

华为云会议

客户端 SDK 参考

文档版本 80.13.6
发布日期 2023-03-24



版权所有 © 华为技术有限公司 2023。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

目录

1 SDK 概述	1
2 Demo 体验	3
2.1 Demo 安装包下载	3
3 SDK 下载	6
3.1 App SDK 下载	6
3.2 APP SDK 包完整性校验	22
4 一分钟跑通 Demo	24
4.1 跑通 Android Demo	24
4.2 跑通 iOS Demo	29
4.3 跑通 Windows Demo	33
4.4 跑通 Mac Demo	36
4.5 跑通 Electron Demo	38
5 Android SDK	46
5.1 概述	46
5.2 修订记录	46
5.3 快速入门	52
5.4 典型场景	66
5.4.1 典型场景接口调用概览	67
5.4.2 场景 1：初始化	69
5.4.3 场景 2：帐号密码方式登录	70
5.4.4 场景 3：App ID 方式登录	71
5.4.5 场景 4：预约会议	72
5.4.6 场景 5：编辑会议	73
5.4.7 场景 6：创建会议	75
5.4.8 场景 7：加入会议	76
5.4.9 场景 8：发起呼叫	77
5.4.10 场景 9：注销	78
5.5 接口参考	79
5.5.1 基础配置	79
5.5.1.1 同步初始化	79
5.5.1.2 异步初始化	80
5.5.1.3 设置接入站点	82

5.5.1.4 自定义会中分享面板.....	82
5.5.1.5 自定义日志保存路径.....	83
5.5.1.6 自定义开关会中聊天功能.....	84
5.5.1.7 自定义是否开启会议高清优先.....	85
5.5.1.8 自定义会议接听时摄像头/麦克风开关.....	85
5.5.1.9 自定义设置来电响铃和震动.....	86
5.5.1.10 自定义隐藏外部标签.....	87
5.5.1.11 禁用提示音.....	87
5.5.1.12 网络模式设置.....	88
5.5.1.13 开启功能.....	89
5.5.1.14 设置语言.....	90
5.5.2 界面定制.....	91
5.5.2.1 自定义底部菜单栏.....	91
5.5.2.2 自定义与会者列表页面中的菜单.....	99
5.5.2.3 自定义会中标题栏点击弹出菜单.....	103
5.5.2.4 自定义会议中加入.....	105
5.5.2.5 自定义会中通知栏.....	107
5.5.2.6 头像拦截显示.....	107
5.5.2.7 自定义视频会议来电时接听方式按钮.....	110
5.5.2.8 自定义会议中水印.....	111
5.5.2.9 自定义状态栏菜单.....	113
5.5.3 帐号操作.....	113
5.5.3.1 帐号密码方式登录.....	113
5.5.3.2 App ID 方式登录.....	115
5.5.3.3 SSO 方式登录.....	117
5.5.3.4 注销.....	118
5.5.3.5 上传个人头像.....	119
5.5.4 会议管理.....	120
5.5.4.1 预约会议.....	120
5.5.4.2 编辑会议.....	126
5.5.4.3 取消会议.....	127
5.5.4.4 获取会议列表.....	128
5.5.4.5 获取会议详情.....	129
5.5.4.6 获取云会议室列表.....	130
5.5.4.7 修改云会议室信息.....	132
5.5.4.8 预约周期会议.....	133
5.5.4.9 编辑周期会议.....	135
5.5.4.10 取消周期会议.....	136
5.5.4.11 编辑周期子会议.....	137
5.5.4.12 取消周期子会议.....	138
5.5.5 会议控制.....	139
5.5.5.1 创建会议.....	139

5.5.5.2 加入会议.....	141
5.5.5.3 添加与会者.....	143
5.5.5.4 离开会议.....	144
5.5.5.5 结束会议.....	145
5.5.5.6 设置共享权限.....	145
5.5.6 获取会议信息.....	146
5.5.6.1 获取与会者列表.....	146
5.5.6.2 获取观众列表.....	147
5.5.6.3 获取用户状态.....	147
5.5.7 音视频数据流.....	148
5.5.7.1 开启第三方视频推流配置.....	148
5.5.7.2 设置第三方推流视频镜像.....	149
5.5.7.3 第三方视频推流.....	150
5.5.8 通知参考.....	151
5.5.8.1 自定义帐号通知事件.....	151
5.5.8.2 自定义会中 UI 行为通知.....	152
5.5.8.3 业务消息通知.....	157
5.5.8.4 自定义来电自动处理通知事件.....	160
5.6 错误码参考.....	161
5.6.1 常见错误码.....	161
5.6.2 错误码表.....	162
5.7 版本更新内容.....	206
5.7.1 80.13.6 更新内容.....	206
5.7.2 80.13.5 更新内容.....	206
5.7.3 80.12.6 更新内容.....	206
5.7.4 80.12.5 更新内容.....	206
5.7.5 80.10.6 更新内容.....	206
5.7.6 80.10.5 更新内容.....	206
5.7.7 80.9.5 更新内容.....	207
5.7.8 80.8.5 更新内容.....	207
5.7.9 80.7.5 更新内容.....	207
5.7.10 80.5.11 更新内容.....	207
5.7.11 80.5.5 更新内容.....	207
5.7.12 70.16.5 更新内容.....	207
5.7.13 70.14.5 更新内容.....	209
5.7.14 70.13.5 更新内容.....	209
5.7.15 70.12.6 更新内容.....	209
5.7.16 70.12.4 更新内容.....	209
5.7.17 70.11.5 更新内容.....	209
5.7.18 70.10.5 更新内容.....	210
5.7.19 70.9.5 更新内容.....	210
5.7.20 70.8.6 更新内容.....	210

5.7.21 70.8.5 更新内容.....	210
5.7.22 70.7.5 更新内容.....	210
5.7.23 70.6.5 更新内容.....	211
5.7.24 70.5.8 更新内容.....	211
5.7.25 70.4.5 更新内容.....	211
5.7.26 70.3.5 更新内容.....	211
5.7.27 60.15.6 更新内容.....	212
5.7.28 60.13.2 更新内容.....	212
5.7.29 60.12.5 更新内容.....	212
5.7.30 60.11.1 更新内容.....	212
5.7.31 60.10.5 更新内容.....	212
5.7.32 60.8.1 更新内容.....	213
5.7.33 60.7.1 更新内容.....	214
6 iOS SDK.....	216
6.1 概述.....	216
6.2 修订记录.....	216
6.3 快速入门.....	221
6.3.1 开发准备.....	221
6.3.2 快速集成.....	221
6.3.3 屏幕共享集成.....	228
6.3.4 业务集成.....	238
6.4 典型场景.....	239
6.4.1 典型场景接口调用概览.....	239
6.4.2 场景 1: 初始化.....	242
6.4.3 场景 2: 登录.....	242
6.4.3.1 帐号密码登录.....	242
6.4.3.2 App ID 登录.....	243
6.4.4 场景 3: 创建会议.....	244
6.4.5 场景 4: 添加与会者.....	245
6.4.6 场景 5: 加入会议.....	245
6.4.7 场景 6: 发起呼叫.....	246
6.4.8 场景 7: 预约会议.....	247
6.4.9 场景 8: 编辑会议.....	248
6.4.10 场景 9: 退出登录.....	248
6.5 接口参考.....	249
6.5.1 基础配置.....	249
6.5.1.1 初始化.....	249
6.5.1.2 设置语言.....	252
6.5.1.3 设置日志路径.....	253
6.5.1.4 设置来电通知方式.....	254
6.5.1.5 自定义会议接听时摄像头/麦克风开关.....	254
6.5.1.6 开启功能.....	255

6.5.1.7 网络模式设置.....	256
6.5.1.8 禁用提示音.....	257
6.5.2 帐号操作.....	258
6.5.2.1 登录.....	258
6.5.2.2 退出登录.....	261
6.5.2.3 上传个人头像.....	262
6.5.3 界面定制.....	262
6.5.3.1 界面配置.....	263
6.5.3.2 清除界面配置.....	270
6.5.3.3 社交分享定制.....	270
6.5.3.4 底部工具栏定制.....	274
6.5.3.5 与会者列表菜单定制.....	278
6.5.3.6 会议页面顶部 titlebar 菜单定制.....	285
6.5.3.7 自定义视频会议来电时接听方式按钮.....	287
6.5.3.8 联系人 UI 定制.....	289
6.5.3.9 头像拦截显示.....	290
6.5.3.10 自定义会中水印.....	292
6.5.3.11 录制按钮操作定制.....	293
6.5.4 会议控制.....	295
6.5.4.1 创建会议.....	295
6.5.4.2 加入会议.....	298
6.5.4.3 添加与会者.....	299
6.5.4.4 离开会议.....	300
6.5.4.5 结束会议.....	301
6.5.4.6 设置共享权限.....	301
6.5.5 会议管理.....	302
6.5.5.1 预约会议.....	302
6.5.5.2 编辑会议.....	306
6.5.5.3 会议列表.....	307
6.5.5.4 会议详情.....	308
6.5.5.5 取消会议.....	310
6.5.5.6 云会议室列表.....	310
6.5.5.7 修改云会议室信息.....	312
6.5.5.8 预约周期会议.....	313
6.5.5.9 编辑周期会议.....	315
6.5.5.10 取消周期会议.....	316
6.5.5.11 编辑周期子会议.....	317
6.5.5.12 取消周期子会议.....	319
6.5.6 获取会议信息.....	319
6.5.6.1 获取与会者列表.....	319
6.5.6.2 获取观众列表.....	320
6.5.6.3 判断自己是否是主持人.....	320

6.5.6.4 获取用户状态.....	321
6.6 通知参考.....	322
6.6.1 注意事项，使用通知前请查看.....	322
6.6.2 帐号状态变更通知.....	322
6.6.2.1 帐号被踢出.....	322
6.6.3 会议信息变更通知.....	323
6.6.3.1 会议状态.....	323
6.6.3.2 被邀加入会议来电通知.....	324
6.6.3.3 会议详情通知.....	325
6.6.3.4 会议结束通知.....	325
6.6.4 通话状态变更通知.....	325
6.6.4.1 通话状态.....	326
6.6.4.2 通话结束.....	326
6.6.5 业务消息通知.....	328
6.6.5.1 主持人角色变化.....	328
6.6.5.2 会议列表变化.....	329
6.6.5.3 视频会议录播权限更新.....	329
6.6.5.4 预约会议发送短信权限更新.....	330
6.6.6 来电接听策略设置.....	330
6.6.6.1 设置点对点来电接听策略.....	330
6.6.6.2 设置会议来电接听策略.....	331
6.7 错误码参考.....	332
6.7.1 常见错误码.....	332
6.7.2 错误码表.....	335
6.8 版本更新内容.....	379
6.8.1 80.13.6 更新内容.....	379
6.8.2 80.13.5 更新内容.....	379
6.8.3 80.12.6 更新内容.....	379
6.8.4 80.12.5 更新内容.....	379
6.8.5 80.10.5 更新内容.....	380
6.8.6 80.9.5 更新内容.....	380
6.8.7 80.8.5 更新内容.....	380
6.8.8 80.7.5 更新内容.....	380
6.8.9 80.5.11 更新内容.....	380
6.8.10 80.5.5 更新内容.....	380
6.8.11 70.16.5 更新内容.....	380
6.8.12 70.14.5 更新内容.....	381
6.8.13 70.13.5 更新内容.....	381
6.8.14 70.12.6 更新内容.....	381
6.8.15 70.12.4 更新内容.....	381
6.8.16 70.11.5 更新内容.....	382
6.8.17 70.10.5 更新内容.....	382

6.8.18 70.9.5 更新内容.....	382
6.8.19 70.8.6 更新内容.....	382
6.8.20 70.8.5 更新内容.....	382
6.8.21 70.7.5 更新内容.....	382
6.8.22 70.6.5 更新内容.....	382
6.8.23 70.5.8 更新内容.....	383
6.8.24 70.4.5 更新内容.....	383
6.8.25 70.3.5 更新内容.....	383
6.8.26 60.15.5 更新内容.....	383
6.8.27 60.13.2 更新内容.....	383
6.8.28 60.12.5 更新内容.....	383
6.8.29 60.11.6 更新内容.....	384
6.8.30 60.10.5 更新内容.....	384
6.8.31 60.8.1 更新内容.....	384
6.8.32 60.7.1 更新内容.....	385
7 Windows SDK.....	386
7.1 概述.....	386
7.2 修订记录.....	386
7.3 快速入门.....	393
7.4 典型场景.....	413
7.4.1 典型场景接口调用概览.....	413
7.4.2 场景 1: 初始化.....	417
7.4.3 场景 2: 登录.....	418
7.4.4 场景 3: 预约会议.....	419
7.4.5 场景 4: 编辑会议.....	421
7.4.6 场景 5: 取消会议.....	423
7.4.7 场景 6: 创建会议.....	424
7.4.8 场景 7: 添加与会人.....	427
7.4.9 场景 8: 加入会议.....	428
7.4.10 场景 9: 邮件链接入会.....	429
7.4.11 场景 10: 发起呼叫.....	430
7.4.12 场景 11: 定制会中“邀请”按钮.....	433
7.4.13 场景 12: 注销登录.....	434
7.4.14 场景 13: 退出.....	435
7.5 接口参考.....	436
7.5.1 基础配置.....	436
7.5.1.1 初始化.....	436
7.5.1.2 退出.....	439
7.5.1.3 设置语言.....	440
7.5.1.4 开启功能.....	442
7.5.1.5 设置用户文件保存路径.....	444
7.5.1.6 网络模式设置.....	445

7.5.1.7 禁用提示音.....	447
7.5.2 帐号操作.....	448
7.5.2.1 登录.....	448
7.5.2.2 SSO 登录.....	450
7.5.2.3 注销.....	452
7.5.2.4 上传个人头像.....	453
7.5.3 界面定制.....	454
7.5.3.1 界面配置.....	454
7.5.3.2 自定义会控按钮.....	466
7.5.3.3 清除界面配置.....	470
7.5.3.4 刷新自定义按钮配置.....	471
7.5.3.5 自定义会中水印.....	472
7.5.3.6 自定义设置界面.....	474
7.5.3.7 显示设置窗口.....	478
7.5.4 会议控制.....	480
7.5.4.1 创建会议.....	480
7.5.4.2 加入会议.....	484
7.5.4.3 Random 入会.....	485
7.5.4.4 设置会场名.....	487
7.5.4.5 添加与会者.....	488
7.5.4.6 开启共享.....	490
7.5.4.7 结束共享.....	492
7.5.4.8 离开会议.....	493
7.5.4.9 结束会议.....	494
7.5.4.10 设置会议来电应答.....	495
7.5.4.11 显示会议主窗口.....	496
7.5.4.12 设置共享策略.....	497
7.5.4.13 设置共享音频.....	498
7.5.4.14 音视频设备控制.....	499
7.5.4.15 设置共享权限.....	500
7.5.5 会议管理.....	501
7.5.5.1 获取 VMR 列表.....	501
7.5.5.2 预约会议.....	503
7.5.5.3 编辑会议.....	508
7.5.5.4 取消会议.....	511
7.5.5.5 获取会议详情.....	512
7.5.5.6 修改云会议室信息.....	514
7.5.5.7 预约周期会议.....	515
7.5.5.8 编辑周期会议.....	517
7.5.5.9 取消周期会议.....	519
7.5.5.10 编辑周期子会议.....	519
7.5.5.11 取消周期子会议.....	521

7.5.6 获取会议信息.....	522
7.5.6.1 获取与会者列表.....	522
7.5.6.2 获取观众列表.....	523
7.5.6.3 获取用户状态.....	523
7.5.7 音视频数据流.....	524
7.5.7.1 获取音频流.....	524
7.6 通知参考.....	526
7.6.1 基础配置变更通知.....	526
7.6.1.1 连接断开.....	526
7.6.1.2 企业配置变更.....	526
7.6.1.3 企业信息变更.....	527
7.6.2 帐号状态变更通知.....	528
7.6.2.1 帐号被踢出.....	528
7.6.3 定制界面事件通知.....	528
7.6.3.1 注入点击.....	528
7.6.3.2 显示会议信息框.....	530
7.6.3.3 销毁会议信息框.....	532
7.6.3.4 查询外部通讯录头像.....	533
7.6.4 会议状态变更通知.....	534
7.6.4.1 会议来电.....	534
7.6.4.2 会议状态.....	535
7.6.4.3 会议信息.....	536
7.6.4.4 会议列表通知.....	538
7.6.4.5 发起数据共享状态.....	540
7.6.4.6 接收数据共享状态.....	541
7.6.4.7 得到远程控制权限.....	542
7.6.4.8 授予他人远程控制权限.....	542
7.6.4.9 共享锁定状态变化通知.....	543
7.6.5 通话状态变更通知.....	543
7.6.5.1 呼叫来电.....	544
7.6.5.2 通话状态.....	544
7.6.5.3 通话信息.....	545
7.6.5.4 通话结束.....	546
7.6.6 音视频设备状态.....	547
7.7 错误码参考.....	547
7.7.1 常见错误码.....	548
7.7.2 错误码表.....	551
7.8 版本更新内容.....	594
7.8.1 80.13.6 更新内容.....	594
7.8.2 80.13.5 更新内容.....	594
7.8.3 80.12.7 更新内容.....	595
7.8.4 80.12.6 更新内容.....	595

7.8.5 80.12.5 更新内容.....	595
7.8.6 80.10.5 更新内容.....	595
7.8.7 80.9.5 更新内容.....	595
7.8.8 80.8.5 更新内容.....	595
7.8.9 80.7.5 更新内容.....	596
7.8.10 80.5.11 更新内容.....	597
7.8.11 80.5.5 更新内容.....	597
7.8.12 70.16.5 更新内容.....	597
7.8.13 70.14.5 更新内容.....	597
7.8.14 70.13.5 更新内容.....	597
7.8.15 70.12.6 更新内容.....	598
7.8.16 70.12.4 更新内容.....	598
7.8.17 70.11.5 更新内容.....	598
7.8.18 70.10.5 更新内容.....	598
7.8.19 70.9.5 更新内容.....	598
7.8.20 70.8.6 更新内容.....	598
7.8.21 70.8.5 更新内容.....	598
7.8.22 70.7.5 更新内容.....	598
7.8.23 70.6.5 更新内容.....	599
7.8.24 70.5.8 更新内容.....	599
7.8.25 70.4.5 更新内容.....	599
7.8.26 70.3.5 更新内容.....	599
7.8.27 60.15.5 更新内容.....	599
7.8.28 60.13.2 更新内容.....	599
7.8.29 60.12.5 更新内容.....	600
7.8.30 60.11.6 更新内容.....	600
7.8.31 60.10.5 更新内容.....	600
7.8.32 60.8.1 更新内容.....	600
8 Mac SDK.....	601
8.1 概述.....	601
8.2 修订记录.....	601
8.3 快速入门.....	605
8.3.1 开发准备.....	605
8.3.2 快速集成.....	605
8.3.3 业务集成.....	609
8.4 典型场景.....	610
8.4.1 典型场景接口调用概览.....	610
8.4.2 场景 1：初始化.....	612
8.4.3 场景 2：登录.....	612
8.4.3.1 帐号密码登录.....	612
8.4.3.2 App ID 登录.....	613
8.4.4 场景 3：创建会议.....	614

8.4.5 场景 4: 添加与会者.....	615
8.4.6 场景 5: 加入会议.....	615
8.4.7 场景 6: 发起呼叫.....	616
8.4.8 场景 7: 预约会议.....	617
8.4.9 场景 8: 编辑会议.....	618
8.4.10 场景 9: 退出登录.....	618
8.5 接口参考.....	619
8.5.1 基础配置.....	619
8.5.1.1 初始化.....	619
8.5.1.2 设置语言.....	621
8.5.1.3 自定义会议接听时摄像头/麦克风开关.....	622
8.5.1.4 自定义本地录制路径.....	623
8.5.1.5 开启功能.....	623
8.5.1.6 网络模式设置.....	627
8.5.1.7 禁用提示音.....	628
8.5.2 帐号操作.....	628
8.5.2.1 登录.....	628
8.5.2.2 退出登录.....	631
8.5.2.3 上传个人头像.....	632
8.5.3 界面定制.....	633
8.5.3.1 界面配置.....	633
8.5.3.2 清除界面配置.....	633
8.5.3.3 底部工具栏定制.....	633
8.5.3.4 共享工具栏定制.....	635
8.5.3.5 底部工具栏更多菜单定制.....	636
8.5.3.6 共享工具栏更多菜单定制.....	637
8.5.3.7 社交分享定制.....	638
8.5.3.8 选择联系人定制.....	639
8.5.3.9 头像拦截显示.....	640
8.5.3.10 其它 UI 定制.....	642
8.5.3.11 自定义会中水印.....	642
8.5.3.12 自定义设置界面.....	643
8.5.3.13 联系人信息定制.....	646
8.5.3.14 录制按钮操作定制.....	648
8.5.3.15 显示设置窗口.....	649
8.5.3.16 与会者列表子项定制.....	650
8.5.4 会议控制.....	651
8.5.4.1 创建会议.....	652
8.5.4.2 加入会议.....	654
8.5.4.3 添加与会者.....	656
8.5.4.4 离开会议.....	656
8.5.4.5 结束会议.....	657

8.5.4.6 显示会议主窗口.....	657
8.5.4.7 设置共享权限.....	658
8.5.5 会议管理.....	658
8.5.5.1 预约会议.....	659
8.5.5.2 编辑会议.....	662
8.5.5.3 会议列表.....	663
8.5.5.4 会议详情.....	664
8.5.5.5 取消会议.....	666
8.5.5.6 云会议室列表.....	666
8.5.5.7 修改云会议室信息.....	668
8.5.5.8 预约周期会议.....	669
8.5.5.9 编辑周期会议.....	671
8.5.5.10 取消周期会议.....	672
8.5.5.11 编辑周期子会议.....	673
8.5.5.12 取消周期子会议.....	675
8.5.6 获取会议信息.....	675
8.5.6.1 获取与会者列表.....	676
8.5.6.2 获取观众列表.....	676
8.5.6.3 获取用户状态.....	677
8.5.6.4 判断自己是否是主持人.....	677
8.5.7 音视频数据流.....	678
8.5.7.1 获取音频流.....	678
8.6 通知参考.....	678
8.6.1 注意事项，使用通知前请查看.....	679
8.6.2 帐号状态变更通知.....	679
8.6.2.1 帐号被踢出.....	679
8.6.3 企业配置信息通知.....	679
8.6.3.1 企业配置信息.....	680
8.6.4 会议来电设置设备状态通知.....	680
8.6.4.1 设置麦克风状态通知.....	681
8.6.4.2 设置摄像头状态通知.....	681
8.6.5 联系人交互通知.....	681
8.6.5.1 打开联系人选择窗口通知.....	681
8.6.5.2 关闭联系人选择窗口通知.....	683
8.6.6 会议交互通知.....	683
8.6.6.1 dock 栏图标及共享时吸附图片通知.....	683
8.6.7 用户头像通知.....	683
8.6.7.1 获取用户图像通知.....	683
8.6.8 社交分享通知.....	684
8.6.8.1 会议信息分享通知.....	684
8.6.8.2 自定义会议信息分享通知.....	686
8.6.9 业务消息通知.....	686

8.6.9.1 个人角色变化通知.....	686
8.6.9.2 会议状态变更通知.....	687
8.6.9.3 全量会议列表通知.....	688
8.6.9.4 会中会议状态信息变更通知.....	690
8.6.9.5 音频数据流.....	690
8.6.10 来电接听策略设置.....	691
8.6.10.1 设置点对点来电接听策略.....	692
8.6.10.2 设置会议来电接听策略.....	692
8.6.11 会议来电状态变更通知.....	692
8.6.12 通话状态变更通知.....	694
8.6.12.1 呼叫来电.....	694
8.6.12.2 通话状态.....	694
8.6.12.3 通话结束.....	695
8.7 错误码参考.....	697
8.7.1 常见错误码.....	697
8.7.2 错误码表.....	698
8.8 版本更新内容.....	742
8.8.1 80.13.6 更新内容.....	742
8.8.2 80.13.5 更新内容.....	742
8.8.3 80.12.7 更新内容.....	742
8.8.4 80.12.6 更新内容.....	742
8.8.5 80.12.5 更新内容.....	742
8.8.6 80.10.5 更新内容.....	742
8.8.7 80.9.5 更新内容.....	743
8.8.8 80.8.5 更新内容.....	743
8.8.9 80.7.5 更新内容.....	743
8.8.10 80.5.11 更新内容.....	743
8.8.11 80.5.5 更新内容.....	743
8.8.12 70.16.5 更新内容.....	743
8.8.13 70.14.5 更新内容.....	744
8.8.14 70.13.5 更新内容.....	744
8.8.15 70.12.6 更新内容.....	744
8.8.16 70.12.4 更新内容.....	744
8.8.17 70.11.5 更新内容.....	745
8.8.18 70.10.5 更新内容.....	745
8.8.19 70.9.5 更新内容.....	745
8.8.20 70.8.6 更新内容.....	745
8.8.21 70.8.5 更新内容.....	745
8.8.22 70.7.5 更新内容.....	745
8.8.23 70.6.5 更新内容.....	746
8.8.24 70.5.8 更新内容.....	746
8.8.25 70.4.5 更新内容.....	746

8.8.26 70.3.5 更新内容.....	746
8.8.27 60.15.5 更新内容.....	746
8.8.28 60.13.2 更新内容.....	746
8.8.29 60.12.5 更新内容.....	747
9 Electron SDK.....	748
9.1 概述.....	748
9.2 修订记录.....	748
9.3 快速入门.....	750
9.4 典型场景.....	754
9.4.1 典型场景接口调用概览.....	754
9.4.2 场景 1: 初始化.....	757
9.4.3 场景 2: 登录.....	758
9.4.4 场景 3: 预约会议.....	759
9.4.5 场景 4: 编辑会议.....	760
9.4.6 场景 5: 取消会议.....	762
9.4.7 场景 6: 创建会议.....	763
9.4.8 场景 7: 添加与会人.....	765
9.4.9 场景 8: 加入会议.....	766
9.4.10 场景 9: 邮件链接入会.....	767
9.4.11 场景 10: 发起呼叫.....	768
9.4.12 场景 11: 定制会中“邀请”按钮.....	770
9.4.13 场景 12: 注销登录.....	771
9.4.14 场景 13: 退出.....	772
9.5 接口参考.....	773
9.5.1 UISDK 接口管理对象.....	773
9.5.1.1 初始化.....	773
9.5.1.2 退出.....	775
9.5.1.3 获取登录服务对象.....	775
9.5.1.4 获取会议控制服务对象.....	775
9.5.1.5 获取会议管理服务对象.....	775
9.5.1.6 获取配置服务对象.....	776
9.5.1.7 获取界面配置服务对象.....	776
9.5.1.8 获取呼叫服务对象.....	776
9.5.1.9 订阅连接断开通知.....	777
9.5.2 登录管理对象.....	777
9.5.2.1 订阅登录被踢通知.....	777
9.5.2.2 订阅企业配置通知.....	778
9.5.2.3 订阅登录错误通知.....	779
9.5.2.4 账号密码登录.....	780
9.5.2.5 Appld 登录.....	781
9.5.2.6 SSO 登录.....	783
9.5.2.7 注销.....	784

9.5.2.8 设置网络模式.....	785
9.5.2.9 上传头像.....	787
9.5.3 会议控制操作对象.....	788
9.5.3.1 订阅会议状态通知.....	788
9.5.3.2 订阅会议结束通知.....	789
9.5.3.3 订阅会议信息通知.....	791
9.5.3.4 订阅共享状态通知.....	793
9.5.3.5 订阅个人角色变化通知.....	795
9.5.3.6 创建会议.....	796
9.5.3.7 加入会议.....	801
9.5.3.8 Random 入会.....	803
9.5.3.9 添加与会者.....	804
9.5.3.10 离开会议.....	807
9.5.3.11 结束会议.....	808
9.5.3.12 获取与会者列表.....	808
9.5.3.13 获取观众列表.....	812
9.5.3.14 获取用户状态.....	815
9.5.3.15 获取自己会中角色.....	816
9.5.3.16 设置会场名.....	818
9.5.4 会议管理对象.....	819
9.5.4.1 订阅会议列表通知.....	819
9.5.4.2 预定会议.....	827
9.5.4.3 取消会议.....	839
9.5.4.4 编辑会议.....	840
9.5.4.5 获取 VMR 列表.....	848
9.5.4.6 获取会议详情.....	850
9.5.4.7 修改云会议室信息.....	861
9.5.4.8 获取会议列表.....	862
9.5.4.9 预约周期会议.....	870
9.5.4.10 编辑周期会议.....	883
9.5.4.11 编辑周期子会议.....	892
9.5.4.12 取消周期会议.....	895
9.5.4.13 取消周期子会议.....	896
9.5.5 通用配置管理对象.....	897
9.5.5.1 功能配置开关.....	897
9.5.5.2 设置语言.....	899
9.5.5.3 设置用户保存路径.....	900
9.5.5.4 设置来电应答方式.....	901
9.5.5.5 会议来电时设备开关配置.....	902
9.5.5.6 获取会议 pstn 接入号码.....	903
9.5.5.7 禁用提示音.....	904
9.5.5.8 设置共享权限.....	905

9.5.6 界面配置管理对象.....	906
9.5.6.1 订阅注入点击通知.....	907
9.5.6.2 订阅与会者列表子项注入点击通知.....	908
9.5.6.3 订阅窗口句柄通知.....	910
9.5.6.4 订阅查询外部通讯录头像通知.....	911
9.5.6.5 界面配置.....	912
9.5.6.6 自定义会中水印.....	913
9.5.6.7 获取窗口句柄列表.....	914
9.5.6.8 显示会议窗口.....	915
9.5.6.9 设置头像.....	916
9.5.6.10 自定义会议窗口任务栏图标.....	917
9.5.6.11 显示设置窗口接口.....	918
9.5.7 呼叫管理对象.....	920
9.5.7.1 订阅通话状态通知.....	920
9.5.7.2 订阅通话结束通知.....	921
9.5.7.3 发起呼叫.....	923
9.6 错误码.....	925
9.7 其他说明.....	984
9.7.1 界面配置 json 说明.....	985
9.7.2 自定义会控按钮.....	992
9.7.3 常见错误码.....	996
9.8 版本更新内容.....	1000
9.8.1 80.13.6 更新内容.....	1000
9.8.2 80.13.5 更新内容.....	1000
9.8.3 80.12.7 更新内容.....	1000
9.8.4 80.12.6 更新内容.....	1000
9.8.5 80.12.5 更新内容.....	1000
9.8.6 80.10.5 更新内容.....	1001
9.8.7 80.9.5 更新内容.....	1001
9.8.8 80.8.5 更新内容.....	1001
9.8.9 80.7.5 更新内容.....	1001
9.8.10 80.5.11 更新内容.....	1002
9.8.11 80.5.5 更新内容.....	1002
10 SDK 已知安全漏洞修复说明.....	1003
11 常见问题.....	1004
11.1 华为云会议资源类型.....	1004
11.2 如何获取 SDK 的日志.....	1004
11.3 【iOS】如何用#import"xxx.h"导入 iOS SDK 中的文件.....	1009
11.4 【iOS】实现通知方法不生效.....	1009
11.5 【iOS】iOS 导入 AFNetworking 报错.....	1009
11.6 【Android】支持 64 位库.....	1010
11.7 【Android】如何去除不需要的 aar 依赖.....	1012

11.8 【 Android 】使用第三方加固软件后，路由跳转错误.....	1012
11.9 【 Android 】 SDK 提示语不支持中英文切换.....	1013
11.10 【 Android 】 SDK 初始化失败，调用创会、加入会议等接口没有回调也没有提示.....	1013
11.11 【 Win 】第三方客户端调用 Exit 接口，需要等待一段时间后才能退出.....	1014
11.12 【 Electron 】运行 Electron demo 时遇 electron failed to install correctly 的错误提示.....	1015
11.13 【 Electron 】 mac 平台环境安装时遇 permission denied 的错误提示.....	1015
11.14 【 Electron 】 mac 平台编译.node 文件时遇 no template named 'remove_cv_t'的错误提示.....	1016
11.15 【 Electron 】 windows 平台编译.node 文件时遇 Could not find any Python installation to use 的错误提示.....	1016
11.16 【 Electron 】 windows 平台编译.node 文件时遇 Could not find any Visual Studio installation to use 的错误提示.....	1017
12 专有名词解释.....	1018
12.1 帐号相关名词.....	1018
12.2 会议 ID 相关名词.....	1018

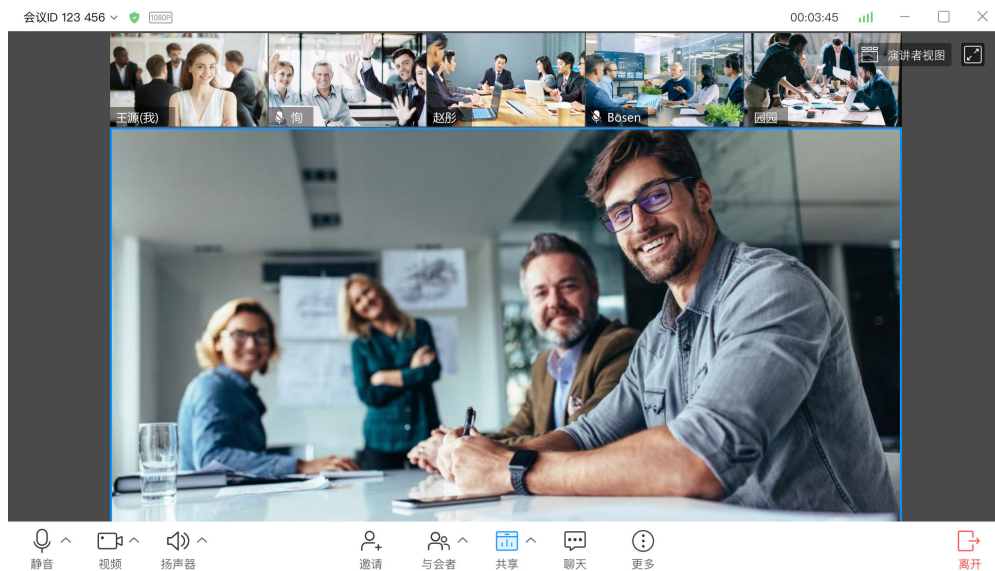
1 SDK 概述

华为云会议服务为开发者提供Windows、Android、iOS、Mac、Electron的客户端 SDK，方便开发者将华为云会议的音频会议、视频会议、桌面共享能力快速集成到自己的应用中。

华为云会议的客户端SDK已经实现了会议中UI界面，包括会场画面显示，会议控制功能，桌面共享等，开发者只需要调用少量接口即可完成华为云会议能力的集成。客户端SDK已经提供的主要UI如下（以Windows SDK为例，其他平台的SDK能力相同，只是界面布局有差异）。

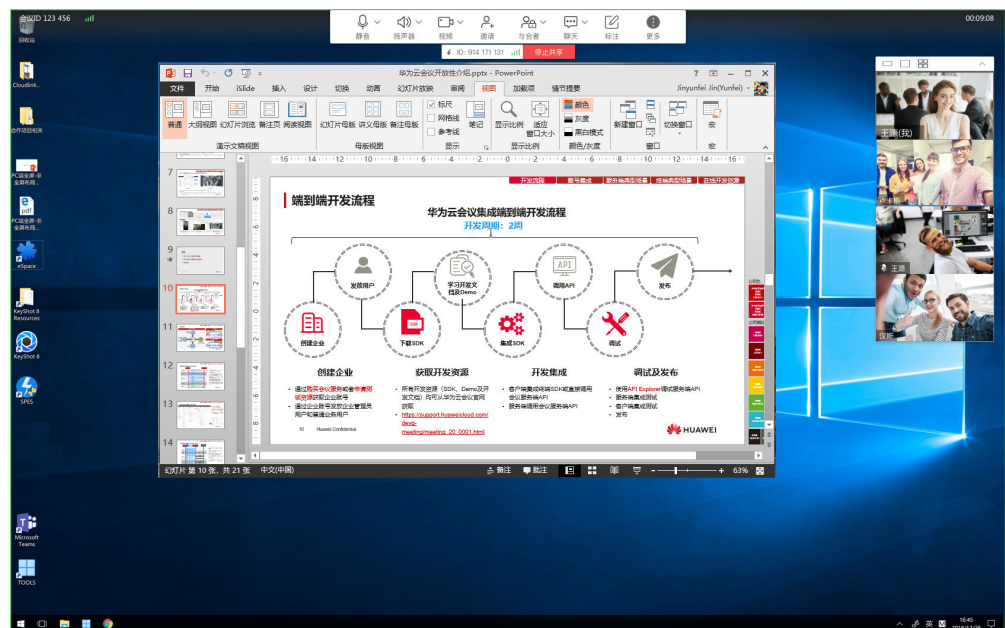
1. 会议中的主界面

图 1-1 会中主界面



2. 桌面共享界面

图 1-2 桌面共享界面



3. 会议来电界面

图 1-3 会议来电界面



说明

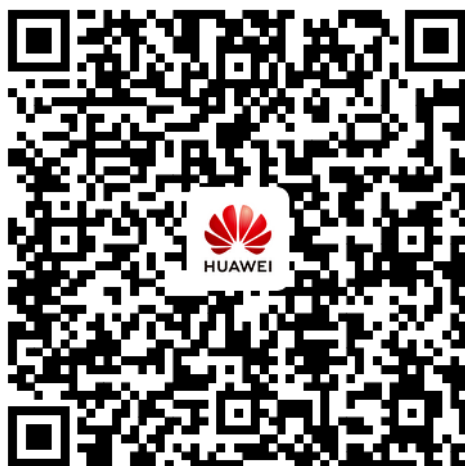
- 来电通知界面只有在应用进程没有被系统杀掉的情况下才能收到来电通知，移动端SDK并不提供VoIP Push能力。

2 Demo 体验

2.1 Demo 安装包下载

1. Android Demo安装包下载

图 2-1 Android Demo 安装包下载二维码



SHA256:
055d00bd65e913d18a2dfe1e1ccb2a884cc6d2f1f1543e2a9c540c207f6fe1d0

说明

使用手机浏览器扫一扫下载

2. Windows Demo安装包下载

图 2-2 Windows Demo 安装包下载链接



32位Windows Demo下载链接: [hmwsdk-win-demo-win32.zip](#)

SHA256:

6b7e99bcc2eecd1739667d2e44935c8812e084bd2dbc7d25e9c408ede49993

64位Windows Demo下载链接: [hmwsdk-win-demo-x64.zip](#)

SHA256:

5c0f4827e9cd14c3a4cdd656d74d936a420d06429068272fb84c7c1b1d4a1ef8

3. Mac Demo安装包下载

图 2-3 Mac Demo 安装包下载链接



Mac Demo下载链接: [hwmsdk-mac-demo.dmg](#)

SHA256:

c72bddb49f3434d616af6984e98fc9662f3180164f5fd4508d43aac328d23e61

4. Electron Demo安装包下载

图 2-4 Electron Demo 安装包下载链接



32位Windows Electron Demo下载链接: [hwmsdk-electron-demo-win-ia32.zip](#)

SHA256:

911790284bde1f0a7fa35ec9e4ebceb60bb5a7f482bb66818642c42b13a3c6f2

64位Windows Electron Demo下载链接: [hwmsdk-electron-demo-win-x64.zip](#)

SHA256:

7d155c57feff96112f7f7b5941ee257acdfb55355cfe4e8ebac108f62ad1be30

Mac Electron Demo下载链接: [hwmsdk-electron-demo-mac.dmg](#)

SHA256:

051db82a5a0f2c76b3dbec115da6d3ac0b76c72a8ca53cc2136a219a06eb5774

3 SDK 下载

3.1 App SDK 下载

说明

华为云会议的UI SDK提供一年的维护周期，如果超过一年的版本使用中有问题，请升级到最新版本。

SDK下载后可参考[3.3章节](#)进行包完整性校验。

表 3-1 Android/iOS/Windows/Mac/Electron UI SDK 资源下载路径

版本	发布日期	资源	下载链接	安装包增量	说明
80.13.6 (最新)	2023-02-22	Android UI SDK	hwmsdk-android-80.13.6.zip	32位: 33.6M 64位: 34.4M 32位+64位: 56.8M	80.13.6补丁版本
		iOS UI SDK	hwmsdk-ios-80.13.6.zip	27.4M	
		Windows UI SDK	hwmsdk-win-80.13.6.zip	约81.1M	
		Mac UI SDK	hwmsdk-mac-80.13.6.zip	约57.6M	
		Electron UI SDK	hwmsdk-electron-80.13.6.zip	Win平台: 约81.1M Mac平台: 约57.6M	

版本	发布日期	资源	下载链接	安装包增量	说明
80.13.5 (维护)	2023-02-16	Android UI SDK	hwmsdk-android-80.13.5.zip	32位: 33.6M 64位: 34.4M 32位+64位: 56.8M	-
		iOS UI SDK	hwmsdk-ios-80.13.5.zip	27.8M	-
		Windows UI SDK	hwmsdk-win-80.13.5.zip	约81.1M	-
		Mac UI SDK	hwmsdk-mac-80.13.5.zip	约57.6M	-
		Electron UI SDK	hwmsdk-electron-80.13.5.zip	Win平台: 约81.1M Mac平台: 约57.6M	-
80.12.7 (维护)	2023-01-16	Windows UI SDK	hwmsdk-win-80.12.7.zip	约81.1M	80.12.5补丁版本
		Mac UI SDK	hwmsdk-mac-80.12.7.zip	约51.3M	
		Electron UI SDK	hwmsdk-electron-80.12.7.zip	Win平台: 约81.1M Mac平台: 约51.3M	
80.12.6 (维护)	2023-01-09	Android UI SDK	hwmsdk-android-80.12.6.zip	32位: 32.6M 64位: 33.2M 32位+64位: 54.6M	80.12.5补丁版本
		iOS UI SDK	hwmsdk-ios-80.12.6.zip	28.1M	
		Windows UI SDK	hwmsdk-win-80.12.6.zip	约81.1M	
		Mac UI SDK	hwmsdk-mac-80.12.6.zip	约51.3M	

版本	发布日期	资源	下载链接	安装包增量	说明
		Electron UI SDK	hwmsdk-electron-80.12.6.zip	Win平台: 约81.1M Mac平台: 约51.3M	
80.12.5 (维护)	2022-12-26	Android UI SDK	hwmsdk-android-80.12.5.zip	32位: 32.6M 64位: 33.2M 32位+64位: 54.6M	-
		iOS UI SDK	hwmsdk-ios-80.12.5.zip	28.1M	-
		Windows UI SDK	hwmsdk-win-80.12.5.zip	约81.1M	-
		Mac UI SDK	hwmsdk-mac-80.12.5.zip	约51.3M	-
		Electron UI SDK	hwmsdk-electron-80.12.5.zip	Win平台: 约81.1M Mac平台: 约51.3M	-
80.10.6 (维护)	2022-11-23	Android UI SDK	hwmsdk-android-80.10.6.zip	32位: 29.7M 64位: 30.3M 32位+64位: 48.8M	80.10.5补丁版本
80.10.5 (维护)	2022-11-16	Android UI SDK	hwmsdk-android-80.10.5.zip	32位: 29.7M 64位: 30.3M 32位+64位: 48.8M	-
		iOS UI SDK	hwmsdk-ios-80.10.5.zip	27.5M	-
		Windows UI SDK	hwmsdk-win-80.10.5.zip	约81.1M	-
		Mac UI SDK	hwmsdk-mac-80.10.5.zip	约48.5M	-

版本	发布日期	资源	下载链接	安装包增量	说明
		Electron UI SDK	hwmsdk-electron-80.10.5.zip	Win平台： 约81.1M Mac平台： 约48.5M	-
80.9.5 (维护)	2022-10-15	Android UI SDK	hwmsdk-android-80.9.5.zip	32位： 29.0M 64位： 29.5M 32位+64位： 48.0M	-
		iOS UI SDK	hwmsdk-ios-80.9.5.zip	33.9M	-
		Windows UI SDK	hwmsdk-win-80.9.5.zip	约80.7M	-
		Mac UI SDK	hwmsdk-mac-80.9.5.zip	约48.4M	-
		Electron UI SDK	hwmsdk-electron-80.9.5.zip	Win平台： 约80.7M Mac平台： 约48.4M	-
80.8.5 (维护)	2022-09-22	Android UI SDK	hwmsdk-android-80.8.5.zip	32位： 28.8M 64位： 29.2M 32位+64位： 47.4M	-
		iOS UI SDK	hwmsdk-ios-80.8.5.zip	33.9M	-
		Windows UI SDK	hwmsdk-win-80.8.5.zip	约71.8M	-
		Mac UI SDK	hwmsdk-mac-80.8.5.zip	约48.6M	-
		Electron UI SDK	hwmsdk-electron-80.8.5.zip	Win平台： 约71.8M Mac平台： 约48.6M	-

版本	发布日期	资源	下载链接	安装包增量	说明
80.7.6 (维护)	2022-08-18	Android UI SDK	hwmsdk-android-80.7.6.zip	32位: 28.0M 64位: 28.7M 32位+64位: 46.5M	80.7.5补丁版本
		iOS UI SDK	hwmsdk-ios-80.7.6.zip	33.1M	
80.7.5 (维护)	2022-07-21	Android UI SDK	hwmsdk-android-80.7.5.zip	32位: 28.0M 64位: 28.7M 32位+64位: 46.5M	-
		iOS UI SDK	hwmsdk-ios-80.7.5.zip	33.1M	-
		Windows UI SDK	hwmsdk-win-80.7.5.zip	约70.5M	-
		Mac UI SDK	hwmsdk-mac-80.7.5.zip	约49M	-
		Electron UI SDK	hwmsdk-electron-80.7.5.zip	Win平台: 约70.5M Mac平台: 约49M	-
80.5.11 (维护)	2022-06-06	Android UI SDK	hwmsdk-android-80.5.11.zip	32位: 27.6M 64位: 28.3M 32位+64位: 46.3M	-
		iOS UI SDK	hwmsdk-ios-80.5.11.zip	33.1M	-
		Windows UI SDK	hwmsdk-win-80.5.11.zip	约69.4M	-
		Mac UI SDK	hwmsdk-mac-80.5.11.zip	约49M	-

版本	发布日期	资源	下载链接	安装包增量	说明
		Electron UI SDK	hwmsdk-electron-80.5.11.zip	Win平台: 约69.4M Mac平台: 约49M	-
80.5.5 (维护)	2022-05-20	Android UI SDK	hwmsdk-android-80.5.5.zip	32位: 27.6M 64位: 28.3M 32位+64位: 46.3M	-
		iOS UI SDK	hwmsdk-ios-80.5.5.zip	33.1M	-
		Windows UI SDK	hwmsdk-win-80.5.5.zip	约69.4M	-
		Mac UI SDK	hwmsdk-mac-80.5.5.zip	约49M	-
		Electron UI SDK	hwmsdk-electron-80.5.5.zip	Win平台: 约69.4M Mac平台: 约49M	-
70.16.5 (维护)	2022-04-15	Android UI SDK	hwmsdk-android-70.16.5.zip	32位: 27.6M 64位: 28.2M 32位+64位: 46.1M	-
		iOS UI SDK	hwmsdk-ios-70.16.5.zip	33.1M	-
		Windows UI SDK	hwmsdk-win-70.16.5.zip	约69.4M	-
		Mac UI SDK	hwmsdk-mac-70.16.5.zip	约49M	-
		Electron UI SDK	hwmsdk-electron-70.16.5.zip	-	Electron UI SDK 的第一个版本

版本	发布日期	资源	下载链接	安装包增量	说明
70.14.5 (维护)	2022-02-16	Android UI SDK	hwmsdk-android-70.14.5.zip	32位: 27.5M 64位: 27.9M 32位+64位: 45.8M	-
		iOS UI SDK	hwmsdk-ios-70.14.5.zip	33.4M	-
		Windows UI SDK	hwmsdk-win-70.14.5.zip	约60.7M	-
		Windows Electron Demo	hwm-electron-demo-src-70.14.5.zip	-	-
		Mac UI SDK	hwmsdk-mac-70.14.5.zip	47.5M	-
		Mac Electron Demo	hwmsdk-electron-mac-70.14.5.zip	-	-
70.13.5 (维护)	2022-01-18	Android UI SDK	hwmsdk-android-70.13.5.zip	32位: 27.1M 64位: 27.5M 32位+64位: 45.4M	△下载前必看, 【重要说明】 “登录”和“退出登录”的回调函数新增了参数
		iOS UI SDK	hwmsdk-ios-70.13.5.zip	33.0M	△下载前必看, 【重要说明】 “登录”和“退出登录”的回调函数新增了参数
		Windows UI SDK	hwmsdk-win-70.13.5.zip	约60.7M	△下载前必看, 【重要说明】 “登录”和“退出登录”的回调函数新增了参数
		Windows Electron Demo	hwm-electron-demo-src-70.13.5.zip	-	△下载前必看, 【重要说明】 “登录”和“退出登录”的回调函数新增了参数

版本	发布日期	资源	下载链接	安装包增量	说明
		Mac UI SDK	hwmsdk-mac-70.13.5.zip	45.4M	△下载前必看， 【重要说明】 “登录”和“退出登录”的回调函数新增了参数 由于App Store限制，本地录制默认路径变更，接入方App需要申请下载文件夹权限
		Mac Electron Demo	hwmsdk-electron-mac-70.13.5.zip	-	△下载前必看， 【重要说明】 “登录”和“退出登录”的回调函数新增了参数 由于App Store限制，本地录制默认路径变更，接入方App需要申请下载文件夹权限
70.12.6 (停止维护)	2021-12-23	Android UI SDK	hwmsdk-android-70.12.6.zip	32位: 28.1M 64位: 28.3M 32位+64位: 47.3M	-
		iOS UI SDK	hwmsdk-ios-70.12.6.zip	33.7M	-
		Windows UI SDK	hwmsdk-win-70.12.6.zip	约59.3M	-
		Windows Electron Demo	hwm-electron-demo-src-70.12.6.zip	-	-
		Mac UI SDK	hwmsdk-mac-70.12.6.zip	46.8M	-
		Mac Electron Demo	hwmsdk-electron-mac-70.12.6.zip	-	-

版本	发布日期	资源	下载链接	安装包增量	说明
70.12.4 (停止维护)	2021-12-14	Android UI SDK	hwmsdk-android-70.12.4.zip	32位: 28.1M 64位: 28.3M 32位+64位: 47.3M	-
		iOS UI SDK	hwmsdk-ios-70.12.4.zip	33.7M	-
		Windows UI SDK	hwmsdk-win-70.12.4.zip	约59.3M	-
		Windows Electron Demo	hwm-electron-demo-src-70.12.4.zip	-	-
		Mac UI SDK	hwmsdk-mac-70.12.4.zip	46.8M	-
		Mac Electron Demo	hwmsdk-electron-mac-70.12.4.zip	-	-
70.11.5 (停止维护)	2021-11-26	Android UI SDK	hwmsdk-android-70.11.5.zip	32位: 28.1M 64位: 28.3M 32位+64位: 47.3M	-
		iOS UI SDK	hwmsdk-ios-70.11.5.zip	34.3M	△下载前必看, 【重要说明】 △ HWRTCEngine.framework更名为 HWMMREngine.framework(只 修改包名, 集成 方式不变)
		Windows UI SDK	hwmsdk-win-70.11.5.zip	约60M	-
		Windows Electron Demo	hwm-electron-demo-src-70.11.5.zip	-	-
		Mac UI SDK	hwmsdk-mac-70.11.5.zip	46.7M	-

版本	发布日期	资源	下载链接	安装包增量	说明
		Mac Electron Demo	hwmsdk-electron-mac-70.11.5.zip	-	-
70.10.5 (停止维护)	2021-10-22	Android UI SDK	hwmsdk-android-70.10.5.zip	32位: 28.4M 64位: 28.6M 32位+64位: 47.5M	-
		iOS UI SDK	hwmsdk-ios-70.10.5.zip	34.4M	-
		Windows UI SDK	hwmsdk-win-70.10.5.zip	约60M	-
		Windows Electron Demo	hwm-electron-demo-src-70.10.5.zip	-	-
		Mac UI SDK	hwmsdk-mac-70.10.5.zip	49.6M	-
		Mac Electron Demo	hwmsdk-electron-mac-70.10.5.zip	-	-
70.9.5 (停止维护)	2021-9-15	Android UI SDK	hwmsdk-android-70.9.5.zip	32位: 28.3M 64位: 28.4M 32位+64位: 47.3M	-
		iOS UI SDK	hwmsdk-ios-70.9.5.zip	35.8M	-
		Windows UI SDK	hwmsdk-win-70.9.5.zip	约60M	-
		Windows Electron Demo	hwm-electron-demo-src-70.9.5.zip	-	-
		Mac UI SDK	hwmsdk-mac-70.9.5.zip	49.8M	-
		Mac Electron Demo	hwmsdk-electron-mac-70.9.5.zip	-	-

版本	发布日期	资源	下载链接	安装包增量	说明
70.8.6 (停止维护)	2021-8-28	Android UI SDK	hwmsdk-android-70.8.6.zip	32位: 27.9M 64位: 28.1M 32位+64位: 46.7M	-
		iOS UI SDK	hwmsdk-ios-70.8.6.zip	35.2M	-
		Windows UI SDK	hwmsdk-win-70.8.6.zip	约60M	-
		Windows Electron Demo	hwm-electron-demo-src-70.8.6.zip	-	-
		Mac UI SDK	hwmsdk-mac-70.8.6.zip	约45M	-
		Mac Electron Demo	hwmsdk-electron-mac-70.8.6.zip	-	-
70.8.5 (停止维护)	2021-08-18	Android UI SDK	hwmsdk-android-70.8.5.zip	32位: 27.9M 64位: 28.1M 32位+64位: 46.7M	△下载前必看, 【重要说明】 △从70.8.5版本开始仅支持AndroidX, 不再支持Android support库
		iOS UI SDK	hwmsdk-ios-70.8.5.zip	35.2M	-
		Windows UI SDK	hwmsdk-win-70.8.5.zip	约60M	-
		Windows Electron Demo	hwm-electron-demo-src-70.8.5.zip	-	-
		Mac UI SDK	hwmsdk-mac-70.8.5.zip	约45M	-
		Mac Electron Demo	hwmsdk-electron-mac-70.8.5.zip	-	-

版本	发布日期	资源	下载链接	安装包增量	说明
70.7.5 (停止维护)	2021-07-16	Android UI SDK	hwmsdk-android-70.7.5.zip	32位: 28.1M 64位: 28.3M 32位+64位: 46.9M	-
		iOS UI SDK	hwmsdk-ios-70.7.5.zip	34.5M	-
		Windows UI SDK	hwmsdk-win-70.7.5.zip	约66M	-
		Windows Electron Demo	hwm-electron-demo-src-70.7.5.zip	-	-
		Mac UI SDK	hwmsdk-mac-70.7.5.zip	约47M	-
		Mac Electron Demo	hwmsdk-electron-mac-70.7.5.zip	-	-
70.6.5 (停止维护)	2021-06-21	Android UI SDK	hwmsdk-android-70.6.5.zip	32位: 28.3M 64位: 28.5M 32位+64位: 47.1M	-
		iOS UI SDK	hwmsdk-ios-70.6.5.zip	34.5M	-
		Windows UI SDK	hwmsdk-win-70.6.5.zip	约46M	-
		Windows Electron Demo	hwm-electron-demo-src-70.6.5.zip	-	-
		Mac UI SDK	hwmsdk-mac-70.6.5.zip	约46M	-
		Mac Electron Demo	hwmsdk-electron-mac-70.6.5.zip	-	-

版本	发布日期	资源	下载链接	安装包增量	说明
70.5.8 (停止维护)	2021-05-17	Android UI SDK	hwmsdk-android-70.5.8.zip	32位: 28.2M 64位: 28.3M 32位+64位: 46.8M	-
		iOS UI SDK	hwmsdk-ios-70.5.8.zip	34.3M	-
		Windows UI SDK	hwmsdk-win-70.5.8.zip	约46M	-
		Windows Electron Demo	hwm-electron-demo-src-70.5.8.zip	-	-
		Mac UI SDK	hwmsdk-mac-70.5.8.zip	约56M	△下载前必看, 【重要说明】
		Mac Electron Demo	hwmsdk-electron-mac-70.5.8.zip	-	-
70.4.5 (停止维护)	2021-04-15	Android UI SDK	hwmsdk-android-70.4.5.zip	32位: 26.8M 64位: 26.9M 32位+64位: 45.2M	-
		iOS UI SDK	hwmsdk-ios-70.4.5.zip	33.1M	-
		Windows UI SDK	hwmsdk-win-70.4.5.zip	约46M	-
		Windows Electron Demo	hwm-electron-demo-src-70.4.5.zip	-	-
		Mac UI SDK	hwmsdk-mac-70.4.5.zip	约49M	△下载前必看, 【重要说明】
		Mac Electron Demo	hwmsdk-electron-mac-70.4.5.zip	-	-

版本	发布日期	资源	下载链接	安装包增量	说明
70.3.5 (停止维护)	2021-03-12	Android UI SDK	hwmsdk-android-70.3.5.zip	32位: 33M 64位: 34.3M 32位 & 64位: 58.6M	-
		iOS UI SDK	hwmsdk-ios-70.3.5.zip	33.1M	-
		Windows UI SDK	hwmsdk-win-70.3.5.zip	约46M	-
		Windows Electron Demo	hwm-electron-demo-src-70.3.5.zip	-	-
		Mac UI SDK	hwmsdk-mac-70.3.5.zip	约47M	-
		Mac Electron Demo	hwmsdk-electron-mac-70.3.5.zip	-	-
60.15.6 (停止维护)	2021-02-22	Android UI SDK	hwmsdk-android-60.15.6.zip	32位: 33.42M 64位: 34.76M 32位 & 64位: 59M	-
		iOS UI SDK	hwmsdk-ios-60.15.5.zip	35.2M	-
		Windows UI SDK	hwmsdk-win-60.15.5.zip	约46M	-
		Windows Electron Demo	hwm-electron-demo-src-60.15.5.zip	-	-
		Mac UI SDK	hwmsdk-mac-60.15.5.zip	约43M	-
		Mac Electron Demo	hwmsdk-electron-mac-60.15.5.zip	-	-

版本	发布日期	资源	下载链接	安装包增量	说明
60.13.2 (停止维护)	2021-01-08	Android UI SDK	hwmsdk-android-60.13.2.zip	32位: 31.7M 64位: 32.5M 32位 & 64位: 54.4M	-
		iOS UI SDK	hwmsdk-ios-60.13.2.zip	38M	-
		Windows UI SDK	hwmsdk-win-60.13.2.zip	-	△下载前必看, 【重要说明】
		Windows Electron Demo	hwm-electron-demo-src-60.13.2.zip	-	-
		Mac UI SDK	hwmsdk-mac-60.13.2.zip	-	-
		Mac Electron Demo	hwmsdk-electron-mac-60.13.2.zip	-	-
60.12.5 (停止维护)	2020-12-07	Android UI SDK	hwmsdk-android-60.12.5.zip	-	-
		iOS UI SDK	hwmsdk-ios-60.12.5.zip	-	△下载前必看, 【重要说明】
		Windows UI SDK	hwmsdk-win-60.12.5.zip	-	-
		Windows Electron Demo	hwm-electron-demo-src-60.12.5.zip	-	-
		Mac UI SDK	hwmsdk-mac-60.12.5.zip	-	-
		Mac Electron Demo	hwmsdk-electron-mac-60.12.5.zip	-	-
60.11.6 (停止维护)	2020-11-06	Android UI SDK	hwmsdk-android-60.11.6.zip	-	-
		iOS UI SDK	hwmsdk-ios-60.11.6.zip	-	-

版本	发布日期	资源	下载链接	安装包增量	说明
		Windows UI SDK	hwmsdk-win-60.11.6.zip	-	-
60.10.5 (停止维护)	2020-10-15	Android UI SDK	hwmsdk-android-60.10.5.zip	-	-
		iOS UI SDK	hwmsdk-ios-60.10.5.zip	-	-
		Windows UI SDK	hwmsdk-ios-60.10.5.zip	-	-
60.8.1 (停止维护)	2020-09-07	Android UI SDK	hwmsdk-android-60.8.1.zip	-	-
		iOS UI SDK	hwmsdk-ios-60.8.1.zip	-	-
		Windows UI SDK	hwmsdk-win-60.8.3.zip	-	-
60.7.1 (停止维护)	2020-08-07	Android UI SDK	hwmsdk-android-60.7.1.zip	-	-
		iOS UI SDK	hwmsdk-ios-60.7.2.zip	-	-
		Windows UI SDK	hwmsdk-win-60.7.1.zip	-	-
60.6.4 (停止维护)	2020-06-29	Android UI SDK	hwmsdk-android-60.6.4.zip	-	-
		iOS UI SDK	hwmsdk-ios-60.6.1.zip	-	-
		Windows UI SDK	hwmsdk-win-60.6.4.zip	-	-
60.6.2 (停止维护)	2020-06-04	Android UI SDK	hwmsdk-android-60.6.2.zip	-	-
60.5.1 (停止维护)	2020-06-06	iOS UI SDK	hwmsdk-ios-60.5.1.zip	-	-

版本	发布日期	资源	下载链接	安装包增量	说明
60.6.3 (停止维护)	2020-06-06	Windows UI SDK	hwmsdk-win-60.6.3.zip	-	-

📖 说明

1. Android SDK如何选择32位和64位版本，请参考"[常见问题8.3](#)"。

3.2 APP SDK 包完整性校验

表 3-2 Android/iOS/Windows/Mac/Electron UI SDK SHA256 值

版本	资源	SHA256值
80.1 3.6 (最新)	Android UI SDK	4dbbac60c23d43dfe186a237cc5ab37367b3b6e20652404558974f7c9c9c30dd
	iOS UI SDK	3c3a7b58997fb431d2e2c9a49c7eeb0e5f7fdf0994f769e7d3573fd74bd548dd
	Windows UI SDK	9e5d7ef35498b4675b5e844a5d1d775fe90fa597f619e17ccca11c9c8d55bd9a
	Mac UI SDK	dcb35dca829b15a58082dadfb085a0c9bfeff8c11d4fdc0b5a072cfcc8d90bd2
	Electron UI SDK	db0aad3a11a324ea041b7cf4a1e8dcdcc02708c746351a55bb41e76d2cd40e0e
80.1 3.5 (最新)	Android UI SDK	d526b5223ddb038f5a93430ed15388b8c4ea07527a6713954d942f6609753c41
	iOS UI SDK	f8eefe147556533355e1507e5cbd63b6ebf7c53f78aeaec087584556d56fe585
	Windows UI SDK	25e442dabcd1e2320fad3e77d45a278b2ee848a0c8c71eaf2c7ec4174b88898
	Mac UI SDK	c9a10deefb37373e5d6d83df83c1da645417a7fc8badbe179530b8ee8571d463
	Electron UI SDK	028c86704cdb654db64e03b167b4692fa78d4cd46452a38d176673482e65e5e3

操作说明

可对下载的SDK包进行完整性校验，判断在下载过程中是否存在篡改和丢包现象。

操作步骤

- 1.在"**App SDK下载**"章节中，“表1 Android/iOS/Windows/Mac/Electron UI SDK资源下载路径”中下载的SDK的包；
- 2.下载SDK包到本地；
- 3.打开本地命令提示符框，输入如下命令，在本地生成已下载的SDK包的SHA256值，其中，“D:\hwmsdk-win-80.12.7.zip”为SDK包的本地存放路径和SDK包名，请根据实际情况修改。

```
certutil -hashfile D:\hwmsdk-win-80.12.7.zip SHA256
```

命令执行结果示例，如下所示：

```
SHA256 的 D:\hwmsdk-win-80.12.7.zip 哈希:  
85a691c46a56572ea092945e9565f7927e6c1b4c612b35d01b1c8a9524f8c498  
CertUtil: -hashfile 命令成功完成。
```

对比查询的SDK包的SHA256值和下载后的SDK包的SHA256值。如果一致，则表示下载过程不存在篡改和丢包。

4 一分钟跑通 Demo

4.1 跑通 Android Demo

本小节描述如何快速编译并运行客户端SDK的Android Demo。

环境要求

在Android Demo的编译运行过程中请满足如下环境要求。

表 4-1 环境要求（强制要求）

名称	要求
JDK版本	>= 1.8.0
最小Android API 版本	API 21
CPU架构支持	ARM64、ARMV7
集成工具	Android Studio 3.3.2
测试资源及App ID申请	请参见“ 开发前准备 ”。

表 4-2 编译配置要求（非强制要求）

名称	要求
jdk配置	sourceCompatibility JavaVersion.VERSION_1_8 targetCompatibility JavaVersion.VERSION_1_8
gradle version	gradle-5.4.1-all
compileSdkVersion	30

名称	要求
targetSdkVersion	30
gradle plugin version	com.android.tools.build:gradle:3.5.2
buildToolsVersion	28.0.3

操作步骤

步骤1 下载SDK和Demo源码

- 参考“[SDK下载](#)”章节，下载SDK的ZIP包。
- 解压ZIP包到DEMO文件夹下，得到如[图1](#)所示的文件及目录。
- Demo示例工程的目录结构如[图2](#)所示。

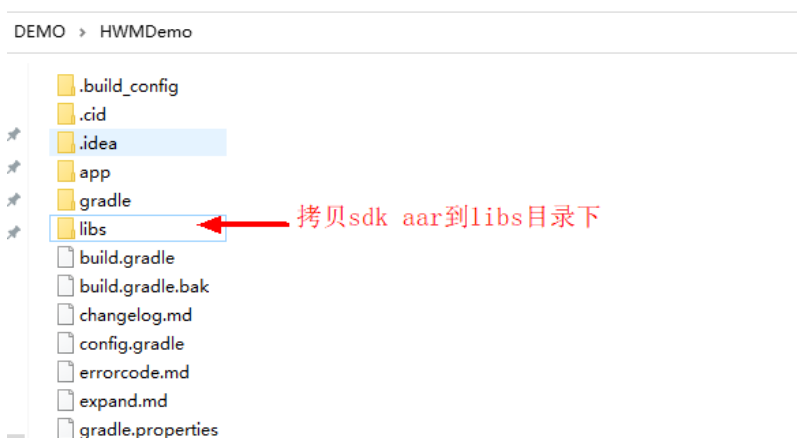
图 4-1 SDK ZIP 包解压后的目录结构



步骤2 拷贝SDK aar文件

- 拷贝SDK aar资源目录下的所有aar资源到Demo工程的libs文件夹中，如图2所示

图 4-2 Demo 示例工程的目录结构



步骤3 编辑和运行

- 打开Android Studio
- 打开Demo示例工程，如[图3](#)所示。
- 打开后如[图4](#)所示。

图 4-3 工程目录

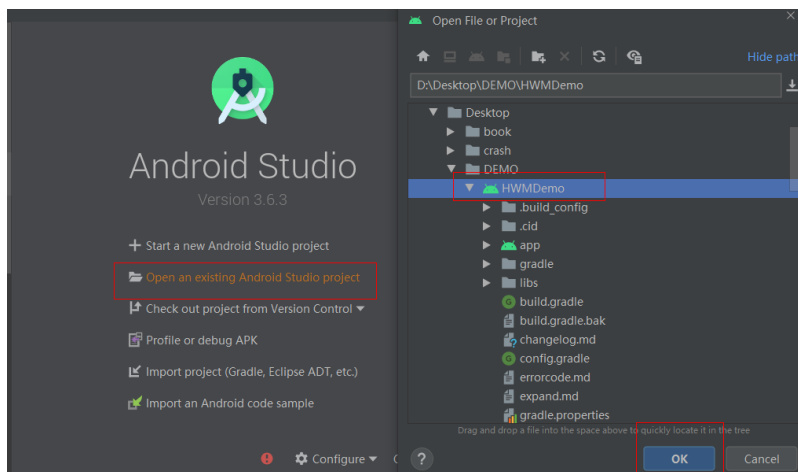
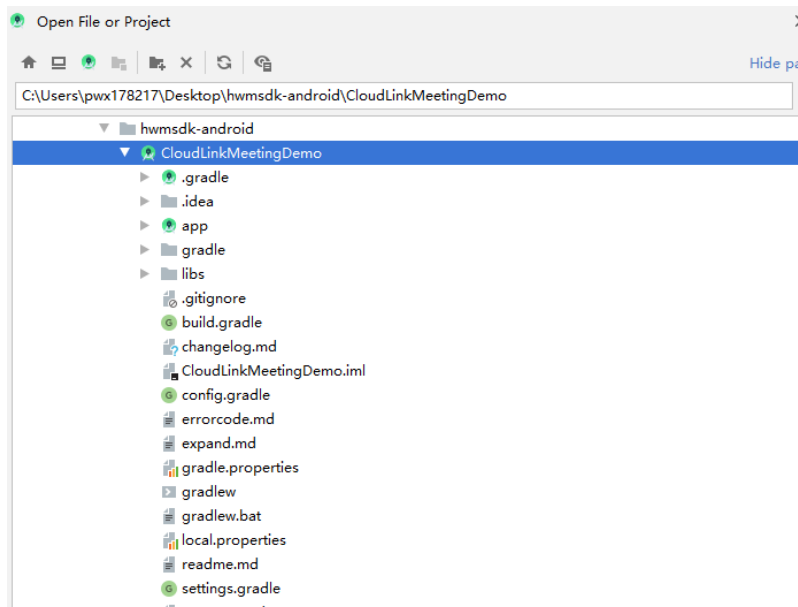


图 4-4 工程目录



- 打开gradle-wrapper.properties，确认是否可以下载到gradle，如果不可以则可以手动到该网址下载，配置成本地依赖。

图 4-5 配置 gradle-wrapper.properties

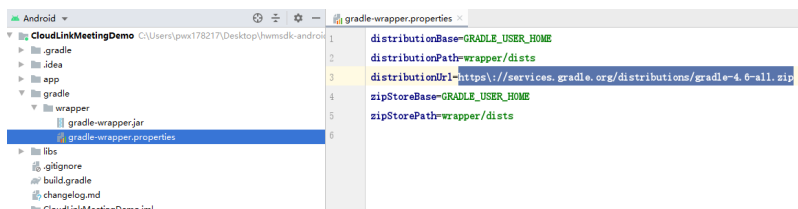
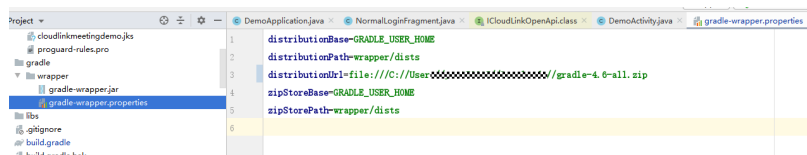


图 4-6 配置本地 gradle



- 打开build.gradle,查看是否可以连接到google,如果不可以则注释掉其他, 并且添加自己可用的maven仓

图 4-7 配置 maven



- 编译并运行demo工程

步骤4 申请App ID

App ID申请步骤请参考“[App ID鉴权介绍](#)”。

将 com.huawei.hwmdemo.utils.AppIdConstants 中的APP_ID替换为申请到的App ID。如下图所示：

图 4-8 替换 App ID

```
package com.huawei.hwmdemo.utils;

public class AppIdContants {

    /**
     * 请将APP_ID替换为您已申请到的App ID
     */
    public final static String APP_ID = "XXXXXXXXXXXXXXXXXXXX";

    /**
     * 1. 请将APP_KEY替换为您已申请到的App Key
     * 2. 将App key保存在Demo App中仅用于演示, 实际商用时请务必将App Key保存在您的服务端, 在服务端计算鉴权签名
     */
    public final static String APP_KEY = "XXXXXXXXXXXXXXXXXXXX";
}
```

步骤5 使用demo

- 登录
- 创建会议
- 会中界面

图 4-9 Android Demo 会中界面



----结束

4.2 跑通 iOS Demo

本小节描述如何快速编译并运行客户端SDK的iOS Demo。

环境要求

在iOS Demo的编译运行过程中请满足如下环境要求。

表 4-3 环境要求

环境和工具名称	版本要求	说明
操作系统	Mac OSX High Sierra 10.14.4及以上	Xcode对Mac系统要求，推荐版本10.15以上
Xcode	Xcode11及以上	Demo及文档以Xcode11.3.1为例
iOS系统版本	iOS10.0及以上，64位设备	屏幕共享功能需iOS12及以上
测试资源及App ID申请	-	请参见“ 开发前准备 ”。

表 4-4 项目配置要求

名称	要求	说明
Enable Bitcode	NO	不支持编译代码优化
Valid Architectures	arm64	支持CPU架构
Other Link Flags	-ObjC	把静态库中所有的Objective-C类和分类都加载到最后的可执行文件中

表 4-5 第三方开源库

开源库名称	建议版本
Masonry	必须使用1.1.0及以上
SSZipArchive	2.1.3
YYModel	1.0.4
CocoaLumberjack	3.4.1
libyuv	1.0.2

操作步骤

步骤1 下载SDK和Demo源码

1. 按照“[SDK下载](#)”章节，下载SDK的ZIP包。
2. 解压ZIP包，得到如下文件及目录。

图 4-10 SDK 包的目录结构

名称	修改日期	大小	种类
HWMSDKDemo	今天 下午 2:45	--	文件夹
libs	今天 下午 3:46	--	文件夹

步骤2 添加依赖库

1. 把HWMUISDK.framework、HWMMREngine.framework、HWMMRMedia.framework拷贝到Demo中。在“TARGETS->HWMSDKDemo->General->Frameworks, Libraries, and Embedded Content”中修改为Embed & Sign。
2. 把HWMExtension.framework拷贝到ScreenShareExtension中,“TARGETS->ScreenShareExtension->Build Settings->Other Linker Flags”中添加-**lc++**。
3. 把导入的HWMUISDK.framework设置为Embed & Sign。

图 4-11 HWMUISDK.framework、HWMMREngine.framework、HWMMRMedia.framework 设置

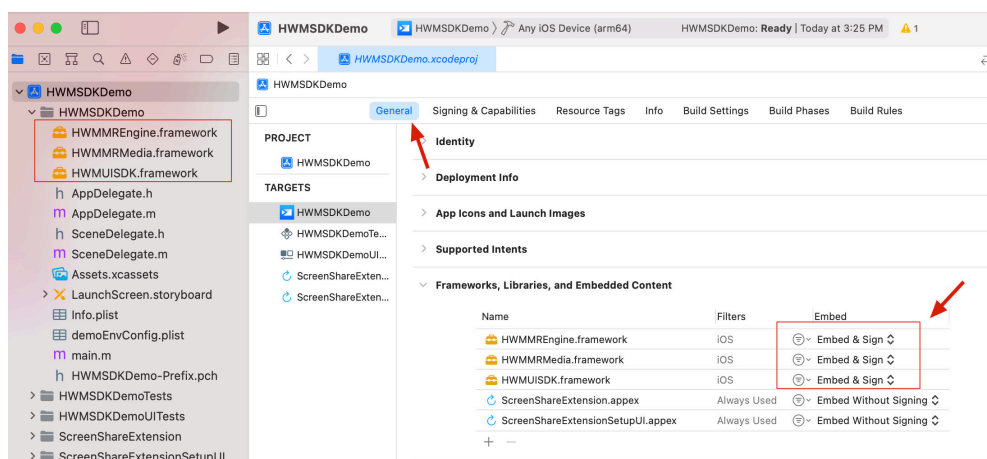
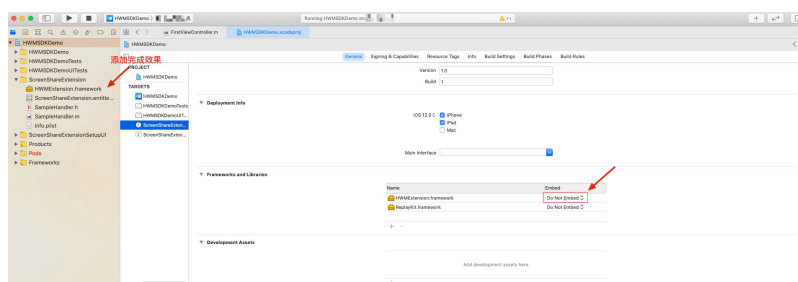


图 4-12 HWMExtension.framework 设置



4. 下载**masonry**, 拷贝源码到工程中。

步骤3 工程配置

1. 修改bundle identifier、App Groups, 安装对应的证书。创建和管理证书地址参见“**证书地址**”。
2. 编译运行。

步骤4 使用Demo

以创建1个即时会议为例:

图 4-14 iOS Demo 会中界面



----结束

4.3 跑通 Windows Demo

本小节描述如何快速编译并运行客户端SDK的Windows MFC Demo。

环境要求

在Windows MFC Demo的编译运行过程中请满足如下环境要求。

表 4-6 环境要求

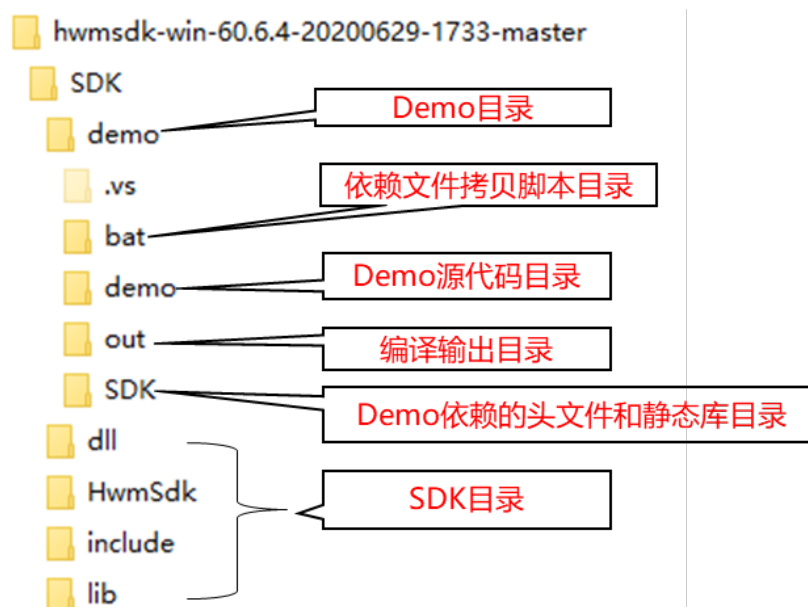
环境和工具名称	版本要求	说明
操作系统	Windows 10专业版	硬件要求： <ul style="list-style-type: none"> • CPU: i5-2400四核 3.1GHz及以上。 • 内存: 4GB及以上。
Microsoft Visual Studio	Visual Studio 2017	NA
测试资源及App ID申请	NA	请参见“ 开发前准备 ”。

操作步骤

步骤1 下载SDK和Demo源码

1. 按照“[SDK下载](#)”章节，下载SDK的ZIP包。
2. 解压ZIP包，得到如下文件及目录。

图 4-15 SDK 包的目录结构



步骤2 拷贝依赖文件

1. 运行bat目录下的批处理脚本。以编译32位Release版本为例，执行 copy_dependency_win32_release.bat脚本，脚本将：
 - SDK的头文件和lib文件拷贝到SDK目录下。
 - SDK的dll文件和资源文件拷贝到out目录下。

步骤3 申请App ID，并替换Demo中的App ID

App ID申请步骤请参考“[App ID鉴权介绍](#)”。

将"SDK\demo\demo\code\demoInitDlg.cpp"中的appId替换为申请到的App ID。如下图所示：

图 4-16 替换 App ID

```

BOOL demoInitDlg::OnInitDialog()
{
    BOOL ret = CDialogEx::OnInitDialog();

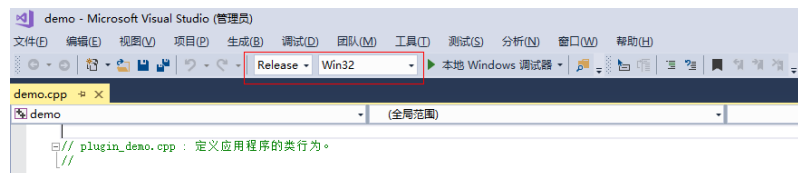
    m_comboSiteType.InsertString(0, _T("中国站点"));
    m_comboSiteType.InsertString(1, _T("国际站点"));
    m_comboSiteType.SetCurSel(0);

    m_editAppId.SetWindowText(_T(""));
    m_editExeName.SetWindowText(_T("HwmSdk"));
    m_logKeepDays.SetWindowText(_T("0"));
    return ret;
}
    
```

步骤4 编译和运行

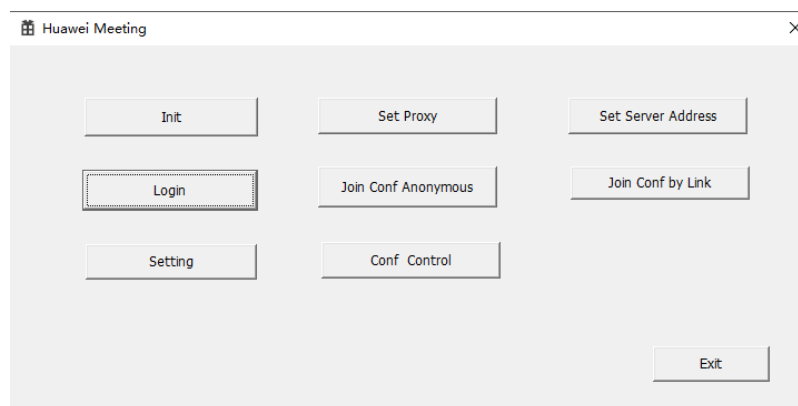
1. 双击demo目录下的demo.sln VS工程文件。
2. 在VS中选择32位Release版本。

图 4-17 选择 32 位 Release 版本



3. 编译并运行demo工程

图 4-18 Windows Demo 登录前界面

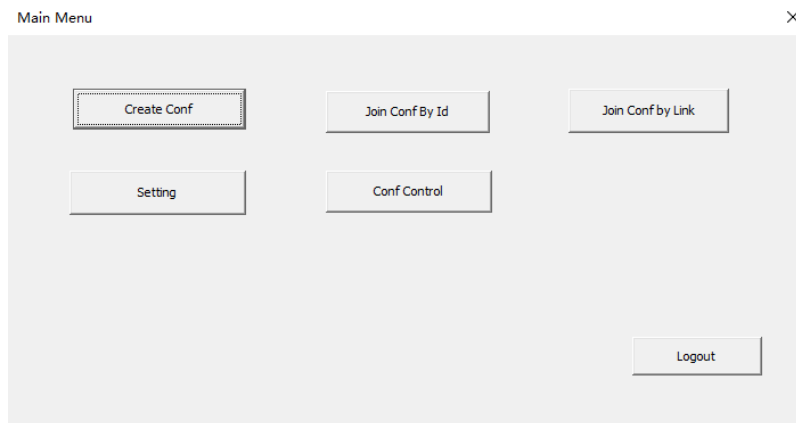


步骤5 使用Demo

假设环境中的网络没有Proxy，PC可以直接访问公网。以创建一个即时会议为例：

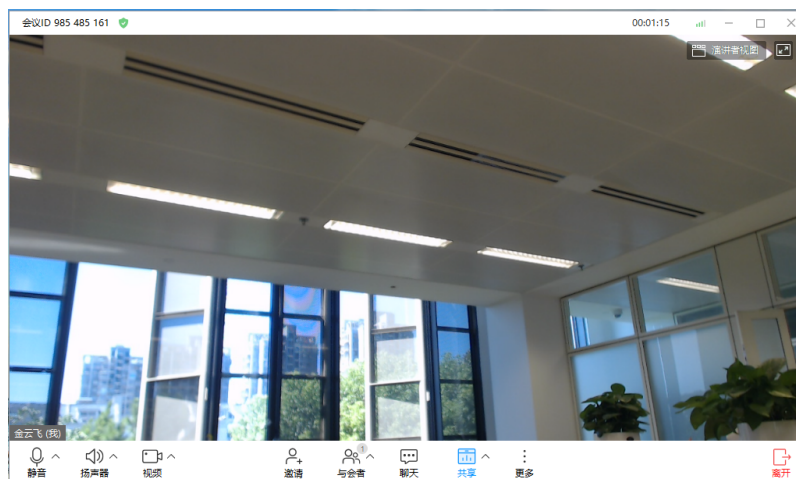
1. 点击“Init”按钮。
2. 点击“Login”按钮，并输入华为云会议帐号和密码。

图 4-19 Windows Demo 登录后界面



3. 点击“Create Conf”按钮，创建会议。

图 4-20 会中界面



----结束

4.4 跑通 Mac Demo

本小节描述如何快速编译并运行客户端SDK的Mac Native Demo。

环境要求

在Mac Native Demo的编译运行过程中请满足如下环境要求。

表 4-7 环境要求

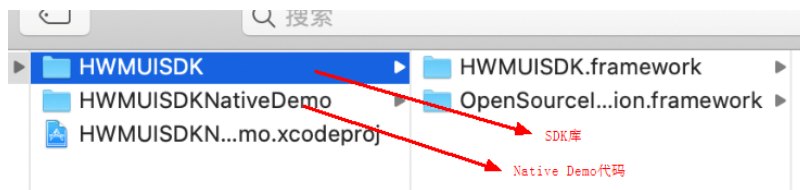
环境和工具名称	版本要求	说明
操作系统	Mac OSX High Sierra 10.14.4及以上	Xcode对Mac系统要求，推荐版本10.15以上
Xcode	Xcode11及以上	Demo及文档以xcode12.0.1为例
测试资源及App ID申请	-	请参见“ 开发前准备 ”。

操作步骤

步骤1 下载SDK和Demo源码

1. 按照“[SDK下载](#)”章节，下载SDK的ZIP包。
2. 解压ZIP包，得到如下文件及目录。

图 4-21 SDK 包的目录结构

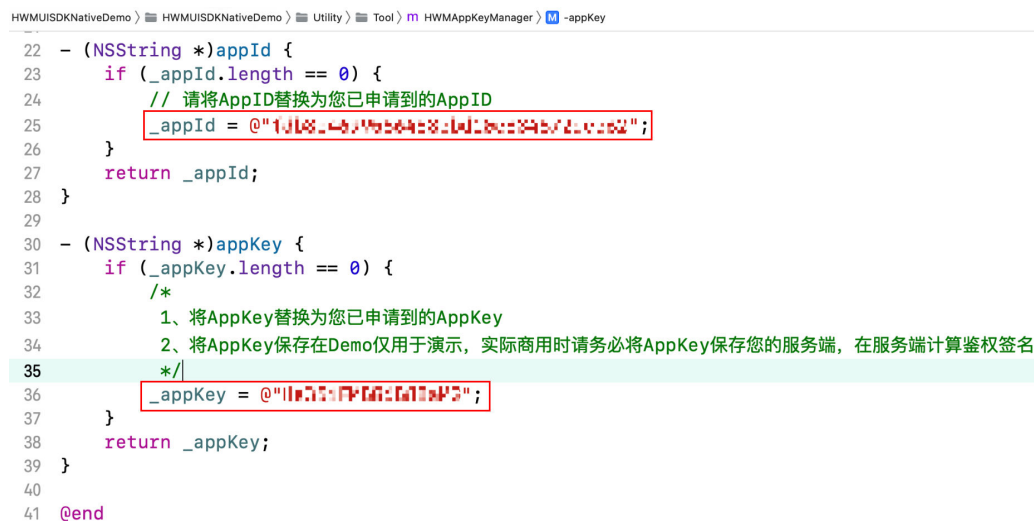


步骤2 申请App ID，并替换Demo中的App ID

App ID申请步骤请参考“[App ID鉴权介绍](#)”。

将 HWMAppKeyManager中的appId替换为申请到的App ID。如下图所示：

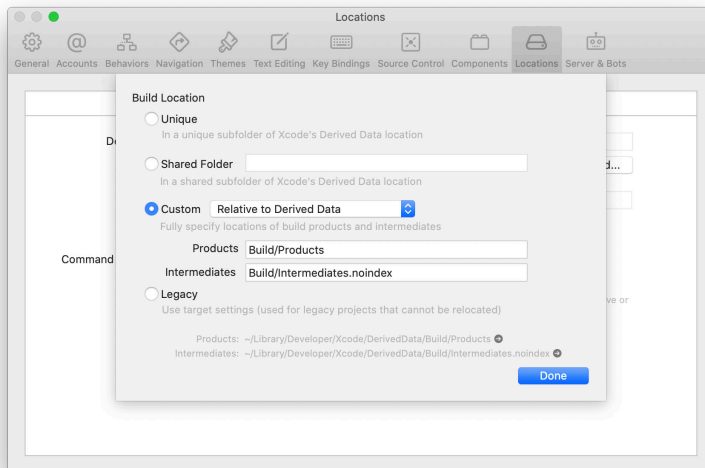
图 4-22 修改 Demo 默认 App ID



步骤3 编辑和运行

1. 修改bundle identifier、安装对应的证书。换成自己公司的对应证书。
2. 修改Xcode Preference的Build Location改成Custom

图 4-23 Xcode Build location 修改



3. 编译运行。

步骤4 使用Demo

以创建1个即时会议为例：

1. 点击“初始化”按钮，初始化SDK。
2. 登录
账号密码登录方式：点击“登录”按钮进入“登录页面”，在“登录页面”输入帐号密码后点击“登录”。
3. 点击“创建会议”按钮，进入创建会议界面，创建会议。

----结束

4.5 跑通 Electron Demo

本小节描述如何快速编译并运行客户端SDK的Electron Demo。

环境要求

在Electron Demo的编译运行过程中请满足如下环境要求。

表 4-8 Windows 环境要求

环境和工具名称	版本要求	说明
操作系统	Windows 10专业版	硬件要求： <ul style="list-style-type: none"> • CPU: i5-2400四核 3.1GHz及以上 • 内存: 4GB及以上
Python	3.x	可选，仅在需要重新编译addon插件时才需要
Microsoft Visual Studio	Visual Studio 2017	可选，仅在需要重新编译addon插件时才需要
Node.js	14.16.0及以上	NA
Electron	13.6.7及以上	NA
测试资源及App ID申请	NA	请参见“ 开发前准备 ”。

📖 说明

Node.js和Electron的版本配套关系，请参考<https://www.electronjs.org/docs/latest/tutorial/electron-timelines>

表 4-9 Mac 环境要求

环境和工具名称	版本要求	说明
操作系统	MacOS 10.15	NA
XCode	12.0及以上	NA
Node.js	14.16.0及以上	NA
Electron	13.6.7及以上	NA
node-gyp	8.1.4及以上	编译JavaScript桥接动态库需要
bindings	1.5.0	编译JavaScript桥接动态库需要
测试资源及App ID申请	NA	请参见“ 开发前准备 ”。

操作步骤

步骤1 下载SDK和Demo源码

1. 按照“[SDK下载](#)”章节，下载Electron SDK的ZIP包（如hwmsdk-electron-70.16.5.zip）。

2. 解压ZIP包，得到如下文件及目录。

图 4-24 SDK 包目录结构



步骤2 环境准备

1. 编译addon插件时，需要有相应编译环境，否则会因为无法编译而报错。

Windows:

- Nodejs
- Python 3.x: 注意配置环境变量
- visual C++编译环境: Visual C++ build tools 或者 Visual Studio Community (安装勾选“使用C++的桌面开发”)
- Electron:以13.6.7为例，运行npm install electron@13.6.7 -g

Mac:

- Nodejs
- XCode
- Electron:以13.6.7为例, 运行npm install -electron@13.6.7 -g
- node-gyp: 运行npm install node-gyp -g
- bindings: 运行npm install bindings -g

2. 如确认Nodejs已经安装, 请先配置npm镜像

安装依赖前先配置npm镜像, 以提高安装速度, 执行以下命令:

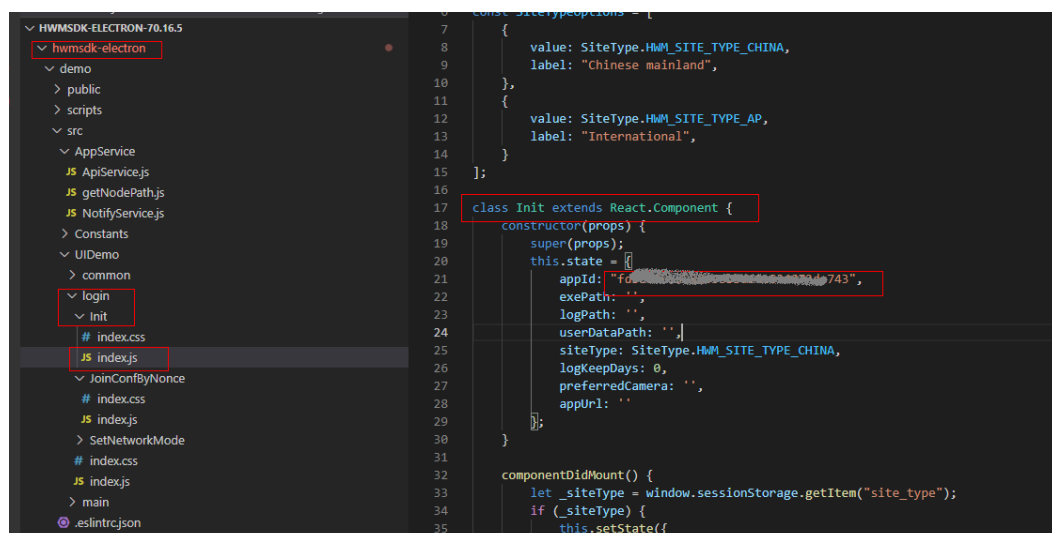
```
npm config set registry http://registry.npm.taobao.org/
npm config set electron_mirror https://npm.taobao.org/mirrors/electron/
```

步骤3 申请App ID, 并替换Demo中的App ID

App ID申请步骤请参考“[App ID鉴权介绍](#)”。

将"hwm sdk-electron\demo\src\UIDemo\login\Init\index.js"中的appId替换为申请的App ID。如下图所示:

图 4-25 替换 App ID



步骤4 编译addon插件

📖 说明

1. 如果已经安装node, 请先确定自己node版本是32位还是64位, 可以执行cmd/shell命令查看node版本的位数, 结果: x64为64位, ia32为32位。

```
$ node -p "process.arch"
```

2. 如果没有安装node, 可到[nodejs官网](#)下载对应版本。

1. 运行hwm sdk-electron目录下编译addon工程的bat或sh脚本。脚本会安装相应的环境并触发编译, 在build\Release目录下生成对应的HwmUisdk.node。
 - 如果需要编译Windows 32位addon插件, node版本是32位, 执行步骤1中解压目录下的build_nodeaddon_win32.bat
 - 如果需要编译Windows 64位addon插件, node版本是64位, 执行步骤1中解压目录下的build_nodeaddon_x64.bat

- 如果需要编译Mac addon插件，执行步骤1中解压目录下的build_nodeaddon_mac.sh

说明

bat或sh脚本的作用：

- 编译addon插件，生成HwmUisdk.node文件，并拷贝到对应平台目录下("sdk/win32"、"sdk/x64"、"sdk/mac")。

步骤5 运行Electron Demo

1.运行hwmsdk-electron目录下编译Demo工程的bat或sh脚本。脚本会安装相应的环境，拷贝sdk库和HwmUisdk.node，将ts转为js，并触发编译，然后运行demo。

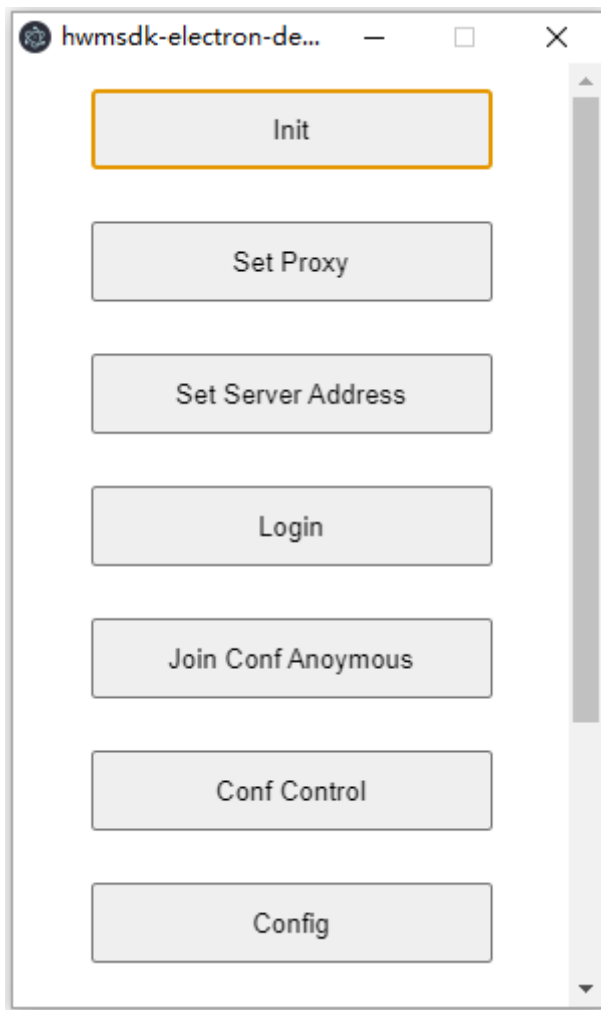
- 如果需要运行Windows 32位demo，node版本是32位，执行run_demo_win32.bat
- 如果需要运行Windows 64位demo，node版本是64位，执行run_demo_x64.bat
- 如果需要运行Mac demo，执行run_demo_mac.sh

说明

bat或sh脚本的作用：

- 编译ts接口，生成js和types文件。
- 将对应平台的的SDK文件拷贝到对应的目录下("sdk/win32"、"sdk/x64"、"sdk/mac")。
- 运行Demo。

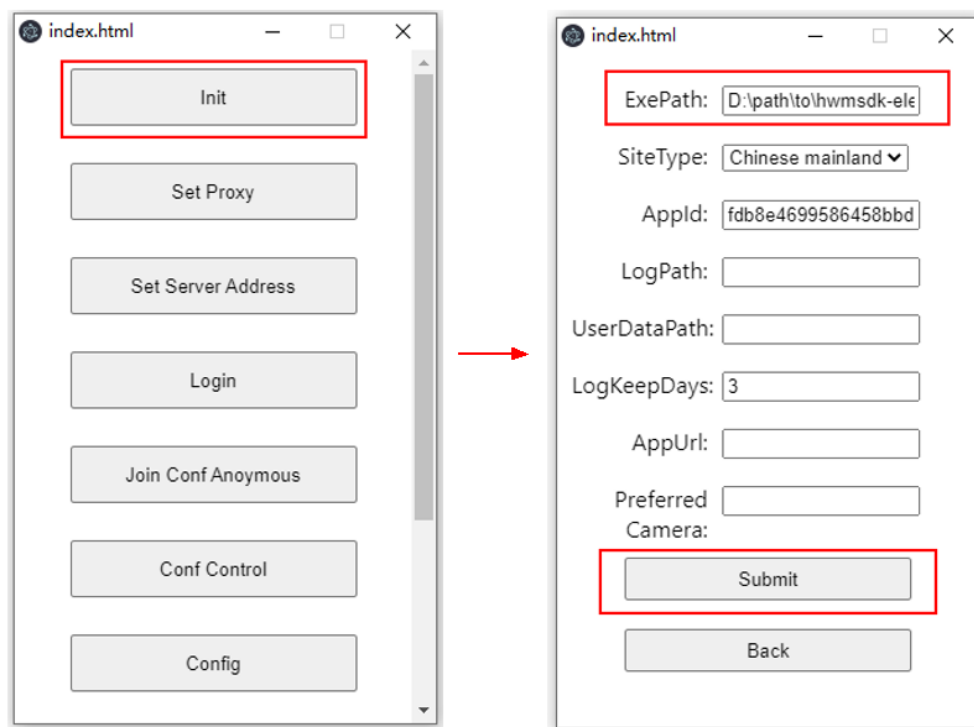
图 4-26 Electron Demo 主界面



步骤6 使用Electron Demo

- 点击“Init”初始化SDK。（如果是32位，ExePath不用填）

图 4-27 初始化 SDK



说明

ExePath是Windows SDK的HwmSdk所在的绝对路径，示例路径：“D:\path\to\hwmsdk-electron-70.16.5-src\hwmsdk-electron\sdk\x64\HwmSdk\HwmSdk.exe”。

图 4-28 登录

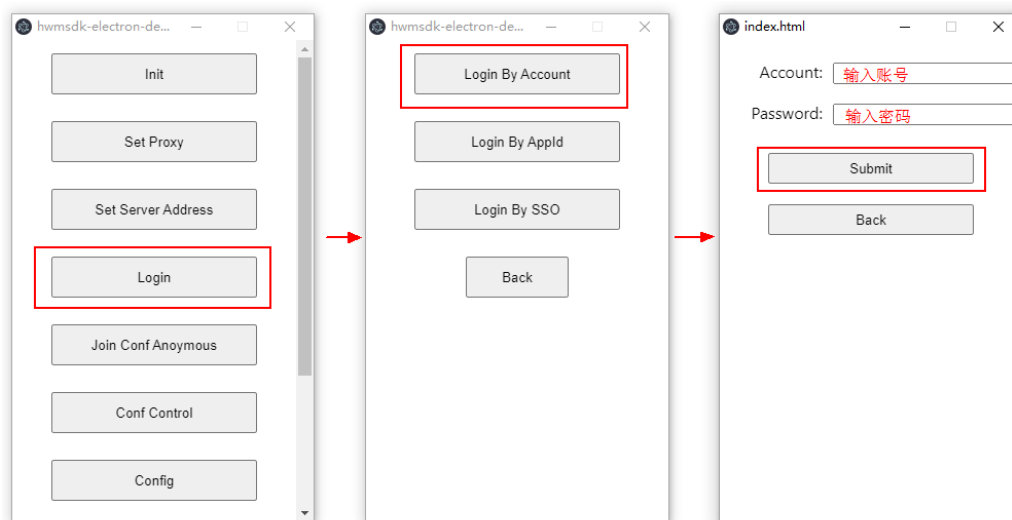
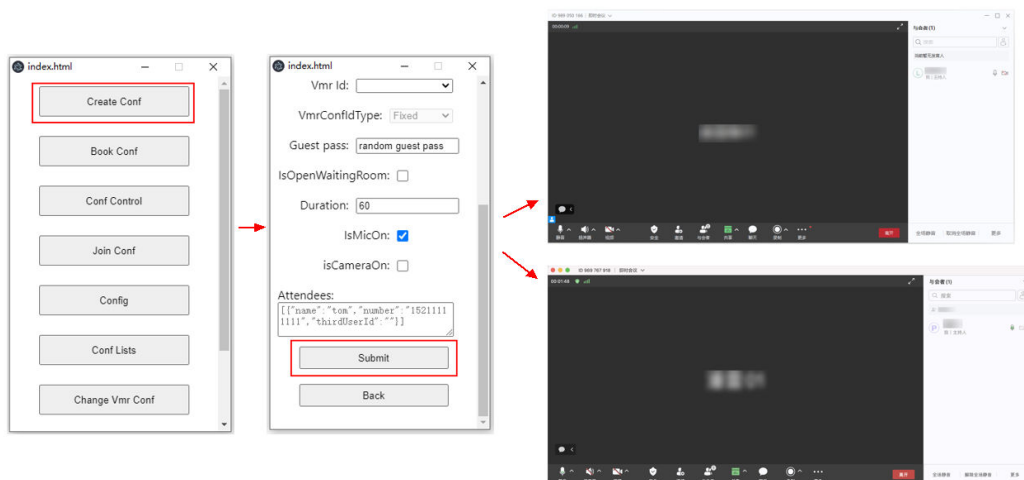


图 4-29 创建会议



----结束

5 Android SDK

5.1 概述

会议二次开发Android SDK 提供了一套完整的接口集合，开发者可以通过调用 HUAWEI CLOUD Meeting SDK（以下简称SDK）开放的API，快速集成音视频会议，包括创建/预约/加入会议、进行屏幕共享、点对点呼叫等。

📖 说明

- Android SDK只能运行在主流的手机和Pad上。如果运行在定制硬件的Android设备上，可能会导致画面分辨率低、画面旋转角度不正确、回声等问题。
- Android SDK不支持在华为智慧屏和为企业智慧屏上做二次开发。

5.2 修订记录

表 5-1 修订记录

日期	版本	修订内容
2023-2-24	80.13.6	1. 问题修复，体验更稳定 具体使用请见 80.13.6更新内容
2023-2-7	80.13.5	1. 上传头像接口限制上传文件的格式和大小，具体请见 上传个人头像 2. 新增设置语言接口，具体请见 设置语言 3. 问题修复，体验更稳定 具体使用请见 80.13.5更新内容

日期	版本	修订内容
2023-1-9	80.12.6	<ol style="list-style-type: none"> 1. 开启功能接口enableFeature 新增是否显示录制安全提示窗枚举值 HWM_FEATURE_TYPE_CONFIRM_RECORD。具体请见枚举 HWMEnableFeatureType说明 2. 问题修复，体验更稳定。 具体使用请见80.12.6更新内容
2022-12-26	80.12.5	<ol style="list-style-type: none"> 1. 修复多个问题
2022-11-22	80.10.6	<ol style="list-style-type: none"> 1. 新增异步初始化接口，具体请见 异步初始化 具体使用请见80.10.6更新内容
2022-11-15	80.10.5	<ol style="list-style-type: none"> 1. 预约周期会议接口废弃cycleCount字段，具体请见 预约周期会议 2. 获取云会议室列表和修改云会议室信息接口新增参数，具体请见 获取云会议室列表、修改云会议室信息 3. 新增设置共享权限接口 setShareScreenPermission, 具体请见设置共享权限 具体使用请见80.10.5更新内容
2022-10-10	80.9.5	<ol style="list-style-type: none"> 1. 编辑会议和会议详情新增字段isLiveOn 是否开启直播。 2. 问题修复 具体使用请见80.9.5更新内容
2022-9-2	80.8.5	<ol style="list-style-type: none"> 1. 创建会议、预约会议、编辑会议支持设置来宾密码。 2. 问题修复 具体使用请见80.8.5更新内容
2022-7-30	80.7.5	<ol style="list-style-type: none"> 1. 问题修复 具体使用请见80.7.5更新内容
2022-6-15	80.5.11	<ol style="list-style-type: none"> 1. 新增禁用提示音接口 2. 问题修复 具体使用请见80.5.11更新内容
2022-5-18	80.5.5	<ol style="list-style-type: none"> 1. 预约会议、编辑会议、获取会议详情接口增加等候室开关功能，新增 isOpenWaitingRoom字段 2. 问题修复 具体使用请见80.5.5更新内容

日期	版本	修订内容
2022-4-15	70.16.5	<ol style="list-style-type: none"> 1. 新增周期会议相关接口 2. 入会参数PasswordJoinConfParam 新增 isStopConflictConf字段，能够强制结束和当前VMR会议资源冲突的会议 3. 会议通知栏支持在来电时可配置是否显示通知 4. sso 登录接口调整 5. 错误码整改，发布新版本错误码描述以及新旧版本错误码对应关系。 <p>具体使用请见70.16.5更新内容</p>
2022-2-19	70.14.5	<ol style="list-style-type: none"> 1. 预约、编辑会议支持设置“来宾在主持人之前加入会议”，并可设置提前入会时间，具体参数请见 预约会议参数说明。 2. 预约云会议室时支持选择预约固定云会议室ID或随机云会议室ID，具体请见 VmridType枚举说明。 3. APPID登录支持指定部门ID，具体请见 AppIdAuthParam参数说明。 4. 问题修复 <p>具体使用请见70.14.5更新内容</p>
2022-1-19	70.13.5	<ol style="list-style-type: none"> 1. 登录、登出接口新增uuid、account、thirdAccount字段，具体请见LoginResult类型说明、LogoutResult类型说明。如果是老版本升级到70.13.5版本,需要修改回调函数的定义。 2. 第三方视频推流接口支持MMR会议，具体请见第三方视频推流。 3. 业务消息通知中新增onConfIncoming接口，允许在入会前获取来电信息，具体请见业务消息通知。 4. 问题修复 <p>具体使用请见70.13.5更新内容</p>
2021-12-23	70.12.6	<ol style="list-style-type: none"> 1. 修复弱网情况下导致清晰度降低后不能恢复的问题 <p>具体使用请见70.12.6更新内容</p>
2021-12-15	70.12.4	<ol style="list-style-type: none"> 1. RxJava升级到3.0.13版本 2. 问题修复 <p>具体使用请见70.12.4更新内容</p>

日期	版本	修订内容
2021-12-3	70.11.5	<ol style="list-style-type: none"> 1. 新增配置日志保存天数 2. HMS Scankit升级到2.1.0.300版本 3. 问题修复 具体使用请见 70.11.5 更新内容
2021-10-22	70.10.5	<ol style="list-style-type: none"> 1. 去除无效资源 2. 问题修复 具体使用请见 70.10.5 更新内容
2021-09-16	70.9.5	<ol style="list-style-type: none"> 1.新增SSO登录方式 2.移除tinypinyin库 具体使用请见 70.9.5 更新内容
2021-08-28	70.8.6	问题修复，体验更稳定
2021-08-18	70.8.5	<ol style="list-style-type: none"> 1.从此版本开始，Android SDK切换为AndroidX依赖 2.支持获取与会者列表（含网络研讨会） 具体使用请见 70.8.5 更新内容
2021-07-16	70.7.5	<ol style="list-style-type: none"> 1.支持会中设置本会场视频画面是否镜像 2.支持安卓11的API 30版本 3.okhttp升级到4.9.1 具体使用请见 70.7.5 更新内容
2021-06-10	70.6.5	<ol style="list-style-type: none"> 1.删除了ZXing的依赖 2.tinypinyin 升级到3.0.0 3.aspectjrt升级到1.9.6 4.gson升级到2.8.6 具体使用请见 70.6.5 更新内容
2021-05-17	70.5.8	<ol style="list-style-type: none"> 1.会中与会者和聊天模块适配横屏体验 2.支持聊天弹幕 具体使用请见 70.5.8 更新内容

日期	版本	修订内容
2021-04-15	70.4.5	<ol style="list-style-type: none"> 1.预约会议接口添加返回值参数说明 2.sdk初始化约束，限制初始化必须在Application的OnCreate主线程中执行 3.新增获取用户当前状态接口 4.会中新增主持人邀请他人共享，主持人也可快速取消共享 5.移动端的前后摄像头切换按钮放到主界面上 6.会议日志大小优化 7.libs文件夹中aar文件整合，使用时需先清空之前历史版本文件，重新导入 具体使用请见 70.4.5 更新内容
2021-03-12	70.3.5	<ol style="list-style-type: none"> 1.移除会中自定义分享链接，自定义分享链接功能可以通过自定义会中分享面板实现 2.去掉短信权限的要求 3.错误码变更 4.观看共享时叠加发言人视频窗口，并支持下拉关闭 5.支持手机、平板同时登录 具体使用请见 70.3.5 更新内容
2021-02-10	60.15.6	<ol style="list-style-type: none"> 1.新增自定义来电自动处理接口 2.新增自定义设置来电响铃和震动 3.sdk 资源包aar依赖变更 4.自定义隐藏外部标签， 具体使用请见 60.15.6 更新内容
2021-01-08	60.13.2	<ol style="list-style-type: none"> 1. 添加与会者 2. 离开会议 3. 结束会议 4. 错误码更新 具体使用请见 60.13.2 更新内容
2020-12-07	60.12.5	<ol style="list-style-type: none"> 1. 创建会议支持指定会议室（vmrId）和入会范围（joinConfRestrictionType） 2. 错误码更新 3. 新增头像上传接口 4. 新增头像拦截接口，可以使用自己的头像显示 5. 自定义菜单样例更新 具体参见 60.12.5更新内容

日期	版本	修订内容
2020-11-07	60.11.1	1、新增支持会议自动录制 2、新增自定义会议来电接听按钮显示 3、新增呼叫设置昵称 具体参见 60.11.1更新内容
2020-10-15	60.10.5	1、新增是否开启第三方数据采集配置： 2、新增自定义接入站点： 3、自定义会中分享面板参数变更： 4、加入会议接口参数变更： 具体参见 60.10.5更新内容
2020-09-04	60.8.1	1、初始化增加App ID，可以传入企业App ID 2、新增App ID登录、预约会议、编辑会议、会议列表、会议详情、取消会议等接口 3、新增自定义是否开启高清优先、自定义会议接听摄像头麦克风开关等配置接口。 4、创建会议参数新增是否开启录制。 5、问题修复，体验更稳定 具体参见 60.8.1更新内容
2020-07-30	60.7.1	1、会议中UI界面优化，菜单目录结构和接口发生变更，逻辑更清晰 2、支持会议中成员文字聊天，支持自定义配置是否显示 3、会控菜单自定义接口优化，部分接口替换，使用更灵活 具体参见 60.7.1更新内容
2020-06-29	60.6.4	新增如下接口 <ul style="list-style-type: none"> ● 新增回调会中是否是主持人 ● 新增接口当前是否是主持人 修改如下接口 <ul style="list-style-type: none"> ● 部分接口修改 参照自定义底部菜单栏
2020-05-08	60.5.3	更新部分使用文档： <ul style="list-style-type: none"> ● 删除“创建会议（携带与会人信息）”接口，和“创建会议”接口合并 ● 优化接口名称和结构
2019-11-12	6.1.223	修复部分已知问题

日期	版本	修订内容
2019-08-31	6.1.203	新增如下接口 <ul style="list-style-type: none"> • 注销功能 修改如下接口 <ul style="list-style-type: none"> • 登录登录: 接口参数发生变更, 登录地址与端口不再需要传入 • 加入会议: 加入会议接口与匿名入会接口合并, 删除匿名入会接口, 加入会议接口参数发生变更
2019-07-23	6.1.195	新增如下接口: <ul style="list-style-type: none"> • 匿名入会: 支持用户匿名入会 (未登录场景下的入会)。 • 支持会议中返回到悬浮窗状态。
2019-04-13	6.1.27	新增如下接口: <ul style="list-style-type: none"> • 创建会议 (携带与会人信息): 支持在创建即时会议的时候携带与会者, 与会者会收到呼叫入会的来电。 新增如下能力: <ul style="list-style-type: none"> • 来电接听: 支持应用在前台收到其他人的来电接听后加入会议。 • 应用集成, 详见基于应用的集成。
2019-03-29	6.1.23	首次发布。

5.3 快速入门

本文面向有一定Android原生开发能力的开发者, 以登录华为云会议并发起和加入视频会议的流程为例, 介绍如何使用SDK 进行二次开发

开发环境准备

在接入SDK前需要准备如下环境:

环境要求 (强制要求)

开发环境, 目的是为了给编译运行源码的人员提供参考:

环境要求 (非强制要求)

SDK 快速集成

步骤1 解压SDK

下载解压软件包, 得到如下几个文件:

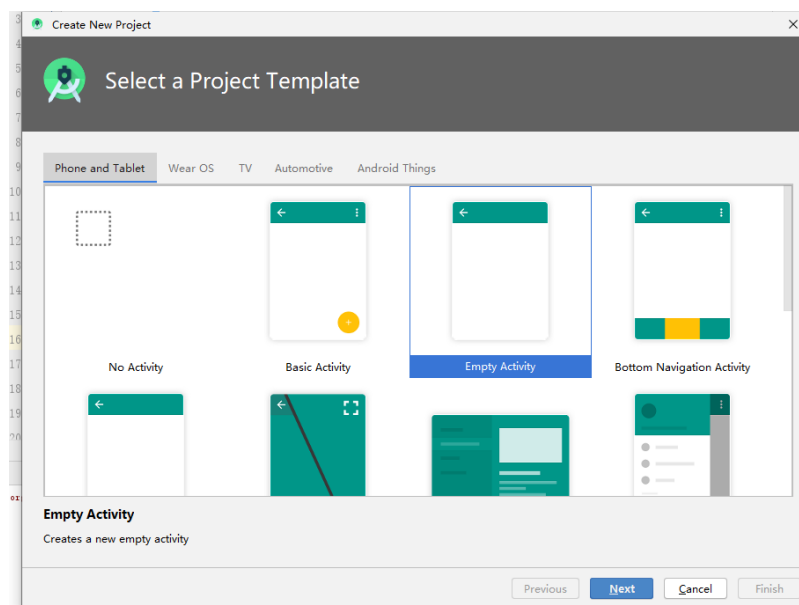
图 5-1 文件目录

名称	修改日期	类型
HWMDemo	2023/2/14 14:34	文件夹
languagetemplates	2023/2/14 14:34	文件夹
libs	2023/2/14 14:34	文件夹
virtualbackgroundlibs	2023/2/14 14:34	文件夹

步骤2 新建工程

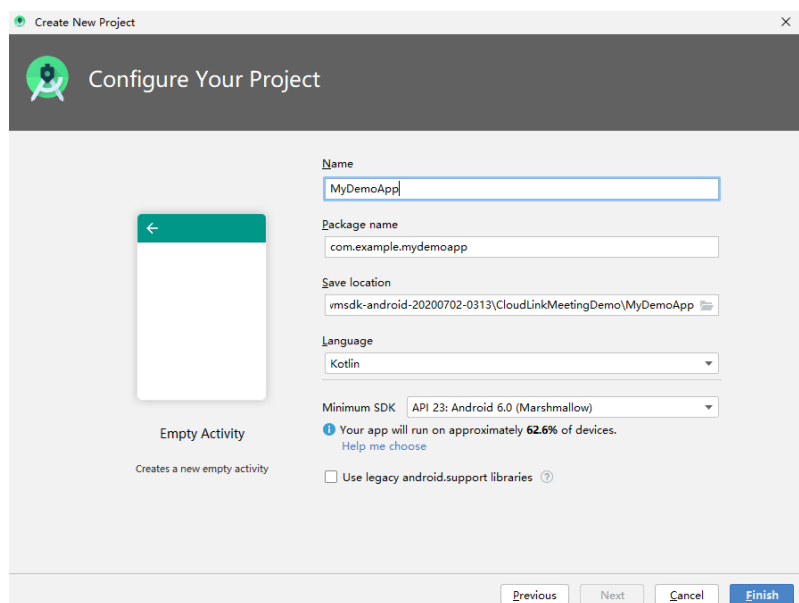
- 打开Android Studio, 单击“File -> New -> New Project...”新建工程, 选择 Phone and Tablet -> Empty Activity 单击Next:

图 5-2 创建工程



- 输入相关项目信息, 单击“Finish”完成工程创建。(请注意最小Android API 版本为23):

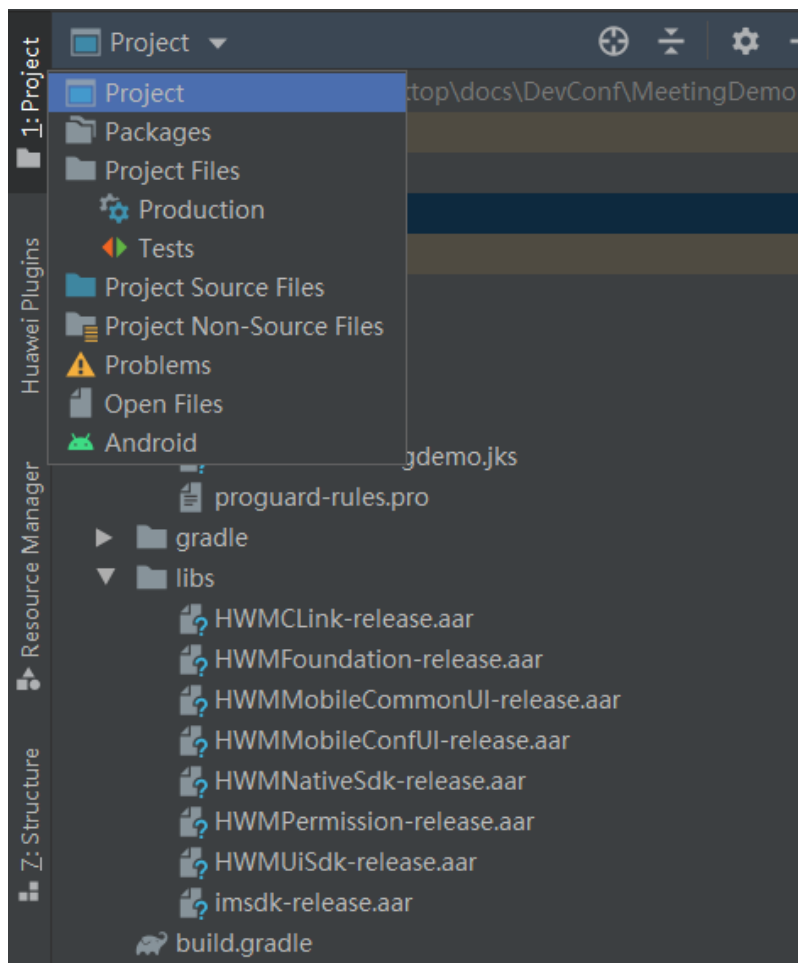
图 5-3 设置名称



步骤3 集成SDK

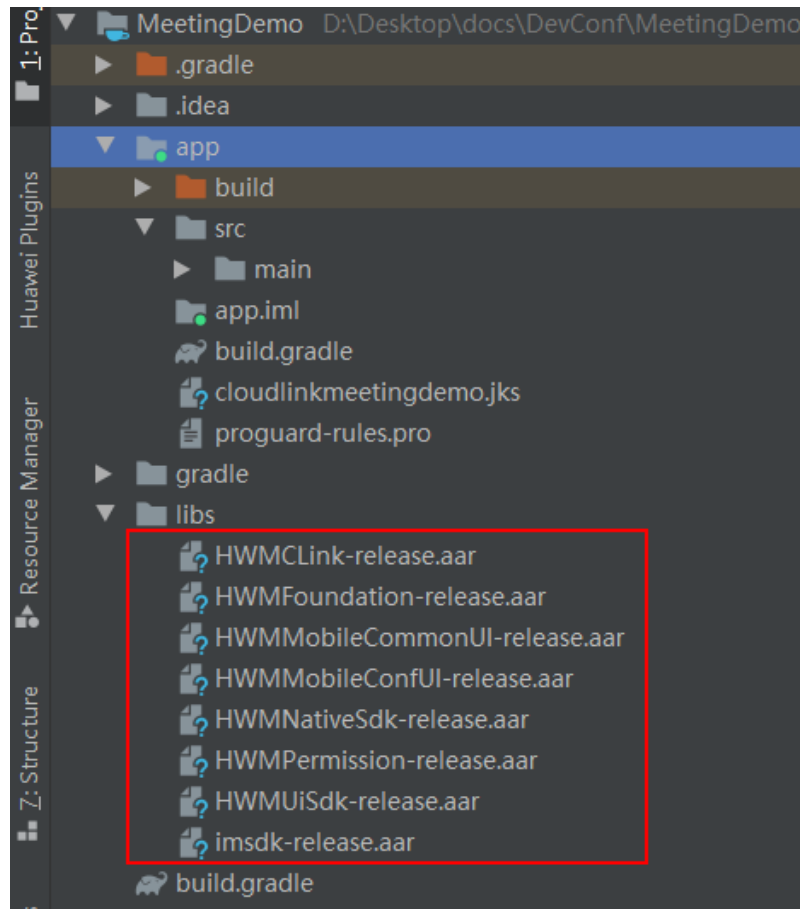
- 在Android Studio中，将工程目录结构切换到Project:

图 5-4 切换到 Project



- 将libs下的aar包拷贝到你的工程目录：

图 5-5 添加 aar



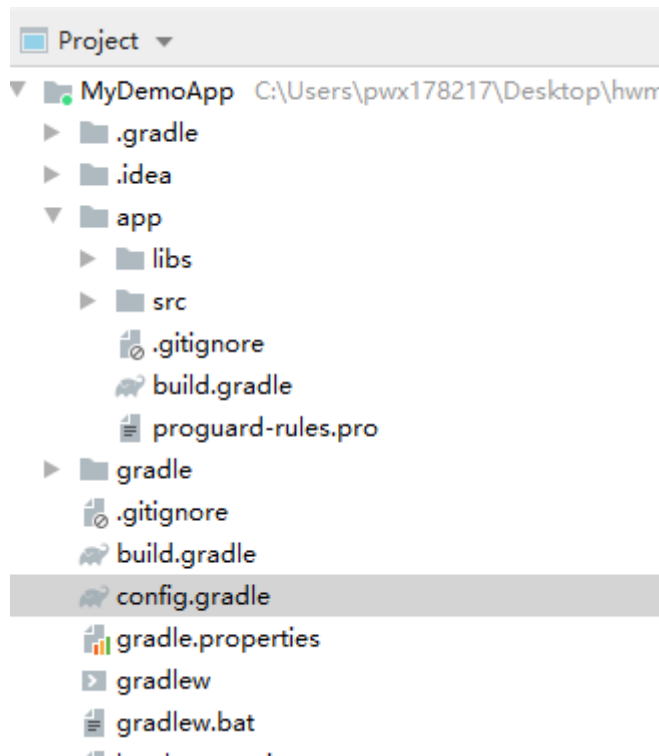
步骤4 添加依赖

- 添加三方依赖。

SDK依赖了一些优秀的开源库，需要在gradle中添加对这些库的依赖，方法如下。

在项目根目录下创建config.gradle（如您已创建统一配置文件，可参考demo中config.gradle中配置版本）：

图 5-6 打开 config.gradle



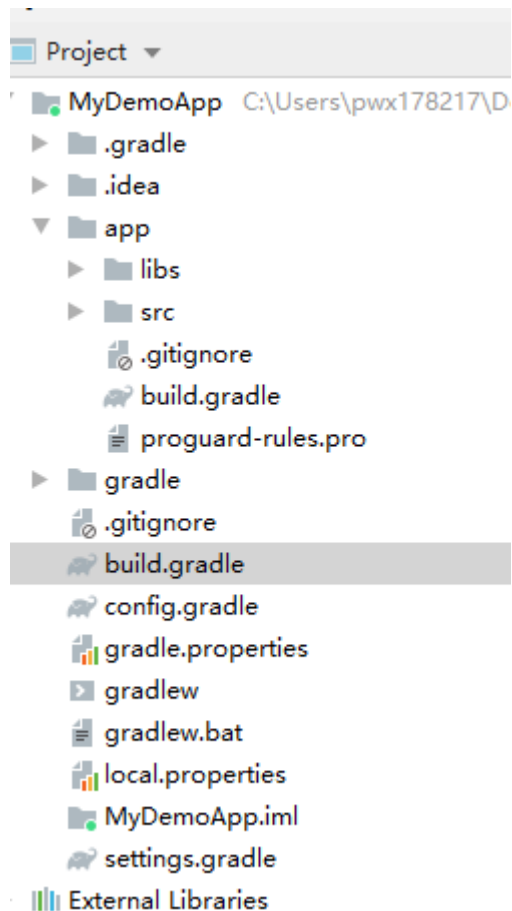
- 在项目build.gradle内首行添加:

```
apply from:"config.gradle"
```

config.gradle内使用到的第三方依赖声明及版本号:

```
"rxandroid" : 'io.reactivex.rxjava3:rxandroid:3.0.2',  
"appcompat-v7" : 'androidx.appcompat:appcompat:1.3.0',  
"vectordrawable" : 'androidx.vectordrawable:vectordrawable:1.2.0-alpha02',  
"rxjava" : 'io.reactivex.rxjava3:rxjava:3.0.13',  
"recyclerview" : 'androidx.recyclerview:recyclerview:1.2.1',  
"aspectjrt" : 'org.aspectj:aspectjrt:1.9.6',  
"okhttp" : 'com.squareup.okhttp3:okhttp:4.9.1',  
"gson" : 'com.google.code.gson:gson:2.8.9',  
"design" : 'com.google.android.material:material:1.3.0',  
"lifecycle-process" : 'androidx.lifecycle:lifecycle-process:2.4.1',  
"eventbus" : 'org.greenrobot:eventbus:3.3.1',  
"constraint-layout" : 'androidx.constraintlayout:constraintlayout:2.0.4',  
"glide" : 'com.github.bumptech.glide:glide:4.13.0',  
"swiperefreshlayout" : 'androidx.swiperefreshlayout:swiperefreshlayout:1.1.0',  
"hms:scan-kit" : 'com.huawei.hms:scan:2.1.0.300',
```


图 5-7 设置 build.gradle



- 在app/build.gradle添加开源包依赖及会议sdk依赖：

📖 说明

由于原Android support库已不再被Google维护演进，将从28版本升级替换为AndroidX。
因此华为云会议从8月发布的70.8.5版本开始，将不再在support库基础上开发，而是替换为AndroidX进行迭代。

如果您仍在原使用原Android support库依赖：

- 我们推荐您与我们一样更新替换使用新的AndroidX依赖，以免您在后续开发时无法正常对接我们采用AndroidX的SDK版本，相关文档请见[快速入门](#)。
- 如果您无法立即更新，我们会提供原有基于support库开发的SDK入口，并持续维护6个月。在维护期间，您仍可以集成我们的旧版SDK（70.7.5及之前版本）进行开发，我们会对于此类版本发现的问题进行修复与更新，但是之后迭代的新功能将只在AndroidX版本SDK上实现。

华为云会议期望给您带来更好的产品体验，如有问题，请拨打4000-955-988 | 950808 按1转1进行快速解答！

```
//sdk 用到的一些三方框架
api rootProject.ext.dependencies["appcompat-v7"]
api rootProject.ext.dependencies["recyclerview"]
api rootProject.ext.dependencies["design"]
api rootProject.ext.dependencies["aspectjrt"]
api rootProject.ext.dependencies["rxandroid"]
api rootProject.ext.dependencies["constraint-layout"]
api rootProject.ext.dependencies["glide"]
```

```
implementation rootProject.ext.dependencies["rxjava"]
implementation rootProject.ext.dependencies["gson"]
implementation rootProject.ext.dependencies["eventbus"]
implementation rootProject.ext.dependencies["okhttp"]
implementation rootProject.ext.dependencies["hms:scan-kit"]
implementation rootProject.ext.dependencies["lifecycle-process"]
implementation rootProject.ext.dependencies["swiperefreshlayout"]
```

- 添加会议依赖:

```
implementation(name: 'HWMFoundation-release', ext: 'aar')
implementation(name: 'HWMMobileCommonUI-release', ext: 'aar')
implementation(name: 'HWMMobileConfUI-release', ext: 'aar')
implementation(name: 'HWMPermission-release', ext: 'aar')
implementation(name: 'HWMNativeSdk-release', ext: 'aar')
implementation(name: 'HWMUiSdk-release', ext: 'aar')
implementation(name: 'HWMCLink-release', ext: 'aar')
if (needConfChat == null || needConfChat.toBoolean()) {
    implementation(name: 'imsdk-release', ext: 'aar')
}
```

📖 说明

部分模块在自己注入或者不打开的情况下，可以不添加依赖，并且删除对应aar
包括:

MobileHybrid模块[自定义会议中加人](#)

imsdk模块[自定义开关会中聊天功能](#)

- 在project/build.gradle allprojects/repositories内添加
flatDir { dirs 'libs' }

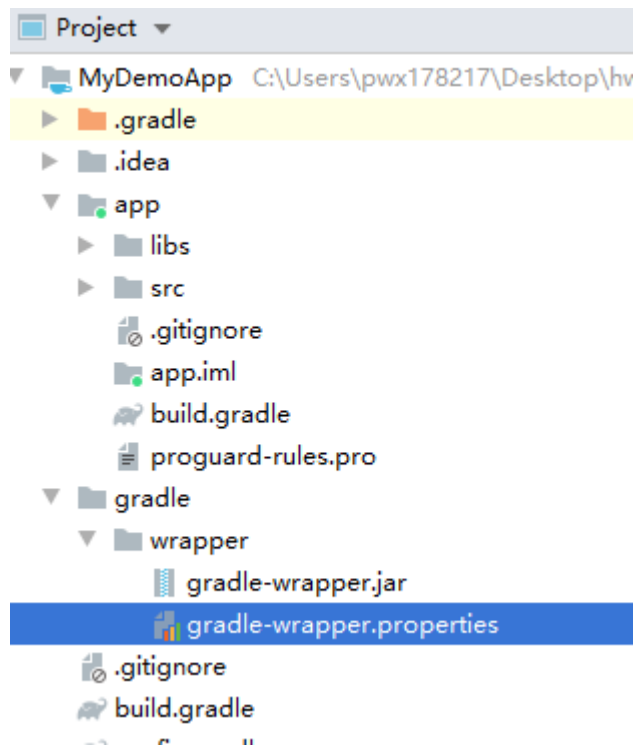
在无法连接谷歌的情况下可以添加自己的maven仓

图 5-8 添加 maven 仓

```
repositories {
    flatDir { dirs 'libs' }
    maven { url 'http://maven.aliyun.com/nexus/content/repositories/jcenter/' }
    maven { url 'http://developer.huawei.com/repo/' }
}
```

- 配置gradle-wrapper.properties, 参照[配置gradle-wrapper.properties](#)

图 5-9 gradle-wrapper.properties



- 单击Build -> make project,等待项目编译完成

📖 说明

如果客户集成libc++_shared.so到app层面,无需配置;如果使用的第三方库的aar中使用到了该动态库,则只需在build.gradle中做如下配置即可:

```
packagingOptions {
    merge lib/armeabi-v7a/libc++shared.so
}
```

步骤5 配置混淆

- 设置混淆,在build.gradle中添加:

```
buildTypes {
    release {
        minifyEnabled true
        proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'
    }
}
```

- 混淆文件proguard-rules.pro中添加

```
# 不开启优化,因为android的dex并不像Java虚拟机需要optimize(优化)和preverify(预检)两个步骤。
-dontoptimize
#不使用大小写混合类名,注意, windows用户必须为ProGuard指定该选项,
#因为windows对文件的大小写是不敏感的,也就是比如a.java和A.java会认为是同一个文件。
#如果不这样做并且你的项目中有超过26个类的话,那么ProGuard就会默认混用大小写文件名,导致class文件相互覆盖
-dontusemixedcaseclassnames
#指定不去忽略非公共库的类和成员
-dontskipnonpubliclibraryclasses
-dontskipnonpubliclibraryclassmembers

#抛出异常时保留代码行号,方便抛出异常时定位问题
```

```

-keepattributes SourceFile,LineNumberTable
#重命名抛出异常时的文件名称，方便抛出异常时定位问题
-renamesourcefileattribute SourceFile
# 混淆后就会生成映射文件
# 包含有类名->混淆后类名的映射关系
# 然后可以使用printmapping指定映射文件的名称
-verbose

# 不做预校验，preverify是proguard的4个步骤之一
# Android不需要preverify，去掉这一步可加快混淆速度
-dontpreverify

# dump.txt文件列出apk包内所有class的内部结构
-dump class_files.txt
# seeds.txt文件列出未混淆的类和成员
-printseeds seeds.txt
# usage.txt文件列出从apk中删除的代码
-printusage usage.txt
# mapping.txt文件列出混淆前后的映射
-printmapping mapping.txt

#忽略library里面非public修饰的类。
#library里的非公开类是不能被程序使用的，忽略掉这些类可以加快混淆速度。
#但是请注意，有一种特殊情况：有些人编写的代码与类库中的类在同一个包下，而且对该包的非public类进行了使用，在这种情况下，就不能使用该选项了。
#-skipnonpubliclibraryclasses

# 保留注解，因为注解是通过反射机制来实现的
-keepattributes *Annotation*
# 保留js接口
-keepattributes *JavascriptInterface*
# 保留exception
-keepattributes Exceptions
# 保留内部类
-keepattributes InnerClasses
#保留泛型
-keepattributes Signature

#保留本地native方法不被混淆
-keepclasseswithmembernames class * {
    native <methods>;
}

#保留枚举类
-keepclassmembers enum * {
    public static **[] values();
    public static ** valueOf(java.lang.String);
}
# 保留Parcelable序列化类不被混淆
-keep class * implements android.os.Parcelable {
    public static final android.os.Parcelable$Creator *;
}
# 保留Serializable序列化的类不被混淆
-keepclassmembers class * implements java.io.Serializable {
    static final long serialVersionUID;
    private static final java.io.ObjectStreamField[] serialPersistentFields;
    !static !transient <fields>;
    !private <fields>;
    !private <methods>;
    private void writeObject(java.io.ObjectOutputStream);
    private void readObject(java.io.ObjectInputStream);
    java.lang.Object writeReplace();
    java.lang.Object readResolve();
}

```

```

}

# 保留资源应用名
-keepclassmembers class **.R$* {
    public static <fields>;
}
# 忽略support包下的警告
-dontwarn android.support.**
# 保留support包下的动画
-keep class android.support.annotation.Keep

-keep @android.support.annotation.Keep class * {*;}
-keepclasseswithmembers class * {
    @android.support.annotation.Keep <methods>;
}
-keepclasseswithmembers class * {
    @android.support.annotation.Keep <fields>;
}
-keepclasseswithmembers class * {
    @android.support.annotation.Keep <init>(…);
}
#保留系统类库
-keep public class * extends android.app.Activity
-keep public class * extends android.app.Application
-keep public class * extends android.app.Service
-keep public class * extends android.content.BroadcastReceiver
-keep public class * extends android.preference.Preference
-keep public class * extends android.content.ContentProvider
-keep public class * extends android.view.View {
    public <init>(android.content.Context);
    public <init>(android.content.Context, android.util.AttributeSet);
    public <init>(android.content.Context, android.util.AttributeSet, int);
    public void set*(…);
}

#保留华为sdk相关
-keep class com.hianalytics.android.**{*;}
-keep class com.huawei.updatesdk.**{*;}
-keep class com.huawei.hms.**{*;}
-keep class com.huawei.android.hms.agent.**{*;}

#OKhttp 混淆配置
#https://github.com/square/okhttp/blob/5fe3cc2d089810032671d6135ad137af6f491d28/
README.md#proguard
-dontwarn okhttp3.**
-dontwarn okio.**
-dontwarn javax.annotation.**
-dontwarn org.conscrypt.**
# A resource is loaded with a relative path so the package of this class must be preserved.
-keepnames class okhttp3.internal.publicsuffix.PublicSuffixDatabase

#Eventbus 混淆配置
#http://greenrobot.org/eventbus/documentation/proguard/
-keepclassmembers class ** {
    @org.greenrobot.eventbus.Subscribe <methods>;
}
-keep enum org.greenrobot.eventbus.ThreadMode { *; }

# Only required if you use AsyncExecutor
-keepclassmembers class * extends org.greenrobot.eventbus.util.ThrowableFailureEvent {
#    <init>(java.lang.Throwable);

```

```
#}

#配置HME java不混淆,C++ 会调用, HMW-Video.jar
-keep class com.huawei.media.audio.AudioDeviceAndroid { *;}
-keep class com.huawei.media.audio.AudioDeviceAndroidService { *;}
-keep class com.huawei.media.audio.JniAudioDeviceImpl { *;}
-keep class com.huawei.media.audio.JniAudioDeviceMeetingImpl { *;}
-keep class com.huawei.media.audio.JniAudioDeviceRtcImpl { *;}
-keep class com.huawei.media.video.Camera2Characteristic { *;}
-keep class com.huawei.media.video.CaptureCapabilityAndroid { *;}
-keep class com.huawei.media.video.DeviceInfo { *;}
-keep class com.huawei.media.video.H264Decoder { *;}
-keep class com.huawei.media.video.HmeDefinitions { *;}
-keep class com.huawei.media.video.JNIBridge { *;}
-keep class com.huawei.media.video.JNIBridgeImpl { *;}
-keep class com.huawei.media.video.JNIMeetingImpl { *;}
-keep class com.huawei.media.video.JNIRtcImpl { *;}
-keep class com.huawei.media.video.KirinMediaCodecEncoder { *;}
-keep class com.huawei.media.video.LogFile { *;}
-keep class com.huawei.media.video.MediaCodecDecoder { *;}
-keep class com.huawei.media.video.MediaCodecEncoder { *;}
-keep class com.huawei.media.video.ScreenCaptureImageActivity { *;}
-keep class com.huawei.media.video.SurfaceEncoder { *;}
-keep class com.huawei.media.video.VideoCapture { *;}
-keep class com.huawei.media.video.VideoCapture2Android { *;}
-keep class com.huawei.media.video.VideoCaptureAndroid { *;}
-keep class com.huawei.media.video.VideoCaptureDeviceInfo { *;}
-keep class com.huawei.media.video.VideoCaptureDeviceInfoAndroid { *;}
-keep class com.huawei.media.video.VideoRender { *;}
-keep class com.huawei.media.video.VideoRenderNoGLES { *;}
-keep class com.huawei.media.video.ViEAndroidGLES20 { *;}
-keep class com.huawei.media.video.ViERenderer { *;}
-keep class com.huawei.media.video.ViESurfaceRenderer { *;}
-keep class com.huawei.media.video.VtNativeDecoder { *;}
-keep class com.huawei.media.video.codec.KirinDecoder { *;}
-keep class com.huawei.media.video.codec.KirinEncoder { *;}
-keep class com.huawei.media.video.codec.** { *;}
-keep class com.huawei.media.video.Camera2VideoCapture { *;}
-keep class com.huawei.media.video.capture.Camera2EncodedImageReader { *;}
-keep class com.huawei.media.video.capture.Camera2TextureImageReader { *;}
-keep class com.huawei.media.video.capture.Camera2VideoCaptureImpl { *;}
-keep class com.huawei.media.video.capture.Camera2YUVImageReader { *;}
-keep class com.huawei.media.video.capture.CaptureRequestExKeyUtils { *;}
-keep class com.huawei.media.video.capture.StreamConfig { *;}
-keep class com.huawei.media.video.capture.StreamResolution { *;}
-keep class com.huawei.media.video.capture.VideoCaptureDevInfoCamera2Impl { *;}
-keep class com.huawei.media.mcuvideo.CaptureCapabilityAndroid { *;}
-keep class com.huawei.media.mcuvideo.HmeDefinitions { *;}
-keep class com.huawei.media.mcuvideo.JNIBridge { *;}
-keep class com.huawei.media.mcuvideo.LogFile { *;}
-keep class com.huawei.media.mcuvideo.MediaCodecDecoder { *;}
-keep class com.huawei.media.mcuvideo.MediaCodecEncoder { *;}
-keep class com.huawei.media.mcuvideo.VideoCapture { *;}
-keep class com.huawei.media.mcuvideo.VideoCaptureDeviceInfo { *;}
-keep class com.huawei.media.mcuvideo.VideoRender { *;}
-keep class com.huawei.media.mcuvideo.VideoRenderNoGLES { *;}
-keep class com.huawei.media.mcuvideo.ViERenderer { *;}
-keep class com.huawei.media.mcuvideo.codec.** { *;}
-keep class com.huawei.media.mcuvideo.capture.Camera2EncodedImageReader { *;}
-keep class com.huawei.media.mcuvideo.capture.Camera2TextureImageReader { *;}
-keep class com.huawei.media.mcuvideo.capture.Camera2VideoCaptureImpl { *;}
-keep class com.huawei.media.mcuvideo.capture.Camera2YUVImageReader { *;}
-keep class com.huawei.media.mcuvideo.capture.CaptureRequestExKeyUtils { *;}
-keep class com.huawei.media.mcuvideo.capture.StreamConfig { *;}
-keep class com.huawei.media.mcuvideo.capture.StreamResolution { *;}
-keep class com.huawei.media.mcuvideo.capture.VideoCaptureDevInfoCamera2Impl { *;}
-keep class com.huawei.media.oldvideo.CaptureCapabilityAndroid { *;}
-keep class com.huawei.media.oldvideo.HmeDefinitions { *;}

```

```

-keep class com.huawei.media.oldvideo.JNIBridge { *;}
-keep class com.huawei.media.oldvideo.LogFile { *;}
-keep class com.huawei.media.oldvideo.MediaCodecDecoder { *;}
-keep class com.huawei.media.oldvideo.MediaCodecEncoder { *;}
-keep class com.huawei.media.oldvideo.VideoCapture { *;}
-keep class com.huawei.media.oldvideo.VideoCaptureDeviceInfo { *;}
-keep class com.huawei.media.oldvideo.VideoRender { *;}
-keep class com.huawei.media.oldvideo.VideoRenderNoGLES { *;}
-keep class com.huawei.media.oldvideo.ViERenderer { *;}
-keep class com.huawei.media.oldvideo.codec.** { *;}

#配置TUP java 不混淆, c++ 会调用
-keep class tupsdk.Tupmedia { *;} #TupCall.jar
-keep class com.huawei.media.data.Conference { *;} #TupConf.jar

-keep class imssdk.** { *;}

# rtc sdk 混淆
-keep class com.huawei.rtc.**{*;}

-keep class com.huawei.allplatform.**{*;}

## native sdk 混淆
-keep class com.huawei.hwmsdk.**{*;}

# WebViewInterface 不能混淆
-keepclassmembers class com.huawei.hwmclink.jsbridge.model.GHConfigModel {
    public *;
}

# API不能被混淆
-keep public class com.huawei.**.*Api { *;}
-keep class com.huawei.cloudlink.openapi.api.ICloudLinkOpenApi{*;}
-keep class com.huawei.cloudlink.openapi.api.impl.CloudLinkOpenApiImpl{*;}
-keep class com.huawei.hwmbiz.IBizOpenApi{*;}
-keep class com.huawei.hwmbiz.impl.BizOpenApiImpl{*;}

#micro service 不能混淆,有反射调用, 参考华为云配置
-keep public class * extends com.huawei.hwmfoundation.microservice.HCMicroService
-keep public class * extends com.mapp.hcmobileframework.microapp.HCMicroApplicationDelegate
-keep public class * implements com.huawei.hwmclink.jsbridge.bridge.** {*;}

#继承自AbsCache 的类, 要保留构造函数, 否则构造函数会被优化掉,这里保留所有的构造函数, 不论有没有在使用
-keepclassmembers class * {
    <init>(...);
}

#RXJava 不能被混淆, 因为内部有hook操作
-keep class io.reactivex.**{*;}

#自定义注解不能被混淆
-keep public class com.huawei.hwmbiz.aspect.CheckToken
-keep public class com.huawei.hwmfoundation.hook.annotation.TimeConsume
-keep public class com.huawei.hwmfoundation.hook.annotation.HookDisable
-keep public class com.huawei.hwmfoundation.hook.annotation.HookNotNeeded

#glide 不混淆
-keep class com.github.bumptech.glide.**{*;}
#需要用json转换成javabean 不混淆
-keep public class * extends com.huawei.hwmfoundation.base.BaseModel{*;}

#HMS SDK不可混淆
-ignorewarnings
-keepattributes *Annotation*
-keepattributes Exceptions

```

```

-keepattributes InnerClasses
-keepattributes Signature
-keepattributes SourceFile,LineNumberTable
-keep class com.hianalytics.android.**{*;}
-keep class com.huawei.updatesdk.**{*;}
-keep class com.huawei.hms.**{*;}

#Androidx 混淆
-keep class com.google.android.material.**{*;}
-keep class androidx.**{*;}
-keep public class * extends androidx.**
-keep interface androidx.**{*;}
-dontwarn com.google.android.material.**
-dontnote com.google.android.material.**
-dontwarn androidx.**
#路由表不能混淆，启动会从固定package下汇总路由表
-keep class com.huawei.cloudlink.router.routermap.**{*;}

# crashreport混淆
-keep public class com.huawei.crashreport.**{*;}

# IMSDK 混淆
-keep class com.huawei.imsdk.ECSProxy{*;}
-keep class com.huawei.imsdk.msg.**{*;}

# HWMUserState用户状态不混淆
-keep public class com.huawei.cloudlink.openapi.model.HWMUserState{*;}

# netty不混淆，投影要用到
-keep class io.netty.**{*;}

```

步骤6 添加权限

在AndroidManifest.xml中添加SDK必要的权限：

```

<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
<uses-permission android:name="android.permission.CAMERA" />
<uses-permission android:name="android.permission.READ_CONTACTS" />
<uses-permission android:name="android.permission.MODIFY_AUDIO_SETTINGS" />
<uses-permission android:name="android.permission.RECORD_AUDIO" />
<uses-permission android:name="android.permission.WAKE_LOCK" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.DISABLE_KEYGUARD" />
<uses-permission android:name="android.permission.SYSTEM_ALERT_WINDOW" />
<uses-permission android:name="android.permission.REQUEST_INSTALL_PACKAGES" />
<uses-permission android:name="com.huawei.android.launcher.permission.CHANGE_BADGE" />
<uses-permission android:name="android.permission.BLUETOOTH" />
<uses-permission android:name="android.permission.VIBRATE" />
<!--会议中会启动前台服务器保活-->
<uses-permission android:name="android.permission.FOREGROUND_SERVICE" />

```

步骤7 添加屏幕共享的服务

如果需要使用会中屏幕共享功能，在AndroidManifest.xml中添加以下服务的申明

```

<!-- 会议sdk 需要增加以下配置，如果是compileSdkVersion >= 29，需要增加属性
android:foregroundServiceType="mediaProjection"，否则去掉该属性-->
<service android:name="com.huawei.hwmconf.sdk.ScreenRecorderService"
    android:exported="false"
    android:enabled="true"/>

```

步骤8 SDK初始化

- 初始化要放在Application的onCreate里，尽量时序提前，以保证sdk内部可以通过监听activity的生命周期的方式来获取当前activity并跳转界面：


```
OpenSDKConfig sdkConfig = new OpenSDKConfig(this)
    .setAppId("fdb8e4699586458bbd10c834872dcc62") //设置唯一App ID, 可填应用英文名, 使用
App ID登录必须要设
    .setServerAddress(DBConfig.Default.getInstance().getServerAddress()) //设置会议服务器地址
    .setServerPort(DBConfig.Default.getInstance().getServerPort()); //设置会议服务器端口
HWMSdk.init(this,sdkConfig);
```

步骤9 接口调用示例。

- 登录接口
HWMSdk.getOpenApi(Application).login(LoginParam, HwmCallback<LoginResult>);
- 创会接口
HWMSdk.getOpenApi(Application).createConf(CreateConfParam, HwmCancelableCallBack<ConfInfo>);
- 入会接口
HWMSdk.getOpenApi(Application).joinConf(JoinConfParam, HwmCancelableCallBack<Void>);
- 呼叫接口
HWMSdk.getOpenApi(Application).startCall(CallParam, HwmCancelableCallBack<Void>);
- 注销接口
HWMSdk.getOpenApi(Application).logout(HwmCallback<LoginCompletedResult>);

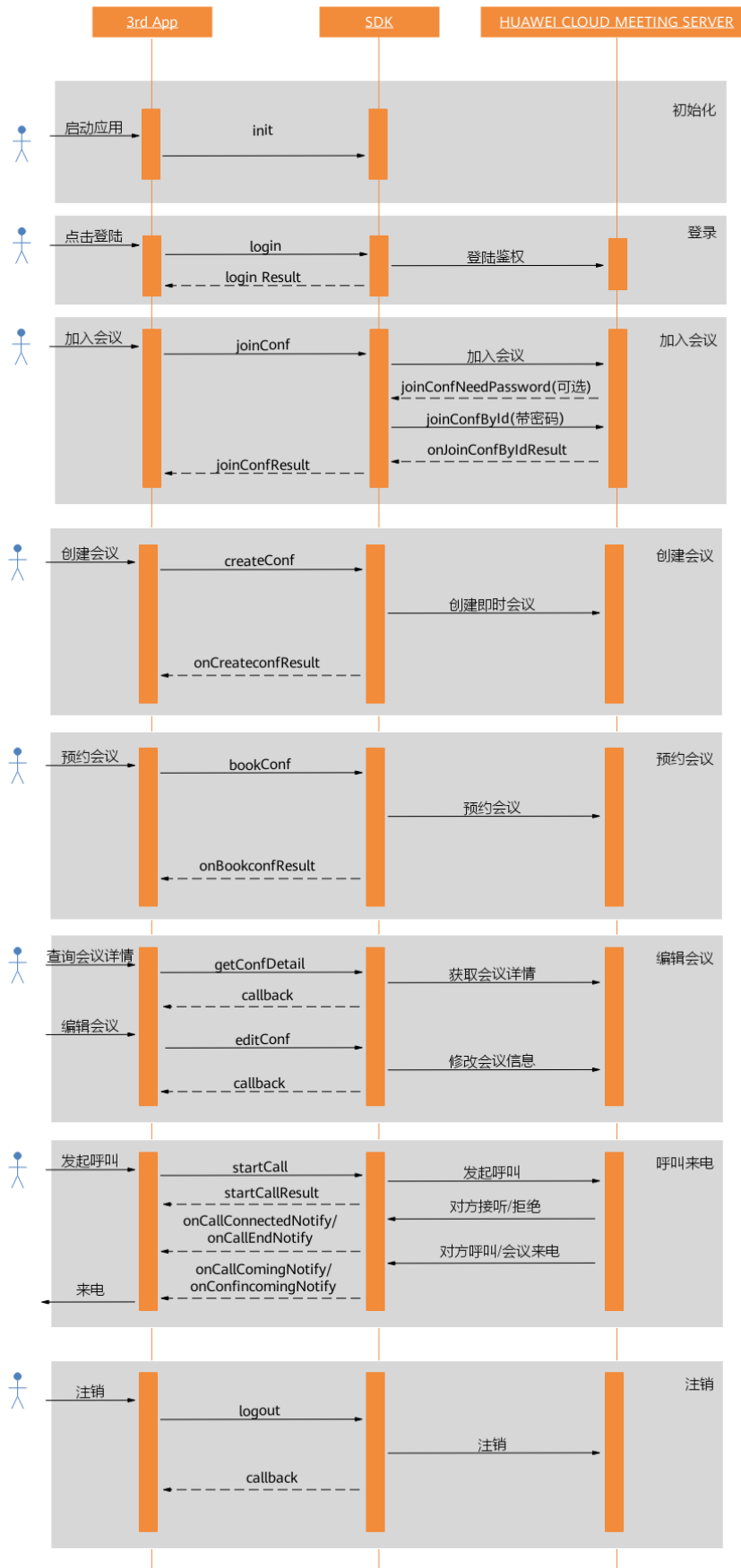
完成上述调用步骤，就可以运行 新建的demo工程。

----结束

5.4 典型场景

5.4.1 典型场景接口调用概览

图 5-10 接口调用流程



📖 说明

- 1、登录不是必须的步骤，在不登录的情况下也能通过会议ID和密码加入已经创建好的会议。
- 2、开启共享、结束共享、离开会议、结束会议等接口，当前SDK的UI上已经提供了对应的功能，如果第三方的App不需要程序控制这些功能则无需调用对应接口。

5.4.2 场景 1：初始化

描述

使用SDK业务组件时，需要先完成SDK的基础组件的初始化。

初始化接口分为两种：1. 同步初始化 2. 异步初始化。初始化过程大约需要1-2秒，若对App启动速度有要求，建议使用异步初始化接口。

1. 同步初始化业务流程

步骤1 接口调用

1. 调用HWMSdk.init(Application, OpenSDKConfig) 初始化SDK。在初始化之前我们可以对SDK进行自定义的相关设置，设置步骤和示例代码请浏览[5.3-8SDK初始化](#)

----结束

2. 异步初始化业务流程

步骤1 注册监听

在Application中注册Activity声明周期的监听：

```
registerActivityLifecycleCallbacks(new ActivityLifecycleHandle());
```

步骤2 接口调用

1. 调用HWMSdk.initAsync(Application, OpenSDKConfig, SdkCallback) 异步初始化SDK。在初始化之前我们可以对SDK进行自定义的相关设置，设置步骤和示例代码请浏览[5.3-8SDK初始化](#)

步骤3 回调处理

1. 监听onSuccess回调，**确保Android SDK初始化完成后再进行其他接口的调用。**

----结束

注意事项

1. 异步初始化时，步骤1 Activity声明周期监听是必做步骤，若忽略会影响SDK的多项功能，**请确保完成步骤1后再调用初始化接口**
2. 异步初始化可放在子线程中进行，**但请确保初始化完成后再使用SDK的其他功能。**
3. 初始化接口的Application参数不能缺省，否则会初始化失败，**如果是App ID 方式登录，请务必通过OpenSDKConfig.setAppId设置。**

5.4.3 场景 2：帐号密码方式登录

描述

在使用云会议服务之前，需要向服务器完成登录鉴权。

📖 说明

在未登录状态下，只能加入已经创建的会议。

业务流程

步骤1 接口调用

1. 创建登录参数LoginParam。
 - a. 选择合适的登录类型。
 - b. 设置帐号密码。
2. 调用login接口进行登录，第1步中的数据作为参数。

步骤2 处理回调

处理回调中的结果。

----结束

示例代码

```
//构建登录对象
LoginParam loginParam = new LoginParam()
    .setLoginAuthType(LoginAuthType.Account_And_Password)
    .setAccount("huawei") //登录帐号
    .setPassword("123456"); //登录密码
//开始登录
HWMSdk.getOpenApi(getApplication()).login(loginParam, new HwmCallback<LoginResult>() {
    @Override
    public void onSuccess(LoginResult loginResult) {
        Log.i(TAG, "登录成功");
    }
    @Override
    public void onFailed(final int retCode, String desc) {
        HwmContext.getInstance().runOnMainThread(new Runnable() {
            @Override
            public void run() {
                Error error = HWMBizSdk.getLoginApi().convertErrorCodeToUI(retCode);
                Log.i(TAG, "登录失败：" + error.getMessage());
            }
        });
    }
});
```

注意事项

必须在HWMSdk.init调用成功后才可以调用该方法登录。


```
public void onSuccess(LoginResult loginResult) {
    dismissLoading();
    dismiss();
    if (loginResult != null) {
        DemoUtil.showToast("登录成功" + loginResult.getUserUid());
    }else{
        DemoUtil.showToast("已登录");
    }
}

@Override
public void onFailed(int retCode, String desc) {
    HwmContext.getInstance().runOnMainThread(() ->{
        dismissLoading();
        Error error = HwMBizSdk.getLoginApi().convertErrorCodeToUI(retCode);
        DemoUtil.showToast("登录错误: " + getLoginErrTips(error));
        dismiss();
    });
}
});
```

注意事项

必须在HWMSdk.init调用成功后才可以调用该方法登录。

5.4.5 场景 4： 预约会议

描述

在使用华为云会议帐号登录后，可以调用预约会议接口预约会议。除了填写预约会议所需的会议参数外，还可以添加与会者。

业务流程

使用SDK预约会议前，处理用户界面输入的预约会议信息，调用bookConf接口，然后处理回调hwmCallback和消息通知BizNotificationHandler的onConfListChanged。

步骤1 接口调用

1. 组装数据结构BookConfParam。
2. 调用bookConf开始创建预约会议，第1步中的数据作为参数。

步骤2 处理回调函数

处理回调函数hwmCallback。

步骤3 处理消息通知

处理消息通知onConfListChanged。

----结束

示例代码

```
// 预约会议结果回调
HwmCallback<ConfInfo> completeHandler = new HwmCallback<ConfInfo>() {
    @Override
    public void onSuccess(ConfInfo result) {
        dismissLoading();
        DemoUtil.showToast("预约会议成功");
    }
}
```

```

@Override
public void onFailed(int retCode, String desc) {
    dismissLoading();
    dismiss();
    String err = ErrorMessageFactory.create(Utils.getApp(), retCode);
    if (TextUtils.isEmpty(err)) {
        err = Utils.getApp().getString(R.string.hwmconf_book_conf_fail);
    }
    DemoUtil.showToast("预约会议失败: " + retCode + ", desc:" + err);
}

boolean needWithMember = true;
List<AttendeeModel> attendeeInfos = new ArrayList<>();
if (needWithMember) {
    //以下三种方式可选
    String name = "huawei123456";
    String number = "+99123456xxx";
    String thirdAccountId = "123456abcd";
    if (!TextUtils.isEmpty(thirdAccountId)) {
        attendeeInfos.add(AttendeeModel.buildAttendeeByThirdAccountId(thirdAccountId, name));
    } else if (number.startsWith("+99")) {
        //sip 号码入会
        attendeeInfos.add(AttendeeModel.buildAttendeeBySipNumber(number, name));
    } else {
        //电话号码入会（手机和座机）
        attendeeInfos.add(AttendeeModel.buildAttendeeByPhone(number, name));
    }
}
BookConfParam bookConfParam = new BookConfParam()
    .setConfSubject("我的会议")
    .setStartTime(1599574798)
    .setTimeZone(56)
    .setDuration(60)
    .setConfType(MeetingType.CONF_VIDEO)
    .setVmrlidFlag(false)
    .setVmrlid("")
    .setNeedConfPwd(true)
    .setJoinConfRestrictionType(JoinConfPermissionType.PERMIT_EVERYONE)
    .setRecordOn(true)
    .setMailOn(true)
    .setSmsOn(true)
    .setEmailCalenderOn(true)
    .setAttendees(attendeeInfos);
HwMBizSdk.getBizOpenApi().bookConf(bookConfParam, this.completeHandler);

```

5.4.6 场景 5：编辑会议

描述

在华为云会议帐号登录后，调用[会议详情](#)接口获取会议详情后调用接口编辑会议。

业务流程

使用SDK发起呼叫时，先调用editConf接口，然后处理回调hwmCallback。

步骤1 接口调用

调用editConf编辑会议。

步骤2 处理回调

处理回调hwmCallback。

----结束

示例代码

```
// 编辑会议参数组装
//会议主题
String subject = "huawei123456的会议";
//会议开始时间、时区、会议时长
String startTimeLocal = startTimeText.getText().toString();
String startTime = BaseDateUtil.transTimeZone(startTimeLocal, TimeZone.getTimeZone("GMT+08:00"),
TimeZone.getTimeZone("GMT+00:00"), BaseDateUtil.FMT_YMDHMS);
long startTimeStamp = BaseDateUtil.dateToTimeStamp(startTime, BaseDateUtil.FMT_YMDHMS);
String timeZomeStr = timezomeText.getText().toString();
String timedurStr = timeDuartionText.getText().toString();
int timeZone = Integer.parseInt(timeZomeStr);
int timeDuration = Integer.parseInt(timedurStr);
//会议类型
MeetingType confType = audioBtn.isChecked() ? MeetingType.CONF_AUDIO : MeetingType.CONF_VIDEO;
String joinConfTypeStr = joinConfPerText.getText().toString();
//入会范围限制
int joinConfPermission = Integer.parseInt(joinConfTypeStr);
JoinConfPermissionType permissionType = JoinConfPermissionType.valueOf(joinConfPermission);
//是否录制
boolean needRecord = recordSwitch.isChecked();
//是否需要邮件通知
boolean needEmail = sendEmailSwitch.isChecked();
//是否需要短信通知
boolean needSms = sendSmsSwitch.isChecked();
//是否需要发邮件日历
boolean needEmailCalendar = sendEmailCalendarSwitch.isChecked();
//与会者列表
List<AttendeeModel> attendeeInfos = this.confDetailModel.getAttendeeModels();
Switch needWithMemberSwitch = findViewById(R.id.edit_switch_with_attend);
boolean needWithMember = needWithMemberSwitch.isChecked();
if (needWithMember) {
    //以下三种方式可选
    String name = "huawei123456";
    String number = "+99123456xxx";
    String thirdAccount = "123456abcd";
    if (!TextUtils.isEmpty(thirdAccount)) {
        if (LoginStatusCache.getLoginAccountInfo() instanceof AppIdAuthParam) {
            //如果是appid的方式登录
            attendeeList.add(AttendeeModel.buildAttendeeByThirdAccountId(thirdAccount,name));
        }
    } else if (number.startsWith("+99")){
        //sip 号码入会
        attendeeList.add(AttendeeModel.buildAttendeeBySipNumber(number, name));
    } else {
        //电话号码入会（手机和座机）
        attendeeList.add(AttendeeModel.buildAttendeeByPhone(number, name));
    }
}
}

EditConfParam editConfParam = new EditConfParam()
    .setConfId(this.confDetailModel.getConfId())
    .setVmridFlag(1 == this.confDetailModel.getVmridFlag())
    .setVmrid(this.confDetailModel.getVmrid())
    .setConfSubject(subject)
    .setStartTime(startTimeStamp)
    .setTimeZone(timeZone)
    .setDuration(timeDuration)
    .setConfType(confType)
    .setNeedConfPwd(!this.confDetailModel.getIsGuestFreePwd())
    .setJoinConfRestrictionType(permissionType)
    .setRecordOn(needRecord)
    .setMailOn(needEmail)
    .setSmsOn(needSms)
    .setEmailCalenderOn(needEmailCalendar)
    .setAttendees(attendeeInfos);
// 编辑会议接口调用
HWMBizSdk.getBizOpenApi().editConf(editConfParam, new HwmCallback<Integer>() {
```

```

@Override
public void onSuccess(Integer integer) {
    dismissLoading();
    DemoUtil.showToast("编辑会议成功: " + integer);
}

@Override
public void onFailed(int retCode, String desc) {
    dismissLoading();
    String err = ErrorMessageFactory.create(Utils.getApp(), retCode);
    if (TextUtils.isEmpty(err)) {
        err = Utils.getApp().getString(R.string.hwmconf_edit_failed);
    }
    DemoUtil.showToast("编辑会议失败: " + retCode + ", desc: " + err);
}
});

```

5.4.7 场景 6：创建会议

描述

创建会议功能会是指创建一个即时会议并立即加入会议，可以通过传参指定创建音频会议还是视频会议，可以指定会议是否需要密码。

业务流程

步骤1 准备创建会议参数

准备创建会议参数，类型为CreateConfParam，定义详见CreateConfParam的接口定义，共需要以下几个参数：会议主题、会议类型、是否需要密码、成员列表（携带与会者时传入）、是否打开摄像头（默认关闭）、是否打开麦克风（默认关闭）、云会议室id（使用时传入）、设置允许入会范围（必填：0-所有人，2-企业内，3-会议邀请人员）。

```

//创建会议对象
CreateConfParam createConfParam = new CreateConfParam()
    .setSubject("华为的会议") //会议名称
    .setConfType(ConfType.CONF_AUDIO) //会议类型，详情请见表2 CreateConfParam参数说明
    .setVmrid(2c9183e07562e60b0175647ea92f59b3) //云会议室id（非必填字段）
    .setJoinConfRestrictionType(JoinConfPermissionType.PERMIT_EVERYONE) //设置允许入会范围限制
    .setCameraOn(true) //摄像头是否开启
    .setMicOn(true) //麦克风是否开启
    .setRecordOn(true) //会议录制功能是否开启
    .setNeedPassword(true); //会议是否需要密码

```

步骤2 是否携带与会者（可选）

如果需要携带与会者创建会议，给步骤1生成的创建会议参数中members赋值一个携带与会者的数组，如果不需要携带与会者可以不传入。

```

List<AttendeeModel> attendeeList = new ArrayList<>();
//以下三种方式可选
String name = "huawei123456";
String number = "+99123456xxx";
String thirdAccount = "123456abcd";
if (!TextUtils.isEmpty(thirdAccount)) {
    if (LoginStatusCache.getLoginAccountInfo() instanceof AppIdAuthParam) {
        //如果是appid的方式登录
        attendeeList.add(AttendeeModel.buildAttendeeByThirdAccountId(thirdAccount,name));
    }
} else if(number.startsWith("+99")){
    //sip 号码入会
    attendeeList.add(AttendeeModel.buildAttendeeBySipNumber(number, name));
} else {
    //电话号码入会（手机和座机）
    attendeeList.add(AttendeeModel.buildAttendeeByPhone(number, name));
}

```

```

    }
//创建会议对象
CreateConfParam createConfParam = new CreateConfParam()
    .setSubject("华为的会议") //会议名称
    .setConfType(ConfType.CONF_AUDIO) //会议类型, 详情请见表2 CreateConfParam参数说明
    .setVmrid(2c9183e07562e60b0175647ea92f59b3) // 云会议室id ( 非必传字段 )
    .setJoinConfRestrictionType(JoinConfPermissionType.PERMIT_EVERYONE) // 设置允许入会范围限制
    .setCameraOn(true) //摄像头是否开启
    .setMicOn(true) //麦克风是否开启
    .setRecordOn(true) //会议录制功能是否开启
    .setNeedPassword(true) //会议是否需要密码
    .setAttendeeMembers(attendeeInfos); //添加与会人信息

```

步骤3 调用接口

使用步骤1和步骤2生成的创建会议参数，并根据返回的回调对执行的结果进行判断和处理，返回回调中onSuccess表示成功，onFailed表示失败，onCancel表示没有网络等取消创建，ErrorMessageFactory.create可以将场景错误码转成国际化的提示。

----结束

示例代码

```

//开始创建会议
HWMSdk.getOpenApi(getActivity()).createConf(createConfParam, new HwmCancelableCallBack
<ConfInfo>() {
    @Override
    public void onSuccess(ConfInfo confInfo) {
        //创建成功 confInfo 会议详情
        //会议id: confInfo.getConfId();
        //会议密码: confInfo.getConfPwd();
    }
    @Override
    public void onFailed(int retCode, String desc) {
        //创建失败
        String err = ErrorMessageFactory.create(Utils.getApp(), retCode);
        if (TextUtils.isEmpty(err)) {
            err =
Utils.getApp().getString(com.huawei.hwmmobileconfui.R.string.conf_create_error);
        }
        Log.i(TAG, "创建会议失败: " + err);
    }
    @Override
    public void onCancel() {
        Log.i(TAG, "取消创建会议");
    }
});

```

注意事项

创建会议的前提是已完成初始化和登录。

5.4.8 场景 7： 加入会议

描述

使用该接口可以在登录和未登录状态下通过会议ID和会议密码加入会议。

业务流程

步骤1 接口调用

1. 创建入会参数JoinConfParam。
 - a. 会议id加入会议使用PasswordJoinConfParam。
 - b. 会议random加入会议使用RandomJoinConfParam。

```
//加入会议对象
PasswordJoinConfParam passwordJoinConfParam = new PasswordJoinConfParam()
    .setConfId("987654321") //加入会议ID
    .setNickname("huawei123456") //设置昵称
    .setCameraOn(true) //是否开启摄像头
    .setMicOn(true); //是否开启麦克风
    .setPassword(true) //入会密码
```

2. 调用joinConf接口加入会议，第1步中的数据作为参数。

步骤2 处理回调

处理回调中的结果。

----结束

示例代码

```
//开始加入会议
HWMSdk.getOpenApi(getActivity()).joinConf(passwordJoinConfParam, new
HwmCancelableCallBack<Void>() {
    @Override
    public void onSuccess(Void ret) {
        Log.i(TAG, "加入会议成功");
    }
    @Override
    public void onFailed(int retCode, String desc) {
        String err = ErrorMessageFactory.create(Utils.getApp(), retCode);
        if (TextUtils.isEmpty(err)) {
            err = Utils.getApp().getString(com.huawei.hwmmobileconfui.R.string.conf_join_fail_tip);
        }
        Log.i(TAG, "加入会议失败: " + err);
    }
    @Override
    public void onCancel() {
        Log.i(TAG, "加入会议取消");
    }
});
```

注意事项

在未登录状态下调用该接口时，会议昵称参数必选。

5.4.9 场景 8：发起呼叫

描述

云会议服务支持用户发起点对点音频呼叫和视频呼叫，呼叫前提是已登录成功。

业务流程

步骤1 接口调用

1. 创建呼叫参数CallParam。

```
//呼叫对象
CallParam callParam = new CallParam()
    .setVideo(true); //是否视频
```

```
String name = "huawei123456";
String number = "+99123456xxx";
String thirdAccountId = "123456abcd";
String accountUuid = "uuidid123456";

//以下四种方式可选
if (!TextUtils.isEmpty(number)) {
    //通过sip号码的方式进行呼叫
    callParam.setNumber(number);
}

if (!TextUtils.isEmpty(account)) {
    //通过华为企业通讯录帐号的方式进行呼叫
    callParam.setAccount(account);
}

if (!TextUtils.isEmpty(thirdAccountId)) {
    //如果是appid的方式登录，可以通过第三方通讯录的联系人id进行呼叫
    callParam.setThirdAccountId(thirdAccountId);
}

if (!TextUtils.isEmpty(accountUuid)) {
    //通过华为企业通讯录id的方式进行呼叫
    callParam.setCalleeUuid(accountUuid);
}
```

2. 调用startCall接口发起呼叫，第1步中的数据作为参数。

步骤2 处理回调

处理回调中的结果。

----结束

示例代码

```
//开始发起呼叫
HWMSdk.getOpenApi(getActivity()).startCall(callParam, new HwmCancelableCallBack<Void>() {
    @Override
    public void onSuccess(Void result) {
        Log.i(TAG, "呼叫成功");
    }
    @Override
    public void onFailed(int retCode, String desc) {
        Log.i(TAG, "呼叫失败: "+ retCode + ", desc: " + desc);
    }
    @Override
    public void onCancel() {
    }
});
```

5.4.10 场景 9： 注销

描述

在使用过程中，可以调用该接口完成已登录帐号的注销。

业务流程

步骤1 接口调用

1. 调用logout接口进行注销。

步骤2 处理回调

处理回调中的结果。

----结束

示例代码

```
HWMSdk.getOpenApi(getActivity()).logout(new HwmCallback<LoginCompletedResult>() {
    @Override
    public void onSuccess(LoginCompletedResult result) {
        DemoUtil.showToast(getContext(), "登出成功");
    }
    @Override
    public void onFailed(int retCode, String desc) {
        DemoUtil.showToast(getContext(), "登出失败 : " + retCode + " ; " + desc);
    }
});
```

5.5 接口参考

5.5.1 基础配置

5.5.1.1 同步初始化

Init

同步初始化

```
HWMSdk.init(Application,OpenSDKConfig)
```

接口描述

该接口用于初始化SDK，进行一些环境准备工作，调用其他接口之前必须先调用这个接口，只需要调用一次，必须放在Application的onCreate里，尽量提前，sdk内部通过监听activity的生命获取当前activity，跳转界面，否则可能出现界面无法跳转的情况。

注意事项

1. 初始化接口耗时约1-2秒，若对初始化速度有要求，请使用 [异步初始化](#)
2. 请传入正确的Application和OpenSDKConfig对象。

方法定义

```
/**
 * 初始化接口
 * @param application 应用application
 * @param sdkConfig sdk配置对象
 */
void init(@NonNull Application application, OpenSDKConfig sdkConfig);
```

参数描述

表 5-2 参数说明

参数	是否必须	类型	描述
application	是	Application	当前程序的Application对象。
sdkConfig	是	OpenSDKConfig	对SDK配置好参数的对象。

表 5-3 初始化信息 OpenSDKConfig

参数	是否必须	类型	描述
appId	是	String	App ID。App ID需要开发者自己申请，请参考《开发指南》中的“ App ID鉴权介绍 ”章节。
serverAddresses	否	String	服务器地址配置
serverPort	否	String	服务器端口配置
needConfChat	否	Boolean	是否需要会中聊天功能
logKeepDays	否	Integer	配置项-日志保存天数，输入有效范围为3-30，默认值0。0为不按天数删除日志。 注意：输入1和2会转换为3，大于30的值会被转换为30。
isDisableIncomingNotification	否	Boolean	是否禁用来电时自动弹出的通知

示例代码

```
OpenSDKConfig sdkConfig = new OpenSDKConfig(this)
    .setAppId("fdb8e4699586458bbd10c834872dcc62")
//App ID需要开发者自己申请，申请流程见 App ID申请
HWMSdk.init(this,sdkConfig);
```

5.5.1.2 异步初始化

InitAsync

初始化

```
HWMSdk.initAsync(Application,OpenSDKConfig, SdkCallback)
```

接口描述

该接口用于异步初始化SDK，进行一些环境准备工作，调用其他接口之前必须先完成异步初始化并收到onSuccess回调。此接口只需要调用一次，调用前必须先注册Activity声明周期的监听，注册行为必须放在Application的onCreate里，主要目的是让sdk内部可以通过监听activity的生命周期获取当前activity，跳转界面，否则可能出现界面无法跳转的情况。

注意事项

1. 对App初始化速度有要求的建议使用异步初始化接口。
2. 请确保初始化完成后再使用SDK的其他功能。
3. 请参照示例代码，在调用异步初始化接口之前先在Application的onCreate里注册Activity生命周期的监听。

方法定义

```
/**
 * sdk 异步初始化接口，确保使用UISDK任何其他功能前已经初始化成功
 *
 * @param application 传入的aplition，内部转化为applicationContext，用户获取系统资源
 * @param sdkConfig sdk配置相关
 * @param sdkCallback sdk回调
 */
void initAsync(@NonNull Application application, OpenSDKConfig sdkConfig, SdkCallback<Integer> sdkCallback)
```

参数描述

表 5-4 参数说明

参数	是否必须	类型	描述
application	是	Application	当前程序的Application对象。
sdkConfig	是	OpenSDKConfig	对SDK配置好参数的对象。
SdkCallback	是	SdkCallback<Integer>	异步初始化结果回调。

示例代码

```
// 【重要】注册Activity生命周期的监听，必须要在Application的onCreate里注册
registerActivityLifecycleCallbacks(new ActivityLifecycleHandle());

// 演示子线程初始化
new Thread(() -> HWMSdk.initAsync(mApp, sdkConfig, new SdkCallback<Integer>() {
    @Override
    public void onFailed(SDKERR sdkerr) {
        HLog.e(TAG, "init failed:" + sdkerr.getDescription());
    }

    @Override
    public void onSuccess(Integer integer) {
        HLog.i(TAG, "init success:" + integer);
        // 继续其他SDK相关逻辑
        mHandler.sendMessage(0);
    }
})).start();
```


5.5.1.3 设置接入站点

在OpenSDKConfig中配置自定义的服务器接入站点（亚太站、中国站等）

默认站点是中国站

使用场景

非中国区用户可以通过配置成亚太站点就近接入

注意事项

SDK初始化的时候必须传入这个配置sdkConfig才会生效

示例代码

```
public class MSiteTypeStrategy implements ISiteTypeStrategy {
    @Override
    public SiteType getSiteType() {
        return SiteType.SITE_TYPE_HEC_AP;
    }
}
sdkConfig.setSiteTypeStrategy(new MSiteTypeStrategy());
```

表 5-5 SiteType 参数说明

枚举值	描述
SiteType.SITE_TYPE_HEC_CN	中国站
SiteType.SITE_TYPE_HEC_AP	亚太站点

📖 说明

- 可以自定义默认接入站点：中国站、亚太站点；选择不同站点，默认登录服务器地址不同。
- 未设置接入站点，默认接入中国站。

5.5.1.4 自定义会中分享面板

在OpenSDKConfig中配置会中自定义的分享面板

分享效果参考图1

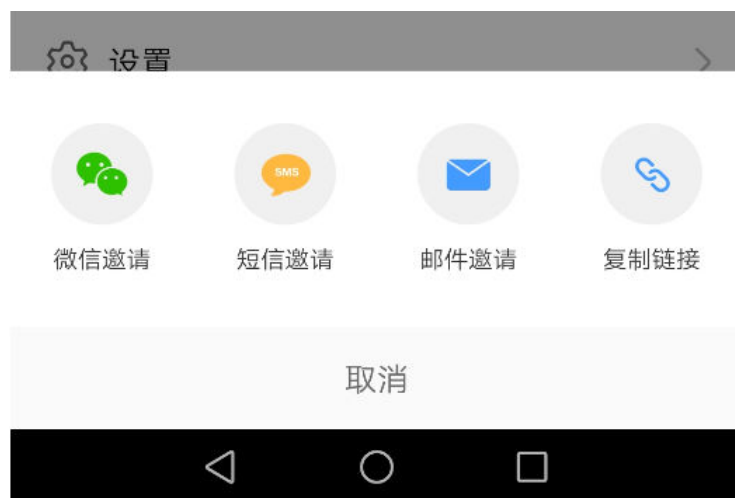
使用场景

在与会者列表页面，点击分享按钮的时候会弹出分享面板，用户可以自定义分享面板的样式和内容

注意事项

1.SDK初始化的时候必须传入这个配置sdkConfig才会生效

图 5-11 自定义分享面板



示例代码

```
//自定义分享菜单
public class CustomShareHandle implements IShareHandle {
    @Override
    public List<IShareModel> buildShareItems(Context context, ConfInfo confInfo, ShareTypeEnum
shareTypeEnum) {
        String subject = confInfo.getConfSubject();
        String guestUrl = confInfo.getConfGuestUri();
        String time = DateUtil.getDateTimeStringForUi(confInfo.getConfStartTime(),
confInfo.getConfEndTime());
        String chairman = confInfo.getConfScheduserName();
        String confId = confInfo.getConfId();
        String guestPwd = TextUtils.isEmpty(confInfo.getConfPwd()) ?
Utils.getApp().getString(com.huawei.hwmmobileconfui.R.string.conf_no_pwd) : confInfo.getConfPwd();
        String content =
String.format(Utils.getApp().getString(com.huawei.hwmmobileconfui.R.string.conf_share_content), subject,
time, chairman, confId, guestPwd, guestUrl);
        //短信分享菜单
        SmsShareModel smsShareModel = new SmsShareModel();
        smsShareModel.setContent(content);
        //复制链接
        CopyShareModel copyShareModel = new CopyShareModel();
        copyShareModel.setContent(content);
        List<IShareModel> shareModels = new ArrayList<>();
        shareModels.add(smsShareModel);
        shareModels.add(copyShareModel);
        return shareModels;
    }
}
sdkConfig.setShareHandle(new CustomShareHandle());
```

5.5.1.5 自定义日志保存路径

在OpenSDKConfig中配置自定义的SDK日志保存的路径

使用场景

SDK默认的日志路径是/sdcard/Android/data/应用的包名/files/log

用户可以根据自己需要修改日志的存放路径

注意事项

- 1.自定义日志保存路径后，取日志也确保使用相应的路径，取日志参考[获取Android SDK的日志](#)
- 2.SDK初始化的时候必须传入这个配置sdkConfig才会生效

示例代码

```
//自定义日志保存路径
public class CustomLoggerHandle implements ILoggerHandle {
    private Application application;
    public CustomLoggerHandle(Application application) {
        this.application = application;
    }
    //自定义日志保存地址
    @Override
    public String getLogPath() {
        return FileUtil.getExternalFilesDir(this.application) + "/log/meeting/";
    }
}
sdkConfig.setLoggerHandle(new CustomLoggerHandle());
```

5.5.1.6 自定义开关会中聊天功能

在OpenSDKConfig中配置是否会中开启聊天功能

开启聊天功能，会中菜单有聊天功能的按钮显示[图1](#)

关闭聊天功能，会中菜单没有聊天功能的按钮[图2](#)

使用场景

应用中不需要会中聊天的功能，可以屏蔽对应的功能

注意事项

- 1.SDK初始化的时候必须传入这个配置sdkConfig才会生效

图 5-12 会中聊天开启



图 5-13 会中聊天关闭



示例代码

```
OpenSDKConfig sdkConfig = new OpenSDKConfig(Context)
    .setNeedConfChat(false)
```

5.5.1.7 自定义是否开启会议高清优先

在OpenSDKConfig中配置是否开启会议高清优先的功能

开启高清优先后

1. 设为开启高清视频后，可以本端发送高清视频
2. 设为开启高清视频后，可以选看高清视频
3. 不根据手机性能等因素降低发送的分辨率

使用场景

对会议的视频高清程度有要求的场景

注意事项

1. SDK初始化的时候必须传入这个配置sdkConfig才会生效

📖 说明

会前设置是否开启优先传输高清画质，开启后就会忽略网络及手机性能等因素，优先传输高清画质

示例代码

```
OpenSDKConfig sdkConfig = new OpenSDKConfig(Context)
    .setHighResolutionFirst(true)
```

5.5.1.8 自定义会议接听时摄像头/麦克风开关

在OpenSDKConfig中配置是否在接听会议的时候自动打开摄像头或者麦克风入会

接听会议的时候sdk通过该回调接口获取摄像头和麦克风是否打开在入会的时候设置是否开启摄像头麦克风

使用场景

一般应用设置中有是否开启摄像头麦克风的用户设置项，通过自定义该回调接口即可实现会议接听时候的摄像头麦克风的开启情况和用户的设置项保持一致

注意事项

- 1.摄像头的开启需要先有权限，没有权限的情况下无法开启摄像头
- 2.SDK初始化的时候必须传入这个配置sdkConfig才会生效

示例代码

```
public class MyDeviceStrategyWhenConfAccept implements IDeviceStrategyWhenConfAccept {
    @Override
    public boolean isTurnOnCameraWhenConfAccept() {
        //自定义返回是否开关摄像头
        return false;
    }

    @Override
    public boolean isTurnOnMicWhenConfAccept() {
        //自定义返回是否开关麦克风
        return false;
    }
}
sdkConfig.setDeviceStrategyWhenConfAccept(new MyDeviceStrategyWhenConfAccept());
```

5.5.1.9 自定义设置来电响铃和震动

```
HWMSdk.getOpenApi.changeInComingNotice(boolean isVibrate, boolean isRing, HwmCallback<Boolean> callback);
```

接口描述

该接口用于修改来电(呼叫/会议)通知方式。

注意事项

该接口需要在登录之后收到来电之前设置才能生效。

方法定义

```
/**
 * 修改来电(呼叫/会议)通知方式
 * @param isVibrate 是否开振动
 * @param isRing 是否开启铃声
 */
void changeInComingNotice(boolean isVibrate, boolean isRing, HwmCallback<Boolean> callback);
```

参数描述

表 5-6 参数说明

参数	是否必须	类型	描述
isVibrate	是	boolean	是否开启振动
isRing	是	boolean	是否开启铃声
callback	是	HwmCallback	设置来电通知方式结果回调

返回值

无。

示例代码

```
HWMSdk.getOpenApi(HWMBizSdk.getApplication()).changeInComingNotice(isOpenVibrate, isOpenRing,
new HwmCallback<Boolean>() {
    @Override
    public void onFailed(int retCode, String desc) {
        dismissLoading();
        dismiss();
        if (!TextUtils.isEmpty(desc)) {
            DemoUtil.showToast(desc);
        } else {
            DemoUtil.showToast("修改来电通知方式失败");
        }
    }

    @Override
    public void onSuccess(Boolean aBoolean) {
        dismissLoading();
        dismiss();
    }
});
```

5.5.1.10 自定义隐藏外部标签

默认情况下与会者列表中外部企业人员、匿名入会人员将呈现”外部“标签
在OpenSDKConfig中可配置是否隐藏外部标签

使用场景

对于不想展示“外部”标签的应用，可以自定义隐藏

注意事项

1.SDK初始化的时候必须传入这个配置sdkConfig才会生效

示例代码

```
//自定义隐藏外部标签
sdkConfig.setHideExternalLabelHandle(new IHideExternalLabelHandle() {
    @Override
    public boolean onHideExternalLabelEvent() {
        // true隐藏标签, false显示标签
        return DemoUtil.needCustom(NeedCustomType.HIDE_EXTERNAL_LABEL);
    }
});

ConfUI.setHideExternalLabelHandle(sdkConfig.getHideExternalLabelHandle());
```

5.5.1.11 禁用提示音

DisableVoicePrompts

接口描述

禁用提示音。

注意事项

1. 该接口非必需调用，如不调用，则默认播放会中提示音。
2. 该接口为同步接口，返回值以错误码形式返回

方法定义

```
SDKERR disableVoicePrompts(VoicePrompts voicePrompts);
```

参数描述

表 5-7 参数说明

参数	是否必须	类型	描述
voicePrompts	是	VoicePrompts	语音提示信息。

表 5-8 结构体 VoicePrompts 参数说明

参数	是否必须	类型	描述
disableMutePrompt	是	boolean	是否禁用静音提示。

返回值

示例代码

```
VoicePrompts voicePrompts = new VoicePrompts();
voicePrompts.setDisableMutePrompt(true);
SDKERR result = HWMBizSdk.getBizOpenApi().disableVoicePrompts(voicePrompts);
Log.i(TAG, "result code:" + result.getValue() + " desc:" + result.getDescription());
```

5.5.1.12 网络模式设置

setNetworkMode

接口描述

如华为云会议的媒体节点下沉部署到企业，并且管理节点也是通过企业内的代理访问，需要将网络模式设置成“企业内外接入”。

📖 说明

企业内的App无法直接访问公网（即媒体节点下沉和管理节点代理下沉场景）的情况下才需要调用该接口。

注意事项

1. 该接口非必需调用，若不设置，则使用默认模式。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
IHwmNetwork.getInstance().setNetworkMode(networkInfo)
```

参数描述

表 5-9 NetWorkModelInfo 说明

参数	是否必须	类型	描述
networkInfo	是	NetWorkModelInfo	网络模式信息

表 5-10 结构体 NetWorkModelInfo 参数说明

参数	是否必须	类型	描述
accessType	是	NetworkAccessType	网络接入类型

表 5-11 枚举 NetworkAccessType 参数说明

值	描述
ACCESS_TYPE_AUTO	默认值，目前和企业外网接入模式一致。
ACCESS_TYPE_INNER	企业内网接入，用于媒体和信令下沉场景，app从内网接入，信令/媒体消息走代理服务器才能访问云会议服务端。
ACCESS_TYPE_OUTER	企业外网接入，用于非媒体和信令下沉场景，互联网接入，信令/媒体信息可以直接访问云会议服务端

表 5-12 返回值

值	描述
int	成功返回0，其他值表示失败。

示例代码

```
NetWorkModelInfo netWorkModelInfo = new NetWorkModelInfo();
netWorkModelInfo.setAccessType(NetworkAccessType.enumOf(NetworkAccessType.ACCESS_TYPE_AUTO));
NativeSDK.getNetworkApi().setNetworkMode(netWorkModelInfo);
```

5.5.1.13 开启功能

enableFeature

接口描述

用于开启特性支持。

注意事项

1. 默认是开启分组会议，若不需要分组会议功能，再调用此接口。
2. 先创建初始化对象（HWMOpenSDKConfig），再调用该接口，最后初始化。

方法定义

```
void enableFeature(HWMEnableFeatureType confCapabilityType, boolean enable)
```

参数描述

表 5-13 枚举 HWMEnableFeatureType 说明

值	描述
HWM_FEATURE_TYPE_BREAKOUT_CONF	分组会议能力
HWM_FEATURE_TYPE_CONFIRM_RECORD	是否显示录制安全提示窗，默认不显示

返回值

无

示例代码

```
// 自定义功能开启或屏蔽
openSDKConfig.enableFeature(HWMEnableFeatureType.HWM_FEATURE_TYPE_CONFIRM_RECORD, true);
```

5.5.1.14 设置语言

setLanguage

接口描述

该接口设置SDK国际化语言。默认是中文。

注意事项

无

方法定义

```
void setLanguage(LanguageType languageType, String language);
```

参数描述

表 5-14 参数说明

参数	是否必须	类型	描述
languageType	是	LanguageType	语言类型枚举。

参数	是否必须	类型	描述
language	是	String	若LanguageType传入的是中文或者英文，则此参数可传空字符串或任意值； 若LanguageType传入的是自定义语言，则需要填对应的语言代码，如“ar”代表阿拉伯语

表 5-15 枚举 LanguageType 说明

枚举值	描述
LANGUAGE_TYPE_CN_ZH	中文。
LANGUAGE_TYPE_EN	英文。
LANGUAGE_TYPE_CUSTOMIZED	自定义语言

返回值

无

示例代码

```
// 设置中文和英文，直接使用枚举
HWMSdk.getOpenApi(this).setLanguage(LanguageType.LANGUAGE_TYPE_EN, "");

// 设置其他语言，添加语言代码，如"ar"代表阿拉伯语，并在主工程的资源文件夹中添加阿拉伯语的翻译
HWMSdk.getOpenApi(this).setLanguage(LanguageType.LANGUAGE_TYPE_CUSTOMIZED, "ar");
```

5.5.2 界面定制

5.5.2.1 自定义底部菜单栏

自定义菜单主要用于实现会中界面底部菜单栏中菜单项的增减和自定义菜单项

底部菜单分四种场景，需要实现四种自定义方法：

- 1.CallAudioToolbar：音频点对点通话底部菜单栏自定义[音频点对点通话自定义菜单](#)
- 2.CallVideoToolbar：视频点对点通话底部菜单栏自定义[视频点对点通话自定义菜单](#)
- 3.ConfAudioToolbar：音频会议底部菜单栏自定义[音频会议自定义菜单](#)
- 4.ConfVideoToolbar：视频会议呼叫底部菜单栏自定义[视频会议自定义菜单](#)

使用场景

需要调整底部菜单栏中菜单项的应用场景

注意事项

- 1.SDK初始化的时候必须传入这个配置sdkConfig才会生效

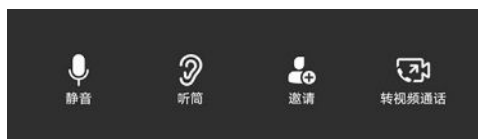
自定义菜单设置

示例代码

```
ToolBarMenuProxy toolBarMenuProxy = new ToolBarMenuProxy();  
toolBarMenuProxy.setCallAudioToolBarHandle(new CustomCallAudioToolBarHandle());  
toolBarMenuProxy.setCallVideoToolBarHandle(new CustomCallVideoToolBarHandle());  
toolBarMenuProxy.setConfAudioToolBarHandle(new CustomConfAudioToolBarHandle());  
toolBarMenuProxy.setConfVideoToolBarHandle(new CustomConfVideoToolBarHandle());  
sdkConfig.setToolBarMenuProxy(toolBarMenuProxy);  
sdkConfig.setParticipantMenuStrategy(new CustomParticipantMenuHandle());
```

音频点对点通话自定义菜单

图 5-14 音频点对点通话自定义菜单



示例代码

```
/**  
 * 演示自定义"音频呼叫"场景下的toolbar菜单  
 */  
public class CustomCallAudioToolBarHandle implements IToolBarMenuStrategy {  
    @Override  
    public List<IConfMenu> buildMenuItems() {  
        List<IConfMenu> confMenus = new ArrayList<>();  
        confMenus.add(new MicMenu());  
        confMenus.add(new SpeakerMenu());  
        if (TupConfig.isNeedScreenShare()) {  
            confMenus.add(new ShareMenu());  
        }  
        confMenus.add(new InviteMenu());  
        confMenus.add(new TransVideoMenu());  
        return confMenus;  
    }  
    @Override  
    public List<IConfMenu> buildMoreMenuItems() {  
        return null;  
    }  
    @Override  
    public List<IConfMenu> buildSettingMenuItems() {  
        return null;  
    }  
}
```

视频点对点通话自定义菜单

图 5-15 视频点对点通话自定义菜单



图 5-16 视频点对点通话更多自定义菜单



图 5-17 视频呼叫会议设置自定义菜单



示例代码

```
/**
 * 演示自定义"视频呼叫"场景下的toolbar菜单
 */
public class CustomCallVideoToolbarHandle implements IToolbarMenuStrategy {
    @Override
    public List<IConfMenu> buildMenuItems() {
        List<IConfMenu> confMenus = new ArrayList<>();
        confMenus.add(new MicMenu());
        confMenus.add(new SpeakerMenu());
        confMenus.add(new CameraMenu());
        if (TupConfig.isNeedScreenShare()) {
            confMenus.add(new ShareMenu());
        }
        confMenus.add(new InviteMenu());
        confMenus.add(new MoreMenu());
        return confMenus;
    }
}
```

```

    }
    @Override
    public List<IConfMenu> buildMoreMenuItems() {
        List<IConfMenu> confMenus = new ArrayList<>();
        confMenus.add(new InviteMoreMenu());
        confMenus.add(new SwitchCameraMenu());
        confMenus.add(new CallVideoToAudioMenu());
        confMenus.add(new CallAudioToVideoMenu());
        confMenus.add(new ConfSettingMenu());
        return confMenus;
    }
    @Override
    public List<IConfMenu> buildSettingMenuItems() {
        List<IConfMenu> menus = new ArrayList<>();

        List<IConfMenu> audioVideomenuChildren = new ArrayList<>();
        IConfMenu beautyMenu = new
        SwitchMenu(com.huawei.hwmmobileconfui.R.id.hwmmconf_confsetting_beauty,
        R.string.hwmmconf_setting_beauty, true);
        IConfMenu pipMenu = new
        SwitchMenu(com.huawei.hwmmobileconfui.R.id.hwmmconf_confsetting_hide_self,
        R.string.hwmmconf_setting_pip, true);
        audioVideomenuChildren.add(beautyMenu);
        audioVideomenuChildren.add(pipMenu);
        MenuContainer audioVideoMenuContainer = new
        MenuContainer(com.huawei.hwmmobileconfui.R.id.hwmmconf_setting_menu_audio_video,
        R.string.hwmmconf_setting_audio_video, audioVideomenuChildren);

        List<IConfMenu> commonMenuChildren = new ArrayList<>();
        //网络检测
        IConfMenu networkDetectMenu = new
        RouterMenu(com.huawei.hwmmobileconfui.R.id.hwmmconf_confsetting_network_detect,
        R.string.hwmmconf_network_check, "cloudlink://hwmeeting/networkdetection");
        commonMenuChildren.add(networkDetectMenu);
        MenuContainer commonMenuContainer = new
        MenuContainer(com.huawei.hwmmobileconfui.R.id.hwmmconf_setting_menu_common,
        R.string.hwmmconf_setting_common, commonMenuChildren);

        menus.add(audioVideoMenuContainer);
        menus.add(commonMenuContainer);
        return menus;
    }
}

```

音频会议自定义菜单

图 5-18 音频会议自定义菜单



图 5-19 音频会议更多自定义菜单



图 5-20 音频会议会议设置菜单



示例代码

```
/**
 * 演示自定义"音频会议"场景下的toolbar菜单
 */
public class CustomConfAudioToolbarHandle implements IToolbarMenuStrategy {
    @Override
    public List<IConfMenu> buildMenuItems() {
        List<IConfMenu> confMenus = new ArrayList<>();
        confMenus.add(new MicMenu());
        confMenus.add(new SpeakerMenu());
        confMenus.add(new ParticipantMenu());
        //【演示】增加一个自定义菜单
        confMenus.add(new CustomMenu());
        confMenus.add(new MoreMenu());
        return confMenus;
    }
    @Override
    public List<IConfMenu> buildMoreMenuItems() {
        List<IConfMenu> confMenus = new ArrayList<>();
        confMenus.add(new ChatMenu());
        confMenus.add(new HandsUpOrDownMenu());
        confMenus.add(new OpenOrCloseRecordMenu());
        confMenus.add(new ConfSettingMenu());
        //【演示】增加一个自定义菜单
        confMenus.add(new CustomMenu());
        return confMenus;
    }
    @Override
    public List<IConfMenu> buildSettingMenuItems() {
        List<IConfMenu> menus = new ArrayList<>();

        List<IConfMenu> audioVideomenuChildren = new ArrayList<>();
        IConfMenu howlDetectMenu = new
SwitchMenu(com.huawei.hwmmobileconfui.R.id.hwmmconf_confsetting_whistle_detection,
R.string.hwmmconf_setting_howling_detect, true);
        audioVideomenuChildren.add(howlDetectMenu);
        MenuContainer audioVideoMenuContainer = new
MenuContainer(com.huawei.hwmmobileconfui.R.id.hwmmconf_setting_menu_audio_video,
R.string.hwmmconf_setting_audio_video, audioVideomenuChildren);
```

```

        List<IConfMenu> securityMenuChildren = new ArrayList<>();
        //锁定会议
        IConfMenu lockConfMenu = new
SwitchMenu(com.huawei.hwmmobileconfui.R.id.hwmmconf_confsetting_lock_meeting,
R.string.hwmmconf_lock, false);
        securityMenuChildren.add(lockConfMenu);
        MenuContainer securityMenuContainer = new
MenuContainer(com.huawei.hwmmobileconfui.R.id.hwmmconf_setting_menu_security,
R.string.hwmmconf_setting_safe, securityMenuChildren);

        List<IConfMenu> attendeePermissionMenuChildren = new ArrayList<>();
        //允许解除静音
        IConfMenu allowReleaseMuteMenu = new
SwitchMenu(com.huawei.hwmmobileconfui.R.id.hwmmconf_confsetting_allow_unmute,
R.string.hwmmconf_allow_ummute_self, false);
        IConfMenu allowShareMuteMenu = new
SwitchMenu(com.huawei.hwmmobileconfui.R.id.hwmmconf_confsetting_allow_sharing,
R.string.hwmmconf_allow_sharing, false);
        attendeePermissionMenuChildren.add(allowReleaseMuteMenu);
        attendeePermissionMenuChildren.add(allowShareMuteMenu);
        MenuContainer attendeePermissionMenuContainer = new
MenuContainer(com.huawei.hwmmobileconfui.R.id.hwmmconf_setting_menu_paticipant_permission,
R.string.hwmmconf_setting_participant_permission, attendeePermissionMenuChildren);

        List<IConfMenu> commonMenuChildren = new ArrayList<>();
        //网络检测
        IConfMenu networkDetectMenu = new
RouterMenu(com.huawei.hwmmobileconfui.R.id.hwmmconf_confsetting_network_detect,
R.string.hwmmconf_network_check, "cloudlink://hwmeeting/networkdetection");
        commonMenuChildren.add(networkDetectMenu);
        MenuContainer commonMenuContainer = new
MenuContainer(com.huawei.hwmmobileconfui.R.id.hwmmconf_setting_menu_common,
R.string.hwmmconf_setting_common, commonMenuChildren);

        menus.add(audioVideoMenuContainer);
        menus.add(securityMenuContainer);
        menus.add(attendeePermissionMenuContainer);
        menus.add(commonMenuContainer);
        return menus;
    }
}

```

视频会议自定义菜单

图 5-21 视频会议自定义菜单



图 5-22 视频会议更多自定义菜单



图 5-23 视频会议会议设置自定义菜单



示例代码

```
/**
 * 演示自定义"视频会议"场景下的toolbar菜单
 */
public class CustomConfVideoToolbarHandle implements IToolbarMenuStrategy {
    @Override
    public List<IConfMenu> buildMenuItems() {
        List<IConfMenu> confMenus = new ArrayList<>();
        confMenus.add(new MicMenu());
        confMenus.add(new CameraMenu());
        confMenus.add(new ParticipantMenu());
        // 【演示】增加一个自定义菜单
        confMenus.add(new CustomMenu());
        confMenus.add(new MoreMenu());
        return confMenus;
    }
    @Override
    public List<IConfMenu> buildMoreMenuItems() {
        List<IConfMenu> confMenus = new ArrayList<>();
        confMenus.add(new ChatMenu());
        confMenus.add(new HandsUpOrDownMenu());
        confMenus.add(new OpenOrCloseRecordMenu());
        confMenus.add(new SwitchCameraMenu());
        confMenus.add(new ConfSettingMenu());
        // 【演示】增加一个自定义菜单
        confMenus.add(new CustomMenu());
        return confMenus;
    }
    @Override
    public List<IConfMenu> buildSettingMenuItems() {
        List<IConfMenu> menus = new ArrayList<>();
```



```

        List<IConfMenu> audioVideomenuChildren = new ArrayList<>();
        IConfMenu howlDetectMenu = new
SwitchMenu(com.huawei.hwmmobileconfui.R.id.hwmmconf_confsetting_whistle_detection,
R.string.hwmmconf_setting_howling_detect, true);
        IConfMenu beautyMenu = new
SwitchMenu(com.huawei.hwmmobileconfui.R.id.hwmmconf_confsetting_beauty,
R.string.hwmmconf_setting_beauty, true);
        IConfMenu pipMenu = new
SwitchMenu(com.huawei.hwmmobileconfui.R.id.hwmmconf_confsetting_hide_self,
R.string.hwmmconf_setting_pip, true);
        audioVideomenuChildren.add(howlDetectMenu);
        audioVideomenuChildren.add(beautyMenu);
        audioVideomenuChildren.add(pipMenu);
        MenuContainer audioVideoMenuContainer = new
MenuContainer(com.huawei.hwmmobileconfui.R.id.hwmmconf_setting_menu_audio_video,
R.string.hwmmconf_setting_audio_video, audioVideomenuChildren);

        List<IConfMenu> securityMenuChildren = new ArrayList<>();
        //锁定会议
        IConfMenu lockConfMenu = new
SwitchMenu(com.huawei.hwmmobileconfui.R.id.hwmmconf_confsetting_lock_meeting,
R.string.hwmmconf_lock, false);
        securityMenuChildren.add(lockConfMenu);
        MenuContainer securityMenuContainer = new
MenuContainer(com.huawei.hwmmobileconfui.R.id.hwmmconf_setting_menu_security,
R.string.hwmmconf_setting_safe, securityMenuChildren);

        List<IConfMenu> attendeePermissionMenuChildren = new ArrayList<>();
        //允许解除静音
        IConfMenu allowReleaseMuteMenu = new
SwitchMenu(com.huawei.hwmmobileconfui.R.id.hwmmconf_confsetting_allow_unmute,
R.string.hwmmconf_allow_ummute_self, false);
        IConfMenu allowShareMuteMenu = new
SwitchMenu(com.huawei.hwmmobileconfui.R.id.hwmmconf_confsetting_allow_sharing,
R.string.hwmmconf_allow_sharing, false);
        attendeePermissionMenuChildren.add(allowReleaseMuteMenu);
        attendeePermissionMenuChildren.add(allowShareMuteMenu);
        /*
        * 【演示】在会议设置里新增一个设置项
        */
        IConfMenu customeSettingMenuTest1 = new SwitchMenu(R.id.conf_setting_menu_test1,
R.string.menu_conf_setting_menu_test1, false);
        attendeePermissionMenuChildren.add(customeSettingMenuTest1);
        MenuContainer attendeePermissionMenuContainer = new
MenuContainer(com.huawei.hwmmobileconfui.R.id.hwmmconf_setting_menu_participant_permission,
R.string.hwmmconf_setting_participant_permission, attendeePermissionMenuChildren);

        List<IConfMenu> commonMenuChildren = new ArrayList<>();
        //网络检测
        IConfMenu networkDetectMenu = new
RouterMenu(com.huawei.hwmmobileconfui.R.id.hwmmconf_confsetting_network_detect,
R.string.hwmmconf_network_check, "cloudlink://hwmeeting/networkdetection");
        commonMenuChildren.add(networkDetectMenu);
        MenuContainer commonMenuContainer = new
MenuContainer(com.huawei.hwmmobileconfui.R.id.hwmmconf_setting_menu_common,
R.string.hwmmconf_setting_common, commonMenuChildren);

        menus.add(audioVideoMenuContainer);
        menus.add(securityMenuContainer);
        menus.add(attendeePermissionMenuContainer);
        menus.add(commonMenuContainer);

        /*
        * 【演示】在会议设置里新增加一个设置项组
        */
    
```

```

List<IConfMenu> customChildren = new ArrayList<>();
//网络检测
IConfMenu customeSettingMenuTest2 = new RouterMenu(R.id.conf_setting_menu_test2,
R.string.menu_conf_setting_menu_test1, "test");
customChildren.add(customeSettingMenuTest2);
MenuContainer customContainer = new MenuContainer(R.id.conf_setting_menu_test_group,
R.string.menu_conf_setting_menu_test_group, customChildren);
menus.add(customContainer);

return menus;
}
}

```

5.5.2.2 自定义与会者列表页面中的菜单

该接口用于自定义与会者列表页面中的菜单，包括下面三种菜单：

1.buildParticipantMoreMenuItems：自定义标题栏右侧菜单项

[图1](#)中区域1位置的菜单

2.buildParticipantItemMenuItems：自定义与会者列表项点击弹出的菜单

点击[图1](#)区域2弹出[图2](#)

3.buildParticipantToolbarMenuItems：自定义与会者页面底部菜单栏

[图1](#)中区域3位置的菜单

4.buildParticipantToolbarMoreMenuItems：自定义点击底部更多菜单项弹出的菜单

点击[图1](#)区域4弹出[图3](#)

使用场景

需要自定义与会者页面四个区域菜单项的场景

注意事项

1.SDK初始化的时候必须传入这个配置sdkConfig才会生效

自定义与会者列表点击菜单设置

示例代码

```

sdkConfig.setParticipantMenuStrategy(new CustomParticipantMenuHandle());

```

图 5-24 与会者列表各个菜单区域图示



与会者列表自定义菜单

图 5-25 与会者列表自定义菜单



图 5-26 与会者列表更多自定义菜单



示例代码

```
public class CustomParticipantMenuHandle implements IParticipantMenuStrategy {
    @Override
    public List<IConfMenu> buildParticipantMoreMenuItems() {
        List<IConfMenu> confMenus = new ArrayList<>();
        confMenus.add(new ParticipantShareMenu());
        confMenus.add(new ContantMenu());
        //【演示】增加一个自定义菜单
        confMenus.add(new CustomMoreMenu());
        return confMenus;
    }
    @Override
    public List<IConfMenu> buildParticipantItemMenuItems() {
        List<IConfMenu> confMenus = new ArrayList<>();
        confMenus.add(new ReleaseOrRequestChairManMenu()); //申请/释放主持人
        confMenus.add(new MuteOrUnmuteMenu()); //静音
        confMenus.add(new BroastAndCacelMenu()); //广播
        confMenus.add(new SetChairmanMenu()); //设为主席
        confMenus.add(new RecallMenu()); //重新呼叫
        if
(LoginConstant.AppEdition.China.getEdition().equals(PreferenceUtils.read(PreferenceUtils.PREFERENCES_NAME, APP_EDITION, LoginConstant.AppEdition.China.getEdition(), Utils.getApp())) {
            confMenus.add(new CallOtherNumberMenu()); //呼叫其他号码
        }
        confMenus.add(new SelectWatchOrCacnelMenu()); //选看
        confMenus.add(new RenameMenu()); //重命名
    }
}
```

```

confMenus.add(new HandsUpOrDownMenu());//举手
confMenus.add(new AttendeeProfileMenu());//查看与会者详情
confMenus.add(new HangupMenu());//挂断
confMenus.add(new RemoveMenu());//移除
//【演示】增加一个自定义菜单
confMenus.add(new CustomMoreMenu());
return confMenus;
}
@Override
public List<IConfMenu> buildParticipantToolbarMenuItems() {
    return null;
}
@Override
public List<IConfMenu> buildParticipantToolbarMoreMenuItems() {
    List<IConfMenu> confMenus = new ArrayList<>();
    confMenus.add(new AllowUnmuteOrNotMenu());
    confMenus.add(new ReleaseOrRequestChairManMenu());
    confMenus.add(new LockOrUnlockConfMenu());
    //【演示】增加一个自定义菜单
    confMenus.add(new CustomMoreMenu());
    return confMenus;
}
}

```

5.5.2.3 自定义会中标题栏点击弹出菜单

通过自定义会中标题栏的点击弹出菜单的接口可以实现点击图1弹出图2菜单面板中显示内容

使用场景

需要调整弹出的会议详情面板中显示的内容，比如隐藏密码、新增自定义view项等

注意事项

1.SDK初始化的时候必须传入这个配置sdkConfig才会生效

自定义会中标题栏点击弹出菜单的设置

示例代码

```

sdkConfig.setConfTitlePopwindowHandle(new CustomConfTitleBarPopwindowHandle());

```

图 5-27 会中标题栏点击区域

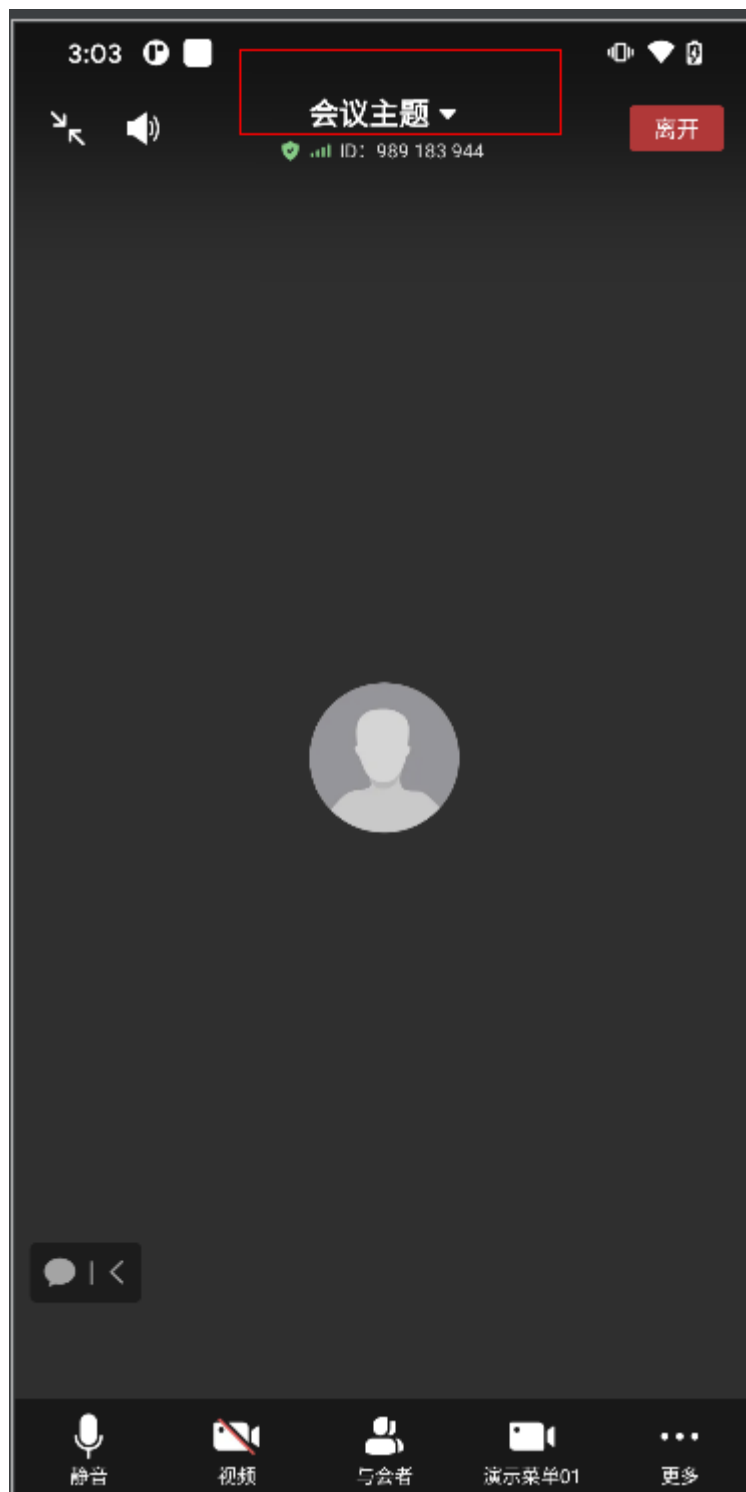


图 5-28 自定义会议详情面板



示例代码

```
//自定义会中标题栏点击弹出菜单, IMenuItemAction定义了一些View的操作函数, 也可直接强转成ViewGroup对子view进行增删改查
public class CustomConfTitleBarPopwindowHandle implements IConfTitleBarPopwindowHandle {
    @Override
    public void buildConfTitleBarPopwindow(IMenuAction iMenuItemAction) {
        if (!HWMBizSdk.getBizOpenApi().isSelfChairMan()) {
            //如果是主席, 不显示密码view
            View mPwdView = iMenuItemAction.findView(BuildInMenuType.CONF_TITLE_PWD_LAYOUT);
            if (mPwdView != null) {
                mPwdView.setVisibility(View.GONE);
            }
        } else {
            //如果是主席, 显示密码view
            View mPwdView = iMenuItemAction.findView(BuildInMenuType.CONF_TITLE_PWD_LAYOUT);
            if (mPwdView != null) {
                mPwdView.setVisibility(View.VISIBLE);
            }
        }
    }
}
```

5.5.2.4 自定义会议中加入人

点击与会者列表右上角加入按钮的时候, 会触发doAddAttendees的回调方法, 用户可以自定义实现该方法,

选择需要加入到会议中的联系人并通过callback的onSuccess回传选择的联系人列表

使用场景

应用自身有自己的企业通讯录等, 所以会中选择入会联系人需要自定义实现

注意事项

1.SDK初始化的时候必须传入这个配置sdkConfig才会生效

自定义加人配置接口:

```
sdkConfig.setAddAttendeesHandle(new CustomAddAttendeesHandle());
```

效果图如下图所示:

示例代码：自定义加人处理方法示例

```
//点击添加与会者界面注入，在方法中跳转到自己的选人界面
public class CustomAddAttendeesHandle implements IAddAttendeesHandle {
    private HwmCallback<List<AttendeeModel>> callback;
    /**
     * curActivity调用的activity
     * selectedAccount 已选择的帐号
     * isScheled 是否预约会议
     * callback 加人完成后返回已选中与会者列表
     */
    @Override
    public void doAddAttendees(Activity curActivity, List<AttendeeModel> selectedAccount, String
selectParam, boolean isScheled, HwmCallback<List<AttendeeModel>> callback) {
        callback = hwmCallback;
        //可以传入已经选择的与会者列表在自定义界面做操作
        Intent intent = new Intent(Utils.getApp(), AddPeopleByAppidAuthActivity.class);
        activity.startActivityForResult(intent,5000);
    }
    @Override
    public void dealContactSelect(Intent intent) {
    }
    @Override
    public void onActivityResult(int requestCode, int resultCode, Intent data) {
        if (requestCode == 5000){
            if(null != callback){
                //回调选中的与会者加入会议
                callback.onSuccess(DemoUtil.selectModels);
            }
        }
    }
}
```

示例代码：自定义加人界面处理，AddPeopleByAppidAuthActivity是示例的选人界面

```
//在activity的加人按钮的点击事件中添加与会人逻辑，并把与会人list传给自定义的Handle
public void onClick(View view) {
    String thirdAccountId = thirdAccountIdTextView.getText().toString();
    String sipNumber = sipNumberTextView.getText().toString();
    String phoneNumber = userPhoneTextView.getText().toString();
    //昵称选填，不填，则以企业通讯录的为准
    String name = userNameTextView.getText().toString();

    members = new ArrayList<>();
    //三种方式选一即可，第三方通讯录的id，仅仅对appid登录方式有效
    if (!TextUtils.isEmpty(thirdAccountId)) {
        AttendeeModel attendeeModel = AttendeeModel.buildAttendeeByThirdUserId(thirdAccountId,
name);
        members.add(attendeeModel);
    } else if (!TextUtils.isEmpty(sipNumber)) {
        //三种方式选一即可，sip号码是会议企业通讯录的账号标示
        AttendeeModel attendeeModel = AttendeeModel.buildAttendeeBySipNumber(sipNumber,name);
        members.add(attendeeModel);
    } else if (!TextUtils.isEmpty(phoneNumber)) {
        //三种方式选一即可，手机号码需要外呼权限
        AttendeeModel attendeeModel = AttendeeModel.buildAttendeeByPhone(phoneNumber, name);
        members.add(attendeeModel);
    } else {
        DemoUtil.showToast("三种入会方式需要选择一种");
        return;
    }
}
```

```
DemoUtil.selectModels = members;  
Intent intent = new Intent();  
setResult(1,intent);  
finish();  
}
```

5.5.2.5 自定义会中通知栏

该接口用于自定义通知栏中的会议通知图标，默认是如图1所示

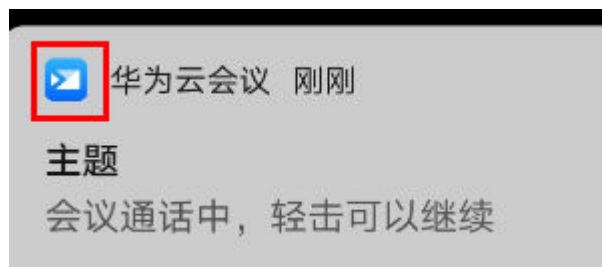
使用场景

通知栏中通知图标使用和自己应用图标一致的场景

注意事项

1.SDK初始化的时候必须传入这个配置sdkConfig才会生效

图 5-29 自定义会中通知栏



示例代码

```
sdkConfig.setNotificationStrategy(new INotificationStrategy() {  
    @Override  
    public int resSmallIcon() {  
        return R.mipmap.ic_launcher;  
    }  
});
```

5.5.2.6 头像拦截显示

接口描述

该接口用于定制登录后的头像显示。

调用场景:

图 5-30 画中画模式

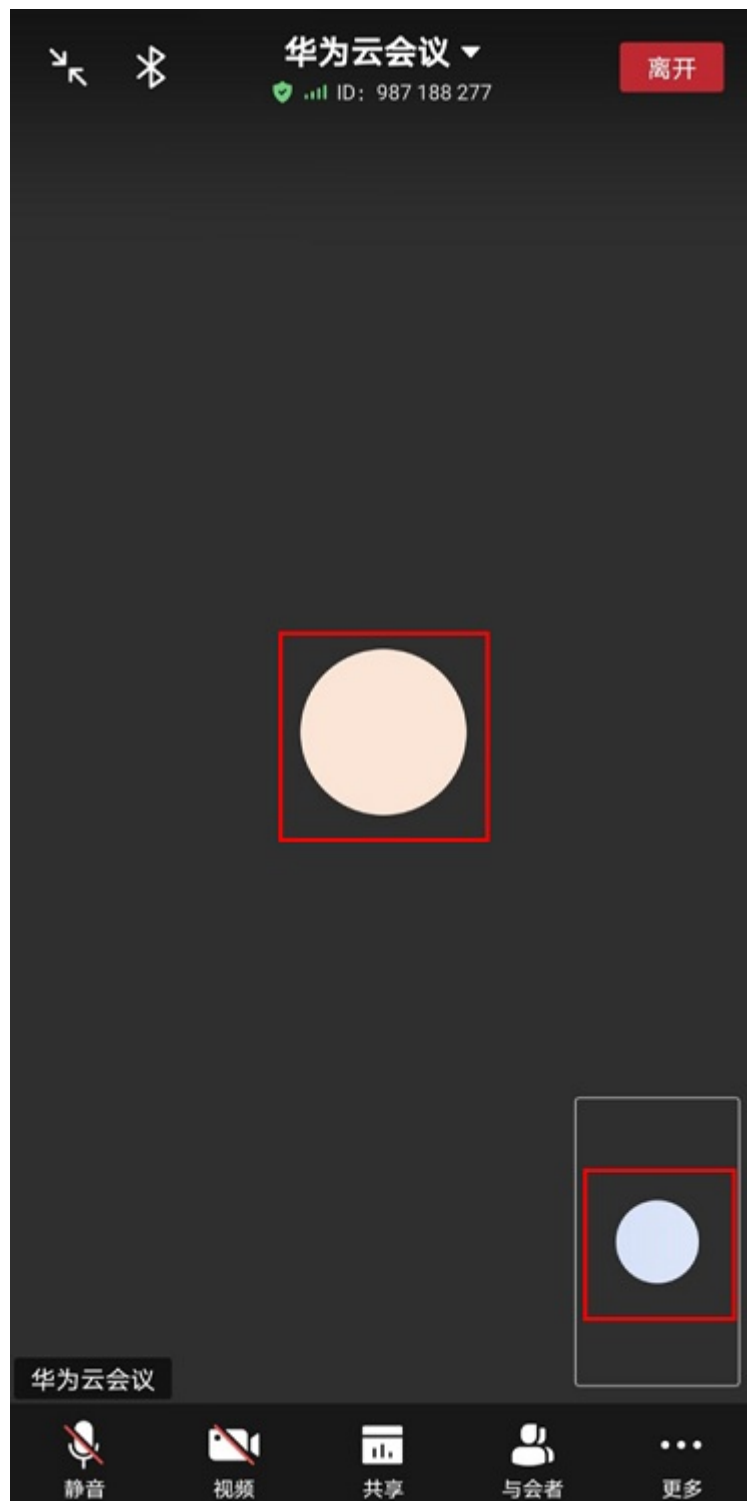


图 5-31 画廊模式

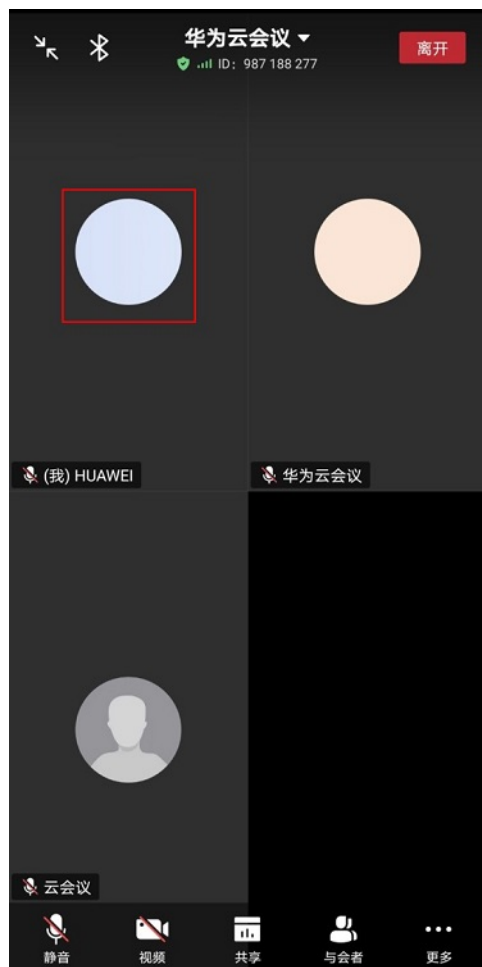


图 5-32 悬浮窗



注意事项

未登录的头像不支持自定义。

方法定义

```
public Bitmap queryContactAvatar(String account, String thirdUserId, String sipNumber)
```

参数描述

表 5-16

参数	类型	描述
account	String	帐号密码登录场景下优先使用此字段
thirdUserId	String	appid登录场景下优先使用此字段
sipNumber	String	跨企业场景优先使用此字段

返回值

表 5-17

参数	类型	描述
bitmap	Bitmap	头像位图类

示例代码

```

sdkConfig.setQueryContactInfoStrategy(new IQueryContactInfoStrategy() {
    @Override
    public Bitmap queryContactAvatar(String account, String thirdUserId, String sipNumber) {
        Log.i(TAG, "setQueryContactInfoStrategy account: " + account + " thirdUserId: " + thirdUserId + " sipNumber: " + sipNumber);
        // 第三方查询通讯录，返回对应用户的头像图片，这里返回一张默认图片作为示例代码
        return BitmapFactory.decodeResource(resources, R.mipmap.ic_launcher);
    }
});

```

5.5.2.7 自定义视频会议来电时接听方式按钮

接口描述

该接口用于自定义视频会议来电时接听按钮。默认情况下视频会议来电时有视频接听和语音接听两种方式，可以通过该接口配置成只显示一种接听方式。

图 5-33 会议来电接听方式按钮自定义



示例代码

```
public class MyUiStrategyWhenConflncoming implements IUiStrategyWhenConflncoming {  
    @Override  
    public AnswerBtnShowType getVideoConflncomingAnswerBtnShowType() {  
        // return AnswerBtnShowType.AnswerBtnShowTypeOnlyVideo; 只显示视频接听按钮  
        // return AnswerBtnShowType.AnswerBtnShowTypeOnlyAudio; 只显示语音接听按钮  
        // return AnswerBtnShowType.AnswerBtnShowTypeAll; 语音接听视频接听按钮都显示  
        return AnswerBtnShowType.AnswerBtnShowTypeAll;  
    }  
}  
  
sdkConfig.setUiStrategyWhenConflncoming(new MyUiStrategyWhenConflncoming());
```

5.5.2.8 自定义会议中水印

在OpenSDKConfig中配置会中共享内容的自定义水印

共享的水印效果参考图1

使用场景

需要共享内容添加水印的场景有：

- 1.内部会议，共享材料比较机密，限部分人员查看
- 2.涉外会议，跨企业会议等，防止共享材料外泄
- 3.其他可能导致重要共享材料泄漏的场景

注意事项

- 1.SDK初始化的时候必须传入这个配置sdkConfig才会生效

图 5-34 自定义水印



说明

在桌面共享、程序共享、白板共享的接收端叠加水印图片，只有服务器上会中水印打开的情况下自定义水印才生效。如果不自定义，默认按照手机号、邮箱、会议ID+昵称优先级顺序显示，字体透明度10%。

示例代码

```
//自定义会中水印
public class CustomWatermarkHandle implements IWatermarkHandle {
    @Override
    public Bitmap getWatermarkBitMap(Activity activity) {
        //设置水印bitmap
        return null;
    }
}
sdkConfig.setWatermarkHandle(new CustomWatermarkHandle());
```

5.5.2.9 自定义状态栏菜单

在OpenSDKConfig中配置会中状态栏的自定义

使用场景

不显示状态栏的录制按钮

注意事项

1.SDK初始化的时候必须传入这个配置sdkConfig才会生效

示例代码

```
public class CustomStatusBarMenuHandle implements IStatusBarMenuStrategy {
    @Override
    public List<IConfMenu> buildRecordMenuItems() {
        List<IConfMenu> confMenus = new ArrayList<>();
        // 如果不需要在会中左上角状态栏位置显示录制按钮，则注释掉下面的这行代码
        confMenus.add(new StatusBarRecordMenu());
        return confMenus;
    }
}

sdkConfig.setStatusBarMenuStrategy(new CustomStatusBarMenuHandle ());
```

5.5.3 帐号操作

5.5.3.1 帐号密码方式登录

```
HWMSdk.getOpenApi.login(LoginParam,HwmCallback )
```

接口描述

该接口用于登录和鉴权。

注意事项

除匿名入会外，在使用其他业务之前必须进行成功登录。

📖 说明

登录成功以后会发送登录成功事件onSuccess回调。

方法定义

```
/**
 * 登录接口
 * @param LoginParam 登录配置对象
 * @param hwmCallback 登录结果回调
 */
void login(LoginParam loginParam HwmCallback<LoginResult> hwmCallback)
```

参数描述

表 5-18 参数说明

参数	是否必须	类型	描述
loginParam	是	LoginParam	登录配置对象

参数	是否必须	类型	描述
hwmCallback	是	HwmCallback<LoginResult>	回调处理对象

返回值

无。

参数拓展

表 5-19 LoginParam 参数说明

参数	是否必须	类型	描述
account	是	String	用户名
password	是	String	密码
huaweiAccountAccessToken	否	String	华为accesstoken帐号
nickName	否	String	昵称

表 5-20 LoginResult 类型说明

参数	类型	描述
userId	String	用户userId，由系统分配的真实的唯一用户标识。
account	String	登录账号
thirdAccount	String	第三方登录账号（AppId登录场景）
isFreeUser	boolean	是否为免费用户
refreshToken	String	刷新token
siteDomain	String	登录域名。只适用于站点切换场景，当用户选择了国际站点，却使用国内账号登录时，会自动切换到国内站点登录，并返回国内站点的域名信息。

示例代码

```

LoginParam loginParam = new LoginParam()
    .setLoginAuthType(LoginAuthType.Account_And_Password) //登录方式的类别
    .setAccount("139571854984") //登录帐号
    .setPassword("Change_Me"); //登录密码
HwMSdk.getOpenApi(getActivity()).login(loginParam, new HwmCallback<LoginResult>() {

```

```

@Override
public void onSuccess(LoginResult loginResult) { //登录成功回调
    if (loginResult != null ) {
        Log.i(TAG, "登录成功" + loginResult.getUserUid());
    }
}
@Override
public void onFailed(int retCode, String desc) { //失败回调
    Error error = HWMBizSdk.getLoginApi().convertErrorCodeToUI(retCode);
    String errorTip = getLoginErrTips(error); //根据返回的错误值，输出错误原因
    Log.e(TAG, errorTip);
}
});
    
```

5.5.3.2 App ID 方式登录

`HWMSdk.getOpenApi.loginByAppId(AppIdAuthParam,HwmCallback)`

接口描述

该接口用于通过App ID的方式登录和鉴权。

注意事项

除匿名入会外，在使用其他业务之前必须进行成功登录。

说明

登录成功以后会发送登录成功事件onSuccess回调。

方法定义

```

/**
 * 登录接口
 * @param appIdAuthParam 登录配置对象
 * @param hwmCallback 登录结果回调
 */
void loginByAppId(AppIdAuthParam appIdAuthParam HwmCallback<LoginResult> hwmCallback)
    
```

参数描述

表 5-21 参数说明

参数	是否必须	类型	描述
appIdAuthParam	是	AppIdAuthParam	登录配置对象
hwmCallback	是	HwmCallback<LoginResult>	回调处理对象

返回值

无。

参数拓展

表 5-22 AppIdAuthParam 参数说明

参数	是否必须	类型	描述
thirdUserId	是	String	第三方帐号的唯一标识，1到64位 说明：App ID的方式鉴权，需要保证App ID的是在会议服务器上申请的，App ID通过HWMSdk.init时，传入OpenSdkConfig，通过OpenSdkConfig.setAppId(appid)传入
nonce	是	String	随机字符串，32到64位
corpId	否	String	设置该用户所在的企业ID。仅在SP模式下需要填写，单企业模式不要填写，否则会鉴权失败。
signature	是	String	签名信息
expireTime	是	long	签名过期时间，单位为秒的Unix时间戳，为0则不会过期
userName	否	String	用户名称(如不填，默认与thirdUserId一致)。
userEmail	否	String	用户邮箱
userPhone	否	String	用户手机号码
deptCode	否	String	部门编码

示例代码

```
//构建登录对象
AppIdAuthParam appIdAuthParam = new AppIdAuthParam();
//设置签名过期时间，单位为秒的Unix时间戳，为0则不会过期
appIdAuthParam.setExpireTime(1599618620);
//设置随机字符串，长度为32-64位
appIdAuthParam.setNonce("602b111272ad4b6989e683c27e6e87d4");
//设置该用户所在的企业ID,仅SP登录场景下需要
appIdAuthParam.setCorpId(123);
//设置第三方帐号的id，长度为1到64位
appIdAuthParam.setUserId("admin@1");
//设置签名信息
appIdAuthParam.setSignature("2a8c780cee3dbfe210384c3f95380732d55dfc81cfa49c5a6c44f3c1b3c2455d");
//设置用户昵称
appIdAuthParam.setUserName("Mike");
//注意邮箱需要符合邮箱格式要求
appIdAuthParam.setUserEmail("huawei123456@huawei.com");
//注意手机号码必须符合手机格式，比如(^$|^[\+]?[0-9]+\$$)
appIdAuthParam.setUserPhone("12345678901");
//开始登录
HWMSdk.getOpenApi(getActivity().getApplication()).loginByAppId(appIdAuthParam, new
HwmCallback<LoginResult>() {
    @Override
    public void onSuccess(LoginResult loginResult) {
```

```
dismissLoading();
dismiss();
if (loginResult != null) {
    DemoUtil.showToast("登录成功" + loginResult.getUserUuid());
}else{
    DemoUtil.showToast("已登录");
}
}

@Override
public void onFailed(int retCode, String desc) {
    HwmContext.getInstance().runOnMainThread() ->{
        dismissLoading();
        Error error = HWMBizSdk.getLoginApi().convertErrorCodeToUI(retCode);
        DemoUtil.showToast("登录错误: " + getLoginErrTips(error));
        dismiss();
    };
}
});
```

5.5.3.3 SSO 方式登录

`HWMSdk.getOpenApi.loginBySSO(SsoAuthParam ssoAuthParam, HwmCallback<LoginResult> callback)`

接口描述

该接口用于通过SSO的方式登录和鉴权。

注意事项

除匿名入会外，在使用其他业务之前必须进行成功登录。

📖 说明

登录成功以后会发送登录成功事件onSuccess回调。

方法定义

```
/**
 * SSO登录接口
 * @param code 授权码
 * @param domain 企业域名
 * @param authServerType 鉴权服务器类型
 * @param callback
 */
void loginBySSO(SsoAuthParam ssoAuthParam, HwmCallback<LoginResult> callback);
```

参数描述

表 5-23 参数说明

参数	是否必须	类型	描述
ssoAuthParam	是	SsoAuthParam	SSO登录鉴权参数
callback	是	HwmCallback<LoginResult>	回调处理对象

表 5-24 SsoAuthParam 参数说明

参数	是否必须	类型	描述
code	是	String	SSO授权码
domain	是	String	SSO企业域名
authServerType	是	AuthServerType	鉴权服务器类型，枚举取值如下 AUTH_SERVER_TYPE_OAUTH2(0, "单点登录")

返回值

无。

示例代码

```
SsoAuthParam ssoAuthParam = new SsoAuthParam();
ssoAuthParam.setDomain(domain);
ssoAuthParam.setCode(code);
ssoAuthParam.setAuthServerType(AuthServerType.AUTH_SERVER_TYPE_OAUTH2);
HWMSdk.getOpenApi(ssoAuthParam, new HwmCallback<LoginResult>() {
    @Override
    public void onSuccess(LoginResult loginResult) {
        dismissLoading();
        if (loginResult != null) {
            DemoUtil.showToast("登录成功" + loginResult.getUserUid());
        } else {
            DemoUtil.showToast("已登录");
        }
    }

    @Override
    public void onFailed(int retCode, String desc) {
        HwmContext.getInstance().runOnUiThread(() -> {
            DemoUtil.showToast("登录错误: code=" + retCode + " desc=" + desc);
            dismissLoading();
        });
    }
});
```

5.5.3.4 注销

```
HWMSdk.getOpenApi.logout(HwmCallback<LogoutResult> hwmCallback)
```

接口描述

该接口用于注销已登录的帐号。

注意事项

请在已登录状态下调用。

方法定义

```
/**
 * 注销接口
 * @param hwmCallback 回调对象
```

```
*/
void logout(HwmCallback<LogoutResult> hwmCallback);
```

参数描述

表 5-25 参数说明

参数	是否必须	类型	描述
hwmCallback	是	HwmCallback <LogoutResult >	回调处理对象

表 5-26 LogoutResult 类型说明

参数	类型	描述
userId	String	用户uuid，由系统分配的真实的唯一用户标识。
account	String	登录账号
thirdAccount	String	第三方登录账号（Appld登录场景）

返回值

无。

示例代码

```
HWMSdk.getOpenApi(getActivity()).logout(new HwmCallback<LogoutResult>() {
    @Override
    public void onSuccess(LogoutResult result) {
        Log.i("", getContext(), "登出成功");
    }

    @Override
    public void onFailed(int retCode, String desc) {
        Log.i("", getContext(), "登出失败 : " + retCode + " ; " + desc);
    }
});
```

5.5.3.5 上传个人头像

```
HWMBizSdk.getBizOpenApi().uploadSelfAvatar(String path, HwmCallback<Void> hwmCallback)
```

接口描述

该接口用于上传当前自己的头像到会议服务器的通讯录

注意事项

1. 请在已登录状态下调用。
2. 请上传png格式或jpg格式的图片，且上传的头像大小不超过200kb。

方法定义

```
/**
 * 上传当前自己的头像到会议服务器的通讯录
 * @param path 本地头像文件路径
 * @param hwmCallback 设置头像结果回调
 */
void uploadSelfAvatar(String path, HwmCallback<Void> hwmCallback);
```

参数描述

表 5-27 参数说明

参数	是否必须	类型	描述
path	是	String	本地头像文件路径
hwmCallback	是	HwmCallback<Void>	回调处理对象

返回值

无。

示例代码

```
HWMBizSdk.getBizOpenApi().uploadSelfAvatar(filepath, new HwmCallback<Void>() {
    @Override
    public void onSuccess(Void aVoid) {
        if (getActivity() != null) {
            ((DemoActivity) getActivity()).dismissLoading();
        }
        DemoUtil.showToast("设置头像成功! ");
    }

    @Override
    public void onFailed(int retCode, String desc) {
        if (getActivity() != null) {
            ((DemoActivity) getActivity()).dismissLoading();
        }
        if (!TextUtils.isEmpty(desc)) {
            DemoUtil.showToast(desc);
        } else {
            DemoUtil.showToast("设置头像失败! ");
        }
    }
});
```

5.5.4 会议管理

5.5.4.1 预约会议

接口描述

该接口用于预约会议。

注意事项

- 不在登录状态下，不应调用该接口。
- 会议开始时间必须晚于当前时间，会议时长应大于15分钟。

- 个人会议或者云会议室的时候要传入vmrId。
- 携带与会者入会的时候需要传入attendees。
- callback中的result返回会议详情信息（[ConfInfo类型说明](#)）

方法定义

```
/**
 * 预约会议
 * @param bookConfParam 预约会议参数
 * @param hwmCallback 结果回调
 */
void bookConf(BookConfParam bookConfParam , HwmCallback<ConfInfo> hwmCallback);
```

参数描述

表 5-28 BookConfParam 参数说明

参数	是否必须	类型	描述
confSubject	是	String	会议主题
startTime	是	long	会议开始时间，单位为秒的UTC时间戳
duration	是	int	会议时长-单位分钟，最小值15分钟，最大值1425分钟
timeZone	否	int	时区码，默认值56-东八区，详情参考 时区表
confType	是	MeetingType	会议类型，默认CONF_VIDEO CONF_AUDIO(0, "语音会议"), CONF_VIDEO(1, "视频会议");
attendees	否	List<AttendeeModel>	与会人列表
vmrIdFlag	否	boolean	预约会议是否使用云会议室ID，默认false
vmrId	否	String	云会议室ID，个人会议或者云会议室时传入(vmrIdFlag 为true时必填)
isNeedConfPwd	否	boolean	是否需要来宾密码，默认true 说明 仅对随机ID会议生效。
guestPwd	否	String	来宾密码，如果不填则来宾密码由服务端生成 说明 仅对随机ID会议生效。

参数	是否必须	类型	描述
joinConfRestrictionType	否	JoinConfPermissionType	<p>允许入会范围限制，默认 PERMIT_EVERYONE</p> <p>PERMIT_EVERYONE("PERMIT_EVERYONE", 0, "允许所有人入会"), PERMIT_ENTERPRISE_USER("PERMIT_ENTERPRISE_USER", 2, "仅允许企业内人员入会"), PERMIT_INVITED_USER("PERMIT_INVITED_USER", 3, "仅允许会议邀请人员入会");</p>
isRecordOn	否	boolean	<p>录制会议开关，会议是否支持录制功能，默认false</p> <p>说明 仅对云录制生效，客户端本地录制不生效。</p>
isAutoRecord	否	boolean	<p>会议开始后是否自动开启录制，默认false。开启自动录制会议时必须开启录制会议</p> <p>说明 仅对云录制生效，客户端本地录制不生效。</p>
isSmsOn	否	boolean	<p>发送短信通知开关，默认false，企业配置里面需要开启，否则不生效</p>
isMailOn	否	boolean	<p>发送邮件通知开关，默认false</p>
isEmailCalendarOn	否	boolean	<p>发送邮件日历开关，默认false</p>
vmrIdType	否	VmrIdType	<p>云会议室的会议ID类型，固定会议ID或者随机会议ID。</p>
allowGuestStartConf	否	boolean	<p>是否允许来宾提前启动会议。</p> <p>说明</p> <ul style="list-style-type: none"> • 仅针对随机会议ID的会议生效。 • 固定会议ID的会议，请使用修改云会议室信息接口设置。

参数	是否必须	类型	描述
allowGuestStartConfTime	否	int	允许来宾提前入会时间范围（单位：分钟）： 0 - 随时 x - 提前x分钟启动会议 说明 <ul style="list-style-type: none"> 仅针对随机会议ID的会议生效。 固定会议ID的会议，不支持设置提前入会时间。
isOpenWaitingRoom	否	boolean	是否开启等候室： 说明 <ul style="list-style-type: none"> 等候室功能开通后才生效。

表 5-29 AttendeeModel 参数说明

参数	是否必须	类型	描述
number	是	String	SIP呼叫号码或者电话号码（number/thirdUserId 二选一）
thirdUserId	是	String	App ID登录场景下的第三方帐号（number/thirdUserId 二选一）
name	是	String	名字
isSelf	否	boolean	是否是用户自己
isAutoInvite	否	boolean	会议时间到了是否自动呼叫
isMute	否	boolean	入会是否静音
accountId	否	String	用户账号唯一ID，企业通讯录的id
type	否	HwmAttendeeType	与会者类型
email	否	String	邮箱
sms	否	String	手机号码
email	否	String	邮箱
orgId	否	String	用户的企业ID, 用于识别是否是企业账号（仅用于获取与会者列表使用）

参数	是否必须	类型	描述
isAnonymous	否	boolean	是否是匿名入会
state	否	ConfAttendeeState	与会者状态（仅用于获取与会者列表使用）

表 5-30 ConfInfo 类型说明

成员名称	类型	描述
confSubject	String	会议主题
confId	String	会议ID
isOtherCorpConf	boolean	是否企业外会议
vmrConferenceId	String	个人会议id
confPwd	String	会议密码
confAccessNum	String	会议接入号
confChairPwd	String	会议主席密码
confGuestUri	String	会议来宾（普通用户）链接
confStartTime	String	会议开始时间
confEndTime	String	会议结束时间
confSchedulerName	String	会议创建者
confOrgId	String	企业id
audienceJoinUri	String	观众入会链接（网络研讨会）
audienceJoinPwd	String	观众入会密码（网络研讨会）
confType	ConfType	会议类型，区分普通会议和网络研讨会
conferenceType	ConferenceType	区分普通会议和周期会议
cycleConfParam	CycleConfParam	周期会议参数
subConfSize	int	周期子会议数量
subConfParam	List<SubCycleConfParam>	周期子会议参数
selfConfRole	ConfRole	本人在会议中的角色
isInBreakoutSubConf	boolean	是否在分组讨论中

表 5-31 VmroleIdType 枚举说明

枚举名称	枚举值	枚举说明
FIXED_ID	0	VmroleId是固定类型
RANDOM_ID	1	VmroleId是随机类型

示例代码

与会人列表List<AttendeeModel>的构造:

```
boolean needWithMember = true;
List<AttendeeModel> attendeeInfos = new ArrayList<>();
if (needWithMember) {
    //以下三种方式可选
    String name = "huawei123456";
    String number = "+99123456xxx";
    String thirdAccountId = "123456abcd";
    if (!TextUtils.isEmpty(thirdAccountId)) {
        attendeeInfos.add(AttendeeModel.buildAttendeeByThirdUserId(thirdUserId, name));
    } else if (number.startsWith("+99")) {
        //sip 号码入会
        attendeeInfos.add(AttendeeModel.buildAttendeeBySipNumber(number, name));
    } else {
        //电话号码入会 (手机和座机)
        attendeeInfos.add(AttendeeModel.buildAttendeeByPhone(number, name));
    }
}
```

构造通过appid登录时的与会者模型, 其他模型可参考demo代码:

```
public static AttendeeModel buildAttendeeByThirdUserId(@NonNull String thirdUserId,
String nickName) {
    AttendeeModel attendeeInfo = new AttendeeModel();
    attendeeInfo.setAppId(FOUNDATION.getAppId());
    attendeeInfo.setThirdUserId(thirdUserId);
    attendeeInfo.setName(nickName);
    attendeeInfo.setIsAutoInvite(true);
    attendeeInfo.setIsMute(true);
    attendeeInfo.setRole(ConfRole.ATTENDEE);
    attendeeInfo.setType(HwmAttendeeType.ATTENDEE_TYPE_NORMAL);
    return attendeeInfo;
}
```

```
private HwmCallback<ConfInfo> completeHandler = new HwmCallback<ConfInfo>() {
    @Override
    public void onSuccess(ConfInfo result) {
        dismissLoading();
        DemoUtil.showToast("预约会议成功");
    }

    @Override
    public void onFailed(int retCode, String desc) {
        dismissLoading();
        dismiss();
        String err = ErrorMessageFactory.create(Utils.getApp(), retCode);
        if (TextUtils.isEmpty(err)) {
            err = Utils.getApp().getString(R.string.hwmconf_book_conf_fail);
        }
        DemoUtil.showToast("预约会议失败: " + retCode + ", desc:" + err);
    }
};
BookConfParam bookConfParam = new BookConfParam()
    .setConfSubject("我的会议")
    .setStartTime(1599574798)
    .setTimeZone(56)
```

```

        .setDuration(60)
        .setConfType(MeetingType.CONF_VIDEO)
        .setVmridFlag(false)
        .setVmrid("")
        .setNeedConfPwd(true)
        .setJoinConfRestrictionType(JoinConfPermissionType.PERMIT_EVERYONE)
        .setRecordOn(true)
        .setMailOn(true)
        .setSmsOn(true)
        .setEmailCalenderOn(true)

        .setAttendees(attendeeInfos);
HwMBizSdk.getBizOpenApi().bookConf(bookConfParam, this.completeHandler);

```

5.5.4.2 编辑会议

接口描述

该接口用于编辑会议。

注意事项

- 会议未开始前可以调用接口，开始后调用则无效。
- 所有字段必填，编辑修改的字段可来源于用户输入，其他不想修改的属性一定要取会议详情对应字段来赋值。

方法定义

```

/**
 * 编辑会议
 * @param editConfParam 编辑会议参数
 * @param hwmCallback 结果回调
 */
void editConf(EditConfParam editConfParam, HwmCallback<Integer> hwmCallback);

```

参数描述

表 5-32 EditConfParam 参数说明

参数	是否必须	类型	描述
confId	是	String	会议ID
isLiveOn	否	boolean	会议直播开关，设置会议是否支持直播功能

其余参数参见[参数说明](#)

示例代码

```

EditConfParam editConfParam = new EditConfParam()
    .setConfId(this.confDetailModel.getConfId())
    .setVmridFlag(1 == this.confDetailModel.getVmridFlag())
    .setVmrid(this.confDetailModel.getVmrid())
    .setConfSubject("我的会议-编辑会议")
    .setStartTime(1599574798)
    .setTimeZone(56)
    .setDuration(60)
    .setConfType(MeetingType.CONF_VIDEO)
    .setNeedConfPwd(!this.confDetailModel.getIsGuestFreePwd())
    .setJoinConfRestrictionType(JoinConfPermissionType.PERMIT_EVERYONE)

```

```

        .setRecordOn(true)
        .setMailOn(true)
        .setSmsOn(true)
        .setEmailCalenderOn(true)
        .setAttendees(attendeeInfos);
HWMBizSdk.getBizOpenApi().editConf(editConfParam, new HwmCallback<Integer>() {
    @Override
    public void onSuccess(Integer integer) {
        dismissLoading();
        DemoUtil.showToast("编辑会议成功: " + integer);
    }

    @Override
    public void onFailed(int retCode, String desc) {
        dismissLoading();
        String err = ErrorMessageFactory.create(Utils.getApp(), retCode);
        if (TextUtils.isEmpty(err)) {
            err = Utils.getApp().getString(R.string.hwmconf_edit_failed);
        }
        DemoUtil.showToast("编辑会议失败: " + retCode + ", desc: " + err);
    }
});

```

5.5.4.3 取消会议

接口描述

该接口用于取消未开始的会议。

注意事项

会议未开始前可以调用接口，开始后调用则无效。

方法定义

```

/**
 * 取消会议
 *
 * @param confId 会议Id
 * @param hwmCallback 结果回调
 */
void cancelConf(String confId, HwmCallback<Integer> hwmCallback);

```

参数描述

表 5-33 参数说明

参数	是否必须	类型	描述
confId	是	String	会议ID

示例代码

```

HWMBizSdk.getBizOpenApi().cancelConf(item.getConfId(), new HwmCallback<Integer>() {
    @Override
    public void onSuccess(Integer result) {
        dismissLoading();
        DemoUtil.showToast("取消会议成功");
    }

    @Override
    public void onFailed(int retCode, String desc) {
        dismissLoading();
    }
});

```

```
dismiss();
String err = ErrorMessageFactory.create(Utils.getApp(), retCode);
if (TextUtils.isEmpty(err)) {
    err = Utils.getApp().getString(R.string.hwmconf_cancel_fail_tip);
}
DemoUtil.showToast("取消会议失败: " + err);
}
});
```

5.5.4.4 获取会议列表

接口描述

该接口用于获取会议列表。

注意事项

已登录状态，首次收到会议列表通知之后，调用才有效，否则返回为空的列表。

方法定义

```
/**
 * 获取会议列表
 * @return
 */
List<ConfBaseInfo> getConfList();
```

参数描述

表 5-34 ConfBaseInfo 参数说明

参数	类型	描述
confId	String	会议ID
vmrConferenceId	String	云会议室ID,云会议室显示时使用
confSubject	String	会议主题
chairmanPwd	String	主席密码（自己是主席时才会有主席密码）
guestPwd	String	来宾密码
startTime	long	会议开始时间，UTC时间戳，精度秒
endTime	long	会议结束时间，UTC时间戳，精度秒
mediaType	int	17：音频会议, 21:视频会议
scheduserName	String	会议预订者姓名
chairJoinUri	String	主持人入会链接（主持人才有）
guestJoinUri	String	普通与会者入会链接
accessNumber	String	会议接入码

示例代码

```
HWMBizSdk.getBizOpenApi().getConfList()
```

5.5.4.5 获取会议详情

接口描述

该接口用于获取会议详情。

注意事项

ConfDetailModel参数说明，继承于[ConfBaseInfo](#)

方法定义

```
/**
 * 根据会议id 获取会议详情
 *
 * @param confId 会议id
 * @param hwmCallback 结果回调
 */
void getConfDetail(String confId, HwmCallback<ConfDetailModel> hwmCallback);
```

参数描述

表 5-35 ConfDetailModel 参数说明，继承于 ConfBaseInfo

参数	类型	描述
vmrId	String	用VMRId创建的会议的标识，编辑会议时用
vmrFlag	int	是否是VMR会议
vmrIdType	VmrIdType	VMR会议ID类型
callInRestrictionType	JoinConfPermissionType	允许入会范围限制 PERMIT_EVERYONE("PERMIT_EVERYONE", 0, "允许所有人入会"), PERMIT_ENTERPRISE_USER("PERMIT_ENTERPRISE_USER", 2, "仅允许企业内人员入会"), PERMIT_INVITED_USER("PERMIT_INVITED_USER", 3, "仅允许会议邀请人员入会");
isRecordOn	boolean	是否是录制会议
isAutoRecord	boolean	会议开始后是否自动开启录制
isLiveOn	boolean	是否开启直播
isMailOn	boolean	是否发送邮件通知
isSmsOn	boolean	是否发送短信通知
isEmailCalenderOn	boolean	是否发送日历通知
isGuestFreePwd	boolean	是否来宾免密码入会
minutes	int	会议时长

参数	类型	描述
allowGuestStartConf	boolean	是否允许来宾提前启动会议
allowGuestStartConfTime	int	允许来宾提前入会时间范围（单位：分钟） 0 - 随时 x - 提前x分钟启动会议
attendeeModels	List<AttendeeModel>	与会者列表
isCycleConf	boolean	是否为周期会议
cycleConfParam	CycleConfParam	周期会议属性
subCycleConfList	List<SubCycleConfParam>	周期子会议列表
isOpenWaitingRoom	boolean	是否开启等候室

示例代码

```
HwMBizSdk.getBizOpenApi().getConfDetail(confid, new HwmCallback<ConfDetailModel>() {
    @Override
    public void onSuccess(ConfDetailModel confDetailModel) {
        DemoUtil.showToast("获取会议详情成功。");
        Message message = Message.obtain(confDetailPageHandler);
        message.what = HANDLE_MESSAGE_GET_CONF_DETAIL_SUCCESS;
        message.obj = confDetailModel;
        message.sendToTarget();
    }

    @Override
    public void onFailed(int retCode, String desc) {
        DemoUtil.showToast("获取会议详情失败: " + retCode + ", " + desc);
    }
});
```

5.5.4.6 获取云会议室列表

接口描述

该接口用于获取VMR(云会议室)列表。

注意事项

无

方法定义

```
/**
 * 获取VMR(云会议室)列表
 * @param hwmCallback 结果回调
 */
void getVmrList(HwmCallback<List<VmrInfoModel>> hwmCallback);
```

参数描述

表 5-36 VmrInfoModel 参数说明

参数	类型	描述
name	String	会议室名称
vmrConferenceId	String	用于云会议室显示的ID
vmrId	String	云会议室唯一ID
chairmanPwd	String	主席密码
guestPwd	String	来宾密码
type	MeetingIdType	会议ID类型 RANDOM_MEETING_ID(0, "随机会议ID"), PERSONAL_MEETING_ID(1, "个人会议ID"), CLOUD_MEETING_ROOM_ID(2, "云会议室ID");
maxParties	int	会议室最大方数
vmrEnable	boolean	云会议室是否可用
confAllowJoinUser	ConfAllowJoinUserType	允许入会的用户类型 CONF_ALLOW_JOIN_ANYONE(0, "Anyone:任何人"), CONF_ALLOW_JOIN_LOGINED_USER(1, "logined user:已登录用户"), CONF_ALLOW_JOIN_IN_COMPANY_USER(2, "Company user:企业内用户"), CONF_ALLOW_JOIN_INVITED_USER(3, "Invited user:被邀请用户")
isOpenWaitingRoom	boolean	是否开启等候室

示例代码

```
HWMBizSdk.getBizOpenApi().getVmrList(new HwmCallback<List<VmrInfoModel>>() {
    @Override
    public void onSuccess(List<VmrInfoModel> vmrInfoModelList) {
        Log.i(TAG, "getVmrList: " + (vmrInfoModelList == null ? 0 : vmrInfoModelList.size()));
        updateVmrList(vmrInfoModelList);
    }

    @Override
    public void onFailed(int retCode, String desc) {

    }
});
```

5.5.4.7 修改云会议室信息

接口描述

该接口用于修改云会议室信息。

注意事项

无

方法定义

```
/**
 * 修改虚拟会议室信息
 * @param vmrParam vmr信息, guestPwd-来宾密码 hostPwd-主席密码 vmrName-虚拟会议室名称 vmrId-
虚拟会议室唯一ID
 * @param hwmCallback 结果回调
 */
void changeVmrInfo(ModifyVmrParam vmrParam, HwmCallback<Void> hwmCallback);
```

参数描述

表 5-37 ModifyVmrParam 参数说明

参数	是否必须	类型	描述
vmrId	是	String	云会议室唯一ID
guestPwd	是	String	来宾密码
hostPwd	是	String	主席密码
subject	是	String	云会议室主题
allowGuestStartConf	否	boolean	是否允许来宾提前启动会议
confAllowJoinUser	否	ConfAllowJoinUserType	允许入会的用户类型： CONF_ALLOW_JOIN_ANYONE(0, "Anyone:任何人"), CONF_ALLOW_JOIN_LOGINED_USER(1, "logged user:已登录用户"), CONF_ALLOW_JOIN_IN_COMPANY_USER(2, "Company user:企业内用户"), CONF_ALLOW_JOIN_INVITED_USER(3, "Invited user:被邀请用户")
isOpenWaitingRoom	否	boolean	是否开启等候室

示例代码

```
ModifyVmrParam vmrParam = new ModifyVmrParam();
vmrParam.setGuestPwd(item.getGuestPwd());
```

```
vmrParam.setHostPwd(item.getChairmanPwd());
vmrParam.setVmrid(item.getVmrid());
vmrParam.setSubject(item.getName());
HwMBizSdk.getBizOpenApi().changeVmrInfo(vmrParam, new HwmCallback<Void>() {
    @Override
    public void onSuccess(Void v) {
        DemoUtil.showToast("修改Vmr来宾密码成功。");
    }

    @Override
    public void onFailed(int retCode, String desc) {
        DemoUtil.showToast("修改Vmr来宾密码失败: " + retCode + ", desc: " + desc);
    }
});
```

5.5.4.8 预约周期会议

接口描述

该接口用于预约周期会议。

注意事项

- 不在登录状态下，不应调用该接口。
- 会议开始时间必须晚于当前时间，会议时长应大于15分钟。
- 个人会议或者云会议室的时候要传入vmrId。
- 携带与会者入会的时候需要传入attendees。
- 周期会议最大跨度一年，会议数量不超过50，超过则只预约前50个会议
- callback中的result返回会议详情信息（[ConfInfo类型说明](#)）、

接口所在类

NativeSDK

方法定义

```
/**
 * 预约周期会议
 * @param bookCycleConfParam 预约周期会议参数
 * @param callback结果回调
 */
void bookCycleConf(BookCycleConfParam bookCycleConfParam, HwmCallback<ConfInfo> hwmCallback);
```

参数描述

表 5-38 参数说明

参数	是否必须	类型	描述
bookCycleConfParam	是	BookCycleConfParam	预约周期会议的参数体
callback	是	HwmCallback<ConfInfo> hwmCallback	创建周期会议的结果回调

表 5-39 BookCycleConfParam 参数说明

参数	是否必须	类型	描述
bookConfParam	是	BookConfParam	预约会议参数
cycleConfParam	是	CycleConfParam	周期子会议参数

表 5-40 CycleConfParam 参数说明

参数	是否必须	类型	描述
startDate	是	long	开始日期-时间戳，精度秒(0时区)
endDate	是	long	结束日期-时间戳，精度秒(0时区)
cycleType	是	CycleType	周期类型
interval	是	int	周期区间： 1、周期类型选择了按天，表示每几天召开一次，取值范围[1,15] 2、周期类型选择了Week，表示每几周召开一次，取值范围[1,5] 3、周期类型选择了Month，interval表示隔几月，取值范围[1,3]
listPoints	是	String	周期内的会议召开点。仅当按周和月时有效。数据格式(中间用逗号隔开):1,3,5 - 当按周时，取值范围[0,6]，0代表周日，1代表周一，以此类推； - 当按月时，取值范围[1,31]，若当月不含31号，则认为是当月最后一天
preRemindDays	是	int	周期子会议提前通知天数

示例代码

```
CycleConfParam cycleConfParam = new CycleConfParam();
cycleConfParam.setCycleType(mSelectedCycleType);
cycleConfParam.setInterval(interval);
cycleConfParam.setListPoints(subDates);
cycleConfParam.setStartDate(startTimeStamp);
cycleConfParam.setEndDate(endTime.getTimeInMillis() / 1000);
cycleConfParam.setPreRemindDays(preMindDays);

BookConfParam bookConfParam = new BookConfParam()
    .setConfSubject("我的会议")
    .setStartTime(1599574798)
    .setTimeZone(56)
    .setDuration(60)
    .setConfType(MeetingType.CONF_VIDEO)
    .setVmridFlag(false)
    .setVmrid("")
```

```

.setNeedConfPwd(true)
.setJoinConfRestrictionType(JoinConfPermissionType.PERMIT_EVERYONE)
.setRecordOn(true)
.setMailOn(true)
.setSmsOn(true)
.setEmailCalenderOn(true)

.setAttendees(attendeeInfos);

BookCycleConfParam bookCycleConfParam = new BookCycleConfParam();
bookCycleConfParam.setBookConfParam(bookConfParam);
bookCycleConfParam.setCycleConfParam(cycleConfParam);
HWMBizSdk.getBizOpenApi().bookCycleConf(bookCycleConfParam, completeHandler);

```

5.5.4.9 编辑周期会议

接口描述

该接口用于编辑周期会议。

注意事项

无

方法定义

```

/**
 * 编辑周期会议
 * @param editConfParam 编辑周期会议参数
 * @param hwmCallback 结果回调
 */
void editCycleConf(EditCycleConfParam editCycleConfParam, HwmCallback<Integer> hwmCallback);

```

参数描述

表 5-41 参数说明

参数	是否必须	类型	描述
editCycleConfParam	是	EditCycleConfParam	编辑周期会议的参数体
callback	是	HwmCallback<Integer>	编辑周期会议的结果回调

表 5-42 EditCycleConfParam 参数说明

参数	是否必须	类型	描述
editConfParam	是	EditConfParam	编辑会议参数
cycleConfParam	是	CycleConfParam	周期子会议参数

示例代码

```

HWMBizSdk.getBizOpenApi().editCycleConf(editCycleConfParam, new HwmCallback<Integer>() {
    @Override
    public void onSuccess(Integer integer) {

```

```

        dismissGlobalLoading();
        DemoUtil.showToast("编辑周期会议成功: " + integer);
    }

    @Override
    public void onFailed(int retCode, String desc) {
        HLog.i(TAG, "edit cycle conf " + retCode + " desc " + desc);
        dismissGlobalLoading();
        String err = ErrorMessageFactory.create(retCode);
        if (TextUtils.isEmpty(err)) {
            err = DemoUtil.getResources().getString(R.string.hwmconf_edit_failed);
        }
        DemoUtil.showToast("编辑周期会议失败: " + retCode + ", desc: " + err);
    }
});

```

5.5.4.10 取消周期会议

接口描述

该接口用于取消未开始的周期会议。

注意事项

周期会议未全部结束前可以调用接口，结束后调用则无效。

方法定义

```

/**
 * 取消周期会议
 *
 * @param cycleConfId 周期会议Id
 * @param hwmCallback 结果回调
 */
void cancelCycleConf(String cycleConfId, HwmCallback<Integer> hwmCallback);

```

参数描述

表 5-43 参数说明

参数	是否必须	类型	描述
confConfId	是	String	周期会议ID

示例代码

```

HWMBizSdk.getBizOpenApi().cancelCycleConf(item.getConfId(), new HwmCallback<Integer>() {
    @Override
    public void onSuccess(Integer result) {
        HLog.i(TAG, "cancelCycleConf onSuccess" + result);
        doDismissDialogAndFragment();
        DemoUtil.showToast("cancel cycle conf success");
    }

    @Override
    public void onFailed(int retCode, String desc) {
        doDismissDialogAndFragment();
        String err = ErrorMessageFactory.create(retCode);
        if (TextUtils.isEmpty(err)) {
            err = DemoUtil.getResources().getString(R.string.hwmconf_cancel_fail_tip);
        }
        DemoUtil.showToast("cancel cycle conf failed: " + err);
    }
});

```

5.5.4.11 编辑周期子会议

接口描述

该接口用于编辑会议。

注意事项

- 会议未开始前可以调用接口，开始后调用则无效。
- 所有字段必填，编辑修改的字段可来源于用户输入，其他不想修改的属性一定要取会议详情对应字段来赋值。

方法定义

```
/**
 * 编辑周期子会议
 * @param modifySubCycleConfParam 编辑周期子会议参数
 * @param hwmCallback 结果回调
 */
void editSubCycleConf(ModifySubCycleConfParam modifySubCycleConfParam, SdkCallback<Void>
hwmCallback);
```

参数描述

表 5-44 参数说明

参数	是否必须	类型	描述
modifySubCycleConfParam	是	ModifySubCycleConfParam	周期子会议参数

表 5-45 ModifySubCycleConfParam 参数说明

参数	是否必须	类型	描述
editConfParam	是	EditConfParam	编辑会议参数
cycleConfParam	是	CycleConfParam	周期子会议参数

其余参数参见[参数说明](#)

示例代码

```
HWMBizSdk.getBizOpenApi().editSubCycleConf(modifySubCycleConfParam, new SdkCallback<Void>() {
    @Override
    public void onSuccess(Void v) {
        DemoUtil.showToast("Edit sub cycle conf success");
        doDismissDialogAndFragment();
        mListener.onEditSubCycleConfSuccess();
    }

    @Override
    public void onFailed(SDKERR retCode) {
        HCLog.i(TAG, "modifySubCycleConf onFailed" + retCode);
        String err = ErrorMessageFactory.create(retCode);
        if (TextUtils.isEmpty(err)) {
```



```

        err = DemoUtil.getResContext().getString(R.string.hwmconf_edit_failed);
    }
    DemoUtil.showToast("Edit sub cycle conf failed: " + retCode + ", desc: " + err);
    doDismissDialogAndFragment();
    }
});
});

```

5.5.4.12 取消周期子会议

接口描述

该接口用于取消未开始的周期子会议。

注意事项

周期子会议未开始前可以调用接口，开始后调用则无效。

方法定义

```

/**
 * 取消周期会议
 *
 * @param cycleConfId 周期会议Id
 * @param subCycleConfId 周期子会议Id
 * @param hwmCallback 结果回调
 */
void cancelSubCycleConf(String cycleConfId, String subCycleConfId, SdkCallback<Void> hwmCallback);

```

参数描述

表 5-46 参数说明

参数	是否必须	类型	描述
cycleConfId	是	String	周期会议ID
subCycleConfId	是	String	周期子会议ID

示例代码

```

HWMBizSdk.getBizOpenApi().cancelSubCycleConf(cycleConfId, item.getConfId(), new SdkCallback<Void>()
{
    @Override
    public void onFailed(SDKERR retCode) {
        HLog.i(TAG, "cancel sub cycle conf onFailed " + retCode);
        String err = ErrorMessageFactory.create(retCode);
        if (TextUtils.isEmpty(err)) {
            err = DemoUtil.getResContext().getString(R.string.hwmconf_cancel_fail_tip);
        }
        DemoUtil.showToast("cancel sub cycle conf failed: " + err);
        confListAdapter.notifyDataSetChanged();
    }

    @Override
    public void onSuccess(Void aVoid) {
        DemoUtil.showToast("cancel sub cycle conf success");
        updateConfDetail(cycleConfId);
    }
});
}

```

5.5.5 会议控制

5.5.5.1 创建会议

```
HWMSdk.getOpenApi.createConf(CreateConfParam, HwmCancelableCallBack )
```

接口描述

该接口用于创建即时会议并立即加入会议。

注意事项

该接口会申请应用程序的audio和camera权限。

方法定义

```
/**
 * 创建会议接口
 * @param createConfParam 会议配置实体
 * @param hwmCallback 会议创建回调
 */
void createConf(CreateConfParam createConfParam , HwmCancelableCallBack <ConfInfo> hwmCallback);
```

参数描述

表 5-47 参数说明

参数	是否必须	类型	描述
createConfParam	是	CreateConfParam	创建会议参数实体
hwmCallback	是	HwmCancelableCallBack	会议创建结果回调

返回值

无。

参数拓展

表 5-48 CreateConfParam 参数设置说明

参数	是否必须	类型	描述
setSubject	是	String	会议主题
setConfType	是	ConfType	CONF_AUDIO(0, "语音会议"), CONF_VIDEO(1, "视频会议"),
setNeedPassword	是	boolean	会议是否需要密码 说明 仅对随机ID会议生效。

参数	是否必须	类型	描述
guestPwd	否	String	来宾密码，如果不填则来宾密码由服务端生成 说明 仅对随机ID会议生效。
setAttendeeMembers	否	List<AttendeeModel>	会议参加者集合
setCameraOn	否	boolean	是否开启摄像头
setMicOn	否	boolean	是否开启麦克风
setRecordOn	否	boolean	会议录制功能是否开启 说明 仅对云录制生效，客户端本地录制不生效。
setAutoRecord	否	boolean	会议开始后是否自动开启录制，默认false。开启自动录制会议时必须开启录制会议 说明 仅对云录制生效，客户端本地录制不生效。
setVmrlId	否	String	云会议室id
setJoinConfRestrictionType	否	JoinConfPermissionType	PERMIT_EVERYONE("PERMIT_EVERYONE", 0, "允许所有人入会") PERMIT_ENTERPRISE_USER("PERMIT_ENTERPRISE_USER", 2, "仅允许企业内人员入会") PERMIT_INVITED_USER("PERMIT_INVITED_USER", 3, "仅允许会议邀请人员入会")

示例代码

```
List<AttendeeModel> attendeeList = new ArrayList<>();
//以下三种方式可选
String name = "huawei123456";
String number = "+99123456xxx";
String thirdAccountId = "123456abcd";
if (!TextUtils.isEmpty(thirdAccount)) {
    if (LoginStatusCache.getLoginAccountInfo() instanceof AppldAuthParam) {
        //如果是appid的方式登录
        attendeeList.add(AttendeeModel.buildAttendeeByThirdAccountId(thirdAccount,name));
    }
} else if(number.startsWith("+99")){
    //sip 号码入会
    attendeeList.add(AttendeeModel.buildAttendeeBySipNumber(number, name));
} else {
    //电话号码入会（手机和座机）
    attendeeList.add(AttendeeModel.buildAttendeeByPhone(number, name));
}
```

```

CreateConfParam createConfParam = new CreateConfParam()
    .setSubject("huawei123456的会议") //会议主题
    .setConfType(ConfType.CONF_AUDIO) //会议类型
    .setVmrlId(2c9183e07562e60b0175647ea92f59b3) //云会议室id（非必填字段）
    .setJoinConfRestrictionType(JoinConfPermissionType.PERMIT_EVERYONE) //设置允许入会范围限制
    .setNeedPassword(true) //是否创建密码
    .setCameraOn(true) //摄像头是否开启
    .setMicOn(true) //麦克风是否开启
    .setRecordOn(true) //会议录制功能是否开启
    .setAttendeeMembers(attendeeList); //添加与会人信息

HWMSdk.getOpenApi(getActivity()).createConf(createConfParam, new
HwmCancelableCallBack<ConfInfo>() {

    //创建会议成功回调
    @Override
    public void onSuccess(ConfInfo confInfo) {
        Log.i(TAG, "创建会议成功: 会议id: " + confInfo.getConfId() +
            ";会议密码: " + confInfo.getConfPwd());
    }

    //创建会议失败回调
    @Override
    public void onFailed(int retCode, String desc) {
        dismissLoading();
        String err = ErrorMessageFactory.create(Utils.getApp(), retCode);
        if (TextUtils.isEmpty(err)) {
            err =
Utils.getApp().getString(com.huawei.hwmmobileconfui.R.string.conf_create_error);
        }
        Log.e(TAG, "创建会议失败: " + err);
    }
    @Override
    public void onCancel() {
    }
});

```

5.5.5.2 加入会议

```
HWMSdk.getOpenApi.joinConf(JoinConfParam, HwmCancelableCallBack)
```

接口描述

该接口可以在登录和未登录状态下加入会议。加入会议的方式有两种：

- 会议ID+会议密码入会：手动输入会议ID和会议密码密码。
- 会议ID+Random入会：通过邮件会议通知中的会议链接入会时（会议链接如：<https://bmeeting.huaweicloud.com/#/j/981924330/e11ddec62ee760734fcf2ba55b79937eac2aa68677caf659>），链接里面的是会议ID（即举例中的981924330）和会议的Random码（即举例中的e11ddec62ee760734fcf2ba55b79937eac2aa68677caf659）。

注意事项

- 在未登录状态下调用该接口时，会议昵称参数必选。
- ID入会参数使用子类**PasswordJoinConfParam**，random入会参数使用子类**RandomJoinConfParam**

方法定义

```

/**
 * @param joinConfParam 加入会议参数实体
 * @param hwmCallback 加入会议结果回调
 */
void joinConf(JoinConfParam joinConfParam, HwmCancelableCallBack<Void> hwmCallback);

```

参数描述

表 5-49 参数说明

参数	是否必须	类型	描述
joinConfParam	是	JoinConfParam	加入会议参数实体
hwmCallback	是	HwmCancelableCall Back	会议加入结果回调

返回值

无。

参数拓展

表 5-50 JoinConfParam 参数说明

参数	是否必须	类型	描述
setConfId	是	String	设置加入会议的ID
setNickname	是	String	设置昵称
setCameraOn	否	boolean	是否开启摄像头
setMicOn	否	boolean	是否开启麦克风

表 5-51 PasswordJoinConfParam 参数说明

参数	是否必须	类型	描述
password	否	String	设置加入会议的密码，会议无密码时不填
isStopConflictConf	否	boolean	强制结束和当前VMR会议资源冲突的会议

表 5-52 RandomJoinConfParam 参数说明

参数	是否必须	类型	描述
setRandom	是	String	设置加入会议的随机数，由会议密码映射生成

示例代码

```
JoinConfParam joinConfParam;
if (!TextUtils.isEmpty(random)) {
```

```

        joinConfParam = new RandomJoinConfParam()
            .setRandom(random)
            .setConfId("987654321")
            .setNickname("huawei123")
            .setCameraOn(true)
            .setMicOn(true);
    } else {
        joinConfParam = new PasswordJoinConfParam()
            .setPassword(password)
            .setConfId("987654321")
            .setNickname("huawei123")
            .setCameraOn(true)
            .setMicOn(true);
    }
}
HWMSdk.getOpenApi(getActivity()).joinConf(joinConfParam, new HwmCancelableCallBack<Void>() {

    //加入会议成功回调
    @Override
    public void onSuccess(Void ret) {
        Log.i(TAG, "加入会议成功");
    }

    //加入会议失败回调
    @Override
    public void onFailed(int retCode, String desc) {
        String err = ErrorMessageFactory.create(Utils.getApp(), retCode);
        if (TextUtils.isEmpty(err)) {
            err = Utils.getApp().getString(com.huawei.hwmmobileconfui.R.string.conf_join_fail_tip);
        }
        Log.e(TAG, "加入会议失败: " + err);
    }
    @Override
    public void onCancel() {

    }
});

```

5.5.5.3 添加与会者

```
HWMSdk.getOpenApi.addAttendee(List<AttendeeModel> attendees, HwmCallback<Integer> hwmCallback);
```

接口描述

该接口用于在会议中添加与会者。

注意事项

1. 必须在会议中，并且只有主持人才可以加入；

方法定义

```

/**
 * 会中添加与会者
 * @param attendees 添加的与会者列表
 * @param hwmCallback 结果回调
 */
void addAttendee(List<AttendeeModel> attendees, HwmCallback<Integer> hwmCallback);

```

参数描述

表 5-53 AttendeeModel 参数说明

参数	是否必须	类型	描述
name	是	String	会场名。

参数	是否必须	类型	描述
number	是	String	号码。如果填分配给帐号的sip号码（如+99111244216210249）则呼叫该帐号的App；如果填PSTN号码（如18700000000），则通过VoIP网关呼叫该号码，前提是该企业已开通PSTN呼叫权限。与thirdUserId二选一，帐号密码鉴权时使用。
thirdUserId	是	String	第三方用户ID。与number二选一，App ID鉴权时使用。

示例代码

```
HWMBizSdk.getBizOpenApi().addAttendee(List<AttendeeModel> attendees, new HwmCallback<Integer>() {
    @Override
    public void onSuccess(Integer integer) {
        Log.i(TAG, "添加与会者成功");
    }

    @Override
    public void onFailed(int retCode, String desc) {
        Log.i(TAG, "添加与会者失败 code: " + retCode);
    }
});
```

5.5.5.4 离开会议

```
HWMSdk.getOpenApi.leaveConf(HwmCallback<Integer> callback)
```

接口描述

该接口用于主动离开会议。

注意事项

- 不在会议中不应调用此接口。
- 调用该接口离开会议，会议不会结束，在会议存在期间可以再次入会。

方法定义

```
/**
 * 离开会议
 * @param callback 结果回调
 */
void leaveConf(HwmCallback<Integer> callback);
```

参数描述

无。

返回值

无

示例代码

```
HWMSdk.getOpenApi(HWMBizSdk.getApplication()).leaveConf(new HwmCallback<Integer>() {
    @Override
```

```

public void onSuccess(Integer integer) {
    Toast.makeText(HWMBizSdk.getApplication(), "已成功离开会议", Toast.LENGTH_SHORT)
        .show();
}

@Override
public void onFailed(int retCode, String desc) {
    Toast.makeText(HWMBizSdk.getApplication(), "自定义离开会议失败: " + retCode + " " + desc,
        Toast.LENGTH_SHORT)
        .show();
}
});

```

5.5.5.5 结束会议

```
HWMSdk.getOpenApi.endConf(HwmCallback<Integer> callback)
```

接口描述

该接口用于主持人主动结束会议。

注意事项

1. 不在会议中不应调用此接口。
2. 只有主持人可以结束会议。
3. 调用该接口，会议将结束，无法再次加入该会议。

方法定义

```

/**
 * 结束会议
 * @param callback 结果回调
 */
void endConf(HwmCallback<Integer> callback);

```

参数描述

无。

返回值

无

示例代码

```

HWMSdk.getOpenApi(HWMBizSdk.getApplication()).endConf(new HwmCallback<Integer>() {
    @Override
    public void onSuccess(Integer integer) {
        Toast.makeText(HWMBizSdk.getApplication(), "已成功结束会议", Toast.LENGTH_SHORT)
            .show();
    }

    @Override
    public void onFailed(int retCode, String desc) {
        Toast.makeText(HWMBizSdk.getApplication(), "自定义结束会议失败: " + retCode + " " + desc,
            Toast.LENGTH_SHORT)
            .show();
    }
});

```

5.5.5.6 设置共享权限

setShareScreenPermission

```
HWMBizSdk.getBizOpenApi().setShareScreenPermission(boolean hasPermission)
```


接口描述

该接口用于设置共享权限。

注意事项

该接口返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回

方法定义

```
/**
 * 设置本端是否有共享权限
 * @param hasPermission
 */
void setShareScreenPermission(boolean hasPermission);
```

参数描述

表 5-54 参数说明

参数	是否必须	类型	描述
hasPermission	是	boolean	是否允许。

返回值

无

示例代码

```
HWMBizSdk.getBizOpenApi().setShareScreenPermission(true);
```

5.5.6 获取会议信息

5.5.6.1 获取与会者列表

getAttendeeList

接口描述

该接口用于获取会议中与会者列表。

注意事项

1. 只能在会议中使用

方法定义

```
/**
 * 获取所有与会者列表
 * @return 与会者列表
 */
List<AttendeeModel> getAttendeeList();
```

参数描述

参数说明见[AttendeeModel](#)。

表 5-55 枚举 ConfAttendeeState 说明

参数	描述
CONF_ATTENDEE_STATE_IN_CONF	会议中
CONF_ATTENDEE_STATE_CALLING	正在呼叫
CONF_ATTENDEE_STATE_LEFT	已经离开

示例代码

```
HWMBizSdk.getBizOpenApi().getAttendeeList();
```

5.5.6.2 获取观众列表

getAudienceList

接口描述

该接口用于获取会议中观众列表。

注意事项

1. 只能在会议中使用
2. 观众身份（可以通过会议详情消息推送获取）不能使用此接口

方法定义

```
/**
 * 获取所有观众列表
 * @return 观众列表
 */
List<AttendeeModel> getAudienceList();
```

参数描述

参数说明见[AttendeeModel](#)。

示例代码

```
HWMBizSdk.getBizOpenApi().getAudienceList();
```

5.5.6.3 获取用户状态

```
HWMSdk.getOpenApi().getHWMUserState();
```

接口描述

该接口用于获取用户当前的状态(是否在呼叫/会议中)。

注意事项

无。

方法定义

```
/**
 * 获取用户当前的状态(是否在呼叫/会议中)
 */
HWMUserState getHWMUserState();
```

参数描述

无。

返回值

表 5-56 HWMUserState 枚举说明

枚举值	描述
HWM_STATE_IDLE	空闲状态
HWM_STATE_IN_CONF	在会议中
HWM_STATE_IN_CALL	在呼叫中

示例代码

```
public void getHWMUserState() {
    HWMUserState hwmUserState =
    HWM SDK.getOpenApi(HWMBizSdk.getApplication()).getHWMUserState();
    DemoUtil.showToast("user state: " + hwmUserState.getDescription());
}
```

5.5.7 音视频数据流

5.5.7.1 开启第三方视频推流配置

在OpenSDKConfig中配置视频流的第三方采集，sdk自身不再采集视频流

共有三个回调方法

onSetLocalVideoMirrorTypeNotify: 会议开始时回调，用户可以在此设置镜像

onStartPushExternalVideoFrameNotify: 会议开始时回调，用户必须要在实现自己的数据采集并进行推流

onEndPushExternalVideoFrameNotify: 会议结束时回调，用户可在此停止数据采集

使用场景

对于非标准手机摄像头采集视频流或者视频流需要额外处理的场景

注意事项

1.SDK初始化的时候必须传入这个配置sdkConfig才会生效

示例代码

```
public class CustomPushExternalVideoFrameStrategy implements IPushExternalVideoFrameStrategy {
    private static CopyOnWriteArrayList<IPushVideoFrameNotification> pushVideoFrameNotifications = new
    CopyOnWriteArrayList<>();
    public static void registerPushVideoFrameListener(IPushVideoFrameNotification confResultNotification) {
```

```

        pushVideoFrameNotifications.add(confResultNotification);
    }

    public static void removePushVideoFrameListener(IPushVideoFrameNotification confResultNotification) {
        pushVideoFrameNotifications.remove(confResultNotification);
    }

    @Override
    public void onSetLocalVideoMirrorTypeNotify() {
        //设置本地视频窗口镜像模式(需要在UI主线程调用) 设置第三方推流视频镜像
    }

    @Override
    public void onStartPushExternalVideoFrameNotify() {
        //开始第三方采集推流
    }

    @Override
    public void onEndPushExternalVideoFrameNotify() {
        //停止第三方采集推流
    }
}
sdkConfig.setPushExternalVideoFrameStrategy(new CustomPushExternalVideoFrameStrategy());

```

5.5.7.2 设置第三方推流视频镜像

接口描述

该接口用于第三方推流时设置本地视频窗口前置摄像头镜像模式。

注意事项

1. 需要放在UI主线程调用，使用前置摄像头设置左右镜像类型，后置摄像头不能镜像。

方法定义

```

/**
 * 设置本地视频窗口镜像模式
 * @param mirrorType 窗口镜像模式 0:不做镜像(默认值) 1:上下镜像(目前未支持) 2:左右镜像
 */
void setLocalVideoMirrorType(MirrorType mirrorType);

```

参数描述

表 5-57 参数说明

参数	是否必须	类型	描述
mirrorType	是	MirrorType	窗口镜像模式 MIRROR_NONE(0, "不做镜像") MIRROR_UP_DOWN(1, "上下镜像")(目前未支持) MIRROR_LEFT_RIGHT(2, "左右镜像")

示例代码

```
HWMBizSdk.getBizOpenApi().setLocalVideoMirrorType(MirrorType.MIRROR_LEFT_RIGHT);
```

5.5.7.3 第三方视频推流

接口描述

对于自定义视频流采集的场景，采集到数据流之后，调用该接口进行第三方推流。

注意事项

1. 会议或者呼叫中调用，会前或者会后调用无效。

方法定义

```
/**
 * 第三方采集推流
 * @param videoFrameParam 视频流参数
 * @param callback 结果回调
 */
void pushExternalVideoFrame(VideoFrameParam videoFrameParam, HwmCallback<Void> callback);
```

参数描述

表 5-58 参数说明

参数	是否必须	类型	描述
videoFrameParam	否	VideoFrameParam	视频流参数
callback	否	HwmCallback	结果回调

返回值

无。

参数扩展

表 5-59 VideoFrameParam 参数说明

参数	是否必须	类型	描述
frameDate	是	byte[]	视频数据
frameWidth	是	int	图像宽度，作为输入时，范围为[90,1920]，必须是4的整数倍
frameHeight	是	int	图像高度，作为输入时，范围为[90,1200]，必须是2的整数倍

参数	是否必须	类型	描述
videoFrameFormat	是	VideoFrameFormat	图像格式，当前demo使用格式为： VIDEO_FRAME_FORMAT_NV21，转换后需设置为105 VIDEO_FRAME_FORMAT_I420(100, "I420") VIDEO_FRAME_FORMAT_YV12(101, "YV12") VIDEO_FRAME_FORMAT_NV21(105, "NV21") VIDEO_FRAME_FORMAT_UNKNOWN(999, "UNKNOWN")
displayRotation	是	DisplayRotation	视频显示的角度 ROTATION_0(0, "顺时针旋转0°") ROTATION_90(1, "顺时针旋转90°") ROTATION_180(2, "顺时针旋转180°") ROTATION_270(3, "顺时针旋转270°")

示例代码

```
ExternalVideoFrameParam videoFrameParam = new ExternalVideoFrameParam();
videoFrameParam.setFrameDate(mData);
videoFrameParam.setFrameWidth(1920);
videoFrameParam.setFrameHeight(1080);
videoFrameParam.setDisplayRotation(DisplayRotation.ROTATION_0);
videoFrameParam.setVideoFrameFormat(VideoFrameFormat.VIDEO_FRAME_FORMAT_NV21);
HWMBizSdk.getBizOpenApi().pushExternalVideoFrame(videoFrameParam, new HwmCallback<Void>() {
    @Override
    public void onSuccess(Void aVoid) {
    }

    @Override
    public void onFailed(int retCode, String desc) {
        Log.i(TAG, "code: " + retCode);
    }
});
```

5.5.8 通知参考

5.5.8.1 自定义帐号通知事件

在OpenSDKConfig中配置登录账号相关通知回调的监听，以监测账号失效等场景

使用场景

账号被踢: onKickedOut

Token失效: onTokenInvalid

账号被锁定: onAccountLocked

账号密码错误: onAccountOrPasswordError

注意事项

SDK初始化的时候必须传入这个配置sdkConfig才会生效

示例代码

```

sdkConfig.setAccountEventHandle(new IAccountEventHandle() {
    @Override
    public void onKickedOut(@Nullable KickOutState kickOutState) {
        Log.i(TAG, "您的帐号在其他终端登录");
    }
    @Override
    public void onAccountOrPasswordError(@Nullable AccountOrPasswordErrorState
accountOrPasswordErrorState) {
        Log.i(TAG, "onAccountOrPasswordError");
    }
    @Overridepublic void onTokenInvalid(@Nullable TokenInvalidState tokenInvalidState) {
        // Token失效后, 需要进行重新登录(退出并登录), 否则会影响创会预约会议等基本功能
        Log.i(TAG, "onTokenInvalid");
    }
    @Override
    public void onAccountLocked(@Nullable AccountLockedState accountLockedState) {
        Log.i(TAG, "onAccountLocked");
    }
});

```

5.5.8.2 自定义会中 UI 行为通知

在OpenSdkConfig中配置会议UI页面变更相关的通知监听

入会的时候UI创建, 离会的时候销毁UI

使用场景

入会场景: onConfDetailNotify

离会场景: onConfEnded、onLeaveConf

点对点呼叫结束场景: onCallEnded

注意事项

SDK初始化的时候必须传入这个配置sdkConfig才会生效

示例代码

```

public class CustomNotifyHandler implements CLMNotifyHandler {
    /**
     * 呼叫结束后, 会收到该通知
     * @param callInfo 呼叫信息
     */
    @Override
    public void onCallEnded(CallInfo callInfo) {
        Log.i(TAG, "呼叫结束了");
    }

    @Override
    public void onConfDetailNotify(ConfInfo confInfo) {

```

```

        Log.i(TAG, "confInfo: confId" + confInfo.getConfId() + ", Subject:" + confInfo.getConfSubject());
    }

    /**
     * 会议结束后，会收到通知
     * @param result 会议结束后的错误码
     */
    @Override
    public void onConfEnded(ConfEndedReason result) {
        Log.i(TAG, "会议已结束");
    }

    @Override
    public void onLeaveConf(int result) {
        Log.i(TAG, "您已离开会议");
    }
}
sdkConfig.setNotifyHandler(new CustomNotifyHandler());

```

离开会议通知

接口描述

主动离会、加入会议失败、重呼失败，收到该通知。

方法定义

```

/**
 * 离开会议的通知
 */
void onLeaveConf(int result);

```

参数描述

表 5-60 参数说明

参数	是否必须	类型	描述
result	是	int	离会原因 Constants.LEAVE_CONF_MODE.LEAVE_NORMAL = 7; //正常离会 Constants.LEAVE_CONF_MODE.LEAVE_OF_NOSTREAM = 8; //无码流重连后离会 Constants.LEAVE_CONF_MODE.LEAVE_OF_OUTNET = 9; //断网重连后离会 Constants.LEAVE_CONF_MODE.LEAVE_OF_TLS_UNESTABLISH = 10; //tls闪断 Constants.LEAVE_CONF_MODE.LEAVE_OF_OUTNET_TIMEOUT = 14; //断网重连超时离会

返回值

无。

会议结束通知

接口描述

会议结束收到该通知。

方法定义

```
/**
 * 会议结束后，会收到通知
 *
 * @param reason 会议结束后的错误码
 */
void onConfEnded(ConfEndedReason reason);
```

参数描述

表 5-61 参数说明

参数	是否必须	类型	描述
reason	是	ConfEndedReason	结束错误码枚举

返回值

无。

参数拓展

表 5-62 枚举 ConfEndedReason 说明

参数	类型	描述
LEAVE_CONF_HANGUP	enum	离开会议
STOP_CONF_HANDUP	enum	会议已经结束
REMOVE_BY_CHAIRMAN_HANDUP	enum	被主持人移出会场
TIMEOUT	enum	网络中断
NO_STREAM_HANDUP	enum	网络异常
CORP_CONFERENCE_RESOURCE_HAS_BEEN_RUN_OUT	enum	企业资源已达上限
UNKONWN	enum	未知原因

会议详情通知

接口描述

获取会议详情收到该通知，入会的时候会收到一次。

方法定义

```
/**
 * 会议详情的上报
 *
 * @param confInfo
 */
void onConfDetailNotify(ConfInfo confInfo);
```

参数描述

表 5-63 参数说明

参数	是否必须	类型	描述
confInfo	是	ConfInfo	会议信息

返回值

无。

参数拓展

表 5-64 ConfInfo 参数说明

参数	是否必须	类型	描述
confSubject	是	String	会议主题
confId	是	String	会议ID
vmrConferen ceId	是	String	个人会议ID
confPwd	否	String	入会密码
confAccessN um	是	String	接入号
confChairPw d	是	String	主持人密码
confGuestUri	是	String	guest入会uri
confStartTim e	是	String	会议开始时间
confEndTime	是	String	会议结束时间

参数	是否必须	类型	描述
confSchedulerName	是	String	创建者名字
confOrgId	是	String	判断是否外部用户
mediaType	是	int	会议类型 CONF_MEDIA_TYPE_AUDIO = 1; 音频 CONF_MEDIA_TYPE_VIDEO = 2; 视频 CONF_MEDIA_TYPE_HDVIDEO = 4; 高清视频 CONF_MEDIA_TYPE_DATA = 16; 数据

呼叫结束通知

接口描述

点对点呼叫结束收到该通知。

方法定义

```
/**
 * 呼叫结束后，会收到该通知
 *
 * @param callInfo 呼叫信息
 */
void onCallEnded(CallInfo callInfo);
```

参数描述

表 5-65 参数说明

参数	是否必须	类型	描述
callInfo	是	CallInfo	呼叫信息

返回值

无。

参数拓展

表 5-66 CallInfo 参数说明

参数	是否必须	类型	描述
peerInfo	是	PeerInfo	点对点呼叫对方信息
startTime	是	long	开始时间
endTime	是	long	结束时间
callType	是	int	呼叫类型： <ul style="list-style-type: none"> 0: 语音呼叫 1: 视频呼叫
callState	否	int	主被叫状态： <ul style="list-style-type: none"> 1: 被叫 2: 主叫
isCallSuccess	是	boolean	是否成功建立呼叫
missReason	是	int	呼叫未成功建立的原因： <ul style="list-style-type: none"> 0: 被叫未接听 1: 被叫拒接
isEncryptCall	否	boolean	音视频数据是否加密
isCalleeBusy	是	boolean	被叫方是否忙碌

表 5-67 PeerInfo 参数说明

参数	是否必须	类型	描述
peerName	是	String	对方成员名称
peerNumber	是	String	对方成员号码
peerAccount	否	String	对方成员账号
peerUuid	否	String	对方成员的uuid
peerThirdAccount	否	String	第三方账号

5.5.8.3 业务消息通知

在OpenSDKConfig中配置业务类通知消息处理的接口来监听会议信息的变更

使用场景

监听会议列表变更

监听自己在会中的主持人角色变更

监听自己在会中的录制权限变更

监听自己预约会议是否有发送短信权限的变更

注意事项

SDK初始化的时候必须传入这个配置sdkConfig才会生效

示例代码

```
public class CustomBizNotficationHandle implements BizNotificationHandler {
    private static CopyOnWriteArrayList<IConfListUpdate> confListUpdates = new CopyOnWriteArrayList<>();

    public static void registerConfListUpdateListener(IConfListUpdate confListUpdate){
        confListUpdates.add(confListUpdate);
    }

    public static void removeConfListUpdateListener(IConfListUpdate confListUpdate){
        confListUpdates.remove(confListUpdate);
    }

    //会议列表更新回调
    @Override
    public void onConfListChanged(List<ConfBaseInfo> list) {
        for (IConfListUpdate listener : confListUpdates) {
            try {
                listener.onConfListUpdateNotify(list);
            } catch (RuntimeException e) {
                Log.e("onConfListChanged", e.toString());
            }
        }
    }

    //个人角色是否是主席更新回调
    @Override
    public void onIsSelfChairManChanged(boolean isChairman) {
    }

    //会议录制权限通知
    @Override
    public void onRecordPermissionChanged(boolean hasPermission) {
    }

    //预约会议发送短信权限通知
    @Override
    public void onBookConfSmsPermissionChanged(boolean hasPermission) {
    }

    //是否支持AI录制
    @Override
    public void onAIConfRecordStateChanged(int state) {
    }

    // 是否有云上录播权限
    @Override
    public void onRecordTypeChanged(int type) {
    }

    // 会议来电信息
    @Override
    public void onConfIncomingNotify(ConfIncomingInfo confIncomingInfo) {
    }
}
sdkConfig.setBizNotificationHandler(new CustomBizNotficationHandle());
```

个人角色变化

接口描述

收到该通知时，表示个人角色发生变化，是否是主持人身份。

方法定义

```
/**
 * 个人角色是否主持人变化通知
 * @param isSelfChairMan
 */
void onIsSelfChairManChanged(boolean isSelfChairMan);
```

参数描述

表 5-68 参数说明

参数	类型	描述
isSelfChairMan	boolean	个人角色是否是主持人

返回值

无。

会议列表变化

接口描述

收到该通知时，表示会议列表发生变化。

方法定义

```
/**
 * 会议列表变化通知
 *
 * @param hwmConfList
 */
void onConfListChanged(List<ConfBaseInfo> hwmConfList);
```

参数描述

参数说明参见[会议列表参数描述](#)

录制会议权限更新

接口描述

收到该通知时，表示是否有录制会议权限。

方法定义

```
/**
 * 录制会议权限更新通知
 *
 * @param hasPermission
 */
void onRecordPermissionChanged(boolean hasPermission);
```

参数描述

表 5-69 参数说明

参数	类型	描述
hasPermission	boolean	是否有录制会议权限

返回值

无。

预约会议发送短信权限更新

接口描述

收到该通知时，表示预约会议发送短信权限更新，是否可以发送短息。

方法定义

```
/**
 * 预约会议发送短信权限更新通知
 *
 * @param hasPermission
 */
void onBookConfSmsPermissionChanged(boolean hasPermission);
```

参数描述

表 5-70 参数说明

参数	类型	描述
hasPermission	boolean	预约会议是否有发送短信通知权限

返回值

无。

5.5.8.4 自定义来电自动处理通知事件

通知描述

在OpenSDKConfig配置中自定义来电事件处理接口

当点对点呼叫或者会议邀请收到该来电通知时，可以自定义来电接听的自动处理方式。若不实现，则默认处理方式是显示接听页面。

使用场景

收到会议邀请的时候直接接听：answerTypeOnConfIncoming返回
HWMIncomingAnswerType.HWM_INCOMING_AUTO_ANSWER

收到会议邀请的时候直接挂断：answerTypeOnConfIncoming返回
HWMIncomingAnswerType.HWM_IMCOMING_AUTO_DECLINE

点对点呼叫类型场景以及其他场景

注意事项

SDK初始化的时候必须传入这个配置sdkConfig才会生效

示例代码

```

sdkConfig.setIncomingHandle(new IncomingHandle() {
    @Override
    public HWMIncomingAnswerType answerTypeOnCallIncoming() {
        // 自定义点对点呼叫的来电处理方式
        return HWMIncomingAnswerType.HWM_IMCOMING_NORMAL;
    }

    @Override
    public HWMIncomingAnswerType answerTypeOnConfIncoming() {
        // 自定义会议邀请的来电处理方式
        return HWMIncomingAnswerType.HWM_IMCOMING_NORMAL;
    }
});
    
```

参考返回值

```

public enum HWMIncomingAnswerType {
    HWM_IMCOMING_NORMAL(0, "HWM_IMCOMING_NORMAL"), //默认处理
    HWM_IMCOMING_AUTO_DECLINE(1, "HWM_IMCOMING_AUTO_DECLINE"), //自动拒接来电
    HWM_INCOMING_AUTO_ANSWER(2, "HWM_INCOMING_AUTO_ANSWER"); //自动接听
}
    
```

5.6 错误码参考

5.6.1 常见错误码

表1 常见错误码对照表

值	描述
68	网络不佳，发起/加入会议超时
91	会议不存在或者无权限访问
95	会议不允许提前入会，请联系主持人
103	会议不允许提前入会，请联系主持人
112	会议已结束
836	网络中断，呼叫已断开
837	呼叫已断开，无码流挂断
838	您已被主持人移出会议（主席挂断）
839	结束会议挂断
11070005	会议不存在

值	描述
11070009	会议鉴权信息校验失败
11071005	会议时间与下一个预定的会议冲突，如需继续请联系云会议室的管理员预约
11071014	会议开始时长超出最大可预定时间范围
11071025	请检查与会人手机号是否正确
11071067	当前会议正在召开，无法取消
11071073	您呼叫的会议不允许匿名用户呼入，请联系会议预订者
11071074	您呼叫的会议只允许企业内用户呼入，请联系会议预订者
11071075	您呼叫的会议只允许被邀请用户呼入，请联系会议预订者
11072031	请检查与会人手机号是否为空
11072033	请检查与会人入会号码是否正确
11072050	会议已经被锁定
11076003	云会议室同时在线人数已达上限，请联系会议主持人或管理员
11081006	密码错误，请重新输入
115022001	Enterprise concurrency resource not enough:企业并发资源不足

5.6.2 错误码表

错误码表

表 5-71 错误码说明

错误码	错误描述
-1	Unknown error.:未知错误
0	Success.:成功
111070001	Request error.:请求错误
111070002	Insufficient permissions.:权限不足
111070003	System processing failure.:系统处理失败
111070004	Meeting authentication failed.:校验失败

错误码	错误描述
111070005	Meeting does not exist.:会议不存在
111070006	Personal cloud meeting room does not exist.:个人云会议室不存在
111070007	Meeting access number does not exist.:会议接入号不存在
111070008	Meeting notification could not be sent.:会议通知模板配置错误
111070009	Meeting password cannot be left blank.:会议密码不能为空
111070010	Insufficient permissions to view this meeting.:权限不足,无法查看会议
111070100	Get started meeting exception.:获取召开会议异常
111070101	Start meeting exception.:开始会议异常
111070102	Apply IVR resource exception.:申请IVR资源异常
111070103	Start IVR meeting exception.:开始IVR会议异常
111070104	Apply site resource exception.:申请会场资源异常
111070105	Start sub meeting exception.:开始子会议异常
111070106	Add channel exception.:创建级联通道异常
111070107	Move and apply site resource exception.:移动会场申请资源异常
111070108	Start meeting exception.:开启会议异常
111070109	Get started cloud meeting exception.:获取召开云会议室会议异常
111070110	Invite site exception.:邀请会场异常
111070111	Request token is null.:请求token为空
111070112	Rename site exception.:重命名会场异常
111070113	Hangup site exception.:会场挂断异常
111070114	Mass hangup site exception.:会场批量挂断异常
111070115	Delete attendee exception.:删除与会者异常
111070116	Mass delete attendee exception.:批量删除与会者异常
111070117	Book cycle meeting exception.:预定周期会议异常
111070118	Modify cycle meeting exception.:修改周期会议异常
111070119	Create cycle meeting exception.:创建周期会议异常
111070120	Region data not found.:区域信息获取失败

错误码	错误描述
111070121	Forbidden subscribe.:禁止订阅
111070122	Cancel subscribe fail.:取消订阅失败
111071001	Sysportal service exception.:Sysportal服务运行异常
111071002	Conf service exception.:Conf服务运行异常
111071003	Failed to create the meeting. Try again later.:创建会议失败, 请稍后重试
111071004	This cloud meeting room does not allow guests to join before the host.:VMR会议不允许来宾在主席入会前入会
111071005	This cloud meeting room has been scheduled for another meeting during this period.:VMR会议时间冲突, 该时间段此会议室已有预约会议
111071006	Insufficient cloud meeting time.:VMR会议时长不足
111071007	Meeting does not exist.:VMR预约关系不存在
111071008	Cannot query meeting list.:查询会议列表失败
111071009	Cannot change cloud meeting into immediate meeting.:不支持修改VMR会议为立即会议
111071010	Cannot modify cloud meeting flag.:不支持修改云会议室标识
111071011	The cloud meeting has not yet started. Please wait.:VMR会议未到准备时间点
111071012	Not cloud meeting.:非VMR会议
111071013	Start time less than current time.:会议开始时间不能早于系统当前时间
111071014	Start time over max schedule time.:会议开始时间超出最大预订时间范围
111071015	Duration over range.:时长超出范围
111071016	Cycle ID list null exception.:取消周期会议ID列表不能为空
111071017	Meeting size over range.:与会方数超出范围
111071018	Invalid media type.:媒体类型非法
111071019	Invalid meeting type.:会议类型非法
111071020	Cycle params null exception.:周期会议时,必须携带周期会议参数
111071021	Duration over range.:会议时长超出范围
111071022	Subject length too long.:会议主题超长
111071023	Group URI length too long.:groupuri超长

错误码	错误描述
111071024	Invalid email address format.:邮箱地址不合法
111071025	Invalid mobile number format.:手机号不合法
111071026	Department name too long.:部门名称过长
111071027	Invalid welcome voice enable value.:入会欢迎词开关不在范围内
111071028	Invalid enter voice enable value.:入会提示音开关不在范围内
111071029	Invalid leave voice enable value.:离会提示音开关不在范围内
111071030	Invalid reminders value.:会议提醒非法
111071031	Invalid allow record value.:会议是否启用网络录制取值非法
111071032	Invalid auto record value.:会议是否自动启动录制取值非法
111071033	Invalid user type value.:会预定者的用户类型非法
111071034	Invalid encrypt mode value.:加密类型非法
111071035	Invalid language value.:语言非法
111071036	Invalid auto prolong value.:是否自动延长会议取值非法
111071037	VmrID too long.:vmrid超长
111071038	Invalid record type value.:录播类型非法
111071039	Live address too long.:主流直播地址超长
111071040	Aux address too long.:辅流直播地址超长
111071041	Null start date for cycle meeting.:周期会议时,起始日期为空
111071042	Null end date for cycle meeting.:周期会议时,结束日期为空
111071043	Invalid cycle type value for cycle meeting.:周期会议时,周期类型非法
111071044	Invalid interval for cycle meeting.:周期会议时,interval非法
111071045	Null point for cycle meeting.:周期会议时,point为空
111071046	Invalid point value for cycle meeting.:周期会议时,point非法
111071047	Invalid vmr flag value.:vmrFlag非法
111071048	Invalid record auxStream.:recordAuxStream非法
111071049	Invalid multiStream flag.:multiStreamFlag非法
111071050	Invalid audience flag.:audienceFlag非法
111071051	Invalid liveChannel ID.:liveChannelID不存在

错误码	错误描述
111071052	Anonymous login is locked. Try again later.:匿名登录锁定,请稍后重试
111071053	Link login is locked. Try again later.:链接登录锁定,请稍后重试
111071054	Not enough Meeting ID.:会议ID不足
111071055	Failed to apply meeting ID.:会议ID申请失败
111071056	Failed to release meeting ID.:会议ID释放失败
111071057	Live room is used.:直播房间已被占用
111071058	The meeting has not yet started. Please wait.:当前会议还未召开,请耐心等待
111071059	Meeting size over range.:与会方数超出上限
111071060	The enterprise cannot create meetings because it has not purchased any concurrent meeting resources.:企业未配置并发会议资源,无法使用个人会议ID和随机会议ID发起会议
111071061	Parameter verify failed.:参数校验失败
111071062	Json conversion failed.:JSON转换失败
111071063	Not cloud meeting room owner.:非云会议室拥有者
111071064	This cloud meeting room has been started for another meeting during this period.:VMR会议冲突,当前已有会议正在召开
111071065	This meeting cannot be modified because it is already started.:正在召开会议,无法编辑会议详情
111071066	Attendees amount over range.:与会方数量超出上限
111071067	This meeting cannot be canceled because it is already started.:会议正在召开中,不允许取消会议
111071068	Meeting password encrypt failed.:会议密码加密失败
111071069	No meeting during this period.:选定的时间中没有会议
111071070	Sub cycle meeting cannot modify live room.:周期会议子会议不允许修改直播房间
111071071	Stop online meeting exception.:结束会议失败
111071072	Meeting is ended.:会议已经结束
111071073	Not allow anonymous user.:会议不允许匿名用户呼入
111071074	Only allow user in organization.:会议只允许企业内用户呼入
111071075	Only allow invited user.:会议只允许被邀请用户呼入

错误码	错误描述
111071076	Batch export task for history meeting list during this period.:当前有正在执行的批量导出任务
111071077	Only allow invited user and user in organization.:会议只允许被邀请或者企业内用户用户呼入
111071078	Max current participants amount reached.:企业单会议并发方数已达上限
111071079	Invalid CTD param.:ctd会议参数不合法
111071080	Invalid guest password.:来宾密码不合法
111071081	Guest password conflict with meeting ID.:来宾密码与会议ID冲突
111071082	Max meeting amount reached.:会议数达到上限
111071083	Invalid meeting random number.:无效的会议随机数
111071084	Max export history meeting list times reached.:导出历史会议次数超出允许的值
111071085	Max multi view layouts reached.:会议多画面布局超出限制
111071086	Max multi view invalid manual type.:会议多画面模式非法
111071087	Max multi view sub view amount reached.:会议多画面子画面个数超出限制
111071088	Max multi view sub view numbers reached.:会议多画面子画面会场数超出限制
111071089	Webinar and common meeting cannot be converted to each other.:网络研讨会不能修改为普通会议,普通会议也不能修改为网络研讨会
111071090	Not allow add audience.:不支持添加观众
111071091	Meeting password authentication failed.:密码错误
111071092	File export task failed.:导出文件任务失败
111071093	File export task is downloading.:导出文件正在处理中
111071094	Not allow audience to join.:会议不允许观众入会
111071095	Need to get WEBINAR_CONF.DTO object.:需要获取WEBINAR_CONF.DTO对象
111071096	User-defined languages over range.:自定义语言个数超出范围
111071097	Your enterprise does not support simultaneous interpretation.:企业不支持同声传译
111071098	Need to set interpreter groups.:需要设置传译组信息

错误码	错误描述
111071099	Invalid interpretation language.:传译组语言非法
111071100	Max languages in a meeting are reached.:单会议最大支持16种语言
111071101	This interpreter group ID already exists.:传译组的组号重复
111071102	Max interpreters in the interpreter group reached.:传译组中传译员的个数达到上限
111071103	This interpreter already exists in an interpreter group.:传译员重复
111071104	Invalid interpreter account.:传译员帐号不存在
111071105	Guest password conflict with chairman password.:来宾密码与主席密码冲突
111071106	Meeting is holding.:云会议室已有会议正在召开
111071107	Invalid audience password.:观众密码不合法
111071108	Audience password conflict with meeting id.:观众密码与会议ID冲突
111071109	Audience password conflict with chairman password.:观众密码与主席密码冲突
111071110	Only RTC meeting support E2E encrypt.:非RTC会议不支持端到端加密
111071111	Audience password conflict with guest password.:观众密码与来宾密码冲突
111071112	Cloud meeting room ID type cannot be modified.:不支持修改会议ID类型
111071113	Cloud meeting room type cannot be modified.:不支持修改VMR类型
111071114	Cloud meeting room flag cannot be modified.:不支持修改VMR标识
111071119	Sub cycle meeting modify time over range.:周期会议修改时间超出范围
111072001	Application for meeting resources failed.:申请资源失败
111072002	Failed to release meeting resources.:释放资源失败
111072003	The meeting is ending.:会议正在结束
111072004	The meeting is starting.:会议正在开始
111072005	The meeting has not yet started.:会议还未开始
111072006	The meeting has ended.:会议已经结束

错误码	错误描述
111072007	Site invitation failed.:邀请会场失败
111072008	Site does not exist.:会场不存在
111072009	Failed to disconnect the site.:挂断会场失败
111072010	Coming soon...:功能暂未支持,敬请期待...
111072011	Meeting status incorrect.:会议状态非法
111072012	Site control operation failed.:会场会控操作失败
111072013	The cloud meeting room is in use by another meeting.:当前VMR会议非目标会议
111072014	The meeting has ended.:VMR会议已过期
111072015	The meeting has not yet started.:VMR会议还未开始
111072016	Meeting does not exist.:VMR预约记录不存在
111072017	Meeting has no attendee.:会议与会者不存在
111072018	Application for chairman password exception.:申请主席密码错误
111072019	Not found site which can be moved exception.:没有找到可以移动的会场
111072020	The current meeting is not a recording meeting. Recording is not supported.:非录播会议不支持录制
111072021	System processing failure.:录播调度失败
111072022	Insufficient recording resources in the system.:录播资源不足
111072023	There are no participants in this meeting.:会议中没有与会人
111072024	Operation failed. No sites have joined this meeting.:请确保会议中已有会场入会
111072025	Voice meeting does not support this function.:语音会议不支持该功能
111072026	The voice site does not support this operation.:语音会场不支持该操作
111072027	Voice sites cannot be displayed in custom continuous presence.:手动设置的多画面中无法显示语音会场
111072028	At least one site must have custom continuous presence configured.:手动设置的多画面中需最少设置一个会场
111072029	At most one presentation pane is allowed in custom continuous presence.:手工设置的多画面最多只允许一个辅流子画面

错误码	错误描述
111072030	In the voice-activated continuous presence mode, each pane can only support a single user.:声控多画面模式下,每个子画面最多只允许一个用户
111072031	The number used for the site to join the meeting cannot be left blank.:与会者的入会号码不允许为空
111072032	The number used for the site to join the meeting is too long.:与会者的入会号码长度超过系统允许的最大值
111072033	The format of the number used for joining the meeting is incorrect. (Correct format: User name name/IP address:Port number or Country code plus phone number):与会者的入会号码格式非法(支持SIP、TEL号码格式)
111072034	Length of site name too long.:与会者的名称长度超过系统允许的最大值
111072035	The site role type is outside the allowed range.:与会者的角色类型不在系统允许的取值范围内
111072036	Length of site email address too long.:与会者的邮箱地址长度超过系统允许的最大值
111072037	Length of site mobile number too long.:与会者的手机号码长度超过系统允许的最大值
111072038	The value for whether a site is muted after it first joins the meeting is outside the allowed range.:与会者入会后是否静音的取值不在系统允许的取值范围内
111072039	The value for whether to automatically invite a site is outside the allowed range.:与会者是否自动邀请的取值不在系统允许的取值范围内
111072040	The host's site viewing type is outside the allowed range.:主席选看类型取值不在系统允许的取值范围内
111072041	The value of the continuous presence setting mode is outside the allowed range.:设置多画面方式取值不在系统允许的取值范围内
111072042	The polling time is outside the allowed range.:轮询显示时间取值不在系统允许的取值范围内
111072043	The pane type cannot be left blank.:画面类型不允许为空
111072044	Invalid pane type.:画面类型非法
111072045	The value of the switch is outside the allowed range.:开关项取值不在系统允许的取值范围内
111072046	The site name cannot be left blank.:与会者的名称不允许为空
111072047	Site name too long.:与会者的名称超过系统允许的最大值

错误码	错误描述
111072048	Continuous presence is not supported. Ask the administrator to configure related data on the MCU.:不支持多画面,请联系管理员开启MCU配置
111072049	Response timed out. Try again later.:等待响应超时
111072050	Meeting locked.:会议被锁定
111072051	The meeting display mode is empty.:会议显示模式为空
111072052	The meeting display mode is outside the allowed range.:会议显示模式不在系统允许的取值范围内
111072053	The meeting pane type is outside the allowed range.:会议画面类型不在系统允许的取值范围内
111072054	The banner or caption type is empty.:横幅或字幕类型为空白
111072055	The banner or caption content is empty.:横幅或字幕的内容为空
111072056	The banner or caption content cannot exceed 2,000 bytes.:横幅或字幕的内容长度超过2000字节
111072057	The request message body is empty.:请求消息体为空
111072058	The pane index is greater than the total number of panes.:子画面索引大于画面总数
111072059	Meeting has no host.:会议中没有主席
111072060	Regular participants cannot perform this operation.:当前普通与会方类型不支持该操作
111072061	You have set host polling and cannot select a specific site to view or set continuous presence.:前主席正在轮询观看会场,不能单独选看其他会场或会议多画面
111072062	You cannot select a site to view on the web page. Perform the operation on the terminal.:当前主席与会方类型不支持该操作
111072063	Applying for recording resources....:录制资源重新申请错误码
111072064	Too many failed login attempts. Try again later.:会控登录锁定
111072065	The meeting ID or password is incorrect. Check the ID and password and try again.:会议不存在或密码错误
111072066	The enterprise has not purchased any recording storage space.:企业未购买录播空间
111072067	Insufficient enterprise recording storage space.:企业录播空间已满
111072068	Failed to start live broadcast. Try again later.:启动直播失败

错误码	错误描述
111072069	Not allow to hand up for host.:不允许指定会议主席举手发言
111072070	The operation is not allowed for guest.:来宾不允许进行该操作
111072071	The operation is only allowed for yourself.:仅能对自己进行操作
111072072	Not allow to apply host for attendees who do not have meeting control abilities.:没有会控能力的与会者不允许设置为主席
111072073	Notify message is expired.:推送消息已过期
111072074	Only participants with a number registered on HUAWEI CLOUD can be set as the host.:云下会场不能成为主席
111072075	Hung up site for meeting ended.:会议结束挂断会场
111072076	Hung up site by host.:主席挂断会场
111072077	Hung up site because host deleted site.:主席删除会议挂断会场
111072078	Hung up site because meeting is recreated.:重建会议挂断会场
111072079	Hung up site for moving site failed.:移动会场失败挂断会场
111072080	Hung up site for inviting site cross region failed.:跨region邀请会场失败挂断会场
111072081	Hung up site for invitation exception.:邀请会场失败挂断会场
111072082	Hung up site for joining meeting repeatedly.:会场重复入会挂断会场
111072083	Lock sharing failed.:锁定共享失败
111072084	This site does not support 1-screen layout.:云下会场禁止设置在多画面一画面中
111072085	Not support realtime subtitle.:会议不支持实时字幕
111072086	Realtime subtitle pid is null.:会议实时字幕指定发言人列表为空
111072087	Invalid realtime subtitle pid.:会议实时字幕指定发言人ID有误或已离会
111072088	Failed to set realtime subtitle.:会议实时字幕设置失败
111072089	Not allow audience to change speaker status.:不允许观众解闭音
111072090	Allow/forbid audience to speak only for audience.:主持人允许/禁止观众说话接口只能操作观众

错误码	错误描述
111072091	This meeting does not support simultaneous interpretation.:会议不支持同声传译
111072092	Interpreter group is empty.:会议当前传译组为空,请先配置传译组
111072093	This meeting has not enable simultaneous interpretation.:会议未开启同声传译
111072094	This site is not interpreter.:当前会场非传译员
111072095	This site has not yet confirmed interpreter identity.:传译员未确认身份
111072096	Invalid site language.:会场选择的语言非法
111072097	This site is confirmed interpreter.:当前会场是已确认的传译员
111072098	This site listen channel is different from speak channel.:普通会场听说频道不一致
111072099	Failed to set interpretation languages. Try again.:设置会议语言频道失败
111072100	Failed to set interpretation channel. Try again.:设置会场语言频道失败
111072101	Max participant amount reached.:超过设置的会议方数
111072102	Max allowed handup amount reached.:观众超过最大允许举手数
111072103	Max allowed speaking amount reached.:观众超过最大允许发言数
111072104	Failed to set virtual audio CSD site.:设置级联通道虚拟语言会场失败
111072105	This meeting does not support cohost.:该会议未启用联席主持人
111072106	This site does not support cohost.:该会场不支持联席主持人
111072107	Host cannot be set to cohost.:不支持设置主持人为联席主持人
111072108	Audience cannot be set to cohost.:不支持设置观众为联席主持人
111072109	Need to check cohost permission.:需要去主会议所在region判断是否为联席主持人
111072110	Cohost cannot operate host.:联席主持人不能操作主持人
111072111	This site does not support share invitation.:当前会场不支持邀请共享能力

错误码	错误描述
111072201	Hung up site because heart beat failed.:与终端心跳中断导致挂断会场
111072202	Insufficient audience cloud meeting room logic resources.:观众数超过VMR套餐中的观众最大并发资源数
111072203	Insufficient participant cloud meeting room logic resources.:主持人与嘉宾超过VMR套餐中的最大并发资源数
111072204	Insufficient webinar resources, so started failed.:网络研讨会套餐已经有在线会议,被占用,激活失败
111072205	Only RTC conf support local record.:非RTC会议不支持本地录制
111072206	Invalid E2E encryption parameter.:端到端加密参数非法
111072207	Not allow to switch role in webinar.:网络研讨会不允许进行角色转换
111072208	The client version used by the user is too early. Cannot start local recording.:客户端不支持本地录制
111072211	Meeting is under AR assist.:会议已经在进行AR协作
111072213	Audience cannot be set to host.:观众不能设置为主持人
111072214	Rename param contains sensitive words.:改名涉及敏感词错误
111072222	Indicates record start exception.:启动云录制失败, 请联系管理员
111072223	Indicates record paused exception.:暂停云录制失败, 请联系管理员
111072224	Indicates record stop exception.:停止云录制失败, 请联系管理员
111072225	Indicates live start exception.:启动直播失败, 请联系管理员
111072226	Indicates live stop exception.:停止直播失败, 请联系管理员
111072504	Max participants in a single meeting reached.:达到系统单个会议人数上限
111072507	waiting room user is leaved.:等候室准入用户已经离会
111072508	waiting room user is joined.:等候室准入用户已经准入
111073001	User does not exist.:用户不存在
111073002	Department does not exist.:组织不存在
111073003	Invalid username format.:用户非法
111073004	Invalid department name format.:组织非法

错误码	错误描述
111073005	Could not query department.:查询组织失败
111073006	Application for anonymous user failed.:申请匿名用户失败
111073007	Invalid cloud meeting room package status.:VMR套餐状态非法
111073008	Failed to Release anonymous user.:释放匿名用户失败
111073009	Not support cycle meeting for welinkC.:WeLinkC 不支持预约周期会议
111073010	Not support cloud meeting room for welinkC.:WeLinkC 不支持预约VMR会议
111073011	Failed to get Welink token.:获取WeLink Proxy Token失败
111073012	Failed to Get M to M token.:获取机机帐号Token失败
111073013	Invalid webinar cloud meeting package status.:网络研讨会VMR套餐状态非法
111073014	Max extend times for cloud meeting trial version reached.:会议延长次数超出VMR试用套餐可延长的最大次数
111073015	Max extend duration for cloud meeting trial version reached.:会议延长时间超出VMR试用套餐单次会议最大时长
111073016	Not support auto extend meeting for cloud meeting trial version.:VMR试用套餐不支持自动延长
111074001	This meeting host already exists.:IDO主席已经存在
111074002	No meeting host permissions.:无IDO主席权限
111074003	Failed to extend the meeting. Max meeting duration reached.:延长会议失败,会议已达到最大时长
111074004	Failed to extend the meeting. The service processing is abnormal.:延长会议失败,服务处理异常
111075001	Failed to start timer.:启动定时器失败
111075002	Failed to end timer.:停止定时器失败
111075003	Json parse failed.:json转换异常
111076001	Released by host.:会场被终端主席挂断
111076002	Released by portal host.:会场被portal主席挂断
111076003	Max number of participants reached.:超出VMR最大方数
111076004	Insufficient enterprise resources.:企业并发数不足
111076005	Not answered.:会场无响应
111076006	User does not exist.:会场不存在

错误码	错误描述
111076007	Offline.:会场离线
111076008	Busy.:忙
111076009	Rejected by user.:拒接
111076010	Released by user.:会场挂断
111076011	Insufficient meeting resources in the system.:媒体资源不足
111076012	Outbound calling disabled.:PSTN服务未开通
111076013	Application for address token failed.:申请地址本鉴权信息失败
111076014	Call failed.:SIP呼叫超时
111076015	Max number of participants reached.:超过设置的会议方数
111076017	Invalid user.:用户失效
111076018	Invalid enterprise.:企业失效
111076019	Invalid SP.:SP失效
111076022	Not allow joining by scanning for hard device.:企业配置来宾不允许扫码邀请硬终端入会*
111076028	Not allow conf control.:会议正在进行AR协作, 禁止共享/分组会议/释放主持人等会控操作*
111076029	Not allow AR Assist.:会议正在进行共享/分组会议, 禁止进行AR协作*
111076099	Common error.:通常失败
111077001	RSE server info does not exist.:RSE录播相关错误码
111077101	Not support AI record.:会议不支持AI会议纪要
111077102	Reapplying for AI record resource.:正在重新申请会议纪要资源
111077103	Insufficient AI record resource.:AI会议纪要资源不足
111077104	In AI recording, the meeting cannot stop live or recording.:会议处于AI会议纪要状态,不允许停止录制/直播
111077201	Record file does not exist or have no permission.:录播文件已删除或您无权观看,请确认后重试
111077202	RSE authentication failed.:RSE机机鉴权失败
111078000	No available region in multi region.:没有可用Region
111078001	No available global in multi region.:没有可用Global

错误码	错误描述
111078002	Dispatch region error in multi region.:多region 分发region错误
111079000	Failed to phrase big data param query xml.:大参数查询xml解析失败
111079001	Failed to phrase anonymous assign xml.:匿名帐号分配xml解析失败
111080001	Bad request.:请求消息参数错误
111080002	Insufficient permissions.:用户无权限进行该操作
111080004	Site data not found or deleted.:会场不存在或已被删除
111081001	Site is inviting.:会场正在通话中
111081002	Site meeting does not exist or is ended.:会场所存在会议不存在或已结束
111081003	Failed to apply site resource.:申请会场资源失败
111081004	Failed to release site resource.:释放会场资源失败
111081005	Failed to apply site media resource.:申请会场媒体资源失败
111081006	Authentication failed.:认证失败
111081007	Meeting is locked.:会议已经被锁定
111081009	Login is locked because max authentication times reached.:会议鉴权失败次数已达上限被锁定
111081010	Waiting user restrictions.:等候用户限制
111082001	Not support this control.:不支持该会控
111082002	No response.:无响应
111082003	Site not exist.:会场不存在
111082004	Site unavailable.:会场离线
111082005	Site busy.:会场正忙
111082006	Site refused.:会场拒接
111082007	Site leave.:会场挂断
111082008	PSTN service is closed.:PSTN服务关闭
111082009	SIP invitation timeout.:SIP呼叫超时
111083001	RSE does not exist.:RSE设备不存在
111084001	Failed to execute DTMF.:执行DTMF失败
111084002	Get started meeting exception.:启动会议异常

错误码	错误描述
111084003	Call MCU side exception.:呼叫MCU端异常
111084004	Join call exception.:加入呼叫异常
111084005	Invite exception.:邀请异常
111084006	Call terminal side exception.:呼叫端侧异常
111084007	Answer exception.:应答异常
111084008	Common CTL exception.:通用CTL异常
111084009	Hangup exception.:挂断异常
111084010	Notify capability exception.:通知能力异常
111084011	Update MT exception.:更新MT异常
111084012	Negotiate exception.:协商异常
111084013	Not support this meeting.:不支持该会议
111084014	Not allowed join meeting in RTC inviting because guests are not invited.:RTC外邀入会,来宾不在被邀请名单中,拒绝入会
111089002	Failed to phrase URI join xml.:链接入会XML解析失败
111090001	Failed to join data meeting.:数据入会失败
111090002	Join data meeting overtime.:数据入会超时
111100001	Failed to get slider image.:获取滑块图片失败
111100002	Slider check failed.:滑块校验失败
111100003	Invalid random number.:无效的随机数
111100004	Invalid slider token.:无效滑块token
111100005	Need to check slider.:需要滑块校验
111100006	Invalid SMS phone number.:非法的手机号
111100007	Failed to send verification code.:发送验证码失败
111100008	Verification code expired.:验证码已过期
111100009	Verification code error.:验证码错误
111100010	Max verification code check times reached.:验证码错误超过最大次数
111100011	The uuid is invalid, you need to re-verify the phone number.:uuid无效,需要重新验证手机号
111100012	The meeting is not on this site.:会议不在本站点

错误码	错误描述
111100013	Need to verify the mobile phone number. Overseas access to China, free enterprises need to verify the mobile phone verification code.:需要验证手机号，海外接入国内，免费企业需要进行手机验证码校验
111110001	Failed to add IM group.:添加群组失败
111110002	Failed to disband IM group.:删除群组失败
111110003	Failed to add IM member.:添加成员失败
111110004	Failed to remove IM member.:删除成员失败
111111007	hangup for rejoin mmr failed.:mmr重入会失败挂断
111112001	Failed to enter the waiting room.:入等候室失败
111112002	Failed to exit the waiting room.:出等候室失败
111112003	Waiting room cache operation failed.:等候室缓存操作失败
111112004	Co-hosts cannot be moved into the waiting room.:联席主持人不能被移入等候室
111112005	Not allowed to enable waiting room.:不允许启用等候室
111112008	Not support this feature according to client capabilities.:客户端能力不支持此特性
111113001	can not find application.:找不到应用信息
111113002	application exist.:应用已存在，目前一个会议只能关联一种应用
111113003	not required to create application.:无权限创建应用
111113004	need verify id.:应用操作需要实名认证（用户需要绑定手机）
111113005	get token failed.:获取应用用户token失败
111113006	url illegal.:应用服务URL地址非法
111113101	shimo doc has not used.:石墨文档应用未启用
111113102	call shimo doc server failed.:调用石墨文档服务失败
111113103	get shimo token failed.:获取石墨应用级token失败
111113104	not in conf,has no right to apply.:用户不在会议中，无权限申请权限
111500301	breakout conf surpass limit.:分组会议子会议超过上限
111500302	breakout conf name error.:分组会议子会议名称错误
111500303	breakout Start failed.:分组会议启动失败

错误码	错误描述
111500304	breakout not exist when delete:分组会议中没有子会议（删除）
111500305	breakout not exist when start.:分组会议中没有子会议（启动）
111500306	breakout is closing by host:分组会议结束中(主持人停止)
111500307	breakout is closing by host:分组会议已经结束或未启动（加入时）
111500308	Attendee in target conf:移动用户在目标分组
111500309	Chair in current conf:主持人在当前分组中（请求帮助）
111500310	not support breakout conf:不支持分组会议（启动时 启动接口）
111500311	breakout conf starting or started:分组会议已经启动或启动中（重复启动）
111500312	subconf not exist:子会议不存在
111500313	not support in mainconf:主会中用户不支持此操作（请求帮助）
111500314	breakout conf state illegal:分组会议状态异常（缓存误删）
111500315	action forbidden by status:分组会议静态、动态操作由于状态原因，不支持当前操作
111500316	apply chair failed due to host in subconf:主持人在分组中，不能申请为主持人
111500317	apply sub conf id failed:申请分组的会议id失败
111500318	query main conf failed:获取主会议信息失败
111500319	attendee not found:没有找到用户信息
111500320	sub conf stop failed:分组会议停止失败
111502000	subject faile and leave conf start:订阅失败需要离会的错误码起始值
111502099	subject faile and leave conf end:订阅失败需要离会的错误码结束值
112010001	Common failure.:通用失败
112010002	Null param.:参数为空
112010003	Participant not found.:未找到参与者
112010004	Meeting not found.:会议不存在
112010005	SIP invite information timeout.:invite消息超时

错误码	错误描述
112010006	SIP invite information failed.:invite消息失败
112010007	SIP information timeout.:info响应超时
112010008	SIP information failed.:info响应失败
112010009	System processing failed.:系统处理失败
112010010	System processing failed.:系统处理失败
112010011	Failed to invite MRS.:呼叫mrs失败
112010012	Failed to invite participant.:呼叫终端失败
112010013	Data meeting join information failed.:数据入会Info失败
112010014	Failed to start meeting.:召开会议失败
112010015	Media negotiation failed.:媒体协商失败
112010016	User rejects.:用户拒绝
112010017	User is busy.:用户忙
112010018	No call permission.:没有呼叫权限
112010019	Unknown callee.:未知被叫人
112010020	No user response.:没有用户响应
112010021	Join meeting timed out.:加入会议超时
112010022	Session has been released.:会话已释放
112010023	Callee is not registered.:被叫人未注册
112010024	SIP invite timed out.:SIP邀请超时
112010025	User Session timer times out and hangs up.:用户Session timer 超时挂断
112010026	The participant bye hangs up on the MRS side.:与会者腿MRS侧bye挂断
112010027	The participant bye hangs up on terminal side.:与会者腿终端侧bye挂断
112010028	No RTP stream hang up.:无码流挂断
112100000	Sysportal exception.:系统异常
112100001	Bad request.:请求异常
112100002	Illegal request.:非法请求
112100003	Data not found.:数据未获取到
112100004	Data duplication.:数据重复

错误码	错误描述
112100006	The requested type is not supported.:请求的类型不支持
112100007	Failed to save notification template.:保存通知模板失败
112100008	Failed to save data.:保存数据失败
112100009	Failed to query data.:查询数据失败
112100010	Failed to delete data.:删除数据失败
112100011	Password reset required.:需要重置密码
112100012	Incorrect old password.:旧密码错误
112100013	The new password cannot be the same as the old password.:新密码不能与旧密码相同
112100014	Insufficient password complexity and low security.:密码复杂度不够,安全性低
112100015	The length does not meet the requirements.:长度不符合要求
112100016	RSE is in meeting.:RSE在会议中
112100017	RSE to be deleted.:RSE待删除
112100018	Failed to query the meeting list.:查询会议列表失败
112100019	User has been locked out. Try again later.:用户已被锁定,请稍后重新登录
112100022	The live room channel name exists.:频道名称已存在
112100023	Failed to modify system configuration items for duplicated meeting ID prefix.:修改系统配置项失败,会议ID前缀重复
112100024	CONFID prefix can only be added, not modified or deleted.:CONFID前缀只能增加,不能修改和删除
112100025	The live room channel url exists.:直播房间地址已存在
112100026	The live room has been occupied.:直播房间已被使用
112100027	The live room does not exist.:直播房间不存在
112100028	Live room release failed.:直播房间释放失败
112100029	The request is missing header or param.:请求缺失头域
112100030	The Live main stream URL already exists.:音视频推流地址已存在
112100031	The Live auxiliary stream URL already exists.:辅流演示推流地址已存在
112100032	Password is weak.:密码为弱密码

错误码	错误描述
112100033	The live room is occupied.:直播房间被占用
112100034	Save part of attendees since max attendees size reached.:因为与会者数量超出上限只保存部分
112100035	Failed to export data.:导出数据失败
112100036	Failed to import data.:导入数据失败
112100037	Max file size reached.:文件大小超出上限
112100038	LVS target address exists.:lvs的目标地址已存在
112100039	LVS port existed.:lvs端口重复
112100040	Not allow to modify global standby config.:Global只有主机可以修改配置;
112100041	MCU insulation IP address exists.:mcu隔离的IP已经存在
112100042	MCU insulation IP address cannot be modified.:mcu隔离的IP不能修改
112100043	SBC address exists.:SBC配置的地址已经存在
112100044	Internal exception.:内部错误异常
112100045	System config delete exception.:系统配置不允许被删除
112100046	No ISBC access address.:企业id无法获取关联SBC地址
112100047	Media type need to be HD because the organization enable Full HD capability.:企业已开启全高清功能,媒体设置不能低于1080P/8M
112100061	Same interpreter language abbreviation exists.:语言缩写重复
112100062	Same interpreter language name exists.:语言名称重复
112100063	MCU type is MIXED and insulation config is not configured, or MCU meeting exists when modifying mcu connection settings.:修改MCU连接配置告警,以下两种场景会导致:1.MCU存在会议 2.修改类型为MIXED未配置隔离
112100064	MCU type is MIXED and insulation config is not configured when adding mcu connection settings.:新增MCU连接配置告警,MCU类型为MIXED且未配置隔离
112100065	MCU type is MIXED when modifying MCU insulation config or disbanding enterprise.:修改MCU隔离配置时,解除企业绑定时MCU类型为MIXED
112100066	MCU type is MIXED when deleting MCU insulation config.:删除MCU隔离配置时,MCU类型为MIXED

错误码	错误描述
112100067	This gateway type is incompatible with MCU.:MCU与网关类型不兼容
115021001	Insufficient media resource.:媒体资源不足
115021002	Meeting does not exist.:会议不存在
115021003	Media resource does not exist.:媒体资源不存在
115021005	Insufficient media bandwidth resource.:带宽资源不足
115021006	Media device fault.:设备故障
115022000	Insufficient common logic resource.:通用的逻辑资源不足
115022001	Insufficient meeting resources in the enterprise.:企业会议并发方数不足
115022002	Insufficient logic port resource.:逻辑资源端口资源不满足
115022003	Live broadcast resources have not been purchased or are insufficient.:企业直播推流方数不足
115022004	Media device does not exist.:媒体设备不存在
115022005	The cloud recording ports purchased by enterprises have been used up.:企业购买的云录制端口已经用完
115023001	Insufficient ID resources.:ID资源不足
115024001	RMS service exception.:RMS服务异常
115025001	The live room has already been occupied.:直播房间已经被使用
115025002	The live room does not exist.:直播房间不存在
115025003	Live room release failed.:直播房间释放失败
115026001	MCU meeting exists.:MCU有会议
115029001	Invalid common parameter.:通用参数值不合法
115029002	Null parameter.:参数值为空
115029003	service is not available.:服务不可用
115029004	Configuration exception.:配置异常
115029005	Duplicately apply resources.:重复申请资源
115029006	Waiting for applying resources.:等待申请资源中
115029999	Unknown error.:未知错误
118000000	User authentication failed.:用户鉴权失败
118000001	User token authentication failed.:usg token 鉴权失败

错误码	错误描述
118000002	User authentication account lockout.:usg鉴权帐号锁定
200000001	Server is busy. Please try again later.:服务器正忙 请稍后再试
200000002	Internal error. Please contact SRE.:服务器正忙 请联系SRE
200000003	Invalid parameter. Please check whether the parameters are valid.:请求参数错误 请检查参数是否合法
200000004	Illegal request. Please check the value of the request header field X-Auth-Token or X-Access-Token.:非法请求 请检查请求头域X-Auth-Token或X-Access-Token的值
200000005	Forbidden access during the version upgrade.:系统正在升级, 暂时不可访问。如对您造成不便, 十分抱歉
201020023	This user is already the owner of another corporation and cannot create a new corporation for it.:该用户已经是其他企业的所有者,无法为其创建新的企业
201040005	Max number of users reached. Please check the purchased meeting users.:用户总数超过限制, 请检查购买的会议用户
201040016	Password security is low. Please set another password.:密码安全性低, 请设置其他密码
201040031	Activation code overdue.:激活码过期
201040042	Pre-authentication expired.:预验证失效
201040043	This user is the owner of another corporation and cannot be invited.:该用户是其他企业的所有者,暂无法邀请
201040044	Invalid invite connection.:邀请连接非法
201040045	User is already in waiting list.:用户已经在等待列表
201040046	User already exists.:用户已存在
201040047	Account modification is not allowed.:不允许修改账户
201050012	Activation code does not exist.:激活码不存在
201050015	Invalid activation code.:激活码错误
201050021	Activation failed.:激活失败
201050022	Terminal does not allow unbinding.:终端不允许解绑
201080000	VMR resource unavailable.:云会议资源不可用
201080006	The host password and the guest password cannot be the same. Please confirm the input parameters.:主席密码和来宾密码不能相同 请确认输入参数
201080017	Same as meeting password.:和会议密码相同

错误码	错误描述
201510001	No invite permission to get an invite code.:没有邀请权限去获取邀请码
201510003	Invalid application.:申请失效
206010000	User name or password error.:用户名或密码错误
206010007	Account is not activated, no permission to log in.:帐号未激活,无权限登录
206010010	Failed to get verification code, Please contact SRE.:获取验证码失败 请联系SRE
206010011	Verification code must be filled.:验证码不能为空
206010012	Verification code check failed.:验证码校验失败
206010013	Jump login failed. Please contact SRE.:跳转登录失败 请联系SRE
206010022	SSO login failed. Please contact SRE.:SSO登录失败 请联系SRE
206010023	SSO login is not supported.:不支持sso登录
206010024	Invalid authentication type.:认证类型非法
206010025	APPID verification failed.:APPID校验失败
206010027	Authentication type error.:校验类型错误
206010029	Network error.:网络异常
206010030	APPID does not allow automatic registration.:APPID不允许自动注册
206010031	APPID auto registration failed.:APPID自动注册失败
206010036	Normal users cannot log in.:普通用户无法登录
206010042	Huawei account authentication failed.:华为账户认证失败
206010057	APPID authentication expires.:APPID认证过期
206010058	APPID account is not bound.:APPID账户未绑定
206010064	Site mismatch.:站点不匹配
206010065	Phone number conflict.:电话冲突
206010066	E-mail conflict.:邮箱冲突
206010070	The number of users has reached the maximum.:用户数超过上限
206010071	WeChat authentication failed.:微信鉴权失败

错误码	错误描述
206010074	International sites support site changing re-login.:国际站点支持切站点重登
206010075	Account has been unbound.:账号已经解绑
206020000	Invalid authentication information. Please check the value of the request header field X-Auth-Token or X-Access-Token.:鉴权信息无效 请检查请求头域X-Auth-Token或X-Access-Token的值
206020001	Token is created by disaster recovery site.:token是容灾站点创建
206020002	Token is invalid due to being kicked.:token由于被踢失效
206030000	The user does not exist.:系统中不存在该用户
206030001	The verification code is incorrect or has expired. Please obtain a new one.:验证码错误或失效，请重新获取
206030002	Incorrect verification code.:验证码错误
206030003	Failed to send the verification code.:发送验证码失败
206030004	Try to get a new verification code 180s later.:请在180秒后重新发送验证码
206030005	The password cannot be changed.:系统不支持修改密码
206030006	The old password is incorrect.:旧密码错误
206030007	The length of the new password does not meet the requirements.:新密码长度不符合要求
206030008	The password complexity does not meet the requirements.:密码复杂度不满足要求
206030009	The new password must be different from the previous old passwords.:设置的密码不能与最近旧密码相同
206030010	The password cannot contain three or more of the same characters consecutively.:密码不能包含3个以上重复字符
206030011	The password cannot be changed within 5 minutes upon the last password change.:修改密码后，5分钟内不能更新密码
206030012	The password cannot contain the account or the reverse order of the account.:密码不能与帐号或其逆序帐号相同
206030013	At least two characters must be different between the new and old passwords.:新密码相较于旧密码至少要有两个不同的字符
206030014	Invalid password format.:密码格式不正确

错误码	错误描述
206030015	The user is locked and will be locked in 60s.:用户已被锁定, 请60秒后重试
206030016	Have no right to reset other's password.:没有权限重置密码
206030018	The number of password retrieval times in a day has reached the maximum(10 times).:一天内忘记密码次数达到最大值 (10次)
206030019	The number of incorrect verification code input has reached the maximum, Please get it again.:验证码输入错误次数超过最大值, 请重新获取
206030023	Failed to send verification code. Please contact SRE.:发送验证码失败 请联系SRE
206030028	The system has enabled SSO login and does not support changing the password. Please retry after the system has disabled SSO login.:系统已开启SSO登录,不支持修改密码 系统关闭SSO登录后重试
206030031	Forgot password for Huawei account.:华为帐号忘记密码
206040000	No corresponding terminal type.:无对应的终端类型
206060008	Sending verification code is locked.:发送校验码锁定
206060009	Slider verification required.:需要滑块校验
206060010	Invalid slider token.:滑块token非法
206060011	Request parameter error.:请求参数错误
206060012	Sending SMS is not supported.:不支持发送短信
206060013	Repeated sending is not supported.:不支持重复发送
206060014	Invalid verification code.:验证码非法
206060015	Max verification codes check times reached.:验证码次数超过最大次数
206060017	Failed to bind Huawei account.:华为账户绑定失败
211030046	Not the ideahub activation code.:不是ideahub激活码
211030050	auth fail too many times, locked.:身份验证失败次数太多,已锁定
211030052	auth fail too many times, permanently locked.:身份验证失败次数太多,已永久锁定
211060003	The maximum times of log uploads has been reached.:日志上传次数达到最大限制
411000001	Common failure.:失败

错误码	错误描述
411000002	Parameter error.:参数错误
411000003	Meeting does not exist.:会议不存在
411000004	A meeting already exists.:已经有会议存在
411000005	Meeting state error.:会议状态异常
411000006	Failed to get device.:获取设备异常
411000007	Failed to set device.:设置设备异常
411000008	No device available.:无可用设备
411000009	This operation is only supported on mobile.:只有移动端支持该操作
411000010	Network access error.:网络接入错误
411000011	Other supplementary service is running.:正在进行其他补充业务
411000012	Memory allocation error.:分配内存错误
411000013	Initialization failed.:初始化失败
411000014	User does not exist.:用户不存在
411000015	Did not join the data meeting.:未加入数据会议
411000016	Neither meeting nor call exists.:会议和呼叫均不存在
411000017	SVN port detection failed.:svn端口探测不通
411000018	Failed to write file.:写文件失败
411000019	RTC does not support STG mode.:RTC不支持STG模式
411000020	Failed to load RTC library. RTC meeting is not supported.:rtc库加载失败，不支持rtc会议
411000021	The meeting is in the audience pause state when changing roles.:转角色的时候会议正处于观众暂停观看状态
411000022	Not in call.:不在呼叫中
411000023	Meeting or call does not exist.:会议或者呼叫不存在
411000024	Pairing state not allowed.:配对状态不允许
411000025	Pairing request failed.:配对请求失败
411000026	Pairing exceeds limit.:配对超出限制
411000027	Pairing common error.:配对通用错误
411000028	Pairing and joining the meeting failed.:配对入会失败
411000029	Pairing and joining the meeting is busy.:配对入会忙

错误码	错误描述
411000030	Pairing and joining the meeting is invalid.:配对入会无效
411000031	Multiple clicks at the same time to pair and join the meeting error.:同时多次点击入会错误
411000032	Pairing is not established or is being established.:配对未建立或正在建立中
411000033	The camera is not turned on in the third-party capture mode.:第三方采集模式下摄像头未打开
411000034	RTC engine creation failed.:RTC 引擎创建失败
411000035	Not in a meeting.:不在会议中
411000036	mute self failed.:静音自己失败并且采用了保护
411000100	End the meeting with no stream due to disconnection of network.:无码流结束会议网络断开
411000101	End the meeting with no stream because fail to re-join meeting.:无码流结束会议重入会失败
411000102	Null pointer exception.:空指针异常
411000103	Parameters cannot be empty.:参数不许为空
411000104	Indicates encrypt and decript func not match.:加解密算法不匹配
411000105	Indicates Loading User-Defined DLLs failed .:加载用户自定义dll失败
411000106	Indicates Applition not support shimo docs .:企业配置不支持打开石墨文档
411000107	Indicates notification of render fail .:数据会议UI渲染失败
411000108	End the meeting with mmr media rejoin failed.:mmr重入会失败
412100001	The message processing thread is not initialized.:消息处理线程未初始化
412100002	Parameter error.:参数错误
412100003	Timeout.:超时
412100004	Memory allocation error.:分配内存错误
412100005	XML parsing error.:XML解析出错
412100006	DNS resolution error.:DNS解析异常
412100007	Request message failed.:请求消息异常
412100008	Authentication failed.:鉴权失败

错误码	错误描述
412100009	Server error.:服务器异常
412100010	Account is locked.:帐号被锁定
412100011	Create timer error.:创建定时器错误
412100012	Server CA certificate verification failed.:服务器CA证书校验失败
412100013	Network error.:网络异常
412100014	Server redirect.:服务器返回重定向地址
412100015	Account is not activated.:帐号未激活
412100016	JSON parsing error.:JSON解析失败
412100017	The proxy communication times out and the TCP link building fails. Probably because the proxy configuration is incorrect.:代理通信超时, tcp建链失败, 大概率是代理配置错误
412100018	Disconnected by the proxy. Probably because the server address is misconfigured.:接被代理断开, 大概率是服务器地址配置错误
412100019	The TCP link establishment timed out. Probably because the proxy domain name or port configuration is incorrect.:tcp建链超时, 大概率是proxy域名或者端口配置错误
412100020	The authentication process failed. The user does not have proxy permission and needs to apply for proxy permission.:认证鉴权过程失败, 用户没有proxy权限, 需要申请proxy权限
412100021	The authentication process failed. The proxy user name or password is incorrect.:认证鉴权过程失败, 代理用户名或者密码错误
412100022	The authentication process failed. Incorrect authentication type. User name and password are required. The user needs to be prompted for input.:认证鉴权过程失败, 错误的认证类型, 同时需要用户名、密码, 需要提示用户输入
412100023	The authentication process failed. Other unknown errors. It may be that the authentication method is not supported. The user needs to be prompted to try again.:认证鉴权过程失败, 其他未知错误, 有可能是不支持认证方式, 建议弹框提示用户重新发起
412100024	The authentication process failed. Unable to resolve gateway.:认证鉴权过程失败, 无法解析网关
412100025	The authentication process failed. Unable to resolve host.:认证鉴权过程失败, 无法解析主机

错误码	错误描述
412100026	The authentication process failed. Unable to connect.:认证鉴权过程失败，无法连接
412100027	The authentication process failed. Operation times out.:认证鉴权过程失败，操作超时
412100028	Message length too long.:消息长度太大
412100029	Base64 encryption failed.:base64加密失败
412100030	XML parsing error.:XML解析出错
412100031	Secure function error.:安全函数失败
412100032	Firewall detection parameter error.:防火墙检测参数错误
412100033	Failed to create STG tunnel.:创建STG隧道失败
412100034	Failed to update STG service.:更新STG服务失败
412100035	Encryption failed.:加密失败
412100036	Do not allow duplicate downloads.:不允许重复下载
412100037	File operation failed.:文件操作失败
412100038	The request message returns empty.:请求消息返回空
412100039	The number of detection servers is 0 error.:探测服务器数为0错误
412100040	Failed to get local IP address.:获取本地IP失败
412100041	XXX.:登录模块消息未初始化
412100042	http return 429 error.:http 返回429错误
412100043	Not logged in, not available:此业务未登录不可用
412200001	Internal error.:内部错误
412200002	Parameter error.:参数错误
412200003	Network timeout.:网络超时
412200004	Network Error.:网络错误
412200005	Need to bind mobile phone.:需要绑定手机
412200006	Need to bind email.:需要绑定邮箱
412200007	Has initialized already.:重复初始化
412200008	State machine execution failed.:状态机执行失败
412200009	Failed to start timer.:开启定时器失败
412200010	The user is prompted to change the password when logging in for the first time.:首次登录，提示用户修改密码

错误码	错误描述
412200011	Unknown authentication type.:未知鉴权类型
413000001	Common error.:一般错误
413000002	Parameter error.:参数错误
413000003	Memory allocation error.:分配内存错误
413000004	System error.:系统错误
413000005	Message sending error.:发送消息错误
413000006	Get system configuration error.:获取系统配置错误
413000007	Physical network error.:物理网络错误
413000008	Network access error.:网络接入错误
413000009	Create timer error.:创建定时器错误
413000010	Call state error.:呼叫状态错误
413000011	Other operation in progress error.:正在进行其他操作错误
413000012	Requesting the master to start call error.:请求主控进行呼叫开始错误
413000013	Requesting audio resource error.:申请音频资源错误
413000014	Other supplementary service is running.:正在进行其他补充业务
413000015	Record call error.:记录通话记录错误
413000016	Media process error.:媒体进程返回的错误
413000017	Maximum number of calls reached.:超过最大呼叫路数
413000018	SIP account ID does not exist.:SIP账户ID不存在
413000019	Call ID does not exist.:呼叫ID不存在
413000020	Failed due to registering.:正在注册中导致失败
413000021	Registration failed.:注册失败
413000022	Deregistration failed.:注销失败
413000023	Setting account information error.:设置账户信息错误
413000024	Setting SIP account error.:设置SIP帐号失败
413000025	Reporting SIP account information error.:上报SIP帐号信息失败
413000026	Server information error.:服务器信息错误
413000027	SIP account information error.:账户信息错误

错误码	错误描述
413000028	SIPC execution error.:SIPC执行错误
413000029	Not registered error.:未注册错误
413000030	Call Sip interface to subscribe error.:调用Sip接口订阅失败
413000031	Canceling subscription error.:注销订阅中错误
413000032	Subscribing error.:订阅中错误
413000033	Only one local meeting is allowed.:只允许一个本地会议
413000034	Local meeting is not created.:本地会议未创建
413000035	The participant line does not match the chair account.:与会者线路与主席账户不匹配
413000036	Local meeting state error.:本地会议状态错误
413000037	Failed to report the linkage state of the IP Phone.:IP Phone端联动状态上报失败
413000038	Meeting ID does not exist.:会议ID不存在
413000039	Video hold error.:视频保持失败
413000040	Video unhold error.:视频恢复失败
413000041	The maximum number of server meeting reached.:服务器会议个数超出
413000042	Get AA random number error.:获取AA随机数失败
413000043	AA login error.:AA登陆失败
413000044	Meeting type mismatch.:会议类型不匹配
413000045	Failed to start video meeting.:视频会议开启失败
413000046	AA no master server.:AA无主服务器
413000047	Password error.:密码错误
413000048	Username error.:用户名错误
413000049	User is logged in.:用户已登录
413000050	Account is locked.:账户已锁定
413000051	Terminal type mismatch.:终端类型不匹配
413000052	Parsing XML error.:解析XML错误
413000053	Connecting to server error.:连接服务器错误
413000054	Get media configuration error.:获取媒体配置失败
413000055	Get service permission error.:获取业务权限失败

错误码	错误描述
413000056	Insufficient service permission.:业务权限不足失败
413000057	Network environment error.:网络环境错误
413000058	Service conflict.:业务冲突
413000059	Connection timeout.:连接超时
413000060	Unknown error.:未知错误
413000061	Add ringing information error.:添加振铃信息失败
413000062	Delete ringing information error.:删除加振铃信息失败
413000063	Create ringing number error.:创建振铃号码失败
413000064	VVM parameter error.:VVM参数错误
413000065	Getting registered service error.:获取登记业务错误
413000066	Getting voicemail error.:获取语音邮箱错误
413000067	Attendee already exists.:与会者已经存在
413000068	Attendee does not exist.:与会者不存在
413000069	Create server video meeting window error.:创建服务器视频会议窗口失败
413000070	Video window already exists.:视频窗口已存在
413000071	Getting meeting list error.:获取会议列表失败
413000072	Operating requires host privileges.:需要主持人权限才能操作
413000073	No video device to operate.:没有视频设备可以操作
413000074	Refresh registration is not turned off.:没有关闭刷新注册
413000075	Online state report error.:在线状态上报失败
413000076	Network address subscription report error.:网络地址本订阅notify上报失败
413000077	Telepresence account was force unregistered. Information report error.:智真帐号被踢,info上报失败
413000078	TLS root certificate error.:TLS根证书错误
413000079	AD authentication error.:AD鉴权失败
413000080	Meeting list is getting.:会议列表正在获取中
413000081	Prohibited calls.:禁止呼叫
413000082	The maximum number of calls has been reached.:呼叫数达上限

错误码	错误描述
413000083	The maximum number of encrypted calls has been reached.:加密呼叫数达上限
413000084	The maximum number of video calls has been reached and needs to downgrade to audio.:超出视频呼叫数，需要降为音频
413000085	H323 account ID does not exist.:H323账户ID不存在
413000086	H323 account information error.:H323账户信息错误
413000087	H323 account information report failed.:H323帐号信息上报失败
413000088	Insufficient local resources.:本端资源不足
413000089	The calling protocol is not supported.:不支持该呼叫协议
413000090	Setting H323 account information error.:设置H323账户信息错误
413000091	Mediatrix meeting access number reporting failed.:mediatrix会议接入号上报界面失败
413000092	Mediatrix VMR information reporting failed.:上报Mediatrix VMR信息到界面失败
413000093	IMS meeting list information reporting failed.:上报获取IMS会议列表信息到界面失败
413000094	TUP initialization is not complete.:tup初始化未完成
413000095	Message return value error.:消息返回值错误
413000096	Message content too large error.:消息内容太大错误
413000097	Secure function error.:安全函数返回错误
413000098	message response is empty error.:消息响应为空错误
413000099	Account is not enabled.:帐号未启用
413000100	Create call control implement error.:创建呼叫控制类错误
413000101	Already in call.:已经在呼叫中
413000102	Call does not exist.:呼叫不存在
413000103	Failed to get media configuration information.:获取媒体配置信息失败
413000104	Failed to set third-party capture. Calling HME_V_Capture_InputData_MCU failed.:设置第三方采集失败，调用HME_V_Capture_InputData_MCU返回失败
414100001	Common error.:一般错误

错误码	错误描述
414100002	Parameter error.:参数错误
414100003	Timeout.:超时
414100004	Memory allocation error.:分配内存错误
414100005	DNS resolution error.:DNS解析异常
414100006	Request message failed.:请求消息异常
414100007	Authentication failed.:鉴权失败
414100008	Server error.:服务器异常
414100009	Create timer error.:创建定时器错误
414100010	Start timer error.:开启定时器错误
414100011	The meeting control block is full. Please release the control block that is not in use.:会控控制块已满，请释放没在使用的控制块
414100012	The proxy communication times out and the TCP link building fails. Probably because the proxy configuration is incorrect.:代理通信超时，tcp建链失败，大概率是代理配置错误
414100013	Disconnected by the proxy. Probably because the server address is misconfigured.:接被代理断开，大概率是服务器地址配置错误
414100014	The TCP link establishment timed out. Probably because the proxy domain name or port configuration is incorrect.:tcp建链超时，大概率是proxy域名或者端口配置错误
414100015	The authentication process failed. The user does not have proxy permission and needs to apply for proxy permission.:认证鉴权过程失败，用户没有proxy权限，需要申请proxy权限
414100016	The authentication process failed. The proxy user name or password is incorrect.:认证鉴权过程失败，代理用户名或者密码错误
414100017	The authentication process failed. Incorrect authentication type. User name and password are required. The user needs to be prompted for input.:认证鉴权过程失败，错误的认证类型，同时需要用户名、密码，需要提示用户输入
414100018	The authentication process failed. Other unknown errors. It may be that the authentication method is not supported. The user needs to be prompted to try again.:认证鉴权过程失败，其他未知错误，有可能是不支持认证方式，建议弹框提示用户重新发起
414100019	Curl proxy exception.:curl代理异常

错误码	错误描述
414100020	Curl host exception.:curl主机异常
414100021	Curl connection exception.:curl连接异常
414100022	Curl operation timed out.:curl操作超时
414100023	Message length too long.:会控消息内容过大
414100024	Failed to create thread.:创建线程失败
414100025	Secure function error.:安全函数返回错误
414100026	JSON parsing failed.:解析json体失败
414100027	WS token is empty.:ws token值为空
414100028	Return JSON data error.:返回json数据异常
414100029	Failed to create meeting control block.:创建会议控制块失败
414100030	Failed to get meeting control block.:获取会议控制块失败
414100031	Failed to get meeting control block by CallId.:通过CallId获取会议控制块失败
414100032	Destroying the meeting control block failed.:销毁会议控制块失败
414100033	Temporary creating meeting information not found.:找不到创会临时信息
414100034	Failed to get participant ID.:获取参会者ID失败
414100035	Failed to get MT by participant ID.:通过ParticipantId获取MT失败
414100036	Failed to get self information by participant ID.:通过ParticipantId获取selfinfo失败
414100037	Failed to get participant information by MT.:通过MT获取会议成员信息失败
414100038	Manual invitation required.:需要手动邀请
414100039	The major parameters of the meeting have been reported.:已上报过会议大参数
414100040	Subtitles are not supported.:不支持字幕
414100041	Meeting Hang Up - Timeout.:会议挂断 - 超时
414100042	Meeting Hang Up - No Streaming.:会议挂断 - 无码流
414100043	Meeting hung up - Hung up by the host.:会议挂断 - 被主持人挂断
414100044	Meeting Hang Up - Meeting has ended.:会议挂断 - 会议已结束

错误码	错误描述
414100045	End of meeting due to leave meeting.:离会导致的会议结束
414100046	Message queue processing function registration failed.:消息队列处理函数注册失败
414100047	Message queue processing function deregistration failed.:消息队列处理函数去注册失败
414100048	Meeting ended due to reject meeting.:拒绝会议导致的会议结束
414100049	Meeting ended by self.:自己结束会议导致的会议结束
414100050	The meeting control module is not initialized.:会控模块未初始化
414100051	The watching special site reaches the maximum.:选看的特殊会场超过限制
414100052	The watching HD site reaches the maximum.:选看的高清会场超过限制
414100053	The watching SD site reaches the maximum.:选看的标清会场超过限制
414100054	The watching specification reaches the maximum.:选看的规格超过限制
414100055	Watching internal errors.:选看内部错误
414100056	Duplicated watch render.:选看的句柄重复
414100057	HTTP request error.:HTTP请求错误
414100058	Websocket request timeout.:websocket请求超时
414100059	Loading failed.:加载错误
414100061	The meeting ends due to the timeout of guest waiting before meeting or the timeout of the meeting control heartbeat.:会前等待页面时间超时或者会控心跳超时导致的会议结束
414100062	The meeting ends due to being canceled.:会议被取消预约导致的会议结束
414100063	The meeting ends due to network error when guest waiting before meeting.:会前等待页面网络异常导致的会议结束
414100064	http return 429 error.:http 返回429错误
414100065	self is chairman.:已是主持人
414100066	self is cohost.:已是联席主持人
414100067	no cohost and chairman.:无主持人和联席主持人

错误码	错误描述
414100068	no supported cohost and chairman.:无支持的主持人和联席主持人
414100069	Invalid user, enterprise or SP.:用户，企业或SP失效
415000001	DB is not initialized.:db未初始化
415000002	Common failure.:失败
415000003	Proxy detection has not been completed.:代理探测还没有完成
416000001	Invalid parameter.:参数错误
416000002	Track module is not initialized.:打点模块未初始化
416000003	Track event does not exist.:打点事件不存在
416000004	Not logged in.:未登录
416000005	Upload is not enabled.:未开启上传
419100001	Invalid parameter.:错误参数
419100002	Incorrect URI.:错误号码
419100003	Memory allocation failed.:内存分配失败
419100004	SDP error.:SDP错误
419100005	Connection does not exist.:连接不存在
419100006	Connection failure.:连接失效
419100007	Operation forbidden.:禁止操作
419100008	State error.:状态错误
419100009	No corresponding line data.:无对应线路数据
419100010	No corresponding registered instance.:无对应注册实例
419100011	No corresponding call instance.:无对应呼叫实例
419100012	No corresponding subscription instance.:无对应订阅实例
419100013	No corresponding Refer instance.:无对应Refer实例
419100014	No corresponding DIM instance.:无对应DIM实例
419100015	Failed to create header.:创建头域失败
419100016	Failed to parse header.:解析头域失败
419100017	Header not found.:没有找到头域
419100018	No SDP message body.:没有sdp消息体
419100019	Failed to get capability.:获取能力失败

错误码	错误描述
419100020	Failed to compare capability.:比较能力失败
419100021	Failed to open logical channel.:打开逻辑通道失败
419100022	Encoding SDP failed.:编码SDP失败
419100023	Decoding SDP failed.:解码SDP失败
419100024	Negotiate to unsupported SDP.:协商到不支持的SDP
419100025	TPTD initialization failed.:TPTD初始化失败
419100026	TPT initialization failed.:TPT初始化失败
419100027	TXN initialization failed.:TXN初始化失败
419100028	UA initialization failed.:UA初始化失败
419100029	GLUE initialization failed.:GLUE初始化失败
419100030	HLLM initialization failed.:HLLM初始化失败
419100031	VCOM initialization failed.:VCOM初始化失败
419100032	SDP initialization failed.:SDP初始化失败
419100033	Multiple instance error.:多实例错误
419100034	Insufficient registration control blocks.:注册控制块不足
419100035	Insufficient subscription control blocks.:订阅控制块不足
419100036	Insufficient call control blocks.:通话控制块不足
419100037	Insufficient IM control blocks.:对话外控制块不足
419100038	Failed to parse URI xml to string.:解析xml的uri失败
419100039	Encryption negotiation failed.:加密协商失败
419100040	Local capabilities are not configured.:没有设置本地能力
419100041	BFCP token error.:bfcpl令牌错误
419100042	DNS resolution error.:dns解析错误
419100043	Failed to get subscription state.:获取订阅状态失败
419100044	IPv4 is not supported.:不支持IPv4
419100045	IPv6 is not supported.:不支持IPv6
419100046	Unsupported host type.:不支持的主机类型
419100047	Failed to get timer.:获取定时器失败
420000001	Cannot receive sharing invitations.:无法接收共享邀请

错误码	错误描述
420000002	ReportDTO parameter verification failed.:reportDTO参数校验失败
420000003	Report folder creation failed.:举报文件夹创建失败
420000004	Report folder compression failure.:举报文件夹压缩失败
420000005	The path of the ZIP file to be uploaded is empty.:待上传的ZIP文件路径为空
420000006	The size of the ZIP file to be uploaded reaches the maximum.:待上传的ZIP文件大小超过限制
420000007	The same person reported more than 100 times within 24 hours.:同一个人24小时内举报次数超过100次
430100001	Common Error.:通用错误
430100002	Invalid parameters.:参数错误
430100003	SIP connection check failed.:SIP连接检查失败
430100004	Timeout.:超时
430100005	In process.:接口正在处理中
430100006	Not initialized or initialization is not complete.:未初始化或初始化尚未完成。
430100007	Not supported in this release.:此版本不支持。
430100008	Operation interrupt.:操作中断。
430100009	Contact not found.:联系人不存在
430100010	Args length overrun.:参数长度超限
430100011	APPID info illegal.:APPID 信息入参不合法
430100012	E2EEncrypt algorithm error.:端到端加密算法不对称
430200001	unlogin.:未登录
430200002	Repeated login.:重复登录
430200003	SIP connection timeout.:SIP连接超时
430200004	Invalid login password.:用于登录的密码参数不合法
430200005	Password or account empty.:账号或密码不能都为空
430300001	The calling number, account number or third-party account number cannot be empty.:呼叫号码、账号或者第三方账号不能为空
430300002	Call account empty.:点对点呼叫账号为空
430300003	Call nickname empty.:点对点呼叫名字为空

错误码	错误描述
430400001	Password required to join the meeting.:加入会议需要密码
430400002	Time zone error.:时区错误
430400003	The meeting duration is less than 1.:会议时长小于1
430400004	Joining the conference is interrupted.:入会被中断
430400005	Password input at least 4 digits.:密码输入至少4位
430400006	User voluntarily cancels.:用户主动取消
430400007	The guest password illegal.:来宾密码非法
430400008	Not in conference or in call.:不在会议或呼叫中
430400009	Already exist conf or call.:已存在一路会议或呼叫
430400010	Confid empty.:会议id不能为空
430400011	Subject too long.:会议主题长度不能超过50
430400012	VMR host password length illegal.:主持人密码长度不合法
430400013	VMR name empty.:VMR名字为空
430400014	Conf starttime early.:会议的开始时间早于当前时间
430400015	Share is locked.:共享被锁定
430400016	Not in data conference.:不在数据会议中
430400017	Role not support operation:角色不支持该操作
430500001	HTTP request, server exception.:HTTP请求，服务器异常
430500002	HTTP request, Connecting to the server times out.:HTTP请求，连接服务器超时
430500003	network unavailable.:网络不可用
430500004	file type illegal.:文件类型不合法
510000403	Server forbidden.:服务器拒绝
510000404	Not found.:未发现
510000480	Temporary unavailable.:临时失效
510000486	Busy here.:这里忙
510000487	Request terminated.:请求终止
510000488	Request is not acceptable here.:这里请求不可接受
510000603	Decline.:丢弃
510000841	Hangup without toast.:挂断不弹框

错误码	错误描述
510000842	Conference is locked.:会议被锁定
910000001	Server error.:服务器异常
910000002	SFU service failure.:sfu服务故障
910000003	Service unavaliable 503.:服务不可达503
910000004	Internal error.:内部错误
910000005	Kicked off.:被踢
910000006	Signature expired.:签名过期
910000007	Reconnect timeout.:重连超时
910000008	Network detection.:网络检测, UI不需要关注该错误码, 不对外体现
910000009	User is removed.:用户移除
910000010	The room is dismissed.:房间解散
990000001	Internal error.:内部错误
990000002	Message too large.:消息长度太大
990000003	Insufficient memory.:内存不足
990000004	Failed to send message synchronously.:同步发送消息失败
990000005	Parameter error.:参数错误
990000006	Call ID order error.:呼叫ID时序错误
990000007	Failed to set local screen.:设置本端画面失败
990000008	Failed to open remote screen.:开启远端画面失败
990000009	Device configuration failed.:设备配置失败
990000010	Initializing.:初始化中
990000011	Deinitializing.:去初始化中
990000012	Uploading log.:正在日志上传中
990000013	Failed to get media port.:获取媒体端口失败
990000014	The maximum number view window reached .:选看窗口超过规格数
990000015	Media negotiation failed.:媒体协商失败
990000016	Server no response.:服务器没有响应
990000017	Role switch failed.:角色切换失败
990000018	Failed to join room.:加入房间失败

错误码	错误描述
990000019	Failed to join room. Status is busy.:非空闲状态
990000020	Failed to join room. Server error.:加入房间失败，服务器异常
990000021	Failed to join room. Service unavaliable.:加入房间失败，服务不可达
990000022	Failed to join room. Authentication failed.:加入房间失败，鉴权失败
990000023	Failed to join room. Authentication retry.:加入房间失败，鉴权重试
990000024	Failed to join room. Clock synchronization.:加入房间失败，时钟同步
990000025	Failed to join room. URL error.:加入房间失败，url错误
990000026	Kicked off.:被踢
990000027	Share failed.:共享失败
990000028	Currently in external media output mode, this operation is forbidden.:当前为外部媒体输出模式，禁用该操作
990000029	Reconnection failed.:重连失败
990000030	Server break down.:服务器宕机
990000031	Signature has expired.:签名已过期
990000032	Failed to set remote window mode.:设置远端窗口模式失败
990000033	Failed to subscribe or unsubscribe audio.:订阅或取消订阅音频失败
990000034	Failed to connect to other room.:连接其他房间失败
990000035	Failed to disconnect other room.:断开连接其他房间失败
990000036	Role switching is not allowed.:不允许角色切换
990000037	Currently in third-party capture mode, this operation is forbidden.:当前为第三方采集模式，禁用该操作
990000038	Failed to set third-party audio capture.:设置第三方音频采集失败
990000039	Failed to set third-party video capture.:设置第三方视频采集失败
990000040	Failed to set share sound switch.:设置共享声音开关失败
990000041	Failed to start or stop upstream audio stream.:启停上行音频流失败

错误码	错误描述
990000042	Failed to start or stop upstream video stream.:启停上行视频流失败
990000043	User is removed.:用户被移除
990000044	The room is dismissed.:房间被解散
990000045	Failed to set remote View.:设置远端View失败
990000052	Failed to join room because room full.:房间已满，加入失败

5.7 版本更新内容

5.7.1 80.13.6 更新内容

1. 问题修复，体验更稳定

5.7.2 80.13.5 更新内容

1. 上传头像接口限制上传文件的格式和大小，具体请见 [上传个人头像](#)
2. 新增设置语言接口，具体请见 [设置语言](#)
3. 问题修复，体验更稳定

5.7.3 80.12.6 更新内容

1. 开启功能接口enableFeature 新增是否显示录制安全提示窗枚举值 HWM_FEATURE_TYPE_CONFIRM_RECORD。具体请见[枚举 HWMEnableFeatureType说明](#)
2. 问题修复，体验更稳定。

5.7.4 80.12.5 更新内容

1. 修复多个问题

5.7.5 80.10.6 更新内容

1. 新增异步初始化接口，具体请见 [异步初始化](#)

5.7.6 80.10.5 更新内容

1. 预约周期会议接口废弃cycleCount字段，具体请见 [预约周期会议](#)
2. 获取云会议室列表和修改云会议室信息接口新增参数，具体请见 [获取云会议室列表、修改云会议室信息](#)
3. 新增设置共享权限接口setShareScreenPermission, 具体请见[设置共享权限](#)
3. 问题修复

5.7.7 80.9.5 更新内容

1. 编辑会议和会议详情新增字段isLiveOn是否开启直播，具体请见 [编辑会议](#)、[获取会议详情](#)
2. 问题修复

5.7.8 80.8.5 更新内容

1. 创建会议、预约会议、编辑会议支持设置来宾密码，具体请见 [预约会议](#)、[创建会议](#)、[编辑会议](#)
2. 问题修复

5.7.9 80.7.5 更新内容

1. 问题修复

5.7.10 80.5.11 更新内容

1. 新增禁用提示音接口disableVoicePrompts，具体请见 [禁用提示音](#)
2. 问题修复

5.7.11 80.5.5 更新内容

1. [预约会议](#)、[编辑会议](#)、[获取会议详情](#)接口增加等候室开关功能，新增isOpenWaitingRoom字段
2. 问题修复

5.7.12 70.16.5 更新内容

1. 新增周期会议相关接口，具体请见[预约周期会议](#)、[编辑周期会议](#)、[取消周期会议](#)、[编辑周期子会议](#)、[取消周期子会议](#)
2. 入会参数PasswordJoinConfParam 新增 isStopConflictConf字段，能够强制结束和当前VMR会议资源冲突的会议，具体请见 [表3](#)
3. 会议通知栏支持在来电时可配置是否显示通知，具体请见 [表2](#)
4. sso 登录接口调整，具体请见 [loginBySSO](#)
5. 错误码整改，新版本错误码描述请见 [表1](#)

表 5-72 新旧版本错误码对应关系

70.14.6 版本及之前的错误码	70.16.5 版本之后的错误码	新版本错误描述
-1	430100001	Common Error.:通用错误
500	430200001	unlogin.:未登录
12	430200002	Repeated login.:重复登录

70.14.6 版本及之前的错误码	70.16.5 版本之后的错误码	新版本错误描述
10001	412100001	The message processing thread is not initialized.:消息处理线程未初始
10001	412100004	Memory allocation error.:分配内存错误
10001	412100005	XML parsing error.:XML解析出错
10001	412100009	Server error.:服务器异常
10001	412100014	Server redirect.:服务器返回重定向地址
10001	412100016	JSON parsing error.:JSON解析失败
10001	412200001	Internal error.:内部错误
10002	412100002	Parameter error.:参数错误
10002	412200002	Parameter error.:参数错误
10003	412100003	Timeout.:超时
10004	412100006	DNS resolution error.:DNS解析异常
10004	412100007	Request message failed.:请求消息异常
10004	412100013	Network error.:网络异常
10004	412200004	Network Error.:网络错误
201	430200003	SIP connection timeout.:SIP连接超
35	430400001	Password required to join the meeting.:加入会议需要密码
2	430200004	Invalid login password.:用于登录的密码参数不合法
420010001	430100003	SIP connection check failed.:SIP连接检查失败
10006	206010000	User name or password error.:用户名或密码错误
10006	206030000	The user does not exist.:系统中不存在该用户

70.14.6 版本及之前的错误码	70.16.5 版本之后的错误码	新版本错误描述
10006	206010025	APPID verification failed.:APPID校验失败
10007	412100010	Account is locked.:帐号被锁定
10007	206030015	The user is locked and will be locked in 60s.:用户已被锁定，请60秒后重试

5.7.13 70.14.5 更新内容

1. 预约、编辑会议支持设置“来宾在主持人之前加入会议”，并可设置提前入会时间，具体参数请见[预约会议参数说明](#)。
2. 预约云会议室时支持选择预约固定云会议室ID或随机云会议室ID，具体请见[VmridType枚举说明](#)。
3. APPID登录支持指定部门ID，具体请见[AppIdAuthParam参数说明](#)。
4. 问题修复。

5.7.14 70.13.5 更新内容

1. 登录、登出接口新增uuid、account、thirdAccount字段，具体请见[LoginResult类型说明](#)、[LogoutResult类型说明](#)。如果是老版本升级到70.13.5版本,需要修改回调函数的定义。
2. 第三方视频推流接口支持MMR会议，具体请见[第三方视频推流](#)。
3. 业务消息通知中新增onConflIncoming接口，允许在入会前获取来电信息，具体请见[业务消息通知](#)。
4. 问题修复

5.7.15 70.12.6 更新内容

1. 修复由于弱网导致视频清晰度下降，网络恢复后视频清晰度无法恢复的问题。

5.7.16 70.12.4 更新内容

1. 升级RxJava版本
2. 问题修复

5.7.17 70.11.5 更新内容

1. 新增配置日志保存天数的能力，具体请见[初始化信息OpenSDKConfig](#)
2. HMS Scankit升级到2.1.0.300版本
3. 问题修复

5.7.18 70.10.5 更新内容

1. 去除无效资源
2. 问题修复

5.7.19 70.9.5 更新内容

1. 新增SSO登录方式，具体请见[SSO方式登录](#)
2. 移除tinypinyin库

5.7.20 70.8.6 更新内容

问题修复，体验更稳定

5.7.21 70.8.5 更新内容

说明

由于原Android support库已不再被Google维护演进，将从28版本升级替换为AndroidX。

因此华为云会议从8月发布的70.8.5版本开始，将不再在support库基础上开发，而是替换为AndroidX进行迭代。

如果您仍在使用原Android support库依赖：

- 我们推荐您与我们一样更新替换使用新的AndroidX依赖，以免您在后续开发时无法正常对接我们采用AndroidX的SDK版本，相关文档请见[快速入门](#)；
- 如果您无法立即更新，我们会提供原有基于support库开发的SDK入口，并持续维护6个月。在维护期间，您仍可以集成我们的旧版SDK（70.7.5及之前版本）进行开发，我们会对于此类版本发现的问题进行修复与更新，但是之后迭代的新功能将只在AndroidX版本SDK上实现。

华为云会议期望给您带来更好的产品体验，如有问题，请拨打4000-955-988 | 950808 按1转1 进行快速解答！

1. 从此版本开始，Android SDK切换为AndroidX依赖，详细配置请见[快速入门](#)
2. 支持获取与会者列表（含网络研讨会），详见[获取与会者列表](#)

5.7.22 70.7.5 更新内容

1. 支持视频镜像能力，打开后，观看自己显示为镜像效果

图 5-35 视频镜像



2.支持安卓11的API 30版本

3.okhttp升级到4.9.1

5.7.23 70.6.5 更新内容

1.删除了ZXing的依赖

2.tinypinyin 升级到3.0.0

3.aspectjrt升级到1.9.6

4.gson升级到2.8.6

5.7.24 70.5.8 更新内容

1.会中与会者和聊天模块适配横屏体验

2.支持聊天弹幕

5.7.25 70.4.5 更新内容

1.预约会议接口添加返回值参数说明，具体请见[表3](#)

2.sdk初始化约束，限制初始化必须在Application的OnCreate主线程中执行

3.新增获取用户当前状态接口，具体请见[获取用户状态](#)

4.会中新增主持人邀请他人共享，主持人也可快速取消共享

5.移动端的前后摄像头切换按钮放到主界面上

6.会议日志大小优化

7.libs文件夹中aar文件整合，使用时需先清空之前历史版本文件，重新导入

5.7.26 70.3.5 更新内容

1.移除会中自定义分享链接，自定义分享链接功能可以通过自定义会中分享面板实现，具体使用请见[自定义会中分享面板](#)

2. 去掉短信权限的要求，具体请见[添加权限步骤6](#)
3. 错误码变更，具体请见[错误码参考](#)
4. 观看共享时叠加发言人视频窗口，并支持下拉关闭
5. 支持手机、平板同时登录

5.7.27 60.15.6 更新内容

1. 新增自定义来电自动处理接口，具体使用请见[自定义来电自动处理通知事件](#)
2. 新增自定义设置来电响铃和震动，具体使用请见[自定义设置来电响铃和震动](#)
3. sdk 资源包aar依赖变更，具体使用请见[快速入门](#)添加会议依赖
4. 自定义隐藏外部标签，具体使用请见[自定义隐藏外部标签](#)

5.7.28 60.13.2 更新内容

1. 添加与会者，具体使用请见[添加与会者](#)
2. 离开会议，具体使用请见[离开会议](#)
3. 结束会议，具体使用请见[结束会议](#)
4. 错误码更新，废弃掉一些未使用的错误码。请见[错误码参考](#)

5.7.29 60.12.5 更新内容

1. 创建会议支持指定会议室（vmrId）和入会范围（joinConfRestrictionType），具体使用请见[表2](#)
2. 错误码更新，具体使用请见[错误码参考](#)
3. 新增头像上传接口，具体使用请见[上传个人头像](#)
4. 新增头像拦截接口，可以使用自己的头像显示，具体使用请见[头像拦截显示](#)
5. 自定义菜单样例更新，具体使用请见[视频点对点通话自定义菜单](#) [音频会议自定义菜单](#) [视频会议自定义菜单](#)

5.7.30 60.11.1 更新内容

1. 新增会议支持自动录制，新增允许自动录制会议isAutoRecord字段
注意：会议开始后是否自动开启录制，默认关闭。开启自动录制会议时必须开启录制会议
 - 创建会议[表2](#)
 - 预约会议、编辑会议[表1](#)
2. 新增自定义会议来电接听按钮显示，具体使用请见[自定义视频会议来电时接听方式按钮](#)

5.7.31 60.10.5 更新内容

- 1、新增是否开启第三方数据采集配置：
 - 新增是否开启第三方数据采集配置，请见[开启第三方视频推流配置](#)
- 2、新增自定义接入站点：

- 新增自定义接入站点配置，请见[设置接入站点](#)
- 3、自定义会中分享面板参数变更：
 - 自定义会中分享面板，入参新增ShareTypeEnum，请见[自定义会中分享面板](#)
- 4、加入会议接口参数变更：
 - 加入会议参数变更，JoinConfParam抽为抽象类，修改为使用子类 PasswordJoinConfParam或RandomJoinConfParam，请见[场景7：加入会议](#)

5.7.32 60.8.1 更新内容

1、初始化增加App ID，可以传入企业App ID：

初始化设置App ID，可以用于App ID方式登录，App ID需要到会议服务器上申请
具体请见[5.3-8SDK初始化](#)

2、新增AppId登录、预约会议、编辑会议、会议列表、会议详情、取消会议等接口：

以下为接口示例，具体使用请见[5.5-接口参考](#)

- 新增接口

```

/**
 * 通过App ID登录
 */
void loginByAppId(AppIdAuthParam appIdAuthParam, HwmCallback<LoginResult> callback);

/**
 * 预约会议
 */
void bookConf(BookConfParam bookConfParam, HwmCallback<ConfInfo> hwmCallback);

/**
 * 编辑会议
 */
void editConf(EditConfParam editConfParam, HwmCallback<Integer> hwmCallback);

/**
 * 获取会议列表
 */
List<ConfBaseInfo> getConfList();

/**
 * 根据会议id 获取会议详情
 */
void getConfDetail(String confId, HwmCallback<ConfDetailModel> hwmCallback);

/**
 * 取消已经预约的会议
 */
void cancelConf(String confId, HwmCallback<Integer> hwmCallback);

/**
 * 获取VMR(云会议室)列表，预约会议可用虚拟会议室进行预约
 */
void getVmrList(HwmCallback<List<VmrInfoModel>> hwmCallback);

/**
 * 修改虚拟会议室信息
 */

```

```
void changeVmrInfo(HwmChangeVmrInfo hwmChangeVmrInfo, HwmCallback<Integer>
hwmCallback);
```

3、新增自定义是否开启高清优先、自定义会议接听摄像头麦克风开关等配置接口：

具体请见[自定义是否开启会议高清优先](#)

[自定义会议接听时摄像头/麦克风开关](#)

4、创建会议参数新增是否开启录制：

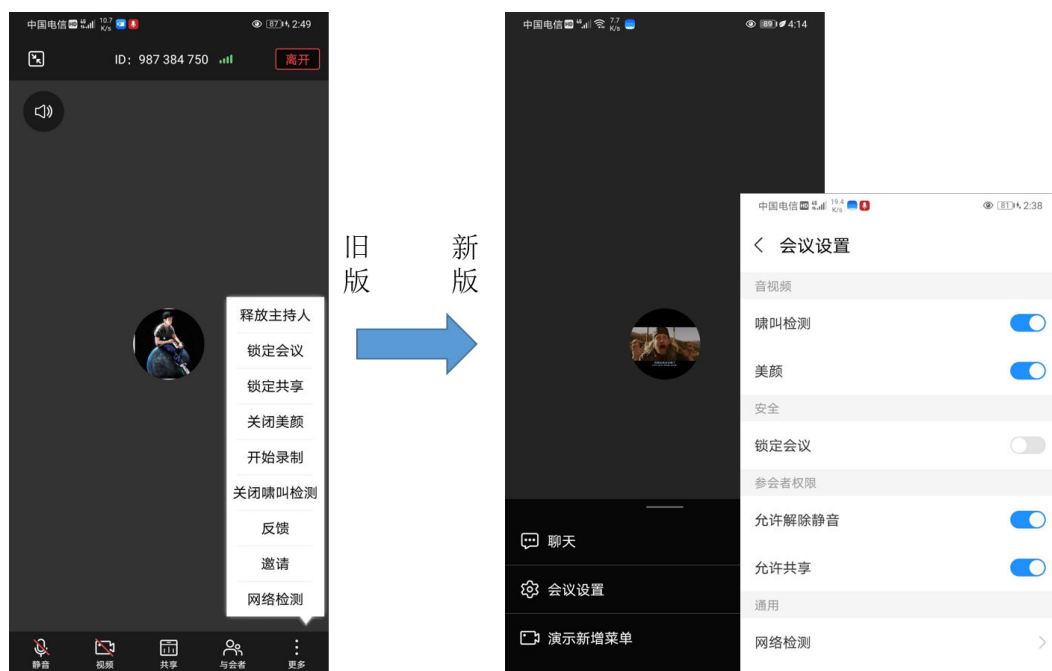
具体请见[场景6：创建会议](#)

5.7.33 60.7.1 更新内容

1、会中UI界面优化、菜单目录结构和接口发生变更：

- 会中界面-顶部会议信息增加会议锁定，页面UI优化
- 原有会中toolbar“更多”菜单中的部分功能调整到“更多”-“会议设置”菜单中

图 5-36 会中菜单



2、会中增加聊天功能，支持自定义配置是否显示，详见[自定义开关会中聊天功能](#)

3、会控菜单自定义接口优化，部分接口替换：

- 会中与会者列表界面，底部toolbar一级菜单支持自定义配置
- 可根据音频/视频呼叫、音频/视频会议四个场景，分别配置菜单栏
- 支持区分与会人身份，自定义会议标题栏点击弹出菜单的内容
- 弃用旧版菜单配置IConfMenuHandle接口

以下为接口示例，具体使用请见[5.5-接口参考](#)

```

/**
 * 会中底部会控工具栏的配置策略，包括：一级菜单；点击一级菜单里更多菜单；点击更多菜单里的会议设置
 */
public interface IToolbarMenuStrategy {

    /**
     * 设置会中底部会控一级菜单
     * @return 会中底部会控一级菜单
     */
    List<IConfMenu> buildMenuItems();

    /**
     * 设置会中底部会控点击更多弹出的菜单项
     * @return 会中底部会控点击更多弹出的菜单项
     */
    List<IConfMenu> buildMoreMenuItems();

    /**
     * 设置会中点击更多里会议设置界面的菜单项，支持二级分组
     * @return 会中点击更多里会议设置界面的菜单项
     */
    List<IConfMenu> buildSettingMenuItems();
}

```

```

/**
 * 菜单的点击事件接口定义
 */
public interface OnMenuItemClickListener {

    /**
     * 当自定义菜单被点击时触发
     * @param confMenu 被点击的菜单
     * @param confInfo 会议信息
     */
    void onCustomMenuItemClick(IConfMenu confMenu, ConfInfo confInfo);
}

```

接口使用:

```

//自定义菜单
if(DemoUtil.needCustom(NeedCustomType.CONF_MENU)){
    sdkConfig.setConfMenuHandle(new CustomConfMenu());

    ToolbarMenuProxy toolbarMenuProxy = new ToolbarMenuProxy();
    //定制 音频呼叫的会中toolbar菜单
    toolbarMenuProxy.setCallAudioToolbarHandle(new CustomCallAudioToolbarHandle());
    //定制 视频呼叫的会中toolbar菜单
    toolbarMenuProxy.setCallVideoToolbarHandle(new CustomCallVideoToolbarHandle());
    //定制 音频会议的会中toolar菜单
    toolbarMenuProxy.setConfAudioToolbarHandle(new CustomConfAudioToolbarHandle());
    //定制 视频会议的会中toolbar菜单
    toolbarMenuProxy.setConfVideoToolbarHandle(new CustomConfVideoToolbarHandle());
    sdkConfig.setToolbarMenuProxy(toolbarMenuProxy);
    //定制会中入会者列表界面的菜单
    sdkConfig.setParticipantMenuStrategy(new CustomParticipantMenuHandle());
    //自定义菜单的点击事件处理
    sdkConfig.setOnMenuItemClickListener(new CustomOnMenuItemClickListener());
}

```

6 iOS SDK

6.1 概述

华为云会议终端iOS SDK提供了一套完整的接口集合，开发者可以通过调用SDK开放的API，在自己的应用程序中快速集成音视频会议，包括创建/预约/加入会议、进行屏幕共享、点对点呼叫等。

6.2 修订记录

表 6-1 修订记录

日期	版本	修订内容
2023-02-22	80.13.6	<ol style="list-style-type: none"> 1. 编译环境升级到XCode14。 2. 问题修复，体验更稳定。 具体使用请见 80.13.6更新内容
2023-02-06	80.13.5	<ol style="list-style-type: none"> 1. 子线程初始化SDK。 2. 增加与会者列表actionSheet，点击头像进入详情。 3. 问题修复，体验更稳定。 具体使用请见 80.13.5更新内容
2023-01-05	80.12.6	<ol style="list-style-type: none"> 1. 开启功能接口enableFeature:isEnabled: 新增分组讨论枚举值HWMFeatureTypeRecoedAlert。 2. 问题修复，体验更稳定。 具体使用请见 80.12.6更新内容
2022-12-23	80.12.5	<ol style="list-style-type: none"> 1. 新增HWMRMedia.framework库。 2. 上传个人头像上传条件变更（增加类型和大小判断）。 3. 问题修复，体验更稳定。 具体使用请见 80.12.5更新内容

日期	版本	修订内容
2022-11-15	80.10.5	<ol style="list-style-type: none"> 1. 获取云会议列表 (HWMVmrInfoModel) 和修改云会议室信息 (HWMModifyVMRParam) 新增字段 confAllowJoinUser允许入会范围限制、isOpenWaitingRoom等候室开启开关。 2. 新增设置共享权限接口setShareScreenPermission。 3. 问题修复, 体验更稳定。 <p>具体使用请见80.10.5更新内容</p>
2022-10-09	80.9.5	<ol style="list-style-type: none"> 1. 编辑会议HWMOrderConfParam.h新增字段 isLiveOn。 2. 开启功能接口enableFeature:isEnabled: 新增分组讨论枚举值HWMFeatureTypeBreakoutConf。 3. 取消周期子会议CancelSubCycleConf中字段 subConfUUID改成subConfID。 4. 问题修复, 体验更稳定。 <p>具体使用请见80.9.5更新内容</p>
2022-09-02	80.8.5	<ol style="list-style-type: none"> 1. 创建会议、预约会议、编辑会议支持设置来宾密码。 2. 问题修复, 体验更稳定 <p>具体使用请见80.8.5更新内容</p>
2022-07-30	80.7.5	<ol style="list-style-type: none"> 1. App支持自定义多语言 2. 问题修复, 体验更稳定 <p>具体使用请见80.7.5更新内容</p>
2022-06-06	80.5.11	<ol style="list-style-type: none"> 1. 新增禁用提示音接口 2. 新增录制按钮操作定制 <p>具体使用请见80.5.11更新内容</p>
2022-05-18	80.5.5	<ol style="list-style-type: none"> 1. 预约会议、编辑会议、获取会议详情接口增加等候室开关功能 2. 问题修复, 体验更稳定 <p>具体使用请见80.5.5更新内容</p>
2022-04-15	70.16.5	<ol style="list-style-type: none"> 1. 新增周期会议相关接口 2. 新增设置网络模式接口 3. 新增音频共享 4. id入会HWMJoinConfParam.h 新增字段 isStopConflictConf 5. 初始化SDK新增字段会议来电时禁止使用本地推送 6. sso 登录接口调整 7. 错误码整改, 发布新版本错误码描述以及新旧版本错误码对应关系。 <p>具体使用请见70.16.5 更新内容</p>

日期	版本	修订内容
2022-02-19	70.14.5	<ol style="list-style-type: none"> 1. 预约、编辑会议支持设置“来宾在主持人之前加入会议”，并可设置提前入会时间。具体参数请见预约会议参数说明。 2. 预约云会议室时支持选择预约固定云会议室ID或随机云会议室ID。具体请见表6。 3. APPID登录支持指定部门ID。具体请见表3。 具体使用请见 70.14.5 更新内容
2022-01-19	70.13.5	<ol style="list-style-type: none"> 1. login 和 logout 相关接口的回调中增加用户的 userUid（原来uuid字段）、account、thirdAccount 2. 支持设置是否显示外部标签字段hideExternalLabel增加场景（转移主持人、聊天） 3. 新增被邀加入会议来电通知 具体使用请见 70.13.5 更新内容
2021-12-23	70.12.6	<ol style="list-style-type: none"> 1. 修复弱网情况下导致清晰度降低后不能恢复的问题 具体使用请见 70.12.6更新内容
2021-12-13	70.12.4	问题修复，体验更稳定 具体使用请见 70.12.4 更新内容
2021-12-03	70.11.5	<ol style="list-style-type: none"> 1. 初始化接口参数新增字段：logKeepDays 日志保存天数 2. UISDK iPhone版本最低支持iOS10.0 3. HWRTCEngine.framework更名为 HWMMREngine.framework 具体使用请见 70.11.5更新内容
2021-10-22	70.10.5	<ol style="list-style-type: none"> 1. 接收共享桌面时支持大小画面切换 2. 问题修复，体验更稳定 具体使用请见 70.10.5 更新内容
2021-09-16	70.9.5	<ol style="list-style-type: none"> 1. 新增SSO登录方式 具体使用请见 70.9.5 更新内容
2021-08-28	70.8.6	问题修复，体验更稳定 具体使用请见 70.8.6更新内容
2021-08-18	70.8.5	<ol style="list-style-type: none"> 1. 支持获取与会者列表（含网络研讨会） 具体使用请见 70.8.5更新内容
2021-07-16	70.7.5	<ol style="list-style-type: none"> 1. 支持会中设置本会场视频画面是否镜像 具体使用请见 70.7.5更新内容
2021-06-10	70.6.5	<ol style="list-style-type: none"> 1. 新增结束会议的状态通知 具体使用请见 70.6.5更新内容

日期	版本	修订内容
2021-05-17	70.5.8	<ol style="list-style-type: none"> 1. 会中与会者和聊天模块适配横屏体验 2. 支持聊天弹幕 具体使用请见 70.5.8更新内容
2021-04-15	70.4.5	<ol style="list-style-type: none"> 1. 新增获取用户当前状态接口 2. 会中新增主持人邀请他人共享，主持人也可快速取消共享 3. 移动端的前后摄像头切换按钮放到主界面上 4. 会中界面优化 5. 会议日志大小优化 具体使用请见 70.4.5更新内容
2021-03-12	70.3.5	<ol style="list-style-type: none"> 1. 观看共享时叠加发言人视频窗口，并支持下拉关闭 2. 支持手机和平板同时登录 具体使用请见 70.3.5更新内容
2021-02-10	60.15.5	<ol style="list-style-type: none"> 1. 会中自定义会议页面顶部titlebar优化（支持自定义cell） 2. 支持隐藏与会者列表外部标签 3. 新增设置来电响铃和震动 4. 新增呼叫和会议来电时接听策略设置（自动接听、自动拒接、默认） 具体参见 60.15.5更新内容
2021-01-08	60.13.2	<ol style="list-style-type: none"> 1. 新增会中添加与会者接口 2. 新增离开会议接口 3. 新增结束会议接口 具体参见 60.13.2更新内容
2020-12-07	60.12.5	<ol style="list-style-type: none"> 1. Extension添加配置“TARGETS—>ScreenShareExtension—>Build Settings->Other Linker Flags”中添加-lc++。 2. 创建会议支持指定会议室和入会范围 3. 新增会中会议详情上报 4. 新增头像上传接口 5. 新增头像拦截接口 6. 预约会议接口返回会议详情信息 具体参见 60.12.5更新内容
2020-11-07	60.11.6	<ol style="list-style-type: none"> 1. 新增支持会议自动录制 2. 新增自定义会议来电接听按钮显示 3. 新增呼叫设置昵称 4. 新增SDK库：HWRTCEngine.framework 具体参见 60.11.6更新内容

日期	版本	修订内容
2020-10-15	60.10.5	<ol style="list-style-type: none"> 1. 初始化增加登录站点类型 2. 支持使用random入会 3. 增加sp鉴权登录 4. 优化水印显示 具体参见 60.10.5更新内容
2020-09-04	60.8.1	<ol style="list-style-type: none"> 1. 初始化增加App ID, 可以传入企业App ID 2. 新增AppId登录、预约会议、编辑会议、会议列表、会议详情、取消会议等接口 3. 语音会议界面调整 4. IMSDK.framework合入到UISDK.framework 5. 问题修复, 体验更稳定 具体参见 60.8.1更新内容
2020-07-30	60.7.1	<ol style="list-style-type: none"> 1. 会议界面会议信息和更多界面UI调整 2. 会中增加聊天功能, 可配置是否显示 3. 音视频通话和会议, 底部会控和更多菜单自定义接口优化, 原接口删除 4. 问题修复, 体验更稳定 具体参见 60.7.1更新内容
2020-06-29	60.6.1	<ol style="list-style-type: none"> 1. 社交分享接口优化 2. 问题修复, 体验更稳定
2020-06-14	60.5.1	<ol style="list-style-type: none"> 1. 支持会控界面自定义 2. 问题修复, 体验更稳定
2020-05-28	60.4.0	<ol style="list-style-type: none"> 1. 去掉cordova 依赖, 提升性能, 初始化和入会速度提升 2. 重新设计接口API, 接入更方便 3. 重新开发demo, 接入更简单 4. 优化架构, 部分页面从web转为原生开发, 体验更流畅
2019-09-02	6.1.203	新增如下接口: 注销 其他接口参数调整、屏幕共享新增groupId在主应用设置
2019-07-23	6.1.195	新增如下接口: 匿名入会:支持用户匿名入会(未登录场景下的入会)
2019-06-21	6.1.187	新增如下接口: 发起呼叫:支持用户主动发起点对点呼叫
2019-05-20	6.1.177	新增如下接口: 会议中屏幕共享

日期	版本	修订内容
2019-04-13	6.1.27	新增如下接口： <ul style="list-style-type: none"> 创建会议(携带与会人信息)：支持在创建即时会议的时候携带与会者，与会者会收到呼叫入会的来电 新增如下能力： <ul style="list-style-type: none"> 来电接听：支持应用在前台收到其他人的来电接听后加入会议。 应用集成，详见基于应用的集成。
2019-03-29	6.1.23	首次发布。

6.3 快速入门

本小节以实现Demo功能为例，介绍如何使用iOS SDK进行二次集成开发。

6.3.1 开发准备

1. 环境及项目要求参见“[环境要求](#)”
2. **可选**: 在Mac电脑上解压下载到的“hwmsdk-ios.zip”软件包。软件包下载地址参见“[下载地址](#)”
3. 项目和Extension开发所需要的证书和文件。创建和管理地址参见“[证书地址](#)”

6.3.2 快速集成

步骤1 将下载的软件包全部解压得到libs（集成使用）和HWMSDKDemo（示例Demo）。

图 6-1 解压出来的文件

名称	修改日期	大小	种类
▶ HWMSDKDemo ← Demo 示例	今天 下午 2:45		-- 文件夹
▶ libs ← SDK 资源	今天 下午 3:46		-- 文件夹

步骤2 打开“Xcode”软件。

步骤3 单击“File->New->Project...” ，选择“Single View App->Next” ，填写应用信息，选择“Create”完成新工程创建，具体步骤可参考下图。

图 6-2 选择创建新工程

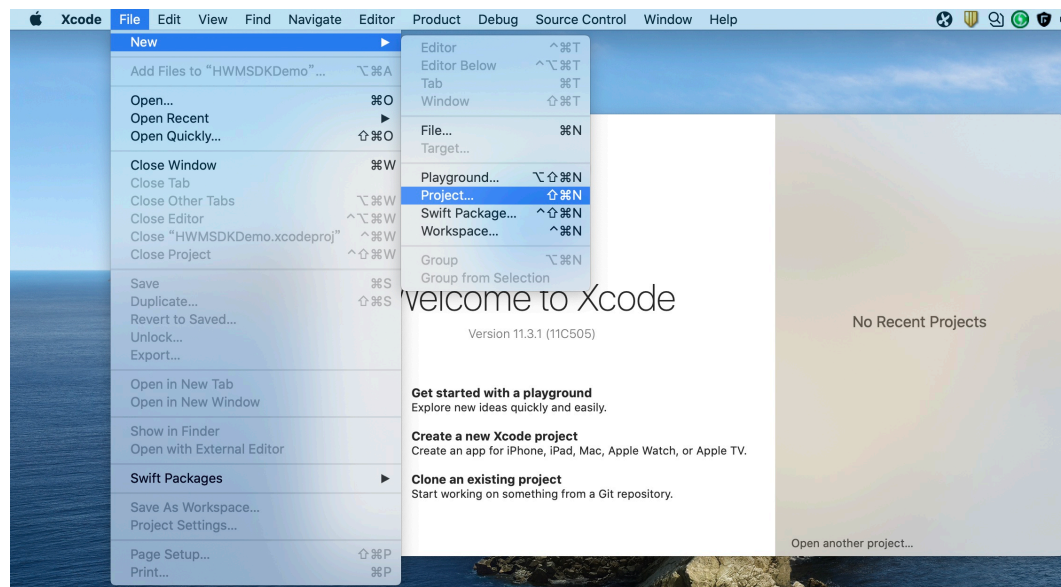


图 6-3 选择新工程模板为 Single View App

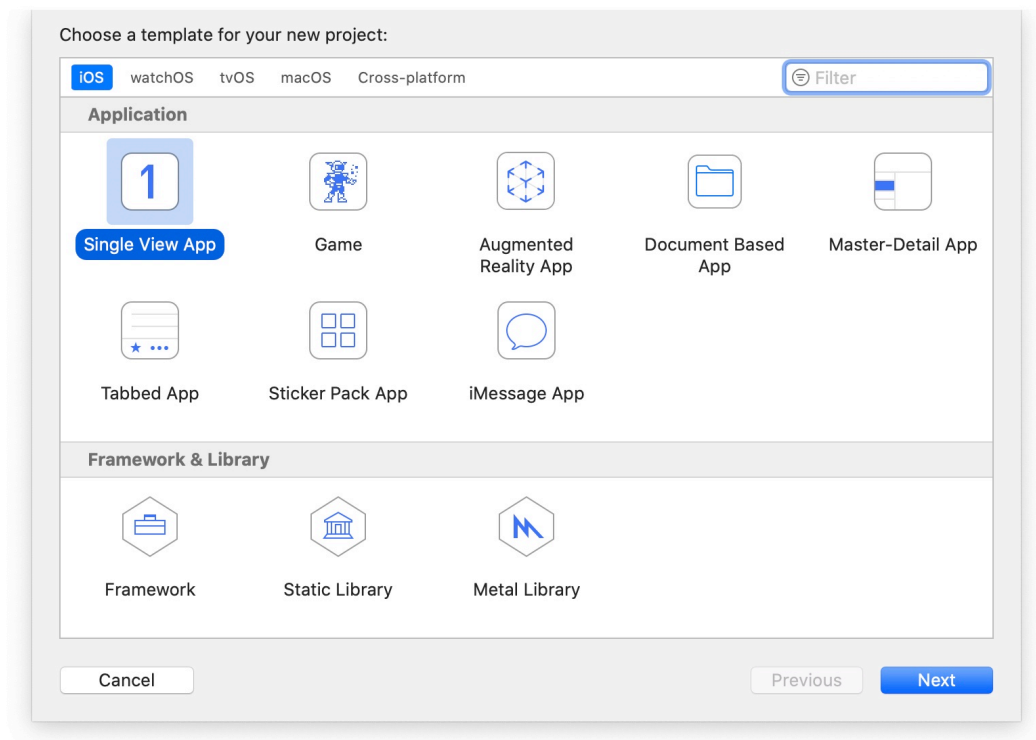


图 6-4 新建工程命名

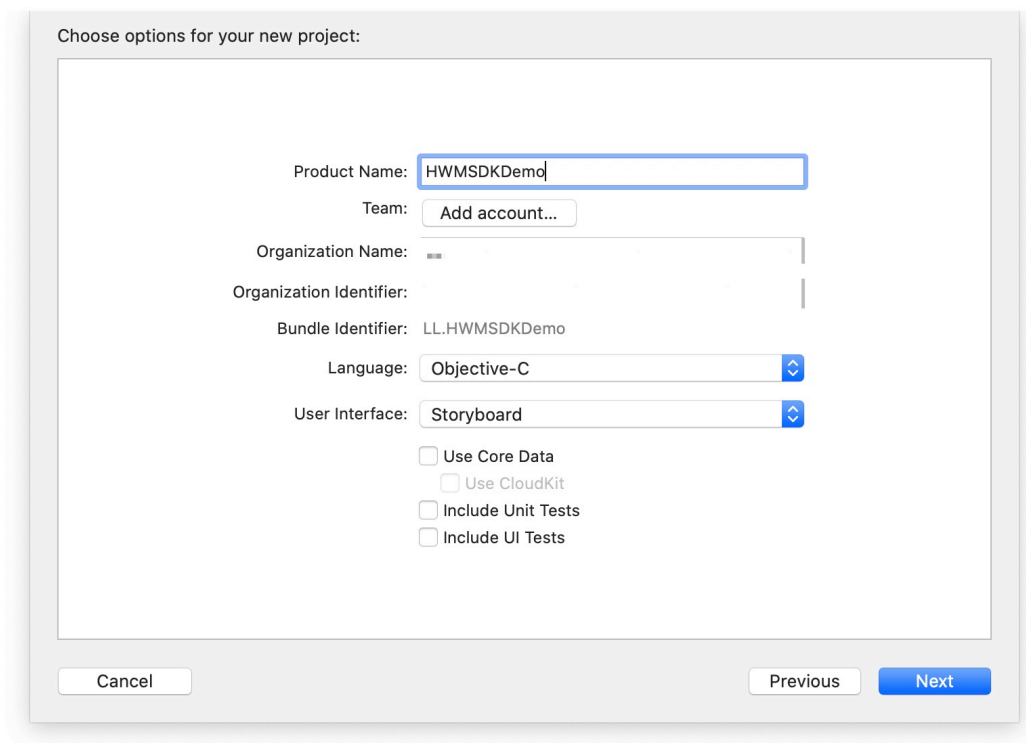
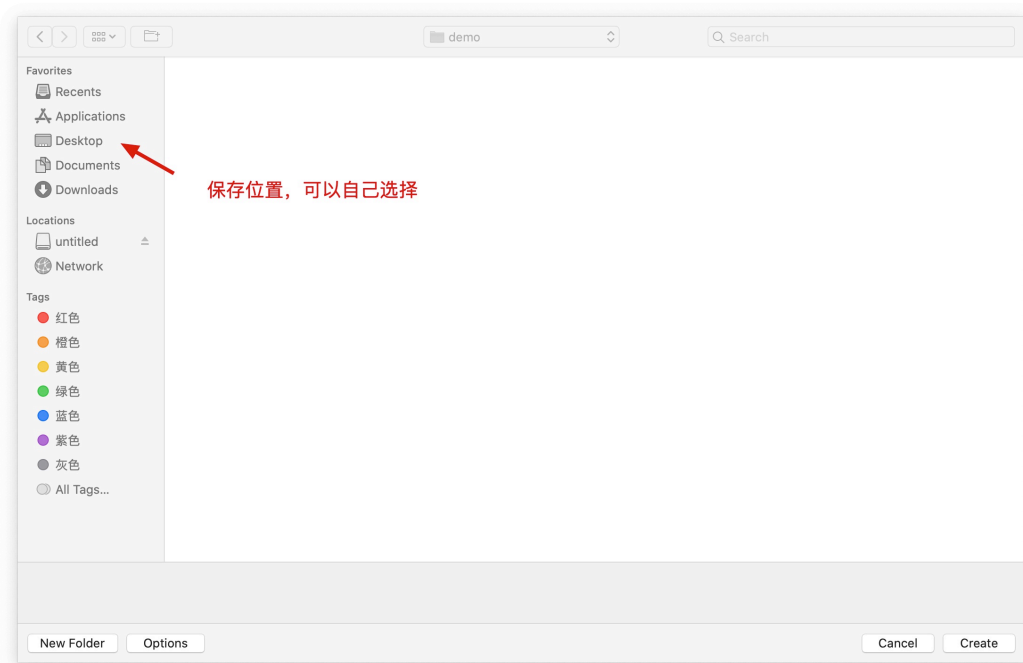


图 6-5 选择保存工程路径，Create 创建项目。



步骤4 将**步骤1**中解压得到的HWMUISDK.framework、HWMMREngine.framework、HWMMRMedia.framework添加到项目中。

图 6-6 添加解压出来的资源

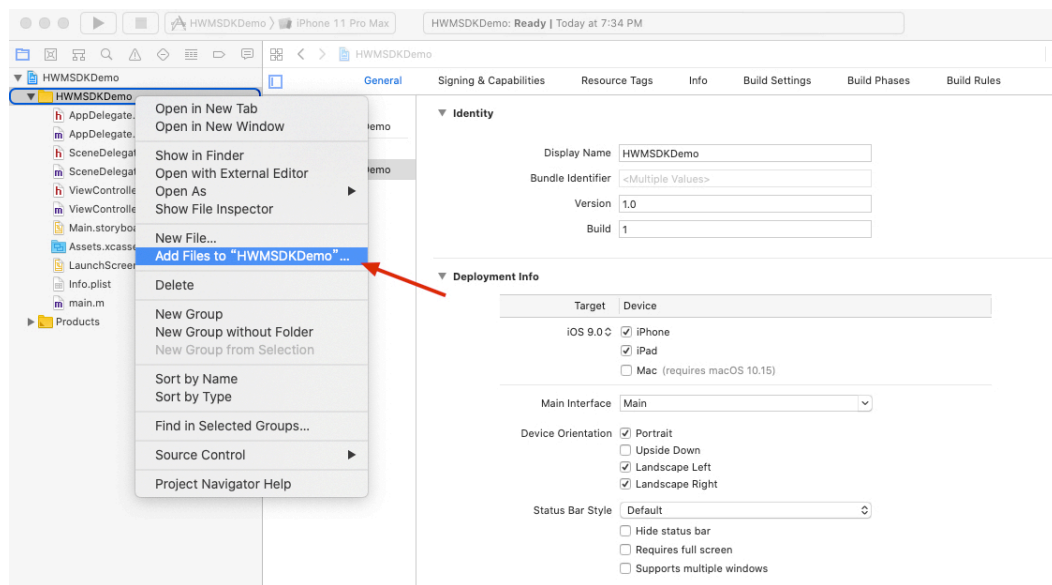
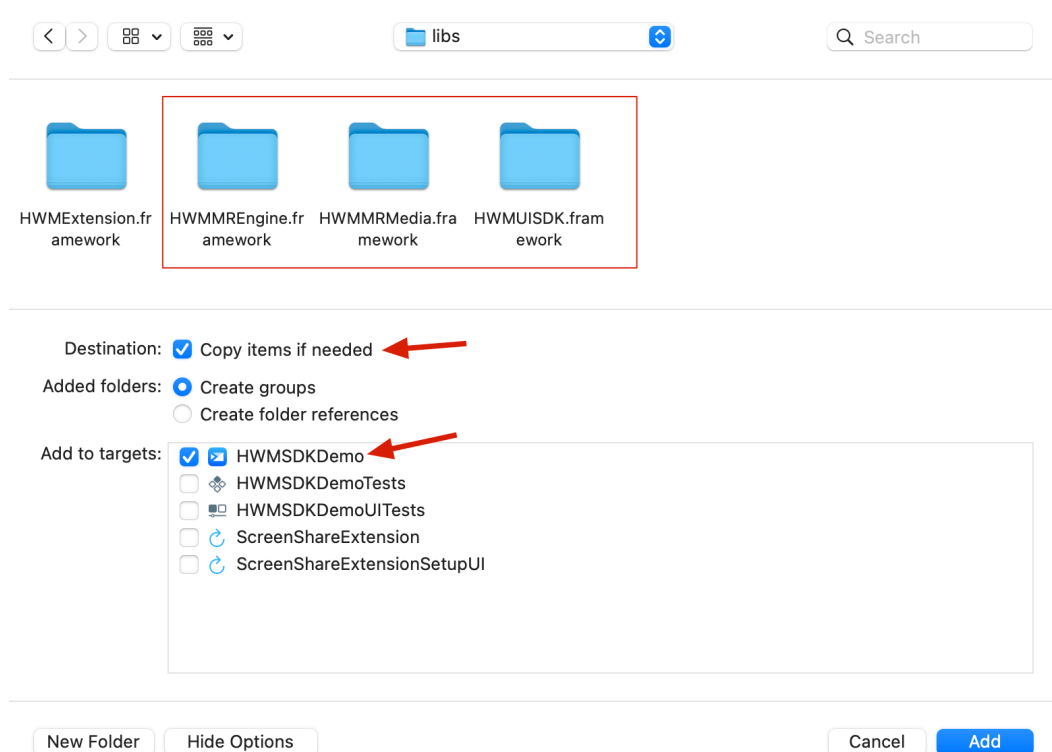
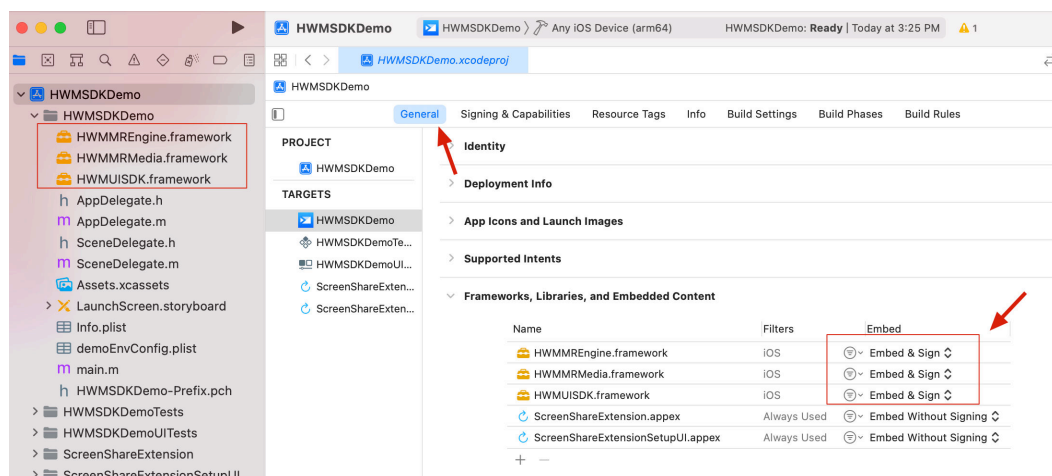


图 6-7 选择添加 HWMUISDK.framework、HWMREngine.framework、HWMRMedia.framework



添加后在“TARGETS->General->Frameworks、Libraries、and Embedded Content”中修改为Embed & Sign。

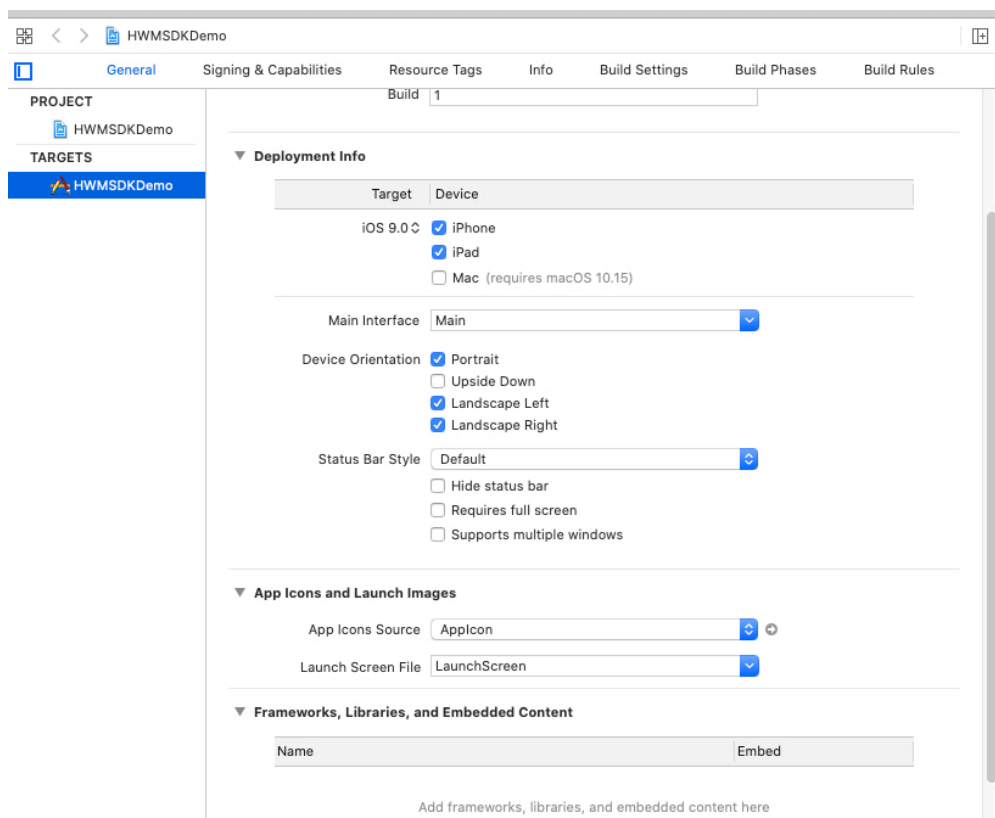
图 6-8 添加完成后的效果，



步骤5 工程文件配置。

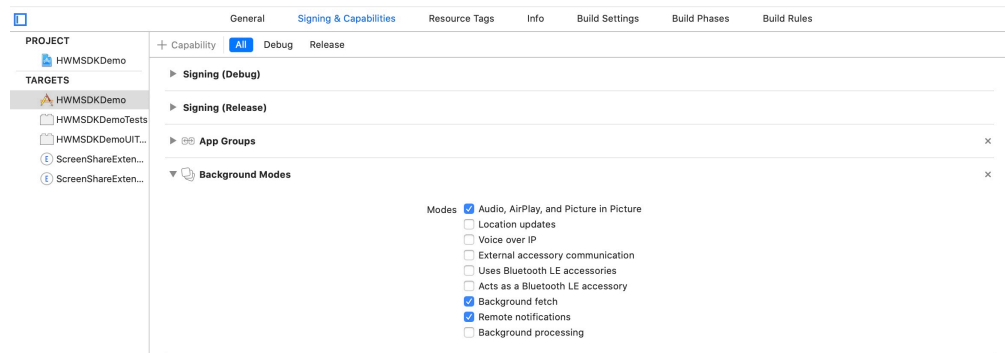
1. General配置，需要配置Deployment Info，开发证书请根据自己需要进行配置。

图 6-9 General 中 Deployment Info 配置



2. Capabilities配置，需要打开应用的Background Modes(支持切后台运行会议，提交审核时附带测试账号给Apple详情描述会议场景否则可能被拒审)。

图 6-10 Background Modes 配置



3. Build Settings配置，需要配置Enable Bitcode设置为NO（SDK暂时不支持YES），Valid Architectures设置为arm64（不支持模拟器运行），添加Other Link Flags为-ObjC。

图 6-11 Build Setting 配置

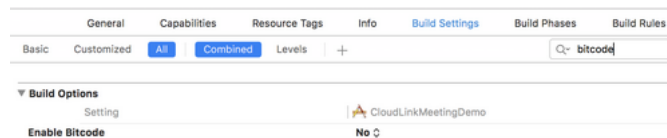


图 6-12 Valid Architectures 设置

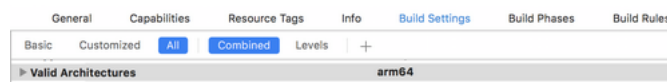
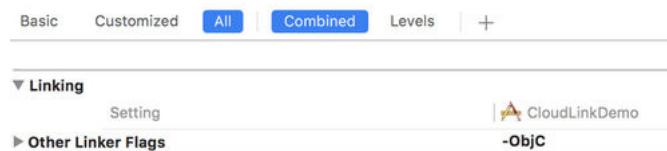
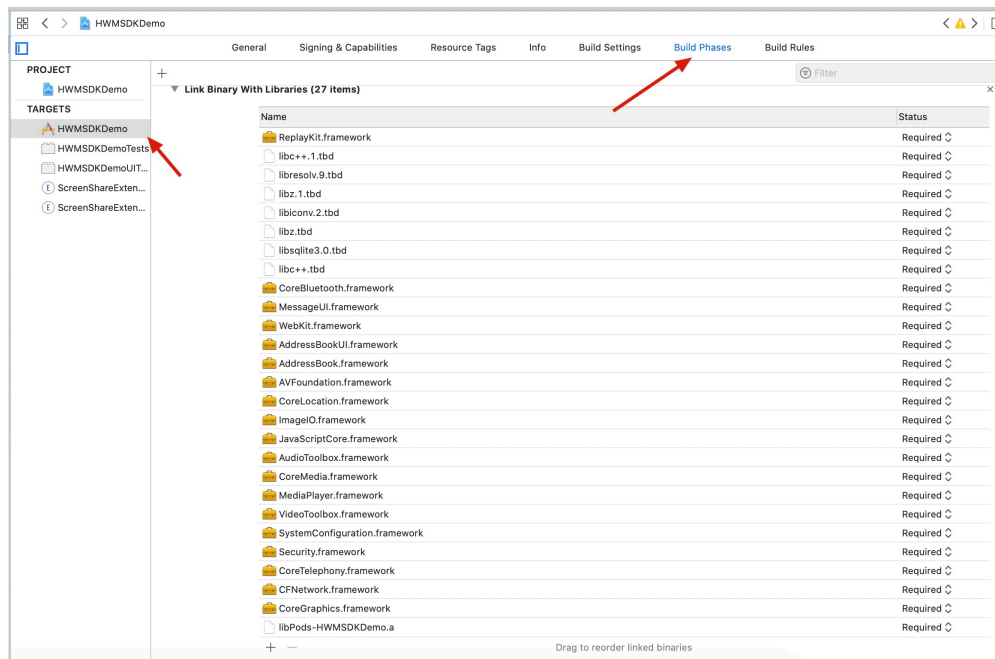


图 6-13 添加 Other Link Flags 为-ObjC



4. Build Phases配置链接文件Link Binary With Libraries。配置如下图所示

图 6-14 Link Binary With Libraries 配置，选择 Link Binary With Libraries 下面的+添加文件

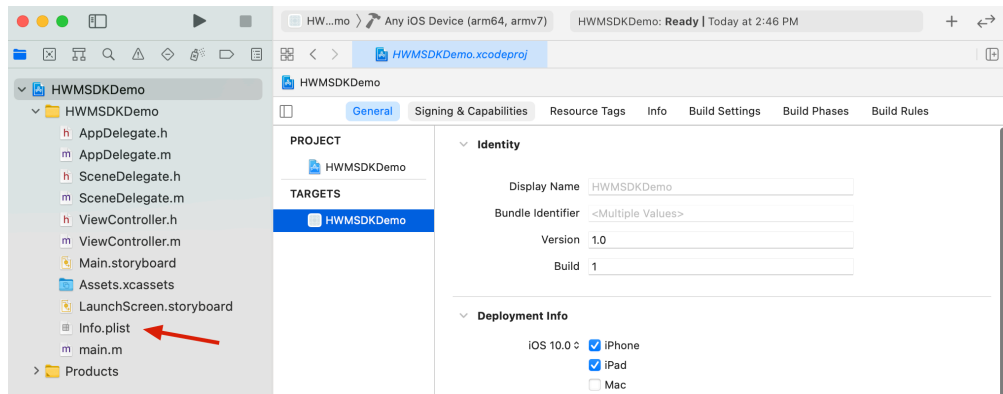


可以逐个复制下面名字，点击加号添加（也可以手动输入）

AddressBook.framework
 AudioToolbox.framework
 AVFoundation.framework
 CFNetwork.framework
 CoreGraphics.framework
 CoreLocation.framework
 CoreMedia.framework
 CoreTelephony.framework
 MediaPlayer.framework
 PushKit.framework
 ReplayKit.framework
 Security.framework
 SystemConfiguration.framework
 VideoToolbox.framework
 WebKit.framework
 libresolv.9.tbd
 libc++.1.tbd
 libz.1.tbd
 libconv.tbd
 libz.tbd
 libc++.tbd

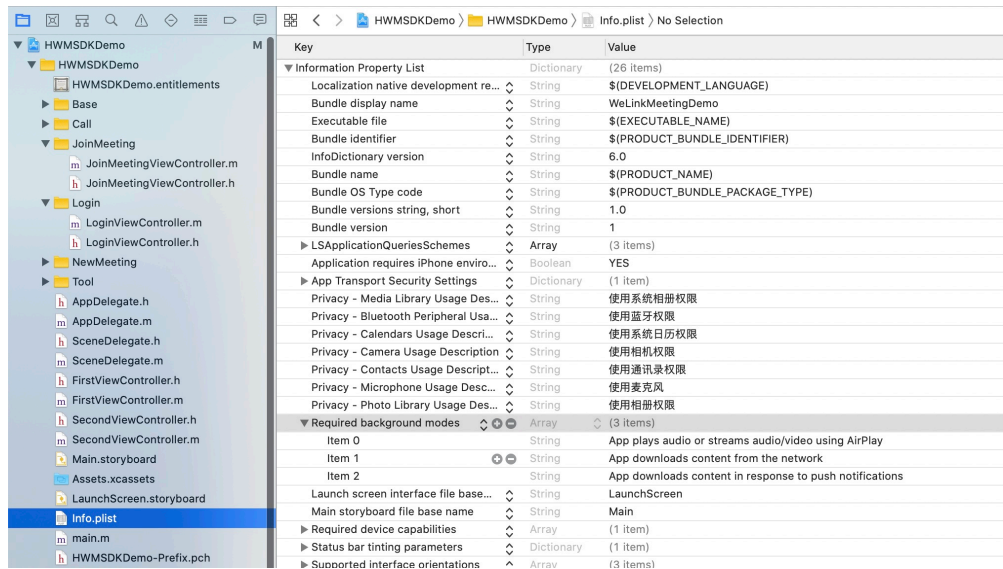
5. Info.plist配置，需要配置权限提示语（语言信息根据自己需要进行设置）和拉起第三方应用等信息，需要配置的字段如下：

图 6-15 Info.plist 位置



- Privacy - Camera Usage Description 使用相机权限提示语
- Privacy - Microphone Usage Description 使用麦克风权限提示语
- Privacy - Contacts Usage Description 使用通讯录权限提示语
- App Transport Security Settings--Allow Arbitrary Loads 需要设置为YES

图 6-16 Info.plist 信息配置完成效果



----结束

6.3.3 屏幕共享集成

说明

屏幕共享：共享自己的手机屏幕的实时界面给其它用户，更加高效的与其它用户沟通。

共享 Bundle Id 相关说明

1. ScreenShareExtension插件bundle Id格式:
主App bundle Id + ScreenShareExtension
2. ScreenShareExtensionSetupUI插件bundle Id格式:

主App bundle Id + ScreenShareExtensionSetupUI

例：
主App bundle Id: com.xx.xx
ScreenShareExtension插件bundle Id: com.xx.xx.ScreenShareExtension
ScreenShareExtensionSetupUI插件bundle Id: com.xx.xx.ScreenShareExtensionSetupUI

iOS端屏幕共享需要添加iOS Extension并实现相关代码，以下步骤用于说明如何集成SDK进行屏幕共享Extension的开发。

📖 说明

由于相关特性对系统能力的依赖，因此需要iOS12及以上版本才能支持屏幕共享功能。详情信息参见“[官方文档](#)”

步骤1 创建新的Extension Target，选择“File->New->Target->Broadcast Upload Extension”，创建Extension Target。

图 6-17 添加选择 Target

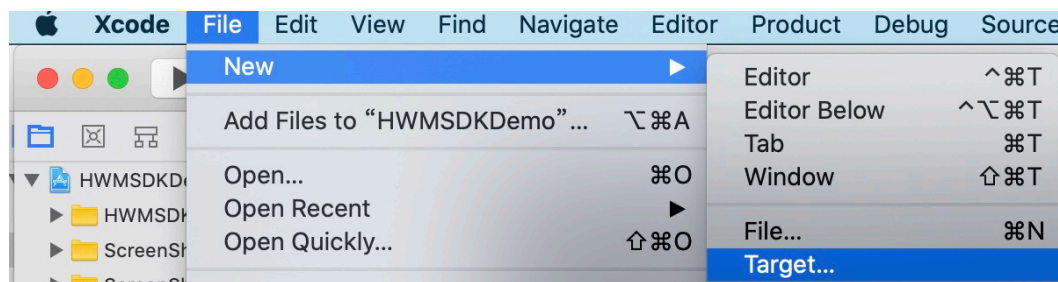


图 6-18 选择 Broadcast Upload Extension，点击“Next”创建

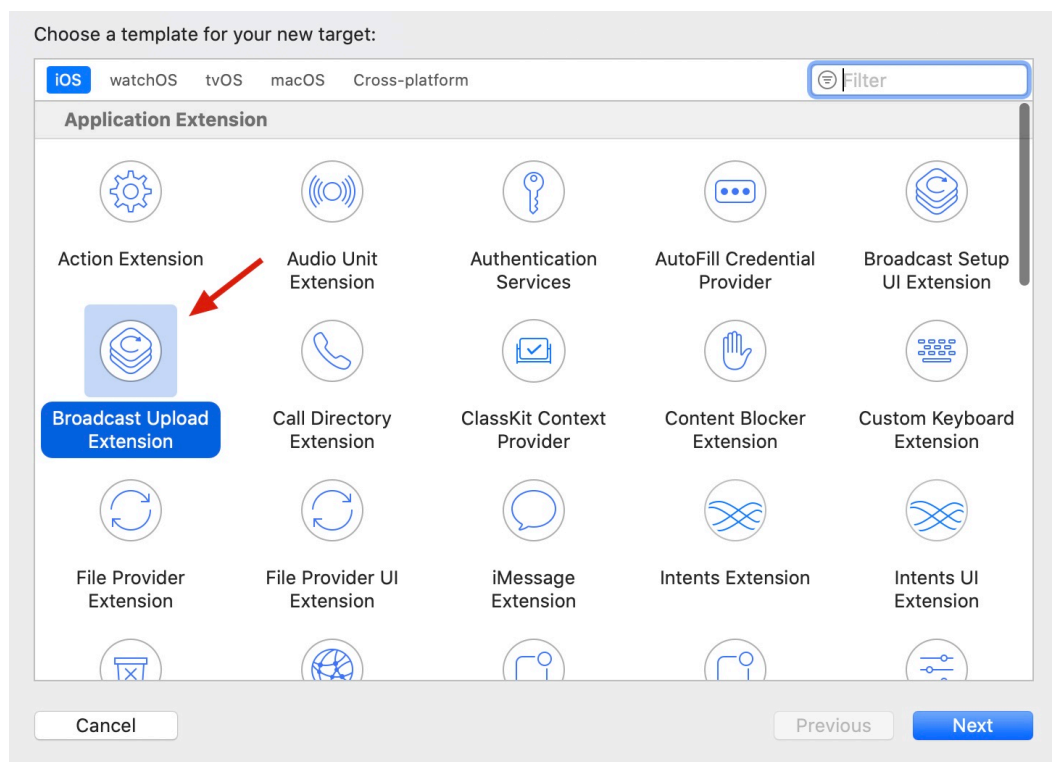
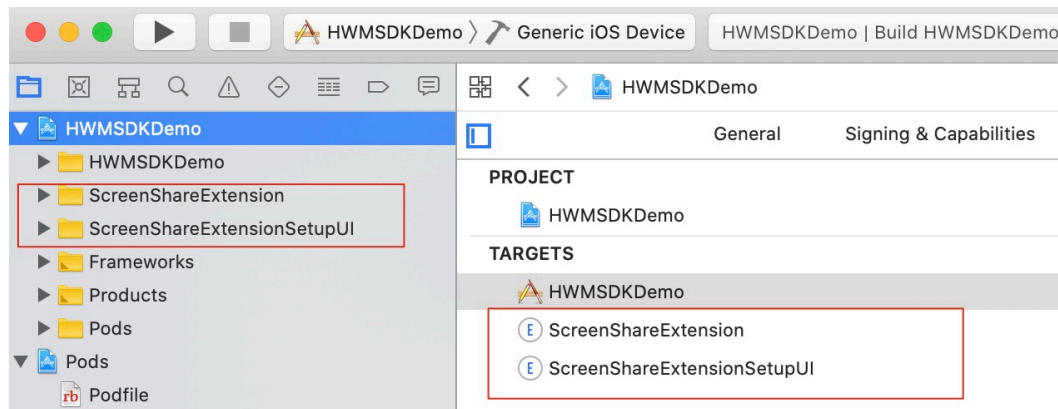


图 6-19 添加 Broadcast Upload Extension 后效果（ScreenShareExtension 是创建时的名称）



步骤2 设置Extension的支持系统版本号12.0

图 6-20 Extension 设置

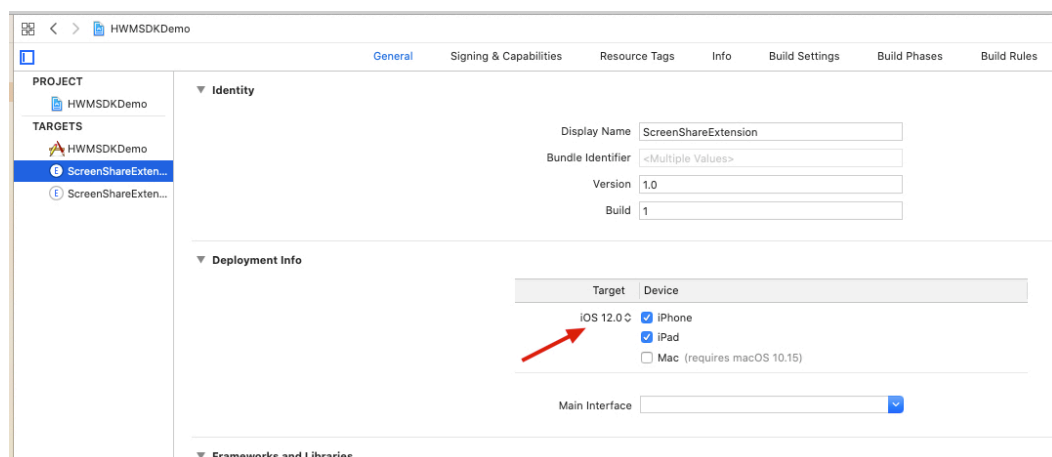
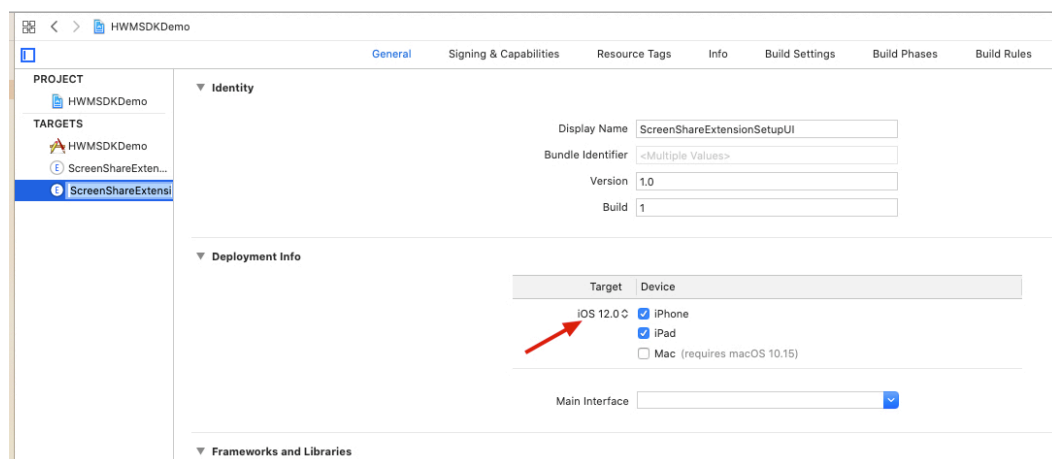


图 6-21 ExtensionUI 设置



步骤3 给主工程和Extension添加App Groups，并设置为在开发准备中申请到的groupId。

图 6-22 默认没有 App Groups

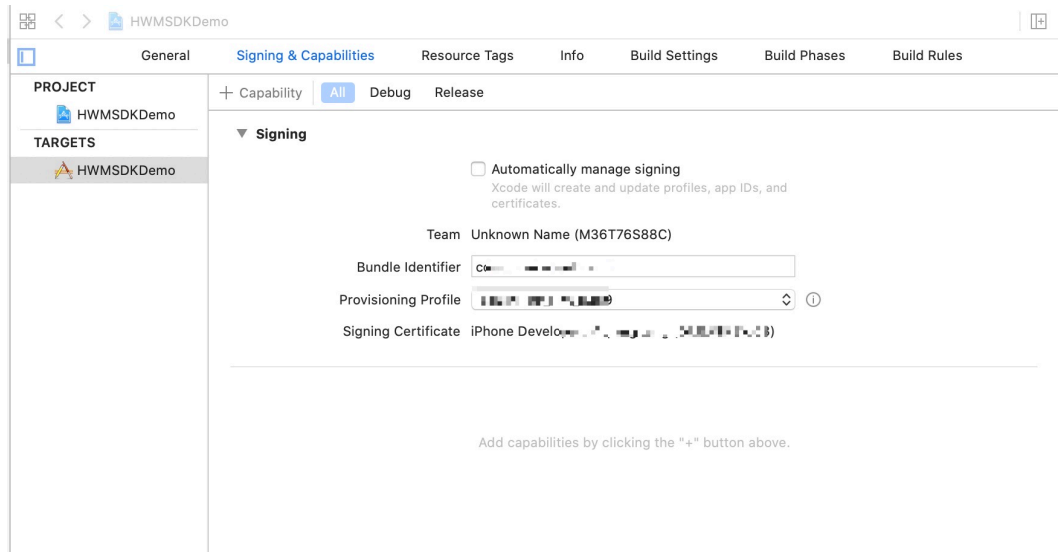


图 6-23 添加 App Groups, 选择 “TARGET->Signing&Capabilities->Capability”

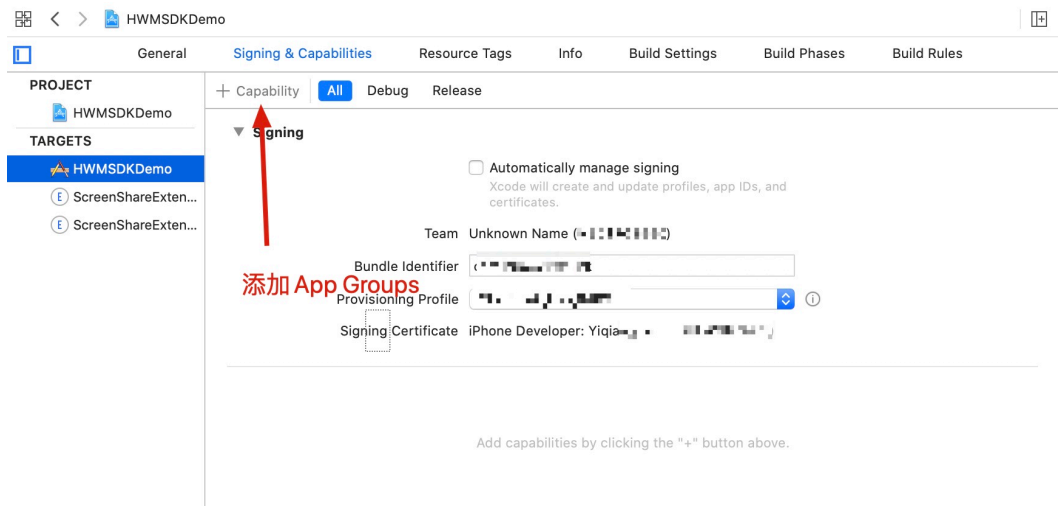
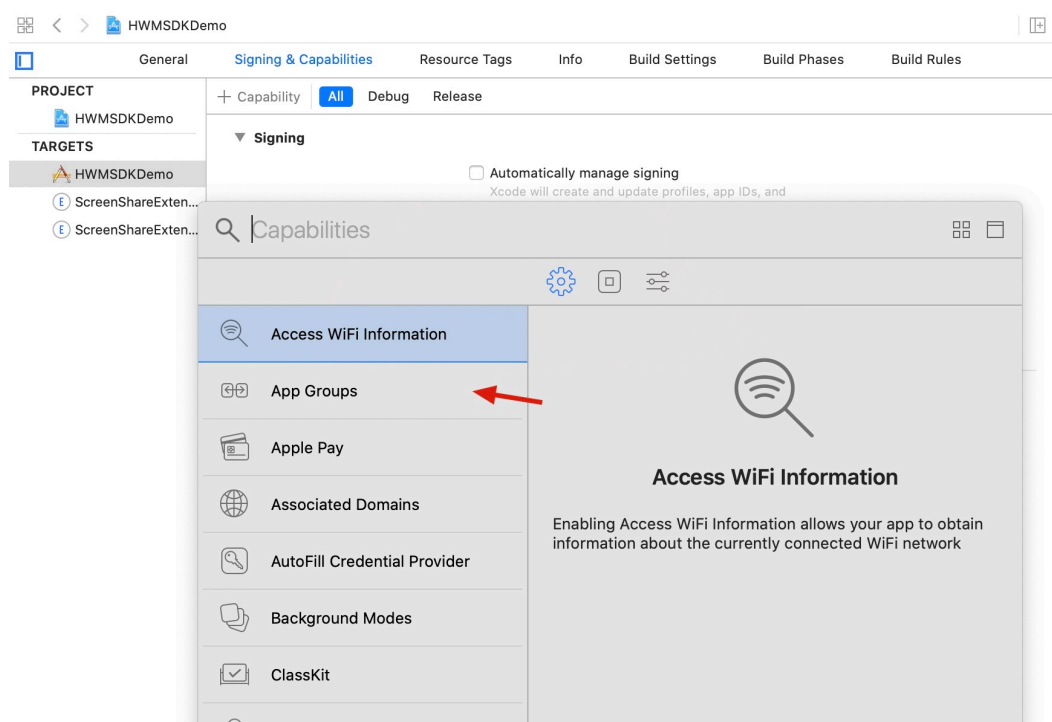
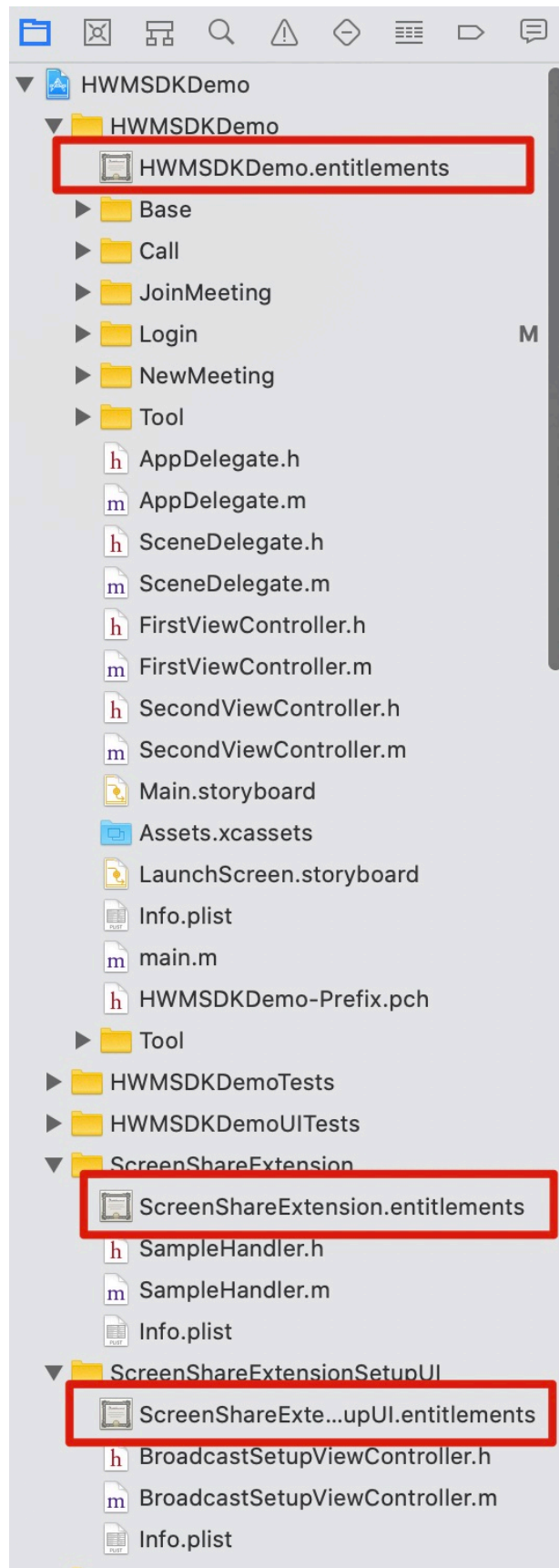


图 6-24 选中 App Groups,



按照上面两个图片，同理分别为ScreenShareExtension和ScreenShareExtensionSetupUI添加App Groups

图 6-25 App Groups 添加完成效果



步骤4 分别给entitlements文件中添加App Groups ID(申请地址 此groupId在初始化的时候也会用到)

图 6-26 应用程序添加 ID 效果



图 6-27 ScreenShareExtension 添加 ID 效果

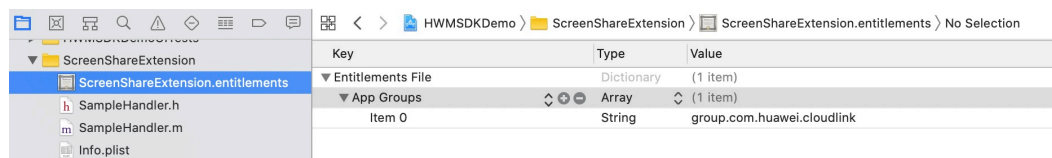
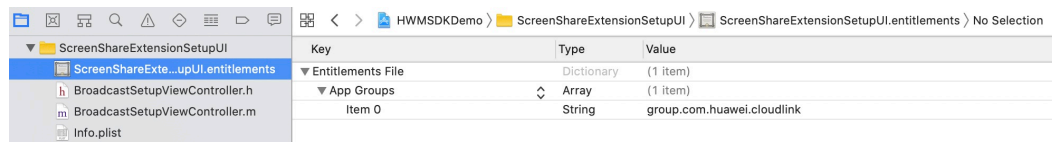


图 6-28 ScreenShareExtensionSetupUI 添加 ID 效果



步骤5 给ScreenShareExtension 添加HWMExtension.framework，添加配置“TARGETS—>ScreenShareExtension—>Build Settings->Other Linker Flags”中添加-lic++。

图 6-29 选择 “ScreenShareExtension->Add Files to”

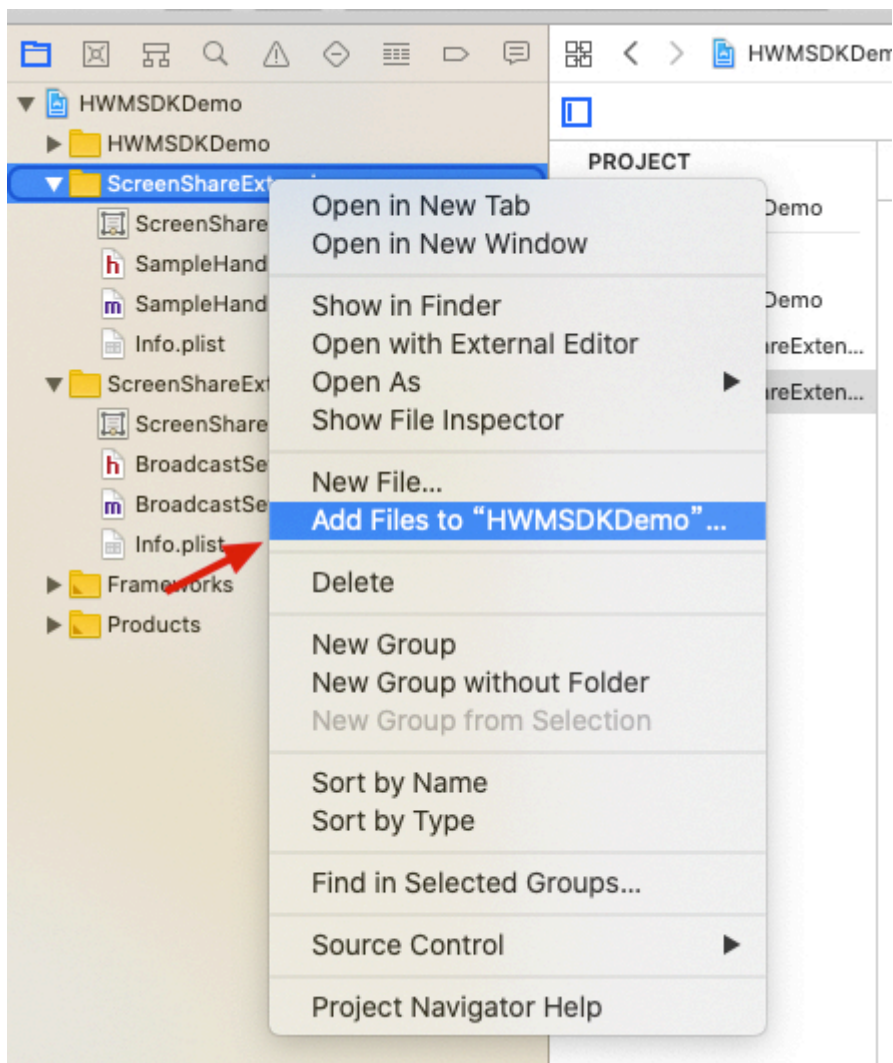


图 6-30 选择 HWMExtension.framework 添加到 Extension

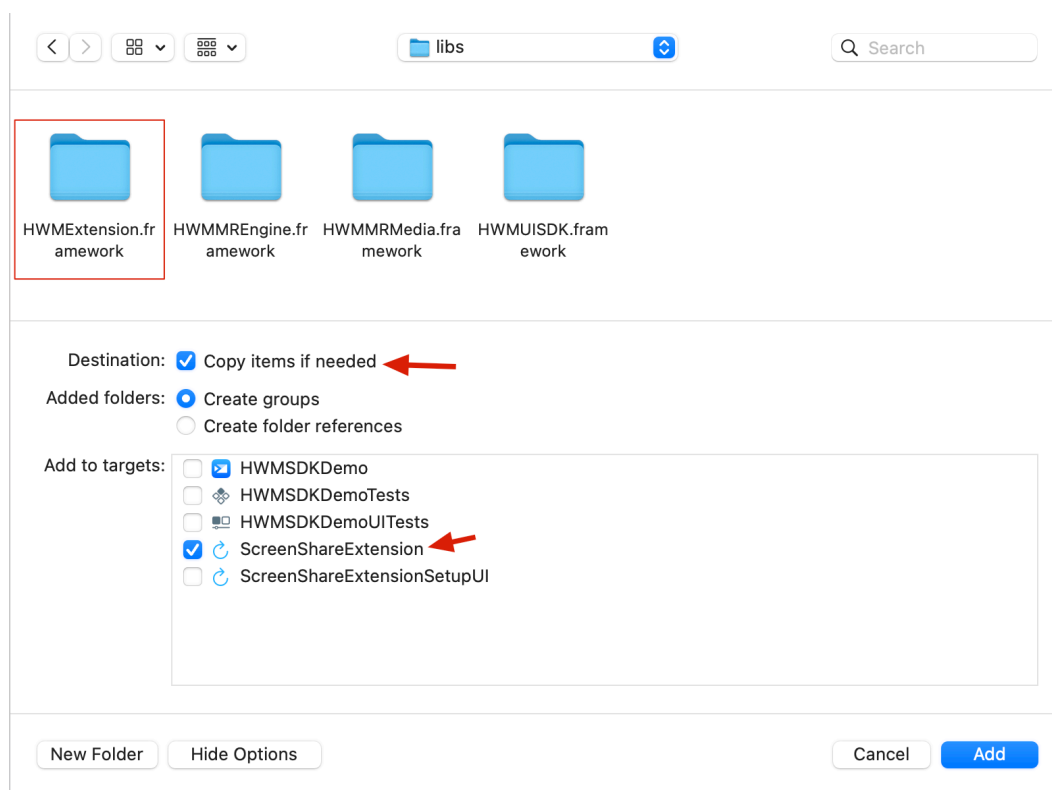
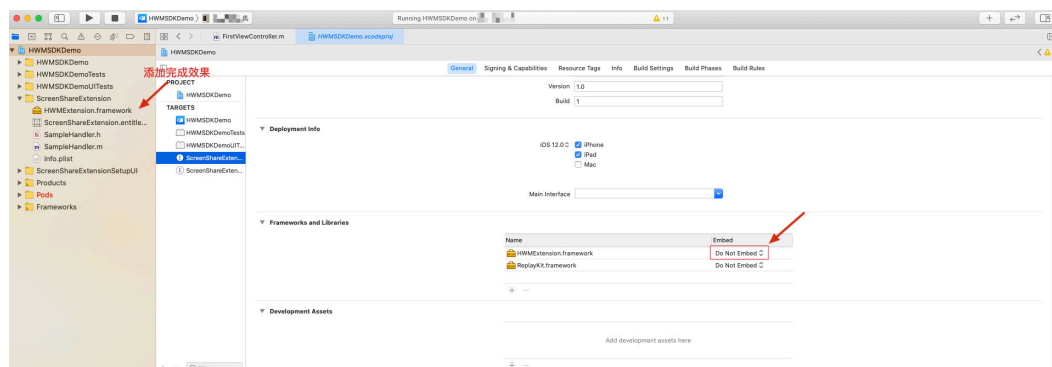


图 6-31 添加完成的效果



步骤6 配置屏幕共享代码，Extensions 的SampleHandler.m文件中添加代码如下

图 6-32 SampleHandler 文件位置



文件内容修改groupId后可以直接使用。

```

#import "SampleHandler.h"
#import <HWMExtension/HWMExtension.h>

@interface SampleHandler()<ScreenShareManagerDelegate>
@property (strong, nonatomic) HWMExtensionDataManager *screenShareManager;
@end

@implementation SampleHandler

- (instancetype)init {
    if (self = [super init]) {
        // [HWMExtensionAppGroup setAppGroup:@"group.com.huawei.xxx"];// 这里替换成自己申请的
        appgroup
        self.screenShareManager = [[HWMExtensionDataManager alloc] init];
        self.screenShareManager.delegate = self;
    }
    return self;
}

- (void)dealloc {
    self.screenShareManager = nil;
}

- (void)broadcastStartedWithSetupInfo:(NSDictionary<NSString *,NSObject *> *)setupInfo {
    // User has requested to start the broadcast. Setup info from the UI extension can be supplied but
    optional.
    [self.screenShareManager broadcastStartedWithSetupInfo:setupInfo];
}

- (void)broadcastPaused {
    // User has requested to pause the broadcast. Samples will stop being delivered.
    [self.screenShareManager broadcastPaused];
}

- (void)broadcastResumed {
    // User has requested to resume the broadcast. Samples delivery will resume.
    [self.screenShareManager broadcastResumed];
}

- (void)broadcastFinished {
    // User has requested to finish the broadcast.
    [self.screenShareManager broadcastFinished];
}

- (void)processSampleBuffer:(CMSampleBufferRef)sampleBuffer withType:
(RPSampleBufferType)sampleBufferType {
    switch (sampleBufferType) {
        case RPSampleBufferTypeVideo:
            // Handle video sample buffer
            [self.screenShareManager processSampleBuffer:sampleBuffer withType:RPSampleBufferTypeVideo];
            break;
        case RPSampleBufferTypeAudioApp:
            // Handle audio sample buffer for app audio
            // 音频共享相关
            if ([self currentIsAudioShare]) {
                [self.screenShareManager processSampleBuffer:sampleBuffer
withType:RPSampleBufferTypeAudioApp];
            }
            break;
        case RPSampleBufferTypeAudioMic:
            // Handle audio sample buffer for mic audio
            break;

        default:
            break;
    }
}

- (void)screenShareManagerFinishBroadcastWithError:(NSError *)error {

```

```

[self.screenShareManager broadcastFinished];
[self finishBroadcastWithError: error];
}
- (BOOL)currentIsAudioShare {
   NSUserDefaults *userDefault = [[NSUserDefaults alloc] initWithSuiteName:[HWMExtensionAppGroup
getAppGroup]];
    NSString *audioShareType = [userDefault objectForKey:kHWMExtensionAudioShareKey];
    if ([audioShareType isEqualToString:kHWMExtensionAudioShareValue]) {
        return YES;
    }
    return NO;
}
@end

```

----结束

6.3.4 业务集成

在调用HWMSDK接口的文件中必须包含以下头文件：

```
#import <HWMUISDK/HWMUISDK.h>
```

SDK初始化：

```

- (void)initSDK {
    HWMOpenSDKConfig *config = [[HWMOpenSDKConfig alloc] init];
    config.appld = @"<#HWMMeetingDemo#>"; // APPID 接入方式时，config.appld 传入申请到的企业APPID；
    非APPID接入方式推荐传入企业名称
    config.appGroupIdentifier = @"<#appGroupIdentifier#>"; // 屏幕共享extension的appGroup。
    BOOL result = [HWMSdk initWithConfig:config];
    if (result) {
        [UIUtil showMessage:@"初始化成功" vc:self];
    }
}

```

接口调用示例

- **登录接口**
[[HWMSdk getOpenApi] login:account password:password callback:^(NSError * _Nullable error, HWMLoginResult * _Nullable result){ }];
- **创会接口**
[[HWMSdk getOpenApi] createConf:param callback:^(NSError * _Nullable error, HWMCreateConfResult * _Nullable result) { }];
- **入会接口**
[[HWMSdk getOpenApi] joinConf:param callback:^(NSError * _Nullable error, id _Nullable result) { }];
- **呼叫接口**
[[HWMSdk getOpenApi] startCall:param callback:^(NSError * _Nullable error, id _Nullable result) { }];
- **预约会议**
[[HWMBizSdk getBizOpenApi] bookConf:param callback:^(NSError * _Nullable error, id _Nullable result) { }];
- **编辑会议**
[[HWMBizSdk getBizOpenApi] editConf:param callback:^(NSError * _Nullable error, id _Nullable result) { }];
- **退出登录接口**
[[HWMSdk getOpenApi] logout:^(NSError * _Nullable error, id _Nullable result) { }];

完成上述接口调用代码后即完成集成SDK，更详细信息请参考HWMSDKDemo样例代码。

6.4 典型场景

本小节描述SDK在场景使用时的接口调用顺序和接口调用示例。

6.4.1 典型场景接口调用概览

3 类接口描述

iOS SDK的接口分为**同步Api**、**异步Api**、**全局回调**。

同步Api：App主动调用SDK的接口，用于完成某项任务。比如初始化接口 `initWithConfig`。

异步Api：调用SDK接口的时候都会以callback回调方式请求结果。

全局回调：由华为云会议服务端或SDK内部触发的事件通知，SDK以消息通知接口的形式通知给App。比如会议来电通知 `onConfStatusChanged`。

典型场景接口的调用顺序及生命周期

图 6-33 典型场景接口调用时序图



📖 说明

- 1、使用SDK功能前需要初始化。
- 2、在不登录的情况下也能通过会议ID和密码加入已经创建好的会议，其它使用场景需要登录。
- 3、可选通知根据需求场景决定是否实现。

6.4.2 场景 1：初始化

描述

需要先初始化SDK，才能调用SDK的功能接口，该接口只需调用一次。

业务流程

SDK初始化时，先调用initWithConfig接口，根据返回值确定接口是否调用成功。

步骤1 接口调用

1. 构建HWMOpenSDKConfig模型参数。
2. 调用initWithConfig接口，完成配置初始化，第1步中的数据作为参数。

步骤2 处理返回值

处理返回值，YES代表初始化成功、NO代表初始化失败。

----结束

示例代码

```
/// 初始化
HWMOpenSDKConfig *config = [[HWMOpenSDKConfig alloc] init];
config.appId = @"<#fdb8e4699586458bbd10c834872dcc62#>"; // 传入申请到的企业AppId
config.appGroupIdentifier = @"<#appGroupIdentifier#>"; // 必须，屏幕共享 extension的appGroup。用于屏幕共享时，共享进程和主进程通信，详情可参考苹果官方Extension相关文档
BOOL result = [HWMsdk initWithConfig:config];
if (result) {
    NSLog(@"初始化成功");
}
```

📖 说明

典型场景和接口参考中的示例代码仅作为示例用的伪代码，不能直接使用。

6.4.3 场景 2：登录

6.4.3.1 帐号密码登录

描述

该接口用于帐号登录。创建会议等接口不支持在未登录状态下调用，若要使用完整的会议功能，必须先登录。

📖 说明

在未登录状态下，只能加入已经创建的会议。

业务流程

使用SDK登录时，先调用login接口，然后处理回调callback。

步骤1 接口调用

1. 获得登录使用的帐号和密码
2. 调用login进行登录，第1步中的数据作为参数。

步骤2 处理回调

处理回调callback。

----结束

示例代码

```

/// 登录处理
- (void)login {
    NSString *account = @"139571854984";
    NSString *password = @"Change_Me";
    [[HWM SDK getOpenApi] login:account password:password callback:^(NSError * _Nullable error,
    HWMLoginResult * _Nullable result) {
        if (error == nil) {
            NSLog(@"登录成功");
        }else{
            NSLog(@"登录失败");
        }
    }
    }
}

```

6.4.3.2 App ID 登录

描述

该接口用于通过App ID登录

📖 说明

在未登录状态下，只能加入已经创建的会议。

业务流程

使用SDK登录时，先调用login接口，然后处理回调callback。

步骤1 接口调用

1. 从接入方服务器获得登录使用的帐号、过期时间、随机数、用户名、加密签名
2. 调用login进行登录，第1步中的数据作为参数。

步骤2 处理回调

处理回调callback。

----结束

示例代码

```

/// AppId 登录接口
- (void)login {
    HWMAppIDLoginParam *appidInfo = [[HWMAppIDLoginParam alloc] init];
}

```

```

appidInfo.thirdUserId = @"test";
appidInfo.expireTime = 1599618620;
appidInfo.nonce = @"nonce";
appidInfo.userName = @"测试帐号";
appidInfo.signature = @"signature";
[[HWMSSdk getOpenApi] login:appidInfo callback:^(NSError * _Nullable error, HWMLoginResult *
_Nullable result) {
    if (error == nil) {
        NSLog(@"登录成功");
    }else{
        NSLog(@"登录失败");
    }
}
}];
}

```

6.4.4 场景 3：创建会议

描述

登录后，可以调用创建会议接口创建立即会议。创建会议时可以携带与会人信息，也可以不携带。

说明

在登录状态下，才能创建会议。

业务流程

使用SDK创建立即会议时，先调用createConf接口，然后处理回调callback。

步骤1 接口调用

1. 组装模型HWMCreateConfParam。
2. 组装模型数组HWMCreateConfMember（与会者可选）。
3. 调用createConf开始创建，第1步中的数据作为参数。

步骤2 处理回调

处理回调callback。

----结束

示例代码

```

/// 创建会议
- (void)creatMeeting {
    HWMCreateConfParam *param = [[HWMCreateConfParam alloc] init];
    param.subject = @"会议标题";
    param.confType = HWMConfTypeVideoData;
    param.needPassword = YES;// 会议是否需要设置密码，默认不需要
    param.isCameraOn = YES;// 是否打开摄像头，默认关闭
    param.isMicOn = YES;// 是否打开麦克风，默认打开
    param.isRecordOn = NO;// 是否打开会议录制，默认关闭
    param.joinConfRestrictionType = HWMJoinConfRestrictionAll;// 允许入会范围
    // 与会者列表
    if (self.selectedMemebrs) {
        __block NSMutableArray *members = [[NSMutableArray alloc] init];
        [self.selectedMemebrs enumerateObjectsUsingBlock:^(HWMContactSelectedModel * _Nonnull obj,
        NSUIInteger idx, BOOL * _Nonnull stop) {
            HWMCreateConfMember *member = [[HWMCreateConfMember alloc] init];
            member.accountId = obj.accountId;
            member.number = obj.number;

```

```

        member.name = obj.name;
        member.thirdUserId = obj.thirdUserId;
        [members addObject:member];
    }];
    param.members = members;
}
[[[HWM SDK getOpenApi] createConf:param callback:^(NSError * _Nullable error, HWMCreateConfResult *
_Nullable result) {
    [self hideLoading];
    if (error) {
        NSLog(@"创会失败%@", error.localizedDescription);
    }else{
        NSLog(@"创会成功");
    }
}]];
}
}

```

📖 说明

与会者列表可以携带与会者创建会议，创建会议成功后服务端会呼叫携带的与会人入会，不需要可以不传。

6.4.5 场景 4：添加与会者

描述

在会议中，可以调用添加与会者的接口把与会人加入到会议中。

业务流程

使用SDK添加与会人时，先打开联系人列表，选择要添加联系人，之后把选中的联系人添加到会议中。也可以自定义加人界面，参考[联系人UI定制](#)。

📖 说明

- 1.AppId 登录场景下，建议自定义添加与会者。
- 2.如果不自定义添加与会者界面，项目中无需添加示例代码。

示例代码

```

/// 定制打开联系人选择页面
- (void)openContactSelectPage:(NSArray<HWMConfContactModel *> *)param
    scene:(HWMSelectedContactScene)scene
    completionHandler:(void(^)(NSArray<HWMContactSelectedModel *> *result, NSError
*error))handler {
    NSLog(@"enter open contact select page");
    ContactSelectViewController * contactSelectVc = [[ContactSelectViewController alloc]
initWithResultHandler:handler];
    [self pushViewControllerWithVc:contactSelectVc];
}

```

6.4.6 场景 5：加入会议

描述

在登录后或者登录前，都可以使用该接口通过会议ID和密码加入会议。

业务流程

使用SDK加入已经存在的会议时，先调用joinConf接口，然后处理回调callback。

步骤1 接口调用

1. 组装模型
 - a. 会议id加入会议使用HWMJoinConfParam。
 - b. 会议random加入会议使用HWMRandomConfParam。
2. 调用joinConf开始加入会议，第1步中的数据作为参数。

步骤2 处理回调

处理回调callback。

----结束

示例代码

```

/// 加入会议
- (void)joinMeeting {
    HWMJoinConfParam *param = [[HWMJoinConfParam alloc] init];
    param.confId = @"989156631";
    param.password = @"807766";
    param.nickname = @"杭州会场";
    param.isCameraOn = NO;
    param.isMicOn = YES;
    [[HWMSdk getOpenApi] joinConf:param callback:^(NSError * _Nullable error, id _Nullable result) {
        if (error) {
            NSLog(@"入会失败%@", error.localizedDescription);
        } else {
            NSLog(@"入会成功");
        }
    }];
}
    
```

📖 说明

在未登录状态下调用该接口时，nickname参数必填。

6.4.7 场景 6：发起呼叫

描述

登录后，可以调用接口发起呼叫。

业务流程

使用SDK发起呼叫时，先调用startCall接口，然后处理回调callback。

步骤1 接口调用

1. 调用startCall发起呼叫。

步骤2 处理回调

处理回调callback。

----结束

示例代码

```

/// 发起呼叫
- (void)startCall {
    /**
    
```

```

* AppId 登录场景：account/number/thirdUserId 三选一，优先级：number>account>thirdUserId
* 非AppId 登录场景：account/number 二选一，优先级：number>account
*/
HWMStartCallParam *param = [[HWMStartCallParam alloc] init];
param.isVideo = YES;
param.account = @"";
param.number = @"+991112988657623";
param.thirdUserId = @"";
[[HWMSdk getOpenApi] startCall:param callback:^(NSError * _Nullable error, id _Nullable result) {
    if (error) {
        NSLog(@"发起呼叫失败%@", error.localizedDescription);
    }else{
        NSLog(@"发起呼叫成功");
    }
}
}
}

```

6.4.8 场景 7：预约会议

描述

登录后，可以调用接口预约会议。

业务流程

使用SDK预约会议时，先调用bookConf接口，然后处理回调callback。

步骤1 接口调用

1. 调用bookConf预约会议。

步骤2 处理回调

处理回调callback。

----结束

示例代码

```

/// 预约会议
- (void)bookConf {
    HWMOrderConfParam * param = [[HWMOrderConfParam alloc] init];
    param.confSubject = @"会议名称";
    param.startTime = 1598759580;
    param.duration = 15;
    param.confType = HWMConfTypeVideo;
    param.isNeedConfPwd = YES;
    param.isRecordOn = NO;
    param.timeZone = 56;
    param.vmrId = @"";
    param.callInRestrictionType = HWMCallRestrictionAll;
    [[HWMBizSdk getBizOpenApi] bookConf:param callback:^(NSError * _Nullable error, id _Nullable result) {
        if (error) {
            NSLog(@"预约会议失败:%@", error.localizedDescription);
        } else {
            NSLog(@"预约会议成功");
        }
    }
}
}
}

```

6.4.9 场景 8：编辑会议

描述

登录后，调用[会议详情](#)接口（也可以使用服务器的rest接口）获取会议详情后调用接口编辑会议。

业务流程

使用SDK编辑会议时，先调用editConf接口，然后处理回调callback。

步骤1 接口调用

1. 调用editConf编辑会议。

步骤2 处理回调

处理回调callback。

---结束

示例代码

```
/// 编辑会议
- (void)editConf {
    HWMOrderConfParam * param = [[HWMOrderConfParam alloc] init];
    param.confSubject = @"会议名称";
    param.startTime = 1598759580;
    param.duration = 15;
    param.confType = HWMConfTypeVideo;
    param.isNeedConfPwd = YES;
    param.isRecordOn = NO;
    param.timeZone = 56;
    param.vmrId = @""; // vmr会议需要传入vmrId
    param.callInRestrictionType = HWMCallRestrictionAll;
    param.confId = @"986927771";
    [[HWMBizSdk getBizOpenApi] editConf:param callback:^(NSError * _Nullable error, id _Nullable result) {
        if (error) {
            NSLog(@"编辑会议失败:%@", error.localizedDescription);
        } else {
            NSLog(@"编辑会议成功");
        }
    }];
}
```

6.4.10 场景 9：退出登录

描述

当需要退出当前会议帐号登录时，可以调用该接口完成已登录会议帐号退出。

业务流程

使用SDK退出登录时，先调用logout接口，然后处理回调callback。

步骤1 接口调用

1. 调用logout进行退出登录。

步骤2 处理回调

处理回调callback。

----结束

示例代码

```

/// 退出登录
- (void)logout {
    [[HWM SDK getOpenApi] logout:^(NSError * _Nullable error, id _Nullable result) {
        if (error == nil) {
            NSLog(@"退出成功");
        } else {
            NSLog(@"退出失败");
        }
    }];
}

```

6.5 接口参考

6.5.1 基础配置

6.5.1.1 初始化

📖 说明

初始化接口有主线程初始化、子线程初始化两种，必须先调用任意一个初始化接口后再调用其它接口。

initWithConfig

接口描述

用于启动并且初始化SDK。

注意事项

1. 该接口用于启动并初始化SDK，在调用其他接口之前必须先调用初始化接口。
2. 此初始化方法必须在主线程调用。
3. 返回值只代表接口是否调用成功。

方法定义

```
+ (BOOL) initWithConfig:(HWMOpenSDKConfig *)config;
```

参数描述

表 6-2 参数说明

参数	是否必须	类型	描述
config	是	HWMOpenSDKConfig	初始化信息集合

表 6-3 初始化信息 HWMSdkOpenSDKConfig

参数	是否必须	类型	描述
appId	是	NSString *	App ID。App ID的申请, 请参考《开发指南》的“ App ID 的申请 ”章节。
appGroupIdentifier	否	NSString *	屏幕共享 extension的 appGroup, 不传屏幕共享功能不能使用。用于屏幕共享时, 共享进程和主进程通信, 详情可参考苹果官方 Extension相关文档
siteType	否	HWMSdkSiteType	登录站点类型, 默认为中国站点 (HWMSdkSiteTypeChina)
serverAddress	否	NSString *	服务器地址配置
serverPort	否	NSInteger	服务器端口配置
hideExternalLabel	否	BOOL	与会者列表、转移主持人、聊天界面是否隐藏外部标签, 默认显示
logKeepDays	否	NSInteger	配置项-日志保存天数, 输入有效范围为3-30, 默认值0。0为不按天数删除日志。 注意: 输入1和2会转换为3, 大于30的值会被转换为30。
disableIncomingLocal Notification	否	BOOL	来电时禁止使用本地推送

说明

App ID的申请请参考《开发指南》的“App ID鉴权介绍”的“[App ID的申请](#)”部分。

表 6-4 枚举 HWMSdkSiteType 说明

枚举值	描述
HWMSdkSiteTypeChina	中国站点
HWMSdkSiteTypeAP	亚太站点

返回值

表 6-5 返回值

类型	描述
BOOL	初始化成功返回YES，失败返回NO。

示例代码

```

/// 初始化
HWMSDKOpenSDKConfig *config = [[HWMSDKOpenSDKConfig alloc] init];
config.appId = @"fdb8e4699586458bbd10c834872dcc62"; // 传入申请到的企业AppId
config.appGroupId = @"appGroupId"; // 必须，屏幕共享 extension 的 appGroup。用于屏幕共享时，共享进程和主进程通信，详情可参考苹果官方 Extension 相关文档
BOOL result = [HWMSdk initWithConfig:config];
if (result) {
    NSLog(@"初始化成功");
}
    
```

说明

- 典型场景和接口参考中的示例代码仅作为示例用的伪代码，不能直接使用。

initWithConfig:callback:

接口描述

用于启动并且子线程初始化SDK。

注意事项

- 该接口用于启动并初始化SDK，在调用其他接口之前必须先调用初始化接口。
- 返回值只代表接口是否调用成功。

方法定义

```

+ (void) initWithConfig:(HWMSDKOpenSDKConfig *)config callback:(__nonnull
HWMSDKInitCompleteHandler)callback;
    
```

参数描述

参见[初始化参数说明](#)

返回值

无

示例代码

```

/// 初始化
HWMSDKOpenSDKConfig *config = [[HWMSDKOpenSDKConfig alloc] init];
config.appId = @"fdb8e4699586458bbd10c834872dcc62"; // 传入申请到的企业AppId
config.appGroupId = @"appGroupId"; // 必须，屏幕共享 extension 的 appGroup。用于屏幕共享时，共享进程和主进程通信，详情可参考苹果官方 Extension 相关文档
[HWMSdk initWithConfig:config callback:^(HWMSDKERR errCode, id _Nullable result) {
    if (errCode == HWMSDKSderrSuccess) {
        NSLog(@"初始化成功");
    } else {
        NSLog(@"初始化失败");
    }
}];
    
```

说明

- 典型场景和接口参考中的示例代码仅作为示例用的伪代码，不能直接使用。

6.5.1.2 设置语言

setLanguage

接口描述

该接口设置SDK国际化语言。默认是中文。

注意事项

无

方法定义

```
+ (void)setLanguage:(HWMSdkLanguageType)language;
```

参数描述

表 6-6 参数说明

参数	是否必须	类型	描述
language	是	HWMSdkLanguageType	语言类型枚举。

表 6-7 枚举 HWMSdkLanguageType 说明

枚举值	描述
HWMSdkLanguageTypeZH	中文。
HWMSdkLanguageTypeEN	英文。
HWMSdkLanguageTypeCustomized	自定义语言

返回值

无

示例代码

```
/// 修改语言
[HWMSdk setLanguage:HWMSdkLanguageType];
```

setLanguage:languageCode:

接口描述

该接口用于自定义设置SDK国际化语言,默认是中文。

注意事项

实现自定义语言,要求在 en.lproj 同级目录下创建对应的国际化文件。比如新增阿拉伯语则可以创建 ar.lproj/Localizable.strings。通过该接口设置的语言应与系统当前语言保持一致

方法定义

```
+ (void)setLanguage:(HWMSdkLanguageType)language languageCode:(NSString *)languageCode;
```

参数描述

表 6-8 参数说明

参数	是否必须	类型	描述
language	是	HWMSdkLanguageType	语言类型枚举。
languageCode	否	NSString *	自定义时对应的语言代码,若是中文/英文,该参数可以传空字符串

返回值

无

示例代码

```
/// 修改语言
[HWMSdk setLanguage:HWMSdkLanguageTypeCustomized languageCode:@"ar"];
```

6.5.1.3 设置日志路径

setLogPath

接口描述

该接口用于设置华为云会议打印日志路径。

注意事项

无

方法定义

```
+ (void)setLogPath:(NSString *)logPath;
```

参数描述

表 6-9 参数说明

参数	是否必须	类型	描述
logPath	是	NSString *	日志路径。

返回值

无

示例代码

```
/// 路径为默认路径
NSString * logPath= [[NSSearchPathForDirectoriesInDomains(NSLibraryDirectory, NSUserDomainMask, YES)
firstObject] stringByAppendingPathComponent:@"log"];
[HWMSdk setLogPath:logPath];
```

6.5.1.4 设置来电通知方式

changeInComingNoticeWithRing:vibrate:

接口描述

该接口用修改来电（呼叫/会议）通知方式。（默认响铃不振动）

注意事项

1. 登录后调用接口
2. 打开振动时需要设置打开iOS系统的振动（设置—>声音与触感—>响铃模式振动和静音模式振动）
3. 返回YES说明接口调用成功，未初始化或者未登录返回NO

方法定义

```
-(BOOL)changeInComingNoticeWithRing:(BOOL)ring vibrate:(BOOL)vibrate;
```

参数描述

表 6-10 参数说明

参数	是否必须	类型	描述
ring	是	BOOL	是否开启铃声
vibrate	是	BOOL	是否开启振动

返回值

无

示例代码

```
BOOL result = [[HWMSdk getOpenApi] changeInComingNoticeWithRing:YES vibrate:NO];
[UIUtil showMessage:result ? @"设置成功" : @"设置失败"];
```

6.5.1.5 自定义会议接听时摄像头/麦克风开关

isOpenMicrophoneIncoming

接口描述

该接口用于自定义被邀入会时麦克风的开关状态。

注意事项

1. 如果设置除主持人外的与会方入会后自动静音则该自定义无效果；

方法定义

```
- (BOOL)isOpenMicrophoneIncoming;
```

参数描述

无

示例代码

```
/// 被邀入会时是否打开麦克风
- (BOOL)isOpenMicrophoneIncoming {
    return NO;
}
```

isOpenCameraIncoming

接口描述

该接口用于自定义被邀入会时摄像头的开关状态。

注意事项

无

方法定义

```
- (BOOL)isOpenCameraIncoming;
```

参数描述

无

示例代码

```
/// 被邀入会时是否打开摄像头
- (BOOL)isOpenCameraIncoming {
    return NO;
}
```

6.5.1.6 开启功能

enableFeature:isEnabled:

接口描述

用于开启特性支持。

注意事项

1. 默认是开启分组会议，若不需要分组会议功能，再调用此接口。
2. 先创建初始化对象（HWMOpenSDKConfig），再调用该接口，最后初始化。

方法定义

```
- (void)enableFeature:(HWMSupportFeatureType)feature isEnabled:(BOOL)enable;
```

参数描述

表 6-11 枚举 HWMEnableFeatureType 说明

值	描述
HWMFeatureTypeBreakoutConf	分组会议能力
HWMFeatureTypeRecoedAlert	是否显示录制安全提示窗，默认不显示

返回值

无

示例代码

```
HWMOpenSDKConfig *config = [[HWMOpenSDKConfig alloc] init];
[config enableFeature:HWMFeatureTypeBreakoutConf isEnabled:YES];
```

6.5.1.7 网络模式设置

setNetworkModeWithNetworkInfo:

接口描述

如华为云会议的媒体节点下沉部署到企业，并且管理节点也是通过企业内的代理访问，需要将网络模式设置成“企业内外接入”。

说明

企业内的App无法直接访问公网（即媒体节点下沉和管理节点代理下沉场景）的情况下才需要调用该接口。

注意事项

1. 该接口非必需调用，若不设置，则使用默认模式。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
-(HWMSDKERR)setNetworkModeWithNetworkInfo:(HWMSDKNetWorkModelInfoModel *)networkInfo;
```

参数描述

表 6-12 HWMSDKNetWorkModelInfoModel 说明

参数	是否必须	类型	描述
accessType	是	HWMNetworkAccessType	网络类型

表 6-13 枚举 HWMNetworkAccessType 说明

值	描述
HWMSDKAccessTypeAuto	默认值
HWMSDKAccessTypeInner	内网接入
HWMSDKAccessTypeOuter	外网接入

返回值

无

示例代码

```
HWMSDKNetWorkModelInfoModel *networkInfo = [[HWMSDKNetWorkModelInfoModel alloc] init];
networkInfo.accessType = HWMSDKAccessTypeAuto;
HWMSDKERR ret = [[HWMSDKNetworkService sharedInstance]
setNetworkModeWithNetworkInfo:networkInfo];
```

6.5.1.8 禁用提示音

disableVoicePrompts:

接口描述

禁用会议提示音。

注意事项

1. 该接口非必需调用，若不设置，则使用默认模式。
2. 返回值只代表接口是否调用成功。

方法定义

```
-(HWMSDKERR)disableVoicePrompts:(HWMSDKVoicePromptsModel *)voicePrompts;
```

参数描述

表 6-14 HWMSDKVoicePromptsModel 说明

参数	是否必须	类型	描述
disableMuteProm pt	是	BOOL	主持人静音与会者提示音

返回值

接口是否调用成功

示例代码

```
HWMSDKVoicePromptsModel * model = [[HWMSDKVoicePromptsModel alloc] init];
model.disableMutePrompt = YES;
[[HWMBizSdk getBizOpenApi] disableVoicePrompts:model];
```


6.5.2 帐号操作

6.5.2.1 登录

登录分为帐号密码鉴权登录和App ID鉴权登录。

login:password:

接口描述

该接口用于通过帐号密码鉴权登录。

注意事项

1. 创建会议，被邀入会等功能只有在登录后才能使用，若不登录，将无法体验完整会议功能。
2. 回调error表示错误信息，nil表示成功。

方法定义

```
-(void)login:(NSString * _Nonnull)account password:(NSString * _Nonnull)password callback:( _Nonnull HWMSDKLoginCompleteHandler)callback;
```

参数描述

表 6-15 参数说明

参数	是否必须	类型	描述
account	是	NSString *	登录帐号
password	是	NSString *	登录密码
callback	否	HWMSDKLoginCompleteHandler	参数Error表示错误信息，nil表示成功。

返回值

表 6-16 登录返回 HWMLoginResult

参数	类型	描述
userId	NSString *	账号uuid
account	NSString *	登录账号
thirdAccount	NSString *	appid登录的三方账号

示例代码

```
/// 登录处理
-(void)login {
```

```
NSString *account = @"139571854984";
NSString *password = @"Change_Me";
[[HWMSdk getOpenApi] login:account password:password callback:^(NSError * _Nullable error,
HWMLoginResult * _Nullable result) {
    if (error == nil) {
        NSLog(@"登录成功");
    }else{
        NSLog(@"登录失败");
    }
}
}];
}
```

login:

接口描述

该接口用于通过App ID鉴权登录。

注意事项

1. 创建会议，被邀入会等功能只有在登录后才能使用，若不登录，将无法体验完整会议功能。
2. 回调error表示错误信息，nil表示成功。

方法定义

```
-(void)login:(HWMAAppIDLoginParam *)appidInfo callback:(HWMSDKLoginCompleteHandler)callback;
```

参数描述

表 6-17 HWMAAppIDLoginParam 参数说明

参数	是否必须	类型	描述
signature	是	NSString *	从第三方服务端获取的鉴权签名信息
thirdUserId	是	NSString *	第三方用户的帐号
expireTime	是	NSInteger	signature 过期时间，单位为秒的时间戳，为0时表示不会过期
nonce	是	NSString *	随机数
userName	否	NSString *	用户名
userEmail	否	NSString *	邮箱
userPhone	否	NSString *	手机号
corpId	否	NSString *	设置该用户所在的企业ID。仅在SP模式下需要填写，单企业模式不要填写，否则会鉴权失败。
deptCode	否	NSString *	部门编码

返回值

参考：表2

示例代码

```

/// AppId 登录接口
- (void)login {
    HWMAppIDLoginParam *appidInfo = [[HWMAppIDLoginParam alloc] init];
    appidInfo.thirdUserId = @"test";
    appidInfo.expireTime = 1599618620;
    appidInfo.nonce = @"0vjeeKUvztUu6N20lqfooegDVGi5Kf30";
    appidInfo.userName = @"测试帐号";
    appidInfo.signature = @"fac06749023bfc887b3b5fa030cc3dbac2c95a3fed85664bae55ba334f9c";
    [[HWMSDK getOpenApi] login:appidInfo callback:^(NSError * _Nullable error, HWMLoginResult *
_Nullable result) {
        if (error == nil) {
            NSLog(@"登录成功");
        }else{
            NSLog(@"登录失败");
        }
    }
    }];
}
    
```

loginBySSO:

接口描述

该接口用于通过SSO鉴权登录。

注意事项

1. 创建会议，被邀入会等功能只有在登录后才能使用，若不登录，将无法体验完整会议功能。
2. 回调error表示错误信息，nil表示成功。

方法定义

```

- (void)loginBySSO:(HWMSSOLoginParam)param callback:(Nonnull
HWMSDKLoginCompleteHandler)callback;
    
```

参数描述

表 6-18 HWMSSOLoginParam 参数说明

参数	是否必须	类型	描述
domain	是	NSString *	企业域名
userTicket	是	NSString *	code
authServerType	是	HWMAuthServerType	鉴权服务器类型

表 6-19 枚举 HWMAuthServerType 说明

值	描述
HWMSDKAuthServerTypeOauth2	单点登录

返回值

表2

示例代码

```
/// SSO 登录接口
[[HWMSdk getOpenApi] loginBySSO:ssoLoginParam callback:^(NSError * _Nullable error, HWMLoginResult * _Nullable result) {
    if (error) {
        [UIUtil showMessageWithError:error];
    } else {
        [UIUtil showMessage:@"sso登录成功" error:error];
    }
}];
```

6.5.2.2 退出登录

logout

接口描述

该接口用于退出已登录的会议帐号。

注意事项

1. 请在已登录的场景下调用。
2. 回调error表示错误信息，nil表示成功。

方法定义

```
- (void)logout:(Nonnull HWMSDKCompleteHandler)callback;
```

参数描述

无。

返回值

表 6-20 退出返回 HWMLogoutResult

参数	类型	描述
userId	NSString *	账号uuid
account	NSString *	登录账号
thirdAccount	NSString *	appid登录的三方账号

示例代码

```
/// 退出登录
- (void)logout {
    [[HWMSdk getOpenApi] logout:^(NSError * _Nullable error, HWMLogoutResult * _Nullable result) {
        if (error == nil) {
            NSLog(@"登出成功");
        } else {
            NSLog(@"登出失败");
        }
    }];
}
```

6.5.2.3 上传个人头像

uploadSelfAvatarWithFilePath:callback:

接口描述

该接口用于上传用户自己头像。

注意事项

1. 文件路径为头像路径。
2. 大小限制200k。
3. 只支持jpg、png。
4. 为了头像展示效果，建议图片宽高比为1:1。
5. 回调error表示错误信息，nil表示成功。

方法定义

```
- (void)uploadSelfAvatarWithFilePath:(NSString * _Nullable)filePath callback:( _Nonnull HWMSDKCompleteHandler)callback;
```

参数描述

表 6-21

参数	是否必须	类型	描述
filePath	否	NSString *	头像文件路径

返回值

无

示例代码

```
/// 上传用户个人头像
- (void)uploadSelfAvatar {
    [[HWMBizSdk getBizOpenApi] uploadSelfAvatarWithFilePath:self.headImgPath callback:^(NSError *
    _Nullable error, id _Nullable result) {
        if (error) {
            [UIUtil showMessage:[NSString stringWithFormat:@"上传失败%@", error.localizedDescription]];
        } else {
            [UIUtil showMessage:@"上传成功"];
        }
    }
    };
```

6.5.3 界面定制

用户可以根据需求自定义一些会控菜单

📖 说明

本章节自定义菜单示例代码为SDK默认，根据自己需求增加（增加的type值不能与SDK中 **HWMSDKConfCtrlType** 的定义重复）或者减少。

6.5.3.1 界面配置

设置自定义界面的代理，根据需求在对应位置设置代理。可以在初始化SDK的时候配置，也可以在使用时配置。

```
[HWMSdk getSdkConfig].socialShareHandler = <#代理对象#>; // 可选，社交分享定制代理，用于定制会中分享能力
[HWMSdk getSdkConfig].audioCallToolBarHandler = <#代理对象#>; // 可选，音频呼叫会控菜单定制代理
[HWMSdk getSdkConfig].videoCallToolBarHandler = <#代理对象#>; // 可选，视频呼叫会控菜单定制代理
[HWMSdk getSdkConfig].audioConfToolBarHandler = <#代理对象#>; // 可选，音频会议会控菜单定制代理
[HWMSdk getSdkConfig].videoConfToolBarHandler = <#代理对象#>; // 可选，视频会议会控菜单定制代理
[HWMSdk getSdkConfig].participantListMenuHandler = <#代理对象#>; // 可选，与会者列表菜单定制代理
[HWMSdk getSdkConfig].confTitleBarMenuHandler = <#代理对象#>; // 可选，会议页面顶部titlebar菜单定制代理
[HWMSdk getSdkConfig].contactUIHandler = <#代理对象#>; // 可选，联系人UI定制代理
[HWMSdk getSdkConfig].headPortraitApiHandler = <#代理对象#>; // 可选，头像拦截显示定制代理
```

界面定制使用item介绍

表 6-22 HWMConfMenuItem 介绍

参数	类型	描述
type	HWMConfCtrlType	会控类型
text	NSString *	正常文本
highlightText	NSString *	高亮文本
selectedText	NSString *	选中文本
isSelected	BOOL	是否选中，默认NO
clickEvent	void(^clickEvent) (HWMConfCtrlType type)	点击item回调

表 6-23 HWMConfToolBarMenuItem 介绍，继承 HWMConfMenuItem

参数	类型	描述
image	UIImage *	普通图片
highlightImage	UIImage *	高亮图片
selectedImage	UIImage *	选中图片
selectedHighlightImage	UIImage *	选中高亮
disableImage	UIImage *	不可点击图片
menuClickEvent	void(^menuClickEvent) (HWMConfCtrlType type, UIButton *menuItemBtn);	点击item回调

表 6-24 HWMConfMenuItem 介绍，继承 HWMConfMenuItem

参数	类型	描述
nemuType	HWMConfMenuItemType	显示类型枚举
image	UIImage *	普通图片
selectedImage	UIImage *	选中图片

表 6-25 HWMConfSettingItem 介绍，继承 HWMConfMenuItem

参数	类型	描述
viewType	HWMConfSettingViewType	显示类型枚举
switchEvent	void (^ switchEvent) (HWMConfCtrlType type, BOOL isSelected)	开关切换的事件回调,只有 HWMConfSettingItemTypeSwitch 类型会回调。isSelected,表示switch是否开启, YES: 开启, NO: 关闭

表 6-26 HWMConfParticipantActionMenuItem 介绍，继承 HWMConfMenuItem

参数	类型	描述
actionClickEvent	void(^actionClickEvent) (HWMConfCtrlType type, HWMConfParticipant *participant)	点击item回调

表 6-27 HWMConfParticipant 介绍

参数	类型	描述
participantId	NSString *	与会者唯一标识
name	NSString *	与会者名字
number	NSString *	号码
accountId	NSString *	uuid,帐号的唯一标识
isMute	BOOL	是否静音
isHandup	BOOL	是否举手
isBroadcast	BOOL	是否被广播

参数	类型	描述
isSelf	BOOL	是否自己
role	ConfRoleType	会议成员角色
state	ConfParticipantStatusType	用户状态
mediaType	NSUInteger	媒体类型
isAnonymous	BOOL	是否匿名用户
isChairman	BOOL	是否是主席
userId	NSUInteger	用户id
clientDeviceType	HWMClientDeviceType	客户端接入类型
isSpeaker	BOOL	是否是最大发言人
displayName	NSString *	默认显示名字，是自己的时候加一个“我”
pinyinName	NSString *	名字的拼音
isAudioJoined	BOOL	是否是音频接入
cameraState	HWMConfCameraState	摄像头状态
orgId	NSString *	所属企业id

表 6-28 枚举 HWMConfCtrlType 介绍（SDK 内部默认会控类型,枚举会持续增加，自定义枚举建议从 1000 开始）

枚举值	值	描述
ConfCtrlMute	1	静音 or 取消静音
ConfCtrlVideo	2	摄像头打开or关闭
ConfCtrlScreenShare	3	屏幕共享 or 停止
ConfCtrlAttendee	4	与会者列表
ConfCtrlMore	5	更多
ConfCtrlLoudSpeaker	6	扬声器 or 麦克风
ConfCtrlTransToVideo	7	语音转视频
ConfCtrlTransToAudio	8	视频转语音
ConfCtrlMuteAll	9	全场静音
ConfCtrlUnmuteAll	10	取消全场静音
ConfCtrlHandup	11	举手

枚举值	值	描述
ConfCtrlRequestChairman	12	申请主持人
ConfCtrlReleaseChairman	13	释放主持人
ConfCtrlTransferChairman	14	转移主持人
ConfCtrlBroadcast	15	广播
ConfCtrlCallOtherNumber	16	呼叫其他号码
ConfCtrlChangeNickName	17	改名
ConfCtrlHangup	18	挂断
ConfCtrlRecall	19	重新呼叫
ConfCtrlWatch	20	选看 or 取消选看
ConfCtrlAllowAttendeeUnmute	21	禁止 or 允许与会者自己解除静音
ConfCtrlRecord	22	开启录制 or 关闭录制
ConfCtrlAddressBook	23	通讯录
ConfCtrlShare	24	社会化分享
ConfCtrlSwitchCamera	25	切换摄像头, 前置 or 后置
ConfCtrlHideSmallVideo	26	隐藏小画面
ConfCtrlLock	27	锁定会议
ConfCtrlInvite	28	邀请
ConfCtrlBeauty	29	美颜
ConfCtrlHowlDetection	30	啸叫检测
ConfCtrlLockShare	31	锁定共享
ConfCtrlQRScan	32	扫一扫
ConfCtrlNetworkDetect	33	网络检测
ConfCtrlFeedback	34	反馈
ConfCtrlRemoveAttendee	35	移除与会者
ConfCtrlChat	36	聊天
ConfCtrlSetting	37	会议设置
ConfCtrlInterpret	38	传译

枚举值	值	描述
ConfCtrlAllowedSpeak	39	允许观众说话
ConfCtrlAllowAudienceWatch	40	允许/禁止观众观看
ConfCtrlCoHost	41	设为/撤销联席主持人
ConfCtrlLocalRecordSetMode	42	本地录制权限模式
ConfCtrlLocalRecordGrantAttendee	43	允许/禁止与会者录制
ConfCtrlSwitchAudienceAndAttendee	44	观众-嘉宾互转
ConfCtrlViewDetails	45	查看详情
ConfCtrlAllowJoinConf	46	允许入会
ConfCtrlWaitingRoom	47	等候室
ConfCtrlAllowChat	48	允许聊天
ConfCtrlMoveToWaitingRoom	49	移至等候室
ConfCtrlAdmitWaitingRoomParticipant	50	准入
ConfCtrlRemoveWaitingRoomParticipant	51	移除
ConfCtrlInviteScreenShare	52	邀请/取消共享
ConfCtrlChatRemind	53	聊天消息提醒
ConfCtrlMirror	54	视频镜像
ConfCtrlAllowAttendeeOpenCamera	55	允许与会者打开摄像头
ConfCtrlReport	56	举报
ConfCtrlAllowRename	57	允许改名
ConfCtrlVote	58	投票
ConfCtrlPrivateChat	59	私聊
ConfCtrlVirtualBackground	60	虚拟背景
ConfCtrlHideFloatMicro	61	隐藏悬浮静音视图
ConfCtrlLiveBroadcast	62	直播

枚举值	值	描述
ConfCtrlHDPREFERRED	63	高清视频优先
ConfCtrlPicRatio	64	视频画面比例
ConfCtrlNoiseReduction	65	音频降噪
ConfCtrlRetrieveChairman	66	收回主持人权限
ConfCtrlEnhanceVideoQuality	67	暗场景增强

表 6-29 枚举 HWMConfMenuItemType 说明

枚举值	描述
HWMConfMenuItemTypeNormal	普通图文
HWMConfMenuItemTypeEnter	左边图文 + 右边箭头进入按钮
HWMConfMenuItemTypeRightBadge	左图文 + 一个右边的badge，暂不开放

表 6-30 枚举 HWMConfSettingViewType 说明

枚举值	描述
HWMConfSettingViewTypeSwitch	左边文字 + 右边开关
HWMConfSettingViewTypeEnter	左文字 + 右箭头

表 6-31 枚举 ConfRoleType 说明

枚举值	描述
ConfRoleTypeAttendee	普通与会者
ConfRoleTypeChairman	主席

表 6-32 枚举 ConfParticipantStatusType 说明

枚举值	描述
ConfParticipantStatusInConf	会议中
ConfParticipantStatusCalling	正在呼叫
ConfParticipantStatusJoining	正在加入会议
ConfParticipantStatusLeft	已经离开
ConfParticipantStatusNotExist	用户不存在
ConfParticipantStatusBusy	被叫用户忙
ConfParticipantStatusNoAnswer	用户无应答
ConfParticipantStatusReject	用户拒绝接听
ConfParticipantStatusCallFailed	呼叫失败

表 6-33 枚举 HWMClientDeviceType 说明

枚举值	描述
HWMClientDeviceTypeDesktop	pc 登录
HWMClientDeviceTypeMobile	移动端登录
HWMClientDeviceTypePad	pad 登录
HWMClientDeviceTypeOther	其他

表 6-34 枚举 HWMConfCameraState 说明

枚举值	描述
HWMConfCameraStateClose	关闭

枚举值	描述
HWMConfCameraStateOpen	打开
HWMConfCameraStateUnknown	未知
HWMConfCameraStateNoCamera	无摄像头

6.5.3.2 清除界面配置

设置相关的代理后如果需要清除对应代理，可以置为nil，根据需求在对应位置设置。

```
[HWMsdk getSdkConfig].socialShareHandler = nil; // 可选，社交分享定制代理，用于定制会中分享能力
[HWMsdk getSdkConfig].audioCallToolBarHandler = nil; // 可选，音频呼叫会控菜单定制代理
[HWMsdk getSdkConfig].videoCallToolBarHandler = nil; // 可选，视频呼叫会控菜单定制代理
[HWMsdk getSdkConfig].audioConfToolBarHandler = nil; // 可选，音频会议会控菜单定制代理
[HWMsdk getSdkConfig].videoConfToolBarHandler = nil; // 可选，视频会议会控菜单定制代理
[HWMsdk getSdkConfig].participantListMenuHandler = nil; // 可选，与会者列表菜单定制代理
[HWMsdk getSdkConfig].confTitleBarMenuHandler = nil; // 可选，会议页面顶部titlebar菜单定制
[HWMsdk getSdkConfig].contactUIHandler = nil; // 可选，联系人UI 定制代理
```

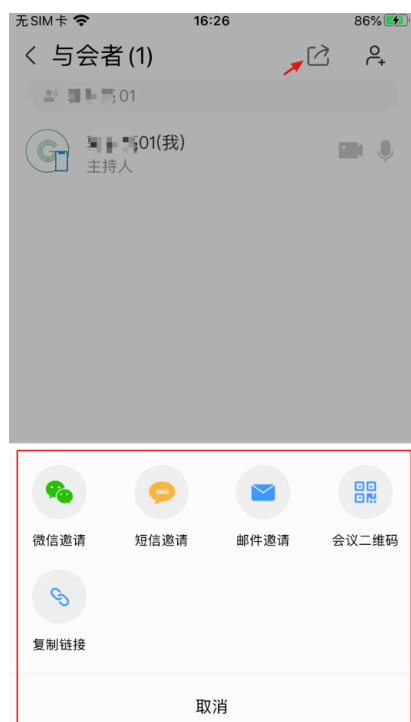
6.5.3.3 社交分享定制

shareConf:isAnonymous:inCtrl

接口描述

该接口用于分享会议拦截处理。实现代理后会拦截SDK默认的分​​享弹窗，如下图红框标注。

图 6-34 分享拦截 SDK 弹窗



注意事项

1. 如果实现该协议方法，则会中调用分享功能，会回调到该方法。
2. shareConf:isAnonymous:inCtrl方法实现后，SDK内部分享UI和会议二维码不显示，所以[getShareLogoImages](#)、[getBrandName](#)设置无效果

方法定义

```
- (void)shareConf:(HWMConfDetail *)confInfo isAnonymous:(BOOL)isAnonymous inCtrl:(UIViewController *)controller;
```

参数描述

表 6-35 参数说明

参数	类型	描述
confInfo	HWMConfDetail *	会议信息
isAnonymous	BOOL	是否匿名入会
controller	UIViewController *	当前的vc，接入方可以在当前controller上弹出自己的定制分享页面

表 6-36 HWMConfDetail 说明

参数	类型	描述
confId	NSString *	会议id。
confSubject	NSString *	会议主题。
accessNumber	NSString *	会议接入码。
chairmanPwd	NSString *	主席密码。
generalPwd	NSString *	来宾密码。
startTime	NSString *	会议开始时间
endTime	NSString *	会议结束时间

返回值

无

示例代码

```
/// 生效后会拦截SDK内部分享弹窗
- (void)shareConf:(HWMConfDetail *)confInfo isAnonymous:(BOOL)isAnonymous inCtrl:(UIViewController *)controller {
    NSLog(@"enter share conf");
    ShareView *shareView = [[ShareView alloc] init];
    [shareView showShareView:^(NSInteger index) {
        [UIUtil showMessage:[NSString stringWithFormat:@"点击了第%zd个按钮", index + 1]];
    }];
}
```

```
});  
}
```

getShareLogolmages

接口描述

该接口用于设置会议二维码分享页面logo。

图 6-35 自定义分享 logo 位置



注意事项

1. logo数组，数组数量为1个。
2. 跟`shareConf:isAnonymous:inCtrl`不能同时使用，同时使用`getShareLogolmages`方法无效果

方法定义

```
-(NSArray <UIImage *>*)getShareLogolmages;
```

参数描述

参数	类型	描述
getShareLogolmages	NSArray <UIImage *>*	二维码分享页面logo数组。

返回值

无

示例代码

```
/// 设置会议二维码分享页面logo数组，数组数量为1个  
-(NSArray<UIImage *> *)getShareLogoImages {  
    UIImage * image = [UIImage imageNamed:@""];  
    return @[image];  
}
```

getBrandName

接口描述

该接口用于设置会议二维码分享页面品牌名。

图 6-36 自定义二维码分享品牌名



注意事项

1. 跟 `shareConf:isAnonymous:inCtrl` 不能同时使用，同时使用 `getBrandName` 方法无效果。

方法定义

```
-(NSString *)getBrandName;
```

参数描述

参数	类型	描述
getBrandName	NSString *	会议二维码分享页面品牌名。

返回值

无

示例代码

```
/// 设置会议二维码分享页面品牌名  
- (NSString *)getBrandName {  
    return @"集成品牌名";  
}
```

6.5.3.4 底部工具栏定制

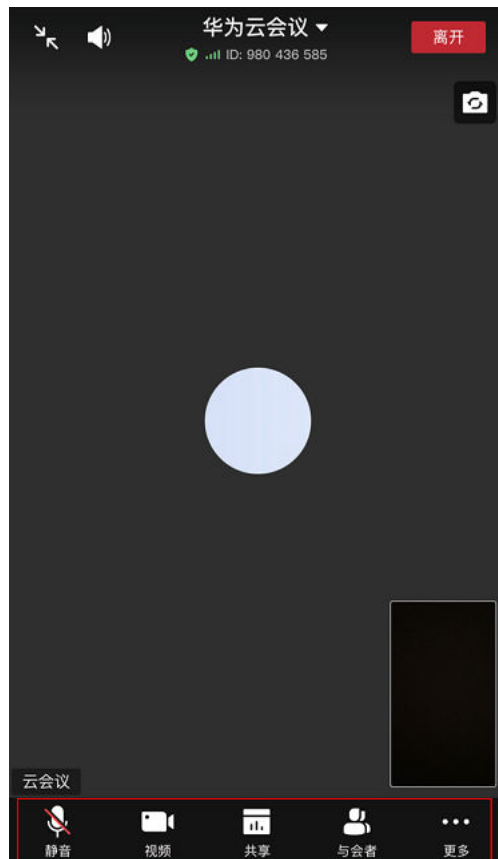
此章节以视频会议为例（音频呼叫、视频呼叫、音频会议同理使用）。

buildToolBarMenuItems

接口描述

该接口用于构建视频会议底部toolbar的会控全量菜单。

图 6-37 视频会议会控菜单定制位置



注意事项

1. 此方法用于视频会议底部toolbar的定制。

方法定义

```
- (NSArray <HWMConfToolBarMenuItem *> *)buildToolBarMenuItems;
```

参数描述

无

返回值

无

示例代码

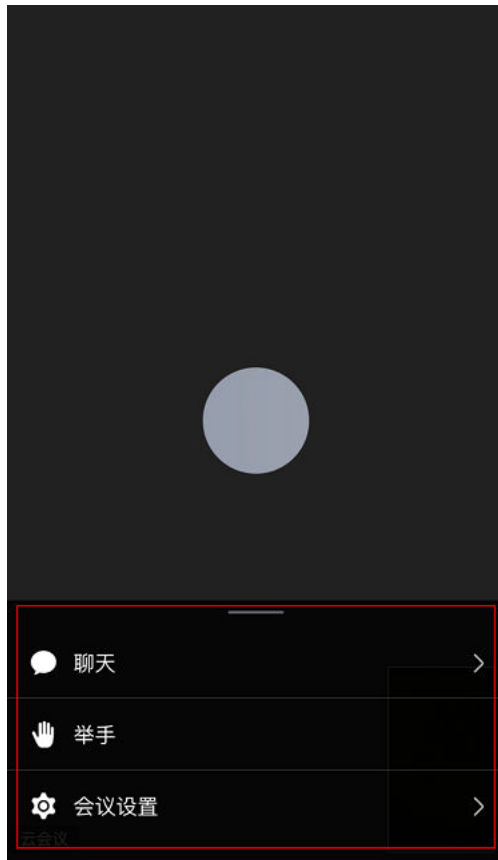
```
/// 构建视频会议底部toolbar的会控全量菜单  
- (NSArray <HWMConfToolBarMenuItem *> *)buildToolBarMenuItems {  
    // 麦克风静音 or 取消静音  
    HWMConfToolBarMenuItem *micItem = [HWMConfToolBarMenuItem defaultMicItem];  
    // 打开 or 关闭摄像头  
    HWMConfToolBarMenuItem *videoItem = [HWMConfToolBarMenuItem defaultVideoItem];  
    // 屏幕共享开始 or 停止  
    HWMConfToolBarMenuItem *screenShareItem = [HWMConfToolBarMenuItem defaultScreenShareItem];  
    // 与会者列表  
    HWMConfToolBarMenuItem *attendeeltem = [HWMConfToolBarMenuItem defaultAttendeeltem];  
    // 更多  
    HWMConfToolBarMenuItem *moreItem = [HWMConfToolBarMenuItem defaultMoreItem];  
  
    return @[micItem, videoItem, screenShareItem, attendeeltem, moreItem];  
}
```

buildToolBarMoreMenuItems

接口描述

该接口用于构建视频会议底部toolbar的“更多”菜单。

图 6-38 视频会议底部 toolbar 的“更多”菜单



注意事项

无

方法定义

```
-(NSArray <HWMConfMenuItem *> *)buildToolBarMoreMenuItems
```

参数描述

无

返回值

无

示例代码

```
/// 构建视频会议底部toolbar的“更多”菜单
-(NSArray <HWMConfMenuItem *> *)buildToolBarMoreMenuItems {
    // 聊天
    HWMConfMenuItem *chatItem = [HWMConfMenuItem defaultChatItem];
    // 邀请
    HWMConfMenuItem *inviteItem = [HWMConfMenuItem defaultInviteItem];
    // 录制
    HWMConfMenuItem *recordItem = [HWMConfMenuItem defaultRecordItem];
    // 举手
    HWMConfMenuItem *raiseHandsItem = [HWMConfMenuItem defaultRaiseHandsItem];
    // 切换摄像头
    HWMConfMenuItem *switchCameraItem = [HWMConfMenuItem defaultSwitchCameraItem];
    // 设置
    HWMConfMenuItem *settingItem = [HWMConfMenuItem defaultSettingItem];
}
```

```
return @[chatItem, inviteItem, recordItem, raiseHandsItem, switchCameraItem, settingItem];
}
```

buildSettingItems

接口描述

该接口用于构建视频会议点击更多菜单的设置弹出的设置列表项目。

图 6-39 视频会议点击更多菜单弹出列表定制位置



注意事项

无

方法定义

```
-(NSArray <HWMConfSettingGroup *> *)buildSettingItems;
```

参数描述

无

返回值

无

示例代码

```
// 构建视频会议点击更多菜单的设置弹出的设置列表项目
-(NSArray <HWMConfSettingGroup *> *)buildSettingItems {
```

```

HWMConfSettingGroup *audioVideoGroup = [[HWMConfSettingGroup alloc] initWithTitle:@"音视频"];
HWMConfSettingItem *howlItem = [HWMConfSettingItem defaultHowlDetection];
HWMConfSettingItem *beautyItem = [HWMConfSettingItem defaultBeautyItem];
HWMConfSettingItem *hideSmallViewItem = [HWMConfSettingItem defaultHideSmallVideoItem];
audioVideoGroup.items = @[howlItem, beautyItem, hideSmallViewItem];

HWMConfSettingGroup *securityGroup = [[HWMConfSettingGroup alloc] initWithTitle:@"安全"];
HWMConfSettingItem *lockItem = [HWMConfSettingItem defaultLockItem];
securityGroup.items = @[lockItem];

HWMConfSettingGroup *authorityGroup = [[HWMConfSettingGroup alloc] initWithTitle:@"权限"];
HWMConfSettingItem *allowUnmuteItem = [HWMConfSettingItem defaultAllowAttendeeUnmuteItem];
HWMConfSettingItem *shareLockItem = [HWMConfSettingItem defaultLockShareItem];
authorityGroup.items = @[allowUnmuteItem, shareLockItem];

HWMConfSettingGroup *commonGroup = [[HWMConfSettingGroup alloc] initWithTitle:@"通用"];
HWMConfSettingItem *networkDetectItem = [HWMConfSettingItem defaultNetworkDetectItem];
HWMConfSettingItem *feedbackItem = [HWMConfSettingItem defaultFeedbackItem];
commonGroup.items = @[networkDetectItem, feedbackItem];

return @[audioVideoGroup, securityGroup, authorityGroup, commonGroup];
}

```

6.5.3.5 与会者列表菜单定制

buildParticipantToolBarMenuItems

接口描述

该接口用于构建与会者列表底部toolbar上的菜单。

图 6-40 与会者列表菜单定制位置



注意事项

无

方法定义

```
- (NSArray <HWMConfToolBarMenuItem *> *)buildParticipantToolBarMenuItems;
```

参数描述

无

返回值

无

示例代码

```
/// 构建与会者列表底部toolbar上的菜单  
- (NSArray <HWMConfToolBarMenuItem *> *)buildParticipantToolBarMenuItems{  
    // 全场静音  
    HWMConfToolBarMenuItem *muteAllItem = [HWMConfToolBarMenuItem defaultMuteAllItem];  
    // 解除全场静音  
    HWMConfToolBarMenuItem *unmuteAllItem = [HWMConfToolBarMenuItem defaultUnmuteAllItem];  
    // 更多  
    HWMConfToolBarMenuItem *moreItem = [HWMConfToolBarMenuItem defaultMemberListMoreItem];  
    // 举手  
    HWMConfToolBarMenuItem *handupItem = [HWMConfToolBarMenuItem defaultHandupItem];  
    // 请求主持人  
    HWMConfToolBarMenuItem *requestChairmanItem = [HWMConfToolBarMenuItem  
    defaultRequestChairmanItem];  
  
    return @[muteAllItem, unmuteAllItem, moreItem, handupItem, requestChairmanItem];  
}
```

buildParticipantToolBarMoreMenuItems

接口描述

该接口用于构建与会者列表底部toolbar上的“更多”菜单。

图 6-41 与会者列表更多定制位置



注意事项

无

方法定义

```
- (NSArray <HWMConfMenuItem *> *)buildParticipantToolBarMoreMenuItems;
```

参数描述

无

返回值

无

示例代码

```
/// 构建与会者列表底部toolbar上的“更多”菜单  
- (NSArray <HWMConfMenuItem *> *)buildParticipantToolBarMoreMenuItems{  
    // 释放主持人  
    HWMConfMenuItem *releaseChairmanItem = [HWMConfMenuItem defaultReleaseChairmanItem];  
    // 锁定会议  
    HWMConfMenuItem *lockItem = [HWMConfMenuItem defaultLockItem];  
    // 允许 or 禁止 与会者自己解除静音  
    HWMConfMenuItem *allowUnmuteItem = [HWMConfMenuItem defaultAllowAttendeeUnmuteItem];  
  
    return @[releaseChairmanItem, lockItem, allowUnmuteItem];  
}
```

buildParticipantTitleBarMenuItems

接口描述

该接口用于构建与会者列表右上角的菜单。

图 6-42 与会者列表右上角的菜单定制位置



注意事项

无

方法定义

```
-(NSArray <HWMConfToolBarMenuItem *> *)buildParticipantTitleBarMenuItems;
```

参数描述

无

返回值

无

示例代码

```
/// 构建与会者列表右上角的菜单
-(NSArray <HWMConfToolBarMenuItem *> *)buildParticipantTitleBarMenuItems{
    HWMConfToolBarMenuItem *shareItem = [HWMConfToolBarMenuItem defaultShareItem];
    HWMConfToolBarMenuItem *addressBookItem = [HWMConfToolBarMenuItem defaultAddressBookItem];

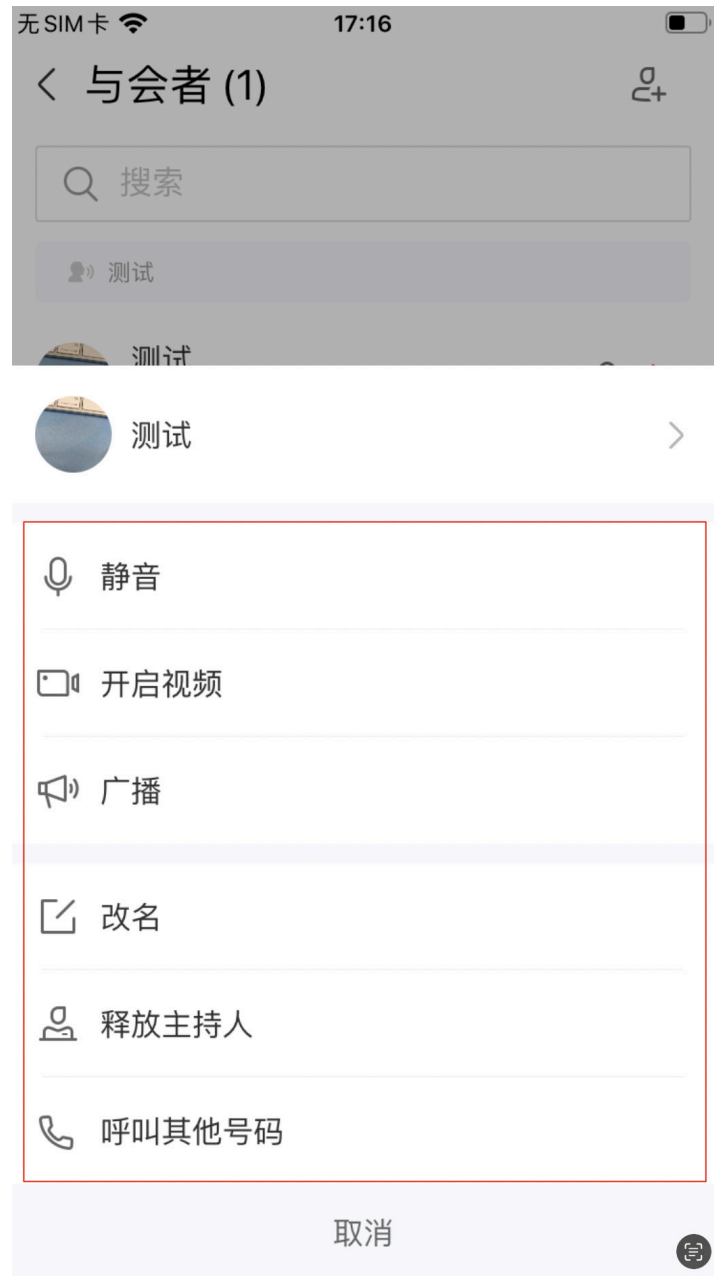
    return @[shareItem, addressBookItem];
}
```


buildParticipantActionSheetItems

接口描述

该接口用于构建与会者列表中成员点击弹出的actionsheet的菜单。

图 6-43 与会者列表中成员点击弹出的 actionsheet 的菜单定制位置



注意事项

无

方法定义

```
-(NSArray <HWMConfParticipantActionMenuItem *> *)buildParticipantActionSheetItems;
```

参数描述

无

返回值

无

示例代码

```

/// 构建与会者列表中成员点击弹出的actionsheet的菜单
- (NSArray <HWMConfParticipantActionMenuItem *> *)buildParticipantActionSheetItems{
    // 请求主持人
    HWMConfParticipantActionMenuItem *requestChairmanItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultRequestChairmanItem];
    // 释放主持人
    HWMConfParticipantActionMenuItem *releaseChairmanItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultReleaseChairmanItem];
    // 广播会场 or 停止
    HWMConfParticipantActionMenuItem *broadcastItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultBroadcastItem];
    // 静音 or 取消静音
    HWMConfParticipantActionMenuItem *muteItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultMuteItem];
    // 挂断
    HWMConfParticipantActionMenuItem *hangupItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultHangupItem];
    // 移除与会者
    HWMConfParticipantActionMenuItem *removeAttendeeItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultRemoveAttendeeItem];
    // 转移主持人
    HWMConfParticipantActionMenuItem *transferItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultTransferChairmanItem];
    // 选看 or 取消选看
    HWMConfParticipantActionMenuItem *watchItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultWatchItem];
    // 举手 or 手放下
    HWMConfParticipantActionMenuItem *handup = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultHandupItem];
    // 呼叫其他号码
    HWMConfParticipantActionMenuItem *callOtherItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultCallOtherNumberItem];
    // 改名
    HWMConfParticipantActionMenuItem *changeNameItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultChangeNickNameItem];
    // 重呼
    HWMConfParticipantActionMenuItem *recallItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultRecallItem];

    return @[releaseChairmanItem, requestChairmanItem, broadcastItem, watchItem, muteItem,
transferItem, recallItem, callOtherItem, changeNameItem, handup, hangupItem, removeAttendeeItem];
}

```

buildParticipantActionSheetItems 头像点击事件自定义

接口描述

该接口用于构建与会者列表中成员点击弹出的actionsheet的菜单后，自定义头像点击事件。

图 6-44 与会者列表中成员点击弹出的 actionsheet 的菜单，头像点击定制位置



注意事项

1. 需要自定义buildParticipantActionSheetItems，添加defaultViewDetails的item。
2. defaultViewDetails设置delegate对象
3. 在代理对象位置实现代理

方法定义

无

参数描述

无

返回值

无

示例代码

```

/// 是否显示头像右边箭头及是否可以点击
- (BOOL)isShowAccessoryView:(HWMConfParticipant *)participant {
    // 根据需要返回是否显示
    return YES;
}

/// 构建与会者列表中成员点击弹出的actionsheet的菜单
- (NSArray <HWMConfParticipantActionMenuItem *> *)buildParticipantActionSheetItems{
    // 查看详情
    HWMConfParticipantActionMenuItem *detailsItem = [HWMConfParticipantActionMenuItem
defaultViewDetails];
    detailsItem.delegate = self;
    detailsItem.actionClickEvent = ^(HWMConfCtrlType type, HWMConfParticipant * _Nonnull participant) {
        NSLog(@"处理点击事件");
    };
    // 以上为查看详情自定义相关

    // 请求主持人
    HWMConfParticipantActionMenuItem *requestChairmanItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultRequestChairmanItem];
    // 释放主持人
    HWMConfParticipantActionMenuItem *releaseChairmanItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultReleaseChairmanItem];
    // 广播会场 or 停止
    HWMConfParticipantActionMenuItem *broadcastItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultBroadcastItem];
    // 静音 or 取消静音
    HWMConfParticipantActionMenuItem *muteItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultMuteItem];
    // 挂断
    HWMConfParticipantActionMenuItem *hangupItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultHangupItem];
    // 移除与会者
    HWMConfParticipantActionMenuItem *removeAttendeeItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultRemoveAttendeeItem];
    // 转移主持人
    HWMConfParticipantActionMenuItem *transferItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultTransferChairmanItem];
    // 选看 or 取消选看
    HWMConfParticipantActionMenuItem *watchItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultWatchItem];
    // 举手 or 手放下
    HWMConfParticipantActionMenuItem *handup = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultHandupItem];
    // 呼叫其他号码
    HWMConfParticipantActionMenuItem *callOtherItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultCallOtherNumberItem];
    // 改名
    HWMConfParticipantActionMenuItem *changeNameItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultChangeNickNameItem];
    // 重呼
    HWMConfParticipantActionMenuItem *recallItem = (HWMConfParticipantActionMenuItem *)
[HWMConfParticipantActionMenuItem defaultRecallItem];

    return @[releaseChairmanItem, requestChairmanItem, broadcastItem, watchItem, muteItem,
transferItem, recallItem, callOtherItem, changeNameItem, handup, hangupItem, removeAttendeeItem];
}

```

6.5.3.6 会议页面顶部 titlebar 菜单定制

buildTitleBarPopViewItems

接口描述

该接口用于构建会议titleBar点击的弹出菜单。

图 6-45 会议页面顶部 titlebar 菜单定制位置



注意事项

1. 主持人密码只有主持人可见。其它定制菜单，所有人显示一致。
2. 自定义cell时需要遵循HWMInMeetingCellDelegate（只修改默认顺序不需要遵循代理）

方法定义

```

- (NSArray <HWMInMeetingCellModel *> *)buildTitleBarPopViewItems;

// 自定义cell使用以下，遵循HWMInMeetingCellDelegate
/// cell 对象代理
/// @param cellType cellType
- (UITableViewCell *)cellForRowWithCellType:(HWMConfInfoCellType)cellType inTableView:(UITableView *)tableView indexPath:(NSIndexPath *)indexPath;
/// cell点击事件代理
/// @param cellType cellType
- (void)cellSelectedWithCellType:(HWMConfInfoCellType)cellType;
    
```

参数描述

无

返回值

无

示例代码

```

/// 构建会议titleBar 点击的弹出菜单
- (NSArray <HWMInMeetingCellModel *> *)buildTitleBarPopViewItems{
    HWMInMeetingCellModel *subjectCell = [HWMInMeetingCellModel defaultSubjectCell];
    }
    
```

```

HWMInMeetingCellModel *encryptCell = [HWMInMeetingCellModel defaultEncryptedCell];
HWMInMeetingCellModel *confldCell = [HWMInMeetingCellModel defaultConfldCell];
HWMInMeetingCellModel *generalPasswordCell = [HWMInMeetingCellModel
defaultGeneralPasswordCell];
HWMInMeetingCellModel *hostPasswordCell = [HWMInMeetingCellModel defaultHostPasswordCell];
HWMInMeetingCellModel *lockCell = [HWMInMeetingCellModel defaultConfLockCell];
HWMInMeetingCellModel *qosCell = [HWMInMeetingCellModel defaultQosCell];

return @[subjectCell,encryptCell, confldCell, generalPasswordCell, hostPasswordCell, lockCell, qosCell];
}
/// cell 对象代理, 高度自适应, 根据需求自定义cell显示, 以下为示例代码仅供参考
/// @param cellType cellType
- (UITableViewCell *)cellForRowWithCellType:(HWMConfInfoCellType)cellType inTableView:(nonnull
UITableView *)tableView indexPath:(nonnull NSIndexPath *)indexPath{
    NSString *reuseIdentifier = [NSString stringWithFormat:@"CustomCell_%ld", cellType];
    UITableViewCell *cell = [tableView dequeueReusableCellWithIdentifier:reuseIdentifier];
    if (cell == nil) {
        cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleDefault
reuseIdentifier:reuseIdentifier];
    }
    cell.contentView.backgroundColor = [UIColor clearColor];
    cell.backgroundColor = [UIColor clearColor];
    cell.selectionStyle = UITableViewCellSelectionStyleNone;

    UIImageView * safeImg = [[UIImageView alloc] initWithImage:[UIImage
imageNamed:@"custom_conf_setting"]];
    [cell.contentView addSubview:safeImg];

    UILabel * safeLabel = [[UILabel alloc] init];
    safeLabel.font = [UIFont systemFontOfSize:14];
    safeLabel.textColor = [UIColor whiteColor];
    safeLabel.text = reuseIdentifier;
    [cell.contentView addSubview:safeLabel];
    [safeImg mas_makeConstraints:^(MASConstraintMaker *make) {
        make.left.mas_equalTo(16);
        make.size.mas_equalTo(CGSizeMake(16, 16));
        make.centerY.mas_equalTo(safeLabel);
    }];

    [safeLabel mas_makeConstraints:^(MASConstraintMaker *make) {
        make.left.mas_equalTo(safeImg.mas_right).offset(10);
        make.top.mas_equalTo(10);
        make.height.mas_equalTo(34);
        make.bottom.mas_equalTo(-10);
    }];
    return cell;
}

/// cell点击事件代理
/// @param cellType cellType
- (void)cellSelectedWithCellType:(HWMConfInfoCellType)cellType{
    [UIUtil showMessage:[NSString stringWithFormat:@"点击了菜单: %lu", (unsigned long)cellType]];
}

```

6.5.3.7 自定义视频会议来电时接听方式按钮

getVideoConfIncomingAnswerBtnShowType

接口描述

该接口用于自定义视频会议来电时接听按钮。默认情况下视频会议来电时有视频接听和语音接听两种方式, 可以通过该接口配置成只显示一种接听方式。

图 6-46 会议来电接听方式按钮自定义



注意事项

1. RTC会议时，禁止语音接通。

方法定义

- (HWMAnswerBtnShowType)getVideoConfIncomingAnswerBtnShowType;

参数描述

表 6-37 枚举 HWMOrderConfType 说明

参数	描述
HWMAnswerBtnShowTypeAll	全部显示
HWMAnswerBtnShowTypeOnlyAudio	只显示音频接听
HWMAnswerBtnShowTypeOnlyVideo	只显示视频接听

示例代码

```
/// 视频会议来电时按钮显示状态  
- (HWMAnswerBtnShowType)getVideoConfIncomingAnswerBtnShowType {  
    return HWMAnswerBtnShowTypeAll;  
}
```

6.5.3.8 联系人 UI 定制

openContactSelectPage:scene:completeHandler:

接口描述

该接口用于定制打开联系人选择页面。

图 6-47 生效后拦截 SDK 选择联系人界面



注意事项

1. 如果自定义需要自己构建加人UI界面，组装添加人员数组。

方法定义

```
- (void)openContactSelectPage:(NSArray<HWMConfContactModel *> *)param  
    scene:(HWMSelectedContactScene)scene  
    completeHandler:(void (^)(NSArray<HWMContactSelectedModel *> *result, NSError  
*error))handler;
```

参数描述

无

返回值

无

示例代码


```

/// 定制打开联系人选择页面
- (void)openContactSelectPage:(NSArray<HWMConfContactModel *> *)param
    scene:(HWMSelectedContactScene)scene
    completionHandler:(void (^)(NSArray<HWMContactSelectedModel *> *result, NSError
*error))handler {
    NSLog(@"enter open contact select page");
    ContactSelectViewController * contactSelectVc = [[ContactSelectViewController alloc]
initWithResultHandler:handler];
    [self pushViewControllerWithVc:contactSelectVc];
}

/// 模拟回调添加与会者人员
- (void)initSlectedModels {
    NSMutableArray *slectedModels = [NSMutableArray array];
    HWMContactSelectedModel *model = [[HWMContactSelectedModel alloc] initWithName:@"测试帐号01"
number:@""" contactType:HWMContactTypeEnterprise];
    [slectedModels addObject:model];
    HWMContactSelectedModel *model2 = [[HWMContactSelectedModel alloc] initWithName:@"测试帐号
02" number:@""" contactType:HWMContactTypeEnterprise];
    [slectedModels addObject:model2];
    HWMContactSelectedModel *model3 = [[HWMContactSelectedModel alloc] initWithName:@"测试帐号
03" number:@""" contactType:HWMContactTypeEnterprise];
    [slectedModels addObject:model3];
    HWMContactSelectedModel *model4 = [[HWMContactSelectedModel alloc] initWithName:@"测试帐号
04" number:@""" contactType:HWMContactTypeEnterprise];
    [slectedModels addObject:model4];

    /// AppId 接入场景示例代码
    HWMContactSelectedModel *model5 = [[HWMContactSelectedModel alloc] initWithName:@"测试帐号
05" number:@""" contactType:HWMContactTypeEnterprise];
    model5.thirdUserId = @"test";
    [slectedModels addObject:model5];

    self.completeHandler(slectedModels, nil);
}

```

📖 说明

- 1.AppId 登录场景建议使用自定义联系人及 UI ，构建选中联系人数据时，需要传入第三方帐号 thirdUserId。
- 2.completeHander直接回调需要添加的人员数组，如果不需要UI界面，可根据需求直接回调相关数据。
- 3.initSlectedModels为模拟测试数据，接入时需要传入真实数据。

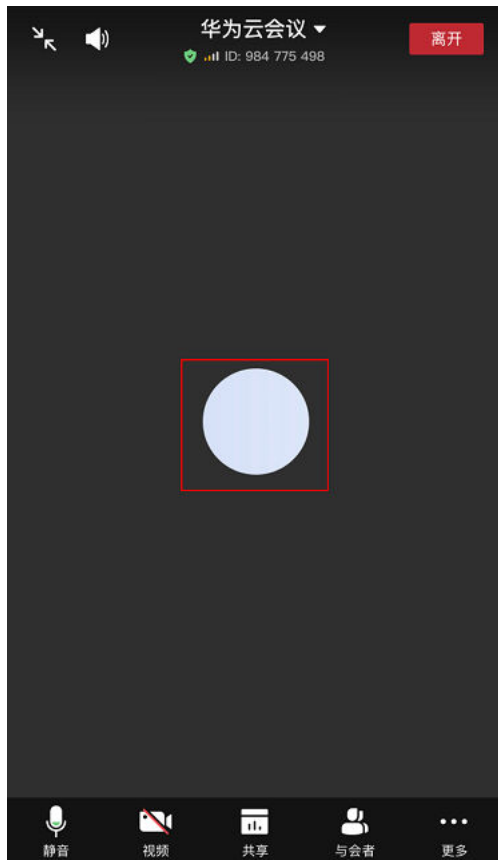
6.5.3.9 头像拦截显示

queryContactAvatarWithAccount:thirdUserId:number:completionBlock:

接口描述

该接口用于定制登录后的会中头像显示。

图 6-48 生效后拦截头像显示效果



注意事项

1. 未登录的头像不支持自定义。

方法定义

```
- (void)queryContactAvatarWithAccount:(NSString *)account thirdUserId:(NSString *)thirdUserId number:(NSString *)number completionBlock:(void (^)(NSError * _Nullable error, UIImage * _Nullable avatarImage))completionBlock;
```

参数描述

表 6-38

参数	类型	描述
account	NSString *	帐号密码登录场景下优先使用此字段
thirdUserId	NSString *	appid登录场景下优先使用此字段
number	NSString *	跨企业场景优先使用此字段

返回值

无

示例代码

```

/// 拦截头像显示
- (void)queryContactAvatarWithAccount:(NSString *)account thirdUserId:(NSString *)thirdUserId number:
(nonnull NSString *)number
    completionBlock:(void(^)(NSError * _Nullable error, UIImage * _Nullable
avatarImage))completionBlock {
    /// 根据account或者thirdUserId 设置图片，下面的仅为示例代码
    UIImage *image = [UIImage imageNamed:@"custom_conf_setting"];
    completionBlock(nil, image);
}
    
```

6.5.3.10 自定义会中水印

screenShareCoverView

接口描述

该接口用于屏幕共享/白板共享设置水印。该水印是在接收端叠加到共享内容上的。服务器上配置后才会显示水印。

说明

- 1、在桌面共享、程序共享、白板共享的接收端叠加水印图片，只有服务器上会中水印打开的情况下自定义水印才生效。如果不自定义，默认按照手机号、邮箱、会议ID+昵称优先级顺序显示，字体透明度10%。
- 2、自定义水印时用户截屏时会有弹窗提示，如果不需要在初始化时配置关闭。

图 6-49 设置水印效果，图片仅为示例



注意事项

1. 注意不要添加全局手势。
2. 自定义视图为全屏View,在共享视图上层，请设置合适透明度防止遮盖分享信息。
3. 自定义弹窗提示是否需要关闭需在SDK初始化时配置参数
enablePromptUserScreenshot

方法定义

```
- (UIView *)screenShareCoverView;
```

参数描述

无

返回值

无

示例代码

```
/// 屏幕共享/白板共享的设置水印。注意不要添加全局手势  
- (UIView *)screenShareCoverView {  
    UIImageView *coverImgView = [[UIImageView alloc] init];  
    coverImgView.image = [UIImage imageNamed:@"img"];  
    coverImgView.alpha = 0.2;  
    return coverImgView;  
}
```

6.5.3.11 录制按钮操作定制

buildRecordMenuItems

接口描述

该接口用于定制录制状态栏旁的录制按钮是否隐藏。

图 6-50 可定制位置



注意事项

1. 暂时只支持录制右侧按钮隐藏，不返回item就不显示。

方法定义

```
-(NSArray <HWMConfStatusBarMenuItem *> *)buildRecordMenuItems;
```

参数描述

无

返回值

无

示例代码

```

/// 录制按钮菜单栏定制
- (NSArray<HWMConfStatusBarMenuItem *> *)buildRecordMenuItems {
    HWMConfStatusBarMenuItem *recordItem = [HWMConfStatusBarMenuItem defaultRecordItem];
    // 如果不需要录制按钮右侧的操作按钮，数组返回为空即可
    return @[recordItem];
}
    
```

6.5.4 会议控制

6.5.4.1 创建会议

createConf

接口描述

该接口用于创建即时会议。

注意事项

1. 该接口默认已将自己作为主席加入会议，调用接口时，在与会者参数中不需要带上自己的信息。
2. 若需要在创建会议的同时邀请其他与会者，需要传入其他与会者信息。
3. 开启自动录制会议时必须开启录制会议。
4. 创会结果在callback中返回。

方法定义

```

- (void)createConf:(HWMCreateConfParam * _Nonnull)param callback:( _Nonnull
HWMSDKCreateConfCompleteHandler)callback;
    
```

参数描述

表 6-39 HWMCreateConfParam 参数说明

参数	是否必须	类型	描述
subject	是	NSString *	会议主题。
confType	是	HWMConfType	会议媒体类型 默认语音会议，如果要支持屏幕共享能力，需传入数据会议。
needPassword	否	BOOL	会议是否需要设置密码，默认不需要。 说明 仅对随机会议ID生效
guestPwd	否	NSString *	来宾密码 说明 仅对随机会议ID生效。

参数	是否必须	类型	描述
members	否	NSArray <HWMCreateConfMember*> *	成员列表，可选
isCameraOn	否	BOOL	是否打开摄像头，默认关闭
isMicOn	否	BOOL	是否打开麦克风，默认打开
isRecordOn	否	BOOL	录制会议开关 说明 仅对云录制生效，客户端本地录制不生效。
isAutoRecord	否	BOOL	会议开始后是否自动开启录制，默认关闭。开启自动录制会议时必须开启录制会议 说明 仅对云录制生效，客户端本地录制不生效。
vmrId	否	NSString *	个人会议或者云会议室Id，创建个人会议或云会议室的时候传入，其它时候传入空
joinConfRestrictionType	否	HWMJoinConfRestrictionType	允许入会范围限制
isOpenWaitingRoom	否	BOOL	等候室开启开关 说明 需要等候室功能开通后才能生效

表 6-40 枚举 HWMConfType 说明

枚举值	描述
HWMConfTypeAudio	音频会议。
HWMConfTypeVideo	视频会议。
HWMConfTypeAudioAndData	音频数据会议，支持屏幕共享能力。
HWMConfTypeVideoData	视频数据会议，支持屏幕共享能力。

表 6-41 HWMCreateConfMember 参数说明

参数	是否必须	类型	描述
name	是	NSString *	与会者名字
number	是	NSString *	Sip呼叫号码或者电话号码（number和thirdUserId二选一）
thirdUserId	是	NSString *	APPID 登录场景下的第三方帐号（number和thirdUserId二选一）
accountId	否	NSString *	用户帐号ID。

返回值

无

示例代码

```

/// 创建会议
- (void)creatMeeting {
    HWMCreateConfParam *param = [[HWMCreateConfParam alloc] init];
    param.subject = @"会议标题";
    param.confType = HWMConfTypeVideoData;
    param.needPassword = YES;// 会议是否需要设置密码，默认不需要
    param.isCameraOn = YES;// 是否打开摄像头，默认关闭
    param.isMicOn = YES;// 是否打开麦克风，默认打开
    param.isRecordOn = NO;// 是否打开会议录制，默认关闭
    param.joinConfRestrictionType = HWMJoinConfRestrictionAll;// 允许入会范围
    param.joinConfRestrictionType = NO;// 是否开启等候室
    // 与会者列表
    if (self.selectedMemebrs) {
        _block NSMutableArray *members = [[NSMutableArray alloc] init];
        [self.selectedMemebrs enumerateObjectsUsingBlock:^(HWMContactSelectedModel * _Nonnull obj,
        NSUInteger idx, BOOL * _Nonnull stop) {
            HWMCreateConfMember *member = [[HWMCreateConfMember alloc] init];
            member.accountId = obj.accountId;
            member.number = obj.number;
            member.name = obj.name;
            member.thirdUserId = obj.thirdUserId;
            [members addObject:member];
        }];
        param.members = members;
    }
    [[HWMsSdk getOpenApi] createConf:param callback:^(NSError * _Nullable error, HWMCreateConfResult *
    _Nullable result) {
        [self hideLoading];
        if (error) {
            NSLog(@"创会失败%@", error.localizedDescription);
        }else{
            NSLog(@"创会成功");
        }
    }];
}

```


6.5.4.2 加入会议

joinConf

接口描述

该接口用于通过会议ID和密码加入已经存在的会议。加入会议的方式有两种：

- 会议ID+会议密码入会：手动输入会议ID和会议密码密码。
- 会议ID+Random入会：通过邮件会议通知中的会议链接入会时（会议链接如：<https://bmeeting.huaweicloud.com/#/j/981924330/e11ddec62ee760734fcf2ba55b79937eac2aa68677caf659>），链接里面的的是会议ID（即举例中的981924330）和会议的Random码（即举例中的e11ddec62ee760734fcf2ba55b79937eac2aa68677caf659）。

注意事项

1. ID入会参数使用子类HWMJoinConfParam，random入会参数使用子类HWMRandomConfParam
2. 在登录和未登录状态下都可以调用该接口。
3. 实际业务处理结果在对应的callback回调中。

方法定义

```
-(void)joinConf:(HWMBaseJoinConfParam * _Nonnull)param callback:(_Nonnull HWMSDKCompleteHandler)callback;
```

参数描述

表 6-42 HWMBaseJoinConfParam

参数	是否必须	类型	描述
confId	是	NSString *	会议ID。
nickname	否	NSString *	会议昵称。在已登录状态下可为空，已登录时空将使用登录的用户名作为会场名。
isCameraOn	是	BOOL	是否打开摄像头，默认关闭。
isMicOn	是	BOOL	是否打开麦克风，默认打开。

表 6-43 HWMJoinConfParam 参数说明，继承 HWMBaseJoinConfParam

参数	是否必须	类型	描述
password	否	NSString *	会议密码。若希望以主席身份入会，必须传入主席密码。若希望以来宾身份入会，会议要求来宾密码时，必须传入来宾密码，会议不要求来宾密码时，可以为空。

参数	是否必须	类型	描述
isStopConflictConf	否	BOOL	强制结束和当前VMR会议资源冲突的会议（必需是vmr资源的拥有者才有权限，否则无效）

表 6-44 HWMRandomConfParam 参数说明，继承 HWMBaseJoinConfParam

参数	是否必须	类型	描述
random	是	NSString *	random入会时传入。

示例代码

```

/// 加入会议
- (void)joinMeeting {
    HWMJoinConfParam *param = [[HWMJoinConfParam alloc] init];
    param.confId = @"989156631";
    param.password = @"807766";
    param.nickname = @"杭州会场";
    param.isCameraOn = NO;
    param.isMicOn = YES;
    [[HWM SDK getOpenApi] joinConf:param callback:^(NSError * _Nullable error, id _Nullable result) {
        if (error) {
            NSLog(@"入会失败%@", error.localizedDescription);
        }else{
            NSLog(@"入会成功");
        }
    }];
}

/// random入会
- (void)joinMeetingByRandom {
    HWMRandomConfParam *param = [[HWMRandomConfParam alloc] init];
    param.confId = @"989156631";
    param.random = @"1da9ec437c5aa24beb1e2b67c783f59d0859c578022ba88c";
    param.nickname = @"杭州会场";
    param.isCameraOn = NO;
    param.isMicOn = YES;
    [[HWM SDK getOpenApi] joinConf:param callback:^(NSError * _Nullable error, id _Nullable result) {
        [self hideLoading];
        if (error) {
            NSLog(@"入会失败%@", error.localizedDescription);
        }else{
            NSLog(@"入会成功");
        }
    }];
}

```

6.5.4.3 添加与会者

addAttendee

接口描述

该接口用于在会议中添加与会者。

注意事项

1. 必须在会议中，并且只有主持人才可以加入；

方法定义

```
-(void)addAttendee:(NSArray<HWMContactSelectedModel *> *)params completionHandler:
(HWMSDKCompleteHandler)handler;
```

参数描述

表 6-45 HWMContactSelectedModel 说明

参数	是否必须	类型	描述
name	是	NSString *	会场名。
number	是	NSString *	号码。如果填分配给帐号的sip号码（如 +99111244216210249）则呼叫该帐号的App；如果填PSTN号码（如 18700000000），则通过VoIP网关呼叫该号码，前提是该企业已开通PSTN呼叫权限。与thirdUserId二选一，帐号密码鉴权时使用。
thirdUserId	是	NSString *	第三方用户ID。与number二选一，App ID鉴权时使用。

示例代码

```
/// 会中加入
[[HWMSdk getOpenApi] addAttendee:attendee completionHandler:^(NSError * _Nullable error, id _Nullable result) {
    [UIUtil showMessage:@"邀请成功" error:error];
}];
```

6.5.4.4 离开会议

leaveConf

接口描述

该接口用于离开会议。

注意事项

1. 不在会议中不应调用此接口。

2. 调用该接口离开会议，会议不会结束，在会议存在期间可以再次入会。

方法定义

```
- (void)leaveConf:(HWMSDKCompleteHandler)callback;
```

参数描述

无。

返回值

无

示例代码

```
/// 离开会议  
[[HWMSdk getOpenApi] leaveConf:^(NSError * _Nullable error, id _Nullable result) {  
    [UIUtil showMessage:@"离开会议" error:error];  
}];
```

6.5.4.5 结束会议

endConf

接口描述

该接口用于结束会议。

注意事项

1. 不在会议中不应调用此接口。
2. 只有主持人可以结束会议。
3. 调用该接口，会议将结束，无法再次加入该会议。

方法定义

```
- (void)endConf:(HWMSDKCompleteHandler)callback;
```

参数描述

无。

返回值

无

示例代码

```
/// 结束会议  
[[HWMSdk getOpenApi] endConf:^(NSError * _Nullable error, id _Nullable result) {  
    [UIUtil showMessage:@"结束会议成功" error:error];  
}];
```

6.5.4.6 设置共享权限

setShareScreenPermission

接口描述

该接口用于设置共享权限。

注意事项

该接口返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回

方法定义

```
- (BOOL)setShareScreenPermission:(BOOL)isPermission;
```

参数描述

表 6-46 参数说明

参数	是否必须	类型	描述
isPermission	是	bool	是否允许。

返回值

无

示例代码

```
/// 设置共享权限
[[HWMsSdk getOpenApi] setShareScreenPermission:isPermission];
```

6.5.5 会议管理

6.5.5.1 预约会议

bookConf

接口描述

该接口用于预约会议。

注意事项

1. 会议开始时间晚于当前时间。
2. 个人会议或者云会议室的时候要传入vmrId。
3. 携带与会者入会的时候需要传入attendee。
4. 开启自动录制会议时必须开启录制会议。
5. callback中的result返回会议详情信息（[HWMConfDetail](#)）。

方法定义

```
- (void)bookConf:(HWMOrderConfParam *_Nonnull)param callback:(_Nonnull HWMsDKCompleteHandler)callback;
```

参数描述

表 6-47 HWMOrderConfParam 参数说明

参数	是否必须	类型	描述
confSubject	是	NSString *	会议主题
startTime	是	NSTimeInterval	会议开始时间，UTC时间戳，精度秒
duration	否	NSUInteger	会议持续时长，单位分钟，最小值15分钟，最大值1425分钟
confType	是	HWMOrderConfType	会议类型
isNeedConfPwd	否	BOOL	是否需要来宾密码 说明 仅对随机会议ID生效。
guestPwd	否	NSString *	来宾密码 说明 仅对随机会议ID生效。
isRecordOn	否	BOOL	录制会议开关，字段即将废弃，不建议使用。 说明 仅对云录制生效，客户端本地录制不生效。
isAutoRecord	否	BOOL	会议开始后是否自动开启录制，默认关闭。开启自动录制会议时必须开启录制会议 说明 仅对云录制生效，客户端本地录制不生效。
timeZone	否	NSUInteger	时区，默认值当前时区（中国默认时区56-东八区），详情参考 时区表 ，参数值1-73可用。
vmrId	否	NSString *	个人会议或者云会议室时传入
vmrIdType	否	HWMVmrIdType	VMR会议ID类型
callInRestrictionType	否	HWMJoinConfRestrictionType	允许入会范围限制
attendee	否	NSArray< HWMAttendeeInfo > *	与会者信息
isSmsOn	否	BOOL	发送短信通知开关，可选参数，企业配置里面需要开启，否则不生效

参数	是否必须	类型	描述
isMailOn	否	BOOL	发送邮件通知开发，可选参数，企业配置里面需要开启，否则不生效
isEmailCalendarOn	否	BOOL	发送邮件日历开关
allowGuestStartConf	否	BOOL	是否允许来宾提前启动会议。 说明 <ul style="list-style-type: none"> 仅针对随机会议ID的会议生效。 固定会议ID的会议，请使用修改云会议室信息接口设置。
allowGuestStartConfTime	否	NSInteger	允许来宾提前入会时间范围（单位：分钟）： 0 - 随时 x - 提前x分钟启动会议 说明 <ul style="list-style-type: none"> 仅针对随机会议ID的会议生效。 固定会议ID的会议，不支持设置提前入会时间。
isOpenWaitingRoom	否	BOOL	等候室开启开关 说明 需要等候室功能开通后才能生效

表 6-48 枚举 HWMOrderConfType 说明

参数	描述
OrderConfTypeAudio	音频会议
OrderConfTypeVideo	视频会议

表 6-49 枚举 HWMJoinConfRestrictionType 说明

参数	值	说明
HWMJoinConfRestrictionAll	0	所有用户
HWMJoinConfRestrictionCompany	2	企业内用户

参数	值	说明
HWMJoinConfRestrictionInvited	3	被邀请用户

表 6-50 HWMAttendeeInfo 参数说明

参数	是否必传	类型	描述
number	否	NSString *	号码（使用thirdUserId时，此参数可以不传入）
thirdUserId	否	NSString *	APPID 登录场景下的第三方帐号
name	否	NSString *	成员名称
email	否	NSString *	邮箱地址
sms	否	NSString *	手机电话
role	否	ConfRoleType	会议成员角色
accountId	否	NSString *	用户UUID（会议详情时字段返回登录帐号）
userUuid	否	NSString *	在编辑会议时这个值由会议详情上报，默认是空字符串
isAnonymous	否	BOOL	是否是匿名入会（仅用于获取与会者列表使用）
state	否	HWMConfAttendeeState	与会者状态（仅用于获取与会者列表使用）
orgId	否	NSString *	用于识别是否是本企业账号（仅用于获取与会者列表使用）

表 6-51 枚举 ConfRoleType 说明

参数	说明
ConfRoleTypeAttendee	普通与会者
ConfRoleTypeChairman	主席

表 6-52 枚举 HWMVmrlIdType 说明

参数	说明
HWMVmrlIdTypeFixed	固定类型
HWMVmrlIdTypeRandom	随机类型

示例代码

```

/// 预约会议
- (void)bookConf {
    HWMOrderConfParam * param = [[HWMOrderConfParam alloc] init];
    param.confSubject = @"会议名称";
    param.startTime = 1598759580;
    param.duration = 15;
    param.confType = HWMConfTypeVideo;
    param.isNeedConfPwd = YES;
    param.isRecordOn = NO;
    param.timeZone = 56;
    param.vmrlId = @"";
    param.callInRestrictionType = HWMCallRestrictionAll;
    [[HWMBizSdk getBizOpenApi] bookConf:param callback:^(NSError *_Nullable error, id _Nullable result) {
        if (error) {
            NSLog(@"预约会议失败:%@", error.localizedDescription);
        } else {
            NSLog(@"预约会议成功");
        }
    }];
}

```

6.5.5.2 编辑会议

editConf

接口描述

该接口用于编辑未开始的预约会议。

注意事项

1. 会议未开始前且自己是主席才可以调用接口。
2. 预约会议中携带的与会者在编辑会议时需要传入userUuid（会议详情中每个与会者带有该参数）。
3. 开启自动录制会议时必须开启录制会议。
4. 所有字段必填，编辑修改的字段可来源于用户输入，其他不想修改的属性通过会议详情参数获取。

方法定义

```

- (void)editConf:(HWMOrderConfParam *_Nonnull)param callback:(_Nonnull
HWMSDKCompleteHandler)callback;

```

参数描述

表 6-53 HWMOrderConfParam 参数说明，其余参数参见预约会议

参数	是否必须	类型	描述
confId	是	NSString *	会议ID。
isLiveOn	否	BOOL	是否开启直播。

示例代码

```

/// 编辑会议
- (void)editConf {
    HWMOrderConfParam * param = [[HWMOrderConfParam alloc] init];
    param.confSubject = @"会议名称";
    param.startTime = 1598759580;
    param.duration = 15;
    param.confType = HWMConfTypeVideo;
    param.isNeedConfPwd = YES;
    param.isRecordOn = NO;
    param.timeZone = 56;
    param.vmrId = @""; // vmr会议需要传入vmrId
    param.callInRestrictionType = HWMCallRestrictionAll;
    param.confId = @"986927771";
    [[HWMBizSdk getBizOpenApi] editConf:param callback:^(NSError *_Nullable error, id _Nullable result) {
        if (error) {
            NSLog(@"编辑会议失败:%@", error.localizedDescription);
        } else {
            NSLog(@"编辑会议成功");
        }
    }];
}

```

6.5.5.3 会议列表

getConfList

接口描述

该接口用于获取会议列表。

注意事项

1. 已登录状态，首次收到[会议列表通知](#)之后，调用才有效，否则返回为空的列表。

方法定义

```
- (NSArray<HWMBaseConfDetailParam *> *)getConfList;
```

参数描述

表 6-54 HWMBaseConfDetailParam 参数说明

参数	类型	描述
confId	NSString *	会议ID。
vmrConferenceId	NSString *	云会议室ID,显示时使用
confSubject	NSString *	会议主题

参数	类型	描述
chairmanPwd	NSString *	主席密码
generalPwd	NSString *	来宾密码
startTime	NSTimeInterval	会议开始时间, UTC时间戳, 精度秒
endTime	NSTimeInterval	会议结束时间, UTC时间戳, 精度秒
mediaType	ConfMediaType	媒体类型 (1和17是语音会议, 其它值是视频会议)
scheduserName	NSString *	预订者姓名
chairJoinUri	NSString *	主持人加入会议uri链接
guestJoinUri	NSString *	普通与会者加入会议uri链接
role	ConfRoleType	会议角色
accessNumber	NSString *	会议接入码

表 6-55 枚举 HWMCallInRestrictionType 说明

参数	值	描述
HWMCallRestrictionAll	0	所有用户
HWMCallRestrictionCompany	2	企业内用户
HWMCallRestrictionInvited	3	被邀请用户

示例代码

```
/// 会议列表
[[HWMBizSdk getBizOpenApi] getConfList];
```

6.5.5.4 会议详情

getConfDetail

接口描述

该接口用于获取会议详情。

注意事项

无

方法定义

- (void)getConfDetail:(NSString *)confId completionHandler:(HWM SDKConfInfoCompleteHandler)handler;

参数描述

HWMConfDetail继承于[HWMBaseConfDetailParam](#)

表 6-56 HWMConfDetail 参数说明，继承于 HWMBaseConfDetailParam

参数	类型	描述
recordType	NSUInteger	录制类型（2是录制会议）
scheduleVmr	BOOL	是否是VMR预约记录：0.普通会议记录；1.VMR预约记录
isGuestFreePwd	BOOL	是否来宾免密
isSendNotify	BOOL	是否发送邮件通知
isSendSms	BOOL	是否发送短信通知
isSendCalendarNotify	BOOL	是否发送日历通知
isAutoRecord	BOOL	会议开始后是否自动开启录制
vmrId	NSString *	用VMRId创建的会议的标识，编辑会议时用
callInRestriction	HWMCallInRestrictionType	呼入限制
vmrFlag	BOOL	是否是VMR会议
attendee	NSArray< HWMAttendeeInfo *> *	与会者列表
vmrIdType	HWMVmrIdType	VMR会议ID类型
allowGuestStartConf	BOOL	是否允许来宾提前启动会议
allowGuestStartConfTime	NSInteger	允许来宾提前入会时间范围（单位：分钟）： 0 - 随时 x - 提前x分钟启动会议
isOpenWaitingRoom	BOOL	等候室开启开关

示例代码

```

/// 获取会议详情
- (void)getConfDetail {
    [[HWMBizSdk getBizOpenApi] getConfDetail:@"986927771" completionHandler:^(NSError * _Nullable error, HWMConfDetail * _Nullable result) {
        if (error) {

```

```

        NSLog(@"%@", error.localizedDescription);
    } else {
        NSLog(@"获取会议详情成功");
    }
}
}];
}

```

6.5.5.5 取消会议

cancelConf

接口描述

该接口用于取消预约的会议。

注意事项

1. 只有主席可以取消会议
2. 会中调用会结束会议

方法定义

```
- (void)cancelConf:(NSString * _Nonnull)confId callback:(_Nonnull HWMSDKCompleteHandler)callback;
```

参数描述

表 6-57 参数说明

参数	是否必须	类型	描述
confId	是	NSString *	会议ID。

示例代码

```

/// 取消会议
- (void)cancelConf {
    [[HWMBizSdk getBizOpenApi] cancelConf:@"986927771" callback:^(NSError * _Nullable error, id
    _Nullable result) {
        if (error) {
            NSLog(@"%@",error.localizedDescription);
        } else {
            NSLog(@"取消会议成功");
        }
    }
}];
}

```

6.5.5.6 云会议室列表

getVmrList

接口描述

该接口用于获取云会议室列表。

注意事项

无

方法定义

```
- (NSArray<HWMVmrInfoModel *> *)getVmrList DEPRECATED_MSG_ATTRIBUTE("Use '- (void)getVmrList: (_Nonnull HWMSDKVmrListCompleteHandler)callback' instead");
- (void)getVmrList:(_Nonnull HWMSDKVmrListCompleteHandler)callback;
```

参数描述

表 6-58 HWMVmrInfoModel 参数说明

参数	类型	描述
name	NSString *	会议室名称
confId	NSString *	用于云会议室显示的ID
vmrId	NSString *	云会议室唯一ID
chairmanPwd	NSString *	主席密码
guestPwd	NSString *	来宾密码
type	HWMConfIdType	会议室类型
maxParties	NSUInteger	会议室最大方数
vmrStatus	BOOL	会议室状态，YES可用，NO不可用
confAllowJoinUser	HWMCallInRestrictionType	允许入会范围
isOpenWaitingRoom	BOOL	是否启用等候室

表 6-59 枚举 HWMConfIdType 说明

值	说明
HWMConfIdPersonal	个人会议ID
HWMConfIdPrivate	专用会议ID
HWMConfIdRandom	随机会议ID

表 6-60 枚举 HWMCallInRestrictionType 说明

值	说明
HWMCallRestrictionAll	所有用户
HWMCallRestrictionNotAnonymouse	非匿名用户
HWMCallRestrictionCompany	企业内用户
HWMCallRestrictionInvited	被邀请用户

示例代码

```

/// 获取云会议室列表
- (void)getVmrList {
    [[HWMBizSdk getBizOpenApi] getVmrList:^(NSError * _Nullable error, NSArray<HWMVmrInfoModel *> *
    _Nullable result) {
        if (error) {
            NSLog(@"获取成功");
        } else {
            NSLog(@"获取失败");
        }
    }
    ];
}
    
```

6.5.5.7 修改云会议室信息

changeVmrInfo

接口描述

该接口用于修改云会议室信息。

注意事项

1. 尽量只修改来宾密码

方法定义

```

- (void)changeVmrInfo:(HWMModifyVmrParam *)info callback:(_Nonnull
HWMSDKCompleteHandler)callback;
    
```

参数描述

表 6-61 HWMModifyVmrParam 参数说明

参数	是否必须	类型	描述
vmrId	是	NSString *	云会议室唯一ID
guestPwd	是	NSString *	来宾密码
chairPwd	是	NSString *	主席密码
vmrName	是	NSString *	云会议室名称
allowGuestStartConf	是	BOOL	允许来宾在主持人之前入会
confAllowJoinUser	否	HWMCallInRestrictionType	允许入会范围
isOpenWaitingRoom	否	BOOL	是否启用等候室

示例代码

```

/// 修改云会议信息
- (void)changeVmrInfo {
    HWMModifyVmrParam * vmrInfo = [[HWMModifyVmrParam alloc] init];
    vmrInfo.guestPwd = @"222222";
}
    
```

```

vmrInfo.chairPwd = @"123456789";
vmrInfo.vmrName = @"云会议室名称"
vmrInfo.vmrId = @"8a8d0d8a718d87d50171a9f6b8de0f40";
[[HWMBizSdk getBizOpenApi] changeVmrInfo:vmrInfo callback:^(NSError * _Nullable error, id _Nullable
result) {
    if (error) {
        NSLog(@"%@", error.localizedDescription);
    } else {
        NSLog(@"修改成功");
    }
}
}];
}

```

6.5.5.8 预约周期会议

bookCycleConf

接口描述

该接口用于预约周期会议。

注意事项

1. 不在登录状态下，不应调用该接口。
2. 会议开始时间必须晚于当前时间，会议时长应大于15分钟。
3. 个人会议或者云会议室的时候要传入vmrId。
4. 携带与会者入会的时候需要传入attendees。
5. 周期会议最大跨度一年，会议数量不超过50，超过则只预约前50个会议callback中的result返回会议详情信息（HWMConfDetail类型说明）

方法定义

```

- (void)bookCycleConf:(HWMOrderCycleConfParam *_Nonnull)param callback:(_Nonnull
HWMSDKCompleteHandler)callback

```

参数描述

表 6-62 HWMOrderCycleConfParam 参数说明

参数	是否必须	类型	描述
orderConfParam	是	HWMOrderConfParam	普通会议参数
cycleParam	是	HWMSDKCycleConfParamModel	周期会议参数

表 6-63 HWMSDKCycleConfParamModel 参数说明

参数	是否必须	类型	描述
startDate	是	NSUInteger	开始日期-时间戳，精度秒(0时区)
endDate	是	NSUInteger	结束日期-时间戳，精度秒(0时区)

参数	是否必须	类型	描述
cycleCount	是	NSUInteger	重复次数
cycleType	是	HWMCycleType	周期类型
interval	是	NSUInteger	周期区间。 1、周期类型选择了按天，表示每几天召开一次，取值范围[1,15]； 2、周期类型选择了Week，表示每几周召开一次，取值范围[1,5]； 3、周期类型选择了Month，interval表示隔几月，取值范围[1,3]
listPoints	是	NSString *	周期内的会议召开点。仅当按周和月时有效。数据格式(中间用逗号隔开):1,3,5,7 - 当按周时，取值范围[0,6]，0代表周日，1代表周一，以此类推； - 当按月时，取值范围[1,31]，若当月不含31号，则认为是当月最后一天
preRemindDays	是	NSUInteger	周期子会议提前通知天数。

表 6-64 枚举 HWMCycleType 说明

值	说明
HWMSDKCycleTypeDay	以天为周期
HWMSDKCycleTypeWeek	以周为周期
HWMSDKCycleTypeMonth	以月为周期

示例代码

```

/// 预约周期会议
- (void)bookCycleConf{
    HWMOrderCycleConfParam * cycleConfParam = [[HWMOrderCycleConfParam alloc] init];
    cycleConfParam.orderConfParam.confSubject = @"我的预约会议";
    cycleConfParam.orderConfParam.confType = OrderConfTypeVideo;
    cycleConfParam.orderConfParam.isNeedConfPwd = YES;
    cycleConfParam.orderConfParam.isAutoRecord = YES;
    cycleConfParam.orderConfParam.isRecordOn = YES;
}

```

```

cycleConfParam.orderConfParam.startTime = 1633017600; //utc时间戳，单位秒，如果获取的时间是本地时间，需要转换成utc时间
cycleConfParam.orderConfParam.duration = 30; //会议持续时长(分钟)
cycleConfParam.orderConfParam.timeZone = 56;
cycleConfParam.orderConfParam.callInRestrictionType = HWMJoinConfRestrictionAll;
HWMAttendeeInfo * attendeeInfo = [[HWMAttendeeInfo alloc] init];
attendeeInfo.name = @"Mike";
attendeeInfo.number = @"+991116003543";
attendeeInfo.isMute = YES;
cycleConfParam.orderConfParam.attendee = @[attendeeInfo];

HWMSDKCycleConfParamModel * cycleConfParamModel = [[HWMSDKCycleConfParamModel alloc] init];
cycleConfParamModel.startDate = 1633017600; //utc时间戳，单位秒，如果获取的时间是本地时间，需要转换成utc时间;
cycleConfParamModel.endDate = 1636560000;
cycleConfParamModel.cycleType = HWMSDKCycleTypeWeek;
cycleConfParamModel.preRemindDays = 1;
cycleConfParam.cycleParam = cycleConfParamModel;

[[HWMBizSdk getBizOpenApi] bookCycleConf:cycleConfParam callback:^(NSError * _Nullable error, id _Nullable result) {
    if (!error) {
        NSLog(@"book cycle conf success");
        [UIUtil showMessage:@"预约周期会议会议成功"];
    }else{
        NSLog(@"book cycle conf fail errorCode : %zd",error.code);
        [UIUtil showMessage:[NSString stringWithFormat:@"预约周期会议失败:%zd %@", error.code, error.localizedDescription]];
    }
}];
}

```

6.5.5.9 编辑周期会议

editCycleConf

接口描述

该接口用于编辑周期会议。

注意事项

1. 该接口在已登录状态下才能调用。
2. 所有字段必填，编辑修改的字段可来源于用户输入，其他不想修改的属性一定要取会议详情对应字段来赋值。

方法定义

```

- (void)editCycleConf:(HWMOrderCycleConfParam * _Nonnull)param callback:(_Nonnull HWMSDKCompleteHandler)callback

```

参数描述

表 6-65 HWMOrderCycleConfParam 参数说明

参数	是否必须	类型	描述
orderConfParam	是	HWMOrderConfParam	普通会议参数
cycleParam	是	HWMSDKCycleConfParamModel	周期会议参数

示例代码

```

/// 编辑周期会议
- (void)editCycleConf{
    HWMOrderCycleConfParam * cycleConfParam = [[HWMOrderCycleConfParam alloc] init];
    cycleConfParam.orderConfParam.confSubject = @"我的预约会议";
    cycleConfParam.orderConfParam.confType = OrderConfTypeVideo;
    cycleConfParam.orderConfParam.isNeedConfPwd = YES;
    cycleConfParam.orderConfParam.isAutoRecord = YES;
    cycleConfParam.orderConfParam.isRecordOn = YES;
    cycleConfParam.orderConfParam.startTime = 1633017600; //utc时间戳，单位秒，如果获取的时间是本地时间，需要转换成utc时间
    cycleConfParam.orderConfParam.duration = 30; //会议持续时长(分钟)
    cycleConfParam.orderConfParam.timeZone = 56;
    cycleConfParam.orderConfParam.callInRestrictionType = HWMJoinConfRestrictionAll;
    HWMAttendeeInfo * attendeeInfo = [[HWMAttendeeInfo alloc] init];
    attendeeInfo.name = @"Mike";
    attendeeInfo.number = @"+991116003543";
    attendeeInfo.isMute = YES;
    cycleConfParam.orderConfParam.attendee = @[attendeeInfo];

    HWMSDKCycleConfParamModel * cycleConfParamModel = [[HWMSDKCycleConfParamModel alloc] init];
    cycleConfParamModel.startDate = 1633017600; //utc时间戳，单位秒，如果获取的时间是本地时间，需要转换成utc时间;
    cycleConfParamModel.endDate = 1636560000;
    cycleConfParamModel.cycleType = HWMSDKCycleTypeWeek;
    cycleConfParamModel.preRemindDays = 1;
    cycleConfParam.cycleParam = cycleConfParamModel;

    [[HWMBizSdk getBizOpenApi] editCycleConf:cycleConfParam callback:^(NSError * _Nullable error, id _Nullable result) {
        if (!error) {
            NSLog(@"book cycle conf success");
            [UIUtil showMessage:@"编辑周期会议会议成功"];
        }else{
            NSLog(@"book cycle conf fail errorCode : %zd",error.code);
            [UIUtil showMessage:[NSString stringWithFormat:@"编辑周期会议失败:%zd %@", error.code, error.localizedDescription]];
        }
    }];
}

```

6.5.5.10 取消周期会议

cancelCycleConfWithParam

接口描述

该接口用于取消至少还有一场子会议未开始的周期会议。

注意事项

1. 该接口在登录状态下才能调用。
2. 最后一场子会议未开始前可以调用接口，开始后调用则无效。

方法定义

```

- (void)cancelCycleConfWithParam:(HWMSDKCancelCycleConfParamModel * _Nonnull)param callback:
(_Nonnull HWMSDKCompleteHandler)callback;

```

参数描述

表 6-66 HWMSDKCancelCycleConfParamModel 参数说明

参数	是否必须	类型	描述
confId	是	NSString *	会议id
endOnlineConf	是	BOOL	是否需要结束在线会议

示例代码

```

/// 取消周期会议
- (void)cancelCycleConf {
    HWMSDKCancelCycleConfParamModel *cancelParam = [[HWMSDKCancelCycleConfParamModel alloc]
init];
    cancelParam.confId = "989156631";
    cancelParam.endOnlineConf = YES;
    [[HWMBizSdk getBizOpenApi] cancelCycleConfWithParam:cancelParam callback:^(NSError * _Nullable
error, id _Nullable result) {
        NSString *tips = !error ? @"取消周期会议成功" : @"取消周期会议失败";
        [UIUtil showMessage:tips error:error];
    }];
}

```

6.5.5.11 编辑周期子会议

editSubCycleConf

接口描述

该接口用于编辑周期子会议。

注意事项

1. 该接口在已登录状态下才能调用。

方法定义

```

- (void)editSubCycleConf:(HWMSDKModifySubCycleConfParamModel *_Nonnull)param callback:(_Nonnull
HWMSDKCompleteHandler)callback;

```

参数描述

表 6-67 HWMSDKModifySubCycleConfParamModel 参数说明

参数	是否必须	类型	描述
confId	是	NSString *	会议id
subConfID	是	NSString *	子会议唯一标识UUID
confMediaType	否	HWMConf MediaType	会议媒体类型
startTime	否	NSInteger	会议开始时间（UTC时间），单位为秒
confLen	否	NSInteger	会议持续时长，单位分钟

参数	是否必须	类型	描述
isAutoRecord	否	BOOL	会议开始后是否自动开启录制
recordAuthType	否	NSUInteger	录播鉴权方式 0:可通过连接观看/下载 (不拼接nonce) 1:企业用户可观看/下载 2:与会者可观看/下载
callInRestriction	否	HWMConfAllowJoinUserType	呼入限制
allowGuestStartConf	否	BOOL	允许来宾启动会议
allowGuestStartConfTime	否	NSInteger	允许来宾提前入会时间范围(单位:分钟) 0-随时, n-提前n分钟启动会议

表 6-68 枚举 HWMConfAllowJoinUserType 说明

值	描述
HWMConfAllowJoinUserTypeAnyone	任何人
HWMConfAllowJoinUserTypeLoggedInUser	已登录用户
HWMConfAllowJoinUserTypeInCompanyUser	企业内用户
HWMConfAllowJoinUserTypeInvitedUser	被邀请用户

示例代码

```

/// 编辑周期会议
- (void)editSubCycleConf{
    HWMSDKModifySubCycleConfParamModel * modifySubCycleConfParamModel =
[[HWMSDKModifySubCycleConfParamModel alloc] init];
    modifySubCycleConfParamModel.confId = confId;
    modifySubCycleConfParamModel.subConfId = subConfId;
    modifySubCycleConfParamModel.confMediaType = HWMConfMediaTypeVideo;
    modifySubCycleConfParamModel.startTime = 1633017600;
    modifySubCycleConfParamModel.confLen = 60;
    modifySubCycleConfParamModel.isAutoRecord = NO;
    modifySubCycleConfParamModel.callInRestriction = HWMConfAllowJoinUserTypeAnyone;
    modifySubCycleConfParamModel.allowGuestStartConf = NO;
    modifySubCycleConfParamModel.allowGuestStartConfTime = 0;
    [[HWMBizSdk getBizOpenApi] editSubCycleConf:modifySubCycleConfParamModel callback:^(NSError *
_Nullable error, id _Nullable result) {
        if (!error) {
            [UIUtil showMessage:@"编辑周期子会议会议成功"];
        }else{
            [UIUtil showMessage:[NSString stringWithFormat:@"编辑周期子会议失败:%zd %@", error.code,
error.localizedDescription]];
        }
    }
    }];
}

```

6.5.5.12 取消周期子会议

cancelSubCycleWithParam:

接口描述

该接口用于取消未开始的周期子会议。

注意事项

1. 该接口在登录状态下才能调用。
2. 子会议未开始前可以调用接口，开始后调用则无效。

方法定义

```
-(void)cancelSubCycleWithParam:(HWMSDKCancelSubCycleConfParamModel *_Nonnull)param callback:
(_Nonnull HWMSDKCompleteHandler)callback
```

参数描述

表 6-69 HWMSDKCancelSubCycleConfParamModel 参数说明

参数	是否必须	类型	描述
confId	是	NSString *	会议id
subConfID	是	NSString *	子会议id
endOnlineConf	是	BOOL	是否需要结束在线会议

示例代码

```
/// 取消周期子会议
- (void)confirmCancelSubConf {
    HWMSDKCancelSubCycleConfParamModel *cancelParma = [[HWMSDKCancelSubCycleConfParamModel
alloc] initWithConfId:confId subConfID:subConfID endOnlineConf:YES];
    [[HWMBizSdk getBizOpenApi] cancelSubCycleWithParam:cancelParma callback:^(NSError *_Nullable
error, id _Nullable result) {
        NSString *tips = !error ? @"取消会议成功" : @"取消会议失败";
        [UIUtil showMessage:tips error:error];
    }];
}
```

6.5.6 获取会议信息

6.5.6.1 获取与会者列表

getAttendeeList

接口描述

该接口用于获取会议中与会者列表。

注意事项

1. 只能在会议中使用

方法定义

```
-(NSArray <HWMAttendeeInfo *> *)getAttendeeList;
```

参数描述

[HWMAttendeeInfo](#) 参数说明

表 6-70 枚举 HWMConfAttendeeState 说明

参数	描述
HWMConfAttendeeStateInConf	会议中
HWMConfAttendeeStateCalling	正在呼叫
HWMConfAttendeeStateLeaved	已经离开

示例代码

```
/// 获取与会者列表
[[HWMBizSdk getBizOpenApi] getAttendeeList];
```

6.5.6.2 获取观众列表

getAudienceList

接口描述

该接口用于获取会议中观众列表。

注意事项

1. 只能在会议中使用
2. 观众身份（可以通过会议详情消息推送获取）不能使用此接口

方法定义

```
-(NSArray <HWMAttendeeInfo *> *)getAudienceList;
```

参数描述

[HWMAttendeeInfo](#) 参数说明

示例代码

```
/// 获取观众列表
[[HWMBizSdk getBizOpenApi] getAudienceList];
```

6.5.6.3 判断自己是否是主持人

selfIsChairMan

接口描述

该接口用于判断自己当前是否是主持人。

注意事项

1. 只有是主席的时候才返回YES，没有进去会议时也会返回NO；

方法定义

```
- (BOOL)selfIsChairMan;
```

参数描述

无

示例代码

```
/// 判断自己是否是主持人  
[[HWMBizSdk getBizOpenApi] selfIsChairMan];
```

6.5.6.4 获取用户状态

getIsInConfOrCall:

接口描述

该接口用于获取用户当前的状态(是否在呼叫/会议中)。

注意事项

无

方法定义

```
- (HWMSDKConfOrCallStatus)getIsInConfOrCall;
```

参数描述

表 6-71 参数说明

枚举值	描述
HWMSDKConfOrCallStatusIdle	空闲
HWMSDKConfOrCallStatusInConf	会议中
HWMSDKConfOrCallStatusInCall	通话中

返回值

无

示例代码

```
[[HWMBizSdk getBizOpenApi] getIsInConfOrCall];
```


6.6 通知参考

注意事项，使用前必看

1. 需要使用通知方法时，必须注册代理对象。
两种方式：
 - a. 初始化时，对HWMOpenSDKConfig对象的属性赋值

```
HWMOpenSDKConfig *config = [[HWMOpenSDKConfig alloc] init];  
//示例根据需求使用对应代理  
config.globalHandler = [NotifyMessageHandler sharedInstance];
```
 - b. 使用前使用[HWMSdk getSdkConfig]获取初始化对象给属性赋值

```
//示例根据需求使用对应代理  
[HWMSdk getSdkConfig].globalHandler = [NotifyMessageHandler sharedInstance];
```
2. 代理方法必须写在代理对象中
3. 使用代理时保证代理对象存在

6.6.1 注意事项，使用通知前请查看

1. 需要使用通知方法时，必须注册代理对象。
两种方式：
 - a. 初始化时，对HWMOpenSDKConfig对象的属性赋值

```
HWMOpenSDKConfig *config = [[HWMOpenSDKConfig alloc] init];  
//示例根据需求使用对应代理  
config.globalHandler = [NotifyMessageHandler sharedInstance];
```
 - b. 使用前使用[HWMSdk getSdkConfig]获取初始化对象给属性赋值

```
//示例根据需求使用对应代理  
[HWMSdk getSdkConfig].globalHandler = [NotifyMessageHandler sharedInstance];
```
2. 代理方法必须写在代理对象中
3. 使用代理时保证代理对象存在

6.6.2 帐号状态变更通知

6.6.2.1 帐号被踢出

通知描述

收到该通知时，表示当前使用的帐号被其他终端踢出，退出登录状态，需要重新登录。

方法定义

```
-(void)onKickedOut:(NSString *)result;
```

注意事项

1. SDK初始化时设置globalHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 6-72 参数说明

参数	类型	描述
result	NSString *	用户uuid

6.6.3 会议信息变更通知

6.6.3.1 会议状态

通知描述

收到该通知时，表示会议状态发生变更。

方法定义

```
-(void)onConfStatusChanged:(HWMSDKConfStatus)status;
```

注意事项

1. SDK初始化时设置globalHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 6-73 参数说明

参数	类型	描述
status	HWMSDKConfStatus	会议通话状态。

表 6-74 枚举 HWMSDKConfStatus 说明

枚举值	描述
HWMSDKConfStatusIdle	空闲状态（会议结束）
HWMSDKConfStatusIncoming	正在来电
HWMSDKConfStatusCallingOut	正在呼出
HWMSDKConfStatusConnected	已接通
HWMSDKConfStatusRecalling	正在重呼

6.6.3.2 被邀加入会议来电通知

通知描述

收到该通知时，被邀请加入会议。

方法定义

```
-(void)onConflncomingNotify:(HWMSDKConflncomingInfoModel *)conflncomingInfo;
```

注意事项

1. SDK初始化时设置globalHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 6-75 参数说明

参数	类型	描述
conflncomingInfo	HWMSDKConflncomingInfoModel	会议详情

表 6-76 HWMSDKConflncomingInfoModel 参数说明

参数	类型	描述
subject	NSString *	会议标题
conflid	NSString *	会议ID
confMediaType	HWMConfMediaType	媒体类型

表 6-77 枚举 HWMConfMediaType 说明

枚举	值	说明
HWMConfMediaTypeAudio	0	音频会议
HWMConfMediaTypeVideo	1	视频会议

6.6.3.3 会议详情通知

通知描述

收到该通知时，会中时会收到会议详情通知，可用于做会中分享等。

方法定义

```
- (void)onConfDetailNotify:(HWMSConfDetail *)confDetail;
```

注意事项

1. SDK初始化时设置globalHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 6-78 参数说明

参数	类型	描述
confDetail	HWMSConfDetail	会议详情

6.6.3.4 会议结束通知

通知描述

收到该通知时，表示当前会议已结束。

方法定义

```
- (void)onConfEndedNotify:(HWMSDKERR)confEndCode;
```

注意事项

1. SDK初始化时设置globalHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 6-79 参数说明

参数	类型	描述
confEndCode	HWMSDKERR	SDK错误信息枚举（具体参照HWMSDKCommonDefine.h文件）

6.6.4 通话状态变更通知

6.6.4.1 通话状态

通知描述

收到该通知时，表示通话状态发生变更。

方法定义

```
-(void)onCallStatusChanged:(HWMSDKCallStatus)status;
```

注意事项

1. SDK初始化时设置globalHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 6-80 参数说明

参数	类型	描述
status	HWMSDKCallStatus	会议通话状态。

表 6-81 枚举 HWMSDKCallStatus 说明

枚举值	描述
HWMSDKCallStatusIdle	空闲状态 (通话结束)
HWMSDKCallStatusIncoming	正在来电
HWMSDKCallStatusCallingOut	正在呼出
HWMSDKCallStatusConnected	已接通

6.6.4.2 通话结束

通知描述

收到该通知时，表示通话结束，返回通话信息。

方法定义

```
-(void)onCallEnded:(HWMCallRecordInfo *)callInfo;
```

注意事项

1. SDK初始化时设置globalHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 6-82 参数说明

参数	类型	描述
callInfo	HWMCallRecordInfo *	结束通话的记录信息。

表 6-83 HWMCallRecordInfo 说明

参数	类型	描述
myInfo	HWMPeerInfo *	本端信息。
peerInfo	HWMPeerInfo *	对端信息。
startTime	NSTimeInterval	开始时间。
endTime	NSTimeInterval	结束时间。
callMediaType	HWMCallMediaType	呼叫类型。
callDirectionType	HWMCallDirectionType	通话发起方向(来电或者呼出)。
callResult	HWMCallEndResult	结束通过方式。
isHangUpFirst	BOOL	主动挂断是YES，其他是NO。
isPeerBusy	BOOL	对方忙。

表 6-84 HWMPeerInfo 说明

参数	类型	描述
peerName	NSString *	名字
peerNumber	NSString *	号码。分配给帐号的sip号码（如 +99111244216210249）
peerContactsId	NSString *	用户Uuid
account	NSString *	账号
thirdUserId	NSString *	APPID 登录场景下的第三方账号

表 6-85 枚举 HWMCallMediaType 说明

枚举值	描述
HWMCallMediaTypeAudio	音频呼叫。
HWMCallMediaTypeVideo	视频呼叫。

表 6-86 枚举 HWMCallDirectionType 说明

枚举值	描述
HWMCallDirectionTypeComing	来电。
HWMCallDirectionTypeOut	呼出。

表 6-87 枚举 HWMCallEndResult 说明

枚举值	描述
HWMCallEndResultFailed	通话失败。
HWMCallEndResultConnectedCallerHangUp	主叫挂断（接通）。
HWMCallEndResultConnectedCalleeHangUp	被叫挂断（接通）。
HWMCallEndResultCallerHangUp	主叫挂断（未接通）。
HWMCallEndResultCalleeHangUp	被叫挂断（未接通）。
HWMCallEndResultCalleeBusy	被叫正忙（未接通）。
HWMCallEndResultCalleeAnswerTimeOut	被叫接听超时（未接通）。

6.6.5 业务消息通知

6.6.5.1 主持人角色变化

通知描述

收到该通知时，表示个人角色发生变化，是否是主持人身份。

方法定义

- (void)onSelfIsChairManChanged:(BOOL)isChairMan;

注意事项

1. SDK初始化时设置bizNotifyHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 6-88 参数说明

参数	类型	描述
isChairMan	BOOL	是否是主持人

6.6.5.2 会议列表变化

通知描述

收到该通知时，表示会议列表发生变化。

方法定义

- (void)onReceiveConfList:(NSArray<HWMBaseConfDetailParam *>)confList;

注意事项

1. SDK初始化时设置bizNotifyHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

参数说明参见[会议列表参数描述](#)

6.6.5.3 视频会议录播权限更新

通知描述

收到该通知时，表示视频会议录播权限更新，是否可以录播。

方法定义

- (void)onRecordPermissionNotify:(BOOL)isHasPermission;

注意事项

1. SDK初始化时设置bizNotifyHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 6-89 参数说明

参数	类型	描述
isHasPermission	BOOL	是否可以录播

6.6.5.4 预约会议发送短信权限更新

通知描述

收到该通知时，表示预约会议发送短信权限更新，是否可以发送短信。

方法定义

```
-(void)onBookConfSmsPermissionNotify:(BOOL)isHasPermission;
```

注意事项

1. SDK初始化时设置bizNotifyHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 6-90 参数说明

参数	类型	描述
isHasPermission	BOOL	是否允许发送短信

6.6.6 来电接听策略设置

6.6.6.1 设置点对点来电接听策略

通知描述

点对点来电时会收到该通知，需要接入者返回来电接听策略。若不实现，默认使用原始体验弹出来电接听框。

方法定义

```
-(HWMInComingAnswerType)answerTypeOnCallIncoming;
```

示例代码

```
-(HWMInComingAnswerType)answerTypeOnCallIncoming {
    return HWMInComingAnswerNormal;// 默认
}
```

注意事项

1. SDK初始化时设置confSettingHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 6-91 HWMInComingAnswerType 枚举说明

参数	描述
HWMInComingAnswerNormal	默认（弹出来电接听界面）
HWMInComingAnswerAutoDecline	自动拒接
HWMInComingAnswerAutoAnswer	自动接听

6.6.6.2 设置会议来电接听策略

通知描述

会议来电时会收到该通知，需要接入者返回来电接听策略。若不实现，默认使用原始体验，弹出会议来电接听框。

方法定义

```
-(HWMInComingAnswerType)answerTypeOnConfIncoming;
```

示例代码

```
-(HWMInComingAnswerType)answerTypeOnConfIncoming {
    return HWMInComingAnswerNormal;// 默认
}
```

注意事项

1. SDK初始化时设置confSettingHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 6-92 HWMInComingAnswerType 枚举说明

参数	描述
HWMInComingAnswerNormal	默认
HWMInComingAnswerAutoDecline	自动拒接

参数	描述
HWMIncomingAnswerAutoAnswer	自动接听

6.7 错误码参考

6.7.1 常见错误码

注意事项

1. 如果使用建议使用枚举。

表 6-93 枚举 HWMUISDKErrorCode 说明

变量值	值	描述
HWMUISDKErrorCodeNumberAndAccountEmpty	40001	联系人不存在
HWMUISDKErrorCodeArgsError	40002	参数错误
HWMUISDKErrorCodeRepeatRequest	40003	重复调用方法
HWMUISDKErrorCodeTimeout	40004	调用方法超时
HWMUISDKErrorCodeUnInit	40005	sdk 未初始化
HWMUISDKErrorCodeArgsLengthOverrun	40006	参数长度超限
HWMUISDKErrorCodePasswordOrAccountEmpty	40007	账号或密码不能都为空
HWMUISDKErrorCodeConfIdEmpty	40008	会议id不能为空
HWMUISDKErrorCodeUnLogin	40009	未登录
HWMUISDKErrorCodeLoginIn	40010	登录中
HWMUISDKErrorCodeSubjectTooLong	40011	会议主题长度不能超过50
HWMUISDKErrorCodeConfIdTooLong	40012	会议ID长度不能超过20

变量值	值	描述
HWMUISDKErrorCodeConfOrCallExit	40013	您正处在会议或者呼叫中
HWMUISDKErrorCodeAPPIDInvalid	40014	APPID登录参数错误
HWMUISDKErrorCodeNetworkUnreachable	40015	网络不可用，请检查网络设置
HWMUISDKErrorCodeFilePathEmpty	40016	上传头像时图片路径不能为空
HWMUISDKErrorCodeNotInMeeting	40017	没有在会议中
HWMUISDKErrorCodeNoPermissionAddAttendee	40018	没有权限添加与会者
HWMUISDKErrorCodeNoPermissionEndConf	40019	没有权限结束会议
HWMUISDKErrorCodeLeaveConfFail	40020	离开会议失败
HWMUISDKErrorCodeLoggedIn	40021	已登录

表 6-94 枚举 HWMErrrorCode 说明

变量值	值	描述
HWMErrrorCodePermissionReject	50000	没有摄像头和麦克风权限
HWMErrrorCodeUserCancel	50001	用户取消输入会议密码
HWMErrrorCodeMultiDeviceOnline	50002	多终端登录
HWMErrrorCodeSipDisable	50003	呼叫不可用
HWMErrrorCodeConfIdError	50004	会议ID不可为空
HWMErrrorCodeNickEmpty	50005	昵称不可为空
HWMErrrorCodeNickTooLong	50006	昵称不能超过64字符
HWMErrrorCodeConfOrCallExist	50007	已经存在一路会议/呼叫
HWMErrrorCodeConfNeedPassword	50008	匿名会议需要密码
HWMErrrorCodeConfUnLogin	50009	未登录

变量值	值	描述
HWMErrorCodeConfLoginIng	50010	登录中
HWMErrorCodeQRCodeNotSupported	50011	不支持的二维码（只支持华为会议二维码）
HWMErrorCodeNetworkUnReachable	50012	网络异常
HWMErrorCodeUserCancelPhoneVerify	50013	用户取消匿名入会手机验证
HWMErrorCodeSeesionExpired	50014	匿名入会时会话过期

表 6-95 枚举 HWMConfSdkErrorCode 说明

变量值	值	描述
HWMConfSdkErrorCodeOperationTimeout	68	网络不佳，发起/加入会议超时
HWMConfSdkErrorCodeRoleAuthenticationFailed	91	会议不存在或者无权限访问
HWMConfSdkErrorCodeVMROnlyAllowChairStartConf	95	会议不允许提前入会，请联系主持人
HWMConfSdkErrorCodeNotAllowedGuestFirstJoin	103	会议不允许提前入会，请联系主持人
HWMConfSdkErrorCodeConfSizeOverCommunalVMRMaxNum	104	云会议室与会人数已达上限，请联系管理员
HWMConfSdkErrorCodeConfClosedRecently	112	会议已结束
HWMConfSdkErrorCodeReasonSessionTimerTimeout	836	网络中断，呼叫已断开
HWMConfSdkErrorCodeReasonNoStreamHangup	837	呼叫已断开，无码流挂断
HWMConfSdkErrorCodeReasonChairHangup	838	您已被主持人移出会议（主席挂断）
HWMConfSdkErrorCodeReasonStopConfHangup	839	结束会议挂断
HWMConfSdkErrorCodeReasonConfLocked	842	会议已经被锁定
HWMConfSdkErrorCodeVMRDataNotFound	11070006	VMR会议室不存在

变量值	值	描述
HWMConfSdkErrorCodeVMRConfConflict	11071005	会议时间与下一个预定的会议冲突，如需继续请联系云会议室的管理员预约
HWMConfSdkErrorCodeConfStartTimeOverMaxScheduleTime	11071014	会议开始时长超出最大可预定时间范围
HWMConfSdkErrorCodeConfSMSIllegal	11071025	请检查与会人手机号是否正确
HWMConfSdkErrorCodeOrgConcurrentParticipantsNumbersIsZero	11071060	企业未购买并发与会方资源，无法创建会议
HWMConfSdkErrorCodeConfCancelFailAsConfStarted	11071067	当前会议正在召开，无法取消
HWMConfSdkErrorCodeConfNotAllowedAnonymous	11071073	您呼叫的会议不允许匿名用户呼入，请联系会议预订者
HWMConfSdkErrorCodeConfOnlyAllowUserInOrg	11071074	您呼叫的会议只允许企业内用户呼入，请联系会议预订者
HWMConfSdkErrorCodeConfOnlyAllowUserInvited	11071075	您呼叫的会议只允许被邀请用户呼入，请联系会议预订者
HWMConfSdkErrorCodeSiteCallNumbersIsEmpty	11072031	请检查与会人手机号是否为空
HWMConfSdkErrorCodeSiteCallNumbersIllegalUri	11072033	请检查与会人入会号码是否正确
HWMConfSdkErrorCodeConfLocked	11072050	会议已经被锁定
HWMConfSdkErrorCodeParticipantAuthenticationFailed	11081006	密码错误，请重新输入
HWMConfSdkErrorCodeCorpConferenceResourceHasBeenRunout	15022001	企业购买的会议资源（人数限制）已达上限，请联系管理员

6.7.2 错误码表

错误码表

表 6-96 错误码说明

错误码	错误描述
-1	Unknown error.:未知错误
0	Success.:成功

错误码	错误描述
111070001	Request error.:请求错误
111070002	Insufficient permissions.:权限不足
111070003	System processing failure.:系统处理失败
111070004	Meeting authentication failed.:校验失败
111070005	Meeting does not exist.:会议不存在
111070006	Personal cloud meeting room does not exist.:个人云会议室不存在
111070007	Meeting access number does not exist.:会议接入号不存在
111070008	Meeting notification could not be sent.:会议通知模板配置错误
111070009	Meeting password cannot be left blank.:会议密码不能为空
111070010	Insufficient permissions to view this meeting.:权限不足,无法查看会议
111070100	Get started meeting exception.:获取召开会议异常
111070101	Start meeting exception.:开始会议异常
111070102	Apply IVR resource exception.:申请IVR资源异常
111070103	Start IVR meeting exception.:开始IVR会议异常
111070104	Apply site resource exception.:申请会场资源异常
111070105	Start sub meeting exception.:开始子会议异常
111070106	Add channel exception.:创建级联通道异常
111070107	Move and apply site resource exception.:移动会场申请资源异常
111070108	Start meeting exception.:开启会议异常
111070109	Get started cloud meeting exception.:获取召开云会议室会议异常
111070110	Invite site exception.:邀请会场异常
111070111	Request token is null.:请求token为空
111070112	Rename site exception.:重命名会场异常
111070113	Hangup site exception.:会场挂断异常
111070114	Mass hangup site exception.:会场批量挂断异常
111070115	Delete attendee exception.:删除与会者异常
111070116	Mass delete attendee exception.:批量删除与会者异常

错误码	错误描述
111070117	Book cycle meeting exception.:预定周期会议异常
111070118	Modify cycle meeting exception.:修改周期会议异常
111070119	Create cycle meeting exception.:创建周期会议异常
111070120	Region data not found.:区域信息获取失败
111070121	Forbidden subscribe.:禁止订阅
111070122	Cancel subscribe fail.:取消订阅失败
111071001	Sysportal service exception.:Sysportal服务运行异常
111071002	Conf service exception.:Conf服务运行异常
111071003	Failed to create the meeting. Try again later.:创建会议失败, 请稍后重试
111071004	This cloud meeting room does not allow guests to join before the host.:VMR会议不允许来宾在主席入会前入会
111071005	This cloud meeting room has been scheduled for another meeting during this period.:VMR会议时间冲突, 该时间段此会议室已有预约会议
111071006	Insufficient cloud meeting time.:VMR会议时长不足
111071007	Meeting does not exist.:VMR预约关系不存在
111071008	Cannot query meeting list.:查询会议列表失败
111071009	Cannot change cloud meeting into immediate meeting.:不支持修改VMR会议为立即会议
111071010	Cannot modify cloud meeting flag.:不支持修改云会议室标识
111071011	The cloud meeting has not yet started. Please wait.:VMR会议未到准备时间点
111071012	Not cloud meeting.:非VMR会议
111071013	Start time less than current time.:会议开始时间不能早于系统当前时间
111071014	Start time over max schedule time.:会议开始时间超出最大预订时间范围
111071015	Duration over range.:时长超出范围
111071016	Cycle ID list null exception.:取消周期会议ID列表不能为空
111071017	Meeting size over range.:与会方数超出范围
111071018	Invalid media type.:媒体类型非法
111071019	Invalid meeting type.:会议类型非法

错误码	错误描述
111071020	Cycle params null exception.:周期会议时,必须携带周期会议参数
111071021	Duration over range.:会议时长超出范围
111071022	Subject length too long.:会议主题超长
111071023	Group URI length too long.:groupuri超长
111071024	Invalid email address format.:邮箱地址不合法
111071025	Invalid mobile number format.:手机号不合法
111071026	Department name too long.:部门名称过长
111071027	Invalid welcome voice enable value.:入会欢迎词开关不在范围内
111071028	Invalid enter voice enable value.:入会提示音开关不在范围内
111071029	Invalid leave voice enable value.:离会提示音开关不在范围内
111071030	Invalid reminders value.:会议提醒非法
111071031	Invalid allow record value.:会议是否启用网络录制取值非法
111071032	Invalid auto record value.:会议是否自动启动录制取值非法
111071033	Invalid user type value.:会预定者的用户类型非法
111071034	Invalid encrypt mode value.:加密类型非法
111071035	Invalid language value.:语言非法
111071036	Invalid auto prolong value.:是否自动延长会议取值非法
111071037	Vmrid too long.:vmrid超长
111071038	Invalid record type value.:录播类型非法
111071039	Live address too long.:主流直播地址超长
111071040	Aux address too long.:辅流直播地址超长
111071041	Null start date for cycle meeting.:周期会议时,起始日期为空
111071042	Null end date for cycle meeting.:周期会议时,结束日期为空
111071043	Invalid cycle type value for cycle meeting.:周期会议时,周期类型非法
111071044	Invalid interval for cycle meeting.:周期会议时,interval非法
111071045	Null point for cycle meeting.:周期会议时,point为空
111071046	Invalid point value for cycle meeting.:周期会议时,point非法
111071047	Invalid vmr flag value.:vmrFlag非法

错误码	错误描述
111071048	Invalid record auxStream.:recordAuxStream非法
111071049	Invalid multiStream flag.:multiStreamFlag非法
111071050	Invalid audience flag.:audienceFlag非法
111071051	Invalid liveChannel ID.:liveChannelID不存在
111071052	Anonymous login is locked. Try again later.:匿名登录锁定,请稍后重试
111071053	Link login is locked. Try again later.:链接登录锁定,请稍后重试
111071054	Not enough Meeting ID.:会议ID不足
111071055	Failed to apply meeting ID.:会议ID申请失败
111071056	Failed to release meeting ID.:会议ID释放失败
111071057	Live room is used.:直播房间已被占用
111071058	The meeting has not yet started. Please wait.:当前会议还未召开,请耐心等待
111071059	Meeting size over range.:与会方数超出上限
111071060	The enterprise cannot create meetings because it has not purchased any concurrent meeting resources.:企业未配置并发会议资源,无法使用个人会议ID和随机会议ID发起会议
111071061	Parameter verify failed.:参数校验失败
111071062	Json conversion failed.:JSON转换失败
111071063	Not cloud meeting room owner.:非云会议室拥有者
111071064	This cloud meeting room has been started for another meeting during this period.:VMR会议冲突,当前已有会议正在召开
111071065	This meeting cannot be modified because it is already started.:正在召开会议,无法编辑会议详情
111071066	Attendees amount over range.:与会方数量超出上限
111071067	This meeting cannot be canceled because it is already started.:会议正在召开中,不允许取消会议
111071068	Meeting password encrypt failed.:会议密码加密失败
111071069	No meeting during this period.:选定的时间中没有会议
111071070	Sub cycle meeting cannot modify live room.:周期会议子会议不允许修改直播房间
111071071	Stop online meeting exception.:结束会议失败

错误码	错误描述
111071072	Meeting is ended.:会议已经结束
111071073	Not allow anonymous user.:会议不允许匿名用户呼入
111071074	Only allow user in organization.:会议只允许企业内用户呼入
111071075	Only allow invited user.:会议只允许被邀请用户呼入
111071076	Batch export task for history meeting list during this period.:当前有正在执行的批量导出任务
111071077	Only allow invited user and user in organization.:会议只允许被邀请或者企业内用户呼入
111071078	Max current participants amount reached.:企业单会议并发方数已达上限
111071079	Invalid CTD param.:ctd会议参数不合法
111071080	Invalid guest password.:来宾密码不合法
111071081	Guest password conflict with meeting ID.:来宾密码与会议ID冲突
111071082	Max meeting amount reached.:会议数达到上限
111071083	Invalid meeting random number.:无效的会议随机数
111071084	Max export history meeting list times reached.:导出历史会议次数超出允许的值
111071085	Max multi view layouts reached.:会议多画面布局超出限制
111071086	Max multi view invalid manual type.:会议多画面模式非法
111071087	Max multi view sub view amount reached.:会议多画面子画面个数超出限制
111071088	Max multi view sub view numbers reached.:会议多画面子画面会场数超出限制
111071089	Webinar and common meeting cannot be converted to each other.:网络研讨会不能修改为普通会议,普通会议也不能修改为网络研讨会
111071090	Not allow add audience.:不支持添加观众
111071091	Meeting password authentication failed.:密码错误
111071092	File export task failed.:导出文件任务失败
111071093	File export task is downloading.:导出文件正在处理中
111071094	Not allow audience to join.:会议不允许观众入会
111071095	Need to get WEBINAR_CONF.DTO object.:需要获取WEBINAR_CONF.DTO对象

错误码	错误描述
111071096	User-defined languages over range.:自定义语言个数超出范围
111071097	Your enterprise does not support simultaneous interpretation.:企业不支持同声传译
111071098	Need to set interpreter groups.:需要设置传译组信息
111071099	Invalid interpretation language.:传译组语言非法
111071100	Max languages in a meeting are reached.:单会议最大支持16种语言
111071101	This interpreter group ID already exists.:传译组的组号重复
111071102	Max interpreters in the interpreter group reached.:传译组中传译员的个数达到上限
111071103	This interpreter already exists in an interpreter group.:传译员重复
111071104	Invalid interpreter account.:传译员帐号不存在
111071105	Guest password conflict with chairman password.:来宾密码与主席密码冲突
111071106	Meeting is holding.:云会议室已有会议正在召开
111071107	Invalid audience password.:观众密码不合法
111071108	Audience password conflict with meeting id.:观众密码与会议ID冲突
111071109	Audience password conflict with chairman password.:观众密码与主席密码冲突
111071110	Only RTC meeting support E2E encrypt.:非RTC会议不支持端到端加密
111071111	Audience password conflict with guest password.:观众密码与来宾密码冲突
111071112	Cloud meeting room ID type cannot be modified.:不支持修改会议ID类型
111071113	Cloud meeting room type cannot be modified.:不支持修改VMR类型
111071114	Cloud meeting room flag cannot be modified.:不支持修改VMR标识
111071119	Sub cycle meeting modify time over range.:周期会议修改时间超出范围
111072001	Application for meeting resources failed.:申请资源失败
111072002	Failed to release meeting resources.:释放资源失败

错误码	错误描述
111072003	The meeting is ending.:会议正在结束
111072004	The meeting is starting.:会议正在开始
111072005	The meeting has not yet started.:会议还未开始
111072006	The meeting has ended.:会议已经结束
111072007	Site invitation failed.:邀请会场失败
111072008	Site does not exist.:会场不存在
111072009	Failed to disconnect the site.:挂断会场失败
111072010	Coming soon...:功能暂未支持,敬请期待...
111072011	Meeting status incorrect.:会议状态非法
111072012	Site control operation failed.:会场会控操作失败
111072013	The cloud meeting room is in use by another meeting.:当前VMR会议非目标会议
111072014	The meeting has ended.:VMR会议已过期
111072015	The meeting has not yet started.:VMR会议还未开始
111072016	Meeting does not exist.:VMR预约记录不存在
111072017	Meeting has no attendee.:会议与会者不存在
111072018	Application for chairman password exception.:申请主席密码错误
111072019	Not found site which can be moved exception.:没有找到可以移动的会场
111072020	The current meeting is not a recording meeting. Recording is not supported.:非录播会议不支持录制
111072021	System processing failure.:录播调度失败
111072022	Insufficient recording resources in the system.:录播资源不足
111072023	There are no participants in this meeting.:会议中没有与会人
111072024	Operation failed. No sites have joined this meeting.:请确保会议中已有会场入会
111072025	Voice meeting does not support this function.:语音会议不支持该功能
111072026	The voice site does not support this operation.:语音会场不支持该操作
111072027	Voice sites cannot be displayed in custom continuous presence.:手动设置的多画面中无法显示语音会场

错误码	错误描述
111072028	At least one site must have custom continuous presence configured.:手动设置的多画面中需最少设置一个会场
111072029	At most one presentation pane is allowed in custom continuous presence.:手工设置的多画面最多只允许一个辅流子画面
111072030	In the voice-activated continuous presence mode, each pane can only support a single user.:声控多画面模式下,每个子画面最多只允许一个用户
111072031	The number used for the site to join the meeting cannot be left blank.:与会者的入会号码不允许为空
111072032	The number used for the site to join the meeting is too long.:与会者的入会号码长度超过系统允许的最大值
111072033	The format of the number used for joining the meeting is incorrect. (Correct format: User name name/IP address:Port number or Country code plus phone number):与会者的入会号码格式非法(支持SIP、TEL号码格式)
111072034	Length of site name too long.:与会者的名称长度超过系统允许的最大值
111072035	The site role type is outside the allowed range.:与会者的角色类型不在系统允许的取值范围内
111072036	Length of site email address too long.:与会者的邮箱地址长度超过系统允许的最大值
111072037	Length of site mobile number too long.:与会者的手机号码长度超过系统允许的最大值
111072038	The value for whether a site is muted after it first joins the meeting is outside the allowed range.:与会者入会后是否静音的取值不在系统允许的取值范围内
111072039	The value for whether to automatically invite a site is outside the allowed range.:与会者是否自动邀请的取值不在系统允许的取值范围内
111072040	The host's site viewing type is outside the allowed range.:主席选看类型取值不在系统允许的取值范围内
111072041	The value of the continuous presence setting mode is outside the allowed range.:设置多画面方式取值不在系统允许的取值范围内
111072042	The polling time is outside the allowed range.:轮询显示时间取值不在系统允许的取值范围内
111072043	The pane type cannot be left blank.:画面类型不允许为空
111072044	Invalid pane type.:画面类型非法

错误码	错误描述
111072045	The value of the switch is outside the allowed range.:开关项取值不在系统允许的取值范围内
111072046	The site name cannot be left blank.:与会者的名称不允许为空
111072047	Site name too long.:与会者的名称超过系统允许的最大值
111072048	Continuous presence is not supported. Ask the administrator to configure related data on the MCU.:不支持多画面,请联系管理员开启MCU配置
111072049	Response timed out. Try again later.:等待响应超时
111072050	Meeting locked.:会议被锁定
111072051	The meeting display mode is empty.:会议显示模式为空
111072052	The meeting display mode is outside the allowed range.:会议显示模式不在系统允许的取值范围内
111072053	The meeting pane type is outside the allowed range.:会议画面类型不在系统允许的取值范围内
111072054	The banner or caption type is empty.:横幅或字幕类型为空白
111072055	The banner or caption content is empty.:横幅或字幕的内容为空
111072056	The banner or caption content cannot exceed 2,000 bytes.:横幅或字幕的内容长度超过2000字节
111072057	The request message body is empty.:请求消息体为空
111072058	The pane index is greater than the total number of panes.:子画面索引大于画面总数
111072059	Meeting has no host.:会议中没有主席
111072060	Regular participants cannot perform this operation.:当前普通与会方类型不支持该操作
111072061	You have set host polling and cannot select a specific site to view or set continuous presence.:前主席正在轮询观看会场,不能单独选看其他会场或会议多画面
111072062	You cannot select a site to view on the web page. Perform the operation on the terminal.:当前主席与会方类型不支持该操作
111072063	Applying for recording resources....:录制资源重新申请错误码
111072064	Too many failed login attempts. Try again later.:会控登录锁定
111072065	The meeting ID or password is incorrect. Check the ID and password and try again.:会议不存在或密码错误

错误码	错误描述
111072066	The enterprise has not purchased any recording storage space.:企业未购买录播空间
111072067	Insufficient enterprise recording storage space.:企业录播空间已满
111072068	Failed to start live broadcast. Try again later.:启动直播失败
111072069	Not allow to hand up for host.:不允许指定会议主席举手发言
111072070	The operation is not allowed for guest.:来宾不允许进行该操作
111072071	The operation is only allowed for yourself.:仅能对自己进行操作
111072072	Not allow to apply host for attendees who do not have meeting control abilites.:没有会控能力的与会者不允许设置为主席
111072073	Notify message is expired.:推送消息已过期
111072074	Only participants with a number registered on HUAWEI CLOUD can be set as the host.:云下会场不能成为主席
111072075	Hung up site for meeting ended.:会议结束挂断会场
111072076	Hung up site by host.:主席挂断会场
111072077	Hung up site because host deleted site.:主席删除会议挂断会场
111072078	Hung up site because meeting is recreated.:重建会议挂断会场
111072079	Hung up site for moving site failed.:移动会场失败挂断会场
111072080	Hung up site for inviting site cross region failed.:跨region邀请会场失败挂断会场
111072081	Hung up site for invitation exception.:邀请会场失败挂断会场
111072082	Hung up site for joining meeting repeatedly.:会场重复入会挂断会场
111072083	Lock sharing failed.:锁定共享失败
111072084	This site does not support 1-screen layout.:云下会场禁止设置在多画面一画面中
111072085	Not support realtime subtitle.:会议不支持实时字幕
111072086	Realtime subtitle pid is null.:会议实时字幕指定发言人列表为空
111072087	Invalid realtime subtitle pid.:会议实时字幕指定发言人ID有误或已离会

错误码	错误描述
111072088	Failed to set realtime subtitle.:会议实时字幕设置失败
111072089	Not allow audience to change speaker status.:不允许观众解闭音
111072090	Allow/forbid audience to speak only for audience.:主持人允许/禁止观众说话接口只能操作观众
111072091	This meeting does not support simultaneous interpretation.:会议不支持同声传译
111072092	Interpreter group is empty.:会议当前传译组为空,请先配置传译组
111072093	This meeting has not enable simultaneous interpretation.:会议未开启同声传译
111072094	This site is not interpreter.:当前会场非传译员
111072095	This site has not yet confirmed interpreter identity.:传译员未确认身份
111072096	Invalid site language.:会场选择的语言非法
111072097	This site is confirmed interpreter.:当前会场是已确认的传译员
111072098	This site listen channel is different from speak channel.:普通会场听说频道不一致
111072099	Failed to set interpretation languages. Try again.:设置会议语言频道失败
111072100	Failed to set interpretation channel. Try again.:设置会场语言频道失败
111072101	Max participant amount reached.:超过设置的会议方数
111072102	Max allowed handup amount reached.:观众超过最大允许举手数
111072103	Max allowed speaking amount reached.:观众超过最大允许发言数
111072104	Failed to set virtual audio CSD site.:设置级联通道虚拟语言会场失败
111072105	This meeting does not support cohost.:该会议未启用联席主持人
111072106	This site does not support cohost.:该会场不支持联席主持人
111072107	Host cannot be set to cohost.:不支持设置主持人为联席主持人
111072108	Audience cannot be set to cohost.:不支持设置观众为联席主持人

错误码	错误描述
111072109	Need to check cohost permission.:需要去主会议所在region判断是否为联席主持人
111072110	Cohost cannot operate host.:联席主持人不能操作主持人
111072111	This site does not support share invitation.:当前会场不支持邀请共享能力
111072201	Hung up site because heart beat failed.:与终端心跳中断导致挂断会场
111072202	Insufficient audience cloud meeting room logic resources.:观众数超过VMR套餐中的观众最大并发资源数
111072203	Insufficient participant cloud meeting room logic resources.:主持人与嘉宾超过VMR套餐中的最大并发资源数
111072204	Insufficient webinar resources, so started failed.:网络研讨会套餐已经有在线会议,被占用,激活失败
111072205	Only RTC conf support local record.:非RTC会议不支持本地录制
111072206	Invalid E2E encryption parameter.:端到端加密参数非法
111072207	Not allow to switch role in webinar.:网络研讨会不允许进行角色转换
111072208	The client version used by the user is too early. Cannot start local recording.:客户端不支持本地录制
111072211	Meeting is under AR assist.:会议已经在进行AR协作
111072213	Audience cannot be set to host.:观众不能设置为主持人
111072214	Rename param contains sensitive words.:改名涉及敏感词错误
111072222	Indicates record start exception.:启动云录制失败, 请联系管理员
111072223	Indicates record paused exception.:暂停云录制失败, 请联系管理员
111072224	Indicates record stop exception.:停止云录制失败, 请联系管理员
111072225	Indicates live start exception.:启动直播失败, 请联系管理员
111072226	Indicates live stop exception.:停止直播失败, 请联系管理员
111072504	Max participants in a single meeting reached.:达到系统单个会议人数上限
111072507	waiting room user is leaved.:等候室准入用户已经离会
111072508	waiting room user is joined.:等候室准入用户已经准入

错误码	错误描述
111073001	User does not exist.:用户不存在
111073002	Department does not exist.:组织不存在
111073003	Invalid username format.:用户非法
111073004	Invalid department name format.:组织非法
111073005	Could not query department.:查询组织失败
111073006	Application for anonymous user failed.:申请匿名用户失败
111073007	Invalid cloud meeting room package status.:VMR套餐状态非法
111073008	Failed to Release anonymous user.:释放匿名用户失败
111073009	Not support cycle meeting for welinkC.:WeLinkC 不支持预约周期会议
111073010	Not support cloud meeting room for welinkC.:WeLinkC 不支持预约VMR会议
111073011	Failed to get Welink token.:获取WeLink Proxy Token失败
111073012	Failed to Get M to M token.:获取机机帐号Token失败
111073013	Invalid webinar cloud meeting package status.:网络研讨会VMR套餐状态非法
111073014	Max extend times for cloud meeting trial version reached.:会议延长次数超出VMR试用套餐可延长的最大次数
111073015	Max extend duration for cloud meeting trial version reached.:会议延长时间超出VMR试用套餐单次会议最大时长
111073016	Not support auto extend meeting for cloud meeting trial version.:VMR试用套餐不支持自动延长
111074001	This meeting host already exists.:IDO主席已经存在
111074002	No meeting host permissions.:无IDO主席权限
111074003	Failed to extend the meeting. Max meeting duration reached.:延长会议失败,会议已达到最大时长
111074004	Failed to extend the meeting. The service processing is abnormal.:延长会议失败,服务处理异常
111075001	Failed to start timer.:启动定时器失败
111075002	Failed to end timer.:停止定时器失败
111075003	Json parse failed.:json转换异常
111076001	Released by host.:会场被终端主席挂断
111076002	Released by portal host.:会场被portal主席挂断

错误码	错误描述
111076003	Max number of participants reached.:超出VMR最大方数
111076004	Insufficient enterprise resources.:企业并发数不足
111076005	Not answered.:会场无响应
111076006	User does not exist.:会场不存在
111076007	Offline.:会场离线
111076008	Busy.:忙
111076009	Rejected by user.:拒接
111076010	Released by user.:会场挂断
111076011	Insufficient meeting resources in the system.:媒体资源不足
111076012	Outbound calling disabled.:PSTN服务未开通
111076013	Application for address token failed.:申请地址本鉴权信息失败
111076014	Call failed.:SIP呼叫超时
111076015	Max number of participants reached.:超过设置的会议方数
111076017	Invalid user.:用户失效
111076018	Invalid enterprise.:企业失效
111076019	Invalid SP.:SP失效
111076022	Not allow joining by scanning for hard device.:企业配置来宾不允许扫码邀请硬终端入会*
111076028	Not allow conf control.:会议正在进行AR协作, 禁止共享/分组会议/释放主持人等会控操作*
111076029	Not allow AR Assist.:会议正在进行共享/分组会议, 禁止进行AR协作*
111076099	Common error.:通常失败
111077001	RSE server info does not exist.:RSE录播相关错误码
111077101	Not support AI record.:会议不支持AI会议纪要
111077102	Reapplying for AI record resource.:正在重新申请会议纪要资源
111077103	Insufficient AI record resource.:AI会议纪要资源不足
111077104	In AI recording, the meeting cannot stop live or recording.:会议处于AI会议纪要状态,不允许停止录制/直播
111077201	Record file does not exist or have no permission.:录播文件已删除或您无权观看,请确认后重试

错误码	错误描述
111077202	RSE authentication failed.:RSE机机鉴权失败
111078000	No available region in multi region.:没有可用Region
111078001	No available global in multi region.:没有可用Global
111078002	Dispatch region error in multi region.:多region 分发region错误
111079000	Failed to phrase big data param query xml.:大参数查询xml解析失败
111079001	Failed to phrase anonymous assign xml.:匿名帐号分配xml解析失败
111080001	Bad request.:请求消息参数错误
111080002	Insufficient permissions.:用户无权限进行该操作
111080004	Site data not found or deleted.:会场不存在或已被删除
111081001	Site is inviting.:会场正在通话中
111081002	Site meeting does not exist or is ended.:会场所在会议不存在或已结束
111081003	Failed to apply site resource.:申请会场资源失败
111081004	Failed to release site resource.:释放会场资源失败
111081005	Failed to apply site media resource.:申请会场媒体资源失败
111081006	Authentication failed.:认证失败
111081007	Meeting is locked.:会议已经被锁定
111081009	Login is locked because max authentication times reached.:会议鉴权失败次数已达上限被锁定
111081010	Waiting user restrictions.:等候用户限制
111082001	Not support this control.:不支持该会控
111082002	No response.:无响应
111082003	Site not exist.:会场不存在
111082004	Site unavailable.:会场离线
111082005	Site busy.:会场正忙
111082006	Site refused.:会场拒接
111082007	Site leave.:会场挂断
111082008	PSTN service is closed.:PSTN服务关闭
111082009	SIP invitation timeout.:SIP呼叫超时

错误码	错误描述
111083001	RSE does not exist.:RSE设备不存在
111084001	Failed to execute DTMF:执行DTMF失败
111084002	Get started meeting exception.:启动会议异常
111084003	Call MCU side exception.:呼叫MCU端异常
111084004	Join call exception.:加入呼叫异常
111084005	Invite exception.:邀请异常
111084006	Call terminal side exception.:呼叫端侧异常
111084007	Answer exception.:应答异常
111084008	Common CTL exception.:通用CTL异常
111084009	Hangup exception.:挂断异常
111084010	Notify capability exception.:通知能力异常
111084011	Update MT exception.:更新MT异常
111084012	Negotiate exception.:协商异常
111084013	Not support this meeting.:不支持该会议
111084014	Not allowed join meeting in RTC inviting because guests are not invited.:RTC外邀入会,来宾不在被邀请名单中,拒绝入会
111089002	Failed to phrase URI join xml.:链接入会XML解析失败
111090001	Failed to join data meeting.:数据入会失败
111090002	Join data meeting overtime.:数据入会超时
111100001	Failed to get slider image.:获取滑块图片失败
111100002	Slider check failed.:滑块校验失败
111100003	Invalid random number.:无效的随机数
111100004	Invalid slider token.:无效滑块token
111100005	Need to check slider.:需要滑块校验
111100006	Invalid SMS phone number.:非法的手机号
111100007	Failed to send verification code.:发送验证码失败
111100008	Verification code expired.:验证码已过期
111100009	Verification code error.:验证码错误
111100010	Max verification code check times reached.:验证码错误超过最大次数

错误码	错误描述
111100011	The uuid is invalid, you need to re-verify the phone number.:uuid无效,需要重新验证手机号
111100012	The meeting is not on this site.:会议不在本站点
111100013	Need to verify the mobile phone number. Overseas access to China, free enterprises need to verify the mobile phone verification code.:需要验证手机号, 海外接入国内, 免费企业需要进行手机验证码校验
111110001	Failed to add IM group.:添加群组失败
111110002	Failed to disband IM group.:删除群组失败
111110003	Failed to add IM member.:添加成员失败
111110004	Failed to remove IM member.:删除成员失败
111111007	hangup for rejoin mmr failed.:mmr重入会失败挂断
111112001	Failed to enter the waiting room.:入等候室失败
111112002	Failed to exit the waiting room.:出等候室失败
111112003	Waiting room cache operation failed.:等候室缓存操作失败
111112004	Co-hosts cannot be moved into the waiting room.:联席主持人不能被移入等候室
111112005	Not allowed to enable waiting room.:不允许启用等候室
111112008	Not support this feature according to client capabilities.:客户端能力不支持此特性
111113001	can not find application.:找不到应用信息
111113002	application exist.:应用已存在, 目前一个会议只能关联一种应用
111113003	not required to create application.:无权限创建应用
111113004	need verify id.:应用操作需要实名认证(用户需要绑定手机)
111113005	get token failed.:获取应用用户token失败
111113006	url illegal.:应用服务URL地址非法
111113101	shimo doc has not used.:石墨文档应用未启用
111113102	call shimo doc server failed.:调用石墨文档服务失败
111113103	get shimo token failed.:获取石墨应用级token失败
111113104	not in conf,has no right to apply.:用户不在会议中, 无权限申请权限
111500301	breakout conf surpass limit.:分组会议子会议超过上限

错误码	错误描述
111500302	breakout conf name error:分组会议子会议名称错误
111500303	breakout Start failed.:分组会议启动失败
111500304	breakout not exist when delete:分组会议中没有子会议（删除）
111500305	breakout not exist when start.:分组会议中没有子会议（启动）
111500306	breakout is closing by host:分组会议结束中(主持人停止)
111500307	breakout is closing by host:分组会议已经结束或未启动（加入时）
111500308	Attendee in target conf:移动用户在目标分组
111500309	Chair in current conf:主持人在当前分组中（请求帮助）
111500310	not support breakout conf:不支持分组会议（启动时 启动接口）
111500311	breakout conf starting or started:分组会议已经启动或启动中（重复启动）
111500312	subconf not exist:子会议不存在
111500313	not support in mainconf:主会中用户不支持此操作（请求帮助）
111500314	breakout conf state illegal:分组会议状态异常（缓存误删）
111500315	action forbidden by status:分组会议静态、动态操作由于状态原因，不支持当前操作
111500316	apply chair failed due to host in subconf:主持人在分组中，不能申请为主持人
111500317	apply sub conf id failed:申请分组的会议id失败
111500318	query main conf failed:获取主会议信息失败
111500319	attendee not found:没有找到用户信息
111500320	sub conf stop failed:分组会议停止失败
111502000	subject faile and leave conf start:订阅失败需要离会的错误码起始值
111502099	subject faile and leave conf end:订阅失败需要离会的错误码结束值
112010001	Common failure.:通用失败
112010002	Null param.:参数为空
112010003	Participant not found.:未找到参与者

错误码	错误描述
112010004	Meeting not found.:会议不存在
112010005	SIP invite information timeout.:invite消息超时
112010006	SIP invite information failed.:invite消息失败
112010007	SIP information timeout.:info响应超时
112010008	SIP information failed.:info响应失败
112010009	System processing failed.:系统处理失败
112010010	System processing failed.:系统处理失败
112010011	Failed to invite MRS.:呼叫mrs失败
112010012	Failed to invite participant.:呼叫终端失败
112010013	Data meeting join information failed.:数据入会Info失败
112010014	Failed to start meeting.:召开会议失败
112010015	Media negotiation failed.:媒体协商失败
112010016	User rejects.:用户拒绝
112010017	User is busy.:用户忙
112010018	No call permission.:没有呼叫权限
112010019	Unknown callee.:未知被叫人
112010020	No user response.:没有用户响应
112010021	Join meeting timed out.:加入会议超时
112010022	Session has been released.:会话已释放
112010023	Callee is not registered.:被叫人未注册
112010024	SIP invite timed out.:SIP邀请超时
112010025	User Session timer times out and hangs up.:用户Session timer 超时挂断
112010026	The participant bye hangs up on the MRS side.:与会者腿MRS侧bye挂断
112010027	The participant bye hangs up on terminal side.:与会者腿终端侧bye挂断
112010028	No RTP stream hang up.:无码流挂断
112100000	Sysportal exception.:系统异常
112100001	Bad request.:请求异常
112100002	Illegal request.:非法请求

错误码	错误描述
112100003	Data not found.:数据未获取到
112100004	Data duplication.:数据重复
112100006	The requested type is not supported.:请求的类型不支持
112100007	Failed to save notification template.:保存通知模板失败
112100008	Failed to save data.:保存数据失败
112100009	Failed to query data.:查询数据失败
112100010	Failed to delete data.:删除数据失败
112100011	Password reset required.:需要重置密码
112100012	Incorrect old password.:旧密码错误
112100013	The new password cannot be the same as the old password.:新密码不能与旧密码相同
112100014	Insufficient password complexity and low security.:密码复杂度不够,安全性低
112100015	The length does not meet the requirements.:长度不符合要求
112100016	RSE is in meeting.:RSE在会议中
112100017	RSE to be deleted.:RSE待删除
112100018	Failed to query the meeting list.:查询会议列表失败
112100019	User has been locked out. Try again later.:用户已被锁定,请稍后重新登录
112100022	The live room channel name exists.:频道名称已存在
112100023	Failed to modify system configuration items for duplicated meeting ID prefix.:修改系统配置项失败,会议ID前缀重复
112100024	CONFID prefix can only be added, not modified or deleted.:CONFID前缀只能增加,不能修改和删除
112100025	The live room channel url exists.:直播房间地址已存在
112100026	The live room has been occupied.:直播房间已被使用
112100027	The live room does not exist.:直播房间不存在
112100028	Live room release failed.:直播房间释放失败
112100029	The request is missing header or param.:请求缺失头域
112100030	The Live main stream URL already exists.:音视频推流地址已存在

错误码	错误描述
112100031	The Live auxiliary stream URL already exists.:辅流演示推流地址已存在
112100032	Password is weak.:密码为弱密码
112100033	The live room is occupied.:直播房间被占用
112100034	Save part of attendees since max attendees size reached.:因为与会者数量超出上限只保存部分
112100035	Failed to export data.:导出数据失败
112100036	Failed to import data.:导入数据失败
112100037	Max file size reached.:文件大小超出上限
112100038	LVS target address exists.:lvs的目标地址已存在
112100039	LVS port existed.:lvs端口重复
112100040	Not allow to modify global standby config.:Global只有主机可以修改配置;
112100041	MCU insulation IP address exists.:mcu隔离的IP已经存在
112100042	MCU insulation IP address cannot be modified.:mcu隔离的IP不能修改
112100043	SBC address exists.:SBC配置的地址已经存在
112100044	Internal exception.:内部错误异常
112100045	System config delete exception.:系统配置不允许被删除
112100046	No ISBC access address.:企业id无法获取关联SBC地址
112100047	Media type need to be HD because the organization enable Full HD capability.:企业已开启全高清功能,媒体设置不能低于1080P/8M
112100061	Same interpreter language abbreviation exists.:语言缩写重复
112100062	Same interpreter language name exists.:语言名称重复
112100063	MCU type is MIXED and insulation config is not configured, or MCU meeting exists when modifying mcu connection settings.:修改MCU连接配置告警,以下两种场景会导致:1.MCU存在会议 2.修改类型为MIXED未配置隔离
112100064	MCU type is MIXED and insulation config is not configured when adding mcu connection settings.:新增MCU连接配置告警,MCU类型为MIXED且未配置隔离
112100065	MCU type is MIXED when modifying MCU insulation config or disbanding enterprise.:修改MCU隔离配置时,解除企业绑定时MCU类型为MIXED

错误码	错误描述
112100066	MCU type is MIXED when deleting MCU insulation config.: 删除MCU隔离配置时,MCU类型为MIXED
112100067	This gateway type is incompatible with MCU.:MCU与网关类型不兼容
115021001	Insufficient media resource.:媒体资源不足
115021002	Meeting does not exist.:会议不存在
115021003	Media resource does not exist.:媒体资源不存在
115021005	Insufficient media bandwidth resource.:带宽资源不足
115021006	Media device fault.:设备故障
115022000	Insufficient common logic resource.:通用的逻辑资源不足
115022001	Insufficient meeting resources in the enterprise.:企业会议并发方数不足
115022002	Insufficient logic port resource.:逻辑资源端口资源不满足
115022003	Live broadcast resources have not been purchased or are insufficient.:企业直播推流方数不足
115022004	Media device does not exist.:媒体设备不存在
115022005	The cloud recording ports purchased by enterprises have been used up.:企业购买的云录制端口已经用完
115023001	Insufficient ID resources.:ID资源不足
115024001	RMS service exception.:RMS服务异常
115025001	The live room has already been occupied.:直播房间已经被使用
115025002	The live room does not exist.:直播房间不存在
115025003	Live room release failed.:直播房间释放失败
115026001	MCU meeting exists.:MCU有会议
115029001	Invalid common parameter.:通用参数值不合法
115029002	Null parameter.:参数值为空
115029003	service is not available.:服务不可用
115029004	Configuration exception.:配置异常
115029005	Duplicately apply resources.:重复申请资源
115029006	Waiting for applying resources.:等待申请资源中
115029999	Unknown error.:未知错误

错误码	错误描述
118000000	User authentication failed.:用户鉴权失败
118000001	User token authentication failed.:usg token 鉴权失败
118000002	User authentication account lockout.:usg鉴权帐号锁定
200000001	Server is busy. Please try again later.:服务器正忙 请稍后再试
200000002	Internal error. Please contact SRE.:服务器正忙 请联系SRE
200000003	Invalid parameter. Please check whether the parameters are valid.:请求参数错误 请检查参数是否合法
200000004	Illegal request. Please check the value of the request header field X-Auth-Token or X-Access-Token.:非法请求 请检查请求头域X-Auth-Token或X-Access-Token的值
200000005	Forbidden access during the version upgrade.:系统正在升级, 暂时不可访问。如对您造成不便, 十分抱歉
201020023	This user is already the owner of another corporation and cannot create a new corporation for it.:该用户已经是其他企业的所有者,无法为其创建新的企业
201040005	Max number of users reached. Please check the purchased meeting users.:用户总数超过限制, 请检查购买的会议用户
201040016	Password security is low. Please set another password.:密码安全性低, 请设置其他密码
201040031	Activation code overdue.:激活码过期
201040042	Pre-authentication expired.:预验证失效
201040043	This user is the owner of another corporation and cannot be invited.:该用户是其他企业的所有者,暂无法邀请
201040044	Invalid invite connection.:邀请连接非法
201040045	User is already in waiting list.:用户已经在等待列表
201040046	User already exists.:用户已存在
201040047	Account modification is not allowed.:不允许修改账户
201050012	Activation code does not exist.:激活码不存在
201050015	Invalid activation code.:激活码错误
201050021	Activation failed.:激活失败
201050022	Terminal does not allow unbinding.:终端不允许解绑
201080000	VMR resource unavailable.:云会议资源不可用

错误码	错误描述
201080006	The host password and the guest password cannot be the same. Please confirm the input parameters.:主席密码和来宾密码不能相同 请确认输入参数
201080017	Same as meeting password.:和会议密码相同
201510001	No invite permission to get an invite code.:没有邀请权限去获取邀请码
201510003	Invalid application.:申请失效
206010000	User name or password error.:用户名或密码错误
206010007	Account is not activated, no permission to log in.:帐号未激活,无权限登录
206010010	Failed to get verification code, Please contact SRE.:获取验证码失败 请联系SRE
206010011	Verification code must be filled.:验证码不能为空
206010012	Verification code check failed.:验证码校验失败
206010013	Jump login failed. Please contact SRE.:跳转登录失败 请联系SRE
206010022	SSO login failed. Please contact SRE.:SSO登录失败 请联系SRE
206010023	SSO login is not supported.:不支持sso登录
206010024	Invalid authentication type.:认证类型非法
206010025	APPID verification failed.:APPID校验失败
206010027	Authentication type error.:校验类型错误
206010029	Network error.:网络异常
206010030	APPID does not allow automatic registration.:APPID不允许自动注册
206010031	APPID auto registration failed.:APPID自动注册失败
206010036	Normal users cannot log in.:普通用户无法登录
206010042	Huawei account authentication failed.:华为账户认证失败
206010057	APPID authentication expires.:APPID认证过期
206010058	APPID account is not bound.:APPID账户未绑定
206010064	Site mismatch.:站点不匹配
206010065	Phone number conflict.:电话冲突
206010066	E-mail conflict.:邮箱冲突

错误码	错误描述
206010070	The number of users has reached the maximum.:用户数超过上限
206010071	WeChat authentication failed.:微信鉴权失败
206010074	International sites support site changing re-login.:国际站点支持切站点重登
206010075	Account has been unbound.:账号已经解绑
206020000	Invalid authentication information. Please check the value of the request header field X-Auth-Token or X-Access-Token.:鉴权信息无效 请检查请求头域X-Auth-Token或X-Access-Token的值
206020001	Token is created by disaster recovery site.:token是容灾站点创建
206020002	Token is invalid due to being kicked.:token由于被踢失效
206030000	The user does not exist.:系统中不存在该用户
206030001	The verification code is incorrect or has expired. Please obtain a new one.:验证码错误或失效，请重新获取
206030002	Incorrect verification code.:验证码错误
206030003	Failed to send the verification code.:发送验证码失败
206030004	Try to get a new verification code 180s later.:请在180秒后重新发送验证码
206030005	The password cannot be changed.:系统不支持修改密码
206030006	The old password is incorrect.:旧密码错误
206030007	The length of the new password does not meet the requirements.:新密码长度不符合要求
206030008	The password complexity does not meet the requirements.:密码复杂度不满足要求
206030009	The new password must be different from the previous old passwords.:设置的密码不能与最近旧密码相同
206030010	The password cannot contain three or more of the same characters consecutively.:密码不能包含3个以上重复字符
206030011	The password cannot be changed within 5 minutes upon the last password change.:修改密码后，5分钟内不能更新密码
206030012	The password cannot contain the account or the reverse order of the account.:密码不能与帐号或其逆序帐号相同

错误码	错误描述
206030013	At least two characters must be different between the new and old passwords.:新密码相较于旧密码至少要有两个不同的字符
206030014	Invalid password format.:密码格式不正确
206030015	The user is locked and will be locked in 60s.:用户已被锁定,请60秒后重试
206030016	Have no right to reset other's password.:没有权限重置密码
206030018	The number of password retrieval times in a day has reached the maximum(10 times):.一天内忘记密码次数达到最大值 (10次)
206030019	The number of incorrect verification code input has reached the maximum, Please get it again.:验证码输入错误次数超过最大值, 请重新获取
206030023	Failed to send verification code. Please contact SRE.:发送验证码失败 请联系SRE
206030028	The system has enabled SSO login and does not support changing the password. Please retry after the system has disabled SSO login.:系统已开启SSO登录,不支持修改密码 系统关闭SSO登录后重试
206030031	Forgot password for Huawei account.:华为帐号忘记密码
206040000	No corresponding terminal type.:无对应的终端类型
206060008	Sending verification code is locked.:发送校验码锁定
206060009	Slider verification required.:需要滑块校验
206060010	Invalid slider token.:滑块token非法
206060011	Request parameter error.:请求参数错误
206060012	Sending SMS is not supported.:不支持发送短信
206060013	Repeated sending is not supported.:不支持重复发送
206060014	Invalid verification code.:验证码非法
206060015	Max verification codes check times reached.:验证码次数超过最大次数
206060017	Failed to bind Huawei account.:华为账户绑定失败
211030046	Not the ideahub activation code.:不是ideahub激活码
211030050	auth fail too many times, locked.:身份验证失败次数太多,已锁定
211030052	auth fail too many times, permanently locked.:身份验证失败次数太多,已永久锁定

错误码	错误描述
211060003	The maximum times of log uploads has been reached.:日志上传次数达到最大限制
411000001	Common failure.:失败
411000002	Parameter error.:参数错误
411000003	Meeting does not exist.:会议不存在
411000004	A meeting already exists.:已经有会议存在
411000005	Meeting state error.:会议状态异常
411000006	Failed to get device.:获取设备异常
411000007	Failed to set device.:设置设备异常
411000008	No device available.:无可用设备
411000009	This operation is only supported on mobile.:只有移动端支持该操作
411000010	Network access error.:网络接入错误
411000011	Other supplementary service is running.:正在进行其他补充业务
411000012	Memory allocation error.:分配内存错误
411000013	Initialization failed.:初始化失败
411000014	User does not exist.:用户不存在
411000015	Did not join the data meeting.:未加入数据会议
411000016	Neither meeting nor call exists.:会议和呼叫均不存在
411000017	SVN port detection failed.:svn端口探测不通
411000018	Failed to write file.:写文件失败
411000019	RTC does not support STG mode.:RTC不支持STG模式
411000020	Failed to load RTC library. RTC meeting is not supported.:rtc库加载失败, 不支持rtc会议
411000021	The meeting is in the audience pause state when changing roles.:转角色的时候会议正处于观众暂停观看状态
411000022	Not in call.:不在呼叫中
411000023	Meeting or call does not exist.:会议或者呼叫不存在
411000024	Pairing state not allowed.:配对状态不允许
411000025	Pairing request failed.:配对请求失败
411000026	Pairing exceeds limit.:配对超出限制

错误码	错误描述
411000027	Pairing common error.:配对通用错误
411000028	Pairing and joining the meeting failed.:配对入会失败
411000029	Pairing and joining the meeting is busy.:配对入会忙
411000030	Pairing and joining the meeting is invalid.:配对入会无效
411000031	Multiple clicks at the same time to pair and join the meeting error.:同时多次点击入会错误
411000032	Pairing is not established or is being established.:配对未建立或正在建立中
411000033	The camera is not turned on in the third-party capture mode.:第三方采集模式下摄像头未打开
411000034	RTC engine creation failed.:RTC 引擎创建失败
411000035	Not in a meeting.:不在会议中
411000036	mute self failed.:静音自己失败并且采用了保护
411000100	End the meeting with no stream due to disconnection of network.:无码流结束会议网络断开
411000101	End the meeting with no stream because fail to re-join meeting.:无码流结束会议重入会失败
411000102	Null pointer exception.:空指针异常
411000103	Parameters cannot be empty.:参数不许为空
411000104	Indicates encrypt and decript func not match.:加解密算法不匹配
411000105	Indicates Loading User-Defined DLLs failed .:加载用户自定义dll失败
411000106	Indicates Applition not support shimo docs .:企业配置不支持打开石墨文档
411000107	Indicates notification of render fail .:数据会议UI渲染失败
411000108	End the meeting with mmr media rejoin failed.:mmr重入会失败
412100001	The message processing thread is not initialized.:消息处理线程未初始化
412100002	Parameter error.:参数错误
412100003	Timeout.:超时
412100004	Memory allocation error.:分配内存错误
412100005	XML parsing error.:XML解析出错

错误码	错误描述
412100006	DNS resolution error.:DNS解析异常
412100007	Request message failed.:请求消息异常
412100008	Authentication failed.:鉴权失败
412100009	Server error.:服务器异常
412100010	Account is locked.:帐号被锁定
412100011	Create timer error.:创建定时器错误
412100012	Server CA certificate verification failed.:服务器CA证书校验失败
412100013	Network error.:网络异常
412100014	Server redirect.:服务器返回重定向地址
412100015	Account is not activated.:帐号未激活
412100016	JSON parsing error.:JSON解析失败
412100017	The proxy communication times out and the TCP link building fails. Probably because the proxy configuration is incorrect.:代理通信超时, tcp建链失败, 大概率是代理配置错误
412100018	Disconnected by the proxy. Probably because the server address is misconfigured.:接被代理断开, 大概率是服务器地址配置错误
412100019	The TCP link establishment timed out. Probably because the proxy domain name or port configuration is incorrect.:tcp建链超时, 大概率是proxy域名或者端口配置错误
412100020	The authentication process failed. The user does not have proxy permission and needs to apply for proxy permission.:认证鉴权过程失败, 用户没有proxy权限, 需要申请proxy权限
412100021	The authentication process failed. The proxy user name or password is incorrect.:认证鉴权过程失败, 代理用户名或者密码错误
412100022	The authentication process failed. Incorrect authentication type. User name and password are required. The user needs to be prompted for input.:认证鉴权过程失败, 错误的认证类型, 同时需要用户名、密码, 需要提示用户输入
412100023	The authentication process failed. Other unknown errors. It may be that the authentication method is not supported. The user needs to be prompted to try again.:认证鉴权过程失败, 其他未知错误, 有可能是不支持认证方式, 建议弹框提示用户重新发起

错误码	错误描述
412100024	The authentication process failed. Unable to resolve gateway.:认证鉴权过程失败，无法解析网关
412100025	The authentication process failed. Unable to resolve host.:认证鉴权过程失败，无法解析主机
412100026	The authentication process failed. Unable to connect.:认证鉴权过程失败，无法连接
412100027	The authentication process failed. Operation times out.:认证鉴权过程失败，操作超时
412100028	Message length too long.:消息长度太大
412100029	Base64 encryption failed.:base64加密失败
412100030	XML parsing error.:XML解析出错
412100031	Secure function error.:安全函数失败
412100032	Firewall detection parameter error.:防火墙检测参数错误
412100033	Failed to create STG tunnel.:创建STG隧道失败
412100034	Failed to update STG service.:更新STG服务失败
412100035	Encryption failed.:加密失败
412100036	Do not allow duplicate downloads.:不允许重复下载
412100037	File operation failed.:文件操作失败
412100038	The request message returns empty.:请求消息返回空
412100039	The number of detection servers is 0 error.:探测服务器数为0错误
412100040	Failed to get local IP address.:获取本地IP失败
412100041	XXX.:登录模块消息未初始化
412100042	http return 429 error.:http 返回429错误
412100043	Not logged in, not available:此业务未登录不可用
412200001	Internal error.:内部错误
412200002	Parameter error.:参数错误
412200003	Network timeout.:网络超时
412200004	Network Error.:网络错误
412200005	Need to bind mobile phone.:需要绑定手机
412200006	Need to bind email.:需要绑定邮箱
412200007	Has initialized already.:重复初始化

错误码	错误描述
412200008	State machine execution failed.:状态机执行失败
412200009	Failed to start timer.:开启定时器失败
412200010	The user is prompted to change the password when logging in for the first time.:首次登录, 提示用户修改密码
412200011	Unknown authentication type.:未知鉴权类型
413000001	Common error.:一般错误
413000002	Parameter error.:参数错误
413000003	Memory allocation error.:分配内存错误
413000004	System error.:系统错误
413000005	Message sending error.:发送消息错误
413000006	Get system configuration error.:获取系统配置错误
413000007	Physical network error.:物理网络错误
413000008	Network access error.:网络接入错误
413000009	Create timer error.:创建定时器错误
413000010	Call state error.:呼叫状态错误
413000011	Other operation in progress error.:正在进行其他操作错误
413000012	Requesting the master to start call error.:请求主控进行呼叫开始错误
413000013	Requesting audio resource error.:申请音频资源错误
413000014	Other supplementary service is running.:正在进行其他补充业务
413000015	Record call error.:记录通话记录错误
413000016	Media process error.:媒体进程返回的错误
413000017	Maximum number of calls reached.:超过最大呼叫路数
413000018	SIP account ID does not exist.:SIP账户ID不存在
413000019	Call ID does not exist.:呼叫ID不存在
413000020	Failed due to registering.:正在注册中导致失败
413000021	Registration failed.:注册失败
413000022	Deregistration failed.:注销失败
413000023	Setting account information error.:设置账户信息错误
413000024	Setting SIP account error.:设置SIP帐号失败

错误码	错误描述
413000025	Reporting SIP account information error.:上报SIP帐号信息失败
413000026	Server information error.:服务器信息错误
413000027	SIP account information error.:账户信息错误
413000028	SIPC execution error.:SIPC执行错误
413000029	Not registered error.:未注册错误
413000030	Call Sip interface to subscribe error.:调用Sip接口订阅失败
413000031	Canceling subscription error.:注销订阅中错误
413000032	Subscribing error.:订阅中错误
413000033	Only one local meeting is allowed.:只允许一个本地会议
413000034	Local meeting is not created.:本地会议未创建
413000035	The participant line does not match the chair account.:与会者线路与主席账户不匹配
413000036	Local meeting state error.:本地会议状态错误
413000037	Failed to report the linkage state of the IP Phone.:IP Phone 端联动状态上报失败
413000038	Meeting ID does not exist.:会议ID不存在
413000039	Video hold error.:视频保持失败
413000040	Video unhold error.:视频恢复失败
413000041	The maximum number of server meeting reached.:服务器会议个数超出
413000042	Get AA random number error.:获取AA随机数失败
413000043	AA login error.:AA登陆失败
413000044	Meeting type mismatch.:会议类型不匹配
413000045	Failed to start video meeting.:视频会议开启失败
413000046	AA no master server.:AA无主服务器
413000047	Password error.:密码错误
413000048	Username error.:用户名错误
413000049	User is logged in.:用户已登录
413000050	Account is locked.:账户已锁定
413000051	Terminal type mismatch.:终端类型不匹配
413000052	Parsing XML error.:解析XML错误

错误码	错误描述
413000053	Connecting to server error.:连接服务器错误
413000054	Get media configuration error.:获取媒体配置失败
413000055	Get service permission error.:获取业务权限失败
413000056	Insufficient service permission.:业务权限不足失败
413000057	Network environment error.:网络环境错误
413000058	Service conflict.:业务冲突
413000059	Connection timeout.:连接超时
413000060	Unknown error.:未知错误
413000061	Add ringing information error.:添加振铃信息失败
413000062	Delete ringing information error.:删除加振铃信息失败
413000063	Create ringing number error.:创建振铃号码失败
413000064	VVM parameter error.:VVM参数错误
413000065	Getting registered service error.:获取登记业务错误
413000066	Getting voicemail error.:获取语音邮箱错误
413000067	Attendee already exists.:与会者已经存在
413000068	Attendee does not exist.:与会者不存在
413000069	Create server video meeting window error.:创建服务器视频会议窗口失败
413000070	Video window already exists.:视频窗口已存在
413000071	Getting meeting list error.:获取会议列表失败
413000072	Operating requires host privileges.:需要主持人权限才能操作
413000073	No video device to operate.:没有视频设备可以操作
413000074	Refresh registration is not turned off.:没有关闭刷新注册
413000075	Online state report error.:在线状态上报失败
413000076	Network address subscription report error.:网络地址本订阅 notify 上报失败
413000077	Telepresence account was force unregistered. Information report error.:智真帐号被踢,info 上报失败
413000078	TLS root certificate error.:TLS根证书错误
413000079	AD authentication error.:AD鉴权失败
413000080	Meeting list is getting.:会议列表正在获取中

错误码	错误描述
413000081	Prohibited calls.:禁止呼叫
413000082	The maximum number of calls has been reached.:呼叫数达上限
413000083	The maximum number of encrypted calls has been reached.:加密呼叫数达上限
413000084	The maximum number of video calls has been reached and needs to downgrade to audio.:超出视频呼叫数，需要降为音频
413000085	H323 account ID does not exist.:H323账户ID不存在
413000086	H323 account information error.:H323账户信息错误
413000087	H323 account information report failed.:H323帐号信息上报失败
413000088	Insufficient local resources.:本端资源不足
413000089	The calling protocol is not supported.:不支持该呼叫协议
413000090	Setting H323 account information error.:设置H323账户信息错误
413000091	Mediatrix meeting access number reporting failed.:mediatrix会议接入号上报界面失败
413000092	Mediatrix VMR information reporting failed.:上报Mediatrix VMR信息到界面失败
413000093	IMS meeting list information reporting failed.:上报获取IMS会议列表信息到界面失败
413000094	TUP initialization is not complete.:tup初始化未完成
413000095	Message return value error.:消息返回值错误
413000096	Message content too large error.:消息内容太大错误
413000097	Secure function error.:安全函数返回错误
413000098	message response is empty error.:消息响应为空错误
413000099	Account is not enabled.:帐号未启用
413000100	Create call control implement error.:创建呼叫控制类错误
413000101	Already in call.:已经在呼叫中
413000102	Call does not exist.:呼叫不存在
413000103	Failed to get media configuration information.:获取媒体配置信息失败

错误码	错误描述
413000104	Failed to set third-party capture. Calling HME_V_Capture_InputData_MCU failed.:设置第三方采集失败, 调用HME_V_Capture_InputData_MCU返回失败
414100001	Common error.:一般错误
414100002	Parameter error.:参数错误
414100003	Timeout.:超时
414100004	Memory allocation error.:分配内存错误
414100005	DNS resolution error.:DNS解析异常
414100006	Request message failed.:请求消息异常
414100007	Authentication failed.:鉴权失败
414100008	Server error.:服务器异常
414100009	Create timer error.:创建定时器错误
414100010	Start timer error.:开启定时器错误
414100011	The meeting control block is full. Please release the control block that is not in use.:会控控制块已满, 请释放没在使用的控制块
414100012	The proxy communication times out and the TCP link building fails. Probably because the proxy configuration is incorrect.:代理通信超时, tcp建链失败, 大概率是代理配置错误
414100013	Disconnected by the proxy. Probably because the server address is misconfigured.:接被代理断开, 大概率是服务器地址配置错误
414100014	The TCP link establishment timed out. Probably because the proxy domain name or port configuration is incorrect.:tcp建链超时, 大概率是proxy域名或者端口配置错误
414100015	The authentication process failed. The user does not have proxy permission and needs to apply for proxy permission.:认证鉴权过程失败, 用户没有proxy权限, 需要申请proxy权限
414100016	The authentication process failed. The proxy user name or password is incorrect.:认证鉴权过程失败, 代理用户名或者密码错误
414100017	The authentication process failed. Incorrect authentication type. User name and password are required. The user needs to be prompted for input.:认证鉴权过程失败, 错误的认证类型, 同时需要用户名、密码, 需要提示用户输入

错误码	错误描述
414100018	The authentication process failed. Other unknown errors. It may be that the authentication method is not supported. The user needs to be prompted to try again.:认证鉴权过程失败, 其他未知错误, 有可能是不支持认证方式, 建议弹框提示用户重新发起
414100019	Curl proxy exception.:curl代理异常
414100020	Curl host exception.:curl主机异常
414100021	Curl connection exception.:curl连接异常
414100022	Curl operation timed out.:curl操作超时
414100023	Message length too long.:会控消息内容过大
414100024	Failed to create thread.:创建线程失败
414100025	Secure function error.:安全函数返回错误
414100026	JSON parsing failed.:解析json体失败
414100027	WS token is empty.:ws token值为空
414100028	Return JSON data error.:返回json数据异常
414100029	Failed to create meeting control block.:创建会议控制块失败
414100030	Failed to get meeting control block.:获取会议控制块失败
414100031	Failed to get meeting control block by CallId.:通过CallId获取会议控制块失败
414100032	Destroying the meeting control block failed.:销毁会议控制块失败
414100033	Temporary creating meeting information not found.:找不到创会临时信息
414100034	Failed to get participant ID.:获取参会者ID失败
414100035	Failed to get MT by participant ID.:通过ParticipantId获取MT失败
414100036	Failed to get self information by participant ID.:通过ParticipantId获取selfinfo失败
414100037	Failed to get participant information by MT.:通过MT获取会议成员信息失败
414100038	Manual invitation required.:需要手动邀请
414100039	The major parameters of the meeting have been reported.:已上报过会议大参数
414100040	Subtitles are not supported.:不支持字幕

错误码	错误描述
414100041	Meeting Hang Up - Timeout.:会议挂断 - 超时
414100042	Meeting Hang Up - No Streaming.:会议挂断 - 无码流
414100043	Meeting hung up - Hung up by the host.:会议挂断 - 被主持人挂断
414100044	Meeting Hang Up - Meeting has ended.:会议挂断 - 会议已结束
414100045	End of meeting due to leave meeting.:离会导致的会议结束
414100046	Message queue processing function registration failed.:消息队列处理函数注册失败
414100047	Message queue processing function deregistration failed.:消息队列处理函数去注册失败
414100048	Meeting ended due to reject meeting.:拒绝会议导致的会议结束
414100049	Meeting ended by self.:自己结束会议导致的会议结束
414100050	The meeting control module is not initialized.:会控模块未初始化
414100051	The watching special site reaches the maximum.:选看的特殊会场超过限制
414100052	The watching HD site reaches the maximum.:选看的高清会场超过限制
414100053	The watching SD site reaches the maximum.:选看的标清会场超过限制
414100054	The watching specification reaches the maximum.:选看的规格超过限制
414100055	Watching internal errors.:选看内部错误
414100056	Duplicated watch render.:选看的句柄重复
414100057	HTTP request error.:HTTP请求错误
414100058	Websocket request timeout.:websocket请求超时
414100059	Loading failed.:加载错误
414100061	The meeting ends due to the timeout of guest waiting before meeting or the timeout of the meeting control heartbeat.:会前等待页面时间超时或者会控心跳超时导致的会议结束
414100062	The meeting ends due to being canceled.:会议被取消预约导致的会议结束

错误码	错误描述
414100063	The meeting ends due to network error when guest waiting before meeting.:会前等待页面网络异常导致的会议结束
414100064	http return 429 error.:http 返回429错误
414100065	self is chairman.:已是主持人
414100066	self is cohost.:已是联席主持人
414100067	no cohost and chairman.:无主持人和联席主持人
414100068	no supported cohost and chairman.:无支持的主持人和联席主持人
414100069	Invalid user, enterprise or SP.:用户，企业或SP失效
415000001	DB is not initialized.:db未初始化
415000002	Common failure.:失败
415000003	Proxy detection has not been completed.:代理探测还没有完成
416000001	Invalid parameter.:参数错误
416000002	Track module is not initialized.:打点模块未初始化
416000003	Track event does not exist.:打点事件不存在
416000004	Not logged in.:未登录
416000005	Upload is not enabled.:未开启上传
419100001	Invalid parameter.:错误参数
419100002	Incorrect URI.:错误号码
419100003	Memory allocation failed.:内存分配失败
419100004	SDP error.:SDP错误
419100005	Connection does not exist.:连接不存在
419100006	Connection failure.:连接失效
419100007	Operation forbidden.:禁止操作
419100008	State error.:状态错误
419100009	No corresponding line data.:无对应线路数据
419100010	No corresponding registered instance.:无对应注册实例
419100011	No corresponding call instance.:无对应呼叫实例
419100012	No corresponding subscription instance.:无对应订阅实例
419100013	No corresponding Refer instance.:无对应Refer实例

错误码	错误描述
419100014	No corresponding DIM instance.:无对应DIM实例
419100015	Failed to create header.:创建头域失败
419100016	Failed to parse header.:解析头域失败
419100017	Header not found.:没有找到头域
419100018	No SDP message body.:没有sdp消息体
419100019	Failed to get capability.:获取能力失败
419100020	Failed to compare capability.:比较能力失败
419100021	Failed to open logical channel.:打开逻辑通道失败
419100022	Encoding SDP failed.:编码SDP失败
419100023	Decoding SDP failed.:解码SDP失败
419100024	Negotiate to unsupported SDP.:协商到不支持的SDP
419100025	TPTD initialization failed.:TPTD初始化失败
419100026	TPT initialization failed.:TPT初始化失败
419100027	TXN initialization failed.:TXN初始化失败
419100028	UA initialization failed.:UA初始化失败
419100029	GLUE initialization failed.:GLUE初始化失败
419100030	HLLM initialization failed.:HLLM初始化失败
419100031	VCOM initialization failed.:VCOM初始化失败
419100032	SDP initialization failed.:SDP初始化失败
419100033	Multiple instance error.:多实例错误
419100034	Insufficient registration control blocks.:注册控制块不足
419100035	Insufficient subscription control blocks.:订阅控制块不足
419100036	Insufficient call control blocks.:通话控制块不足
419100037	Insufficient IM control blocks.:对话外控制块不足
419100038	Failed to parse URI xml to string.:解析xml的uri失败
419100039	Encryption negotiation failed.:加密协商失败
419100040	Local capabilities are not configured.:没有设置本地能力
419100041	BFCP token error.:bfcpl令牌错误
419100042	DNS resolution error.:dns解析错误
419100043	Failed to get subscription state.:获取订阅状态失败

错误码	错误描述
419100044	IPv4 is not supported.:不支持IPv4
419100045	IPv6 is not supported.:不支持IPv6
419100046	Unsupported host type.:不支持的主机类型
419100047	Failed to get timer.:获取定时器失败
420000001	Cannot receive sharing invitations.:无法接收共享邀请
420000002	ReportDTO parameter verification failed.:reportDTO参数校验失败
420000003	Report folder creation failed.:举报文件夹创建失败
420000004	Report folder compression failure.:举报文件夹压缩失败
420000005	The path of the ZIP file to be uploaded is empty.:待上传的ZIP文件路径为空
420000006	The size of the ZIP file to be uploaded reaches the maximum.:待上传的ZIP文件大小超过限制
420000007	The same person reported more than 100 times within 24 hours.:同一个人24小时内举报次数超过100次
430100001	Common Error.:通用错误
430100002	Invalid parameters.:参数错误
430100003	SIP connection check failed.:SIP连接检查失败
430100004	Timeout.:超时
430100005	In process.:接口正在处理中
430100006	Not initialized or initialization is not complete.:未初始化或初始化尚未完成。
430100007	Not supported in this release.:此版本不支持。
430100008	Operation interrupt.:操作中断。
430100009	Contact not found.:联系人不存在
430100010	Args length overrun.:参数长度超限
430100011	APPID info illegal.:APPID 信息入参不合法
430100012	E2EEncrypt algorithm error.:端到端加密算法不对称
430200001	unlogin.:未登录
430200002	Repeated login.:重复登录
430200003	SIP connection timeout.:SIP连接超时
430200004	Invalid login password.:用于登录的密码参数不合法

错误码	错误描述
430200005	Password or account empty.:账号或密码不能都为空
430300001	The calling number, account number or third-party account number cannot be empty.:呼叫号码、账号或者第三方账号不能为空
430300002	Call account empty.:点对点呼叫账号为空
430300003	Call nickname empty.:点对点呼叫名字为空
430400001	Password required to join the meeting.:加入会议需要密码
430400002	Time zone error.:时区错误
430400003	The meeting duration is less than 1.:会议时长小于1
430400004	Joining the conference is interrupted.:入会被中断
430400005	Password input at least 4 digits.:密码输入至少4位
430400006	User voluntarily cancels.:用户主动取消
430400007	The guest password illegal.:来宾密码非法
430400008	Not in conference or in call.:不在会议或呼叫中
430400009	Already exist conf or call.:已存在一路会议或呼叫
430400010	Confid empty.:会议id不能为空
430400011	Subject too long.:会议主题长度不能超过50
430400012	VMR host password length illegal.:主持人密码长度不合法
430400013	VMR name empty.:VMR名字为空
430400014	Conf starttime early.:会议的开始时间早于当前时间
430400015	Share is locked.:共享被锁定
430400016	Not in data conference.:不在数据会议中
430400017	Role not support operation:角色不支持该操作
430500001	HTTP request, server exception.:HTTP请求，服务器异常
430500002	HTTP request, Connecting to the server times out.:HTTP请求，连接服务器超时
430500003	network unavailable.:网络不可用
430500004	file type illegal.:文件类型不合法
510000403	Server forbidden.:服务器拒绝
510000404	Not found.:未发现
510000480	Temporary unavailable.:临时失效

错误码	错误描述
510000486	Busy here.:这里忙
510000487	Request terminated.:请求终止
510000488	Request is not acceptable here.:这里请求不可接受
510000603	Decline.:丢弃
510000841	Hangup without toast.:挂断不弹框
510000842	Conference is locked.:会议被锁定
910000001	Server error.:服务器异常
910000002	SFU service failure.:sfu服务故障
910000003	Service unavaliable 503.:服务不可达503
910000004	Internal error.:内部错误
910000005	Kicked off.:被踢
910000006	Signature expired.:签名过期
910000007	Reconnect timeout.:重连超时
910000008	Network detection.:网络检测, UI不需要关注该错误码, 不对外体现
910000009	User is removed.:用户移除
910000010	The room is dismissed.:房间解散
990000001	Internal error.:内部错误
990000002	Message too large.:消息长度太大
990000003	Insufficient memory.:内存不足
990000004	Failed to send message synchronously.:同步发送消息失败
990000005	Parameter error.:参数错误
990000006	Call ID order error.:呼叫ID时序错误
990000007	Failed to set local screen.:设置本端画面失败
990000008	Failed to open remote screen.:开启远端画面失败
990000009	Device configuration failed.:设备配置失败
990000010	Initializing.:初始化中
990000011	Deinitializing.:去初始化中
990000012	Uploading log.:正在日志上传中
990000013	Failed to get media port.:获取媒体端口失败

错误码	错误描述
990000014	The maximum number view window reached .:选看窗口超过规格数
990000015	Media negotiation failed.:媒体协商失败
990000016	Server no response.:服务器没有响应
990000017	Role switch failed.:角色切换失败
990000018	Failed to join room.:加入房间失败
990000019	Failed to join room. Status is busy.:非空闲状态
990000020	Failed to join room. Server error.:加入房间失败, 服务器异常
990000021	Failed to join room. Service unavaliable.:加入房间失败, 服务不可达
990000022	Failed to join room. Authentication failed.:加入房间失败, 鉴权失败
990000023	Failed to join room. Authentication retry.:加入房间失败, 鉴权重试
990000024	Failed to join room. Clock synchronization.:加入房间失败, 时钟同步
990000025	Failed to join room. URL error.:加入房间失败, url错误
990000026	Kicked off.:被踢
990000027	Share failed.:共享失败
990000028	Currently in external media output mode, this operation is forbidden.:当前为外部媒体输出模式, 禁用该操作
990000029	Reconnection failed.:重连失败
990000030	Server break down.:服务器宕机
990000031	Signature has expired.:签名已过期
990000032	Failed to set remote window mode.:设置远端窗口模式失败
990000033	Failed to subscribe or unsubscribe audio.:订阅或取消订阅音频失败
990000034	Failed to connect to other room.:连接其他房间失败
990000035	Failed to disconnect other room.:断开连接其他房间失败
990000036	Role switching is not allowed.:不允许角色切换
990000037	Currently in third-party capture mode, this operation is forbidden.:当前为第三方采集模式, 禁用该操作
990000038	Failed to set third-party audio capture.:设置第三方音频采集失败

错误码	错误描述
990000039	Failed to set third-party video capture.:设置第三方视频采集失败
990000040	Failed to set share sound switch.:设置共享声音开关失败
990000041	Failed to start or stop upstream audio stream.:启停上行音频流失败
990000042	Failed to start or stop upstream video stream.:启停上行视频流失败
990000043	User is removed.:用户被移除
990000044	The room is dismissed.:房间被解散
990000045	Failed to set remote View.:设置远端View失败
990000052	Failed to join room because room full.:房间已满，加入失败

6.8 版本更新内容

6.8.1 80.13.6 更新内容

1. 编译环境升级到XCode14。
2. 问题修复，体验更稳定。

6.8.2 80.13.5 更新内容

1. 子线程初始化SDK。具体请见[子线程初始化SDK](#)。
2. 增加与会者列表sheetAction，点击头像进入详情。具体请见[buildParticipantActionSheetItems头像点击事件自定义](#)。
3. 问题修复，体验更稳定。

6.8.3 80.12.6 更新内容

1. 开启功能接口enableFeature:isEnabled: 新增是否显示录制安全提示窗枚举值HWMFeatureTypeRecoedAlert。具体请见[枚举HWMEnableFeatureType](#)。
2. 问题修复，体验更稳定。

6.8.4 80.12.5 更新内容

1. 新增HWMRMedia.framework库。具体请见[快速入门](#)相关，使用方法和HWMUISDK.framework库类似。
2. 上传个人头像上传条件变更（增加类型和大小判断）。具体请见[上传个人头像](#)注意事项。
3. 问题修复，体验更稳定。

6.8.5 80.10.5 更新内容

1. 获取云会议列表（HWMVmrInfoModel）和修改云会议室信息（HWMModifyVMRParam）新增字段confAllowJoinUser允许入会范围限制、isOpenWaitingRoom等候室开启开关。具体请见[HWMVmrInfoModel参数说明](#)、[HWMModifyVMRParam参数说明](#)。
2. 新增设置共享权限接口setShareScreenPermission。具体请见[设置共享权限](#)。
3. 问题修复，体验更稳定。

6.8.6 80.9.5 更新内容

1. 编辑会议HWMOrderConfParam.h新增字段isLiveOn。具体请见[HWMOrderConfParam参数说明](#)。
2. 开启功能接口enableFeature:isEnabled: 新增分组讨论枚举值HWMFeatureTypeBreakoutConf。具体请见[开启功能](#)。
3. 取消周期子会议CancelSubCycleConf中字段subConfUUID改成subConfID。具体请见[HWMSDKCancelSubCycleConfParamModel参数说明](#)。
4. 问题修复，体验更稳定

6.8.7 80.8.5 更新内容

1. 创建会议、预约会议、编辑会议支持设置来宾密码。具体请见[HWMCreateConfParam参数说明](#)、[HWMOrderConfParam参数说明](#)。
2. 问题修复，体验更稳定

6.8.8 80.7.5 更新内容

1. App支持自定义多语言
2. 问题修复，体验更稳定

6.8.9 80.5.11 更新内容

1. 新增禁用提示音接口，具体请见[禁用提示音](#)
2. 新增录制按钮操作定制，具体请见[录制按钮操作定制](#)

6.8.10 80.5.5 更新内容

1. 预约会议、编辑会议、获取会议详情接口增加等候室开关功能，具体请见[HWMCreateConfParam参数说明](#)、[HWMOrderConfParam参数说明](#)
2. 问题修复，体验更稳定

6.8.11 70.16.5 更新内容

1. 新增周期会议相关接口，具体请见[预约周期会议](#)、[编辑周期会议](#)、[取消周期会议](#)、[编辑周期子会议](#)、[取消周期子会议](#)
2. 新增设置网络模式接口，具体请见[网络模式设置](#)
3. 新增音频共享，具体请见[SampleHandler文件修改](#)
4. id入会HWMJoinConfParam.h 新增字段isStopConflictConf，具体请见[HWMJoinConfParam参数说明](#)

5. 初始化SDK (HWMOpenSDKConfig) 新增字段 (disableIncomingLocalNotification) 会议来电时禁止使用本地推送, 具体请见[初始化信息HWMOpenSDKConfig](#)
6. sso 登录接口调整, 具体请见[loginBySSO](#):
7. 错误码整改, 新版本错误码描述请见 [新旧版本错误码对应关系](#)

表 6-97 新旧版本错误码对应关系

70.14.6 版本及之前的错误码	70.16.5 版本之后的错误码	新版本错误描述
1002	430400007	The guest password illegal.:来宾密码非
1004	430400002	Time zone error.:时区错误
1005	430400003	The meeting duration is less than 1.:会议时长小于1
1006	430400004	Joining the conference is interrupted.:入会被中断

6.8.12 70.14.5 更新内容

1. 预约、编辑会议支持设置“来宾在主持人之前加入会议”，并可设置提前入会时间 (allowGuestStartConf、allowGuestStartConfTime)，具体参数请见[HWMOrderConfParam参数说明](#)。
2. 预约云会议室时支持选择预约固定云会议室ID或随机云会议室ID，具体请见[VmridType枚举说明](#)。
3. APPID登录支持指定部门ID，具体请见 [AppldAuthParam参数说明](#)。

6.8.13 70.13.5 更新内容

1. login 和 logout 相关接口的回调中增加用户的userUuid (原来uuid字段)、account、thirdAccount, , 详见[登录返回HWMLoginResult](#)、[退出返回HWMLogoutResult](#)。如果是老版本升级到70.13.5版本,需要修改回调函数的定义。
2. 支持设置是否显示外部标签字段hideExternalLabel增加场景 (转移主持人、聊天), 详见[初始化信息HWMOpenSDKConfig](#)
3. 新增被邀加入会议来电通知, 详见[被邀加入会议来电通知](#)

6.8.14 70.12.6 更新内容

1. 修复由于弱网导致视频清晰度下降, 网络恢复后视频清晰度无法恢复的问题。

6.8.15 70.12.4 更新内容

问题修复, 体验更稳定

6.8.16 70.11.5 更新内容

1. 初始化接口参数新增字段：logKeepDays 日志保存天数，详见[初始化信息](#)
[HWMOpenSDKConfig](#)
2. UISDK iPhone版本最低支持iOS10.0
3. HWRTCEngine.framework更名为HWMMREngine.framework(只修改包名，集成方式不变)，详见[HWMMREngine.framework更名](#)

6.8.17 70.10.5 更新内容

1. 接收共享桌面时支持大小画面切换
2. 问题修复，体验更稳定

6.8.18 70.9.5 更新内容

- 1.新增SSO登录方式，具体请见[SSO登录](#)

6.8.19 70.8.6 更新内容

问题修复，体验更稳定

6.8.20 70.8.5 更新内容

1. 支持获取与会者列表（含网络研讨会），详见[获取与会者列表](#)

6.8.21 70.7.5 更新内容

1. 支持视频镜像能力，打开后，观看自己显示为镜像效果

图 6-51 视频镜像



6.8.22 70.6.5 更新内容

1. 新增结束会议的状态通知，具体使用请见[会议结束通知](#)

6.8.23 70.5.8 更新内容

1. 会中与参会者和聊天模块适配横屏体验
2. 支持聊天弹幕

6.8.24 70.4.5 更新内容

1. 新增获取用户当前状态接口，具体使用请见[获取用户状态](#)
2. 会中新增主持人邀请他人共享，主持人也可快速取消共享
3. 移动端的前后摄像头切换按钮放到主界面上
4. 会中界面优化
5. 会议日志大小优化

6.8.25 70.3.5 更新内容

1. 观看共享时叠加发言人视频窗口，并支持下拉关闭
2. 支持手机、平板同时登录，具体使用请见[登录](#)

6.8.26 60.15.5 更新内容

1. 会中自定义会议页面顶部titlebar优化（支持自定义cell），具体使用请见[会议页面顶部titlebar菜单定制](#)
2. 新增支持隐藏与会者列表外部标签，初始化设置配置参数hideExternalLabel，默认显示
3. 新增设置来电响铃和震动，具体使用请见[来电通知](#)
4. 新增呼叫和会议来电时接听策略设置（自动接听、自动拒接、默认），具体使用请见[来电接听策略设置](#)

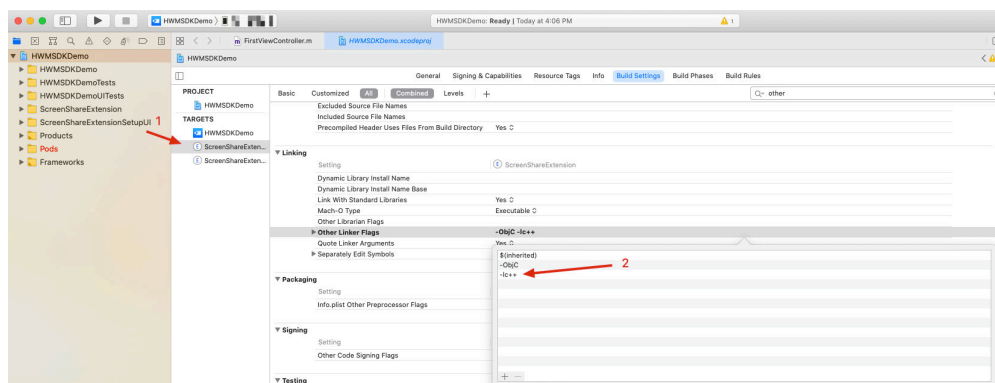
6.8.27 60.13.2 更新内容

1. 添加与会者，具体使用请见[添加与会者](#)
2. 离开会议，具体使用请见[离开会议](#)
3. 结束会议，具体使用请见[结束会议](#)

6.8.28 60.12.5 更新内容

1. Extension添加配置“TARGETS—>ScreenShareExtension—>Build Settings—>Other Linker Flags”中添加-lc++。

图 6-52 添加配置截图



2. 创建会议支持指定会议室（vmrId）和入会范围（joinConfRestrictionType），具体使用请见[HWMCreateConfParam参数说明](#)
3. 新增会中会议详情上报，具体使用请见[会议详情通知](#)
4. 新增头像上传接口，具体使用请见[上传个人头像](#)
5. 新增头像拦截接口，可以使用自己的头像显示，具体使用请见[头像拦截显示](#)
6. 预约会议接口在result中返回会议详情信息，具体使用请见[result中返回详情](#)

6.8.29 60.11.6 更新内容

1. 新增会议支持自动录制，新增允许自动录制会议isAutoRecord字段，
注意：会议开始后是否自动开启录制，默认关闭。开启自动录制会议时必须开启录制会议
 - 创建会议[HWMCreateConfParam参数说明](#)
 - 预约会议、编辑会议[HWMOrderConfParam参数说明](#)
2. 新增自定义会议来电接听按钮显示，具体使用请见[自定义视频会议来电时接听方式按钮](#)
3. 新增SDK库：HWRTCEngine.framework，具体使用请见[新增HWRTCEngine库](#)

6.8.30 60.10.5 更新内容

1. 初始化增加登录站点类型，具体使用请见[初始化](#)
2. 支持使用random入会，具体使用请见[加入会议](#)
3. 增加sp鉴权登录，增加字段corpId。具体使用请见[App ID鉴权登录](#)
4. 优化水印显示，具体使用请见[screenShareCoverView](#)

6.8.31 60.8.1 更新内容

1. 初始化增加AppID，优化App区分
2. 新增AppId登录、预约会议、编辑会议、会议列表、会议详情、取消会议等接口
 - 新增接口

```

/// 获取会议列表
- (NSArray<HWMBaseConfDetailParam *> *)getConfList;
/// 获取会议详情
- (void)getConfDetail:(NSString *)confId completeHandler:
(HWMSDKConfInfoCompleteHandler)handler;
/// 预约会议
- (void)bookConf:(HWMOrderConfParam *_Nonnull)param callback:(_Nonnull
HWMSDKCompleteHandler)callback;
/// 取消会议
- (void)cancelConf:(NSString *_Nonnull)confId callback:(_Nonnull HWMSDKCompleteHandler)callback;
/// 编辑会议
- (void)editConf:(HWMOrderConfParam *_Nonnull)param callback:(_Nonnull
HWMSDKCompleteHandler)callback;
/// 获取VMR ID列表，预约会议时可从次列表中选择固定会议ID进行预约
- (NSArray<HWMVmrInfoModel *> *)getVmrList;
/// 修改vmr会议室信息
- (void)changeVmrInfo:(HWMModifyVMRParam *)info callback:(_Nonnull
HWMSDKCompleteHandler)callback;

```

3. 语音会议界面调整
语音会议会中界面调整。
4. IMSDK.framework合入到UISDK.framework

6.8.32 60.7.1 更新内容

1. 会议界面UI界面优化、位置调整
 - 会议界面-顶部会议信息增加会议锁定显示，显示优化
 - 会议界面-“更多”菜单调整，部分功能移动到“更多”-“会议设置”中
2. 会控菜单自定义接口优化，部分接口替换

以下为接口示例，具体使用请见[接口参考](#)

- 新增接口

```
/// 会议设置菜单栏  
- (NSArray <HWMConfSettingGroup *> *)buildSettingItems;  
/// 构建会议titleBar 点击的弹出菜单  
- (NSArray <HWMInMeetingCellModel *> *)buildTitleBarPopViewItems;  
/// 屏幕共享/白板共享的封面视图，可放水印等。注意不要添加全局手势  
- (UIView *)screenShareCoverView;
```

- 替换接口

```
/// 构建音频呼叫和音频会议底部toolbar的会控全量菜单  
- (NSArray <HWMConfToolBarMenuItem *> *)buildAudioCallAndConfToolBarMenuItems;  
/// 构建视频呼叫和视频会议底部toolbar的会控全量菜单  
- (NSArray <HWMConfToolBarMenuItem *> *)buildVideoCallAndConfToolBarMenuItems;  
///构建点对点视频通话底部toolbar上的"更多"菜单  
- (NSArray <HWMConfMenuItem *> *)buildVideoCallToolBarMoreMenuItems;  
/// 构建视频会议底部toolbar上的“更多”菜单  
- (NSArray <HWMConfMenuItem *> *)buildVideoConfToolBarMoreMenuItems;
```

接口合并替换为

```
/// 构建底部toolbar的会控全量菜单  
- (NSArray <HWMConfToolBarMenuItem *> *)buildToolBarMenuItems;  
/// 构建底部toolbar的“更多”菜单  
- (NSArray <HWMConfMenuItem *> *)buildToolBarMoreMenuItems;
```

- 删除接口

```
/// 构建音频呼叫和音频会议底部toolbar的会控全量菜单  
- (NSArray <HWMConfToolBarMenuItem *> *)buildAudioCallAndConfToolBarMenuItems;  
/// 构建视频呼叫和视频会议底部toolbar的会控全量菜单  
- (NSArray <HWMConfToolBarMenuItem *> *)buildVideoCallAndConfToolBarMenuItems;  
///构建点对点视频通话底部toolbar上的"更多"菜单  
- (NSArray <HWMConfMenuItem *> *)buildVideoCallToolBarMoreMenuItems;  
/// 构建视频会议底部toolbar上的“更多”菜单  
- (NSArray <HWMConfMenuItem *> *)buildVideoConfToolBarMoreMenuItems;
```


7 Windows SDK

7.1 概述

华为云会议终端Windows SDK提供了一套完整的接口集合，开发者可以通过调用SDK开放的API，在自己的应用程序中快速集成音视频会议，包括创建/预约/加入会议、进行屏幕共享/程序共享/白板标注、进行全场静音等功能。

7.2 修订记录

表 7-1 修订记录

日期	版本	修订内容
2023-2-22	80.13.6	1.问题修复，体验更稳定 具体使用请见 80.13.6更新内容
2023-2-7	80.13.5	1.新增设置语言代码接口 SetLanguageWithCode 具体使用请见 80.13.5更新内容
2023-1-17	80.12.7	1.问题修复，体验更稳定 具体使用请见 80.12.7更新内容
2023-1-9	80.12.6	1.会议相关功能配置开关接口 EnableFeature新增录制确认开关枚举值 HWM_ENABLE_FEATURE_CONFIRM_RECORD 具体使用请见 80.12.6更新内容

日期	版本	修订内容
2022-12-23	80.12.5	<p>1.新增上传头像接口UploadSelfAvatar</p> <p>2.新增会议是否共享锁定状态变化通知OnConflsShareLockedChanged</p> <p>3.企业配置通知OnCorpConfig中HwmCorpConfigInfo新增字段 是否支持通讯录显示 enableShowAddressBookStructure</p> <p>4.新增会议结束原因错误码，达到系统单个会议人数上限 111072504 HWM_CONF_END_REASON_PARTICIPANT_REACH_MAX_NUM</p> <p>具体使用请见80.12.5更新内容</p>
2022-11-16	80.10.5	<p>1.修改云会议室信息ChangeVmrlInfoParam和云会议室信息HwmVmrlInfo新增字段 joinConfRestrictionType允许入会范围限制、isOpenWaitingRoom等候室开启开关</p> <p>2.新增设置共享权限接口 SetShareScreenPermission</p> <p>具体使用请见80.10.5更新内容</p>
2022-10-8	80.9.5	<p>1.取消周期子会议CancelSubCycleConf中字段subConfUUID改成subConfID</p> <p>2.预约周期会议BookCycleConf和编辑周期会议ModifyCycleConf、周期会议详情中 cycleCount字段废弃</p> <p>3.会议相关功能配置开关接口 EnableFeature新增分组讨论功能开关枚举值 HWM_ENABLE_FEATURE_BREAKOUT_CONF</p> <p>4.会议信息HwmConfInfo新增字段是否在讨论组中isInBreakoutSubConf</p> <p>具体使用请见80.9.5更新内容</p>
2022-9-2	80.8.5	<p>1.错误码细化，以及新增错误码10011鉴权信息过期</p> <p>2.OnConfInfo接口参数HwmConfInfo新增字段isCreator，是否是会议预订者</p> <p>3.会议详情HwmConfDetail和编辑会议HwmEditConfParam新增字段isLiveOn</p> <p>具体使用请见80.8.5更新内容</p>

日期	版本	修订内容
2022-7-30	80.7.5	<p>1.新增显示设置界面接口 ShowSettingWindow</p> <p>2.Config接口新增字段 moveall_to_waitingroom, 支持全员移入等候室可隐藏</p> <p>3.Config接口新增字段participant_detail, 支持与会者头像点击注入, 点击注入OnClickInjectBtn通知里面按钮类型里面增加 HWM_CLICK_INJECT_BTN_PARTICIPANT_DETAIL 与会者详情按钮, data信息是 HwmConfAttendee</p> <p>4.HwmSettingWindowConfig 添加 HwmScreenSettingTab 屏幕设置项, 将双屏设置从 HwmNormalSettingTab 移到 HwmScreenSettingTab</p> <p>5.会中信息通知 (OnConfInfo) 中 HwmConfInfo参数新增字段 isOtherCorpConf bool 是否是外部企业的会议</p> <p>6.企业信息变更通知 (OnUserCorpInfo) 中HwmUserCorpInfoNotifyType参数新增枚举值 HWM_USER_CORP_INFO_NOTIFY_TYPE_USER_NAME_CHANGE 用户名变更和 HWM_USER_CORP_INFO_NOTIFY_TYPE_USER_STATUS_CHANGE 用户状态变化</p> <p>7.会中邀人 (AddAttendee) 接口 HwmConfAttendee参数新增字段 accountId华为云会议帐号</p> <p>8.新增错误码</p> <p>9.增加会议结束原因错误码, 与会方超出专用VMR套餐的最大与会人数 HWM_CONF_END_REASON_NUM_OF_PAR_EXCEEDS_UPPER_LIMIT</p> <p>10.新增通话结束通知OnCallRecordInfo 具体使用请见80.7.5更新内容</p>
2022-5-30	80.5.11	<p>1.Config接口新增字段statusBar, 支持录制状态栏按钮可隐藏</p> <p>2.新增禁用提示音接口 DisableVoicePrompts 具体使用请见80.5.11更新内容</p>

日期	版本	修订内容
2022-5-18	80.5.5	<p>1.会议结束原因HwmConfEndReason新增： HWM_CONF_END_REASON_SERVER_ERROR 服务器异常 具体使用请见80.5.5更新内容</p>
2022-3-25	70.16.5	<p>1.新增周期会议相关接口 2.加入会议接口HwmJoinConfByIdInfo参数新增字段：isStopConflictConf强制结束和当前VMR会议资源冲突的会议 3.Config接口配置自定义按钮，支持增加二级菜单 4.新增错误码 5.Config接口菜单按钮新增 共享画面质量策略：清晰度优先、流畅度优先 "share_strategy"、共享时隐藏视频窗口 "share_hide_video"、共享时包含电脑声音 "share_audio"配置可隐藏。复制会议信息 "copy_conf_info"按钮支持配置可注入 6.会议角色HwmConfRole新增枚举值 HWM_CONF_ROLE_AUDIENCE观众， HWM_CONF_ROLE_COHOST联席主持人和 HWM_CONF_ROLE_WAITING等候室成员 具体使用请见70.16.5 更新内容</p>
2022-2-19	70.14.5	<p>1.新增设置网络模式接口SetNetworkMode 2.登录接口HwmAppIdAuthInfo参数新增 字段：deptCode 部门id 3.预约、编辑会议支持设置“来宾在主持人之前加入会议”，并可设置提前入会时间 具体使用请见70.14.5 更新内容</p>

日期	版本	修订内容
2022-01-19	70.13.5	<p>1.OnLoginResult、OnLoginBySSOResult 回调增加HwmLoginResult参数； OnLogoutResult回调增加 HwmLogoutResult参数</p> <p>2.修改winUI日志后缀名为txt，修改了四个日志命名：hwm_winconfui.txt，hwmSdk.txt，hwmSdkAgent.txt，hwmSdkAgent32.txt</p> <p>3.Config接口中新增配置项： isUploadKeyLog 是否自动收集日志的开关</p> <p>4.Init接口HwmlnitInfo参数新增字段： preferredCamera 优选摄像头列表参数</p> <p>5.新增接口AVDeviceCtrl音视频设备控制和结果回调OnAVDeviceCtrlResult，新增对应通知OnAVDeviceCtrlStateChanged音视频设备状态</p> <p>6.Config接口配置自定义按钮，支持动态刷新底部/共享工具栏</p> <p>具体使用请见70.13.5更新内容</p>
2021-12-23	70.12.6	<p>1. 修复弱网情况下导致清晰度降低后不能恢复的问题</p> <p>2. 修复隐藏投票功能</p> <p>具体使用请见70.12.6更新内容</p>
2021-12-13	70.12.4	<p>1.CreateConf接口HwmCreateConfInfo参数新增选填字段：duration 会议时长</p> <p>2.新增自定义设置界面接口 ConfigSettingWindow</p> <p>3.新增企业信息变更通知OnUserCorpInfo</p> <p>具体使用请见70.12.4更新内容</p>
2021-12-3	70.11.5	<p>1.Init接口HwmlnitInfo参数新增字段： logKeepDays 日志保存天数</p> <p>具体使用请见70.11.5更新内容</p>
2021-10-22	70.10.5	<p>1.Config接口新增配置项： isHideTransferHostBeforeLeave 是否隐藏离会时指定主持人功能按钮</p> <p>具体使用请见70.10.5更新内容</p>
2021-09-16	70.9.5	<p>1.登录新增“SSO登录”接口</p> <p>具体使用请见70.9.5更新内容</p>
2021-08-28	70.8.6	<p>1. 新增“刷新自定义按钮配置”接口</p> <p>2. 修复问题，体验更稳定</p> <p>具体使用请见70.8.6更新内容</p>

日期	版本	修订内容
2021-08-18	70.8.5	1.支持获取与会者列表（含网络研讨会） 2.支持获取音频数据流 具体使用请见 70.8.5更新内容
2021-07-16	70.7.5	1.支持会中设置本会场视频画面是否镜像 2.提供32位&64位两个版本SDK 3.支持会控工具栏的自定义 4.支持会中背景替换和背景虚化 5.支持共享作为广播多画面子画面显示，该模式下视频画面和共享画面可自由切换 6. 新增设置共享策略、设置共享音频接口 具体使用请见 70.7.5更新内容
2021-06-10	70.6.5	1.新增“显示会议主窗口”接口 2.会中一键打开有视频画面，观看更方便 3.设置会议入会范围后，非指定类型用户加入会议或会议锁定时，错误码从原来的加入会议时的回调中上报，改成从OnConfState离会原因中上报 具体使用请见 70.6.5更新内容
2021-05-17	70.5.8	1.优化会中视觉体验 具体使用请见 70.5.8更新内容
2021-04-15	70.4.5	1.预约会议接口添加返回值参数说明 2.新增获取用户当前状态接口 3.支持安全菜单、字幕功能、锁定共享按钮可配置隐藏 4.会中新增主持人邀请他人共享，主持人也可快速取消共享 5.支持阿拉伯语文件路径下安装使用 6.会议日志大小优化 7.优化回调函数实现方式从纯虚函数变更有默认实现的虚函数，可以不实现定义所有的回调函数接口。 具体使用请见 70.4.5更新内容
2021-03-12	70.3.5	1.会中导出与会者信息优化 具体使用请见 70.3.5更新内容

日期	版本	修订内容
2021-02-10	60.15.5	<ol style="list-style-type: none"> 1.初始化回调结果未返回，调用其他任何接口都将返回失败，增加未初始化或初始化尚未完成错误码 HWM_AGENT_ERR_NOT_INITED_END 2.新增与会者列表外部标签是否隐藏功能 3.新增呼叫来电弹窗是否隐藏、会议来电弹窗是否隐藏的功能 4.新增设置来电应答接口 5.新增本地录制功能，支持主持人控制与会者本地录制权限、支持配置录制文件保存路径 6.会中新增“安全”模块，统一管理权限 具体使用请见60.15.5更新内容
2021-01-08	60.13.2	<ol style="list-style-type: none"> 1.一般应用升级前需要杀死进程，为了便于升级管理，支持初始化之前修改HwmSdk文件夹下HwmSdk.exe文件的名称为用户自己的名称 2.初始化接口参数exePath的值从HwmSdk文件夹的绝对路径变更为exe文件的绝对路径，并且要求是utf8格式， 3.初始化接口新增参数logPath和userDataPath，可以指定日志文件和用户数据文件的保存路径 具体使用请见60.13.2更新内容
2020-12-07	60.12.5	<ol style="list-style-type: none"> 1.创建会议支持指定会议室（vmrld）和入会范围（callInRestriction） 2.新增自定义会议信息框 3.新增查询外部通讯录头像，可以在会中关闭摄像头时，显示自定义头像 具体参见60.12.5更新内容
2020-11-07	60.11.1	<ol style="list-style-type: none"> 1.新增支持会议自动录制 2.新增点对点呼叫功能 3.新增点对点呼叫设置昵称 具体参见60.11.6更新内容
2020-10-15	60.10.5	<ol style="list-style-type: none"> 1.初始化增加登录站点类型 2.支持使用random入会 3.增加sp鉴权登录 4.优化水印显示 具体参见60.10.5更新内容

日期	版本	修订内容
2020-09-04	60.8.1	1.增加预约会议相关会管接口 2.增加三方App ID登录场景 具体参见 60.8.1更新内容
2020-06-24	60.6.4	首次发布

7.3 快速入门

本小节以实现初始化SDK为例，介绍如何使用Windows SDK进行二次集成开发。

开发环境准备

在开发的过程中请满足如下环境要求。

表 7-2 环境要求

环境和工具名称	版本要求	说明
操作系统	Windows 10/11专业版	硬件要求： <ul style="list-style-type: none"> • CPU: i5-2400四核 3.1GHz及以上。 • 内存: 4GB及以上。
Microsoft Visual Studio	Visual Studio 2017	-
测试资源及App ID申请	-	请参见“ 开发前准备 ”。

SDK 快速集成

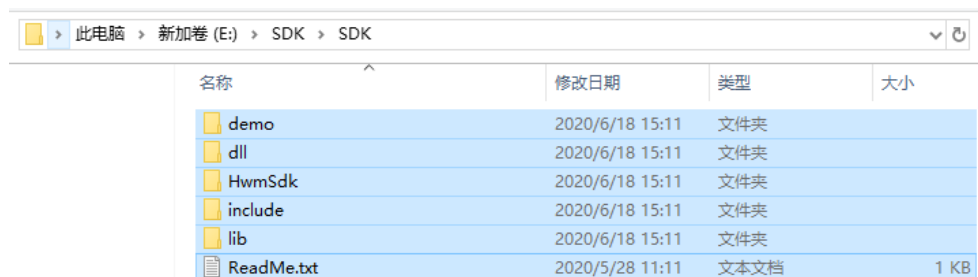
说明

集成SDK的exe需要打开每个监视器高dpi识别开关、否则在高dpi的电脑（缩放比例大于100%）上面会显示异常（图片模糊、共享时接收端观看画面不全、倾斜等。）

示例：[图12 配置“DPI识别功能”](#)

步骤1 解压SDK压缩包到本地。

图 7-1 解压 SDK 压缩包到本地



include: 该文件夹下放置编译所需头文件。

lib: 该文件夹下有win32和x64两个子文件夹, 分别放置对应平台编译所依赖的.lib文件。

dll: 该文件夹下有win32和x64两个子文件夹, 分别放置对应平台运行所依赖的.dll和SDK可执行.exe文件。

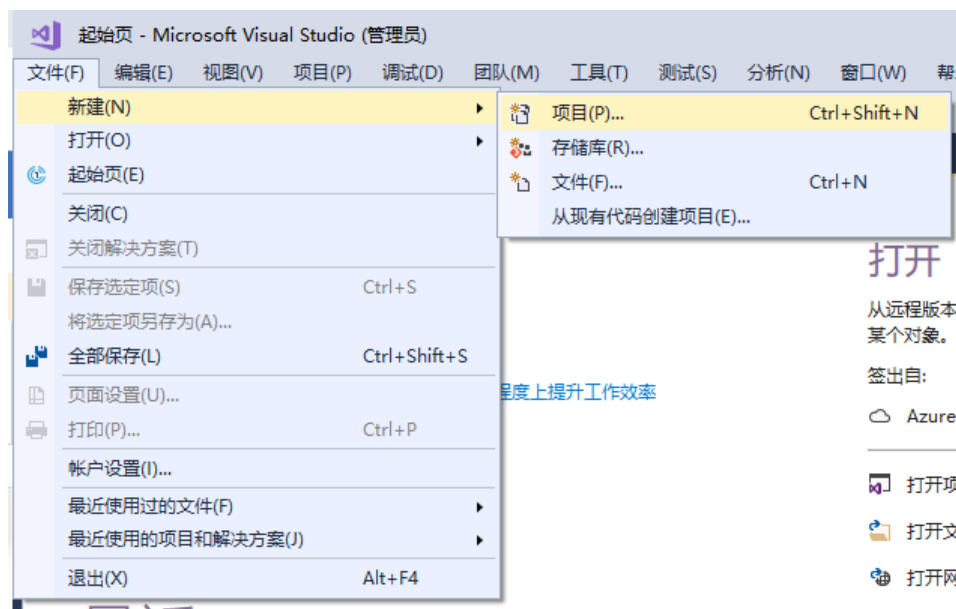
HwmSdk: 该文件夹下放置win32和x64公共使用的依赖库。

demo: 该文件夹下放置demo代码文件与资源文件。

步骤2 创建工程项目。

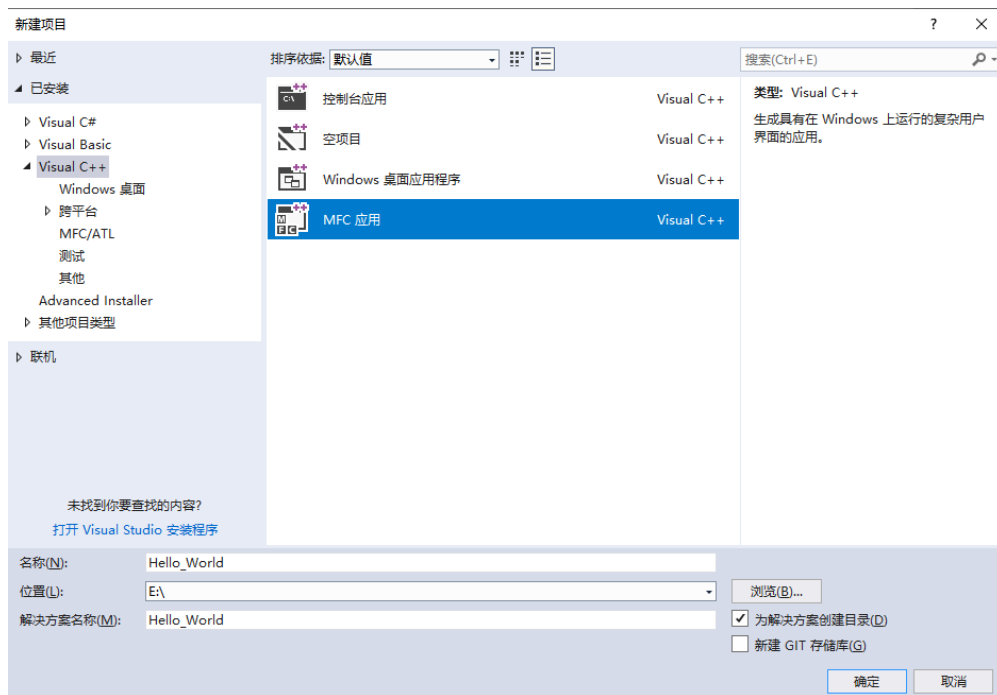
1. 打开Visual Studio 2017, 单击菜单栏的“文件> 新建> 项目”, 打开“新建项目”窗口。

图 7-2 打开新建项目



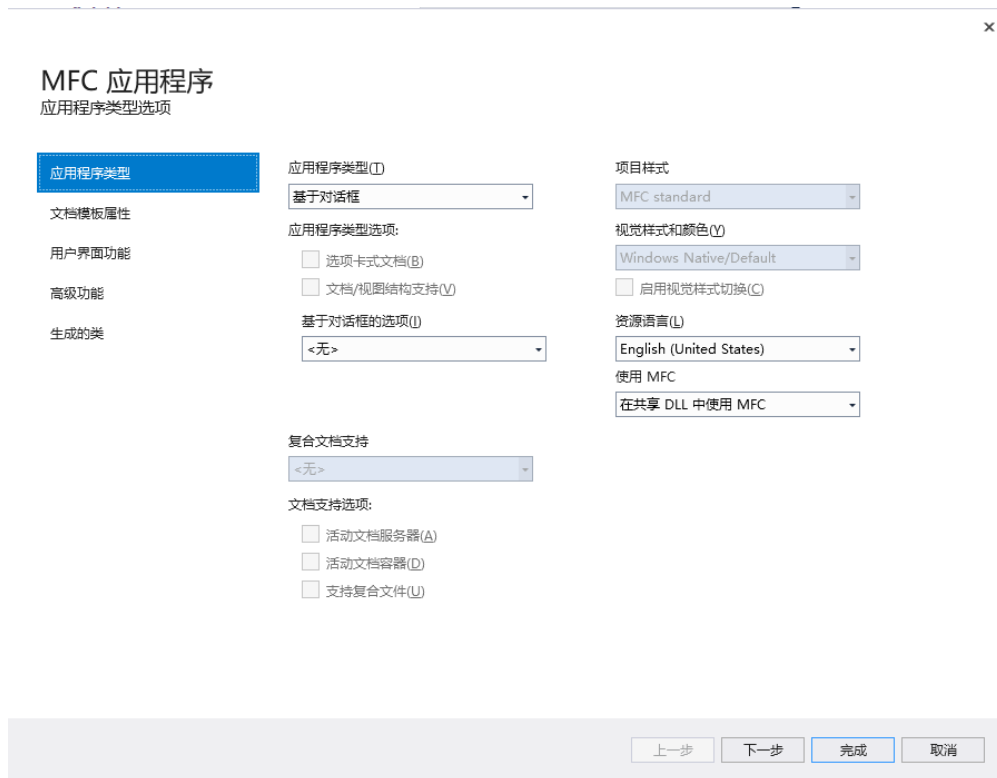
2. 在“新建项目”左侧选择“已安装> Visual C++”, 然后在右侧模板中选择“MFC 应用程序”, 在“名称”输入工程名称“Hello_World”, 在“位置”选择工程保存位置, 此处选择“E:\”, 完成后点击“确定”。

图 7-3 新建项目



3. 在“应用程序类型选项”中选择“应用程序类型”为“基于对话框”，其余设置均保持默认值，点击完成。

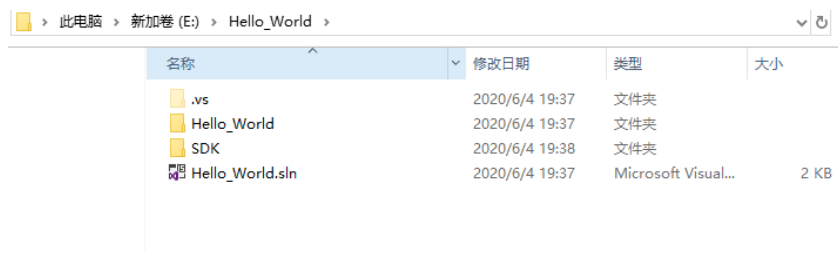
图 7-4 应用程序类型选项



步骤3 拷贝静态库文件与头文件到项目中。

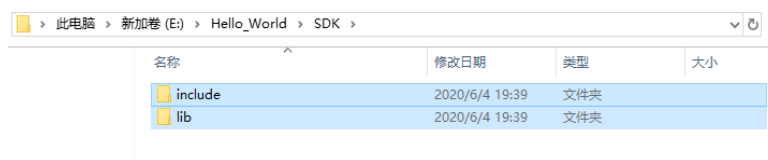
1. 在“Hello_World”工程目录下新建“SDK”目录，与.sln文件同级。

图 7-5 新建“SDK”



2. 将“SDK”目录下的“lib”和“include”目录拷贝到“Hello_World\SDK”目录下。

图 7-6 拷贝文件



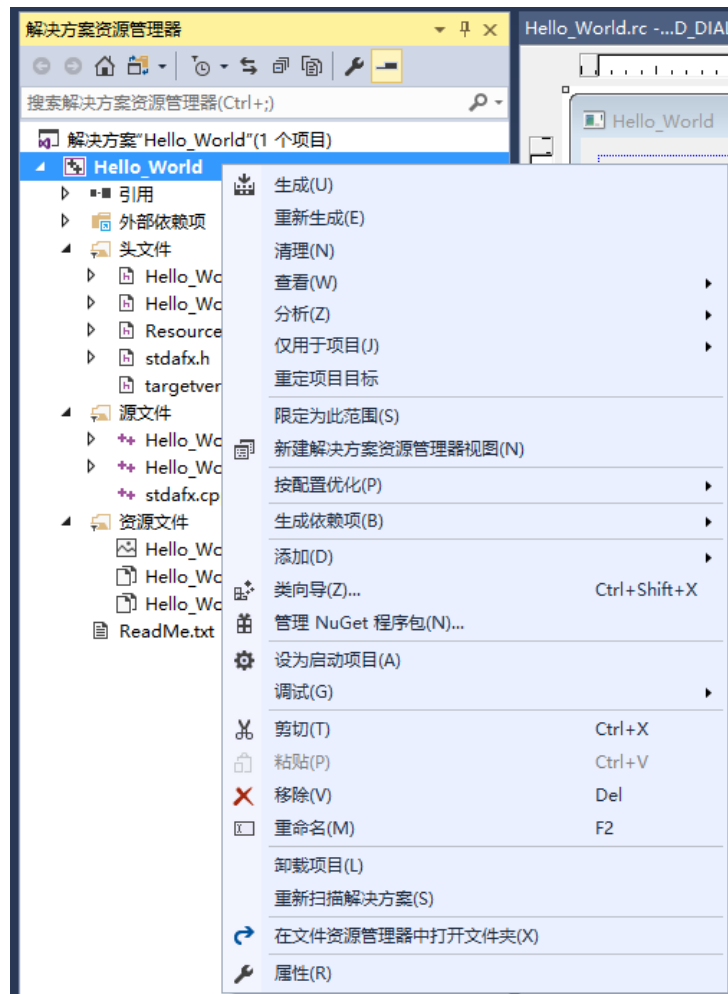
步骤4 修改项目“配置属性”。

📖 说明

以下操作基于项目配置Debug与平台Win32，更改配置和平台时需要重新添加附加包含目录和附加依赖库。

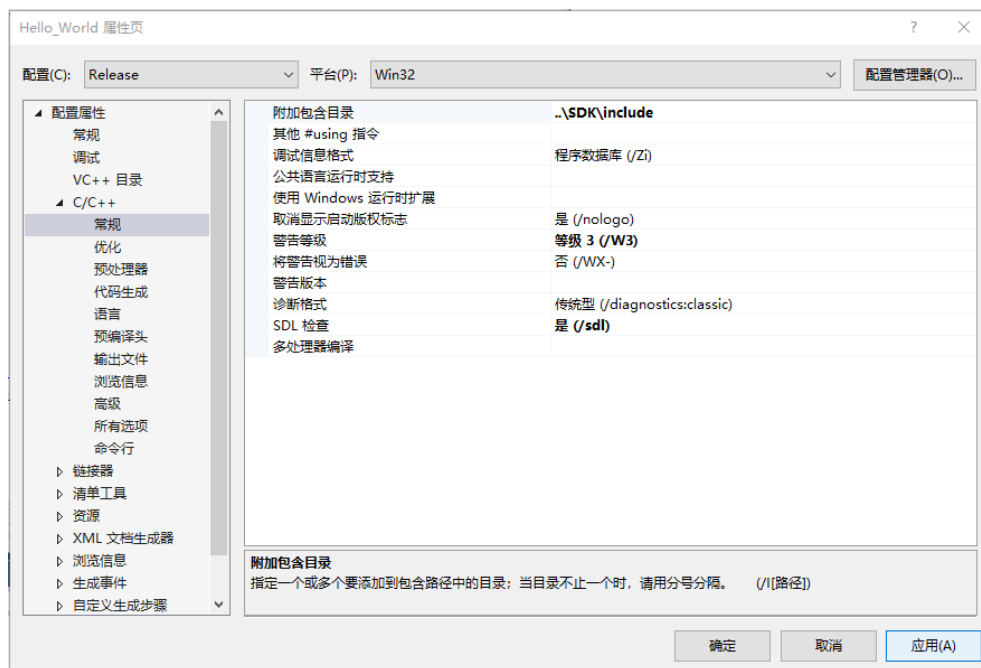
1. 在“Hello_World”项目右键单击“属性”。

图 7-7 项目属性



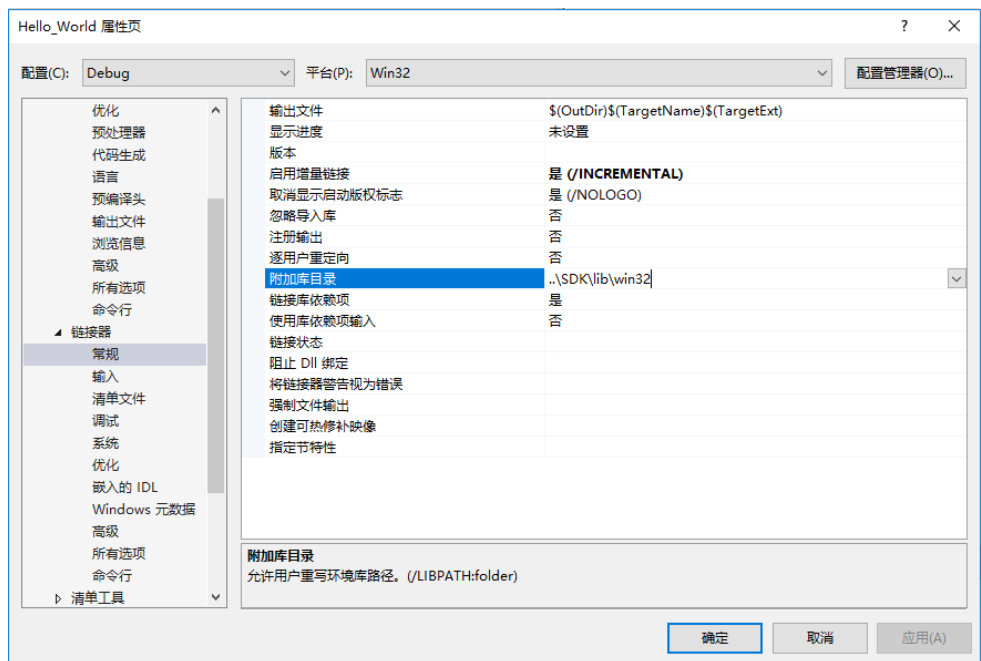
2. 在属性页左侧选择“配置属性 > C/C++ > 常规”，将“附加包含目录”的值修改为“..\SDK\include”，点击“应用”按钮。

图 7-8 配置“附加包含目录”



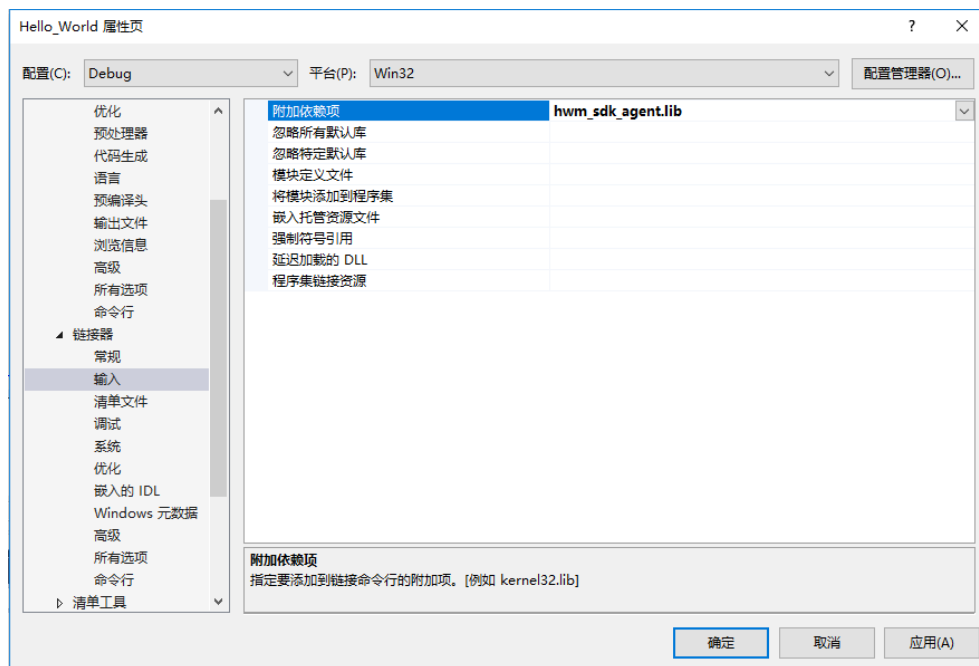
3. 在属性页左侧选择“配置属性 > 链接器 > 常规”，将“附加库目录”的值修改为“..\SDK\lib\win32”，点击“应用”按钮。

图 7-9 配置“附加库目录”



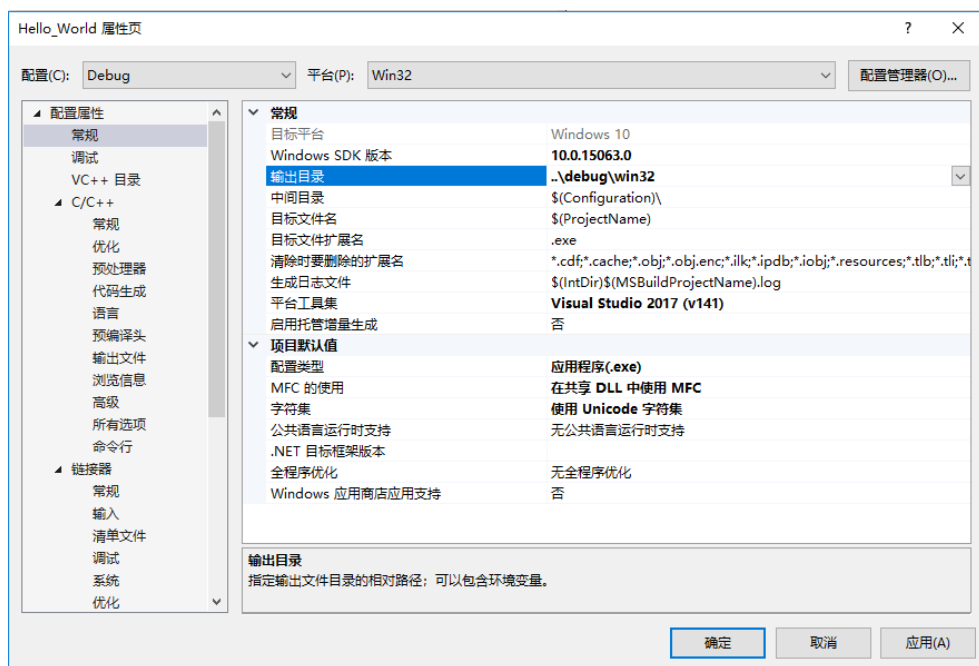
4. 在属性页左侧选择“配置属性 > 链接器 > 输入”，将“附加依赖项”的值修改为“hwm_sdk_agent.lib”，点击“应用”。

图 7-10 添加“附加依赖项”



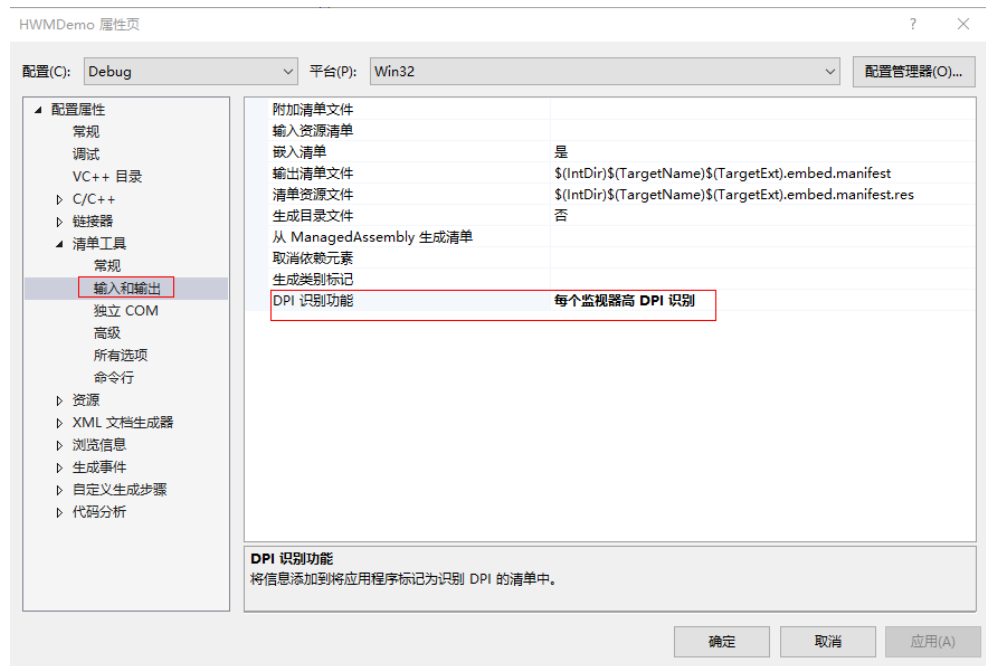
5. 在属性页左侧列表中选择“配置属性 > 常规”，“输出目录”值修改为“..\debug\win32”，点击“确定”按钮。

图 7-11 配置“输出目录”



6. 在属性页左侧列表中选择“配置属性 > 清单工具 > 输入和输出”，“ DPI识别功能”值修改为“每个监视器高 DPI 识别”，点击“确定”按钮。

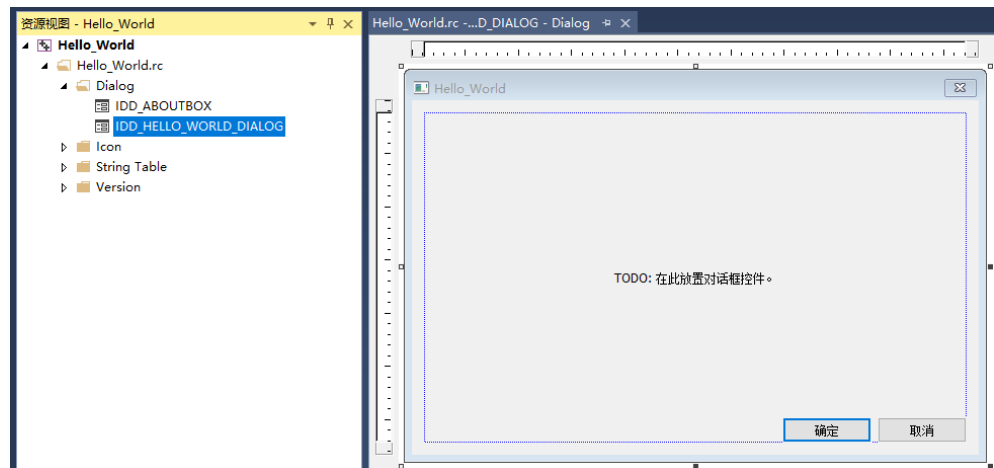
图 7-12 配置 “DPI 识别功能”



步骤5 添加界面资源控件。

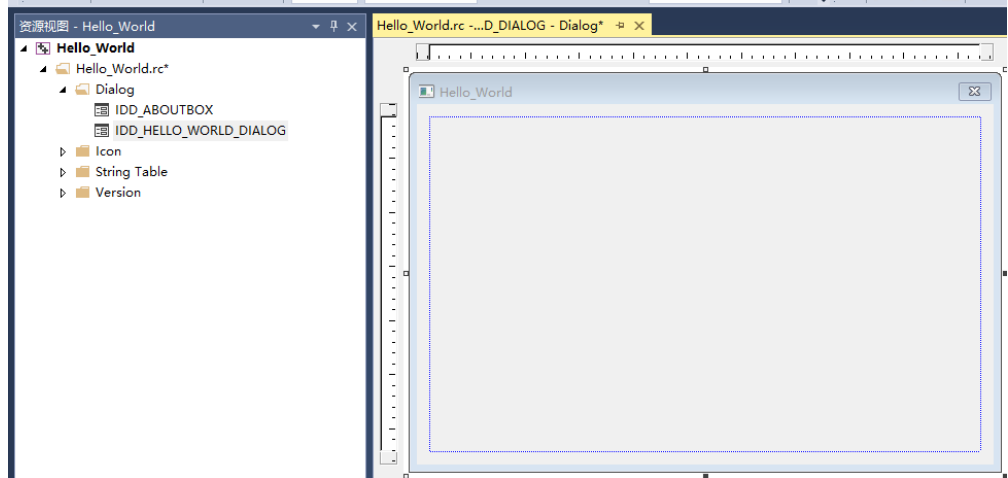
1. 在“资源视图”中，打开工程对应的Dialog页面。

图 7-13 工程 Dialog 页面



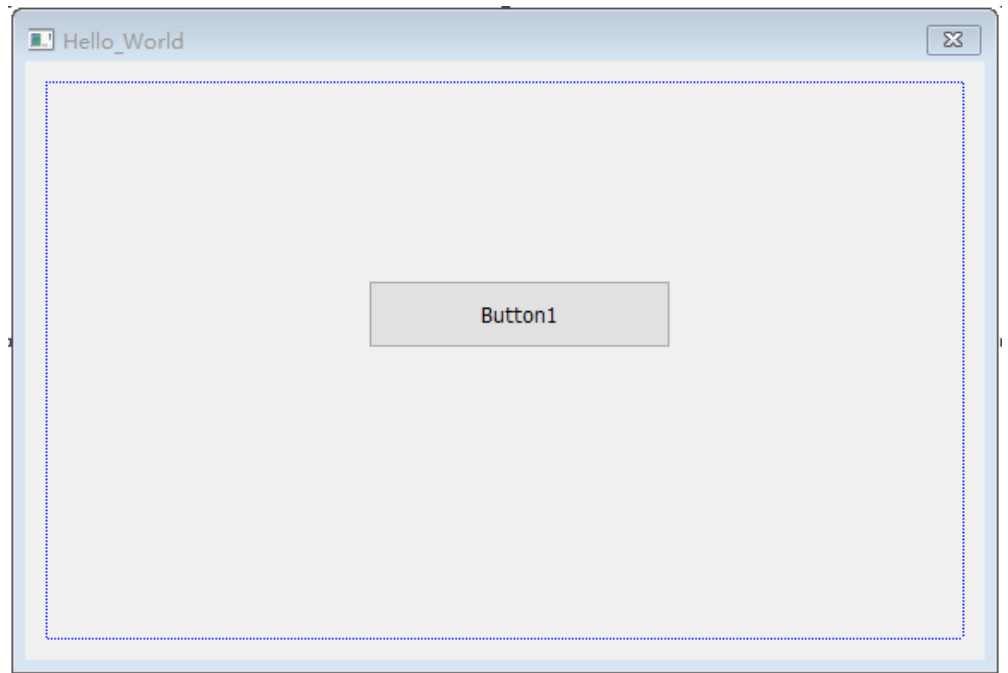
2. 选中页面上原有全部控件，按Delete键删除。

图 7-14 删除原有控件



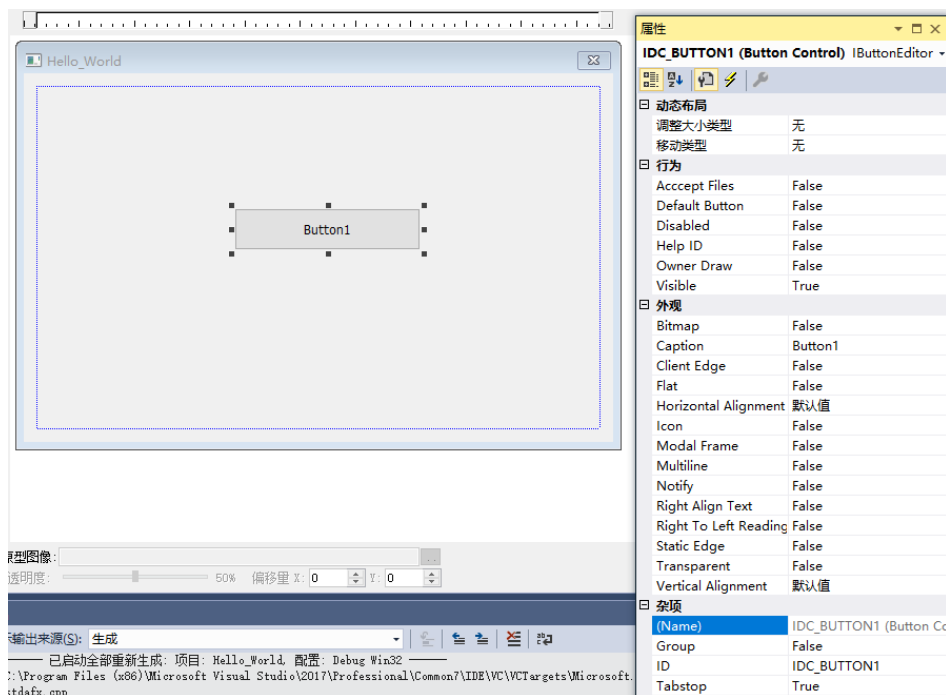
3. 添加初始化按钮。
 - a. 添加初始化按钮控件：打开“工具箱”页面，选中“Button”控件，拖至对话框页面上并调整宽高。

图 7-15 添加初始化按钮



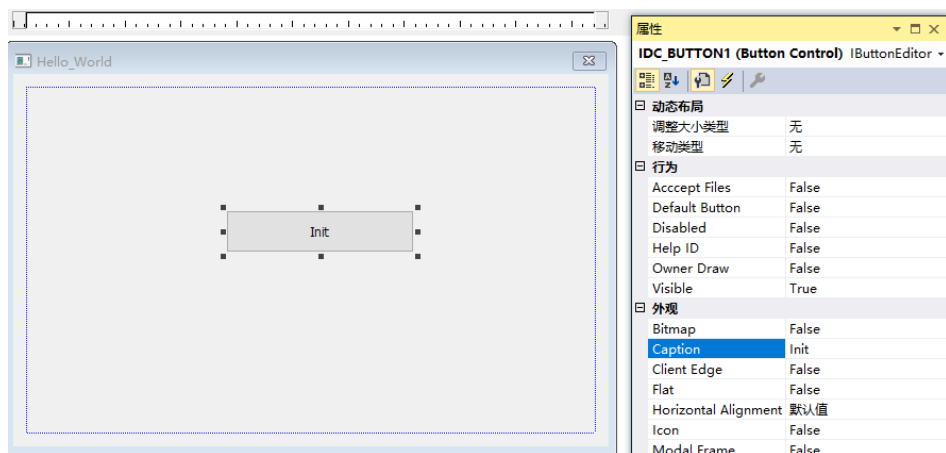
- b. 修改按钮上面的注释：选中按钮，右击选择“属性”。

图 7-16 打开按钮属性



- c. 在对应的属性页上，找到“Caption”项，将值修改为“Init”。

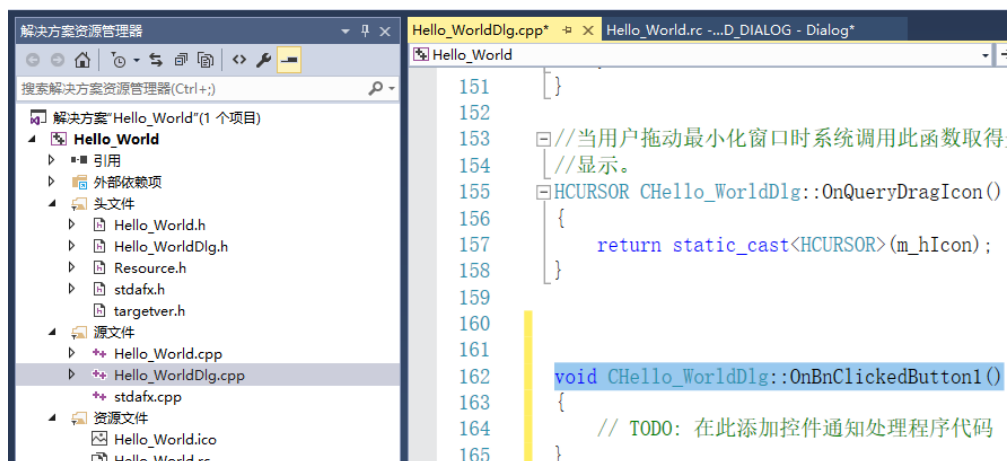
图 7-17 修改按钮属性



步骤6 添加按钮点击事件处理。

1. 双击“Init”控件，系统会自动在Hello_WorldDlg.cpp中添加初始化按钮事件方法 OnBnClickedButton1()。

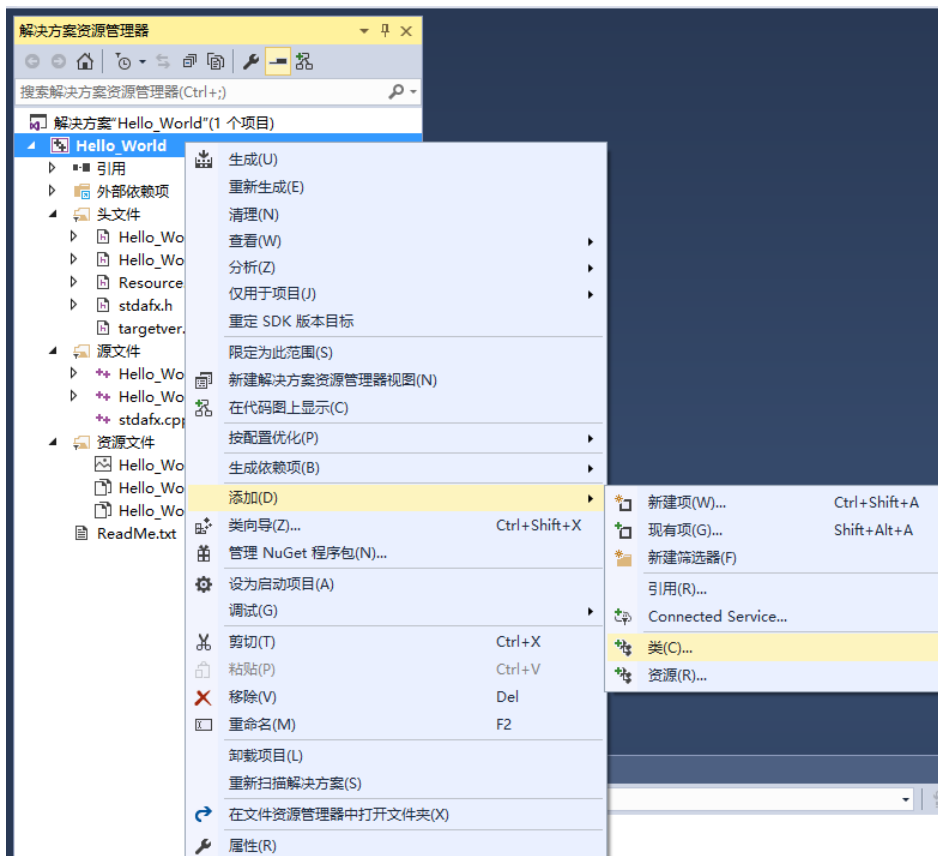
图 7-18 创建初始化按钮事件方法



步骤7 添加代码文件。

1. 添加回调处理类。
 - a. 选中项目右键“添加 > 类”。

图 7-19 添加类



- b. 在“一般C++类向导”中填写类名“callbackProc”，其他保持默认，点击“完成”，保持现有工程。

图 7-20 添加 “callbackProc 类”



2. 添加通知处理类，类名设置为 “notifyProc”。

图 7-21 添加 “notifyProc 类”



步骤8 添加逻辑代码。

1. 在callbackProc.h文件中包含SDK回调类头文件HwmAgentCallback.h，使callbackProc类继承于基类HwmAgentCallback，可以只拷贝实现HwmAgentCallback.h文件中需要关注的接口调用回调函数。

```
#pragma once
#include "HwmAgentCallback.h"

class callbackProc : public hwmsdkagent::HwmAgentCallback
{
public:
    callbackProc();
    ~callbackProc();
    /**
     * @brief [en] Callback of Init interface.
     *
     * @param [in] hwmsdk::HwmErrCode ret [en] Return code
     * @param [in] const char* reason [en] Fail reason
     * @attention [en] NA
     */
    void OnInitResult(hwmsdk::HwmErrCode ret, const char* reason);

    /**
     * @brief [en] Callback of Exit interface.
     *
     * @param [in] hwmsdk::HwmErrCode ret [en] Return code
     * @param [in] const char* reason [en] Fail reason
     * @attention [en] NA
     */
    void OnExitResult(hwmsdk::HwmErrCode ret, const char* reason);

    //为节省篇幅，此处省略其他若干回调接口，实际操作时请注意拷贝完整关注的回调函数。
}

```

📖 说明

为节省篇幅，上文代码中有部分省略，请根据SDK头文件实现完整代码。

2. 在callbackProc.cpp文件中定义接口。

```
#include "stdafx.h"
#include "callbackProc.h"

callbackProc::callbackProc()
{
}

callbackProc::~callbackProc()
{
}

/**
 * 初始化回调
 */
void callbackProc::OnInitResult(hwmsdk::HwmErrCode ret, const char* reason)
{
    CString retStr;
    retStr.Format(_T("%d"), ret);
    CString tips = _T("OnInitResult code:") + retStr;
    AfxMessageBox(tips);
}

/**
 * 退出回调*/
void callbackProc::OnExitResult(hwmsdk::HwmErrCode ret, const char* reason)
{
    CString retStr;
    retStr.Format(_T("%d"), ret);
    CString tips = _T("OnExitResult code:") + retStr;
    AfxMessageBox(tips);
}
```

//为节省篇幅，此处省略若干代码，实际操作时请注意定义实现关注的回调函数接口。

📖 说明

为节省篇幅，上文代码中有部分省略，请根据SDK头文件实现完整代码。不同版本使用预编译头文件的头文件（stdafx.h或pch.h）可能不一样，若有编译错误，以本地默认头文件为准。

3. 在notifyProc.h文件中包含SDK回调类头文件HwmAgentNotify.h，使callbackProc类继承于基类HwmAgentNotify，可以只拷贝实现HwmAgentNotify.h文件中需要关注的接口调用回调函数。

```
#pragma once
#include "HwmAgentNotify.h"

class notifyProc : public hwmsdkagent::HwmAgentNotify
{
public:
    notifyProc();
    ~notifyProc();

    /**
     * @brief [en] Notify of sdk disconnected.
     *
     * @attention [en] NA
     */
    void OnSdkDisconnected();

    //为节省篇幅，此处省略其他若干回调接口，实际操作时请注意拷贝完整关注的回调函数。
}
```

📖 说明

为节省篇幅，上文代码中有部分省略，请根据SDK头文件实现完整代码。

- 在notifyProc.cpp文件中定义接口。

```
#include "stdafx.h"
#include "notifyProc.h"

notifyProc::notifyProc()
{
}

notifyProc::~notifyProc()
{
}

//SDK连接断开通知
void notifyProc::OnSdkDisconnected()
{
    CString tips = _T("OnSdkDisconnected");
    AfxMessageBox(tips);
}

//为节省篇幅，此处省略其他若干回调接口，实际操作时请注意拷贝完整关注的回调函数。
```

📖 说明

为节省篇幅，上文代码中有部分省略，请根据SDK头文件实现完整代码。不同版本使用预编译头文件的头文件（stdafx.h或pch.h）可能不一样，若有编译错误，以本地默认头文件为准。

- 在Hello_WorldDlg.h文件中包含需要用到的头文件。

```
#include "HwmErrorCodeDef.h"
```

- 在Hello_WorldDlg.h文件中声明将要使用的函数。

```
public:
    //初始化接口
    hwmsdk::HwmErrCode Init();
    //退出接口
    static hwmsdk::HwmErrCode Exit();
```

- 在Hello_WorldDlg.cpp文件中包含用到的头文件。

```
#include "notifyProc.h"
#include "callBackProc.h"
#include "HwmSdkAgent.h"
#include <string>
```

- 在Hello_WorldDlg.cpp定义回调和通知对象。

```
static notifyProc *notifyObj = new notifyProc();
static callbackProc *callbackObj = new callbackProc();
```

- 在Hello_WorldDlg.cpp文件中定义初始化接口。

```
/**
 *初始化接口
 */
hwmsdk::HwmErrCode CHello_WorldDlg::Init()
{
    //组装入参结构体
    hwmsdkagent::HwmlnitInfo initParam;
    memset(&initParam, 0, sizeof(hwmsdkagent::HwmlnitInfo));

    //设置SDK路径，若HwmSdk文件夹实际存放路径与下文不一致，以实际存放路径为准
    std::string path = "E:\\Hello_World\\debug\\win32\\HwmSdk\\HwmSdk.exe";
    //将path路径转换为UTF8格式后拷贝给initParam，此处省略转换代码
    strcpy_s(initParam.exePath, HWM_MAX_FILE_PATH_LEN, path.c_str());
    //指定日志路径，将路径转换为UTF8格式后拷贝给initParam，此处省略转换代码
    std::string logPath = "E:\\Hello_World\\debug\\win32\\MySdk\\log\\";
    strcpy_s(initParam.logPath, HWM_MAX_FILE_PATH_LEN, logPath.c_str());
    //指定数据路径，将路径转换为UTF8格式后拷贝给initParam，此处省略转换代码
```

```
std::string userDataPath = "E:\\Hello_World\\debug\\win32\\MySdk\\UserData\\";
strcpy_s(initParam.userDataPath, HWM_MAX_FILE_PATH_LEN, userDataPath.c_str());

std::string appld = "Hello_World";
strncpy_s(initParam.appld, appld.c_str(), HWM_MAX_APPID_LEN);

initParam.notify = notifyObj;
initParam.callback = callbackObj;
hwmsdk::HwmErrCode ret = hwmsdkagent::Init(&initParam);

return ret;
}
```

10. 在Hello_WorldDlg.cpp文件中实现初始化按钮点击事件处理。

```
void CHello_WorldDlg::OnBnClickedButton1()
{
    // TODO: 在此添加控件通知处理程序代码
    hwmsdk::HwmErrCode ret = Init();
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("Init error"));
    }
}
```

📖 说明

上代码部分与工程代码重复，注意区分，不要重复添加。

11. 在Hello_WorldDlg.cpp定义退出接口

```
/**
 *退出接口
 */
hwmsdk::HwmErrCode CHello_WorldDlg::Exit()
{
    hwmsdk::HwmErrCode ret = hwmsdkagent::Exit();
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("Exit error"));
    }
    return ret;
}
```

12. 在Hello_World.cpp文件中找到InitInstance接口，在窗口退出时调用Exit接口。

```
// CHello_WorldApp 初始化

BOOL CHello_WorldApp::InitInstance()
{
    // 如果一个运行在 Windows XP 上的应用程序清单指定要
    // 使用 ComCtl32.dll 版本 6 或更高版本来启用可视化方式，
    //则需要 InitCommonControlEx()。 否则，将无法创建窗口。
    INITCOMMONCONTROLSEX InitCtrls;
    InitCtrls.dwSize = sizeof(InitCtrls);
    // 将它设置为包括所有要在应用程序中使用的
    // 公共控件类。
    InitCtrls.dwICC = ICC_WIN95_CLASSES;
    InitCommonControlEx(&InitCtrls);

    CWinApp::InitInstance();

    AfxEnableControlContainer();

    // 创建 shell 管理器，以防对话框包含
    // 任何 shell 树视图控件或 shell 列表视图控件。
    CShellManager *pShellManager = new CShellManager;

    // 激活“Windows Native”视觉管理器，以便在 MFC 控件中启用主题
    CMFCVisualManager::SetDefaultManager(RUNTIME_CLASS(CMFCVisualManagerWindows));

    // 标准初始化
```

```
// 如果未使用这些功能并希望减小
// 最终可执行文件的大小, 则应移除下列
// 不需要的特定初始化例程
// 更改用于存储设置的注册表项
// TODO: 应适当修改该字符串,
// 例如修改为公司或组织名
SetRegistryKey(_T("应用程序向导生成的本地应用程序"));

CHello_WorldDlg dlg;
m_pMainWnd = &dlg;
INT_PTR nResponse = dlg.DoModal();
if (nResponse == IDOK)
{
    // TODO: 在此放置处理何时用
    // “确定” 来关闭对话框的代码
    CHello_WorldDlg::Exit();
}
else if (nResponse == IDCANCEL)
{
    // TODO: 在此放置处理何时用
    // “取消” 来关闭对话框的代码
    CHello_WorldDlg::Exit();
}
else if (nResponse == -1)
{
    TRACE(traceAppMsg, 0, "警告: 对话框创建失败, 应用程序将意外终止。\\n");
    TRACE(traceAppMsg, 0, "警告: 如果您在对话框上使用 MFC 控件, 则无法 #define
_AFX_NO_MFC_CONTROLS_IN_DIALOGS。\\n");
}

// 删除上面创建的 shell 管理器。
if (pShellManager != nullptr)
{
    delete pShellManager;
}

#ifdef _AFXDLL && !defined(_AFX_NO_MFC_CONTROLS_IN_DIALOGS)
    ControlBarCleanUp();
#endif

// 由于对话框已关闭, 所以将返回 FALSE 以便退出应用程序,
// 而不是启动应用程序的消息泵。
return FALSE;
}
```

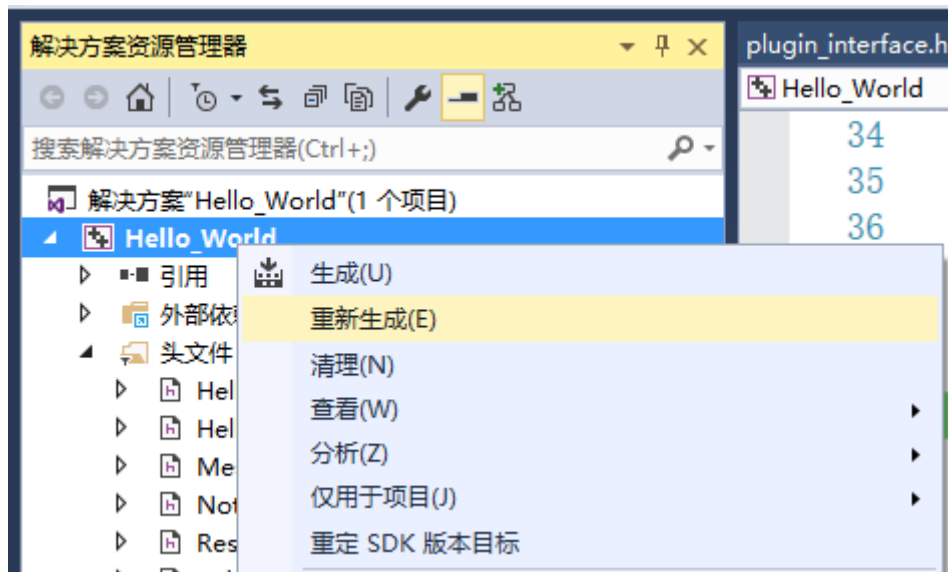
说明

上代码部分与工程代码重复, 注意区分, 不要重复添加。

步骤9 编译及调试工程。

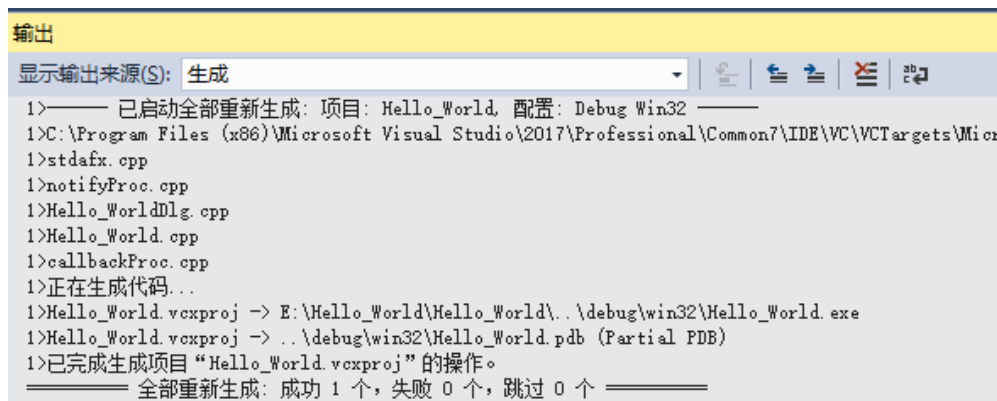
1. 选中工程, 右键选择“重新生成”, 进行编译。

图 7-22 生成文件



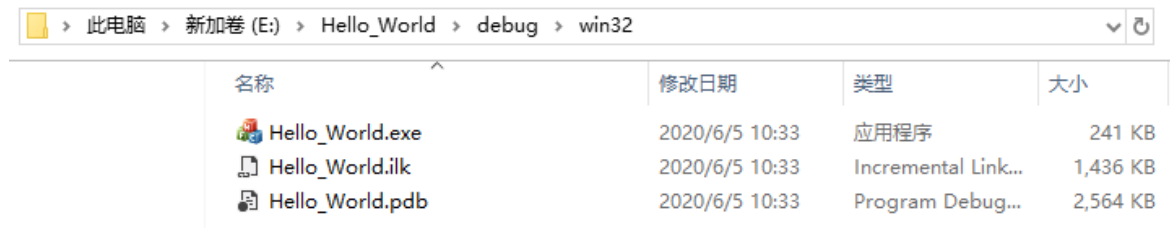
2. 观察“Output”窗口，出现信息“全部重新生成: 成功 1 个，失败 0 个，跳过 0 个”，代表编译成功。

图 7-23 生成结果



3. 编译成功后，生成的可执行程序Hello_World.exe在输出目录下，当前输出目录路径为\${path}\Hello_World\debug\win32。

图 7-24 exe 文件



4. 拷贝“SDK\dll”目录和“SDK\HwmSdk”目录下对应文件到输出目录。
如果配置为win32平台时，拷贝SDK\dll\win32内dll到输出目录，拷贝SDK\HwmSdk内所有文件到输出目录

图 7-25 拷贝 SDK\dll\win32 内 dll 到输出目录:

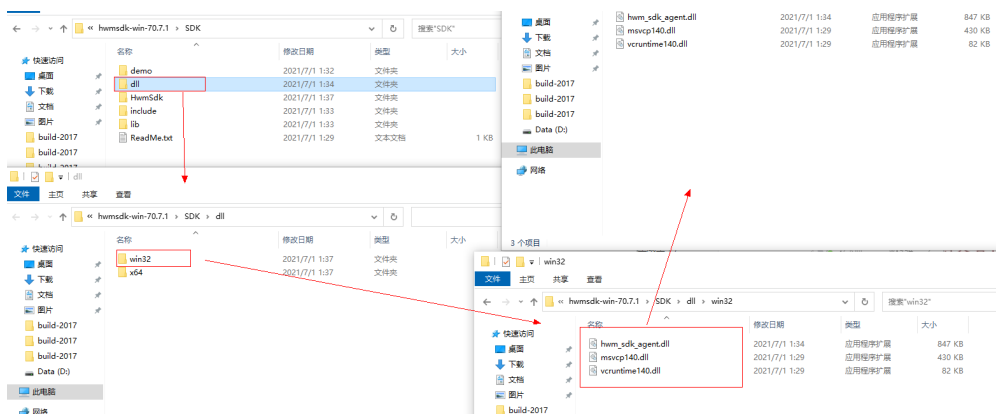
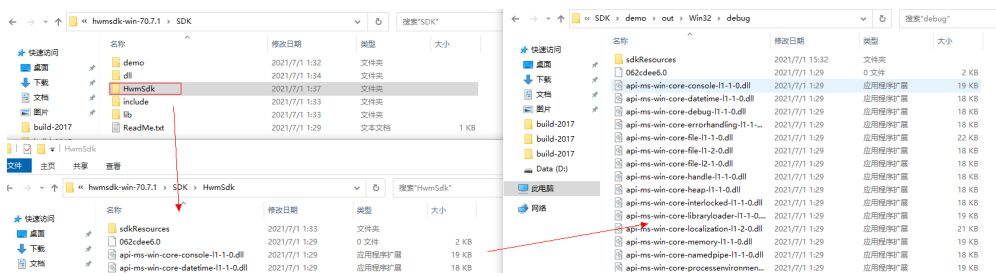


图 7-26 拷贝 SDK\HwmSdk 内所有文件到输出目录



如果配置为x64平台时，拷贝SDK\dll\x64内dll到输出目录，拷贝SDK\HwmSdk整个文件夹到输出目录，拷贝SDK\dll\x64内HwmSdk.exe到输出目录HwmSdk文件夹下。

图 7-27 拷贝 SDK\dll\x64 内 dll 到输出目录

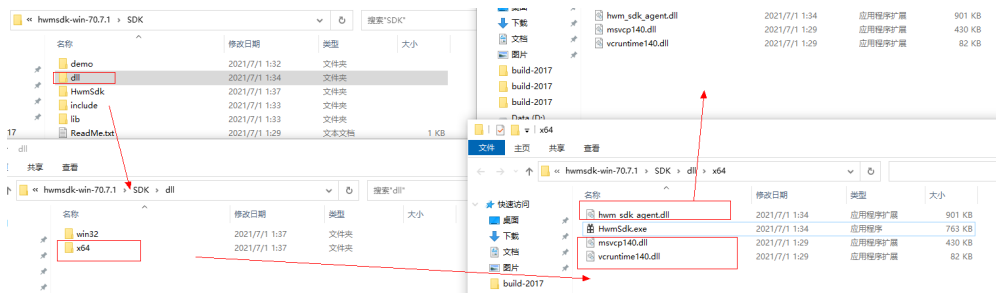


图 7-28 拷贝 SDK\HwmSdk 整个文件夹到输出目录

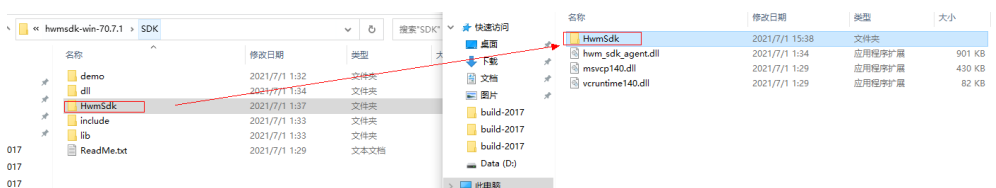
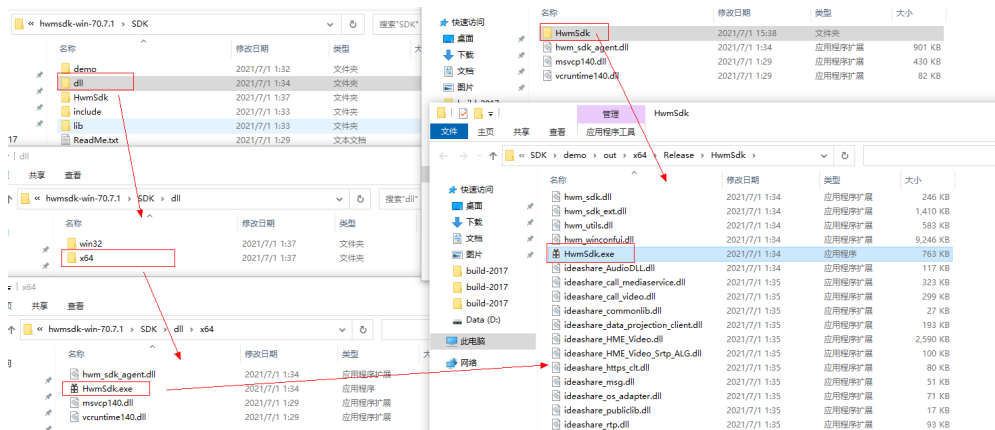


图 7-29 拷贝 SDK\dll\x64 内 HwmSdk.exe 到输出目录 HwmSdk 文件夹下

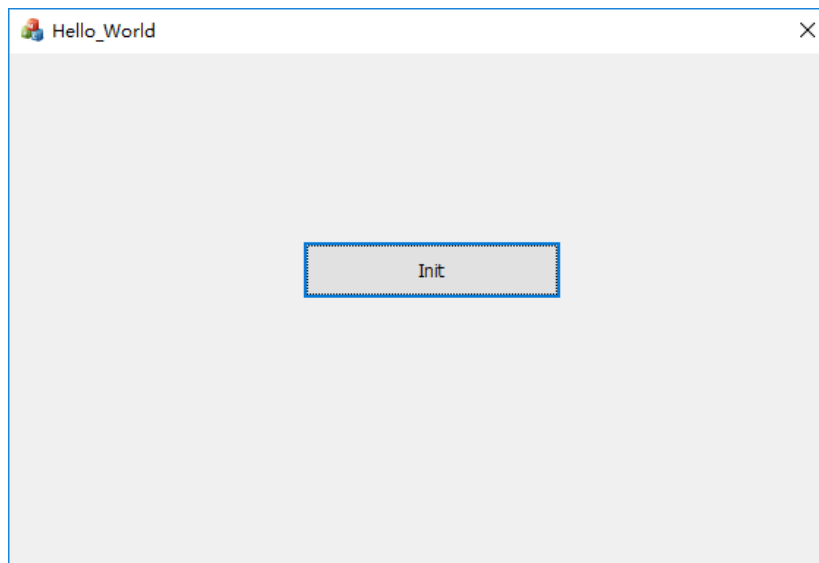


说明

配置为Win32平台时，目录地址为SDK\dll\win32，配置为x64平台时，目录地址为SDK\dll\x64。HwmSdk内为公共依赖库文件

5. 双击Hello_World.exe文件，启动界面。

图 7-30 启动页面



6. 点击“Init”按钮登录，初始化结果有弹框提示，0表示成功，其他表示失败，初始化成功后有启动通知弹框。

图 7-31 结果提示

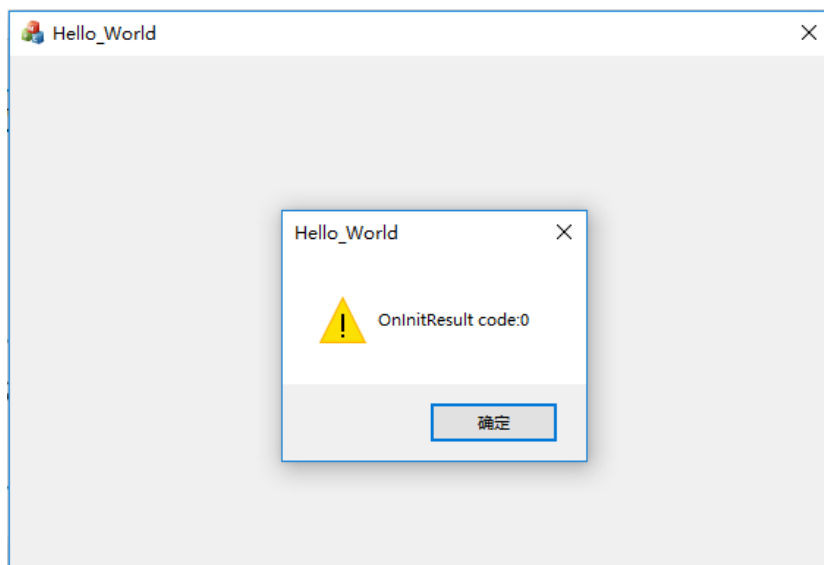
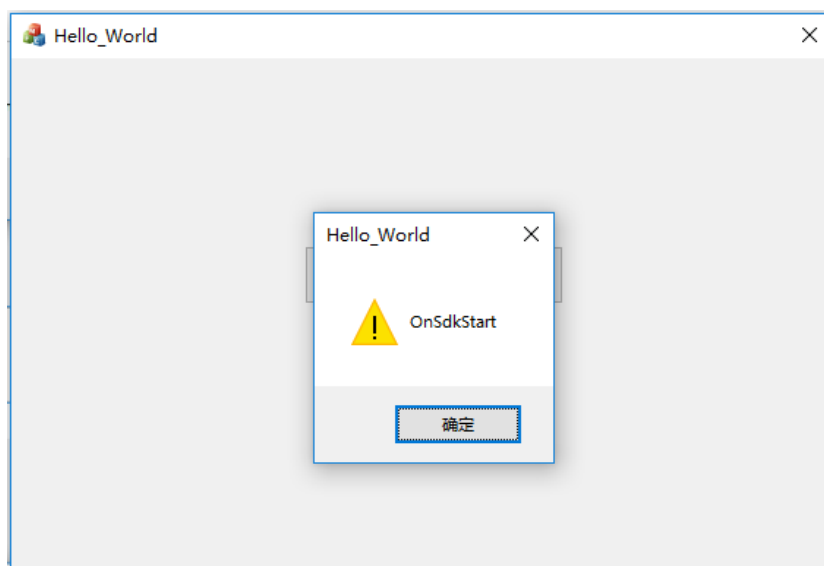


图 7-32 启动通知



----结束

7.4 典型场景

本小节描述华为云会议在场景使用场景下的接口调用顺序和接口调用示例。

7.4.1 典型场景接口调用概览

3 类接口描述

Windows SDK的接口分为主动调用接口、回调接口、消息通知接口。

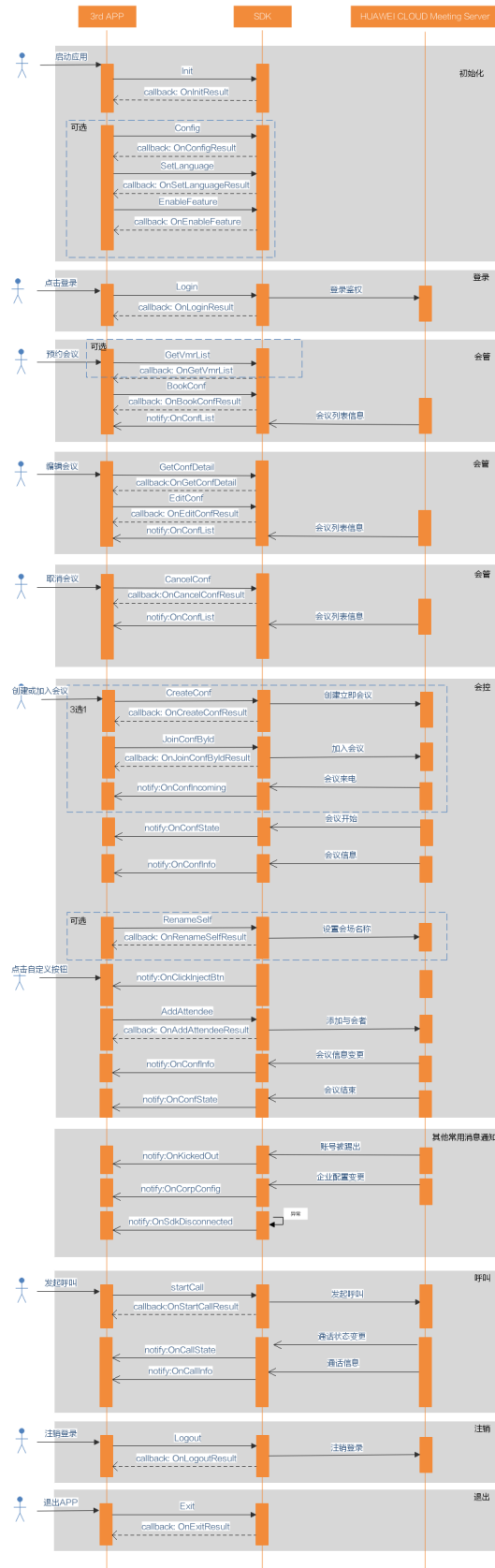
主动调用接口： App主动调用SDK的接口，用于完成某项任务。比如初始化接口Init。

回调接口：由于主动调用接口和SDK内部任务处理采用异步处理方式，所有SDK某项任务处理完成后，以回调接口的方式通知App。比如初始化结果回调接口OnInitResult。

消息通知接口：由华为云会议服务端触发或SDK内部触发的事件通知，SDK以消息通知接口的形式通知给App。比如会议来电通知OnConflncoming。

典型场景接口的调用顺序及生命周期

图 7-33 典型场景接口调用时序图



📖 说明

- 1、一般应用升级前需要杀死进程，为了便于升级管理，初始化之前请修改HwmSdk文件夹下HwmSdk.exe文件的名称为用户自己的名称。
- 2、登录不是必须的步骤，在不登录的情况下也能通过会议ID和密码加入已经创建好的会议。
- 3、开启共享、结束共享、离开会议、结束会议等接口，当前SDK的UI上已经提供了对应的功能，如果第三方的App不需要程序控制这些功能则无需调用对应接口。
- 4、设置会场名（RenameSelf）接口在会议前和会议中都可以调用。
- 5、会议开始后，OnConfInfo会上报多次。

7.4.2 场景 1：初始化

描述

在第三方APP启动时，需要先初始化SDK，才能调用SDK的功能接口，该接口只需调用一次。

业务流程

SDK初始化时，先修改HwmSdk文件夹下HwmSdk.exe文件的名称，再调用Init接口，然后处理回调函数OnInitResult。

步骤1 修改exe文件名称

修改HwmSdk文件夹下HwmSdk.exe文件名称为MySdk.exe。

步骤2 接口调用

1. 构建数据结构HwmInitInfo。
 - a. 定义派生类demoNotifyProc继承于基类HwmAgentNotify，新建派生类对象。

```
static demoNotifyProc *notifyObj = new demoNotifyProc();
```
 - b. 定义派生类demoCallbackProc继承于基类HwmAgentCallback，新建派生类对象。

```
static demoCallbackProc *callbackObj = new demoCallbackProc();
```
 - c. 构建其他变量。
2. 调用Init接口，完成配置初始化，第1步中的数据作为参数。

步骤3 处理回调函数

处理回调函数OnInitResult。

----结束

示例代码

```
/**
 *初始化接口，拉起应用
 */
int CdemoBeforeLoginDlg::Init()
{
    //组装入参结构体
    hwmagent::HwmInitInfo initParam;
    memset(&initParam, 0, sizeof(HwmInitInfo));

    //(win32平台不需要)x64平台需要设置sdk路径，假设HwmSdk目录在E:\Hello_World\debug\win32\，假设exe文件名称为MySdk.exe
```



```
std::string path = "E:\\Hello_World\\debug\\win32\\HwmSdk\\MySdk.exe";
//将路径转换为UTF8格式后拷贝给initParam, 此处省略转换代码
strcpy_s(initParam.exePath, HWM_MAX_FILE_PATH_LEN, path.c_str());
//指定日志路径, 将路径转换为UTF8格式后拷贝给initParam, 此处省略转换代码
std::string logPath = "E:\\Hello_World\\debug\\win32\\MySdk\\log\\";
strcpy_s(initParam.logPath, HWM_MAX_FILE_PATH_LEN, logPath.c_str());
//指定数据路径, 将路径转换为UTF8格式后拷贝给initParam, 此处省略转换代码
std::string userDataPath = "E:\\Hello_World\\debug\\win32\\MySdk\\UserData\\";
strcpy_s(initParam.userDataPath, HWM_MAX_FILE_PATH_LEN, userDataPath.c_str());

string appld("602b111272ad4b6989e683c27e6e87d4");
strncpy_s(initParam.appld, appld.c_str(), HWM_MAX_APPID_LEN);

initParam.notify = notifyObj;
initParam.callback = callbackObj;
initParam.authType = hwmsdkagent::HWM_AUTH_TYPE_APPID;
int ret = hwmsdkagent::Init(&initParam);

return ret;
}
```

📖 说明

exePath参数取值是HwmSdk目录的上级父目录, 例如HwmSdk目录是E:\Hello_World\debug\win32\HwmSdk时, exePath的取值是E:\Hello_World\debug\win32\。

```
/**
 * 初始化回调
 */
void demoCallbackProc::OnInitResult(hwmsdk::HwmErrCode ret, const char* msg, HwmSdkInfo *sdkInfo)
{
    CString codeStr;
    codeStr.Format(_T("%d"), ret);
    string msgStr = CTools::UTF82MultiByte(msg);
    string verStr = "";
    if (sdkInfo != NULL) {
        verStr = CTools::UTF82MultiByte(sdkInfo->version);
    }
    CString tips = _T("OnInitResult code:") + codeStr + _T(", msg:") + CString(msgStr.c_str()) + _T(" version:")
+ CString(verStr.c_str());
    AfxMessageBox(tips);
}
```

📖 说明

典型场景和接口参考中的示例代码仅作为示例用的伪代码, 不能直接使用。

7.4.3 场景 2: 登录

描述

可用华为云会议帐号或App ID或登录华为云会议。创建会议等接口不支持在未登录状态下调用, 若要使用完整的会议功能, 必须先登录华为云会议。

📖 说明

在未登录状态下, 只能加入已经创建的会议。

业务流程

使用SDK登录时, 先调用Login接口, 然后处理回调函数OnLoginResult。

步骤1 接口调用

1. 组装数据结构HwmLoginInfo。
 - a. 选择合适的登录类型。
 - b. 根据登录类型给对应的变量赋值。
2. 调用Login进行登录，第1步中的数据作为参数。

步骤2 处理回调函数

处理回调函数OnLoginResult。

----结束

示例代码

```

/**
 * 登录处理
 */
int demoLoginDlg::Login()
{
    int ret;
    //设置App ID登录参数
    string userId= "admin@1";
    string userName= "Mike";
    string email= "xxx@xxx.com";
    string phoneNumber= "1234567890";
    /**
     * 此处省略第三方服务器做的数据转换，最后获取到nonce和signature
     */
    strncpy_s(loginParam.appldAuthInfo.thirdUserId, userId.c_str(), HWM_MAX_USER_ID_LEN);
    strncpy_s(loginParam.appldAuthInfo.userName, userName.c_str(), HWM_MAX_USER_NAME_LEN);
    strncpy_s(loginParam.appldAuthInfo.signature, signature.c_str(), HWM_MAX_SIGNATURE_LEN);
    strncpy_s(loginParam.appldAuthInfo.nonce, nonce.c_str(), HWM_APPID_NONCE_LEN );
    strncpy_s(loginParam.appldAuthInfo.email, email.c_str(), HWM_MAX_EMAIL_LEN);
    strncpy_s(loginParam.appldAuthInfo.phoneNumber, phoneNumber.c_str(),
HWM_MAX_PHONE_NUM_LEN);
    loginParam.appldAuthInfo.expireTime = 1598398920; // utc时间，单位s

    loginParam.authType = hwmsdkagent::HWM_AUTH_TYPE_APPID;
    //调用登录接口
    ret = hwmsdkagent::Login(&loginParam);

    return ret;
}

/**
 * 登录接口回调
 */
void demoCallbackProc::OnLoginResult(hwmsdk::HwmErrCode ret, const char* msg)
{
    CString codeStr;
    codeStr.Format(_T("%d"), ret);
    string msgStr = CTools::UTF82MultiByte(msg);
    CString tips = _T("OnLoginResult code:") + codeStr + _T(", msg:") + CString(msgStr.c_str());
    AfxMessageBox(tips);
}

```

7.4.4 场景 3：预约会议

描述

在使用华为云会议帐号登录后，可以调用预约会议接口预约会议。预约会议时，除了填写预约会议所需的会议参数以外，还可以添加与会者。

业务流程

调用BookConf接口预约会议，然后处理回调函数OnBookConfResult和OnConfList通知。

若要预约个人会议ID的会议或者云会议室的会议，则需要先调用GetVmrList接口查询个人会议ID和云会议室信息，然后处理回调函数OnGetVmrList返回的数据，该数据可用于预约会议。

步骤1 接口调用

1. 组装数据结构HwmBookConfParam（包含结构内的HwmAttendeeInfo）。
2. 调用BookConf开始创建预约会议，第1步中的数据作为参数。

步骤2 处理回调函数

处理回调函数OnBookConfResult。

步骤3 处理消息通知

处理消息通知OnConfList。

----结束

示例代码

```
/**
 * 预约会议
 */
int CdemoBookConfDlg::OnBnClickedBookConf()
{
    hwmsdkagent::HwmBookConfParam data = {0};

    string confSubject = u8"我的预约会议"; // 此处应是utf-8编码字符串
    strncpy_s(data.subject, confSubject.c_str(), HWM_MAX_SUBJECT_LEN);
    data.confType = hwmsdkagent::HwmConfMediaType::HWM_VIDEO_AND_DATA;
    data.isNeedConfPwd = true;
    data.isRecordOn = false; //默认会议不开启允许录制
    data.isAutoRecordOn = false; //默认会议不开启自动录制
    data.startTime = 1598398920; //utc时间戳，单位秒，如果获取的时间是本地时间，需要转换成utc时间
    data.duration = 30; //会议持续时长(分钟)
    data.timeZone = 56; //时区，56表示东八区
    data.joinConfRestrictionType = hwmsdkagent::HwmJoinConfPermissionType::RESTRICTION_CALL_IN_ALL;

    std::vector<hwmsdkagent::HwmAttendeeInfo> attendeeList;
    //被邀与会者
    hwmsdkagent::HwmAttendeeInfo attendeeInfo = { 0 };
    string attName = "Mike";
    string attNumber = "+991116003543";
    string attEmail = "";
    string attSms = "";
    string attUserId = "";
    strncpy_s(attendeeInfo.name, attName.c_str(), HWM_MAX_USER_NAME_LEN);
    strncpy_s(attendeeInfo.number, attNumber.c_str(), HWM_MAX_NUMBER_LEN);
    strncpy_s(attendeeInfo.email, attEmail.c_str(), HWM_MAX_EMAIL_LEN);
    strncpy_s(attendeeInfo.sms, attSms.c_str(), HWM_MAX_PHONE_NUM_LEN);
    strncpy_s(attendeeInfo.thirdUserId, attUserId.c_str(), HWM_MAX_USER_ID_LEN);
    attendeeInfo.isMute = true;
    attendeeInfo.isAutoInvite = true;
    attendeeList.push_back(attendeeInfo);

    data.attendees = attendeeList.data();
    data.attendeeLen = attendeeList.size();
    data.isSmsOn = true;
}
```

```

data.isMailOn = true;
data.isEmailCalendarOn = true;
return hwmsdkagent::BookConf(&data);
}

```

```

/**
 * 预约会议接口回调
 */
void demoCallbackProc::OnBookConfResult(hwmsdk::HwmErrCode ret, const char* msg)
{
    CString codeStr;
    codeStr.Format(_T("%d"), ret);
    string msgStr = CTools::UTF82MultiByte(msg);
    CString tips = _T("OnBookConfResult code:") + codeStr + _T(", msg:") + CString(msgStr.c_str());
    AfxMessageBox(tips);
}

```

```

/**
 * 会议列表信息通知
 */
void demoNotifyProc::OnConfList(HwmConfListUpdateType updateType, const HwmConfListInfo*
confInfoList, unsigned int confInfoLen)
{
    if (confInfoLen <= 0 || confInfoList == nullptr)
    {
        return;
    }
    //登录后全量返回会议列表
    if (updateType == CONFLIST_UPDTAE_ALL)
    {
        //处理全量会议列表
    }
    else
    {
        if (updateType == CONFLIST_UPDATE_ADD)
        {
            //处理新增的会议或被编辑的会议
        }
        else if (updateType == CONFLIST_UPDATE_DELETE)
        {
            //处理取消的会议
        }
    }

    CString tips = _T("OnConfList");
    for (int i = 0; i < confInfoLen; i++)
    {
        CString strConfId = CTools::UTF2UNICODE(confInfoList->confId);
        CString strConfSubject = CTools::UTF2UNICODE(confInfoList->confSubject);
        CString tip = _T(", confId:") + CString(strConfId)
            + _T(", subject:") + CString(strConfSubject);
        tips += tip;
        confInfoList++;
    }
    AfxMessageBox(tips);
}

```

7.4.5 场景 4：编辑会议

描述

预约会议成功后，可以通过编辑会议接口来编辑会议信息，在编辑会议时可以添加与会者。

业务流程

得到会议列表以后，若要编辑会议，需要先通过GetConfDetail接口查询会议详情，在接口回调OnGetConfDetail中处理详情数据，再调用EditConf接口对会议详情进行修改，并处理回调函数OnEditConfResult和消息通知OnConfList。

步骤1 获取会议详情

1. 调用GetConfDetail接口。

步骤2 处理回调函数

处理回调OnGetConfDetail，得到会议详情数据。

步骤3 调用编辑会议接口

1. 使用步骤2获取的数据，组装数据结构HwmEditConfParam（包含结构内的HwmAttendeeInfo）。
2. 调用EditConf接口，第1步中的数据作为入参。

步骤4 处理回调函数

处理回调函数OnEditConfResult。

步骤5 处理消息通知

处理消息通知OnConfList。

----结束

示例代码

```
/**
 * 获取会议详情
 */
int demoEditConfDlg::GetConfDetail()
{
    string confID = CTools::UNICODE2UTF(CString("989156631"));
    int ret =hwmsdkagent::GetConfDetail(confID); // 使用会议id查询会议详细信息
    return ret;
}
```

```
/**
 * 会议详情回调处理
 */
void demoCallbackProc::OnGetConfDetail(hwmsdk::HwmErrCode ret, const char* reason, const
HwmConfDetail* confDetail)
{
    if (confDetail == nullptr)
    {
        return;
    }
    // 此处省略缓存会议详情数据代码
    CString codeStr;
    codeStr.Format(_T("%d"), ret);
    string msgStr = CTools::UTF82MultiByte(msg);
    CString tips = _T("OnGetConfDetail code:") + codeStr + _T(", msg:") + CString(msgStr.c_str());
    AfxMessageBox(tips);
}
```

```
/**
 * 编辑会议接口
 */
```

```
int demoEditConfDlg::EditConf()
{
    hwmsdkagent::HwmConfDetail confDetail{0}; //从缓存中得到会议详情数据赋予confDetail
    hwmsdkagent::HwmEditConfParam editConfParam{ 0 };
    //根据实际需求拷贝confDetail的数据
    strncpy_s(editConfParam.confId, confDetail.confListInfo.confId, HWM_MAX_CONF_ID_LEN);
    editConfParam.vmrFlag = confDetail.vmrFlag;
    strncpy_s(editConfParam.vmrId, confInfo.vmrId, HWM_MAX_VMR_CONF_ID_LEN);
    // 此处省略部分赋值代码
    editConfParam.timeZone = 56; // 本地时区, 用户依据自身情况自己调整, 56东八区
    editConfParam.isRecordOn = false; //默认会议不启用允许录制
    editConfParam.isAutoRecordOn = false; //默认会议不启用自动录制
    editConfParam.startTime = 1598398920; //utc时间戳
    editConfParam.duration = 0 * 60 + 30; //会议时长
    editConfParam.joinConfRestrictionType =
hwmsdkagent::HwmJoinConfPermissionType::RESTRICTION_CALL_IN_ALL; //取编辑会议上面的允许入会用户类型
    return hwmsdkagent::EditConf(&editConfParam);
}

/**
 * 编辑会议接口回调
 */
void demoCallbackProc::OnEditConfResult()
{
    CString codeStr;
    codeStr.Format(_T("%d"), ret);
    string msgStr = CTools::UTF82MultiByte(msg);
    CString tips = _T("OnEditConfResult code:") + codeStr + _T(", msg:") + CString(msgStr.c_str());
    AfxMessageBox(tips);
}
```

📖 说明

OnConfList事件通知跟预约会议场景相同, 请参考[预约会议的示例代码](#)。

7.4.6 场景 5: 取消会议

描述

预约会议成功后, 可以取消会议接口来取消会议。

业务流程

若要取消会议, 可以调用CancelConf接口, 并处理回调函数OnCancelConfResult和消息通知OnConfList。

步骤1 接口调用

调用CancelConf取消会议。

步骤2 处理回调函数

处理回调函数OnCancelConfResult。

步骤3 处理消息通知

处理消息通知OnConfList。

----结束

示例代码

```
/**
 * 取消会议接口
```

```

*/
int demoConfListDlg::OnBnCancelConf(hwmsdk::HwmErrCode ret, const char* msg)
{
    string confId = CTools::UNICODE2UTF(CString("989156631"));
    return hwmsdkagent::CancelConf(confId);
}

/**
 * 取消会议接口回调
 */
void demoCallbackProc::OnCancelConfResult(hwmsdk::HwmErrCode ret, const char* msg)
{
    CString codeStr;
    codeStr.Format(_T("%d"), ret);
    string msgStr = CTools::UTF82MultiByte(msg);
    CString tips = _T("OnCancelConfResult code:") + codeStr + _T(", msg:") + CString(msgStr.c_str());
    AfxMessageBox(tips);
}

```

📖 说明

OnConfList事件通知跟预约会议场景相同，请参考[预约会议的示例代码](#)。

7.4.7 场景 6：创建会议

描述

在使用华为云会议帐号登录后，可以调用创建会议接口创建立即会议。创建会议时可以携带与会人信息，也可以不携带。

业务流程

若要创建个人会议ID的会议或者云会议室的会议，则需要先调用GetVmrList接口查询个人会议ID和云会议室信息，然后处理回调函数OnGetVmrList返回的数据，该数据可用于创建会议。

使用SDK创建立即会议时，先调用CreateConf接口，然后处理回调函数OnCreateConfResult和消息通知OnConfState、OnConfInfo。

步骤1 接口调用

1. 组装数据结构HwmCreateConfInfo。
2. 组装数据结构HwmConfAttendee。
3. 调用CreateConf开始创建，第1步中的数据作为参数。

步骤2 处理回调函数

处理回调函数OnCreateConfResult。

步骤3 处理消息通知

处理消息通知OnConfState。

步骤4 处理消息通知

处理消息通知OnConfInfo。

步骤5 处理消息通知

处理消息通知OnConfList。

----结束

示例代码

```

/**
 * 创建会议
 */
int demoCreateConfWithAttendeeDlg::clickCreatConfWithAttendee()
{
    int ret;
    hwmsdkagent::HwmCreateConfInfo data;
    memset(&data, 0, sizeof(hwmsdkagent::HwmCreateConfInfo));

    //设置会议主题
    string meetingSubjectStr = CTools::UNICODE2UTF(CString("我的会议"));
    strncpy_s(data.subject, meetingSubjectStr.c_str(), HWM_MAX_SUBJECT_LEN);
    //设置会议类型
    data.mediaType = hwmsdkagent::HWM_VIDEO_AND_DATA
    //设置会议是否需要来宾密码
    data.needPassword = true;
    data.callInRestriction = hwmsdkagent::RESTRICTION_CALL_IN_ALL;
    //设置vmrId, 若创建随机会议, 则vmrId为空 此处m_confIdTypeCombo排序是把随机会议放在首位
    //因此vmrIdSelect > 0 为个人会议id与云会议室id场景
    const auto vmrIdSelect = m_confIdTypeCombo.GetCurSel();
    if (vmrIdSelect > 0 && vmrIdSelect - 1 < vmrList.size())
    {
        strcpy_s(data.vmrId, HWM_MAX_VMR_CONF_ID_LEN, vmrList[vmrIdSelect - 1].vmrId);
    }

    //设置与会人
    //获取与会人信息, 实际使用是根据实际情况获取
    CString tempCString;
    m_attendeesEdit.GetWindowText(tempCString);
    string tempString = CTools::UNICODE2UTF(tempCString);
    vector<string> list = CTools::split(tempString, ',');
    vector<string> temp;
    int count = list.size();
    int realCount = 0;
    if (count > 0)
    {
        //申请结构体内存
        hwmsdkagent::HwmConfAttendee* participants;
        participants = (hwmsdkagent::HwmConfAttendee*)malloc(sizeof(hwmsdkagent::HwmConfAttendee) *
count);
        if (participants == NULL)
        {
            return -1;
        }
        memset(participants, 0, sizeof(hwmsdkagent::HwmConfAttendee)*count);

        hwmsdkagent::HwmConfAttendee* participantsTemp = participants;
        for (int i = 0; i < count; i++)
        {
            temp = CTools::split(list[i], '-');
            if (temp.size() == 2)
            {
                //name赋值
                strncpy_s(participantsTemp->name, (char *)temp[0].c_str(), HWM_MAX_DISPLAY_NAME_LEN);
                //number赋值
                strncpy_s(participantsTemp->number, (char *)temp[1].c_str(), HWM_MAX_NUMBER_LEN);
                //指针个数加1
                realCount++;
                participantsTemp++;
            }
        }
        //调用SDK接口, 创建会议
        ret = hwmsdkagent::CreateConf(&data, participants, realCount);

        //释放内存空间
        free(participants);
        participants = NULL;
    }
}

```



```

    }
    else
    {
        //不携带与会人创建会议
        ret = hwmsdkagent::CreateConf(&data, NULL, 0);
    }
    return ret;
}

```

📖 说明

Windows SDK接口中的字符串参数，比如会议主题、与会人姓名等，需要编码成UTF8，否则接口会报错。

```

/**
 * 创建会议接口回调
 */
void demoCallbackProc::OnCreateConfResult(hwmsdk::HwmErrCode ret, const char* msg)
{
    CString codeStr;
    codeStr.Format(_T("%d"), ret);
    string msgStr = CTools::UTF82MultiByte(msg);
    CString tips = _T("OnCreateConfResult code:") + codeStr + _T(", msg:") + CString(msgStr.c_str());
    AfxMessageBox(tips);
}

```

```

/**
 * 会议状态消息通知
 */
void demoNotifyProc::OnConfState(HwmConfStateInfo *confStateInfo)
{
    CString str;
    str.Format(_T("%d"), confStateInfo->state);
    CString reason;
    reason.Format(_T("%d"), confStateInfo->reason);
    CString tips = _T("OnConfState state:") + str + _T(", reason:") + reason;
    AfxMessageBox(tips);
}

```

```

/**
 * 会议信息消息通知
 */
void demoNotifyProc::OnConfInfo(HwmConfInfo *confInfo)
{
    string urlStr = CTools::UTF82MultiByte(confInfo->url);
    string confIdStr = CTools::UTF82MultiByte(confInfo->confId);
    string chairmanPwdStr = CTools::UTF82MultiByte(confInfo->chairmanPwd);
    string generalPwdStr = CTools::UTF82MultiByte(confInfo->generalPwd);
    string subjectStr = CTools::UTF82MultiByte(confInfo->subject);

    CString role;
    role.Format(_T("%d"), confInfo->role);

    CString tips = _T("OnConfInfo confUrl:") + CString(urlStr.c_str())
        + _T(", confId:") + CString(confIdStr.c_str()) + _T(", confRole:") + CString(role)
        + _T(", chairmanPwd:") + CString(chairmanPwdStr.c_str())
        + _T(", generalPwd:") + CString(generalPwdStr.c_str())
        + _T(", subject:") + CString(subjectStr.c_str());
    AfxMessageBox(tips);
}

```

📖 说明

OnConfList事件通知跟预约会议场景相同，请参考[预约会议的示例代码](#)。

Windows SDK回调函数或者消息通知中的字符串都是UTF8编码的。

7.4.8 场景 7：添加与会人

描述

在会议中，可以调用添加与会人的接口把与会人加入到会议中。

业务流程

使用SDK添加与会人时，先调用AddAttendee接口，然后处理回调函数OnAddAttendeeResult。

步骤1 接口调用

1. 组装数据结构HwmConfAttendee。
2. 调用AddAttendee开始创建，第1步中的数据作为参数。

步骤2 处理回调函数

处理回调函数OnAddAttendeeResult。

----结束

示例代码

```
/**
 * 添加与会人
 */
int demoAddAttendeeDlg::clickAddAttendee()
{
    int ret;
    CString tempCString;

    //获取与会人信息，实际使用是根据实际情况获取
    m_numbersEdit.GetWindowText(tempCString);
    string tempString = CTools::UNICODE2UTF(tempCString);
    vector<string> list = CTools::split(tempString, ',');
    vector<string> temp;
    int count = list.size();

    int realCount = 0;
    if (count > 0)
    {
        //申请结构体内存
        hwmsdkagent::HwmConfAttendee* participants;
        participants = (hwmsdkagent::HwmConfAttendee*)malloc(sizeof(hwmsdkagent::HwmConfAttendee) *
count);
        if (participants == NULL)
        {
            return -1;
        }
        memset(participants, 0, sizeof(hwmsdkagent::HwmConfAttendee)*count);

        hwmsdkagent::HwmConfAttendee* participantsTemp = participants;
        for (int i = 0; i < count; i++)
        {
            temp = CTools::split(list[i], '-');
            if (temp.size() == 2)
            {
                //name赋值
                strncpy_s(participantsTemp->name, (char *)temp[0].c_str(), HWM_MAX_DISPLAY_NAME_LEN);
                //number赋值
                strncpy_s(participantsTemp->number, (char *)temp[1].c_str(), HWM_MAX_NUMBER_LEN);
                //指针个数加1
                realCount++;
            }
        }
    }
}
```

```

        participantsTemp++;
    }
}

ret = hwmsdkagent::AddAttendee(participants, realCount);

//释放内存空间
free(participants);
participants = NULL;
}

return ret;
}

```

```

/**
 * 添加与会者接口回调
 */
void demoCallbackProc::OnAddAttendeeResult(hwmsdk::HwmErrCode ret, const char* msg)
{
    CString codeStr;
    codeStr.Format(_T("%d"), ret);
    string msgStr = CTools::UTF82MultiByte(msg);
    CString tips = _T("OnAddAttendeeResult code:") + codeStr + _T(", msg:") + CString(msgStr.c_str());
    AfxMessageBox(tips);
}

```

7.4.9 场景 8：加入会议

描述

在华为云会议帐号登录后或者登录前，都可以使用该接口通过会议ID和密码加入会议。

业务流程

使用SDK加入已经存在的会议时，先调用JoinConfById接口，然后处理回调函数OnJoinConfByIdResult和消息通知OnConfState、OnConfInfo。

步骤1 接口调用

1. 组装数据结构HwmJoinConfByIdInfo。
2. 调用JoinConfById开始创建，第1步中的数据作为参数。

步骤2 处理回调函数

处理回调函数OnJoinConfByIdResult。

步骤3 处理消息通知

处理消息通知OnConfState。

步骤4 处理消息通知

处理消息通知OnConfInfo。

----结束

示例代码

```

// 通过会议ID加入会议
int demoJoinConfByIdDlg::clickJoinConfById()
{

```

```
// 填写会议ID和会议密码；未登录场景下加入会议时，也可以配置本会场的会场名称
string meetingID = CTools::UNICODE2UTF(CString("989156631"));
string accessCode = CTools::UNICODE2UTF(CString("807766"));
string participantName = CTools::UNICODE2UTF(CString("杭州会场"));

hwmsdkagent::HwmJoinConfByIdInfo data;
memset(&data, 0, sizeof(hwmsdkagent::HwmJoinConfByIdInfo));

strncpy_s(data.confId, meetingID.c_str(), HWM_MAX_CONF_ID_LEN);
strncpy_s(data.password, accessCode.c_str(), HWM_MAX_PASSWORD_LEN);
strncpy_s(data.name, participantName.c_str(), HWM_MAX_DISPLAY_NAME_LEN);

int ret = hwmsdkagent::JoinConfById(&data);
return ret;
}

/**
 * 会议id入会接口回调
 */
void demoCallbackProc::OnJoinConfByIdResult(hwmsdk::HwmErrCode ret, const char* msg)
{
    CString codeStr;
    codeStr.Format(_T("%d"), ret);
    string msgStr = CTools::UTF82MultiByte(msg);
    CString tips = _T("OnJoinConfByIdResult code:") + codeStr + _T(", msg:") + CString(msgStr.c_str());
    AfxMessageBox(tips);
}

```

📖 说明

OnConfState和OnConfInfo的事件通知跟创建会议场景相同，请参考[创建会议的示例代码](#)。

7.4.10 场景 9：邮件链接入会

描述

通过邮件会议通知中的会议链接入会时（会议链接如：<https://bmeeting.huaweicloud.com/#/j/981924330/e11ddec62ee760734fcf2ba55b79937eac2aa68677caf659>），链接里面的的是会议ID（即举例中的981924330）和会议的Random码（即举例中的e11ddec62ee760734fcf2ba55b79937eac2aa68677caf659）。

在华为云会议帐号登录后或者登录前，都可以使用该接口通过分享会议链接中的会议ID和48位密码映射值Random加入会议。

业务流程

使用SDK加入已经存在的会议时，先调用JoinConfByRandom接口，然后处理回调函数OnJoinConfByRandom和消息通知OnConfState、OnConfInfo

步骤1 接口调用

1. 组装数据结构HwmJoinConfByRandomInfo。
2. 调用JoinConfByRandom开始入会，第1步中的数据作为参数。

步骤2 处理回调函数

处理回调函数OnJoinConfByRandomResult。

步骤3 处理消息通知

处理消息通知OnConfState。

步骤4 处理消息通知

处理消息通知OnConfInfo。

----结束

示例代码

```
// 通过会议ID和Random加入会议
int demoJoinConfByRandomDlg::clickJoinConfByRandom()
{
    // 填写会议ID和密码映射值等; Random入会时, 支持配置会场名称
    string meetingID = CTools::UNICODE2UTF(CString("988148352"));
    string accessCode =
CTools::UNICODE2UTF(CString("c4f92bc34e685bbba6c23589741be0431997f30da5d6e47b"));
    string participantName = CTools::UNICODE2UTF(CString("杭州会场"));

    hwmsdkagent::HwmJoinConfByRandomInfo data;
    memset(&data, 0, sizeof(hwmsdkagent::HwmJoinConfByRandomInfo));

    strncpy_s(data.confId, meetingID.c_str(), HWM_MAX_CONF_ID_LEN);
    strncpy_s(data.random, accessCode.c_str(), HWM_MAX_RANDOM_LEN);
    strncpy_s(data.name, participantName.c_str(), HWM_MAX_DISPLAY_NAME_LEN);

    int ret = hwmsdkagent::JoinConfByRandom(&data);
    return ret;
}

/**
 * 会议Id和Random入会接口回调
 */
void demoCallbackProc::OnJoinConfByRandomResult(hwmsdk::HwmErrCode ret, const char* reason)
{
    CString codeStr;
    codeStr.Format(_T("%d"), ret);
    string msgStr = CTools::UTF82MultiByte(msg);
    CString tips = _T("OnJoinConfByRandomResult code:") + codeStr + _T(", msg:") + CString(msgStr.c_str());
    AfxMessageBox(tips);
}
```

📖 说明

OnConfState和OnConfInfo的事件通知跟创建会议场景相同, 请参考[创建会议的示例代码](#)。

48位密码映射值为分享会议/复制会议信息中的链接所携带的部分字符串。

<https://bmeeting.huaweicloud.com/#/j/987297025/227b2d708b7e5a24ce8de2159772239b1672c0a85ba70d85>

7.4.11 场景 10: 发起呼叫

描述

在使用华为云会议帐号登录后, 可以调用发起呼叫接口发起一路呼叫。

业务流程

使用SDK发起时, 先调用StartCall接口, 然后处理回调函数OnStartCallResult和消息通知OnCallState、OnCallInfo、OnCallRecordInfo。

步骤1 接口调用

1. 组装数据结构HwmStartCallInfo。
2. 调用StartCall开始创建, 第1步中的数据作为参数。

步骤2 处理回调函数

处理回调函数OnStartCallResult。

步骤3 处理消息通知

处理消息通知OnCallState。

步骤4 处理消息通知

处理消息通知OnCallInfo。

---结束

示例代码

```
/**
 * 发起呼叫
 */
int demoStartCallDlg::clickStartCall()
{
    hwmsdkagent::HwmStartCallInfo data;
    memset(&data, 0, sizeof(hwmsdkagent::HwmStartCallInfo));

    //被叫人姓名
    CString calleeNameCstr;
    m_calleeNameEdit.GetWindowText(calleeNameCstr);
    string calleeName = CTools::UNICODE2UTF(calleeNameCstr);
    errno_t nRet = strcpy_s(data.callee.name, sizeof(data.callee.name), calleeName.c_str());

    //被叫人号码
    CString calleeNumberCstr;
    m_calleeNumberEdit.GetWindowText(calleeNumberCstr);
    string calleeNumber = CTools::UNICODE2UTF(calleeNumberCstr);
    nRet += strcpy_s(data.callee.number, sizeof(data.callee.number), calleeNumber.c_str());

    //被叫人三方账号
    CString calleeThirdUserIdCstr;
    m_calleeThirdUserIdEdit.GetWindowText(calleeThirdUserIdCstr);
    string calleeThirdUserId = CTools::UNICODE2UTF(calleeThirdUserIdCstr);
    nRet += strcpy_s(data.callee.thirdUserId, sizeof(data.callee.thirdUserId), calleeThirdUserId.c_str());

    //主叫人姓名
    CString callerNameCstr;
    m_callerNameEdit.GetWindowText(callerNameCstr);
    string callerName = CTools::UNICODE2UTF(callerNameCstr);
    nRet += strcpy_s(data.name, sizeof(data.name), callerName.c_str());

    if (nRet != 0)
    {
        AfxMessageBox(_T("String Copy Error!"));
    }

    //视频或音频
    CString callTypeCString;
    m_callTypeCombo.GetWindowText(callTypeCString);
    hwmsdkagent::HwmCallType type = hwmsdkagent::CALL_TYPE_VIDEO;
    if (_T("Video") == callTypeCString)
    {
        data.callType = hwmsdkagent::CALL_TYPE_VIDEO;
    }
    else if (_T("Audio") == callTypeCString)
    {
        data.callType = hwmsdkagent::CALL_TYPE_AUDIO;
    }

    int ret = hwmsdkagent::StartCall(&data);
}
```

```
return ret;
}
```

📖 说明

Windows SDK接口中的字符串参数，比如被叫方姓名、呼叫方姓名等，需要编码成UTF8，否则接口会报错。

```
/**
 * 发起呼叫接口回调
 */
void demoCallbackProc::OnStartCallResult(hwmsdk::HwmErrCode ret, const char * reason)
{
    Cdemo* app = (Cdemo*)AfxGetApp();
    if (!app || reason == NULL)
    {
        //窗口已经关闭
        return;
    }
    CString codeStr;
    codeStr.Format(_T("%d"), ret);
    string msgStr = CTools::UTF82MultiByte(reason);
    CString tips = _T("OnStartCallResult code:") + codeStr + _T(", msg:") + CString(msgStr.c_str());
    AfxMessageBox(tips);
}
```

```
/**
 * 通话状态消息通知
 */
void demoNotifyProc::OnCallState(const HwmCallStateInfo *callStateInfo)
{
    Cdemo* app = (Cdemo*)AfxGetApp();
    if (!app || callStateInfo == NULL)
    {
        //窗口已经关闭
        return;
    }

    CString state;
    state.Format(_T("%d"), callStateInfo->state);
    CString callId;
    callId.Format(_T("%d"), callStateInfo->callId);

    CString tips = _T("OnCallState state:") + state + _T(", callId:" + callId);
    AfxMessageBox(tips);
}
```

```
/**
 * 通话信息消息通知
 */
void demoNotifyProc::OnCallInfo(const HwmCallInfo *callInfo)
{
    Cdemo* app = (Cdemo*)AfxGetApp();
    if (!app || callInfo == NULL)
    {
        //窗口已经关闭
        return;
    }

    CString callType;
    callType.Format(_T("%d"), callInfo->callType);

    string number = CTools::UTF82MultiByte(callInfo->number);
    string name = CTools::UTF82MultiByte(callInfo->name);
    string startTime = CTools::UTF82MultiByte(callInfo->startTime);
    string endTime = CTools::UTF82MultiByte(callInfo->endTime);

    CString isCallOut;
```

```
isCallOut.Format(_T("%d"), callInfo->isCallOut);
CString callId;
callId.Format(_T("%d"), callInfo->callId);

CString tips = _T("OnCallInfo callType:") + callType + _T(", number:") + CString(number.c_str())
    + _T(", name:") + CString(name.c_str()) + _T(", startTime:") + CString(startTime.c_str())
    + _T(", endTime:") + CString(endTime.c_str()) + _T(", isCallOut:") + isCallOut + _T(", callId:") + callId;
AfxMessageBox(tips);
}
```

说明

Windows SDK回调函数或者消息通知中的字符串都是UTF8编码的。

7.4.12 场景 11：定制会中“邀请”按钮

描述

考虑到大部分使用SDK二次开发的场景下，第三方App不会使用华为云会议的通讯录，因此Windows SDK提供的会议界面上“邀请”按钮，只提供了一个按钮界面，并没有提供完整邀请与会者的功能，需要开发者自己实现。

业务流程

使用SDK实现“邀请”功能时，在SDK初始化完成后先调用Config接口，然后处理回调函数OnConfigResult。会议中，在“邀请”按钮单击后，再处理消息通知OnClickInjectBtn。

步骤1 接口调用

1. 在SDK初始化后，调用Config进行“邀请”按钮的定制。

步骤2 处理回调函数

处理回调函数OnConfigResult。

步骤3 处理消息通知

当“邀请”按钮单击后，再处理消息通知OnClickInjectBtn

----结束

示例代码

```
/**
 * 配置邀请按钮
 */
int hwmSDKConfigUI::hwmSDKStartUIConfig()
{
    //配置“邀请”按钮
    CString inviteBtn = ("{"frame":{"confMenu":{"toolbar":{"button":{"id":"invite"},"showAsAction":
    "ifRoom"},"isCustomizedClick":true}}}}");
    string uiConfig CTools::UNICODE2UTF(inviteBtn);
    ret = hwmsdkagent::Config(uiConfig);
    return ret;
}

/**
 * 配置接口回调
 */
void demoCallbackProc::OnConfigResult(hwmsdk::HwmErrCode ret, const char* msg)
```



```

{
    CString codeStr;
    codeStr.Format(_T("%d"), ret);
    string msgStr = CTools::UTF82MultiByte(msg);
    CString tips = _T("OnConfigResult code:") + codeStr + _T(", msg:") + CString(msgStr.c_str());
    AfxMessageBox(tips);
}

/**
 * 按钮点击消息通知
 */
void notifyProc::OnClickInjectBtn(hwmsdkagent::HwmClickInjectBtn injectBtn, void* data, const char* id)
{
    INT_PTR nRes;
    switch (injectBtn)
    {
        // “邀请”按钮点击消息通知
        case hwmsdkagent::HWM_CLICK_INJECT_BTN_INVITE:
        {
            // 显示带通讯录的邀请与会人界面
            hwm_ui_demo_inviteDlg inviteDlg;
            nRes = inviteDlg.DoModal();
            break;
        }
    }
}

```

7.4.13 场景 12：注销登录

描述

当需要退出当前的华为云会议帐号的登录时，可以调用该接口完成已登录华为云帐号帐号的注销。

业务流程

使用SDK注销登录时，先调用Logout接口，然后处理回调函数OnLogoutResult。

步骤1 接口调用

1. 调用Logout进行注销登录。

步骤2 处理回调函数

处理回调函数OnLogoutResult。

----结束

示例代码

```

/**
 * 注销
 */
void demoMainMenuDlg::OnBnClickedLogout()
{
    //调用登录接口
    int ret = hwmsdkagent::Logout();
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("Logout error"));
    }
}

/**
 * 注销接口回调

```

```
*/
void demoCallbackProc::OnLogoutResult(hwmsdk::HwmErrCode ret, const char* msg)
{
    CString codeStr;
    codeStr.Format(_T("%d"), ret);
    string msgStr = CTools::UTF82MultiByte(msg);
    CString tips = _T("OnLogoutResult code:") + codeStr + _T(", msg:") + CString(msgStr.c_str());
    AfxMessageBox(tips);
}
```

7.4.14 场景 13：退出

描述

在第三方APP退出时，需要去初始化SDK，否则名称为HwmSdk.exe的进程（Windows SDK的主进程）一直在后台运行。

业务流程

需要退出SDK时，先调用Exit接口，然后处理回调函数OnExitResult。

步骤1 接口调用

1. 调用Exit退出SDK。

步骤2 处理回调函数

处理回调函数OnExitResult。

----结束

示例代码

```
/**
 * 退出sdk
 */
void CdemoBeforeLoginDlg::OnBnClickedButtonExit()
{
    int ret = hwmsdkagent::Exit();
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("Exit error"));
    }
}
```

```
/**
 * 退出回调
 */
void demoCallbackProc::OnExitResult(hwmsdk::HwmErrCode ret, const char* msg)
{
    CString codeStr;
    codeStr.Format(_T("%d"), ret);
    string msgStr = CTools::UTF82MultiByte(msg);
    CString tips = _T("OnExitResult code:") + codeStr + _T(", msg:") + CString(msgStr.c_str());
    AfxMessageBox(tips);
}
```

7.5 接口参考

说明

Windows SDK接口中的字符数组（char[]）类型的参数都需要使用UTF8编码。例外情况在参数说明中描述。

7.5.1 基础配置

7.5.1.1 初始化

Init

接口描述

用于启动并且初始化SDK。

注意事项

1. 调用该接口之前请先修改HwmSdk文件夹下HwmSdk.exe文件的名称。
2. 该接口用于启动并初始化SDK，在调用其他接口之前必须先调用该接口并等待初始化回调结果。
3. 每次调用初始化接口会重新启动SDK，之前所有设置与操作将不复存在。
4. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode Init(HwmInitInfo *initInfo);
```

回调函数

```
virtual void OnInitResult(hwmsdk::HwmErrCode ret, const char* reason, HwmSdkInfo *sdkInfo) {};
```

参数描述

表 7-3 结构体 HwmInitInfo 参数说明

参数	是否必须	类型	描述
exePath	x64平台是必须，win32平台不用传	char[]	SDK启动路径，SDK压缩包中HwmSdk文件夹下的exe文件所在的绝对路径。使用前请将SDK压缩包中HwmSdk文件夹下HwmSdk.exe文件名修改为期望的名称。如果HwmSdk文件夹的路径为：D:\MyApp\SDK\HwmSdk\，exe文件的名称为MySdk.exe，则exePath=D:\MyApp\SDK\HwmSdk\MySdk.exe。 注：需要UTF8编码。

参数	是否必须	类型	描述
logPath	是	char[]	SDK日志路径，如果不设置，则使用默认路径。 注：需要UTF8编码。
userDataPath	是	char[]	SDK用户数据路径，如果不设置，则使用默认路径。 注：需要UTF8编码。
appId	是	char[]	App ID。App ID的申请，请参考《开发指南》的“ App ID的申请 ”章节。
notify	是	HwmAgentNotify	通知对象。继承于基类HwmAgentNotify的类的对象。
callback	是	HwmAgentCallback	接口回调对象。继承于基类HwmAgentCallback的类的对象。
authType	是	HwmAuthType	鉴权方式。
siteType	否	HwmSiteType	站点类型。默认使用国内站点（值置0）。
logKeepDays	否	unsigned int	配置项-日志保存天数，输入有效范围为3-30，默认值0。0为不按天数删除日志。 注意：输入1和2会转换为3，大于30的值会被转换为30。
preferredCamera	否	char[]	优选摄像头列表，填写后，初次打开摄像头按列表中的顺序选择第一个可用的摄像头。可以不填，不填按照默认顺序打开可用的摄像头。

表 7-4 枚举 HwmAuthType 说明

枚举值	描述
HWM_AUTH_TYPE_ACCOUNT_AND_PASSWORD	帐号密码鉴权方式。
HWM_AUTH_TYPE_APPID	App ID鉴权方式。

表 7-5 枚举 HwmSiteType 说明

枚举值	描述
SITE_TYPE_CHINA	国内站点

枚举值	描述
SITE_TYPE_AP	亚太站点

表 7-6 结构体 HwmSdkInfo 参数说明

参数	类型	描述
version	char[]	SDK版本号

说明

推荐使用App ID鉴权方式。App ID的申请，请参考《[开发指南](#)》中的“App ID鉴权介绍”章节。

返回值

表 7-7 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
//创建消息通知处理对象
static demoNotifyProc *notifyObj = new demoNotifyProc();
//创建回调函数处理对象
static demoCallbackProc *callbackObj = new demoCallbackProc();

/**
 *初始化接口，拉起应用
 */
int CdemoBeforeLoginDlg::Init()
{
    //组装入参结构体
    hwm sdkagent::HwmInitInfo initParam;
    memset(&initParam, 0, sizeof(HwmInitInfo));

    //设置sdk路径，假设HwmSdk目录在E:\Hello_World\debug\win32\，假设exe文件名称为MySdk.exe
    std::string path = "E:\\Hello_World\\debug\\win32\\HwmSdk\\MySdk.exe";
    //将路径转换为UTF8格式后拷贝给initParam，此处省略转换代码
    strcpy_s(initParam.exePath, HWM_MAX_FILE_PATH_LEN, path.c_str());
    //指定日志路径，将路径转换为UTF8格式后拷贝给initParam，此处省略转换代码
    std::string logPath = "E:\\Hello_World\\debug\\win32\\MySdk\\log\\";
    strcpy_s(initParam.logPath, HWM_MAX_FILE_PATH_LEN, logPath.c_str());
    //指定数据路径，将路径转换为UTF8格式后拷贝给initParam，此处省略转换代码
    std::string userDataPath = "E:\\Hello_World\\debug\\win32\\MySdk\\UserData\\";
    strcpy_s(initParam.userDataPath, HWM_MAX_FILE_PATH_LEN, userDataPath.c_str());

    //传入申请到的企业AppId
    string appId("fdb8e4699586458bbd10c834872dcc62");
    strncpy_s(initParam.appId, appId.c_str(), HWM_MAX_APPID_LEN);
    initParam.notify = notifyObj;
    initParam.callback = callbackObj;
    initParam.authType = hwm sdkagent::HWM_AUTH_TYPE_APPID;
}
```

```
int ret = hwmsdkagent::Init(&initParam);

return ret;
}
```

说明

- 典型场景和接口参考中的示例代码仅作为示例用的伪代码，不能直接使用。
- 消息通知处理类demoNotifyProc和回调函数处理类demoCallbackProc的定义，请参考SDK包中的Demo代码。

7.5.1.2 退出

Exit

接口描述

用于退出SDK。

注意事项

1. 该接口用于退出SDK，调用该接口之后SDK不可使用。
2. 应用退出时请调用该接口，若不调用该接口，则SDK无法退出，应用退出后SDK进程将继续存在。
3. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode Exit();
```

回调函数

```
virtual void OnExitResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

无。

返回值

表 7-8 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 退出sdk
 */
void CdemoBeforeLoginDlg::OnBnClickedButtonExit()
{
    int ret = hwmsdkagent::Exit();
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
```

```
AfxMessageBox(_T("Exit error"));
}
```

7.5.1.3 设置语言

SetLanguage

接口描述

该接口设置界面显示语言。默认是中文。

注意事项

该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode SetLanguage(HwmLanguage language);
```

回调函数

```
virtual void OnSetLanguageResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

表 7-9 参数说明

参数	是否必须	类型	描述
language	是	HwmLanguage	界面语言枚举。

表 7-10 枚举 HwmLanguage 说明

枚举值	描述
HWM_LANGUAGE_ZH_CN	中文。
HWM_LANGUAGE_EN_GB	英文。
HWM_LANGUAGE_CUSTOMIZED	自定义语言

返回值

表 7-11 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 修改语言
 */
void demoSettingDlg::OnBnClickedButtonSetLanCode()
{
    // 设置语言为中文
    hwmsdkagent::HwmLanguage languageCode = hwmsdkagent::HWM_LANGUAGE_ZH_CN;

    int ret = hwmsdkagent::SetLanguage((hwmsdkagent::HwmLanguage)languageCode);
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("SetLanguage error"));
    }
}
```

SetLanguageWithCode

接口描述

该接口用于自定义设置SDK国际化语言。默认是中文。

注意事项

1. 实现自定义语言，要求在 \HwmSdk\sdkResources\language\ 目录下创建对应的国际化文件。如新增阿拉伯语则可以创建 \ar\temp.js，通过该接口设置的语言应与系统当前语言保持一致
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode SetLanguageWithCode(HwmLanguage language, const char* languageCode);
```

回调函数

```
virtual void OnSetLanguageResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

表 7-12 参数说明

参数	是否必须	类型	描述
language	是	HwmLanguage	界面语言枚举。
languageCode	否	char[]	自定义时对应的语言代码，若是中文/英文，该参数可以传空字符串

返回值

表 7-13 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 修改语言
 */
void demoSettingDlg::OnBnClickedButtonSetLanCode()
{
    // 设置自定义语言为阿拉伯语
    hwmsdkagent::HwmLanguage languageCode = hwmsdkagent::HWM_LANGUAGE_CUSTOMIZED;
    CString code = "ar";
    int ret = hwmsdkagent::SetLanguageWithCode((hwmsdkagent::HwmLanguage)languageCode,
    CTools::UNICODE2UTF(code).c_str());
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("SetLanguageWithCode error"));
    }
}
```

7.5.1.4 开启功能

EnableFeature

接口描述

Windows SDK中有些功能特性是通过开关控制是否打开的，该接口用于配置这些功能的打开或者关闭。

注意事项

1. 枚举中的功能项均有默认值，若对开关状态不做要求，可以不必调用该接口。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode EnableFeature(HwmEnableFeatureType type, bool isEnabled);
```

回调函数

```
virtual void OnEnableFeature(hwmsdk::HwmErrCode ret, const char* msg) {};
```

参数描述

表 7-14 参数说明

参数	是否必须	类型	描述
type	是	HwmEnableFeatureType	功能开关枚举。

参数	是否必须	类型	描述
isEnabled	是	bool	开关状态。

表 7-15 枚举 HwmEnableFeatureType 说明

枚举值	描述
HWM_ENABLE_FEATURE_TYPE_MULTI_MONITOR	是否启动多显示器模式开关。在多显示器场景下，在两个显示器上同时显示桌面共享窗口和视频窗口；在单显示器模式下，开启共享后，隐藏视频窗口。默认为关。
HWM_ENABLE_FEATURE_TYPE_SHOW_HANGUP_ATTENDEE	是否在与会者窗口上显示离会的与会者开关。默认为开。
HWM_ENABLE_FEATURE_TYPE_HOWLING_DETECTION	是否启用啸叫抑制开关状态。默认为开。
HWM_ENABLE_FEATURE_TYPE_HIGH_RESOLUTION	是否启用会议中大画面的1080P高清显示。默认为开。
HWM_ENABLE_FEATURE_TYPE_MIC_SWITCH	麦克风在会议中的初始化开关状态。默认为开。
HWM_ENABLE_FEATURE_TYPE_CAM_SWITCH	摄像头在会议中的初始化开关状态。默认为关。
HWM_ENABLE_FEATURE_TYPE_HIDE_EXTERNAL_LABEL	外部标签在会议中的显示/隐藏状态。默认为关(显示外部标签)。
HWM_ENABLE_FEATURE_AUTO_MIC_VOLUME	自动麦克风音量调整开关，默认为开，如果外接麦克风设备自带有AGC效果时建议关闭
HWM_ENABLE_FEATURE_BREAKOUT_CONF	分组讨论功能开关状态。默认为开。
HWM_ENABLE_FEATURE_CONFIRM_RECORD	录制确认开关状态。默认为关。

返回值

表 7-16 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 多显示器模式开关配置
 */
void demoEnableFeatureDlg::OnBnClickedButtonMultiMonitor()
{
    bool isCheck = true;
    int ret = hwmsdkagent::EnableFeature(hwmsdkagent::HWM_ENABLE_FEATURE_TYPE_MULTI_MONITOR,
isCheck);
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("Enable multi monitor failed"));
    }
}
```

7.5.1.5 设置用户文件保存路径

SetUserSavePath

接口描述

该接口用于设置用户文件（包括本地录制文件）的保存路径。

注意事项

1. 该接口非必需调用，若不设置，则使用默认路径。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode SetUserSavePath(HWMSavePathType type, const char* path);
```

回调函数

```
virtual void OnSetUserSavePath(hwmsdk::HwmErrCode ret, const char* msg) {};
```

参数描述

表 7-17 参数说明

参数	是否必须	类型	描述
type	是	HWMSavePathType	路径类型枚举。
path	是	const char*	绝对路径。 注：需要UTF8编码。

表 7-18 枚举 HwmsavePathType 说明

枚举值	描述
HWM_SAVE_PATH_LOCAL_RECORD	本地录制文件保存路径。

返回值

表 7-19 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 设置用户保存路径
 */
void demoSetUserSavePathDlg::OnBnClickedSetUserSavePath()
{
    std::string path = "E:\\";
    int ret = hwmsdkagent::SetUserSavePath(hwmsdkagent::HWM_SAVE_PATH_LOCAL_RECORD,
    path.c_str());
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("Set save path failed"));
    }
}
```

7.5.1.6 网络模式设置

SetNetworkMode

接口描述

如华为云会议的媒体节点下沉部署到企业，并且管理节点也是通过企业内的代理访问，需要将网络模式设置成“企业内外接入”。

说明

企业内的App无法直接访问公网（即媒体节点下沉和管理节点代理下沉场景）的情况下才需要调用该接口。

注意事项

1. 该接口非必需调用，若不设置，则使用默认模式。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode SetNetworkMode(const HwmNetWorkModeInfo*
networkInfo);
```

回调函数

```
virtual void OnSetNetworkModeResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

表 7-20 参数说明

参数	是否必须	类型	描述
networkInfo	是	HwmNetWork ModelInfo	网络模式信息。

表 7-21 结构体 HwmNetWorkModelInfo 参数说明

参数	是否必须	类型	描述
accessType	是	HwmNetwork AccessType	网络接入类型。
isForcedVpn	是	bool	是否强制走vpn，默认false不强制。

表 7-22 枚举 HwmNetworkAccessType 参数说明

枚举值	描述
ACCESS_TYPE_AUTO	默认值，目前和企业外网接入模式一致。
ACCESS_TYPE_INNER	企业内网接入，用于媒体和信令下沉场景，app从内网接入，信令/媒体消息走代理服务器才能访问云会议服务端。
ACCESS_TYPE_OUTER	企业外网接入，用于非媒体和信令下沉场景，互联网接入，信令/媒体消息可以直接访问云会议服务端。

返回值

表 7-23 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 设置网络模式
```

```

*/
void demoSetNetworkModeDlg::SetNetworkMode()
{
    hwmsdkagent::HwmNetWorkModeInfo networkMode{};
    networkMode.accessType = hwmsdkagent::HwmNetworkAccessType::ACCESS_TYPE_AUTO;
    int ret = hwmsdkagent::SetNetworkMode(&networkMode);
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("Set network mode failed"));
    }
}

```

7.5.1.7 禁用提示音

DisableVoicePrompts

接口描述

禁用提示音。

注意事项

1. 该接口非必需调用，如不调用，则默认播放会中提示音。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```

HWM_SDK_AGENT_API hwmsdk::HwmErrCode DisableVoicePrompts(const HwmVoicePromptsInfo*
voicePromptsInfo);

```

回调函数

```

virtual void OnDisableVoicePromptsResult(hwmsdk::HwmErrCode ret, const char* reason) {};

```

参数描述

表 7-24 参数说明

参数	是否必须	类型	描述
voicePromptsInfo	是	HwmVoicePromptsInfo	语音提示信息。

表 7-25 结构体 HwmVoicePromptsInfo 参数说明

参数	是否必须	类型	描述
isDisableMutePrompt	是	bool	是否禁用静音提示。

返回值

表 7-26 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 禁用提示音
 */
void demoDisableVoicePromptsDlg::DisableVoicePrompts()
{
    hwmsdkagent::HwmVoicePromptsInfo info{};
    info.isDisableMutePrompt = true;

    int ret = hwmsdkagent::DisableVoicePrompts(&info);
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("disable voice prompts error"));
        return;
    }
    CDialogEx::OnOK();
}
```

7.5.2 帐号操作

7.5.2.1 登录

Login

接口描述

该接口用于登录。登录分为华为云会议账户登录和AppID登录，由authType参数决定。

注意事项

1. 创建会议，被邀入会等功能只有在登录后才能使用，若不登录，将无法体验完整会议功能。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode Login(HwmLoginInfo *loginInfo);
```

回调函数

```
virtual void OnLoginResult(hwmsdk::HwmErrCode ret, const char* reason, const HwmLoginResult* loginResult) {};
```

参数描述

表 7-27 结构体 HwmLoginInfo 参数说明

参数	是否必须	类型	描述
authType	是	HwmAuthType	登录类型。参考表2
accountAndPasswordAuthInfo	否	HwmAccountAndPasswordAuthInfo	帐号密码登录信息。选择“帐号密码鉴权方式”时需要填写。
appldAuthInfo	否	HwmAppldAuthInfo	App ID登录信息。选择“App ID鉴权方式”时需要填写。

表 7-28 结构体 HwmAccountAndPasswordAuthInfo 参数说明

参数	是否必须	类型	描述
account	是	char[]	登录帐号。
password	是	char[]	登录密码。

表 7-29 结构体 HwmAppldAuthInfo 参数说明

参数	是否必须	类型	描述
thirdUserId	是	char[]	第三方用户ID。
corpId	否	char[]	设置该用户所在的企业ID。仅在SP模式下需要填写，单企业模式不要填写，否则会鉴权失败。
userName	否	char[]	用户名称(如不填，默认与thirdUserId一致)。
signature	是	char[]	从第三方服务端获取的鉴权签名。
nonce	是	char[]	从第三方服务端获取的Nonce值，用于华为云服务端验证鉴权签名有效性。
expireTime	是	long long	从第三方服务端获取的鉴权签名有效期的时间戳。0表示永不超时。
email	否	char[]	用户邮箱信息。
phoneNumber	否	char[]	用户手机号码。
deptCode	否	char[]	部门ID

返回值

表 7-30 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。
HwmLoginResult	登录结果信息。

表 7-31 结构体 HwmLoginResult 参数说明

参数	类型	描述
userId	char[]	UUID(由系统分配的真实的唯一用户标识)。
account	char[]	帐号。
thirdAccount	char[]	第三方帐号。

示例代码

```
/**
 * 登录处理
 */
int demoLoginDlg::Login()
{
    int ret;
    //设置App ID登录参数
    string userId= "admin@1";
    string userName= "Mike";
    string email= "xxx@xxx.com";
    string phoneNumber= "1234567890";
    /**
     * 此处省略第三方服务器做的数据转换，最后获取到nonce和signature
     */
    strncpy_s(loginParam.appIdAuthInfo.thirdUserId, userId.c_str(), HWM_MAX_USER_ID_LEN);
    strncpy_s(loginParam.appIdAuthInfo.userName, userName.c_str(), HWM_MAX_USER_NAME_LEN);
    strncpy_s(loginParam.appIdAuthInfo.signature, signature.c_str(), HWM_MAX_SIGNATURE_LEN);
    strncpy_s(loginParam.appIdAuthInfo.nonce, nonce.c_str(), HWM_APPID_NONCE_LEN );
    strncpy_s(loginParam.appIdAuthInfo.email, email.c_str(), HWM_MAX_EMAIL_LEN);
    strncpy_s(loginParam.appIdAuthInfo.phoneNumber, phoneNumber.c_str(),
    HWM_MAX_PHONE_NUM_LEN);
    loginParam.appIdAuthInfo.expireTime = 1598398920; // utc时间，单位s

    loginParam.authType = hwmsdkagent::HWM_AUTH_TYPE_APPID;
    //调用登录接口
    ret = hwmsdkagent::Login(&loginParam);
    return ret;
}
```

7.5.2.2 SSO 登录

LoginBySSO

接口描述

该接口用于通过SSO的方式登录和鉴权。需要先通过GetSSOAuthorizeUrl接口获取到鉴权code。GetSSOAuthorizeUrl通过传域名获取到鉴权网站，用户在网站上输入账号+密码或者扫码完成鉴权获取到临时code(使用code完成登录)。

注意事项

1. 创建会议，被邀入会等功能只有在登录后才能使用，若不登录，将无法体验完整会议功能。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode LoginBySSO(HwmSSOAuthInfo *ssoAuthInfo);
```

回调函数

```
virtual void OnLoginBySSOResult(hwmsdk::HwmErrCode ret, const char* reason, const HwmLoginResult* loginResult) {};
```

参数描述

表 7-32 结构体 HwmSSOAuthInfo 参数说明

参数	是否必须	类型	描述
code	是	char[]	授权码
domain	是	char[]	企业域名。
authServerType	是	AuthServerType	鉴权服务类型

表 7-33 枚举 AuthServerType 参数说明

参数	描述
AUTH_SERVER_TYPE_OAUTH2	单点登录。

返回值

表 7-34 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。
HwmLoginResult	登录结果信息。参考“ 结构体HwmLoginResult参数说明 ”。

示例代码

```
/**
 * 登录处理
 */
int demoLoginBySSODlg::LoginBySSO()
{
    int ret;
    //填入登录参数
    string domain= "admin@1";
    string code= "XXXXXXXXXXXXXXXXXXXXXXX";
    hwmsdkagent::HwmSSOAuthInfo ssoAuthInfo{};
    strncpy_s(ssoAuthInfo.domain, domain.c_str(), HWM_MAX_SSO_DOMAIN_LEN);
    strncpy_s(ssoAuthInfo.code, code.c_str(), HWM_MAX_SSO_CODE_LEN);
    ssoAuthInfo.authServerType = hwmsdkagent::AUTH_SERVER_TYPE_OAUTH2;

    //调用登录接口
    ret = hwmsdkagent::LoginBySSO(&ssoAuthInfo);
    return ret;
}
```

7.5.2.3 注销

Logout

接口描述

该接口用于注销已登录的华为云会议帐号。

注意事项

1. 请在已登录的场景下调用。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode Logout();
```

回调函数

```
virtual void OnLogoutResult(hwmsdk::HwmErrCode ret, const char* reason, const HwmLogoutResult* logoutResult) {};
```

参数描述

无。

返回值

表 7-35 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。
HwmLogoutResult	登出结果信息。

表 7-36 结构体 HwmLogoutResult 参数说明

参数	类型	描述
userId	char[]	UUID(由系统分配的真实的唯一用户标识)。
account	char[]	帐号。
thirdAccount	char[]	第三方帐号。

示例代码

```
/**
 * 注销
 */
void demoMainMenuDlg::OnBnClickedLogout()
{
    //调用注销接口
    int ret = hwmsdkagent::Logout();
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("Logout error"));
    }
}
```

7.5.2.4 上传个人头像

UploadSelfAvatar

接口描述

该接口用于上传当前自己的头像到会议服务器的通讯录。

注意事项

1. 请在已登录状态下调用。
2. 上传的图片文件格式仅支持jpg/png格式，大小需要限制在200KB以下。
3. 为保证头像展示效果，建议图片宽高比为1:1

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode UploadSelfAvatar(const HwmSelfAvatarInfo *selfAvatarInfo);
```

回调函数

```
virtual void OnUploadSelfAvatarResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

表 7-37 结构体 HwmSelfAvatarInfo 参数说明

参数	是否必须	类型	描述
path	是	char[]	图片全路径。

```

/**
 * 上传个人头像
 */
int demoUploadSelfAvatarDlg::UploadSelfAvatar()
{
    int ret;
    //填入个人头像参数
    string path= "to/path/image.png";
    hwmsdkagent::HwmSelfAvatarInfo selfAvatarInfo{};
    strncpy_s(selfAvatarInfo.path, path.c_str(), HWM_MAX_FILE_PATH_LEN);

    //调用上传个人头像接口
    ret = hwmsdkagent::UploadSelfAvatar(&selfAvatarInfo);
    return ret;
}

```

7.5.3 界面定制

7.5.3.1 界面配置

Config

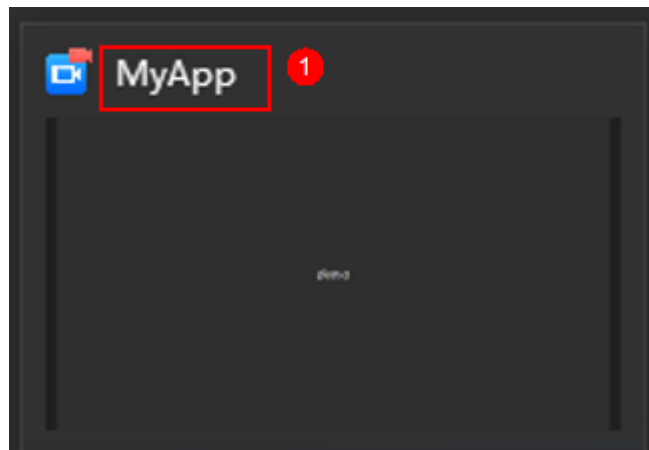
接口描述

Windows SDK中已经提供了会议中的界面，界面上的部分元素是可以自定义的。可以自定义的界面元素如下：

- **会议窗口名称**

会议开始后，在Windows任务栏上会生成一个会议的图标，鼠标指向该图标时会显示会议的缩略窗口，该窗口的名称可以自定义。

图 7-34 会议窗口名称



📖 说明

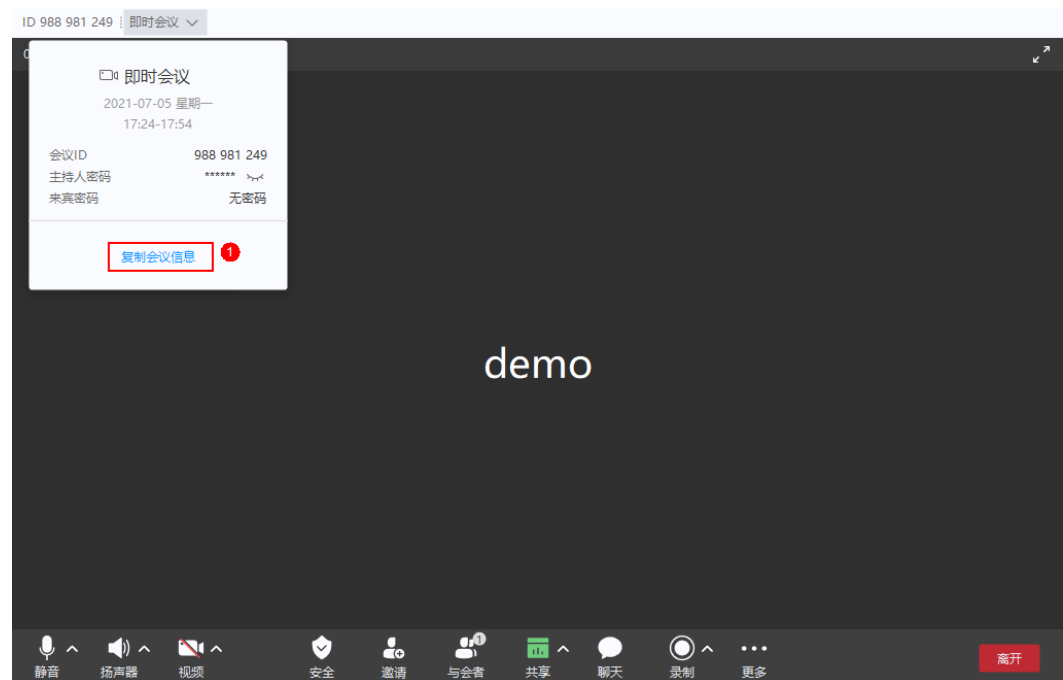
会议窗口的图标，可以替换HwmSdk\sdkResources\nativeimg目录下的video.ico图标文件。

- **会议信息及会议链接前缀**

会议链接的域名前缀默认是<https://bmeeting.huaweicloud.com/>（如<https://bmeeting.huaweicloud.com/#/j/982989574>），如果入会Portal是第三方自己开发的，这个前缀需要自定义。

整个会议信息也可以自定义显示或者隐藏。

图 7-35 会议信息



- **来电窗口**

呼叫来电窗口和会议来电窗口默认在收到来电时弹出，可配置成隐藏，在收到来电通知时调用呼叫来电应答接口、会议来电应答接口来应答：

图 7-36 来电窗口



- **会控按钮**

1. 删除或者修改SDK中自带的会控按钮。具体可以删除和修改的按钮见[按钮ID定义表](#)。
2. 自定义新增按钮。具体用例见[自定义会控按钮](#)。

图 7-37 底部工具栏和更多菜单

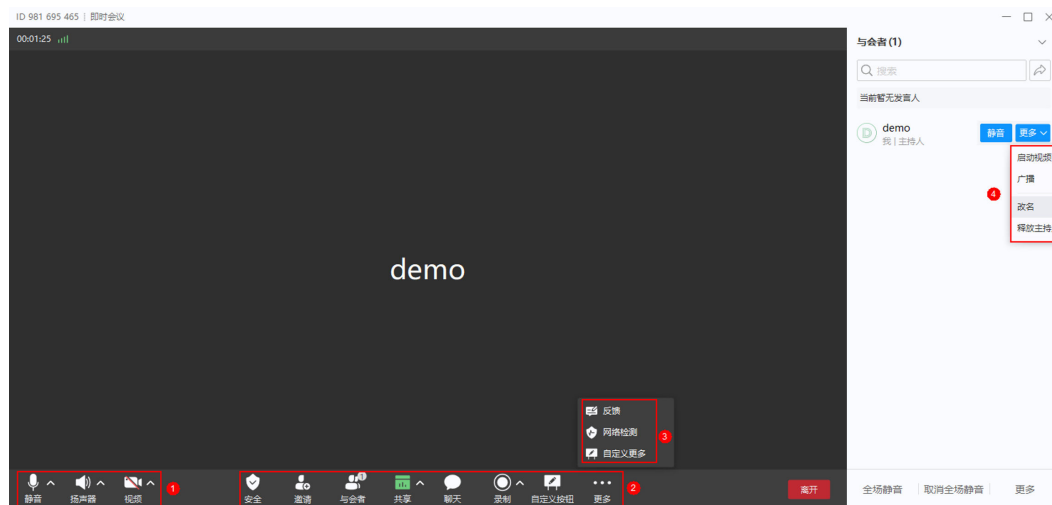


图 7-38 共享工具栏和更多菜单

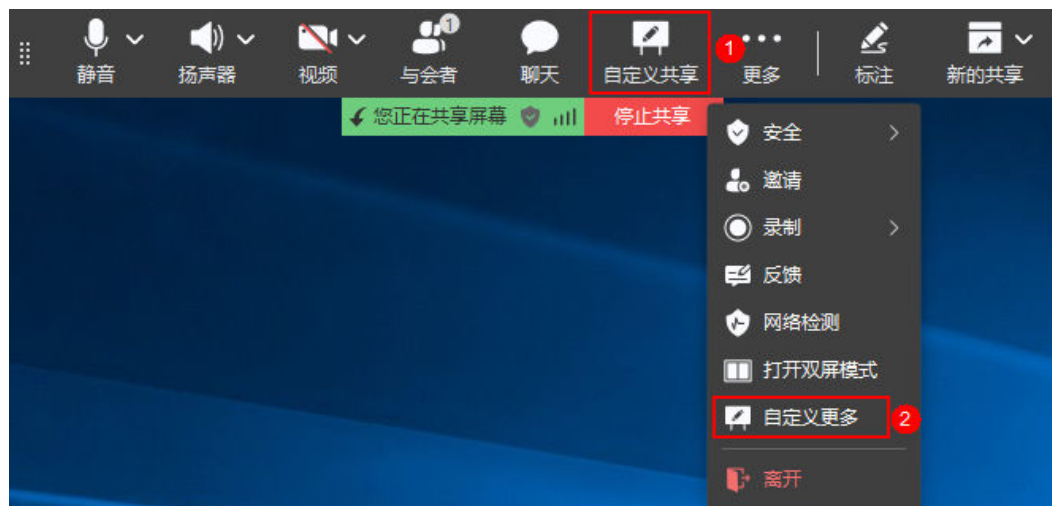


图 7-39 自定义子菜单项：



- 自定义会议信息窗口

图 7-40 自定义会议信息框

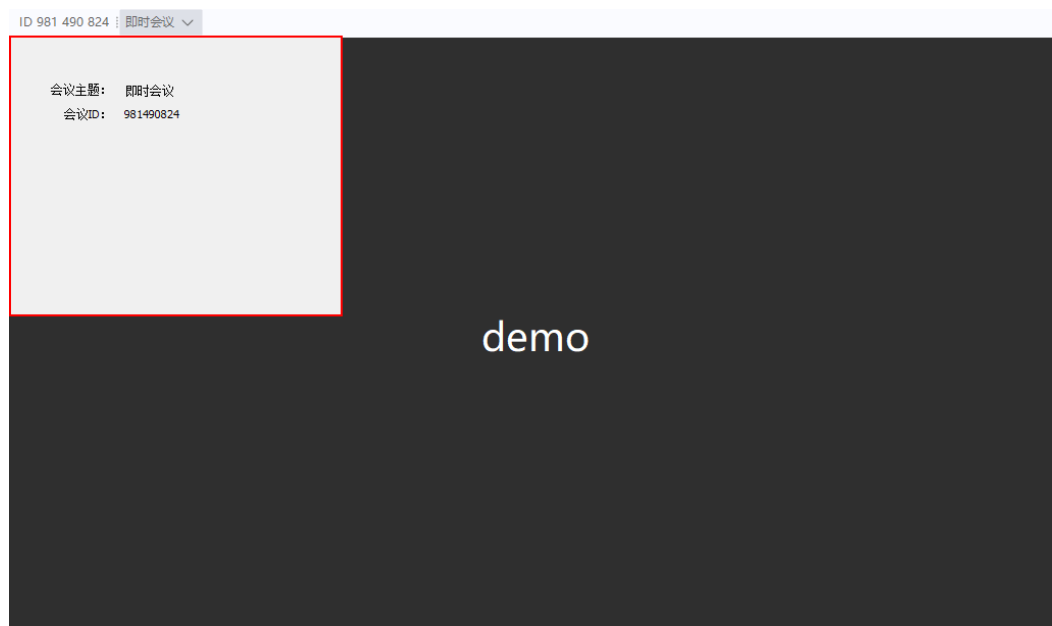


图 7-41 状态工具栏

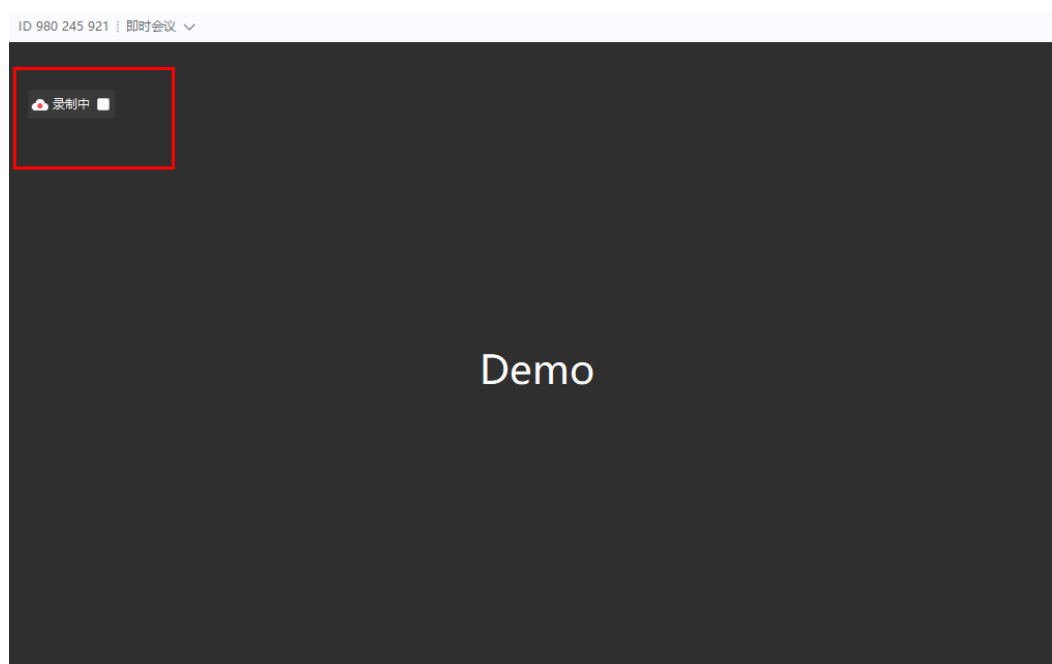


图 7-42 与会者详情头像点击



说明

自定义会议信息窗口请参考“[显示会议信息框](#)”和“[销毁会议信息框](#)”。

注意事项

1. 只有需要更改会议界面默认显示样式时，才需要调用该接口，若默认样式已满足要求，可以不调用该接口。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode Config(char uiConfig[HWM_MAX_CONFIG_LEN]);
```

说明

Config接口不能连续调用，如果需要调用两次Config接口，一定要等第一次OnConfigResult回调函数处理完成后，再调用第二次Config。

回调函数

```
virtual void OnConfigResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

表 7-38 参数说明

参数	是否必须	类型	描述
uiConfig	是	char[]	Json对象字符串{"frame":object}，object对象有格式要求，若不匹配则配置不生效。详细规则见下表2、表3、表4、表5。

表 7-39 frame 对象介绍

key	value 类型	描述
name	string	会议窗口名称。如图1中的窗口名称（ MyApp ）。
linkPrefix	string	会议窗口中分享的会议链接前缀。如图2中复制出来的会议链接。
confMenu	object	会议窗口上的按钮配置对象。如图4和图5中的4类按钮。
isUseExternalConfInfo Wnd	bool	会议窗口中自定义分享会议信息框开关。如图6中的窗口。"true"：开启；"false"：关闭，默认 "false" 关闭。
isUseExternalAddressBook	bool	会议窗口中使用外部通讯录头像信息的开关。"true": 使用；"false": 不使用，默认 "false" 不使用
isHideCallIncomingWindow	bool	是否隐藏呼叫来电弹窗。"true":隐藏，呼叫来电不弹窗；"false"不隐藏，正常弹窗，默认不隐藏，正常弹窗。
isHideConfIncomingWindow	bool	是否隐藏会议来电弹窗。"true":隐藏，会议来电不弹窗；"false"不隐藏，正常弹窗，默认不隐藏，正常弹窗。
defaultWindowMode	int	会议窗口模式。0 通常；1最大化；2全屏。默认通常。
isCloseSideWindow	bool	是否关闭侧边窗口: "false": 不关闭; "true":关闭。默认不关闭。
flowControlInitShowPos	int	共享工具栏初始化位置: 0: 顶部 1: 底部。默认显示顶部。
defaultWindowMonitor	int	会议窗口入会时默认创建在哪个显示器: 0主显示器, 1副显示器。默认在主显示器。
isHideTransferHostBeforeLeave	bool	是否隐藏离会时指定主持人功能按钮: "false": 不隐藏; "true":隐藏。默认不隐藏。
isUploadKeyLog	bool	是否自动收集日志的开关。"true": 收集; "false": 不收集。默认"true"收集。

表 7-40 confMenu 对象介绍

key	value 类型	描述
toolBar	object	会议窗口上的按钮配置对象。如图4和图5中的4类按钮。

key	value类型	描述
titleBar	object	会议窗口上的标题栏配置对象。如图2中的会议信息。
statusBar	object	会议窗口左上角的状态栏配置对象。如图8中的按钮

表 7-41 toolBar 对象子 key 介绍

key	value类型	描述
button	object[]	会议窗口上的按钮配置对象。如图4和图5中的4类按钮和菜单。
customButton	object[]	会议窗口上的第三方自定义按钮配置对象。如图4中的③④和图5中的①②。

表 7-42 statusBar 对象子 key 介绍

key	value类型	描述
button	object[]	会议窗口左上角的状态栏上的按钮配置对象。如图8中的按钮。

表 7-43 button 对象子 key 介绍

key	value类型	描述
id	string	按钮的唯一标识。具体id与按钮的对应关系见下表6。
showAsAction	string	是否显示。只有2个取值，“never”：不显示，“ifRoom”：显示。
isCustomizedClick	bool	是否点击注入。true：点击注入，用户点击按钮时发送通知消息，SDK不做点击响应处理。

表 7-44 customButton 对象子 key 介绍

key	value类型	描述
id	string	第三方自定义的按钮id。如“customMenu”。
title	string	第三方自定义按钮的显示的名称。如“自定义按钮”。

key	value类型	描述
buttonImg	string	第三方自定义按钮的图标路径，使用绝对路径。如“to/path/image.svg”，支持svg/png等格式 注：需要UTF8编码。
buttonPos	int	第三方自定义按钮的位置。0：工具栏更多菜单，1：底部工具栏，2：共享工具栏
subMenu	object[]	第三方自定义按钮的子菜单项。*（最多配置到三级：深入两次）

表 7-45 subMenu 对象子 key 介绍

key	value类型	描述
id	string	第三方自定义的按钮id。如“customMenu”。
title	string	第三方自定义按钮的显示的名称。如“自定义按钮”。
buttonImg	string	第三方自定义按钮的图标路径，使用绝对路径。如“to/path/image.svg”，支持svg/png等格式 注：需要UTF8编码。
subMenu	object[]	第三方自定义按钮的子菜单项。

表 7-46 按钮 id 介绍

按钮名称	按钮id	说明
麦克风	"microphone"	可隐藏
扬声器	"speaker"	可隐藏
摄像头	"camera"	可隐藏
邀请	"invite"	可隐藏，可注入
离开	"leave"	可隐藏，可注入
共享	"share"	可隐藏，可注入
与会者	"attendee"	可隐藏，可注入
呼叫其它号码	"call_other_number"	可隐藏，可注入
反馈	"feedback"	可隐藏，可注入
举手/手放下	"hands_up"	可隐藏

按钮名称	按钮id	说明
申请主持人/释放主持人	"change_chairman_auth"	可隐藏
改名	"rename"	可隐藏
锁定会议	"lock_conf"	可隐藏
标注	"annotation"	可隐藏
授予远程控制	"remote_control"	可隐藏
停止共享	"stop_share"	可隐藏
聊天	"chat"	可隐藏，可注入
导出与会者	"export_participants"	可隐藏
允许入会	"attendee_access_control"	可隐藏
允许共享	"lock_Share"	可隐藏
字幕功能：开启字幕、字幕翻译	"subtitle"	可隐藏
安全	"security"	可隐藏
邀请共享、取消共享	"invite_or_cancel_share"	可隐藏
隐藏无视频与会者	"hide_no_video_view"	可隐藏
视频画面智能排序	"video_smart_sorting"	可隐藏
虚拟背景	"virtual_background"	可隐藏
网络检测	"network_detection"	可隐藏
全场静音/取消全场静音	"mute_all"	可隐藏
允许与会者解除静音	"allow_unmute"	可隐藏
打开/关闭双屏模式	"dual_screen"	可隐藏
录制	"record"	可隐藏
投票	"vote"	可隐藏
共享画面质量策略：清晰度优先、流畅度优先	"share_strategy"	可隐藏
共享时隐藏视频窗口	"share_hide_video"	可隐藏
共享时包含电脑声音	"share_audio"	可隐藏
复制会议信息	"copy_conf_info"	可注入
全员移入等候室	"moveall_to_waitingroom"	可隐藏

按钮名称	按钮id	说明
与会者详情	"participant_detail"	可注入，见图9

表 7-47 titleBar 对象子 key 介绍

key	value类型	描述
confDetail	object[]	如图2中的会议信息。showAsAction设置成never后，不显示会议信息。

表 7-48 confDetail 的对象 key 介绍

key	value类型	描述
showAsAction	string	是否显示。只有2个取值，“never”：不显示，“ifRoom”：显示。
button	object[]	会议信息窗口上的按钮配置对象。

按照上面的定义，完整的可定制的界面元素的json格式配置项如下：

```
{
  "frame":{
    "name":"MyApp",
    "linkPrefix":"https://www.demo.com/",
    "isUseExternalConfInfoWnd":false,
    "isUseExternalAddressBook":false,
    "confMenu":{
      "titleBar":{
        "confDetail":{
          "showAsAction":"never"
        }
        "button":[
          {
            "id":"copy_conf_info",
            "isCustomizedClick":false
          }
        ]
      },
      "statusBar": {
        "button": [
          {
            "id": "record",
            "showAsAction": "never"
          }
        ]
      },
      "toolBar":{
        "customButton": [
          {
            "id": "customMenu",
            "title": "自定义按钮",
            "buttonImg": "../native/default/svg/icon-whiteboard.svg",
            "buttonPos": 1
          }
        ],
        "button":[
          {
```

```
"id": "microphone",
"showAsAction": "ifRoom"
},
{
  "id": "speaker",
  "showAsAction": "ifRoom"
},
{
  "id": "camera",
  "showAsAction": "ifRoom"
},
{
  "id": "invite",
  "showAsAction": "ifRoom",
  "isCustomizedClick": true
},
{
  "id": "leave",
  "showAsAction": "ifRoom",
  "isCustomizedClick": true
},
{
  "id": "share",
  "showAsAction": "never",
  "isCustomizedClick": true
},
{
  "id": "attendee",
  "showAsAction": "ifRoom",
  "isCustomizedClick": true
},
{
  "id": "call_other_number",
  "showAsAction": "ifRoom",
  "isCustomizedClick": true
},
{
  "id": "feedback",
  "showAsAction": "never"
},
{
  "id": "hands_up",
  "showAsAction": "never"
},
{
  "id": "change_chairman_auth",
  "showAsAction": "never",
  "isCustomizedClick": false
},
{
  "id": "rename",
  "showAsAction": "never"
},
{
  "id": "lock_conf",
  "showAsAction": "never"
},
{
  "id": "annotation",
  "showAsAction": "never"
},
{
  "id": "remote_control",
  "showAsAction": "never"
},
{
  "id": "stop_share",
  "showAsAction": "never"
},
}
```

```
{
    {
        "id":"chat",
        "showAsAction":"ifRoom",
        "isCustomizedClick":true
    },
    {
        "id":"export_participants",
        "showAsAction":"never"
    },
    {
        "id":"attendee_access_control",
        "showAsAction":"never"
    },
    {
        "id":"network_detection",
        "showAsAction":"never"
    },
    {
        "id":"mute_all",
        "showAsAction":"never"
    },
    {
        "id":"allow_unmute",
        "showAsAction":"never"
    },
    {
        "id":"dual_screen",
        "showAsAction":"never"
    },
    {
        "id":"record",
        "showAsAction":"never"
    },
    {
        "id":"vote",
        "showAsAction":"never"
    }
}
}
```

说明

由于函数Config的uiConfig是字符数组，需要把上述的json转换成字符串形式，请参考下面的示例代码。

示例代码

```
/**
 * UI界面配置
 */
int hwmSDKConfigUI::hwmSDKStartUIConfigTotal()
{
    CString start = "{\\\"frame\\\":{";

    //会议窗口名称
    CString name = "\\\"name\\\":\\\"MyApp\\\"";

    //会议链接前缀
    CString linkPrefix = "\\\"linkPrefix\\\":\\\"https://www.demo.com/\\\"";
    //自定义外部会议信息框开关
    CString isUseExternalConfInfoWnd = "\\\"isUseExternalConfInfoWnd\\\":\\\"false\\\"";
    //外部通讯录开关
    CString isUseExternalAddressBook = "\\\"isUseExternalAddressBook\\\":\\\"false\\\"";
    CString menuStart = "\\\"confMenu\\\":{";
    //会议信息
    CString titleBar = "\\\"titleBar\\\":{\\\"confDetail\\\":{\\\"showAsAction\\\":\\\"ifRoom\\\"}}";
    CString toolBarStart = "\\\"toolBar\\\":{";
```



```

//自定义按钮
CString customMenuStart = "\\customButton\":[
CString customMenu1 = "{id:\customMenu1\,title:\自定义按钮1\}";
CString customMenu2 = "{id:\customMenu2\,title:\自定义按钮2\}";
CString customMenuEnd = "]";

//会控按钮
CString buttonStart = "\\button\":[
CString inviteButton = "{id:\invite\,showAsAction:\ifRoom\,isCustomizedClick:true}";
CString leaveButton = "{id:\leave\,showAsAction:\ifRoom\,isCustomizedClick:true}";
CString callButton = "{id:\call_other_number\,showAsAction:\ifRoom\,isCustomizedClick:true}";
CString attendeeButton = "{id:\attendee\,showAsAction:\ifRoom\,isCustomizedClick:true}";
CString shareButton = "{id:\share\,showAsAction:\ifRoom\,isCustomizedClick:true}";
CString annotationButton = "{id:\annotation\,showAsAction:\never\}";
CString stopshareButton = "{id:\stop_share\,showAsAction:\never\}";
CString feedbackButton = "{id:\feedback\,showAsAction:\never\}";
CString remotecontrolButton = "{id:\remote_control\,showAsAction:\never\}";
CString handsupButton = "{id:\hands_up\,showAsAction:\never\}";
CString changechairmanButton = "{id:\change_chairman_auth\,showAsAction:\never\}";
CString renameButton = "{id:\rename\,showAsAction:\never\}";
CString lockconfButton = "{id:\lock_conf\,showAsAction:\never\}";
CString chatButton = "{id:\chat\,showAsAction:\never\}";
CString exportButton = "{id:\export_participants\,showAsAction:\never\}";
CString accessButton = "{id:\attendee_access_control\,showAsAction:\never\}";

CString buttonEnd = "]";
CString toolBarEnd = "}";
CString menuEnd = "}";
CString end = "}}";

CString customMenu = customMenuStart + customMenu1 + "," + customMenu2 + customMenuEnd;
CString button = buttonStart + inviteButton + "," + leaveButton + "," + callButton + "," + attendeeButton + "," + shareButton + "," + annotationButton + "," + stopshareButton + "," + feedbackButton + "," + remotecontrolButton + "," + handsupButton + "," + changechairmanButton + "," + renameButton + "," + lockconfButton + "," + chatButton + "," + exportButton + "," + accessButton + buttonEnd;

CString menu = menuStart + "," + titleBar + toolBarStart + customMenu + "," + button + toolBarEnd + menuEnd;
CString config = start + name + "," + linkPrefix + "," + isUseExternalConfInfoWnd + "," + isUseExternalAddressBook + "," + menu + end;

//转换成UTF8格式
string uiConfig = CTools::UNICODE2UTF(config);
int ret = hwmsdkagent::Config(uiConfig);
return ret;
}

```

7.5.3.2 自定义会控按钮

- 自定义新增按钮。底部工具栏按钮、共享工具栏和更多菜单中的按钮支持第三方开发者自定义添加。目前底部工具栏和共享工具栏限制自定义按钮个数为1，超出的自定义按钮会移入更多菜单，自定义底部/共享工具栏按钮的顺序固定在“更多”按钮前；自定义更多菜单按钮会在底部和共享工具栏更多按钮中同时生效，底部工具栏的更多菜单中自定义按钮添加在菜单栏最后，共享工具栏的更多菜单中自定义按钮添加在“离开”按钮前。
- 删除自定义按钮。通过Config接口删除customButton中的按钮配置，关于清空自定义按钮操作，若赋值customButton: []，则清空全量自定义按钮；若customButton下未配置某位置下的自定义按钮，则清空该位置下的自定义按钮。
- 自定义按钮子菜单功能说明：
 - 当 subMenu 为空时，点击按钮通知对应按钮 id 被点击；

- 当 subMenu 不为空时，点击按钮显示 subMenu 二级菜单，并且不通知；

📖 说明

通过Config接口增删改自定义按钮时，需要传对应位置的全量自定义按钮，支持动态刷新底部/共享工具栏，更多菜单不支持，而是在重新生成时刷新。刷新单个自定义按钮请参考[刷新自定义按钮配置](#)。

- **增加自定义按钮**

以添加自定义带子菜单的底部工具栏按钮、共享工具栏按钮和更多菜单按钮为例：

```
{
  "frame":{
    "confMenu":{
      "toolBar":{
        "customButton": [
          {
            "id": "customMenu",
            "title": "自定义按钮",
            "buttonImg": "../../native/default/svg/icon-whiteboard.svg",
            "buttonPos": 1,
            "subMenu":[
              {
                "id": "customSecondMenu",
                "title": "二级菜单",
                "buttonImg": "../../native/default/svg/icon-whiteboard.svg"
              }
            ]
          },
          {
            "id": "customMore",
            "title": "自定义更多",
            "buttonImg": "../../native/default/svg/icon-whiteboard.svg",
            "buttonPos": 0
          },
          {
            "id": "customShare",
            "title": "自定义共享",
            "buttonImg": "../../native/default/svg/icon-whiteboard.svg",
            "buttonPos": 2
          }
        ]
      }
    }
  }
}
```



- **更新自定义按钮**

注意确保id和buttonPos正确，修改按钮的其他信息。以重命名底部工具栏按钮为例：

```
{
  "frame":{
    "confMenu":{
```

```

"toolBar":{
  "customButton": [
    {
      "id": "customMenu",
      "title": "重命名按钮",
      "buttonImg": "../../native/default/svg/icon-whiteboard.svg",
      "buttonPos": 1,
      "subMenu":[
        {
          "id": "customSecondMenu",
          "title": "二级菜单",
          "buttonImg": "../../native/default/svg/icon-whiteboard.svg"
        }
      ]
    },
    {
      "id": "customMore",
      "title": "自定义更多",
      "buttonImg": "../../native/default/svg/icon-whiteboard.svg",
      "buttonPos": 0
    },
    {
      "id": "customShare",
      "title": "自定义共享",
      "buttonImg": "../../native/default/svg/icon-whiteboard.svg",
      "buttonPos": 2
    }
  ]
}
}
}
}
}
}

```



- **删除部分自定义按钮**

customButton删除对应按钮，同时需要保留其他自定义按钮。以删除底部工具栏自定义按钮为例：

```

{
  "frame":{
    "confMenu":{
      "toolBar":{
        "customButton": [
          {
            "id": "customMore",
            "title": "自定义更多",
            "buttonImg": "../../native/default/svg/icon-whiteboard.svg",
            "buttonPos": 0
          },
          {
            "id": "customShare",
            "title": "自定义共享",
            "buttonImg": "../../native/default/svg/icon-whiteboard.svg",
            "buttonPos": 2
          }
        ]
      }
    }
  }
}

```

```
}  
}  
}
```



- **清空某个位置下的自定义按钮**

customButton中删除该位置下的所有自定义按钮即可，若customButton下未配置某位置下的自定义按钮，则会清空该位置下的自定义按钮。以清空底部工具栏自定义按钮为例：

```
{  
  "frame":{  
    "confMenu":{  
      "toolBar":{  
        "customButton": [  
          {  
            "id": "customMore",  
            "title": "自定义更多",  
            "buttonImg": "../../native/default/svg/icon-whiteboard.svg",  
            "buttonPos": 0  
          },  
          {  
            "id": "customShare",  
            "title": "自定义共享",  
            "buttonImg": "../../native/default/svg/icon-whiteboard.svg",  
            "buttonPos": 2  
          },  
        ]  
      }  
    }  
  }  
}
```



- **清空全量自定义按钮**

customButton传入空数组，会清空所有位置下自定义按钮。清空示例如下：

```
{  
  "frame":{  
    "confMenu":{  
      "toolBar":{  
        "customButton": [  
        ]  
      }  
    }  
  }  
}
```

```

    }
  }
}
}
}

```



7.5.3.3 清除界面配置

CleanConfig

接口描述

该接口清除所有会议界面相关的配置，恢复默认状态。

注意事项

1. 有调用Config接口并且希望清除Config接口所做的配置时，才需要调用该接口。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode CleanConfig();
```

回调函数

```
virtual void OnCleanConfigResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

无。

返回值

表 7-49 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```

/**
 * UI界面配置清除
 */
int hwmSDKConfigUI::hwmSDKConfigUI()
{
    //清除配置
    int ret = hwmsdkagent::CleanConfig();
}

```

```
LOG_INFO("Call hwmsdkagent::CleanConfig ret is " << ret);
}
```

7.5.3.4 刷新自定义按钮配置

UpdateCustomButtonInfo

接口描述

会中刷新自定义按钮相关的配置，支持底部工具栏和共享工具栏。

注意事项

1. 有调用Config接口并希望在会中刷新自定义按钮配置。
2. 只对Config中已配置的自定义按钮生效。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode UpdateCustomButtonInfo(const HwmCustomButtonInfo *btnInfo);
```

回调函数

```
virtual void OnUpdateCustomButtonInfoResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

表 7-50 参数说明

参数	是否必须	类型	描述
btnInfo	是	HwmCustomButtonInfo	自定义按钮信息

表 7-51 结构体 HwmCustomButtonInfo 参数说明

参数	是否必须	类型	描述
HwmCustomButtonPos	是	HwmCustomButtonPos	自定义按钮的位置。
buttonKey	是	char[]	自定义按钮标识，需要和config中配置的自定义按钮id一致。
buttonTitle	是	char[]	自定义按钮显示名称。
buttonImg	是	char[]	自定义按钮的图标路径，使用绝对路径。

表 7-52 枚举 HwmCustomButtonPos 参数说明

参数	描述
CUSTOM_BUTTON_POS_TOOL_BAR	底部工具栏
CUSTOM_BUTTON_POS_SHARE_TOOL_BAR	共享工具栏

返回值

表 7-53 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 更新自定义按钮配置
 */
int hwmSDKConfigUI::hwmSDKConfigUI()
{
    hwmsdkagent::HwmCustomButtonInfo btnInfo{};
    btnInfo.buttonPos = hwmsdkagent::CUSTOM_BUTTON_POS_TOOL_BAR;
    string buttonKey = CTools::UNICODE2UTF(CString("customMenu"));
    strcpy_s(btnInfo.buttonKey, sizeof(btnInfo.buttonKey), buttonKey.c_str());
    string buttonTitle = CTools::UNICODE2UTF(CString("update"));
    strcpy_s(btnInfo.buttonTitle, sizeof(btnInfo.buttonTitle), buttonTitle.c_str());
    string buttonIconPath = CTools::UNICODE2UTF(CString("path/to/image"));
    strcpy_s(btnInfo.buttonImg, sizeof(btnInfo.buttonImg), buttonIconPath.c_str());
    //更新自定义按钮配置
    int ret = hwmsdkagent::UpdateCustomButtonInfo(&btnInfo);
    LOG_INFO("Call hwmsdkagent::UpdateCustomButtonInfo ret is " << ret);
}
```

7.5.3.5 自定义会中水印

SetWaterMark

接口描述

该接口用于设置水印自定义内容注入，若未进行设置，将默认使用内部优先级来展示水印内容。

图 7-43 自定义水印



注意事项

1. 该接口必须在会前调用，会中调用不会即时生效。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。
3. 企业级水印开关未打开的场景下，水印自定义内容注入也不会生效。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode SetWaterMark(const HwmWaterMarkInfo * wtInfo);
```

回调函数

```
virtual void OnSetWaterMarkInjectInfoResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

表 7-54 参数说明

参数	是否必须	类型	描述
wtInfo	是	HwmWaterMarkInfo	设置水印注入信息。

表 7-55 结构体 HwmWaterMarkInfo 参数说明

参数	类型	描述
content	char[]	设置用户自定义水印注入内容。
color	HwmWaterMarkColorType	设置水印字体颜色。

表 7-56 枚举 HwmWaterMarkColorType 说明

枚举值	描述
HWM_WATERMARK_COLOR_TYPE_LIGHT	浅色 #CCCCCC。
HWM_WATERMARK_COLOR_TYPE_STANDARD	标准色 #999999。
HWM_WATERMARK_COLOR_TYPE_DARKER	深色 #666666。

返回值

表 7-57 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 设置水印
 */
void demoWaterMarkInfoDlg::OnBnClickedSetWaterMarkInjectInfo()
{
    std::string content = "admin@1";
    unsigned int color = hwmsdkagent::HWM_WATERMARK_COLOR_TYPE_LIGHT;
    hwmsdkagent::HwmWaterMarkInfo injectInfo = {0};

    strncpy_s(injectInfo.content, content.c_str(), HWM_MAX_WATERMARK_CONTENT_LEN);
    injectInfo.color = (hwmsdkagent::HwmWaterMarkColorType)color;
    int ret = hwmsdkagent::SetWaterMark(&injectInfo);
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("SetWaterMark inject info failed"));
    }
}
```

7.5.3.6 自定义设置界面

ConfigSettingWindow

接口描述

该接口用于自定义设置界面中各标签页和功能区的显示隐藏，若未进行设置，则默认都显示

图 7-44 设置界面



注意事项

1. 该接口必须在设置界面打开前调用，设置界面打开时调用不会即时生效。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode ConfigSettingWindow(const HwmSettingWindowConfig * configInfo);
```

回调函数

```
virtual void OnConfigSettingWindowResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

表 7-58 参数说明

参数	是否必须	类型	描述
configInfo	是	HwmSettingWindowConfig	设置窗口定制信息。

表 7-59 结构体 HwmSettingWindowConfig 参数说明

参数	类型	描述
normalTab	HwmNormalSettingTab	基本设置页定制信息。
videoTab	HwmVideoSettingTab	视频设置页定制信息。
audioTab	HwmAudioSettingTab	音频设置页定制信息。
screenTab	HwmScreenSettingTab	屏幕设置页定制信息。
recordTab	HwmRecordSettingTab	录制设置页定制信息。
virtualBackgroundTab	HwmVirtualBackgroundSettingTab	虚拟背景设置页定制信息。
hotkeyTab	HwmHotKeySettingTab	快捷键设置页定制信息

表 7-60 结构体 HwmNormalSettingTab 参数说明

参数	类型	描述
isHide	bool	是否隐藏。
callOrConfArea	HwmSettingArea	会议呼叫设置区域定制信息。
chatNotifyArea	HwmSettingArea	聊天消息提醒设置区域定制信息。
subtitleArea	HwmSettingArea	字幕设置区域定制信息。
shareArea	HwmSettingArea	共享设置区域定制信息

表 7-61 结构体 HwmVideoSettingTab 参数说明

参数	类型	描述
isHide	bool	是否隐藏。
cameraArea	HwmSettingArea	摄像头设置区域定制信息。
videoViewArea	HwmSettingArea	视频画面设置区域定制信息。

表 7-62 结构体 HwmAudioSettingTab 参数说明

参数	类型	描述
isHide	bool	是否隐藏。
micArea	HwmSettingArea	麦克风设置区域定制信息。
spkArea	HwmSettingArea	扬声器设置区域定制信息。

参数	类型	描述
advancedArea	HwmSettingArea	高级设置区域定制信息。

表 7-63 结构体 HwmScreenSettingTab 参数说明

参数	类型	描述
isHide	bool	是否隐藏。
dualScreenArea	HwmSettingArea	双屏设置区域定制信息。
displayTopologyArea	HwmSettingArea	显示器显示模式设置区域定制信息。
swapContentArea	HwmSettingArea	切换显示内容设置区域定制信息。

表 7-64 结构体 HwmRecordSettingTab 参数说明

参数	类型	描述
isHide	bool	是否隐藏。
localRecordArea	HwmSettingArea	本地录制设置区域定制信息。

表 7-65 结构体 HwmVirtualBackgroundSettingTab 参数说明

参数	类型	描述
isHide	bool	是否隐藏。
virtualBackgroundArea	HwmSettingArea	虚拟背景设置区域定制信息。

表 7-66 结构体 HwmHotKeySettingTab 参数说明

参数	类型	描述
isHide	bool	是否隐藏。
hotkeyArea	HwmSettingArea	快捷键设置区域定制信息。

表 7-67 结构体 HwmSettingArea 参数说明

参数	类型	描述
isHide	bool	是否隐藏。

返回值

表 7-68 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 定制设置界面
 */
void demoConfigSettingWindowDlg::OnBnClickedOk()
{
    hwmsdkagent::HwmSettingWindowConfig data{ 0 };

    data.normalTab.isHide = true;
    data.videoTab.videoViewArea.isHide = true;

    int ret = hwmsdkagent::ConfigSettingWindow(&data);
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("config setting window error"));
        return;
    }
}
```

7.5.3.7 显示设置窗口

ShowSettingWindow

接口描述

该接口显示设置窗口，会前会中都可以调用。

注意事项

1. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode ShowSettingWindow(const HwmShowSettingWindowInfo* showInfo);
```

回调函数

```
virtual void OnShowSettingWindowResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

表 7-69 参数说明

参数	是否必须	类型	描述
showInfo	是	HwmShowSettingWindowInfo	显示设置窗口信息。

表 7-70 结构体 HwmShowSettingWindowInfo 参数说明

参数	类型	描述
isShow	bool	是否显示。
defaultPage	HwmSettingPage	默认显示标签页。

表 7-71 枚举 HwmSettingPage 说明

枚举值	描述
SETTING_PAGE_NORMAL	基本设置页。
SETTING_PAGE_VIDEO	视频设置页。
SETTING_PAGE_AUDIO	音频设置页。
SETTING_PAGE_RECORD	录制设置页。
SETTING_PAGE_HOTKEY	快捷键设置页。
SETTING_PAGE_SCREEN	屏幕设置页。
SETTING_PAGE_COUNT	设置页数量。

示例代码

```
/**
 * 显示设置窗口
 */
void demoShowSettingWindowDlg::OnBnClickedOk()
{
    hwmsdkagent::HwmShowSettingWindowInfo showInfo{ 0 };

    showInfo.isShow = true;
    showInfo.defaultPage = hwmsdkagent::HwmSettingPage::SETTING_PAGE_NORMAL;

    int ret = hwmsdkagent::ShowSettingWindow(&showInfo);
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("show setting window error"));
        return;
    }
}
```

7.5.4 会议控制

7.5.4.1 创建会议

CreateConf

接口描述

该接口用于创建即时会议。

注意事项

1. 该接口默认已将自己作为主席加入会议，调用接口时，在与会者参数中不需要带上自己的信息。
2. 若需要在创建会议的同时邀请其他与会者，需要传入其他与会者信息。若不需要，与会者信息和与会者个数可以为空。
3. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode CreateConf(HwmCreateConfInfo *createConfInfo,
HwmConfAttendee *attendees, unsigned int count);
```

回调函数

```
virtual void OnCreateConfResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

参数	是否必须	类型	描述
createConfInfo	是	HwmCreateConfInfo	会议信息。
attendees	否	HwmConfAttendee	与会者信息。若不需要带与会者，参数值可以为NULL。
count	否	unsigned int	与会者个数。若不需要带与会者，参数值可以为0。

表 7-72 结构体 HwmCreateConfInfo 参数说明

参数	是否必须	类型	描述
subject	是	char[]	会议主题。
meetingType	是	HwmConfMediaType	会议类型。
needPassword	否	bool	是否需要会议密码。 说明 仅对随机会议ID生效

参数	是否必须	类型	描述
isAutoRecord	否	bool	是否开启自动录制会议。开启自动录制会议开关，内部会默认支持录制会议，录制会议开关的打开与关闭并不影响。 说明 仅对云录制生效，客户端本地录制不生效。
allowRecord	否	bool	是否支持录制会议。 说明 仅对云录制生效，客户端本地录制不生效。
callInRestriction	否	HwmJoinConfPermissionType	允许入会范围限制。默认所有用户。
vmrId	否	char[]	云会议室唯一ID。预约随机会议时，设置为空字符串。
vmrConfIdType	否	HwmVmrConfIdType	云会议室的会议ID类型
guestPwd	否	char[]	普通与会者密码，如果空则服务器随机生成 说明 仅对随机会议ID生效
isOpenWaitingRoom	否	bool	等候室开启开关 说明 需要等候室功能开通后才能生效
duration	否	int	会议时长，单位分钟，最小值15分钟，最大值1440分钟。 注：不填时由服务端取默认会议时长

表 7-73 枚举 HwmConfMediaType 说明

枚举值	描述
HWM_AUDIO_AND_DATA	音频数据会议。
HWM_VIDEO_AND_DATA	视频数据会议。

表 7-74 枚举 HwmJoinConfPermissionType 说明

枚举值	描述
RESTRICTION_CALL_IN_ALL	所有用户。
RESTRICTION_CALL_IN_COMPANY	企业内用户。
RESTRICTION_CALL_IN_INVITED	被邀请用户。

表 7-75 结构体 HwmConfAttendee 参数说明

参数	是否必须	类型	描述
name	是	char[]	与会者名称。
number	是	char[]	号码。如果填分配给帐号的 sip 号码（如 +99111244216210249）则呼叫该帐号的 App；如果填 PSTN 号码（如 18700000000），则通过 VoIP 网关呼叫该号码，前提是该企业已开通 PSTN 呼叫权限。与 thirdUserId 二选一，帐号密码鉴权时使用。
thirdUserId	是	char[]	第三方用户 ID。与 number 二选一，App ID 鉴权时使用。

表 7-76 枚举 HwmVmrConfIdType 说明

枚举值	描述
HWM_VMR_CONF_ID_TYPE_FIXED	云会议室的会议 ID 类型是固定类型。
HWM_VMR_CONF_ID_TYPE_RANDOM	云会议室的会议 ID 类型是随机类型。

返回值

表 7-77 返回值

类型	描述
HwmErrCode	成功返回 0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```

/**
 * 创建会议
 */
int demoCreateConfWithAttendeeDlg::clickCreatConfWithAttendee()
{
    int ret;
    hwmsdkagent::HwmCreateConfInfo data;
    memset(&data, 0, sizeof(hwmsdkagent::HwmCreateConfInfo));

    //设置会议主题
    string meetingSubjectStr = CTools::UNICODE2UTF(CString("我的会议"));
    strncpy_s(data.subject, meetingSubjectStr.c_str(), HWM_MAX_SUBJECT_LEN);
    //设置会议类型
    data.mediaType = hwmsdkagent::HWM_VIDEO_AND_DATA
    //设置会议是否需要来宾密码
    data.needPassword = true;
    //设置会议呼入限制
    data.callInRestriction = hwmsdkagent::RESTRICTION_CALL_IN_ALL;
    //设置vmrId, 若创建随机会议, 则vmrId为空 此处m_confIdTypeCombo排序是把随机会议放在首位
    //因此vmrIdSelect > 0 为个人会议id与云会议室id场景
    const auto vmrIdSelect = m_confIdTypeCombo.GetCurSel();
    if (vmrIdSelect > 0 && vmrIdSelect - 1 < vmrList.size())
    {
        strcpy_s(data.vmrId, HWM_MAX_VMR_CONF_ID_LEN, vmrList[vmrIdSelect - 1].vmrId);
    }

    //设置与会人
    //获取与会人信息, 实际使用是根据实际情况获取
    CString tempCString;
    m_attendeesEdit.GetWindowText(tempCString);
    string tempString = CTools::UNICODE2UTF(tempCString);
    vector<string> list = CTools::split(tempString, ',');
    vector<string> temp;
    int count = list.size();
    int realCount = 0;
    if (count > 0)
    {
        //申请结构体内存
        hwmsdkagent::HwmConfAttendee* participants;
        participants = (hwmsdkagent::HwmConfAttendee*)malloc(sizeof(hwmsdkagent::HwmConfAttendee) *
count);
        if (participants == NULL)
        {
            return -1;
        }
        memset(participants, 0, sizeof(hwmsdkagent::HwmConfAttendee)*count);

        hwmsdkagent::HwmConfAttendee* participantsTemp = participants;
        for (int i = 0; i < count; i++)
        {
            temp = CTools::split(list[i], '-');
            if (temp.size() == 2)
            {
                //name赋值
                strncpy_s(participantsTemp->name, (char *)temp[0].c_str(), HWM_MAX_DISPLAY_NAME_LEN);
                //number赋值
                strncpy_s(participantsTemp->number, (char *)temp[1].c_str(), HWM_MAX_NUMBER_LEN);
                //指针个数加1
                realCount++;
                participantsTemp++;
            }
        }
        //调用SDK接口, 创建会议
        ret = hwmsdkagent::CreateConf(&data, participants, realCount);
    }
}

```

```

//释放内存空间
free(participants);
participants = NULL;
}
else
{
//不携带与会人创建会议
ret = hwmsdkagent::CreateConf(&data, NULL, 0);
}
return ret;
}

```

7.5.4.2 加入会议

JoinConfById

接口描述

该接口用于通过会议ID和密码加入已经存在的会议。

注意事项

1. 在登录和未登录状态下都可以调用该接口。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode JoinConfById(HwmJoinConfByIdInfo *joinConfByIdInfo);
```

回调函数

```
virtual void OnJoinConfByIdResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

表 7-78 结构体 HwmJoinConfByIdInfo 参数说明

参数	是否必须	类型	描述
confId	是	char[]	会议ID。
password	否	char[]	会议密码。若希望以主席身份入会，必须传入主席密码。若希望以来宾身份入会，会议要求来宾密码时，必须传入来宾密码，会议不要求来宾密码时，可以为空字符串。
name	否	char[]	会议昵称。在已登录状态下可为空，已登录时空将使用登录的用户名作为会场名。匿名入会必须传该名字或者提前设置会场名
isCloseSpk	否	bool	是否关闭扬声器，true关闭扬声器；false不关闭扬声器。

参数	是否必须	类型	描述
isStopConflictConf	否	bool	强制结束和当前VMR会议资源冲突的会议（必需是vmr资源的拥有者才有权限，否则无效）

返回值

表 7-79 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
// 通过会议ID加入会议
int demoJoinConfByDlg::clickJoinConfById()
{
    // 填写会议ID和会议密码；未登录场景下加入会议时，也可以配置本会场的会场名称
    string meetingID = CTools::UNICODE2UTF(CString("989156631"));
    string accessCode = CTools::UNICODE2UTF(CString("807766"));
    string participantName = CTools::UNICODE2UTF(CString("杭州会场"));

    hwm sdkagent::HwmJoinConfByIdInfo data;
    memset(&data, 0, sizeof(hwm sdkagent::HwmJoinConfByIdInfo));

    strncpy_s(data.confId, meetingID.c_str(), HWM_MAX_CONF_ID_LEN);
    strncpy_s(data.password, accessCode.c_str(), HWM_MAX_PASSWORD_LEN);
    strncpy_s(data.name, participantName.c_str(), HWM_MAX_DISPLAY_NAME_LEN);
    data.isCloseSpk = false;

    // 设置麦克风和摄像头开关
    int ret = EnableFeature(hwm sdkagent::HWM_ENABLE_FEATURE_TYPE_MIC_SWITCH, true);
    ret = EnableFeature(hwm sdkagent::HWM_ENABLE_FEATURE_TYPE_CAM_SWITCH, false);
    ret = hwm sdkagent::JoinConfById(&data);
    return ret;
}
```

说明

在加入会议前可通过调用EnableFeature修改麦克风、摄像头状态。

7.5.4.3 Random 入会

JoinConfByRandom

接口描述

通过邮件会议通知中的会议链接入会时（会议链接如：<https://bmeeting.huaweicloud.com/#/j/981924330/e11ddec62ee760734fcf2ba55b79937eac2aa68677caf659>），链接里面的的是会议ID（即举例中的981924330）和会议的Random码（即举例中的

e11ddec62ee760734fcf2ba55b79937eac2aa68677caf659)。第三方App需要调用本接口加入会议。

说明

如果预约会议时选择没有来宾密码，会议链接只有会议ID，没有Random码。

注意事项

1. 在登录和未登录状态下都可以调用该接口。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode JoinConfByRandom(HwmJoinConfByRandomInfo *joinConfByRandomInfo);
```

回调函数

```
virtual void OnJoinConfByRandomResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

表 7-80 结构体 HwmJoinConfByRandomInfo 参数说明

参数	是否必须	类型	描述
confld	是	char[]	会议ID。
random	否	char[]	48位会议密码映射值。来源于分享会议信息中的会议链接尾部。会议不要求来宾密码时，可以为空字符串。
name	否	char[]	会议昵称。在已登录状态下可为空，已登录时空将使用登录的用户名作为会场名。匿名入会必须传该名字或者提前设置会场名
isCloseSpk	否	bool	是否关闭扬声器，true关闭扬声器；false不关闭扬声器。

返回值

表 7-81 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
// 通过会议ID和Random加入会议
int demoJoinConfByRandomDlg::clickJoinConfByRandom()
{
    // 填写会议ID和密码映射值等; Random入会时, 支持配置会场名称
    string meetingID = CTools::UNICODE2UTF(CString("988148352"));
    string accessCode =
CTools::UNICODE2UTF(CString("c4f92bc34e685bbba6c23589741be0431997f30da5d6e47b"));
    string participantName = CTools::UNICODE2UTF(CString("杭州会场"));

    hwmsdkagent::HwmJoinConfByRandomInfo data;
    memset(&data, 0, sizeof(hwmsdkagent::HwmJoinConfByRandomInfo));

    strncpy_s(data.confId, meetingID.c_str(), HWM_MAX_CONF_ID_LEN);
    strncpy_s(data.random, accessCode.c_str(), HWM_MAX_RANDOM_LEN);
    strncpy_s(data.name, participantName.c_str(), HWM_MAX_DISPLAY_NAME_LEN);
    data.isCloseSpk = false;

    // 设置麦克风和摄像头开关
    int ret = EnableFeature(hwmsdkagent::HWM_ENABLE_FEATURE_TYPE_MIC_SWITCH, true);
    ret = EnableFeature(hwmsdkagent::HWM_ENABLE_FEATURE_TYPE_CAM_SWITCH, false);
    ret = hwmsdkagent::JoinConfByRandom(&data);
    return ret;
}
```

📖 说明

在加入会议前可通过调用EnableFeature修改麦克风、摄像头状态。

7.5.4.4 设置会场名

RenameSelf

接口描述

该接口用于设置会场名。

注意事项

该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode RenameSelf(const char
name[HWM_MAX_DISPLAY_NAME_LEN]);
```

回调函数

```
virtual void OnRenameSelfResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

表 7-82 参数说明

参数	是否必须	类型	描述
name	是	char[]	会场名。

返回值

表 7-83 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
void demoRenameSelfDlg::OnBnClickedOk()
{
    char name[HWM_MAX_DISPLAY_NAME_LEN] = {0};
    string tempString = CTools::UNICODE2UTF(CString("杭州会场"));
    //设置会场名
    strncpy_s(name, (char *)tempString.c_str(), HWM_MAX_DISPLAY_NAME_LEN);
    int ret = hwmsdkagent::RenameSelf(name);
    if(ret != hwmsdk::HWM_COMMON_SUCCESS)
    {
        AfxMessageBox(_T("Rename self error"));
        return;
    }
}
```

7.5.4.5 添加与会者

AddAttendee

接口描述

该接口用于在会议或呼叫中添加与会者。

注意事项

1. 不在会议或呼叫中，调用该接口将返回失败。
2. 呼叫中调用该接口，呼叫则转成会议。
3. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode AddAttendee(HwmConfAttendee *attendees, unsigned int count);
```

回调函数

```
virtual void OnAddAttendeeResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

表 7-84 结构体 HwmConfAttendee 参数说明

参数	是否必须	类型	描述
name	是	char[]	会场名。

参数	是否必须	类型	描述
number	是	char[]	号码。如果填分配给帐号的sip号码（如+99111244216210249）则呼叫该帐号的App；如果填PSTN号码（如18700000000），则通过VoIP网关呼叫该号码，前提是该企业已开通PSTN呼叫权限。与thirdUserId二选一，帐号密码鉴权时使用。
thirdUserId	是	char[]	第三方用户ID。与number二选一，App ID鉴权时使用。
accountId	否	char[]	华为云会议帐号。

返回值

表 7-85 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 添加与会人
 */
int demoAddAttendeeDlg::clickAddAttendee()
{
    int ret;
    CString tempCString;

    //获取与会人信息，实际使用是根据实际情况获取
    m_numbersEdit.GetWindowText(tempCString);
    string tempString = CTools::UNICODE2UTF(tempCString);
    vector<string> list = CTools::split(tempString, ',');
    vector<string> temp;
    int count = list.size();

    int realCount = 0;
    if (count > 0)
    {
        //申请结构体内存
        hwmsdkagent::HwmConfAttendee* participants;
        participants = (hwmsdkagent::HwmConfAttendee*)malloc(sizeof(hwmsdkagent::HwmConfAttendee) *
count);
        if (participants == NULL)
        {
            return -1;
        }
        memset(participants, 0, sizeof(hwmsdkagent::HwmConfAttendee)*count);

        hwmsdkagent::HwmConfAttendee* participantsTemp = participants;
```



```

for (int i = 0; i < count; i++)
{
    temp = CTools::split(list[i], '-');
    if (temp.size() == 2)
    {
        //name赋值
        strncpy_s(participantsTemp->name, (char *)temp[0].c_str(), HWM_MAX_DISPLAY_NAME_LEN);
        //number赋值
        strncpy_s(participantsTemp->number, (char *)temp[1].c_str(), HWM_MAX_NUMBER_LEN);
        //指针个数加1
        realCount++;
        participantsTemp++;
    }
}

ret = hwmsdkagent::AddAttendee(participants, realCount);

//释放内存空间
free(participants);
participants = NULL;
}

return ret;
}

```

7.5.4.6 开启共享

StartShare

接口描述

该接口用于开启会议共享。SDK的UI已经实现了“共享”功能，如果第三方App不需要通过程序控制开启共享，则不用调用该接口。

注意事项

1. 不在会议或呼叫中调用该接口将返回失败。
2. 呼叫中调用该接口，呼叫转成会议
3. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode StartShare(HwmStartShareInfo *startShareInfo);
```

回调函数

```
virtual void OnStartShareResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

表 7-86 结构体 HwmStartShareInfo 参数说明

参数	是否必须	类型	描述
shareType	是	enum	共享类型。
monitorShareInfo	否	HwmMonitorShareInfo	屏幕共享信息。当共享类型不是屏幕共享时，该变量不需要赋值。

参数	是否必须	类型	描述
appShareInfo	否	HwmAppShareInfo	程序共享信息。当共享类型不是程序共享时，该变量不需要赋值。
operatableUiMonitor	是	HwmMonitor	共享时，工具栏所在的显示器。若值为0，则不指定显示器。

表 7-87 枚举 HwmShareType 说明

枚举值	描述
HWM_SHARE_TYPE_MONITOR	屏幕共享。
HWM_SHARE_TYPE_APP	程序共享。

表 7-88 结构体 HwmMonitorShareInfo 参数说明

参数	是否必须	类型	描述
monitor	是	HwmMonitor	显示器类型。

表 7-89 枚举 HwmMonitor 说明

枚举值	描述
HWM_MONITOR_MAIN	主显示器。
HWM_MONITOR_SECONDARY	副显示器。

表 7-90 结构体 HwmAppShareInfo 参数说明

参数	是否必须	类型	描述
appHandle	是	HWND	程序句柄。

返回值

表 7-91 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 开启共享
 */
int demoStartShareDlg::clickStartShare()
{
    hwmsdkagent::HwmStartShareInfo data;
    memset(&data, 0, sizeof(hwmsdkagent::HwmStartShareInfo));
    //设置共享类型
    data.shareType = hwmsdkagent::HwmShareType::HWM_SHARE_TYPE_MONITOR;
    //设置共享的显示器
    data.monitorShareInfo.monitor = hwmsdkagent::HWM_MONITOR_MAIN;
    //设置工具栏所在的显示器
    data.operatableUiMonitor = hwmsdkagent::HWM_MONITOR_MAIN;

    int ret = hwmsdkagent::StartShare(&data);
    return ret;
}
```

7.5.4.7 结束共享

StopShare

接口描述

该接口用于结束会中共享。SDK的UI已经实现了“停止共享”功能，如果第三方App不需要通过程序控制停止共享，则不用调用该接口。

注意事项

1. 不在共享中不应调用此接口。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode StopShare();
```

回调函数

```
virtual void OnStopShareResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

无。

返回值

表 7-92 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 结束共享
 */
void demoConfCtrlDlg::OnBnClickedButtonStopShare()
{
    //结束共享
    int ret = hwmsdkagent::StopShare();
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("StopShare error"));
    }
}
```

7.5.4.8 离开会议

LeaveConf

接口描述

该接口用于离开会议。

注意事项

1. 不在会议中不应调用此接口。
2. 调用该接口离开会议，会议不会结束，在会议存在期间可以再次入会。
3. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode LeaveConf();
```

回调函数

```
virtual void OnLeaveConfResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

无。

返回值

表 7-93 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 离开会议
 */
void demoConfCtrlDlg::OnBnClickedButtonLeaveConf()
{
    int ret = hwmsdkagent::LeaveConf();
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("LeaveConf error"));
    }
}
```

7.5.4.9 结束会议

EndConf

接口描述

该接口用于结束会议。

注意事项

1. 不在会议中不应调用此接口。
2. 调用该接口，会议将结束，无法再次加入该会议。
3. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode EndConf();
```

回调函数

```
virtual void OnEndConfResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

无。

返回值

表 7-94 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 结束会议
 */
void demoConfCtrlDlg::OnBnClickedButtonStopConf()
{
    int ret = hwmsdkagent::EndConf();
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {

```

```
AfxMessageBox(_T("EndConf error"));
}
```

7.5.4.10 设置会议来电应答

SetConfIncomingAnswerResult

接口描述

该接口用于收到会议来电时设置应答。一般用于设置会议来电的自动拒绝或者自动接听。

注意事项

1. 只有在收到会议来电后才能调用该接口（在通知接口 [OnConfIncoming](#) 中调用），其他情况不应调用该接口。
2. 调用该接口设置应答类型：弹窗选择、拒绝、接听。
3. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

说明

如果是自动拒接来电，需要在Init初始化成功后，需要调用 [Config](#) 接口将 `isHideConfIncomingWindow` 设置成 `TRUE`。

需要拒接时 `SetCallIncomingAnswerResult` 的参数传 `HWM_INCOMING_ANSWER_DECLINE`，需要正常弹窗接听时传 `HWM_INCOMING_ANSWER_NORMAL`。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode type(HWMIncomingAnswerType type);
```

回调函数

```
virtual void OnSetConfIncomingAnswerResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

表 7-95 枚举 HWMIncomingAnswerType 说明

枚举值	描述
HWM_INCOMING_ANSWER_NORMAL	来电时应答采用原始体验弹窗选择。
HWM_INCOMING_ANSWER_DECLINE	来电时拒接
HWM_INCOMING_ANSWER_ANSWER	来电时接听

返回值

表 7-96 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 收到会议来电后拒接来电
 */
void demoConfCtrlDlg::OnBnClickedSetConfIncomingAnswerResult()
{
    hwmsdkagent::HWMInComingAnswerType type = HWM_INCOMING_ANSWER_DECLINE;
    int ret = hwmsdkagent::SetConfIncomingAnswerResult(type);
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("SetConfIncomingAnswerResult error"));
    }
}
```

7.5.4.11 显示会议主窗口

showMeetingWindow

接口描述

当会议主窗口处于最小化或者被其他窗口覆盖时，使用该接口将会议主窗口拉到最上层。一般用于从会议列表中拉起正在进行的会议窗口。

注意事项

无

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode ShowMeetingWindow();
```

参数描述

无

返回值

表 7-97 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 显示会议主窗口
```

```

*/
void demoSettingDlg::OnBnClickedButtonShowMeetingWindow()
{
    int ret = hwmsdkagent::ShowMeetingWindow();
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox("ShowMeetingWindow error" );
    }
}

```

7.5.4.12 设置共享策略

SetShareStrategy

接口描述

该接口用于设置屏幕共享的策略是清晰度优先还是流畅度优先。

注意事项

该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```

HWM_SDK_AGENT_API hwmsdk::HwmErrCode SetShareStrategy(HwmShareStrategyType
shareStrategyType);

```

回调函数

```

virtual void OnSetShareStrategyResult(hwmsdk::HwmErrCode ret, const char* reason) {};

```

参数描述

表 7-98 参数说明

参数	是否必须	类型	描述
shareStrategyType	是	HwmShareStrategyType	屏幕共享策略类型枚举。

表 7-99 枚举 HwmShareStrategyType 说明

枚举值	描述
HWM_SHARE_STRATEGY_DEFINITON_FIRST	清晰度优先
HWM_SHARE_STRATEGY_FLUENCY_FIRST	流畅度优先

返回值

表 7-100 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 设置共享策略
 */
void demoSetShareStrategyDlg::OnBnClickedButtonSetShareStrategy()
{
    hwmsdkagent::HwmShareStrategyType type = HWM_SHARE_STRATEGY_DEFINITION_FIRST;
    int ret = hwmsdkagent::SetShareStrategy(type);
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("Set Share Strategy Result failed"));
    }
}
```

7.5.4.13 设置共享音频

SetShareAudio

接口描述

该接口用于设置共享时是否包含电脑声音开关。

注意事项

该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode SetShareAudio(bool isEnabled);
```

回调函数

```
virtual void OnSetShareAudioResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

表 7-101 参数说明

参数	是否必须	类型	描述
isEnabled	是	bool	是否开启功能开关。

返回值

表 7-102 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 设置共享音频
 */
void demoSetShareAudioDlg::OnBnClickedButtonSetShareAudio()
{
    bool isEnabled = true;
    int ret = hwmsdkagent::SetShareAudio(isEnabled);
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("Set Share Audio Result failed"));
    }
}
```

7.5.4.14 音视频设备控制

AVDeviceCtrl

接口描述

该接口用于控制本地麦克风、扬声器和摄像头开关。

注意事项

该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode AVDeviceCtrl(HwmDeviceType deviceType, bool isOpen);
```

回调函数

```
virtual void OnAVDeviceCtrlResult(hwmsdk::HwmErrCode ret, const char* reason, HwmDeviceType deviceType) {};
```

参数描述

表 7-103 参数说明

参数	是否必须	类型	描述
HwmDeviceType	是	HwmDeviceType	设备类型枚举。
isOpen	是	bool	是否开启设备。

表 7-104 枚举 HwmDeviceType 说明

枚举值	描述
HWM_DEVICE_TYPE_MIC	麦克风设备
HWM_DEVICE_TYPE_SPEAKER	扬声器设备
HWM_DEVICE_TYPE_CAMERA	视频设备

返回值

表 7-105 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 音视频设备控制
 */
int demoAVDeviceCtrlDlg::ClickAVDeviceCtrl()
{
    hwmsdkagent::HwmDeviceType deviceType = hwmsdkagent::HwmDeviceType::HWM_DEVICE_TYPE_MIC;
    bool isOpen = true;
    int ret = hwmsdkagent::AVDeviceCtrl(deviceType, isOpen);
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("AVDeviceCtrl error"));
    }
}
```

7.5.4.15 设置共享权限

SetShareScreenPermission

接口描述

该接口用于设置共享权限。

注意事项

该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode SetShareScreenPermission(bool isAllowed);
```

回调函数

```
virtual void OnSetShareScreenPermissionResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

表 7-106 参数说明

参数	是否必须	类型	描述
isAllowed	是	bool	是否允许。

返回值

表 7-107 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 设置共享权限
 */
int demoSetShareScreenPermissionDlg::ClickSetShareScreenPermission()
{
    bool isOpen = true;
    int ret = hwmsdkagent::SetShareScreenPermission(isOpen);
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("SetShareScreenPermission error"));
    }
}
```

7.5.5 会议管理

7.5.5.1 获取 VMR 列表

GetVmrList

接口描述

该接口用于获取VMR列表。

注意事项

1. 该接口在已登录状态下才能调用。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode GetVmrList();
```

回调函数

```
virtual void OnGetVmrList(hwmsdk::HwmErrCode ret, const char* reason, const HwmVmrInfo* vmrInfoList, unsigned int listLen) {};
```

参数描述

无。

返回值

表 7-108 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

表 7-109 结构体 HwmVmrlInfo 参数说明

参数	类型	描述
vmrConferenceId	char[]	云会议室上的固定会议ID。
vmrId	char[]	云会议室唯一ID。
chairmanPwd	char[]	云会议室主席密码。
guestPwd	char[]	云会议室来宾密码。
type	HwmVmrlIdType	云会议室类型。
maxParties	int	云会议室最大方数。
name	char[]	会议室名称。
allowGuestStartConf	bool	是否允许来宾提前启动会议。
joinConfRestrictionType	HwmJoinConfPermissionType	允许入会范围限制。
isOpenWaitingRoom	bool	等候室开启开关。

表 7-110 枚举 HwmVmrlIdType 说明

枚举值	描述
PERSONAL_MEETING_ID	个人会议ID。
CLOUD_MEETING_ROOM_ID	云会议室ID。

示例代码

```
/**
 * 获取Vmr列表
```

```

*/
void demoBookConfDlg::GetVmrList()
{
    int ret = hwmsdkagent::GetVmrList();
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("GetVmrList error"));
    }
}

```

7.5.5.2 预约会议

BookConf

接口描述

该接口用于预约会议。

注意事项

1. 该接口在已登录状态下才能调用。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。
3. 调用该接口，与会者列表中需要添加本会场，并且将本会场设置为主席身份。
4. 回调函数中会返回预约成功后的会议详情，详见[HwmConfDetail](#)结构体表。如果参数内只有会议id有价值则获取会议详情失败。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode BookConf(const HwmBookConfParam *param);
```

回调函数

```
virtual void OnBookConfResult(hwmsdk::HwmErrCode ret, const char* reason, const HwmConfDetail* confDetail) {};
```

参数描述

表 7-111 结构体 HwmBookConfParam 参数说明

参数	是否必须	类型	描述
subject	是	char[]	会议主题。
startTime	是	long long	会议开始时间时间戳(utc时间)，单位是秒。
duration	是	int	会议时长，单位分钟，最小值15分钟，最大值1425分钟。
confType	否	HwmConfMediaType	会议类型。默认视频会议。
isNeedConfPwd	否	bool	是否需要密码。默认不需要密码。 说明 仅对随机会议ID生效

参数	是否必须	类型	描述
isRecordOn	否	bool	录制会议开关。 说明 仅对云录制生效，客户端本地录制不生效。
isAutoRecordOn	否	bool	会议自动录制开关。默认不开启。 开启会议自动录制开关，内部会默认支持录制会议，录制会议开关的打开与关闭并不影响。 说明 仅对云录制生效，客户端本地录制不生效。
timeZone	是	int	时区码，请参考 时区表 。用于发送会议通知。
vmrFlag	否	bool	会议是否使用虚拟会议室。预约并发会议时，设置为false。
vmrId	否	char[]	云会议室唯一ID。预约并发会议时，设置为空字符串。
joinConfRestrictionType	否	HwmJoinConfPermissionType	允许入会范围限制。默认所有用户。
isSmsOn	否	bool	发送短信通知开关。企业配置里面需要开启，否则不生效。
isMailOn	否	bool	发送邮件通知开关。
isEmailCalendarOn	否	bool	发送邮件日历开关。
attendees	是	HwmAttendeeInfo*	与会者列表。
attendeeLen	是	unsigned int	与会者列表长度。
vmrConfIdType	否	HwmVmrConfIdType	云会议室的会议ID类型。
guestPwd	否	char[]	普通与会者密码，如果不填则来宾密码由服务端生成。 说明 仅对随机ID会议生效。
isOpenWaitingRoom	否	bool	等候室开启开关。 说明 需要等候室功能开通后才能生效。

参数	是否必须	类型	描述
allowGuestStartConf	否	bool	是否允许来宾提前启动会议。 说明 <ul style="list-style-type: none"> 仅针对随机会议ID的会议生效。 固定会议ID的会议，请使用修改云会议室信息接口设置。
allowGuestStartConfTime	否	unsigned int	允许来宾提前入会时间范围（单位：分钟）： 0 - 随时 x - 提前x分钟启动会议 说明 <ul style="list-style-type: none"> 仅针对随机会议ID的会议生效。 固定会议ID的会议，不支持设置提前入会时间。

表 7-112 枚举 HwmConfMediaType 说明

枚举值	描述
HWM_AUDIO_AND_DATA	音频数据会议。
HWM_VIDEO_AND_DATA	视频数据会议。

表 7-113 枚举 HwmJoinConfPermissionType 说明

枚举值	描述
RESTRICTION_CALL_IN_ALL	所有用户。
RESTRICTION_CALL_IN_COMPANY	企业内用户。
RESTRICTION_CALL_IN_INVITED	被邀请用户。

表 7-114 结构体 HwmAttendeelInfo 参数说明

参数	是否必须	类型	描述
number	是	char[]	号码。如果填分配给帐号的sip号码（如+99111244216210249）则呼叫该帐号的App；如果填PSTN号码（如18700000000），则通过VoIP网关呼叫该号码，前提是该企业已开通PSTN呼叫权限。与thirdUserId二选一，帐号密码鉴权时使用。
thirdUserId	是	char[]	第三方用户ID。与number二选一，App ID鉴权时使用。
name	是	char[]	与会者名称。
email	否	char[]	电子邮箱地址，用于发送邮件通知。
sms	否	char[]	短信通知手机号码，用于发送短信通知。
isMute	否	bool	是否闭音。
isAutoInvite	否	bool	是否自动外邀。
role	是	HwmConf Role	会议角色。
isAnonymous	否	BOOL	是否是匿名入会（仅用于获取与会者列表使用）
callStatus	否	ConfAttendeelState	与会者状态（仅用于获取与会者列表使用）
orgId	否	char[]	用于识别是否是企业账号（仅用于获取与会者列表使用）

表 7-115 会议成员角色 HwmConfRole 枚举值

枚举值	描述
HWM_CONF_ROLE_ATTENDEE	普通与会者。
HWM_CONF_ROLE_CHAIRMAN	主席。

表 7-116 枚举 HwmVmrConfIdType 说明

枚举值	描述
HWM_VMR_CONF_ID_TYPE_FIXED	云会议室的会议ID类型是固定类型。
HWM_VMR_CONF_ID_TYPE_RANDOM	云会议室的会议ID类型是随机类型。

表 7-117 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

表 7-118 结构体 HwmConfDetail 字段名

字段	类型	描述
confListInfo	HwmConfListInfo	会议信息，参见 表3 。
vmrFlag	bool	是否是vmr会议。
vmrId	char[]	云会议室唯一ID。
isRecordOn	bool	是否有录制会议权限。
isAutoRecord	bool	是否开启自动录制会议
isGuestFreePwd	bool	是否需要来宾免密。
isMailOn	bool	是否发送邮件通知。
isSmsOn	bool	是否发送短信通知。
isEmailCalendarOn	bool	是否发送日历通知。
joinConfRestrictionType	HwmJoinConfPermissionType	会议准入设置 表3 。
attendees	HwmAttendeeInfo*	与会者列表，字段信息参见 预约会议 中表述。
attendeeLen	unsigned int	与会者列表长度。
confServerType	HwmConfServerType	会议服务器类型 1:RTC or 0:MCU
isOpenWaitingRoom	bool	是否开启等候室

示例代码

```
/**
 * 预约会议
 */
```

```
int CdemoBookConfDlg::OnBnClickedBookConf()
{
    hwmsdkagent::HwmBookConfParam data = {0};
    string confSubject = u8"我的预约会议"; // 此处应是utf-8编码字符串
    strncpy_s(data.subject, confSubject.c_str(), HWM_MAX_SUBJECT_LEN);
    data.confType = hwmsdkagent::HwmConfMediaType::HWM_VIDEO_AND_DATA;
    data.isNeedConfPwd = true;
    data.isAutoRecordOn = true;
    data.isRecordOn = true;
    data.startTime = 1598398920; //utc时间戳, 单位秒, 如果获取的时间是本地时间, 需要转换成utc时间
    data.duration = 30; //会议持续时长(分钟)
    data.timeZone = 56; //时区, 56表示东八区
    data.joinConfRestrictionType = hwmsdkagent::HwmJoinConfPermissionType::RESTRICTION_CALL_IN_ALL;

    std::vector<hwmsdkagent::HwmAttendeeInfo> attendeeList;
    //被邀与会者
    hwmsdkagent::HwmAttendeeInfo attendeeInfo = { 0 };
    string attName = "Mike";
    string attNumber = "+991116003543";
    string attEmail = "";
    string attSms = "";
    string attUserId = "";
    strncpy_s(attendeeInfo.name, attName.c_str(), HWM_MAX_USER_NAME_LEN);
    strncpy_s(attendeeInfo.number, attNumber.c_str(), HWM_MAX_NUMBER_LEN);
    strncpy_s(attendeeInfo.email, attEmail.c_str(), HWM_MAX_EMAIL_LEN);
    strncpy_s(attendeeInfo.sms, attSms.c_str(), HWM_MAX_PHONE_NUM_LEN);
    strncpy_s(attendeeInfo.thirdUserId, attUserId.c_str(), HWM_MAX_USER_ID_LEN);
    attendeeInfo.isMute = true;
    attendeeInfo.isAutoInvite = true;
    attendeeList.push_back(attendeeInfo);

    data.attendees = attendeeList.data();
    data.attendeeLen = attendeeList.size();
    data.isSmsOn = true;
    data.isMailOn = true;
    data.isEmailCalendarOn = true;
    return hwmsdkagent::BookConf(&data);
}
```

7.5.5.3 编辑会议

EditConf

接口描述

该接口用于编辑已预约的会议。

注意事项

1. 该接口在已登录状态下才能调用。
2. 调用该接口时, 仅能编辑尚未开始的会议, 若会议已召开, 接口将返回失败。
3. 该接口为异步接口, 返回值只代表接口是否调用成功, 实际业务处理结果在对应的回调函数中返回。
4. 所有字段必填, 编辑会议前需要查询会议详情, 不需要变更的数据请使用查询会议详情返回的数据赋值。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode EditConf(const HwmEditConfParam *confParam);
```

回调函数

```
virtual void OnEditConfResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

表 7-119 结构体 HwmEditConfParam 参数说明

参数	是否必须	类型	描述
confId	是	char[]	会议ID。
subject	是	char[]	会议主题。
startTime	是	long long	会议开始时间戳(utc时间), second。
duration	是	int	会议时长, minute。
confType	是	HwmConfMediaType	会议类型。默认视频会议。
isNeedConfPwd	是	bool	是否需要密码。默认不需要密码。
isRecordOn	是	bool	录制会议开关。(字段即将废弃, 不建议使用)
isLiveOn	否	bool	保留。编辑会议时透传会议详情此字段。
isAutoRecordOn	是	bool	会议自动录制开关。
timeZone	是	int	时区码, 请参考 时区表 。
vmrFlag	是	bool	是否是vmr会议。不可修改, 填入会议详情返回值
vmrId	否	char[]	云会议室唯一ID。不可修改, 填入会议详情返回值
joinConfRestrictionType	是	HwmJoinConfPermissionType	允许入会范围限制。默认所有用户。
isSmsOn	是	bool	发送短信通知开关, 企业配置里面需要开启, 否则不生效。
isMailOn	是	bool	发送邮件通知开关。
isEmailCalendarOn	是	bool	发送邮件日历开关。
attendees	是	HwmAttendeeInfo*	与会者列表。
attendeeLen	是	unsigned int	与会者列表长度。

参数	是否必须	类型	描述
vmrConfIdType	是	HwmVmrConfIdType	云会议室的会议ID类型。不可修改，会议详情中vmrConferenceId非空，填入 HWM_VMR_CONF_ID_TYPE_FIXED；会议详情中vmrConferenceId为空，填入 HWM_VMR_CONF_ID_TYPE_RANDOM
isOpenWaitingRoom	否	bool	等候室开启开关 说明 需要等候室功能开通后才能生效
allowGuestStartConf	否	bool	是否允许来宾提前启动会议。 说明 <ul style="list-style-type: none"> 仅针对随机会议ID的会议生效。 固定会议ID的会议，请使用修改云会议室信息接口设置。
allowGuestStartConfTime	否	unsigned int	允许来宾提前入会时间范围(单位：分钟)：0-随时，n-提前n分钟启动会议。

返回值

表 7-120 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 编辑会议接口
 */
int demoEditConfDlg::EditConf()
{
    hwmsdkagent::HwmConfDetail confDetail{0}; //从缓存中得到会议详情数据赋予confDetail
    hwmsdkagent::HwmEditConfParam editConfParam{ 0 };
    //根据实际需求拷贝confDetail的数据
    strncpy_s(editConfParam.confId, confDetail.confListInfo.confId, HWM_MAX_CONF_ID_LEN);
    editConfParam.vmrFlag = confDetail.vmrFlag;
    strncpy_s(editConfParam.vmrId, confInfo.vmrId, HWM_MAX_VMR_CONF_ID_LEN);
    // 此处省略部分赋值代码
    editConfParam.timeZone = 56; // 本地时区，用户依据自身情况自己调整，56东八区
    editConfParam.isRecordOn = false;
    editConfParam.isAutoRecordOn = false;
    editConfParam.startTime = 1598398920; //utc时间戳
    editConfParam.duration = 0 * 60 + 30; //会议时长
}
```

```

editConfParam.joinConfRestrictionType =
hwmsdkagent::HwmJoinConfPermissionType::RESTRICTION_CALL_IN_ALL; //取编辑会议上面的允许入会用户类型
// vmrConferenceId非空是固定ID,为空则是随机id
editConfParam.vmrConfIdType = (strlen(confInfo.confListInfo.vmrConferenceId) != 0) ?
hwmsdkagent::HwmVmrConfIdType::HWM_VMR_CONF_ID_TYPE_FIXED :
hwmsdkagent::HwmVmrConfIdType::HWM_VMR_CONF_ID_TYPE_RANDOM;
return hwmsdkagent::EditConf(&editConfParam);
}

```

7.5.5.4 取消会议

CancelConf

接口描述

该接口用于取消已预约的会议。

注意事项

1. 该接口在登录状态下才能调用。
2. 调用该接口时，仅能取消尚未开始的会议，若会议已召开，则接口返回失败。
3. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode CancelConf(const char confId[HWM_MAX_CONF_ID_LEN]);
```

回调函数

```
virtual void OnCancelConfResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

参数	是否必须	类型	描述
confId	是	char[]	会议ID。

返回值

表 7-121 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```

/**
 * 取消会议接口
 */
int demoConfListDlg::OnBnCancelConf(hwmsdk::HwmErrCode ret, const char* msg)
{
    string confId= CTools::UNICODE2UTF(CString("989156631"));
}

```

```
return hwmsdkagent::CancelConf(confId);
}
```

7.5.5.5 获取会议详情

GetConfDetail

接口描述

该接口用于获取会议的详细信息。

注意事项

1. 该接口在已登录状态下才能调用。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode GetConfDetail(const char
confId[HWM_MAX_CONF_ID_LEN]);
```

回调函数

```
virtual void OnGetConfDetail(hwmsdk::HwmErrCode ret, const char* reason, const HwmConfDetail*
vmrInfoList) {};
```

参数描述

表 7-122 接口入参

参数	是否必须	类型	描述
confId	是	char[]	会议ID。

表 7-123 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

表 7-124 结构体 HwmConfDetail 字段名

字段	类型	描述
confListInfo	HwmConfListInfo	会议信息，参见 表3 。
vmrFlag	bool	是否是vmr会议。
vmrId	char[]	云会议室唯一ID。
isRecordOn	bool	是否有录制会议权限。

字段	类型	描述
isLiveOn	bool	保留。编辑会议时透传会议详情此字段。
isAutoRecord	bool	是否开启自动录制会议
isGuestFreePwd	bool	是否需要来宾免密。
isMailOn	bool	是否发送邮件通知。
isSmsOn	bool	是否发送短信通知。
isEmailCalendarOn	bool	是否发送日历通知。
joinConfRestrictionType	HwmJoinConfPermissionType	会议准入设置 表3 。
attendees	HwmAttendeeInfo*	与会者列表，字段信息参见 预约会议 中表述。只能查询会前预约时添加的与会人和会中主持人邀请的与会人。如果是与会人主动加入的，这个接口查不出来。
attendeeLen	unsigned int	与会者列表长度。
confServerType	HwmConfServerType	会议服务器类型 1:RTC or 0:MCU
isOpenWaitingRoom	bool	是否开启等候室
vmrConfIdType	HwmVmrConfIdType	云会议室的会议ID类型，参见 HwmVmrConfIdType 。
allowGuestStartConf	bool	是否允许来宾提前启动会议。
allowGuestStartConfTime	unsigned int	允许来宾提前入会时间范围(单位：分钟)：0-随时，n-提前n分钟启动会议。

示例代码

```
/**
 * 获取会议详情
 */
int demoEditConfDlg::GetConfDetail()
{
    string confID = CTools::UNICODE2UTF(CString("989156631"));
    int ret =hwmsdkagent::GetConfDetail(confID); // 使用会议id查询会议详细信息
    return ret;
}
```


7.5.5.6 修改云会议室信息

ChangeVmrInfo

接口描述

该接口用于修改云会议室信息。

注意事项

1. 该接口在已登录状态下才能调用。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。
3. 所有字段必填，调用该接口前需要获取VMR列表，不需要变更的数据请使用获取VMR列表接口返回的数据赋值。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode ChangeVmrInfo(const ChangeVmrInfoParam *param);
```

回调函数

```
virtual void OnChangeVmrInfo(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

参数	是否必须	类型	描述
vmrId	是	char[]	云会议室唯一ID。
vmrName	是	char[]	云会议室名称。
chairPwd	是	char[]	云会议室主持人密码。
guestPwd	是	char[]	云会议室来宾密码。
allowGuestStartConf	否	bool	是否允许来宾提前启动会议。
joinConfRestrictionType	否	HwmJoinConfRestrictionType	允许入会范围限制。
isOpenWaitingRoom	否	bool	等候室开启开关。

表 7-125 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```

/**
 * 修改云会议室信息
 */
int demoCreateConfWithAttendeeDlg::onClickEditVmrSubmit()
{
    hwmsdkagent::ChangeVmrInfoParam infoParam = {0};
    string name = CTools::UNICODE2UTF(CString("我的会场"));
    string chairPwd = CTools::UNICODE2UTF(CString("807766"));
    string guestPwd = CTools::UNICODE2UTF(CString("807755"));
    strcpy_s(infoParam.vmrName, sizeof(infoParam.vmrName), name.c_str());
    strcpy_s(infoParam.chairPwd, sizeof(infoParam.chairPwd), chairPwd.c_str());
    strcpy_s(infoParam.guestPwd, sizeof(infoParam.guestPwd), guestPwd.c_str());
    return hwmsdkagent::ChangeVmrInfo(&infoParam);
}

```

7.5.5.7 预约周期会议

BookCycleConf

接口描述

该接口用于预约周期会议。

注意事项

1. 该接口在已登录状态下才能调用。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。
3. 会议开始时间必须晚于当前时间，会议时长应大于15分钟。
4. 携带与会者入会的时候需要传入attendees。
5. 周期会议最大跨度一年，会议数量不超过50，超过则只预约前50个会议。
6. 回调函数中会返回预约成功后的会议详情，详见[HwmConfDetail](#)结构体表。如果参数内只有会议id有值则获取会议详情失败。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode BookCycleConf(const HwmBookCycleConfParam *bookCycleConfParam);
```

回调函数

```
virtual void OnBookCycleConfResult(hwmsdk::HwmErrCode ret, const char* reason, const HwmConfDetail* confDetail) {};
```

参数描述

表 7-126 结构体 HwmBookCycleConfParam 参数说明

参数	是否必须	类型	描述
bookCycleConfParam	是	HwmBookCycleConfParam	预约会议参数
cycleParam	是	HwmCycleConfParam	周期会议参数

表 7-127 结构体 HwmCycleConfParam 参数说明

参数	是否必须	类型	描述
startDate	是	long long	开始日期-时间戳, 精度秒(0时区)
endDate	是	long long	结束日期-时间戳, 精度秒(0时区)
cycleType	是	HwmCycleType	周期类型
interval	是	unsigned int	周期区间。 1、周期类型选择了按天, 表示每几天召开一次, 取值范围[1,15]; 2、周期类型选择了Week, 表示每几周召开一次, 取值范围[1,5]; 3、周期类型选择了Month, interval表示隔几月, 取值范围[1,3]
listPoints	是	char[]	周期内的会议召开点。仅当按周和月时有效。数据格式(中间用逗号隔开):1,3,5,7 - 当按周时, 取值范围[0,6], 0代表周日, 1代表周一, 以此类推; - 当按月时, 取值范围[1,31], 若当月不含31号, 则认为是当月最后一天
preRemindDays	是	unsigned int	周期子会议提前通知天数。

表 7-128 枚举 HwmCycleType 说明

枚举值	描述
CYCLE_TYPE_DAY	以天为周期。
CYCLE_TYPE_WEEK	以周为周期。
CYCLE_TYPE_MONTH	以月为周期。

示例代码

```
/**
 * 预约会议
 */
int CdemoBookConfDlg::OnBnClickedBookCycleConf()
{
    hwmsdkagent::HwmBookConfParam data{};
    string confSubject = u8"我的预约会议"; // 此处应是utf-8编码字符串
    strncpy_s(data.subject, confSubject.c_str(), HWM_MAX_SUBJECT_LEN);
    data.confType = hwmsdkagent::HwmConfMediaType::HWM_VIDEO_AND_DATA;
    data.isNeedConfPwd = true;
    data.isAutoRecordOn = true;
    data.isRecordOn = true;
    data.startTime = 1633017600; //utc时间戳, 单位秒, 如果获取的时间是本地时间, 需要转换成utc时间
    data.duration = 30; //会议持续时长(分钟)
```

```

data.timeZone = 56; //时区, 56表示东八区
data.joinConfRestrictionType = hwmsdkagent::HwmJoinConfPermissionType::RESTRICTION_CALL_IN_ALL;

std::vector<hwmsdkagent::HwmAttendeeInfo> attendeeList;
//被邀与会者
hwmsdkagent::HwmAttendeeInfo attendeeInfo{};
string attName = "Mike";
string attNumber = "+991116003543";
string attEmail = "";
string attSms = "";
string attUserId = "";
strncpy_s(attendeeInfo.name, attName.c_str(), HWM_MAX_USER_NAME_LEN);
strncpy_s(attendeeInfo.number, attNumber.c_str(), HWM_MAX_NUMBER_LEN);
strncpy_s(attendeeInfo.email, attEmail.c_str(), HWM_MAX_EMAIL_LEN);
strncpy_s(attendeeInfo.sms, attSms.c_str(), HWM_MAX_PHONE_NUM_LEN);
strncpy_s(attendeeInfo.thirdUserId, attUserId.c_str(), HWM_MAX_USER_ID_LEN);
attendeeInfo.isMute = true;
attendeeInfo.isAutoInvite = true;
attendeeList.push_back(attendeeInfo);

data.attendees = attendeeList.data();
data.attendeeLen = attendeeList.size();
data.isSmsOn = true;
data.isMailOn = true;
data.isEmailCalendarOn = true;
hwmsdkagent::HwmCycleConfParam cycleConfParam{};
cycleConfParam.startDate = 1633017600; //utc时间戳, 单位秒, 如果获取的时间是本地时间, 需要转换成
utc时间
cycleConfParam.endDate = 1636560000; //utc时间戳, 单位秒, 如果获取的时间是本地时间, 需要转换成utc
时间
cycleConfParam.cycleType = hwmsdkagent::CYCLE_TYPE_WEEK;
strcpy_s(cycleConfParam.listPoints, sizeof(cycleConfParam.listPoints), "1,5"); // 每周的星期1, 星期5
cycleConfParam.preRemindDays = 1;

hwmsdkagent::HwmBookCycleConfParam bookCycleConfParam{};
bookCycleConfParam.bookConfParam = data;
bookCycleConfParam.cycleParam = cycleConfParam;
return hwmsdkagent::BookCycleConf(&bookCycleConfParam);
}

```

7.5.5.8 编辑周期会议

ModifyCycleConf

接口描述

该接口用于编辑周期会议。

注意事项

1. 该接口在已登录状态下才能调用。
2. 该接口为异步接口, 返回值只代表接口是否调用成功, 实际业务处理结果在对应的回调函数中返回。
3. 所有字段必填, 编辑修改的字段可来源于用户输入, 其他不想修改的属性一定要取会议详情对应字段来赋值。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode ModifyCycleConf(const HwmModifyCycleConfParam *modifyParam);
```

回调函数

```
virtual void OnModifyCycleConfResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

表 7-129 结构体 HwmModifyCycleConfParam 参数说明

参数	是否必须	类型	描述
editConfParam	是	HwmEditConfParam	编辑会议参数
cycleParam	是	HwmCycleConfParam	周期会议参数

示例代码

```

/**
 * 编辑周期会议接口
 */
int demoEditConfDlg::ModifyCycleConf()
{
    hwmsdkagent::HwmConfDetail confDetail{}; //从缓存中得到会议详情数据赋予confDetail
    hwmsdkagent::HwmEditConfParam editConfParam{};
    //根据实际需求拷贝confDetail的数据
    strncpy_s(editConfParam.confId, confDetail.confListInfo.confId, HWM_MAX_CONF_ID_LEN);
    editConfParam.vmrFlag = confDetail.vmrFlag;
    strncpy_s(editConfParam.vmrId, confInfo.vmrId, HWM_MAX_VMR_CONF_ID_LEN);
    //此处省略部分赋值代码
    editConfParam.timeZone = 56; // 本地时区, 用户依据自身情况自己调整, 56东八区
    editConfParam.isRecordOn = false;
    editConfParam.isAutoRecordOn = false;
    editConfParam.startTime = 1633017600; //utc时间戳
    editConfParam.duration = 0 * 60 + 30; //会议时长
    editConfParam.joinConfRestrictionType =
hwmsdkagent::HwmJoinConfPermissionType::RESTRICTION_CALL_IN_ALL; //取编辑会议上面的允许入会用户类型
    // vmrConferenceId非空是固定ID,为空则是随机id
    editConfParam.vmrConfIdType = (strlen(confInfo.confListInfo.vmrConferenceId) != 0)
        ? hwmsdkagent::HwmVmrConfIdType::HWM_VMR_CONF_ID_TYPE_FIXED
        : hwmsdkagent::HwmVmrConfIdType::HWM_VMR_CONF_ID_TYPE_RANDOM;
    editConfParam.allowGuestStartConf = false;
    editConfParam.allowGuestStartConfTime = 0;

    hwmsdkagent::HwmCycleConfParam cycleConfParam{};
    cycleConfParam.startDate = 1633017600; //utc时间戳, 单位秒, 如果获取的时间是本地时间, 需要转换成
utc时间
    cycleConfParam.endDate = 1636560000; //utc时间戳, 单位秒, 如果获取的时间是本地时间, 需要转换成
utc时间
    cycleConfParam.cycleType = 1;
    strcpy_s(cycleConfParam.listPoints, sizeof(cycleConfParam.listPoints), "5"); // 星期五
    cycleConfParam.preRemindDays = 1;
    cycleConfParam.interval = 1;

    hwmsdkagent::HwmModifyCycleConfParam editCycleConfParam{};
    editCycleConfParam.editConfParam = editConfParam;
    editCycleConfParam.cycleParam = cycleConfParam;

    return hwmsdkagent::ModifyCycleConf(&editCycleConfParam);
}

```

7.5.5.9 取消周期会议

CancelCycleConf

接口描述

该接口用于取消至少还有一场子会议未开始的周期会议。

注意事项

1. 该接口在登录状态下才能调用。
2. 最后一场子会议未开始前可以调用接口，开始后调用则无效。
3. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode CancelCycleConf(const HwmCancelCycleConfParam
*cancelCycleConfParam);
```

回调函数

```
virtual void OnCancelCycleConfResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

表 7-130 结构体 HwmCancelCycleConfParam 参数说明

参数	是否必须	类型	描述
confId	是	char[]	会议id
endOnlineConf	是	bool	是否需要结束在线会议

示例代码

```
/**
 * 取消周期会议接口
 */
int demoConfListDlg::OnBnCancelCycleConf()
{
    hwmsdkagent::HwmCancelCycleConfParam cancelCycleConfParam{};
    cancelCycleConfParam.confId= CTools::UNICODE2UTF(CString("989156631"));
    cancelCycleConfParam.endOnlineConf = true;
    return hwmsdkagent::CancelCycleConf(cancelCycleConfParam);
}
```

7.5.5.10 编辑周期子会议

ModifySubCycleConf

接口描述

该接口用于编辑周期子会议。

注意事项

1. 该接口在已登录状态下才能调用。
2. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode ModifySubCycleConf(const HwmModifySubCycleConfParam *modifySubCycleConfParam);
```

回调函数

```
virtual void OnModifySubCycleConfResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

参数描述

表 7-131 结构体 HwmModifySubCycleConfParam 参数说明

参数	是否必须	类型	描述
confId	是	char[]	会议id
subConfID	是	char[]	子会议唯一标识UUID
confMediaType	否	HwmConfMediaType	会议媒体类型
startTime	否	long long	会议开始时间（UTC时间），单位为秒
confLen	否	unsigned int	会议持续时长，单位分钟
isAutoRecord	否	bool	会议开始后是否自动开启录制
recordAuthType	否	unsigned int	录播鉴权方式 0:可通过连接观看/下载（不拼接nonce）1:企业用户可观看/下载 2:与会者可观看/下载
callInRestriction	否	HwmJoinConfPermissionType	呼入限制
allowGuestStartConf	否	bool	允许来宾启动会议
allowGuestStartConfTime	否	unsigned int	允许来宾提前入会时间范围(单位：分钟) 0-随时，n-提前n分钟启动会议

示例代码

```
/**
 * 编辑周期子会议接口
 */
int demoConfListDlg::ModifySubCycleConf()
{
    hwmsdkagent::HwmModifySubCycleConfParam SubCycleConfParam{};
    strcpy_s(SubCycleConfParam.confId, sizeof(SubCycleConfParam.confId), confId.c_str());
    strcpy_s(SubCycleConfParam.subConfID, sizeof(SubCycleConfParam.subConfID), subConfUuid.c_str());
    SubCycleConfParam.confMediaType = 1;
}
```

```

SubCycleConfParam.startTime = 1633017600;
SubCycleConfParam.confLen = 60;
SubCycleConfParam.isAutoRecord = false;
SubCycleConfParam.callInRestriction = 0;
SubCycleConfParam.allowGuestStartConf = false;
SubCycleConfParam.allowGuestStartConfTime = 0;

return hwmsdkagent::ModifySubCycleConf(&SubCycleConfParam);
}

```

7.5.5.11 取消周期子会议

CancelSubCycleConf

接口描述

该接口用于取消未开始的周期子会议。

注意事项

1. 该接口在登录状态下才能调用。
2. 子会议未开始前可以调用接口，开始后调用则无效。
3. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```

HWM_SDK_AGENT_API hwmsdk::HwmErrCode CancelSubCycleConf(const HwmCancelSubCycleConfParam
*cancelSubCycleConfParam);

```

回调函数

```

virtual void OnCancelSubCycleConfResult(hwmsdk::HwmErrCode ret, const char* reason) {};

```

参数描述

表 7-132 结构体 HwmCancelCycleConfParam 参数说明

参数	是否必须	类型	描述
confId	是	char[]	会议id
subConfID	是	char[]	子会议唯一标识UUID
endOnlineConf	是	bool	是否需要结束在线会议

示例代码

```

/**
 * 取消周期子会议接口
 */
int demoSubCycleConfList::CancelSubCycleConf()
{
    hwmsdkagent::HwmCancelSubCycleConfParam subCycleConfParam;
    strcpy_s(subCycleConfParam.confId, sizeof(subCycleConfParam.confId), m_confInfo.confListInfo.confId);
    strcpy_s(subCycleConfParam.subConfID, sizeof(subCycleConfParam.subConfID),
m_subCycleConfID.c_str());
    subCycleConfParam.endOnlineConf = true;
}

```



```
hwmsdkagent::CancelSubCycleConf(&subCycleConfParam);
}
```

7.5.6 获取会议信息

7.5.6.1 获取与会者列表

GetAttendeeList

接口描述

该接口用于获取会议中与会者列表。

注意事项

1. 只能在会议中使用

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode GetAttendeeList();
```

回调函数

```
virtual void OnGetAttendeeListResult(const HwmAttendeeInfo* attendeeList, unsigned int length) override{};
```

参数描述

[HWMAttendeeInfo](#) 参数说明

表 7-133 枚举 ConfAttendeeState 说明

参数	描述
CONF_ATTENDEE_STATE_IN_CONF	会议中
CONF_ATTENDEE_STATE_CALLING	正在呼叫
CONF_ATTENDEE_STATE_LEAVED	已经离开

示例代码

```
/**
 * 获取与会者列表
 */
void demoSettingDlg::OnBnClickedButtonGetAttendeeList()
{
    hwmsdkagent::GetAttendeeList();
}

/**
 * 获取与会者列表回调
 */
void demoCallbackProc::OnGetAttendeeListResult(const HwmAttendeeInfo* attendeeList, unsigned int length)
{
    std::vector< HwmAttendeeInfo> attendeeListInfo(attendeeList, attendeeList + length);
}
```

7.5.6.2 获取观众列表

GetAudienceList

接口描述

该接口用于获取会议中观众列表。

注意事项

1. 只能在会议中使用
2. 观众身份（可以通过会议详情消息推送获取）不能使用此接口

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode GetAudienceList();
```

回调函数

```
virtual void OnGetAudienceListResult(const HwmAttendeeInfo* audienceList, unsigned int length)override {};
```

参数描述

[HWMAttendeeInfo](#)参数说明

7.5.6.3 获取用户状态

GetUserState

接口描述

该接口用于获取当前用户终端的状态。

注意事项

1. 该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode GetUserState();
```

回调函数

```
virtual void OnGetUserStateResult(hwmsdk::HwmErrCode ret, const char* reason, HwmUserState userState) {};
```

参数描述

表 7-134 枚举 HwmUserState 说明

枚举值	描述
HWM_STATE_IDLE	空闲状态
HWM_STATE_IN_CONF	会中状态
HWM_STATE_IN_CALL	呼叫中状态

返回值

表 7-135 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

示例代码

```
/**
 * 获取用户状态
 */
void demoSetUserSavePathDlg::OnBnClickedGetUserState()
{
    int ret = hwmsdkagent::GetUserState();
    if (hwmsdk::HWM_COMMON_SUCCESS != ret)
    {
        AfxMessageBox(_T("Get user state failed"));
    }
}
```

7.5.7 音视频数据流

7.5.7.1 获取音频流

接口描述

该接口用于获取会议中各会场混音后的音频数据流。

注意事项

该接口为异步接口，回调函数返回值只代表接口是否调用成功，实际业务处理结果在对应的通知函数中返回。

在进入会议前就可以调用此接口，创建或加入会议后就会有通知。

方法定义

```
HWM_SDK_AGENT_API hwmsdk::HwmErrCode SetAudioRawDataOutputConfig(bool isOpen);
```

回调函数

```
virtual void OnSetAudioRawDataOutputConfigResult(hwmsdk::HwmErrCode ret, const char* reason) {};
```

通知函数

```
virtual void OnAudioFrameNotify(const AudioFrameData* pFrame) {};
```

参数描述

表 7-136 参数说明

参数	是否必须	类型	描述
isOpen	是	bool	是否开启功能开关。

表 7-137 返回值

类型	描述
HwmErrCode	成功返回0，其他值表示失败。失败返回值见 错误码参考 。

表 7-138 结构体 AudioFrameData 参数说明

参数	类型	描述
eFrameType	AudioFrameType	音频帧类型
iSamples	signed int	每个声道的采样点数
iSamplesPerSec	signed int	采样率
iBytesPerSample	signed int	每个采样点的字节数,pcm数据一般16bit
iChannels	signed int	声道数
pBuffer	char *	数据缓冲区,长度 = iSamples*iBytesPerSample*iChannels
uiDataLen	signed int	编码后或解码前使用该字段表示数据长度，加解密处理后需更新此字段值
uiBufferLen	signed int	缓冲区长度，编码后数据进行第三方加密，数据长度会变大，防止溢出

表 7-139 枚举 AudioFrameType 说明

枚举值	描述
AUDIO_FRAME_TYPE_PCM16	16位精度PCM

示例代码

```
/**
 * 上报音频码流帧数据通知
 */
void demoNotifyProc::OnAudioFrameNotify(const AudioFrameData* pFrame)
{
    if(nullptr != pFrame)
    {
        FILE* recordFile = nullptr;
        std::string Path ("D:\\AudioFrame.pcm");
        auto error = fopen_s(&recordFile,Path.c_str(), "ab+");
        if (error != 0)
        {
            CTools::OutputRetStr("open file error = " + std::to_string(error));
        }
    }
}
```

```
        if (nullptr != recordFile)
        {
            fwrite(pFrame->pBuffer, sizeof(char), pFrame->iSamples * pFrame->iBytesPerSample * pFrame->iChannels, recordFile);
            fclose(recordFile);
        }
    }
}
```

7.6 通知参考

7.6.1 基础配置变更通知

7.6.1.1 连接断开

通知描述

收到该通知时，表示与SDK的连接断开了。

方法定义

```
virtual void OnSdkDisconnected() {};
```

注意事项

收到该通知，表明SDK发生异常断开，需要重新调用初始化接口，否则SDK无法使用。

参数描述

无。

7.6.1.2 企业配置变更

通知描述

收到该通知时，表示企业配置信息变更。

方法定义

```
virtual void OnCorpConfig(HwmCorpConfigInfo* data) {};
```

注意事项

无。

参数描述

表 7-140 结构体 HwmCorpConfigInfo 参数说明

参数	类型	描述
enablePstn	bool	是否支持手机呼叫。
enableSms	bool	是否支持短信通知。
enableRecord	bool	是否支持会议录制。
allowGuestStartConf	bool	是否允许来宾提前启动会议。
allowGuestStartConfTime	unsigned int	允许来宾提前入会时间范围(单位: 分钟): 0-随时, n-提前n分钟启动会议。
enableShowAddressBookStructure	bool	是否支持通讯录显示。

7.6.1.3 企业信息变更

通知描述

收到该通知时, 表示客户企业信息变更。

方法定义

```
virtual void OnUserCorpInfo(HwmUserCorpInfoNotifyType* type) = 0;
```

注意事项

无。

参数描述

表 7-141 枚举 HwmUserCorpInfoNotifyType 说明

参数	描述
HWM_USER_CORP_INFO_NOTIFY_TYPE_USER_DELETE	用户已注销。
HWM_USER_CORP_INFO_NOTIFY_TYPE_USER_JOIN_NEW_CORP	用户已加入新的企业(接受邀请、创建企业变成新企业的用户)。
HWM_USER_CORP_INFO_NOTIFY_TYPE_USER_LEAVE_CURRENT_CORP	用户已被移出当前企业(企业管理员删除、解散、用户主动离开企业变成注册用户)。

参数	描述
HWM_USER_CORP_INFO_NOTIFY_TYPE_USER_RESET	用户重置（Smartrooms账号被重置激活码或者账号被删除）。
HWM_USER_CORP_INFO_NOTIFY_TYPE_USER_VMR_CHANGE	用户vmr变化。
HWM_USER_CORP_INFO_NOTIFY_TYPE_USER_NAME_CHANGE	用户姓名变更。
HWM_USER_CORP_INFO_NOTIFY_TYPE_USER_STATUS_CHANGE	接入账号状态变化。

7.6.2 帐号状态变更通知

7.6.2.1 帐号被踢出

通知描述

收到该通知时，表示当前使用的帐号被其他终端踢出，退出登录状态，需要重新登录。

方法定义

```
virtual void OnKickedOut() {};
```

注意事项

无。

参数描述

无。

7.6.3 定制界面事件通知

7.6.3.1 注入点击

通知描述

收到该通知时，用户点击了配置过点击注入的按钮。

方法定义

```
virtual void OnClickInjectBtn(HwmClickInjectBtn injectBtn, void* data, const char* id) {};
```

注意事项

无。

参数描述

表 7-142 参数说明

参数	类型	描述
injectBtn	HwmClickInjectBtn	按钮类型。
data	void*	按钮参数，可能为空。 <ul style="list-style-type: none"> 若按钮类型为 HWM_CLICK_INJECT_BTN_CALL_OTHER_NUM，该参数为期望呼叫的与会者信息； 若按钮类型为 HWM_CLICK_INJECT_BTN_INVITE，该参数为空。
id	const char*	自定义按钮id。 <ul style="list-style-type: none"> 当按钮类型为 HWM_TOOLS_INJECT_BTN_CUSTOM 时，该参数为自定义按钮的id； 当按钮类型为 HWM_CLICK_INJECT_BTN_PARTICIPANT_DETAIL，该参数为 HwmConfAttendee； 当按钮类型为其他值时，该参数为空。

表 7-143 枚举 HwmClickInjectBtn 说明

枚举值	描述
HWM_CLICK_INJECT_BTN_INVITE	邀请按钮。
HWM_CLICK_INJECT_BTN_CALL_OTHER_NUM	呼叫其他号码按钮。
HWM_CLICK_INJECT_BTN_ATTENDEE	与会者按钮。
HWM_CLICK_INJECT_BTN_SHARE	共享按钮。
HWM_CLICK_INJECT_BTN_LEAVE	离开按钮。

枚举值	描述
HWM_TOOLS_INJECT_BTN_CUSTOM	用户自定义按钮。
HWM_CLICK_INJECT_BTN_COPY_CONF_INFO	复制会议信息按钮。
HWM_CLICK_INJECT_BTN_PARTICIPANT_DETAIL	与会者详情按钮。

7.6.3.2 显示会议信息框

通知描述

收到该通知时，表示用户将鼠标移至会中界面左上角的会议ID处。

方法定义

```
virtual void OnShowExternalConfInfoWnd(int left,int top) {};
```

注意事项

1 收到该通知的前提需要打开 [7.5.3.1 表2 frame对象介绍](#) 中的 isUseExternalConfInfoWnd开关为true。

2 通知方法OnShowExternalConfInfoWnd中添加线程切换，抛到用户自己线程中，再显示用户自定义的

会议信息框(会议信息框显示需做置顶操作)。

3 显示用户自定义的会议信息框接口中，需要添加定时器检测(2s):

鼠标若未移入自定义会议信息框区域，定时器时间到则销毁会议信息框。

鼠标若移入自定义会议信息框区域，则待鼠标移出信息框区域直接销毁会议信息框。

(若未添加3处理，无法覆盖所有的场景销毁会议信息框)

参数描述

表 7-144

参数	类型	描述
left	int	会议信息框左上角坐标的x。
top	int	会议信息框左上角坐标的y。

示例代码

```
/**
 * 显示外部会议信息框
```

```

*/
LRESULT demoMainMenuDlg::OnShowExternalConfInfoWnd(WPARAM wParam, LPARAM lParam)
{
    CPoint* pt = (CPoint*)wParam;
    demoConfInfoDlg* dlg = demoData::GetInstance().GetConfInfoDlgObj();
    // 若窗口对象不存在, 则创建窗口对象
    if (dlg == NULL)
    {
        dlg = new demoConfInfoDlg;
        dlg->Create(IDD_CONF_INFO_DIALOG, NULL);
        demoData::GetInstance().SetConfInfoDlgObj(dlg);
    }
    // 若窗口存在, 则发起定时器检测窗口销毁
    if (IsWindow(dlg->GetSafeHwnd()))
    {
        SetTimer(1, 2000, NULL);
    }
    // 显示窗口且置顶
    CRect rectDlg(0, 0, 0, 0);
    dlg->GetClientRect(rectDlg);
    CRect rt(pt->x, pt->y, pt->x + rectDlg.Width(), pt->y + rectDlg.Height());
    dlg->MoveWindow(rt);
    dlg->SetWindowPos(&CWnd::wndTopMost, pt->x, pt->y, rectDlg.Width(), rectDlg.Height(),
    SWP_NOMOVE | SWP_NOSIZE | SWP_NOACTIVATE);
    dlg->ShowWindow(SW_SHOW);

    delete pt;
    pt = NULL;
    return 0;
}
/**
 * 定时器检测销毁会议信息框
 */
void demoMainMenuDlg::OnTimer(UINT_PTR nIDEvent)
{
    if (nIDEvent == 1)
    {
        demoConfInfoDlg* dlg = demoData::GetInstance().GetConfInfoDlgObj();
        if (dlg == NULL)
        {
            return;
        }

        if (IsWindow(dlg->m_hWnd))
        {
            CPoint ptMouse(0, 0);
            GetCursorPos(&ptMouse);

            CRect rtConfInfo(0,0,0,0);
            dlg->GetWindowRect(&rtConfInfo);
            // 鼠标不在会议信息框范围内则销毁
            if (!rtConfInfo.PtInRect(ptMouse))
            {
                {
                    dlg->CloseWindow();
                    demoData::GetInstance().SetConfInfoDlgObj(NULL);
                    delete dlg;
                    dlg = NULL;
                    KillTimer(nIDEvent);
                }
            }
        }
    }
    CDialogEx::OnTimer(nIDEvent);
}
/**
 * 会议信息框添加鼠标移出销毁自身窗口
 */
void demoConfInfoDlg::OnMouseMove(UINT nFlags, CPoint point)
{
    if (!m_bTracking)

```

```

{
    TRACKMOUSEEVENT t_MouseEvent;
    t_MouseEvent.cbSize = sizeof(TRACKMOUSEEVENT);
    t_MouseEvent.dwFlags = TME_HOVER | TME_LEAVE;
    t_MouseEvent.hwndTrack = m_hWnd;
    t_MouseEvent.dwHoverTime = 30;

    // 触发OnMouseLeave or OnMouseHover事件, m_bTracking 防止频繁触发事件
    if (::TrackMouseEvent(&t_MouseEvent))
    {
        m_bTracking = true;
    }
}
CDialogEx::OnMouseMove(nFlags, point);
}

void demoConfInfoDlg::OnMouseLeave()
{
    m_bTracking = false;
    CDialogEx::OnMouseLeave();
    demoData::GetInstance().SetConfInfoDlgObj(NULL);
    OnCancel();
}

```

7.6.3.3 销毁会议信息框

通知描述

收到该通知时，用户存在的操作场景较多，如点击会中共享按钮、点击会中离开按钮、离会、结束会议、点击标题栏、会控方式共享等操作。

方法定义

```
virtual void OnDestroyExternalConfInfoWnd() {};
```

注意事项

判断会议信息框窗口对象是否存在，若存在才做销毁操作。

参数描述

无。

示例代码

```

/**
 * 销毁外部会议信息框
 */
LRESULT demoMainMenuDlg::OnDestroyExternalConfInfoWnd(WPARAM wParam, LPARAM lParam)
{
    demoConfInfoDlg* dlg = demoData::GetInstance().GetConfInfoDlgObj();
    if (dlg == NULL)
    {
        return 1;
    }

    dlg->CloseWindow();
    demoData::GetInstance().SetConfInfoDlgObj(NULL);
    delete dlg;
    dlg = NULL;
    return 0;
}

```

7.6.3.4 查询外部通讯录头像

通知描述

该接口用于实现视频画面窗口中，未打开摄像头时，显示自定义头像。使用该接口需要先在“[界面配置](#)”接口中把isUseExternalAddressBook设置成true，并在这个通知接口中返回需要需要显示头像的图片给SDK。

方法定义

```
virtual void OnQueryContactAvatar(const char account[HWM_MAX_ACCOUNT_LEN], const char thirdUserId[HWM_MAX_ACCOUNT_LEN], const char sipNumber[HWM_MAX_NUMBER_LEN], char* avatarPath,int pathLen) {};
```

注意事项

用户侧需要在OnQueryContactAvatar通知方法里，根据account账号、thirdUserId第三方账户、sipNumber（sip号码）查询自己的通讯录，获取到avatarPath头像路径，并赋值到avatarPath里面，路径长度上限为pathLen。

参数描述

表 7-145

参数	类型	描述
account	char[]	登录华为云会议的帐号信息。
thirdUserId	char[]	登录华为云会议的第三方userId。
sipNumber	char[]	帐号对应的sip号码。
avatarPath	char*	查询通讯录返回的头像路径。
pathLen	int	最大路径长度。

示例代码

```
/**
 * 查询外部通讯录头像
 */
void demoNotifyProc::OnQueryContactAvatar(const char* account, const char* thirdUserId, const char* sipNumber, char* avatarPath, int pathLen)
{
    Cdemo* app = (Cdemo*)AfxGetApp();
    if (!app || (strlen(account) == 0 && strlen(thirdUserId) == 0 && strlen(sipNumber) == 0))
    {
        //窗口已经关闭或数据为空
        return;
    }

    string sAccount = account;
    string sThirdUserId = thirdUserId;
    string sSipNumber = sipNumber;

    /* 根据用户层自己的逻辑 + account thirdUserId sipNumber 获取到头像图片路径信息并复制到avatarPath char数组内*/
    // 测试使用时，在D:\picture\路径下放置一张图片test.png
```

```
strcpy_s(avatarPath, pathLen, "D:\\picture\\test.png");
}
```

7.6.4 会议状态变更通知

7.6.4.1 会议来电

通知描述

收到该通知时，表示收到会议来电。

方法定义

```
virtual void OnConfIncoming(HwmConfIncomingInfo* data) {};
```

注意事项

无。

参数描述

表 7-146 结构体 HwmConfIncomingInfo 参数说明

参数	类型	描述
state	HwmConfComingState	来电状态
mediaType	HwmMediaType	会议媒体类型。
confId	char[]	会议id。

表 7-147 枚举 HwmConfComingState 说明

枚举值	描述
HWM_CONF_COMING_STATE_START	初始状态，来电振铃中。
HWM_CONF_COMING_STATE_ACCEPT	接听状态，已接听来电。
HWM_CONF_COMING_STATE_REJECT	拒绝状态，已拒绝来电。
HWM_CONF_COMING_STATE_TIMEOUT	未响应状态，呼叫超时或者主叫已挂断。

表 7-148 枚举 HwmMediaType 说明

枚举值	描述
HWM_MEDIA_TYPE_AUDIO	音频。
HWM_MEDIA_TYPE_VIDEO	视频。

7.6.4.2 会议状态

通知描述

收到该通知时，表示会议状态发生变更。

方法定义

```
virtual void OnConfState(HwmConfStateInfo *confStateInfo) {};
```

注意事项

无。

参数描述

表 7-149 结构体 HwmConfStateInfo 参数说明

参数	类型	描述
state	HwmState	会议状态。
reason	HwmConfEndReason	离会原因。

表 7-150 枚举 HwmState 说明

枚举值	描述
HWM_STATE_OFF	会议未开始或已离开状态。
HWM_STATE_ON	会议已开始状态。

表 7-151 枚举 HwmConfEndReason 说明

枚举值	描述
HWM_CONF_END_REASON_LEAVE_CONF_HANGUP	离开会议。

枚举值	描述
HWM_CONF_END_REASON_STOP_CONF_HANGUP	会议已结束。
HWM_CONF_END_REASON_CHAIR_HANGUP	被主持人移出会场。
HWM_CONF_END_REASON_SESSION_TIMER_TIMEOUT	网络中断。
HWM_CONF_END_REASON_NOSTREAM_HANGUP	网络异常。
HWM_CONF_END_REASON_CORP_CONFERENCE_RESOURCE_HAS_BEEN_RUN_OUT	企业资源已达上限。
HWM_CONF_END_REASON_CONF_ONLY_SAME_ENTERPRISE	只允许企业内用户加入会议
HWM_CONF_END_REASON_CONF_ONLY_INVITED	只允许被邀请者加入会议
HWM_CONF_END_REASON_CONF_IS_LOCKED	会议被锁定
HWM_CONF_END_ERR_VMR_NOT_BEEN_HOLD	VMR会议来宾不允许激活会议
HWM_CONF_END_REASON_VMR_CONF_CONFLICT	VMR会议冲突
HWM_CONF_END_REASON_SERVER_ERROR	服务器异常
HWM_CONF_END_REASON_NUM_OF_PAR_EXCEEDS_UPPER_LIMIT	与会方超出专用VMR套餐的最大与会人数
HWM_CONF_END_REASON_PARTICIPANT_REACH_MAX_NUM = 111072504	达到系统单个会议人数上限

7.6.4.3 会议信息

通知描述

收到该通知时，表示会议中的信息发生变更。

方法定义

```
virtual void OnConfInfo(HwmConfInfo *confInfo) {};
```

注意事项

会议开始后，会议信息会上报多次，不是一次把HwmConfInfo中的信息全部上报上来的。

参数描述

表 7-152 结构体 HwmConfInfo 参数说明

参数	类型	描述
url	char[]	当前会议链接，可以分享通过浏览器打开进行入会。
confId	char[]	会议id。
role	HwmConfRole	会议角色。主席、普通与会者等。
chairmanPwd	char[]	会议主席密码。
generalPwd	char[]	会议来宾密码。
subject	char[]	会议主题。
isOtherCorpConf	bool	是否是外部企业的会议。如果是加入外部企业的会议或者匿名入会是true，否则是false。
isCreator	bool	是否是会议预订者。
isInBreakoutSubConf	bool	是否在讨论组中。

表 7-153 枚举 HwmConfRole 说明

枚举值	描述
HWM_CONF_ROLE_ATTENDEE	普通与会者。
HWM_CONF_ROLE_CHAIRMAN	主席。
HWM_CONF_ROLE_AUDIENCE	观众。
HWM_CONF_ROLE_COHOST	联席主持人。
HWM_CONF_ROLE_WAITING	等候室成员。

7.6.4.4 会议列表通知

通知描述

收到该通知时，表示会议列表信息发生变更。

方法定义

```
virtual void OnConfList(HwmConfListUpdateType updateType, const HwmConfListInfo *confInfoList,
    unsigned int confInfoLen) {};
```

注意事项

登录成功后，会上报一次该通知，后续每当会议列表发生变更会再次上报。

预约会议、编辑会议、取消会议接口都会触发该通知，其中编辑会议触发该通知上报的更新类型是CONFLIST_UPDATE_ADD。

当更新类型(updateType)是会议列表全量更新(CONFLIST_UPDTAE_ALL)时如果confInfoLen为0时需要清空会议列表。

参数描述

表 7-154 参数说明

参数	类型	描述
updateType	HwmConfListUpdateType	更新类型。
confInfoList	HwmConfListInfo*	会议列表参数信息。
confInfoLen	unsigned int	会议列表参数信息的长度。

表 7-155 枚举 HwmConfListUpdateType 说明

枚举值	描述
CONFLIST_UPDTAE_ALL	会议列表全量更新。
CONFLIST_UPDATE_ADD	会议列表有增加或者修改。编辑会议导致会议列表变更会上报该值。
CONFLIST_UPDATE_DELETE	会议列表有删除。

表 7-156 结构体 HwmConfListInfo 参数说明

参数	类型	描述
confId	char[]	会议ID。

参数	类型	描述
confSubject	char[]	会议主题。
chairmanPwd	char[]	主席密码。
guestPwd	char[]	来宾密码。
audiencePwd	char[]	观众密码。
startTime	long long	会议开始时间(utc时间)，单位秒。
endTime	long long	会议结束时间(utc时间)，单位秒。
timeZone	char[]	时区码
mediaType	HwmMediaType	媒体类型 表2 。
scheduserName	char[]	会议预定者名称。
vmrConferencelid	char[]	用于云会议显示的ID。
chairJoinUri	char[]	主持人加入会议uri链接
guestJoinUri	char[]	来宾加入会议uri链接
audienceJoinUri	char[]	观众加入会议的链接
confStateType	HwmConfState	会议状态
isWebinar	bool	是否是网络研讨会
selfConfRole	HwmConfRole	本人在会议中的角色 表5 。

表 7-157 枚举 HwmConfState 说明

枚举值	描述
HWM_CONF_STATE_SCHE DULE	预定状态。
HWM_CONF_STATE_CREAT ING	正在创建状态。
HWM_CONF_STATE_GOIN G	会议已经开始。
HWM_CONF_STATE_DESTR OYED	会议已经关闭

7.6.4.5 发起数据共享状态

通知描述

收到该通知时，表示当前发起数据共享的状态有所变更。

方法定义

```
virtual void OnShareState(HwmShareStateInfo *shareStateInfo) {};
```

注意事项

无。

参数描述

表 7-158 结构体 HwmShareStateInfo 参数说明

参数	类型	描述
state	HwmState	共享状态。
name	char[]	共享者名称。当state为HWM_STATE_ON时才有数据。
number	char[]	共享者号码。当state为HWM_STATE_ON时才有数据。
shareType	HwmShareType	共享类型。
monitor	HwmMonitor	共享显示器类型（主屏、副屏）。
appHandle	HWND	程序句柄。

表 7-159 枚举 HwmState 说明

枚举值	描述
HWM_STATE_OFF	共享已关闭。
HWM_STATE_ON	共享已开启。

表 7-160 枚举 HwmShareType 说明

枚举值	描述
HWM_SHARE_TYPE_MONITOR	屏幕共享。
HWM_SHARE_TYPE_APP	程序共享。

枚举值	描述
HWM_SHARE_TYPE_WHITE_BOARD	白板共享。

表 7-161 枚举 HwmMonitor 说明

枚举值	描述
HWM_MONITOR_MAIN	主显示器。
HWM_MONITOR_SECONDARY	副显示器。

7.6.4.6 接收数据共享状态

通知描述

收到该通知时，表示当前接收数据共享的状态发生了变更。

方法定义

```
virtual void OnRecvShareState(HwmRecvShareStateInfo *shareStateInfo) {};
```

注意事项

无。

参数描述

表 7-162 结构体 HwmRecvShareStateInfo 参数说明

参数	类型	描述
state	HwmState	共享状态。
name	char[]	共享者名称。当state为HWM_STATE_ON时才有数据。
number	char[]	共享者号码。当state为HWM_STATE_ON时才有数据。

表 7-163 枚举 HwmState 说明

枚举值	描述
HWM_STATE_OFF	接收的共享已关闭。

枚举值	描述
HWM_STATE_ON	开始接收到共享。

7.6.4.7 得到远程控制权限

通知描述

收到该通知时，表示当前远程控制权限状态发生变化：state为1时，从name处获得远程控制权限，state为0时，name的远程控制权限被释放、收回。

方法定义

```
virtual void OnRemoteControlState(HwmRemoteControlStateInfo *remoteControlStateInfo) {};
```

注意事项

无。

参数描述

表 7-164 结构体 HwmRemoteControlStateInfo 参数说明

参数	类型	描述
state	HwmState	状态。
name	char[]	权限所有者名称。
number	char[]	权限所有者号码。

表 7-165 枚举 HwmState 说明

枚举值	描述
HWM_STATE_OFF	得到的远程控制权限被收回。
HWM_STATE_ON	得到远程控制权限。

7.6.4.8 授予他人远程控制权限

通知描述

收到该通知时，表示当前远程控制权限状态发生变化：state为1时，把远程控制权限授予给name，state为0时，name的远程控制权限被释放、收回。

方法定义

```
virtual void OnBeRemoteControlledState(HwmRemoteControlStateInfo *remoteControlStateInfo) {};
```

注意事项

无。

参数描述

表 7-166 结构体 HwmRemoteControlStateInfo 参数说明

参数	类型	描述
state	HwmState	状态。
name	char[]	权限所有者名称。
number	char[]	权限所有者号码。

表 7-167 枚举 HwmState 说明

枚举值	描述
HWM_STATE_OFF	收回授予的远程控制权限。
HWM_STATE_ON	授予他人远程控制权限。

7.6.4.9 共享锁定状态变化通知

通知描述

会议是否共享锁定状态变化通知。

方法定义

```
virtual void OnConfIsShareLockedChanged(bool isShareLocked) {};
```

注意事项

无

参数描述

表 7-168 参数说明

参数	类型	描述
isShareLocked	bool	是否共享锁定。

7.6.5 通话状态变更通知

7.6.5.1 呼叫来电

通知描述

收到该通知时，表示收到呼叫来电。

方法定义

```
virtual void OnCallIncoming(const HwmCallIncomingInfo *callIncomingInfo) {};
```

注意事项

无。

参数描述

表 7-169 结构体 HwmCallIncomingInfo 参数说明

参数	类型	描述
callType	HwmCallType	呼叫类型。
number	char	来电者号码
name	char	来电者姓名
callId	unsigned int	通话id

表 7-170 枚举 HwmCallType 说明

枚举值	描述
CALL_TYPE_AUDIO	音频呼叫。
CALL_TYPE_VIDEO	视频呼叫

7.6.5.2 通话状态

通知描述

收到该通知时，表示通话状态发生变更。

方法定义

```
virtual void OnCallState(const HwmCallStateInfo *callStateInfo) {};
```

注意事项

无。

参数描述

表 7-171 结构体 HwmCallStateInfo 参数说明

参数	类型	描述
state	HwmCallState	通话状态。
callId	unsigned int	通话id

表 7-172 枚举 HwmCallState 说明

枚举值	描述
HWM_CALL_STATE_INCOMING	来电状态。
HWM_CALL_STATE_HANGUP	挂断状态

7.6.5.3 通话信息

通知描述

收到该通知时，表示通话中的信息发生变更。

方法定义

```
virtual void OnCallInfo(const HwmCallInfo *callInfo) {};
```

注意事项

无

参数描述

表 7-173 结构体 HwmCallInfo 参数说明

参数	类型	描述
callType	HwmCallType	呼叫类型。
number	char[]	对方号码。
name	enum	对方名称。
startTime	char[]	呼叫开始时间。
endTime	char[]	呼叫结束时间（通话结束前为空）。
isCallOut	bool	是否为主动呼出

参数	类型	描述
callId	unsigned int	通话id

7.6.5.4 通话结束

通知描述

收到该通知时，表示通话结束，返回通话信息。

方法定义

```
virtual void OnCallRecordInfo(const HwmCallRecordInfo *callRecordInfo) {};
```

注意事项

无

参数描述

表 7-174 结构体 HwmCallRecordInfo 参数说明

参数	类型	描述
callInfo	HwmCallInfo	呼叫信息。见表1
reason	HwmCallEndReason	呼叫结束原因。

表 7-175 枚举 HwmCallEndReason 说明

枚举值	描述
HWM_CALL_END_REASON_INVALID	无。
HWM_CALL_END_REASON_CONNECTED_HANGUP_BY_CALLER	主叫挂断（接通）。
HWM_CALL_END_REASON_CONNECTED_HANGUP_BY_CALLEE	被叫挂断（接通）。
HWM_CALL_END_REASON_HANGUP_BY_CALLER	主叫挂断（未接通）。
HWM_CALL_END_REASON_HANGUP_BY_CALLEE	被叫挂断（未接通）。

枚举值	描述
HWM_CALL_END_REASON_CALLEE_BUSY	被叫通话中（未接通）
HWM_CALL_END_REASON_CALLEE_ANSWER_TIMEOUT	被叫超时未接听（未接通）。
HWM_CALL_END_REASON_CALLEE_NUMBER_ERROR	被叫号码错误（未接通）。
HWM_CALL_END_REASON_FAILED	通话失败。

7.6.6 音视频设备状态

通知描述

收到该通知时，表示会议中麦克风、扬声器或摄像头的状态信息发生变更。

方法定义

```
virtual void OnAVDeviceCtrlStateChanged(HwmDeviceType deviceType, bool isOpen) {};
```

注意事项

无

参数描述

表 7-176 参数说明

参数	是否必须	类型	描述
HwmDeviceType	是	HwmDeviceType	设备类型枚举。
isOpen	是	bool	设备是否开启。

7.7 错误码参考

7.7.1 常见错误码

表 7-177 HwmErrCode 枚举说明

变量	值	描述
HWM_COMMON_ERR_FAILED	1	通用错误。
HWM_COMMON_ERR_PARAM_ERROR	2	参数错误。
HWM_LOGIN_ERR_ACCOUNT_OR_PASSWORD_ERROR	10006	帐号或密码错误。
HWM_LOGIN_ERR_ACCOUNT_LOCKED	10007	帐号被锁定。
HWM_LOGIN_ERR_IP_OR_DEVICE_FORBIDDEN	10008	IP或设备被禁止接入。
HWM_LOGIN_ERR_ACCOUNT_OR_ENTERPRISE_NOT_ACTIVE	10009	帐号或企业未激活。
HWM_LOGIN_ERR_AUTH_INFO_TIMEOUT	10011	鉴权信息过期（是否使用UTC时间，是否经过校准）。
HWM_LOGIN_ERR_ACCOUNT_STOPED	10013	账号被停用
HWM_CONF_ERR_CONF_ID_OR_PASSWORD_ERROR	20001	会议id或密码错误。
HWM_CONF_ERR_CONF_LOCKED	20002	会议被锁定。
HWM_CONF_ERR_NOT_IN_CONF	20003	不在会议中，接口调用非法。
HWM_CONF_ERR_VMR_NOT_BEEN_HOLD	20004	VMR会议，来宾不允许激活会议
HWM_CONF_ERR_SHARE_LOCKED	20005	共享被锁定
HWM_CONF_ERR_NOT_IN_DATA_CONF	20006	不在数据会议中
HWM_CONF_ERR_CONF_ONLY_SAME_ENTERPRISE	20007	只允许企业内用户加入会议
HWM_CONF_ERR_CONF_ONLY_INVITED	20008	只允许被邀请者加入会议
HWM_CALL_ERR_THIRD_USER_ID_NO_EXIST	20009	发起呼叫的第三方账号不存在
HWM_ERR_NOT_IN_CONF_OR_CALL	20010	不在会议或呼叫中
HWM_ERR_ALREADY_EXIST_CONF_OR_CALL	20011	已存在一路会议或呼叫
HWM_CONF_ERR_PASSWORD_ERROR_LOCKED	20012	多次输入密码错误导致会议暂时被锁定

变量	值	描述
HWM_CONF_ERR_CONF_ENDED	20013	会议已结束
HWM_CONF_ERR_SAME_PWD_WITH_CHAIR	20014	来宾密码与主持人密码相同
HWM_CONF_ERR_NUM_OF_PAR_EXCEEDS_UPPER_LIMIT	20015	与会方超出专用VMR套餐的最大与会人数
HWM_CONF_ERR_CONF_NOT_ON_LOCAL_STATION	20016	会议不在本站点
HWM_CONF_ERR_ENTERPRISE_CONCURRENCY_RESOURCE_NOT_ENOUGH	115022001	企业并发资源不足
HWM_CONF_ERR_CONF_DATA_NOT_FOUND	111070005	会议不存在
HWM_CONF_ERR_VMR_DATA_NOT_FOUND	111070006	个人云会议室不存在
HWM_CONF_ERR_CONF_ROLE_AUTHENTICATION_FAILED	111070010	权限不足,无法查看会议
HWM_CONF_ERR_VMR_CONF_CONFLICT	11107106	VMR会议冲突
HWM_CONF_ERR_MODIFY_FAIL_AS_CONF_ALREADY_STARTED	111071065	正在召开会议,无法编辑会议详情
HWM_ORG_CONCURRENT_PARTICIPANTS_NUMBER_IS_ZERO	111071060	企业未配置并发会议资源,无法使用个人会议ID和随机会议ID发起会议
HWM_CONF_USER_NOT_VMR_OWNER	111071063	非云会议室拥有者
HWM_CONF_ERR_CANCEL_FAIL_AS_CONF_STARTED	111071067	会议正在召开中,不允许取消会议
HWM_CONF_ERR_START_TIME_LESS_THAN_CURRENT_TIME	111071013	开始时间早于当前时间
HWM_CONF_ERR_START_TIME_OVER_MAXIMUM_SCHEDULE_TIME	111071014	会议开始时间超出最大预订时间范围
HWM_CONF_ERR_VMR_PWD_SAME_AS_MEETING_ID	201080017	来宾密码不能和会议ID相同
HWM_CONF_ERR_PARTICIPANT_CONF_NOT_SUPPORT	111084013	不支持该会议
HWM_CONF_ERR_ATTENDEES_NUMBER_EXCEED_LIMIT	111071066	与会方数量超出上限
HWM_CONF_ERR_VMR_DATA_STATUS_ILLEGAL	111073007	VMR套餐状态非法

变量	值	描述
HWM_CONF_ERR_WAIT_APPLY_RESOURCE	115029006	等待申请资源中
HWM_CONF_ERR_SDK_SVN_DETECT_FAILURE	411000017	svn端口探测不通
HWM_CONF_ERR_RTC_NOT_SUPPORT_STG	411000019	RTC不支持STG模式
HWM_CONF_ERR_OVER_MAX_PARTICIPANT_NUMBER	111072101	超过设置的会议方数
HWM_CONF_ERR_AUDIENCE_VMR_LOGIC_RESOURCE_NOT_ENOUGH	111072202	观众数超过VMR套餐中的观众最大并发资源数
HWM_CONF_ERR_PARTICIPANT_VMR_LOGIC_RESOURCE_NOT_ENOUGH	111072203	主持人与嘉宾超过VMR套餐中的最大并发资源数
HWM_CONF_ERR_PARTICIPANT_CONF_NOT_EXIST	111081002	会场所在会议不存在或已结束
HWM_VMR_RESOURCE_UNAVAILABLE	201080000	云会议资源不可用
HWM_LOGIN_ERR_ACCOUNT_IS_UNBOUND	206010075	账号已经解绑
HWM_CONF_ERR_SERVER_ERROR	990000018	连接不上服务器
HWM_CONF_ERR_JOIN_ROOM_FAIL	990000019	无法加入会议
HWM_CONF_ERR_SERVER_BUSY	990000020	服务器繁忙
HWM_CONF_ERR_RTC_JOIN_ROOM_AUTH_FAIL	990000022	会议鉴权信息校验失败
HWM_CONF_ERR_SERVER_CONNECTION_ERROR	990000025	服务器连接异常
HWM_LOGIN_ERR_CA_VERIFY_FAILED	412100012	服务器CA证书校验失败
HWM_LOGIN_ERR_CURLE_EXTERN_CONNECT_TIMEOUT	412100017	代理通信超时, tcp建链失败, 大概率是代理配置错误
HWM_LOGIN_ERR_CURLE_EXTERN_CLOSED_BY_PROXY	412100018	链接被代理断开, 大概率是服务器地址配置错误
HWM_LOGIN_ERR_CURLE_COULDNT_RESOLVE_PROXY	412100024	proxy域名解析失败
HWM_LOGIN_ERR_CURLE_COULDNT_RESOLVE_HOST	412100025	服务器域名解析失败

变量	值	描述
HWM_AGENT_ERR_CALLBACK_TIMEOUT	90001	等待回调超时
HWM_AGENT_ERR_IN_PROCESS	90002	接口正在处理中
HWM_AGENT_ERR_NOT_LOGIN	90003	未登录，接口调用非法。
HWM_AGENT_ERR_NOT_INITED_END	90004	未初始化或初始化尚未完成
HWM_AGENT_ERR_NOT_SUPPORT_END	90005	此版本不支持
HWM_AGENT_ERR_OPERATION_INTERRUPT	90012	操作中断

7.7.2 错误码表

错误码表

表 7-178 错误码说明

错误码	错误描述
-1	Unknown error.:未知错误
0	Success.:成功
111070001	Request error.:请求错误
111070002	Insufficient permissions.:权限不足
111070003	System processing failure.:系统处理失败
111070004	Meeting authentication failed.:校验失败
111070005	Meeting does not exist.:会议不存在
111070006	Personal cloud meeting room does not exist.:个人云会议室不存在
111070007	Meeting access number does not exist.:会议接入号不存在
111070008	Meeting notification could not be sent.:会议通知模板配置错误
111070009	Meeting password cannot be left blank.:会议密码不能为空
111070010	Insufficient permissions to view this meeting.:权限不足,无法查看会议
111070100	Get started meeting exception.:获取召开会议异常
111070101	Start meeting exception.:开始会议异常
111070102	Apply IVR resource exception.:申请IVR资源异常

错误码	错误描述
111070103	Start IVR meeting exception.:开始IVR会议异常
111070104	Apply site resource exception.:申请会场资源异常
111070105	Start sub meeting exception.:开始子会议异常
111070106	Add channel exception.:创建级联通道异常
111070107	Move and apply site resource exception.:移动会场申请资源异常
111070108	Start meeting exception.:开启会议异常
111070109	Get started cloud meeting exception.:获取召开云会议室会议异常
111070110	Invite site exception.:邀请会场异常
111070111	Request token is null.:请求token为空
111070112	Rename site exception.:重命名会场异常
111070113	Hangup site exception.:会场挂断异常
111070114	Mass hangup site exception.:会场批量挂断异常
111070115	Delete attendee exception.:删除与会者异常
111070116	Mass delete attendee exception.:批量删除与会者异常
111070117	Book cycle meeting exception.:预定周期会议异常
111070118	Modify cycle meeting exception.:修改周期会议异常
111070119	Create cycle meeting exception.:创建周期会议异常
111070120	Region data not found.:区域信息获取失败
111070121	Forbidden subscribe.:禁止订阅
111070122	Cancel subscribe fail.:取消订阅失败
111071001	Sysportal service exception.:Sysportal服务运行异常
111071002	Conf service exception.:Conf服务运行异常
111071003	Failed to create the meeting. Try again later.:创建会议失败, 请稍后重试
111071004	This cloud meeting room does not allow guests to join before the host.:VMR会议不允许来宾在主席入会前入会
111071005	This cloud meeting room has been scheduled for another meeting during this period.:VMR会议时间冲突, 该时间段此会议室已有预约会议
111071006	Insufficient cloud meeting time.:VMR会议时长不足

错误码	错误描述
111071007	Meeting does not exist.:VMR预约关系不存在
111071008	Cannot query meeting list.:查询会议列表失败
111071009	Cannot change cloud meeting into immediate meeting.:不支持修改VMR会议为立即会议
111071010	Cannot modify cloud meeting flag.:不支持修改云会议室标识
111071011	The cloud meeting has not yet started. Please wait.:VMR会议未到准备时间点
111071012	Not cloud meeting.:非VMR会议
111071013	Start time less than current time.:会议开始时间不能早于系统当前时间
111071014	Start time over max schedule time.:会议开始时间超出最大预订时间范围
111071015	Duration over range.:时长超出范围
111071016	Cycle ID list null exception.:取消周期会议ID列表不能为空
111071017	Meeting size over range.:与会方数超出范围
111071018	Invalid media type.:媒体类型非法
111071019	Invalid meeting type.:会议类型非法
111071020	Cycle params null exception.:周期会议时,必须携带周期会议参数
111071021	Duration over range.:会议时长超出范围
111071022	Subject length too long.:会议主题超长
111071023	Group URI length too long.:groupuri超长
111071024	Invalid email address format.:邮箱地址不合法
111071025	Invalid mobile number format.:手机号不合法
111071026	Department name too long.:部门名称过长
111071027	Invalid welcome voice enable value.:入会欢迎词开关不在范围内
111071028	Invalid enter voice enable value.:入会提示音开关不在范围内
111071029	Invalid leave voice enable value.:离会提示音开关不在范围内
111071030	Invalid reminders value.:会议提醒非法
111071031	Invalid allow record value.:会议是否启用网络录制取值非法
111071032	Invalid auto record value.:会议是否自动启动录制取值非法

错误码	错误描述
111071033	Invalid user type value.:会预定者的用户类型非法
111071034	Invalid encrypt mode value.:加密类型非法
111071035	Invalid language value.:语言非法
111071036	Invalid auto prolong value.:是否自动延长会议取值非法
111071037	VmrID too long.:vmrid超长
111071038	Invalid record type value.:录播类型非法
111071039	Live address too long.:主流直播地址超长
111071040	Aux address too long.:辅流直播地址超长
111071041	Null start date for cycle meeting.:周期会议时,起始日期为空
111071042	Null end date for cycle meeting.:周期会议时,结束日期为空
111071043	Invalid cycle type value for cycle meeting.:周期会议时,周期类型非法
111071044	Invalid interval for cycle meeting.:周期会议时,interval非法
111071045	Null point for cycle meeting.:周期会议时,point为空
111071046	Invalid point value for cycle meeting.:周期会议时,point非法
111071047	Invalid vmr flag value.:vmrFlag非法
111071048	Invalid record auxStream.:recordAuxStream非法
111071049	Invalid multiStream flag.:multiStreamFlag非法
111071050	Invalid audience flag.:audienceFlag非法
111071051	Invalid liveChannel ID.:liveChannelID不存在
111071052	Anonymous login is locked. Try again later.:匿名登录锁定,请稍后重试
111071053	Link login is locked. Try again later.:链接登录锁定,请稍后重试
111071054	Not enough Meeting ID.:会议ID不足
111071055	Failed to apply meeting ID.:会议ID申请失败
111071056	Failed to release meeting ID.:会议ID释放失败
111071057	Live room is used.:直播房间已被占用
111071058	The meeting has not yet started. Please wait.:当前会议还未召开,请耐心等待
111071059	Meeting size over range.:与会方数超出上限

错误码	错误描述
111071060	The enterprise cannot create meetings because it has not purchased any concurrent meeting resources.:企业未配置并发会议资源,无法使用个人会议ID和随机会议ID发起会议
111071061	Parameter verify failed.:参数校验失败
111071062	Json conversion failed.:JSON转换失败
111071063	Not cloud meeting room owner.:非云会议室拥有者
111071064	This cloud meeting room has been started for another meeting during this period.:VMR会议冲突,当前已有会议正在召开
111071065	This meeting cannot be modified because it is already started.:正在召开会议,无法编辑会议详情
111071066	Attendees amount over range.:与会方数量超出上限
111071067	This meeting cannot be canceled because it is already started.:会议正在召开中,不允许取消会议
111071068	Meeting password encrypt failed.:会议密码加密失败
111071069	No meeting during this period.:选定的时间中没有会议
111071070	Sub cycle meeting cannot modify live room.:周期会议子会议不允许修改直播房间
111071071	Stop online meeting exception.:结束会议失败
111071072	Meeting is ended.:会议已经结束
111071073	Not allow anonymous user.:会议不允许匿名用户呼入
111071074	Only allow user in organization.:会议只允许企业内用户呼入
111071075	Only allow invited user.:会议只允许被邀请用户呼入
111071076	Batch export task for history meeting list during this period.:当前有正在执行的批量导出任务
111071077	Only allow invited user and user in organization.:会议只允许被邀请或者企业内用户呼入
111071078	Max current participants amount reached.:企业单会议并发方数已达上限
111071079	Invalid CTD param.:ctd会议参数不合法
111071080	Invalid guest password.:来宾密码不合法
111071081	Guest password conflict with meeting ID.:来宾密码与会议ID冲突
111071082	Max meeting amount reached.:会议数达到上限

错误码	错误描述
111071083	Invalid meeting random number.:无效的会议随机数
111071084	Max export history meeting list times reached.:导出历史会议次数超出允许的值
111071085	Max multi view layouts reached.:会议多画面布局超出限制
111071086	Max multi view invalid manual type.:会议多画面模式非法
111071087	Max multi view sub view amount reached.:会议多画面子画面个数超出限制
111071088	Max multi view sub view numbers reached.:会议多画面子画面会场数超出限制
111071089	Webinar and common meeting cannot be converted to each other.:网络研讨会不能修改为普通会议,普通会议也不能修改为网络研讨会
111071090	Not allow add audience.:不支持添加观众
111071091	Meeting password authentication failed.:密码错误
111071092	File export task failed.:导出文件任务失败
111071093	File export task is downloading.:导出文件正在处理中
111071094	Not allow audience to join.:会议不允许观众入会
111071095	Need to get WEBINAR_CONF_DTO object.:需要获取WEBINAR_CONF_DTO对象
111071096	User-defined languages over range.:自定义语言个数超出范围
111071097	Your enterprise does not support simultaneous interpretation.:企业不支持同声传译
111071098	Need to set interpreter groups.:需要设置传译组信息
111071099	Invalid interpretation language.:传译组语言非法
111071100	Max languages in a meeting are reached.:单会议最大支持16种语言
111071101	This interpreter group ID already exists.:传译组的组号重复
111071102	Max interpreters in the interpreter group reached.:传译组中传译员的个数达到上限
111071103	This interpreter already exists in an interpreter group.:传译员重复
111071104	Invalid interpreter account.:传译员帐号不存在
111071105	Guest password conflict with chairman password.:来宾密码与主席密码冲突

错误码	错误描述
111071106	Meeting is holding.:云会议室已有会议正在召开
111071107	Invalid audience password.:观众密码不合法
111071108	Audience password conflict with meeting id.:观众密码与会议ID冲突
111071109	Audience password conflict with chairman password.:观众密码与主席密码冲突
111071110	Only RTC meeting support E2E encrypt.:非RTC会议不支持端到端加密
111071111	Audience password conflict with guest password.:观众密码与来宾密码冲突
111071112	Cloud meeting room ID type cannot be modified.:不支持修改会议ID类型
111071113	Cloud meeting room type cannot be modified.:不支持修改VMR类型
111071114	Cloud meeting room flag cannot be modified.:不支持修改VMR标识
111071119	Sub cycle meeting modify time over range.:周期会议修改时间超出范围
111072001	Application for meeting resources failed.:申请资源失败
111072002	Failed to release meeting resources.:释放资源失败
111072003	The meeting is ending.:会议正在结束
111072004	The meeting is starting.:会议正在开始
111072005	The meeting has not yet started.:会议还未开始
111072006	The meeting has ended.:会议已经结束
111072007	Site invitation failed.:邀请会场失败
111072008	Site does not exist.:会场不存在
111072009	Failed to disconnect the site.:挂断会场失败
111072010	Coming soon...:功能暂未支持,敬请期待...
111072011	Meeting status incorrect.:会议状态非法
111072012	Site control operation failed.:会场会控操作失败
111072013	The cloud meeting room is in use by another meeting.:当前VMR会议非目标会议
111072014	The meeting has ended.:VMR会议已过期
111072015	The meeting has not yet started.:VMR会议还未开始

错误码	错误描述
111072016	Meeting does not exist.:VMR预约记录不存在
111072017	Meeting has no attendee.:会议与会者不存在
111072018	Application for chairman password exception.:申请主席密码错误
111072019	Not found site which can be moved exception.:没有找到可以移动的会场
111072020	The current meeting is not a recording meeting. Recording is not supported.:非录播会议不支持录制
111072021	System processing failure.:录播调度失败
111072022	Insufficient recording resources in the system.:录播资源不足
111072023	There are no participants in this meeting.:会议中没有与会人
111072024	Operation failed. No sites have joined this meeting.:请确保会议中已有会场入会
111072025	Voice meeting does not support this function.:语音会议不支持该功能
111072026	The voice site does not support this operation.:语音会场不支持该操作
111072027	Voice sites cannot be displayed in custom continuous presence.:手动设置的多画面中无法显示语音会场
111072028	At least one site must have custom continuous presence configured.:手动设置的多画面中需最少设置一个会场
111072029	At most one presentation pane is allowed in custom continuous presence.:手工设置的多画面最多只允许一个辅流子画面
111072030	In the voice-activated continuous presence mode, each pane can only support a single user.:声控多画面模式下,每个子画面最多只允许一个用户
111072031	The number used for the site to join the meeting cannot be left blank.:与会者的入会号码不允许为空
111072032	The number used for the site to join the meeting is too long.:与会者的入会号码长度超过系统允许的最大值
111072033	The format of the number used for joining the meeting is incorrect. (Correct format: User name name/IP address:Port number or Country code plus phone number):与会者的入会号码格式非法(支持SIP、TEL号码格式)
111072034	Length of site name too long.:与会者的名称长度超过系统允许的最大值

错误码	错误描述
111072035	The site role type is outside the allowed range.:与会者的角色类型不在系统允许的取值范围内
111072036	Length of site email address too long.:与会者的邮箱地址长度超过系统允许的最大值
111072037	Length of site mobile number too long.:与会者的手机号码长度超过系统允许的最大值
111072038	The value for whether a site is muted after it first joins the meeting is outside the allowed range.:与会者入会后是否静音的取值不在系统允许的取值范围内
111072039	The value for whether to automatically invite a site is outside the allowed range.:与会者是否自动邀请的取值不在系统允许的取值范围内
111072040	The host's site viewing type is outside the allowed range.:主席选看类型取值不在系统允许的取值范围内
111072041	The value of the continuous presence setting mode is outside the allowed range.:设置多画面方式取值不在系统允许的取值范围内
111072042	The polling time is outside the allowed range.:轮询显示时间取值不在系统允许的取值范围内
111072043	The pane type cannot be left blank.:画面类型不允许为空
111072044	Invalid pane type.:画面类型非法
111072045	The value of the switch is outside the allowed range.:开关项取值不在系统允许的取值范围内
111072046	The site name cannot be left blank.:与会者的名称不允许为空
111072047	Site name too long.:与会者的名称超过系统允许的最大值
111072048	Continuous presence is not supported. Ask the administrator to configure related data on the MCU.:不支持多画面,请联系管理员开启MCU配置
111072049	Response timed out. Try again later.:等待响应超时
111072050	Meeting locked.:会议被锁定
111072051	The meeting display mode is empty.:会议显示模式为空
111072052	The meeting display mode is outside the allowed range.:会议显示模式不在系统允许的取值范围内
111072053	The meeting pane type is outside the allowed range.:会议画面类型不在系统允许的取值范围内
111072054	The banner or caption type is empty.:横幅或字幕类型为空

错误码	错误描述
111072055	The banner or caption content is empty.:横幅或字幕的内容为空
111072056	The banner or caption content cannot exceed 2,000 bytes.:横幅或字幕的内容长度超过2000字节
111072057	The request message body is empty.:请求消息体为空
111072058	The pane index is greater than the total number of panes.:子画面索引大于画面总数
111072059	Meeting has no host.:会议中没有主席
111072060	Regular participants cannot perform this operation.:当前普通与会方类型不支持该操作
111072061	You have set host polling and cannot select a specific site to view or set continuous presence.:前主席正在轮询观看会场,不能单独选看其他会场或会议多画面
111072062	You cannot select a site to view on the web page. Perform the operation on the terminal.:当前主席与会方类型不支持该操作
111072063	Applying for recording resources...:录制资源重新申请错误码
111072064	Too many failed login attempts. Try again later.:会控登录锁定
111072065	The meeting ID or password is incorrect. Check the ID and password and try again.:会议不存在或密码错误
111072066	The enterprise has not purchased any recording storage space.:企业未购买录播空间
111072067	Insufficient enterprise recording storage space.:企业录播空间已满
111072068	Failed to start live broadcast. Try again later.:启动直播失败
111072069	Not allow to hand up for host.:不允许指定会议主席举手发言
111072070	The operation is not allowed for guest.:来宾不允许进行该操作
111072071	The operation is only allowed for yourself.:仅能对自己进行操作
111072072	Not allow to apply host for attendees who do not have meeting control abilites.:没有会控能力的与会者不允许设置为主席
111072073	Notify message is expired.:推送消息已过期
111072074	Only participants with a number registered on HUAWEI CLOUD can be set as the host.:云下会场不能成为主席

错误码	错误描述
111072075	Hung up site for meeting ended.:会议结束挂断会场
111072076	Hung up site by host.:主席挂断会场
111072077	Hung up site because host deleted site.:主席删除会议挂断会场
111072078	Hung up site because meeting is recreated.:重建会议挂断会场
111072079	Hung up site for moving site failed.:移动会场失败挂断会场
111072080	Hung up site for inviting site cross region failed.:跨region邀请会场失败挂断会场
111072081	Hung up site for invitation exception.:邀请会场失败挂断会场
111072082	Hung up site for joining meeting repeatedly.:会场重复入会挂断会场
111072083	Lock sharing failed.:锁定共享失败
111072084	This site does not support 1-screen layout.:云下会场禁止设置在多画面一画面中
111072085	Not support realtime subtitle.:会议不支持实时字幕
111072086	Realtime subtitle pid is null.:会议实时字幕指定发言人列表为空
111072087	Invalid realtime subtitle pid.:会议实时字幕指定发言人ID有误或已离会
111072088	Failed to set realtime subtitle.:会议实时字幕设置失败
111072089	Not allow audience to change speaker status.:不允许观众解闭音
111072090	Allow/forbid audience to speak only for audience.:主持人允许/禁止观众说话接口只能操作观众
111072091	This meeting does not support simultaneous interpretation.:会议不支持同声传译
111072092	Interpreter group is empty.:会议当前传译组为空,请先配置传译组
111072093	This meeting has not enable simultaneous interpretation.:会议未开启同声传译
111072094	This site is not interpreter.:当前会场非传译员
111072095	This site has not yet confirmed interpreter identity.:传译员未确认身份
111072096	Invalid site language.:会场选择的语言非法

错误码	错误描述
111072097	This site is confirmed interpreter.:当前会场是已确认的传译员
111072098	This site listen channel is different from speak channel.:普通会场听说频道不一致
111072099	Failed to set interpretation languages. Try again.:设置会议语言频道失败
111072100	Failed to set interpretation channel. Try again.:设置会场语言频道失败
111072101	Max participant amount reached.:超过设置的会议方数
111072102	Max allowed handup amount reached.:观众超过最大允许举手数
111072103	Max allowed speaking amount reached.:观众超过最大允许发言数
111072104	Failed to set virtual audio CSD site.:设置级联通道虚拟语言会场失败
111072105	This meeting does not support cohost.:该会议未启用联席主持人
111072106	This site does not support cohost.:该会场不支持联席主持人
111072107	Host cannot be set to cohost.:不支持设置主持人为联席主持人
111072108	Audience cannot be set to cohost.:不支持设置观众为联席主持人
111072109	Need to check cohost permission.:需要去主会议所在region判断是否为联席主持人
111072110	Cohost cannot operate host.:联席主持人不能操作主持人
111072111	This site does not support share invitation.:当前会场不支持邀请共享能力
111072201	Hung up site because heart beat failed.:与终端心跳中断导致挂断会场
111072202	Insufficient audience cloud meeting room logic resources.:观众数超过VMR套餐中的观众最大并发资源数
111072203	Insufficient participant cloud meeting room logic resources.:主持人与嘉宾超过VMR套餐中的最大并发资源数
111072204	Insufficient webinar resources, so started failed.:网络研讨会套餐已经有在线会议,被占用,激活失败
111072205	Only RTC conf support local record.:非RTC会议不支持本地录制
111072206	Invalid E2E encryption parameter.:端到端加密参数非法

错误码	错误描述
111072207	Not allow to switch role in webinar.:网络研讨会不允许进行角色转换
111072208	The client version used by the user is too early. Cannot start local recording.:客户端不支持本地录制
111072211	Meeting is under AR assist.:会议已经在进行AR协作
111072213	Audience cannot be set to host.:观众不能设置为主持人
111072214	Rename param contains sensitive words.:改名涉及敏感词错误
111072222	Indicates record start exception.:启动云录制失败, 请联系管理员
111072223	Indicates record paused exception.:暂停云录制失败, 请联系管理员
111072224	Indicates record stop exception.:停止云录制失败, 请联系管理员
111072225	Indicates live start exception.:启动直播失败, 请联系管理员
111072226	Indicates live stop exception.:停止直播失败, 请联系管理员
111072504	Max participants in a single meeting reached.:达到系统单个会议人数上限
111072507	waiting room user is leaved.:等候室准入用户已经离会
111072508	waiting room user is joined.:等候室准入用户已经准入
111073001	User does not exist.:用户不存在
111073002	Department does not exist.:组织不存在
111073003	Invalid username format.:用户非法
111073004	Invalid department name format.:组织非法
111073005	Could not query department.:查询组织失败
111073006	Application for anonymous user failed.:申请匿名用户失败
111073007	Invalid cloud meeting room package status.:VMR套餐状态非法
111073008	Failed to Release anonymous user.:释放匿名用户失败
111073009	Not support cycle meeting for welinkC.:WeLinkC 不支持预约周期会议
111073010	Not support cloud meeting room for welinkC.:WeLinkC 不支持预约VMR会议
111073011	Failed to get Welink token.:获取WeLink Proxy Token失败

错误码	错误描述
111073012	Failed to Get M to M token.:获取机机帐号Token失败
111073013	Invalid webinar cloud meeting package status.:网络研讨会VMR套餐状态非法
111073014	Max extend times for cloud meeting trial version reached.:会议延长次数超出VMR试用套餐可延长的最大次数
111073015	Max extend duration for cloud meeting trial version reached.:会议延长时间超出VMR试用套餐单次会议最大时长
111073016	Not support auto extend meeting for cloud meeting trial version.:VMR试用套餐不支持自动延长
111074001	This meeting host already exists.:IDO主席已经存在
111074002	No meeting host permissions.:无IDO主席权限
111074003	Failed to extend the meeting. Max meeting duration reached.:延长会议失败,会议已达到最大时长
111074004	Failed to extend the meeting. The service processing is abnormal.:延长会议失败,服务处理异常
111075001	Failed to start timer.:启动定时器失败
111075002	Failed to end timer.:停止定时器失败
111075003	Json parse failed.:json转换异常
111076001	Released by host.:会场被终端主席挂断
111076002	Released by portal host.:会场被portal主席挂断
111076003	Max number of participants reached.:超出VMR最大方数
111076004	Insufficient enterprise resources.:企业并发数不足
111076005	Not answered.:会场无响应
111076006	User does not exist.:会场不存在
111076007	Offline.:会场离线
111076008	Busy.:忙
111076009	Rejected by user.:拒接
111076010	Released by user.:会场挂断
111076011	Insufficient meeting resources in the system.:媒体资源不足
111076012	Outbound calling disabled.:PSTN服务未开通
111076013	Application for address token failed.:申请地址本鉴权信息失败
111076014	Call failed.:SIP呼叫超时

错误码	错误描述
111076015	Max number of participants reached.:超过设置的会议方数
111076017	Invalid user.:用户失效
111076018	Invalid enterprise.:企业失效
111076019	Invalid SP.:SP失效
111076022	Not allow joining by scanning for hard device.:企业配置来宾不允许扫码邀请硬终端入会*
111076028	Not allow conf control.:会议正在进行AR协作，禁止共享/分组会议/释放主持人等会控操作*
111076029	Not allow AR Assist.:会议正在进行共享/分组会议，禁止进行AR协作*
111076099	Common error.:通常失败
111077001	RSE server info does not exist.:RSE录播相关错误码
111077101	Not support AI record.:会议不支持AI会议纪要
111077102	Reapplying for AI record resource.:正在重新申请会议纪要资源
111077103	Insufficient AI record resource.:AI会议纪要资源不足
111077104	In AI recording, the meeting cannot stop live or recording.:会议处于AI会议纪要状态,不允许停止录制/直播
111077201	Record file does not exist or have no permission.:录播文件已删除或您无权观看,请确认后重试
111077202	RSE authentication failed.:RSE机鉴权失败
111078000	No available region in multi region.:没有可用Region
111078001	No available global in multi region.:没有可用Global
111078002	Dispatch region error in multi region.:多region 分发region错误
111079000	Failed to phrase big data param query xml.:大参数查询xml解析失败
111079001	Failed to phrase anonymous assign xml.:匿名帐号分配xml解析失败
111080001	Bad request.:请求消息参数错误
111080002	Insufficient permissions.:用户无权限进行该操作
111080004	Site data not found or deleted.:会场不存在或已被删除
111081001	Site is inviting.:会场正在通话中

错误码	错误描述
111081002	Site meeting does not exist or is ended.:会场所在会议不存在或已结束
111081003	Failed to apply site resource.:申请会场资源失败
111081004	Failed to release site resource.:释放会场资源失败
111081005	Failed to apply site media resource.:申请会场媒体资源失败
111081006	Authentication failed.:认证失败
111081007	Meeting is locked.:会议已经被锁定
111081009	Login is locked because max authentication times reached.:会议鉴权失败次数已达上限被锁定
111081010	Waiting user restrictions.:等候用户限制
111082001	Not support this control.:不支持该会控
111082002	No response.:无响应
111082003	Site not exist.:会场不存在
111082004	Site unavailable.:会场离线
111082005	Site busy.:会场正忙
111082006	Site refused.:会场拒接
111082007	Site leave.:会场挂断
111082008	PSTN service is closed.:PSTN服务关闭
111082009	SIP invitation timeout.:SIP呼叫超时
111083001	RSE does not exist.:RSE设备不存在
111084001	Failed to execute DTMF.:执行DTMF失败
111084002	Get started meeting exception.:启动会议异常
111084003	Call MCU side exception.:呼叫MCU端异常
111084004	Join call exception.:加入呼叫异常
111084005	Invite exception.:邀请异常
111084006	Call terminal side exception.:呼叫端侧异常
111084007	Answer exception.:应答异常
111084008	Common CTL exception.:通用CTL异常
111084009	Hangup exception.:挂断异常
111084010	Notify capability exception.:通知能力异常
111084011	Update MT exception.:更新MT异常

错误码	错误描述
111084012	Negotiate exception.:协商异常
111084013	Not support this meeting.:不支持该会议
111084014	Not allowed join meeting in RTC inviting because guests are not invited.:RTC外邀入会,来宾不在被邀请名单中,拒绝入会
111089002	Failed to phrase URI join xml.:链接入会XML解析失败
111090001	Failed to join data meeting.:数据入会失败
111090002	Join data meeting overtime.:数据入会超时
111100001	Failed to get slider image.:获取滑块图片失败
111100002	Slider check failed.:滑块校验失败
111100003	Invalid random number.:无效的随机数
111100004	Invalid slider token.:无效滑块token
111100005	Need to check slider.:需要滑块校验
111100006	Invalid SMS phone number.:非法的手机号
111100007	Failed to send verification code.:发送验证码失败
111100008	Verification code expired.:验证码已过期
111100009	Verification code error.:验证码错误
111100010	Max verification code check times reached.:验证码错误超过最大次数
111100011	The uuid is invalid, you need to re-verify the phone number.:uuid无效,需要重新验证手机号
111100012	The meeting is not on this site.:会议不在本站点
111100013	Need to verify the mobile phone number. Overseas access to China, free enterprises need to verify the mobile phone verification code.:需要验证手机号,海外接入国内,免费企业需要进行手机验证码校验
111110001	Failed to add IM group.:添加群组失败
111110002	Failed to disband IM group.:删除群组失败
111110003	Failed to add IM member.:添加成员失败
111110004	Failed to remove IM member.:删除成员失败
111111007	hangup for rejoin mmr failed.:mmr重入会失败挂断
111112001	Failed to enter the waiting room.:入等候室失败
111112002	Failed to exit the waiting room.:出等候室失败

错误码	错误描述
111112003	Waiting room cache operation failed.:等候室缓存操作失败
111112004	Co-hosts cannot be moved into the waiting room.:联席主持人不能被移入等候室
111112005	Not allowed to enable waiting room.:不允许启用等候室
111112008	Not support this feature according to client capabilities.:客户端能力不支持此特性
111113001	can not find application.:找不到应用信息
111113002	application exist.:应用已存在，目前一个会议只能关联一种应用
111113003	not required to create application.:无权限创建应用
111113004	need verify id.:应用操作需要实名认证（用户需要绑定手机）
111113005	get token failed.:获取应用用户token失败
111113006	url illegal.:应用服务URL地址非法
111113101	shimo doc has not used.:石墨文档应用未启用
111113102	call shimo doc server failed.:调用石墨文档服务失败
111113103	get shimo token failed.:获取石墨应用级token失败
111113104	not in conf,has no right to apply.:用户不在会议中，无权限申请权限
111500301	breakout conf surpass limit.:分组会议子会议超过上限
111500302	breakout conf name error.:分组会议子会议名称错误
111500303	breakout Start failed.:分组会议启动失败
111500304	breakout not exist when delete:分组会议中没有子会议（删除）
111500305	breakout not exist when start.:分组会议中没有子会议（启动）
111500306	breakout is closing by host:分组会议结束中(主持人停止)
111500307	breakout is closing by host:分组会议已经结束或未启动（加入时）
111500308	Attendee in target conf:移动用户在目标分组
111500309	Chair in current conf:主持人在当前分组中（请求帮助）
111500310	not support breakout conf:不支持分组会议（启动时 启动接口）
111500311	breakout conf starting or started:分组会议已经启动或启动中（重复启动）

错误码	错误描述
111500312	subconf not exist:子会议不存在
111500313	not support in mainconf:主会中用户不支持此操作（请求帮助）
111500314	breakout conf state illegal:分组会议状态异常（缓存误删）
111500315	action forbidden by status:分组会议静态、动态操作由于状态原因，不支持当前操作
111500316	apply chair failed due to host in subconf:主持人在分组中，不能申请为主持人
111500317	apply sub conf id failed:申请分组的会议id失败
111500318	query main conf failed:获取主会议信息失败
111500319	attendee not found:没有找到用户信息
111500320	sub conf stop failed:分组会议停止失败
111502000	subject faile and leave conf start:订阅失败需要离会的错误码起始值
111502099	subject faile and leave conf end:订阅失败需要离会的错误码结束值
112010001	Common failure.:通用失败
112010002	Null param.:参数为空
112010003	Participant not found.:未找到参与者
112010004	Meeting not found.:会议不存在
112010005	SIP invite information timeout.:invite消息超时
112010006	SIP invite information failed.:invite消息失败
112010007	SIP information timeout.:info响应超时
112010008	SIP information failed.:info响应失败
112010009	System processing failed.:系统处理失败
112010010	System processing failed.:系统处理失败
112010011	Failed to invite MRS.:呼叫mrs失败
112010012	Failed to invite participant.:呼叫终端失败
112010013	Data meeting join information failed.:数据入会Info失败
112010014	Failed to start meeting.:召开会议失败
112010015	Media negotiation failed.:媒体协商失败
112010016	User rejects.:用户拒绝

错误码	错误描述
112010017	User is busy.:用户忙
112010018	No call permission.:没有呼叫权限
112010019	Unknown callee.:未知被叫人
112010020	No user response.:没有用户响应
112010021	Join meeting timed out.:加入会议超时
112010022	Session has been released.:会话已释放
112010023	Callee is not registered.:被叫人未注册
112010024	SIP invite timed out.:SIP邀请超时
112010025	User Session timer times out and hangs up.:用户Session timer 超时挂断
112010026	The participant bye hangs up on the MRS side.:与会者腿MRS侧bye挂断
112010027	The participant bye hangs up on terminal side.:与会者腿终端侧bye挂断
112010028	No RTP stream hang up.:无码流挂断
112100000	Sysportal exception.:系统异常
112100001	Bad request.:请求异常
112100002	Illegal request.:非法请求
112100003	Data not found.:数据未获取到
112100004	Data duplication.:数据重复
112100006	The requested type is not supported.:请求的类型不支持
112100007	Failed to save notification template.:保存通知模板失败
112100008	Failed to save data.:保存数据失败
112100009	Failed to query data.:查询数据失败
112100010	Failed to delete data.:删除数据失败
112100011	Password reset required.:需要重置密码
112100012	Incorrect old password.:旧密码错误
112100013	The new password cannot be the same as the old password.:新密码不能与旧密码相同
112100014	Insufficient password complexity and low security.:密码复杂度不够,安全性低

错误码	错误描述
112100015	The length does not meet the requirements.:长度不符合要求
112100016	RSE is in meeting.:RSE在会议中
112100017	RSE to be deleted.:RSE待删除
112100018	Failed to query the meeting list.:查询会议列表失败
112100019	User has been locked out. Try again later.:用户已被锁定,请稍后重新登录
112100022	The live room channel name exists.:频道名称已存在
112100023	Failed to modify system configuration items for duplicated meeting ID prefix.:修改系统配置项失败,会议ID前缀重复
112100024	CONFID prefix can only be added, not modified or deleted.:CONFID前缀只能增加,不能修改和删除
112100025	The live room channel url exists.:直播房间地址已存在
112100026	The live room has been occupied.:直播房间已被使用
112100027	The live room does not exist.:直播房间不存在
112100028	Live room release failed.:直播房间释放失败
112100029	The request is missing header or param.:请求缺失头域
112100030	The Live main stream URL already exists.:音视频推流地址已存在
112100031	The Live auxiliary stream URL already exists.:辅流演示推流地址已存在
112100032	Password is weak.:密码为弱密码
112100033	The live room is occupied.:直播房间被占用
112100034	Save part of attendees since max attendees size reached.:因为与会者数量超出上限只保存部分
112100035	Failed to export data.:导出数据失败
112100036	Failed to import data.:导入数据失败
112100037	Max file size reached.:文件大小超出上限
112100038	LVS target address exists.:lvs的目标地址已存在
112100039	LVS port existed.:lvs端口重复
112100040	Not allow to modify global standby config.:Global只有主机可以修改配置;
112100041	MCU insulation IP address exists.:mcu隔离的IP已经存在

错误码	错误描述
112100042	MCU insulation IP address cannot be modified.:mcu隔离的IP不能修改
112100043	SBC address exists.:SBC配置的地址已经存在
112100044	Internal exception.:内部错误异常
112100045	System config delete exception.:系统配置不允许被删除
112100046	No ISBC access address.:企业id无法获取关联SBC地址
112100047	Media type need to be HD because the organization enable Full HD capability.:企业已开启全高清功能,媒体设置不能低于1080P/8M
112100061	Same interpreter language abbreviation exists.:语言缩写重复
112100062	Same interpreter language name exists.:语言名称重复
112100063	MCU type is MIXED and insulation config is not configured, or MCU meeting exists when modifying mcu connection settings.:修改MCU连接配置告警,以下两种场景会导致:1.MCU存在会议 2.修改类型为MIXED未配置隔离
112100064	MCU type is MIXED and insulation config is not configured when adding mcu connection settings.:新增MCU连接配置告警,MCU类型为MIXED且未配置隔离
112100065	MCU type is MIXED when modifying MCU insulation config or disbanding enterprise.:修改MCU隔离配置时,解除企业绑定时MCU类型为MIXED
112100066	MCU type is MIXED when deleting MCU insulation config.:删除MCU隔离配置时,MCU类型为MIXED
112100067	This gateway type is incompatible with MCU.:MCU与网关类型不兼容
115021001	Insufficient media resource.:媒体资源不足
115021002	Meeting does not exist.:会议不存在
115021003	Media resource does not exist.:媒体资源不存在
115021005	Insufficient media bandwidth resource.:带宽资源不足
115021006	Media device fault.:设备故障
115022000	Insufficient common logic resource.:通用的逻辑资源不足
115022001	Insufficient meeting resources in the enterprise.:企业会议并发方数不足
115022002	Insufficient logic port resource.:逻辑资源端口资源不满足

错误码	错误描述
115022003	Live broadcast resources have not been purchased or are insufficient.:企业直播推流方数不足
115022004	Media device does not exist.:媒体设备不存在
115022005	The cloud recording ports purchased by enterprises have been used up.:企业购买的云录制端口已经用完
115023001	Insufficient ID resources.:ID资源不足
115024001	RMS service exception.:RMS服务异常
115025001	The live room has already been occupied.:直播房间已经被使用
115025002	The live room does not exist.:直播房间不存在
115025003	Live room release failed.:直播房间释放失败
115026001	MCU meeting exists.:MCU有会议
115029001	Invalid common parameter.:通用参数值不合法
115029002	Null parameter.:参数值为空
115029003	service is not available.:服务不可用
115029004	Configuration exception.:配置异常
115029005	Duplicately apply resources.:重复申请资源
115029006	Waiting for applying resources.:等待申请资源中
115029999	Unknown error.:未知错误
118000000	User authentication failed.:用户鉴权失败
118000001	User token authentication failed.:usg token 鉴权失败
118000002	User authentication account lockout.:usg鉴权帐号锁定
200000001	Server is busy. Please try again later.:服务器正忙 请稍后再试
200000002	Internal error. Please contact SRE.:服务器正忙 请联系SRE
200000003	Invalid parameter. Please check whether the parameters are valid.:请求参数错误 请检查参数是否合法
200000004	Illegal request. Please check the value of the request header field X-Auth-Token or X-Access-Token.:非法请求 请检查请求头域X-Auth-Token或X-Access-Token的值
200000005	Forbidden access during the version upgrade.:系统正在升级, 暂时不可访问。如对您造成不便, 十分抱歉
201020023	This user is already the owner of another corporation and cannot create a new corporation for it.:该用户已经是其他企业的所有者,无法为其创建新的企业

错误码	错误描述
201040005	Max number of users reached. Please check the purchased meeting users.:用户总数超过限制, 请检查购买的会议用户
201040016	Password security is low. Please set another password.:密码安全性低, 请设置其他密码
201040031	Activation code overdue.:激活码过期
201040042	Pre-authentication expired.:预验证失效
201040043	This user is the owner of another corporation and cannot be invited.:该用户是其他企业的所有者,暂无法邀请
201040044	Invalid invite connection.:邀请连接非法
201040045	User is already in waiting list.:用户已经在等待列表
201040046	User already exists.:用户已存在
201040047	Account modification is not allowed.:不允许修改账户
201050012	Activation code does not exist.:激活码不存在
201050015	Invalid activation code.:激活码错误
201050021	Activation failed.:激活失败
201050022	Terminal does not allow unbinding.:终端不允许解绑
201080000	VMR resource unavailable.:云会议资源不可用
201080006	The host password and the guest password cannot be the same. Please confirm the input parameters.:主席密码和来宾密码不能相同 请确认输入参数
201080017	Same as meeting password.:和会议密码相同
201510001	No invite permission to get an invite code.:没有邀请权限去获取邀请码
201510003	Invalid application.:申请失效
206010000	User name or password error.:用户名或密码错误
206010007	Account is not activated, no permission to log in.:帐号未激活,无权限登录
206010010	Failed to get verification code, Please contact SRE.:获取验证码失败 请联系SRE
206010011	Verification code must be filled.:验证码不能为空
206010012	Verification code check failed.:验证码校验失败
206010013	Jump login failed. Please contact SRE.:跳转登录失败 请联系SRE

错误码	错误描述
206010022	SSO login failed. Please contact SRE.:SSO登录失败 请联系 SRE
206010023	SSO login is not supported.:不支持sso登录
206010024	Invalid authentication type.:认证类型非法
206010025	APPID verification failed.:APPID校验失败
206010027	Authentication type error.:校验类型错误
206010029	Network error.:网络异常
206010030	APPID does not allow automatic registration.:APPID不允许自动注册
206010031	APPID auto registration failed.:APPID自动注册失败
206010036	Normal users cannot log in.:普通用户无法登录
206010042	Huawei account authentication failed.:华为账户认证失败
206010057	APPID authentication expires.:APPID认证过期
206010058	APPID account is not bound.:APPID账户未绑定
206010064	Site mismatch.:站点不匹配
206010065	Phone number conflict.:电话冲突
206010066	E-mail conflict.:邮箱冲突
206010070	The number of users has reached the maximum.:用户数超过上限
206010071	WeChat authentication failed.:微信鉴权失败
206010074	International sites support site changing re-login.:国际站点支持切站点重登
206010075	Account has been unbound.:账号已经解绑
206020000	Invalid authentication information. Please check the value of the request header field X-Auth-Token or X-Access-Token.:鉴权信息无效 请检查请求头域X-Auth-Token或X-Access-Token的值
206020001	Token is created by disaster recovery site.:token是容灾站点创建
206020002	Token is invalid due to being kicked.:token由于被踢失效
206030000	The user does not exist.:系统中不存在该用户
206030001	The verification code is incorrect or has expired. Please obtain a new one.:验证码错误或失效, 请重新获取
206030002	Incorrect verification code.:验证码错误

错误码	错误描述
206030003	Failed to send the verification code.:发送验证码失败
206030004	Try to get a new verification code 180s later.:请在180秒后重新发送验证码
206030005	The password cannot be changed.:系统不支持修改密码
206030006	The old password is incorrect.:旧密码错误
206030007	The length of the new password does not meet the requirements.:新密码长度不符合要求
206030008	The password complexity does not meet the requirements.:密码复杂度不满足要求
206030009	The new password must be different from the previous old passwords.:设置的密码不能与最近旧密码相同
206030010	The password cannot contain three or more of the same characters consecutively.:密码不能包含3个以上重复字符
206030011	The password cannot be changed within 5 minutes upon the last password change.:修改密码后, 5分钟内不能更新密码
206030012	The password cannot contain the account or the reverse order of the account.:密码不能与帐号或其逆序帐号相同
206030013	At least two characters must be different between the new and old passwords.:新密码相较于旧密码至少要有两个不同的字符
206030014	Invalid password format.:密码格式不正确
206030015	The user is locked and will be locked in 60s.:用户已被锁定, 请60秒后重试
206030016	Have no right to reset other's password.:没有权限重置密码
206030018	The number of password retrieval times in a day has reached the maximum(10 times):一天内忘记密码次数达到最大值 (10次)
206030019	The number of incorrect verification code input has reached the maximum, Please get it again.:验证码输入错误次数超过最大值, 请重新获取
206030023	Failed to send verification code. Please contact SRE.:发送验证码失败 请联系SRE
206030028	The system has enabled SSO login and does not support changing the password. Please retry after the system has disabled SSO login.:系统已开启SSO登录,不支持修改密码 系统关闭SSO登录后重试
206030031	Forgot password for Huawei account.:华为帐号忘记密码

错误码	错误描述
206040000	No corresponding terminal type.:无对应的终端类型
206060008	Sending verification code is locked.:发送校验码锁定
206060009	Slider verification required.:需要滑块校验
206060010	Invalid slider token.:滑块token非法
206060011	Request parameter error.:请求参数错误
206060012	Sending SMS is not supported.:不支持发送短信
206060013	Repeated sending is not supported.:不支持重复发送
206060014	Invalid verification code.:验证码非法
206060015	Max verification codes check times reached.:验证码次数超过最大次数
206060017	Failed to bind Huawei account.:华为账户绑定失败
211030046	Not the ideahub activation code.:不是ideahub激活码
211030050	auth fail too many times, locked.:身份验证失败次数太多,已锁定
211030052	auth fail too many times, permanently locked.:身份验证失败次数太多,已永久锁定
211060003	The maximum times of log uploads has been reached.:日志上传次数达到最大限制
411000001	Common failure.:失败
411000002	Parameter error.:参数错误
411000003	Meeting does not exist.:会议不存在
411000004	A meeting already exists.:已经有会议存在
411000005	Meeting state error.:会议状态异常
411000006	Failed to get device.:获取设备异常
411000007	Failed to set device.:设置设备异常
411000008	No device available.:无可设备
411000009	This operation is only supported on mobile.:只有移动端支持该操作
411000010	Network access error.:网络接入错误
411000011	Other supplementary service is running.:正在进行其他补充业务
411000012	Memory allocation error.:分配内存错误

错误码	错误描述
411000013	Initialization failed.:初始化失败
411000014	User does not exist.:用户不存在
411000015	Did not join the data meeting.:未加入数据会议
411000016	Neither meeting nor call exists.:会议和呼叫均不存在
411000017	SVN port detection failed.:svn端口探测不通
411000018	Failed to write file.:写文件失败
411000019	RTC does not support STG mode.:RTC不支持STG模式
411000020	Failed to load RTC library. RTC meeting is not supported.:rtc库加载失败，不支持rtc会议
411000021	The meeting is in the audience pause state when changing roles.:转角色的时候会议正处于观众暂停观看状态
411000022	Not in call.:不在呼叫中
411000023	Meeting or call does not exist.:会议或者呼叫不存在
411000024	Pairing state not allowed.:配对状态不允许
411000025	Pairing request failed.:配对请求失败
411000026	Pairing exceeds limit.:配对超出限制
411000027	Pairing common error.:配对通用错误
411000028	Pairing and joining the meeting failed.:配对入会失败
411000029	Pairing and joining the meeting is busy.:配对入会忙
411000030	Pairing and joining the meeting is invalid.:配对入会无效
411000031	Multiple clicks at the same time to pair and join the meeting error.:同时多次点击入会错误
411000032	Pairing is not established or is being established.:配对未建立或正在建立中
411000033	The camera is not turned on in the third-party capture mode.:第三方采集模式下摄像头未打开
411000034	RTC engine creation failed.:RTC 引擎创建失败
411000035	Not in a meeting.:不在会议中
411000036	mute self failed.:静音自己失败并且采用了保护
411000100	End the meeting with no stream due to disconnection of network.:无码流结束会议网络断开
411000101	End the meeting with no stream because fail to re-join meeting.:无码流结束会议重入会失败

错误码	错误描述
411000102	Null pointer exception.:空指针异常
411000103	Parameters cannot be empty.:参数不许为空
411000104	Indicates encrypt and decript func not match.:加解密算法不匹配
411000105	Indicates Loading User-Defined DLLs failed .:加载用户自定义dll失败
411000106	Indicates Applition not support shimo docs .:企业配置不支持打开石墨文档
411000107	Indicates notification of render fail .:数据会议UI渲染失败
411000108	End the meeting with mmr media rejoin failed.:mmr重入会失败
412100001	The message processing thread is not initialized.:消息处理线程未初始化
412100002	Parameter error.:参数错误
412100003	Timeout.:超时
412100004	Memory allocation error.:分配内存错误
412100005	XML parsing error.:XML解析出错
412100006	DNS resolution error.:DNS解析异常
412100007	Request message failed.:请求消息异常
412100008	Authentication failed.:鉴权失败
412100009	Server error.:服务器异常
412100010	Account is locked.:帐号被锁定
412100011	Create timer error.:创建定时器错误
412100012	Server CA certificate verification failed.:服务器CA证书校验失败
412100013	Network error.:网络异常
412100014	Server redirect.:服务器返回重定向地址
412100015	Account is not activated.:帐号未激活
412100016	JSON parsing error.:JSON解析失败
412100017	The proxy communication times out and the TCP link building fails. Probably because the proxy configuration is incorrect.:代理通信超时，tcp建链失败，大概率是代理配置错误

错误码	错误描述
412100018	Disconnected by the proxy. Probably because the server address is misconfigured.:接被代理断开，大概率是服务器地址配置错误
412100019	The TCP link establishment timed out. Probably because the proxy domain name or port configuration is incorrect.:tcp建链超时，大概率是proxy域名或者端口配置错误
412100020	The authentication process failed. The user does not have proxy permission and needs to apply for proxy permission.:认证鉴权过程失败，用户没有proxy权限，需要申请proxy权限
412100021	The authentication process failed. The proxy user name or password is incorrect.:认证鉴权过程失败，代理用户名或者密码错误
412100022	The authentication process failed. Incorrect authentication type. User name and password are required. The user needs to be prompted for input.:认证鉴权过程失败，错误的认证类型，同时需要用户名、密码，需要提示用户输入
412100023	The authentication process failed. Other unknown errors. It may be that the authentication method is not supported. The user needs to be prompted to try again.:认证鉴权过程失败，其他未知错误，有可能是不支持认证方式，建议弹框提示用户重新发起
412100024	The authentication process failed. Unable to resolve gateway.:认证鉴权过程失败，无法解析网关
412100025	The authentication process failed. Unable to resolve host.:认证鉴权过程失败，无法解析主机
412100026	The authentication process failed. Unable to connect.:认证鉴权过程失败，无法连接
412100027	The authentication process failed. Operation times out.:认证鉴权过程失败，操作超时
412100028	Message length too long.:消息长度太大
412100029	Base64 encryption failed.:base64加密失败
412100030	XML parsing error.:XML解析出错
412100031	Secure function error.:安全函数失败
412100032	Firewall detection parameter error.:防火墙检测参数错误
412100033	Failed to create STG tunnel.:创建STG隧道失败
412100034	Failed to update STG service.:更新STG服务失败
412100035	Encryption failed.:加密失败
412100036	Do not allow duplicate downloads.:不允许重复下载

错误码	错误描述
412100037	File operation failed.:文件操作失败
412100038	The request message returns empty.:请求消息返回空
412100039	The number of detection servers is 0 error.:探测服务器数为0错误
412100040	Failed to get local IP address.:获取本地IP失败
412100041	XXX.:登录模块消息未初始化
412100042	http return 429 error.:http 返回429错误
412100043	Not logged in, not available:此业务未登录不可用
412200001	Internal error.:内部错误
412200002	Parameter error.:参数错误
412200003	Network timeout.:网络超时
412200004	Network Error.:网络错误
412200005	Need to bind mobile phone.:需要绑定手机
412200006	Need to bind email.:需要绑定邮箱
412200007	Has initialized already.:重复初始化
412200008	State machine execution failed.:状态机执行失败
412200009	Failed to start timer.:开启定时器失败
412200010	The user is prompted to change the password when logging in for the first time.:首次登录, 提示用户修改密码
412200011	Unknown authentication type.:未知鉴权类型
413000001	Common error.:一般错误
413000002	Parameter error.:参数错误
413000003	Memory allocation error.:分配内存错误
413000004	System error.:系统错误
413000005	Message sending error.:发送消息错误
413000006	Get system configuration error.:获取系统配置错误
413000007	Physical network error.:物理网络错误
413000008	Network access error.:网络接入错误
413000009	Create timer error.:创建定时器错误
413000010	Call state error.:呼叫状态错误
413000011	Other operation in progress error.:正在进行其他操作错误

错误码	错误描述
413000012	Requesting the master to start call error.:请求主控进行呼叫开始错误
413000013	Requesting audio resource error.:申请音频资源错误
413000014	Other supplementary service is running.:正在进行其他补充业务
413000015	Record call error.:记录通话记录错误
413000016	Media process error.:媒体进程返回的错误
413000017	Maximum number of calls reached.:超过最大呼叫路数
413000018	SIP account ID does not exist.:SIP账户ID不存在
413000019	Call ID does not exist.:呼叫ID不存在
413000020	Failed due to registering.:正在注册中导致失败
413000021	Registration failed.:注册失败
413000022	Deregistration failed.:注销失败
413000023	Setting account information error.:设置账户信息错误
413000024	Setting SIP account error.:设置SIP帐号失败
413000025	Reporting SIP account information error.:上报SIP帐号信息失败
413000026	Server information error.:服务器信息错误
413000027	SIP account information error.:账户信息错误
413000028	SIPC execution error.:SIPC执行错误
413000029	Not registered error.:未注册错误
413000030	Call Sip interface to subscribe error.:调用Sip接口订阅失败
413000031	Canceling subscription error.:注销订阅中错误
413000032	Subscribing error.:订阅中错误
413000033	Only one local meeting is allowed.:只允许一个本地会议
413000034	Local meeting is not created.:本地会议未创建
413000035	The participant line does not match the chair account.:与会者线路与主席账户不匹配
413000036	Local meeting state error.:本地会议状态错误
413000037	Failed to report the linkage state of the IP Phone.:IP Phone端联动状态上报失败
413000038	Meeting ID does not exist.:会议ID不存在

错误码	错误描述
413000039	Video hold error.:视频保持失败
413000040	Video unhold error.:视频恢复失败
413000041	The maximum number of server meeting reached.:服务器会议个数超出
413000042	Get AA random number error.:获取AA随机数失败
413000043	AA login error.:AA登陆失败
413000044	Meeting type mismatch.:会议类型不匹配
413000045	Failed to start video meeting.:视频会议开启失败
413000046	AA no master server.:AA无主服务器
413000047	Password error.:密码错误
413000048	Username error.:用户名错误
413000049	User is logged in.:用户已登录
413000050	Account is locked.:账户已锁定
413000051	Terminal type mismatch.:终端类型不匹配
413000052	Parsing XML error.:解析XML错误
413000053	Connecting to server error.:连接服务器错误
413000054	Get media configuration error.:获取媒体配置失败
413000055	Get service permission error.:获取业务权限失败
413000056	Insufficient service permission.:业务权限不足失败
413000057	Network environment error.:网络环境错误
413000058	Service conflict.:业务冲突
413000059	Connection timeout.:连接超时
413000060	Unknown error.:未知错误
413000061	Add ringing information error.:添加振铃信息失败
413000062	Delete ringing information error.:删除加振铃信息失败
413000063	Create ringing number error.:创建振铃号码失败
413000064	VVM parameter error.:VVM参数错误
413000065	Getting registered service error.:获取登记业务错误
413000066	Getting voicemail error.:获取语音邮箱错误
413000067	Attendee already exists.:与会者已经存在

错误码	错误描述
413000068	Attendee does not exist.:与会者不存在
413000069	Create server video meeting window error.:创建服务器视频会议窗口失败
413000070	Video window already exists.:视频窗口已存在
413000071	Getting meeting list error.:获取会议列表失败
413000072	Operating requires host privileges.:需要主持人权限才能操作
413000073	No video device to operate.:没有视频设备可以操作
413000074	Refresh registration is not turned off.:没有关闭刷新注册
413000075	Online state report error.:在线状态上报失败
413000076	Network address subscription report error.:网络地址本订阅 notify 上报失败
413000077	Telepresence account was force unregistered. Information report error.:智真帐号被踢,info 上报失败
413000078	TLS root certificate error.:TLS 根证书错误
413000079	AD authentication error.:AD 鉴权失败
413000080	Meeting list is getting.:会议列表正在获取中
413000081	Prohibited calls.:禁止呼叫
413000082	The maximum number of calls has been reached.:呼叫数达上限
413000083	The maximum number of encrypted calls has been reached.:加密呼叫数达上限
413000084	The maximum number of video calls has been reached and needs to downgrade to audio.:超出视频呼叫数,需要降为音频
413000085	H323 account ID does not exist.:H323 账户 ID 不存在
413000086	H323 account information error.:H323 账户信息错误
413000087	H323 account information report failed.:H323 帐号信息上报失败
413000088	Insufficient local resources.:本端资源不足
413000089	The calling protocol is not supported.:不支持该呼叫协议
413000090	Setting H323 account information error.:设置 H323 账户信息错误
413000091	Mediatrix meeting access number reporting failed.:mediatrix 会议接入号上报界面失败

错误码	错误描述
413000092	Mediav VMR information reporting failed.:上报Mediav VMR信息到界面失败
413000093	IMS meeting list information reporting failed.:上报获取IMS会议列表信息到界面失败
413000094	TUP initialization is not complete.:tup初始化未完成
413000095	Message return value error.:消息返回值错误
413000096	Message content too large error.:消息内容太大错误
413000097	Secure function error.:安全函数返回错误
413000098	message response is empty error.:消息响应为空错误
413000099	Account is not enabled.:帐号未启用
413000100	Create call control implement error.:创建呼叫控制类错误
413000101	Already in call.:已经在呼叫中
413000102	Call does not exist.:呼叫不存在
413000103	Failed to get media configuration information.:获取媒体配置信息失败
413000104	Failed to set third-party capture. Calling HME_V_Capture_InputData_MCU failed.:设置第三方采集失败, 调用HME_V_Capture_InputData_MCU返回失败
414100001	Common error.:一般错误
414100002	Parameter error.:参数错误
414100003	Timeout.:超时
414100004	Memory allocation error.:分配内存错误
414100005	DNS resolution error.:DNS解析异常
414100006	Request message failed.:请求消息异常
414100007	Authentication failed.:鉴权失败
414100008	Server error.:服务器异常
414100009	Create timer error.:创建定时器错误
414100010	Start timer error.:开启定时器错误
414100011	The meeting control block is full. Please release the control block that is not in use.:会控控制块已满, 请释放没在使用的控制块

错误码	错误描述
414100012	The proxy communication times out and the TCP link building fails. Probably because the proxy configuration is incorrect.:代理通信超时，tcp建链失败，大概率是代理配置错误
414100013	Disconnected by the proxy. Probably because the server address is misconfigured.:接被代理断开，大概率是服务器地址配置错误
414100014	The TCP link establishment timed out. Probably because the proxy domain name or port configuration is incorrect.:tcp建链超时，大概率是proxy域名或者端口配置错误
414100015	The authentication process failed. The user does not have proxy permission and needs to apply for proxy permission.:认证鉴权过程失败，用户没有proxy权限，需要申请proxy权限
414100016	The authentication process failed. The proxy user name or password is incorrect.:认证鉴权过程失败，代理用户名或者密码错误
414100017	The authentication process failed. Incorrect authentication type. User name and password are required. The user needs to be prompted for input.:认证鉴权过程失败，错误的认证类型，同时需要用户名、密码，需要提示用户输入
414100018	The authentication process failed. Other unknown errors. It may be that the authentication method is not supported. The user needs to be prompted to try again.:认证鉴权过程失败，其他未知错误，有可能是不支持认证方式，建议弹框提示用户重新发起
414100019	Curl proxy exception.:curl代理异常
414100020	Curl host exception.:curl主机异常
414100021	Curl connection exception.:curl连接异常
414100022	Curl operation timed out.:curl操作超时
414100023	Message length too long.:会控消息内容过大
414100024	Failed to create thread.:创建线程失败
414100025	Secure function error.:安全函数返回错误
414100026	JSON parsing failed.:解析json体失败
414100027	WS token is empty.:ws token值为空
414100028	Return JSON data error.:返回json数据异常
414100029	Failed to create meeting control block.:创建会议控制块失败
414100030	Failed to get meeting control block.:获取会议控制块失败

错误码	错误描述
414100031	Failed to get meeting control block by CallId.:通过CallId获取会议控制块失败
414100032	Destroying the meeting control block failed.:销毁会议控制块失败
414100033	Temporary creating meeting information not found.:找不到创会临时信息
414100034	Failed to get participant ID.:获取参会者ID失败
414100035	Failed to get MT by participant ID.:通过ParticipantId获取MT失败
414100036	Failed to get self information by participant ID.:通过ParticipantId获取selfinfo失败
414100037	Failed to get participant information by MT.:通过MT获取会议成员信息失败
414100038	Manual invitation required.:需要手动邀请
414100039	The major parameters of the meeting have been reported.:已上报过会议大参数
414100040	Subtitles are not supported.:不支持字幕
414100041	Meeting Hang Up - Timeout.:会议挂断 - 超时
414100042	Meeting Hang Up - No Streaming.:会议挂断 - 无码流
414100043	Meeting hung up - Hung up by the host.:会议挂断 - 被主持人挂断
414100044	Meeting Hang Up - Meeting has ended.:会议挂断 - 会议已结束
414100045	End of meeting due to leave meeting.:离会导致的会议结束
414100046	Message queue processing function registration failed.:消息队列处理函数注册失败
414100047	Message queue processing function deregistration failed.:消息队列处理函数去注册失败
414100048	Meeting ended due to reject meeting.:拒绝会议导致的会议结束
414100049	Meeting ended by self.:自己结束会议导致的会议结束
414100050	The meeting control module is not initialized.:会控模块未初始化
414100051	The watching special site reaches the maximum.:选看的特殊会场超过限制

错误码	错误描述
414100052	The watching HD site reaches the maximum.:选看的高清会场超过限制
414100053	The watching SD site reaches the maximum.:选看的标清会场超过限制
414100054	The watching specification reaches the maximum.:选看的规格超过限制
414100055	Watching internal errors.:选看内部错误
414100056	Duplicated watch render.:选看的句柄重复
414100057	HTTP request error.:HTTP请求错误
414100058	Websocket request timeout.:websocket请求超时
414100059	Loading failed.:加载错误
414100061	The meeting ends due to the timeout of guest waiting before meeting or the timeout of the meeting control heartbeat.:会前等待页面时间超时或者会控心跳超时导致的会议结束
414100062	The meeting ends due to being canceled.:会议被取消预约导致的会议结束
414100063	The meeting ends due to network error when guest waiting before meeting.:会前等待页面网络异常导致的会议结束
414100064	http return 429 error.:http 返回429错误
414100065	self is chairman.:已是主持人
414100066	self is cohost.:已是联席主持人
414100067	no cohost and chairman.:无主持人和联席主持人
414100068	no supported cohost and chairman.:无支持的主持人和联席主持人
414100069	Invalid user, enterprise or SP.:用户, 企业或SP失效
415000001	DB is not initialized.:db未初始化
415000002	Common failure.:失败
415000003	Proxy detection has not been completed.:代理探测还没有完成
416000001	Invalid parameter.:参数错误
416000002	Track module is not initialized.:打点模块未初始化
416000003	Track event does not exist.:打点事件不存在
416000004	Not logged in.:未登录

错误码	错误描述
416000005	Upload is not enabled.:未开启上传
419100001	Invalid parameter.:错误参数
419100002	Incorrect URI.:错误号码
419100003	Memory allocation failed.:内存分配失败
419100004	SDP error.:SDP错误
419100005	Connection does not exist.:连接不存在
419100006	Connection failure.:连接失效
419100007	Operation forbidden.:禁止操作
419100008	State error.:状态错误
419100009	No corresponding line data.:无对应线路数据
419100010	No corresponding registered instance.:无对应注册实例
419100011	No corresponding call instance.:无对应呼叫实例
419100012	No corresponding subscription instance.:无对应订阅实例
419100013	No corresponding Refer instance.:无对应Refer实例
419100014	No corresponding DIM instance.:无对应DIM实例
419100015	Failed to create header.:创建头域失败
419100016	Failed to parse header.:解析头域失败
419100017	Header not found.:没有找到头域
419100018	No SDP message body.:没有sdp消息体
419100019	Failed to get capability.:获取能力失败
419100020	Failed to compare capability.:比较能力失败
419100021	Failed to open logical channel.:打开逻辑通道失败
419100022	Encoding SDP failed.:编码SDP失败
419100023	Decoding SDP failed.:解码SDP失败
419100024	Negotiate to unsupported SDP.:协商到不支持的SDP
419100025	TPTD initialization failed.:TPTD初始化失败
419100026	TPT initialization failed.:TPT初始化失败
419100027	TXN initialization failed.:TXN初始化失败
419100028	UA initialization failed.:UA初始化失败
419100029	GLUE initialization failed.:GLUE初始化失败

错误码	错误描述
419100030	HLLM initialization failed.:HLLM初始化失败
419100031	VCOM initialization failed.:VCOM初始化失败
419100032	SDP initialization failed.:SDP初始化失败
419100033	Multiple instance error.:多实例错误
419100034	Insufficient registration control blocks.:注册控制块不足
419100035	Insufficient subscription control blocks.:订阅控制块不足
419100036	Insufficient call control blocks.:通话控制块不足
419100037	Insufficient IM control blocks.:对话外控制块不足
419100038	Failed to parse URI xml to string.:解析xml的uri失败
419100039	Encryption negotiation failed.:加密协商失败
419100040	Local capabilities are not configured.:没有设置本地能力
419100041	BFCP token error.:bfcpl令牌错误
419100042	DNS resolution error.:dns解析错误
419100043	Failed to get subscription state.:获取订阅状态失败
419100044	IPv4 is not supported.:不支持IPv4
419100045	IPv6 is not supported.:不支持IPv6
419100046	Unsupported host type.:不支持的主机类型
419100047	Failed to get timer.:获取定时器失败
420000001	Cannot receive sharing invitations.:无法接收共享邀请
420000002	ReportDTO parameter verification failed.:reportDTO参数校验失败
420000003	Report folder creation failed.:举报文件夹创建失败
420000004	Report folder compression failure.:举报文件夹压缩失败
420000005	The path of the ZIP file to be uploaded is empty.:待上传的ZIP文件路径为空
420000006	The size of the ZIP file to be uploaded reaches the maximum.:待上传的ZIP文件大小超过限制
420000007	The same person reported more than 100 times within 24 hours.:同一个人24小时内举报次数超过100次
430100001	Common Error.:通用错误
430100002	Invalid parameters.:参数错误
430100003	SIP connection check failed.:SIP连接检查失败

错误码	错误描述
430100004	Timeout.:超时
430100005	In process.:接口正在处理中
430100006	Not initialized or initialization is not complete.:未初始化或初始化尚未完成。
430100007	Not supported in this release.:此版本不支持。
430100008	Operation interrupt.:操作中断。
430100009	Contact not found.:联系人不存在
430100010	Args length overrun.:参数长度超限
430100011	APPID info illegal.:APPID 信息入参不合法
430100012	E2EEncrypt algorithm error.:端到端加密算法不对称
430200001	unlogin.:未登录
430200002	Repeated login.:重复登录
430200003	SIP connection timeout.:SIP连接超时
430200004	Invalid login password.:用于登录的密码参数不合法
430200005	Password or account empty.:账号或密码不能都为空
430300001	The calling number, account number or third-party account number cannot be empty.:呼叫号码、账号或者第三方账号不能为空
430300002	Call account empty.:点对点呼叫账号为空
430300003	Call nickname empty.:点对点呼叫名字为空
430400001	Password required to join the meeting.:加入会议需要密码
430400002	Time zone error.:时区错误
430400003	The meeting duration is less than 1.:会议时长小于1
430400004	Joining the conference is interrupted.:入会被中断
430400005	Password input at least 4 digits.:密码输入至少4位
430400006	User voluntarily cancels.:用户主动取消
430400007	The guest password illegal.:来宾密码非法
430400008	Not in conference or in call.:不在会议或呼叫中
430400009	Already exist conf or call.:已存在一路会议或呼叫
430400010	Confld empty.:会议id不能为空
430400011	Subject too long.:会议主题长度不能超过50

错误码	错误描述
430400012	VMR host password length illegal.:主持人密码长度不合法
430400013	VMR name empty.:VMR名字为空
430400014	Conf starttime early.:会议的开始时间早于当前时间
430400015	Share is locked.:共享被锁定
430400016	Not in data conference.:不在数据会议中
430400017	Role not support operation:角色不支持该操作
430500001	HTTP request, server exception.:HTTP请求，服务器异常
430500002	HTTP request, Connecting to the server times out.:HTTP请求，连接服务器超时
430500003	network unavailable.:网络不可用
430500004	file type illegal.:文件类型不合法
510000403	Server forbidden.:服务器拒绝
510000404	Not found.:未发现
510000480	Temporary unavailable.:临时失效
510000486	Busy here.:这里忙
510000487	Request terminated.:请求终止
510000488	Request is not acceptable here.:这里请求不可接受
510000603	Decline.:丢弃
510000841	Hangup without toast.:挂断不弹框
510000842	Conference is locked.:会议被锁定
910000001	Server error.:服务器异常
910000002	SFU service failure.:sfu服务故障
910000003	Service unavaliable 503.:服务不可达503
910000004	Internal error.:内部错误
910000005	Kicked off.:被踢
910000006	Signature expired.:签名过期
910000007	Reconnect timeout.:重连超时
910000008	Network detection.:网络检测，UI不需要关注该错误码，不对外体现
910000009	User is removed.:用户移除
910000010	The room is dismissed.:房间解散

错误码	错误描述
990000001	Internal error.:内部错误
990000002	Message too large.:消息长度太大
990000003	Insufficient memory.:内存不足
990000004	Failed to send message synchronously.:同步发送消息失败
990000005	Parameter error.:参数错误
990000006	Call ID order error.:呼叫ID时序错误
990000007	Failed to set local screen.:设置本端画面失败
990000008	Failed to open remote screen.:开启远端画面失败
990000009	Device configuration failed.:设备配置失败
990000010	Initializing.:初始化中
990000011	Deinitializing.:去初始化中
990000012	Uploading log.:正在日志上传中
990000013	Failed to get media port.:获取媒体端口失败
990000014	The maximum number view window reached .:选看窗口超过规格数
990000015	Media negotiation failed.:媒体协商失败
990000016	Server no response.:服务器没有响应
990000017	Role switch failed.:角色切换失败
990000018	Failed to join room.:加入房间失败
990000019	Failed to join room. Status is busy.:非空闲状态
990000020	Failed to join room. Server error.:加入房间失败，服务器异常
990000021	Failed to join room. Service unavaliable.:加入房间失败，服务不可达
990000022	Failed to join room. Authentication failed.:加入房间失败，鉴权失败
990000023	Failed to join room. Authentication retry.:加入房间失败，鉴权重试
990000024	Failed to join room. Clock synchronization.:加入房间失败，时钟同步
990000025	Failed to join room. URL error.:加入房间失败，url错误
990000026	Kicked off.:被踢
990000027	Share failed.:共享失败

错误码	错误描述
990000028	Currently in external media output mode, this operation is forbidden.:当前为外部媒体输出模式，禁用该操作
990000029	Reconnection failed.:重连失败
990000030	Server break down.:服务器宕机
990000031	Signature has expired.:签名已过期
990000032	Failed to set remote window mode.:设置远端窗口模式失败
990000033	Failed to subscribe or unsubscribe audio.:订阅或取消订阅音频失败
990000034	Failed to connect to other room.:连接其他房间失败
990000035	Failed to disconnect other room.:断开连接其他房间失败
990000036	Role switching is not allowed.:不允许角色切换
990000037	Currently in third-party capture mode, this operation is forbidden.:当前为第三方采集模式，禁用该操作
990000038	Failed to set third-party audio capture.:设置第三方音频采集失败
990000039	Failed to set third-party video capture.:设置第三方视频采集失败
990000040	Failed to set share sound switch.:设置共享声音开关失败
990000041	Failed to start or stop upstream audio stream.:启停上行音频流失败
990000042	Failed to start or stop upstream video stream.:启停上行视频流失败
990000043	User is removed.:用户被移除
990000044	The room is dismissed.:房间被解散
990000045	Failed to set remote View.:设置远端View失败
990000052	Failed to join room because room full.:房间已满，加入失败

7.8 版本更新内容

7.8.1 80.13.6 更新内容

- 1.问题修复，体验更稳定

7.8.2 80.13.5 更新内容

- 1.新增设置语言代码接口SetLanguageWithCode，详见[表2](#)

7.8.3 80.12.7 更新内容

- 1.问题修复，体验更稳定

7.8.4 80.12.6 更新内容

- 1.会议相关功能配置开关接口EnableFeature新增录制确认开关枚举值 HWM_ENABLE_FEATURE_CONFIRM_RECORD，详见[表2](#)

7.8.5 80.12.5 更新内容

- 1.新增上传头像接口UploadSelfAvatar，详见[UploadSelfAvatar](#)
- 2.新增会议是否共享锁定状态变化通知OnConfIsShareLockedChanged，详见[共享锁定状态变化通知](#)
- 3.企业配置通知OnCorpConfig中HwmCorpConfigInfo新增字段 是否支持通讯录显示 enableShowAddressBookStructure，详见[表1](#)
- 4.新增会议结束原因错误码，达到系统单个会议人数上限 111072504 HWM_CONF_END_REASON_PARTICIPANT_REACH_MAX_NUM，详见[表3](#)

7.8.6 80.10.5 更新内容

- 1.修改云会议室信息ChangeVmrlInfoParam和云会议室信息HwmVmrlInfo新增字段 joinConfRestrictionType允许入会范围限制、isOpenWaitingRoom等候室开启开关，详见

[表2](#)和[ChangeVmrlInfo](#)

- 2.新增设置共享权限接口SetShareScreenPermission，详见[SetShareScreenPermission](#)

7.8.7 80.9.5 更新内容

- 1.取消周期子会议CancelSubCycleConf中字段subConfUUID改成subConfID，详见[表1](#)
- 2.预约周期会议BookCycleConf和编辑周期会议ModifyCycleConf、周期会议详情中 cycleCount字段废弃，详见[表2](#)
- 3.会议相关功能配置开关接口EnableFeature新增分组讨论功能开关枚举值 HWM_ENABLE_FEATURE_BREAKOUT_CONF，详见[表2](#)
- 4.会议信息HwmConfInfo新增字段是否在讨论组中isInBreakoutSubConf，详见[表1](#)

7.8.8 80.8.5 更新内容

- 1.错误码细化，以及新增错误码10011鉴权信息过期，详见[常见错误码](#)
- 2.OnConfInfo接口参数HwmConfInfo新增字段isCreator，是否是会议预订者，详见[会议信息](#)
- 3.会议详情HwmConfDetail和编辑会议HwmEditConfParam新增字段isLiveOn，详见[获取会议详情](#)和[编辑会议](#)

表 7-179 新旧版本错误码对应关系

80.7.5 版本及之前的错误码	80.8.5 版本之后的错误码	新版本错误描述
10006	10006	用户名或密码错误
	206010025	APPID校验失败
	206030000	系统中不存在该用户
20001	20001	会议id或密码错误
	111070005	会议不存在
	111070006	个人云会议室不存在
	111070010	权限不足,无法查看会议
	111081002	会场所在会议不存在或已结束
20015	20015	与会方超出专用VMR套餐的最大与会人数
	111072101	超过设置的会议方数

7.8.9 80.7.5 更新内容

- 1.新增显示设置界面接口ShowSettingWindow，接口详见[显示设置窗口](#)
- 2.Config接口新增字段moveall_to_waitingroom，支持全员移入等候室可隐藏，详见[界面配置](#)
- 3.Config接口新增字段participant_detail，支持与会者头像点击注入，详见[界面配置](#)，点击注入OnClickInjectBtn通知里面按钮类型里面增加HWM_CLICK_INJECT_BTN_PARTICIPANT_DETAIL 与会者详情按钮，data信息是HwmConfAttendee，详见[注入点击](#)
- 4.HwmSettingWindowConfig 添加 HwmScreenSettingTab 屏幕设置项，将双屏设置从 HwmNormalSettingTab 移到 HwmScreenSettingTab，接口详见[自定义设置界面](#)
- 5.会中信息通知（ OnConfInfo ）中HwmConfInfo参数新增字段isOtherCorpConf bool 是否是外部企业的会议，详见[会议信息](#)
- 6.企业信息变更通知（ OnUserCorpInfo ）中HwmUserCorpInfoNotifyType参数新增枚举值HWM_USER_CORP_INFO_NOTIFY_TYPE_USER_NAME_CHANGE 用户名变更和HWM_USER_CORP_INFO_NOTIFY_TYPE_USER_STATUS_CHANGE 用户状态变化，详见[企业信息变更](#)
- 7.会中邀人（ AddAttendee ）接口HwmConfAttendee参数新增字段accountId华为云会议帐号，接口详见[添加与会者](#)
- 8.新增错误码，详见[常见错误码](#)
- 9.增加会议结束原因错误码，与会方超出专用VMR套餐的最大与会人数HWM_CONF_END_REASON_NUM_OF_PAR_EXCEEDS_UPPER_LIMIT，接口详见[会议状态](#)

10.新增通话结束通知OnCallRecordInfo，通知详见[通话结束](#)

7.8.10 80.5.11 更新内容

- 1.Config接口新增字段statusBar，支持录制状态栏按钮可隐藏，详见[界面配置](#)
- 2.新增禁用提示音接口DisableVoicePrompts，接口详见[禁用提示音](#)

7.8.11 80.5.5 更新内容

- 1.会议结束原因HwmConfEndReason新增：
HWM_CONF_END_REASON_SERVER_ERROR 服务器异常，详见[HwmConfEndReason说明](#)

7.8.12 70.16.5 更新内容

- 1.新增周期会议相关接口，接口详见[预约周期会议](#)，[编辑周期会议](#)，[取消周期会议](#)，[编辑周期子会议](#)，[取消周期子会议](#)
- 2.加入会议接口HwmJoinConfByIdInfo参数新增字段：isStopConflictConf强制结束和当前VMR会议资源冲突的会议，接口详见[加入会议](#)
- 3.Config接口配置自定义按钮，支持增加二级菜单，接口详见[界面配置](#)
- 4.新增错误码，详见[常见错误码](#)
- 5.Config接口菜单按钮新增 共享画面质量策略：清晰度优先、流畅度优先 "share_strategy"、共享时隐藏视频窗口"share_hide_video"、共享时包含电脑声音 "share_audio"配置可隐藏。复制会议信息"copy_conf_info"按钮支持配置可注入，详见[界面配置](#)和[HwmClickInjectBtn说明](#)
- 6.会议角色HwmConfRole新增枚举值HWM_CONF_ROLE_AUDIENCE观众，HWM_CONF_ROLE_COHOST联席主持人和HWM_CONF_ROLE_WAITING等候室成员，详见[HwmConfRole说明](#)

7.8.13 70.14.5 更新内容

- 1.新增设置网络模式接口SetNetworkMode，接口详见[网络模式设置](#)
- 2.登录接口HwmAppIdAuthInfo参数新增字段：deptCode 部门id，接口详见[登录](#)
- 3.预约会议支持设置“来宾在主持人之前加入会议”，并可设置提前入会时间，接口详见[预约会议](#)

7.8.14 70.13.5 更新内容

- 1.OnLoginResult、OnLoginBySSOResult回调增加HwmLoginResult参数；OnLogoutResult回调增加HwmLogoutResult参数，接口详见[帐号操作](#)。如果是老版本升级到70.13.5版本,需要修改回调函数的定义。
- 2.修改winUI日志后缀名为txt，修改了四个日志命名：hwm_winconfui.txt，hwmSdk.txt，hwmSdkAgent.txt，hwmSdkAgent32.txt
- 3.Config接口中新增配置项：isUploadKeyLog 是否自动收集日志的开关，接口详见[7.5.3.1-界面配置](#)
- 4.Init接口HwmInitInfo参数新增字段：preferredCamera 优选摄像头列表参数，接口详见[7.5.1.1-初始化](#)

5.新增接口AVDeviceCtrl音视频设备控制和结果回调OnAVDeviceCtrlResult，接口详见[音视频设备控制](#)；新增对应通知OnAVDeviceCtrlStateChanged音视频设备状态，通知详见[音视频设备状态](#)

6.Config接口配置自定义按钮，支持动态刷新底部/共享工具栏，接口详见[界面配置](#)

7.8.15 70.12.6 更新内容

- 1、修复由于弱网导致视频清晰度下降，网络恢复后视频清晰度无法恢复的问题。
- 2、修复隐藏投票功能。

7.8.16 70.12.4 更新内容

- 1.CreateConf接口HwmCreateConfInfo参数新增选填字段：duration 会议时长，详见[创建会议](#)
- 2.新增自定义设置界面接口ConfigSettingWindow，详见[自定义设置界面](#)
- 3.新增企业信息变更通知OnUserCorpInfo，详见[企业信息变更](#)

7.8.17 70.11.5 更新内容

- 1.Init接口HwmInitInfo参数新增字段：logKeepDays 日志保存天数
接口详见[7.5.1.1-初始化](#)

7.8.18 70.10.5 更新内容

- 1.Config接口新增配置项：isHideTransferHostBeforeLeave 是否隐藏离会时指定主持人功能按钮
接口详见[界面配置](#)

7.8.19 70.9.5 更新内容

- 1.登录新增“SSO登录”接口，详见[SSO登录](#)

7.8.20 70.8.6 更新内容

- 1.新增“刷新自定义按钮配置”接口，详见[刷新自定义按钮配置](#)
- 2.修复问题，体验更稳定

7.8.21 70.8.5 更新内容

- 1.支持获取与会者列表（含网络研讨会），详见[获取与会者列表](#)
- 2.支持获取音频数据流，详见[音频数据流开放](#)

7.8.22 70.7.5 更新内容

- 1.支持视频镜像能力，打开后，观看自己显示为镜像效果
- 2.提供32位&64位两个版本SDK，详见[快速入门](#)
- 3.支持会控工具栏的自定义，详见[界面配置](#)

- 4.支持会中背景替换和背景虚化
- 5.支持共享作为广播多画面子画面显示，该模式下视频画面和共享画面可自由切换
6. 新增[设置共享策略](#)、[设置共享音频](#)接口

7.8.23 70.6.5 更新内容

- 1.新增“显示会议主窗口”接口，详见[显示会议主窗口](#)
- 2.会中一键打开有视频画面，观看更方便
- 3.设置会议入会范围后，非指定类型用户加入会议或会议锁定时，错误码从原来的加入会议时的回调中上报，改成从OnConfState离会原因中上报。详见[会议状态](#)

7.8.24 70.5.8 更新内容

- 1.优化会中视觉体验

7.8.25 70.4.5 更新内容

- 1.预约会议接口添加返回值参数说明，具体使用请见[预约会议](#)
- 2.新增获取用户当前状态接口，具体使用请见[获取用户状态](#)
- 3.支持安全、字幕、锁定共享可配置隐藏，具体使用请见[界面配置](#)
- 4.会中新增主持人邀请他人共享，主持人也可快速取消共享
- 5.支持阿拉伯语文件路径下安装使用
- 6.会议日志大小优化

7.8.26 70.3.5 更新内容

1. 会中导出与会者信息优化

7.8.27 60.15.5 更新内容

1. 初始化回调结果未返回，调用其他任何接口都将返回失败，增加未初始化或初始化尚未完成错误码HWM_AGENT_ERR_NOT_INITED_END。
2. 新增与会者列表外部标签是否隐藏功能。具体参见[7.5.1.4开启功能](#)
3. 新增呼叫来电弹窗是否隐藏、会议来电弹窗是否隐藏的功能。具体参见[7.5.3.1界面配置](#)
5. 新增本地录制功能，支持主持人控制与会者本地录制权限、支持配置录制文件保存路径[设置用户文件保存路径](#)
6. 会中新增“安全”模块，统一管理权限

7.8.28 60.13.2 更新内容

1. 支持初始化之前修改HwmSdk文件夹下HwmSdk.exe文件的名称，具体使用请见[表1](#)
2. 初始化接口参数exePath的值从HwmSdk文件夹的绝对路径变更为exe文件的绝对路径，**并且要求是utf8格式**。具体使用请见[表1](#)

3. 初始化接口新增参数logPath和userDataPath，可以指定日志文件和用户数据文件的保存路径，**并且要求是utf8格式**。具体使用请见[表1](#)

7.8.29 60.12.5 更新内容

1. 创建会议支持指定会议室（vmrId）和入会范围（callInRestriction），具体使用见[表1](#)
2. 新增自定义会议信息框，具体使用请见[图7](#)，[显示会议信息框](#)，[销毁会议信息框](#)
3. 新增查询外部通讯录头像，可以在会中关闭摄像头时，显示自定义头像，具体使用请见[查询外部通讯录头像](#)

7.8.30 60.11.6 更新内容

1. 新增会议支持自动录制，新增允许自动录制会议isAutoRecord字段，
注意：会议开始后是否自动开启录制，默认关闭。开启自动录制会议时必须开启录制会议
 - 创建会议[7.5.4.1-表1 结构体HwmCreateConfInfo参数说明](#)
 - 预约会议、编辑会议[表1](#)

7.8.31 60.10.5 更新内容

1. 初始化增加登录站点类型，具体使用请见[7.5.1.1-初始化](#)
2. 支持使用random入会，具体使用请见[7.5.4.3-Random入会](#)
3. 增加sp鉴权登录，增加字段corpId。具体使用请见[7.5.2.1-登录](#)
4. 优化水印显示，具体使用请见[7.5.1.5-设置水印](#)

7.8.32 60.8.1 更新内容

1. 初始化增加AppId，优化App区分
2. 新增AppId登录、预约会议、编辑会议、会议列表、会议详情、取消会议等接口

8 Mac SDK

8.1 概述

华为云会议终端Mac SDK提供了一套完整的接口集合，开发者可以通过调用SDK开放的API，在自己的应用程序中快速集成会议功能，包括创建/预约/加入会议、进行屏幕共享等。

8.2 修订记录

表 8-1 修订记录

日期	版本	修订内容
2023-2-23	80.13.6	1. 问题修复，体验更稳定。 具体使用请见 80.13.6更新内容
2023-2-7	80.13.5	1. 问题修复，体验更稳定。 具体使用请见 80.13.5更新内容
2023-1-17	80.12.7	1. 问题修复，体验更稳定。 具体使用请见 80.12.7更新内容
2023-1-9	80.12.6	1. 开启功能接口enableSDKCapability新增是否显示录制安全提示窗枚举值HWMSDKCapabilitySupportConfirmRecord。 2. 问题修复，体验更稳定。 具体使用请见 80.12.6更新内容
2022-12-24	80.12.5	1. 上传个人头像上传条件变更（增加类型和大小判断）。 2. 问题修复，体验更稳定。 具体使用请见 80.12.5更新内容

日期	版本	修订内容
2022-11-16	80.10.5	<ol style="list-style-type: none"> 1. 云会议室列表HWMVmrInfoModel.h、修改云会议室信息HWMModifyVMRParam.h新增字段confAllowJoinUser、isOpenWaitingRoom。 2. 新增设置共享权限接口setShareScreenPermission。 具体使用请见80.10.5更新内容
2022-10-8	80.9.5	<ol style="list-style-type: none"> 1. 会议详情HWMConfDetail.h、编辑会议HWMOrderConfParam.h新增字段isRecordOn、isLiveOn。 2. 开启功能接口enableSDKCapability新增分组讨论枚举值HWMSDKCapabilitySupportBreakoutConf。 3. 取消周期子会议CancelSubCycleConf中字段subConfUUID改成subConfID。 4. 预约周期会议和编辑周期会议接口，HWMCycleConfParam.h中cycleCount字段废弃。 具体使用请见80.9.5更新内容
2022-9-2	80.8.5	<ol style="list-style-type: none"> 1. 问题修复，体验更稳定。 具体使用请见80.8.5更新内容
2022-7-30	80.7.5	<ol style="list-style-type: none"> 1. App支持自定义多语言。 2. 新增显示设置界面接口showSettingWindowWithInfo。 3. 新增与会者列表子项注入点击通知onClickInjectParticipantItemBtnNotify。 4. 新增错误码。 具体使用请见80.7.5更新内容
2022-6-17	80.5.11	<ol style="list-style-type: none"> 1. 新增禁用提示音接口。 2. 新增录制按钮操作定制。 具体使用请见80.5.11更新内容
2022-5-18	80.5.5	<ol style="list-style-type: none"> 1. 创建会议HWMCreateConfParam.h、预约会议和编辑会议HWMOrderConfParam.h、查询会议详情HWMConfDetail.h新增字段isOpenWaitingRoom。 2. 支持关闭和开启会议设置中隐藏非视频与会者选项。 具体使用请见80.5.5更新内容
2022-4-15	70.16.5	<ol style="list-style-type: none"> 1. 新增周期会议相关接口。 2. id入会HWMJoinConfParam.h 新增字段isStopConflictConf。 3. sso 登录接口调整。 4. 错误码整改，发布新版本错误码描述以及新旧版本错误码对应关系。 具体使用请见70.16.5更新内容

日期	版本	修订内容
2022-2-19	70.14.5	<ol style="list-style-type: none"> 1. 预约云会议室时支持选择预约固定云会议室ID或随机云会议室ID。 2. 增加点对点发起呼叫接口，以及通话状态通知和通话结束通知。 3. 预约、编辑会议支持设置“来宾在主持人之前加入会议”，并可设置提前入会时间（allowGuestStartConf、allowGuestStartConfTime）。 4. APPID登录支持指定部门ID。 5. 新增设置网络模式接口。 具体使用请见 70.14.5 更新内容
2022-01-19	70.13.5	<ol style="list-style-type: none"> 1. login 和 logout 相关接口的回调中增加用户的 uuid（原来uuid字段）、account、thirdAccount， 2. 新增会议来电状态变更通知 3. 新增呼叫来电通知 4. 修改了获取窗口句柄参数 5. 新增开启共享声音功能 6. 本地录制默认路径变更，接入方App需要申请下载文件夹权限 具体使用请见 70.13.5更新内容
2021-12-23	70.12.6	<ol style="list-style-type: none"> 1. 修复弱网情况下导致清晰度降低后不能恢复的问题 具体使用请见 70.12.6更新内容
2021-12-15	70.12.4	<ol style="list-style-type: none"> 1. 支持自定义会议设置窗口 2. 支持开启特性支持 具体请见 70.12.4 更新内容
2021-12-3	70.11.5	<ol style="list-style-type: none"> 1. 工程配置的签名脚本和Runpath Search Paths配置有变更 2. 初始化接口参数新增字段：logKeepDays 日志保存天数 具体请见 70.11.5更新内容
2021-10-22	70.10.5	<ol style="list-style-type: none"> 1. 支持自定义水印 2. 支持自定义本地录制路径 具体使用请见 70.10.5 更新内容
2021-09-16	70.9.5	<ol style="list-style-type: none"> 1.新增SSO登录方式 具体使用请见 70.9.5更新内容
2021-08-28	70.8.6	问题修复，体验更稳定

日期	版本	修订内容
2021-08-18	70.8.5	<ul style="list-style-type: none"> 1.支持获取与会者列表（含网络研讨会） 2.支持获取音频数据流 具体使用请见 70.8.5更新内容
2021-07-16	70.7.5	<ul style="list-style-type: none"> 1.支持会中设置本会场视频画面是否镜像 2.会中UI界面更新 3.支持会控工具栏的自定义 具体使用请见 70.7.5更新内容
2021-06-10	70.6.5	<ul style="list-style-type: none"> 1. 新增“显示会议主窗口”接口 具体使用请见 70.6.5更新内容
2021-05-17	70.5.8	<ul style="list-style-type: none"> 1. 工程配置的签名脚本有变更，新增代码如下，详见（修改库的签名） 2. 支持共享作为广播多画面子画面显示，该模式下视频画面和共享画面可自由切换 3. 支持聊天弹幕 具体使用请见 70.5.8更新内容
2021-04-15	70.4.5	<ul style="list-style-type: none"> 1. 工程配置的签名脚本有变更，见（修改库的签名） 2. 查询当前终端是否在会中，具体使用请见获取用户状态 3. 支持安全可配置隐藏，具体使用请见初始化 4. 会中新增主持人邀请他人共享，主持人也可快速取消共享 5. 日志大小优化 具体使用请见 70.4.5更新内容
2021-03-12	70.3.5	<ul style="list-style-type: none"> 1. 会中导出与会者信息优化 具体使用请见 70.3.5更新内容
2021-02-10	60.15.5	<ul style="list-style-type: none"> 1. 新增会议来电接听策略通知接口 2. 新增入会扬声器设置参数 3. 新增创建、预约、编辑会议密码及语音会议参数 4. 会中新增“安全”模块，统一管理权限 具体使用请见 60.15.5更新内容
2021-01-08	60.13.2	<ul style="list-style-type: none"> 1. 新增会中添加与会者接口 2. 新增离开会议接口 3. 新增结束会议接口 4. 新增设置语言接口 具体使用请见 60.13.2更新内容
2020-12-07	60.12.5	首次发布。

8.3 快速入门

本小节以实现初始化SDK为例，介绍如何使用Mac SDK进行二次集成开发。

8.3.1 开发准备

1. 环境及项目要求参见“[环境要求](#)”
2. **可选:** 在Mac电脑上解压下载到的“hwmsdk-mac.zip”软件包。软件包下载地址参见“[下载地址](#)”

8.3.2 快速集成

步骤1 将下载的软件包全部解压得到HWMUISDK（集成使用）和HWMUISDKNativeDemo（示例Demo）。

图 8-1 解压出来的文件



说明

HWMUISDK中的AW_PortraitSDK.framework是背景虚化功能使用。如果需要集成背景虚化功能，请联系华为销售人员。

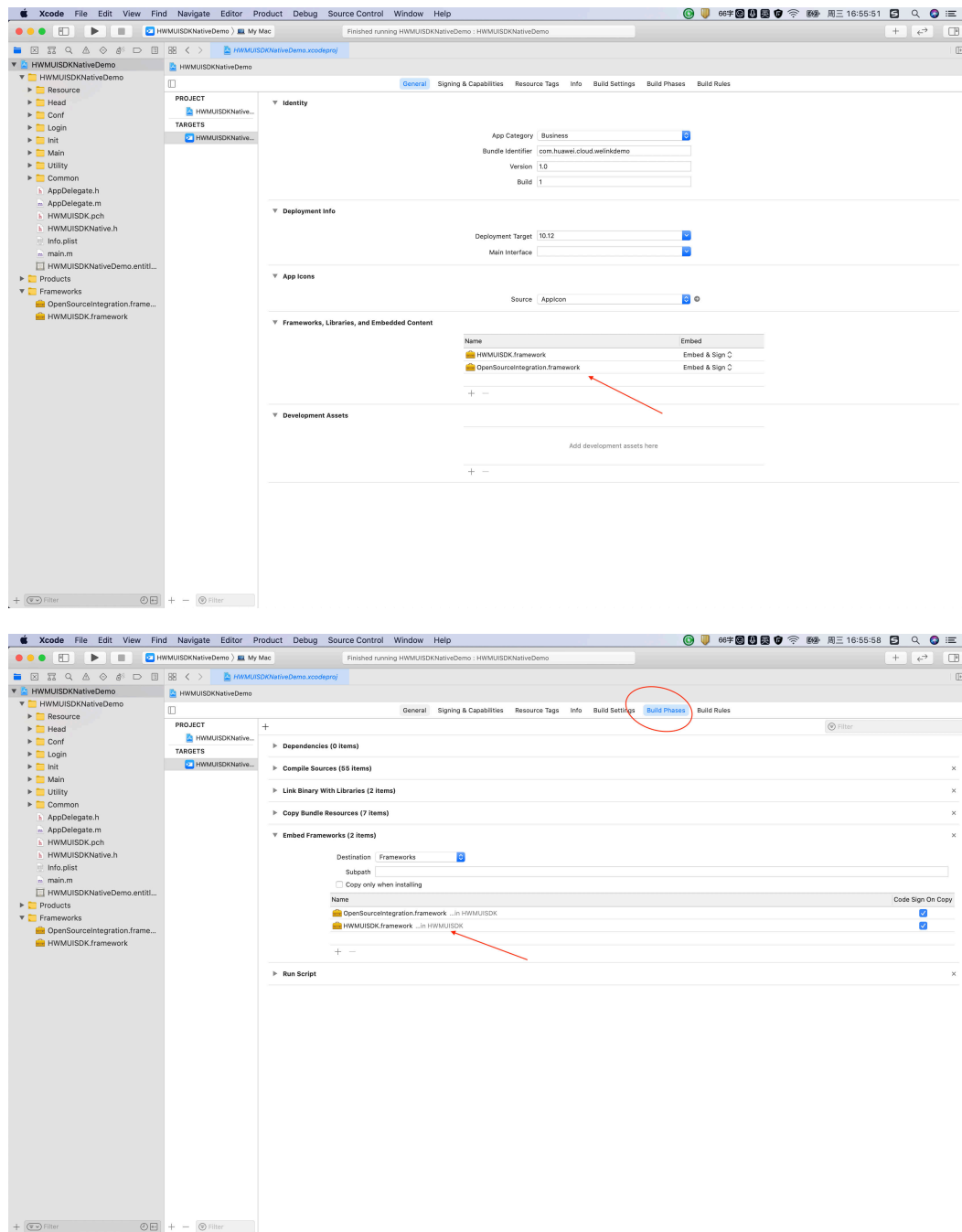
步骤2 打开“Xcode”软件。

步骤3 单击“File->New->Project...”，选择“App->Next”，填写应用信息，选择“Create”完成新工程创建。

步骤4 将**步骤1**中解压得到的HWMUISDK.framework、OpenSourceIntegration.framework添加到项目中。

添加后在“TARGETS->General->Frameworks、Libraries、and Embedded Content”中修改为Embed & Sign。

图 8-2 添加完成后的效果



步骤5 工程文件配置。

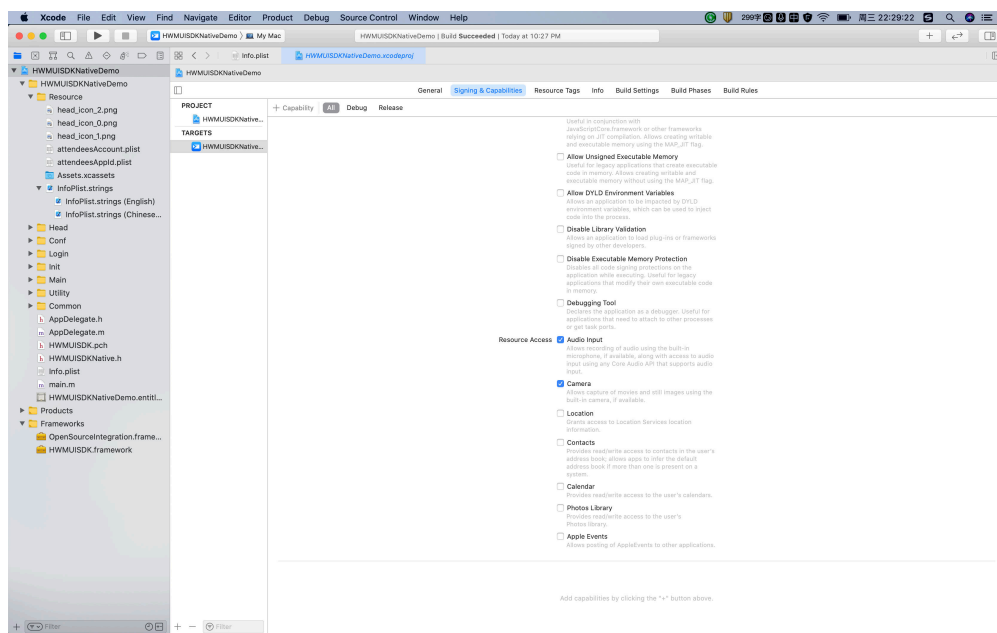
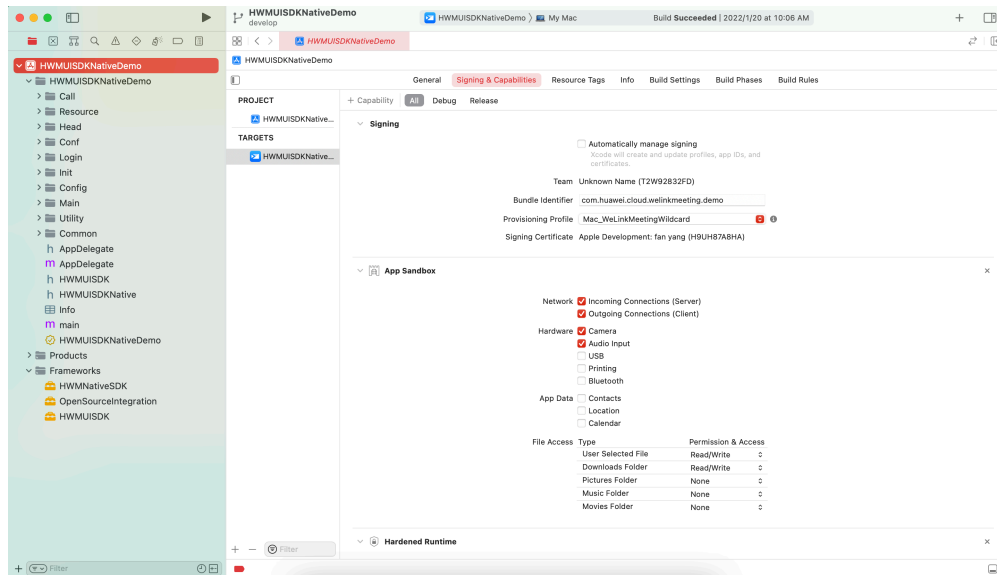
1. 开启sandbox对应的项目。若不上应用市场也可以不开启，支持不开启sandbox情况。

设置以下sandbox配置项key的value为true

```
com.apple.security.app-sandbox
com.apple.security.device.audio-input
com.apple.security.device.camera
com.apple.security.files.user-selected.read-write
com.apple.security.files.downloads.read-write
```

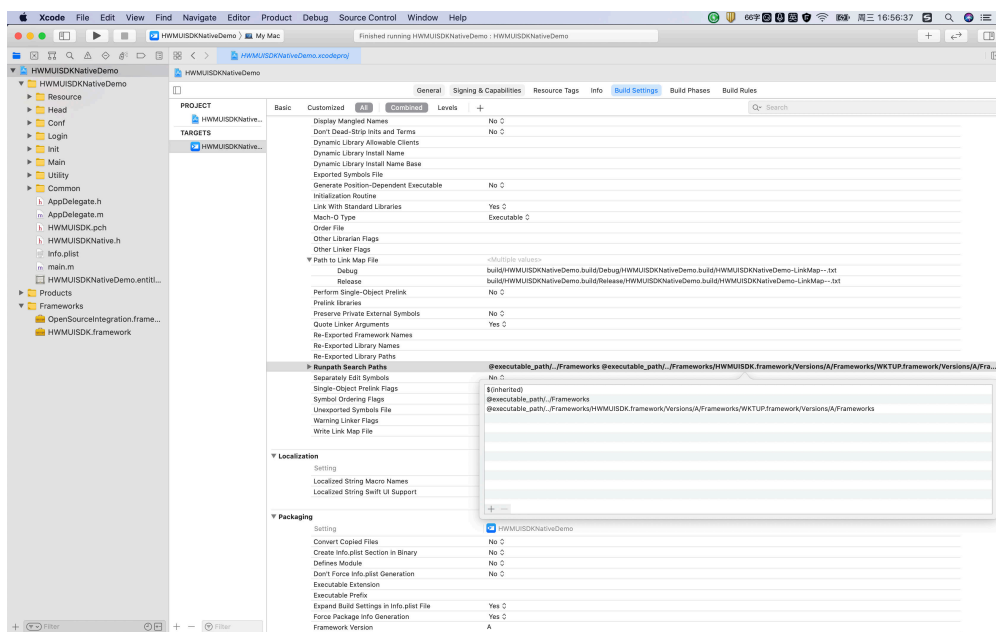
com.apple.security.network.client
com.apple.security.network.server

图 8-3 sandbox 对应开启



- Build Settings配置，需要配置Runpath Search Paths，增加
@executable_path/../Frameworks
@executable_path/../Frameworks/HWMUISDK.framework/Versions/A/
Frameworks
两个相对路径

图 8-4 Runpath Search Paths 路径增加配置项



3. Build Phases增加Run Script修改提供库的重签名，与应用App的签名保持一致。

```
cd "$BUILT_PRODUCTS_DIR"/
#xxxx替换成应用自己的app名字
cd ./xxxx.app/Contents/Frameworks/
echo "##### HWMUISDK.framework xattr #####"
xattr -r -d com.apple.quarantine ./HWMUISDK.framework
xattr -r -d com.apple.quarantine ./OpenSourceIntegration.framework
echo "##### finish #####"

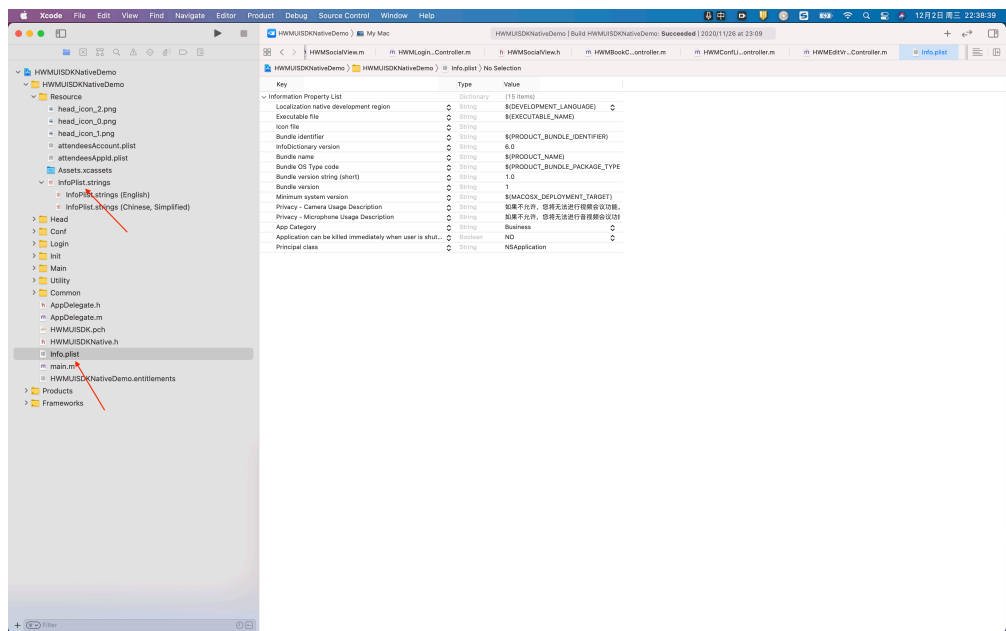
echo "##### HWMUISDK.framework codeSgin #####"
codesign -f -s "$EXPANDED_CODE_SIGN_IDENTITY" --timestamp=none ./
*.framework
codesign -f -s "$EXPANDED_CODE_SIGN_IDENTITY" --timestamp=none ./
HWMUISDK.framework/Versions/A/Frameworks/*.framework
codesign -f -s "$EXPANDED_CODE_SIGN_IDENTITY" --timestamp=none ./
HWMUISDK.framework/Versions/A/Frameworks/HWMNativeSDK.framework/
Versions/A/Frameworks/WKTUP.framework/Versions/A/Frameworks/*.dylib
codesign -f -s "$EXPANDED_CODE_SIGN_IDENTITY" --timestamp=none ./
HWMUISDK.framework/Versions/A/Frameworks/HWMNativeSDK.framework/
Versions/A/Frameworks/WKTUP.framework
codesign -f -s "$EXPANDED_CODE_SIGN_IDENTITY" --timestamp=none ./
HWMUISDK.framework/Versions/A/Frameworks/HWMNativeSDK.framework/
Versions/A/Frameworks/WKTUP.framework/Versions/A/Frameworks/
*.framework
```

图 8-5 修改库的签名



4. Info.plist配置，需要配置权限提示语（语言信息根据自己需要进行设置）以及程序退出相关属性设置，需要配置的字段如下：
 - Privacy - Camera Usage Description 使用相机权限提示语
 - Privacy - Microphone Usage Description 使用麦克风权限提示语
 - Application can be killed immediately when user is shutting down or logging out 设置NO

图 8-6 Info.plist 信息配置完成效果



----结束

8.3.3 业务集成

在调用HwMSDK接口的文件中必须包含以下头文件：

```
#import <HwMUISDK/HwMUISDK.h>
```

SDK初始化：**初始化时必须**在主线程调用


```

- (void)initSDK {
    HWMOpenSDKConfig *config = [[HWMOpenSDKConfig alloc] init];
    config.appId = @"fdb8e4699586458bbd10c834872dcc62";// APPID 接入方式时, config.appId 传入申请到的
    企业APPID; 非APPID接入方式推荐传入应用的英文名称
    BOOL result = [HWMSdk initWithConfig:config];
    if (result) {
        [UIUtil showMessage:@"初始化成功" vc:self];
    }
}

```

接口调用示例

- **登录接口**
[[HWMSdk getOpenApi] login:account password:password callback:^(NSError * _Nullable error, HWMLoginResult * _Nullable result){ }];
- **创会接口**
[[HWMSdk getOpenApi] createConf:param callback:^(NSError * _Nullable error, HWMCreateConfResult * _Nullable result) { }];
- **入会接口**
[[HWMSdk getOpenApi] joinConf:param callback:^(NSError * _Nullable error, id _Nullable result) { }];
- **预约会议**
[[HWMBizSdk getBizOpenApi] bookConf:param callback:^(NSError * _Nullable error, id _Nullable result) { }];
- **编辑会议**
[[HWMBizSdk getBizOpenApi] editConf:param callback:^(NSError * _Nullable error, id _Nullable result) { }];
- **退出接口**
[[HWMSdk getOpenApi] logout:^(NSError * _Nullable error, id _Nullable result) { }];

完成上述接口调用代码后即完成集成SDK，更详细信息请参考 HWMUISDKNativeDemo 样例代码。

8.4 典型场景

本小节描述SDK在场景使用时的接口调用顺序和接口调用示例。

8.4.1 典型场景接口调用概览

3 类接口描述

Mac SDK的接口分为**同步Api**、**异步Api**、**全局回调**。

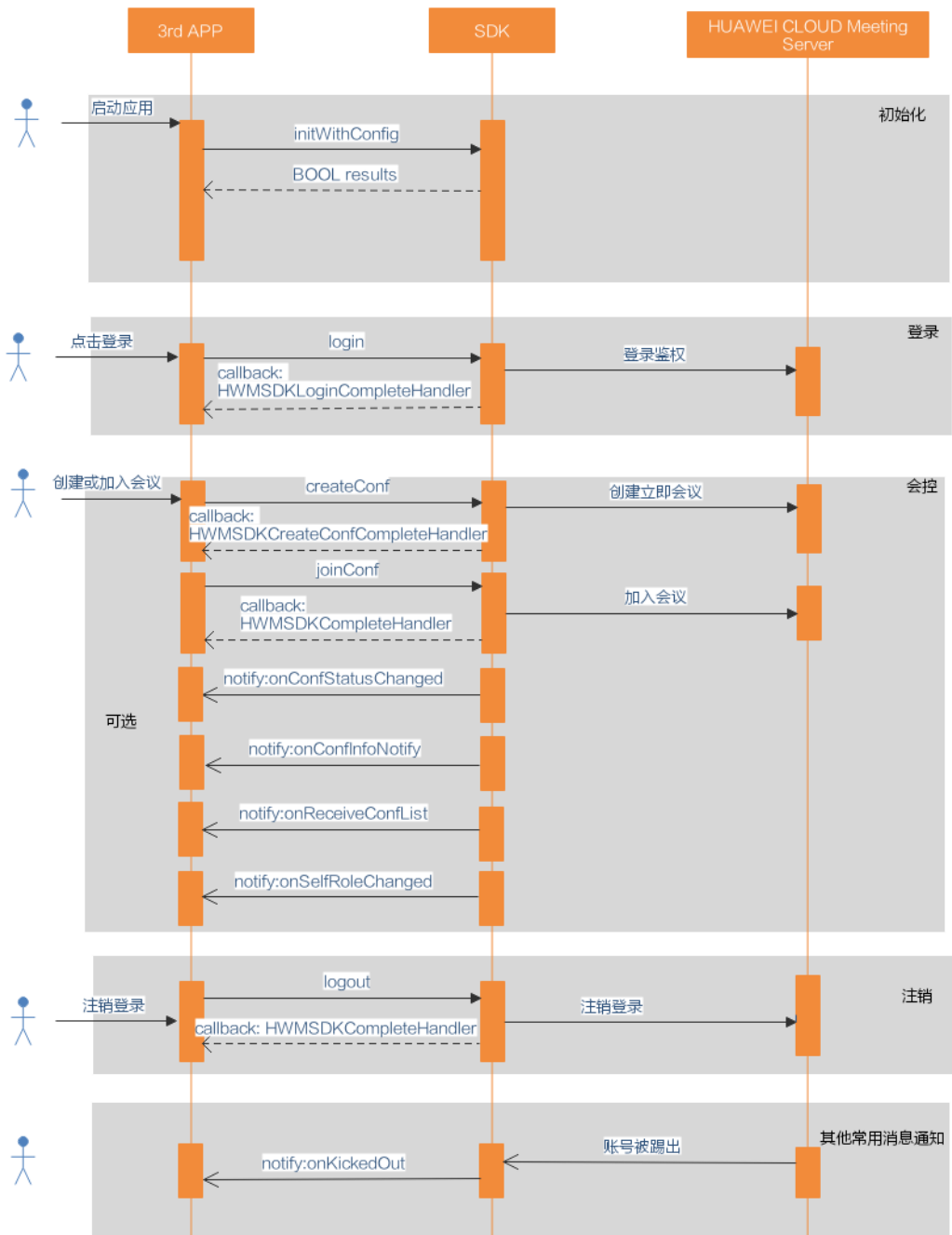
同步Api: App主动调用SDK的接口，用于完成某项任务。比如初始化接口 initWithConfig。

异步Api: 调用SDK接口的时候都会以callback回调方式请求结果。

全局回调: 由华为云会议服务端或SDK内部触发的事件通知，SDK以消息通知接口的形式通知给App。比如会议来电通知onConfStatusChanged。

典型场景接口的调用顺序及生命周期

图 8-7 典型场景接口调用时序图



说明

- 1、使用SDK功能前需要初始化。
- 2、在不登录的情况下也能通过会议ID和密码加入已经创建好的会议，其它使用场景需要登录。
- 3、可选通知根据需求场景决定是否实现。

8.4.2 场景 1：初始化

描述

在第三方APP启动时，需要先初始化SDK，才能调用SDK的功能接口，该接口只需调用一次。

业务流程

SDK初始化时，先调用initWithConfig接口，根据返回值确定接口是否调用成功。

步骤1 接口调用

1. 构建HWMOpenSDKConfig模型参数。
2. 调用initWithConfig接口，完成配置初始化，第1步中的数据作为参数。

步骤2 处理返回值

处理返回值，YES代表初始化成功、NO代表初始化失败。

----结束

示例代码

```
/// 初始化
HWMOpenSDKConfig *config = [[HWMOpenSDKConfig alloc] init];
config.appld = @"fdb8e4699586458bbd10c834872dcc62"; // 传入申请到的企业Appld
BOOL result = [HWMSdk initWithConfig:config];
if (result) {
    NSLog(@"初始化成功");
}
```

说明

典型场景和接口参考中的示例代码仅作为示例用的伪代码，不能直接使用。

8.4.3 场景 2：登录

8.4.3.1 帐号密码登录

描述

该接口用于帐号登录。创建会议等接口不支持在未登录状态下调用，若要使用完整的会议功能，必须先登录。

说明

在未登录状态下，只能加入已经创建的会议。

业务流程

使用SDK登录时，先调用login接口，然后处理回调callback。

步骤1 接口调用

1. 获得登录使用的帐号和密码

2. 调用login进行登录，第1步中的数据作为参数。

步骤2 处理回调

处理回调callback。

----结束

示例代码

```
/// 登录处理
- (void)login {
    NSString *account = @"439571854984";
    NSString *password = @"Change_Me";
    [[HWM SDK getOpenApi] login:account password:password callback:^(NSError * _Nullable error,
HWMLoginResult * _Nullable result) {
        if (error == nil) {
            NSLog(@"登录成功");
        }else{
            NSLog(@"登录失败");
        }
    }
};
}
```

8.4.3.2 App ID 登录

描述

该接口用于通过App ID登录

📖 说明

在未登录状态下，只能加入已经创建的会议。

业务流程

使用SDK登录时，先调用login接口，然后处理回调callback。

步骤1 接口调用

1. 从接入方服务器获得登录使用的帐号、过期时间、随机数、用户名、加密签名
2. 调用login进行登录，第1步中的数据作为参数。

步骤2 处理回调

处理回调callback。

----结束

示例代码

```
/// AppId 登录接口
- (void)login {
    HWMAppIDLoginParam *appidInfo = [[HWMAppIDLoginParam alloc] init];
    appidInfo.thirdUserId = @"test";
    appidInfo.expireTime = 1599618620;
    appidInfo.nonce = @"0vjeeKUvztUu6N20IqfooegDVGi5Kf30";
    appidInfo.userName = @"测试帐号";
    appidInfo.signature = @"fac06749023bfcfbfc887b3b5fa030cc3dbacb2c95a3fed85664bae55ba334f9c";
    [[HWM SDK getOpenApi] login:appidInfo callback:^(NSError * _Nullable error, HWMLoginResult *
_Nullable result) {
        if (error == nil) {
```

```

        NSLog(@"登录成功");
    }else{
        NSLog(@"登录失败");
    }
}
}
};
}

```

8.4.4 场景 3：创建会议

描述

登录后，可以调用创建会议接口创建立即会议。创建会议时可以携带与会人信息，也可以不携带。

说明

在登录状态下，才能创建会议。

业务流程

使用SDK创建立即会议时，先调用CreateConf接口，然后处理回调callback。

步骤1 接口调用

1. 组装模型HWMCreateConfParam。
2. 组装模型数组HWMAttendeeInfo（与会者可选）。
3. 调用CreateConf开始创建，第1步中的数据作为参数。

步骤2 处理回调

处理回调callback。

----结束

示例代码

```

/// 创建会议
- (void)creatMeeting {
    HWMCreateConfParam *param = [[HWMCreateConfParam alloc] init];
    param.subject = @"会议标题";
    param.isCameraOn = YES;// 是否打开摄像头, 默认关闭
    param.isMicOn = YES;// 是否打开麦克风, 默认打开
    param.isAutoRecord = NO;// 是否打开会议录制, 默认关闭
    param.callInRestrictionType = HWMJoinConfRestrictionAll;// 允许入会范围
    // 与会者列表
    if (self.selectedMemebrs) {
        __block NSMutableArray *members = [[NSMutableArray alloc] init];
        [self.selectedMemebrs enumerateObjectsUsingBlock:^(HWMContactSelectedModel * _Nonnull obj,
        NSUInteger idx, BOOL * _Nonnull stop) {
            HWMAttendeeInfo *member = [[HWMAttendeeInfo alloc] init];
            member.accountId = obj.accountId;
            member.number = obj.number;
            member.name = obj.name;
            member.thirdUserId = obj.thirdUserId;
            [members addObject:member];
        }];
        param.members = members;
    }
    [[HWMSdk getOpenApi] createConf:param callback:^(NSError * _Nullable error, HWMCreateConfResult *
    _Nullable result) {
        [self hideLoading];
        if (error) {

```

```

        NSLog(@"创会失败%@", error.localizedDescription);
    }else{
        NSLog(@"创会成功");
    }
}
}];
}

```

📖 说明

与会者列表可以携带与会者创建会议，创建会议成功后服务端会呼叫携带的与会人入会，不需要可以不传。

8.4.5 场景 4：添加与会者

描述

在会议中，可以调用添加与会者的接口把与会人加入到会议中。

业务流程

使用SDK添加与会人时，需要自定义加人界面，选择要添加的联系人，之后把选中的联系人添加到会议中。

📖 说明

1.Mac SDK 仅支持自定义添加与会者。

示例代码

```

/// 自定义联系人
- (void)openContactSelectWindow:(NSArray<HWMAttendeeInfo *> *)param scene:
(HWMSelectedContactScene)scene completeHandler:(void (^)(NSArray<HWMAttendeeInfo *> *
_Nullable))handler {
    HWMSelectContactViewController *contact = [[HWMSelectContactViewController alloc] init];
    [contact setSelectAttendeeHandler:^(NSArray *attendees) {
        NSMutableArray *arr = @[].mutableCopy;
        for (HWMContactInfoModel *model in attendees) {
            HWMAttendeeInfo *info = [[HWMAttendeeInfo alloc] init];
            info.accountId = model.accountId;
            info.name = model.name;
            info.number = model.number;
            info.thirdUserId = model.thirdUserId;
            info.email = model.email;
            info.mute = model.mute;
            info.role = model.role;
            info.sms = model.sms;
            [arr addObject:info];
        }
        if (handler) {
            handler(arr);
        }
    }];
    self.selectContactVC = contact;
    [self presentViewControllerAsModalWindow:contact];
}

```

8.4.6 场景 5：加入会议

描述

在登录后或者登录前，都可以使用该接口通过会议ID和密码加入会议。

业务流程

使用SDK加入已经存在的会议时，先调用joinConf接口，然后处理回调callback。

步骤1 接口调用

1. 组装模型
 - a. 会议id加入会议使用HWMJoinConfParam。
 - b. 会议random加入会议使用HWMRandomConfParam。
2. 调用joinConf开始加入会议，第1步中的数据作为参数。

步骤2 处理回调

处理回调callback。

----结束

示例代码

```
/// 加入会议
- (void)joinMeeting {
    HWMJoinConfParam *param = [[HWMJoinConfParam alloc] init];
    param.confId = @"989156631";
    param.password = @"807766";
    param.nickname = @"杭州会场";
    param.isCameraOn = NO;
    param.isMicOn = YES;
    [[HWMSdk getOpenApi] joinConf:param callback:^(NSError * _Nullable error, id _Nullable result) {
        if (error) {
            NSLog(@"入会失败%@", error.localizedDescription);
        } else {
            NSLog(@"入会成功");
        }
    }];
}
```

📖 说明

在未登录状态下调用该接口时，nickname参数必填。

8.4.7 场景 6：发起呼叫

描述

登录后，可以调用接口发起呼叫。

业务流程

使用SDK发起呼叫时，先调用startCall接口，然后处理回调callback。

步骤1 接口调用

1. 调用startCall发起呼叫。

步骤2 处理回调

处理回调callback。

----结束

示例代码

```

/// 发起呼叫
- (void)startCall {
    /**
     * AppId 登录场景：account/number/thirdUserId 三选一，优先级：number>account>thirdUserId
     * 非AppId 登录场景：account/number 二选一，优先级：number>account
     */
    HWMStartCallParam *param = [[HWMStartCallParam alloc] init];
    param.isVideo = YES;
    param.account = @"";
    param.number = @"+991112988657623";
    param.thirdUserId = @"";
    [[HWMSdk getOpenApi] startCall:param callback:^(NSError * _Nullable error, id _Nullable result) {
        if (error) {
            NSLog(@"发起呼叫失败%@", error.localizedDescription);
        } else {
            NSLog(@"发起呼叫成功");
        }
    }
    }
}

```

8.4.8 场景 7：预约会议

描述

登录后，可以调用接口预约会议。

业务流程

使用SDK预约会议时，先调用bookConf接口，然后处理回调callback。

步骤1 接口调用

1. 调用bookConf预约会议。

步骤2 处理回调

处理回调callback。

----结束

示例代码

```

/// 预约会议
- (void)bookConf {
    HWMOrderConfParam * param = [[HWMOrderConfParam alloc] init];
    param.confSubject = @"会议名称";
    param.startTime = 1598759580;
    param.duration = 15;
    param.isAutoRecord = NO;
    param.timeZone = 56;
    param.vmrId = @"";
    param.callInRestrictionType = HWMJoinConfRestrictionAll;
    [[HWMBizSdk getBizOpenApi] bookConf:param callback:^(NSError * _Nullable error, id _Nullable result) {
        if (error) {
            NSLog(@"预约会议失败:%@", error.localizedDescription);
        } else {
            NSLog(@"预约会议成功");
        }
    }
    }
}

```


8.4.9 场景 8：编辑会议

描述

登录后，调用[会议详情](#)接口（也可以使用服务器的rest接口）获取会议详情后调用接口编辑会议。

业务流程

使用SDK编辑会议时，先调用editConf接口，然后处理回调callback。

步骤1 接口调用

1. 调用editConf编辑会议。

步骤2 处理回调

处理回调callback。

----结束

示例代码

```
/// 编辑会议
- (void)editConf {
    HWMOrderConfParam * param = [[HWMOrderConfParam alloc] init];
    param.confSubject = @"会议名称";
    param.startTime = 1598759580;
    param.duration = 15;
    param.isAutoRecord = NO;
    param.timeZone = 56;
    param.vmrId = @""; // vmr会议需要传入vmrId
    param.callInRestrictionType = HWMJoinCallRestrictionAll;
    param.confId = @"986927771";
    [[HWMBizSdk getBizOpenApi] editConf:param callback:^(NSError *_Nullable error, id _Nullable result) {
        if (error) {
            NSLog(@"编辑会议失败:%@", error.localizedDescription);
        } else {
            NSLog(@"编辑会议成功");
        }
    }];
}
```

8.4.10 场景 9：退出登录

描述

当需要退出当前会议帐号登录时，可以调用该接口完成已登录会议帐号退出。

业务流程

使用SDK退出登录时，先调用logout接口，然后处理回调callback。

步骤1 接口调用

1. 调用logout进行退出登录。

步骤2 处理回调

处理回调callback。

----结束

示例代码

```
/// 退出登录
- (void)logout {
    [[HWMSdk getOpenApi] logout:^(NSError * _Nullable error, id _Nullable result) {
        if (error == nil) {
            NSLog(@"登出成功");
        } else {
            NSLog(@"登出失败");
        }
    }];
}
```

8.5 接口参考

8.5.1 基础配置

8.5.1.1 初始化

initWithConfig

接口描述

用于启动并且初始化SDK。

注意事项

1. 该接口用于启动并初始化SDK，在调用其他接口之前必须先调用该接口。
2. 初始化方法必须在主线程调用。
3. 返回值只代表接口是否调用成功。

方法定义

```
+ (BOOL)initWithConfig:(HWMSdkConfig *)config;
```

参数描述

表 8-2 参数说明

参数	是否必须	类型	描述
config	是	HWMSdkConfig	初始化信息集合

表 8-3 初始化信息 HWMOpenSDKConfig

参数	是否必须	类型	描述
appid	是	NSString *	App ID。App ID的申请, 请参考《开发指南》的" App ID的申请 "章节。
siteType	否	HWMSiteType	登录站点类型, 默认为中国站点 (HWMSiteTypeChina)
hideExternalLabel	否	BOOL	是否显示与会者列表中外部标签, 默认显示
hideAllowJoinConf Menu	否	BOOL	是否隐藏安全中的允许入会菜单, 默认不隐藏
logKeepDays	否	NSInteger	配置项-日志保存天数, 输入有效范围为3-30, 默认值0。0为不按天数删除日志。 注意: 输入1和2会转换为3, 大于30的值会被转换为30。
appUrl	否	NSString *	官网下载渠道链接

说明

App ID的申请请参考《开发指南》的“App ID鉴权介绍”的[App ID的申请](#)部分。

表 8-4 枚举 HWMSdkSiteType 说明

枚举值	描述
HWMSiteTypeChina	中国站点
HWMSiteTypeAP	亚太站点

返回值

表 8-5 返回值

类型	描述
BOOL	初始化成功返回YES, 失败返回NO。

示例代码

```
/// 初始化
HWMOpenSDKConfig *config = [[HWMOpenSDKConfig alloc] init];
config.appid = @"fdb8e4699586458bbd10c834872dcc62"; //传入申请到的企业AppId
BOOL result = [HWMSdk initWithConfig:config];
```

```
if (result) {
    NSLog(@"初始化成功");
}
```

说明

- 典型场景和接口参考中的示例代码仅作为示例用的伪代码，不能直接使用。

8.5.1.2 设置语言

setLanguage

接口描述

用于设置当前应用的语言，设置后进入会议，会议界面中的文字会使用设置的语言。

方法定义

```
+ (void)setLanguage:(HWMSdkLanguageType)language;
```

参数描述

表 8-6 参数说明

参数	是否必须	类型	描述
language	是	HWMSdkLanguageType	要设置的语言，目前只有中文和英文可以选择

表 8-7 枚举 HWMSdkLanguageType 说明

枚举值	描述
HWMSdkLanguageTypeZH	中文
HWMSdkLanguageTypeEN	英文
HWMSdkLanguageTypeCustomized	自定义语言

示例代码

```
/// 语言初始化
[HWMSdk setLanguage:HWMSdkLanguageTypeEN];
```

说明

1. 该接口建议在初始化后进行设置。
2. 该接口不支持在会议中时设置，如果进行设置，则会议中界面的文字不会立刻改变。
3. 目前只支持设置中文和英文。

setLanguage:languageCode:

接口描述

该接口用于自定义设置SDK国际化语言,默认是中文。

注意事项

实现自定义语言,要求在 en.lproj 同级目录下创建对应的国际化文件。比如新增阿拉伯语则可以创建 ar.lproj/Localizable.strings。通过该接口设置的语言应与系统当前语言保持一致

方法定义

```
+ (void)setLanguage:(HWMSdkLanguageType)language languageCode:(NSString *)languageCode;
```

参数描述

表 8-8 参数说明

参数	是否必须	类型	描述
language	是	HWMSdkLanguageType	语言类型枚举
languageCode	否	NSString *	自定义时对应的语言代码,若是中文/英文,该参数可以传空字符串

返回值

无

示例代码

```
/// 修改语言
[HWMSdk setLanguage:HWMSdkLanguageTypeCustomized languageCode:@"ar"];
```

8.5.1.3 自定义会议接听时摄像头/麦克风开关

isOpenMicrophoneIncoming

接口描述

该接口用于自定义被邀入会时麦克风的开关状态。

注意事项

1. 如果设置除主持人外的与会方入会后自动静音则该自定义无效果;

方法定义

```
- (BOOL)isOpenMicrophoneIncoming;
```

参数描述

无

示例代码

```
/// 被邀入会时是否打开麦克风
- (BOOL)isOpenMicrophoneIncoming {
    return NO;
}
```

isOpenCameraIncoming

接口描述

该接口用于自定义被邀入会时摄像头的开关状态。

注意事项

无

方法定义

```
- (BOOL)isOpenCameraIncoming;
```

参数描述

无

示例代码

```
/// 被邀入会时是否打开摄像头  
- (BOOL)isOpenCameraIncoming {  
    return NO;  
}
```

8.5.1.4 自定义本地录制路径

setLocalRecordPath

接口描述

用于自定义本地录制文件的存储路径。

注意事项

如果用户调用该接口指定了录制文件夹，则开启沙盒后不能重新创建，会写到默认路径。

方法定义

```
- (void)setLocalRecordPath:(NSURL *)url;
```

示例代码

```
NSOpenPanel *select = [NSOpenPanel openPanel];  
select.canChooseFiles = NO;  
select.canChooseDirectories = YES;  
select.prompt = @"确定";  
[select beginSheetModalForWindow:self.view.window completionHandler:^(NSModalResponse result) {  
    if (result == 1) {  
        [[HWMSdk getOpenSDK] setLocalRecordPath:select.URL];  
    }  
}];
```

8.5.1.5 开启功能

enableSDKCapability

接口描述

用于开启特性支持。

方法定义

- (void)enableSDKCapability:(HWMSDKCapabilitySupport)type;

参数描述

表 8-9 HWMSDKCapabilitySupport 说明

参数	类型	描述
HWMSDKCapabilitySupportVideoHD	NSUInteger	开启720P高清视频，默认360P。
HWMSDKCapabilitySupportShareAudio	NSUInteger	开启显示共享时包含电脑声音选项，默认不开启。
HWMSDKCapabilitySupportHideNoCameraView	NSUInteger	开启显示隐藏非视频与会者选项，默认开启。
HWMSDKCapabilitySupportBreakoutConf	NSUInteger	开启分组会议，默认开启。
HWMSDKCapabilitySupportConfirmRecord	NSUInteger	录制安全提示，默认不开启。

开启支持高清视频

示例代码

```
/// 开启支持高清视频
[[HWMBizSdk getBizOpenApi] enableSDKCapability:HWMSDKCapabilitySupportVideoHD];
```

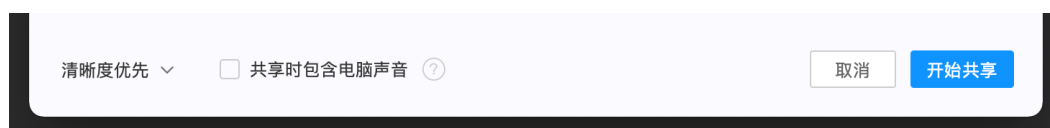
开启共享声音功能

步骤1 配置开启共享时包含电脑声音功能，调用enableSDKCapability接口

示例代码

```
/// 开启共享声音
[[HWMBizSdk getBizOpenApi] enableSDKCapability:HWMSDKCapabilitySupportShareAudio];
```

配置开启共享时包含电脑声音功能后，用户发起共享时，在开始共享窗口会显示【共享时包含电脑声音】选项。若未配置开启该功能，则窗口上不显示该选项。如下图：



步骤2 取消 App Sandbox 功能

SDK 自动安装音频驱动插件需要关闭 App Sandbox 功能，并获取 root 权限。

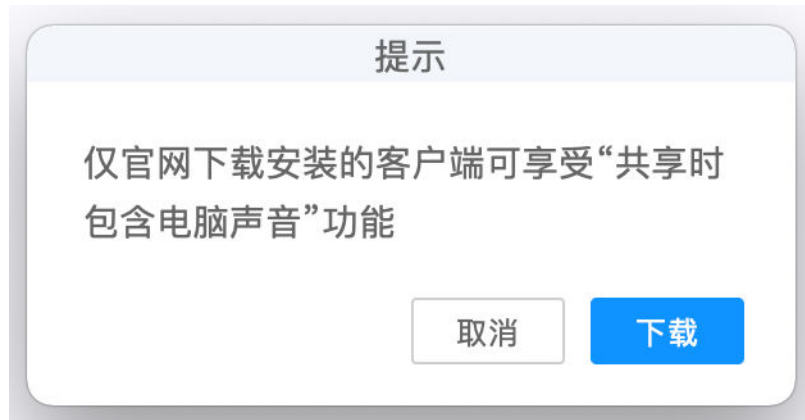
在 App 的 entitlements 描述文件中，删除 App Sandbox 条目。如下图：

Key	Type	Value
HWMUISDKNativeDemo > HWMUISDKNativeDemo > HWMUISDKNativeDemo > No Selection		
Entitlements File	Dictionary	(7 items)
App Sandbox	Boolean	YES
Audio Input	Boolean	YES
Camera	Boolean	YES
com.apple.security.files.downloads.read-write	Boolean	1
com.apple.security.files.user-selected.read-write	Boolean	1
com.apple.security.network.client	Boolean	1
com.apple.security.network.server	Boolean	1

取消 App Sandbox 功能后，首次点击勾选【共享时包含电脑声音】选项时，会获取 root 权限。在用户单击【好】后，开始自动安装音频驱动插件。如下图：



若未取消 App Sandbox 功能，首次点击勾选【共享时包含电脑声音】选项时，会弹框提示到官网下载。在用户单击【下载】后，会跳转到官网。如下图：



下载链接可通过initWithConfig接口中的appUrl字段配置，该接口用于启动并初始化 SDK，参考[8.5.1.1 初始化部分](#)。若未配置appUrl，则不显示下载。

----结束

📖 说明

App Sandbox 功能取消后，App 内获取到的用户路径会发生变化，App 会无法上架到 Mac App Store。

开启隐藏非视频与会者功能

示例代码

```
/// 开启显示隐藏非视频与会者选项
[[HWMBizSdk getBizOpenApi] enableSDKCapability:HWMSDKCapabilitySuportHideNoCameraView];
```

开启后，在会议设置中可以看到隐藏非视频与会者选项：



示例代码

```
/// 关闭隐藏非视频与会者选项
[[HWMBizSdk getBizOpenApi] disableSDKCapability:HWMSDKCapabilitySuportHideNoCameraView];
```

关闭后，会议设置不显示隐藏非视频与会者选项。

8.5.1.6 网络模式设置

setNetworkModeWithAccessType

接口描述

如华为云会议的媒体节点下沉部署到企业，并且管理节点也是通过企业内的代理访问，需要将网络模式设置成“企业内网接入”。

说明

企业内的App无法直接访问公网（即媒体节点下沉和管理节点代理下沉场景）的情况下才需要调用该接口。

注意事项

1. 该接口非必需调用，若不设置，则使用默认模式。
2. 该接口仅支持 macOS 10.13 及以上版本。

方法定义

```
/// 设置网络模式
+ (void)setNetworkModeWithAccessType:(HWMAccessType)accessType API_AVAILABLE(macos(10.13));
```

参数描述

表 8-10 参数说明

参数	是否必须	类型	描述
accessType	是	HWMAccessType	网络接入类型

表 8-11 枚举 HWMAccessType 参数说明

枚举值	描述
HWMAccessTypeAuto	默认值。
HWMAccessTypeInner	公司内网接入，用于媒体和信令下沉场景，app从内网接入，信令/媒体消息走代理服务器才能访问云会议服务端。
HWMAccessTypeOuter	公司外网接入，用于非媒体和信令下沉场景，互联网接入，信令/媒体消息可以直接访问云会议服务端。

示例代码

```
/**
 * 设置网络模式
 */
[HWMSdk setNetworkModeWithAccessType:HWMAccessTypeInner];
```

8.5.1.7 禁用提示音

disableVoicePrompts:

接口描述

禁用会议提示音。

注意事项

1. 该接口非必需调用，若不设置，则使用默认模式。
2. 返回值只代表接口是否调用成功。

方法定义

```
-(HWMSDKERR)disableVoicePrompts:(HWMSDKVoicePromptsModel *)voicePrompts;
```

参数描述

表 8-12 HWMSDKVoicePromptsModel 说明

参数	是否必须	类型	描述
disableMutePrompt	是	BOOL	主持人静音与会者提示音

返回值

接口是否调用成功

示例代码

```
HWMSDKVoicePromptsModel * model = [[HWMSDKVoicePromptsModel alloc] init];
model.disableMutePrompt = YES;
[[HWMBizSdk getBizOpenApi] disableVoicePrompts:model];
```

8.5.2 帐号操作

8.5.2.1 登录

登录分为帐号密码鉴权登录和App ID鉴权登录。

login:password:

接口描述

该接口用于通过帐号密码鉴权登录。

注意事项

1. 创建会议，被邀入会等功能只有在登录后才能使用，若不登录，将无法体验完整会议功能。
2. 回调error表示错误信息，nil表示成功。

方法定义

```
- (void)login:( NSString * _Nonnull )account password:( NSString * _Nonnull )password callback:( _Nonnull  
HWMSDKLoginCompleteHandler)callback;
```

参数描述

表 8-13 参数说明

参数	是否必须	类型	描述
account	是	NSString *	登录帐号
password	是	NSString *	登录密码
callback	否	HWMSDKLoginCompleteHandler	参数Error表示错误信息，nil表示成功。

返回值

表 8-14 登录返回 HWMLLoginResult

参数	类型	描述
userId	NSString *	账号uuid
account	NSString *	登录账号
thirdAccount	NSString *	appid登录的三方账号

示例代码

```
/// 登录处理
- (void)login {
    NSString *account = @"139571854984";
    NSString *password = @"Change_Me";
    [[HWMSdk getOpenApi] login:account password:password callback:^(NSError * _Nullable error,
    HWMLLoginResult * _Nullable result) {
        if (error == nil) {
            NSLog(@"登录成功");
        }else{
            NSLog(@"登录失败");
        }
    }
    };
```

login:

接口描述

该接口用于通过App ID鉴权登录。

注意事项

1. 创建会议，被邀入会等功能只有在登录后才能使用，若不登录，将无法体验完整会议功能。

2. 回调error表示错误信息，nil表示成功。

方法定义

```
- (void)login:(HWMAppIDLoginParam *)appidInfo callback:(HWMSDKLoginCompleteHandler)callback;
```

参数描述

表 8-15 HWMAppIDLoginParam 参数说明

参数	是否必须	类型	描述
signature	是	NSString *	从第三方服务端获取的鉴权签名信息
thirdUserId	是	NSString *	第三方用户的帐号
expireTime	是	NSInteger	signature 过期时间，单位为秒的时间戳，为0时表示不会过期
nonce	是	NSString *	随机数
userName	否	NSString *	用户名
userEmail	否	NSString *	邮箱
userPhone	否	NSString *	手机号
corpId	否	NSString *	SP下有多个企业的时候需要传入corpId
deptCode	否	NSString *	部门编码

返回值

参考：[表登录返回HWMLoginResult](#)

示例代码

```
/// AppId 登录接口
- (void)login {
    HWMAppIDLoginParam *appidInfo = [[HWMAppIDLoginParam alloc] init];
    appidInfo.thirdUserId = @"test";
    appidInfo.expireTime = 1599618620;
    appidInfo.nonce = @"0vjeeKUvztUu6N20lqfooegDVGi5Kf30";
    appidInfo.userName = @"测试帐号";
    appidInfo.signature = @"fac06749023bfc887b3b5fa030cc3dbacb2c95a3fed85664bae55ba334f9c";
    [[HWM SDK getOpenApi] login:appidInfo callback:^(NSError * _Nullable error, HWMLoginResult *
    _Nullable result) {
        if (error == nil) {
            NSLog(@"登录成功");
        }else{
            NSLog(@"登录失败");
        }
    }
    }];
}
```

loginBySSO:

接口描述

该接口用于通过SSO的方式登录和鉴权。

注意事项

除匿名入会外，在使用其他业务之前必须进行成功登录。

方法定义

```
- (void)loginBySSO:(HWMSSOLoginParam *)param callback:(HWMSDKLoginCompleteHandler)callback;
```

参数描述

表 8-16 HWMSSOLoginParam 参数说明

参数	是否必须	类型	描述
domain	是	NSString *	企业域名
userTicket	是	NSString *	code
authServerType	是	HWMOpenAuthServerType	鉴权服务器类型

表 8-17 枚举 HWMOpenAuthServerType 说明

值	描述
HWMOpenAuthServerTypeOauth2	单点登录

返回值

参考：[表登录返回HWMLoginResult](#)

示例代码

```
/// SSO 登录接口
- (void)login {
    HWMSSOLoginParam *param = [[HWMSSOLoginParam alloc] init];
    param.domain = domain;
    param.code = code;
    param.authServerType = HWMOpenAuthServerTypeOauth2;
    [[HWMSdk getOpenApi] loginBySSO:param callback:^(NSError * _Nullable error, HWMLoginResult * _Nullable result) {
        if (error == nil) {
            NSLog(@"登录成功");
        }else{
            NSLog(@"登录失败");
        }
    }];
}
```

8.5.2.2 退出登录

logout

接口描述

该接口用于退出已登录的会议帐号。

注意事项

1. 请在已登录的场景下调用。
2. 回调error表示错误信息，nil表示成功。。

方法定义

```
- (void)logout:(Nonnull HWMSDKCompleteHandler)callback;
```

参数描述

无。

返回值

表 8-18 退出返回 HWMLogoutResult

参数	类型	描述
userId	NSString *	账号uuid
account	NSString *	登录账号
thirdAccount	NSString *	appid登录的三方账号

示例代码

```
/// 退出登录
- (void)logout {
    [[HWMSdk getOpenApi] logout:^(NSError * _Nullable error, HWMLogoutResult* _Nullable result) {
        if (error == nil) {
            NSLog(@"登出成功");
        } else {
            NSLog(@"登出失败");
        }
    }
};
}
```

8.5.2.3 上传个人头像

uploadSelfAvatar: success: fail:

接口描述

该接口用于上传用户自己头像。

注意事项

1. 传入头像文件路径path。
2. 大小限制200k。
3. 只支持jpg、png。
4. 为了头像展示效果，建议图片宽高比为1:1。
5. 回调success表示成功，fail表示失败，error错误信息。

方法定义

```
- (void)uploadSelfAvatar:(NSString *)path success:(void (^)(void))success fail:(void (^)(NSError *error))fail;
```

参数描述

表 8-19

参数	是否必须	类型	描述
path	是	NSString	图片的全路径

返回值

无

示例代码

```
/// 上传用户个人头像
- (void)uploadSelfAvatar {
    [[HWMBizSdk getBizOpenApi] uploadSelfAvatar:path success:^(
        NSLog(@"set succeeded");
    ) fail:^(NSError *_Nonnull error) {
        NSLog(@"%@", error.description);
    }];
}
```

8.5.3 界面定制

用户可以根据需求自定义一些界面

8.5.3.1 界面配置

设置自定义界面的代理，根据需求在对应位置设置代理。可以在初始化SDK的时候配置，也可以在使用时配置。

```
[HWMSdk getSdkConfig].socialShareHandler = 代理对象; // 可选，社交分享定制代理，用于定制会中分享能力
[HWMSdk getSdkConfig].confUIHandler = 代理对象; // 可选，会中部分UI代理
[HWMSdk getSdkConfig].contactUIHandler = 代理对象; // 可选，会中邀请联系人UI 定制代理
[HWMSdk getSdkConfig].headPortraitApiHandler = 代理对象; // 可选，头像拦截显示定制代理
```

8.5.3.2 清除界面配置

设置相关的代理后如果需要清除对应代理，可以置为nil，根据需求在对应位置设置。

```
[HWMSdk getSdkConfig].socialShareHandler = nil;
[HWMSdk getSdkConfig].confUIHandler = nil;
[HWMSdk getSdkConfig].contactUIHandler = nil;
[HWMSdk getSdkConfig].headPortraitApiHandler = nil;
```

8.5.3.3 底部工具栏定制

meetingToolBarItemsForScene

接口描述

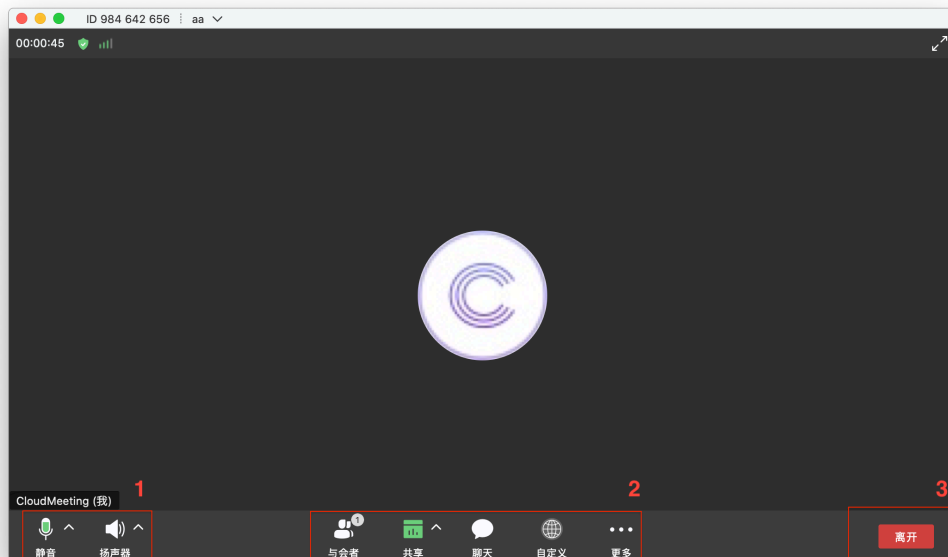
该接口用于构建底部toolbar的会控全量菜单。

下图中1区域内支持调整按钮展示顺序或者隐藏按钮。

下图中2区域支持最多8个按钮，多余按钮将不会展示。

下图中3区域按钮仅支持隐藏/显示。

图 8-8 视频会议会控菜单定制位置



注意事项

- 1、返回的全量数组中，如果隐藏某个区域的按钮，需使用空数组(@[])占位;
- 2、传入的按钮如果当前场景不支持或当前角色不支持将不会被展示(如普通与会者角色将不会展示邀请功能。)

方法定义

```
- (nullable NSArray <NSArray <HWMToolBarItemModel * > * >)meetingToolBarItemsForScene:
(HWMConfToolBarScene)scene;
```

参数描述

scene 为需要定制底部工具栏的场景，各场景对应取值如下：

- 语音呼叫：HWMConfToolBarSceneAudioCall;
- 视频呼叫：HWMConfToolBarSceneVideoCall;
- 语音会议：HWMConfToolBarSceneAudioConf;
- 视频会议：HWMConfToolBarSceneVideoConf.

返回值

无

示例代码

以视频会议为例：

以下代码隐藏了视频按钮，并添加自定义按钮。效果如图1所示。

```

- (nullable NSArray <NSArray <HWMToolBarItemModel * > * > *)meetingToolBarItemsForScene:
(HWMConfToolBarScene)scene {
    // 构建视频会议底部工具栏按钮
    if (scene == HWMConfToolBarSceneVideoConf) {
        // 麦克风
        HWMToolBarItemModel *micItem = [HWMToolBarItemModel defaultMicrophoneItem];
        // 扬声器
        HWMToolBarItemModel *speakerItem = [HWMToolBarItemModel defaultSpeakerItem];
        // 与会者
        HWMToolBarItemModel *participantItem = [HWMToolBarItemModel defaultParticipantItem];
        // 共享
        HWMToolBarItemModel *shareItem = [HWMToolBarItemModel defaultDataShareItem];
        // 聊天
        HWMToolBarItemModel *chatItem = [HWMToolBarItemModel defaultChatItem];
        // 自定义
        HWMToolBarItemModel *customItem = [[HWMToolBarItemModel alloc] init];
        customItem.title = @"自定义";
        customItem.selectedTitle = @"选中自定义";
        customItem.iconImage = [UIImage imageNamed:@"icon-customize-select-normal"];
        customItem.selectedIconImage = [UIImage imageNamed:@"icon-customize-select-selected"];
        customItem.disabledIconImage = [UIImage imageNamed:@"icon-customize-select-normal"];
        // 更多
        HWMToolBarItemModel *moreItem = [HWMToolBarItemModel defaultMoreItem];
        // 离开
        HWMToolBarItemModel *leaveItem = [HWMToolBarItemModel defaultLeaveItem];

        // 左边区域
        NSArray *leadingArray = @[micItem, speakerItem];
        // 中间区域
        NSArray *centerArray = @[participantItem, shareItem, chatItem, customItem, moreItem];
        // 右边区域
        NSArray *trailingArray = @[leaveItem];

        // 返回全量数组 需要隐藏某个区域全部按钮时 需使用空数组@[ ]占位
        return @[leadingArray, centerArray, trailingArray];
    }
    return nil;
}

```

8.5.3.4 共享工具栏定制

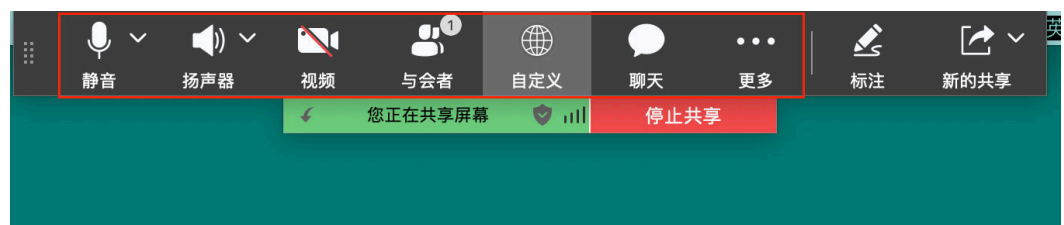
screenShareToolBarItems

接口描述

该接口用于共享工具栏左侧区域按钮的定制。

下图中框选部分按钮最多可定制8个按钮，多余按钮将不会被展示。

图 8-9 共享按钮定制位置



注意事项

传入的按钮如果当前场景不支持或当前角色不支持将不会被展示；

方法定义

```
- (nullable NSArray <HWMToolBarItemModel *> *)screenShareToolBarItems;
```

参数描述

无

返回值

无

示例代码

```
// 构建共享工具栏按钮
- (nullable NSArray <HWMToolBarItemModel *> *)screenShareToolBarItems {
    // 麦克风
    HWMToolBarItemModel *micItem = [HWMToolBarItemModel defaultMicrophoneItem];
    // 扬声器
    HWMToolBarItemModel *speakerItem = [HWMToolBarItemModel defaultSpeakerItem];
    // 摄像头
    HWMToolBarItemModel *cameraItem = [HWMToolBarItemModel defaultCameraItem];
    // 与会者
    HWMToolBarItemModel *participantItem = [HWMToolBarItemModel defaultParticipantItem];
    // 自定义
    HWMToolBarItemModel *customItem = [[HWMToolBarItemModel alloc] init];
    customItem.title = @"自定义";
    customItem.selectedTitle = @"选中自定义";
    customItem.iconImage = [NSImage imageNamed:@"icon-customize-select-normal"];
    customItem.selectedIconImage = [NSImage imageNamed:@"icon-customize-select-selected"];
    customItem.disabledIconImage = [NSImage imageNamed:@"icon-customize-select-normal"];
    // 聊天
    HWMToolBarItemModel *chatItem = [HWMToolBarItemModel defaultChatItem];
    // 更多
    HWMToolBarItemModel *moreItem = [HWMToolBarItemModel defaultMoreItem];

    return @[micItem, speakerItem, cameraItem, participantItem, customItem, chatItem, moreItem];
}
```

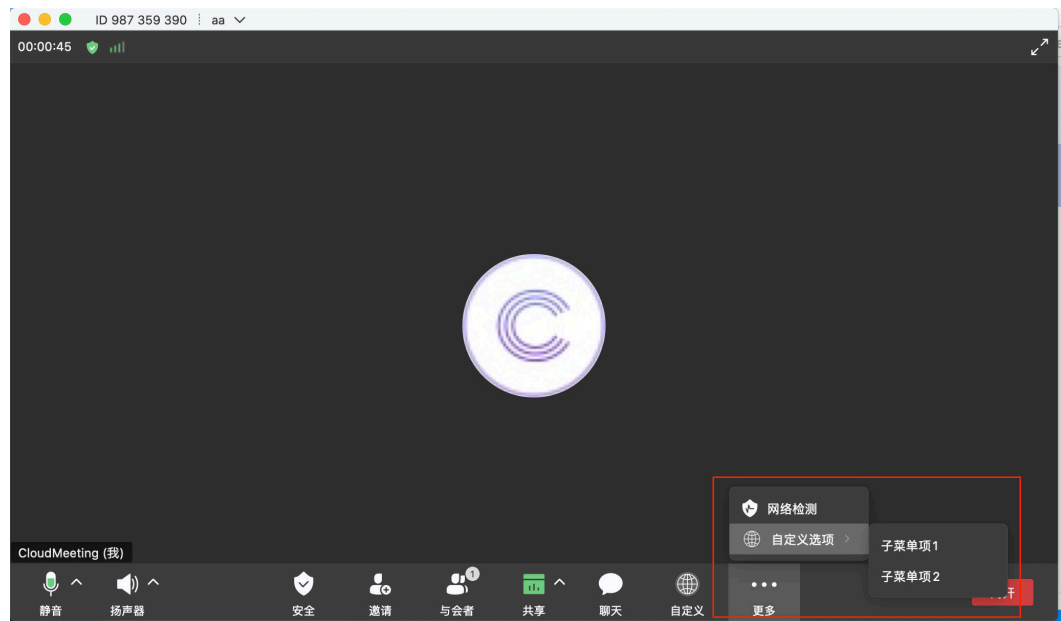
8.5.3.5 底部工具栏更多菜单定制

meetingToolBarMoreMenuItems

接口描述

该接口用于构建会控工具栏更多菜单选项。

图 8-10 会控工具栏更多菜单选项定制位置



注意事项

无

方法定义

```
- (nullable NSArray <HWMMenuItemModel *> *)meetingToolbarMoreMenuItems;
```

参数描述

无

返回值

无

示例代码

```
// 构建底部工具栏更多菜单
- (nullable NSArray <HWMMenuItemModel *> *)meetingToolbarMoreMenuItems {
    // 网络检测
    HWMMenuItemModel *netWorkDetectionItem = [HWMMenuItemModel
defaultNetworkDetectionModel];
    // 自定义选项
    HWMMenuItemModel *customItem = [[HWMMenuItemModel alloc] init];
    customItem.title = @"自定义选项";
    customItem.image = [NSImage imageNamed:@"icon-customize-select-normal"];
    HWMMenuItemModel *subItem1 = [[HWMMenuItemModel alloc] init];
    subItem1.title = @"子菜单项1";
    HWMMenuItemModel *subItem2 = [[HWMMenuItemModel alloc] init];
    subItem2.title = @"子菜单项2";
    customItem.submenus = @[subItem1, subItem2];
    return @[netWorkDetectionItem, customItem];
}
```

8.5.3.6 共享工具栏更多菜单定制

screenShareToolbarMoreMenuItems

接口描述

该接口用于构建共享工具栏更多菜单选项

图 8-11 共享工具栏更多菜单选项定制位置



注意事项

无

方法定义

```
- (nullable NSArray <HWMMenulItemModel *>)screenShareToolbarMoreMenuItems;
```

参数描述

无

返回值

无

示例代码

```
// 构建共享工具栏更多菜单选项
- (nullable NSArray <HWMMenulItemModel *>)screenShareToolbarMoreMenuItems {
    // 网络检测
    HWMMenulItemModel *networkDetectionItem = [HWMMenulItemModel
defaultNetworkDetectionModel];
    // 自定义选项
    HWMMenulItemModel *customItem = [[HWMMenulItemModel alloc]init];
    customItem.title = @"自定义选项";
    customItem.image = [UIImage imageNamed:@"icon-customize-select-normal"];
    HWMMenulItemModel *subItem1 = [[HWMMenulItemModel alloc]init];
    subItem1.title = @"子菜单项1";
    HWMMenulItemModel *subItem2 = [[HWMMenulItemModel alloc]init];
    subItem2.title = @"子菜单项2";
    customItem.submenus = @[subItem1,subItem2];
    // 离开
    HWMMenulItemModel *leaveItem = [HWMMenulItemModel defaultLeaveMeetingModel];

    return @[networkDetectionItem, customItem, leaveItem];
}
```

8.5.3.7 社交分享定制

```
- (NSView *)socialShareView:(HWMMConfStateInfo *)confInfo;
```

接口描述

该接口用于分享会议拦截处理。实现代理后会拦截SDK调起分享界面的方法，如下图红框标注。

图 8-12 分享拦截 SDK 弹窗



注意事项

1. 如果实现该协议方法，则会中调用分享功能，会回调到该方法。窗口的位置以及界面的显示隐藏都有SDK内部控制。

示例代码

```
- (NSView *)socialShareView:(HWMConfStateInfo *)confDetail {
    if (!self.socialView) {
        self.socialView = [[HWMSocialView alloc] initWithFrame:NSMakeRange(0, 0, 500, 300)];
        self.socialView.wantsLayer = YES;
        self.socialView.layer.backgroundColor = [NSColor whiteColor].CGColor;
    }
    self.socialView.confDetail = confDetail;
    return self.socialView;
}
```

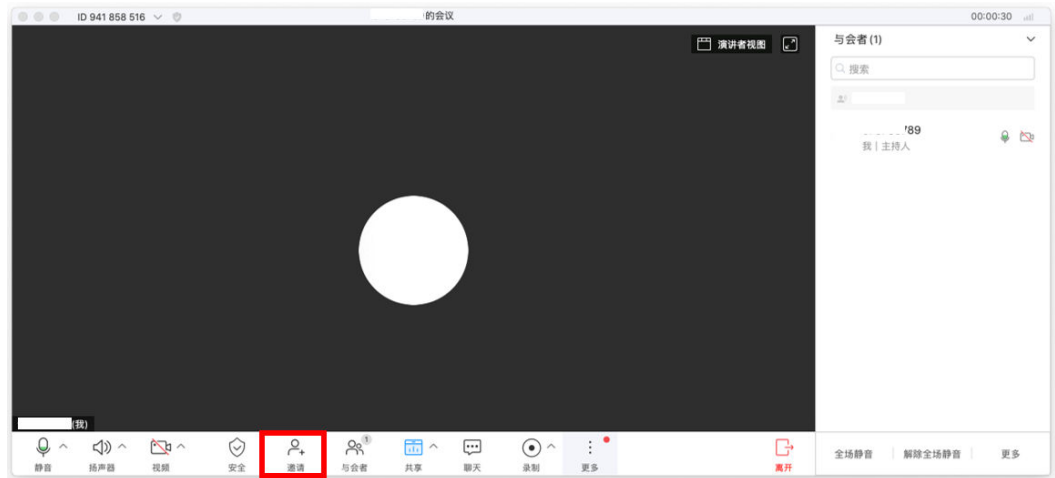
8.5.3.8 选择联系人定制

openContactSelectWindow:scene:completeHandler:

接口描述

该接口用于定制打开联系人选择窗口。

图 8-13 定制选择联系人触发位置



注意事项

无

方法定义

```
- (void)openContactSelectWindow:(NSArray<HWMAttendeeInfo *> *)param scene:  
(HWMSelectedContactScene)scene completionHandler:(void (^)(NSArray<HWMAttendeeInfo *> *  
_Nullable))handler ;
```

参数描述

无

返回值

无

示例代码

```
- (void)openContactSelectWindow:(NSArray<HWMAttendeeInfo *> *)param scene:  
(HWMSelectedContactScene)scene completionHandler:(void (^)(NSArray<HWMAttendeeInfo *> *  
_Nullable))handler {  
    HWMSelectContactViewController *contact = [[HWMSelectContactViewController alloc] init];  
    [contact setSelectAttendeeInfoHandler:^(NSArray *attendees) {  
        if (handler) {  
            handler(attendees);  
        }  
    }];  
    self.selectContactVC = contact;  
    [self presentViewControllerAsModalWindow:contact];  
}
```

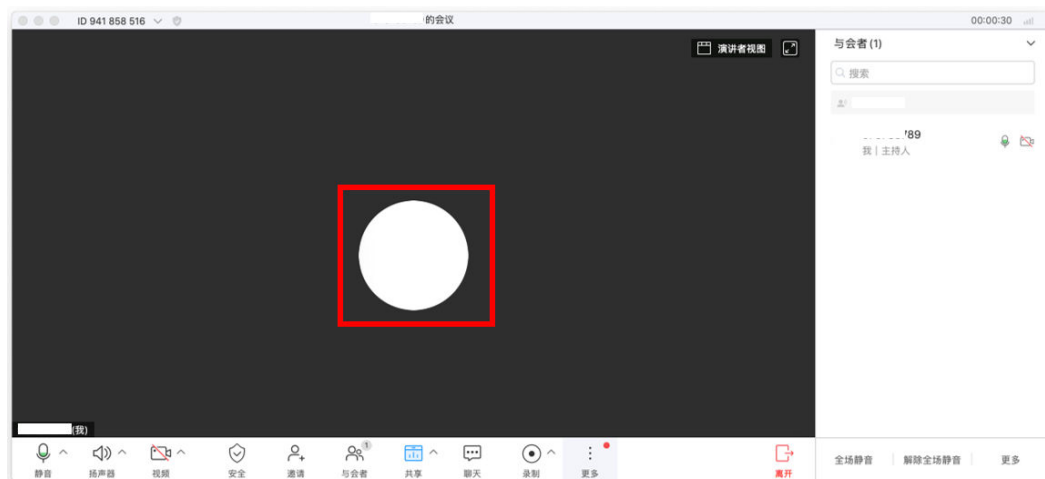
8.5.3.9 头像拦截显示

**downloadHeadPortraitWithThirdAccount: account: number:
completionBlock:**

接口描述

该接口用于定制登录后的头像显示。

图 8-14 生效后拦截头像显示效果



注意事项

1. 未登录的头像不支持自定义。
2. 头像支持两种情况：一种主动上传，见上传个人头像接口；一种为注入实现，若两种都实现只会进行拦截的实现，不会从服务端下载上传的头像。

方法定义

```
- (void)downloadHeadPortraitWithThirdAccount:(NSString *)thirdAccount account:(NSString *)account
number:(NSString *)number completionBlock:(void (^)(NSError * _Nullable, NSString *
_Nullable))completionBlock
```

参数描述

表 8-20

参数	类型	描述
thirdAccount	NSString	appid登录场景下优先使用此字段
account	NSString	帐号密码登录场景下 优先使用此字段
number	NSString	跨企业场景优先使用此字段

返回值

无

示例代码

```
- (void)downloadHeadPortraitWithThirdAccount:(NSString *)thirdAccount account:(NSString *)account
number:(NSString *)number completionBlock:(void (^)(NSError * _Nullable, NSString *
_Nullable))completionBlock {
    if (completionBlock) {
        /// 将图片路径返回给SDK
        NSInteger index = arc4random() % 3;
```



```
NSString *path = [[NSBundle bundleForClass:[self class]] pathForResource:[NSString
stringWithFormat:@"head_icon_%lu", index] ofType:@"png"];

    completionBlock(nil, path);
}
}
```

8.5.3.10 其它 UI 定制

videoWindowDockIcon

接口描述

该接口用于定制视频通话时，视频窗口最小化dock栏图标以及共享情况下吸附窗口显示的图标。

方法定义

```
-(NSImage *)videoWindowDockIcon;
```

参数描述

无

返回值

无

示例代码

```
-(NSImage *)videoWindowDockIcon {
    return [NSImage imageNamed:@"iconMeeting"];
}
```

8.5.3.11 自定义会中水印

setWaterMark

接口描述

该接口用于自定义共享时的水印。

方法定义

```
-(void)setWaterMark:(nullable NSAttributedString *)waterMarkString;
```

参数描述

水印要展示的内容。

返回值

无

示例代码

```
NSAttributedString *waterMarkString = [[NSAttributedString alloc] initWithString:@"水印内容"];
[[HWMSSdk getOpenApi] setWaterMark:waterMarkString];
```

8.5.3.12 自定义设置界面

buildSettingTabItems

接口描述

该接口用于自定义设置界面中各标签页和功能区的显示隐藏，若未进行设置，则默认都显示。

图 8-15 自定义设置界面



方法定义

```
-(NSArray <HWMSettingTabItem *> *)buildSettingTabItems;
```

参数描述

表 8-21 HWMSettingTabItem

参数	类型	描述
type	HWMSettingTabType	标签页类型
text	NSString	标题
image	NSImage	图片
hoverImage	NSImage	鼠标移入图片

参数	类型	描述
settingMenuItems	NSArray	右侧功能列表

表 8-22 HWMSettingMenuItem

参数	类型	描述
type	HWMSettingMenuItemType	功能类型
text	NSString	标题

表 8-23 HWMSettingTabType 说明

参数	类型	描述
HWMSettingTabNormal	NSUInteger	常规设置页
HWMSettingTabVideo	NSUInteger	视频设置页
HWMSettingTabAudio	NSUInteger	音频设置页
HWMSettingTabRecord	NSUInteger	录制设置页
HWMSettingTabHotKey	NSUInteger	快捷键设置页

表 8-24 HWMSettingMenuItemType 说明

参数	类型	描述
HWMSettingMenuItemLogin	NSUInteger	登录设置
HWMSettingMenuItemCall	NSUInteger	会议呼叫设置
HWMSettingMenuItemChatNotify	NSUInteger	聊天消息提醒设置
HWMSettingMenuItemSubtitle	NSUInteger	字幕设置
HWMSettingMenuItemMicrophone	NSUInteger	麦克风设置
HWMSettingMenuItemSpeaker	NSUInteger	扬声器设置
HWMSettingMenuItemAdvanced	NSUInteger	高级设置
HWMSettingMenuItemCamera	NSUInteger	摄像头设置

参数	类型	描述
HWMSettingMenuLayout	NSUInteger	画面设置
HWMSettingMenuRecord	NSUInteger	录制设置
HWMSettingMenuHotKey	NSUInteger	快捷键设置

返回值

无

示例代码

```

- (NSArray <HWMSettingTabItem *> *)buildSettingTabItems{
    HWMSettingTabItem *normalTab = [HWMSettingTabItem defaultTabNormal];
    HWMSettingTabItem *videoTab = [HWMSettingTabItem defaultTabVideo];
    HWMSettingTabItem *audioTab = [HWMSettingTabItem defaultTabAudio];
    HWMSettingTabItem *recordTab = [HWMSettingTabItem defaultTabRecord];
    HWMSettingTabItem *hotKeyTab = [HWMSettingTabItem defaultTabHotKey];
    return @[normalTab, videoTab, audioTab, recordTab, hotKeyTab];
};

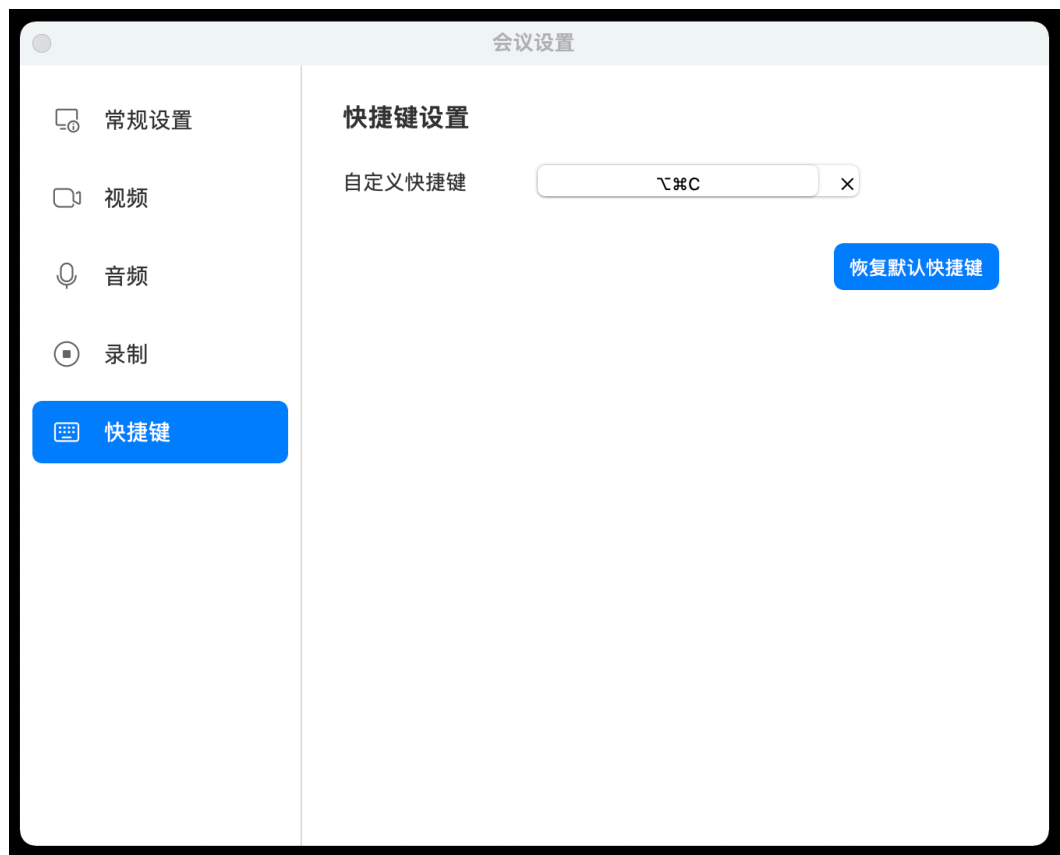
```

buildSettingHotKeyItems

接口描述

该接口用于自定义设置窗口中快捷键标签页，若未进行设置，则默认不显示。

图 8-16 自定义快捷键



方法定义

```
-(NSArray<HWMSettingHotKeyItem *> *)buildSettingHotKeyItem;
```

参数描述

表 8-25 HWMSettingHotKeyItem

参数	类型	描述
type	HWMSettingHotKeyType	快捷键类型
keyCode	NSInteger	快捷键默认值
keyName	NSString	快捷键key值
text	NSString	快捷键标题
action	dispatch_block_t	快捷键响应事件

表 8-26 结构体 HWMSettingHotKeyType 参数说明

参数	类型	描述
HWMSettingHotKeyOpenMain	NSUInteger	打开主面板
HWMSettingHotKeyActivateSearch	NSUInteger	激活搜索

返回值

无

示例代码

```
-(NSArray<HWMSettingHotKeyItem *> *)buildSettingHotKeyItems {
    HWMSettingHotKeyItem *item = [[HWMSettingHotKeyItem alloc] initWithType:2 text:@"自定义快捷键"
    keyCode:kVK_ANSI_C keyName:@"HWMHotKeyCustomKey"];
    item.action = ^{
        NSLog(@"action");
    };
    return @[item];
};
```

8.5.3.13 联系人信息定制

queryUserDetailByNumbers: completionBlock:

接口描述

该接口用于联系人信息注入。

注意事项

1. 使用时需要订阅HWMContactApiHandler代理对象

图 8-17 tooltip 显示注入的部门信息

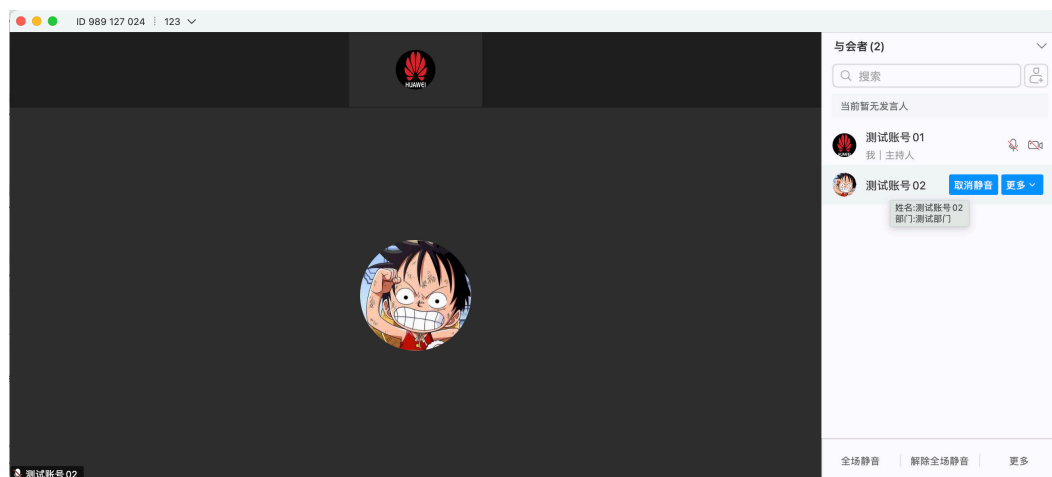
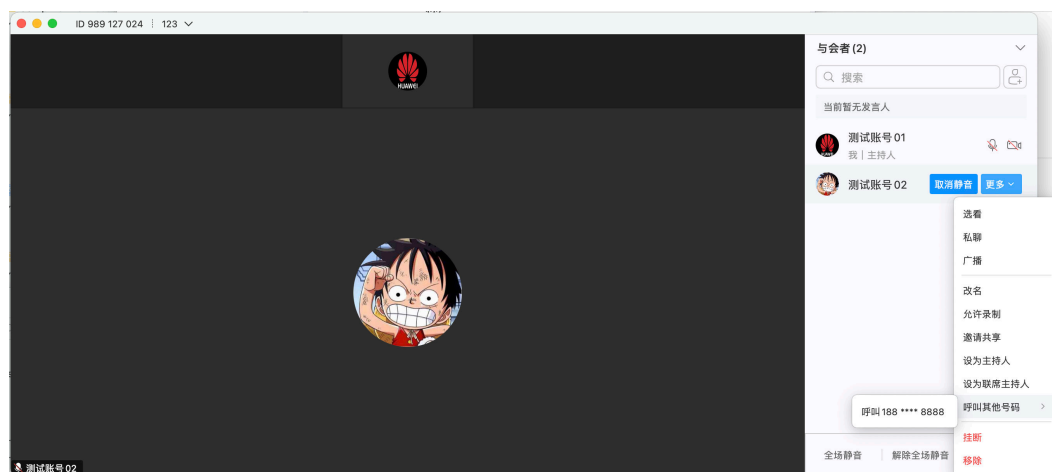


图 8-18 手机号码的自定义展示形式



方法定义

```

/// 通过number ( sip号 ) 查询联系人
/// @param numbers number 集合
/// @param completionBlock 结果回调 若NSArray中有值 error不会返回错误
/// 需要将查询到的结果 ( userInfos ) 回传给sdk
- (void)queryUserDetailByNumbers:(NSArray<NSString *> *)numbers
    completionBlock:(void (^)(NSError *_Nullable error, NSArray<HWMEnterpriseContact *>
*_Nullable userInfos))completionBlock;
    
```

参数描述

表 8-27

参数	类型	描述
numbers	NSArray	需要注入的联系人 sip 号

返回值

无

示例代码

```
- (void)queryUserDetailByNumbers:(NSArray<NSString *> *)numbers completionBlock:(void (^)(NSError *  
_Nullable, NSArray<HWMEnterpriseContact *> *_Nullable))completionBlock {  
    NSArray *data = [self searchContactsWithKeys:numbers isNumber:YES];// 通过numbers数组查询到的联系  
    人数组  
    completionBlock(nil,data); // 将数据源回传至SDK  
}
```

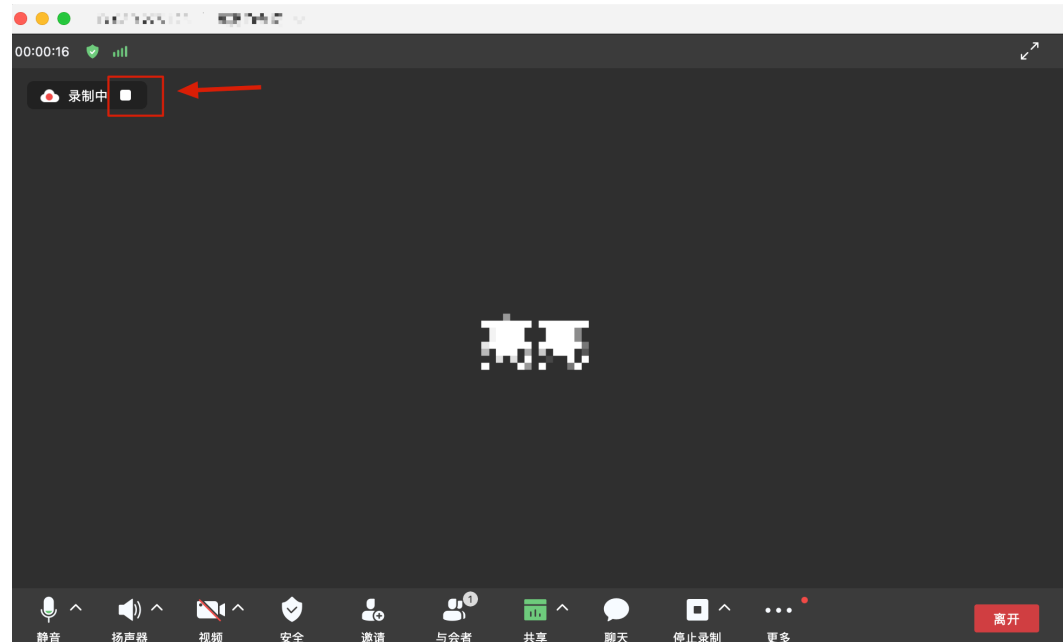
8.5.3.14 录制按钮操作定制

buildRecordMenuItems

接口描述

该接口用于定制录制状态栏旁的录制按钮是否隐藏。

图 8-19 可定制位置



注意事项

1. 暂时只支持录制右侧按钮隐藏，不返回item就不显示。

方法定义

```
- (NSArray <HWMStatusBarItemModel *> *)buildRecordMenuItems;
```

参数描述

无

返回值

无

示例代码

```

/// 录制按钮菜单栏定制
- (NSArray <HWMStatusBarItemModel *> *)buildRecordMenuItems {
    HWMStatusBarItemModel *recordItem = [WKCFStatusBarItemModel defaultStatusBarRecordMenu];
    // 如果不需要录制按钮右侧的操作按钮，数组返回为空即可
    return @[recordItem];
}
    
```

8.5.3.15 显示设置窗口

showSettingWindowWithInfo

接口描述

该接口显示设置窗口。

方法定义

```
- (void)showSettingWindowWithInfo:(HWMUISDKShowSettingWindowInfoModel *)info;
```

参数描述

表 8-28 参数说明

参数	是否必须	类型	描述
info	是	HWMUISDKShowSettingWindowInfoModel	显示设置窗口信息

表 8-29 结构体 HWMUISDKShowSettingWindowInfoModel 参数说明

参数	类型	描述
isShow	BOOL	是否显示
defaultPage	HWMUISDKSettingPage	默认显示标签页

表 8-30 枚举 HWMUISDKSettingPage 说明

枚举值	描述
HWMUISDKSettingPageNormal	基本设置页
HWMUISDKSettingPageVideo	视频设置页
HWMUISDKSettingPageAudio	音频设置页
HWMUISDKSettingPageRecord	录制设置页

枚举值	描述
HWMUISDKSettingPageHotkey	快捷键设置页
HWMUISDKSettingPageScreen	屏幕设置页
HWMUISDKSettingPageCount	设置页数量

返回值

无

示例代码

```
- (void)settingEvent:(NSButton *)button {
    if (![HWMsSdk hasInit]) {
        // 初始化后可以调用
        return;
    }
    HWMUISDKShowSettingWindowInfoModel *model = [[HWMUISDKShowSettingWindowInfoModel alloc]
init];
    model.isShow = YES;// 设置是否显示
    model.defaultPage = HWMUISDKSettingPageNormal;// 设置显示标签页
    [[HWMBizSdk getBizOpenApi] showSettingWindowWithInfo:model];
}
```

8.5.3.16 与会者列表子项定制

onClickInjectParticipantItemBtnNotify

接口描述

该接口用于定制与会者列表的子项点击事件。

方法定义

```
- (void)onClickInjectParticipantItemBtnNotify:(HWMUISDKInjectParticipantItemBtnInfoModel *)injectBtnInfo;
```

参数描述

表 8-31 参数说明

参数	是否必须	类型	描述
injectBtnInfo	是	HWMUISDKInjectParticipantItemBtnInfoModel	显示设置窗口信息

表 8-32 结构体 HWMUISDKInjectParticipantItemBtnInfoModel 参数说明

参数	类型	描述
injectBtn	HWMUISDKClickInjectParticipantItemBtn	按钮类型
participantInfo	HWMUISDKParticipantItemInfoModel	与会者子项信息
btnId	NSString *	按钮Id

表 8-33 枚举 HWMUISDKClickInjectParticipantItemBtn 说明

枚举值	描述
HWMUISDKClickInjectParticipantItemBtnCallOtherNum	呼叫其他号码按钮
HWMUISDKClickInjectParticipantItemBtnParticipantDetail	查看与会者详情

表 8-34 结构体 HWMUISDKParticipantItemInfoModel 参数说明

参数	类型	描述
number	NSString *	号码
nickName	NSString *	与会者昵称
thirdUserId	NSString *	与会者三方帐号，appId登录时使用
accountId	NSString *	华为云会议帐号

返回值

无

示例代码

```

- (void)onClickInjectParticipantItemBtnNotify:(HWMUISDKInjectParticipantItemBtnInfoModel *)injectBtnInfo
{
    // 收到通知后，对上报信息进行处理，示例代码处理方式为打印上报信息
    NSString *info = [injectBtnInfo yy_modelDescription];
    [HWMConsoleManager sharedInstance].console = [NSString stringWithFormat:@"participant: %@", info];
}

```

8.5.4 会议控制

8.5.4.1 创建会议

createConf

接口描述

该接口用于创建即时会议。

注意事项

1. 该接口默认已将自己作为主席加入会议，调用接口时，在与会者参数中不需要带上自己的信息。
2. 若需要在创建会议的同时邀请其他与会者，需要传入其他与会者信息。
3. 创会结果在callback中返回。

方法定义

```
- (void)createConf:(HWMCreateConfParam * _Nonnull)param callback:( _Nonnull  
HWMSDKCreateConfCompleteHandler)callback;
```

参数描述

表 8-35 HWMCreateConfParam 参数说明

参数	是否必须	类型	描述
subject	是	NSString *	会议主题。
members	否	NSArray <HWMAttendeeInfo*> *	成员列表，可选
isCameraOn	否	BOOL	是否打开摄像头，默认关闭
isMicOn	否	BOOL	是否打开麦克风，默认打开
isAutoRecord	否	BOOL	会议开始后是否自动开启录制，默认关闭。开启自动录制会议时必须开启录制会议
vmrId	否	NSString *	个人会议或者云会议室Id，创建个人会议或云会议室的时候传入，其它时候传入空
callInRestrictionType	否	HWMJoinConfRestrictionType	允许入会范围限制
noPassword	否	BOOL	会议是否无密码，默认为否（只针对随机VMR会议有效）

参数	是否必须	类型	描述
guestPwd	否	NSString	指定来宾入会密码，默认空表示随机生成密码（只针对随机VMR会议有效）
isVideo	否	BOOL	是否创建视频会议，默认是YES。选NO则创建语音会议
vmrConfIdType	否	HWMVmrConfIdType	云会议室的会议ID类型，默认固定类型
isOpenWaitingRoom	否	BOOL	是否开启等候室 说明 需要等候室功能开通后才能生效。

表 8-36 HWMAttendeeInfo 参数说明

参数	是否必须	类型	描述
name	是	NSString	与会者名字
number	否	NSString	Sip呼叫号码或者电话号码（账号、密码鉴权场景情况下必填）
thirdUserId	否	NSString	APPID登录场景下的第三方账号（第三方用户帐号AppID鉴权方式下必填）
accountId	否	NSString	用户登录帐号（若为用户名/密码登录场景 必填）
email	否	NSString	电子邮箱地址
sms	否	NSString	短信通知手机号码
mute	否	BOOL	是否闭音
role	否	HWMConfRoleType	会议成员角色

返回值

无

示例代码

```

/// 创建会议
- (void)creatMeeting {
    HWMCreateConfParam *param = [[HWMCreateConfParam alloc] init];
    param.subject = @"会议标题";
    param.isCameraOn = YES;// 是否打开摄像头, 默认关闭
    param.isMicOn = YES;// 是否打开麦克风, 默认打开
    param.isAutoRecord = NO;// 是否打开会议自动录制, 默认关闭
    param.joinConfRestrictionType = HWMJoinConfRestrictionAll;// 允许入会范围

    [[HWM SDK getOpenApi] createConf:param callback:^(NSError * _Nullable error, HWMCreateConfResult *
    _Nullable result) {
        [self hideLoading];
        if (error) {
            NSLog(@"创会失败%@", error.localizedDescription);
        }else{
            NSLog(@"创会成功");
        }
    }
    }];
}
    
```

8.5.4.2 加入会议

joinConf

接口描述

该接口用于通过会议ID和密码加入已经存在的会议。加入会议的方式有两种：

- 会议ID+会议密码入会：手动输入会议ID和会议密码密码。
- 会议ID+Random入会：通过邮件会议通知中的会议链接入会时（会议链接如：<https://bmeeting.huaweicloud.com/#/j/981924330/e11ddec62ee760734fcf2ba55b79937eac2aa68677caf659>），链接里面的的是会议ID（即举例中的981924330）和会议的Random码（即举例中的e11ddec62ee760734fcf2ba55b79937eac2aa68677caf659）。

注意事项

1. ID入会参数使用子类HWMJoinConfParam，random入会参数使用子类HWMRandomConfParam
2. 在登录和未登录状态下都可以调用该接口。
3. 实际业务处理结果在对应的callback回调中。

方法定义

```

- (void)joinConf:(HWMBaseJoinConfParam * _Nonnull)param callback:( _Nonnull
HWM SDKCompleteHandler)callback;
    
```

参数描述

表 8-37 HWMBaseJoinConfParam

参数	是否必须	类型	描述
confId	是	NSString *	会议ID。

参数	是否必须	类型	描述
nickname	否	NSString *	会议昵称。在已登录状态下可为空，已登录时空将使用登录的用户名作为会场名。
isCameraOn	否	BOOL	是否打开摄像头，默认关闭。
isMicOn	否	BOOL	是否打开麦克风，默认打开。
isSpkOn	否	BOOL	是否打开扬声器，默认打开

表 8-38 HWMJoinConfParam 参数说明，继承 HWMBaseJoinConfParam

参数	是否必须	类型	描述
password	否	NSString *	会议密码。若希望以主席身份入会，必须传入主席密码。若希望以来宾身份入会，会议要求来宾密码时，必须传入来宾密码，会议不要求来宾密码时，可以为空。
isStopConflictConf	否	BOOL	强制结束和当前VMR会议资源冲突的会议（必需是vmr资源的拥有者才有权限，否则无效）

表 8-39 HWMRandomConfParam 参数说明，继承 HWMBaseJoinConfParam

参数	是否必须	类型	描述
random	是	NSString *	random入会时传入。

示例代码

```

/// 加入会议
- (void)joinMeeting {
    HWMJoinConfParam *param = [[HWMJoinConfParam alloc] init];
    param.confId = @"989156631";
    param.password = @"807766";
    param.nickname = @"杭州会场";
    param.isCameraOn = NO;
    param.isMicOn = YES;
    [[HWM SDK getOpenApi] joinConf:param callback:^(NSError * _Nullable error, id _Nullable result) {
        if (error) {
            NSLog(@"入会失败%@", error.localizedDescription);
        }else{
            NSLog(@"入会成功");
        }
    }
    }];
}
/// random入会
- (void)joinMeetingByRandom {
    HWMRandomConfParam *param = [[HWMRandomConfParam alloc] init];
}

```

```
param.confId = @"989156631";
param.random = @"1da9ec437c5aa24beb1e2b67c783f59d0859c578022ba88c";
param.nickname = @"杭州会场";
param.isCameraOn = NO;
param.isMicOn = YES;
[[HWMSdk getOpenApi] joinConf:param callback:^(NSError * _Nullable error, id _Nullable result) {
    [self hideLoading];
    if (error) {
        NSLog(@"入会失败%@", error.localizedDescription);
    }else{
        NSLog(@"入会成功");
    }
}
}];
}
```

8.5.4.3 添加与会者

addAttendee

接口描述

该接口用于在会议中添加与会者。

注意事项

1. 必须在会议中，并且只有主持人才可以添加与会者；

方法定义

```
- (void)addAttendee:(NSArray<HWMAAttendeeInfo *> *) attendee callback:(_Nonnull
HWMSDKCompleteHandler)callback;
```

参数描述

attendee是存储要入会的成员信息，详细参数参考[HWMAAttendeeInfo](#)的说明

示例代码

```
/// 会议中添加与会人
[[HWMBizSdk getBizOpenApi] addAttendee:arr callback:^(NSError * _Nullable error, id _Nullable result) {
    if (!error) {
        NSLog(@"success");
    }
}
}];
```

8.5.4.4 离开会议

leaveConf

接口描述

该接口用于离开会议。

注意事项

1. 不在会议中不应调用此接口。
2. 调用该接口离开会议，会议不会结束，在会议存在期间可以再次入会。

方法定义

```
- (void)leaveConf:(HWMSDKCompleteHandler)callback;
```

参数描述

无。

返回值

无

示例代码

```
/// 离开会议
[[HWMSdk getBizOpenApi] leaveConf:^(NSError * _Nullable error, id _Nullable result) {
    [UIUtil showMessage:@"离开会议" error:error];
}];
```

8.5.4.5 结束会议

endConf

接口描述

该接口用于结束会议。

注意事项

1. 不在会议中不应调用此接口。
2. 只有主持人可以结束会议。
3. 调用该接口，会议将结束，无法再次加入该会议。

方法定义

```
- (void)endConf:(HWMSDKCompleteHandler)callback;
```

参数描述

无。

返回值

无

示例代码

```
/// 结束会议
[[HWMSdk getBizOpenApi] endConf:^(NSError * _Nullable error, id _Nullable result) {
    [UIUtil showMessage:@"结束会议成功" error:error];
}];
```

8.5.4.6 显示会议主窗口

showMeetingWindow

接口描述

当会议主窗口处于最小化或者被其他窗口覆盖时，使用该接口将会议主窗口拉到最上层。

注意事项

无

方法定义

```
- (void)showMeetingWindow;
```


参数描述

无

返回值

无

示例代码

```
[[HWMBizSdk getBizOpenApi] showMeetingWindow];
```

8.5.4.7 设置共享权限

setShareScreenPermission

接口描述

该接口用于设置共享权限。

注意事项

该接口返回值只代表接口是否调用成功。

方法定义

```
-(BOOL)setShareScreenPermission:(BOOL)isPermission;
```

参数描述

表 8-40 参数说明

参数	是否必须	类型	描述
isPermission	是	bool	是否允许。

返回值

表 8-41 返回值

类型	描述
BOOL	接口调用成功返回YES，失败返回NO。

示例代码

```
/// 设置共享权限
[[HWMSdk getOpenApi] setShareScreenPermission:isPermission];
```

8.5.5 会议管理

8.5.5.1 预约会议

bookConf

接口描述

该接口用于预约会议。

注意事项

1. 会议开始时间晚于当前时间。
2. 个人会议或者云会议室的时候要传入vmrId。
3. 携带与会者入会的时候需要传入attendee。
4. 开启自动录制会议时必须开启录制会议。
5. callback中的result返回会议详情信息（ [HWMConfDetail](#) ）。

方法定义

```
- (void)bookConf:(HWMOrderConfParam *_Nonnull)param callback:(_Nonnull HWMSDKCompleteHandler)callback;
```

参数描述

表 8-42 HWMOrderConfParam 参数说明

参数	是否必须	类型	描述
confSubject	是	NSString *	会议主题
startTime	是	NSTimeInterval	会议开始时间，UTC时间戳，精度秒
duration	否	NSInteger	会议持续时长，单位分钟，最小值15分钟，最大值1425分钟
isAutoRecord	否	BOOL	会议开始后是否自动开启录制，默认关闭。开启自动录制会议时必须开启录制会议
timeZone	是	NSInteger	时区，默认值当前时区（中国默认时区56-东八区），详情参考 时区表 ，参数值1-73可用。
vmrId	否	NSString *	个人会议或者云会议室时传入
callInRestrictionType	否	HWMJoinConfRestrictionType	允许入会范围限制
attendee	否	NSArray< HWMAttendeeInfo *> *	与会者信息

参数	是否必须	类型	描述
isSmsOn	否	BOOL	发送短信通知开关，可选参数，企业配置里面需要开启，否则不生效
isMailOn	否	BOOL	发送邮件通知开发，可选参数，企业配置里面需要开启，否则不生效
isEmailCalendarOn	否	BOOL	发送邮件日历开关
noPassword	否	BOOL	会议是否无密码，默认为否（只针对随机VMR会议有效）
guestPwd	否	NSString	指定来宾入会密码，默认空表示随机生成密码（只针对随机VMR会议有效）
isVideo	否	BOOL	是否创建视频会议，默认是YES。选NO则创建语音会议
vmrConfIdType	否	HWMVmrConfIdType	云会议室的会议ID类型，默认固定类型
allowGuestStartConf	否	BOOL	是否允许来宾提前启动会议。 说明 <ul style="list-style-type: none"> 仅针对随机会议ID的会议生效。 固定会议ID的会议，请使用修改云会议室信息接口设置。
allowGuestStartConfTime	否	NSUInteger	允许来宾提前入会时间范围（单位：分钟）： 0 - 随时 x - 提前x分钟启动会议 说明 <ul style="list-style-type: none"> 仅针对随机会议ID的会议生效。 固定会议ID的会议，不支持设置提前入会时间。
isOpenWaitingRoom	否	BOOL	是否开启等候室 说明 需要等候室功能开通后才能生效。
isRecordOn	否	BOOL	是否开启录制

表 8-43 枚举 HWMJoinConfRestrictionType 说明

参数	值	说明
HWMJoinConfRestrictionAll	0	所有用户
HWMJoinConfRestrictionCompany	2	企业内用户
HWMJoinConfRestrictionInvited	3	被邀请用户

表 8-44 HWMAttendeeInfo 参数说明

参数	是否必传	类型	描述
number	否	NSString *	账号、密码场鉴权情况下必填
thirdUserId	否	NSString *	第三方帐号，APP ID 鉴权场景下必填
name	否	NSString *	会场名称
email	否	NSString *	邮箱地址
sms	否	NSString *	手机电话
role	否	HWMConfRoleType	会议成员角色
accountId	否	NSString *	用户UUID（会议详情时字段返回登录帐号）
mute	否	BOOL	是否静音
isAnonymous	否	BOOL	是否是匿名入会（仅用于获取与会者列表使用）
state	否	HWMAttendeeState	与会者状态（仅用于获取与会者列表使用）
orgId	否	NSString *	用于识别是否是本企业账号（仅用于获取与会者列表使用）

表 8-45 枚举 HWMConfRoleType 说明

参数	说明
HWMConfRoleTypeAttendee	普通与会者
HWMConfRoleTypeChairman	主席

表 8-46 枚举 HWMVmrConfIdType 说明

枚举值	描述
HWMVmrConfIdTypeFixed	云会议室的会议ID类型是固定类型。
HWMVmrConfIdTypeRandom	云会议室的会议ID类型是随机类型。

示例代码

```

/// 预约会议
- (void)bookConf {
    HWMOrderConfParam * param = [[HWMOrderConfParam alloc] init];
    param.confSubject = @"会议名称";
    param.startTime = 1598759580;
    param.duration = 15;
    param.isAutoRecord = NO;
    param.timeZone = 56;
    param.vmrId = @"";
    param.attendee = self.attendeeArr;
    param.callInRestrictionType = HWMCallRestrictionAll;
    [[HWMBizSdk getBizOpenApi] bookConf:param callback:^(NSError * _Nullable error, id _Nullable result) {
        if (error) {
            NSLog(@"预约会议失败:%@", error.localizedDescription);
        } else {
            NSLog(@"预约会议成功");
        }
    }];
}

```

8.5.5.2 编辑会议

editConf

接口描述

该接口用于编辑未开始的预约会议。

注意事项

1. 会议未开始前且自己是主席才可以调用接口。
2. 预约会议中携带的与会者在编辑会议时需要传入userUuid（会议详情中每个与会者带有该参数）。
3. 所有字段必填，编辑修改的字段可来源于用户输入，其他不想修改的属性通过会议详情参数获取。

方法定义

```

- (void)editConf:(HWMOrderConfParam * _Nonnull)param callback:( _Nonnull
HWMSDKCompleteHandler)callback;

```

参数描述

表 8-47 HWMOrderConfParam 参数说明

参数	是否必须	类型	描述
confId	是	NSString *	会议ID。
isLiveOn	否	BOOL	是否开启直播。

其余参数参见[预约会议](#)。

示例代码

```

/// 编辑会议
- (void)editConf {
    HWMOrderConfParam * param = [[HWMOrderConfParam alloc] init];
    param.confSubject = @"会议名称";
    param.startTime = 1598759580;
    param.duration = 15;
    param.isAutoRecord = NO;
    param.timeZone = 56;
    param.vmrId = @""; // vmr会议需要传入vmrId
    param.callInRestrictionType = HWMCallRestrictionAll;
    param.confId = @"986927771";
    [[HWMBizSdk getBizOpenApi] editConf:param callback:^(NSError *_Nullable error, id _Nullable result) {
        if (error) {
            NSLog(@"编辑会议失败:%@", error.localizedDescription);
        } else {
            NSLog(@"编辑会议成功");
        }
    }];
}

```

8.5.5.3 会议列表

getConfList

接口描述

该接口用于获取会议列表。

注意事项

1. 已登录状态，首次收到会议列表通知之后，调用才有效，否则返回为空的列表。

方法定义

```

- (NSArray<HWMConfListItemInfo *> *)getConfList;

```

参数描述

表 8-48 HWMConfListItemInfo 参数说明

参数	类型	描述
confId	NSString *	会议ID。
vmrConferenceId	NSString *	云会议室ID,显示时使用
confSubject	NSString *	会议主题

参数	类型	描述
chairmanPwd	NSString *	主席密码
generalPwd	NSString *	来宾密码
startTime	NSTimeInterval	会议开始时间, UTC时间戳, 精度秒
endTime	NSTimeInterval	会议结束时间, UTC时间戳, 精度秒
videoConf	BOOL	是否是视频会议
scheduserName	NSString *	预订者姓名
chairJoinUri	NSString *	主持人加入会议uri链接
guestJoinUri	NSString *	普通与会者加入会议uri链接
role	HWMConfRoleType	会议角色
isAutoRecord	BOOL	是否自动开启会议录制
accessNumber	NSString *	会议接入码
scheduleVmr	BOOL	是否是VMR预约记录: 0.普通会议记录; 1.VMR预约记录

表 8-49 枚举 HWMConfRoleType 说明

参数	说明
HWMConfRoleTypeAttendee	普通与会者
HWMConfRoleTypeChairman	主席

示例代码

```
/// 会议列表
[[HWMBizSdk getBizOpenApi] getConfList];
```

8.5.5.4 会议详情

getConfDetail

接口描述

该接口用于获取会议详情。

注意事项

无

方法定义

- (void)getConfDetail:(NSString *)confId completionHandler:(HWM SDKConfInfoCompleteHandler)handler;

参数描述

HWMConfDetail参数说明，继承于[HWMConfListItemInfo](#)

表 8-50 HWMConfDetail 参数说明，继承于 HWMConfListItemInfo

参数	类型	描述
guestFreePwd	BOOL	是否来宾免密
sendNotify	BOOL	是否发送邮件通知
sendSms	BOOL	是否发送短信通知
sendCalendarNotify	BOOL	是否发送日历通知
isAutoRecord	BOOL	会议开始后是否自动开启录制
vmrId	NSString *	用VMRId创建的会议的标识，编辑会议时用
callInRestriction	HWMCallInRestrictionType	呼入限制
vmrFlag	BOOL	是否是VMR会议
attendee	NSArray< HWMAttendeeInfo *> *	与会者列表
allowGuestStartConf	BOOL	是否允许来宾提前进入会议
allowGuestStartConfTime	NSInteger	允许来宾提前入会时间范围(单位：分钟)，预约随机会议id会议可选 0 - 随时，n - 提前n分钟启动会议（n可取值：0,5,10,15,30）。默认为0(即可随时入会)
vmrConfIdType	HWMVmrConfIdType	云会议室的会议ID类型
isOpenWaitingRoom	BOOL	是否开启等候室
isRecordOn	BOOL	是否开启录制
isLiveOn	BOOL	是否开启直播

示例代码

```

/// 获取会议详情
- (void)getConfDetail {
    [[HWMBizSdk getBizOpenApi] getConfDetail:@"986927771" completionHandler:^(NSError * _Nullable error, HWMConfDetail * _Nullable result) {
        if (error) {
            NSLog(@"%@", error.localizedDescription);
        }
    }];
}

```



```

    } else {
        NSLog(@"获取会议详情成功");
    }
}];
}

```

8.5.5.5 取消会议

cancelConf

接口描述

该接口用于取消未开始的会议。

注意事项

1. 只有主席可以取消会议

方法定义

```
- (void)cancelConf:(NSString * _Nonnull)confId callback:( _Nonnull HWMSDKCompleteHandler)callback;
```

参数描述

表 8-51 参数说明

参数	是否必须	类型	描述
confId	是	NSString *	会议ID。

示例代码

```

/// 取消会议
- (void)cancelConf {
    [[HWMBizSdk getBizOpenApi] cancelConf:@"986927771" callback:^(NSError * _Nullable error, id
    _Nullable result) {
        if (error) {
            NSLog(@"%@",error.localizedDescription);
        } else {
            NSLog(@"取消会议成功");
        }
    }
}];
}

```

8.5.5.6 云会议室列表

getVmrList

接口描述

该接口用于获取云会议室列表。

注意事项

无

方法定义

```
- (void)getVmrList:( _Nonnull HWMSDKVmrListCompleteHandler)callback;
```

参数描述

表 8-52 HWMVmrInfoModel 参数说明

参数	类型	描述
name	NSString *	会议室名称
confId	NSString *	用于云会议室显示的ID
vmrId	NSString *	云会议室唯一ID
chairmanPwd	NSString *	主席密码
guestPwd	NSString *	来宾密码
type	HWMConfIdType	会议室类型
maxParties	NSUInteger	会议室最大方数
usable	BOOL	会议室状态，YES可用，NO不可用
allowGuestStartConf	BOOL	是否允许来宾提前启动会议
confAllowJoinUser	HWMJoinConfRestrictionType	允许入会范围
isOpenWaitingRoom	BOOL	是否启用等候室

表 8-53 枚举 HWMConfIdType 说明

参数	描述
HWMConfIdPersonal	个人会议ID
HWMConfIdMeetingRoom	云会议室ID
HWMConfIdRandom	随机会议ID

表 8-54 枚举 HWMJoinConfRestrictionType 说明

参数	描述
HWMJoinConfRestrictionAll	所有用户
HWMJoinConfRestrictionCompany	企业内用户
HWMJoinConfRestrictionInvited	被邀请用户

示例代码

```

/// 获取云会议室列表
- (void)getVmrList {
    [[HWMBizSdk getBizOpenApi] getVmrList:^(NSError * _Nullable error, NSArray<HWMVmrInfoModel *> *
    _Nullable result) {
        if (error) {
            NSLog(@"获取成功");
        } else {
            NSLog(@"获取失败");
        }
    }];
}
    
```

8.5.5.7 修改云会议室信息

changeVmrInfo

接口描述

该接口用于修改云会议室信息。

注意事项

1. 尽量只修改来宾密码

方法定义

```

- (void)changeVmrInfo:(HWMModifyVmrParam *)info callback:(_Nonnull
HWMSDKCompleteHandler)callback;
    
```

参数描述

表 8-55 HWMModifyVmrParam 参数说明

参数	是否必须	类型	描述
vmrId	是	NSString *	云会议室唯一ID
guestPwd	是	NSString *	来宾密码
chairPwd	是	NSString *	主席密码
vmrName	是	BOOL	云会议室名称
allowGuestStartConf	是	BOOL	是否允许来宾提前启动会议
confAllowJoinUser	否	HWMJoinConfRestrictionType	允许入会范围
isOpenWaitingRoom	否	BOOL	是否启用等候室

表 8-56 枚举 HWMJoinConfRestrictionType 说明

参数	描述
HWMJoinConfRestrictionAll	所有用户

参数	描述
HWMJoinConfRestrictionCompany	企业内用户
HWMJoinConfRestrictionInvited	被邀请用户

示例代码

```

/// 修改云会议信息
- (void)changeVmrInfo {
    HWMMModifyVmrParam * vmrInfo = [[HWMMModifyVmrParam alloc] init];
    vmrInfo.guestPwd = @"222222";
    vmrInfo.chairPwd = @"123456789";
    vmrInfo.vmrName = @"云会议室名称";
    vmrInfo.vmrId = @"8a8d0d8a718d87d50171a9f6b8de0f40";
    vmrInfo.allowGuestStartConf = YES;
    [[HWMBizSdk getBizOpenApi] changeVmrInfo:vmrInfo callback:^(NSError * _Nullable error, id _Nullable result) {
        if (error) {
            NSLog(@"%@", error.localizedDescription);
        } else {
            NSLog(@"修改成功");
        }
    }];
}

```

8.5.5.8 预约周期会议

bookCycleConf

接口描述

该接口用于预约周期会议。

注意事项

- 不在登录状态下，不应调用该接口。
- 会议开始时间必须晚于当前时间，会议时长应大于15分钟。
- 个人会议或者云会议室的时候要传入vmrId。
- 携带与会者入会的时候需要传入attendees。
- 周期会议最大跨度一年，会议数量不超过50，超过则只预约前50个会议
- callback中的result返回会议详情信息（[HWMConfDetail](#)）

方法定义

```

- (void)bookCycleConf:(HWMBookCycleConfParam * _Nonnull)param callback:(_Nonnull HWMSDKCompleteHandler)callback;

```

参数描述

表 8-57 HWMBookCycleConfParam 参数说明

参数	是否必传	类型	描述
confParam	是	HWMOrderConfParam	预约会议参数

参数	是否必传	类型	描述
cycleConfParam	是	HWMCycleConfParam	周期子会议参数

表 8-58 HWMCycleConfParam 参数说明

参数	是否必须	类型	描述
startDate	是	NSDateInterval	开始日期-时间戳，精度秒(0时区)
endDate	是	NSDateInterval	结束日期-时间戳，精度秒(0时区)
cycleMode	是	HWMCycleMode	周期类型
interval	是	NSUInteger	周期区间。 1、周期类型选择了按天，表示每几天召开一次，取值范围[1,15]； 2、周期类型选择了Week，表示每几周召开一次，取值范围[1,5]； 3、周期类型选择了Month，interval表示隔几月，取值范围[1,3]
listPoints	是	NSString *	周期内的会议召开点。仅当按周和月时有效。数据格式(中间用逗号隔开):1,3,5,7
preRemindDays	是	NSUInteger	周期子会议提前通知天数。

表 8-59 枚举 HWMCycleMode 说明

值	说明
HWMCycleModeDay	以天为周期
HWMCycleModeWeek	以周为周期
HWMCycleModeMonth	以月为周期

示例代码

```

/// 预约周期会议
- (void)bookCycleConf{
    HWMOrderConfParam * orderConfParam = [[HWMOrderConfParam alloc] init];

```

```

orderConfParam.confSubject = @"我的预约会议";
orderConfParam.isAutoRecord = YES;
orderConfParam.startTime = 1633017600; //utc时间戳，单位秒，如果获取的时间是本地时间，需要转换成
utc时间
orderConfParam.duration = 30; //会议持续时长(分钟)
orderConfParam.timeZone = 56;
orderConfParam.callInRestrictionType = HWMCallRestrictionAll;
HWMAttendeeInfo * attendeeInfo = [[HWMAttendeeInfo alloc] init];
attendeeInfo.name = @"Mike";
attendeeInfo.number = @"+991116003543";
attendeeInfo.isMute = YES;
orderConfParam.attendee = @[attendeeInfo];

HWMCycleConfParamModel * cycleConfParamModel = [[HWMCycleConfParamModel alloc] init];
cycleConfParamModel.startDate = 1633017600; //utc时间戳，单位秒，如果获取的时间是本地时间，需要转
换成utc时间;
cycleConfParamModel.endDate = 1636560000;
cycleConfParamModel.cycleType = HWMSDKCycleTypeWeek;
cycleConfParamModel.preRemindDays = 1;

HWMBookCycleConfParam *param = [[HWMBookCycleConfParam alloc] init];
param.confParam = orderConfParam;
param.cycleConfParam = cycleConfParamModel;
[[HWMBizSdk getBizOpenApi] bookCycleConf:param callback:^(NSError * _Nullable error, id _Nullable
result) {
    if (!error) {
        NSLog(@"book cycle conf success");
    }else{
        NSLog(@"book cycle conf fail errorCode : %zd",error.code);
    }
}
]];
}

```

8.5.5.9 编辑周期会议

editCycleConf

接口描述

该接口用于编辑周期会议。

注意事项

1. 该接口在已登录状态下才能调用。
2. 所有字段必填，编辑修改的字段可来源于用户输入，其他不想修改的属性一定要取会议详情对应字段来赋值。

方法定义

```

- (void)editCycleConf:(HWMBookCycleConfParam * _Nonnull)param callback:( _Nonnull
HWMSDKCompleteHandler)callback

```

参数描述

表 8-60 HWMBookCycleConfParam 参数说明

参数	是否必须	类型	描述
confParam	是	HWMOrderConfParam	普通会议参数

参数	是否必须	类型	描述
cycleConfParam	是	HWMCycleConfParam	周期会议参数

示例代码

```

/// 编辑周期会议
- (void)editCycleConf{
HWMOrderConfParam * orderConfParam = [[HWMOrderConfParam alloc] init];
orderConfParam.confSubject = @"我的预约会议";
orderConfParam.isAutoRecord = YES;
orderConfParam.startTime = 1633017600; //utc时间戳, 单位秒, 如果获取的时间是本地时间, 需要转换成
utc时间
orderConfParam.duration = 30; //会议持续时长(分钟)
orderConfParam.timeZone = 56;
orderConfParam.callInRestrictionType = HWMCallRestrictionAll;
HWMAttendeeInfo * attendeeInfo = [[HWMAttendeeInfo alloc] init];
attendeeInfo.name = @"Mike";
attendeeInfo.number = @"+991116003543";
attendeeInfo.isMute = YES;
orderConfParam.attendee = @[attendeeInfo];

HWMCycleConfParamModel * cycleConfParamModel = [[HWMCycleConfParamModel alloc] init];
cycleConfParamModel.startDate = 1633017600; //utc时间戳, 单位秒, 如果获取的时间是本地时间, 需要转
换成utc时间;
cycleConfParamModel.endDate = 1636560000;
cycleConfParamModel.cycleType = HWMSDKCycleTypeWeek;
cycleConfParamModel.preRemindDays = 1;

HWMBookCycleConfParam *param = [[HWMBookCycleConfParam alloc] init];
param.confParam = orderConfParam;
param.cycleConfParam = cycleConfParamModel;

[[HWMBizSdk getBizOpenApi] editCycleConf:param callback:^(NSError * _Nullable error, id _Nullable
result) {
    if (!error) {
        NSLog(@"book cycle conf success");
    }else{
        NSLog(@"book cycle conf fail errorCode : %zd",error.code);
    }
}
}
}

```

8.5.5.10 取消周期会议

cancelCycleConf

接口描述

该接口用于取消至少还有一场子会议未开始的周期会议。

注意事项

1. 该接口在登录状态下才能调用。
2. 最后一场子会议未开始前可以调用接口, 开始后调用则无效。

方法定义

```
- (void)cancelCycleConf:(NSString * _Nonnull)confId callback:( _Nonnull HWMSDKCompleteHandler)callback;
```

参数描述

表 8-61 参数说明

参数	是否必须	类型	描述
confId	是	NSString *	会议id

示例代码

```

/// 取消周期会议
- (void)cancelCycleConf {
    NSString *confId = "989156631";
    [[HWMBizSdk getBizOpenApi] cancelCycleConfWithParam:cancelParam callback:^(NSError * _Nullable
error, id _Nullable result) {
        NSString *tips = !error ? @"取消周期会议成功" : @"取消周期会议失败";
        NSLog(@"%@@",tips);
    }
};
}

```

8.5.5.11 编辑周期子会议

editSubCycleConf

接口描述

该接口用于编辑周期子会议。

注意事项

1. 该接口在已登录状态下才能调用。

方法定义

```

- (void)editSubCycleConf:(HWMSDKModifySubCycleConfParamModel *_Nonnull)param callback:(_Nonnull
HWMSDKCompleteHandler)callback;

```

参数描述

表 8-62 HWMSDKModifySubCycleConfParamModel 参数说明

参数	是否必须	类型	描述
confId	是	NSString *	会议id
subConfID	是	NSString *	子会议唯一标识UUID
confMediaType	否	HWMConfMediaType	会议媒体类型
startTime	否	NSInteger	会议开始时间（UTC时间），单位为秒
confLen	否	NSInteger	会议持续时长，单位分钟
isAutoRecord	否	BOOL	会议开始后是否自动开启录制
recordAuthType	否	NSInteger	录播鉴权方式 0:可通过连接观看/下载（不拼接nonce）1:企业用户可观看/下载 2:与会者可观看/下载

参数	是否必须	类型	描述
callInRestriction	否	HWMConfAllowJoinUserType	呼入限制
allowGuestStartConf	否	BOOL	允许来宾启动会议
allowGuestStartConfTime	否	NSInteger	允许来宾提前入会时间范围(单位: 分钟) 0-随时, n-提前n分钟启动会议

表 8-63 枚举 HWMConfMediaType 说明

枚举	值	说明
HWMConfMediaTypeAudio	0	音频会议
HWMConfMediaTypeVideo	1	视频会议

表 8-64 枚举 HWMConfAllowJoinUserType 说明

值	描述
HWMConfAllowJoinUserTypeAnyone	任何人
HWMConfAllowJoinUserTypeLoggedInUser	已登录用户
HWMConfAllowJoinUserTypeInCompanyUser	企业内用户
HWMConfAllowJoinUserTypeInvitedUser	被邀请用户

示例代码

```

/// 编辑周期会议
- (void)editSubCycleConf{
    HWMSDKModifySubCycleConfParamModel * modifySubCycleConfParamModel =
    [[HWMSDKModifySubCycleConfParamModel alloc] init];
    modifySubCycleConfParamModel.confId = confId;
    modifySubCycleConfParamModel.subConfId = subConfId;
    modifySubCycleConfParamModel.confMediaType = HWMConfMediaTypeVideo;
    modifySubCycleConfParamModel.startTime = 1633017600;
    modifySubCycleConfParamModel.confLen = 60;
    modifySubCycleConfParamModel.isAutoRecord = NO;
    modifySubCycleConfParamModel.callInRestriction = HWMConfAllowJoinUserTypeAnyone;
    modifySubCycleConfParamModel.allowGuestStartConf = NO;
    modifySubCycleConfParamModel.allowGuestStartConfTime = 0;
    [[HWMBizSdk getBizOpenApi] editSubCycleConf:modifySubCycleConfParamModel callback:^(NSError *
    _Nullable error, id _Nullable result) {
        if (!error) {
    
```

```

        NSLog(@"编辑周期子会议会议成功");
    }else{
        NSLog(@"编辑周期子会议失败:%zd %@", error.code, error.localizedDescription);
    }
}
}];
}

```

8.5.5.12 取消周期子会议

cancelSubCycleConf

接口描述

该接口用于取消未开始的周期子会议。

注意事项

1. 该接口在登录状态下才能调用。
2. 子会议未开始前可以调用接口，开始后调用则无效。

方法定义

```

- (void)cancelSubCycleConf:(HWMSDKCancelSubCycleConfParamModel *_Nonnull)param callback:
(_Nonnull HWMSDKCompleteHandler)callback

```

参数描述

表 8-65 HWMSDKCancelSubCycleConfParamModel 参数说明

参数	是否必须	类型	描述
confId	是	NSString *	会议id
subConfID	是	NSString *	子会议id
endOnlineConf	是	BOOL	是否需要结束在线会议

示例代码

```

/// 取消周期子会议
- (void)confirmCancelSubConf {
    HWMSDKCancelSubCycleConfParamModel *cancelParma = [[HWMSDKCancelSubCycleConfParamModel
alloc] initWithConfId:confId subConfID:subConfID endOnlineConf:YES];
    [[HWMBizSdk getBizOpenApi] cancelSubCycleConf:cancelParma callback:^(NSError * _Nullable error, id
_Nullable result) {
        NSString *tips = !error ? @"取消会议成功" : @"取消会议失败";
        NSLog(@"%@",tips);
    }];
}

```

8.5.6 获取会议信息

8.5.6.1 获取与会者列表

getAttendeeList

接口描述

该接口用于获取会议中与会者列表。

注意事项

1. 只能在会议中使用

方法定义

```
- (NSArray <HWMAttendeeInfo *> *)getAttendeeList;
```

参数描述

[HWMAttendeeInfo](#)参数说明

表 8-66 枚举 HWMAttendeeState 说明

参数	描述
HWMAttendeeStateInConf	会议中
HWMAttendeeStateCalling	正在呼叫
HWMAttendeeStateLeaved	已经离开

示例代码

```
/// 获取与会者列表
[[HWMBizSdk getBizOpenApi] getAttendeeList];
```

8.5.6.2 获取观众列表

getAudienceList

接口描述

该接口用于获取网络研讨会中观众列表。

注意事项

1. 只能在网络研讨会中使用
2. 观众身份（可以通过会议详情消息推送获取）不能使用此接口

方法定义

```
- (NSArray <HWMAttendeeInfo *> *)getAudienceList;
```

参数描述

[HWMAttendeeInfo](#)参数说明

示例代码

```
/// 获取观众列表  
[[HWMBizSdk getBizOpenApi] getAudienceList];
```

8.5.6.3 获取用户状态

getUserState

接口描述

该接口用于获取用户当前的状态(是否在呼叫/会议中)。

注意事项

无

方法定义

```
-(HWMUserState)getUserState;
```

参数描述

表 8-67 枚举 HWMUserState 说明

参数	值	描述
HWMUserStateIdle	0	空闲
HWMUserStateInConf	1	会议中
HWMUserStateInCall	2	呼叫中

返回值

无

示例代码

```
[[HWMSdk getOpenApi] getUserState];
```

8.5.6.4 判断自己是否是主持人

selfIsChairMan

接口描述

该接口用于判断自己当前是否是主持人。

注意事项

1. 只有是主席的时候才返回YES，没有进去会议时也会返回NO；

方法定义

```
-(BOOL)selfIsChairMan;
```

参数描述

无

示例代码

```
/// 判断自己是否是主持人  
[[HWMBizSdk getBizOpenApi] selfIsChairMan];
```

8.5.7 音视频数据流

8.5.7.1 获取音频流

setAudioRawDataOutputConfig

接口描述

该接口用于获取音频数据流。

注意事项

调用该接口且传入参数为YES时，实际业务处理结果会在对应的通知（[音频数据流](#)）中返回。

在初始化后就可以调用此接口，创建或加入会议后就会有通知。

方法定义

```
-(void)setAudioRawDataOutputConfig:(BOOL)enable;
```

参数描述

传入YES为开启，NO为不开启，默认为不开启。

返回值

无

示例代码

```
[[HWMBizSdk getBizOpenApi] setAudioRawDataOutputConfig:YES];
```

8.6 通知参考

注意事项，使用前必看

1. 需要使用通知方法时，必须注册代理对象。
两种方式：
 - a. 初始化时，对HWMOpenSDKConfig对象的属性赋值

```
HWMOpenSDKConfig *config = [[HWMOpenSDKConfig alloc] init];  
//示例根据需求使用对应代理  
config.globalHandler = [NotifyMessageHandler sharedInstance];
```
 - b. 使用前使用[HWMSdk getSdkConfig]获取初始化对象给属性赋值

```
//示例根据需求使用对应代理  
[HWMSdk getSdkConfig].globalHandler = [NotifyMessageHandler sharedInstance];
```
2. 代理方法必须写在代理对象中
3. 使用代理时保证代理对象存在

8.6.1 注意事项，使用通知前请查看

1. 需要使用通知方法时，必须注册代理对象。

两种单代理注册方式：

- a. 初始化时，对HWMOpenSDKConfig对象的属性赋值

```
HWMOpenSDKConfig *config = [[HWMOpenSDKConfig alloc] init];
//示例根据需求使用对应代理
config.confSettingHandler = [NotifyMessageHandler sharedInstance];
```
- b. 使用前使用[HWMSdk getSdkConfig]获取初始化对象给属性赋值

```
//示例根据需求使用对应代理
[HWMSdk getSdkConfig].confSettingHandler = [NotifyMessageHandler sharedInstance];
```

多代理注册方式：

- a. 订阅代理

```
//示例根据需求使用对应代理
HWMOpenSDKConfig *config = [HWMSdk getSdkConfig].confSettingHandler;
[config subscribeGlobalNotificationHandler:[NotifyMessageHandler sharedInstance]];
```
- b. 取消订阅

```
HWMOpenSDKConfig *config = [HWMSdk getSdkConfig].confSettingHandler;
//示例根据需求使用对应代理
[config unsubscribeGlobalNotificationHandler:[NotifyMessageHandler sharedInstance]];
```

2. 代理方法必须写在代理对象中
3. 使用代理时保证代理对象存在

8.6.2 帐号状态变更通知

8.6.2.1 帐号被踢出

通知描述

收到该通知时，表示当前使用的帐号被其他终端踢出，退出登录状态，需要重新登录。如果需要继续操作，需要重新调用登录接口。

方法定义

```
- (void)onKickedOut;
```

示例代码

```
/// 被挤下线
- (void)onKickedOut {
    NSLog(@"您的账号已在其他端登录");
}
```

注意事项

1. SDK初始化时订阅HWMNotifyHandler代理对象或者在需要处理的界面设置代理对象。

8.6.3 企业配置信息通知

8.6.3.1 企业配置信息

通知描述

登录成功后，收到该通知，下发登录的企业配置信息。

方法定义

```
-(void)onCorpConfigNotify:(HWMCorpConfigInfo *)corpConfig;
```

示例代码

```
-(void)onCorpConfigNotify:(HWMCorpConfigInfo *)corpConfig {
    if (corpConfig.hasSMSPerm) {
        NSLog(@"有发送短信通知权限");
    }
    if (corpConfig.hasRecordPerm) {
        NSLog(@"有会议录制权限");
    }
    if (corpConfig.hasPstnPerm) {
        NSLog(@"有PSTN外呼权限");
    }
}
```

注意事项

1. SDK初始化时订阅HWMNotifyHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 8-68 参数说明

参数	类型	描述
corpConfig	HWMCorpConfigInfo *	配置信息

表 8-69 HWMCorpConfigInfo 说明

参数	类型	描述
hasRecordPerm	BOOL	是否有会议录制权限
hasPstnPerm	BOOL	是否有PSTN外呼权限
hasSMSPerm	BOOL	是否发送短信通知权限

8.6.4 会议来电设置设备状态通知

8.6.4.1 设置麦克风状态通知

通知描述

会议和点对点来电时收到该通知时，需要接入者返回是否打开麦克风。若不实现，默认开启。

方法定义

```
- (BOOL)isOpenMicrophoneIncoming;
```

示例代码

```
- (BOOL)isOpenMicrophoneIncoming {  
    return YES;  
}
```

注意事项

1. SDK初始化时设置confSettingHandler代理对象或者在需要处理的界面设置代理对象。

8.6.4.2 设置摄像头状态通知

通知描述

会议和点对点来电时收到该通知时，需要接入者返回是否打开摄像头。若不实现，默认关闭。

方法定义

```
- (BOOL)isOpenCameraIncoming;
```

示例代码

```
- (BOOL)isOpenCameraIncoming {  
    return YES;  
}
```

注意事项

1. SDK初始化时设置confSettingHandler代理对象或者在需要处理的界面设置代理对象。

8.6.5 联系人交互通知

8.6.5.1 打开联系人选择窗口通知

通知描述

收到该通知时，接入方返回定制联系人，sdk打开联系人选择窗口。

方法定义

```
- (void)openContactSelectWindow:(nullable NSArray<HWMAttendeeInfo *> *)param scene:
(HWMSelectedContactScene *)scene
completeHandler:(void (^)(NSArray<HWMAttendeeInfo *> * _Nullable result))handler;
```

示例代码

```
/// 自定义联系人
- (void)openContactSelectWindow:(NSArray<HWMAttendeeInfo *> *)param scene:
(HWMSelectedContactScene)scene completeHandler:(void (^)(NSArray<HWMAttendeeInfo *> *
_Nullable))handler {
    HWMSelectContactViewController *contact = [[HWMSelectContactViewController alloc] init];
    [contact setSelectAttendeeInfoHandler:^(NSArray *attendees) {
        if (handler) {
            handler(attendees);
        }
    }];
    self.selectContactVC = contact;
    [self presentViewControllerAsModalWindow:contact];
}
```

注意事项

1. SDK初始化时设置contactUIHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 8-70 参数说明

参数	类型	描述
param	NSArray<HWMAttendeeInfo *> *	会议中已有的联系人。
scene	HWMSelectedContactScene	选择联系人场景，默认会议中。
handler	void (^)(NSArray<HWMAttendeeInfo *> * _Nullable result)	结果，接入方需将选中的联系人封装成HWMAttendeeInfo，调用handler返回给sdk。

表 8-71 枚举 HWMSelectedContactScene 说明

参数	描述
HWMSelectedContactInConf = 0	正常会议中 win可以使用模态框弹出样式、非模态框弹出样式。
HWMSelectedContactInShare	会议中的共享场景 win使用非模态框弹出样式，不建议使用模态框模式。

8.6.5.2 关闭联系人选择窗口通知

通知描述

收到该通知时，接入端需要关闭联系人选择窗口。

方法定义

```
- (void)closeContactSelectWindow;
```

示例代码

```
/// 关闭选择联系人窗口  
- (void)closeContactSelectWindow {  
    if (self.selectContactVC) {  
        [self.selectContactVC.view.window close];  
    }  
}
```

注意事项

1. SDK初始化时设置contactUIHandler代理对象或者在需要处理的界面设置代理对象。

8.6.6 会议交互通知

8.6.6.1 dock 栏图标及共享时吸附图片通知

通知描述

收到该通知时，接入方需要返回图片。图片将作为视频通话时，视频窗口最小化dock栏图标，以及共享情况下吸附窗口显示的图标。

方法定义

```
- (NSImage *)videoWindowDockIcon;
```

示例代码

```
/// 设置视频通话时 视频窗口最小化dock栏图标 以及共享情况下吸附窗口显示的图标  
- (NSImage *)videoWindowDockIcon {  
    return [NSImage imageNamed:@"iconMeeting"];  
}
```

注意事项

1. SDK初始化时设置confUIHandler代理对象或者在需要处理的界面设置代理对象。

8.6.7 用户头像通知

8.6.7.1 获取用户图像通知

通知描述

收到该通知时，接入方需要返回用户唯一标识对应的头像。

方法定义

```
- (void)downloadHeadPortraitWithThirdAccount:(NSString *)thirdAccount account:(NSString *)account
number:(nonnull NSString *)number completionBlock:(void (^)(NSError *_Nullable error, NSString *_Nullable
headImagePath))completionBlock;
```

示例代码

```
- (void)downloadHeadPortraitWithThirdAccount:(NSString *)thirdAccount account:(NSString *)account
number:(NSString *)number completionBlock:(void (^)(NSError *_Nullable, NSString *
_Nullable))completionBlock {
    NSLog(@"thirdAccount:%@ -- account:%@ -- number:%@", thirdAccount, account, number);
    if (completionBlock) {
        /// 将图片路径返回给SDK
        NSInteger index = arc4random() % 3;
        NSString *path = [[NSBundle mainBundle] pathForResource:[NSString
stringWithFormat:@"head_icon_%lu", index] ofType:@"png"];
        completionBlock(nil, path);
    }
}
```

注意事项

1. SDK初始化时设置headPortraitApiHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 8-72 参数说明

参数	类型	描述
thirdAccount	NSString *	appid登录场景下优先使用此字段。
account	NSString *	账号 密码登录场景下 优先使用此字。
number	NSString *	跨企业场景优先使用此字段。
completionBlock	void (^)(NSError *_Nullable error, NSString *_Nullable headImagePath)	结果回调。

8.6.8 社交分享通知

8.6.8.1 会议信息分享通知

通知描述

收到该通知时，会返回用于分享会中状态的页面。会中鼠标至标题栏出现的分享页面和大小等由外部指定。显示位置、显示和隐藏时机内部控制。原生实现使用此接口，方便简单。

方法定义

```
- (NSView *)socialShareView:(HWMCConfStateInfo *)confInfo;
```

示例代码

```

- (NSView *)socialShareView:(HWMConfStateInfo *)confDetail {
    if (!self.socialView) {
        self.socialView = [[HWMSocialView alloc] initWithFrame:NSMakeRange(0, 0, 500, 300)];
        self.socialView.wantsLayer = YES;
        self.socialView.layer.backgroundColor = [NSColor whiteColor].CGColor;
    }
    self.socialView.confDetail = confDetail;
    return self.socialView;
}
    
```

表 8-73 参数说明

参数	类型	描述
confInfo	HWMConfStateInfo *	会中会议状态详情

表 8-74 HWMConfStateInfo 说明

参数	类型	描述
confId	NSString *	会议id。
confSubject	NSString *	会议主题。
chairmanPwd	NSString *	主持人密码。
generalPwd	NSString *	来宾密码。
startTime	NSTimeInterval	会议开始时间，UTC时间戳，精度秒。
endTime	NSTimeInterval	会议开始时间，UTC时间戳，精度秒。
videoConf	BOOL	是否是视频会议。
confScheduleName	NSString *	预订者姓名。
chairJoinUri	NSString *	主持人加入会议uri链接。
guestJoinUri	NSString *	普通与会者加入会议uri链接。
vmrConferenceId	NSString *	VMR会议显示ID。
role	HWMConfRoleType	角色。
orgId	NSString *	会议所属的企业id。用户判断是否是外部人员入会，登录结果中有个corpId上报，登录的企业ID与会议的企业ID不是一个，说明为外部。

注意事项

1. SDK初始化时设置HWMSocialShareHandler代理对象或者在需要处理的界面设置代理对象。

8.6.8.2 自定义会议信息分享通知

通知描述

收到该通知时，外部展示会中状态分享页面。会中鼠标至标题出现分享View全部外部控制，包括大小，显示位置和显示隐藏时机等。JS显示可以使用此接口。

📖 说明

此接口主要为Electron JS接口使用，原生建议使用[8.6.8.1 会议信息分享通知](#)

方法定义

```
- (void)showSocialShareView:(NSPoint)point shareInfo:(HWMConfStateInfo *)confInfo;
```

表 8-75 参数说明

参数	类型	描述
point	NSPoint	鼠标位置
confInfo	HWMConfStateInfo *	会中会议状态详情

注意事项

1. SDK初始化时设置HWMSocialShareHandler代理对象或者在需要处理的界面设置代理对象。
2. 会议信息分享通知和自定义会议信息分享通知只需要实现一个。若两个都实现，只会取showSocialShareView: shareInfo会议信息分享页面通知。

8.6.9 业务消息通知

8.6.9.1 个人角色变化通知

通知描述

收到该通知时，表示个人角色发生变化。

方法定义

```
- (void)onSelfRoleChanged:(HWMConfRoleType)confRole;
```

示例代码

```
- (void)onSelfRoleChanged:(HWMConfRoleType)confRole {
    switch (confRole) {
        case HWMConfRoleTypeAttendee:
```

```

        NSLog(@"与会者");
        break;
    case HWMConfRoleTypeAudience:
        NSLog(@"观众");
        break;
    case HWMConfRoleTypeChairman:
        NSLog(@"主持人");
        break;
    default:
        break;
    }
}

```

注意事项

1. SDK初始化时订阅HWMBizNotificationHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 8-76 参数说明

参数	类型	描述
confRole	HWMConfRoleType	角色

表 8-77 枚举 HWMConfRoleType 说明

枚举值	描述
HWMConfRoleTypeInvalid = -1	无效角色
HWMConfRoleTypeAttende e = 0	普通与会者
HWMConfRoleTypeChairm an	主席
HWMConfRoleTypeAudienc e	观众

8.6.9.2 会议状态变更通知

通知描述

收到该通知时，表示会议状态发生变更。

方法定义

```
-(void)onConfStatusChanged:(HWMConfStatus)status;
```

示例代码

```

- (void)onConfStatusChanged:(HWMConfStatus)status {
    switch (status) {
        case HWMConfStatusIdle:
            NSLog(@"会议结束");
            break;
        case HWMConfStatusIncoming:
            NSLog(@"正在来电");
            break;
        case HWMConfStatusCallingOut:
            NSLog(@"正在呼出");
            break;
        case HWMConfStatusConnected:
            NSLog(@"已接通");
            break;

        default:
            break;
    }
}

```

注意事项

1. SDK初始化时订阅HWMBizNotificationHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 8-78 参数说明

参数	类型	描述
status	HWMConfStatus	会议通话状态。

表 8-79 枚举 HWMSDKConfStatus 说明

枚举值	描述
HWMConfStatusIdle = 0	空闲状态（会议结束）
HWMConfStatusIncoming	正在来电
HWMConfStatusCallingOut	正在呼出
HWMConfStatusConnected	已接通

8.6.9.3 全量会议列表通知

通知描述

收到该通知时，表示会议列表变化。

方法定义

```

- (void)onReceiveConfList:(NSArray<HWMConfListItemInfo *> *)confList;

```

示例代码

```

- (void)onReceiveConfList:(NSArray<HWMConfListItemInfo *> *)confList {
    NSLog(@"%lu", confList.count);
    [self.dataArr removeAllObjects];
    [self.dataArr addObjectsFromArray:confList];
    [self.listView reloadData];
}
    
```

注意事项

1. SDK初始化时订阅HWMBizNotificationHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 8-80 参数说明

参数	类型	描述
confList	NSArray<HWMConfListItemInfo *> *	全量会议列表

表 8-81 HWMConfListItemInfo 说明

参数	类型	描述
confId	NSString *	会议id。
confSubject	NSString *	会议主题。
accessNumber	NSString *	会议接入码。
chairmanPwd	NSString *	主持人密码。
generalPwd	NSString *	来宾密码。
startTime	NSTimeInterval	会议开始时间，UTC时间戳，精度秒。
endTime	NSTimeInterval	会议开始时间，UTC时间戳，精度秒。
videoConf	BOOL	是否是视频会议。
confScheduleName	NSString *	预订者姓名。
chairJoinUri	NSString *	主持人加入会议uri链接。
guestJoinUri	NSString *	普通与会者加入会议uri链接。
vmrConferenceld	NSString *	VMR会议显示ID。
isAutoRecord	BOOL	是否会议开启自动录制。
scheduleVmr	BOOL	可选，是否是VMR预约记录：0：普通会议记录；1：VMR会议记录。

参数	类型	描述
role	HWMLConfRoleType	角色。

8.6.9.4 会中会议状态信息变更通知

通知描述

收到该通知时，表示会中会议状态信息变化。

方法定义

```
- (void)onConfInfoNotify:(HWMLConfStateInfo *)confInfo;
```

示例代码

```
- (void)onConfInfoNotify:(HWMLConfStateInfo *)confInfo {
    NSLog(@"会中信息上报 onConfInfoNotify :%@", [confInfo yy_modelDescription]);
}
```

注意事项

1. SDK初始化时订阅HWMLBizNotificationHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 8-82 参数说明

参数	类型	描述
confInfo	HWMLConfStateInfo *	会议信息

8.6.9.5 音频数据流

通知描述

收到该通知时，表示有音频数据流上来。

方法定义

```
- (void)onAudioFrameDataNotify:(HWMAudioFrameDataModel *)pFrame;
```

示例代码

```
- (void)onAudioFrameDataNotify:(HWMAudioFrameDataModel *)pFrame{
    NSLog(@"会中音频流上报:%@", [pFrame yy_modelDescription]);
}
```

注意事项

1. SDK初始化时订阅HWMBizNotificationHandler代理对象或者在需要处理的界面设置代理对象。
2. 需要调用[音频数据流开放](#)接口，并且设置参数为YES，该通知才会有值上抛。

参数描述

表 8-83 参数说明

参数	类型	描述
pFrame	HWMAudioFrameDataModel *	音频流信息

表 8-84 HWMAudioFrameDataModel 说明

参数	类型	描述
iSamples	NSInteger	每个声道的采样点数, $iSamples = iSamplesPerSec * 10ms / 1000$
iSamplesPerSec	NSInteger	采样率
iBytesPerSample	NSInteger	每个采样点的字节数, pcm数据一般16bit
iChannels	NSInteger	声道数
pBuffer	NSData	数据缓冲区, 长度 = $iSamples * iBytesPerSample * iChannels$
uiDataLen	NSInteger	编码后或解码前使用该字段表示数据长度, 加解密处理后需更新此字段值
eFrameType	HWMAudioRowDataFrameType	音频帧类型

表 8-85 枚举 HWMAudioRowDataFrameType 说明

枚举值	描述
HWMAudioRowDataFrameTypePcm16 = 0	Pcm16

8.6.10 来电接听策略设置

8.6.10.1 设置点对点来电接听策略

通知描述

点对点来电时会收到该通知，需要接入者返回来电接听策略。若不实现，默认使用原始体验弹出来电接听框。

方法定义

```
- (HWMIncomingAnswerType)answerTypeOnCallIncoming;
```

示例代码

```
- (HWMIncomingAnswerType)answerTypeOnCallIncoming {  
    return HWMIncomingAnswerAutoAnswer; //自动接听  
}
```

注意事项

1. SDK初始化时设置incomingAnswerHandler代理对象或者在需要处理的界面设置代理对象。

8.6.10.2 设置会议来电接听策略

通知描述

会议来电时会收到该通知，需要接入者返回来电接听策略。若不实现，默认使用原始体验，弹出会议来电接听框。

方法定义

```
- (HWMIncomingAnswerType)answerTypeOnConflIncoming;
```

示例代码

```
- (HWMIncomingAnswerType)answerTypeOnConflIncoming {  
    return HWMIncomingAnswerAutoDecline; //自动拒接  
}
```

注意事项

1. SDK初始化时设置incomingAnswerHandler代理对象或者在需要处理的界面设置代理对象。

8.6.11 会议来电状态变更通知

通知描述

收到该通知时，表示收到会议来电。

方法定义

```
- (void)onConflIncoming:(HWMConflIncomingInfo *)conflIncomingInfo
```

示例代码

```
- (void)onConflncoming:(HWMConflncomingInfo *)conflncomingInfo {
    NSLog(@"%@@", [conflncomingInfo yy_modelDescription]);
}
```

注意事项

1. SDK初始化时订阅HWMNotifyHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 8-86 结构体 HWMConflncomingInfo 参数说明

参数	类型	描述
state	HWMConflncomingState	来电状态
mediaType	HWMSDKConfMediaType	会议媒体类型。
confld	NSString	会议id。

表 8-87 枚举 HWMConflncomingState 说明

枚举值	描述
HWMConflncomingStateStart	初始状态，来电振铃中。
HWMConflncomingStateAccept	接听状态，已接听来电。
HWMConflncomingStateReject	拒绝状态，已拒绝来电。
HWMConflncomingStateTimeout	未响应状态，呼叫超时或者主叫已挂断。

表 8-88 枚举 HWMSDKConfMediaType 说明

枚举值	描述
HWMSDKConfMediaTypeAudio	音频。
HWMSDKConfMediaTypeVideo	视频。

8.6.12 通话状态变更通知

8.6.12.1 呼叫来电

通知描述

收到该通知时，表示收到呼叫来电。

方法定义

```
- (void)onCallIncoming:(HWMCallIncomingInfo *)callIncomingInfo
```

示例代码

```
- (void)onCallIncoming:(HWMCallIncomingInfo *)callIncomingInfo {
    NSLog(@"%@@", [callIncomingInfo yy_modelDescription]);
}
```

注意事项

1. SDK初始化时订阅HWMCallNotifyHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 8-89 结构体 HWMCallIncomingInfo 参数说明

参数	类型	描述
callType	HWMCallMediaType	呼叫类型。
number	NSString	来电者号码
name	NSString	来电者姓名
callId	NSUInteger	通话id

表 8-90 枚举 HWMCallMediaType 说明

枚举值	描述
HWMCallMediaTypeAudio	音频呼叫。
HWMCallMediaTypeVideo	视频呼叫

8.6.12.2 通话状态

通知描述

收到该通知时，表示通话状态发生变更。

方法定义

- (void)onCallStatusChanged:(HWMCallStatus)status;

注意事项

1. SDK初始化时设置HWMCallNotifyHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 8-91 参数说明

参数	类型	描述
status	HWMCallStatus	会议通话状态。

表 8-92 枚举 HWMCallStatus 说明

枚举值	描述
HWMCallStatusIdle	空闲状态 (通话结束)
HWMCallStatusIncoming	正在来电
HWMCallStatusStart	正在呼出
HWMCallStatusConnected	已接通

8.6.12.3 通话结束

通知描述

收到该通知时，表示通话结束，返回通话信息。

方法定义

- (void)onCallEnded:(HWMCallRecordInfo *)callInfo;

注意事项

1. SDK初始化时设置HWMCallNotifyHandler代理对象或者在需要处理的界面设置代理对象。

参数描述

表 8-93 参数说明

参数	类型	描述
callInfo	HWMCallRecordInfo	结束通话的记录信息。

表 8-94 HWMCallRecordInfo 说明

参数	类型	描述
callType	HWMCallMediaType	呼叫类型。
srtP	BOOL	是否加密。
displayName	NSString	来电者姓名。
telNum	NSString	来电号码或作为主叫时的呼出号码。
startTime	NSTimeInterval	呼叫开始时间。
callSuccess	BOOL	呼叫是否建立。
reasonCode	NSInteger	失败原因码。
reasonText	NSString	结束通话原因描述。
callRingBack	BOOL	是否为回呼。
callOut	BOOL	是否为主动呼出。
duration	NSTimeInterval	通话时长。
state	HWMCallEndReason	呼叫结束或者错误对应的state。

表 8-95 枚举 HWMCallMediaType 说明

枚举值	描述
HWMCallMediaTypeAudio	音频呼叫。
HWMCallMediaTypeVideo	视频呼叫。

表 8-96 枚举 HWMCallEndReason 说明

枚举值	描述
HWM_CALL_END_REASON_INVALID	通话无效。

枚举值	描述
HWM_CALL_END_REASON_CONNECTED_HANGUP_BY_CALLER	主叫挂断（接通）。
HWM_CALL_END_REASON_CONNECTED_HANGUP_BY_CALLEE	被叫挂断（接通）。
HWM_CALL_END_REASON_HANGUP_BY_CALLER	主叫挂断（未接通）。
HWM_CALL_END_REASON_HANGUP_BY_CALLEE	被叫挂断（未接通）。
HWM_CALL_END_REASON_CALLEE_BUSY	被叫正忙（未接通）。
HWM_CALL_END_REASON_CALLEE_ANSWER_TIMEOUT	被叫接听超时（未接通）。
HWM_CALL_END_REASON_FAILED	通话失败

8.7 错误码参考

8.7.1 常见错误码

表 8-97 枚举 HWMUISDKErrorCode 说明

变量值	值	描述
HWMUISDKErrorCodeContactNotFound	40000	联系人不存在
HWMUISDKErrorCodeNumberAndAccountEmpty	40001	账号和号码不能都为空
HWMUISDKErrorCodeArgsError	40002	参数错误
HWMUISDKErrorCodeRepeatRequest	40003	重复调用方法
HWMUISDKErrorCodeTimeout	40004	超时
HWMUISDKErrorCodeUnInit	40005	未初始化
HWMUISDKErrorCodeArgsLengthOverrun	40006	参数长度超限
HWMUISDKErrorCodePasswordOrAccountEmpty	40007	账号或密码不能都为空

变量值	值	描述
HWMUISDKErrorCodeConfldEmpty	40008	会议id不能为空
HWMUISDKErrorCodeUnLogin	40009	未登录
HWMUISDKErrorCodeLoginIn	40010	登录中
HWMUISDKErrorCodeSubjectTooLong	40011	会议主题长度不能超过50
HWMUISDKErrorCodeConfldTooLong	40012	会议ID长度不能超过20
HWMUISDKErrorCodeConfOrCallExit	40013	正在会议或呼叫中
HWMUISDKErrorCodeAPPIDInvalid	40014	APPID 信息入参不合法
HWMUISDKErrorCodeVMRHostPasswordLengthlegal	40015	主持人密码长度不合法
HWMUISDKErrorCodeVMRNameEmpty	40016	VMR名字为空
HWMUISDKErrorCodeCallInterfaceError	40110	底层调用会议的接口失败
HWMUISDKErrorCodeConfStartTimeEarly	40115	会议的开始时间早于当前时间
HWMUISDKErrorCodeNotInConf	40116	不在会议中
HWMUISDKErrorE2EEncryptAlgorithmError	41117	端到端加密算法不对称
HWMUISDKErrorCodeCallResourceNoEnough	115022001	企业并发资源不足

8.7.2 错误码表

错误码表

表 8-98 错误码说明

错误码	错误描述
-1	Unknown error.:未知错误

错误码	错误描述
0	Success.:成功
111070001	Request error.:请求错误
111070002	Insufficient permissions.:权限不足
111070003	System processing failure.:系统处理失败
111070004	Meeting authentication failed.:校验失败
111070005	Meeting does not exist.:会议不存在
111070006	Personal cloud meeting room does not exist.:个人云会议室不存在
111070007	Meeting access number does not exist.:会议接入号不存在
111070008	Meeting notification could not be sent.:会议通知模板配置错误
111070009	Meeting password cannot be left blank.:会议密码不能为空
111070010	Insufficient permissions to view this meeting.:权限不足,无法查看会议
111070100	Get started meeting exception.:获取召开会议异常
111070101	Start meeting exception.:开始会议异常
111070102	Apply IVR resource exception.:申请IVR资源异常
111070103	Start IVR meeting exception.:开始IVR会议异常
111070104	Apply site resource exception.:申请会场资源异常
111070105	Start sub meeting exception.:开始子会议异常
111070106	Add channel exception.:创建级联通道异常
111070107	Move and apply site resource exception.:移动会场申请资源异常
111070108	Start meeting exception.:开启会议异常
111070109	Get started cloud meeting exception.:获取召开云会议室会议异常
111070110	Invite site exception.:邀请会场异常
111070111	Request token is null.:请求token为空
111070112	Rename site exception.:重命名会场异常
111070113	Hangup site exception.:会场挂断异常
111070114	Mass hangup site exception.:会场批量挂断异常
111070115	Delete attendee exception.:删除与会者异常

错误码	错误描述
111070116	Mass delete attendee exception.:批量删除与会者异常
111070117	Book cycle meeting exception.:预定周期会议异常
111070118	Modify cycle meeting exception.:修改周期会议异常
111070119	Create cycle meeting exception.:创建周期会议异常
111070120	Region data not found.:区域信息获取失败
111070121	Forbidden subscribe.:禁止订阅
111070122	Cancel subscribe fail.:取消订阅失败
111071001	Sysportal service exception.:Sysportal服务运行异常
111071002	Conf service exception.:Conf服务运行异常
111071003	Failed to create the meeting. Try again later.:创建会议失败, 请稍后重试
111071004	This cloud meeting room does not allow guests to join before the host.:VMR会议不允许来宾在主席入会前入会
111071005	This cloud meeting room has been scheduled for another meeting during this period.:VMR会议时间冲突, 该时间段此会议室已有预约会议
111071006	Insufficient cloud meeting time.:VMR会议时长不足
111071007	Meeting does not exist.:VMR预约关系不存在
111071008	Cannot query meeting list.:查询会议列表失败
111071009	Cannot change cloud meeting into immediate meeting.:不支持修改VMR会议为立即会议
111071010	Cannot modify cloud meeting flag.:不支持修改云会议室标识
111071011	The cloud meeting has not yet started. Please wait.:VMR会议未到准备时间点
111071012	Not cloud meeting.:非VMR会议
111071013	Start time less than current time.:会议开始时间不能早于系统当前时间
111071014	Start time over max schedule time.:会议开始时间超出最大预订时间范围
111071015	Duration over range.:时长超出范围
111071016	Cycle ID list null exception.:取消周期会议ID列表不能为空
111071017	Meeting size over range.:与会方数超出范围
111071018	Invalid media type.:媒体类型非法

错误码	错误描述
111071019	Invalid meeting type.:会议类型非法
111071020	Cycle params null exception.:周期会议时,必须携带周期会议参数
111071021	Duration over range.:会议时长超出范围
111071022	Subject length too long.:会议主题超长
111071023	Group URI length too long.:groupuri超长
111071024	Invalid email address format.:邮箱地址不合法
111071025	Invalid mobile number format.:手机号不合法
111071026	Department name too long.:部门名称过长
111071027	Invalid welcome voice enable value.:入会欢迎词开关不在范围内
111071028	Invalid enter voice enable value.:入会提示音开关不在范围内
111071029	Invalid leave voice enable value.:离会提示音开关不在范围内
111071030	Invalid reminders value.:会议提醒非法
111071031	Invalid allow record value.:会议是否启用网络录制取值非法
111071032	Invalid auto record value.:会议是否自动启动录制取值非法
111071033	Invalid user type value.:会预定者的用户类型非法
111071034	Invalid encrypt mode value.:加密类型非法
111071035	Invalid language value.:语言非法
111071036	Invalid auto prolong value.:是否自动延长会议取值非法
111071037	VmriD too long.:vmriD超长
111071038	Invalid record type value.:录播类型非法
111071039	Live address too long.:主流直播地址超长
111071040	Aux address too long.:辅流直播地址超长
111071041	Null start date for cycle meeting.:周期会议时,起始日期为空
111071042	Null end date for cycle meeting.:周期会议时,结束日期为空
111071043	Invalid cycle type value for cycle meeting.:周期会议时,周期类型非法
111071044	Invalid interval for cycle meeting.:周期会议时,interval非法
111071045	Null point for cycle meeting.:周期会议时,point为空
111071046	Invalid point value for cycle meeting.:周期会议时,point非法

错误码	错误描述
111071047	Invalid vmr flag value.:vmrFlag非法
111071048	Invalid record auxStream.:recordAuxStream非法
111071049	Invalid multiStream flag.:multiStreamFlag非法
111071050	Invalid audience flag.:audienceFlag非法
111071051	Invalid liveChannel ID.:liveChannelID不存在
111071052	Anonymous login is locked. Try again later.:匿名登录锁定,请稍后重试
111071053	Link login is locked. Try again later.:链接登录锁定,请稍后重试
111071054	Not enough Meeting ID.:会议ID不足
111071055	Failed to apply meeting ID.:会议ID申请失败
111071056	Failed to release meeting ID.:会议ID释放失败
111071057	Live room is used.:直播房间已被占用
111071058	The meeting has not yet started. Please wait.:当前会议还未召开,请耐心等待
111071059	Meeting size over range.:与会方数超出上限
111071060	The enterprise cannot create meetings because it has not purchased any concurrent meeting resources.:企业未配置并发会议资源,无法使用个人会议ID和随机会议ID发起会议
111071061	Parameter verify failed.:参数校验失败
111071062	Json conversion failed.:JSON转换失败
111071063	Not cloud meeting room owner.:非云会议室拥有者
111071064	This cloud meeting room has been started for another meeting during this period.:VMR会议冲突,当前已有会议正在召开
111071065	This meeting cannot be modified because it is already started.:正在召开会议,无法编辑会议详情
111071066	Attendees amount over range.:与会方数量超出上限
111071067	This meeting cannot be canceled because it is already started.:会议正在召开中,不允许取消会议
111071068	Meeting password encrypt failed.:会议密码加密失败
111071069	No meeting during this period.:选定的时间中没有会议
111071070	Sub cycle meeting cannot modify live room.:周期会议子会议不允许修改直播房间

错误码	错误描述
111071071	Stop online meeting exception.:结束会议失败
111071072	Meeting is ended.:会议已经结束
111071073	Not allow anonymous user.:会议不允许匿名用户呼入
111071074	Only allow user in organization.:会议只允许企业内用户呼入
111071075	Only allow invited user.:会议只允许被邀请用户呼入
111071076	Batch export task for history meeting list during this period.:当前有正在执行的批量导出任务
111071077	Only allow invited user and user in organization.:会议只允许被邀请或者企业内用户呼入
111071078	Max current participants amount reached.:企业单会议并发方数已达上限
111071079	Invalid CTD param.:ctd会议参数不合法
111071080	Invalid guest password.:来宾密码不合法
111071081	Guest password conflict with meeting ID.:来宾密码与会议ID冲突
111071082	Max meeting amount reached.:会议数达到上限
111071083	Invalid meeting random number.:无效的会议随机数
111071084	Max export history meeting list times reached.:导出历史会议次数超出允许的值
111071085	Max multi view layouts reached.:会议多画面布局超出限制
111071086	Max multi view invalid manual type.:会议多画面模式非法
111071087	Max multi view sub view amount reached.:会议多画面子画面个数超出限制
111071088	Max multi view sub view numbers reached.:会议多画面子画面会场数超出限制
111071089	Webinar and common meeting cannot be converted to each other.:网络研讨会不能修改为普通会议,普通会议也不能修改为网络研讨会
111071090	Not allow add audience.:不支持添加观众
111071091	Meeting password authentication failed.:密码错误
111071092	File export task failed.:导出文件任务失败
111071093	File export task is downloading.:导出文件正在处理中
111071094	Not allow audience to join.:会议不允许观众入会

错误码	错误描述
111071095	Need to get WEBINAR_CONF.DTO object.:需要获取WEBINAR_CONF.DTO对象
111071096	User-defined languages over range.:自定义语言个数超出范围
111071097	Your enterprise does not support simultaneous interpretation.:企业不支持同声传译
111071098	Need to set interpreter groups.:需要设置传译组信息
111071099	Invalid interpretation language.:传译组语言非法
111071100	Max languages in a meeting are reached.:单会议最大支持16种语言
111071101	This interpreter group ID already exists.:传译组的组号重复
111071102	Max interpreters in the interpreter group reached.:传译组中传译员的个数达到上限
111071103	This interpreter already exists in an interpreter group.:传译员重复
111071104	Invalid interpreter account.:传译员帐号不存在
111071105	Guest password conflict with chairman password.:来宾密码与主席密码冲突
111071106	Meeting is holding.:云会议室已有会议正在召开
111071107	Invalid audience password.:观众密码不合法
111071108	Audience password conflict with meeting id.:观众密码与会议ID冲突
111071109	Audience password conflict with chairman password.:观众密码与主席密码冲突
111071110	Only RTC meeting support E2E encrypt.:非RTC会议不支持端到端加密
111071111	Audience password conflict with guest password.:观众密码与来宾密码冲突
111071112	Cloud meeting room ID type cannot be modified.:不支持修改会议ID类型
111071113	Cloud meeting room type cannot be modified.:不支持修改VMR类型
111071114	Cloud meeting room flag cannot be modified.:不支持修改VMR标识
111071119	Sub cycle meeting modify time over range.:周期会议修改时间超出范围

错误码	错误描述
111072001	Application for meeting resources failed.:申请资源失败
111072002	Failed to release meeting resources.:释放资源失败
111072003	The meeting is ending.:会议正在结束
111072004	The meeting is starting.:会议正在开始
111072005	The meeting has not yet started.:会议还未开始
111072006	The meeting has ended.:会议已经结束
111072007	Site invitation failed.:邀请会场失败
111072008	Site does not exist.:会场不存在
111072009	Failed to disconnect the site.:挂断会场失败
111072010	Coming soon...:功能暂未支持,敬请期待...
111072011	Meeting status incorrect.:会议状态非法
111072012	Site control operation failed.:会场会控操作失败
111072013	The cloud meeting room is in use by another meeting.:当前VMR会议非目标会议
111072014	The meeting has ended.:VMR会议已过期
111072015	The meeting has not yet started.:VMR会议还未开始
111072016	Meeting does not exist.:VMR预约记录不存在
111072017	Meeting has no attendee.:会议与会者不存在
111072018	Application for chairman password exception.:申请主席密码错误
111072019	Not found site which can be moved exception.:没有找到可以移动的会场
111072020	The current meeting is not a recording meeting. Recording is not supported.:非录播会议不支持录制
111072021	System processing failure.:录播调度失败
111072022	Insufficient recording resources in the system.:录播资源不足
111072023	There are no participants in this meeting.:会议中没有与会人
111072024	Operation failed. No sites have joined this meeting.:请确保会议中已有会场入会
111072025	Voice meeting does not support this function.:语音会议不支持该功能
111072026	The voice site does not support this operation.:语音会场不支持该操作

错误码	错误描述
111072027	Voice sites cannot be displayed in custom continuous presence.:手动设置的多画面中无法显示语音会场
111072028	At least one site must have custom continuous presence configured.:手动设置的多画面中需最少设置一个会场
111072029	At most one presentation pane is allowed in custom continuous presence.:手工设置的多画面最多只允许一个辅流子画面
111072030	In the voice-activated continuous presence mode, each pane can only support a single user.:声控多画面模式下,每个子画面最多只允许一个用户
111072031	The number used for the site to join the meeting cannot be left blank.:与会者的入会号码不允许为空
111072032	The number used for the site to join the meeting is too long.:与会者的入会号码长度超过系统允许的最大值
111072033	The format of the number used for joining the meeting is incorrect. (Correct format: User name name/IP address:Port number or Country code plus phone number):与会者的入会号码格式非法(支持SIP、TEL号码格式)
111072034	Length of site name too long.:与会者的名称长度超过系统允许的最大值
111072035	The site role type is outside the allowed range.:与会者的角色类型不在系统允许的取值范围内
111072036	Length of site email address too long.:与会者的邮箱地址长度超过系统允许的最大值
111072037	Length of site mobile number too long.:与会者的手机号码长度超过系统允许的最大值
111072038	The value for whether a site is muted after it first joins the meeting is outside the allowed range.:与会者入会后是否静音的取值不在系统允许的取值范围内
111072039	The value for whether to automatically invite a site is outside the allowed range.:与会者是否自动邀请的取值不在系统允许的取值范围内
111072040	The host's site viewing type is outside the allowed range.:主席选看类型取值不在系统允许的取值范围内
111072041	The value of the continuous presence setting mode is outside the allowed range.:设置多画面方式取值不在系统允许的取值范围内
111072042	The polling time is outside the allowed range.:轮询显示时间取值不在系统允许的取值范围内

错误码	错误描述
111072043	The pane type cannot be left blank.:画面类型不允许为空
111072044	Invalid pane type.:画面类型非法
111072045	The value of the switch is outside the allowed range.:开关项取值不在系统允许的取值范围内
111072046	The site name cannot be left blank.:与会者的名称不允许为空
111072047	Site name too long.:与会者的名称超过系统允许的最大值
111072048	Continuous presence is not supported. Ask the administrator to configure related data on the MCU.:不支持多画面,请联系管理员开启MCU配置
111072049	Response timed out. Try again later.:等待响应超时
111072050	Meeting locked.:会议被锁定
111072051	The meeting display mode is empty.:会议显示模式为空
111072052	The meeting display mode is outside the allowed range.:会议显示模式不在系统允许的取值范围内
111072053	The meeting pane type is outside the allowed range.:会议画面类型不在系统允许的取值范围内
111072054	The banner or caption type is empty.:横幅或字幕类型为为空
111072055	The banner or caption content is empty.:横幅或字幕的内容为空
111072056	The banner or caption content cannot exceed 2,000 bytes.:横幅或字幕的内容长度超过2000字节
111072057	The request message body is empty.:请求消息体为空
111072058	The pane index is greater than the total number of panes.:子画面索引大于画面总数
111072059	Meeting has no host.:会议中没有主席
111072060	Regular participants cannot perform this operation.:当前普通与会方类型不支持该操作
111072061	You have set host polling and cannot select a specific site to view or set continuous presence.:前主席正在轮询观看会场,不能单独选看其他会场或会议多画面
111072062	You cannot select a site to view on the web page. Perform the operation on the terminal.:当前主席与会方类型不支持该操作
111072063	Applying for recording resources....:录制资源重新申请错误码

错误码	错误描述
111072064	Too many failed login attempts. Try again later.:会控登录锁定
111072065	The meeting ID or password is incorrect. Check the ID and password and try again.:会议不存在或密码错误
111072066	The enterprise has not purchased any recording storage space.:企业未购买录播空间
111072067	Insufficient enterprise recording storage space.:企业录播空间已满
111072068	Failed to start live broadcast. Try again later.:启动直播失败
111072069	Not allow to hand up for host.:不允许指定会议主席举手发言
111072070	The operation is not allowed for guest.:来宾不允许进行该操作
111072071	The operation is only allowed for yourself.:仅能对自己进行操作
111072072	Not allow to apply host for attendees who do not have meeting control abilites.:没有会控能力的与会者不允许设置为主席
111072073	Notify message is expired.:推送消息已过期
111072074	Only participants with a number registered on HUAWEI CLOUD can be set as the host.:云下会场不能成为主席
111072075	Hung up site for meeting ended.:会议结束挂断会场
111072076	Hung up site by host.:主席挂断会场
111072077	Hung up site because host deleted site.:主席删除会议挂断会场
111072078	Hung up site because meeting is recreated.:重建会议挂断会场
111072079	Hung up site for moving site failed.:移动会场失败挂断会场
111072080	Hung up site for inviting site cross region failed.:跨region邀请会场失败挂断会场
111072081	Hung up site for invitation exception.:邀请会场失败挂断会场
111072082	Hung up site for joining meeting repeatedly.:会场重复入会挂断会场
111072083	Lock sharing failed.:锁定共享失败
111072084	This site does not support 1-screen layout.:云下会场禁止设置在多画面一画面中
111072085	Not support realtime subtitle.:会议不支持实时字幕

错误码	错误描述
111072086	Realtime subtitle pid is null.:会议实时字幕指定发言人列表为空
111072087	Invalid realtime subtitle pid.:会议实时字幕指定发言人ID有误或已离会
111072088	Failed to set realtime subtitle.:会议实时字幕设置失败
111072089	Not allow audience to change speaker status.:不允许观众解闭音
111072090	Allow/forbid audience to speak only for audience.:主持人允许/禁止观众说话接口只能操作观众
111072091	This meeting does not support simultaneous interpretation.:会议不支持同声传译
111072092	Interpreter group is empty.:会议当前传译组为空,请先配置传译组
111072093	This meeting has not enable simultaneous interpretation.:会议未开启同声传译
111072094	This site is not interpreter.:当前会场非传译员
111072095	This site has not yet confirmed interpreter identity.:传译员未确认身份
111072096	Invalid site language.:会场选择的语言非法
111072097	This site is confirmed interpreter.:当前会场是已确认的传译员
111072098	This site listen channel is different from speak channel.:普通会场听说频道不一致
111072099	Failed to set interpretation languages. Try again.:设置会议语言频道失败
111072100	Failed to set interpretation channel. Try again.:设置会场语言频道失败
111072101	Max participant amount reached.:超过设置的会议方数
111072102	Max allowed handup amount reached.:观众超过最大允许举手数
111072103	Max allowed speaking amount reached.:观众超过最大允许发言数
111072104	Failed to set virtual audio CSD site.:设置级联通道虚拟语言会场失败
111072105	This meeting does not support cohost.:该会议未启用联席主持人
111072106	This site does not support cohost.:该会场不支持联席主持人

错误码	错误描述
111072107	Host cannot be set to cohost.:不支持设置主持人为联席主持人
111072108	Audience cannot be set to cohost.:不支持设置观众为联席主持人
111072109	Need to check cohost permission.:需要去主会议所在region判断是否为联席主持人
111072110	Cohost cannot operate host.:联席主持人不能操作主持人
111072111	This site does not support share invitation.:当前会场不支持邀请共享能力
111072201	Hung up site because heart beat failed.:与终端心跳中断导致挂断会场
111072202	Insufficient audience cloud meeting room logic resources.:观众数超过VMR套餐中的观众最大并发资源数
111072203	Insufficient participant cloud meeting room logic resources.:主持人与嘉宾超过VMR套餐中的最大并发资源数
111072204	Insufficient webinar resources, so started failed.:网络研讨会套餐已经有在线会议,被占用,激活失败
111072205	Only RTC conf support local record.:非RTC会议不支持本地录制
111072206	Invalid E2E encryption parameter.:端到端加密参数非法
111072207	Not allow to switch role in webinar.:网络研讨会不允许进行角色转换
111072208	The client version used by the user is too early. Cannot start local recording.:客户端不支持本地录制
111072211	Meeting is under AR assist.:会议已经在进行AR协作
111072213	Audience cannot be set to host.:观众不能设置为主持人
111072214	Rename param contains sensitive words.:改名涉及敏感词错误
111072222	Indicates record start exception.:启动云录制失败, 请联系管理员
111072223	Indicates record paused exception.:暂停云录制失败, 请联系管理员
111072224	Indicates record stop exception.:停止云录制失败, 请联系管理员
111072225	Indicates live start exception.:启动直播失败, 请联系管理员
111072226	Indicates live stop exception.:停止直播失败, 请联系管理员

错误码	错误描述
111072504	Max participants in a single meeting reached.:达到系统单个会议人数上限
111072507	waiting room user is leaved.:等候室准入用户已经离会
111072508	waiting room user is joined.:等候室准入用户已经准入
111073001	User does not exist.:用户不存在
111073002	Department does not exist.:组织不存在
111073003	Invalid username format.:用户非法
111073004	Invalid department name format.:组织非法
111073005	Could not query department.:查询组织失败
111073006	Application for anonymous user failed.:申请匿名用户失败
111073007	Invalid cloud meeting room package status.:VMR套餐状态非法
111073008	Failed to Release anonymous user.:释放匿名用户失败
111073009	Not support cycle meeting for welinkC.:WeLinkC 不支持预约周期会议
111073010	Not support cloud meeting room for welinkC.:WeLinkC 不支持预约VMR会议
111073011	Failed to get Welink token.:获取WeLink Proxy Token失败
111073012	Failed to Get M to M token.:获取机机帐号Token失败
111073013	Invalid webinar cloud meeting package status.:网络研讨会VMR套餐状态非法
111073014	Max extend times for cloud meeting trial version reached.:会议延长次数超出VMR试用套餐可延长的最大次数
111073015	Max extend duration for cloud meeting trial version reached.:会议延长时间超出VMR试用套餐单次会议最大时长
111073016	Not support auto extend meeting for cloud meeting trial version.:VMR试用套餐不支持自动延长
111074001	This meeting host already exists.:IDO主席已经存在
111074002	No meeting host permissions.:无IDO主席权限
111074003	Failed to extend the meeting. Max meeting duration reached.:延长会议失败,会议已达到最大时长
111074004	Failed to extend the meeting. The service processing is abnormal.:延长会议失败,服务处理异常
111075001	Failed to start timer.:启动定时器失败

错误码	错误描述
111075002	Failed to end timer.:停止定时器失败
111075003	Json parse failed.:json转换异常
111076001	Released by host.:会场被终端主席挂断
111076002	Released by portal host.:会场被portal主席挂断
111076003	Max number of participants reached.:超出VMR最大方数
111076004	Insufficient enterprise resources.:企业并发数不足
111076005	Not answered.:会场无响应
111076006	User does not exist.:会场不存在
111076007	Offline.:会场离线
111076008	Busy.:忙
111076009	Rejected by user.:拒接
111076010	Released by user.:会场挂断
111076011	Insufficient meeting resources in the system.:媒体资源不足
111076012	Outbound calling disabled.:PSTN服务未开通
111076013	Application for address token failed.:申请地址本鉴权信息失败
111076014	Call failed.:SIP呼叫超时
111076015	Max number of participants reached.:超过设置的会议方数
111076017	Invalid user.:用户失效
111076018	Invalid enterprise.:企业失效
111076019	Invalid SP.:SP失效
111076022	Not allow joining by scanning for hard device.:企业配置来宾不允许扫码邀请硬终端入会*
111076028	Not allow conf control.:会议正在进行AR协作，禁止共享/分组会议/释放主持人等会控操作*
111076029	Not allow AR Assist.:会议正在进行共享/分组会议，禁止进行AR协作*
111076099	Common error.:通常失败
111077001	RSE server info does not exist.:RSE录播相关错误码
111077101	Not support AI record.:会议不支持AI会议纪要
111077102	Reapplying for AI record resource.:正在重新申请会议纪要资源

错误码	错误描述
111077103	Insufficient AI record resource.:AI会议纪要资源不足
111077104	In AI recording, the meeting cannot stop live or recording.:会议处于AI会议纪要状态,不允许停止录制/直播
111077201	Record file does not exist or have no permission.:录播文件已删除或您无权观看,请确认后重试
111077202	RSE authentication failed.:RSE机机鉴权失败
111078000	No available region in multi region.:没有可用Region
111078001	No available global in multi region.:没有可用Global
111078002	Dispatch region error in multi region.:多region 分发region错误
111079000	Failed to phrase big data param query xml.:大参数查询xml解析失败
111079001	Failed to phrase anonymous assign xml.:匿名帐号分配xml解析失败
111080001	Bad request.:请求消息参数错误
111080002	Insufficient permissions.:用户无权限进行该操作
111080004	Site data not found or deleted.:会场不存在或已被删除
111081001	Site is inviting.:会场正在通话中
111081002	Site meeting does not exist or is ended.:会场所存在会议不存在或已结束
111081003	Failed to apply site resource.:申请会场资源失败
111081004	Failed to release site resource.:释放会场资源失败
111081005	Failed to apply site media resource.:申请会场媒体资源失败
111081006	Authentication failed.:认证失败
111081007	Meeting is locked.:会议已经被锁定
111081009	Login is locked because max authentication times reached.:会议鉴权失败次数已达上限被锁定
111081010	Waiting user restrictions.:等候用户限制
111082001	Not support this control.:不支持该会控
111082002	No response.:无响应
111082003	Site not exist.:会场不存在
111082004	Site unavailable.:会场离线
111082005	Site busy.:会场正忙

错误码	错误描述
111082006	Site refused.:会场拒接
111082007	Site leave.:会场挂断
111082008	PSTN service is closed.:PSTN服务关闭
111082009	SIP invitation timeout.:SIP呼叫超时
111083001	RSE does not exist.:RSE设备不存在
111084001	Failed to execute DTMF.:执行DTMF失败
111084002	Get started meeting exception.:启动会议异常
111084003	Call MCU side exception.:呼叫MCU端异常
111084004	Join call exception.:加入呼叫异常
111084005	Invite exception.:邀请异常
111084006	Call terminal side exception.:呼叫端侧异常
111084007	Answer exception.:应答异常
111084008	Common CTL exception.:通用CTL异常
111084009	Hangup exception.:挂断异常
111084010	Notify capability exception.:通知能力异常
111084011	Update MT exception.:更新MT异常
111084012	Negotiate exception.:协商异常
111084013	Not support this meeting.:不支持该会议
111084014	Not allowed join meeting in RTC inviting because guests are not invited.:RTC外邀入会,来宾不在被邀请名单中,拒绝入会
111089002	Failed to phrase URI join xml.:链接入会XML解析失败
111090001	Failed to join data meeting.:数据入会失败
111090002	Join data meeting overtime.:数据入会超时
111100001	Failed to get slider image.:获取滑块图片失败
111100002	Slider check failed.:滑块校验失败
111100003	Invalid random number.:无效的随机数
111100004	Invalid slider token.:无效滑块token
111100005	Need to check slider.:需要滑块校验
111100006	Invalid SMS phone number.:非法的手机号
111100007	Failed to send verification code.:发送验证码失败

错误码	错误描述
111100008	Verification code expired.:验证码已过期
111100009	Verification code error.:验证码错误
111100010	Max verification code check times reached.:验证码错误超过最大次数
111100011	The uuid is invalid, you need to re-verify the phone number.:uuid无效,需要重新验证手机号
111100012	The meeting is not on this site.:会议不在本站点
111100013	Need to verify the mobile phone number. Overseas access to China, free enterprises need to verify the mobile phone verification code.:需要验证手机号, 海外接入国内, 免费企业需要进行手机验证码校验
111110001	Failed to add IM group.:添加群组失败
111110002	Failed to disband IM group.:删除群组失败
111110003	Failed to add IM member.:添加成员失败
111110004	Failed to remove IM member.:删除成员失败
111111007	hangup for rejoin mmr failed.:mmr重入会失败挂断
111112001	Failed to enter the waiting room.:入等候室失败
111112002	Failed to exit the waiting room.:出等候室失败
111112003	Waiting room cache operation failed.:等候室缓存操作失败
111112004	Co-hosts cannot be moved into the waiting room.:联席主持人不能被移入等候室
111112005	Not allowed to enable waiting room.:不允许启用等候室
111112008	Not support this feature according to client capabilities.:客户端能力不支持此特性
111113001	can not find application.:找不到应用信息
111113002	application exist.:应用已存在, 目前一个会议只能关联一种应用
111113003	not required to create application.:无权限创建应用
111113004	need verify id.:应用操作需要实名认证(用户需要绑定手机)
111113005	get token failed.:获取应用用户token失败
111113006	url illegal.:应用服务URL地址非法
111113101	shimo doc has not used.:石墨文档应用未启用
111113102	call shimo doc server failed.:调用石墨文档服务失败

错误码	错误描述
111113103	get shimo token failed.:获取石墨应用级token失败
111113104	not in conf,has no right to apply.:用户不在会议中，无权限申请权限
111500301	breakout conf surpass limit.:分组会议子会议超过上限
111500302	breakout conf name error.:分组会议子会议名称错误
111500303	breakout Start failed.:分组会议启动失败
111500304	breakout not exist when delete:分组会议中没有子会议（删除）
111500305	breakout not exist when start.:分组会议中没有子会议（启动）
111500306	breakout is closing by host:分组会议结束中(主持人停止)
111500307	breakout is closing by host:分组会议已经结束或未启动（加入时）
111500308	Attendee in target conf:移动用户在目标分组
111500309	Chair in current conf:主持人在当前分组中（请求帮助）
111500310	not support breakout conf:不支持分组会议（启动时 启动接口）
111500311	breakout conf starting or started:分组会议已经启动或启动中（重复启动）
111500312	subconf not exist:子会议不存在
111500313	not support in mainconf:主会中用户不支持此操作（请求帮助）
111500314	breakout conf state illegal:分组会议状态异常（缓存误删）
111500315	action forbidden by status:分组会议静态、动态操作由于状态原因，不支持当前操作
111500316	apply chair failed due to host in subconf:主持人在分组中，不能申请为主持人
111500317	apply sub conf id failed:申请分组的会议id失败
111500318	query main conf failed:获取主会议信息失败
111500319	attendee not found:没有找到用户信息
111500320	sub conf stop failed:分组会议停止失败
111502000	subject faile and leave conf start:订阅失败需要离会的错误码起始值

错误码	错误描述
111502099	subject faile and leave conf end:订阅失败需要离会的错误码结束值
112010001	Common failure.:通用失败
112010002	Null param.:参数为空
112010003	Participant not found.:未找到参与者
112010004	Meeting not found.:会议不存在
112010005	SIP invite information timeout.:invite消息超时
112010006	SIP invite information failed.:invite消息失败
112010007	SIP information timeout.:info响应超时
112010008	SIP information failed.:info响应失败
112010009	System processing failed.:系统处理失败
112010010	System processing failed.:系统处理失败
112010011	Failed to invite MRS.:呼叫mrs失败
112010012	Failed to invite participant.:呼叫终端失败
112010013	Data meeting join information failed.:数据入会Info失败
112010014	Failed to start meeting.:召开会议失败
112010015	Media negotiation failed.:媒体协商失败
112010016	User rejects.:用户拒绝
112010017	User is busy.:用户忙
112010018	No call permission.:没有呼叫权限
112010019	Unknown callee.:未知被叫人
112010020	No user response.:没有用户响应
112010021	Join meeting timed out.:加入会议超时
112010022	Session has been released.:会话已释放
112010023	Callee is not registered.:被叫人未注册
112010024	SIP invite timed out.:SIP邀请超时
112010025	User Session timer times out and hangs up.:用户Session timer 超时挂断
112010026	The participant bye hangs up on the MRS side.:与会者腿MRS侧bye挂断
112010027	The participant bye hangs up on terminal side.:与会者腿终端侧bye挂断

错误码	错误描述
112010028	No RTP stream hang up.:无码流挂断
112100000	Sysportal exception.:系统异常
112100001	Bad request.:请求异常
112100002	Illegal request.:非法请求
112100003	Data not found.:数据未获取到
112100004	Data duplication.:数据重复
112100006	The requested type is not supported.:请求的类型不支持
112100007	Failed to save notification template.:保存通知模板失败
112100008	Failed to save data.:保存数据失败
112100009	Failed to query data.:查询数据失败
112100010	Failed to delete data.:删除数据失败
112100011	Password reset required.:需要重置密码
112100012	Incorrect old password.:旧密码错误
112100013	The new password cannot be the same as the old password.:新密码不能与旧密码相同
112100014	Insufficient password complexity and low security.:密码复杂度不够,安全性低
112100015	The length does not meet the requirements.:长度不符合要求
112100016	RSE is in meeting.:RSE在会议中
112100017	RSE to be deleted.:RSE待删除
112100018	Failed to query the meeting list.:查询会议列表失败
112100019	User has been locked out. Try again later.:用户已被锁定,请稍后重新登录
112100022	The live room channel name exists.:频道名称已存在
112100023	Failed to modify system configuration items for duplicated meeting ID prefix.:修改系统配置项失败,会议ID前缀重复
112100024	CONFID prefix can only be added, not modified or deleted.:CONFID前缀只能增加,不能修改和删除
112100025	The live room channel url exists.:直播房间地址已存在
112100026	The live room has been occupied.:直播房间已被使用
112100027	The live room does not exist.:直播房间不存在

错误码	错误描述
112100028	Live room release failed.:直播房间释放失败
112100029	The request is missing header or param.:请求缺失头域
112100030	The Live main stream URL already exists.:音视频推流地址已存在
112100031	The Live auxiliary stream URL already exists.:辅流演示推流地址已存在
112100032	Password is weak.:密码为弱密码
112100033	The live room is occupied.:直播房间被占用
112100034	Save part of attendees since max attendees size reached.:因为与会者数量超出上限只保存部分
112100035	Failed to export data.:导出数据失败
112100036	Failed to import data.:导入数据失败
112100037	Max file size reached.:文件大小超出上限
112100038	LVS target address exists.:lvs的目标地址已存在
112100039	LVS port existed.:lvs端口重复
112100040	Not allow to modify global standby config.:Global只有主机可以修改配置;
112100041	MCU insulation IP address exists.:mcu隔离的IP已经存在
112100042	MCU insulation IP address cannot be modified.:mcu隔离的IP不能修改
112100043	SBC address exists.:SBC配置的地址已经存在
112100044	Internal exception.:内部错误异常
112100045	System config delete exception.:系统配置不允许被删除
112100046	No ISBC access address.:企业id无法获取关联SBC地址
112100047	Media type need to be HD because the organization enable Full HD capability.:企业已开启全高清功能,媒体设置不能低于1080P/8M
112100061	Same interpreter language abbreviation exists.:语言缩写重复
112100062	Same interpreter language name exists.:语言名称重复
112100063	MCU type is MIXED and insulation config is not configured, or MCU meeting exists when modifying mcu connection settings.:修改MCU连接配置告警,以下两种场景会导致:1.MCU存在会议 2.修改类型为MIXED未配置隔离

错误码	错误描述
112100064	MCU type is MIXED and insulation config is not configured when adding mcu connection settings.:新增MCU连接配置告警,MCU类型为MIXED且未配置隔离
112100065	MCU type is MIXED when modifying MCU insulation config or disbanding enterprise.:修改MCU隔离配置时,解除企业绑定时MCU类型为MIXED
112100066	MCU type is MIXED when deleting MCU insulation config.:删除MCU隔离配置时,MCU类型为MIXED
112100067	This gateway type is incompatible with MCU.:MCU与网关类型不兼容
115021001	Insufficient media resource.:媒体资源不足
115021002	Meeting does not exist.:会议不存在
115021003	Media resource does not exist.:媒体资源不存在
115021005	Insufficient media bandwidth resource.:带宽资源不足
115021006	Media device fault.:设备故障
115022000	Insufficient common logic resource.:通用的逻辑资源不足
115022001	Insufficient meeting resources in the enterprise.:企业会议并发方数不足
115022002	Insufficient logic port resource.:逻辑资源端口资源不满足
115022003	Live broadcast resources have not been purchased or are insufficient.:企业直播推流方数不足
115022004	Media device does not exist.:媒体设备不存在
115022005	The cloud recording ports purchased by enterprises have been used up.:企业购买的云录制端口已经用完
115023001	Insufficient ID resources.:ID资源不足
115024001	RMS service exception.:RMS服务异常
115025001	The live room has already been occupied.:直播房间已经被使用
115025002	The live room does not exist.:直播房间不存在
115025003	Live room release failed.:直播房间释放失败
115026001	MCU meeting exists.:MCU有会议
115029001	Invalid common parameter.:通用参数值不合法
115029002	Null parameter.:参数值为空
115029003	service is not available.:服务不可用

错误码	错误描述
115029004	Configuration exception.:配置异常
115029005	Duplicately apply resources.:重复申请资源
115029006	Waiting for applying resources.:等待申请资源中
115029999	Unknown error.:未知错误
118000000	User authentication failed.:用户鉴权失败
118000001	User token authentication failed.:usg token 鉴权失败
118000002	User authentication account lockout.:usg鉴权帐号锁定
200000001	Server is busy. Please try again later.:服务器正忙 请稍后再试
200000002	Internal error. Please contact SRE.:服务器正忙 请联系SRE
200000003	Invalid parameter. Please check whether the parameters are valid.:请求参数错误 请检查参数是否合法
200000004	Illegal request. Please check the value of the request header field X-Auth-Token or X-Access-Token.:非法请求 请检查请求头域X-Auth-Token或X-Access-Token的值
200000005	Forbidden access during the version upgrade.:系统正在升级, 暂时不可访问。如对您造成不便, 十分抱歉
201020023	This user is already the owner of another corporation and cannot create a new corporation for it.:该用户已经是其他企业的所有者,无法为其创建新的企业
201040005	Max number of users reached. Please check the purchased meeting users.:用户总数超过限制, 请检查购买的会议用户
201040016	Password security is low. Please set another password.:密码安全性低, 请设置其他密码
201040031	Activation code overdue.:激活码过期
201040042	Pre-authentication expired.:预验证失效
201040043	This user is the owner of another corporation and cannot be invited.:该用户是其他企业的所有者,暂无法邀请
201040044	Invalid invite connection.:邀请连接非法
201040045	User is already in waiting list.:用户已经在等待列表
201040046	User already exists.:用户已存在
201040047	Account modification is not allowed.:不允许修改账户
201050012	Activation code does not exist.:激活码不存在
201050015	Invalid activation code.:激活码错误
201050021	Activation failed.:激活失败

错误码	错误描述
201050022	Terminal does not allow unbinding.:终端不允许解绑
201080000	VMR resource unavailable.:云会议资源不可用
201080006	The host password and the guest password cannot be the same. Please confirm the input parameters.:主席密码和来宾密码不能相同 请确认输入参数
201080017	Same as meeting password.:和会议密码相同
201510001	No invite permission to get an invite code.:没有邀请权限去获取邀请码
201510003	Invalid application.:申请失效
206010000	User name or password error.:用户名或密码错误
206010007	Account is not activated, no permission to log in.:帐号未激活,无权限登录
206010010	Failed to get verification code, Please contact SRE.:获取验证码失败 请联系SRE
206010011	Verification code must be filled.:验证码不能为空
206010012	Verification code check failed.:验证码校验失败
206010013	Jump login failed. Please contact SRE.:跳转登录失败 请联系SRE
206010022	SSO login failed. Please contact SRE.:SSO登录失败 请联系SRE
206010023	SSO login is not supported.:不支持sso登录
206010024	Invalid authentication type.:认证类型非法
206010025	APPID verification failed.:APPID校验失败
206010027	Authentication type error.:校验类型错误
206010029	Network error.:网络异常
206010030	APPID does not allow automatic registration.:APPID不允许自动注册
206010031	APPID auto registration failed.:APPID自动注册失败
206010036	Normal users cannot log in.:普通用户无法登录
206010042	Huawei account authentication failed.:华为账户认证失败
206010057	APPID authentication expires.:APPID认证过期
206010058	APPID account is not bound.:APPID账户未绑定
206010064	Site mismatch.:站点不匹配

错误码	错误描述
206010065	Phone number conflict.:电话冲突
206010066	E-mail conflict.:邮箱冲突
206010070	The number of users has reached the maximum.:用户数超过上限
206010071	WeChat authentication failed.:微信鉴权失败
206010074	International sites support site changing re-login.:国际站点支持切站点重登
206010075	Account has been unbound.:账号已经解绑
206020000	Invalid authentication information. Please check the value of the request header field X-Auth-Token or X-Access-Token.:鉴权信息无效 请检查请求头域X-Auth-Token或X-Access-Token的值
206020001	Token is created by disaster recovery site.:token是容灾站点创建
206020002	Token is invalid due to being kicked.:token由于被踢失效
206030000	The user does not exist.:系统中不存在该用户
206030001	The verification code is incorrect or has expired. Please obtain a new one.:验证码错误或失效，请重新获取
206030002	Incorrect verification code.:验证码错误
206030003	Failed to send the verification code.:发送验证码失败
206030004	Try to get a new verification code 180s later.:请在180秒后重新发送验证码
206030005	The password cannot be changed.:系统不支持修改密码
206030006	The old password is incorrect.:旧密码错误
206030007	The length of the new password does not meet the requirements.:新密码长度不符合要求
206030008	The password complexity does not meet the requirements.:密码复杂度不满足要求
206030009	The new password must be different from the previous old passwords.:设置的密码不能与最近旧密码相同
206030010	The password cannot contain three or more of the same characters consecutively.:密码不能包含3个以上重复字符
206030011	The password cannot be changed within 5 minutes upon the last password change.:修改密码后，5分钟内不能更新密码

错误码	错误描述
206030012	The password cannot contain the account or the reverse order of the account.:密码不能与帐号或其逆序帐号相同
206030013	At least two characters must be different between the new and old passwords.:新密码相较于旧密码至少要有两个不同的字符
206030014	Invalid password format.:密码格式不正确
206030015	The user is locked and will be locked in 60s.:用户已被锁定,请60秒后重试
206030016	Have no right to reset other's password.:没有权限重置密码
206030018	The number of password retrieval times in a day has reached the maximum(10 times).:一天内忘记密码次数达到最大值 (10次)
206030019	The number of incorrect verification code input has reached the maximum, Please get it again.:验证码输入错误次数超过最大值, 请重新获取
206030023	Failed to send verification code. Please contact SRE.:发送验证码失败 请联系SRE
206030028	The system has enabled SSO login and does not support changing the password. Please retry after the system has disabled SSO login.:系统已开启SSO登录,不支持修改密码 系统关闭SSO登录后重试
206030031	Forgot password for Huawei account.:华为帐号忘记密码
206040000	No corresponding terminal type.:无对应的终端类型
206060008	Sending verification code is locked.:发送校验码锁定
206060009	Slider verification required.:需要滑块校验
206060010	Invalid slider token.:滑块token非法
206060011	Request parameter error.:请求参数错误
206060012	Sending SMS is not supported.:不支持发送短信
206060013	Repeated sending is not supported.:不支持重复发送
206060014	Invalid verification code.:验证码非法
206060015	Max verification codes check times reached.:验证码次数超过最大次数
206060017	Failed to bind Huawei account.:华为账户绑定失败
211030046	Not the ideahub activation code.:不是ideahub激活码
211030050	auth fail too many times, locked.:身份验证失败次数太多,已锁定

错误码	错误描述
211030052	auth fail too many times, permanently locked.:身份验证失败次数太多,已永久锁定
211060003	The maximum times of log uploads has been reached.:日志上传次数达到最大限制
411000001	Common failure.:失败
411000002	Parameter error.:参数错误
411000003	Meeting does not exist.:会议不存在
411000004	A meeting already exists.:已经有会议存在
411000005	Meeting state error.:会议状态异常
411000006	Failed to get device.:获取设备异常
411000007	Failed to set device.:设置设备异常
411000008	No device available.:无可用设备
411000009	This operation is only supported on mobile.:只有移动端支持该操作
411000010	Network access error.:网络接入错误
411000011	Other supplementary service is running.:正在进行其他补充业务
411000012	Memory allocation error.:分配内存错误
411000013	Initialization failed.:初始化失败
411000014	User does not exist.:用户不存在
411000015	Did not join the data meeting.:未加入数据会议
411000016	Neither meeting nor call exists.:会议和呼叫均不存在
411000017	SVN port detection failed.:svn端口探测不通
411000018	Failed to write file.:写文件失败
411000019	RTC does not support STG mode.:RTC不支持STG模式
411000020	Failed to load RTC library. RTC meeting is not supported.:rtc库加载失败, 不支持rtc会议
411000021	The meeting is in the audience pause state when changing roles.:转角色的时候会议正处于观众暂停观看状态
411000022	Not in call.:不在呼叫中
411000023	Meeting or call does not exist.:会议或者呼叫不存在
411000024	Pairing state not allowed.:配对状态不允许

错误码	错误描述
411000025	Pairing request failed.:配对请求失败
411000026	Pairing exceeds limit.:配对超出限制
411000027	Pairing common error.:配对通用错误
411000028	Pairing and joining the meeting failed.:配对入会失败
411000029	Pairing and joining the meeting is busy.:配对入会忙
411000030	Pairing and joining the meeting is invalid.:配对入会无效
411000031	Multiple clicks at the same time to pair and join the meeting error.:同时多次点击入会错误
411000032	Pairing is not established or is being established.:配对未建立或正在建立中
411000033	The camera is not turned on in the third-party capture mode.:第三方采集模式下摄像头未打开
411000034	RTC engine creation failed.:RTC 引擎创建失败
411000035	Not in a meeting.:不在会议中
411000036	mute self failed.:静音自己失败并且采用了保护
411000100	End the meeting with no stream due to disconnection of network.:无码流结束会议网络断开
411000101	End the meeting with no stream because fail to re-join meeting.:无码流结束会议重入会失败
411000102	Null pointer exception.:空指针异常
411000103	Parameters cannot be empty.:参数不许为空
411000104	Indicates encrypt and decript func not match.:加解密算法不匹配
411000105	Indicates Loading User-Defined DLLs failed .:加载用户自定义dll失败
411000106	Indicates Applition not support shimo docs .:企业配置不支持打开石墨文档
411000107	Indicates notification of render fail .:数据会议UI渲染失败
411000108	End the meeting with mmr media rejoin failed.:mmr重入会失败
412100001	The message processing thread is not initialized.:消息处理线程未初始化
412100002	Parameter error.:参数错误
412100003	Timeout.:超时

错误码	错误描述
412100004	Memory allocation error.:分配内存错误
412100005	XML parsing error.:XML解析出错
412100006	DNS resolution error.:DNS解析异常
412100007	Request message failed.:请求消息异常
412100008	Authentication failed.:鉴权失败
412100009	Server error.:服务器异常
412100010	Account is locked.:帐号被锁定
412100011	Create timer error.:创建定时器错误
412100012	Server CA certificate verification failed.:服务器CA证书校验失败
412100013	Network error.:网络异常
412100014	Server redirect.:服务器返回重定向地址
412100015	Account is not activated.:帐号未激活
412100016	JSON parsing error.:JSON解析失败
412100017	The proxy communication times out and the TCP link building fails. Probably because the proxy configuration is incorrect.:代理通信超时，tcp建链失败，大概率是代理配置错误
412100018	Disconnected by the proxy. Probably because the server address is misconfigured.:接被代理断开，大概率是服务器地址配置错误
412100019	The TCP link establishment timed out. Probably because the proxy domain name or port configuration is incorrect.:tcp建链超时，大概率是proxy域名或者端口配置错误
412100020	The authentication process failed. The user does not have proxy permission and needs to apply for proxy permission.:认证鉴权过程失败，用户没有proxy权限，需要申请proxy权限
412100021	The authentication process failed. The proxy user name or password is incorrect.:认证鉴权过程失败，代理用户名或者密码错误
412100022	The authentication process failed. Incorrect authentication type. User name and password are required. The user needs to be prompted for input.:认证鉴权过程失败，错误的认证类型，同时需要用户名、密码，需要提示用户输入

错误码	错误描述
412100023	The authentication process failed. Other unknown errors. It may be that the authentication method is not supported. The user needs to be prompted to try again.:认证鉴权过程失败, 其他未知错误, 有可能是不支持认证方式, 建议弹框提示用户重新发起
412100024	The authentication process failed. Unable to resolve gateway.:认证鉴权过程失败, 无法解析网关
412100025	The authentication process failed. Unable to resolve host.:认证鉴权过程失败, 无法解析主机
412100026	The authentication process failed. Unable to connect.:认证鉴权过程失败, 无法连接
412100027	The authentication process failed. Operation times out.:认证鉴权过程失败, 操作超时
412100028	Message length too long.:消息长度太大
412100029	Base64 encryption failed.:base64加密失败
412100030	XML parsing error.:XML解析出错
412100031	Secure function error.:安全函数失败
412100032	Firewall detection parameter error.:防火墙检测参数错误
412100033	Failed to create STG tunnel.:创建STG隧道失败
412100034	Failed to update STG service.:更新STG服务失败
412100035	Encryption failed.:加密失败
412100036	Do not allow duplicate downloads.:不允许重复下载
412100037	File operation failed.:文件操作失败
412100038	The request message returns empty.:请求消息返回空
412100039	The number of detection servers is 0 error.:探测服务器数为0错误
412100040	Failed to get local IP address.:获取本地IP失败
412100041	XXX.:登录模块消息未初始化
412100042	http return 429 error.:http 返回429错误
412100043	Not logged in, not available:此业务未登录不可用
412200001	Internal error.:内部错误
412200002	Parameter error.:参数错误
412200003	Network timeout.:网络超时
412200004	Network Error.:网络错误

错误码	错误描述
412200005	Need to bind mobile phone.:需要绑定手机
412200006	Need to bind email.:需要绑定邮箱
412200007	Has initialized already.:重复初始化
412200008	State machine execution failed.:状态机执行失败
412200009	Failed to start timer.:开启定时器失败
412200010	The user is prompted to change the password when logging in for the first time.:首次登录, 提示用户修改密码
412200011	Unknown authentication type.:未知鉴权类型
413000001	Common error.:一般错误
413000002	Parameter error.:参数错误
413000003	Memory allocation error.:分配内存错误
413000004	System error.:系统错误
413000005	Message sending error.:发送消息错误
413000006	Get system configuration error.:获取系统配置错误
413000007	Physical network error.:物理网络错误
413000008	Network access error.:网络接入错误
413000009	Create timer error.:创建定时器错误
413000010	Call state error.:呼叫状态错误
413000011	Other operation in progress error.:正在进行其他操作错误
413000012	Requesting the master to start call error.:请求主控进行呼叫开始错误
413000013	Requesting audio resource error.:申请音频资源错误
413000014	Other supplementary service is running.:正在进行其他补充业务
413000015	Record call error.:记录通话记录错误
413000016	Media process error.:媒体进程返回的错误
413000017	Maximum number of calls reached.:超过最大呼叫路数
413000018	SIP account ID does not exist.:SIP账户ID不存在
413000019	Call ID does not exist.:呼叫ID不存在
413000020	Failed due to registering.:正在注册中导致失败
413000021	Registration failed.:注册失败

错误码	错误描述
413000022	Deregistration failed.:注销失败
413000023	Setting account information error.:设置账户信息错误
413000024	Setting SIP account error.:设置SIP帐号失败
413000025	Reporting SIP account information error.:上报SIP帐号信息失败
413000026	Server information error.:服务器信息错误
413000027	SIP account information error.:账户信息错误
413000028	SIPC execution error.:SIPC执行错误
413000029	Not registered error.:未注册错误
413000030	Call Sip interface to subscribe error.:调用Sip接口订阅失败
413000031	Canceling subscription error.:注销订阅中错误
413000032	Subscribing error.:订阅中错误
413000033	Only one local meeting is allowed.:只允许一个本地会议
413000034	Local meeting is not created.:本地会议未创建
413000035	The participant line does not match the chair account.:与会者线路与主席账户不匹配
413000036	Local meeting state error.:本地会议状态错误
413000037	Failed to report the linkage state of the IP Phone.:IP Phone端联动状态上报失败
413000038	Meeting ID does not exist.:会议ID不存在
413000039	Video hold error.:视频保持失败
413000040	Video unhold error.:视频恢复失败
413000041	The maximum number of server meeting reached.:服务器会议个数超出
413000042	Get AA random number error.:获取AA随机数失败
413000043	AA login error.:AA登陆失败
413000044	Meeting type mismatch.:会议类型不匹配
413000045	Failed to start video meeting.:视频会议开启失败
413000046	AA no master server.:AA无主服务器
413000047	Password error.:密码错误
413000048	Username error.:用户名错误
413000049	User is logged in.:用户已登录

错误码	错误描述
413000050	Account is locked.:账户已锁定
413000051	Terminal type mismatch.:终端类型不匹配
413000052	Parsing XML error.:解析XML错误
413000053	Connecting to server error.:连接服务器错误
413000054	Get media configuration error.:获取媒体配置失败
413000055	Get service permission error.:获取业务权限失败
413000056	Insufficient service permission.:业务权限不足失败
413000057	Network environment error.:网络环境错误
413000058	Service conflict.:业务冲突
413000059	Connection timeout.:连接超时
413000060	Unknown error.:未知错误
413000061	Add ringing information error.:添加振铃信息失败
413000062	Delete ringing information error.:删除加振铃信息失败
413000063	Create ringing number error.:创建振铃号码失败
413000064	VVM parameter error.:VVM参数错误
413000065	Getting registered service error.:获取登记业务错误
413000066	Getting voicemail error.:获取语音邮箱错误
413000067	Attendee already exists.:与会者已经存在
413000068	Attendee does not exist.:与会者不存在
413000069	Create server video meeting window error.:创建服务器视频会议窗口失败
413000070	Video window already exists.:视频窗口已存在
413000071	Getting meeting list error.:获取会议列表失败
413000072	Operating requires host privileges.:需要主持人权限才能操作
413000073	No video device to operate.:没有视频设备可以操作
413000074	Refresh registration is not turned off.:没有关闭刷新注册
413000075	Online state report error.:在线状态上报失败
413000076	Network address subscription report error.:网络地址本订阅notify上报失败
413000077	Telepresence account was force unregistered. Information report error.:智真帐号被踢,info上报失败

错误码	错误描述
413000078	TLS root certificate error.:TLS根证书错误
413000079	AD authentication error.:AD鉴权失败
413000080	Meeting list is getting.:会议列表正在获取中
413000081	Prohibited calls.:禁止呼叫
413000082	The maximum number of calls has been reached.:呼叫数达上限
413000083	The maximum number of encrypted calls has been reached.:加密呼叫数达上限
413000084	The maximum number of video calls has been reached and needs to downgrade to audio.:超出视频呼叫数，需要降为音频
413000085	H323 account ID does not exist.:H323账户ID不存在
413000086	H323 account information error.:H323账户信息错误
413000087	H323 account information report failed.:H323帐号信息上报失败
413000088	Insufficient local resources.:本端资源不足
413000089	The calling protocol is not supported.:不支持该呼叫协议
413000090	Setting H323 account information error.:设置H323账户信息错误
413000091	Mediatrix meeting access number reporting failed.:mediatrix会议接入号上报界面失败
413000092	Mediatrix VMR information reporting failed.:上报Mediatrix VMR信息到界面失败
413000093	IMS meeting list information reporting failed.:上报获取IMS会议列表信息到界面失败
413000094	TUP initialization is not complete.:tup初始化未完成
413000095	Message return value error.:消息返回值错误
413000096	Message content too large error.:消息内容太大错误
413000097	Secure function error.:安全函数返回错误
413000098	message response is empty error.:消息响应为空错误
413000099	Account is not enabled.:帐号未启用
413000100	Create call control implement error.:创建呼叫控制类错误
413000101	Already in call.:已经在呼叫中
413000102	Call does not exist.:呼叫不存在

错误码	错误描述
413000103	Failed to get media configuration information.:获取媒体配置信息失败
413000104	Failed to set third-party capture. Calling HME_V_Capture_InputData_MCU failed.:设置第三方采集失败, 调用HME_V_Capture_InputData_MCU返回失败
414100001	Common error.:一般错误
414100002	Parameter error.:参数错误
414100003	Timeout.:超时
414100004	Memory allocation error.:分配内存错误
414100005	DNS resolution error.:DNS解析异常
414100006	Request message failed.:请求消息异常
414100007	Authentication failed.:鉴权失败
414100008	Server error.:服务器异常
414100009	Create timer error.:创建定时器错误
414100010	Start timer error.:开启定时器错误
414100011	The meeting control block is full. Please release the control block that is not in use.:会控控制块已满, 请释放没在使用的控制块
414100012	The proxy communication times out and the TCP link building fails. Probably because the proxy configuration is incorrect.:代理通信超时, tcp建链失败, 大概率是代理配置错误
414100013	Disconnected by the proxy. Probably because the server address is misconfigured.:接被代理断开, 大概率是服务器地址配置错误
414100014	The TCP link establishment timed out. Probably because the proxy domain name or port configuration is incorrect.:tcp建链超时, 大概率是proxy域名或者端口配置错误
414100015	The authentication process failed. The user does not have proxy permission and needs to apply for proxy permission.:认证鉴权过程失败, 用户没有proxy权限, 需要申请proxy权限
414100016	The authentication process failed. The proxy user name or password is incorrect.:认证鉴权过程失败, 代理用户名或者密码错误
414100017	The authentication process failed. Incorrect authentication type. User name and password are required. The user needs to be prompted for input.:认证鉴权过程失败, 错误的认证类型, 同时需要用户名、密码, 需要提示用户输入

错误码	错误描述
414100018	The authentication process failed. Other unknown errors. It may be that the authentication method is not supported. The user needs to be prompted to try again.:认证鉴权过程失败, 其他未知错误, 有可能是不支持认证方式, 建议弹框提示用户重新发起
414100019	Curl proxy exception.:curl代理异常
414100020	Curl host exception.:curl主机异常
414100021	Curl connection exception.:curl连接异常
414100022	Curl operation timed out.:curl操作超时
414100023	Message length too long.:会控消息内容过大
414100024	Failed to create thread.:创建线程失败
414100025	Secure function error.:安全函数返回错误
414100026	JSON parsing failed.:解析json体失败
414100027	WS token is empty.:ws token值为空
414100028	Return JSON data error.:返回json数据异常
414100029	Failed to create meeting control block.:创建会议控制块失败
414100030	Failed to get meeting control block.:获取会议控制块失败
414100031	Failed to get meeting control block by CallId.:通过CallId获取会议控制块失败
414100032	Destroying the meeting control block failed.:销毁会议控制块失败
414100033	Temporary creating meeting information not found.:找不到创会临时信息
414100034	Failed to get participant ID.:获取参会者ID失败
414100035	Failed to get MT by participant ID.:通过ParticipantId获取MT失败
414100036	Failed to get self information by participant ID.:通过ParticipantId获取selfinfo失败
414100037	Failed to get participant information by MT.:通过MT获取会议成员信息失败
414100038	Manual invitation required.:需要手动邀请
414100039	The major parameters of the meeting have been reported.:已上报过会议大参数
414100040	Subtitles are not supported.:不支持字幕

错误码	错误描述
414100041	Meeting Hang Up - Timeout.:会议挂断 - 超时
414100042	Meeting Hang Up - No Streaming.:会议挂断 - 无码流
414100043	Meeting hung up - Hung up by the host.:会议挂断 - 被主持人挂断
414100044	Meeting Hang Up - Meeting has ended.:会议挂断 - 会议已结束
414100045	End of meeting due to leave meeting.:离会导致的会议结束
414100046	Message queue processing function registration failed.:消息队列处理函数注册失败
414100047	Message queue processing function deregistration failed.:消息队列处理函数去注册失败
414100048	Meeting ended due to reject meeting.:拒绝会议导致的会议结束
414100049	Meeting ended by self.:自己结束会议导致的会议结束
414100050	The meeting control module is not initialized.:会控模块未初始化
414100051	The watching special site reaches the maximum.:选看的特殊会场超过限制
414100052	The watching HD site reaches the maximum.:选看的高清会场超过限制
414100053	The watching SD site reaches the maximum.:选看的标清会场超过限制
414100054	The watching specification reaches the maximum.:选看的规格超过限制
414100055	Watching internal errors.:选看内部错误
414100056	Duplicated watch render.:选看的句柄重复
414100057	HTTP request error.:HTTP请求错误
414100058	Websocket request timeout.:websocket请求超时
414100059	Loading failed.:加载错误
414100061	The meeting ends due to the timeout of guest waiting before meeting or the timeout of the meeting control heartbeat.:会前等待页面时间超时或者会控心跳超时导致的会议结束
414100062	The meeting ends due to being canceled.:会议被取消预约导致的会议结束

错误码	错误描述
414100063	The meeting ends due to network error when guest waiting before meeting.:会前等待页面网络异常导致的会议结束
414100064	http return 429 error.:http 返回429错误
414100065	self is chairman.:已是主持人
414100066	self is cohost.:已是联席主持人
414100067	no cohost and chairman.:无主持人和联席主持人
414100068	no supported cohost and chairman.:无支持的主持人和联席主持人
414100069	Invalid user, enterprise or SP.:用户，企业或SP失效
415000001	DB is not initialized.:db未初始化
415000002	Common failure.:失败
415000003	Proxy detection has not been completed.:代理探测还没有完成
416000001	Invalid parameter.:参数错误
416000002	Track module is not initialized.:打点模块未初始化
416000003	Track event does not exist.:打点事件不存在
416000004	Not logged in.:未登录
416000005	Upload is not enabled.:未开启上传
419100001	Invalid parameter.:错误参数
419100002	Incorrect URI.:错误号码
419100003	Memory allocation failed.:内存分配失败
419100004	SDP error.:SDP错误
419100005	Connection does not exist.:连接不存在
419100006	Connection failure.:连接失效
419100007	Operation forbidden.:禁止操作
419100008	State error.:状态错误
419100009	No corresponding line data.:无对应线路数据
419100010	No corresponding registered instance.:无对应注册实例
419100011	No corresponding call instance.:无对应呼叫实例
419100012	No corresponding subscription instance.:无对应订阅实例
419100013	No corresponding Refer instance.:无对应Refer实例

错误码	错误描述
419100014	No corresponding DIM instance.:无对应DIM实例
419100015	Failed to create header.:创建头域失败
419100016	Failed to parse header.:解析头域失败
419100017	Header not found.:没有找到头域
419100018	No SDP message body.:没有sdp消息体
419100019	Failed to get capability.:获取能力失败
419100020	Failed to compare capability.:比较能力失败
419100021	Failed to open logical channel.:打开逻辑通道失败
419100022	Encoding SDP failed.:编码SDP失败
419100023	Decoding SDP failed.:解码SDP失败
419100024	Negotiate to unsupported SDP.:协商到不支持的SDP
419100025	TPTD initialization failed.:TPTD初始化失败
419100026	TPT initialization failed.:TPT初始化失败
419100027	TXN initialization failed.:TXN初始化失败
419100028	UA initialization failed.:UA初始化失败
419100029	GLUE initialization failed.:GLUE初始化失败
419100030	HLLM initialization failed.:HLLM初始化失败
419100031	VCOM initialization failed.:VCOM初始化失败
419100032	SDP initialization failed.:SDP初始化失败
419100033	Multiple instance error.:多实例错误
419100034	Insufficient registration control blocks.:注册控制块不足
419100035	Insufficient subscription control blocks.:订阅控制块不足
419100036	Insufficient call control blocks.:通话控制块不足
419100037	Insufficient IM control blocks.:对话外控制块不足
419100038	Failed to parse URI xml to string.:解析xml的uri失败
419100039	Encryption negotiation failed.:加密协商失败
419100040	Local capabilities are not configured.:没有设置本地能力
419100041	BFCP token error.:bfcpl令牌错误
419100042	DNS resolution error.:dns解析错误
419100043	Failed to get subscription state.:获取订阅状态失败

错误码	错误描述
419100044	IPv4 is not supported.:不支持IPv4
419100045	IPv6 is not supported.:不支持IPv6
419100046	Unsupported host type.:不支持的主机类型
419100047	Failed to get timer.:获取定时器失败
420000001	Cannot receive sharing invitations.:无法接收共享邀请
420000002	ReportDTO parameter verification failed.:reportDTO参数校验失败
420000003	Report folder creation failed.:举报文件夹创建失败
420000004	Report folder compression failure.:举报文件夹压缩失败
420000005	The path of the ZIP file to be uploaded is empty.:待上传的ZIP文件路径为空
420000006	The size of the ZIP file to be uploaded reaches the maximum.:待上传的ZIP文件大小超过限制
420000007	The same person reported more than 100 times within 24 hours.:同一个人24小时内举报次数超过100次
430100001	Common Error.:通用错误
430100002	Invalid parameters.:参数错误
430100003	SIP connection check failed.:SIP连接检查失败
430100004	Timeout.:超时
430100005	In process.:接口正在处理中
430100006	Not initialized or initialization is not complete.:未初始化或初始化尚未完成。
430100007	Not supported in this release.:此版本不支持。
430100008	Operation interrupt.:操作中断。
430100009	Contact not found.:联系人不存在
430100010	Args length overrun.:参数长度超限
430100011	APPID info illegal.:APPID 信息入参不合法
430100012	E2EEncrypt algorithm error.:端到端加密算法不对称
430200001	unlogin.:未登录
430200002	Repeated login.:重复登录
430200003	SIP connection timeout.:SIP连接超时
430200004	Invalid login password.:用于登录的密码参数不合法

错误码	错误描述
430200005	Password or account empty.:账号或密码不能都为空
430300001	The calling number, account number or third-party account number cannot be empty.:呼叫号码、账号或者第三方账号不能为空
430300002	Call account empty.:点对点呼叫账号为空
430300003	Call nickname empty.:点对点呼叫名字为空
430400001	Password required to join the meeting.:加入会议需要密码
430400002	Time zone error.:时区错误
430400003	The meeting duration is less than 1.:会议时长小于1
430400004	Joining the conference is interrupted.:入会被中断
430400005	Password input at least 4 digits.:密码输入至少4位
430400006	User voluntarily cancels.:用户主动取消
430400007	The guest password illegal.:来宾密码非法
430400008	Not in conference or in call.:不在会议或呼叫中
430400009	Already exist conf or call.:已存在一路会议或呼叫
430400010	Confd empty.:会议id不能为空
430400011	Subject too long.:会议主题长度不能超过50
430400012	VMR host password length illegal.:主持人密码长度不合法
430400013	VMR name empty.:VMR名字为空
430400014	Conf starttime early.:会议的开始时间早于当前时间
430400015	Share is locked.:共享被锁定
430400016	Not in data conference.:不在数据会议中
430400017	Role not support operation:角色不支持该操作
430500001	HTTP request, server exception.:HTTP请求，服务器异常
430500002	HTTP request, Connecting to the server times out.:HTTP请求，连接服务器超时
430500003	network unavailable.:网络不可用
430500004	file type illegal.:文件类型不合法
510000403	Server forbidden.:服务器拒绝
510000404	Not found.:未发现
510000480	Temporary unavailable.:临时失效

错误码	错误描述
510000486	Busy here.:这里忙
510000487	Request terminated.:请求终止
510000488	Request is not acceptable here.:这里请求不可接受
510000603	Decline.:丢弃
510000841	Hangup without toast.:挂断不弹框
510000842	Conference is locked.:会议被锁定
910000001	Server error.:服务器异常
910000002	SFU service failure.:sfu服务故障
910000003	Service unavaliable 503.:服务不可达503
910000004	Internal error.:内部错误
910000005	Kicked off.:被踢
910000006	Signature expired.:签名过期
910000007	Reconnect timeout.:重连超时
910000008	Network detection.:网络检测，UI不需要关注该错误码，不对外体现
910000009	User is removed.:用户移除
910000010	The room is dismissed.:房间解散
990000001	Internal error.:内部错误
990000002	Message too large.:消息长度太大
990000003	Insufficient memory.:内存不足
990000004	Failed to send message synchronously.:同步发送消息失败
990000005	Parameter error.:参数错误
990000006	Call ID order error.:呼叫ID时序错误
990000007	Failed to set local screen.:设置本端画面失败
990000008	Failed to open remote screen.:开启远端画面失败
990000009	Device configuration failed.:设备配置失败
990000010	Initializing.:初始化中
990000011	Deinitializing.:去初始化中
990000012	Uploading log.:正在日志上传中
990000013	Failed to get media port.:获取媒体端口失败

错误码	错误描述
990000014	The maximum number view window reached .:选看窗口超过规格数
990000015	Media negotiation failed.:媒体协商失败
990000016	Server no response.:服务器没有响应
990000017	Role switch failed.:角色切换失败
990000018	Failed to join room.:加入房间失败
990000019	Failed to join room. Status is busy.:非空闲状态
990000020	Failed to join room. Server error.:加入房间失败, 服务器异常
990000021	Failed to join room. Service unavaliable.:加入房间失败, 服务不可达
990000022	Failed to join room. Authentication failed.:加入房间失败, 鉴权失败
990000023	Failed to join room. Authentication retry.:加入房间失败, 鉴权重试
990000024	Failed to join room. Clock synchronization.:加入房间失败, 时钟同步
990000025	Failed to join room. URL error.:加入房间失败, url错误
990000026	Kicked off.:被踢
990000027	Share failed.:共享失败
990000028	Currently in external media output mode, this operation is forbidden.:当前为外部媒体输出模式, 禁用该操作
990000029	Reconnection failed.:重连失败
990000030	Server break down.:服务器宕机
990000031	Signature has expired.:签名已过期
990000032	Failed to set remote window mode.:设置远端窗口模式失败
990000033	Failed to subscribe or unsubscribe audio.:订阅或取消订阅音频失败
990000034	Failed to connect to other room.:连接其他房间失败
990000035	Failed to disconnect other room.:断开连接其他房间失败
990000036	Role switching is not allowed.:不允许角色切换
990000037	Currently in third-party capture mode, this operation is forbidden.:当前为第三方采集模式, 禁用该操作
990000038	Failed to set third-party audio capture.:设置第三方音频采集失败

错误码	错误描述
990000039	Failed to set third-party video capture.:设置第三方视频采集失败
990000040	Failed to set share sound switch.:设置共享声音开关失败
990000041	Failed to start or stop upstream audio stream.:启停上行音频流失败
990000042	Failed to start or stop upstream video stream.:启停上行视频流失败
990000043	User is removed.:用户被移除
990000044	The room is dismissed.:房间被解散
990000045	Failed to set remote View.:设置远端View失败
990000052	Failed to join room because room full.:房间已满，加入失败

8.8 版本更新内容

8.8.1 80.13.6 更新内容

1. 问题修复，体验更稳定。

8.8.2 80.13.5 更新内容

1. 问题修复，体验更稳定。

8.8.3 80.12.7 更新内容

1. 问题修复，体验更稳定。

8.8.4 80.12.6 更新内容

1. 开启功能接口enableSDKCapability新增是否显示录制安全提示窗枚举值HWMSDKCapabilitySuportConfirmRecord。具体请见[HWMSDKCapabilitySuport枚举说明](#)
2. 问题修复，体验更稳定。

8.8.5 80.12.5 更新内容

1. 上传个人头像上传条件变更（增加类型和大小判断）。具体请见[上传个人头像注意事项](#)。
2. 问题修复，体验更稳定。

8.8.6 80.10.5 更新内容

1. 云会议室列表（HWMVmrInfoModel.h）和修改云会议室信息（HWMMModifyVMRParam.h）新增字段confAllowJoinUser允许入会范围限制、

isOpenWaitingRoom等候室开启开关。具体请见[HWMVmrInfoModel参数说明](#)、[HWMMModifyVMRParam参数说明](#)。

2. 新增设置共享权限接口setShareScreenPermission。具体请见[设置共享权限](#)。

8.8.7 80.9.5 更新内容

1. 会议详情HWMConfDetail.h、编辑会议HWMOrderConfParam.h新增字段isRecordOn、isLiveOn，具体详见[HWMConfDetail参数说明](#)、[HWMOrderConfParam参数说明](#)。
2. 开启功能接口enableSDKCapability新增分组讨论枚举值HWMSDKCapabilitySupportBreakoutConf，具体详见[HWMSDKCapabilitySupport参数说明](#)。
3. 取消周期子会议CancelSubCycleConf中字段subConfUUID改成subConfID，具体详见[HWMSDKCancelSubCycleConfParamModel参数说明](#)。
4. 预约周期会议和编辑周期会议接口，HWMCycleConfParam.h中cycleCount字段废弃，具体详见[HWMCycleConfParam参数说明](#)。

8.8.8 80.8.5 更新内容

1. 问题修复，体验更稳定。

8.8.9 80.7.5 更新内容

1. App支持自定义多语言，接口详见[设置语言](#)。
2. 新增显示设置界面接口showSettingWindowWithInfo，接口详见[显示设置窗口](#)。
3. 新增与会者列表子项注入点击通知onClickInjectParticipantItemBtnNotify，通知详见[与会者列表子项定制](#)。
4. 新增错误码，详见[常见错误码](#)。

8.8.10 80.5.11 更新内容

1. 新增禁用提示音接口disableVoicePrompts，接口详见[禁用提示音](#)。
2. 新增录制按钮操作定制，具体请见[录制按钮操作定制](#)。

8.8.11 80.5.5 更新内容

1. 创建会议HWMCreateConfParam.h、预约会议和编辑会议HWMOrderConfParam.h、查询会议详情HWMConfDetail.h新增字段isOpenWaitingRoom。具体请见[创建会议](#)、[预约会议](#)、[编辑会议](#)、[会议详情](#)。
2. 支持关闭和开启会议设置中隐藏非视频与会者选项，具体请见[开启隐藏非视频与会者功能](#)。

8.8.12 70.16.5 更新内容

1. 新增周期会议相关接口，具体请见[预约周期会议](#)、[编辑周期会议](#)、[取消周期会议](#)、[编辑周期子会议](#)、[取消周期子会议](#)。
2. id入会HWMJoinConfParam.h 新增字段isStopConflictConf，具体请见[HWMJoinConfParam参数说明](#)。

3. sso 登录接口调整，具体请见[loginBySSO](#)。
4. 错误码整改，新版本错误码描述请见 [新旧版本错误码对应关系](#)

表 8-99 新旧版本错误码对应关系

70.14.6 版本及之前的错误码	70.16.5 版本之后的错误码	新版本错误描述
40117	430400007	The guest password illegal.:来宾密码非法
40017	430400001	The calling number, account number or third-party account number cannot be empty.:呼叫号码、账号或者第三方账号不能为空

8.8.13 70.14.5 更新内容

1. 预约云会议室时支持选择预约固定云会议室ID或随机云会议室ID，具体请见[HWMVmrConfIdType](#)说明。
3. 预约、编辑会议支持设置“来宾在主持人之前加入会议”，并可设置提前入会时间（allowGuestStartConf、allowGuestStartConfTime），具体参数请见[HWMOrderConfParam](#)参数说明。
4. APPID登录支持指定部门ID，具体请见 [HWMAppIDLoginParam](#)参数说明。
5. 新增设置网络模式接口，具体请见[网络模式设置](#)

8.8.14 70.13.5 更新内容

1. login 和 logout 相关接口的回调中增加用户的userUuid（原来uuid字段）、account、thirdAccount，，详见[登录返回HWMLoginResult](#)、[退出返回HWMLogoutResult](#)。如果是老版本升级到70.13.5版本,需要修改回调函数的定义。
2. 新增会议来电状态变更通知，具体请见[会议来电状态变更通知](#)
3. 新增呼叫来电通知，具体请见[呼叫来电](#)
4. 新增开启共享声音功能，具体请见[开启共享声音功能](#)
5. 本地录制默认路径变更，接入方App需要申请下载文件夹权限，具体请见[工程文件配置](#)。

8.8.15 70.12.6 更新内容

- 1、修复由于弱网导致视频清晰度下降，网络恢复后视频清晰度无法恢复的问题。

8.8.16 70.12.4 更新内容

1. 支持自定义会议设置窗口，具体请见[自定义设置界面](#)
2. 支持开启特性支持，具体请见[开启特性支持](#)

8.8.17 70.11.5 更新内容

1. 工程配置的签名脚本和Runpath Search Paths配置有变更，具体请见[8.3.2-5 工程文件配置](#)
2. 初始化接口参数新增字段：logKeepDays 日志保存天数，具体请见[初始化信息 HWMOpenSDKConfig](#)

8.8.18 70.10.5 更新内容

1. 支持自定义水印，具体请见[自定义水印](#)
2. 支持自定义本地录制路劲，具体请见[自定义本地录制路径](#)

8.8.19 70.9.5 更新内容

1. 新增SSO登录方式，具体请见[SSO登录](#)

8.8.20 70.8.6 更新内容

问题修复，体验更稳定

8.8.21 70.8.5 更新内容

- 1.支持获取与会者列表（含网络研讨会），详见[获取与会者列表](#)
- 2.支持获取音频数据流，详见[音频数据流开放](#)

8.8.22 70.7.5 更新内容

- 1.支持视频镜像能力，打开后，观看自己显示为镜像效果

图 8-20 视频镜像



- 2.会中界面视觉优化

- 3.支持会控工具栏的自定义，详见[界面定制](#)

8.8.23 70.6.5 更新内容

1. 新增“显示会议主窗口”接口，详见[显示会议主窗口](#)

8.8.24 70.5.8 更新内容

1. 工程配置的签名脚本有变更，新增代码如下，详见（[图5](#)）

```
codesign -f -s "$EXPANDED_CODE_SIGN_IDENTITY" ./HWMUISDK.framework/  
Versions/A/Frameworks/WKTUP.framework/Versions/A/Frameworks/  
libHRTCEngine.framework/Versions/A/libHRTCEngine
```
2. 支持共享作为广播多画面子画面显示，该模式下视频画面和共享画面可自由切换
3. 支持聊天弹幕

8.8.25 70.4.5 更新内容

1. 工程配置的签名脚本有变更，新增代码如下，详见（[图5](#)）

```
codesign -f -s "$EXPANDED_CODE_SIGN_IDENTITY" ./HWMUISDK.framework/  
Versions/A/Frameworks/WKTUP.framework/Versions/A/Frameworks/  
libHME_Video.framework/Versions/A/libHME_Video
```
2. 查询当前终端是否在会中，具体使用请见[获取用户状态](#)
3. 支持安全可配置隐藏，具体使用请见[初始化](#)
4. 会中新增主持人邀请他人共享，主持人也可快速取消共享
5. 日志大小优化

8.8.26 70.3.5 更新内容

1. 会中导出与会者信息优化

8.8.27 60.15.5 更新内容

- 1.新增会议来电接听策略通知接口，具体使用请见[来电接听策略设置](#)
- 2.新增入会扬声器设置参数，具体使用请见[表1](#)
- 3.新增创建、预约、编辑会议密码及语音会议参数
- 4.会中新增“安全”模块，统一管理权限
- 5.新增与会者列表外部标签是否隐藏功能。具体参见[初始化](#)

8.8.28 60.13.2 更新内容

1. 添加与会者，具体使用请见[添加与会者](#)
2. 添加离开会议接口说明，具体使用请见[离开会议](#)
3. 添加结束会议接口说明，具体使用请见[结束会议](#)
4. 添加设置语言接口说明，具体使用请见[设置语言](#)

8.8.29 60.12.5 更新内容

1. 添加来电接听策略通知，具体使用请见[来电接听策略设置](#)
2. 增加加入会议是否打开扬声器的参数，具体使用请见[表1](#)
3. 增加创建会议密码相关参数及创建语音会议参数，具体使用请见[表1](#)
4. 增加预约编辑会议密码相关参数及是否是视频会议参数，具体使用请见[表1](#)

9 Electron SDK

9.1 概述

华为云会议终端Electron SDK提供了一套完整的接口集合，适用于windows和mac两个平台。开发者可以通过调用SDK开放的API，在自己的应用程序中快速集成音视频会议，包括创建/预约/加入会议、进行屏幕共享/程序共享/白板标注、进行全场静音等功能。

9.2 修订记录

表 9-1 修订记录

日期	版本	修订内容
2023-2-23	80.13.6	1. 问题修复，体验更稳定。 具体使用请见 80.13.6更新内容
2023-2-7	80.13.5	1. 设置语言接口setLanguage新增自定义语言功能。 具体使用请见 80.13.5更新内容
2023-1-17	80.12.7	1. 问题修复，体验更稳定。 具体使用请见 80.12.7更新内容
2023-1-9	80.12.6	1. 功能配置开关接口enableCapability新增是否显示录制安全提示窗枚举值HWM_CAPABILITY_CONFIRM_RECORD。 2. 问题修复，体验更稳定。 具体使用请见 80.12.6更新内容
2022-12-24	80.12.5	1.上传头像接口上传条件变更（增加类型和大小判断），win支持上传头像接口。 2. 问题修复，体验更稳定。 具体使用请见 80.12.5更新内容

日期	版本	修订内容
2022-11-16	80.10.5	<ol style="list-style-type: none"> 1. 获取VMR列表接口和修改云会议室信息接口新增参数confAllowJoinUser、isOpenWaitingRoom。 2. 新增设置共享权限接口SetShareScreenPermission。 3. 加入会议接口新增参数isStopConflictConf。 4. 获取会议详情新增参数vmrConfIdType。 5. Appld登录接口新增参数deptCode。 具体使用请见 80.10.5更新内容
2022-10-8	80.9.5	<ol style="list-style-type: none"> 1. Mac补充获取会议详情、编辑会议接口新增参数allowLive。 2. 新增周期会议相关接口。 3. 功能配置开关接口新增分组讨论功能开关枚举值HWM_CAPABILITY_BREAKOUT_CONF。 4. 会议信息通知新增参数isInBreakoutSubConf。 5. Electron SDK补充常见错误码。 具体使用请见 80.9.5更新内容
2022-9-2	80.8.5	<ol style="list-style-type: none"> 1. 获取会议详情、编辑会议接口新增参数allowLive。 2. 会议信息通知新增参数isCreator。 3. 预约会议、编辑会议支持设置“来宾在主持人之前加入会议”，并可设置提前入会时间（allowGuestStartConf、allowGuestStartConfTime）。 具体使用请见 80.8.5更新内容
2022-7-30	80.7.5	<ol style="list-style-type: none"> 1. 新增显示设置界面接口showSettingWindow。 2. 会议信息通知OnConfInfoNotify新增参数isOtherCorpConf。 3. 与会者列表子项注入点击通知OnClickInjectParticipantItemBtnNotify新增与会者详情按钮类型HWM_CLICK_INJECT_PARTICIPANT_ITEM_BTN_PARTICIPANT_DETAIL，新增华为云会议帐号参数accountId。 4. Config接口新增字段participant_detail，支持与与会者头像点击注入。 具体使用请见 80.7.5更新内容

日期	版本	修订内容
2022-6-17	80.5.11	1.config接口新增字段statusBar，支持录制状态栏按钮可隐藏。 2.新增禁用提示音接口disableVoicePrompts。 具体使用请见 80.5.11更新内容
2022-5-18	80.5.5	1. Mac补充exit接口。 2. Mac创建会议、预约会议、编辑会议、获取会议详情接口补充等候室开关功能。 3. 功能配置开关接口新增隐藏非视频与会者配置。 4. 会议结束通知新增服务器异常错误码。 具体使用请见 80.5.5 更新内容
2022-04-16	70.16.5	首次发布

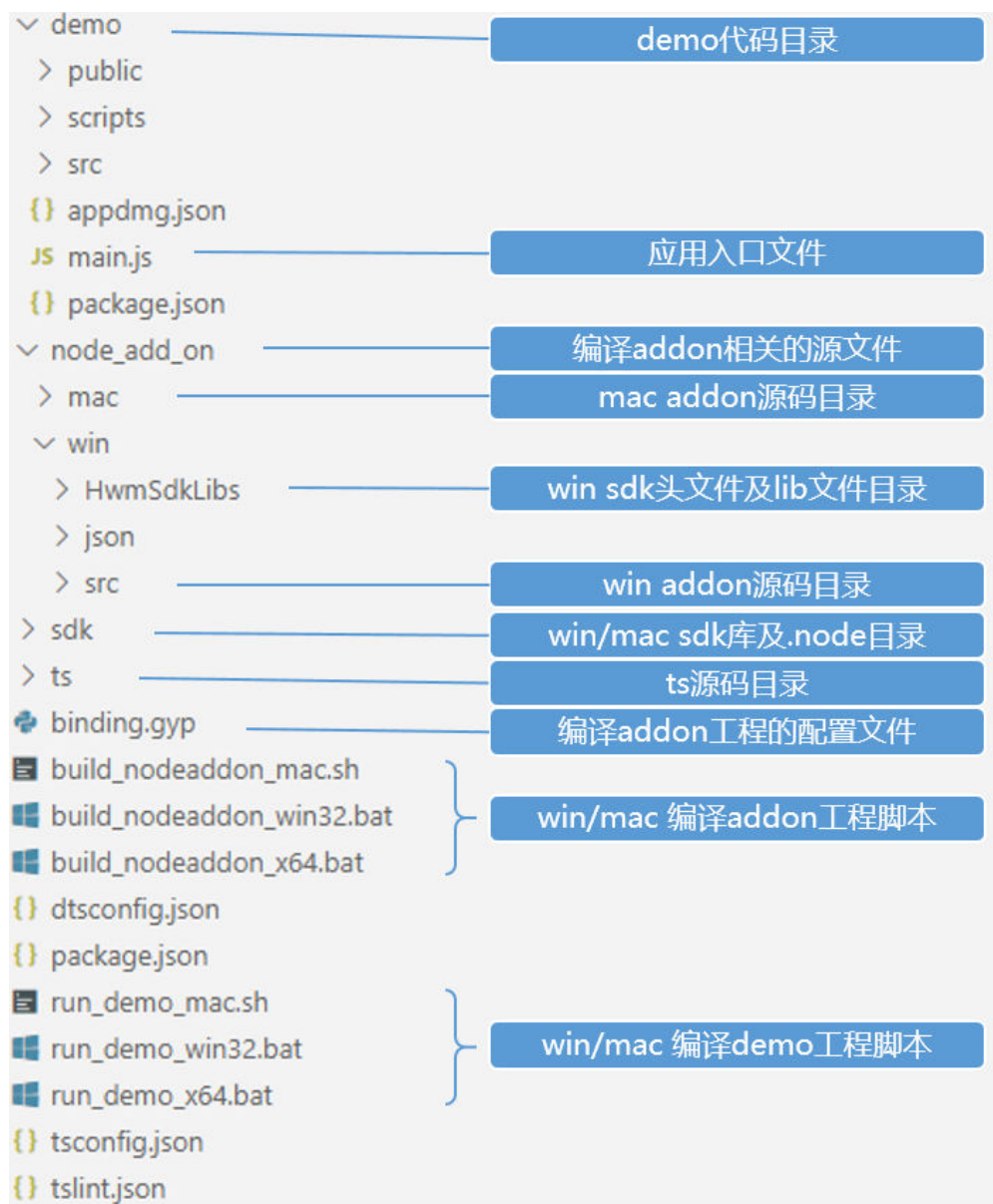
9.3 快速入门

在将Electron SDK集成到第三方应用中前，请先按照“[跑通Electron Demo](#)”中的指导熟悉Electron客户端中集成华为云会议Electron SDK的基本流程。

将华为云会议Windows SDK集成到第三方的Electron客户端的步骤如下：

- 步骤1** 下载[Electron SDK](#)。
- 步骤2** 解压Electron SDK，得到如下目录及文件：

图 9-1 SDK 包目录结构



步骤3 拷贝Electron SDK文件

说明

1. 如果已经安装node，请先确定自己node版本是32位还是64位，可以执行cmd/shell命令查看node版本的位数，结果：x64为64位，ia32为32位。

```
$ node -p "process.arch"
```

2. 如果没有安装node，可到[nodejs官网](https://nodejs.org/)下载对应版本。

- 如果平台是windows，node版本是32位，执行步骤1中解压目录下的 `run_demo_win32.bat`
- 如果平台是windows，node版本是64位，执行步骤1中解压目录下的 `run_demo_x64.bat`
- 如果平台是mac，执行步骤1中解压目录下的 `run_demo_mac.sh`

📖 说明

bat或sh脚本的作用：

- 编译ts接口，生成js和types文件。
- 将对应平台的的SDK文件拷贝到对应的目录下("sdk/win32"、"sdk/x64"、"sdk/mac")。
- 运行Demo。
- 复制js和types目录及其下的子目录和文件、根目录下面的package.json到项目的某个目录下（如果项目使用的是TypeScript，则复制ts目录及其下的子目录和文件）。

📖 说明

第三方应用需要调用的JavaScript接口都在js文件夹中，入口文件为js\index.js（如果项目使用的是TypeScript，入口文件为ts\index.ts）。

步骤4 增加依赖

参考demo中package.json增加依赖。

```
"dependencies": {
  "electron": "^13.6.7"
},
```

步骤5 通过import方式，引用Electron sdk的接口文件所在目录，初始化UISDKService对象，同时设置node路径加载sdk

参考demo中的AppService.js/NotifyService.js代码。

```
import UISDKService from '../././';
const os = window.require('os');
const platform = os.platform();
const arch = os.arch();
const path = window.require("path");

let instance = null;
export default class ApiService {
  constructor() {
    if (!instance) {
      instance = this;
      // node路径请根据实际的路径传入UISDKService构造函数
      let _path = (platform == 'darwin' ? '.././sdk/mac/' : arch == 'x64' ? '.././sdk/x64/' : '.././sdk/win32/');
      let nodePath = path.join(window.__dirname, (_path + "HwmUisdk.node"));
      console.log("ApiService nodePath", nodePath);
      this.uisdkService = new UISDKService(nodePath);
    }
    return instance;
  }
}
```

步骤6 接口调用

华为云会议的接口调用和消息通知的处理时序请参考“[典型场景接口调用概览](#)”。

第三方应用可以调用的JavaScript接口都在js/api/中，可以参考Demo中的AppService/ApiService.js，调用这些接口。

```
import UISDKService from '../././';
const os = window.require('os');
const platform = os.platform();
const arch = os.arch();
const path = window.require("path");

let instance = null;
```

```
export default class ApiService {
  constructor() {
    if (!instance) {
      instance = this;
      // node路径请根据实际的路径传入UISDKService构造函数
      let _path = (platform == 'darwin' ? '../sdk/mac/' : arch == 'x64' ? '../sdk/x64/' : '../sdk/win32/');
      let nodePath = path.join(window.__dirname, (_path + "HwmUisdk.node"));
      console.log("ApiService nodePath", nodePath);
      this.uisdkService = new UISDKService(nodePath);
    }
    return instance;
  }

  init(initInfo) {
    console.log("init, in param = ", initInfo);
    let ret = this.uisdkService.init(initInfo);
    console.log("init, out data = ", ret);
    return ret;
  }
}
```

步骤7 订阅事件通知

订阅第三方可以收到的通知JavaScript接口也都在js/api/中，可以参考Demo中的AppService/NotifyService.js，订阅这些通知。

```
import UISDKService from '../..';
const os = window.require('os');
const platform = os.platform();
const arch = os.arch();
const path = window.require("path");

export default class NotifyService {
  static setNotifyFunc() {

    let _path = (platform == 'darwin' ? '../sdk/mac/' : arch == 'x64' ? '../sdk/x64/' : '../sdk/win32/');
    let nodePath = path.join(window.__dirname, (_path + "HwmUisdk.node"));
    console.log("NotifyService nodePath", nodePath);
    let uisdkService = new UISDKService(nodePath);
    uisdkService.getConfMgrApi().setOnConfListChangedCB(NotifyService.handleOnConfListChanged);
  }

  static handleOnConfListChanged(confListInfo) {
    console.log('OnConfListChanged', ' ', confListInfo = ' ', confListInfo);
    let _confListInfo = window.sessionStorage.getItem("confListInfo");
    _confListInfo = _confListInfo ? JSON.parse(_confListInfo) : [];
    console.log("sessionStorage confList:", _confListInfo);
    let confListNew = confListInfo.confListItem;
    let _data = JSON.stringify(confListNew ? confListNew : []);
    window.sessionStorage.setItem("confListInfo", _data);
  }
}
```

步骤8 使用Electron sdk提供的枚举（可选）

第三方可以直接使用Electron sdk提供的枚举定义，拷贝js/Api/enum.js和types/Api/enum.d.ts到项目的某个目录下，参考Demo中的src/Constants/typeOption.js，使用这些枚举定义。

```
import { MediaType } from './enum';

export const MediaTypeOptions = [
  {
    value: MediaType.HWM_MEDIA_TYPE_AUDIO,
    label: "Audio",
  },
  {
    value: MediaType.HWM_MEDIA_TYPE_VIDEO,
    label: "Video",
  }
]
```



```
}  
];
```

----结束

9.4 典型场景

本小节描述华为云会议在场景使用场景下的接口调用顺序和接口调用示例。

9.4.1 典型场景接口调用概览

3 类接口描述

Electron SDK的接口分为**主动调用同步接口**、**主动调用异步接口**、**订阅消息通知接口**。

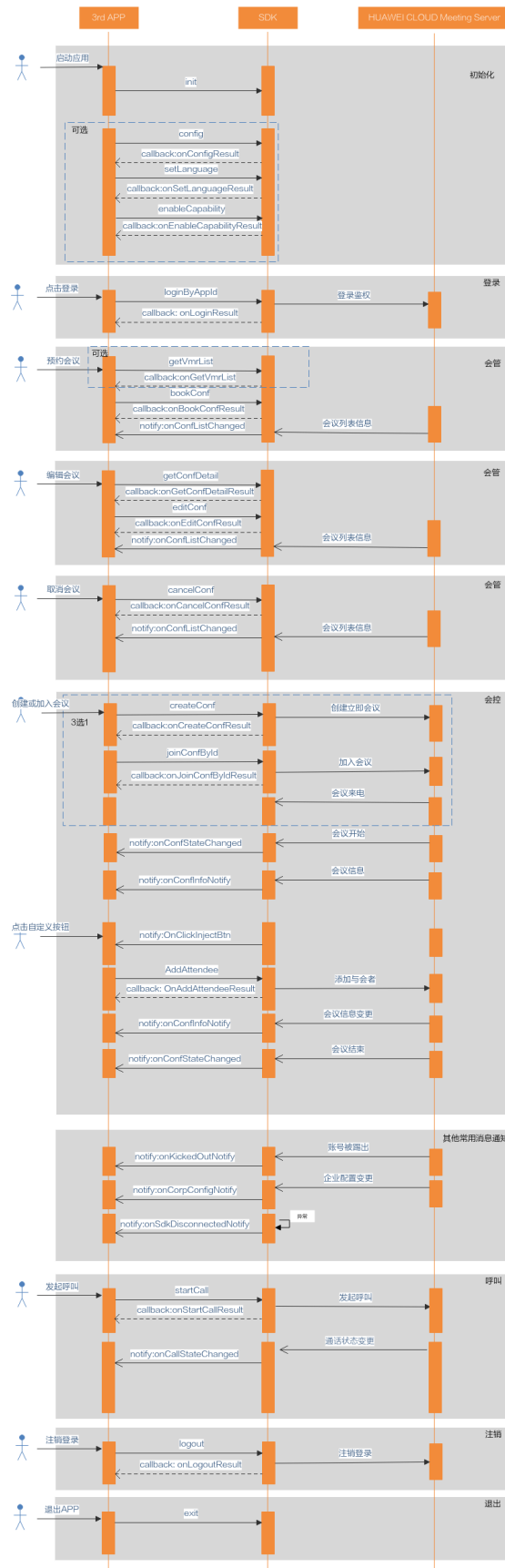
主动调用同步接口：App主动调用SDK的接口，用于完成某项任务。比如初始化接口Init。

主动调用异步接口：App主动调用接口和SDK内部任务处理采用异步处理方式，所有SDK某项任务处理完成后，以接口回调的方式通知App。比如创建会议接口，入参包含创会需要的参数和创会的结果回调onCreateConfResult。

订阅消息通知接口：由华为云会议服务端触发或SDK内部触发的事件通知，调用对应的订阅接口后，SDK以消息通知接口的形式通知给App。比如订阅会议列表通知接口setOnConfListChangedCB。

典型场景接口的调用顺序及生命周期

图 9-2 典型场景接口调用时序图



📖 说明

- 1、一般应用升级前需要杀死进程，为了便于升级管理，如果是windows 64位平台，初始化之前请修改HwmSdk文件夹下HwmSdk.exe文件的名称为用户自己的名称。（仅针对windows 64位平台，其他平台不涉及）
- 2、登录不是必须的步骤，在不登录的情况下也能通过会议ID和密码加入已经创建好的会议。
- 3、离开会议、结束会议等接口，当前SDK的UI上已经提供了对应的功能，如果第三方的App不需要程序控制这些功能则无需调用对应接口。
- 4、会议开始后，订阅的会议信息onConfInfoNotify通知会上报多次。

9.4.2 场景 1：初始化

描述

在第三方APP启动时，需要先初始化SDK，才能调用SDK的功能接口，该接口只需调用一次。

业务流程

SDK初始化时，如果是windows 64位平台，先修改HwmSdk文件夹下HwmSdk.exe文件的名称，其他平台不涉及；再调用init同步接口。

步骤1 修改exe文件名称（仅针对windows 64位平台，其他平台不涉及）

修改HwmSdk文件夹下HwmSdk.exe文件名称为MySdk.exe。

步骤2 接口调用

1. 构建数据结构InitInfo。
2. 调用Init接口，完成配置初始化，第1步中的数据作为参数。

----结束

示例代码

```
/**
 *初始化接口，拉起应用
 */
async goToInit() {
  let param = {
    exePath: "E:\\Hello_World\\debug\\win32\\HwmSdk\\MySdk.exe", // 只有windows 64位平台需要设置sdk
    路径, 假设HwmSdk目录在E:\\Hello_World\\debug\\win32\\, 假设exe文件名称为MySdk.exe ( 路径需要utf8编
    码)
    logPath: "E:\\Hello_World\\debug\\win32\\MySdk\\log\\", // 指定日志路径 ( 路径需要utf8编码)
    userDataPath: "E:\\Hello_World\\debug\\win32\\MySdk\\UserData\\", // 指定数据路径 ( 路径需要utf8编
    码)
    appId: "602b111272ad4b6989e683c27e6e87d4"
  };
  const apiService = new ApiService();
  let setResult = await apiService.init(param);
  if (setResult != 0) {
    window.electron.ipcRenderer.send("show-error-alert", "init error = " + setResult.ret);
  }
}
/**
 * ApiService中init定义
 */
init(initInfo) {
  console.log("init, in param = ", initInfo);
  let ret = this.uisdkService.init(initInfo);
  console.log("init, out data = ", ret);
}
```

```
return ret;
}
```

📖 说明

典型场景和接口参考中的示例代码仅作为示例用的伪代码，不能直接使用。

9.4.3 场景 2：登录

描述

可用App ID（App ID申请请参考“[App ID鉴权介绍](#)”）或华为云会议帐号登录华为云会议。创建会议等接口不支持在未登录状态下调用，若要使用完整的会议功能，必须先登录华为云会议。

📖 说明

在未登录状态下，只能加入已经创建的会议。

业务流程

使用SDK登录时，以App ID登录方式为例，调用loginByAppId接口，同时设置接口回调onLoginResult。

步骤1 接口调用

1. 组装数据结构AppIdAuthInfo和接口回调函数onLoginResult。
2. 调用loginByAppId进行登录，第1步中的数据作为参数。

步骤2 处理回调函数

处理回调函数onLoginResult。

----结束

示例代码

```
/**
 * 登录处理
 */
async goToLoginByAppId(){
  /**
   * 此处省略第三方服务器做的数据转换，最后获取到nonce和signature
   */
  let param = {
    thirdUserId: "admin@1",
    userName: "Mike",
    signature: signature,
    nonce: nonce,
    expireTime: 1598398920,
    userEmail: "xxx@xxx.com",
    userPhone: "1234567890",
  }
  const apiService = new ApiService();
  let setResult = await apiService.loginByAppId(param);
  if(setResult.ret == 0){
    this.props.history.push('/main');
  }else{
    window.electron.ipcRenderer.send("show-error-alert", "loginByAppId error = " + setResult.ret);
  }
}
```

```
/**
 * ApiService中loginByAppld接口定义
 */
async loginByAppld(appldAuthInfo) {
    return new Promise((resolve) => {
        let resultCallback = (ret, reason, loginResult) => {
            console.log("loginByAppld, out data = ", { ret, reason, loginResult });
            resolve({ ret, reason, loginResult });
        };
        console.log("loginByAppld, in param = ", appldAuthInfo);
        this.uisdkService.getLoginApi().loginByAppld(appldAuthInfo, resultCallback);
    });
}
```

9.4.4 场景 3：预约会议

描述

在使用华为云会议帐号登录后，可以调用预约会议接口预约会议。预约会议时，除了填写预约会议所需的会议参数以外，还可以添加与会者。

业务流程

调用bookConf接口预约会议，然后处理接口回调函数onBookConfResult和订阅的onConfListChanged通知。

若要预约个人会议ID的会议或者云会议室的会议，则需要先调用getVmrList接口查询个人会议ID和云会议室信息，然后处理接口回调函数onGetVmrListResult返回的数据，该数据可用于预约会议。

步骤1 接口调用

1. 组装数据结构BookConfParam（包含结构内Array<AttendeeBaseInfo>）和接口回调函数onBookConfResult。
2. 调用bookConf开始创建预约会议，第1步中的数据作为参数。

步骤2 处理回调函数

处理回调函数onBookConfResult。

步骤3 处理消息通知

处理消息通知onConfListChanged。

----结束

示例代码

```
/**
 * 预约会议
 */
async goToBookConf() {
    let _attendees = [{
        nickName: "Mike",
        number: "+991116003543",
        email: "",
        sms: "",
        thirdUserId: "",
        isMute: true,
        isAutoInvite: true,
    }];
}
```

```

let param = {
  startTime: 1598398920; //utc时间戳, 单位秒, 如果获取的时间是本地时间, 需要转换成utc时间
  duration: 30; //会议持续时长(分钟)
  isSendCalendar: true,
  confCommonParam: {
    subject: "我的预约会议", // 此处应是utf-8编码字符串
    mediaType: MediaType.HWM_MEDIA_TYPE_VIDEO,
    needPassword: true,
    timezone: 56, // 时区, 56表示东八区
    confAllowJoinUser: ConfAllowJoinUserType.CONF_ALLOW_JOIN_ANYONE,
    isSendSms: true,
    isSendEmail: true,
    attendees: _attendees,
    isAutoRecord: false, //默认会议不开启自动录制
  }
}
const apiService = new ApiService();
let setResult = await apiService.bookConf(param);
if (setResult.ret != 0) {
  window.electron.ipcRenderer.send("show-error-alert", "bookConf error = " + setResult.ret);
}
}

```

```

/**
 * ApiService中bookConf接口定义
 */
async bookConf(bookConfParam) {
  return new Promise((resolve) => {
    let resultCallback = (ret, reason, confDetail) => {
      console.log("bookConf, out data = ", { ret, reason, confDetail });
      resolve({ ret, reason, confDetail });
    };
    console.log("bookConf, in param = ", bookConfParam);
    this.uisdkService.getConfMgrApi().bookConf(bookConfParam, resultCallback);
  });
}

```

```

/**
 * 订阅会议列表信息onConfListChanged通知
 */
uisdkService.getConfMgrApi().setOnConfListChangedCB(NotifyService.handleOnConfListChanged);

/**
 * NotifyService中handleOnConfListChanged定义
 */
static handleOnConfListChanged(confListInfo) {
  console.log('OnConfListChanged', ' ', confListInfo = ', confListInfo);
  let _confListInfo = window.sessionStorage.getItem("confListInfo");
  _confListInfo = _confListInfo ? JSON.parse(_confListInfo) : [];
  console.log("sessionStorage confList:", _confListInfo);
  let confListNew = confListInfo.confListItem;
  let _data = JSON.stringify(confListNew ? confListNew : []);
  window.sessionStorage.setItem("confListInfo", _data);
}

```

9.4.5 场景 4：编辑会议

描述

预约会议成功后，可以通过编辑会议接口来编辑会议信息，在编辑会议时可以添加与会者。

业务流程

得到会议列表以后，若要编辑会议，需要先通过getConfDetail接口查询会议详情，在接口回调onGetConfDetailResult中处理详情数据，再调用editConf接口对会议详情进行修改，并处理接口回调函数onEditConfResult和订阅的onConfListChanged通知。

步骤1 获取会议详情

1. 组装数据结构GetConfDetailParam和接口回调函数onGetConfDetailResult
2. 调用GetConfDetail接口，第1步中的数据作为参数。

步骤2 处理回调函数

处理回调onGetConfDetailResult，得到会议详情数据。

步骤3 调用编辑会议接口

1. 使用步骤2获取的数据，组装数据结构EditConfParam（包含结构内的Array<AttendeeBaseInfo>）和回调函数onEditConfResult。
2. 调用editConf接口，第1步中的数据作为入参。

步骤4 处理回调函数

处理回调函数onEditConfResult。

步骤5 处理消息通知

处理消息通知onConfListChanged。

---结束

示例代码

```
/**
 * 获取会议详情
 */
async getConfInfoById(confId) {
    let param = {
        confId: "989156631",
    }
    const apiService = new ApiService();
    let setResult = await apiService.getConfDetail(param);
    if (setResult.ret == 0) {
        let _data = setResult.confDetail;
        console.log("confDetail:", _data);
        // 此处省略缓存会议详情数据代码
    } else {
        window.electron.ipcRenderer.send("show-error-alert", "getConfDetail error = " + setResult.ret);
    }
}
```

```
/**
 * ApiService中getConfDetail接口定义
 */
async getConfDetail(getConfDetailParam) {
    return new Promise((resolve) => {
        let resultCallback = (ret, reason, confDetail) => {
            console.log("getConfDetail, out data = ", { ret, reason, confDetail });
            resolve({ ret, reason, confDetail });
        };
        console.log("getConfDetail, in param = ", getConfDetailParam);
        this.uisdkService.getConfMgrApi().getConfDetail(getConfDetailParam, resultCallback);
    });
}
```



```

    });
}

/**
 * 编辑会议接口
 */
async editConf() {
    // 从缓存中得到会议详情数据存在state中,从state中读取
    let { confId, subject, startTime, duration, mediaType, needPassword, timezone, vmrFlag, vmrId,
        confAllowJoinUser, isSendSms, isSendEmail,
        isSendCalendar, attendees, isAutoRecord, isOpenWaitingRoom, vmrConfId } = this.state;
    let _attendees = attendees ? JSON.parse(attendees) : "";
    let param = {
        confId: confId,
        startTime: 1598398920, // utc时间戳
        duration: 0 * 60 + 30, // 会议时长
        isSendCalendar: isSendCalendar,
        confCommonParam: {
            subject: subject,
            mediaType: mediaType,
            needPassword: needPassword,
            timezone: 56, // 本地时区, 用户依据自身情况自己调整, 56东八区
            vmrId: vmrId,
            vmrConfIdType: (vmrConfId != "") ? 0 : 1,
            confAllowJoinUser: ConfAllowJoinUserType.CONF_ALLOW_JOIN_ANYONE, // 取编辑会议上面的允许
            入会用户类型
            isSendSms: isSendSms,
            isSendEmail: isSendEmail,
            attendees: _attendees,
            isAutoRecord: false; // 默认会议不启用自动录制
            isOpenWaitingRoom: isOpenWaitingRoom,
        }
    }
    const apiService = new ApiService();
    let setResult = await apiService.editConf(param);
    if (setResult.ret != 0) {
        window.electron.ipcRenderer.send("show-error-alert", "editConf error = " + setResult.ret);
    }
}
/**
 * ApiService中editConf接口定义
 */
async editConf(editConfParam) {
    return new Promise((resolve) => {
        let resultCallback = (ret, reason) => {
            console.log("editConf, out data = ", { ret, reason });
            resolve({ ret, reason });
        };
        console.log("editConf, in param = ", editConfParam);
        this.uisdkService.getConfMgrApi().editConf(editConfParam, resultCallback);
    });
}
}

```

📖 说明

onConfListChanged事件通知跟预约会议场景相同, 请参考[预约会议的示例代码](#)。

9.4.6 场景 5: 取消会议

描述

预约会议成功后, 可以取消会议接口来取消会议。

业务流程

若要取消会议，可以调用cancelConf接口，并处理回调函数onCancelConfResult和订阅的onConfListChanged通知。。

步骤1 接口调用

1. 组装数据结构CancelConfParam和接口回调函数onCancelConfResult。
2. 调用cancelConf取消会议，第1步中的数据作为参数。

步骤2 处理回调函数

处理回调函数onCancelConfResult。

步骤3 处理消息通知

处理消息通知onConfListChanged。

----结束

示例代码

```
/**
 * 取消会议接口
 */
async cancelConfById() {
  let param = {
    confId: "989156631",
  }
  console.log(param);
  const apiService = new ApiService();
  let setResult = await apiService.cancelConf(param);
  if (setResult.ret != 0) {
    window.electron.ipcRenderer.send("show-error-alert", "cancelConf error = " + setResult.ret);
  }
  this.props.history.push('/main');
}
```

```
/**
 * ApiService中cancelConf接口定义
 */
async cancelConf(cancelConfParam) {
  return new Promise((resolve) => {
    let resultCallback = (ret, reason) => {
      console.log("cancelConf, out data = ", { ret, reason });
      resolve({ ret, reason });
    };
    console.log("cancelConf, in param = ", cancelConfParam);
    this.uisdkService.getConfMgrApi().cancelConf(cancelConfParam, resultCallback);
  });
}
```

说明

onConfListChanged事件通知跟预约会议场景相同，请参考[预约会议的示例代码](#)。

9.4.7 场景 6：创建会议

描述

在使用华为云会议帐号登录后，可以调用创建会议接口创建立即会议。创建会议时可以携带与会人信息，也可以不携带。

业务流程

若要创建个人会议ID的会议或者云会议室的会议，则需要先调用getVmrList接口查询个人会议ID和云会议室信息，然后处理回调函数onGetVmrListResult返回的数据，该数据可用于创建会议。

使用SDK创建立即会议时，先调用createConf接口，然后处理接口回调函数onCreateConfResult和订阅的消息通知onConfStateChanged、onConfInfoNotify。

步骤1 接口调用

1. 组装数据结构CreateConfInfo（包含结构内Array<AttendeeBaseInfo>，可不带）和接口回调函数onCreateConfResult。
2. 调用createConf接口开始创建，第1步中的数据作为参数。

步骤2 处理回调函数

处理回调函数onCreateConfResult。

步骤3 处理消息通知

处理消息通知onConfStateChanged。

步骤4 处理消息通知

处理消息通知onConfInfoNotify。

步骤5 处理消息通知

处理消息通知onConfListChanged。

----结束

示例代码

```
/**
 * 创建会议
 */
async goToCreateConf() {
  let param = {
    subject: "我的会议",
    mediaType: MediaType.HWM_MEDIA_TYPE_VIDEO,
    needPassword: true,
    confAllowJoinUser: ConfAllowJoinUserType.CONF_ALLOW_JOIN_ANYONE,
    isMicOn: true, // 进入会议后默认麦克风开启
    isCameraOn: false, // 进入会议后默认摄像头关闭
  }
  const apiService = new ApiService();
  let setResult = await apiService.createConf(param);
  if (setResult.ret != 0) {
    window.electron.ipcRenderer.send("show-error-alert", "createConf error = " + setResult.ret);
  }
}
/**
 * ApiService中createConf接口定义
 */
async createConf(createConfInfo) {
  return new Promise((resolve) => {
    let resultCallback = (ret, reason, createConfResult) => {
      console.log("createConf, out data = ", { ret, reason, createConfResult });
      resolve({ ret, reason, createConfResult });
    };
    console.log("createConf, in param = ", createConfInfo);
    this.uidkService.getConfCtrlApi().createConf(createConfInfo, resultCallback);
  });
}
```

```
});  
}
```

📖 说明

Electron SDK接口中的字符串参数，比如会议主题、与会人姓名等，需要编码成UTF8，否则接口会报错。

```
/**  
 * 订阅会议状态消息onConfStateChanged通知  
 */  
uisdkService.getConfCtrlApi().setOnConfStateChangedCB(NotifyService.handleOnConfStateChanged);  
  
/**  
 * NotifyService中handleOnConfStateChanged定义  
 */  
static handleOnConfStateChanged(confStateInfo) {  
    console.log('OnConfStateChanged', confStateInfo);  
}
```

```
/**  
 * 订阅会议信息消息onConfInfoNotify通知  
 */  
uisdkService.getConfCtrlApi().setOnConfInfoNotifyCB(NotifyService.handleOnConfInfoNotify);  
  
/**  
 * NotifyService中handleOnConfInfoNotify定义  
 */  
static handleOnConfInfoNotify(confInfo) {  
    console.log('OnConfInfoNotify', confInfo);  
}
```

📖 说明

onConfListChanged事件通知跟预约会议场景相同，请参考[预约会议的示例代码](#)。
Electron SDK回调函数或者消息通知中的字符串都是UTF8编码的。

9.4.8 场景 7：添加与会人

描述

在会议中，可以调用添加与会人的接口把与会人加入到会议中。

业务流程

使用SDK添加与会人时，先调用addAttendee接口，然后处理接口回调函数onAddAttendeeResult。

步骤1 接口调用

1. 组装数据结构AddAttendeeInfo和接口回调函数onAddAttendeeResult。
2. 调用addAttendee接口，第1步中的数据作为参数。

步骤2 处理回调函数

处理回调函数onAddAttendeeResult。

----结束

示例代码

```
/**  
 * 添加与会人
```

```

*/
async goToAddAttendee() {
  let _attendees = [{
    nickName: "Mike",
    number: "+991116003543"
  }];
  let param = {
    attendees: _attendees
  }
  const apiService = new ApiService();
  let setResult = await apiService.addAttendee(param);
  if(setResult.ret != 0){
    window.electron.ipcRenderer("show-error-alert", "addAttendee error = " + setResult.ret);
  }
}
/**
 * ApiService中getConfDetail接口定义
 */
*/
async addAttendee(addAttendeeInfo) {
  return new Promise((resolve) => {
    let resultCallback = (ret, reason) => {
      console.log("addAttendee, out data = ", { ret, reason });
      resolve({ ret, reason });
    };
    console.log("addAttendee, in param = ", addAttendeeInfo);
    this.uisdkService.getConfCtrlApi().addAttendee(addAttendeeInfo, resultCallback);
  });
}
}

```

9.4.9 场景 8：加入会议

描述

在华为云会议帐号登录后或者登录前，都可以使用该接口通过会议ID和密码加入会议。

业务流程

使用SDK加入已经存在的会议时，先调用joinConfById接口，然后处理接口回调函数onJoinConfByIdResult和订阅的消息通知onConfStateChanged、onConfInfoNotify。

步骤1 接口调用

1. 组装数据结构JoinConfByIdInfo和接口回调函数onJoinConfByIdResult。
2. 调用joinConfById接口，第1步中的数据作为参数。

步骤2 处理回调函数

处理回调函数onJoinConfByIdResult。

步骤3 处理消息通知

处理消息通知onConfStateChanged。

步骤4 处理消息通知

处理消息通知onConfInfoNotify。

----结束

示例代码

```

async goToJoinConfById() {
  let { confId, password, nickName, isSpeakerOff, isMicOn, isCameraOn } = this.state;

```

```

let param = {
  confId: "989156631",
  password: "807766",
  nickName: "杭州会场",
  isSpeakerOff: false, // 入会后默认扬声器不关闭
  isMicOn: true, // 入会后默认麦克风打开
  isCameraOn: false, // 入会后默认摄像头关闭
}
const apiService = new ApiService();
let setResult = await apiService.joinConfById(param);
if(setResult.ret != 0){
  window.electron.ipcRenderer.send("show-error-alert", "joinConfById error = " + setResult.ret);
}
}
}
/**
 * ApiService中joinConfById接口定义
 */
async joinConfById(joinConfByIdInfo) {
  return new Promise((resolve) => {
    let resultCallback = (ret, reason) => {
      console.log("joinConfById, out data = ", { ret, reason });
      resolve({ ret, reason });
    };
    console.log("joinConfById, in param = ", joinConfByIdInfo);
    this.uisdkService.getConfCtrlApi().joinConfById(joinConfByIdInfo, resultCallback);
  });
}
}

```

📖 说明

onConfStateChanged和onConfInfoNotify的事件通知跟创建会议场景相同，请参考[创建会议的示例代码](#)。

9.4.10 场景 9：邮件链接入会

描述

通过邮件会议通知中的会议链接入会时（会议链接如：<https://bmeeting.huaweicloud.com/#/j/981924330/e11ddec62ee760734fcf2ba55b79937eac2aa68677caf659>），链接里面的的是会议ID（即举例中的981924330）和会议的Random码（即举例中的e11ddec62ee760734fcf2ba55b79937eac2aa68677caf659）。

在华为云会议帐号登录后或者登录前，都可以使用该接口通过分享会议链接中的会议ID和48位密码映射值Random加入会议。

业务流程

使用SDK加入已经存在的会议时，先调用joinConfByRandom接口，然后处理接口回调函数onJoinConfByRandomResult和订阅的消息通知onConfStateChanged、onConfInfoNotify

步骤1 接口调用

1. 组装数据结构JoinConfByRandomInfo和回调函数onJoinConfByRandomResult。
2. 调用joinConfByRandom开始入会，第1步中的数据作为参数。

步骤2 处理回调函数

处理回调函数onJoinConfByRandomResult。

步骤3 处理消息通知

处理消息通知onConfStateChanged。

步骤4 处理消息通知

处理消息通知onConfInfoNotify。

----结束

示例代码

```
// 通过会议ID和Random加入会议
async goToJoinConfByRandom(){
  let { confId, random, nickName, isSpeakerOff, isMicOn, isCameraOn } = this.state;
  let param = {
    confId: "988148352",
    random: "c4f92bc34e685bbba6c23589741be0431997f30da5d6e47b",
    nickName: "杭州会场",
    isSpeakerOff: false, // 入会后默认扬声器不关闭
    isMicOn: true, // 入会后默认麦克风打开
    isCameraOn: false, // 入会后默认摄像头关闭
  }
  const apiService = new ApiService();
  let setResult = await apiService.joinConfByRandom(param);
  if(setResult.ret != 0){
    window.electron.ipcRenderer.send("show-error-alert", "joinConfByRandom error = " + setResult.ret);
  }
}
```

```
/**
 * ApiService中joinConfByRandom接口定义
 */
async joinConfByRandom(joinConfByRandomInfo) {
  return new Promise((resolve) => {
    let resultCallback = (ret, reason) => {
      console.log("joinConfByRandom, out data = ", { ret, reason });
      resolve({ ret, reason });
    };
    console.log("joinConfByRandom, in param = ", joinConfByRandomInfo);
    this.uisdkService.getConfCtrlApi().joinConfByRandom(joinConfByRandomInfo, resultCallback);
  });
}
```

📖 说明

onConfStateChanged和onConfInfoNotify的事件通知跟创建会议场景相同，请参考[创建会议的示例代码](#)。

48位密码映射值为分享会议/复制会议信息中的链接所携带的部分字符串。

<https://bmeeting.huaweicloud.com/#/j/987297025/227b2d708b7e5a24ce8de2159772239b1672c0a85ba70d85>

9.4.11 场景 10：发起呼叫

描述

在使用华为云会议帐号登录后，可以调用发起呼叫接口发起一路呼叫。

业务流程

使用SDK发起时，先调用startCall接口，然后处理接口回调函数onStartCallResult和订阅的消息通知onCallStateChanged。

步骤1 接口调用

1. 组装数据结构StartCallInfo和回调函数onStartCallResult。
2. 调用startCall接口发起呼叫，第1步中的数据作为参数。

步骤2 处理回调函数

处理回调函数onStartCallResult。

步骤3 处理消息通知

处理消息通知onCallStateChanged。

步骤4 处理消息通知

处理消息通知onCallEndedNotify。

----结束

示例代码

```
/**
 * 发起呼叫
 */
async goToStartCall() {
  let { callerNickName, calleeNickName, calleeNumber, calleeThirdUserId, mediaType } = this.state;
  let param = {
    callerInfo: {
      nickName: "" // 主叫人姓名
    },
    calleeInfo: {
      nickName: "Mike", // 被叫人姓名
      number: "+991116003543", // 被叫人号码
      thirdUserId: "" // 被叫人三方账号
    },
    mediaType: mediaType,
  }
  const apiService = new ApiService();
  let setResult = await apiService.startCall(param);
  if (setResult.ret != 0) {
    window.electron.ipcRenderer.send("show-error-alert", "startCall error = " + setResult.ret);
  }
}
/**
 * ApiService中startCall定义
 */
async startCall(startCallInfo) {
  return new Promise((resolve) => {
    let resultCallback = (ret, reason) => {
      console.log("startCall, out data = ", { ret, reason });
      resolve({ ret, reason });
    };
    console.log("startCall, in param = ", startCallInfo);
    this.uisdkService.getCallApi().startCall(startCallInfo, resultCallback);
  });
}
```

说明

Electron SDK接口中的字符串参数，比如被叫方姓名、呼叫方姓名等，需要编码成UTF8，否则接口会报错。

```
/**
 * 订阅通话状态消息onCallStateChanged通知
 */
uisdkService.getCallApi().setOnCallStateChangedCB(NotifyService.handleOnCallStateChanged);
/**
```



```

* NotifyService中handleOnCallStateChanged定义
*/
static handleOnCallStateChanged(callStateInfo) {
    console.log('OnCallStateChanged', callStateInfo);
}

```

📖 说明

Electron SDK回调函数或者消息通知中的字符串都是UTF8编码的。

9.4.12 场景 11：定制会中“邀请”按钮

描述

考虑到大部分使用SDK二次开发的场景下，第三方App不会使用华为云会议的通讯录，因此Electron SDK提供的会议界面上“邀请”按钮，只提供了一个按钮界面，并没有提供完整邀请与会者的功能，需要开发者自己实现。

业务流程

使用SDK实现“邀请”功能时，在SDK初始化完成后先调用config接口，然后处理接口回调函数onConfigResult。会议中，在“邀请”按钮单击后，再处理订阅的消息通知onClickInjectBtnNotify。

步骤1 接口调用

1. 组装包含配置信息的字符串和接口回调函数onConfigResult。
2. 调用config接口，第1步中的数据作为参数。

步骤2 处理回调函数

处理回调函数onConfigResult。

步骤3 处理消息通知

当“邀请”按钮单击后，再处理订阅的消息通知onClickInjectBtnNotify

----结束

示例代码

```

/**
 * 配置邀请按钮
 */
async setInvite(){
    let param = {
        "frame": {
            "confMenu": {
                "toolBar": {
                    "button": [{
                        "buttonId": "invite",
                        "showAsAction": "ifRoom",
                        "isCustomizedClick": true
                    }]
                }
            }
        }
    }
    const apiService = new ApiService();
    let setResult = await apiService.config(JSON.stringify(param));
    if(setResult.ret != 0){
        window.electron.ipcRenderer.send("show-error-alert", "config error = " + setResult.ret);
    }
}

```

```

    }
  }
  /**
   * ApiService中config接口定义
   */
  async config(uiConfig) {
    return new Promise((resolve) => {
      let resultCallback = (ret, reason) => {
        console.log("config, out data = ", { ret, reason });
        resolve({ ret, reason });
      };
      console.log("config, in param = ", uiConfig);
      this.uisdkService.getUIConfigApi().config(uiConfig, resultCallback);
    });
  }

  /**
   * 订阅按钮点击消息onClickInjectBtnNotify通知
   */
  uisdkService.getUIConfigApi().setOnClickInjectBtnNotifyCB(NotifyService.handleOnClickInjectBtnNotify);

  /**
   * NotifyService中handleOnClickInjectBtnNotify定义
   */
  static handleOnClickInjectBtnNotify(injectBtnInfo) {
    console.log('OnClickInjectBtnNotify', injectBtnInfo);
    switch (injectBtnInfo.injectBtn) {
      // “邀请”按钮点击消息通知
      case ClickInjectBtn.HWM_CLICK_INJECT_BTN_INVITE:
        // 显示添加与会人界面
        window.location.hash = "#/confCtrlPage/addAttendee";
        window.electron.ipcRenderer.send('show-add-attendee-window');
        break;
      default:
        break;
    }
  }
}

```

9.4.13 场景 12：注销登录

描述

当需要退出当前的华为云会议帐号的登录时，可以调用该接口完成已登录华为云帐号帐号的注销。

业务流程

使用SDK注销登录时，先调用logout接口，然后处理接口回调函数onLogoutResult。

步骤1 接口调用

1. 组装接口回调函数onLogoutResult。
2. 调用Logout进行注销登录，第1步中的数据作为参数。

步骤2 处理回调函数

处理回调函数onLogoutResult。

----结束

示例代码

```

/**
 * 注销

```

```

*/
async Logout(){
    const apiService = new ApiService();
    let setResult = await apiService.logout();
    if(setResult.ret === 0){
        console.log("logoutresult:", setResult.logoutResult);
        window.sessionStorage.setItem("confListInfo", "");
        this.props.history.push('/');
    }else{
        window.electron.ipcRenderer.send("show-error-alert", "logout error = " + setResult.ret);
    }
}

/**
 * ApiService中logout接口定义
 */
*/
async logout() {
    return new Promise((resolve) => {
        let resultCallback = (ret, reason, logoutResult) => {
            console.log("logout, out data = ", { ret, reason, logoutResult });
            resolve({ ret, reason, logoutResult });
        };
        console.log("logout, in param = ");
        this.uisdkService.getLoginApi().logout(resultCallback);
    });
}

```

9.4.14 场景 13：退出

描述

在第三方APP退出时，需要去初始化SDK，否则名称为HwmSdk.exe的进程（ Electron SDK的windows 64位下的主进程）一直在后台运行，仅针对windows 64位平台。

业务流程

需要退出SDK时，调用Exit同步接口。

步骤1 接口调用

1. 调用Exit退出SDK。

----结束

示例代码

```

/**
 * 退出sdk
 */
*/
async exitSdk(){
    const apiService = await new ApiService();
    let setResult = await apiService.exit();
    if(setResult != 0){
        window.electron.ipcRenderer.send("show-error-alert", "exit error = " + setResult);
    }
}

/**
 * ApiService中exit接口定义
 */
*/
exit() {
    console.log("exit, in param = ");
    let ret = this.uisdkService.exit();
    console.log("exit, out data = ", ret);
}

```

```
return ret;
}
```

9.5 接口参考

9.5.1 UISDK 接口管理对象

UISDK接口管理对象

9.5.1.1 初始化

接口功能

该接口用于启动并且初始化SDK。注意事项：1、调用该接口之前请先修改HwmSdk文件夹下HwmSdk.exe文件的名称；2、该接口用于启动并初始化SDK，在调用其他接口之前必须先调用该接口并等待初始化回调结果；3、每次调用初始化接口会重新启动SDK，之前所有设置与操作将不复存在。

函数原型

init(initInfo: InitInfo): SDKERR

请求参数

表 9-2 入参说明

参数名称	是否必选	参数类型	参数说明
initInfo	Y	InitInfo	初始化信息

表 9-3 InitInfo

参数名称	是否必选	参数类型	参数说明
exePath	Y	string	SDK启动路径，SDK压缩包中HwmSdk文件夹下的exe文件所在的绝对路径。使用前请将SDK压缩包中HwmSdk文件夹下HwmSdk.exe文件名修改为期望的名称。如果HwmSdk文件夹的路径为：D:\MyApp\SDK\HwmSdk\，exe文件的名称为MySdk.exe，则exePath=D:\MyApp\SDK\HwmSdk\MySdk.exe。注：需要UTF8编码。(仅对win64平台生效)

参数名称	是否必选	参数类型	参数说明
logPath	Y	string	SDK日志路径，如果不设置，则使用默认路径。注：需要UTF8编码。
userDataPath	Y	string	SDK用户数据路径，如果不设置，则使用默认路径。注：需要UTF8编码。(仅对win平台生效)
appId	Y	string	App ID。App ID的申请，请参考“App ID鉴权介绍”章节。
siteType	N	SiteType	站点类型。默认使用国内站点（值置0）。
logKeepDays	N	number	配置项-日志保存天数，输入有效范围为3-30，默认值0。0为不按天数删除日志。注意：输入1和2会转换为3，大于30的值会被转换为30。
preferredCamera	N	string	优选摄像头列表，填写后，初次打开摄像头按列表中的顺序选择第一个可用的摄像头。可以不填，不填按照默认顺序打开可用的摄像头。(仅对win平台生效)
appUrl	N	string	官网下载渠道链接(仅对mac平台生效)

表 9-4 SiteType 枚举说明

枚举名称	枚举值	枚举说明
HWM_SITE_TYPE_CHINA	0	国内站点
HWM_SITE_TYPE_AP	1	亚太站点

返回值

SDKERR请参考[错误码参考](#)

9.5.1.2 退出

接口功能

该接口用于退出SDK。注意事项：1、该接口用于退出SDK，调用该接口之后SDK不可使用；2、应用退出时请调用该接口，若不调用该接口，则SDK无法退出，应用退出后SDK进程将继续存在；3、mac请在杀进程场景调用。

函数原型

exit(): SDKERR

返回值

SDKERR请参考[错误码参考](#)

9.5.1.3 获取登录服务对象

接口功能

获取登录服务对象

函数原型

getLoginApi(): ILogin

返回值

ILogin

9.5.1.4 获取会议控制服务对象

接口功能

获取会议控制服务对象

函数原型

getConfCtrlApi(): IConfCtrl

返回值

IConfCtrl

9.5.1.5 获取会议管理服务对象

接口功能

获取会议管理服务对象

函数原型

getConfMgrApi(): IConfMgr

返回值

IConfMgr

9.5.1.6 获取配置服务对象

接口功能

获取配置服务对象

函数原型

getConfigApi(): IConfig

返回值

IConfig

9.5.1.7 获取界面配置服务对象

接口功能

获取界面配置服务对象

函数原型

getUIConfigApi(): IUIConfig

返回值

IUIConfig

9.5.1.8 获取呼叫服务对象

接口功能

获取呼叫服务对象

函数原型

getCallApi(): ICall

返回值

ICall

9.5.1.9 订阅连接断开通知

接口功能

收到该通知时，表示与SDK的连接断开了。

函数原型

```
setOnSdkDisconnectedNotifyCB(onSdkDisconnectedNotify: () => void): void
```

请求参数

表 9-5 入参说明

参数名称	是否必选	参数类型	参数说明
onSdkDisconnectedNotify	Y	() => void	收到该通知时，表示与SDK的连接断开了。

返回值

void

9.5.2 登录管理对象

登录管理对象

9.5.2.1 订阅登录被踢通知

接口功能

收到该通知时，表示当前使用的帐号被其他终端踢出，退出登录状态，需要重新登录。

函数原型

```
setOnKickedOutNotifyCB(onKickedOutNotify: () => void): void
```

请求参数

表 9-6 入参说明

参数名称	是否必选	参数类型	参数说明
onKickedOutNotify	Y	() => void	收到该通知时，表示当前使用的帐号被其他终端踢出，退出登录状态，需要重新登录。

返回值

void

9.5.2.2 订阅企业配置通知

接口功能

收到该通知时，表示企业配置信息变更。

函数原型

```
setOnCorpConfigNotifyCB(onCorpConfigNotify: (configInfo: CorpConfigInfo) => void): void
```

请求参数

表 9-7 入参说明

参数名称	是否必选	参数类型	参数说明
onCorpConfigNotify	Y	(configInfo: CorpConfigInfo) => void	收到该通知时，表示企业配置信息变更。

返回值

void

通知方法参数说明：

```
(configInfo: CorpConfigInfo) => void
```

表 9-8 入参说明

参数名称	参数类型	参数说明
configInfo	CorpConfigInfo	配置信息

表 9-9 CorpConfigInfo

参数名称	参数类型	参数说明
pstnEnable	boolean	是否支持手机呼叫
smsEnable	boolean	是否支持短信通知
recordEnable	boolean	是否支持会议录制

参数名称	参数类型	参数说明
pstnNumber	string	pstn呼叫号码
allowGuestStartConf	boolean	是否允许来宾提前启动会议默认值
allowGuestStartConfTime	number	允许来宾提前入会时间范围默认值(单位: 分钟) 0-随时, n-提前n分钟启动会议
enableShowAddressBookStructure	boolean	是否支持通讯录显示 (仅对win平台生效)

9.5.2.3 订阅登录错误通知

接口功能

收到该通知时，表示登录失败。

函数原型

```
setOnLoginErrorNotifyCB(onLoginErrorNotify: (ret: SDKERR, reason: string) => void): void
```

请求参数

表 9-10 入参说明

参数名称	是否必选	参数类型	参数说明
onLoginErrorNotify	Y	(ret: SDKERR, reason: string) => void	收到该通知时，表示登录失败。

返回值

void

通知方法参数说明:

(ret: SDKERR, reason: string) => void

表 9-11 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码

参数名称	参数类型	参数说明
reason	string	错误描述

9.5.2.4 账号密码登录

接口功能

该接口用于账号密码登录。注意事项：1、创建会议，被邀入会等功能只有在登录后才能使用，若不登录，将无法体验完整会议功能。

函数原型

```
loginByAccount(accountAuthInfo: AccountAuthInfo, onLoginResult?: (ret: SDKERR, reason: string, loginResult: LoginResult) => void): void
```

请求参数

表 9-12 入参说明

参数名称	是否必选	参数类型	参数说明
accountAuthInfo	Y	AccountAuthInfo	账号密码鉴权信息
onLoginResult	Y	(ret: SDKERR, reason: string, loginResult: LoginResult) => void	登录回调

表 9-13 AccountAuthInfo

参数名称	是否必选	参数类型	参数说明
account	Y	string	登录帐号
password	Y	string	登录密码

返回值

void

回调方法参数说明：

```
(ret: SDKERR, reason: string, loginResult: LoginResult) => void
```

表 9-14 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述
loginResult	LoginResult	登录结果信息

表 9-15 LoginResult

参数名称	参数类型	参数说明
userId	string	UUID(由系统分配的真实的唯一用户标识)
account	string	账号
thirdAccount	string	第三方帐号

9.5.2.5 AppId 登录

接口功能

该接口用于AppId登录。注意事项：1、创建会议，被邀入会等功能只有在登录后才能使用，若不登录，将无法体验完整会议功能。

函数原型

```
loginByAppId(appIdAuthInfo: AppIdAuthInfo, onLoginResult?: (ret: SDKERR, reason: string, loginResult: LoginResult) => void): void
```

请求参数

表 9-16 入参说明

参数名称	是否必选	参数类型	参数说明
appIdAuthInfo	Y	AppIdAuthInfo	AppId鉴权信息
onLoginResult	Y	(ret: SDKERR, reason: string, loginResult: LoginResult) => void	登录回调

表 9-17 AppIdAuthInfo

参数名称	是否必选	参数类型	参数说明
corpId	N	string	企业Id, 仅在SP模式下需要填写, 单企业模式不要填写, 否则会鉴权失败
thirdUserId	Y	string	第三方账号唯一标识
userName	N	string	用户名称 (如不填, 默认与 thirdUserId一致)
signature	Y	string	从第三方服务端获取的鉴权签名
nonce	Y	string	从第三方服务端获取的Nonce值, 用于华为云服务端验证鉴权签名有效性
expireTime	Y	number	从第三方服务端获取的鉴权签名有效期的时间戳 (s), 0表示永不超时
userEmail	N	string	用户邮箱信息
userPhone	N	string	用户手机号码
deptCode	N	string	部门id

返回值

void

回调方法参数说明:

(ret: SDKERR, reason: string, loginResult: LoginResult) => void

表 9-18 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述
loginResult	LoginResult	登录结果信息

表 9-19 LoginResult

参数名称	参数类型	参数说明
userId	string	UUID(由系统分配的真实的唯一用户标识)
account	string	账号
thirdAccount	string	第三方帐号

9.5.2.6 SSO 登录

接口功能

该接口用于SSO登录。注意事项：1、创建会议，被邀入会等功能只有在登录后才能使用，若不登录，将无法体验完整会议功能。

函数原型

loginBySSO(ssoAuthInfo: SSOAuthInfo, onLoginResult?: (ret: SDKERR, reason: string, loginResult: LoginResult) => void): void

请求参数

表 9-20 入参说明

参数名称	是否必选	参数类型	参数说明
ssoAuthInfo	Y	SSOAuthInfo	SSO鉴权信息
onLoginResult	Y	(ret: SDKERR, reason: string, loginResult: LoginResult) => void	登录回调

表 9-21 SSOAuthInfo

参数名称	是否必选	参数类型	参数说明
code	Y	string	授权码
domain	Y	string	企业域名
authServerType	Y	AuthServerType	鉴权服务类型

表 9-22 AuthServerType 枚举说明

枚举名称	枚举值	枚举说明
AUTH_SERVER_TYPE_OAUTH2	0	单点登录

返回值

void

回调方法参数说明:

(ret: SDKERR, reason: string, loginResult: LoginResult) => void

表 9-23 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述
loginResult	LoginResult	登录结果信息

表 9-24 LoginResult

参数名称	参数类型	参数说明
userId	string	UUID(由系统分配的真实的唯一用户标识)
account	string	账号
thirdAccount	string	第三方帐号

9.5.2.7 注销

接口功能

该接口用于注销已登录的华为云会议帐号。注意事项：1、请在已登录的场景下调用。

函数原型

logout(onLogoutResult?: (ret: SDKERR, reason: string, logoutResult: LogoutResult) => void): void

请求参数

表 9-25 入参说明

参数名称	是否必选	参数类型	参数说明
onLogoutResult	Y	(ret: SDKERR, reason: string, logoutResult: LogoutResult) => void	登出回调

返回值

void

回调方法参数说明：

(ret: SDKERR, reason: string, logoutResult: LogoutResult) => void

表 9-26 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述
logoutResult	LogoutResult	登出结果信息

表 9-27 LogoutResult

参数名称	参数类型	参数说明
userId	string	UUID(由系统分配的真实的唯一用户标识)
account	string	账号
thirdAccount	string	第三方帐号

9.5.2.8 设置网络模式

接口功能

该接口用于设置网络模式。注意事项：1、企业内的App无法直接访问公网（即媒体节点下沉和管理节点代理下沉场景）的情况下才需要调用该接口。如华为云会议的媒体

节点下沉部署到企业，并且管理节点也是通过企业内的代理访问，需要将网络模式设置成“企业内外接入”；2、该接口非必需调用，若不设置，则使用默认模式。

函数原型

```
setNetworkMode(networkInfo: NetWorkModeInfo, onSetNetworkModeResult?:  
(ret: SDKERR, reason: string) => void): void
```

请求参数

表 9-28 入参说明

参数名称	是否必选	参数类型	参数说明
networkInfo	Y	NetWorkModeInfo	网络模式信息
onSetNetworkModeResult	Y	(ret: SDKERR, reason: string) => void	设置网络模式结果回调

表 9-29 NetWorkModeInfo

参数名称	是否必选	参数类型	参数说明
accessType	Y	NetworkAccesssType	网络接入类型

表 9-30 NetworkAccessType 枚举说明

枚举名称	枚举值	枚举说明
ACCESS_TYPE_AUTO	0	默认值，目前和企业外网接入模式一致。
ACCESS_TYPE_INNER	1	公司内网接入，用于媒体和信令下沉场景，app从内网接入，信令/媒体消息走代理服务器才能访问云会议服务端
ACCESS_TYPE_OUTER	2	公司外网接入，用于非媒体和信令下沉场景，互联网接入，信令/媒体消息可以直接访问云会议服务端

返回值

void

回调方法参数说明：

(ret: SDKERR, reason: string) => void

表 9-31 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.2.9 上传头像

接口功能

该接口用于上传当前自己的头像到会议服务器的通讯录。注意事项：1、请在已登录状态下调用；2、上传的图片文件格式仅支持jpg/png格式，大小需要限制在200KB以下；3、为了头像展示效果，建议图片宽高比为1:1

函数原型

uploadSelfAvatar(selfAvatarInfo: SelfAvatarInfo, onUploadSelfAvatarResult?: (ret: SDKERR, reason: string) => void): void

请求参数

表 9-32 入参说明

参数名称	是否必选	参数类型	参数说明
selfAvatarInfo	Y	SelfAvatarInfo	自己的头像信息
onUploadSelfAvatarResult	Y	(ret: SDKERR, reason: string) => void	上传个人头像结果回调

表 9-33 SelfAvatarInfo

参数名称	是否必选	参数类型	参数说明
path	Y	string	图片全路径

返回值

void

回调方法参数说明：

(ret: SDKERR, reason: string) => void

表 9-34 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.3 会议控制操作对象

会议控制操作对象

9.5.3.1 订阅会议状态通知

接口功能

收到该通知时，表示会议状态发生变更。

函数原型

setOnConfStateChangedCB(onConfStateChanged: (confStateInfo: ConfStateInfo)
=> void): void

请求参数

表 9-35 入参说明

参数名称	是否必选	参数类型	参数说明
onConfStateChanged	Y	(confStateInfo: ConfStateInfo) => void	收到该通知时，表示会议状态发生变更。

返回值

void

通知方法参数说明：

(confStateInfo: ConfStateInfo) => void

表 9-36 入参说明

参数名称	参数类型	参数说明
confStateInfo	ConfStateInfo	会议状态信息

表 9-37 ConfStateInfo

参数名称	参数类型	参数说明
state	State	会议状态。

表 9-38 State 枚举说明

枚举名称	枚举值	枚举说明
HWM_STATE_OFF	0	关闭状态
HWM_STATE_ON	1	开启状态

9.5.3.2 订阅会议结束通知

接口功能

收到该通知时，表示会议已结束。

函数原型

```
setOnConfEndedNotifyCB(onConfEndedNotify: (confEndInfo: ConfEndInfo) => void): void
```

请求参数

表 9-39 入参说明

参数名称	是否必选	参数类型	参数说明
onConfEndedNotify	Y	(confEndInfo: ConfEndInfo) => void	收到该通知时，表示会议已结束。

返回值

void

通知方法参数说明：

(confEndInfo: ConfEndInfo) => void

表 9-40 入参说明

参数名称	参数类型	参数说明
confEndInfo	ConfEndInfo	已结束的会议信息

表 9-41 ConfEndInfo

参数名称	参数类型	参数说明
reason	ConfEndReason	离会原因。

表 9-42 ConfEndReason 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_END_REASON_LEAVE_CONF_HANGUP	0	离开会议。
HWM_CONF_END_REASON_STOP_CONF_HANGUP	1	会议已结束。
HWM_CONF_END_REASON_HOST_HANGUP	2	被主持人移出会场。
HWM_CONF_END_REASON_SESSION_TIMER_TIMEOUT	3	网络中断。
HWM_CONF_END_REASON_NOSTREAM_HANGUP	4	网络异常。
HWM_CONF_END_REASON_CORP_CONFERENCERESOURCE_HAS_BEEN_RUN_OUT	5	企业资源已达上限。
HWM_CONF_END_REASON_CONF_ONLY_SAME_ENTERPRISE	6	只允许企业内用户加入会议。
HWM_CONF_END_REASON_CONF_ONLY_INVITED	7	只允许被邀请者加入会议。
HWM_CONF_END_REASON_CONF_IS_LOCKED	8	会议被锁定。
HWM_CONF_END_ERROR_VMR_NOT_BEEN_HOLD	9	VMR会议来宾不允许激活会议。

枚举名称	枚举值	枚举说明
HWM_CONF_END_REASON_VMR_CONF_CONFLICT	10	VMR会议冲突。
HWM_CONF_END_REASON_SERVER_ERROR	11	服务器异常。
HWM_CONF_END_REASON_NUMBER_OF_PARTICIPANTS_EXCEEDS_UPPER_LIMIT	12	与会方超出专用VMR套餐的最大与会人数
HWM_CONF_END_REASON_PARTICIPANT_REACH_MAX_NUMBER	1110 7250 4	达到系统单个会议人数上限

9.5.3.3 订阅会议信息通知

接口功能

收到该通知时，表示会议中的信息发生变更。

函数原型

```
setOnConflInfoNotifyCB(onConflInfoNotify: (conflInfo: ConflInfo) => void): void
```

请求参数

表 9-43 入参说明

参数名称	是否必选	参数类型	参数说明
onConflInfoNotify	Y	(conflInfo: ConflInfo) => void	收到该通知时，表示会议中的信息发生变更。

返回值

void

通知方法参数说明：

```
(conflInfo: ConflInfo) => void
```

表 9-44 入参说明

参数名称	参数类型	参数说明
confInfo	ConfInfo	会议信息

表 9-45 ConfInfo

参数名称	参数类型	参数说明
hostJoinUri	string	主持人入会链接(仅对mac平台生效)
guestJoinUri	string	来宾加入会议uri链接，可以分享通过浏览器打开进行入会。
confId	string	会议id
selfConfRole	ConfRole	本人在会议中的角色。主席、来宾等。
hostPwd	string	会议主席密码。
guestPwd	string	会议来宾密码。
subject	string	会议主题。
startTime	number	会议开始时间。
endTime	number	会议结束时间。
schedulerName	string	会议预定者名称。
mediaType	MediaType	会议类型。(仅对mac平台生效)
vmrConfId	string	用于云会议显示的ID。(仅对mac平台生效)
corpId	string	企业id。(仅对mac平台生效)
isOtherCorpConf	boolean	是否是外部企业的会议：如果是加入外部企业的会议或者匿名入会是true，否则是false
isCreator	boolean	是否是会议预订者(仅对win平台生效)
isInBreakoutSubConf	boolean	是否在讨论组中

表 9-46 ConfRole 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_ROLE_ATTENDEE	0	普通与会者

枚举名称	枚举值	枚举说明
HWM_CONF_ROLE_HOST	1	主席
HWM_CONF_ROLE_AUDIENCE	2	观众
HWM_CONF_ROLE_COHOST	3	联席主持人
HWM_CONF_ROLE_WAITING	4	等候室成员

表 9-47 MediaType 枚举说明

枚举名称	枚举值	枚举说明
HWM_MEDIA_TYPE_AUDIO	0	音频类型
HWM_MEDIA_TYPE_VIDEO	1	视频类型

9.5.3.4 订阅共享状态通知

接口功能

收到该通知时，表示当前共享状态发生变更。

函数原型

```
setOnShareStateChangedCB(onShareStateChanged: (shareStateInfo: ShareStateInfo) => void): void
```

请求参数

表 9-48 入参说明

参数名称	是否必选	参数类型	参数说明
onShareStateChanged	Y	(shareStateInfo: ShareStateInfo) => void	收到该通知时，表示当前共享状态发生变更。

返回值

void

通知方法参数说明：

```
(shareStateInfo: ShareStateInfo) => void
```


表 9-49 入参说明

参数名称	参数类型	参数说明
shareStateInfo	ShareStateInfo	共享状态信息

表 9-50 ShareStateInfo

参数名称	参数类型	参数说明
state	State	共享状态。
nickName	string	共享者名称。当state为HWM_STATE_ON时才有数据。
number	string	共享者号码。当state为HWM_STATE_ON时才有数据。
monitor	Monitor	共享显示器类型（主屏、副屏）。
shareType	ShareType	共享类型。
appHandle	number	程序句柄。

表 9-51 State 枚举说明

枚举名称	枚举值	枚举说明
HWM_STATE_OFF	0	关闭状态
HWM_STATE_ON	1	开启状态

表 9-52 Monitor 枚举说明

枚举名称	枚举值	枚举说明
HWM_MONITOR_MAIN	1	主显示器
HWM_MONITOR_SECONDARY	2	副显示器

表 9-53 ShareType 枚举说明

枚举名称	枚举值	枚举说明
HWM_SHARE_TYPE_MONITOR	0	屏幕共享

枚举名称	枚举值	枚举说明
HWM_SHARE_TYPE_APP	1	程序共享
HWM_SHARE_TYPE_WHITE_BOARD	2	白板共享

9.5.3.5 订阅个人角色变化通知

接口功能

收到该通知时，表示个人角色发生变化(仅对mac平台生效)。

函数原型

setOnSelfRoleChangedCB(onSelfRoleChanged: (role: ConfRole) => void): void

请求参数

表 9-54 入参说明

参数名称	是否必选	参数类型	参数说明
onSelfRoleChanged	Y	(role: ConfRole) => void	收到该通知时，表示个人角色发生变化(仅对mac平台生效)。

返回值

void

通知方法参数说明:

(role: ConfRole) => void

表 9-55 入参说明

参数名称	参数类型	参数说明
role	ConfRole	自己在会议中的角色

表 9-56 ConfRole 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_ROLE_ATTENDEE	0	普通与会者
HWM_CONF_ROLE_HOST	1	主席
HWM_CONF_ROLE_AUDIENCE	2	观众
HWM_CONF_ROLE_COHOST	3	联席主持人
HWM_CONF_ROLE_WAITING	4	等候室成员

9.5.3.6 创建会议

接口功能

该接口用于创建即时会议。注意事项：1、该接口默认已将自己作为主席加入会议，调用接口时，在与会者参数中不需要带上自己的信息；2、若需要在创建会议的同时邀请其他与会者，需要传入其他与会者信息。若不需要，与会者信息和与会者个数可以为空。

函数原型

```
createConf(createConfInfo: CreateConfInfo, onCreateConfResult?: (ret: SDKERR, reason: string, createConfResult: CreateConfResult) => void): void
```

请求参数

表 9-57 入参说明

参数名称	是否必选	参数类型	参数说明
createConfInfo	Y	CreateConfInfo	创会信息
onCreateConfResult	Y	(ret: SDKERR, reason: string, createConfResult: CreateConfResult) => void	创建会议回调

表 9-58 CreateConfInfo

参数名称	是否必选	参数类型	参数说明
subject	Y	string	会议主题。

参数名称	是否必选	参数类型	参数说明
mediaType	Y	MediaType	会议类型。
needPassword	Y	boolean	是否需要来宾密码（仅对随机会议ID生效）。
isAutoRecord	N	boolean	是否开启自动录制会议。开启自动录制会议开关，内部会默认支持录制会议，录制会议开关的打开与关闭并不影响。说明：仅对云录制生效，客户端本地录制不生效。
allowRecord	N	boolean	是否支持录制会议。（仅对win平台生效）说明：仅对云录制生效，客户端本地录制不生效。
confAllowJoinUser	N	ConfAllowJoinUserType	允许入会范围限制。默认所有用户。
vmrId	N	string	云会议室唯一ID。预约随机会议时，设置为空字符串。
vmrConfIdType	N	VmrConfIdType	云会议室的会议ID类型
guestPwd	N	string	来宾密码，如果空则服务器随机生成（仅对随机会议ID生效）。
isOpenWaitingRoom	N	boolean	等候室开启开关。说明：需要等候室功能开通后才能生效
duration	N	number	会议时长，单位分钟，最小值15分钟，最大值1440分钟。注：不填时由服务端取默认会议时长
isMicOn	N	boolean	是否开启麦克风，true开启麦克风；false关闭麦克风。
isCameraOn	N	boolean	是否开启摄像头，true开启摄像头；false关闭摄像头。
numOfAttendee	Y	number	与会者个数。若不需要带与会者，参数值可以为0。
attendees	Y	AttendeeBaseInfo	与会者信息。若不需要带与会者，参数值可以为空。

表 9-59 MediaType 枚举说明

枚举名称	枚举值	枚举说明
HWM_MEDIA_TYPE_AUDIO	0	音频类型
HWM_MEDIA_TYPE_VIDEO	1	视频类型

表 9-60 ConfAllowJoinUserType 枚举说明

枚举名称	枚举值	枚举说明
CONF_ALLOW_JOIN_ANYONE	0	所有用户。
CONF_ALLOW_JOIN_LOGINED_USER	1	已登录用户。
CONF_ALLOW_JOIN_IN_COMPANY_USER	2	企业内用户。
CONF_ALLOW_JOIN_INVITED_USER	3	被邀请用户。

表 9-61 VmrConfIdType 枚举说明

枚举名称	枚举值	枚举说明
HWM_VMR_CONF_ID_TYPE_FIXED	0	云会议室的会议ID类型是固定类型。
HWM_VMR_CONF_ID_TYPE_RANDOM	1	云会议室的会议ID类型是随机类型。

表 9-62 AttendeeBaseInfo

参数名称	是否必选	参数类型	参数说明
nickName	Y	string	与会者名称。

参数名称	是否必选	参数类型	参数说明
number	Y	string	号码。如果填分配给帐号的sip号码（如+99111244216210249）则呼叫该帐号的App；如果填PSTN号码（如18700000000），则通过VoIP网关呼叫该号码，前提是该企业已开通PSTN呼叫权限。与thirdUserId二选一，帐号密码鉴权时使用。
thirdUserId	Y	string	第三方用户ID。与number二选一，App ID鉴权时使用。
accountId	N	string	华为云会议帐号
email	N	string	电子邮箱地址，用于发送邮件通知。
sms	N	string	短信通知手机号码，用于发送短信通知。
isAutoInvite	N	boolean	是否自动外邀。
isMute	N	boolean	是否闭音。
role	N	ConfRole	会议角色。
type	N	AttendeeType	成员类型

表 9-63 ConfRole 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_ROLE_ATTENDEE	0	普通与会者
HWM_CONF_ROLE_HOST	1	主席
HWM_CONF_ROLE_AUDIENCE	2	观众
HWM_CONF_ROLE_COHOST	3	联席主持人
HWM_CONF_ROLE_WAITING	4	等候室成员

表 9-64 AttendeeType 枚举说明

枚举名称	枚举值	枚举说明
ATTENDEE_TYPE_NORMAL	0	语音、高清、标清与会者地址

枚举名称	枚举值	枚举说明
ATTENDEE_TYPE_TELEPRESENCE	1	智真与会者地址类型，单屏、三屏智真均属此类
ATTENDEE_TYPE_TERMINAL	2	会议室或硬终端
ATTENDEE_TYPE_OUTSIDE	3	外部与会人
ATTENDEE_TYPE_CUSTOMNUMBER	4	custom number
ATTENDEE_TYPE_MOBILE	5	软终端用户手机
ATTENDEE_TYPE_ANONYMOUS	6	匿名用户
ATTENDEE_TYPE_TELEPHONE	7	电话用户
ATTENDEE_TYPE_BOARD	8	白板用户
ATTENDEE_TYPE_IDEAHUB	9	海思芯片的大屏设备
ATTENDEE_TYPE_HWVISION	10	智慧屏
ATTENDEE_TYPE_WELINKC	11	welink大众版

返回值

void

回调方法参数说明：

(ret: SDKERR, reason: string, createConfResult: CreateConfResult) => void

表 9-65 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述
createConfResult	CreateConfResult	创建会议结果

表 9-66 CreateConfResult

参数名称	参数类型	参数说明
confId	string	会议id

参数名称	参数类型	参数说明
vmrConfId	string	VMR会议id(仅对mac平台生效)
hostPwd	string	主席密码
guestPwd	string	来宾密码
mediaType	MediaType	会议媒体类型(仅对mac平台生效)
guestJoinUri	string	来宾链接
selfConfRole	ConfRole	本人在会议中的角色
subject	string	会议主题
startTime	number	会议开始时间, 精度秒(0时区)
endTime	number	会议结束时间
schedulerName	string	预约者名字

9.5.3.7 加入会议

接口功能

该接口用于通过会议ID和密码加入已经存在的会议。在登录和未登录状态下都可以调用该接口。

函数原型

```
joinConfById(joinConfByIdInfo: JoinConfByIdInfo, onJoinConfByIdResult?: (ret: SDKERR, reason: string) => void): void
```

请求参数

表 9-67 入参说明

参数名称	是否必选	参数类型	参数说明
joinConfByIdInfo	Y	JoinConfByIdInfo	id入会信息
onJoinConfByIdResult	Y	(ret: SDKERR, reason: string) => void	通过会议id入会回调

表 9-68 JoinConfByIdInfo

参数名称	是否必选	参数类型	参数说明
confId	Y	string	会议ID。
password	N	string	会议密码。若希望以主席身份入会，必须传入主席密码。若希望以来宾身份入会，会议要求来宾密码时，必须传入来宾密码，会议不要求来宾密码时，可以为空字符串。
nickName	N	string	会议昵称。在已登录状态下可为空，已登录时空将使用登录的用户名作为会场名。匿名入会必须传该名字
isSpeakerOff	N	boolean	是否关闭扬声器，true关闭扬声器；false不关闭扬声器。
isMicOn	N	boolean	是否开启麦克风，true开启麦克风；false关闭麦克风。
isCameraOn	N	boolean	是否开启摄像头，true开启摄像头；false关闭摄像头。
isStopConflictConf	N	boolean	强制结束和当前VMR会议资源冲突的会议（必需是vmr资源的拥有者才有权限，否则无效）

返回值

void

回调方法参数说明：

(ret: SDKERR, reason: string) => void

表 9-69 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.3.8 Random 入会

接口功能

通过邮件会议通知中的会议链接入会时（会议链接如：<https://bmeeting.huaweicloud.com/#/j/981924330/e11ddec62ee760734fcf2ba55b79937eac2aa68677caf659>），链接里面的会议ID（即举例中的981924330）和会议的Random码（即举例中的e11ddec62ee760734fcf2ba55b79937eac2aa68677caf659）。第三方App需要调用本接口加入会议。

函数原型

```
joinConfByRandom(joinConfByRandomInfo: JoinConfByRandomInfo,
onJoinConfByRandomResult?: (ret: SDKERR, reason: string) => void): void
```

请求参数

表 9-70 入参说明

参数名称	是否必选	参数类型	参数说明
joinConfByRandomInfo	Y	JoinConfByRandomInfo	random入会信息
onJoinConfByRandomResult	Y	(ret: SDKERR, reason: string) => void	通过会议random入会回调

表 9-71 JoinConfByRandomInfo

参数名称	是否必选	参数类型	参数说明
confId	Y	string	会议ID。
random	N	string	48位会议密码映射值。来源于分享会议信息中的会议链接尾部。会议不要求来宾密码时，可以为空字符串。
nickName	N	string	会议昵称。在已登录状态下可为空，已登录时空将使用登录的用户名作为会场名。匿名入会必须传该名字
isSpeakerOff	N	boolean	是否关闭扬声器，true关闭扬声器；false不关闭扬声器。
isMicOn	N	boolean	是否开启麦克风，true开启麦克风；false关闭麦克风。

参数名称	是否必选	参数类型	参数说明
isCameraOn	N	boolean	是否开启摄像头，true开启摄像头；false关闭摄像头。

返回值

void

回调方法参数说明：

(ret: SDKERR, reason: string) => void

表 9-72 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.3.9 添加与会者

接口功能

该接口用于在会议或呼叫中添加与会者。注意事项：1、不在会议或呼叫中，调用该接口将返回失败；2、呼叫中调用该接口，呼叫则转成会议。

函数原型

addAttendee(addAttendeeInfo: AddAttendeeInfo, onAddAttendeeResult?: (ret: SDKERR, reason: string) => void): void

请求参数

表 9-73 入参说明

参数名称	是否必选	参数类型	参数说明
addAttendeeInfo	Y	AddAttendeeInfo	添加与会者信息
onAddAttendeeResult	Y	(ret: SDKERR, reason: string) => void	添加与会者回调

表 9-74 AddAttendeeInfo

参数名称	是否必选	参数类型	参数说明
attendeeSize	Y	number	添加与会者的个数
attendees	Y	AttendeeBaseInfo	添加与会者的列表

表 9-75 AttendeeBaseInfo

参数名称	是否必选	参数类型	参数说明
nickName	Y	string	与会者名称。
number	Y	string	号码。如果填分配给帐号的sip号码（如+99111244216210249）则呼叫该帐号的App；如果填PSTN号码（如18700000000），则通过VoIP网关呼叫该号码，前提是该企业已开通PSTN呼叫权限。与thirdUserId二选一，帐号密码鉴权时使用。
thirdUserId	Y	string	第三方用户ID。与number二选一，App ID鉴权时使用。
accountId	N	string	华为云会议帐号
email	N	string	电子邮箱地址，用于发送邮件通知。
sms	N	string	短信通知手机号码，用于发送短信通知。
isAutoInvite	N	boolean	是否自动外邀。
isMute	N	boolean	是否闭音。
role	N	ConfRole	会议角色。
type	N	AttendeeType	成员类型

表 9-76 ConfRole 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_ROLE_ATTENDEE	0	普通与会者
HWM_CONF_ROLE_HOST	1	主席

枚举名称	枚举值	枚举说明
HWM_CONF_ROLE_AUDIENCE	2	观众
HWM_CONF_ROLE_COHOST	3	联席主持人
HWM_CONF_ROLE_WAITING	4	等候室成员

表 9-77 AttendeeType 枚举说明

枚举名称	枚举值	枚举说明
ATTENDEE_TYPE_NORMAL	0	语音、高清、标清与会者地址
ATTENDEE_TYPE_TELEPRESENCE	1	智真与会者地址类型，单屏、三屏智真均属此类
ATTENDEE_TYPE_TERMINAL	2	会议室或硬终端
ATTENDEE_TYPE_OUTSIDE	3	外部与会人
ATTENDEE_TYPE_CUSTOMNUMBER	4	custom number
ATTENDEE_TYPE_MOBILE	5	软终端用户手机
ATTENDEE_TYPE_ANONYMOUS	6	匿名用户
ATTENDEE_TYPE_TELEPHONE	7	电话用户
ATTENDEE_TYPE_BOARD	8	白板用户
ATTENDEE_TYPE_IDEAHUB	9	海思芯片的大屏设备
ATTENDEE_TYPE_HWVISION	10	智慧屏
ATTENDEE_TYPE_WELINKC	11	welink大众版

返回值

void

回调方法参数说明：

(ret: SDKERR, reason: string) => void

表 9-78 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.3.10 离开会议

接口功能

该接口用于离开会议。注意事项：1、不在会议中不应调用此接口；2、调用该接口离开会议，会议不会结束，在会议存在期间可以再次入会。

函数原型

leaveConf(onLeaveConfResult?: (ret: SDKERR, reason: string) => void): void

请求参数

表 9-79 入参说明

参数名称	是否必选	参数类型	参数说明
onLeaveConfResult	Y	(ret: SDKERR, reason: string) => void	离开会议回调

返回值

void

回调方法参数说明：

(ret: SDKERR, reason: string) => void

表 9-80 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.3.11 结束会议

接口功能

该接口用于结束会议。注意事项：1、不在会议中不应调用此接口；2、调用该接口，会议将结束，无法再次加入该会议。

函数原型

```
endConf(onEndConfResult?: (ret: SDKERR, reason: string) => void): void
```

请求参数

表 9-81 入参说明

参数名称	是否必选	参数类型	参数说明
onEndConfResult	Y	(ret: SDKERR, reason: string) => void	结束会议回调

返回值

void

回调方法参数说明：

```
(ret: SDKERR, reason: string) => void
```

表 9-82 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.3.12 获取与会者列表

接口功能

该接口用于获取会议中与会者列表。注意事项：1、只能在会议中使用。

函数原型

```
getAttendeeList(onGetAttendeeListResult?: (ret: SDKERR, reason: string, attendeeList: AttendeeListInfo) => void): void
```

请求参数

表 9-83 入参说明

参数名称	是否必选	参数类型	参数说明
onGetAttendeeListResult	Y	(ret: SDKERR, reason: string, attendeeList: AttendeeListInfo) => void	获取与会者列表接口回调

返回值

void

回调方法参数说明：

(ret: SDKERR, reason: string, attendeeList: AttendeeListInfo) => void

表 9-84 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述
attendeeList	AttendeeListInfo	与会者列表

表 9-85 AttendeeListInfo

参数名称	参数类型	参数说明
numOfAttendee	number	入会者列表的长度
attendees	AttendeeInfo	与会者基础信息

表 9-86 AttendeeInfo

参数名称	参数类型	参数说明
baseInfo	AttendeeBaseInfo	与会者基本信息
isSelf	boolean	是否是自己
state	ConfAttendeeState	与会者状态

参数名称	参数类型	参数说明
isAnonymous	boolean	是否是匿名入会
corpId	string	企业id，用于识别是否是本企业账号

表 9-87 AttendeeBaseInfo

参数名称	参数类型	参数说明
nickName	string	与会者名称。
number	string	号码。如果填分配给帐号的sip号码（如+99111244216210249）则呼叫该帐号的App；如果填PSTN号码（如18700000000），则通过VoIP网关呼叫该号码，前提是该企业已开通PSTN呼叫权限。与thirdUserId二选一，帐号密码鉴权时使用。
thirdUserId	string	第三方用户ID。与number二选一，App ID鉴权时使用。
accountId	string	华为云会议帐号
email	string	电子邮箱地址，用于发送邮件通知。
sms	string	短信通知手机号码，用于发送短信通知。
isAutoInvite	boolean	是否自动外邀。
isMute	boolean	是否闭音。
role	ConfRole	会议角色。
type	AttendeeType	成员类型

表 9-88 ConfRole 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_ROLE_ATTENDEE	0	普通与会者
HWM_CONF_ROLE_HOST	1	主席
HWM_CONF_ROLE_AUDIENCE	2	观众
HWM_CONF_ROLE_COHOST	3	联席主持人
HWM_CONF_ROLE_WAITING	4	等候室成员

表 9-89 AttendeeType 枚举说明

枚举名称	枚举值	枚举说明
ATTENDEE_TYPE_NORMAL	0	语音、高清、标清与会者地址
ATTENDEE_TYPE_TELEPRESENCE	1	智真与会者地址类型，单屏、三屏智真均属此类
ATTENDEE_TYPE_TERMINAL	2	会议室或硬终端
ATTENDEE_TYPE_OUTSIDE	3	外部与会人
ATTENDEE_TYPE_CUSTOMNUMBER	4	custom number
ATTENDEE_TYPE_MOBILE	5	软终端用户手机
ATTENDEE_TYPE_ANONYMOUS	6	匿名用户
ATTENDEE_TYPE_TELEPHONE	7	电话用户
ATTENDEE_TYPE_BOARD	8	白板用户
ATTENDEE_TYPE_IDEAHUB	9	海思芯片的大屏设备
ATTENDEE_TYPE_HWVISION	10	智慧屏
ATTENDEE_TYPE_WELINKC	11	welink大众版

表 9-90 ConfAttendeeState 枚举说明

枚举名称	枚举值	枚举说明
CONF_ATTENDEE_STATE_IN_CONF	0	会议中
CONF_ATTENDEE_STATE_CALLING	1	正在呼叫
CONF_ATTENDEE_STATE_JOINING	2	正在加入会议
CONF_ATTENDEE_STATE_LEAVED	3	已经离开
CONF_ATTENDEE_STATE_NO_EXIST	4	用户不存在
CONF_ATTENDEE_STATE_BUSY	5	被叫用户忙
CONF_ATTENDEE_STATE_NO_ANSWER	6	用户无应答

枚举名称	枚举值	枚举说明
CONF_ATTENDEE_STATE_REJECT	7	用户拒绝接听
CONF_ATTENDEE_STATE_CALL_FAILED	8	呼叫失败

9.5.3.13 获取观众列表

接口功能

该接口用于获取会议中观众列表。注意事项：1、只能在会议中使用；2、观众身份（可以通过会议详情消息推送获取）不能使用此接口。

函数原型

```
getAudienceList(onGetAudienceListResult?: (ret: SDKERR, reason: string, audienceList: AttendeeListInfo) => void): void
```

请求参数

表 9-91 入参说明

参数名称	是否必选	参数类型	参数说明
onGetAudienceListResult	Y	(ret: SDKERR, reason: string, audienceList: AttendeeListInfo) => void	获取观众列表接口回调

返回值

void

回调方法参数说明：

```
(ret: SDKERR, reason: string, audienceList: AttendeeListInfo) => void
```

表 9-92 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

参数名称	参数类型	参数说明
audienceList	AttendeeListInfo	观众列表

表 9-93 AttendeeListInfo

参数名称	参数类型	参数说明
numOfAttendee	number	入会者列表的长度
attendees	AttendeeInfo	与会者基础信息

表 9-94 AttendeeInfo

参数名称	参数类型	参数说明
baseInfo	AttendeeBaseInfo	与会者基本信息
isSelf	boolean	是否是自己
state	ConfAttendeeState	与会者状态
isAnonymous	boolean	是否是匿名入会
corpId	string	企业id，用于识别是否是本企业账号

表 9-95 AttendeeBaseInfo

参数名称	参数类型	参数说明
nickName	string	与会者名称。
number	string	号码。如果填分配给帐号的sip号码（如+99111244216210249）则呼叫该帐号的App；如果填PSTN号码（如18700000000），则通过VoIP网关呼叫该号码，前提是该企业已开通PSTN呼叫权限。与thirdUserId二选一，帐号密码鉴权时使用。
thirdUserId	string	第三方用户ID。与number二选一，App ID鉴权时使用。
accountId	string	华为云会议帐号
email	string	电子邮箱地址，用于发送邮件通知。
sms	string	短信通知手机号码，用于发送短信通知。

参数名称	参数类型	参数说明
isAutoInvite	boolean	是否自动外邀。
isMute	boolean	是否闭音。
role	ConfRole	会议角色。
type	AttendeeType	成员类型

表 9-96 ConfRole 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_ROLE_ATTENDEE	0	普通与会者
HWM_CONF_ROLE_HOST	1	主席
HWM_CONF_ROLE_AUDIENCE	2	观众
HWM_CONF_ROLE_COHOST	3	联席主持人
HWM_CONF_ROLE_WAITING	4	等候室成员

表 9-97 AttendeeType 枚举说明

枚举名称	枚举值	枚举说明
ATTENDEE_TYPE_NORMAL	0	语音、高清、标清与会者地址
ATTENDEE_TYPE_TELEPRESENCE	1	智真与会者地址类型，单屏、三屏智真均属此类
ATTENDEE_TYPE_TERMINAL	2	会议室或硬终端
ATTENDEE_TYPE_OUTSIDE	3	外部与会人
ATTENDEE_TYPE_CUSTOMNUMBER	4	custom number
ATTENDEE_TYPE_MOBILE	5	软终端用户手机
ATTENDEE_TYPE_ANONYMOUS	6	匿名用户
ATTENDEE_TYPE_TELEPHONE	7	电话用户
ATTENDEE_TYPE_BOARD	8	白板用户
ATTENDEE_TYPE_IDEAHUB	9	海思芯片的大屏设备

枚举名称	枚举值	枚举说明
ATTENDEE_TYPE_HWVISION	10	智慧屏
ATTENDEE_TYPE_WELINKC	11	welink大众版

表 9-98 ConfAttendeeState 枚举说明

枚举名称	枚举值	枚举说明
CONF_ATTENDEE_STATE_IN_CONF	0	会议中
CONF_ATTENDEE_STATE_CALLING	1	正在呼叫
CONF_ATTENDEE_STATE_JOINING	2	正在加入会议
CONF_ATTENDEE_STATE_LEAVED	3	已经离开
CONF_ATTENDEE_STATE_NO_EXIST	4	用户不存在
CONF_ATTENDEE_STATE_BUSY	5	被叫用户忙
CONF_ATTENDEE_STATE_NO_ANSWER	6	用户无应答
CONF_ATTENDEE_STATE_REJECT	7	用户拒绝接听
CONF_ATTENDEE_STATE_CALL_FAILED	8	呼叫失败

9.5.3.14 获取用户状态

接口功能

该接口用于获取用户当前的状态(是否在呼叫/会议中)。

函数原型

```
getUserState(onGetUserStateResult?: (ret: SDKERR, reason: string, userState:
UserState) => void): void
```

请求参数

表 9-99 入参说明

参数名称	是否必选	参数类型	参数说明
onGetUserStateResult	Y	(ret: SDKERR, reason: string, userState: UserState) => void	获取用户状态结果回调

返回值

void

回调方法参数说明：

(ret: SDKERR, reason: string, userState: UserState) => void

表 9-100 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述
userState	UserState	用户当前的状态,判断当前用户是否处于忙碌状态

表 9-101 UserState 枚举说明

枚举名称	枚举值	枚举说明
HWM_STATE_IDLE	0	空闲状态
HWM_STATE_IN_CONF	1	会中状态
HWM_STATE_IN_CALL	2	呼叫中状态

9.5.3.15 获取自己会中角色

接口功能

该接口用于获取自己会中角色(仅对mac平台生效)

函数原型

```
getSelfRole(onGetSelfRoleResult?: (ret: SDKERR, reason: string, selfRole: ConfRole) => void): void
```

请求参数

表 9-102 入参说明

参数名称	是否必选	参数类型	参数说明
onGetSelfRoleResult	Y	(ret: SDKERR, reason: string, selfRole: ConfRole) => void	获取自己会中角色结果回调

返回值

void

回调方法参数说明:

(ret: SDKERR, reason: string, selfRole: ConfRole) => void

表 9-103 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述
selfRole	ConfRole	自己在会议中的角色

表 9-104 ConfRole 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_ROLE_ATTENDEE	0	普通与会者
HWM_CONF_ROLE_HOST	1	主席
HWM_CONF_ROLE_AUDIENCE	2	观众
HWM_CONF_ROLE_COHOST	3	联席主持人
HWM_CONF_ROLE_WAITING	4	等候室成员

9.5.3.16 设置会场名

接口功能

该接口用于设置会场名。

函数原型

```
renameSelf(renameSelfInfo: RenameSelfInfo, onRenameSelfResult?: (ret: SDKERR, reason: string) => void): void
```

请求参数

表 9-105 入参说明

参数名称	是否必选	参数类型	参数说明
renameSelfInfo	Y	RenameSelfInfo	设置自己会场名信息
onRenameSelfResult	Y	(ret: SDKERR, reason: string) => void	设置会场名回调

表 9-106 RenameSelfInfo

参数名称	是否必选	参数类型	参数说明
nickName	Y	string	会场名。

返回值

void

回调方法参数说明：

```
(ret: SDKERR, reason: string) => void
```

表 9-107 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.4 会议管理对象

会议管理对象

9.5.4.1 订阅会议列表通知

接口功能

收到该通知时，表示会议列表信息发生变更。

函数原型

```
setOnConfListChangedCB(onConfListChanged: (confListInfo: ConfListInfo) => void): void
```

请求参数

表 9-108 入参说明

参数名称	是否必选	参数类型	参数说明
onConfListChanged	Y	(confListInfo: ConfListInfo) => void	收到该通知时，表示会议列表信息发生变更。

返回值

void

通知方法参数说明：

```
(confListInfo: ConfListInfo) => void
```

表 9-109 入参说明

参数名称	参数类型	参数说明
confListInfo	ConfListInfo	会议列表信息

表 9-110 ConfListInfo

参数名称	参数类型	参数说明
confListSize	number	会议列表参数信息的长度。
confListItem	ConfListItem	会议列表参数信息。

表 9-111 ConfListItem

参数名称	参数类型	参数说明
confId	string	会议ID。
subject	string	会议主题。
hostPwd	string	主席密码。
guestPwd	string	来宾密码。
audiencePwd	string	观众密码。(仅对win平台生效)
startTime	number	会议开始时间(utc时间), 单位秒。
endTime	number	会议结束时间(utc时间), 单位秒。
timezone	Timezone	时区码(仅对win平台生效)
mediaType	MediaType	媒体类型
schedulerName	string	会议预定者名称。
vmrConfId	string	用于云会议显示的ID。
hostJoinUri	string	主持人加入会议uri链接
guestJoinUri	string	来宾加入会议uri链接
audienceJoinUri	string	观众加入会议的链接(仅对win平台生效)
confStateType	ConfState	会议状态(仅对win平台生效)
isWebinar	boolean	是否是网络研讨会(仅对win平台生效)
selfConfRole	ConfRole	本人在会议中的角色
conferenceType	ConferenceType	会议类型
cycleParam	CycleConfParam	周期会议的参数
subConfSize	number	周期子会议数量
subConfParam	SubCycleConfParam	周期子会议参数

表 9-112 Timezone 枚举说明

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_DEFAULT	0	MediAx默认时区值
HWM_TIMEZONE_INTERDATE_LINE	1	日界线西

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_MIDWAY_ISLAND	2	中途岛, 萨摩亚群岛
HWM_TIMEZONE_HAWAII	3	夏威夷
HWM_TIMEZONE_ALASKA	4	阿拉斯加
HWM_TIMEZONE_PACIFIC_TIME	5	太平洋时间(美国和加拿大);蒂华纳
HWM_TIMEZONE_ARIZONA	6	亚利桑那
HWM_TIMEZONE_MOUNTAIN_TIME	7	山地时间(美国和加拿大)
HWM_TIMEZONE_CENTRAL_AMERICA	8	中美洲
HWM_TIMEZONE_CENTRAL_TIME	9	中间时间(美国和加拿大)
HWM_TIMEZONE_MEXICO_CITY	10	墨西哥城
HWM_TIMEZONE_SASKATCHEWAN	11	萨斯喀彻温
HWM_TIMEZONE_BOGOTA	12	波哥大, 利马, 基多
HWM_TIMEZONE_EASTERN_TIME	13	东部时间(美国和加拿大)
HWM_TIMEZONE_INDIANA	14	印第安纳(东部)
HWM_TIMEZONE_ATLANTIC_TIME	15	大西洋时间(加拿大)
HWM_TIMEZONE_CARACAS	16	加拉加斯, 拉巴斯
HWM_TIMEZONE_SANTIAGO	17	圣地亚哥
HWM_TIMEZONE_NEWFOUNDLAND	18	纽芬兰
HWM_TIMEZONE_BRASILIA	19	巴西利亚
HWM_TIMEZONE_BUENOS_AIRES	20	布宜诺斯艾利斯, 乔治敦
HWM_TIMEZONE_GREENLAND	21	格陵兰
HWM_TIMEZONE_MID_ATLANTIC	22	中大西洋
HWM_TIMEZONE_AZORES	23	亚速尔群岛

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_CAPEVERDE	24	佛得角群岛
HWM_TIMEZONE_MONROVIA	25	蒙罗维亚
HWM_TIMEZONE_GREENWICH_MEANTIME	26	格林威治标准时：都柏林，爱丁堡，里斯本，伦敦
HWM_TIMEZONE_AMSTERDAM	27	阿姆斯特丹，柏林，伯尔尼，罗马，斯德哥尔摩，维也纳
HWM_TIMEZONE_BELGRADE	28	贝尔格莱德，布拉迪斯拉发，布达佩斯，卢布尔亚纳，布拉格
HWM_TIMEZONE_BRUSSELS	29	布鲁塞尔，哥本哈根，马德里，巴黎
HWM_TIMEZONE_SARAJEVO	30	萨拉热窝，斯科普里，维尔纽斯，索非亚，华沙，萨格勒布
HWM_TIMEZONE_WESTCENTRAL_AFRICA	31	中非西部
HWM_TIMEZONE_ATHENS	32	雅典，伊斯坦布尔，明斯克
HWM_TIMEZONE_BUCHAREST	33	布加勒斯特
HWM_TIMEZONE_CAIRO	34	开罗
HWM_TIMEZONE_HARARE	35	哈拉雷，比勒陀利亚
HWM_TIMEZONE_HELSINKI	36	赫尔辛基，里加，塔林
HWM_TIMEZONE_JERUSALEM	37	耶路撒冷
HWM_TIMEZONE_BAGHDAD	38	巴格达
HWM_TIMEZONE_KUWAIT	39	科威特，利雅得
HWM_TIMEZONE_MOSCOW	40	莫斯科，圣彼得堡，喀山，伏尔加格勒
HWM_TIMEZONE_NAIROBI	41	内罗毕
HWM_TIMEZONE_TEHRAN	42	德黑兰
HWM_TIMEZONE_ABU_DHABI	43	阿布扎比，马斯喀特
HWM_TIMEZONE_TBILISI	44	第比利斯，埃里温
HWM_TIMEZONE_KABUL	45	喀布尔
HWM_TIMEZONE_EKATERINBURG	46	叶卡捷琳堡
HWM_TIMEZONE_ISLAMABAD	47	伊斯兰堡，卡拉奇，塔什干
HWM_TIMEZONE_CALCUTTA	48	加尔各答，马德拉斯，孟买，新德里

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_KATHMANDU	49	加德满都
HWM_TIMEZONE_ALMATY	50	阿拉木图
HWM_TIMEZONE_ASTANA	51	阿斯塔纳, 达卡
HWM_TIMEZONE_SRIJAYA	52	斯里哈亚华登尼普拉
HWM_TIMEZONE_RANGOON	53	仰光
HWM_TIMEZONE_BANGKOK	54	曼谷, 雅加达, 河内
HWM_TIMEZONE_NOVOSIBIRSK	55	新西伯利亚
HWM_TIMEZONE_BEIJING	56	北京, 重庆, 香港, 乌鲁木齐, 台北
HWM_TIMEZONE_KRASNOYARSK	57	克拉斯诺亚尔斯克, 乌兰巴托
HWM_TIMEZONE_KUALALUMPUR	58	吉隆坡, 新加坡
HWM_TIMEZONE_PERTH	59	珀斯
HWM_TIMEZONE_OSAKA	60	东京, 大阪, 札幌
HWM_TIMEZONE_SEOUL	61	汉城
HWM_TIMEZONE_YAKUTSK	62	雅库茨克
HWM_TIMEZONE_ADELAIDE	63	阿德莱德
HWM_TIMEZONE_DARWIN	64	达尔文
HWM_TIMEZONE_BRISBANE	65	布里斯班
HWM_TIMEZONE_CANBERRA	66	堪培拉, 墨尔本, 悉尼
HWM_TIMEZONE_GUAM	67	关岛, 莫尔兹比港
HWM_TIMEZONE_HOBART	68	霍巴特
HWM_TIMEZONE_VLADIVOSTOK	69	符拉迪沃斯托克
HWM_TIMEZONE_SOLOMON	70	所罗门群岛, 新喀里多尼亚
HWM_TIMEZONE_AUCKLAND	71	奥克兰, 惠灵顿
HWM_TIMEZONE_FIJI	72	富士, 堪察加半岛, 马绍尔群岛
HWM_TIMEZONE_NUKUALOFA	73	努库阿洛法
HWM_TIMEZONE_IRKUTSK	74	伊尔库茨克

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_CASABLANCA	75	卡萨布兰卡
HWM_TIMEZONE_BAKU	76	巴库
HWM_TIMEZONE_MAGADAN	77	马加丹

表 9-113 MediaType 枚举说明

枚举名称	枚举值	枚举说明
HWM_MEDIA_TYPE_AUDIO	0	音频类型
HWM_MEDIA_TYPE_VIDEO	1	视频类型

表 9-114 ConfState 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_STATE_SCHEDULE	0	预定状态。
HWM_CONF_STATE_CREATING	1	正在创建状态。
HWM_CONF_STATE_GOING	2	会议已经开始。
HWM_CONF_STATE_DESTROYED	3	会议已经关闭。

表 9-115 ConfRole 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_ROLE_ATTENDEE	0	普通与会者
HWM_CONF_ROLE_HOST	1	主席
HWM_CONF_ROLE_AUDIENCE	2	观众
HWM_CONF_ROLE_COHOST	3	联席主持人
HWM_CONF_ROLE_WAITING	4	等候室成员

表 9-116 ConferenceType 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_TYPE_COMMON	0	普通会议
HWM_CONF_TYPE_CYCLE	1	周期会议

表 9-117 CycleConfParam

参数名称	参数类型	参数说明
startDate	number	开始日期-时间戳, 精度秒(0时区)
endDate	number	结束日期-时间戳, 精度秒(0时区)
cycleType	CycleType	周期类型
interval	number	周期区间 1、周期类型选择了按天, 表示每几天召开一次, 取值范围[1,15]; 2、周期类型选择了Week, 表示每几周召开一次, 取值范围[1,5]; 3、周期类型选择了Month, interval表示隔几月, 取值范围[1,3]
listPoints	string	周期内的会议召开点。仅当按周和月时有效。数据格式(中间用逗号隔开): 1,3,5,7
preRemindDays	number	周期子会议提前通知天数。

表 9-118 CycleType 枚举说明

枚举名称	枚举值	枚举说明
CYCLE_TYPE_DAY	0	以天为周期
CYCLE_TYPE_WEEK	1	以周为周期
CYCLE_TYPE_MONTH	2	以月为周期

表 9-119 SubCycleConfParam

参数名称	参数类型	参数说明
subConfID	string	子会议唯一标识UUID
mediaType	MediaType	媒体类型

参数名称	参数类型	参数说明
startTime	number	会议起始时间
endTime	number	会议结束时间
isAutoRecord	boolean	会议开始后是否自动开启录制
recordAuthType	RecordAuthType	录播鉴权方式
confAllowJoinUser	ConfAllowJoinUserType	允许入会范围限制。默认所有用户。
allowGuestStartConf	boolean	允许来宾启动会议
allowGuestStartConfTime	number	允许来宾提前入会时间范围(单位: 分钟) 0-随时, n-提前n分钟启动会议

表 9-120 RecordAuthType 枚举说明

枚举名称	枚举值	枚举说明
REOCDR_AUTH_TYPE_ANYONE	0	可通过连接观看/下载 (不拼接nonce)
REOCDR_AUTH_TYPE_IN_COMPANY_USER	1	企业用户可观看/下载
REOCDR_AUTH_TYPE_ATTENDEE	2	与会者可观看/下载

表 9-121 ConfAllowJoinUserType 枚举说明

枚举名称	枚举值	枚举说明
CONF_ALLOW_JOIN_ANYONE	0	所有用户。
CONF_ALLOW_JOIN_LOGINED_USER	1	已登录用户。
CONF_ALLOW_JOIN_IN_COMPANY_USER	2	企业内用户。
CONF_ALLOW_JOIN_INVITED_USER	3	被邀请用户。

9.5.4.2 预定会议

接口功能

该接口用于预约会议。注意事项：1、该接口在已登录状态下才能调用；2、调用该接口，与会者列表中需要添加本会场，并且将本会场设置为主席身份；3、回调函数中会返回预约成功后的会议详情，详见HwmConfDetail结构体表。如果参数内只有会议id有值则获取会议详情失败。

函数原型

```
bookConf(bookConfParam: BookConfParam, onBookConfResult?: (ret: SDKERR, reason: string, confDetail: ConfDetail) => void): void
```

请求参数

表 9-122 入参说明

参数名称	是否必选	参数类型	参数说明
bookConfParam	Y	BookConfParam	预约会议的参数体
onBookConfResult	Y	(ret: SDKERR, reason: string, confDetail: ConfDetail) => void	预约会议回调

表 9-123 BookConfParam

参数名称	是否必选	参数类型	参数说明
startTime	Y	number	会议开始时间时间戳(utc时间)，单位是秒。
duration	Y	number	会议时长，单位分钟，最小值15分钟，最大值1425分钟。
confCommonParam	Y	ConfCommonParam	会议公用参数
isSendCalendar	N	boolean	发送邮件日历开关。

表 9-124 ConfCommonParam

参数名称	是否必选	参数类型	参数说明
subject	Y	string	会议主题。

参数名称	是否必选	参数类型	参数说明
mediaType	N	MediaType	会议类型。默认音频会议。
needPassword	N	boolean	是否需要来宾密码。默认不需要来宾密码（仅对随机会议ID生效）。
isAutoRecord	N	boolean	会议自动录制开关。默认不开启。开启会议自动录制开关，内部会默认支持录制会议，录制会议开关的打开与关闭并不影响。说明：仅对云录制生效，客户端本地录制不生效。
allowRecord	N	boolean	是否支持录制会议。（仅对win平台生效）说明：仅对云录制生效，客户端本地录制不生效。
allowLive	N	boolean	保留。编辑会议时透传会议详情此字段
timezone	Y	Timezone	时区码。用于发送会议通知。
vmrId	N	string	云会议室唯一ID。预约并发起会议时，设置为空字符串。编辑会议时不可修改，填入会议详情返回值。
confAllowJoinUser	N	ConfAllowJoinUserType	允许入会范围限制。默认所有用户。
isSendSms	N	boolean	发送短信通知开关。企业配置里面需要开启，否则不生效。
isSendEmail	N	boolean	发送邮件通知开关。
vmrConfIdType	N	VmrConfIdType	云会议室的会议ID类型。编辑会议时不可修改会议详情中vmrConfId非空，填入HWM_VMR_CONF_ID_TYPE_FIXED；会议详情中vmrConfId为空，填入HWM_VMR_CONF_ID_TYPE_RANDOM
guestPwd	N	string	来宾密码，如果空则服务器随机生成（仅对随机会议ID生效）。
isOpenWaitingRoom	N	boolean	等候室开启开关。说明：需要等候室功能开通后才能生效
numOfAttendee	Y	number	与会者列表长度。

参数名称	是否必选	参数类型	参数说明
attendees	Y	AttendeeBaseInfo	与会者列表。
allowGuestStartConf	N	boolean	允许来宾启动会议
allowGuestStartConfTime	N	number	允许来宾提前入会时间范围(单位: 分钟) 0-随时, n-提前n分钟启动会议

表 9-125 MediaType 枚举说明

枚举名称	枚举值	枚举说明
HWM_MEDIA_TYPE_AUDIO	0	音频类型
HWM_MEDIA_TYPE_VIDEO	1	视频类型

表 9-126 Timezone 枚举说明

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_DEFAULT	0	MediaX默认时区值
HWM_TIMEZONE_INTERDATE_LINE	1	日界线西
HWM_TIMEZONE_MIDWAY_ISLAND	2	中途岛, 萨摩亚群岛
HWM_TIMEZONE_HAWAII	3	夏威夷
HWM_TIMEZONE_ALASKA	4	阿拉斯加
HWM_TIMEZONE_PACIFIC_TIME	5	太平洋时间(美国和加拿大);蒂华纳
HWM_TIMEZONE_ARIZONA	6	亚利桑那
HWM_TIMEZONE_MOUNTAIN_TIME	7	山地时间(美国和加拿大)
HWM_TIMEZONE_CENTRAL_AMERICA	8	中美洲
HWM_TIMEZONE_CENTRAL_TIME	9	中间时间(美国和加拿大)

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_MEXICO_CITY	10	墨西哥城
HWM_TIMEZONE_SASKATCHEWAN	11	萨斯喀彻温
HWM_TIMEZONE_BOGOTA	12	波哥大, 利马, 基多
HWM_TIMEZONE_EASTERN_TIME	13	东部时间(美国和加拿大)
HWM_TIMEZONE_INDIANA	14	印第安纳(东部)
HWM_TIMEZONE_ATLANTIC_TIME	15	大西洋时间(加拿大)
HWM_TIMEZONE_CARACAS	16	加拉加斯, 拉巴斯
HWM_TIMEZONE_SANTIAGO	17	圣地亚哥
HWM_TIMEZONE_NEWFOUNDLAND	18	纽芬兰
HWM_TIMEZONE_BRASILIA	19	巴西利亚
HWM_TIMEZONE_BUENOS_AIRES	20	布宜诺斯艾利斯, 乔治敦
HWM_TIMEZONE_GREENLAND	21	格陵兰
HWM_TIMEZONE_MID_ATLANTIC	22	中大西洋
HWM_TIMEZONE_AZORES	23	亚速尔群岛
HWM_TIMEZONE_CAPEVERDE	24	佛得角群岛
HWM_TIMEZONE_MONROVIA	25	蒙罗维亚
HWM_TIMEZONE_GREENWICH_MEANTIME	26	格林威治标准时: 都柏林, 爱丁堡, 里斯本, 伦敦
HWM_TIMEZONE_AMSTERDAM	27	阿姆斯特丹, 柏林, 伯尔尼, 罗马, 斯德哥尔摩, 维也纳
HWM_TIMEZONE_BELGRADE	28	贝尔格莱德, 布拉迪斯拉发, 布达佩斯, 卢布尔亚纳, 布拉格
HWM_TIMEZONE_BRUSSELS	29	布鲁塞尔, 哥本哈根, 马德里, 巴黎
HWM_TIMEZONE_SARAJEVO	30	萨拉热窝, 斯科普里, 维尔纽斯, 索非亚, 华沙, 萨格勒布
HWM_TIMEZONE_WESTCENTRAL_AFRICA	31	中非西部

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_ATHENS	32	雅典, 伊斯坦布尔, 明斯克
HWM_TIMEZONE_BUCHAREST	33	布加勒斯特
HWM_TIMEZONE_CAIRO	34	开罗
HWM_TIMEZONE_HARARE	35	哈拉雷, 比勒陀利亚
HWM_TIMEZONE_HELSINKI	36	赫尔辛基, 里加, 塔林
HWM_TIMEZONE_JERUSALEM	37	耶路撒冷
HWM_TIMEZONE_BAGHDAD	38	巴格达
HWM_TIMEZONE_KUWAIT	39	科威特, 利雅得
HWM_TIMEZONE_MOSCOW	40	莫斯科, 圣彼得堡, 喀山, 伏尔加格勒
HWM_TIMEZONE_NAIROBI	41	内罗毕
HWM_TIMEZONE_TEHRAN	42	德黑兰
HWM_TIMEZONE_ABU_DHABI	43	阿布扎比, 马斯喀特
HWM_TIMEZONE_TBILISI	44	第比利斯, 埃里温
HWM_TIMEZONE_KABUL	45	喀布尔
HWM_TIMEZONE_EKATERINBURG	46	叶卡捷琳堡
HWM_TIMEZONE_ISLAMABAD	47	伊斯兰堡, 卡拉奇, 塔什干
HWM_TIMEZONE_CALCUTTA	48	加尔各答, 马德拉斯, 孟买, 新德里
HWM_TIMEZONE_KATHMANDU	49	加德满都
HWM_TIMEZONE_ALMATY	50	阿拉木图
HWM_TIMEZONE_ASTANA	51	阿斯塔纳, 达卡
HWM_TIMEZONE_SRIJAYA	52	斯里哈亚华登尼普拉
HWM_TIMEZONE_RANGOON	53	仰光
HWM_TIMEZONE_BANGKOK	54	曼谷, 雅加达, 河内
HWM_TIMEZONE_NOVOSIBIRSK	55	新西伯利亚
HWM_TIMEZONE_BEIJING	56	北京, 重庆, 香港, 乌鲁木齐, 台北
HWM_TIMEZONE_KRASNOYARSK	57	克拉斯诺亚尔斯克, 乌兰巴托

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_KUALA_LUMPUR	58	吉隆坡，新加坡
HWM_TIMEZONE_PERTH	59	珀斯
HWM_TIMEZONE_OSAKA	60	东京，大阪，札幌
HWM_TIMEZONE_SEOUL	61	汉城
HWM_TIMEZONE_YAKUTSK	62	雅库茨克
HWM_TIMEZONE_ADELAIDE	63	阿德莱德
HWM_TIMEZONE_DARWIN	64	达尔文
HWM_TIMEZONE_BRISBANE	65	布里斯班
HWM_TIMEZONE_CANBERRA	66	堪培拉，墨尔本，悉尼
HWM_TIMEZONE_GUAM	67	关岛，莫尔兹比港
HWM_TIMEZONE_HOBART	68	霍巴特
HWM_TIMEZONE_VLADIVOSTOK	69	符拉迪沃斯托克
HWM_TIMEZONE_SOLOMON	70	所罗门群岛，新喀里多尼亚
HWM_TIMEZONE_AUCKLAND	71	奥克兰，惠灵顿
HWM_TIMEZONE_FIJI	72	富士，堪察加半岛，马绍尔群岛
HWM_TIMEZONE_NUKUALOFA	73	努库阿洛法
HWM_TIMEZONE_IRKUTSK	74	伊尔库茨克
HWM_TIMEZONE_CASABLANCA	75	卡萨布兰卡
HWM_TIMEZONE_BAKU	76	巴库
HWM_TIMEZONE_MAGADAN	77	马加丹

表 9-127 ConfAllowJoinUserType 枚举说明

枚举名称	枚举值	枚举说明
CONF_ALLOW_JOIN_ANYONE	0	所有用户。
CONF_ALLOW_JOIN_LOGINED_USER	1	已登录用户。

枚举名称	枚举值	枚举说明
CONF_ALLOW_JOIN_IN_COMP ANY_USER	2	企业内用户。
CONF_ALLOW_JOIN_INVITED_ USER	3	被邀请用户。

表 9-128 VmrConfIdType 枚举说明

枚举名称	枚举值	枚举说明
HWM_VMR_CONF_ID_TYPE_FI XED	0	云会议室的会议ID类型是固定类型。
HWM_VMR_CONF_ID_TYPE_R ANDOM	1	云会议室的会议ID类型是随机类型。

表 9-129 AttendeeBaseInfo

参数名称	是否必选	参数类型	参数说明
nickName	Y	string	与会者名称。
number	Y	string	号码。如果填分配给帐号的sip号码（如+99111244216210249）则呼叫该帐号的App；如果填PSTN号码（如18700000000），则通过VoIP网关呼叫该号码，前提是该企业已开通PSTN呼叫权限。与thirdUserId二选一，帐号密码鉴权时使用。
thirdUserId	Y	string	第三方用户ID。与number二选一，App ID鉴权时使用。
accountId	N	string	华为云会议帐号
email	N	string	电子邮箱地址，用于发送邮件通知。
sms	N	string	短信通知手机号码，用于发送短信通知。
isAutoInvite	N	boolean	是否自动外邀。
isMute	N	boolean	是否闭音。
role	N	ConfRole	会议角色。

参数名称	是否必选	参数类型	参数说明
type	N	AttendeeType	成员类型

表 9-130 ConfRole 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_ROLE_ATTENDEE	0	普通与会者
HWM_CONF_ROLE_HOST	1	主席
HWM_CONF_ROLE_AUDIENCE	2	观众
HWM_CONF_ROLE_COHOST	3	联席主持人
HWM_CONF_ROLE_WAITING	4	等候室成员

表 9-131 AttendeeType 枚举说明

枚举名称	枚举值	枚举说明
ATTENDEE_TYPE_NORMAL	0	语音、高清、标清与会者地址
ATTENDEE_TYPE_TELEPRESENCE	1	智真与会者地址类型，单屏、三屏智真均属此类
ATTENDEE_TYPE_TERMINAL	2	会议室或硬终端
ATTENDEE_TYPE_OUTSIDE	3	外部与会人
ATTENDEE_TYPE_CUSTOMNUMBER	4	custom number
ATTENDEE_TYPE_MOBILE	5	软终端用户手机
ATTENDEE_TYPE_ANONYMOUS	6	匿名用户
ATTENDEE_TYPE_TELEPHONE	7	电话用户
ATTENDEE_TYPE_BOARD	8	白板用户
ATTENDEE_TYPE_IDEAHUB	9	海思芯片的大屏设备
ATTENDEE_TYPE_HWVISION	10	智慧屏
ATTENDEE_TYPE_WELINKC	11	welink大众版

返回值

void

回调方法参数说明:

(ret: SDKERR, reason: string, confDetail: ConfDetail) => void

表 9-132 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述
confDetail	ConfDetail	会议详情

表 9-133 ConfDetail

参数名称	参数类型	参数说明
confListInfo	ConfListItem	会议基本信息
vmrId	string	云会议室唯一ID。
vmrFlag	boolean	是否是vmr会议。
isAutoRecord	boolean	是否开启自动录制会议。
allowRecord	boolean	是否支持录制会议
allowLive	boolean	保留。编辑会议时透传会议详情此字段
needPassword	boolean	是否需要密码。
isSendSms	boolean	是否发送短信通知。
isSendEmail	boolean	是否发送邮件通知。
isSendCalendar	boolean	是否发送日历通知。
confAllowJoinUser	ConfAllowJoinUserType	会议准入设置。
numOfAttendee	number	与会者列表长度。
attendees	AttendeeBaseInfo	与会者列表。
isOpenWaitingRoom	boolean	是否开启等候室
allowGuestStartConf	boolean	允许来宾启动会议

参数名称	参数类型	参数说明
allowGuestStartConftime	number	允许来宾提前入会时间范围(单位: 分钟) 0-随时, n-提前n分钟启动会议
vmrConfIdType	VmrConfIdType	云会议室的会议ID类型

表 9-134 ConfListItem

参数名称	参数类型	参数说明
confId	string	会议ID。
subject	string	会议主题。
hostPwd	string	主席密码。
guestPwd	string	来宾密码。
audiencePwd	string	观众密码。(仅对win平台生效)
startTime	number	会议开始时间(utc时间), 单位秒。
endTime	number	会议结束时间(utc时间), 单位秒。
schedulerName	string	会议预定者名称。
vmrConfId	string	用于云会议显示的ID。
hostJoinUri	string	主持人加入会议uri链接
guestJoinUri	string	来宾加入会议uri链接
audienceJoinUri	string	观众加入会议的链接(仅对win平台生效)
confStateType	ConfState	会议状态(仅对win平台生效)
isWebinar	boolean	是否是网络研讨会(仅对win平台生效)
conferenceType	ConferenceType	会议类型

表 9-135 ConfState 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_STATE_SCHEDULE	0	预定状态。
HWM_CONF_STATE_CREATING	1	正在创建状态。
HWM_CONF_STATE_GOING	2	会议已经开始。

枚举名称	枚举值	枚举说明
HWM_CONF_STATE_DESTROYED	3	会议已经关闭。

表 9-136 ConferenceType 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_TYPE_COMMON	0	普通会议
HWM_CONF_TYPE_CYCLE	1	周期会议

表 9-137 CycleConfParam

参数名称	参数类型	参数说明
startDate	number	开始日期-时间戳, 精度秒(0时区)
endDate	number	结束日期-时间戳, 精度秒(0时区)
cycleType	CycleType	周期类型
interval	number	周期区间 1、周期类型选择了按天, 表示每几天召开一次, 取值范围[1,15]; 2、周期类型选择了Week, 表示每几周召开一次, 取值范围[1,5]; 3、周期类型选择了Month, interval表示隔几月, 取值范围[1,3]
listPoints	string	周期内的会议召开点。仅当按周和月时有效。数据格式(中间用逗号隔开): 1,3,5,7
preRemindDays	number	周期子会议提前通知天数。

表 9-138 CycleType 枚举说明

枚举名称	枚举值	枚举说明
CYCLE_TYPE_DAY	0	以天为周期
CYCLE_TYPE_WEEK	1	以周为周期
CYCLE_TYPE_MONTH	2	以月为周期

表 9-139 SubCycleConfParam

参数名称	参数类型	参数说明
subConfID	string	子会议唯一标识UUID
startTime	number	会议起始时间
endTime	number	会议结束时间
isAutoRecord	boolean	会议开始后是否自动开启录制
recordAuthType	RecordAuthType	录播鉴权方式

表 9-140 RecordAuthType 枚举说明

枚举名称	枚举值	枚举说明
REOCDR_AUTH_TYPE_ANYONE	0	可通过连接观看/下载（不拼接nonce）
REOCDR_AUTH_TYPE_IN_COMPANY_USER	1	企业用户可观看/下载
REOCDR_AUTH_TYPE_ATTENDEE	2	与会者可观看/下载

表 9-141 AttendeeBaseInfo

参数名称	参数类型	参数说明
nickName	string	与会者名称。
number	string	号码。如果填分配给帐号的sip号码（如+99111244216210249）则呼叫该帐号的App；如果填PSTN号码（如18700000000），则通过VoIP网关呼叫该号码，前提是该企业已开通PSTN呼叫权限。与thirdUserId二选一，帐号密码鉴权时使用。
thirdUserId	string	第三方用户ID。与number二选一，App ID鉴权时使用。
accountId	string	华为云会议帐号
email	string	电子邮箱地址，用于发送邮件通知。
sms	string	短信通知手机号码，用于发送短信通知。
isAutoInvite	boolean	是否自动外邀。

参数名称	参数类型	参数说明
isMute	boolean	是否闭音。

9.5.4.3 取消会议

接口功能

该接口用于取消已预约的会议。注意事项：1、该接口在登录状态下才能调用；2、调用该接口时，仅能取消尚未开始的会议，若会议已召开，则接口返回失败。

函数原型

cancelConf(cancelConfParam: CancelConfParam, onCancelConfResult?: (ret: SDKERR, reason: string) => void): void

请求参数

表 9-142 入参说明

参数名称	是否必选	参数类型	参数说明
cancelConfParam	Y	CancelConfParam	取消会议的参数体
onCancelConfResult	Y	(ret: SDKERR, reason: string) => void	取消会议预约回调

表 9-143 CancelConfParam

参数名称	是否必选	参数类型	参数说明
confId	Y	string	会议ID。
isForceCancel	N	boolean	是否强制取消，会议开始后 可以强制取消(仅对win平台生效)

返回值

void

回调方法参数说明：

(ret: SDKERR, reason: string) => void

表 9-144 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.4.4 编辑会议

接口功能

该接口用于编辑已预约的会议。注意事项：1、该接口在已登录状态下才能调用；2、调用该接口时，仅能编辑尚未开始的会议，若会议已召开，接口将返回失败；3、所有字段必填，编辑会议前需要查询会议详情，不需要变更的数据请使用查询会议详情返回的数据赋值。

函数原型

```
editConf(editConfParam: EditConfParam, onEditConfResult?: (ret: SDKERR, reason: string) => void): void
```

请求参数

表 9-145 入参说明

参数名称	是否必选	参数类型	参数说明
editConfParam	Y	EditConfParam	编辑会议的参数体
onEditConfResult	Y	(ret: SDKERR, reason: string) => void	修改会议预约回调

表 9-146 EditConfParam

参数名称	是否必选	参数类型	参数说明
confId	Y	string	会议ID。
startTime	Y	number	会议开始时间，如果没有指定开始时间，则表示会议马上开始，精度秒(0时区)
duration	Y	number	会议持续时长，单位分钟
confCommonParam	Y	ConfCommonParam	会议公用参数

参数名称	是否必选	参数类型	参数说明
isSendCalendar	N	boolean	是否发送日历通知

表 9-147 ConfCommonParam

参数名称	是否必选	参数类型	参数说明
subject	Y	string	会议主题。
mediaType	N	MediaType	会议类型。默认音频会议。
needPassword	N	boolean	是否需要来宾密码。默认不需要来宾密码（仅对随机会议ID生效）。
isAutoRecord	N	boolean	会议自动录制开关。默认不开启。开启会议自动录制开关，内部会默认支持录制会议，录制会议开关的打开与关闭并不影响。说明：仅对云录制生效，客户端本地录制不生效。
allowRecord	N	boolean	是否支持录制会议。(仅对win平台生效) 说明：仅对云录制生效，客户端本地录制不生效。
allowLive	N	boolean	保留。编辑会议时透传会议详情此字段
timezone	Y	Timezone	时区码。用于发送会议通知。
vmrId	N	string	云会议室唯一ID。预约并发起会议时，设置为空字符串。编辑会议时不可修改，填入会议详情返回值。
confAllowJoinUser	N	ConfAllowJoinUserType	允许入会范围限制。默认所有用户。
isSendSms	N	boolean	发送短信通知开关。企业配置里面需要开启，否则不生效。
isSendEmail	N	boolean	发送邮件通知开关。
vmrConfIdType	N	VmrConfIdType	云会议室的会议ID类型。编辑会议时不可修改会议详情中vmrConfId非空，填入HWM_VMR_CONF_ID_TYPE_FIXED；会议详情中vmrConfId为空，填入HWM_VMR_CONF_ID_TYPE_RANDOM

参数名称	是否必选	参数类型	参数说明
guestPwd	N	string	来宾密码，如果空则服务器随机生成（仅对随机会议ID生效）。
isOpenWaitingRoom	N	boolean	等候室开启开关。说明：需要等候室功能开通后才能生效
numOfAttendee	Y	number	与会者列表长度。
attendees	Y	AttendeeBaseInfo	与会者列表。
allowGuestStartConf	N	boolean	允许来宾启动会议
allowGuestStartConfTime	N	number	允许来宾提前入会时间范围(单位：分钟) 0-随时，n-提前n分钟启动会议

表 9-148 MediaType 枚举说明

枚举名称	枚举值	枚举说明
HWM_MEDIA_TYPE_AUDIO	0	音频类型
HWM_MEDIA_TYPE_VIDEO	1	视频类型

表 9-149 Timezone 枚举说明

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_DEFAULT	0	Mediax默认时区值
HWM_TIMEZONE_INTERDATE_LINE	1	日界线西
HWM_TIMEZONE_MIDWAY_ISLAND	2	中途岛，萨摩亚群岛
HWM_TIMEZONE_HAWAII	3	夏威夷
HWM_TIMEZONE_ALASKA	4	阿拉斯加
HWM_TIMEZONE_PACIFIC_TIME	5	太平洋时间(美国和加拿大);蒂华纳
HWM_TIMEZONE_ARIZONA	6	亚利桑那

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_MOUNTAIN_TIME	7	山地时间(美国和加拿大)
HWM_TIMEZONE_CENTRAL_AMERICA	8	中美洲
HWM_TIMEZONE_CENTRAL_TIME	9	中间时间(美国和加拿大)
HWM_TIMEZONE_MEXICO_CITY	10	墨西哥城
HWM_TIMEZONE_SASKATCHEWAN	11	萨斯喀彻温
HWM_TIMEZONE_BOGOTA	12	波哥大, 利马, 基多
HWM_TIMEZONE_EASTERN_TIME	13	东部时间(美国和加拿大)
HWM_TIMEZONE_INDIANA	14	印第安纳(东部)
HWM_TIMEZONE_ATLANTIC_TIME	15	大西洋时间(加拿大)
HWM_TIMEZONE_CARACAS	16	加拉加斯, 拉巴斯
HWM_TIMEZONE_SANTIAGO	17	圣地亚哥
HWM_TIMEZONE_NEWFOUNDLAND	18	纽芬兰
HWM_TIMEZONE_BRASILIA	19	巴西利亚
HWM_TIMEZONE_BUENOS_AIRES	20	布宜诺斯艾利斯, 乔治敦
HWM_TIMEZONE_GREENLAND	21	格陵兰
HWM_TIMEZONE_MID_ATLANTIC	22	中大西洋
HWM_TIMEZONE_AZORES	23	亚速尔群岛
HWM_TIMEZONE_CAPEVERDE	24	佛得角群岛
HWM_TIMEZONE_MONROVIA	25	蒙罗维亚
HWM_TIMEZONE_GREENWICH_MEANTIME	26	格林威治标准时: 都柏林, 爱丁堡, 里斯本, 伦敦
HWM_TIMEZONE_AMSTERDAM	27	阿姆斯特丹, 柏林, 伯尔尼, 罗马, 斯德哥尔摩, 维也纳

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_BELGRADE	28	贝尔格莱德, 布拉迪斯拉发, 布达佩斯, 卢布尔亚纳, 布拉格
HWM_TIMEZONE_BRUSSELS	29	布鲁塞尔, 哥本哈根, 马德里, 巴黎
HWM_TIMEZONE_SARAJEVO	30	萨拉热窝, 斯科普里, 维尔纽斯, 索非亚, 华沙, 萨格勒布
HWM_TIMEZONE_WESTCENTRAL_AFRICA	31	中非西部
HWM_TIMEZONE_ATHENS	32	雅典, 伊斯坦布尔, 明斯克
HWM_TIMEZONE_BUCHAREST	33	布加勒斯特
HWM_TIMEZONE_CAIRO	34	开罗
HWM_TIMEZONE_HARARE	35	哈拉雷, 比勒陀利亚
HWM_TIMEZONE_HELSINKI	36	赫尔辛基, 里加, 塔林
HWM_TIMEZONE_JERUSALEM	37	耶路撒冷
HWM_TIMEZONE_BAGHDAD	38	巴格达
HWM_TIMEZONE_KUWAIT	39	科威特, 利雅得
HWM_TIMEZONE_MOSCOW	40	莫斯科, 圣彼得堡, 喀山, 伏尔加格勒
HWM_TIMEZONE_NAIROBI	41	内罗毕
HWM_TIMEZONE_TEHRAN	42	德黑兰
HWM_TIMEZONE_ABU_DHABI	43	阿布扎比, 马斯喀特
HWM_TIMEZONE_TBILISI	44	第比利斯, 埃里温
HWM_TIMEZONE_KABUL	45	喀布尔
HWM_TIMEZONE_EKATERINBURG	46	叶卡捷琳堡
HWM_TIMEZONE_ISLAMABAD	47	伊斯兰堡, 卡拉奇, 塔什干
HWM_TIMEZONE_CALCUTTA	48	加尔各答, 马德拉斯, 孟买, 新德里
HWM_TIMEZONE_KATHMANDU	49	加德满都
HWM_TIMEZONE_ALMATY	50	阿拉木图
HWM_TIMEZONE_ASTANA	51	阿斯塔纳, 达卡
HWM_TIMEZONE_SRIJAYA	52	斯里哈亚华登尼普拉

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_RANGOON	53	仰光
HWM_TIMEZONE_BANGKOK	54	曼谷, 雅加达, 河内
HWM_TIMEZONE_NOVOSIBIRSK	55	新西伯利亚
HWM_TIMEZONE_BEIJING	56	北京, 重庆, 香港, 乌鲁木齐, 台北
HWM_TIMEZONE_KRASNOYARSK	57	克拉斯诺亚尔斯克, 乌兰巴托
HWM_TIMEZONE_KUALALUMPUR	58	吉隆坡, 新加坡
HWM_TIMEZONE_PERTH	59	珀斯
HWM_TIMEZONE_OSAKA	60	东京, 大阪, 札幌
HWM_TIMEZONE_SEOUL	61	汉城
HWM_TIMEZONE_YAKUTSK	62	雅库茨克
HWM_TIMEZONE_ADELAIDE	63	阿德莱德
HWM_TIMEZONE_DARWIN	64	达尔文
HWM_TIMEZONE_BRISBANE	65	布里斯班
HWM_TIMEZONE_CANBERRA	66	堪培拉, 墨尔本, 悉尼
HWM_TIMEZONE_GUAM	67	关岛, 莫尔兹比港
HWM_TIMEZONE_HOBART	68	霍巴特
HWM_TIMEZONE_VLADIVOSTOK	69	符拉迪沃斯托克
HWM_TIMEZONE_SOLOMON	70	所罗门群岛, 新喀里多尼亚
HWM_TIMEZONE_AUCKLAND	71	奥克兰, 惠灵顿
HWM_TIMEZONE_FIJI	72	富士, 堪察加半岛, 马绍尔群岛
HWM_TIMEZONE_NUKUALOFA	73	努库阿洛法
HWM_TIMEZONE_IRKUTSK	74	伊尔库茨克
HWM_TIMEZONE_CASABLANCA	75	卡萨布兰卡
HWM_TIMEZONE_BAKU	76	巴库
HWM_TIMEZONE_MAGADAN	77	马加丹

表 9-150 ConfAllowJoinUserType 枚举说明

枚举名称	枚举值	枚举说明
CONF_ALLOW_JOIN_ANYONE	0	所有用户。
CONF_ALLOW_JOIN_LOGINED_USER	1	已登录用户。
CONF_ALLOW_JOIN_IN_COMPANY_USER	2	企业内用户。
CONF_ALLOW_JOIN_INVITED_USER	3	被邀请用户。

表 9-151 VmrConfIdType 枚举说明

枚举名称	枚举值	枚举说明
HWM_VMR_CONF_ID_TYPE_FIXED	0	云会议室的会议ID类型是固定类型。
HWM_VMR_CONF_ID_TYPE_RANDOM	1	云会议室的会议ID类型是随机类型。

表 9-152 AttendeeBaseInfo

参数名称	是否必选	参数类型	参数说明
nickName	Y	string	与会者名称。
number	Y	string	号码。如果填分配给帐号的sip号码（如+99111244216210249）则呼叫该帐号的App；如果填PSTN号码（如18700000000），则通过VoIP网关呼叫该号码，前提是该企业已开通PSTN呼叫权限。与thirdUserId二选一，帐号密码鉴权时使用。
thirdUserId	Y	string	第三方用户ID。与number二选一，App ID鉴权时使用。
accountId	N	string	华为云会议帐号
email	N	string	电子邮箱地址，用于发送邮件通知。
sms	N	string	短信通知手机号码，用于发送短信通知。

参数名称	是否必选	参数类型	参数说明
isAutoInvite	N	boolean	是否自动外邀。
isMute	N	boolean	是否闭音。
role	N	ConfRole	会议角色。
type	N	AttendeeType	成员类型

表 9-153 ConfRole 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_ROLE_ATTENDEE	0	普通与会者
HWM_CONF_ROLE_HOST	1	主席
HWM_CONF_ROLE_AUDIENCE	2	观众
HWM_CONF_ROLE_COHOST	3	联席主持人
HWM_CONF_ROLE_WAITING	4	等候室成员

表 9-154 AttendeeType 枚举说明

枚举名称	枚举值	枚举说明
ATTENDEE_TYPE_NORMAL	0	语音、高清、标清与会者地址
ATTENDEE_TYPE_TELEPRESENCE	1	智真与会者地址类型，单屏、三屏智真均属此类
ATTENDEE_TYPE_TERMINAL	2	会议室或硬终端
ATTENDEE_TYPE_OUTSIDE	3	外部与会人
ATTENDEE_TYPE_CUSTOMNUMBER	4	custom number
ATTENDEE_TYPE_MOBILE	5	软终端用户手机
ATTENDEE_TYPE_ANONYMOUS	6	匿名用户
ATTENDEE_TYPE_TELEPHONE	7	电话用户
ATTENDEE_TYPE_BOARD	8	白板用户
ATTENDEE_TYPE_IDEAHUB	9	海思芯片的大屏设备

枚举名称	枚举值	枚举说明
ATTENDEE_TYPE_HWVISION	10	智慧屏
ATTENDEE_TYPE_WELINKC	11	welink大众版

返回值

void

回调方法参数说明:

(ret: SDKERR, reason: string) => void

表 9-155 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.4.5 获取 VMR 列表

接口功能

该接口用于获取VMR列表。注意事项：1、该接口在已登录状态下才能调用。

函数原型

getVmrList(onGetVmrListResult?: (ret: SDKERR, reason: string, vmrListInfo: VmrListInfo) => void): void

请求参数

表 9-156 入参说明

参数名称	是否必选	参数类型	参数说明
onGetVmrListResult	Y	(ret: SDKERR, reason: string, vmrListInfo: VmrListInfo) => void	Vmr列表通知，获取Vmr列表请求返回

返回值

void

回调方法参数说明:

(ret: SDKERR, reason: string, vmrListInfo: VmrListInfo) => void

表 9-157 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述
vmrListInfo	VmrListInfo	Vmr列表信息

表 9-158 VmrListInfo

参数名称	参数类型	参数说明
numOfVmrs	number	云会议室个数
vmrs	VmrInfo	云会议室信息

表 9-159 VmrInfo

参数名称	参数类型	参数说明
vmrConfId	string	云会议室上的固定会议ID。
vmrId	string	云会议室唯一ID。
hostPwd	string	云会议室主席密码。
guestPwd	string	云会议室来宾密码。
vmrType	VmrIdType	云会议室类型。
maxParties	number	云会议室最大方数。
isEnabled	boolean	云会议室是否可用
name	string	会议室名称。
allowGuestStartConf	boolean	允许来宾启动会议
confAllowJoinUser	ConfAllowJoinUserType	允许入会范围限制。默认所有用户。

参数名称	参数类型	参数说明
isOpenWaitingRoom	boolean	等候室开启开关。说明：需要等候室功能开通后才能生效

表 9-160 VmridType 枚举说明

枚举名称	枚举值	枚举说明
RANDOM_MEETING_ID	0	随机会议id
PERSONAL_MEETING_ID	1	个人会议id
CLOUD_MEETING_ROOM_ID	2	云会议室id

表 9-161 ConfAllowJoinUserType 枚举说明

枚举名称	枚举值	枚举说明
CONF_ALLOW_JOIN_ANYONE	0	所有用户。
CONF_ALLOW_JOIN_LOGINED_USER	1	已登录用户。
CONF_ALLOW_JOIN_IN_COMPANY_USER	2	企业内用户。
CONF_ALLOW_JOIN_INVITED_USER	3	被邀请用户。

9.5.4.6 获取会议详情

接口功能

该接口用于获取会议的详细信息。注意事项：1、该接口在已登录状态下才能调用。

函数原型

```
getConfDetail(getConfDetailParam: GetConfDetailParam, onGetConfDetailResult?: (ret: SDKERR, reason: string, confDetail: ConfDetail) => void): void
```

请求参数

表 9-162 入参说明

参数名称	是否必选	参数类型	参数说明
getConfigDetailParam	Y	GetConfDetailParam	获取会议详情的参数体
onGetConfDetailResult	Y	(ret: SDKERR, reason: string, confDetail: ConfDetail) => void	会议详情通知，获取会议详情返回

表 9-163 GetConfDetailParam

参数名称	是否必选	参数类型	参数说明
confId	Y	string	会议ID。

返回值

void

回调方法参数说明：

(ret: SDKERR, reason: string, confDetail: ConfDetail) => void

表 9-164 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述
confDetail	ConfDetail	会议详情

表 9-165 ConfDetail

参数名称	参数类型	参数说明
confListInfo	ConfListItem	会议基本信息
vmrId	string	云会议室唯一ID。
vmrFlag	boolean	是否是vmr会议。

参数名称	参数类型	参数说明
isAutoRecord	boolean	是否开启自动录制会议。
allowRecord	boolean	是否支持录制会议
allowLive	boolean	保留。编辑会议时透传会议详情此字段
needPassword	boolean	是否需要密码。
isSendSms	boolean	是否发送短信通知。
isSendEmail	boolean	是否发送邮件通知。
isSendCalendar	boolean	是否发送日历通知。
confAllowJoinUser	ConfAllowJoinUserType	会议准入设置。
numOfAttendee	number	与会者列表长度。
attendees	AttendeeBaseInfo	与会者列表。
isOpenWaitingRoom	boolean	是否开启等候室
allowGuestStartConf	boolean	允许来宾启动会议
allowGuestStartConfTime	number	允许来宾提前入会时间范围(单位: 分钟) 0-随时, n-提前n分钟启动会议
vmrConfIdType	VmrConfIdType	云会议室的会议ID类型

表 9-166 ConfListItem

参数名称	参数类型	参数说明
confId	string	会议ID。
subject	string	会议主题。
hostPwd	string	主席密码。
guestPwd	string	来宾密码。
audiencePwd	string	观众密码。(仅对win平台生效)
startTime	number	会议开始时间(utc时间), 单位秒。
endTime	number	会议结束时间(utc时间), 单位秒。
timezone	Timezone	时区码(仅对win平台生效)
mediaType	MediaType	媒体类型
schedulerName	string	会议预定者名称。

参数名称	参数类型	参数说明
vmrConfId	string	用于云会议显示的ID。
hostJoinUri	string	主持人加入会议uri链接
guestJoinUri	string	来宾加入会议uri链接
audienceJoinUri	string	观众加入会议的链接(仅对win平台生效)
confStateType	ConfState	会议状态(仅对win平台生效)
isWebinar	boolean	是否是网络研讨会(仅对win平台生效)
selfConfRole	ConfRole	本人在会议中的角色
conferenceType	ConferenceType	会议类型
cycleParam	CycleConfParam	周期会议的参数
subConfSize	number	周期子会议数量
subConfParam	SubCycleConfParam	周期子会议参数

表 9-167 Timezone 枚举说明

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_DEFAULT	0	Mediav默认时区值
HWM_TIMEZONE_INTERDATE_LINE	1	日界线西
HWM_TIMEZONE_MIDWAY_ISLAND	2	中途岛, 萨摩亚群岛
HWM_TIMEZONE_HAWAII	3	夏威夷
HWM_TIMEZONE_ALASKA	4	阿拉斯加
HWM_TIMEZONE_PACIFIC_TIME	5	太平洋时间(美国和加拿大);蒂华纳
HWM_TIMEZONE_ARIZONA	6	亚利桑那
HWM_TIMEZONE_MOUNTAIN_TIME	7	山地时间(美国和加拿大)
HWM_TIMEZONE_CENTRAL_AMERICA	8	中美洲
HWM_TIMEZONE_CENTRAL_TIME	9	中间时间(美国和加拿大)

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_MEXICO_CITY	10	墨西哥城
HWM_TIMEZONE_SASKATCHEWAN	11	萨斯喀彻温
HWM_TIMEZONE_BOGOTA	12	波哥大, 利马, 基多
HWM_TIMEZONE_EASTERN_TIME	13	东部时间(美国和加拿大)
HWM_TIMEZONE_INDIANA	14	印第安纳(东部)
HWM_TIMEZONE_ATLANTIC_TIME	15	大西洋时间(加拿大)
HWM_TIMEZONE_CARACAS	16	加拉加斯, 拉巴斯
HWM_TIMEZONE_SANTIAGO	17	圣地亚哥
HWM_TIMEZONE_NEWFOUNDLAND	18	纽芬兰
HWM_TIMEZONE_BRASILIA	19	巴西利亚
HWM_TIMEZONE_BUENOS_AIRES	20	布宜诺斯艾利斯, 乔治敦
HWM_TIMEZONE_GREENLAND	21	格陵兰
HWM_TIMEZONE_MID_ATLANTIC	22	中大西洋
HWM_TIMEZONE_AZORES	23	亚速尔群岛
HWM_TIMEZONE_CAPEVERDE	24	佛得角群岛
HWM_TIMEZONE_MONROVIA	25	蒙罗维亚
HWM_TIMEZONE_GREENWICH_MEANTIME	26	格林威治标准时: 都柏林, 爱丁堡, 里斯本, 伦敦
HWM_TIMEZONE_AMSTERDAM	27	阿姆斯特丹, 柏林, 伯尔尼, 罗马, 斯德哥尔摩, 维也纳
HWM_TIMEZONE_BELGRADE	28	贝尔格莱德, 布拉迪斯拉发, 布达佩斯, 卢布尔亚纳, 布拉格
HWM_TIMEZONE_BRUSSELS	29	布鲁塞尔, 哥本哈根, 马德里, 巴黎
HWM_TIMEZONE_SARAJEVO	30	萨拉热窝, 斯科普里, 维尔纽斯, 索非亚, 华沙, 萨格勒布
HWM_TIMEZONE_WESTCENTRAL_AFRICA	31	中非西部

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_ATHENS	32	雅典, 伊斯坦布尔, 明斯克
HWM_TIMEZONE_BUCHAREST	33	布加勒斯特
HWM_TIMEZONE_CAIRO	34	开罗
HWM_TIMEZONE_HARARE	35	哈拉雷, 比勒陀利亚
HWM_TIMEZONE_HELSINKI	36	赫尔辛基, 里加, 塔林
HWM_TIMEZONE_JERUSALEM	37	耶路撒冷
HWM_TIMEZONE_BAGHDAD	38	巴格达
HWM_TIMEZONE_KUWAIT	39	科威特, 利雅得
HWM_TIMEZONE_MOSCOW	40	莫斯科, 圣彼得堡, 喀山, 伏尔加格勒
HWM_TIMEZONE_NAIROBI	41	内罗毕
HWM_TIMEZONE_TEHRAN	42	德黑兰
HWM_TIMEZONE_ABU_DHABI	43	阿布扎比, 马斯喀特
HWM_TIMEZONE_TBILISI	44	第比利斯, 埃里温
HWM_TIMEZONE_KABUL	45	喀布尔
HWM_TIMEZONE_EKATERINBURG	46	叶卡捷琳堡
HWM_TIMEZONE_ISLAMABAD	47	伊斯兰堡, 卡拉奇, 塔什干
HWM_TIMEZONE_CALCUTTA	48	加尔各答, 马德拉斯, 孟买, 新德里
HWM_TIMEZONE_KATHMANDU	49	加德满都
HWM_TIMEZONE_ALMATY	50	阿拉木图
HWM_TIMEZONE_ASTANA	51	阿斯塔纳, 达卡
HWM_TIMEZONE_SRIJAYA	52	斯里哈亚华登尼普拉
HWM_TIMEZONE_RANGOON	53	仰光
HWM_TIMEZONE_BANGKOK	54	曼谷, 雅加达, 河内
HWM_TIMEZONE_NOVOSIBIRSK	55	新西伯利亚
HWM_TIMEZONE_BEIJING	56	北京, 重庆, 香港, 乌鲁木齐, 台北
HWM_TIMEZONE_KRASNOYARSK	57	克拉斯诺亚尔斯克, 乌兰巴托

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_KUALA_LUMPUR	58	吉隆坡，新加坡
HWM_TIMEZONE_PERTH	59	珀斯
HWM_TIMEZONE_OSAKA	60	东京，大阪，札幌
HWM_TIMEZONE_SEOUL	61	汉城
HWM_TIMEZONE_YAKUTSK	62	雅库茨克
HWM_TIMEZONE_ADELAIDE	63	阿德莱德
HWM_TIMEZONE_DARWIN	64	达尔文
HWM_TIMEZONE_BRISBANE	65	布里斯班
HWM_TIMEZONE_CANBERRA	66	堪培拉，墨尔本，悉尼
HWM_TIMEZONE_GUAM	67	关岛，莫尔兹比港
HWM_TIMEZONE_HOBART	68	霍巴特
HWM_TIMEZONE_VLADIVOSTOK	69	符拉迪沃斯托克
HWM_TIMEZONE_SOLOMON	70	所罗门群岛，新喀里多尼亚
HWM_TIMEZONE_AUCKLAND	71	奥克兰，惠灵顿
HWM_TIMEZONE_FIJI	72	富士，堪察加半岛，马绍尔群岛
HWM_TIMEZONE_NUKUALOFUA	73	努库阿洛法
HWM_TIMEZONE_IRKUTSK	74	伊尔库茨克
HWM_TIMEZONE_CASABLANCA	75	卡萨布兰卡
HWM_TIMEZONE_BAKU	76	巴库
HWM_TIMEZONE_MAGADAN	77	马加丹

表 9-168 MediaType 枚举说明

枚举名称	枚举值	枚举说明
HWM_MEDIA_TYPE_AUDIO	0	音频类型
HWM_MEDIA_TYPE_VIDEO	1	视频类型

表 9-169 ConfState 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_STATE_SCHEDULE	0	预定状态。
HWM_CONF_STATE_CREATING	1	正在创建状态。
HWM_CONF_STATE_GOING	2	会议已经开始。
HWM_CONF_STATE_DESTROYED	3	会议已经关闭。

表 9-170 ConfRole 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_ROLE_ATTENDEE	0	普通与会者
HWM_CONF_ROLE_HOST	1	主席
HWM_CONF_ROLE_AUDIENCE	2	观众
HWM_CONF_ROLE_COHOST	3	联席主持人
HWM_CONF_ROLE_WAITING	4	等候室成员

表 9-171 ConferenceType 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_TYPE_COMMON	0	普通会议
HWM_CONF_TYPE_CYCLE	1	周期会议

表 9-172 CycleConfParam

参数名称	参数类型	参数说明
startDate	number	开始日期-时间戳, 精度秒(0时区)
endDate	number	结束日期-时间戳, 精度秒(0时区)
cycleType	CycleType	周期类型

参数名称	参数类型	参数说明
interval	number	周期区间 1、周期类型选择了按天，表示每几天召开一次，取值范围[1,15]；2、周期类型选择了Week，表示每几周召开一次，取值范围[1,5]；3、周期类型选择了Month，interval表示隔几月，取值范围[1,3]
listPoints	string	周期内的会议召开点。仅当按周和月时有效。数据格式(中间用逗号隔开): 1,3,5,7
preRemindDays	number	周期子会议提前通知天数。

表 9-173 CycleType 枚举说明

枚举名称	枚举值	枚举说明
CYCLE_TYPE_DAY	0	以天为周期
CYCLE_TYPE_WEEK	1	以周为周期
CYCLE_TYPE_MONTH	2	以月为周期

表 9-174 SubCycleConfParam

参数名称	参数类型	参数说明
subConfID	string	子会议唯一标识UUID
startTime	number	会议起始时间
endTime	number	会议结束时间
isAutoRecord	boolean	会议开始后是否自动开启录制
recordAuthType	RecordAuthType	录播鉴权方式
confAllowJoinUser	ConfAllowJoinUserType	允许入会范围限制。默认所有用户。
allowGuestStartConf	boolean	允许来宾启动会议

表 9-175 RecordAuthType 枚举说明

枚举名称	枚举值	枚举说明
REOCDR_AUTH_TYPE_ANYONE	0	可通过连接观看/下载（不拼接nonce）
REOCDR_AUTH_TYPE_IN_COMPANY_USER	1	企业用户可观看/下载
REOCDR_AUTH_TYPE_ATTENDEE	2	与会者可观看/下载

表 9-176 ConfAllowJoinUserType 枚举说明

枚举名称	枚举值	枚举说明
CONF_ALLOW_JOIN_ANYONE	0	所有用户。
CONF_ALLOW_JOIN_LOGINED_USER	1	已登录用户。
CONF_ALLOW_JOIN_IN_COMPANY_USER	2	企业内用户。
CONF_ALLOW_JOIN_INVITED_USER	3	被邀请用户。

表 9-177 AttendeeBaseInfo

参数名称	参数类型	参数说明
nickName	string	与会者名称。
number	string	号码。如果填分配给帐号的sip号码（如+99111244216210249）则呼叫该帐号的App；如果填PSTN号码（如18700000000），则通过VoIP网关呼叫该号码，前提是该企业已开通PSTN呼叫权限。与thirdUserId二选一，帐号密码鉴权时使用。
thirdUserId	string	第三方用户ID。与number二选一，App ID鉴权时使用。
accountId	string	华为云会议帐号
email	string	电子邮箱地址，用于发送邮件通知。
sms	string	短信通知手机号码，用于发送短信通知。

参数名称	参数类型	参数说明
isAutoInvite	boolean	是否自动外邀。
isMute	boolean	是否闭音。

表 9-178 AttendeeType 枚举说明

枚举名称	枚举值	枚举说明
ATTENDEE_TYPE_NORMAL	0	语音、高清、标清与会者地址
ATTENDEE_TYPE_TELEPRESENCE	1	智真与会者地址类型，单屏、三屏智真均属此类
ATTENDEE_TYPE_TERMINAL	2	会议室或硬终端
ATTENDEE_TYPE_OUTSIDE	3	外部与会人
ATTENDEE_TYPE_CUSTOMNUMBER	4	custom number
ATTENDEE_TYPE_MOBILE	5	软终端用户手机
ATTENDEE_TYPE_ANONYMOUS	6	匿名用户
ATTENDEE_TYPE_TELEPHONE	7	电话用户
ATTENDEE_TYPE_BOARD	8	白板用户
ATTENDEE_TYPE_IDEAHUB	9	海思芯片的大屏设备
ATTENDEE_TYPE_HWVISION	10	智慧屏
ATTENDEE_TYPE_WELINKC	11	welink大众版

表 9-179 VmrConfIdType 枚举说明

枚举名称	枚举值	枚举说明
HWM_VMR_CONF_ID_TYPE_FIXED	0	云会议室的会议ID类型是固定类型。
HWM_VMR_CONF_ID_TYPE_RANDOM	1	云会议室的会议ID类型是随机类型。

9.5.4.7 修改云会议室信息

接口功能

该接口用于修改云会议室信息。注意事项：1、该接口在已登录状态下才能调用；2、所有字段必填，调用该接口前需要获取VMR列表，不需要变更的数据请使用获取VMR列表接口返回的数据赋值。

函数原型

```
changeVmrInfo(vmrInfoParam: ChangeVmrInfoParam, onChangeVmrInfoResult?: (ret: SDKERR, reason: string) => void): void
```

请求参数

表 9-180 入参说明

参数名称	是否必选	参数类型	参数说明
vmrInfoParam	Y	ChangeVmrInfoParam	虚拟会议室的修改信息
onChangeVmrInfoResult	Y	(ret: SDKERR, reason: string) => void	修改虚拟会议室信息回调

表 9-181 ChangeVmrInfoParam

参数名称	是否必选	参数类型	参数说明
vmrId	Y	string	云会议室唯一ID。
vmrName	Y	string	云会议室名称。
guestPwd	Y	string	云会议室来宾密码。
hostPwd	Y	string	云会议室主持人密码。
allowGuestStartConf	N	boolean	是否允许来宾提前启动会议
confAllowJoinUser	N	ConfAllowJoinUserType	允许入会范围限制。默认所有用户。
isOpenWaitingRoom	N	boolean	等候室开启开关。说明：需要等候室功能开通后才能生效

表 9-182 ConfAllowJoinUserType 枚举说明

枚举名称	枚举值	枚举说明
CONF_ALLOW_JOIN_ANYONE	0	所有用户。
CONF_ALLOW_JOIN_LOGINED_USER	1	已登录用户。
CONF_ALLOW_JOIN_IN_COMPANY_USER	2	企业内用户。
CONF_ALLOW_JOIN_INVITED_USER	3	被邀请用户。

返回值

void

回调方法参数说明:

(ret: SDKERR, reason: string) => void

表 9-183 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.4.8 获取会议列表

接口功能

该接口用于获取会议列表(仅对mac平台生效)。注意事项：1、已登录状态，首次收到会议列表通知之后，调用才有效，否则返回为空的列表

函数原型

getConfList(onGetConfListResult?: (ret: SDKERR, reason: string, confListInfo: ConfListInfo) => void): void

请求参数

表 9-184 入参说明

参数名称	是否必选	参数类型	参数说明
onGetConfListResult	Y	(ret: SDKERR, reason: string, confListInfo: ConfListInfo) => void	会议列表通知，获取会议列表请求返回

返回值

void

回调方法参数说明：

(ret: SDKERR, reason: string, confListInfo: ConfListInfo) => void

表 9-185 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述
confListInfo	ConfListInfo	会议列表信息

表 9-186 ConfListInfo

参数名称	参数类型	参数说明
confListSize	number	会议列表参数信息的长度。
confListItem	ConfListItem	会议列表参数信息。

表 9-187 ConfListItem

参数名称	参数类型	参数说明
confId	string	会议ID。
subject	string	会议主题。
hostPwd	string	主席密码。

参数名称	参数类型	参数说明
guestPwd	string	来宾密码。
audiencePwd	string	观众密码。(仅对win平台生效)
startTime	number	会议开始时间(utc时间), 单位秒。
endTime	number	会议结束时间(utc时间), 单位秒。
timezone	Timezone	时区码(仅对win平台生效)
mediaType	MediaType	媒体类型
schedulerName	string	会议预定者名称。
vmrConfId	string	用于云会议显示的ID。
hostJoinUri	string	主持人加入会议uri链接
guestJoinUri	string	来宾加入会议uri链接
audienceJoinUri	string	观众加入会议的链接(仅对win平台生效)
confStateType	ConfState	会议状态(仅对win平台生效)
isWebinar	boolean	是否是网络研讨会(仅对win平台生效)
selfConfRole	ConfRole	本人在会议中的角色
conferenceType	ConferenceType	会议类型
cycleParam	CycleConfParam	周期会议的参数
subConfSize	number	周期子会议数量
subConfParam	SubCycleConfParam	周期子会议参数

表 9-188 Timezone 枚举说明

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_DEFAULT	0	Mediav默认时区值
HWM_TIMEZONE_INTERDATE_LINE	1	日界线西
HWM_TIMEZONE_MIDWAY_ISLAND	2	中途岛, 萨摩亚群岛
HWM_TIMEZONE_HAWAII	3	夏威夷
HWM_TIMEZONE_ALASKA	4	阿拉斯加
HWM_TIMEZONE_PACIFIC_TIME	5	太平洋时间(美国和加拿大);蒂华纳

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_ARIZONA	6	亚利桑那
HWM_TIMEZONE_MOUNTAIN_TIME	7	山地时间(美国和加拿大)
HWM_TIMEZONE_CENTRAL_AMERICA	8	中美洲
HWM_TIMEZONE_CENTRAL_TIME	9	中间时间(美国和加拿大)
HWM_TIMEZONE_MEXICO_CITY	10	墨西哥城
HWM_TIMEZONE_SASKATCHEWAN	11	萨斯喀彻温
HWM_TIMEZONE_BOGOTA	12	波哥大, 利马, 基多
HWM_TIMEZONE_EASTERN_TIME	13	东部时间(美国和加拿大)
HWM_TIMEZONE_INDIANA	14	印第安纳(东部)
HWM_TIMEZONE_ATLANTIC_TIME	15	大西洋时间(加拿大)
HWM_TIMEZONE_CARACAS	16	加拉加斯, 拉巴斯
HWM_TIMEZONE_SANTIAGO	17	圣地亚哥
HWM_TIMEZONE_NEWFOUNDLAND	18	纽芬兰
HWM_TIMEZONE_BRASILIA	19	巴西利亚
HWM_TIMEZONE_BUENOS_AIRES	20	布宜诺斯艾利斯, 乔治敦
HWM_TIMEZONE_GREENLAND	21	格陵兰
HWM_TIMEZONE_MID_ATLANTIC	22	中大西洋
HWM_TIMEZONE_AZORES	23	亚速尔群岛
HWM_TIMEZONE_CAPEVERDE	24	佛得角群岛
HWM_TIMEZONE_MONROVIA	25	蒙罗维亚
HWM_TIMEZONE_GREENWICH_MEANTIME	26	格林威治标准时: 都柏林, 爱丁堡, 里斯本, 伦敦
HWM_TIMEZONE_AMSTERDAM	27	阿姆斯特丹, 柏林, 伯尔尼, 罗马, 斯德哥尔摩, 维也纳

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_BELGRADE	28	贝尔格莱德, 布拉迪斯拉发, 布达佩斯, 卢布尔亚纳, 布拉格
HWM_TIMEZONE_BRUSSELS	29	布鲁塞尔, 哥本哈根, 马德里, 巴黎
HWM_TIMEZONE_SARAJEVO	30	萨拉热窝, 斯科普里, 维尔纽斯, 索非亚, 华沙, 萨格勒布
HWM_TIMEZONE_WESTCENTRAL_AFRICA	31	中非西部
HWM_TIMEZONE_ATHENS	32	雅典, 伊斯坦布尔, 明斯克
HWM_TIMEZONE_BUCHAREST	33	布加勒斯特
HWM_TIMEZONE_CAIRO	34	开罗
HWM_TIMEZONE_HARARE	35	哈拉雷, 比勒陀利亚
HWM_TIMEZONE_HELSINKI	36	赫尔辛基, 里加, 塔林
HWM_TIMEZONE_JERUSALEM	37	耶路撒冷
HWM_TIMEZONE_BAGHDAD	38	巴格达
HWM_TIMEZONE_KUWAIT	39	科威特, 利雅得
HWM_TIMEZONE_MOSCOW	40	莫斯科, 圣彼得堡, 喀山, 伏尔加格勒
HWM_TIMEZONE_NAIROBI	41	内罗毕
HWM_TIMEZONE_TEHRAN	42	德黑兰
HWM_TIMEZONE_ABU_DHABI	43	阿布扎比, 马斯喀特
HWM_TIMEZONE_TBILISI	44	第比利斯, 埃里温
HWM_TIMEZONE_KABUL	45	喀布尔
HWM_TIMEZONE_EKATERINBURG	46	叶卡捷琳堡
HWM_TIMEZONE_ISLAMABAD	47	伊斯兰堡, 卡拉奇, 塔什干
HWM_TIMEZONE_CALCUTTA	48	加尔各答, 马德拉斯, 孟买, 新德里
HWM_TIMEZONE_KATHMANDU	49	加德满都
HWM_TIMEZONE_ALMATY	50	阿拉木图
HWM_TIMEZONE_ASTANA	51	阿斯塔纳, 达卡
HWM_TIMEZONE_SRIJAYA	52	斯里哈亚华登尼普拉

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_RANGOON	53	仰光
HWM_TIMEZONE_BANGKOK	54	曼谷, 雅加达, 河内
HWM_TIMEZONE_NOVOSIBIRSK	55	新西伯利亚
HWM_TIMEZONE_BEIJING	56	北京, 重庆, 香港, 乌鲁木齐, 台北
HWM_TIMEZONE_KRASNOYARSK	57	克拉斯诺亚尔斯克, 乌兰巴托
HWM_TIMEZONE_KUALALUMPUR	58	吉隆坡, 新加坡
HWM_TIMEZONE_PERTH	59	珀斯
HWM_TIMEZONE_OSAKA	60	东京, 大阪, 札幌
HWM_TIMEZONE_SEOUL	61	汉城
HWM_TIMEZONE_YAKUTSK	62	雅库茨克
HWM_TIMEZONE_ADELAIDE	63	阿德莱德
HWM_TIMEZONE_DARWIN	64	达尔文
HWM_TIMEZONE_BRISBANE	65	布里斯班
HWM_TIMEZONE_CANBERRA	66	堪培拉, 墨尔本, 悉尼
HWM_TIMEZONE_GUAM	67	关岛, 莫尔兹比港
HWM_TIMEZONE_HOBART	68	霍巴特
HWM_TIMEZONE_VLADIVOSTOK	69	符拉迪沃斯托克
HWM_TIMEZONE_SOLOMON	70	所罗门群岛, 新喀里多尼亚
HWM_TIMEZONE_AUCKLAND	71	奥克兰, 惠灵顿
HWM_TIMEZONE_FIJI	72	富士, 堪察加半岛, 马绍尔群岛
HWM_TIMEZONE_NUKUALOFA	73	努库阿洛法
HWM_TIMEZONE_IRKUTSK	74	伊尔库茨克
HWM_TIMEZONE_CASABLANCA	75	卡萨布兰卡
HWM_TIMEZONE_BAKU	76	巴库
HWM_TIMEZONE_MAGADAN	77	马加丹

表 9-189 MediaType 枚举说明

枚举名称	枚举值	枚举说明
HWM_MEDIA_TYPE_AUDIO	0	音频类型
HWM_MEDIA_TYPE_VIDEO	1	视频类型

表 9-190 ConfState 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_STATE_SCHEDULE	0	预定状态。
HWM_CONF_STATE_CREATING	1	正在创建状态。
HWM_CONF_STATE_GOING	2	会议已经开始。
HWM_CONF_STATE_DESTROYED	3	会议已经关闭。

表 9-191 ConfRole 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_ROLE_ATTENDEE	0	普通与会者
HWM_CONF_ROLE_HOST	1	主席
HWM_CONF_ROLE_AUDIENCE	2	观众
HWM_CONF_ROLE_COHOST	3	联席主持人
HWM_CONF_ROLE_WAITING	4	等候室成员

表 9-192 ConferenceType 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_TYPE_COMMON	0	普通会议
HWM_CONF_TYPE_CYCLE	1	周期会议

表 9-193 CycleConfParam

参数名称	参数类型	参数说明
startDate	number	开始日期-时间戳, 精度秒(0时区)
endDate	number	结束日期-时间戳, 精度秒(0时区)
cycleType	CycleType	周期类型
interval	number	周期区间 1、周期类型选择了按天, 表示每几天召开一次, 取值范围[1,15]; 2、周期类型选择了Week, 表示每几周召开一次, 取值范围[1,5]; 3、周期类型选择了Month, interval表示隔几月, 取值范围[1,3]
listPoints	string	周期内的会议召开点。仅当按周和月时有效。数据格式(中间用逗号隔开): 1,3,5,7
preRemindDays	number	周期子会议提前通知天数。

表 9-194 CycleType 枚举说明

枚举名称	枚举值	枚举说明
CYCLE_TYPE_DAY	0	以天为周期
CYCLE_TYPE_WEEK	1	以周为周期
CYCLE_TYPE_MONTH	2	以月为周期

表 9-195 SubCycleConfParam

参数名称	参数类型	参数说明
subConfID	string	子会议唯一标识UUID
startTime	number	会议起始时间
endTime	number	会议结束时间
isAutoRecord	boolean	会议开始后是否自动开启录制
recordAuthType	RecordAuthType	录播鉴权方式
confAllowJoinUser	ConfAllowJoinUserType	允许入会范围限制。默认所有用户。
allowGuestStartConf	boolean	允许来宾启动会议

表 9-196 RecordAuthType 枚举说明

枚举名称	枚举值	枚举说明
REOCDR_AUTH_TYPE_ANYONE	0	可通过连接观看/下载（不拼接nonce）
REOCDR_AUTH_TYPE_IN_COMPANY_USER	1	企业用户可观看/下载
REOCDR_AUTH_TYPE_ATTENDEE	2	与会者可观看/下载

表 9-197 ConfAllowJoinUserType 枚举说明

枚举名称	枚举值	枚举说明
CONF_ALLOW_JOIN_ANYONE	0	所有用户。
CONF_ALLOW_JOIN_LOGINED_USER	1	已登录用户。
CONF_ALLOW_JOIN_IN_COMPANY_USER	2	企业内用户。
CONF_ALLOW_JOIN_INVITED_USER	3	被邀请用户。

9.5.4.9 预约周期会议

接口功能

该接口用于预约周期会议。注意事项：1、该接口在已登录状态下才能调用。2、该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。3、会议开始时间必须晚于当前时间，会议时长应大于15分钟。4、携带与会者入会的时候需要传入attendees。5、周期会议最大跨度一年，会议数量不超过50，超过则只预约前50个会议。6、回调函数中会返回预约成功后的会议详情，详见HwmConfDetail结构体表。如果参数内只有会议id有值则获取会议详情失败。

函数原型

```
bookCycleConf(bookCycleConfParam: BookCycleConfParam,
onBookCycleConfResult?: (ret: SDKERR, reason: string, confDetail: ConfDetail) =>
void): void
```

请求参数

表 9-198 入参说明

参数名称	是否必选	参数类型	参数说明
bookCycleConfParam	Y	BookCycleConfParam	预约周期会议的参数体
onBookCycleConfResult	Y	(ret: SDKERR, reason: string, confDetail: ConfDetail) => void	创建周期会议的结果回调

表 9-199 BookCycleConfParam

参数名称	是否必选	参数类型	参数说明
bookConfParam	N	BookConfParam	预约会议参数
cycleParam	Y	CycleConfParam	子会议参数

表 9-200 BookConfParam

参数名称	是否必选	参数类型	参数说明
startTime	Y	number	会议开始时间时间戳(utc时间), 单位是秒。
duration	Y	number	会议时长, 单位分钟, 最小值15分钟, 最大值1425分钟。
confCommonParam	Y	ConfCommonParam	会议公用参数
isSendCalendar	N	boolean	发送邮件日历开关。

表 9-201 ConfCommonParam

参数名称	是否必选	参数类型	参数说明
subject	Y	string	会议主题。
mediaType	N	MediaType	会议类型。默认音频会议。

参数名称	是否必选	参数类型	参数说明
needPassword	N	boolean	是否需要来宾密码。默认不需要来宾密码（仅对随机会议ID生效）。
isAutoRecord	N	boolean	会议自动录制开关。默认不开启。开启会议自动录制开关，内部会默认支持录制会议，录制会议开关的打开与关闭并不影响。说明：仅对云录制生效，客户端本地录制不生效。
allowRecord	N	boolean	是否支持录制会议。（仅对win平台生效）说明：仅对云录制生效，客户端本地录制不生效。
allowLive	N	boolean	保留。编辑会议时透传会议详情此字段
timezone	Y	Timezone	时区码。用于发送会议通知。
vmrId	N	string	云会议室唯一ID。预约并发会议时，设置为空字符串。编辑会议时不可修改，填入会议详情返回值。
confAllowJoinUser	N	ConfAllowJoinUserType	允许入会范围限制。默认所有用户。
isSendSms	N	boolean	发送短信通知开关。企业配置里面需要开启，否则不生效。
isSendEmail	N	boolean	发送邮件通知开关。
vmrConfIdType	N	VmrConfIdType	云会议室的会议ID类型。编辑会议时不可修改会议详情中vmrConfId非空，填入HWM_VMR_CONF_ID_TYPE_FIXED；会议详情中vmrConfId为空，填入HWM_VMR_CONF_ID_TYPE_RANDOM
guestPwd	N	string	来宾密码，如果空则服务器随机生成（仅对随机会议ID生效）。
isOpenWaitingRoom	N	boolean	等候室开启开关。说明：需要等候室功能开通后才能生效
numOfAttendee	Y	number	与会者列表长度。
attendees	Y	AttendeeBaseInfo	与会者列表。

参数名称	是否必选	参数类型	参数说明
allowGuestStartConf	N	boolean	允许来宾启动会议
allowGuestStartConfTime	N	number	允许来宾提前入会时间范围(单位: 分钟) 0-随时, n-提前n分钟启动会议

表 9-202 MediaType 枚举说明

枚举名称	枚举值	枚举说明
HWM_MEDIA_TYPE_AUDIO	0	音频类型
HWM_MEDIA_TYPE_VIDEO	1	视频类型

表 9-203 Timezone 枚举说明

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_DEFAULT	0	Mediax默认时区值
HWM_TIMEZONE_INTERDATE_LINE	1	日界线西
HWM_TIMEZONE_MIDWAY_ISLAND	2	中途岛, 萨摩亚群岛
HWM_TIMEZONE_HAWAII	3	夏威夷
HWM_TIMEZONE_ALASKA	4	阿拉斯加
HWM_TIMEZONE_PACIFIC_TIME	5	太平洋时间(美国和加拿大);蒂华纳
HWM_TIMEZONE_ARIZONA	6	亚利桑那
HWM_TIMEZONE_MOUNTAIN_TIME	7	山地时间(美国和加拿大)
HWM_TIMEZONE_CENTRAL_AMERICA	8	中美洲
HWM_TIMEZONE_CENTRAL_TIME	9	中间时间(美国和加拿大)
HWM_TIMEZONE_MEXICO_CITY	10	墨西哥城

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_SASKATCHEWAN	11	萨斯喀彻温
HWM_TIMEZONE_BOGOTA	12	波哥大, 利马, 基多
HWM_TIMEZONE_EASTERN_TIME	13	东部时间(美国和加拿大)
HWM_TIMEZONE_INDIANA	14	印第安纳(东部)
HWM_TIMEZONE_ATLANTIC_TIME	15	大西洋时间(加拿大)
HWM_TIMEZONE_CARACAS	16	加拉加斯, 拉巴斯
HWM_TIMEZONE_SANTIAGO	17	圣地亚哥
HWM_TIMEZONE_NEWFOUNDLAND	18	纽芬兰
HWM_TIMEZONE_BRASILIA	19	巴西利亚
HWM_TIMEZONE_BUENOSAIRES	20	布宜诺斯艾利斯, 乔治敦
HWM_TIMEZONE_GREENLAND	21	格陵兰
HWM_TIMEZONE_MIDATLANTIC	22	中大西洋
HWM_TIMEZONE_AZORES	23	亚速尔群岛
HWM_TIMEZONE_CAPEVERDE	24	佛得角群岛
HWM_TIMEZONE_MONROVIA	25	蒙罗维亚
HWM_TIMEZONE_GREENWICH_MEANTIME	26	格林威治标准时: 都柏林, 爱丁堡, 里斯本, 伦敦
HWM_TIMEZONE_AMSTERDAM	27	阿姆斯特丹, 柏林, 伯尔尼, 罗马, 斯德哥尔摩, 维也纳
HWM_TIMEZONE_BELGRADE	28	贝尔格莱德, 布拉迪斯拉发, 布达佩斯, 卢布尔亚纳, 布拉格
HWM_TIMEZONE_BRUSSELS	29	布鲁塞尔, 哥本哈根, 马德里, 巴黎
HWM_TIMEZONE_SARAJEVO	30	萨拉热窝, 斯科普里, 维尔纽斯, 索非亚, 华沙, 萨格勒布
HWM_TIMEZONE_WESTCENTRAL_AFRICA	31	中非西部
HWM_TIMEZONE_ATHENS	32	雅典, 伊斯坦布尔, 明斯克

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_BUCHAREST	33	布加勒斯特
HWM_TIMEZONE_CAIRO	34	开罗
HWM_TIMEZONE_HARARE	35	哈拉雷, 比勒陀利亚
HWM_TIMEZONE_HELSINKI	36	赫尔辛基, 里加, 塔林
HWM_TIMEZONE_JERUSALEM	37	耶路撒冷
HWM_TIMEZONE_BAGHDAD	38	巴格达
HWM_TIMEZONE_KUWAIT	39	科威特, 利雅得
HWM_TIMEZONE_MOSCOW	40	莫斯科, 圣彼得堡, 喀山, 伏尔加格勒
HWM_TIMEZONE_NAIROBI	41	内罗毕
HWM_TIMEZONE_TEHRAN	42	德黑兰
HWM_TIMEZONE_ABU_DHABI	43	阿布扎比, 马斯喀特
HWM_TIMEZONE_TBILISI	44	第比利斯, 埃里温
HWM_TIMEZONE_KABUL	45	喀布尔
HWM_TIMEZONE_EKATERINBURG	46	叶卡捷琳堡
HWM_TIMEZONE_ISLAMABAD	47	伊斯兰堡, 卡拉奇, 塔什干
HWM_TIMEZONE_CALCUTTA	48	加尔各答, 马德拉斯, 孟买, 新德里
HWM_TIMEZONE_KATHMANDU	49	加德满都
HWM_TIMEZONE_ALMATY	50	阿拉木图
HWM_TIMEZONE_ASTANA	51	阿斯塔纳, 达卡
HWM_TIMEZONE_SRIJAYA	52	斯里哈亚华登尼普拉
HWM_TIMEZONE_RANGOON	53	仰光
HWM_TIMEZONE_BANGKOK	54	曼谷, 雅加达, 河内
HWM_TIMEZONE_NOVOSIBIRSK	55	新西伯利亚
HWM_TIMEZONE_BEIJING	56	北京, 重庆, 香港, 乌鲁木齐, 台北
HWM_TIMEZONE_KRASNOYARSK	57	克拉斯诺亚尔斯克, 乌兰巴托

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_KUALA_LUMPUR	58	吉隆坡，新加坡
HWM_TIMEZONE_PERTH	59	珀斯
HWM_TIMEZONE_OSAKA	60	东京，大阪，札幌
HWM_TIMEZONE_SEOUL	61	汉城
HWM_TIMEZONE_YAKUTSK	62	雅库茨克
HWM_TIMEZONE_ADELAIDE	63	阿德莱德
HWM_TIMEZONE_DARWIN	64	达尔文
HWM_TIMEZONE_BRISBANE	65	布里斯班
HWM_TIMEZONE_CANBERRA	66	堪培拉，墨尔本，悉尼
HWM_TIMEZONE_GUAM	67	关岛，莫尔兹比港
HWM_TIMEZONE_HOBART	68	霍巴特
HWM_TIMEZONE_VLADIVOSTOK	69	符拉迪沃斯托克
HWM_TIMEZONE_SOLOMON	70	所罗门群岛，新喀里多尼亚
HWM_TIMEZONE_AUCKLAND	71	奥克兰，惠灵顿
HWM_TIMEZONE_FIJI	72	富士，堪察加半岛，马绍尔群岛
HWM_TIMEZONE_NUKUALOFA	73	努库阿洛法
HWM_TIMEZONE_IRKUTSK	74	伊尔库茨克
HWM_TIMEZONE_CASABLANCA	75	卡萨布兰卡
HWM_TIMEZONE_BAKU	76	巴库
HWM_TIMEZONE_MAGADAN	77	马加丹

表 9-204 ConfAllowJoinUserType 枚举说明

枚举名称	枚举值	枚举说明
CONF_ALLOW_JOIN_ANYONE	0	所有用户。
CONF_ALLOW_JOIN_LOGINED_USER	1	已登录用户。

枚举名称	枚举值	枚举说明
CONF_ALLOW_JOIN_IN_COMP ANY_USER	2	企业内用户。
CONF_ALLOW_JOIN_INVITED_ USER	3	被邀请用户。

表 9-205 VmrConfIdType 枚举说明

枚举名称	枚举值	枚举说明
HWM_VMR_CONF_ID_TYPE_FI XED	0	云会议室的会议ID类型是固定类型。
HWM_VMR_CONF_ID_TYPE_R ANDOM	1	云会议室的会议ID类型是随机类型。

表 9-206 AttendeeBaseInfo

参数名称	是否必选	参数类型	参数说明
nickName	Y	string	与会者名称。
number	Y	string	号码。如果填分配给帐号的sip号码（如+99111244216210249）则呼叫该帐号的App；如果填PSTN号码（如18700000000），则通过VoIP网关呼叫该号码，前提是该企业已开通PSTN呼叫权限。与thirdUserId二选一，帐号密码鉴权时使用。
thirdUserId	Y	string	第三方用户ID。与number二选一，App ID鉴权时使用。
accountId	N	string	华为云会议帐号
email	N	string	电子邮箱地址，用于发送邮件通知。
sms	N	string	短信通知手机号码，用于发送短信通知。
isAutoInvite	N	boolean	是否自动外邀。
isMute	N	boolean	是否闭音。
role	N	ConfRole	会议角色。

参数名称	是否必选	参数类型	参数说明
type	N	AttendeeType	成员类型

表 9-207 ConfRole 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_ROLE_ATTENDEE	0	普通与会者
HWM_CONF_ROLE_HOST	1	主席
HWM_CONF_ROLE_AUDIENCE	2	观众
HWM_CONF_ROLE_COHOST	3	联席主持人
HWM_CONF_ROLE_WAITING	4	等候室成员

表 9-208 AttendeeType 枚举说明

枚举名称	枚举值	枚举说明
ATTENDEE_TYPE_NORMAL	0	语音、高清、标清与会者地址
ATTENDEE_TYPE_TELEPRESENCE	1	智真与会者地址类型，单屏、三屏智真均属此类
ATTENDEE_TYPE_TERMINAL	2	会议室或硬终端
ATTENDEE_TYPE_OUTSIDE	3	外部与会人
ATTENDEE_TYPE_CUSTOMNUMBER	4	custom number
ATTENDEE_TYPE_MOBILE	5	软终端用户手机
ATTENDEE_TYPE_ANONYMOUS	6	匿名用户
ATTENDEE_TYPE_TELEPHONE	7	电话用户
ATTENDEE_TYPE_BOARD	8	白板用户
ATTENDEE_TYPE_IDEAHUB	9	海思芯片的大屏设备
ATTENDEE_TYPE_HWVISION	10	智慧屏
ATTENDEE_TYPE_WELINKC	11	welink大众版

表 9-209 CycleConfParam

参数名称	是否必选	参数类型	参数说明
startDate	Y	number	开始日期-时间戳，精度秒(0时区)
endDate	Y	number	结束日期-时间戳，精度秒(0时区)
cycleType	Y	CycleType	周期类型
interval	Y	number	周期区间 1、周期类型选择了按天，表示每几天召开一次，取值范围[1,15]；2、周期类型选择了Week，表示每几周召开一次，取值范围[1,5]；3、周期类型选择了Month，interval表示隔几月，取值范围[1,3]
listPoints	Y	string	周期内的会议召开点。仅当按周和月时有效。数据格式(中间用逗号隔开):1,3,5,7
preRemindDays	Y	number	周期子会议提前通知天数。

表 9-210 CycleType 枚举说明

枚举名称	枚举值	枚举说明
CYCLE_TYPE_DAY	0	以天为周期
CYCLE_TYPE_WEEK	1	以周为周期
CYCLE_TYPE_MONTH	2	以月为周期

返回值

void

回调方法参数说明：

(ret: SDKERR, reason: string, confDetail: ConfDetail) => void

表 9-211 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码

参数名称	参数类型	参数说明
reason	string	错误描述
confDetail	ConfDetail	会议详情

表 9-212 ConfDetail

参数名称	参数类型	参数说明
confListInfo	ConfListItem	会议基本信息
vmrId	string	云会议室唯一ID。
vmrFlag	boolean	是否是vmr会议。
isAutoRecord	boolean	是否开启自动录制会议。
allowRecord	boolean	是否支持录制会议
allowLive	boolean	保留。编辑会议时透传会议详情此字段
needPassword	boolean	是否需要密码。
isSendSms	boolean	是否发送短信通知。
isSendEmail	boolean	是否发送邮件通知。
isSendCalendar	boolean	是否发送日历通知。
numOfAttendee	number	与会者列表长度。
isOpenWaitingRoom	boolean	是否开启等候室

表 9-213 ConfListItem

参数名称	参数类型	参数说明
confId	string	会议ID。
subject	string	会议主题。
hostPwd	string	主席密码。
guestPwd	string	来宾密码。
audiencePwd	string	观众密码。(仅对win平台生效)
startTime	number	会议开始时间(utc时间), 单位秒。
endTime	number	会议结束时间(utc时间), 单位秒。
schedulerName	string	会议预定者名称。

参数名称	参数类型	参数说明
vmrConfId	string	用于云会议显示的ID。
hostJoinUri	string	主持人加入会议uri链接
guestJoinUri	string	来宾加入会议uri链接
audienceJoinUri	string	观众加入会议的链接(仅对win平台生效)
confStateType	ConfState	会议状态(仅对win平台生效)
isWebinar	boolean	是否是网络研讨会(仅对win平台生效)

表 9-214 ConfState 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_STATE_SCHEDULE	0	预定状态。
HWM_CONF_STATE_CREATING	1	正在创建状态。
HWM_CONF_STATE_GOING	2	会议已经开始。
HWM_CONF_STATE_DESTROYED	3	会议已经关闭。

表 9-215 ConferenceType 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_TYPE_COMMON	0	普通会议
HWM_CONF_TYPE_CYCLE	1	周期会议

表 9-216 CycleConfParam

参数名称	参数类型	参数说明
startDate	number	开始日期-时间戳, 精度秒(0时区)
endDate	number	结束日期-时间戳, 精度秒(0时区)

参数名称	参数类型	参数说明
interval	number	周期区间 1、周期类型选择了按天，表示每几天召开一次，取值范围[1,15]；2、周期类型选择了Week，表示每几周召开一次，取值范围[1,5]；3、周期类型选择了Month，interval表示隔几月，取值范围[1,3]
listPoints	string	周期内的会议召开点。仅当按周和月时有效。数据格式(中间用逗号隔开): 1,3,5,7

表 9-217 SubCycleConfParam

参数名称	参数类型	参数说明
subConfID	string	子会议唯一标识UUID
startTime	number	会议起始时间
endTime	number	会议结束时间
isAutoRecord	boolean	会议开始后是否自动开启录制
recordAuthType	RecordAuthType	录播鉴权方式

表 9-218 RecordAuthType 枚举说明

枚举名称	枚举值	枚举说明
REOCDRD_AUTH_TYPE_ANYONE	0	可通过连接观看/下载（不拼接nonce）
REOCDRD_AUTH_TYPE_IN_COMPANY_USER	1	企业用户可观看/下载
REOCDRD_AUTH_TYPE_ATTENDEE	2	与会者可观看/下载

表 9-219 AttendeeBaseInfo

参数名称	参数类型	参数说明
nickName	string	与会者名称。

参数名称	参数类型	参数说明
number	string	号码。如果填分配给帐号的sip号码（如+99111244216210249）则呼叫该帐号的App；如果填PSTN号码（如18700000000），则通过VoIP网关呼叫该号码，前提是该企业已开通PSTN呼叫权限。与thirdUserId二选一，帐号密码鉴权时使用。
thirdUserId	string	第三方用户ID。与number二选一，App ID鉴权时使用。
accountId	string	华为云会议帐号
email	string	电子邮箱地址，用于发送邮件通知。
sms	string	短信通知手机号码，用于发送短信通知。
isAutoInvite	boolean	是否自动外邀。
isMute	boolean	是否闭音。

9.5.4.10 编辑周期会议

接口功能

该接口用于编辑周期会议。注意事项：1、该接口在已登录状态下才能调用。2、该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。3、所有字段必填，编辑修改的字段可来源于用户输入，其他不想修改的属性一定要取会议详情对应字段来赋值。

函数原型

```
modifyCycleConf(modifyParam: ModifyCycleConfParam,
onModifyCycleConfResult?: (ret: SDKERR, reason: string) => void): void
```

请求参数

表 9-220 入参说明

参数名称	是否必选	参数类型	参数说明
modifyParam	Y	ModifyCycleConfParam	编辑周期会议的参数体
onModifyCycleConfResult	Y	(ret: SDKERR, reason: string) => void	编辑周期会议的结果回调

表 9-221 ModifyCycleConfParam

参数名称	是否必选	参数类型	参数说明
editConfParam	N	EditConfParam	编辑会议参数
cycleParam	Y	CycleConfParam	子会议参数

表 9-222 EditConfParam

参数名称	是否必选	参数类型	参数说明
confId	Y	string	会议ID。
startTime	Y	number	会议开始时间，如果没有指定开始时间，则表示会议马上开始，精度秒(0时区)
duration	Y	number	会议持续时长，单位分钟
confCommonParam	Y	ConfCommonParam	会议公用参数
isSendCalendar	N	boolean	是否发送日历通知

表 9-223 ConfCommonParam

参数名称	是否必选	参数类型	参数说明
subject	Y	string	会议主题。
mediaType	N	MediaType	会议类型。默认音频会议。
needPassword	N	boolean	是否需要来宾密码。默认不需要来宾密码（仅对随机会议ID生效）。
isAutoRecord	N	boolean	会议自动录制开关。默认不开启。开启会议自动录制开关，内部会默认支持录制会议，录制会议开关的打开与关闭并不影响。说明：仅对云录制生效，客户端本地录制不生效。
allowRecord	N	boolean	是否支持录制会议。（仅对win平台生效）说明：仅对云录制生效，客户端本地录制不生效。
allowLive	N	boolean	保留。编辑会议时透传会议详情此字段

参数名称	是否必选	参数类型	参数说明
timezone	Y	Timezone	时区码。用于发送会议通知。
vmrId	N	string	云会议室唯一ID。预约并发会议时，设置为空字符串。编辑会议时不可修改，填入会议详情返回值。
confAllowJoinUser	N	ConfAllowJoinUserType	允许入会范围限制。默认所有用户。
isSendSms	N	boolean	发送短信通知开关。企业配置里面需要开启，否则不生效。
isSendEmail	N	boolean	发送邮件通知开关。
vmrConfIdType	N	VmrConfIdType	云会议室的会议ID类型。编辑会议时不可修改会议详情中vmrConfId非空，填入HWM_VMR_CONF_ID_TYPE_FIXED；会议详情中vmrConfId为空，填入HWM_VMR_CONF_ID_TYPE_RANDOM
guestPwd	N	string	来宾密码，如果空则服务器随机生成（仅对随机会议ID生效）。
isOpenWaitingRoom	N	boolean	等候室开启开关。说明：需要等候室功能开通后才能生效
numOfAttendee	Y	number	与会者列表长度。
attendees	Y	AttendeeBaseInfo	与会者列表。
allowGuestStartConf	N	boolean	允许来宾启动会议
allowGuestStartConfTime	N	number	允许来宾提前入会时间范围(单位：分钟) 0-随时，n-提前n分钟启动会议

表 9-224 MediaType 枚举说明

枚举名称	枚举值	枚举说明
HWM_MEDIA_TYPE_AUDIO	0	音频类型
HWM_MEDIA_TYPE_VIDEO	1	视频类型

表 9-225 Timezone 枚举说明

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_DEFAULT	0	Mediav默认时区值
HWM_TIMEZONE_INTERDATE_LINE	1	日界线西
HWM_TIMEZONE_MIDWAY_ISLAND	2	中途岛, 萨摩亚群岛
HWM_TIMEZONE_HAWAII	3	夏威夷
HWM_TIMEZONE_ALASKA	4	阿拉斯加
HWM_TIMEZONE_PACIFIC_TIME	5	太平洋时间(美国和加拿大);蒂华纳
HWM_TIMEZONE_ARIZONA	6	亚利桑那
HWM_TIMEZONE_MOUNTAIN_TIME	7	山地时间(美国和加拿大)
HWM_TIMEZONE_CENTRAL_AMERICA	8	中美洲
HWM_TIMEZONE_CENTRAL_TIME	9	中间时间(美国和加拿大)
HWM_TIMEZONE_MEXICO_CITY	10	墨西哥城
HWM_TIMEZONE_SASKATCHEWAN	11	萨斯喀彻温
HWM_TIMEZONE_BOGOTA	12	波哥大, 利马, 基多
HWM_TIMEZONE_EASTERN_TIME	13	东部时间(美国和加拿大)
HWM_TIMEZONE_INDIANA	14	印第安纳(东部)
HWM_TIMEZONE_ATLANTIC_TIME	15	大西洋时间(加拿大)
HWM_TIMEZONE_CARACAS	16	加拉加斯, 拉巴斯
HWM_TIMEZONE_SANTIAGO	17	圣地亚哥
HWM_TIMEZONE_NEWFOUNDLAND	18	纽芬兰
HWM_TIMEZONE_BRASILIA	19	巴西利亚
HWM_TIMEZONE_BUENOS_AIRES	20	布宜诺斯艾利斯, 乔治敦

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_GREENLAND	21	格陵兰
HWM_TIMEZONE_MID_ATLANTIC	22	中大西洋
HWM_TIMEZONE_AZORES	23	亚速尔群岛
HWM_TIMEZONE_CAPEVERDE	24	佛得角群岛
HWM_TIMEZONE_MONROVIA	25	蒙罗维亚
HWM_TIMEZONE_GREENWICH_MEANTIME	26	格林威治标准时：都柏林，爱丁堡，里斯本，伦敦
HWM_TIMEZONE_AMSTERDAM	27	阿姆斯特丹，柏林，伯尔尼，罗马，斯德哥尔摩，维也纳
HWM_TIMEZONE_BELGRADE	28	贝尔格莱德，布拉迪斯拉发，布达佩斯，卢布尔亚纳，布拉格
HWM_TIMEZONE_BRUSSELS	29	布鲁塞尔，哥本哈根，马德里，巴黎
HWM_TIMEZONE_SARAJEVO	30	萨拉热窝，斯科普里，维尔纽斯，索非亚，华沙，萨格勒布
HWM_TIMEZONE_WESTCENTRAL_AFRICA	31	中非西部
HWM_TIMEZONE_ATHENS	32	雅典，伊斯坦布尔，明斯克
HWM_TIMEZONE_BUCHAREST	33	布加勒斯特
HWM_TIMEZONE_CAIRO	34	开罗
HWM_TIMEZONE_HARARE	35	哈拉雷，比勒陀利亚
HWM_TIMEZONE_HELSINKI	36	赫尔辛基，里加，塔林
HWM_TIMEZONE_JERUSALEM	37	耶路撒冷
HWM_TIMEZONE_BAGHDAD	38	巴格达
HWM_TIMEZONE_KUWAIT	39	科威特，利雅得
HWM_TIMEZONE_MOSCOW	40	莫斯科，圣彼得堡，喀山，伏尔加格勒
HWM_TIMEZONE_NAIROBI	41	内罗毕
HWM_TIMEZONE_TEHRAN	42	德黑兰
HWM_TIMEZONE_ABU_DHABI	43	阿布扎比，马斯喀特
HWM_TIMEZONE_TBILISI	44	第比利斯，埃里温
HWM_TIMEZONE_KABUL	45	喀布尔

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_EKATERINBURG	46	叶卡捷琳堡
HWM_TIMEZONE_ISLAMABAD	47	伊斯兰堡, 卡拉奇, 塔什干
HWM_TIMEZONE_CALCUTTA	48	加尔各答, 马德拉斯, 孟买, 新德里
HWM_TIMEZONE_KATHMANDU	49	加德满都
HWM_TIMEZONE_ALMATY	50	阿拉木图
HWM_TIMEZONE_ASTANA	51	阿斯塔纳, 达卡
HWM_TIMEZONE_SRIJAYA	52	斯里哈亚华登尼普拉
HWM_TIMEZONE_RANGOON	53	仰光
HWM_TIMEZONE_BANGKOK	54	曼谷, 雅加达, 河内
HWM_TIMEZONE_NOVOSIBIRSK	55	新西伯利亚
HWM_TIMEZONE_BEIJING	56	北京, 重庆, 香港, 乌鲁木齐, 台北
HWM_TIMEZONE_KRASNOYARSK	57	克拉斯诺亚尔斯克, 乌兰巴托
HWM_TIMEZONE_KUALALUMPUR	58	吉隆坡, 新加坡
HWM_TIMEZONE_PERTH	59	珀斯
HWM_TIMEZONE_OSAKA	60	东京, 大阪, 札幌
HWM_TIMEZONE_SEOUL	61	汉城
HWM_TIMEZONE_YAKUTSK	62	雅库茨克
HWM_TIMEZONE_ADELAIDE	63	阿德莱德
HWM_TIMEZONE_DARWIN	64	达尔文
HWM_TIMEZONE_BRISBANE	65	布里斯班
HWM_TIMEZONE_CANBERRA	66	堪培拉, 墨尔本, 悉尼
HWM_TIMEZONE_GUAM	67	关岛, 莫尔兹比港
HWM_TIMEZONE_HOBART	68	霍巴特
HWM_TIMEZONE_VLADIVOSTOK	69	符拉迪沃斯托克
HWM_TIMEZONE_SOLOMON	70	所罗门群岛, 新喀里多尼亚

枚举名称	枚举值	枚举说明
HWM_TIMEZONE_AUCKLAND	71	奥克兰，惠灵顿
HWM_TIMEZONE_FIJI	72	富士，堪察加半岛，马绍尔群岛
HWM_TIMEZONE_NUKUALOFA	73	努库阿洛法
HWM_TIMEZONE_IRKUTSK	74	伊尔库茨克
HWM_TIMEZONE_CASABLANCA	75	卡萨布兰卡
HWM_TIMEZONE_BAKU	76	巴库
HWM_TIMEZONE_MAGADAN	77	马加丹

表 9-226 ConfAllowJoinUserType 枚举说明

枚举名称	枚举值	枚举说明
CONF_ALLOW_JOIN_ANYONE	0	所有用户。
CONF_ALLOW_JOIN_LOGINED_USER	1	已登录用户。
CONF_ALLOW_JOIN_IN_COMPANY_USER	2	企业内用户。
CONF_ALLOW_JOIN_INVITED_USER	3	被邀请用户。

表 9-227 VmrConfIdType 枚举说明

枚举名称	枚举值	枚举说明
HWM_VMR_CONF_ID_TYPE_FIXED	0	云会议室的会议ID类型是固定类型。
HWM_VMR_CONF_ID_TYPE_RANDOM	1	云会议室的会议ID类型是随机类型。

表 9-228 AttendeeBaseInfo

参数名称	是否必选	参数类型	参数说明
nickName	Y	string	与会者名称。

参数名称	是否必选	参数类型	参数说明
number	Y	string	号码。如果填分配给帐号的sip号码（如+99111244216210249）则呼叫该帐号的App；如果填PSTN号码（如18700000000），则通过VoIP网关呼叫该号码，前提是该企业已开通PSTN呼叫权限。与thirdUserId二选一，帐号密码鉴权时使用。
thirdUserId	Y	string	第三方用户ID。与number二选一，App ID鉴权时使用。
accountId	N	string	华为云会议帐号
email	N	string	电子邮箱地址，用于发送邮件通知。
sms	N	string	短信通知手机号码，用于发送短信通知。
isAutoInvite	N	boolean	是否自动外邀。
isMute	N	boolean	是否闭音。
role	N	ConfRole	会议角色。
type	N	AttendeeType	成员类型

表 9-229 ConfRole 枚举说明

枚举名称	枚举值	枚举说明
HWM_CONF_ROLE_ATTENDEE	0	普通与会者
HWM_CONF_ROLE_HOST	1	主席
HWM_CONF_ROLE_AUDIENCE	2	观众
HWM_CONF_ROLE_COHOST	3	联席主持人
HWM_CONF_ROLE_WAITING	4	等候室成员

表 9-230 AttendeeType 枚举说明

枚举名称	枚举值	枚举说明
ATTENDEE_TYPE_NORMAL	0	语音、高清、标清与会者地址

枚举名称	枚举值	枚举说明
ATTENDEE_TYPE_TELEPRESENCE	1	智真与会者地址类型，单屏、三屏智真均属此类
ATTENDEE_TYPE_TERMINAL	2	会议室或硬终端
ATTENDEE_TYPE_OUTSIDE	3	外部与会人
ATTENDEE_TYPE_CUSTOMNUMBER	4	custom number
ATTENDEE_TYPE_MOBILE	5	软终端用户手机
ATTENDEE_TYPE_ANONYMOUS	6	匿名用户
ATTENDEE_TYPE_TELEPHONE	7	电话用户
ATTENDEE_TYPE_BOARD	8	白板用户
ATTENDEE_TYPE_IDEAHUB	9	海思芯片的大屏设备
ATTENDEE_TYPE_HWVISION	10	智慧屏
ATTENDEE_TYPE_WELINKC	11	welink大众版

表 9-231 CycleConfParam

参数名称	是否必选	参数类型	参数说明
startDate	Y	number	开始日期-时间戳，精度秒(0时区)
endDate	Y	number	结束日期-时间戳，精度秒(0时区)
cycleType	Y	CycleType	周期类型
interval	Y	number	周期区间 1、周期类型选择了按天，表示每几天召开一次，取值范围[1,15]；2、周期类型选择了Week，表示每几周召开一次，取值范围[1,5]；3、周期类型选择了Month，interval表示隔几月，取值范围[1,3]
listPoints	Y	string	周期内的会议召开点。仅当按周和月时有效。数据格式(中间用逗号隔开):1,3,5,7
preRemindDays	Y	number	周期子会议提前通知天数。

表 9-232 CycleType 枚举说明

枚举名称	枚举值	枚举说明
CYCLE_TYPE_DAY	0	以天为周期
CYCLE_TYPE_WEEK	1	以周为周期
CYCLE_TYPE_MONTH	2	以月为周期

返回值

void

回调方法参数说明:

(ret: SDKERR, reason: string) => void

表 9-233 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.4.11 编辑周期子会议

接口功能

该接口用于编辑周期子会议。注意事项：1、该接口在已登录状态下才能调用。2、该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

函数原型

modifySubCycleConf(modifySubCycleConfParam: ModifySubCycleConfParam, onModifySubCycleConfResult?: (ret: SDKERR, reason: string) => void): void

请求参数

表 9-234 入参说明

参数名称	是否必选	参数类型	参数说明
modifySubCycleConfParam	Y	ModifySubCycleConfParam	编辑周期会议的参数体

参数名称	是否必选	参数类型	参数说明
onModifySubCycleConfResult	Y	(ret: SDKERR, reason: string) => void	编辑周期子会议的结果回调

表 9-235 ModifySubCycleConfParam

参数名称	是否必选	参数类型	参数说明
confId	Y	string	会议id
subConfId	Y	string	子会议唯一标识UUID
mediaType	Y	MediaType	会议类型。
startTime	N	number	会议开始时间（UTC时间），单位为秒。
duration	Y	number	会议持续时长，单位分钟
isAutoRecord	N	boolean	是否开启自动录制会议。开启自动录制会议开关，内部会默认支持录制会议，录制会议开关的打开与关闭并不影响。说明：仅对云录制生效，客户端本地录制不生效。
recordAuthType	N	RecordAuthType	录播鉴权方式
confAllowJoinUser	N	ConfAllowJoinUserType	允许入会范围限制。默认所有用户。
allowGuestStartConf	N	boolean	允许来宾启动会议
allowGuestStartConfTime	N	number	允许来宾提前入会时间范围(单位：分钟) 0-随时，n-提前n分钟启动会议

表 9-236 MediaType 枚举说明

枚举名称	枚举值	枚举说明
HWM_MEDIA_TYPE_AUDIO	0	音频类型
HWM_MEDIA_TYPE_VIDEO	1	视频类型

表 9-237 RecordAuthType 枚举说明

枚举名称	枚举值	枚举说明
REOCDR_AUTH_TYPE_ANYONE	0	可通过连接观看/下载（不拼接nonce）
REOCDR_AUTH_TYPE_IN_COMPANY_USER	1	企业用户可观看/下载
REOCDR_AUTH_TYPE_ATTENDEE	2	与会者可观看/下载

表 9-238 ConfAllowJoinUserType 枚举说明

枚举名称	枚举值	枚举说明
CONF_ALLOW_JOIN_ANYONE	0	所有用户。
CONF_ALLOW_JOIN_LOGINED_USER	1	已登录用户。
CONF_ALLOW_JOIN_IN_COMPANY_USER	2	企业内用户。
CONF_ALLOW_JOIN_INVITED_USER	3	被邀请用户。

返回值

void

回调方法参数说明：

(ret: SDKERR, reason: string) => void

表 9-239 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.4.12 取消周期会议

接口功能

该接口用于取消至少还有一场子会议未开始的周期会议。注意事项：1、该接口在登录状态下才能调用。2、最后一场子会议未开始前可以调用接口，开始后调用则无效。3、该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

函数原型

```
cancelCycleConf(cancelCycleConfParam: CancelCycleConfParam,
onCancelCycleConfResult?: (ret: SDKERR, reason: string) => void): void
```

请求参数

表 9-240 入参说明

参数名称	是否必选	参数类型	参数说明
cancelCycleConfParam	Y	CancelCycleConfParam	取消周期会议参数
onCancelCycleConfResult	Y	(ret: SDKERR, reason: string) => void	取消周期会议的结果回调

表 9-241 CancelCycleConfParam

参数名称	是否必选	参数类型	参数说明
confId	Y	string	会议id
endOnlineConf	Y	boolean	是否需要结束在线会议

返回值

void

回调方法参数说明：

(ret: SDKERR, reason: string) => void

表 9-242 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码

参数名称	参数类型	参数说明
reason	string	错误描述

9.5.4.13 取消周期子会议

接口功能

该接口用于取消周期子会议。注意事项：1、该接口在已登录状态下才能调用。2、子会议未开始前可以调用接口，开始后调用则无效 3、该接口为异步接口，返回值只代表接口是否调用成功，实际业务处理结果在对应的回调函数中返回。

函数原型

```
cancelSubCycleConf(cancelSubCycleConfParam: CancelSubCycleConfParam,
onCancelSubCycleConfResult?: (ret: SDKERR, reason: string) => void): void
```

请求参数

表 9-243 入参说明

参数名称	是否必选	参数类型	参数说明
cancelSubCycleConfParam	Y	CancelSubCycleConfParam	取消周期子会议参数
onCancelSubCycleConfResult	Y	(ret: SDKERR, reason: string) => void	取消周期子会议的结果回调

表 9-244 CancelSubCycleConfParam

参数名称	是否必选	参数类型	参数说明
confId	Y	string	会议id
subConfId	Y	string	子会议唯一标识UUID
endOnlineConf	Y	boolean	是否需要结束在线会议

返回值

void

回调方法参数说明：

```
(ret: SDKERR, reason: string) => void
```

表 9-245 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.5 通用配置管理对象

通用配置管理对象

9.5.5.1 功能配置开关

接口功能

该接口用于修改会议相关功能配置开关。注意事项：1、枚举中的功能项均有默认值，若对开关状态不做要求，可以不必调用该接口。

函数原型

```
enableCapability(capabilityInfo: CapabilityInfo, onEnableCapabilityResult?: (ret: SDKERR, reason: string) => void): void
```

请求参数

表 9-246 入参说明

参数名称	是否必选	参数类型	参数说明
capabilityInfo	Y	CapabilityInfo	功能配置信息
onEnableCapabilityResult	Y	(ret: SDKERR, reason: string) => void	功能开关配置接口回调

表 9-247 CapabilityInfo

参数名称	是否必选	参数类型	参数说明
type	Y	CapabilityType	功能类型
isEnabled	Y	boolean	开启状态

表 9-248 CapabilityType 枚举说明

枚举名称	枚举值	枚举说明
HWM_CAPABILITY_TYPE_SHOW_HANGUP_ATTENDEE	0	显示离会与会者开关(仅对win平台生效)
HWM_CAPABILITY_TYPE_HIDE_EXTERNAL_LABEL	1	隐藏外部标签
HWM_CAPABILITY_TYPE_RESOLUTION_1080P	2	1080P开关(仅对win平台生效)
HWM_CAPABILITY_TYPE_RESOLUTION_720P	3	720p开关(仅对mac平台生效)
HWM_CAPABILITY_TYPE_MULTI_MONITOR	4	多显示器模式开关(仅对win平台生效)
HWM_CAPABILITY_TYPE_HOWLING_DETECTION	5	啸叫检测开关(仅对win平台生效)
HWM_CAPABILITY_AUTO_MIC_VOLUME	6	自动调节麦克风音量开关(仅对win平台生效)
HWM_CAPABILITY_SHARE_AUDIO	7	共享音频开关
HWM_CAPABILITY_HIDE_NON_VIDEO_ATTENDEE_CAMERA_VIEW	8	隐藏非视频与会者开关(仅对mac平台生效)
HWM_CAPABILITY_BREAKOUT_CONF	9	分组讨论开关
HWM_CAPABILITY_CONFIRM_RECORD	10	录制确认

返回值

void

回调方法参数说明:

(ret: SDKERR, reason: string) => void

表 9-249 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.5.2 设置语言

接口功能

该接口用于设置语言，默认是中文。

函数原型

```
setLanguage(languageInfo: LanguageInfo, onSetLanguageResult?: (ret: SDKERR, reason: string) => void): void
```

请求参数

表 9-250 入参说明

参数名称	是否必选	参数类型	参数说明
languageInfo	Y	LanguageInfo	语言类型信息
onSetLanguageResult	Y	(ret: SDKERR, reason: string) => void	设置语言接口回调

表 9-251 LanguageInfo

参数名称	是否必选	参数类型	参数说明
type	Y	Language	语言类型
languageCode	N	string	自定义语言区域名

表 9-252 Language 枚举说明

枚举名称	枚举值	枚举说明
HWM_LANGUAGE_ZH_CN	0	中文
HWM_LANGUAGE_EN_GB	1	英文
HWM_LANGUAGE_CUSTOMIZED	2	自定义语言

返回值

void

回调方法参数说明：

(ret: SDKERR, reason: string) => void

表 9-253 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.5.3 设置用户保存路径

接口功能

该接口用于设置用户文件（包括本地录制文件）的保存路径。注意事项：1、该接口非必需调用，若不设置，则使用默认路径。

函数原型

setUserSavePath(userSavePathInfo: UserSavePathInfo, onSetUserSavePathResult?: (ret: SDKERR, reason: string) => void): void

请求参数

表 9-254 入参说明

参数名称	是否必选	参数类型	参数说明
userSavePathInfo	Y	UserSavePathInfo	保存路径信息
onSetUserSavePathResult	Y	(ret: SDKERR, reason: string) => void	设置用户保存路径结果回调

表 9-255 UserSavePathInfo

参数名称	是否必选	参数类型	参数说明
pathType	Y	SavePathType	保存路径类型
path	Y	string	保存路径，注：绝对路径，需要UTF8编码

表 9-256 SavePathType 枚举说明

枚举名称	枚举值	枚举说明
HWM_SAVE_PATH_LOCAL_RECORD	0	本地录制文件保存路径

返回值

void

回调方法参数说明：

(ret: SDKERR, reason: string) => void

表 9-257 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.5.4 设置来电应答方式

接口功能

该接口用于设置来电应答: 原始弹框、自动拒绝或自动接听。

函数原型

setAnswerType(inComingAnswerTypeInfo: InComingAnswerTypeInfo, onSetAnswerTypeResult?: (ret: SDKERR, reason: string) => void): void

请求参数

表 9-258 入参说明

参数名称	是否必选	参数类型	参数说明
inComingAnswerTypeInfo	Y	InComingAnswerTypeInfo	来电应答信息
onSetAnswerTypeResult	Y	(ret: SDKERR, reason: string) => void	设置来电应答结果回调

表 9-259 InComingAnswerTypeInfo

参数名称	是否必选	参数类型	参数说明
type	Y	InComingAnswerType	来电应答类型

表 9-260 InComingAnswerType 枚举说明

枚举名称	枚举值	枚举说明
HWM_INCOMING_ANSWER_NORMAL	0	默认，来电时弹出来电接听界面
HWM_INCOMING_ANSWER_DECLINE	1	来电时拒接
HWM_INCOMING_ANSWER_ANSWER	2	来电时接听

返回值

void

回调方法参数说明：

(ret: SDKERR, reason: string) => void

表 9-261 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.5.5 会议来电时设备开关配置

接口功能

该接口用于设置接听会议来电时设备开关状态，包括麦克风和摄像头。

函数原型

```
setDeviceConfigForConflncoming(deviceConfig: DeviceConfig,
onSetDeviceConfigForConflncomingResult?: (ret: SDKERR, reason: string) => void):
void
```

请求参数

表 9-262 入参说明

参数名称	是否必选	参数类型	参数说明
deviceConfig	Y	DeviceConfig	设备配置信息
onSetDeviceConfigForIncomingResult	Y	(ret: SDKERR, reason: string) => void	设置接听会议来电时麦克风/摄像头开关状态结果回调

表 9-263 DeviceConfig

参数名称	是否必选	参数类型	参数说明
isMicOn	N	boolean	是否开启麦克风
isCameraOn	N	boolean	是否开启摄像头

返回值

void

回调方法参数说明:

(ret: SDKERR, reason: string) => void

表 9-264 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.5.6 获取会议 pstn 接入号码

接口功能

该接口用于获取会议pstn接入号码。

函数原型

getConfigPstnAccessNumber(onGetConfPstnAccessNumberResult?: (ret: SDKERR, reason: string, pstnAccessNumberInfo: PstnAccessNumberInfo) => void): void

请求参数

表 9-265 入参说明

参数名称	是否必选	参数类型	参数说明
onGetConfPstnAccessNumberResult	Y	(ret: SDKERR, reason: string, pstnAccessNumberInfo: PstnAccessNumberInfo) => void	获取会议pstn接入号码结果回调

返回值

void

回调方法参数说明:

(ret: SDKERR, reason: string, pstnAccessNumberInfo: PstnAccessNumberInfo) => void

表 9-266 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述
pstnAccessNumberInfo	PstnAccessNumberInfo	会议pstn接入号码信息

表 9-267 PstnAccessNumberInfo

参数名称	参数类型	参数说明
pstnAccessNumber	string	会议pstn接入号码

9.5.5.7 禁用提示音

接口功能

该接口用于禁用提示音。

函数原型

```
disableVoicePrompts(voicePromptsInfo: VoicePromptsInfo,
onDisableVoicePromptsResult?: (ret: SDKERR, reason: string) => void): void
```

请求参数

表 9-268 入参说明

参数名称	是否必选	参数类型	参数说明
voicePromptsInfo	Y	VoicePromptsInfo	语音提示信息
onDisableVoicePromptsResult	Y	(ret: SDKERR, reason: string) => void	禁用语音提示接口结果回调

表 9-269 VoicePromptsInfo

参数名称	是否必选	参数类型	参数说明
isDisableMutePrompt	Y	boolean	是否禁用静音提示

返回值

void

回调方法参数说明:

```
(ret: SDKERR, reason: string) => void
```

表 9-270 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.5.8 设置共享权限

接口功能

该接口用于设置共享权限，默认是允许共享。

函数原型

```
setShareScreenPermission(permission: SharePermission,
onSetShareScreenPermissionResult?: (ret: SDKERR, reason: string) => void): void
```

请求参数

表 9-271 入参说明

参数名称	是否必选	参数类型	参数说明
permission	Y	SharePermission	权限类型信息
onSetShareScreenPermissionResult	Y	(ret: SDKERR, reason: string) => void	设置共享权限接口回调

表 9-272 SharePermission

参数名称	是否必选	参数类型	参数说明
isAllowed	Y	boolean	是否禁用共享权限

返回值

void

回调方法参数说明:

(ret: SDKERR, reason: string) => void

表 9-273 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.6 界面配置管理对象

界面配置管理对象

9.5.6.1 订阅注入点击通知

接口功能

收到该通知时，用户点击了配置过点击注入的按钮。

函数原型

```
setOnClickInjectBtnNotifyCB(onClickInjectBtnNotify: (injectBtnInfo: InjectBtnInfo) => void): void
```

请求参数

表 9-274 入参说明

参数名称	是否必选	参数类型	参数说明
onClickInjectBtnNotify	Y	(injectBtnInfo: InjectBtnInfo) => void	收到该通知时，用户点击了配置过点击注入的按钮。

返回值

void

通知方法参数说明：

```
(injectBtnInfo: InjectBtnInfo) => void
```

表 9-275 入参说明

参数名称	参数类型	参数说明
injectBtnInfo	InjectBtnInfo	注入按钮点击通知信息

表 9-276 InjectBtnInfo

参数名称	参数类型	参数说明
injectBtn	ClickInjectBtn	按钮类型
btnId	string	按钮id

表 9-277 ClickInjectBtn 枚举说明

枚举名称	枚举值	枚举说明
HWM_CLICK_INJECT_BTN_INVITE	0	邀请按钮
HWM_CLICK_INJECT_BTN_ATTENDEE	1	与会者按钮
HWM_CLICK_INJECT_BTN_CHAT	2	聊天按钮
HWM_CLICK_INJECT_BTN_SHARE	3	共享按钮
HWM_CLICK_INJECT_BTN_LEAVE	4	离开会议按钮
HWM_CLICK_INJECT_BTN_CUSTOM	5	用户自定义按钮
HWM_CLICK_INJECT_BTN_GUEST_INVITE	6	来宾邀请按钮
HWM_CLICK_INJECT_BTN_FEEDBACK	7	反馈按钮
HWM_CLICK_INJECT_BTN_COPY_CONF_INFO	8	复制会议信息按钮

9.5.6.2 订阅与会者列表子项注入点击通知

接口功能

收到该通知时，用户点击了与会者列表子项中配置过点击注入的按钮。

函数原型

```
setOnClickInjectParticipantItemBtnNotifyCB(onClickInjectParticipantItemBtnNotify:
(injectBtnInfo: InjectParticipantItemBtnInfo) => void): void
```

请求参数

表 9-278 入参说明

参数名称	是否必选	参数类型	参数说明
onClickInjectParticipantItemBtnNotify	Y	(injectBtnInfo: InjectParticipantItemBtnInfo) => void	收到该通知时，用户点击了与会者列表子项中配置过点击注入的按钮。

返回值

void

通知方法参数说明：

(injectBtnInfo: InjectParticipantItemBtnInfo) => void

表 9-279 入参说明

参数名称	参数类型	参数说明
injectBtnInfo	InjectParticipantItemBtnInfo	注入按钮点击通知信息

表 9-280 InjectParticipantItemBtnInfo

参数名称	参数类型	参数说明
injectBtn	ClickInjectParticipantItemBtn	按钮类型
participantInfo	ParticipantItemInfo	与会者子项信息
btnId	string	按钮id

表 9-281 ClickInjectParticipantItemBtn 枚举说明

枚举名称	枚举值	枚举说明
HWM_CLICK_INJECT_PARTICIPANT_ITEM_BTN_CALL_OTHER_NUM	0	呼叫其他号码按钮
HWM_CLICK_INJECT_PARTICIPANT_ITEM_BTN_PARTICIPANT_DETAIL	1	查看与会者详情

表 9-282 ParticipantItemInfo

参数名称	参数类型	参数说明
number	string	号码
nickName	string	与会者昵称

参数名称	参数类型	参数说明
thirdUserId	string	与会者三方帐号，appId登录时使用
accountId	string	华为云会议帐号

9.5.6.3 订阅窗口句柄通知

接口功能

收到该通知时，窗口句柄信息发生变更。

函数原型

```
setOnRenderWindowInfoListChangedCB(onRenderWindowInfoListChanged:
(hwndInfo: HwndListInfo) => void): void
```

请求参数

表 9-283 入参说明

参数名称	是否必选	参数类型	参数说明
onRenderWindowInfoListChanged	Y	(hwndInfo: HwndListInfo) => void	收到该通知时，窗口句柄信息发生变更。

返回值

void

通知方法参数说明：

```
(hwndInfo: HwndListInfo) => void
```

表 9-284 入参说明

参数名称	参数类型	参数说明
hwndInfo	HwndListInfo	窗口句柄信息

表 9-285 HwndListInfo

参数名称	参数类型	参数说明
hwndInfoList	HwndInfo	视频窗口句柄列表

参数名称	参数类型	参数说明
hwndListLength	number	窗口句柄列表的实际长度

9.5.6.4 订阅查询外部通讯录头像通知

接口功能

收到该通知时，用户打开了界面配置中的外部通讯录头像信息的开关，入会且未打开摄像头。

函数原型

```
setOnQueryContactAvatarNotifyCB(onQueryContactAvatarNotify: (contactInfo: ContactInfo) => void): void
```

请求参数

表 9-286 入参说明

参数名称	是否必选	参数类型	参数说明
onQueryContactAvatarNotify	Y	(contactInfo: ContactInfo) => void	收到该通知时，用户打开了界面配置中的外部通讯录头像信息的开关，入会且未打开摄像头。

返回值

void

通知方法参数说明：

```
(contactInfo: ContactInfo) => void
```

表 9-287 入参说明

参数名称	参数类型	参数说明
contactInfo	ContactInfo	查询外部头像的联系人信息

表 9-288 ContactInfo

参数名称	参数类型	参数说明
accountId	string	帐号，帐号密码登录场景下优先使用此字段
thirdUserId	string	与会者三方帐号，appid登录场景下优先使用此字段
number	string	号码，跨企业场景下优先使用此字段
userId	string	UUID(由系统分配的真实的唯一用户标识)(仅在win平台下有效)

9.5.6.5 界面配置

接口功能

该接口用于配置定制化参数，自定义会议中界面的部分元素。

函数原型

```
config(uiConfig: string, onConfigResult?: (ret: SDKERR, reason: string) => void): void
```

请求参数

表 9-289 入参说明

参数名称	是否必选	参数类型	参数说明
uiConfig	Y	string	配置信息，为Json格式字符串，详细请参考 界面配置json说明
onConfigResult	Y	(ret: SDKERR, reason: string) => void	配置定制化参数回调

返回值

void

回调方法参数说明:

```
(ret: SDKERR, reason: string) => void
```

表 9-290 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.6.6 自定义会中水印

接口功能

该接口用于设置水印自定义内容注入，若未进行设置，将默认使用内部优先级来展示水印内容。注意事项：1、该接口必须在会前调用，会中调用不会即时生效；2、企业级水印开关未打开的场景下，水印自定义内容注入也不会生效。

函数原型

```
setWaterMark(waterMarkInfo: WaterMarkInfo, onSetWaterMarkResult?: (ret: SDKERR, reason: string) => void): void
```

请求参数

表 9-291 入参说明

参数名称	是否必选	参数类型	参数说明
waterMarkInfo	Y	WaterMarkInfo	水印注入信息
onSetWaterMarkResult	Y	(ret: SDKERR, reason: string) => void	设置水印注入信息回调

表 9-292 WaterMarkInfo

参数名称	是否必选	参数类型	参数说明
content	N	string	用户自定义水印注入内容
color	N	WaterMarkColorType	水印字体颜色，0-浅色，1-标准色，2-深色

表 9-293 WaterMarkColorType 枚举说明

枚举名称	枚举值	枚举说明
HWM_WATERMARK_COLOR_TYPE_LIGHT	0	浅色 #CCCCCC
HWM_WATERMARK_COLOR_TYPE_STANDARD	1	标准色 #999999
HWM_WATERMARK_COLOR_TYPE_DARKER	2	深色 #666666

返回值

void

回调方法参数说明:

(ret: SDKERR, reason: string) => void

表 9-294 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.6.7 获取窗口句柄列表

接口功能

该接口用于获取窗口句柄列表。

函数原型

getRenderWindowInfoList(onGetRenderWindowInfoListResult?: (ret: SDKERR, reason: string, hwndInfo: HwndListInfo) => void): void

请求参数

表 9-295 入参说明

参数名称	是否必选	参数类型	参数说明
onGetRenderWindowInfoListResult	Y	(ret: SDKERR, reason: string, hwndInfo: HwndListInfo) => void	窗口句柄列表回调, GetRenderWindowInfoList 请求返回

返回值

void

回调方法参数说明:

(ret: SDKERR, reason: string, hwndInfo: HwndListInfo) => void

表 9-296 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述
hwndInfo	HwndListInfo	窗口句柄信息

表 9-297 HwndListInfo

参数名称	参数类型	参数说明
hwndInfoList	HwndInfo	视频窗口句柄列表
hwndListLength	number	窗口句柄列表的实际长度

9.5.6.8 显示会议窗口

接口功能

当会议主窗口处于最小化或者被其他窗口覆盖时,使用该接口将会议主窗口拉到最上层。一般用于从会议列表中拉起正在进行的会议窗口。

函数原型

showMeetingWindow(onShowMeetingWindowResult?: (ret: SDKERR, reason: string) => void): void

请求参数

表 9-298 入参说明

参数名称	是否必选	参数类型	参数说明
onShowMeetingWindowResult	Y	(ret: SDKERR, reason: string) => void	显示会议窗口

返回值

void

回调方法参数说明:

(ret: SDKERR, reason: string) => void

表 9-299 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.6.9 设置头像

接口功能

该接口用于设置头像信息。

函数原型

setAvatar(avatarInfo: AvatarInfo, onSetAvatarResult?: (ret: SDKERR, reason: string) => void): void

请求参数

表 9-300 入参说明

参数名称	是否必选	参数类型	参数说明
avatarInfo	Y	AvatarInfo	头像信息
onSetAvatarResult	Y	(ret: SDKERR, reason: string) => void	设置头像回调

表 9-301 AvatarInfo

参数名称	是否必选	参数类型	参数说明
number	Y	string	号码参数
avatarPath	Y	string	头像路径

返回值

void

回调方法参数说明：

(ret: SDKERR, reason: string) => void

表 9-302 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.6.10 自定义会议窗口任务栏图标

接口功能

该接口用于设置会议窗口任务栏图标信息(仅对mac平台生效)，win平台可以通过替换HwmSdk\sdkResources\nativeimg目录下的video.ico图标文件实现。

函数原型

```
setMeetingWindowIcon(windowIconInfo: WindowIconInfo,
onSetMeetingWindowIconResult?: (ret: SDKERR, reason: string) => void): void
```

请求参数

表 9-303 入参说明

参数名称	是否必选	参数类型	参数说明
windowIconInfo	Y	WindowIconInfo	窗口图标信息
onSetMeetingWindowIconResult	Y	(ret: SDKERR, reason: string) => void	设置会议窗口图标回调

表 9-304 WindowIconInfo

参数名称	是否必选	参数类型	参数说明
iconPath	Y	string	图标路径

返回值

void

回调方法参数说明：

(ret: SDKERR, reason: string) => void

表 9-305 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.6.11 显示设置窗口接口

接口功能

该接口用于显示设置窗口接口。

函数原型

showSettingWindow(showInfo: ShowSettingWindowInfo, onShowSettingWindowResult?: (ret: SDKERR, reason: string) => void): void

请求参数

表 9-306 入参说明

参数名称	是否必选	参数类型	参数说明
showInfo	Y	ShowSettingWindowInfo	显示设置窗口信息
onShowSettingWindowResult	Y	(ret: SDKERR, reason: string) => void	显示设置界面接口结果回调

表 9-307 ShowSettingWindowInfo

参数名称	是否必选	参数类型	参数说明
isShow	Y	boolean	是否显示
defaultPage	Y	SettingPage	默认显示标签页

表 9-308 SettingPage 枚举说明

枚举名称	枚举值	枚举说明
SETTING_PAGE_NORMAL	0	基本设置页
SETTING_PAGE_VIDEO	1	视频设置页
SETTING_PAGE_AUDIO	2	音频设置页
SETTING_PAGE_RECORD	3	录制设置页
SETTING_PAGE_HOTKEY	4	快捷键设置页
SETTING_PAGE_SCREEN	5	屏幕设置页
SETTING_PAGE_COUNT	6	设置页数量

返回值

void

回调方法参数说明:

(ret: SDKERR, reason: string) => void

表 9-309 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.5.7 呼叫管理对象

呼叫管理对象

9.5.7.1 订阅通话状态通知

接口功能

收到该通知时，表示通话状态发生变更。

函数原型

```
setOnCallStateChangedCB(onCallStateChanged: (callStateInfo: CallStateInfo) => void): void
```

请求参数

表 9-310 入参说明

参数名称	是否必选	参数类型	参数说明
onCallStateChanged	Y	(callStateInfo: CallStateInfo) => void	收到该通知时，表示通话状态发生变更。

返回值

void

通知方法参数说明：

```
(callStateInfo: CallStateInfo) => void
```

表 9-311 入参说明

参数名称	参数类型	参数说明
callStateInfo	CallStateInfo	通话状态信息

表 9-312 CallStateInfo

参数名称	参数类型	参数说明
state	CallState	通话状态

表 9-313 CallState 枚举说明

枚举名称	枚举值	枚举说明
HWM_CALL_STATE_IDLE	0	空闲状态
HWM_CALL_STATE_BUSY	1	忙碌状态

9.5.7.2 订阅通话结束通知

接口功能

收到该通知时，表示呼叫已结束，同时返回通话记录信息。

函数原型

```
setOnCallEndedNotifyCB(onCallEndedNotify: (callRecordInfo: CallRecordInfo) => void): void
```

请求参数

表 9-314 入参说明

参数名称	是否必选	参数类型	参数说明
onCallEndedNotify	Y	(callRecordInfo: CallRecordInfo) => void	收到该通知时，表示呼叫已结束，同时返回通话记录信息。

返回值

void

通知方法参数说明：

```
(callRecordInfo: CallRecordInfo) => void
```


表 9-315 入参说明

参数名称	参数类型	参数说明
callRecordInfo	CallRecordInfo	通话记录信息

表 9-316 CallRecordInfo

参数名称	参数类型	参数说明
callInfo	CallInfo	呼叫信息
reason	CallEndReason	呼叫结束原因

表 9-317 CallInfo

参数名称	参数类型	参数说明
mediaType	MediaType	媒体类型
number	string	对方号码
nickName	string	对方名称
startTime	string	呼叫开始时间
endTime	string	呼叫结束时间（通话结束前为空）
isCallOut	boolean	是否为主动呼出

表 9-318 MediaType 枚举说明

枚举名称	枚举值	枚举说明
HWM_MEDIA_TYPE_AUDIO	0	音频类型
HWM_MEDIA_TYPE_VIDEO	1	视频类型

表 9-319 CallEndReason 枚举说明

枚举名称	枚举值	枚举说明
HWM_CALL_END_REASON_CONNECTED_HANGUP_BY_CALLER	1	主叫挂断（接通）

枚举名称	枚举值	枚举说明
HWM_CALL_END_REASON_CONNECTED_HANGUP_BY_CALLEE	2	被叫挂断（接通）
HWM_CALL_END_REASON_HANGUP_BY_CALLER	3	主叫挂断（未接通）
HWM_CALL_END_REASON_HANGUP_BY_CALLEE	4	被叫挂断（未接通）
HWM_CALL_END_REASON_CALLEE_BUSY	5	被叫通话中（未接通）
HWM_CALL_END_REASON_CALLEE_ANSWER_TIMEOUT	6	被叫超时未接听（未接通）
HWM_CALL_END_REASON_CALLEE_NUMBER_ERROR	7	被叫号码错误（未接通）
HWM_CALL_END_REASON_FAILED	8	通话失败

9.5.7.3 发起呼叫

接口功能

该接口用于通过呼叫号码或者呼叫第三方账号发起呼叫。注意事项：1、在登录状态下调用该接口。

函数原型

```
startCall(startCallInfo: StartCallInfo, onStartCallResult?: (ret: SDKERR, reason: string) => void): void
```

请求参数

表 9-320 入参说明

参数名称	是否必选	参数类型	参数说明
startCallInfo	Y	StartCallInfo	发起呼叫参数信息
onStartCallResult	Y	(ret: SDKERR, reason: string) => void	发起呼叫回调

表 9-321 StartCallInfo

参数名称	是否必选	参数类型	参数说明
callerInfo	Y	CallerInfo	主叫信息
calleeInfo	Y	CalleeInfo	被叫信息
mediaType	Y	MediaType	媒体类型

表 9-322 CallerInfo

参数名称	是否必选	参数类型	参数说明
nickName	N	string	主叫姓名

表 9-323 CalleeInfo

参数名称	是否必选	参数类型	参数说明
nickName	N	string	被叫姓名
number	Y	string	被叫号码，如果填分配给帐号的sip号码（如+99111244216210249）则呼叫该帐号的App；如果填PSTN号码（如18700000000），则通过VoIP网关呼叫该号码，前提是该企业已开通PSTN呼叫权限。与thirdUserId二选一（优先级：number>thirdUserId），帐号密码鉴权时使用。
thirdUserId	Y	string	被叫三方帐号，与number二选一（优先级：number>thirdUserId），appl鉴权时使用

表 9-324 MediaType 枚举说明

枚举名称	枚举值	枚举说明
HWM_MEDIA_TYPE_AUDIO	0	音频类型
HWM_MEDIA_TYPE_VIDEO	1	视频类型

返回值

void

回调方法参数说明:

(ret: SDKERR, reason: string) => void

表 9-325 入参说明

参数名称	参数类型	参数说明
ret	SDKERR 请参考 错误码参考	错误码
reason	string	错误描述

9.6 错误码

错误码参考

表 9-326 SDKError 枚举说明

变量	值	描述
SDKERR_UNKOWN	-1	未知错误
SDKERR_SUCCESS	0	成功
CMS_CONF_BAD_REQUEST	111070001	请求错误
CMS_CONF_INSUFFICIENT_PERMISSIONS	111070002	权限不足
CMS_CONF_EXCEPTION	111070003	系统处理失败
CMS_CONF_AUTH_FAIL	111070004	校验失败
CMS_CONF_DATA_NOT_FOUND	111070005	会议不存在
CMS_VMR_DATA_NOT_FOUND	111070006	个人云会议室不存在
CMS_ACCESS_NUMBER_NOT_FOUND	111070007	会议接入号不存在
CMS_CONF_NOTICE_MODEL_CONFIG_ERROR	111070008	会议通知模板配置错误

变量	值	描述
CMS_CONF_AUTHEN_NEED_PASSWORD	111070009	会议密码不能为空
CMS_CONF_ROLE_AUTHENTICATION_FAILED	111070010	权限不足,无法查看会议
CMS_CONF_GET_STARTED_CONF_EXCEPTION	111070100	获取召开会议异常
CMS_CONF_START_EXCEPTION	111070101	开始会议异常
CMS_CONF_APPLY_IVR_RESOURCE_EXCEPTION	111070102	申请IVR资源异常
CMS_START_IVRCONF_EXCEPTION	111070103	开始IVR会议异常
CMS_CONF_APPLY_PARTICIPANT_RESOURCE_EXCEPTION	111070104	申请会场资源异常
CMS_START_SUBCONF_EXCEPTION	111070105	开始子会议异常
CMS_CONF_ADD_CHANNEL_EXCEPTION	111070106	创建级联通道异常
CMS_CONF_MOVE_AND_APPLY_RESOURCE_EXCEPTION	111070107	移动会场申请资源异常
CMS_START_CONF_EXCEPTION	111070108	开启会议异常
CMS_CONF_GET_STARTED_VMRCONF_EXCEPTION	111070109	获取召开云会议室会议异常
CMS_CONF_INVITE_PARTICIPANT_EXCEPTION	111070110	邀请会场异常
CMS_REQUEST_TOKEN_IS_NULL	111070111	请求token为空
CMS_CONF_RENAME_PARTICIPANT_EXCEPTION	111070112	重命名会场异常
CMS_CONF_HANGUP_EXCEPTION	111070113	会场挂断异常

变量	值	描述
CMS_CONF_BULK_HANGUP_EXCEPTION	111070114	会场批量挂断异常
CMS_CONF_DELETE_ATTENDEE_EXCEPTION	111070115	删除与会者异常
CMS_CONF_BULK_DELETE_ATTENDEE_EXCEPTION	111070116	批量删除与会者异常
CMS_SCHEDULE_CYCLE_CONF_EXCEPTION	111070117	预定周期会议异常
CMS_MODIFY_CYCLE_CONF_EXCEPTION	111070118	修改周期会议异常
CMS_CREATE_CYCLE_CONF_EXCEPTION	111070119	创建周期会议异常
CMS_REGION_DATA_NOT_FOUND	111070120	区域信息获取失败
CMS_FORBIDDEN_SUBSCRIBE	111070121	禁止订阅
CMS_CANCEL_SUBSCRIBE_FAIL	111070122	取消订阅失败
CMS_CONF_SYS_PORTAL_SERVICE_EXCEPTION	111071001	Sysportal服务运行异常
CMS_CONF_EXECUTOR_RUN_EXCEPTION	111071002	Conf服务运行异常
CMS_CONF_ID_NOT_EXIST	111071003	创建会议失败，请稍后重试
CMS_VMR_ONLY_ALLOW_CHAIR_START_CONF	111071004	VMR会议不允许来宾在主席入会前入会
CMS_VMR_CONF_CONFLICT	111071005	VMR会议时间冲突，该时间段此会议室已有预约会议
CMS_VMR_IMMEDIATELY_CONF_DURATION_NOT_ENOUGH	111071006	VMR会议时长不足
CMS_VMR_SCHEDULE_RELATION_NOT_EXIST	111071007	VMR预约关系不存在
CMS_CONF_QUERY_LIST_FAIL	111071008	查询会议列表失败

变量	值	描述
CMS_CONF_MODIFY_IMMEDIATELY_VMR_FAIL	111071009	不支持修改VMR会议为立即会议
CMS_CONF_MODIFY_VMR_FLAG_FAIL	111071010	不支持修改云会议室标识
CMS_VMR_CONF_NOT_REACH_PREPARE_TIME	111071011	VMR会议未到准备时间点
CMS_CONF_NOT_VMR	111071012	非VMR会议
CMS_CONF_START_TIME_LESS_THAN_CURRENT_TIME	111071013	会议开始时间不能早于系统当前时间
CMS_CONF_START_TIME_OVER_MAX_SCHEDULE_TIME	111071014	会议开始时间超出最大预订时间范围
CMS_CONF_DURATION_OVER_RANGE	111071015	时长超出范围
CMS_CONF_PARAMS_NULL_EXCEPTION	111071016	取消周期会议ID列表不能为空
CMS_CONF_SIZE_OVER_RANGE	111071017	与会方数超出范围
CMS_CONF_MEDIA_TYPES_ILLEGAL	111071018	媒体类型非法
CMS_CONF_CONFERENCE_TYPE_ILLEGAL	111071019	会议类型非法
CMS_CONF_CYCLE_PARAMS_NULL_EXCEPTION	111071020	周期会议时,必须携带周期会议参数
CMS_CONF_LENGTH_OVER_RANGE	111071021	会议时长超出范围
CMS_CONF_SUBJECT_LENGTH_TOO_LONG	111071022	会议主题超长
CMS_CONF_GROUPURI_LENGTH_TOO_LONG	111071023	groupuri超长
CMS_CONF_EMAIL_ILLEGAL	111071024	邮箱地址不合法
CMS_CONF_SMS_ILLEGAL	111071025	手机号不合法

变量	值	描述
CMS_CONF_DEPTNAME_TOO_LONG	111071026	部门名称过长
CMS_CONF_WELCOME_VOICE_NOT_IN_RANGE	111071027	入会欢迎词开关不在范围内
CMS_CONF_ENTER_PROMPT_NOT_IN_RANGE	111071028	入会提示音开关不在范围内
CMS_CONF_LEAVE_PROMPT_NOT_IN_RANGE	111071029	离会提示音开关不在范围内
CMS_CONF_REMINDERS_ILLEGAL	111071030	会议提醒非法
CMS_CONF_ALLOW_RECORD_NOT_IN_RANGE	111071031	会议是否启用网络录制取值非法
CMS_CONF_AUTO_RECORD_NOT_IN_RANGE	111071032	会议是否自动启动录制取值非法
CMS_CONF_USER_TYPE_NOT_IN_RANGE	111071033	会预定者的用户类型非法
CMS_CONF_ENCRYPT_MODE_NOT_IN_RANGE	111071034	加密类型非法
CMS_CONF_LANGUAGE_NOT_IN_RANGE	111071035	语言非法
CMS_CONF_AUTO_PROLONG_NOT_IN_RANGE	111071036	是否自动延长会议取值非法
CMS_CONF_VMRID_TOO_LONG	111071037	vmrid超长
CMS_CONF_RECORD_TYPE_NOT_IN_RANGE	111071038	录播类型非法
CMS_CONF_LIVE_ADDRESS_TOO_LONG	111071039	主流直播地址超长
CMS_CONF_AUX_ADDRESS_TOO_LONG	111071040	辅流直播地址超长
CMS_CONF_CYCLE_PARAMS_STARTDATE_NULL	111071041	周期会议时,起始日期为空

变量	值	描述
CMS_CONF_CYCLE_PARAMETERS_ENDDATE_NULL	111071042	周期会议时,结束日期为空
CMS_CONF_CYCLE_PARAMETERS_CYCLE_ILLEGAL	111071043	周期会议时,周期类型非法
CMS_CONF_CYCLE_PARAMETERS_INTERVAL_NOT_IN_RANGE	111071044	周期会议时,interval非法
CMS_CONF_CYCLE_PARAMETERS_POINT_NULL_EXCEPTION	111071045	周期会议时,point为空
CMS_CONF_CYCLE_PARAMETERS_POINT_NOT_IN_RANGE	111071046	周期会议时,point非法
CMS_CONF_VMR_FLAG_NOT_IN_RANGE	111071047	vmrFlag非法
CMS_CONF_RECORD_AUX_STREAM_NOT_IN_RANGE	111071048	recordAuxStream非法
CMS_CONF_MULTI_STREAM_FLAG_NOT_IN_RANGE	111071049	multiStreamFlag非法
CMS_CONF_AUDIENCE_FLAG_NOT_IN_RANGE	111071050	audienceFlag非法
CMS_CONF_LIVE_CHANNEL_ID_NOT_EXISTED	111071051	liveChannelID不存在
CMS_CONF_ANONYMOUS_LOGIN_LOCKED	111071052	匿名登录锁定,请稍后重试
CMS_CONF_LINK_LOGIN_LOCKED	111071053	链接登录锁定,请稍后重试
CMS_CONF_IDENTIFIER_NOT_ENOUGH	111071054	会议ID不足
CMS_CONF_IDENTIFIER_APPLY_FAILED	111071055	会议ID申请失败
CMS_CONF_IDENTIFIER_RELEASE_FAILED	111071056	会议ID释放失败
CMS_CONF_LIVE_ROOM_ISUSED	111071057	直播房间已被占用

变量	值	描述
CMS_NOT_ALLOWED_GUEST_FIRST_JOIN	111071058	当前会议还未召开,请耐心等待
CMS_CONF_SIZE_OVER_COMMUNAL_VMR_MAXNUM	111071059	与会方数超出上限
CMS_ORG_CONCURRENT_PARTICIPANTS_NUMBER_IS_ZERO	111071060	企业未配置并发会议资源,无法使用个人会议ID和随机会议ID发起会议
CMS_PARAMETER_VERIFIED_FAIL	111071061	参数校验失败
CMS_JSON_CONVERSION_FAILED	111071062	JSON转换失败
CMS_CONF_USER_NOT_VMR_OWNER	111071063	非云会议室拥有者
CMS_VMR_CONF_CONFLICT_AS_VMR_ALREADY_STARTED	111071064	VMR会议冲突,当前已有会议正在召开
CMS_CONF_MODIFY_FAIL_AS_CONF_ALREADY_STARTED	111071065	正在召开会议,无法编辑会议详情
CMS_CONF_ATTENDEES_NUMBER_EXCEED_LIMIT	111071066	与会方数量超出上限
CMS_CONF_CANCEL_FAIL_AS_CONF_STARTED	111071067	会议正在召开中,不允许取消会议
CMS_CONF_PWD_ENCRYPT_FAILED	111071068	会议密码加密失败
CMS_CONF_NOCONF_IN_SELECTED_TIME	111071069	选定的时间中没有会议
CMS_SUB_CYCCONF_CANNOT_MODIFY_LIVE_ROOM	111071070	周期会议子会议不允许修改直播房间
CMS_STOP_ONLINE_CONF_EXCEPTION	111071071	结束会议失败
CMS_CONF_CLOSED_RECENTLY	111071072	会议已经结束
CMS_CONF_NOT_ALLOWED_ANONYMOUS	111071073	会议不允许匿名用户呼入

变量	值	描述
CMS_CONF_ONLY_ALLOW_USER_IN_ORG	111071074	会议只允许企业内用户呼入
CMS_CONF_ONLY_ALLOW_USER_INVITED	111071075	会议只允许被邀请用户呼入
CMS_BATCH_EXPORT_HIS_CONF_LIST_CONFLICT	111071076	当前有正在执行的批量导出任务
CMS_CONF_ONLY_ALLOW_USER_INVITED_AND_USER_IN_ORG	111071077	会议只允许被邀请或者企业内用户呼入
CMS_WELINK_C_CURRENT_PARTICIPANTS_NUMBER_IS_NOT_ENOUGH	111071078	企业单会议并发方数已达上限
CMS_CONF_CTD_PARAMETER_ILLEGAL	111071079	ctd会议参数不合法
CMS_GUEST_PWD_ILLEGAL	111071080	来宾密码不合法
CMS_GUEST_PWD_CONFLICT_WITH_CONF_ID	111071081	来宾密码与会议ID冲突
CMS_CONF_NUM_UP_TO_LIMIT	111071082	会议数达到上限
CMS_CONF_RANDOM_INVALID	111071083	无效的会议随机数
CMS_EXPORT_HIS_CONF_UP_TO_LIMIT	111071084	导出历史会议次数超出允许的值
CMS_MULTI_PIC_TOO_MANY_LAYOUTS	111071085	会议多画面布局超出限制
CMS_MULTI_PIC_MANUAL_TYPE_INVALID	111071086	会议多画面模式非法
CMS_MULTI_PIC_TOO_MANY_SUBPICS	111071087	会议多画面子画面个数超出限制
CMS_MULTI_PIC_TOO_MANY_SUBNUMBERS	111071088	会议多画面子画面会场数超出限制
CMS_WEBINAR_COMMON_NON_CONVERT	111071089	网络研讨会不能修改为普通会议,普通会议也不能修改为网络研讨会
CMS_NOT_ALLOW_AUDIENCE	111071090	不支持添加观众

变量	值	描述
CMS_CONF_PWD_AUTH_FAILED	111071091	密码错误
CMS_FILE_EXPORT_TASK_FAILED	111071092	导出文件任务失败
CMS_FILE_EXPORT_TASK_DOWNLOADING	111071093	导出文件正在处理中
CMS_CONF_FORBID_AUDIENCE_JOIN	111071094	会议不允许观众入会
CMS_CONF_NEED_GET_WEBINAR_CONF_DTO	111071095	需要获取WEBINAR_CONF_DTO对象
CMS_CUSTOM_LANGUAGE_NUM_OVER_RANGE	111071096	自定义语言个数超出范围
CMS_ORG_NOT_SUPPORT_SIMULTANEOUS_INTERPRETATION	111071097	企业不支持同声传译
CMS_NEED_SET_INTERPRETER_GROUPS	111071098	需要设置传译组信息
CMS_INTERPRETER_LANGUAGE_ILLEGAL	111071099	传译组语言非法
CMS_CONF_LANGUAGE_NUM_UP_TO_MAX	111071100	单会议最大支持16种语言
CMS_INTERPRETER_GROUP_ID_REPEATED	111071101	传译组的组号重复
CMS_INTERPRETER_NUM_IN_ONE_GROUP_UP_TO_MAX	111071102	传译组中传译员的个数达到上限
CMS_INTERPRETER_REPEATED	111071103	传译员重复
CMS_INVALID_INTERPRETER	111071104	传译员帐号不存在
CMS_GUEST_PWD_CONFLICT_WITH_CHAIR_PWD	111071105	来宾密码与主席密码冲突
CMS_ONLINE_CONF_IN_COMMUNAL	111071106	云会议室已有会议正在召开
CMS_AUDIENCE_PWD_ILLEGAL	111071107	观众密码不合法

变量	值	描述
CMS_AUDIENCE_PWD_CONFLICT_WITH_CONF_ID	111071108	观众密码与会议ID冲突
CMS_AUDIENCE_PWD_CONFLICT_WITH_CHAIR_PWD	111071109	观众密码与主席密码冲突
CMS_ONLY_RTC_CONF_SUPPORT_E2E_ENCRYPT	111071110	非RTC会议不支持端到端加密
CMS_AUDIENCE_PWD_CONFLICT_WITH_GUEST_PWD	111071111	观众密码与来宾密码冲突
CMS_VMRIDTYPE_CANNOT_MODIFY	111071112	不支持修改会议ID类型
CMS_VMRID_CANNOT_MODIFY	111071113	不支持修改VMR类型
CMS_VMR_FLAG_CANNOT_MODIFY	111071114	不支持修改VMR标识
CMS_SUB_CYCONF_MODIFY_TIME_OUT_OF_RANGE	111071119	周期会议修改时间超出范围
CMS_CONF_APPLY_RESOURCE_FAILED	111072001	申请资源失败
CMS_CONF_RELEASE_RESOURCE_FAILED	111072002	释放资源失败
CMS_CONF_IS_CLOSING	111072003	会议正在结束
CMS_CONF_IS_STARTING	111072004	会议正在开始
CMS_CONF_NOT_STARTED	111072005	会议还未开始
CMS_CONF_IS_CLOSED	111072006	会议已经结束
CMS_CONF_INVITE_PARTICIPANT_FAILED	111072007	邀请会场失败
CMS_CONF_PARTICIPANT_NOT_EXIST	111072008	会场不存在
CMS_CONF_PARTICIPANT_HANG_UP_FAILED	111072009	挂断会场失败

变量	值	描述
CMS_CONF_NOT_SUPPORT	111072010	功能暂未支持,敬请期待...
CMS_CONF_STATUS_ILLEGAL	111072011	会议状态非法
CMS_CONF_GENERIC_CTL_PARTICIPANT_FAILED	111072012	会场会控操作失败
CMS_VMR_CURRENT_CONF_NOT_GOAL_CONF	111072013	当前VMR会议非目标会议
CMS_VMR_CONF_IS_EXPIRED	111072014	VMR会议已过期
CMS_VMR_CONF_IS_NOT_ARRIVE_START_TIME	111072015	VMR会议还未开始
CMS_VMR_CONF_WITH_SCHEDULE_NOT_EXIST	111072016	VMR预约记录不存在
CMS_CONF_HAS_NO_ATTENDEE	111072017	会议与会者不存在
CMS_CONF_CHAIRMAN_PWD_EXCEPTION	111072018	申请主席密码错误
CMS_CONF_NOT_FIND_CANMOVESITE_EXCEPTION	111072019	没有找到可以移动的会场
CMS_CONF_NOT_SUPPORT_RECORD	111072020	非录播会议不支持录制
CMS_RECORD_RESOURCE_NOT_ENOUGH	111072021	录播调度失败
CMS_RECORD_SCHEDULE_FAIL	111072022	录播资源不足
CMS_CONF_NOT_HAVE_PART	111072023	会议中没有与会人
CMS_CONF_NO_PARTICIPANT_JOINED	111072024	请确保会议中已有会场入会
CMS_AUDIO_CONF_NOT_SUPPORT	111072025	语音会议不支持该功能
CMS_AUDIO_SITE_NOT_SUPPORT	111072026	语音会场不支持该操作

变量	值	描述
CMS_AUDIO_SITE_IN_MULTI_PIC	111072027	手动设置的多画面中无法显示语音会场
CMS_NO_SITE_IN_MULTI_PIC	111072028	手动设置的多画面中需最少设置一个会场
CMS_ASSISTANT_PIC_NUM_MORE_THAN_ONE	111072029	手工设置的多画面最多只允许一个辅流子画面
CMS_VAS_SUB_PIC_CO NTAINS_USER_MORE_THAN_ONE	111072030	声控多画面模式下,每个子画面最多只允许一个用户
CMS_SITE_CALL_NUMBER_IS_EMPTY	111072031	与会者的入会号码不允许为空
CMS_SITE_CALL_NUMBER_LENGTH_TOO_LONG	111072032	与会者的入会号码长度超过系统允许的最大值
CMS_SITE_CALL_NUMBER_ILLEGAL_URI	111072033	与会者的入会号码格式非法(支持SIP、TEL号码格式)
CMS_SITE_NAME_LENGTH_TOO_LONG	111072034	与会者的名称长度超过系统允许的最大值
CMS_SITE_ROLE_NOT_IN_RANGE	111072035	与会者的角色类型不在系统允许的取值范围内
CMS_SITE_EMAIL_LENGTH_TOO_LONG	111072036	与会者的邮箱地址长度超过系统允许的最大值
CMS_SITE_SMS_LENGTH_TOO_LONG	111072037	与会者的手机号码长度超过系统允许的最大值
CMS_SITE_MUTE_NOT_IN_RANGE	111072038	与会者入会后是否静音的取值不在系统允许的取值范围内
CMS_SITE_AUTO_INVITE_NOT_IN_RANGE	111072039	与会者是否自动邀请的取值不在系统允许的取值范围内
CMS_VIEW_TYPE_NOT_IN_RANGE	111072040	主席选看类型取值不在系统允许的取值范围内
CMS_SET_MULTI_PIC_MANUAL_SET_NOT_IN_RANGE	111072041	设置多画面方式取值不在系统允许的取值范围内
CMS_SWITCH_TIME_NOT_IN_RANGE	111072042	轮询显示时间取值不在系统允许的取值范围内
CMS_SET_MULTI_PIC_IMAGE_TYPE_IS_EMPTY	111072043	画面类型不允许为空

变量	值	描述
CMS_SET_MULTI_PICTURE_TYPE_IS_NOT_MATCH	111072044	画面类型非法
CMS_ON_OFF_NOT_IN_RANGE	111072045	开关项取值不在系统允许的取值范围内
CMS_SITE_NAME_IS_EMPTY	111072046	与会者的名称不允许为空
CMS_SITE_NEW_NAME_LENGTH_TOO_LONG	111072047	与会者的名称超过系统允许的最大值
CMS_MCU_NOT_SUPPORT_SET_MULTI_PICTURE	111072048	不支持多画面,请联系管理员开启MCU配置
CMS_RESPONSE_TIME_OUT	111072049	等待响应超时
CMS_CONF_LOCKED	111072050	会议被锁定
CMS_CONF_MODE_IS_EMPTY	111072051	会议显示模式为空
CMS_CONF_MODE_NOT_IN_RANGE	111072052	会议显示模式不在系统允许的取值范围内
CMS_CONF_PICTURE_TYPE_NOT_IN_RANGE	111072053	会议画面类型不在系统允许的取值范围内
CMS_TEXT_TYPE_IS_EMPTY	111072054	横幅或字幕类型为空
CMS_TEXT_IS_EMPTY	111072055	横幅或字幕的内容为空
CMS_TEXT_IS_TOO_LONG	111072056	横幅或字幕的内容长度超过2000字节
CMS_REQUEST_BODY_IS_NULL	111072057	请求消息体为空
CMS_SUB_PICTURE_INDEX_MORE_THAN_PICTURE_NUMBER	111072058	子画面索引大于画面总数
CMS_CONF_NO_CHAIR_SITE	111072059	会议中没有主席
CMS_NORMAL_SITE_NOT_SUPPORT_VIEW	111072060	当前普通与会方类型不支持该操作
CMS_CHAIR_POLLING_FORBIDDEN_VIEW	111072061	前主席正在轮询观看会场,不能单独选看其他会场或会议多画面
CMS_SVC_SITE_NOT_SUPPORT_VIEW	111072062	当前主席与会方类型不支持该操作

变量	值	描述
CMS_RSE_RESREAPPLY	111072063	录制资源重新申请错误码
CMS_CONF_CTR_LOGI N_LOCKED	111072064	会控登录锁定
CMS_CONF_NOT_FOU ND_OR_AUTH_FAILED	111072065	会议不存在或密码错误
CMS_ORG_NOT_BUY_R ECSPACE	111072066	企业未购买录播空间
CMS_ORG_RECSPACE_ NOT_ENOUGH	111072067	企业录播空间已满
CMS_CONF_SETLIVE_F AILED	111072068	启动直播失败
CMS_CONF_CHAIR_FO RBIDDEN_HAND	111072069	不允许指定会议主席举手发言
CMS_GUEST_NO_PER MISSION_OPERATION	111072070	来宾不允许进行该操作
CMS_NO_PERMISSION _OPERATE_OTHERS	111072071	仅能对自己进行操作
CMS_NO_CONF_CTL_C AP_NO_CHAIR	111072072	没有会控能力的与会者不允许设置为主席
CMS_NOTIFY_MSG_IS_ EXPIRED	111072073	推送消息已过期
CMS_TRUNK_APPLY_C HAIR_FORBID	111072074	云下会场不能成为主席
CMS_CONF_CLOSE_HU NG_UP	111072075	会议结束挂断会场
CMS_CHAIR_HUNG_UP	111072076	主席挂断会场
CMS_CHAIR_DEL_SITE_ HUNG_UP	111072077	主席删除会议挂断会场
CMS_RECREATE_CONF _HUNG_UP	111072078	重建会议挂断会场
CMS_MOVE_SITE_FAIL_ HUNG_UP	111072079	移动会场失败挂断会场
CMS_CROSS_REGION_I NVITE_SITE_FAIL_HUN G_UP	111072080	跨region邀请会场失败挂断会场
CMS_INVITE_EXCEPTIO N_HUNG_UP	111072081	邀请会场失败挂断会场

变量	值	描述
CMS_REPEAT_JOIN_HUNG_UP	111072082	会场重复入会挂断会场
CMS_LOCK_SHARE_FAILED	111072083	锁定共享失败
CMS_TRUNK_SET_ONE_MULTIPIC_FORBID	111072084	云下会场禁止设置在多画面一画面中
CMS_CONF_NOT_SUPPORT_REALTIME_SUBTITLE	111072085	会议不支持实时字幕
CMS_CONF_REALTIME_SUBTITLE_PID_NULL	111072086	会议实时字幕指定发言人列表为空
CMS_CONF_REALTIME_SUBTITLE_PID_WRONG	111072087	会议实时字幕指定发言人ID有误或已离会
CMS_CONF_REALTIME_SUBTITLE_SET_FAIL	111072088	会议实时字幕设置失败
CMS_CONF_NOT_ALLOW_CHANGE_AUDIENCE_SPEAK_STATUS	111072089	不允许观众解闭音
CMS_CONF_SPEAKABLE_ONLY_FOR_AUDIENCE	111072090	主持人允许/禁止观众说话接口只能操作观众
CMS_CONF_NOT_SUPPORT_SIMULTANEOUS_INTERPRETATION	111072091	会议不支持同声传译
CMS_CONF_INTERPRETER_GROUP_IS_EMPTY	111072092	会议当前传译组为空,请先配置传译组
CMS_CONF_NOT_ENABLE_SIMULTANEOUS_INTERPRETATION	111072093	会议未开启同声传译
CMS_PARTICIPANT_IS_NOT_INTERPRETER	111072094	当前会场非传译员
CMS_PARTICIPANT_NOT_CONFIRM_INTERPRETER	111072095	传译员未确认身份
CMS_PARTICIPANT_LANGUAGE_ILLEGAL	111072096	会场选择的语言非法
CMS_PARTICIPANT_IS_CONFIRMED_INTERPRETER	111072097	当前会场是已确认的传译员

变量	值	描述
CMS_PARTICIPANT_LISTEN_NOT_EQUAL_SPEAK	111072098	普通会场听说频道不一致
CMS_SET_CONF_LANGUAGE_CHANNEL_LIST_FAILED	111072099	设置会议语言频道失败
CMS_SET_PARTICIPANT_LANGUAGE_CHANNEL_FAILED	111072100	设置会场语言频道失败
CMS_OVER_MAX_PARTICIPANT_NUMBER	111072101	超过设置的会议方数
CMS_OVER_MAX_ALLOWED_HAND_NUMBER	111072102	观众超过最大允许举手数
CMS_OVER_MAX_ALLOWED_SPEAK_NUMBER	111072103	观众超过最大允许发言数
CMS_SET_VIRTUAL_AUDIO_CSD_SITE_FAILED	111072104	设置级联通道虚拟语言会场失败
CMS_CONF_NOT_SUPPORT_COHOST	111072105	该会议未启用联席主持人
CMS_PARTICIPANT_NOT_SUPPORT_COHOST	111072106	该会场不支持联席主持人
CMS_CHAIR_CANNOT_BE_COHOST	111072107	不支持设置主持人为联席主持人
CMS_AUDIENCE_CANNOT_BE_COHOST	111072108	不支持设置观众为联席主持人
CMS_NEED_CHECK_COHOST_PERMISSION	111072109	需要去主会议所在region判断是否为联席主持人
CMS_COHOST_CANNOT_OPERATE_CHAIR	111072110	联席主持人不能操作主持人
CMS_PARTICIPANT_NOT_SUPPORT_INVITE_SHARE	111072111	当前会场不支持邀请共享能力
CMS_HEART_BEAT_FAILING_UP	111072201	与终端心跳中断导致挂断会场
CMS_AUDIENCE_VMR_LOGIC_RESOURCE_NOT_ENOUGH	111072202	观众数超过VMR套餐中的观众最大并发资源数

变量	值	描述
CMS_PARTICIPANT_VMR_LOGIC_RESOURCE_NOT_ENOUGH	111072203	主持人与嘉宾超过VMR套餐中的最大并发资源数
CMS_STARTED_FAIL_WEBINAR_RESOURCE_NOT_ENOUGH	111072204	网络研讨会套餐已经有在线会议,被占用,激活失败
CMS_ONLY_RTC_CONF_SUPPORT_LOCAL_RECORD	111072205	非RTC会议不支持本地录制
CMS_E2E_PARAMETER_ILLEGAL	111072206	端到端加密参数非法
CMS_DISABLE_ROLE_SWITCH_OVER	111072207	网络研讨会不允许进行角色转换
CMS_TERMINAL_NOT_SUPPORT_LOCAL_RECORD	111072208	客户端不支持本地录制
CMS_CONF_IS_UNDER_AR_ASSIST	111072211	会议已经在进行AR协作
CMS_AUDIENCE_CANNOT_BE_CHAIRMAN	111072213	观众不能设置为主持人
CMS_RENAME_CONTAINS_SENSITIVWORDS	111072214	改名涉及敏感词错误
CMS_RECORD_START_EXCEPTION	111072222	启动云录制失败, 请联系管理员
CMS_RECORD_PAUSED_EXCEPTION	111072223	暂停云录制失败, 请联系管理员
CMS_RECORD_STOP_EXCEPTION	111072224	停止云录制失败, 请联系管理员
CMS_LIVE_START_EXCEPTION	111072225	启动直播失败, 请联系管理员
CMS_LIVE_STOP_EXCEPTION	111072226	停止直播失败, 请联系管理员
CMS_MMR_PARTICIPANT_REACH_MAX_NUM	111072504	达到系统单个会议人数上限
CMS_USER_DATA_USER_INFO_NOT_EXIST	111073001	用户不存在
CMS_USER_DATA_ORG_INFO_NOT_EXIST	111073002	组织不存在

变量	值	描述
CMS_USER_DATA_USER_INFO_ILLEGAL	111073003	用户非法
CMS_USER_DATA_ORG_INFO_ILLEGAL	111073004	组织非法
CMS_USER_DATA_ORG_INFO_QUERY_FAILED	111073005	查询组织失败
CMS_ASSIGN_ANONYMOUS_FAILED	111073006	申请匿名用户失败
CMS_VMR_DATA_STATUSES_ILLEGAL	111073007	VMR套餐状态非法
CMS_RELEASE_ANONYMOUS_FAILED	111073008	释放匿名用户失败
CMS_WELINK_C_NOT_SUPPORT_CYCLE_CONF	111073009	WeLinkC 不支持预约周期会议
CMS_WELINK_C_NOT_SUPPORT_VMR_CONF	111073010	WeLinkC 不支持预约VMR会议
CMS_GET_WE_LINK_TOKEN_FAILED	111073011	获取WeLink Proxy Token失败
CMS_GET_M_TO_M_TOKEN_FAILED	111073012	获取机机帐号Token失败
CMS_WEBINAR_VMR_DATA_STATUS_ILLEGAL	111073013	网络研讨会VMR套餐状态非法
CMS_PROLONG_TIMES_OVER_LIMIT	111073014	会议延长次数超出VMR试用套餐可延长的最大次数
CMS_PROLONG_DURATION_OVER_LIMIT	111073015	会议延长时间超出VMR试用套餐单次会议最大时长
CMS_RFEE_VMR_NOT_SUPPORT_AUTO_PROLONG	111073016	VMR试用套餐不支持自动延长
CMS_IDO_CONF_CHAIR_EXIST	111074001	IDO主席已经存在
CMS_IDO_NOT_CONF_CHAIR	111074002	无IDO主席权限
CMS_IDO_OVER_CONF_MAX_DURATION	111074003	延长会议失败,会议已达到最大时长
CMS_IDO_SET_CONF_DURATION_FAIL	111074004	延长会议失败,服务处理异常

变量	值	描述
CMS_START_TIMER_FAIL	111075001	启动定时器失败
CMS_MODIFY_TIMER_FAIL	111075002	停止定时器失败
CMS_JSON_PARSE_FAIL	111075003	json转换异常
CMS_CONF_PORTAL_PARTICIPANT_HANGUP_BY_CHAIR_IDO	111076001	会场被终端主席挂断
CMS_CONF_PORTAL_PARTICIPANT_HANGUP_BY_CHAIR_REST	111076002	会场被portal主席挂断
CMS_CONF_PORTAL_PARTICIPANT_VMR_LOGIC_RESOURCE_NOT_ENOUGH	111076003	超出VMR最大方数
CMS_CONF_PORTAL_PARTICIPANT_RESOURCE_NOT_ENOUGH	111076004	企业并发数不足
CMS_CONF_PORTAL_PARTICIPANT_NO_RESPONSE	111076005	会场无响应
CMS_CONF_PORTAL_PARTICIPANT_NOT_EXIST	111076006	会场不存在
CMS_CONF_PORTAL_PARTICIPANT_UNAVAILABLE	111076007	会场离线
CMS_CONF_PORTAL_PARTICIPANT_BUSY	111076008	忙
CMS_CONF_PORTAL_PARTICIPANT_REFUSED	111076009	拒接
CMS_CONF_PORTAL_PARTICIPANT_LEAVE	111076010	会场挂断
CMS_CONF_PORTAL_PARTICIPANT_MEDIA_RESOURCE_NOT_ENOUGH	111076011	媒体资源不足
CMS_PSTN_SERVICE_UNAVAILABLE	111076012	PSTN服务未开通

变量	值	描述
CMS_CONF_APPLY_ADDRESS_TOKEN_FAILED	111076013	申请地址本鉴权信息失败
CMS_CONF_PORTAL_PARTICIPANT_TIMEOUT	111076014	SIP呼叫超时
CMS_CONF_PORTAL_OVER_MAX_PARTICIPANT_NUMBER	111076015	超过设置的会议方数
CMS_CONF_GUEST_INVITE_NOT_ALLOW	111076022	企业配置来宾不允许扫码邀请硬终端入会
CMS_CONF_NOT_ALLOWED_CONFCTRL	111076028	会议正在进行AR协作, 禁止共享/分组会议/释放主持人等会控操作
CMS_CONF_NOT_ALLOWED_AR_ASSIST	111076029	会议正在进行共享/分组会议, 禁止进行AR协作
CMS_CONF_PORTAL_PARTICIPANT_COMMON_ERROR	111076099	通常失败
CMS_RSE_SERVER_INFO_NOT_EXIST	111077001	RSE录播相关错误码
CMS_CONF_NOT_SUPPORT_AI_RECORD	111077101	会议不支持AI会议纪要
CMS_CONF_RE_APPLY_AI_RECORD_RES	111077102	正在重新申请会议纪要资源
CMS_AI_RECORD_RESOURCE_NOT_ENOUGH	111077103	AI会议纪要资源不足
CMS_CONF_IS_IN_AI_RECORD	111077104	会议处于AI会议纪要状态,不允许停止录制/直播
CMS_RECORD_FILE_NOT_EXIST_OR_HAVE_NO_RIGHT	111077201	录播文件已删除或您无权观看,请确认后重试
CMS_RSE_AUTH_FAILED	111077202	RSE机机鉴权失败
CMS_MULTI_REGION_NO_AVAILABLE_REGION	111078000	没有可用Region
CMS_MULTI_REGION_NO_AVAILABLE_GLOBAL	111078001	没有可用Global

变量	值	描述
CMS_MULTI_REGION_DISPATCH_REGION_ERROR	111078002	多region 分发region错误
CMS_DATA_PARAM_QUERY_XML_FAILED	111079000	大参数查询xml解析失败
CMS_ANONYMOUS_ASSIGN_XML_FAILED	111079001	匿名帐号分配xml解析失败
CMS_PARTICIPANT_BAD_REQUEST	111080001	请求消息参数错误
CMS_PARTICIPANT_INSUFFICIENT_PERMISSIONS	111080002	用户无权限进行该操作
CMS_PARTICIPANT_DATA_NOT_FOUND	111080004	会场不存在或已被删除
CMS_PARTICIPANT_IS_INVITING	111081001	会场正在通话中
CMS_PARTICIPANT_CONF_NOT_EXIST	111081002	会场所在会议不存在或已结束
CMS_PARTICIPANT_APPLY_RESOURCE_FAILED	111081003	申请会场资源失败
CMS_PARTICIPANT_RELEASE_RESOURCE_FAILED	111081004	释放会场资源失败
CMS_PARTICIPANT_APPLY_MEDIA_RESOURCE_FAILED	111081005	申请会场媒体资源失败
CMS_PARTICIPANT_AUTHENTICATION_FAILED	111081006	认证失败
CMS_PARTICIPANT_CONF_IS_LOCKED	111081007	会议已经被锁定
CMS_PARTICIPANT_CONF_LOGIN_LOCKED	111081009	会议鉴权失败次数已达上限被锁定
CMS_WAITING_USER_LIMIT	111081010	等候用户限制
CMS_PARTICIPANT_CONTROL_NOT_SUPPORT	111082001	不支持该会控
CMS_PARTICIPANT_NO_RESPONSE	111082002	无响应

变量	值	描述
CMS_PARTICIPANT_NOT_EXIST	111082003	会场不存在
CMS_PARTICIPANT_UNAVAILABLE	111082004	会场离线
CMS_PARTICIPANT_BUSY	111082005	会场正忙
CMS_PARTICIPANT_REFUSED	111082006	会场拒接
CMS_PARTICIPANT_LEAVE	111082007	会场挂断
CMS_PSTN_SERVICE_CLOSED	111082008	PSTN服务关闭
CMS_PARTICIPANT_INVITE_TIMEOUT	111082009	SIP呼叫超时
CMS_PARTICIPANT_RSE_NOT_EXIST	111083001	RSE设备不存在
CMS_PARTICIPANT_EXECUTE_DTMF_FAILED	111084001	执行DTMF失败
CMS_PARTICIPANT_GET_STARTED_CONF_EXCEPTION	111084002	启动会议异常
CMS_PARTICIPANT_CALL_MCUSIDE_EXCEPTION	111084003	呼叫MCU端异常
CMS_PARTICIPANT_JOIN_CALL_EXCEPTION	111084004	加入呼叫异常
CMS_PARTICIPANT_INVITE_EXCEPTION	111084005	邀请异常
CMS_PARTICIPANT_CALL_TERMINALSIDE_EXCEPTION	111084006	呼叫端侧异常
CMS_PARTICIPANT_ANSWER_EXCEPTION	111084007	应答异常
CMS_PARTICIPANT_GENERIC_CTL_EXCEPTION	111084008	通用CTL异常
CMS_PARTICIPANT_HANGUP_EXCEPTION	111084009	挂断异常

变量	值	描述
CMS_PARTICIPANT_INFORMATION_CAPABILITY_EXCEPTION	111084010	通知能力异常
CMS_PARTICIPANT_UPDATE_MT_EXCEPTION	111084011	更新MT异常
CMS_PARTICIPANT_NEGOTIATE_EXCEPTION	111084012	协商异常
CMS_PARTICIPANT_CONFIG_NOT_SUPPORT	111084013	不支持该会议
CMS_RTC_INVITE_NOT_ALLOWED_JOIN_CONF	111084014	RTC外邀入会,来宾不在被邀请名单中,拒绝入会
CMS_URI_JOIN_XML_FAILED	111089002	链接入会XML解析失败
CMS_DATA_JOIN_FAIL	111090001	数据入会失败
CMS_DATA_JOIN_OVERTIME	111090002	数据入会超时
CMS_AUTH_SLIDE_GET_IMAGE_FAILED	111100001	获取滑块图片失败
CMS_AUTH_SLIDE_VERIFY_CHECK_FAILED	111100002	滑块校验失败
CMS_AUTH_INVALID_RANDOM	111100003	无效的随机数
CMS_AUTH_INVALID_SLIDER_TOKEN	111100004	无效滑块token
CMS_AUTH_NEED_CHECK_SLIDER	111100005	需要滑块校验
CMS_AUTH_INVALID_SMS	111100006	非法的手机号
CMS_SEND_VERIFY_CODE_FAIL	111100007	发送验证码失败
CMS_AUTH_VERIFY_CODE_EXPIRED	111100008	验证码已过期
CMS_AUTH_VERIFY_CODE_FAIL	111100009	验证码错误
CMS_AUTH_VERIFY_CODE_FAIL_CNT_EXCEEDED	111100010	验证码错误超过最大次数

变量	值	描述
CMS_AUTH_INVALID_UUID	111100011	uuid无效,需要重新验证手机号
CMS_CONF_NOT_ON_LOCAL_STATION	111100012	会议不在本站点
CMS_AUTH_NEED_MOBILE_VERIFY	111100013	需要验证手机号, 海外接入国内, 免费企业需要进行手机验证码校验
CMS_IM_ADD_GROUP_FAILED	111110001	添加群组失败
CMS_IM_DISBAND_GROUP_FAILED	111110002	删除群组失败
CMS_IM_ADD_MEMBER_FAILED	111110003	添加成员失败
CMS_IM_REMOVE_MEMBER_FAILED	111110004	删除成员失败
CMS_WAITING_ROOM_ADD_MEMBER_FAILED	111112001	入等候室失败
CMS_WAITING_ROOM_MOVE_MEMBER_FAILED	111112002	出等候室失败
CMS_WAITING_ROOM_CACHE_OPT_FAILED	111112003	等候室缓存操作失败
CMS_NOT_ALLOW_COHOST_TO_WAITING_ROOM	111112004	联席主持人不能被移入等候室
CMS_NOT_ALLOW_ENABLE_WAITING_ROOM	111112005	不允许启用等候室
CMS_CLIENT_NOT_SUPPORT	111112008	客户端能力不支持此特性
CMS_APPLICATION_NOT_FOUND	111113001	找不到应用信息
CMS_APPLICATION_EXISTS	111113002	应用已存在, 目前一个会议只能关联一种应用
CMS_APPLICATION_CREATE_FORBIDDEN	111113003	无权限创建应用
CMS_APPLICATION_NEED_REAL_NAME	111113004	应用操作需要实名认证 (用户需要绑定手机)

变量	值	描述
CMS_APPLICATION_GET_USER_TOKEN_FAILED	111113005	获取应用用户token失败
CMS_APPLICATION_URL_INVALID	111113006	应用服务URL地址非法
CMS_APPLICATION_SHIMO_DOC_NOT_ENABLED	111113101	石墨文档应用未启用
CMS_APPLICATION_SHIMO_DOC_CALL_FAILED	111113102	调用石墨文档服务失败
CMS_APPLICATION_SHIMO_GET_TOKEN_FAILED	111113103	获取石墨应用级token失败
CMS_APPLICATION_SHIMO_NOT_IN_CONFERENCE	111113104	用户不在会议中，无权限申请权限
CMS_BREAKOUT_CONF_SUB_OVER_LIMIT	111500301	分组会议子会议超过上限
CMS_BREAKOUT_CONF_NAME_ERROR	111500302	分组会议子会议名称错误
CMS_BREAKOUT_CONF_START_FAILED	111500303	分组会议启动失败
CMS_BREAKOUT_CONF_NOT_EXIST_WHEN_DELETE	111500304	分组会议中没有子会议（删除）
CMS_BREAKOUT_CONF_NOT_EXIST_WHEN_START	111500305	分组会议中没有子会议（启动）
CMS_BREAKOUT_CONF_CLOSING	111500306	分组会议结束中(主持人停止)
CMS_BREAKOUT_CONF_END_OR_NOT_START	111500307	分组会议已经结束或未启动（加入时）
CMS_BREAKOUT_CONF_ATTENDEE_IN_TARGET_CONF	111500308	移动用户在目标分组
CMS_BREAKOUT_CONF_CHAIR_IN_CURRENT_CONF	111500309	主持人在当前分组中（请求帮助）

变量	值	描述
CMS_BREAKOUT_CONF_NOT_SUPPORT_BREAKOUT	111500310	不支持分组会议（启动时 启动接口）
CMS_BREAKOUT_CONF_STARTING_OR_STARTED	111500311	分组会议已经启动或启动中（重复启动）
CMS_BREAKOUT_CONF_NOT_EXIST	111500312	子会议不存在
CMS_BREAKOUT_CONF_NOT_SUPPORT_IN_MAIN_CONF	111500313	主会中用户不支持此操作（请求帮助）
CMS_BREAKOUT_CONF_STATE_ILLEGAL	111500314	分组会议状态异常（缓存误删）
CMS_BREAKOUT_CONF_ACTION_FORBIDDEN_BY_STATUS	111500315	分组会议静态、动态操作由于状态原因，不支持当前操作
CMS_BREAKOUT_CONF_APPLY_CHAIR_FAILED	111500316	主持人在分组中，不能申请为主持人
CMS_BREAKOUT_CONF_APPLY_CONF_ID_FAILED	111500317	申请分组的会议id失败
CMS_BREAKOUT_CONF_QUERY_MAIN_CONF_FAILED	111500318	获取主会议信息失败
CMS_BREAKOUT_CONF_ATTENDEE_NOT_FOUND	111500319	没有找到用户信息
CMS_BREAKOUT_CONF_STOP_FAILED	111500320	分组会议停止失败
CMS_SUBJECT_FAIL_TO_LEAVE_CONF_START	111502000	订阅失败需要离会的错误码起始值
CMS_SUBJECT_FAIL_TO_LEAVE_CONF_END	111502099	订阅失败需要离会的错误码结束值
CMS_FAILED	112010001	通用失败
CMS_NULL_PARAM	112010002	参数为空
CMS_PARTICIPANT_NOT_FOUND	112010003	未找到参与者

变量	值	描述
CMS_CONFERENCE_NOT_FOUND	112010004	会议不存在
CMS_SIP_INVITE_TIMEOUT_NO_RCVD	112010005	invite消息超时
CMS_SIP_INVITE_FAILED	112010006	invite消息失败
CMS_SIP_INFO_TIMEOUT	112010007	info响应超时
CMS_SIP_INFO_FAILED	112010008	info响应失败
CMS_WAIT_FIRST_CALL_FAILED	112010009	系统处理失败
CMS_PARTICIPANT_DIALOG_FAILED	112010010	系统处理失败
CMS_INVITE_MRS_FAILED	112010011	呼叫mrs失败
CMS_INVITE_PARTICIPANT_FAILED	112010012	呼叫终端失败
CMS_INFORM_CONF_INFO_FAILED	112010013	数据入会Info失败
CMS_START_CONF_FAILED	112010014	召开会议失败
CMS_MEDIA_NEGOTIATION_FAILURE	112010015	媒体协商失败
CMS_USER_REJECTED	112010016	用户拒绝
CMS_USER_BUSY	112010017	用户忙
CMS_NO_CALL_RIGHT	112010018	没有呼叫权限
CMS_UNKNOWN_CALLER	112010019	未知被叫人
CMS_NO_USER_RESPONDING	112010020	没有用户响应
CMS_JOIN_CONF_TIMEOUT	112010021	加入会议超时
CMS_SESSION_RELEASED	112010022	会话已释放
CMS_CALLEE_NOT_REGISTERED	112010023	被叫人未注册

变量	值	描述
CMS_SIP_INVITE_TIME OUT_RCV1XX	112010024	SIP邀请超时
CMS_USER_SESSION_T IMER_TIMEOUT_HANG UP	112010025	用户Session timer 超时挂断
CMS_RCV_USER_BYE_ MRS_HANGUP	112010026	与会者腿MRS侧bye挂断
CMS_RCV_USER_BYE_P ART_HANGUP	112010027	与会者腿终端侧bye挂断
CMS_NO_RTP_STREAM _HUNG_UP	112010028	无码流挂断
CMS_SYSPORTAL_EXCE PTION	112100000	系统异常
CMS_BAD_REQUEST	112100001	请求异常
CMS_ILLEGAL_REQUES T	112100002	非法请求
CMS_DATA_NOT_FOU ND	112100003	数据未获取到
CMS_DATA_DUPLICAT E	112100004	数据重复
CMS_APPLY_TYPE_NOT _SUPPORT	112100006	请求的类型不支持
CMS_SAVE_NOTICE_M ODEL_FAILED	112100007	保存通知模板失败
CMS_SAVE_DATA_FAIL ED	112100008	保存数据失败
CMS_QUERY_DATA_FAI LED	112100009	查询数据失败
CMS_DELETE_DATA_FA ILED	112100010	删除数据失败
CMS_NEED_RESET_PAS SWORD	112100011	需要重置密码
CMS_AUTH_FAILED	112100012	旧密码错误
CMS_NEW_PASSWORD _EQUALS_OLD_PASSW ORD	112100013	新密码不能与旧密码相同
CMS_PASSWORD_TOO _SIMPLE	112100014	密码复杂度不够,安全性低

变量	值	描述
CMS_LEHGTH_NOTMET_STANDARD	112100015	长度不符合要求
CMS_RSE_IS_IN_CONFERENCE	112100016	RSE在会议中
CMS_RSE_IS_DELETING	112100017	RSE待删除
CMS_RSE_LIST_CONF_FAILED	112100018	查询会议列表失败
CMS_AUTH_LOCKED	112100019	用户已被锁定,请稍后重新登录
CMS_LIVE_CHANNEL_NAME_EXISTED	112100022	频道名称已存在
CMS_CONFID_PREFIX_CANNOT_DECREASE	112100023	修改系统配置项失败,会议ID前缀重复
CMS_CONFID_PREFIX_DUPLICATE	112100024	CONFID前缀只能增加,不能修改和删除
CMS_LIVE_CHANNEL_URL_EXISTED	112100025	直播房间地址已存在
CMS_LIVE_ROOM_ISUSED	112100026	直播房间已被使用
CMS_LIVE_ROOM_NOT_EXIST	112100027	直播房间不存在
CMS_LIVE_ROOM_RELEASE_FAIL	112100028	直播房间释放失败
CMS_REQUEST_MISSING_HEADER_OR_PARAMETER	112100029	请求缺失头域
CMS_LIVE_MAIN_URL_EXISTED	112100030	音视频推流地址已存在
CMS_LIVE_AUX_URL_EXISTED	112100031	辅流演示推流地址已存在
CMS_PWD_IS_WEAK_PASSWORD	112100032	密码为弱密码
CMS_LIVE_ROOM_ISUSED_FAIL	112100033	直播房间被占用
CMS_SAVE_PART_ATTENDEES_SINCE_TOO_MANY	112100034	因为与会者数量超出上限只保存部分
CMS_EXPORT_DATA_FAIL	112100035	导出数据失败

变量	值	描述
CMS_IMPORT_DATA_FAIL	112100036	导入数据失败
CMS_FILE_EXCEEDS_MAXIMUM_SIZE	112100037	文件大小超出上限
CMS_LVS_TARGET_ADDRESS_EXISTED	112100038	lvs的目标地址已存在
CMS_LVS_PORT_EXISTED	112100039	lvs端口重复
CMS_NOT_ALLOWED_MODIFY_GLOBAL_STANDBY	112100040	Global只有主机可以修改配置;
CMS_MCU_INSULATE_ADDR_EXIST	112100041	mcu隔离的IP已经存在
CMS_MCU_INSULATE_ADDR_CANT_MODIFY	112100042	mcu隔离的IP不能修改
CMS_SBC_ADDR_EXIST	112100043	SBC配置的地址已经存在
CMS_INTERNAL_EXCEPTION	112100044	内部错误异常
CMS_SYS_CONFIG_DELETE_EXCEPTION	112100045	系统配置不允许被删除
CMS_NO_ISBC_ACCESS_ADDR	112100046	企业id无法获取关联SBC地址
CMS_MEDIA_TYPE_NEEDED_HD	112100047	企业已开启全高清功能,媒体设置不能低于1080P/8M
CMS_INTERPRETER_LANGUAGE_ABBREVIATION_REPEATED	112100061	语言缩写重复
CMS_INTERPRETER_LANGUAGE_NAME_REPEATED	112100062	语言名称重复
CMS_MODIFY_MCU_WARNING	112100063	修改MCU连接配置告警,以下两种场景会导致:1.MCU存在会议 2.修改类型为MIXED未配置隔离
CMS_ADD_MCU_WARNING	112100064	新增MCU连接配置告警,MCU类型为MIXED且未配置隔离
CMS_MODIFY_MCUINSULATE_WARNING	112100065	修改MCU隔离配置时,解除企业绑定时MCU类型为MIXED
CMS_DEL_MCUINSULATE_WARNING	112100066	删除MCU隔离配置时,MCU类型为MIXED

变量	值	描述
CMS_MCU_TYPE_INCOMPATIBLE	112100067	MCU与网关类型不兼容
CMS_MEDIARES_MEDIA_RESOURCE_NOT_ENOUGH	115021001	媒体资源不足
CMS_MEDIARES_CONF_NOT_EXIST	115021002	会议不存在
CMS_MEDIARES_MEDIA_RESOURCE_NOT_EXIST	115021003	媒体资源不存在
CMS_MEDIARES_BANDWIDTH_RESOURCE_NOT_ENOUGH	115021005	带宽资源不足
CMS_MEDIARES_DEVICE_FAULT	115021006	设备故障
CMS_LOGIC_RES_NOT_ENOUGH	115022000	通用的逻辑资源不足
CMS_LOGICRES_PNUM_RESOURCE_NOT_ENOUGH	115022001	企业会议并发方数不足
CMS_LOGICRES_PORT_RESOURCE_NOT_ENOUGH	115022002	逻辑资源端口资源不满足
CMS_LOGICRES_LSNUM_RESOURCE_NOT_ENOUGH	115022003	企业直播推流方数不足
CMS_MEDIARES_DEVICE_NOT_EXIST	115022004	媒体设备不存在
CMS_ID_RESOURCE_NOT_ENOUGH	115023001	ID资源不足
CMS_RMS_SERVICE_ERROR	115024001	RMS服务异常
CMS_LIVE_ROOM_IN_USE	115025001	直播房间已经被使用
CMS_LIVE_ROOM_NOT_EXISTED	115025002	直播房间不存在
CMS_LIVE_ROOM_RELEASE_FAILED	115025003	直播房间释放失败
CMS_MCU_HAS_CONF	115026001	MCU有会议

变量	值	描述
CMS_COMMON_INVALID_PARAMS	115029001	通用参数值不合法
CMS_COMMON_NULL_PARAMS	115029002	参数值为空
CMS_COMMON_SERVICE_INVALID	115029003	服务不可用
CMS_CONFIG_ERROR	115029004	配置异常
CMS_DUPLICATE_APPLY_RESOURCE	115029005	重复申请资源
CMS_WAIT_APPLY_RESOURCE	115029006	等待申请资源中
CMS_COMMON_UNKNOWN	115029999	未知错误
CMS_USER_AUTHENTICATION_FAILED	118000000	用户鉴权失败
CMS_USER_TOKEN_AUTH_FAILED	118000001	usg token 鉴权失败
CMS_USER_AUTHENTICATION_LOCKED	118000002	usg鉴权帐号锁定
USG_SERVER_BUSY	200000001	服务器正忙 请稍后再试
USG_INTERNAL_ERR	200000002	服务器正忙 请联系SRE
USG_PARAM_ERR	200000003	请求参数错误 请检查参数是否合法
USG_ILLEGAL_REQ	200000004	非法请求 请检查请求头域X-Auth-Token或X-Access-Token的值
USG_UPGRADING	200000005	系统正在升级，暂时不可访问。如对您造成不便，十分抱歉
USG_OTHER_CORP_OWNER	201020023	该用户已经是其他企业的所有者,无法为其创建新的企业
USG_CORP_USER_NUMBER_MAX	201040005	用户总数超过限制，请检查购买的会议用户
USG_PASSWORD_SECURITY_LOW	201040016	密码安全性低，请设置其他密码
USG_ACTIVE_CODE_OVERDUE	201040031	激活码过期
USG_PRE_VERIFY_EXPIRE	201040042	预验证失效

变量	值	描述
USG_NO_INVITE_OTHER_CORP_OWNER	201040043	该用户是其他企业的所有者,暂无法邀请
USG_INVITE_LINK_INVALID	201040044	邀请连接非法
USG_USER_IN_WAIT_LIST	201040045	用户已经在等待列表
USG_USER_EXIST	201040046	用户已存在
USG_NOT_ALLOW_MOD_ACCOUNT	201040047	不允许修改账户
USG_ACTIVE_CODE_NOT_EXIST	201050012	激活码不存在
USG_ACTIVE_INVALID_ACTIVE_CODE	201050015	激活码错误
USG_ACTIVE_FAIL	201050021	激活失败
USG_CAN_NOT_UNBOUND	201050022	终端不允许解绑
USG_VMR_RESOURCE_UNAVAILABLE	201080000	云会议资源不可用
USG_VMR_PWD_SAME_AS_CHAIRMAN_PWD	201080006	主席密码和来宾密码不能相同 请确认输入参数
USG_VMR_PWM_SAME_AS_MEETING_PWD	201080017	和会议密码相同
USG_INVITATION_ENABLE_INVALID	201510001	没有邀请权限去获取邀请码
USG_CORP_APPLICANT_INVALID	201510003	申请失效
USG_ERR_PASSWORD	206010000	用户名或密码错误
USG_APPID_ACCOUNT_DISABLED	206010007	帐号未激活,无权限登录
USG_SEND_SLIDER_FAILED	206010010	获取验证码失败 请联系SRE
USG_NEED_VERIFY_CODE	206010011	验证码不能为空
USG_CHECK_SLIDER_FAILED	206010012	验证码校验失败
USG_JUMP_LOGIN_FAIL	206010013	跳转登录失败 请联系SRE

变量	值	描述
USG_SSO_AUTH_FAILED	206010022	SSO登录失败 请联系SRE
USG_AUTH_NOT_SUPPORT_SSO	206010023	不支持sso登录
USG_INVALID_AUTH_TYPE	206010024	认证类型非法
USG_APPID_AUTH_FAILED	206010025	APPID校验失败
USG_AUTH_TYPE_ERR	206010027	校验类型错误
USG_NETWORK_ERROR	206010029	网络异常
USG_APPID_NOT_ALLOW_AUTO_REG	206010030	APPID不允许自动注册
USG_APPID_AUTO_REG_FAILED	206010031	APPID自动注册失败
USG_NORMAL_USER_CANNOT_LOGIN	206010036	普通用户无法登录
USG_HWACCOUNT_AUTH_FAILED	206010042	华为账户认证失败
USG_APPID_AUTH_EXPIRE	206010057	APPID认证过期
USG_APPID_ACCOUNT_NOT_BIND	206010058	APPID账户未绑定
USG_SITE_MISMATCHED	206010064	站点不匹配
USG_CONFLICT_PHONE	206010065	电话冲突
USG_CONFLICT_EMAIL	206010066	邮箱冲突
USG_USER_NUM_EXCEEDED	206010070	用户数超过上限
AUTH_WECHAT_LOGIN_FAILED	206010071	微信鉴权失败
USG_CHANGE_SITE_AND_NEXT_REQUEST	206010074	国际站点支持切站点重登
USG_ACCOUNT_IS_UNBOUND	206010075	账号已经解绑

变量	值	描述
USG_INVALID_TOKEN	206020000	鉴权信息无效 请检查请求头域X-Auth-Token或X-Access-Token的值
USG_TOKEN_IS_CREATED_BY_OTHERSITE	206020001	token是容灾站点创建
USG_TOKEN_IS_INVALID_BY_KICKOFF	206020002	token由于被踢失效
USG_USER_NOT_EXIST	206030000	系统中不存在该用户
USG_FIND_VERIFY_CODE_FAIL	206030001	验证码错误或失效, 请重新获取
USG_VERIFY_CODE_ERR	206030002	验证码错误
USG_NOT_PHONE_AND_EMAIL	206030003	发送验证码失败
USG_RESEND_NOT_ALLOWED	206030004	请在180秒后重新发送验证码
USG_NOT_SUPPORT_MODIFY_PWD	206030005	系统不支持修改密码
USG_OLD_PWD_WRONG	206030006	旧密码错误
USG_INVALID_NEW_PASSWORD_LEN	206030007	新密码长度不符合要求
USG_INVALID_PASSWORD_LEVEL	206030008	密码复杂度不满足要求
USG_CANNOT_SAME_WITH_HISTROY_PWD	206030009	设置的密码不能与最近旧密码相同
USG_REPEAT_CHARACTER_NUM_TOO_MUCH	206030010	密码不能包含3个以上重复字符
USG_PWD_TOO_FREQUENTLY	206030011	修改密码后, 5分钟内不能更新密码
USG_CANNOT_CONTAIN_ACCOUNT	206030012	密码不能与帐号或其逆序帐号相同
USG_MANY_SAME_WITH_OLD_PWD	206030013	新密码相较于旧密码至少要有两个不同的字符
USG_CONTAINS_INVALID	206030014	密码格式不正确
USG_USER_LOCKED	206030015	用户已被锁定, 请60秒后重试
USG_NO_RIGHT_TO_RESET_PWD	206030016	没有权限重置密码

变量	值	描述
USG_FROGET_PWD_O VER_10_TIMES	206030018	一天内忘记密码次数达到最大值（10次）
USG_USG_VERIFY_COD E_CHECK_EXCCED_MA X	206030019	验证码输入错误次数超过最大值，请重新获取
USG_SEND_VERIFY_CO DE_FAILED	206030023	发送验证码失败 请联系SRE
USG_AUTO_USER_NOT _ALLOW_MOD_PWD	206030028	系统已开启SSO登录,不支持修改密码 系统关闭SSO登录后重试
USG_HUAWEI_MODIFY _PWD	206030031	华为帐号忘记密码
USG_NO_CORRESPON D_TERMIAL_TYPE	206040000	无对应的终端类型
USG_SEND_VERIFY_CO DE_IS_LOCKED	206060008	发送校验码锁定
USG_NEED_SLIDER_VE RIFY	206060009	需要滑块校验
USG_SLIDER_TOKEN_I NVAILD	206060010	滑块token非法
USG_REQUEST_PARAM _ERROR	206060011	请求参数错误
USG_NOT_SUPPORT_S END_SMS	206060012	不支持发送短信
USG_NOT_ALLOW_REP EAT_SEND	206060013	不支持重复发送
USG_LOGINVERIFY_CO DE_INVAILD	206060014	验证码非法
USG_LOGINVERIFY_CO DE_EXCCED_MAX	206060015	验证码次数超过最大次数
USG_HWACCOUNT_BI ND_FAILED	206060017	华为账户绑定失败
USG_ACTIVE_NOT_IDE AHUB_CODE	211030046	不是ideahub激活码
USG_AUTH_FAIL_TOO MANY_TIMES_LOCKED	211030050	身份验证失败次数太多,已锁定
USG_AUTH_FAIL_TOO MANY_TIMES_PERMA NENTLY_LOCKED	211030052	身份验证失败次数太多,已永久锁定

变量	值	描述
USG_LOG_UPLOAD_TIMES_EXCEED	211060003	日志上传次数达到最大限制
SDK_COMMON_FAIL	411000001	失败
SDK_PARAM_ERROR	411000002	参数错误
SDK_CONF_NOT_EXIST	411000003	会议不存在
SDK_CONF_EXIST	411000004	已经有会议存在
SDK_CONF_STATE_ERROR	411000005	会议状态异常
SDK_CALL_GET_DEVICE_FAILED	411000006	获取设备异常
SDK_CALL_SET_DEVICE_FAILED	411000007	设置设备异常
SDK_NO_DEVICE_CAN_USED	411000008	无可用设备
SDK_ONLY_MOBILE_SUPPORT	411000009	只有移动端支持该操作
SDK_ACCESS_ERROR	411000010	网络接入错误
SDK_SERVICE_DOING	411000011	正在进行其他补充业务
SDK_ERR_MEM_ERROR	411000012	分配内存错误
SDK_ERR_INITIALIZE_FAILED	411000013	初始化失败
SDK_USER_DOES_NOT_EXIST	411000014	用户不存在
SDK_DATACONF_NOT_JOIN	411000015	未加入数据会议
SDK_CALL_AND_CONF_NOT_EXIST	411000016	会议和呼叫均不存在
SDK_SVN_DETECT_FAILED	411000017	svn端口探测不通
SDK_ERR_WRITE_FILE_FAILED	411000018	写文件失败
SDK_ERR_RTC_NOT_SUPPORT_STG	411000019	RTC不支持STG模式
SDK_ERR_RTC_LIB_LOAD_FAILED	411000020	rtc库加载失败，不支持rtc会议

变量	值	描述
SDK_ERR_SWITCH_ROLE_CONF_PAUSE	411000021	转角色的时候会议正处于观众暂停观看状态
SDK_CONF_NOT_IN_CALL	411000022	不在呼叫中
SDK_CONF_OR_CALL_NOT_EXIST	411000023	会议或者呼叫不存在
SDK_CONF_PAIR_STATE_NOT_ALLOW	411000024	配对状态不允许
SDK_CONF_PAIR_QUERY_FAIL	411000025	配对请求失败
SDK_CONF_PAIR_OVER_LIMIT	411000026	配对超出限制
SDK_CONF_PAIR_ERROR_COMMON	411000027	配对通用错误
SDK_CONF_PAIR_ERROR_FAILED	411000028	配对入会失败
SDK_CONF_PAIR_ERROR_BUSY	411000029	配对入会忙
SDK_CONF_PAIR_ERROR_INVALID	411000030	配对入会无效
SDK_CONF_PAIR_ERROR_MULTI	411000031	同时多次点击入会错误
SDK_CONF_PAIR_UNESTABLISHED_OR_PENDING	411000032	配对未建立或正在建立中
SDK_CONF_EXTERNAL_INPUT_CAMERA_NOT_OPEN	411000033	第三方采集模式下摄像头未打开
SDK_CONF_CREATE_RTC_ENGINE_FAILED	411000034	RTC 引擎创建失败
SDK_CONF_NOT_IN_CONF	411000035	不在会议中
SDK_CONF_MUTE_SELF_FAILED	411000036	静音自己失败并且采用了保护
SDK_NO_STREAM_END_CONF_NET_BREAK	411000100	无码流结束会议网络断开
SDK_NO_STREAM_END_CONF_REJOIN_FAIL	411000101	无码流结束会议重入会失败

变量	值	描述
SDK_CHECK_PARAM_NULL_POINTER	411000102	空指针异常
SDK_CHECK_PARAM_NOT_ALLOW_NULL	411000103	参数不许为空
SDK_ENCRYPT_DECRYPT_FUNC_NOT_MATCH	411000104	加解密算法不匹配
SDK_LOAD_USER_DEFINED_DLL_FAILED	411000105	加载用户自定义dll失败
SDK_APPLICATION_NOT_SUPPORT_SHIMODOCS	411000106	企业配置不支持打开石墨文档
SDK_DATACONF_SCREEN_RENDER_FAIL	411000107	数据会议UI渲染失败
SDK_LOGIN_MSG_THREAD_UNINIT	412100001	消息处理线程未初始化
SDK_LOGIN_PARAM_ERROR	412100002	参数错误
SDK_LOGIN_TIMEOUT	412100003	超时
SDK_LOGIN_MEM_ERROR	412100004	分配内存错误
SDK_LOGIN_XML_ERROR	412100005	XML解析出错
SDK_LOGIN_DNS_ERROR	412100006	DNS解析异常
SDK_LOGIN_REQUEST_FAILED	412100007	请求消息异常
SDK_LOGIN_AUTH_FAILED	412100008	鉴权失败
SDK_LOGIN_SERVICE_ERROR	412100009	服务器异常
SDK_LOGIN_ACCOUNT_LOCKED	412100010	帐号被锁定
SDK_LOGIN_TIMER_ERROR	412100011	创建定时器错误
SDK_LOGIN_CA_VERIFY_FAILED	412100012	服务器CA证书校验失败

变量	值	描述
SDK_LOGIN_NETWORK_ERROR	412100013	网络异常
SDK_LOGIN_SERVER_REDIRECT	412100014	服务器返回重定向地址
SDK_LOGIN_AUTH_ACCOUNT_DIACTIVE	412100015	帐号未激活
SDK_LOGIN_JSON_ERROR	412100016	JSON解析失败
SDK_LOGIN_CURLE_EXTENSION_CONNECT_TIMEOUT	412100017	代理通信超时, tcp建链失败, 大概率是代理配置错误
SDK_LOGIN_CURLE_EXTENSION_CLOSED_BY_PROXY	412100018	接被代理断开, 大概率是服务器地址配置错误
SDK_LOGIN_CURLE_EXTENSION_PROXY_ADDRESS_OR_PORT_ERROR	412100019	tcp建链超时, 大概率是proxy域名或者端口配置错误
SDK_LOGIN_CURLE_EXTENSION_PROXY_PERMISSION_DENIED	412100020	认证鉴权过程失败, 用户没有proxy权限, 需要申请proxy权限
SDK_LOGIN_CURLE_EXTENSION_USER_OR_PASSWORD_ERROR	412100021	认证鉴权过程失败, 代理用户名或者密码错误
SDK_LOGIN_CURLE_EXTENSION_AUTH_TYPE_ERROR	412100022	认证鉴权过程失败, 错误的认证类型, 同时需要用户名、密码, 需要提示用户输入
SDK_LOGIN_CURLE_EXTENSION_AUTH_ERROR	412100023	认证鉴权过程失败, 其他未知错误, 有可能是不支持认证方式, 建议弹框提示用户重新发起
SDK_LOGIN_CURLE_COULDNT_RESOLVE_PROXY	412100024	认证鉴权过程失败, 无法解析网关
SDK_LOGIN_CURLE_COULDNT_RESOLVE_HOST	412100025	认证鉴权过程失败, 无法解析主机
SDK_LOGIN_CURLE_COULDNT_CONNECT	412100026	认证鉴权过程失败, 无法连接
SDK_LOGIN_CURLE_OPERATION_TIMEOUT	412100027	认证鉴权过程失败, 操作超时

变量	值	描述
SDK_LOGIN_MSG_TOO_LARGE	412100028	消息长度太大
SDK_LOGIN_BASE64_ENCODING_ERROR	412100029	base64加密失败
SDK_LOGIN_XML_PARSE_ERROR	412100030	XML解析出错
SDK_LOGIN_SECURE_LIB_ERROR	412100031	安全函数失败
SDK_LOGIN_FIREWALL_DETECT_PARAMETER_ERROR	412100032	防火墙检测参数错误
SDK_LOGIN_BUILD_STG_TUNNEL_FAIL	412100033	创建STG隧道失败
SDK_LOGIN_UPDATE_STG_SERVER_ERROR	412100034	更新STG服务失败
SDK_LOGIN_TUP_COPY_T_ENCRYPT_ERROR	412100035	加密失败
SDK_LOGIN_IN_DOWNLOADING_CANNOT_DOWNLOAD_AGAIN	412100036	不允许重复下载
SDK_LOGIN_FILE_OPERATION_FAIL	412100037	文件操作失败
SDK_LOGIN_RSP_BODY_NULL	412100038	请求消息返回空
SDK_LOGIN_PROBE_SERVER_NUMBER_ZERO_ERROR	412100039	探测服务器数为0错误
SDK_LOGIN_GET_BEST_LOCAL_IP_FAIL	412100040	获取本地IP失败
SDK_LOGIN_E_ERROR_LOGIN_NO_INIT	412100041	登录模块消息未初始化
SDK_LOGIN_E_ERROR_HTTP_BUSY	412100042	http 返回429错误
SDK_LOGIN_E_ERROR_NOT_LOGINED	412100043	此业务未登录不可用
SDK_LOGINLOGIC_GENERAL	412200001	内部错误
SDK_LOGINLOGIC_PARAMETER_ERROR	412200002	参数错误

变量	值	描述
SDK_LOGINLOGIC_REQUEST_TIMEOUT	412200003	网络超时
SDK_LOGINLOGIC_NETWORK_ERROR	412200004	网络错误
SDK_LOGINLOGIC_NEED_MOBILEBIND	412200005	需要绑定手机
SDK_LOGINLOGIC_NEED_MAILBIND	412200006	需要绑定邮箱
SDK_LOGINLOGIC_HAS_INITED	412200007	重复初始化
SDK_LOGINLOGIC_STATE_MACHINE_RUN_FAILED	412200008	状态机执行失败
SDK_LOGINLOGIC_START_TIMER_ERROR	412200009	开启定时器失败
SDK_LOGINLOGIC_NEED_MODIFY_PASSWORD	412200010	首次登录，提示用户修改密码
SDK_LOGINLOGIC_UNKNOWN_AUTH_TYPE	412200011	未知鉴权类型
SDK_CALL_GENERAL_ERROR	413000001	一般错误
SDK_CALL_PARAMETER_ERROR	413000002	参数错误
SDK_CALL_MEMORY_ERROR	413000003	分配内存错误
SDK_CALL_SYSTEM_ERROR	413000004	系统错误
SDK_CALL_MESSAGE_ERROR	413000005	发送消息错误
SDK_CALL_CFG_ERROR	413000006	获取系统配置错误
SDK_CALL_NETWORK_ERROR	413000007	物理网络错误
SDK_CALL_ACCESS_ERROR	413000008	网络接入错误
SDK_CALL_TIMER_ERROR	413000009	创建定时器错误
SDK_CALL_STATE_ERROR	413000010	呼叫状态错误

变量	值	描述
SDK_CALL_OPERATE_ERROR	413000011	正在进行其他操作错误
SDK_CALL_REQSRV_ERROR	413000012	请求主控进行呼叫开始错误
SDK_CALL_REQRES_ERROR	413000013	申请音频资源错误
SDK_CALL_SERVICE_DOING	413000014	正在进行其他补充业务
SDK_CALL_CALLRECORD_ERROR	413000015	记录通话记录错误
SDK_CALL_MPROC_ERROR	413000016	媒体进程返回的错误
SDK_CALL_MAX_CALL_ERROR	413000017	超过最大呼叫路数
SDK_CALL_SIPID_NOTEXIST_ERROR	413000018	SIP账户ID不存在
SDK_CALL_CALLID_NOEXIT_ERROR	413000019	呼叫ID不存在
SDK_CALL_REGISTERING_ERROR	413000020	正在注册中导致失败
SDK_CALL_REGIST_FAIL_ERROR	413000021	注册失败
SDK_CALL_DEREGIST_FAIL_ERROR	413000022	注销失败
SDK_CALL_SET_SIP_INFO_ERROR	413000023	设置账户信息错误
SDK_CALL_SETACCOUNT_ERROR	413000024	设置SIP帐号失败
SDK_CALL_NOTIFYSIPACCOUNT_ERROR	413000025	上报SIP帐号信息失败
SDK_CALL_SIPSEVER_ERROR	413000026	服务器信息错误
SDK_CALL_SIPACCOUNTINFO_ERROR	413000027	账户信息错误
SDK_CALL_SIPC_ERROR	413000028	SIPC执行错误
SDK_CALL_UNREGISTER_ERROR	413000029	未注册错误

变量	值	描述
SDK_CALL_SUB_FALL_ERROR	413000030	调用Sip接口订阅失败
SDK_CALL_DESUBING_ERROR	413000031	注销订阅中错误
SDK_CALL_SUBING_ERROR	413000032	订阅中错误
SDK_CALL_LOCAL_MAX_CONFER_ERROR	413000033	只允许一个本地会议
SDK_CALL_LOCAL_CONFER_NOEXIT_ERROR	413000034	本地会议未创建
SDK_CALL_LOCAL_CONFER_NOMATCH_ERROR	413000035	与会者线路与主席账户不匹配
SDK_CALL_CONF_STATE_ERROR	413000036	本地会议状态错误
SDK_CALL_JOINTSTATE_NOTIFY_ERROR	413000037	IP Phone端联动状态上报失败
SDK_CALL_CONFID_NOTEXIST_ERROR	413000038	会议ID不存在
SDK_CALL_CONF_VIDEO_HOLD_ERROR	413000039	视频保持失败
SDK_CALL_CONF_VIDEO_UNHOLD_ERROR	413000040	视频恢复失败
SDK_CALL_MAX_SERVER_CONF_ERROR	413000041	服务器会议个数超出
SDK_CALL_AA_REQRANDOMNUM_ERROR	413000042	获取AA随机数失败
SDK_CALL_AA_LOGIN_ERROR	413000043	AA登陆失败
SDK_CALL_CONF_VIDEO_TYPE_NO_MATCH	413000044	会议类型不匹配
SDK_CALL_CONF_VIDEO_START_ERROR	413000045	视频会议开启失败
SDK_CALL_AA_NOMAINSERVER_ERROR	413000046	AA无主服务器
SDK_CALL_AA_PASSWORD_ERROR	413000047	密码错误

变量	值	描述
SDK_CALL_AA_USERNAME_ERROR	413000048	用户名错误
SDK_CALL_AA_USERLOGINED_ERROR	413000049	用户已登录
SDK_CALL_AA_ACCOUNTLOCKED_ERROR	413000050	账户已锁定
SDK_CALL_AA_TERMINALTYPE_ERROR	413000051	终端类型不匹配
SDK_CALL_AA_PARSEXML_ERROR	413000052	解析XML错误
SDK_CALL_AA_CONNECT_ERROR	413000053	连接服务器错误
SDK_CALL_MEDIA_CONFIG_ERROR	413000054	获取媒体配置失败
SDK_CALL_AA_GETSRINFO_ERROR	413000055	获取业务权限失败
SDK_CALL_AA_LACKSR_ERROR	413000056	业务权限不足失败
SDK_CALL_NETWORKENV_ERROR	413000057	网络环境错误
SDK_CALL_SERVICE_CONFLICT_ERROR	413000058	业务冲突
SDK_CALL_AA_TIMEOUT_ERROR	413000059	连接超时
SDK_CALL_AA_UNKNOWN_ERROR	413000060	未知错误
SDK_CALL_PICKUP_ADDTIP_ERROR	413000061	添加振铃信息失败
SDK_CALL_PICKUP_REMOVE_TIP_ERROR	413000062	删除加振铃信息失败
SDK_CALL_PICKUP_CREATINGNUM_ERROR	413000063	创建振铃号码失败
SDK_CALL_VVM_PARAMETER_ERROR	413000064	VVM参数错误
SDK_CALL_GET_IP_INFO_ERROR	413000065	获取登记业务错误

变量	值	描述
SDK_CALL_VVM_GETV OICEMAIL_ERR	413000066	获取语音邮箱错误
SDK_CALL_ATTENDEE_ ALREADY_EXIST	413000067	与会者已经存在
SDK_CALL_ATTENDEE_ NOT_EXIST	413000068	与会者不存在
SDK_CALL_CREATE_CO NF_WND_ERROR	413000069	创建服务器视频会议窗口失败
SDK_CALL_CREATE_CO NF_WND_EXIST	413000070	视频窗口已存在
SDK_CALL_GET_CONF_ LIST_INFO_ERROR	413000071	获取会议列表失败
SDK_CALL_NEED_CHAI RMAN_TO_OPERATE	413000072	需要主持人权限才能操作
SDK_CALL_NO_VIDEO_ DEVICE_TO_OPERATE	413000073	没有视频设备可以操作
SDK_CALL_NOT_STOP_ REFRESHREG	413000074	没有关闭刷新注册
SDK_CALL_NOTIFY_ON LINE_STATE_ERROR	413000075	在线状态上报失败
SDK_CALL_NOTIFY_NE TADDR_ERROR	413000076	网络地址本订阅notify上报失败
SDK_CALL_INFO_FORC EUNREG_ERROR	413000077	智真帐号被踢,info上报失败
SDK_CALL_TLSROOTCE RT_ERROR	413000078	TLS根证书错误
SDK_CALL_AD_AUTH_F AIL	413000079	AD鉴权失败
SDK_CALL_GET_CONF_ LIST_INFO_DOING	413000080	会议列表正在获取中
SDK_CALL_CALL_PROH IBITED	413000081	禁止呼叫
SDK_CALL_CALL_NUM BER_UPPER_LIMIT	413000082	呼叫数达上限
SDK_CALL_ENCRYPT_C ALL_UPPER_LIMIT	413000083	加密呼叫数达上限
SDK_CALL_SHOULD_D EGRADE_TO_AUDIO	413000084	超出视频呼叫数, 需要降为音频

变量	值	描述
SDK_CALL_H323ID_NO TEXIST_ERROR	413000085	H323账户ID不存在
SDK_CALL_H323_ACCO UNTINFO_ERROR	413000086	H323账户信息错误
SDK_CALL_NOTIFY_H3 23_ACCOUNT_ERROR	413000087	H323帐号信息上报失败
SDK_CALL_LOCAL_RES OURCE_NOT_ENOUGH	413000088	本端资源不足
SDK_CALL_CALL_PROT OCOL_NOT_SUPPORT	413000089	不支持该呼叫协议
SDK_CALL_SET_H323_I NFO_ERROR	413000090	设置H323账户信息错误
SDK_CALL_NOTIFY_AC CESSNUM_ERROR	413000091	mediax会议接入号上报界面失败
SDK_CALL_NOTIFY_VM RINFO_ERROR	413000092	上报MediAx VMR信息到界面失败
SDK_CALL_NOTIFY_CO NFLIST_INFO_ERROR	413000093	上报获取IMS会议列表信息到界面失败
SDK_CALL_TUP_INIT_B LOCK	413000094	tup初始化未完成
SDK_CALL_MSG_RETU RN_VALUE_ERROR	413000095	消息返回值错误
SDK_CALL_MSG_TOO LARGE_ERROR	413000096	消息内容太大错误
SDK_CALL_SECURE_FU NC_RETURN_ERROR	413000097	安全函数返回错误
SDK_CALL_MSG_RESP_ NULL_ERROR	413000098	消息响应为空错误
SDK_CALL_ACCOUNTI D_NOT_ENABLE_ERRO R	413000099	帐号未启用
SDK_CALL_CREAT_CAL L_IMPL_ERROR	413000100	创建呼叫控制类错误
SDK_CALL_ALREADY_I N_CALL_ERROR	413000101	已经在呼叫中
SDK_CALL_NO_CALL_E XIST_ERROR	413000102	呼叫不存在

变量	值	描述
SDK_CALL_GET_MEDIA_CONFIG_FAILED	413000103	获取媒体配置信息失败
SDK_CALL_CAPTURE_INPUTDATA_FAILED	413000104	设置第三方采集失败, 调用 HME_V_Capture_InputData_MCU返回失败
SDK_CONFCTRL_GENERAL	414100001	一般错误
SDK_CONFCTRL_PARAMETER_ERROR	414100002	参数错误
SDK_CONFCTRL_TIMEOUT	414100003	超时
SDK_CONFCTRL_MEMORY_ERROR	414100004	分配内存错误
SDK_CONFCTRL_DNS_ERROR	414100005	DNS解析异常
SDK_CONFCTRL_REQUEST_FAILED	414100006	请求消息异常
SDK_CONFCTRL_AUTH_FAILED	414100007	鉴权失败
SDK_CONFCTRL_SERVICE_ERROR	414100008	服务器异常
SDK_CONFCTRL_TIMER_ERROR	414100009	创建定时器错误
SDK_CONFCTRL_START_TIMER_ERROR	414100010	开启定时器错误
SDK_CONFCTRL_CONFIG_BLOCK_FULL	414100011	会控控制块已满, 请释放没在使用的控制块
SDK_CONFCTRL_CURL_E_EXTERN_CONNECT_TIMEOUT	414100012	代理通信超时, tcp建链失败, 大概率是代理配置错误
SDK_CONFCTRL_CURL_E_EXTERN_CLOSED_BY_PROXY	414100013	接被代理断开, 大概率是服务器地址配置错误
SDK_CONFCTRL_CURL_E_EXTERN_PROXY_ADDR_OR_PORT_ERROR	414100014	tcp建链超时, 大概率是proxy域名或者端口配置错误
SDK_CONFCTRL_CURL_E_EXTERN_PROXY_PERMISSION_DENIED	414100015	认证鉴权过程失败, 用户没有proxy权限, 需要申请proxy权限

变量	值	描述
SDK_CONFCTRL_CURL E_EXTERN_USER_OR_P WD_ERROR	414100016	认证鉴权过程失败，代理用户名或者密码错误
SDK_CONFCTRL_CURL E_EXTERN_AUTH_TYPE _ERROR	414100017	认证鉴权过程失败，错误的认证类型，同时需要用户名、密码，需要提示用户输入
SDK_CONFCTRL_CURL E_EXTERN_AUTH_ERR OR	414100018	认证鉴权过程失败，其他未知错误，有可能是不支持认证方式，建议弹框提示用户重新发起
SDK_CONFCTRL_CURL E_COULDNT_RESOLVE _PROXY	414100019	curl代理异常
SDK_CONFCTRL_CURL E_COULDNT_RESOLVE _HOST	414100020	curl主机异常
SDK_CONFCTRL_CURL E_COULDNT_CONNEC T	414100021	curl连接异常
SDK_CONFCTRL_CURL E_OPERATION_TIMED OUT	414100022	curl操作超时
SDK_CONFCTRL_MSG_ TOO_LARGE	414100023	会控消息内容过大
SDK_CONFCTRL_PTHR EAD_CREAT_ERROR	414100024	创建线程失败
SDK_CONFCTRL_SEC_F UNC_RETURN_ERROR	414100025	安全函数返回错误
SDK_CONFCTRL_JSON_ PARSE_FAILED	414100026	解析json体失败
SDK_CONFCTRL_TOKE N_IS_NULL_ERROR	414100027	ws token值为空
SDK_CONFCTRL_RETU RN_JSON_ERROR	414100028	返回json数据异常
SDK_CONFCTRL_CREA T_CONF_BASIC_ERROR	414100029	创建会议控制块失败
SDK_CONFCTRL_GET_C ONF_BASIC_BY_HAN DLE_ERROR	414100030	获取会议控制块失败

变量	值	描述
SDK_CONFCTRL_GET_CONF_BASIC_BY_CALLID_ERROR	414100031	通过CallId获取会议控制块失败
SDK_CONFCTRL_DESTORY_CONF_BASIC_ERROR	414100032	销毁会议控制块失败
SDK_CONFCTRL_CREAT_CONF_TMP_INFO_NOT_FOUND	414100033	找不到创会临时信息
SDK_CONFCTRL_GET_PARTICIPANT_ID_FAILED	414100034	获取参会者ID失败
SDK_CONFCTRL_GET_MT_BY_PARTICIPANT_ID_FAILED	414100035	通过ParticipantId获取MT失败
SDK_CONFCTRL_GET_SELFINFO_ERROR	414100036	通过ParticipantId获取selfinfo失败
SDK_CONFCTRL_GET_PARTICIPANT_BY_MT_FAILED	414100037	通过MT获取会议成员信息失败
SDK_CONFCTRL_NEED_MANUAL_INVITE_ERROR	414100038	需要手动邀请
SDK_CONFCTRL_HAD_NOTIFY_CONFINFO_BEFORE	414100039	已上报过会议大参数
SDK_CONFCTRL_NOT_SUPPORT_SUBTITLE	414100040	不支持字幕
SDK_CONFCTRL_CONF_END_TIMEOUOT	414100041	会议挂断 - 超时
SDK_CONFCTRL_CONF_END_NOSTREAM	414100042	会议挂断 - 无码流
SDK_CONFCTRL_CONF_END_CHAIR_HANGUP	414100043	会议挂断 - 被主持人挂断
SDK_CONFCTRL_CONF_END_STOP_CONF	414100044	会议挂断 - 会议已结束
SDK_CONFCTRL_CONF_END_LEAVE_CONF	414100045	离会导致的会议结束
SDK_CONFCTRL_MSG_PROC_FUN_REG_FAIL	414100046	消息队列处理函数注册失败

变量	值	描述
SDK_CONFCTRL_MSG_PROC_FUN_UNREG_FAIL	414100047	消息队列处理函数去注册失败
SDK_CONFCTRL_CONF_END_REJECT_CONF	414100048	拒绝会议导致的会议结束
SDK_CONFCTRL_CONF_END_SELF_END	414100049	自己结束会议导致的会议结束
SDK_CONFCTRL_VIEWER_EXCEEDS_LIMIT	414100050	会控模块未初始化
SDK_CONFCTRL_SPECIAL_WATCH_EXCEEDS_LIMIT	414100051	选看的特殊会场超过限制
SDK_CONFCTRL_WATCH_HD_EXCEEDS_LIMIT	414100052	选看的高清会场超过限制
SDK_CONFCTRL_WATCH_SD_EXCEEDS_LIMIT	414100053	选看的标清会场超过限制
SDK_CONFCTRL_WATCH_SPECIFICATIONS_EXCEEDS_LIMIT	414100054	选看的规格超过限制
SDK_CONFCTRL_WATCH_INNER_ERROR	414100055	选看内部错误
SDK_CONFCTRL_WATCH_RENDER_REPEATS	414100056	选看的句柄重复
SDK_CONFCTRL_HTTP_RECV_FAILED	414100057	HTTP请求错误
SDK_CONFCTRL_TRACK_REQUEST_TIMEOUT	414100058	websocket请求超时
SDK_CONFCTRL_TRACK_LOAD_FAILED	414100059	加载错误
SDK_CONFCTRL_CONF_END_TIMEOUT_HANGUP	414100061	会前等待页面时间超时或者会控心跳超时导致的会议结束
SDK_CONFCTRL_CONF_END_CONCLUDE	414100062	会议被取消预约导致的会议结束
SDK_CONFCTRL_CONF_END_NETWORK_FAILED	414100063	会前等待页面网络异常导致的会议结束
SDK_CONFCTRL_E_ERR_HTTP_BUSY	414100064	http 返回429错误

变量	值	描述
SDK_CONFCTRL_E_ERR_SELF_IS_CHAIRMAN	414100065	已是主持人
SDK_CONFCTRL_E_ERR_SELF_IS_COHOST	414100066	已是联席主持人
SDK_CONFCTRL_E_ERR_NO_COHOST_AND_CHAIRMAN	414100067	无主持人和联席主持人
SDK_CONFCTRL_E_ERR_UNUPPORT_COHOST_AND_CHAIRMAN	414100068	无支持的主持人和联席主持人
SDK_COMMON_SQLITE_E_NOT_INITED	415000001	db未初始化
SDK_COMMON_E_FAIL	415000002	失败
SDK_COMMON_E_ERR_PROXY_UNFINISH	415000003	代理探测还没有完成
SDK_TRACK_E_INVALID_PARAMETER	416000001	参数错误
SDK_TRACK_E_NOT_INITED	416000002	打点模块未初始化
SDK_TRACK_E_EVENT_NOT_EXIST	416000003	打点事件不存在
SDK_TRACK_E_NOT_LOGINED	416000004	未登录
SDK_TRACK_E_NO_UPLOAD	416000005	未开启上传
SDK_SIP_INVALIDPARAM	419100001	错误参数
SDK_SIP_WRONGURI	419100002	错误号码
SDK_SIP_MEMALLOCFAILURE	419100003	内存分配失败
SDK_SIP_SDPROLEFAILURE	419100004	SDP错误
SDK_SIP_NOCONNECTION	419100005	连接不存在
SDK_SIP_CONNECTION_EXPIRED	419100006	连接失效
SDK_SIP_FORBIDDEN	419100007	禁止操作

变量	值	描述
SDK_SIP_WRONGSTATE	419100008	状态错误
SDK_SIP_NOLINEDATA	419100009	无对应线路数据
SDK_SIP_NOREGISTER	419100010	无对应注册实例
SDK_SIP_NODIALOG	419100011	无对应呼叫实例
SDK_SIP_NOSUBSCRIBE	419100012	无对应订阅实例
SDK_SIP_NOREFER	419100013	无对应Refer实例
SDK_SIP_NODIM	419100014	无对应DIM实例
SDK_SIP_CREATEHEADER	419100015	创建头域失败
SDK_SIP_PARSEHEADER	419100016	解析头域失败
SDK_SIP_FIELDEXIST	419100017	没有找到头域
SDK_SIP_NOSDPBODY	419100018	没有sdp消息体
SDK_SIP_GETCAP	419100019	获取能力失败
SDK_SIP_COMMCAP	419100020	比较能力失败
SDK_SIP_OPENLCFAILED	419100021	打开逻辑通道失败
SDK_SIP_ENCODESDP	419100022	编码SDP失败
SDK_SIP_DECODESDP	419100023	解码SDP失败
SDK_SIP_UNSUPPORTSDP	419100024	协商到不支持的SDP
SDK_SIP_TPTDINIT	419100025	TPTD初始化失败
SDK_SIP_TPTINIT	419100026	TPT初始化失败
SDK_SIP_TXNINIT	419100027	TXN初始化失败
SDK_SIP_UAINIT	419100028	UA初始化失败
SDK_SIP_GLUEINIT	419100029	GLUE初始化失败
SDK_SIP_HLLMINIT	419100030	HLLM初始化失败
SDK_SIP_VCOMINIT	419100031	VCOM初始化失败
SDK_SIP_SDPINIT	419100032	SDP初始化失败
SDK_SIP_MULTIIINS	419100033	多实例错误

变量	值	描述
SDK_SIP_NOREGCB	419100034	注册控制块不足
SDK_SIP_NOSUBCB	419100035	订阅控制块不足
SDK_SIP_NODIACB	419100036	通话控制块不足
SDK_SIP_NODIMCB	419100037	对话外控制块不足
SDK_SIP_PARSEURITOSTR	419100038	解析xml的uri失败
SDK_SIP_ENCRYPTNEGOTERR	419100039	加密协商失败
SDK_SIP_NOLOCALCAP	419100040	没有设置本地能力
SDK_SIP_BFCPTOKENERR	419100041	bfcpl令牌错误
SDK_SIP_DNSERR	419100042	dns解析错误
SDK_SIP_GET_SUB_STATE_ERR	419100043	获取订阅状态失败
SDK_SIP_NOT_SUPPORT_IPV4	419100044	不支持IPv4
SDK_SIP_NOT_SUPPORT_IPV6	419100045	不支持IPv6
SDK_SIP_NOT_SUPPORT_HOST_TYPE	419100046	不支持的主机类型
SDK_SIP_GET_TIMER_ERROR	419100047	获取定时器失败
APP_CANNOT_ACCEPT_INVITE_SHARE	420000001	无法接收共享邀请
APP_REPORT_CHECK_PARAMETER_FAILED	420000002	reportDTO参数校验失败
APP_REPORT_FOLD_CREATE_FAILED	420000003	举报文件夹创建失败
APP_REPORT_COMPRESS_FAILED	420000004	举报文件夹压缩失败
APP_REPORT_ZIP_PATH_EMPTY	420000005	待上传的ZIP文件路径为空
APP_REPORT_ZIP_SIZE_EXCEED_LIMIT	420000006	待上传的ZIP文件大小超过限制
APP_REPORT_EXCEED_LIMIT_DAILY	420000007	同一个人24小时内举报次数超过100次

变量	值	描述
UISDK_COMMON_ERROR	430100001	通用错误
UISDK_API_INVALID_PARAMETERS	430100002	参数错误
UISDK_SIP_CHECK_TIMEOUT	430100003	SIP连接检查失败
UISDK_TIMEOUT	430100004	超时
UISDK_IN_PROCESS	430100005	接口正在处理中
UISDK_NOT_INITED_END	430100006	未初始化或初始化尚未完成。
UISDK_NOT_SUPPORT_END	430100007	此版本不支持。
UISDK_OPERATION_INTERRUPT	430100008	操作中断。
UISDK_CONTACT_NOT_FOUND	430100009	联系人不存在
UISDK_ARGS_LENGTH_OVERRUN	430100010	参数长度超限
UISDK_APPID_INFO_ILLEGAL	430100011	APPID 信息入参不合法
UISDK_E2EENCRYPT_ALGORITHM_ERROR	430100012	端到端加密算法不对称
UISDK_UNLOGIN_ERROR	430200001	未登录
UISDK_REPEATED_LOGIN_ERROR	430200002	重复登录
UISDK_SIP_CONNECTION_TIMEOUT	430200003	SIP连接超时
UISDK_LOGIN_INVALID_PASSWORD_PARAMETER	430200004	用于登录的密码参数不合法
UISDK_PASSWORD_OR_ACCOUNT_EMPTY	430200005	账号或密码不能都为空
UISDK_CALL_INVALID_ACCOUNT_PARAMETER	430300001	呼叫号码、账号或者第三方账号不能为空
UISDK_CALL_ACCOUNT_EMPTY	430300002	点对点呼叫账号为空

变量	值	描述
UISDK_CALL_NICKNAME_EMPTY	430300003	点对点呼叫名字为空
UISDK_JOINCONF_NEED_PASSWORD	430400001	加入会议需要密码
UISDK_INVALID_TIMEZONE_PARAMETER	430400002	时区错误
UISDK_INVALID_CONF_DURATION_PARAMETER	430400003	会议时长小于1
UISDK_CONF_INTERRUPTED	430400004	入会被中断
UISDK_JOINCONF_INPUT_INVALID_PASSWORD	430400005	密码输入至少4位
UISDK_JOINCONF_USER_CANCEL	430400006	用户主动取消
UISDK_INVALID_GUEST_PASSWORD_PARAMETER	430400007	来宾密码非法
UISDK_NOT_IN_CONF_OR_CALL	430400008	不在会议或呼叫中
UISDK_ALREADY_EXIST_CONF_OR_CALL	430400009	已存在一路会议或呼叫
UISDK_CONFID_EMPTY	430400010	会议id不能为空
UISDK_SUBJECT_TOO_LONG	430400011	会议主题长度不能超过50
UISDK_VMR_HOST_PASSWORD_LENGTH_LEGAL	430400012	主持人密码长度不合法
UISDK_VMR_NAME_EMPTY	430400013	VMR名字为空
UISDK_CONF_STARTTIME_EARLY	430400014	会议的开始时间早于当前时间
UISDK_SHARE_LOCKED	430400015	共享被锁定
UISDK_NOT_IN_DATACONF	430400016	不在数据会议中
UISDK_ROLE_NOT_SUPPORT_OPERATION	430400017	角色不支持该操作

变量	值	描述
UISDK_HTTP_SERVER_ERROR	430500001	HTTP请求, 服务器异常
UISDK_HTTP_CONNECT_SERVER_TIMEOUT_ERROR	430500002	HTTP请求, 连接服务器超时
UISDK_NO_NETWORK_ERROR	430500003	网络不可用
UISDK_FILE_TYPE_ILLEGAL	430500004	文件类型不合法
SDK_CALL_REASON_CODE_403_FORBIDDEN	510000403	服务器拒绝
SDK_CALL_REASON_CODE_404_NOT_FOUND	510000404	未发现
SDK_CALL_REASON_CODE_480_TEMP_UNAVAILABLE	510000480	临时失效
SDK_CALL_REASON_CODE_486_BUSYHERE	510000486	这里忙
SDK_CALL_REASON_CODE_487_REQ_TERMINATED	510000487	请求终止
SDK_CALL_REASON_CODE_488_NOT_ACCEPTABLE_HERE	510000488	这里请求不可接受
SDK_CALL_REASON_CODE_603_DECLINE	510000603	丢弃
SDK_CALL_REASON_CODE_841_HANGUP_WITHOUT_TOAST	510000841	挂断不弹框
SDK_CALL_REASON_CODE_842_CONFERENCE_IS_LOCK	510000842	会议被锁定
RTC_SDK_LEAVE_REASON_SERVER_ERROR	910000001	服务器异常
RTC_SDK_LEAVE_REASON_BREAKDOWN	910000002	sfu服务故障
RTC_SDK_LEAVE_REASON_SERVICE_UNREACHABLE	910000003	服务不可达503

变量	值	描述
RTC_SDK_LEAVE_REASON_INTERNAL_ERROR	910000004	内部错误
RTC_SDK_LEAVE_REASON_KICKED_OFF	910000005	被踢
RTC_SDK_LEAVE_REASON_SIGNATURE_EXPIRED	910000006	签名过期
RTC_SDK_LEAVE_REASON_RECONNECT_FAILED	910000007	重连超时
RTC_SDK_LEAVE_REASON_NETWORK_TEST	910000008	网络检测，UI不需要关注该错误码，不对外体现
RTC_SDK_LEAVE_REASON_USER_REMOVED	910000009	用户移除
RTC_SDK_LEAVE_REASON_ROOM_DISMISSED	910000010	房间解散
RTC_SDK_SDK_INTERNAL_ERROR	990000001	内部错误
RTC_SDK_MSG_TOOLARGE	990000002	消息长度太大
RTC_SDK_MEM_NOT_ENOUGH	990000003	内存不足
RTC_SDK_SYNSEND_MSG_ERR	990000004	同步发送消息失败
RTC_SDK_PARAM_ERROR	990000005	参数错误
RTC_SDK_API_CALLED_IN_WRONG_ORDER	990000006	呼叫ID时序错误
RTC_SDK_SETUP_LOCAL_VIEW_FAIL	990000007	设置本端画面失败
RTC_SDK_START_REMOTE_STREAM_VIEW_FAIL	990000008	开启远端画面失败
RTC_SDK_SET_DEVICE_FAIL	990000009	设备配置失败
RTC_SDK_INITIALIZING	990000010	初始化中
RTC_SDK_UN_INITIALIZING	990000011	去初始化中

变量	值	描述
RTC_SDK_LOG_UPLOADING	990000012	正在日志上传中
RTC_SDK_MEDIA_PORT_ERROR	990000013	获取媒体端口失败
RTC_SDK_WATCH_VIDEO_TOO_MUCH	990000014	选看窗口超过规格数
RTC_SDK_MEDIA_CONNECT_ERROR	990000015	媒体协商失败
RTC_SDK_SERVER_NO_RESPONSE	990000016	服务器没有响应
RTC_SDK_USER_ROLE_CHANGE_FAIL	990000017	角色切换失败
RTC_SDK_JOIN_ROOM_FAIL	990000018	加入房间失败
RTC_SDK_JOIN_ROOM_STATUS_BUSY	990000019	非空闲状态
RTC_SDK_JOIN_ROOM_SERVER_ERROR	990000020	加入房间失败，服务器异常
RTC_SDK_JOIN_ROOM_SERVICE_UNREACHABLE	990000021	加入房间失败，服务不可达
RTC_SDK_JOIN_ROOM_AUTH_FAIL	990000022	加入房间失败，鉴权失败
RTC_SDK_JOIN_ROOM_AUTH_RETRY	990000023	加入房间失败，鉴权重试
RTC_SDK_JOIN_ROOM_AUTH_CLOCK_SYNC	990000024	加入房间失败，时钟同步
RTC_SDK_JOIN_ROOM_URL_NOT_RIGHT	990000025	加入房间失败，url错误
RTC_SDK_KICKED_OFF	990000026	被踢
RTC_SDK_SCREEN_CAPTURE_FAIL	990000027	共享失败
RTC_SDK_EXT_MEDIA_OUTPUT	990000028	当前为外部媒体输出模式，禁用该操作
RTC_SDK_RECONNECT_FAILED	990000029	重连失败
RTC_SDK_SERVER_BREAKDOWN	990000030	服务器宕机

变量	值	描述
RTC_SDK_SIGNATURE_EXPIRED	990000031	签名已过期
RTC_SDK_SET_REMOTE_RENDER_MODE_FAIL	990000032	设置远端窗口模式失败
RTC_SDK_SET_REMOTE_AUDIO_MUTE_FAIL	990000033	订阅或取消订阅音频失败
RTC_SDK_CONNECT_OTHER_ROOM_FAIL	990000034	连接其他房间失败
RTC_SDK_DISCONNECT_OTHER_ROOM_FAIL	990000035	断开连接其他房间失败
RTC_SDK_SET_USERROLE_NOT_ALLOWED	990000036	不允许角色切换
RTC_SDK_EXT_MEDIA_CAPTURE_INPUT	990000037	当前为第三方采集模式，禁用该操作
RTC_SDK_SET_EXTAUDIO_CAPTURE_FAIL	990000038	设置第三方音频采集失败
RTC_SDK_SET_EXTVIDEO_CAPTURE_FAIL	990000039	设置第三方视频采集失败
RTC_SDK_SET_SHARE_COMPUTER_SOUND_FAIL	990000040	设置共享声音开关失败
RTC_SDK_SET_LOCAL_AUDIO_MUTE_FAIL	990000041	启停上行音频流失败
RTC_SDK_SET_LOCAL_VIDEO_MUTE_FAIL	990000042	启停上行视频流失败
RTC_SDK_USER_REMOVED	990000043	用户被移除
RTC_SDK_ROOM_DISSUED	990000044	房间被解散
RTC_SDK_SETUP_REMOTE_VIEW_FAIL	990000045	设置远端View失败
RTC_SDK_ROOM_FULL	990000052	房间已满，加入失败

9.7 其他说明

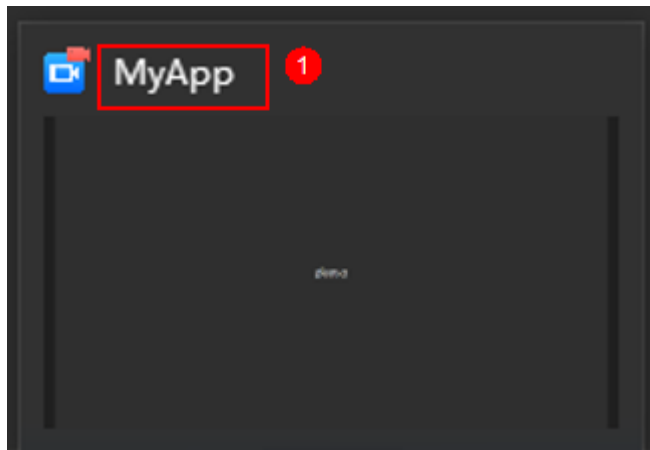
9.7.1 界面配置 json 说明

会议中的界面，界面上的部分元素是通过config接口进行自定义。可以自定义的界面元素如下：

- 会议窗口名称（仅针对windows平台）

会议开始后，在Windows任务栏上会生成一个会议的图标，鼠标指向该图标时会显示会议的缩略窗口，该窗口的名称可以自定义。

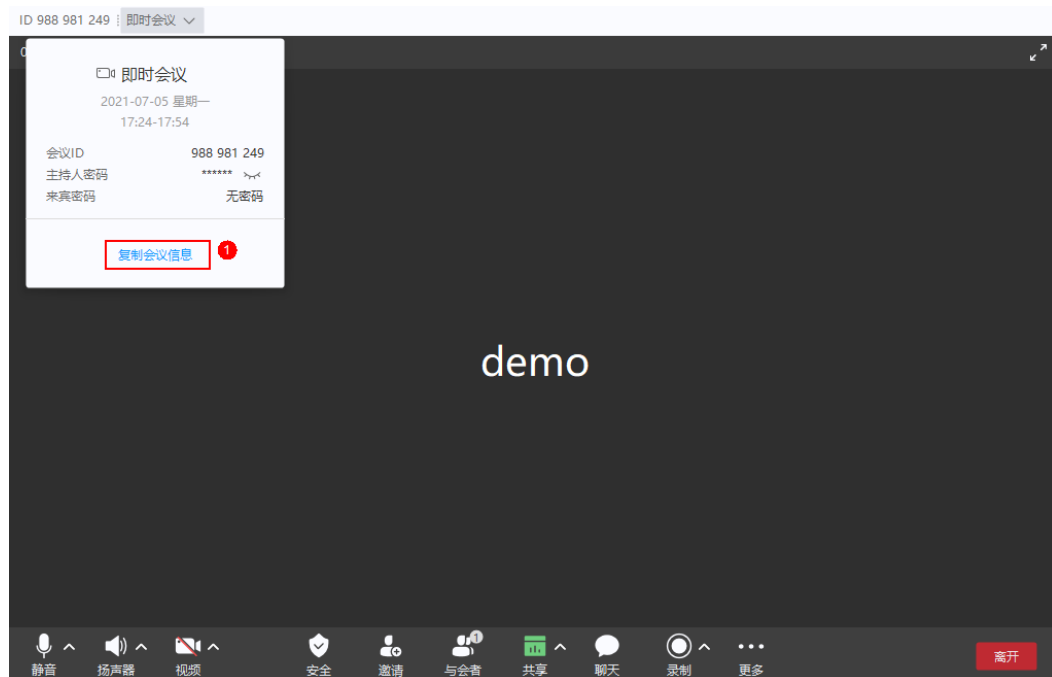
图 9-3 会议窗口名称



- 会议信息页面点击复制会议详情的内容

会议链接的域名前缀默认是https://bmeeting.huaweicloud.com/（如https://bmeeting.huaweicloud.com/#/j/982989574），如果入会Portal是第三方自己开发的，会议信息页面点击复制会议详情按钮后，会议详情内容需要用户自定义。

图 9-4 会议信息



- **会控按钮**

1. 删除或者修改SDK中自带的会控按。具体可以删除和修改的按钮见[按钮ID定义表](#)。
2. 自定义新增按钮。具体用例见[自定义会控按钮](#)。

图 9-5 底部工具栏和更多菜单

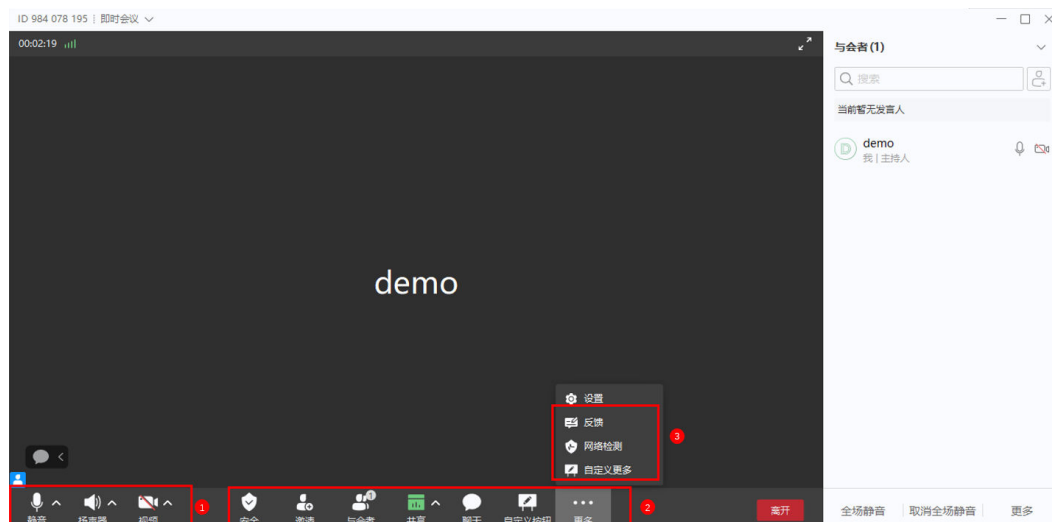


图 9-6 共享工具栏和更多菜单

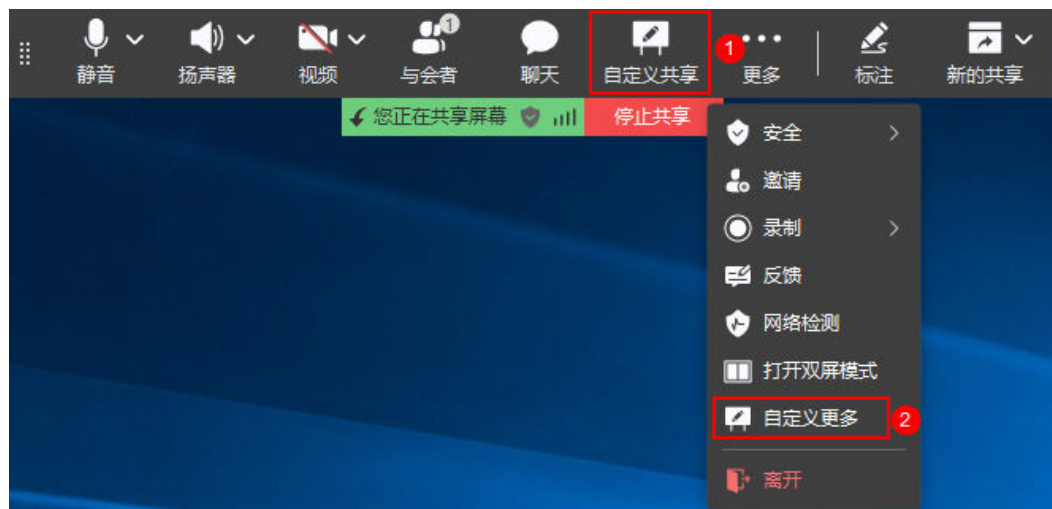
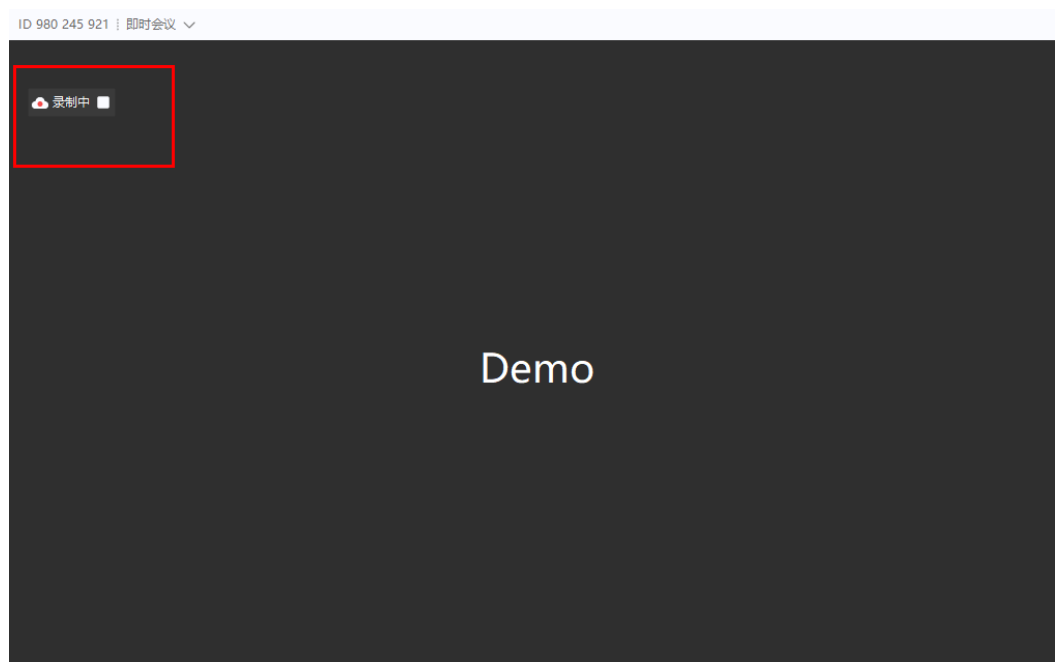


图 9-7 状态工具栏



config接口参数描述

表 9-327 参数说明

参数	是否必须	类型	描述
uiConfig	是	string	Json对象字符串{"frame":object}", object对象有格式要求, 若不匹配则配置不生效。详细规则见下表2、表3、表4、表5。

表 9-328 frame 对象介绍

key	value 类型	描述
name	string	会议窗口名称。如图1中的窗口名称 (MyApp)。
confMenu	object	会议窗口上的按钮配置对象。如图3和图4中所示按钮。

表 9-329 confMenu 对象介绍

key	value类型	描述
toolBar	object	会议窗口上的按钮配置对象。如图3和图4中所示按钮。
titleBar	object	会议窗口上的标题栏配置对象。如图2中的会议信息。
statusBar	object	会议窗口左上角的状态栏配置对象。如图5中的按钮。

表 9-330 toolBar 对象子 key 介绍

key	value类型	描述
button	object[]	会议窗口上的按钮配置对象。如图3和图4中所示按钮和菜单。
customButton	object[]	会议窗口上的第三方自定义按钮配置对象。如图3中的②③和图4中的①②。

表 9-331 statusBar 对象子 key 介绍

key	value类型	描述
button	object[]	会议窗口左上角的状态栏上的按钮配置对象。如图5中的按钮。

表 9-332 titleBar 对象介绍

key	value类型	描述
confDetail	object[]	如图2中的会议信息

表 9-333 confDetail 的对象 key 介绍

key	value类型	描述
button	object[]	会议信息窗口上的按钮配置对象。

表 9-334 toolBar 对象和 confDetail 对象中 button 对象子 key 介绍

key	value 类型	描述
buttonId	string	按钮的唯一标识。具体buttonId与按钮的对应关系见表9 按钮buttonId介绍。
showAsAction	string	是否显示。只有2个取值，“never”：不显示，“ifRoom”：显示。
isCustomizedClick	bool	是否点击注入。true：点击注入，用户点击按钮时发送通知消息，SDK不做点击响应处理。

表 9-335 customButton 对象子 key 介绍

key	value 类型	描述
buttonId	string	第三方自定义的按钮id。如“customMenu”。
title	string	第三方自定义按钮的显示的名称。如“自定义按钮”。
buttonImg	string	第三方自定义按钮的图标路径，使用绝对路径。如“to/path/image.svg”，支持svg/png等格式 注：需要UTF8编码。
buttonPos	int	第三方自定义按钮的位置。0：工具栏更多菜单，1：底部工具栏，2：共享工具栏

表 9-336 按钮 buttonId 介绍

按钮名称	按钮id	说明
麦克风	"microphone"	可隐藏
扬声器	"speaker"	可隐藏
摄像头	"camera"	可隐藏
邀请	"invite"	可隐藏，可注入
离开	"leave"	可隐藏，可注入
共享	"share"	可隐藏，可注入
与会者	"attendee"	可隐藏，可注入
反馈	"feedback"	可隐藏，可注入
标注	"annotation"	可隐藏

按钮名称	按钮id	说明
授予远程控制	"remote_control"	可隐藏
停止共享	"stop_share"	可隐藏
聊天	"chat"	可隐藏, 可注入
字幕功能: 开启字幕、字幕翻译	"subtitle"	可隐藏
安全	"security"	可隐藏
网络检测	"network_detection"	可隐藏
允许与会者解除静音	"allow_unmute"	可隐藏
打开/关闭双屏模式	"dual_screen"	可隐藏
录制	"record"	可隐藏
投票	"vote"	可隐藏
复制会议信息	"copy_conf_info"	可注入
与会者详情	"participant_detail"	可注入 (mac无需注入)

按照上面的定义, 完整的可定制的界面元素的json格式配置项如下:

```
{
  "frame": {
    "name": "demo",
    "confMenu": {
      "titleBar": {
        "confDetail": {
          "button": [
            {
              "buttonId": "copy_conf_info",
              "isCustomizedClick": true
            }
          ]
        }
      }
    },
    "statusBar": {
      "button": [
        {
          "buttonId": "record",
          "showAsAction": "never"
        }
      ]
    },
    "toolBar": {
      "customButton": [
        {
          "buttonId": "customMenu",
          "title": "自定义按钮",
          "buttonPos": 1,
          "buttonImg": "imagePath",
          "subMenu": ""
        },
        {
          "buttonId": "customMore",
          "title": "自定义更多",

```

```
    "buttonPos": 0,  
    "buttonImg": "imagePath"  
  },  
  {  
    "buttonId": "customShare",  
    "title": "自定义共享",  
    "buttonPos": 2,  
    "buttonImg": "imagePath"  
  }  
],  
"button": [  
  {  
    "buttonId": "microphone",  
    "showAsAction": "never"  
  },  
  {  
    "buttonId": "speaker",  
    "showAsAction": "never"  
  },  
  {  
    "buttonId": "camera",  
    "showAsAction": "never"  
  },  
  {  
    "buttonId": "invite",  
    "showAsAction": "never",  
    "isCustomizedClick": true  
  },  
  {  
    "buttonId": "leave",  
    "showAsAction": "never",  
    "isCustomizedClick": true  
  },  
  {  
    "buttonId": "attende",  
    "showAsAction": "never",  
    "isCustomizedClick": true,  
    "dialogPos": "center"  
  },  
  {  
    "buttonId": "share",  
    "showAsAction": "never",  
    "isCustomizedClick": true  
  },  
  {  
    "buttonId": "chat",  
    "showAsAction": "never",  
    "isCustomizedClick": true,  
    "dialogPos": "center"  
  },  
  {  
    "buttonId": "annotation",  
    "showAsAction": "never"  
  },  
  {  
    "buttonId": "stop_share",  
    "showAsAction": "never"  
  },  
  {  
    "buttonId": "feedback",  
    "showAsAction": "ifRoom",  
    "isCustomizedClick": true  
  },  
  {  
    "buttonId": "dual_screen",  
    "showAsAction": "never"  
  },  
  {  
    "buttonId": "remote_control",
```


底部工具栏的更多菜单中自定义按钮添加在菜单栏最后，共享工具栏的更多菜单中自定义按钮添加在“离开”按钮前。

- 删除自定义按钮。通过Config接口删除customButton中的按钮配置，关于清空自定义按钮操作，若赋值customButton: []，则清空全量自定义按钮；若customButton下未配置某位置下的自定义按钮，则清空该位置下的自定义按钮。

📖 说明

通过Config接口增删改自定义按钮时，需要传对应位置的全量自定义按钮，支持动态刷新底部/共享工具栏，更多菜单不支持，而是在重新生成时刷新。

- **增加自定义按钮**

以添加自定义带子菜单的底部工具栏按钮、共享工具栏按钮和更多菜单按钮为例：

```
{
  "frame": {
    "confMenu": {
      "toolBar": {
        "customButton": [
          {
            "buttonId": "customMenu",
            "title": "自定义按钮",
            "buttonImg": "../../native/default/svg/icon-whiteboard.svg",
            "buttonPos": 1
          },
          {
            "buttonId": "customMore",
            "title": "自定义更多",
            "buttonImg": "../../native/default/svg/icon-whiteboard.svg",
            "buttonPos": 0
          },
          {
            "buttonId": "customShare",
            "title": "自定义共享",
            "buttonImg": "../../native/default/svg/icon-whiteboard.svg",
            "buttonPos": 2
          }
        ]
      }
    }
  }
}
```



- **更新自定义按钮**

以重命名底部工具栏按钮为例：

```
{
  "frame": {
    "confMenu": {
      "toolBar": {
        "customButton": [
          {
            "buttonId": "customMenu",
```



```
        "title": "重命名按钮",
        "buttonImg": "../../native/default/svg/icon-whiteboard.svg",
        "buttonPos": 1
    },
    {
        "buttonId": "customMore",
        "title": "自定义更多",
        "buttonImg": "../../native/default/svg/icon-whiteboard.svg",
        "buttonPos": 0
    },
    {
        "buttonId": "customShare",
        "title": "自定义共享",
        "buttonImg": "../../native/default/svg/icon-whiteboard.svg",
        "buttonPos": 2
    }
    ]
}
}
```



- **删除部分自定义按钮**

customButton删除对应按钮，同时需要保留其他自定义按钮。以删除底部工具栏自定义按钮为例：

```
{
  "frame": {
    "confMenu": {
      "toolBar": {
        "customButton": [
          {
            "buttonId": "customMore",
            "title": "自定义更多",
            "buttonImg": "../../native/default/svg/icon-whiteboard.svg",
            "buttonPos": 0
          },
          {
            "buttonId": "customShare",
            "title": "自定义共享",
            "buttonImg": "../../native/default/svg/icon-whiteboard.svg",
            "buttonPos": 2
          }
        ]
      }
    }
  }
}
```



- **清空某个位置下的自定义按钮**

customButton中删除该位置下的所有自定义按钮即可，若customButton下未配置某位置下的自定义按钮，则会清空该位置下的自定义按钮。以底部工具栏为例：

```
{
  "frame": {
    "confMenu": {
      "toolBar": {
        "customButton": [
          {
            "buttonId": "customMore",
            "title": "自定义更多",
            "buttonImg": "../../native/default/svg/icon-whiteboard.svg",
            "buttonPos": 0
          },
          {
            "buttonId": "customShare",
            "title": "自定义共享",
            "buttonImg": "../../native/default/svg/icon-whiteboard.svg",
            "buttonPos": 2
          }
        ]
      }
    }
  }
}
```



- **清空全量自定义按钮**

customButton传入空数组，会清空所有位置下自定义按钮。清空示例如下：

```
{
  "frame": {
    "confMenu": {
      "toolBar": {
        "customButton": [
        ]
      }
    }
  }
}
```



9.7.3 常见错误码

表 9-337 SDKError 枚举说明

变量	值	描述
UISDK_COMMON_ERROR	43010001	通用错误
UISDK_API_INVALID_PARAMETERS	43010002	参数错误
USG_ERR_PASSWORD	20601000	帐号或密码错误
SDK_LOGIN_ACCOUNT_LOCKEDwx1116180	41210010	帐号被锁定
SDK_LOGIN_AUTH_ACCOUNT_DIACTIVE	41210015	帐号或企业未激活
USG_APPID_AUTH_EXPIRE	20601057	APPID认证过期
USG_APPID_ACCOUNT_DISABLED	20601007	帐号未激活,无权限登录
CMS_CONF_AUTHEN_NEED_PASSWORD	11107009	会议密码不能为空
CMS_CONF_NOT_FOUND_OR_AUTH_FAILED	111072065	会议不存在或密码错误
CMS_CONF_LOCKED	111072050	会议被锁定
UISDK_NOT_IN_CONF_OR_CALL	43040008	不在会议或呼叫中
CMS_VMR_ONLY_ALLOW_CHAIR_START_CONF	111071004	VMR会议不允许来宾在主席入会前入会
UISDK_SHARE_LOCKED	43040015	共享被锁定
UISDK_NOT_IN_DATACONF	43040016	不在数据会议中

变量	值	描述
CMS_CONF_ONLY_ALLOW_USER_IN_OR G	111071 074	会议只允许企业内用户呼入
CMS_CONF_ONLY_ALLOW_USER_INVITE D	111071 075	会议只允许被邀请用户呼入
UISDK_ALREADY_EXIST_CONF_OR_CALL	430400 009	已存在一路会议或呼叫
CMS_CONF_ANONYMOUS_LOGIN_LOCK ED	111071 052	匿名登录锁定,请稍后重试
CMS_CONF_LINK_LOGIN_LOCKED	111071 053	链接登录锁定,请稍后重试
CMS_CONF_IS_CLOSED	111072 006	会议已结束
USG_VMR_PWD_SAME_AS_CHAIRMAN_P WD	201080 006	主席密码和来宾密码不能相同 请确认输入参数
CMS_CONF_PORTAL_PARTICIPANT_VMR_ LOGIC_RESOURCE_NOT_ENOUGH	111076 003	超出VMR最大方数
CMS_CONF_PORTAL_PARTICIPANT_RESO URCE_NOT_ENOUGH	111076 004	企业并发数不足
CMS_CONF_NOT_ON_LOCAL_STATION	111100 012	会议不在本站点
CMS_LOGICRES_PNUM_RESOURCE_NOT_ ENOUGH	115022 001	企业并发资源不足
CMS_CONF_DATA_NOT_FOUND	111070 005	会议不存在
CMS_VMR_DATA_NOT_FOUND	111070 006	个人云会议室不存在
CMS_CONF_ROLE_AUTHENTICATION_FA ILED	111070 010	权限不足,无法查看会议
CMS_ONLINE_CONF_IN_COMMUNAL	111071 106	云会议室已有会议正在召开
CMS_CONF_MODIFY_FAIL_AS_CONF_ALR EADY_STARTED	111071 065	正在召开会议,无法编辑会议详情
CMS_ORG_CONCURRENT_PARTICIPANTS _NUMBER_IS_ZERO	111071 060	企业未配置并发会议资源,无法使用个人会议ID和随机会议ID发起会议
CMS_CONF_USER_NOT_VMR_OWNER	111071 063	非云会议室拥有者

变量	值	描述
CMS_CONF_CANCEL_FAIL_AS_CONF_STARTED	111071067	会议正在召开中，不允许取消会议
CMS_CONF_START_TIME_LESS_THAN_CURRENT_TIME	111071013	会议开始时间不能早于系统当前时间
CMS_CONF_START_TIME_OVER_MAX_SCHEDULE_TIME	111071014	会议开始时间超出最大预订时间范围
USG_VMR_PWM_SAME_AS_MEETING_PWD	201080017	和会议密码相同
CMS_PARTICIPANT_CONF_NOT_SUPPORTED	111084013	不支持该会议
CMS_CONF_ATTENDEES_NUMBER_EXCEEDED_LIMIT	111071066	与会方数量超出上限
CMS_VMR_DATA_STATUS_ILLEGAL	111073007	VMR套餐状态非法
CMS_WAIT_APPLY_RESOURCE	115029006	等待申请资源中
SDK_SVN_DETECT_FAILED	411000017	svn端口探测不通
SDK_ERR_RTC_NOT_SUPPORT_STG	411000019	RTC不支持STG模式
CMS_OVER_MAX_PARTICIPANT_NUMBER	111072101	超过设置的会议方数
CMS_AUDIENCE_VMR_LOGIC_RESOURCE_NOT_ENOUGH	111072202	观众数超过VMR套餐中的观众最大并发资源数
CMS_PARTICIPANT_VMR_LOGIC_RESOURCE_NOT_ENOUGH	111072203	主持人与嘉宾超过VMR套餐中的最大并发资源数
CMS_PARTICIPANT_CONF_NOT_EXIST	111081002	会场所在会议不存在或已结束
USG_VMR_RESOURCE_UNAVAILABLE	201080000	云会议资源不可用
USG_ACCOUNT_IS_UNBOUND	206010075	账号已经解绑
RTC_SDK_JOIN_ROOM_FAIL	990000018	加入房间失败
RTC_SDK_JOIN_ROOM_STATUS_BUSY	990000019	非空闲状态
RTC_SDK_JOIN_ROOM_SERVER_ERROR	990000020	加入房间失败，服务器异常

变量	值	描述
RTC_SDK_JOIN_ROOM_AUTH_FAIL	990000 022	加入房间失败，鉴权失败
RTC_SDK_JOIN_ROOM_URL_NOT_RIGHT	990000 025	加入房间失败，url错误
SDK_LOGIN_CA_VERIFY_FAILED	412100 012	服务器CA证书校验失败
SDK_LOGIN_CURLE_EXTERN_CONNECT_TIMEOUT	412100 017	代理通信超时，tcp建链失败，大概率是代理配置错误
SDK_LOGIN_CURLE_EXTERN_CLOSED_BY_PROXY	412100 018	链接被代理断开，大概率是服务器地址配置错误
SDK_LOGIN_CURLE_COULDNT_RESOLVE_PROXY	412100 024	认证鉴权过程失败，无法解析网关
SDK_LOGIN_CURLE_COULDNT_RESOLVE_HOST	412100 025	认证鉴权过程失败，无法解析主机
UISDK_TIMEOUT	430100 004	超时
UISDK_IN_PROCESS	430100 005	接口正在处理中
UISDK_UNLOGIN_ERROR	430200 001	未登录
UISDK_NOT_INITED_END	430100 006	未初始化或初始化尚未完成
UISDK_NOT_SUPPORT_END	430100 007	此版本不支持
UISDK_OPERATION_INTERRUPT	430100 008	操作中断
UISDK_CONTACT_NOT_FOUND	430100 009	联系人不存在
UISDK_ARGS_LENGTH_OVERRUN	430100 010	参数长度超限
UISDK_APPID_INFO_ILLEGAL	430100 011	APPID 信息入参不合法
UISDK_E2EENCRYPT_ALGORITHM_ERROR	430100 012	端到端加密算法不对称
UISDK_REPEATED_LOGIN_ERROR	430200 002	重复登录
UISDK_PASSWORD_OR_ACCOUNT_EMPTY	430200 005	账号或密码不能都为空

变量	值	描述
UISDK_CALL_ACCOUNT_EMPTY	43030002	点对点呼叫账号为空
UISDK_CALL_NICKNAME_EMPTY	43030003	点对点呼叫名字为空
UISDK_INVALID_GUEST_PASSWORD_PARAMETER	43040007	来宾密码非法
UISDK_CONFID_EMPTY	43040010	会议id不能为空
UISDK_SUBJECT_TOO_LONG	43040011	会议主题长度不能超过50
UISDK_VMR_HOST_PASSWORD_LENGTH_ILLEGAL	43040012	主持人密码长度不合法
UISDK_VMR_NAME_EMPTY	43040013	VMR名字为空

9.8 版本更新内容

9.8.1 80.13.6 更新内容

1. 问题修复，体验更稳定。

9.8.2 80.13.5 更新内容

1. 设置语言接口setLanguage新增自定义语言功能（仅win支持），具体详见[设置语言](#)。

9.8.3 80.12.7 更新内容

1. 问题修复，体验更稳定。

9.8.4 80.12.6 更新内容

1. 功能配置开关接口enableCapability新增是否显示录制安全提示窗枚举值HWM_CAPABILITY_CONFIRM_RECORD，具体详见[功能配置开关](#)。
2. 问题修复，体验更稳定。

9.8.5 80.12.5 更新内容

1. 上传头像接口上传条件变更（增加类型和大小判断），win支持上传头像接口，详见[上传头像](#)。
2. 问题修复，体验更稳定。

9.8.6 80.10.5 更新内容

1. 获取VMR列表接口和修改云会议室信息接口新增参数confAllowJoinUser、isOpenWaitingRoom，具体详见

[获取VMR列表](#)和[修改云会议室信息](#)。

2. 新增设置共享权限接口SetShareScreenPermission，具体详见[设置共享权限](#)。

3. 加入会议接口新增参数isStopConflictConf，具体详见[加入会议](#)。

4. 获取会议详情新增参数vmrConfIdType，具体详见[获取会议详情](#)。

5. Appld登录接口新增参数deptCode，具体详见[Appld登录](#)。

9.8.7 80.9.5 更新内容

1. Mac补充获取会议详情、编辑会议接口新增参数allowLive，具体详见[获取会议详情](#)、[编辑会议](#)。

2. 新增周期会议相关接口，具体详见[预约周期会议](#)、[编辑周期会议](#)、[编辑周期子会议](#)、[取消周期会议](#)、[取消周期子会议](#)。

3. 功能配置开关接口新增分组讨论功能开关枚举值HWM_CAPABILITY_BREAKOUT_CONF，具体请见[功能配置开关](#)。

4. 会议信息通知新增参数isInBreakoutSubConf，具体详见[会议信息通知](#)。Mac在下一个版本支持。

5. Electron SDK补充常见错误码，具体详见[常见错误码](#)。

9.8.8 80.8.5 更新内容

1. 获取会议详情、编辑会议接口新增参数allowLive，具体详见[获取会议详情](#)、[编辑会议](#)。Mac在下一个版本支持。

2. 会议信息通知新增参数isCreator，具体详见[会议信息通知](#)。

3. 预约会议、编辑会议支持设置“来宾在主持人之前加入会议”，并可设置提前入会时间（allowGuestStartConf、allowGuestStartConfTime），具体详见[预约会议](#)、[编辑会议](#)。

9.8.9 80.7.5 更新内容

1. 新增显示设置界面接口showSettingWindow，接口详见[显示设置窗口](#)。

2. 会议信息通知OnConfInfoNotify新增参数isOtherCorpConf，具体详见[会议信息通知](#)。

3. 与会者列表子项注入点击通知OnClickInjectParticipantItemBtnNotify新增与会者详情按钮类型HWM_CLICK_INJECT_PARTICIPANT_ITEM_BTN_PARTICIPANT_DETAIL，新增华为云会议帐号参数accountId，具体详见[与会者列表子项注入点击通知](#)。

4. Config接口新增字段participant_detail，支持与与会者头像点击注入，详见[界面配置json说明](#)。

9.8.10 80.5.11 更新内容

1. config接口新增字段statusBar，支持录制状态栏按钮可隐藏，详见[界面配置json说明](#)。
2. 新增禁用提示音接口disableVoicePrompts，接口详见[禁用提示音](#)。

9.8.11 80.5.5 更新内容

1. Mac补充exit接口，具体请见[exit接口](#)。
2. Mac创建会议、预约会议、编辑会议、获取会议详情接口补充等候室开关功能，具体请见[创建会议](#)、[预约会议](#)、[编辑会议](#)、[获取会议详情](#)。
3. 功能配置开关接口新增隐藏非视频与会者配置，具体请见[功能配置开关](#)。
4. 会议结束通知，新增服务器异常错误码，具体请见[会议结束通知](#)。

10 SDK 已知安全漏洞修复说明

表 10-1 SDK 已知安全漏洞及修复版本

安全漏洞描述	涉及SDK	修复版本	版本发布时间
更新OpenSSL版本到1.1.1.n, 修复漏洞: CVE-2022-0778 CVE-2021-4160	Android SDK/iOS SDK/ Windows SDK/Mac SDK/ Electron SDK	80.9.5	2022-10-15
更新curl版本到7.84.0, 修复漏洞: CVE-2022-32207 CVE-2022-32206 CVE-2022-32208 CVE-2022-32205	Android SDK/iOS SDK/ Windows SDK/Mac SDK/ Electron SDK		
更新wxSQLite3版本到4.8.2, 修复漏洞: CVE-2022-35737	Android SDK/iOS SDK/ Windows SDK/Mac SDK/ Electron SDK		

11 常见问题

11.1 华为云会议资源类型

华为云会议资源有两种类型：

1. 并发会议：如果企业购买的资源是会议并发数，每个会议中的与会人数可以随意组合，只要所有会议的在线总与会人数不超过企业的并发数即可。单场会议，视频与会人数不超过500方。
2. 云会议室会议：如果企业购买的资源是云会议室，每个会议室的与会人数是固定的（10方/50方/100方/200方/300方/400方/500方）。比如购买了3个100方的云会议室资源，最多只能同时开3场会，哪怕每场会的实际参会人数都不足100方。

两种会议资源可以组合购买，创建会议时需要指定会议开在哪种资源上（不指定默认是开在并发会议资源上）。更多参考场景请见：[推荐购买方案](#)。

11.2 如何获取 SDK 的日志

获取 Android SDK 的日志

如果SDK初始化时没有修改过日志存放路径，默认的日志路径在“/sdcard/Android/data/应用的包名/files/log下”，可以使用设备上的文件管理工具将整个log目录打包。也可以通过adb获取日志文件，步骤如下：

步骤1 连接手机USB并打开开发者模式开关，打开cmd，输入adb devices查看是否已连接设备

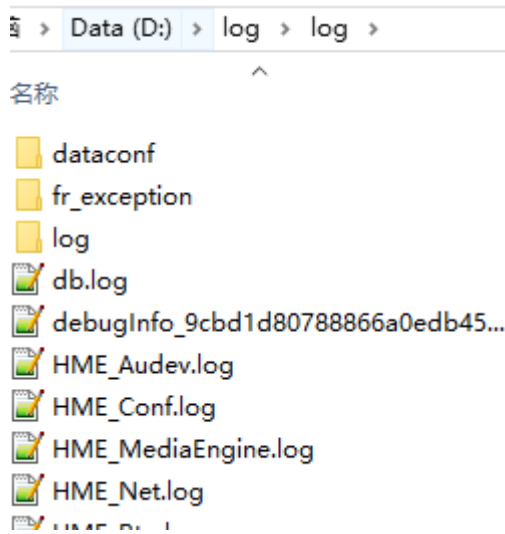
步骤2 输入adb pull 日志路径 目标路径，并回车。

图 11-1 命令行取日志

```
C:\Users\pwx178217>adb pull /storage/sdcard0/Android/data/com.huawei.hicloud.sdk/files/log d:\log
/storage/sdcard0/Android/data/com.huawei.hicloud.sdk/files/log/: 46 files pulled. 33.1 MB/s (28781983 bytes in 0.830s)
```

步骤3 找到D盘目录，SDK的日志在d:\log\log下。

图 11-2 日志目录

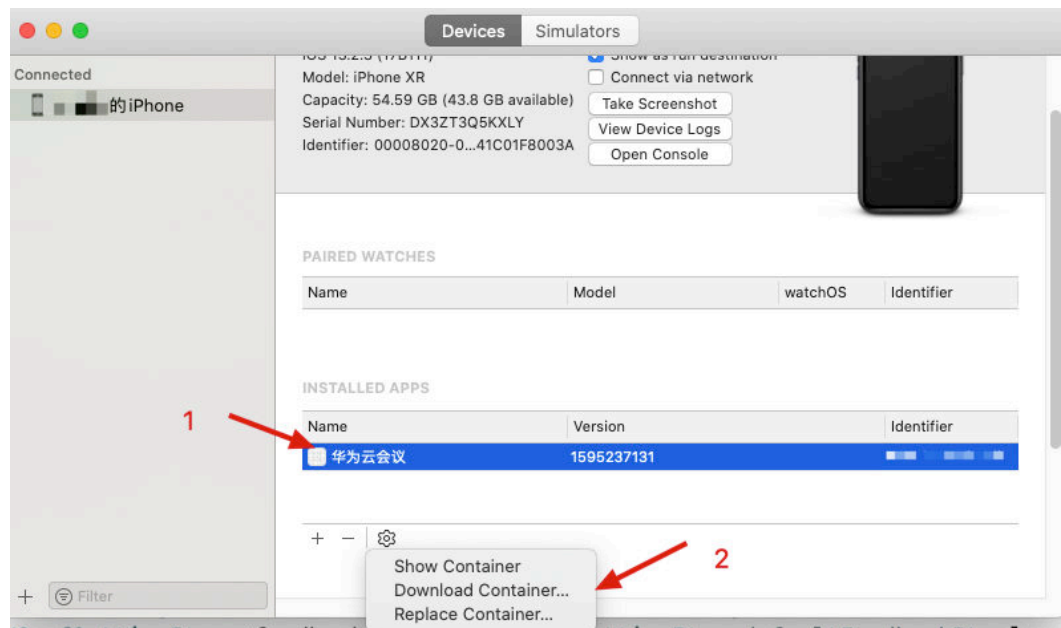


----结束

获取 iOS SDK 的日志

步骤1 打开Xcode ->Window -> Devices and Simulators

图 11-3 下载文件



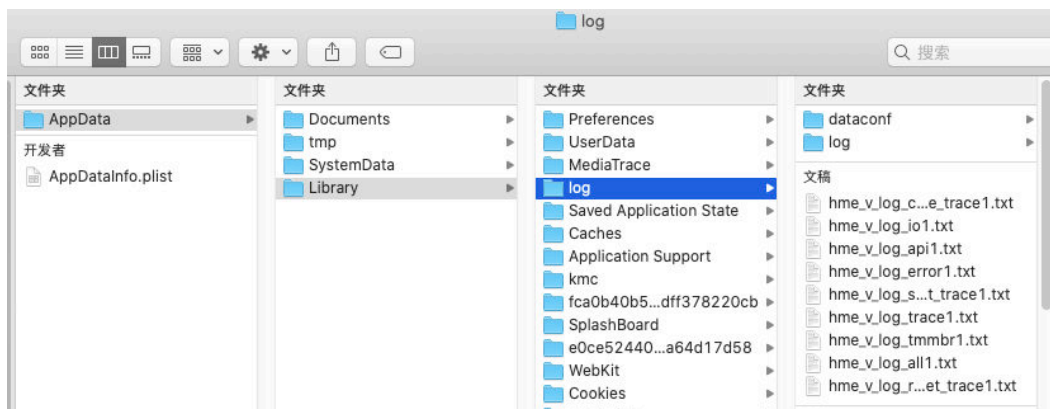
步骤2 1中下载到的文件，鼠标右键显示选择显示包内容

图 11-4 显示包内容



步骤3 在显示的包内容中查看日志

图 11-5 日志路径



----结束

获取 Windows SDK 的日志

注：如已自定义日志路径，则前往自定义路径查找。如使用默认日志路径，则按如下步骤获取日志。

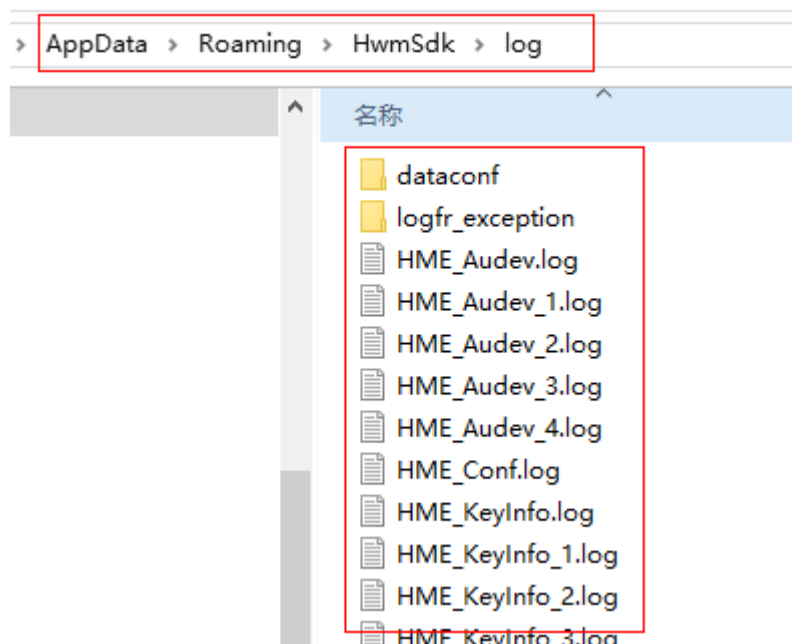
步骤1 打开“文件资源管理器”，并输入%appdata%，并回车。

图 11-6 文件资源管理器



步骤2 找到HwmSdk目录，SDK的日志在HwmSdk\log下。

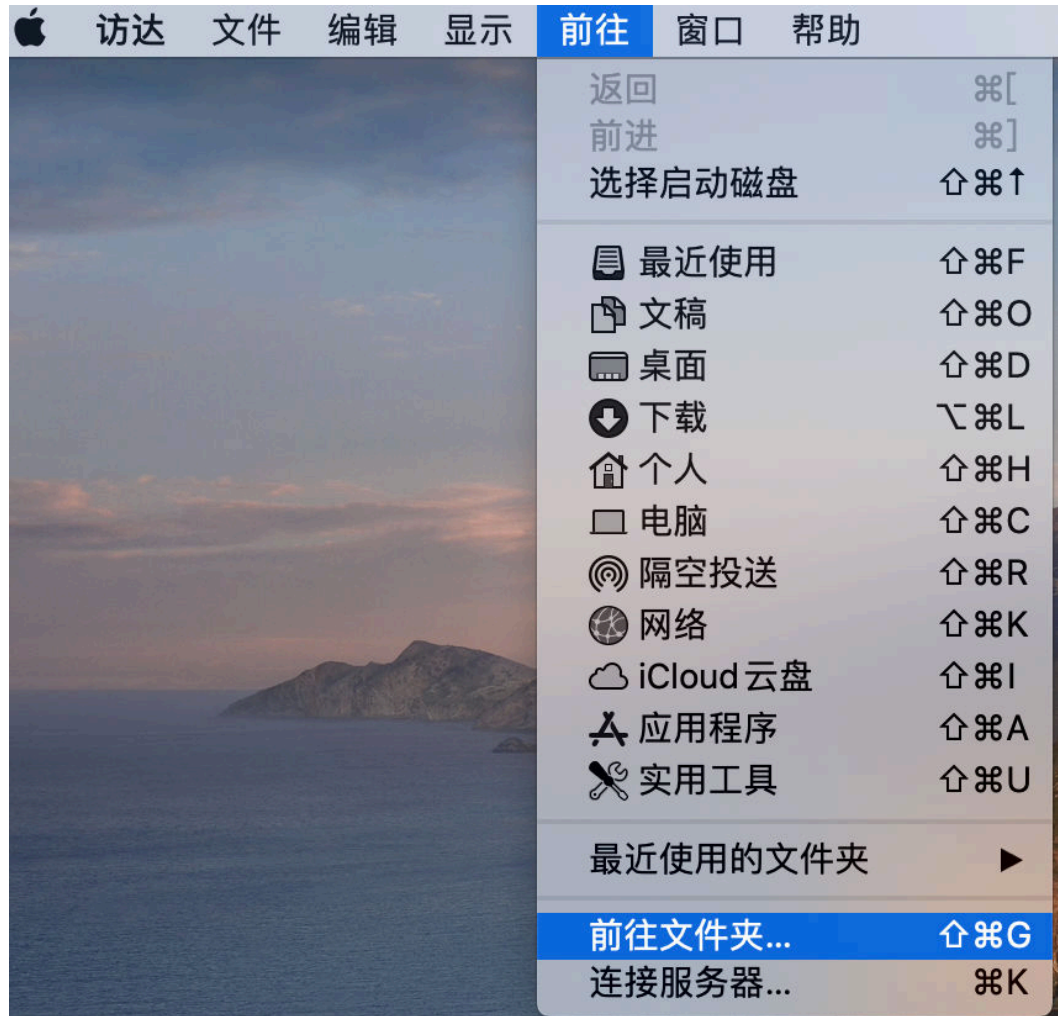
图 11-7 HwmSdk 目录



----结束

获取 Mac SDK 的日志

步骤1 打开 访达 -> 前往 -> 前往文件夹



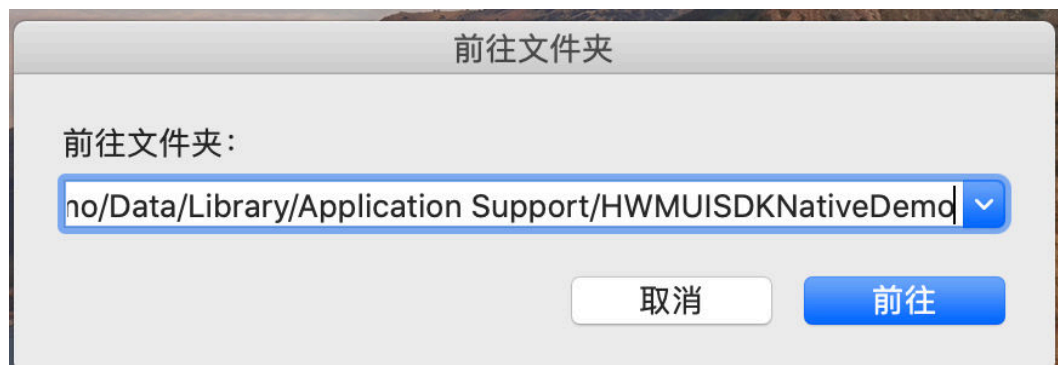
步骤2 输入路径

开启 app sandbox，日志路径

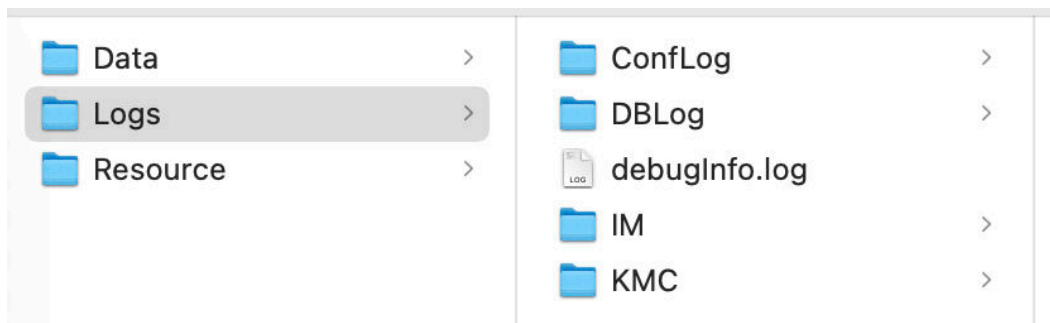
```
~/Library/Containers/com.huawei.cloud.welinkdemo/Data/Library/Application Support/HWMUISDKNativeDemo
```

关闭 app sandbox，日志路径

```
~/Library/Application Support/HWMUISDKNativeDemo
```



步骤3 找到并打开Logs文件夹，查找对应模块的日志

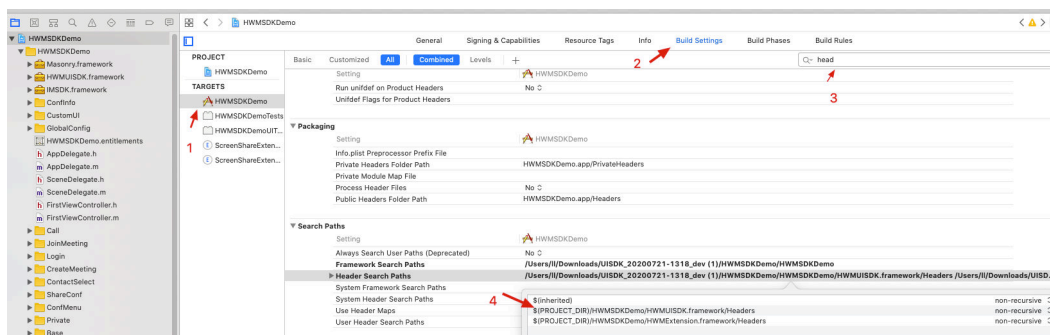


----结束

11.3 【iOS】如何用#import"xxx.h"导入 iOS SDK 中的文件

配置HWMUISDK.framework真实路径，即可使用#import"xxxx.h"导入文件。

图 11-8 配置路径，图示为示例路径，根据导入 framework 真实路径配置。



11.4 【iOS】实现通知方法不生效

请查看[注意事项](#)，[使用通知前请查看](#)。

如果上述注意事项不能解决问题请使用代理方法前打印查看,如果不能打印出代理对象方法存在，[请查找代码原因](#)

```
// 示例代码，根据自己情况查看对应对象
id <HWMNotifyHandler> globalHandler = [HWM SDK getSdkConfig].globalHandler;
// 输出打印代理对象
NSLog(@"查看是否存在代理对象: %@", globalHandler);
// 查看代理对象的方法是否存在
if(globalHandler && [globalHandler respondsToSelector:@selector(onKickedOut:)]) {
    NSLog(@"代理对象方法存在");
}
```

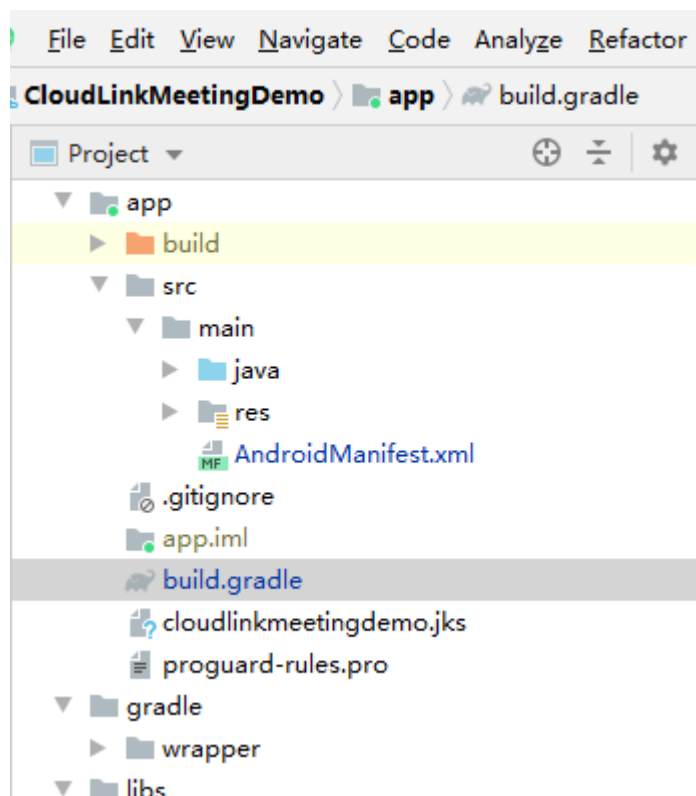
11.5 【iOS】iOS 导入 AFNetworking 报错

General->HWMUISDKDemo->Framework,Libraries,and Embedded content;添加
CFNetwork, SystemConfiguration, MobileCoreServices, libz,

11.6 【Android】支持 64 位库

步骤1 打开工程目录下build.gradle

图 11-9 工程目录



步骤2 在build.gradle中添加

```
ndk {  
    abiFilters "armeabi-v7a", "arm64-v8a"  
}
```

图 11-10 支持 32 位和 64 位库

```
defaultConfig {  
    applicationId "com.huawei.cloudlinkmeetingdemo"  
    minSdkVersion 21  
    targetSdkVersion 26  
    versionCode 1  
    versionName "1.0.0"  
    multiDexEnabled true  
    ndk {  
        abiFilters "armeabi-v7a", "arm64-v8a"  
    }  
}
```

图 11-11 只支持 32 位

```
defaultConfig {  
    applicationId "com.huawei.cloudlinkmeetingdemo"  
    minSdkVersion 21  
    targetSdkVersion 26  
    versionCode 1  
    versionName "1.0.0"  
    multiDexEnabled true  
    ndk {  
        abiFilters "armeabi-v7a"  
    }  
}
```

图 11-12 只支持 64 位

```
defaultConfig {
    applicationId "com.huawei.cloudlinkmeetingdemo"
    minSdkVersion 21
    targetSdkVersion 26
    versionCode 1
    versionName "1.0.0"
    multiDexEnabled true
    ndk {
        abiFilters "arm64-v8a"
    }
}
```

----结束

11.7 【 Android 】如何去掉不需要的 aar 依赖

部分模块在自己注入或者不打开的情况下，可以不添加依赖，并且删除对应aar
包括：

- MobileHybrid模块[自定义会议中加入](#)
- imsdk模块[自定义开关会中聊天功能](#)

11.8 【 Android 】使用第三方加固软件后，路由跳转错误

因为我们用到了类似ARouter的路由方案，如果加固后出现路由跳转错误，解决方案如下：

- **永久解决方案：**联系第三方加固软件的技术支持，把我们的路由自动生成的类所在的包给过滤掉，包路径：com.huawei.cloudlink.router.routermap。
- **临时解决方案：**详见下图；注：如使用此方案，每次SDK发版都需要重新设置一次。

图 11-13 临时解决方案

The image shows a file explorer view of an Android project. The file tree is as follows:

File	Raw File Size	Download Size	% of Total Download
res	4.2 MB	4.2 MB	76.5%
classes.jar	1.2 MB	1.2 MB	22.4%
com	1.2 MB	1.2 MB	21.6%
huawei	1.2 MB	1.2 MB	21.6%
hwmconf	1 MB	1 MB	18.9%
hwmchat	150.2 KB	150.1 KB	2.7%
cloudlink	818 B	818 B	0%
router	818 B	818 B	0%
routermap	818 B	818 B	0%
RouterBinding_1605260221227.class	818 B	818 B	0%
hwmmobileconfui	818 B	818 B	0%
R.txt	461 B	461 B	0%
AndroidManifest.xml	65.1 KB	61.8 KB	1.1%
xml	706 B	702 B	0%
values			0%
raw			0%
mipmap-xxhdpi-v4			0%
layout			0%
drawable-xxhdpi-v4			0%
drawable-xhdpi-v4			0%
drawable-xhdpi-v4			0%
drawable-mdpi-v4			0%
drawable			0%
color			0%
anim			0%

Annotations for the file explorer:

1. 解压HWMobileConfUI.aar
2. 反编译里面的classes.jar查找com.huawei.cloudlink.router.routermap下的类，找到类名

```

try {
    HashMap<String, String> map = new HashMap<>();
    Set<String> classNames = new HashSet<>();
    classNames.add("com.huawei.cloudlink.router.routermap.RouterBinding_1605258662203");
    for (String className : classNames) {
        Class bindViewClass = Class.forName(className);
        Method method = bindViewClass.getMethod("bind");
        Object object = method.invoke(bindViewClass.newInstance());
        if (object instanceof HashMap) {
            HashMap tempMap = (HashMap) object;
            map.putAll(tempMap);
        }
    }

    Field privateMapField = RouterMap.class.getDeclaredField("map");
    privateMapField.setAccessible(true);
    privateMapField.set(null, map);
} catch (NoSuchFieldException | ClassNotFoundException | NoSuchMethodException e) {
    e.printStackTrace();
} catch (IllegalAccessException e) {
    e.printStackTrace();
} catch (InstantiationException e) {
    e.printStackTrace();
} catch (InvocationTargetException e) {
    e.printStackTrace();
}
    
```

Annotations for the code snippet:

3. 将得到的类名替换选择框类的类名
4. 将这段代码，放在sdk初始化之前

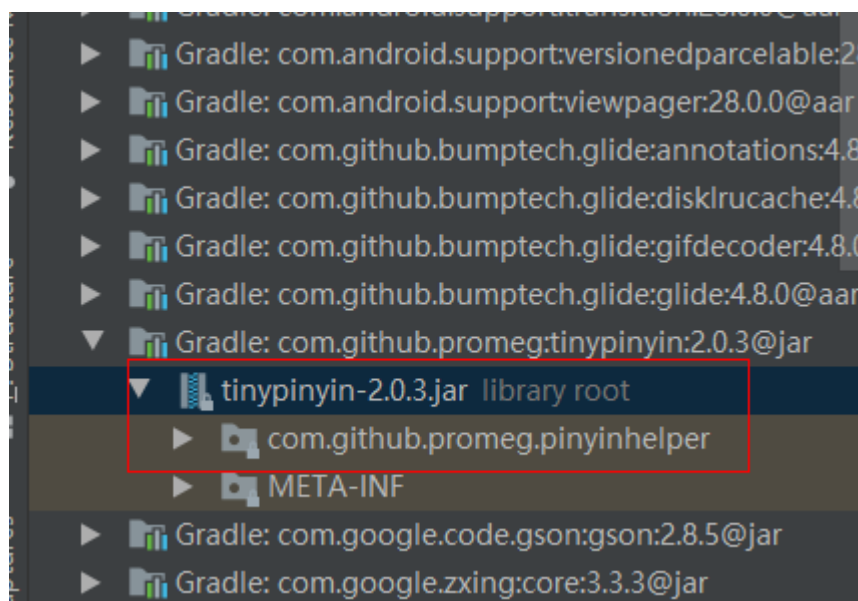
11.9 【 Android 】 SDK 提示语不支持中英文切换

解决方案：检查是否启用了resConfigs，如果启用，需要带上中英文：
resConfigs("zh", "zh-rCN", "en")

11.10 【 Android 】 SDK 初始化失败，调用创会、加入会议等接口没有回调也没有提示

解决方案：检查是否成功下载tinypinyin-2.0.3.jar，如果没有则下载到本地并引用到工程中。

图 11-14 tinypinyin-2.0.3.jar 文件



11.11 【Win】第三方客户端调用 Exit 接口，需要等待一段时间后才能退出

客户案例：

某客户集成了会议Win SDK，客户应用退出后，发现会议进程仍在运行，exit回调时间很长。

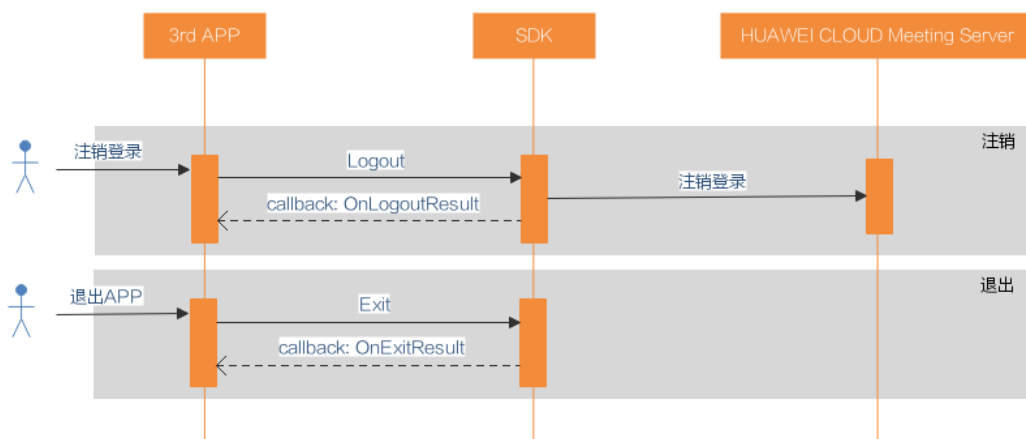
原因分析：

该客户在logout的回调里面调用exit接口，exit接口要等待logout的回调结束才能销毁资源，logout的回调在等exit退出，造成线程死锁，导致如上现象。

正确方案：

在logout回调结束后，再调用exit接口，详见下图。点击查看[更多接口调用时序图](#)。

图 11-15 接口调用时序图



11.12 【 Electron 】运行 Electron demo 时遇 electron failed to install correctly 的错误提示

客户案例:

某客户集成了会议Electron Sdk，运行启动时提示electron failed to install correctly。

```
> electron . --auto

/Users/p00576758/mac/222/hwmsdk-electron/demo/node_modules/electron/index.js:17
  throw new Error('Electron failed to install correctly, please delete node_modules/electron and try installing again');
  ^

Error: Electron failed to install correctly, please delete node_modules/electron and try installing again
    at getElectronPath (/Users/p00576758/mac/222/hwmsdk-electron/demo/node_modules/electron/index.js:17:11)
    at Object.<anonymous> (/Users/p00576758/mac/222/hwmsdk-electron/demo/node_modules/electron/index.js:21:18)
    at Module._compile (node:internal/modules/cjs/loader:1103:14)
    at Object.Module._extensions..js (node:internal/modules/cjs/loader:1157:10)
    at Module.load (node:internal/modules/cjs/loader:981:32)
    at Function.Module._load (node:internal/modules/cjs/loader:822:12)
    at Module.require (node:internal/modules/cjs/loader:1005:19)
    at require (node:internal/modules/cjs/helpers:102:18)
    at Object.<anonymous> (/Users/p00576758/mac/222/hwmsdk-electron/demo/node_modules/electron/cli.js:3:18)
    at Module._compile (node:internal/modules/cjs/loader:1103:14)
```

原因分析:

造成原因一般是用户本地node版本和安装的electron版本不配套导致。

解决方案:

前往node官网下载electron版本建议的node版本（配套关系参见<https://www.electronjs.org/zh/docs/latest/tutorial/electron-timelines>）替换安装后，重新编译启动工程即可解决。

11.13 【 Electron 】mac 平台环境安装时遇 permission denied 的错误提示

客户案例:

某客户集成了会议Electron Sdk，在安装所需环境时提示permission denied。

```
> node install.js

Downloading electron-v13.6.9-darwin-x64.zip: [=====] 100% ETA: 0.0 seconds
Error: EACCES: permission denied, mkdir '/Users/p00576758/Library/Caches/electron/8236d1ab05b1110f7e0f0e480467a10a2fe0646765fceab593d4e085ca70
01e'
npm WARN @babel/plugin-bugfix-v8-spread-parameters-in-optional-chaining@7.16.7 requires a peer of @babel/core@^7.13.0 but none is installed. You
u must install peer dependencies yourself.
npm WARN @babel/plugin-proposal-class-static-block@7.17.6 requires a peer of @babel/core@^7.12.0 but none is installed. You must install peer d
ependencies yourself.
npm WARN tsutils@3.21.0 requires a peer of typescript@>=2.8.0 || >= 3.2.0-dev || >= 3.3.0-dev || >= 3.4.0-dev || >= 3.5.0-dev || >= 3.6.0-dev |
| >= 3.6.0-beta || >= 3.7.0-dev || >= 3.7.0-beta but none is installed. You must install peer dependencies yourself.
npm WARN hwmsdk-electron-demo@0.1.0 No repository field.
npm WARN hwmsdk-electron-demo@0.1.0 No license field.

npm ERR! code ELIFECYCLE
npm ERR! errno 1
npm ERR! electron@13.6.9 postinstall: `node install.js`
npm ERR! Exit status 1
npm ERR!
npm ERR! Failed at the electron@13.6.9 postinstall script.
npm ERR! This is probably not a problem with npm. There is likely additional logging output above.

npm ERR! A complete log of this run can be found in:
npm ERR!   /Users/p00576758/.npm/_logs/2022-04-07T12_13_03_860Z-debug.log
```

原因分析:

文件权限问题。

解决方案:

使用chmod命令修改对应报错文件的权限，如：chmod 777 XXX；或者在安装对应依赖包时添加参数：--unsafe-perm=true --allow-root，

如 `npm install -electron@13.6.7 -g --unsafe-perm=true --allow-root`

11.14 【 Electron 】 mac 平台编译.node 文件时遇 no template named 'remove_cv_t'的错误提示

客户案例:

某客户集成了会议Electron Sdk，在编译.node文件时提示no template named 'remove_cv_t'。

```
/Users/p00576758/Library/Caches/node-gyp/13.6.7/include/node/v8-internal.h:454:38: error: no template named 'remove_cv_t' in namespace 'std';  
did you mean 'remove_cv'?  
    !std::is_same<Data, std::remove_cv_t<T>>::value>::Perform(data);  
                                ~~~~~^~~~~~  
                                remove_cv  
/Applications/Xcode.app/Contents/Developer/Toolchains/XcodeDefault.xctoolchain/usr/bin/../include/c++/v1/type_traits:697:50: note: 'remove_cv'  
declared here  
template <class _Tp> struct _LIBCPP_TEMPLATE_VIS remove_cv  
                                ^  
                                A
```

原因分析:

造成原因是remove_cv_t从c++14才开始支持，而node依赖文件binding.gyp中指定的c++版本还是c++11。

解决方案:

首先，建议node版本和electron版本要匹配安装，前往node官网下载electron版本建议的node版本（配套关系参见<https://www.electronjs.org/zh/docs/latest/tutorial/electron-timelines>）。其次，如果本问题依然存在，可从如下两种解决方案选择一种进行修复：

1.到对应报错文件v8-internal.h中，将remove_cv_t修改为remove_cv。

2.将binding.gyp中“OTHER_CPLUSPLUSFLAGS”的配置项“-std=c++11”修改为“-std=c++14”。

11.15 【 Electron 】 windows 平台编译.node 文件时遇 Could not find any Python installation to use 的错误提示

客户案例:

windows平台，在编译.node文件时提示Could not find any Python installation to use。

```
Error: Could not find any Python installation to use  
  
Error: node-gyp failed to rebuild 'D:\js uisdk\hwmsdk-electron-70.16.5-src-x64\hwmsdk-electron'.  
For more information, rerun with the DEBUG environment variable set to "electron-rebuild".  
  
Error: Could not find any Python installation to use  
  
    at NodeGyp.rebuildModule (D:\js uisdk\hwmsdk-electron-70.16.5-src-x64\hwmsdk-electron\node_modules\electron-rebuild  
lib\src\module-type\node-gyp.js:117:19)  
    at processTicksAndRejections (internal/process/task_queues.js:93:5)  
    at async ModuleRebuilder.rebuildNodeGypModule (D:\js uisdk\hwmsdk-electron-70.16.5-src-x64\hwmsdk-electron\node_mod  
les\electron-rebuild\lib\src\module-rebuilder.js:94:9)  
    at async ModuleRebuilder.rebuild (D:\js uisdk\hwmsdk-electron-70.16.5-src-x64\hwmsdk-electron\node_modules\electron  
rebuild\lib\src\module-rebuilder.js:124:14)  
    at async Rebuilder.rebuildModuleAt (D:\js uisdk\hwmsdk-electron-70.16.5-src-x64\hwmsdk-electron\node_modules\electr  
n-rebuild\lib\src\rebuild.js:145:13)  
    at async Rebuilder.rebuild (D:\js uisdk\hwmsdk-electron-70.16.5-src-x64\hwmsdk-electron\node_modules\electron-rebuil  
d\lib\src\rebuild.js:108:17)  
    at async D:\js uisdk\hwmsdk-electron-70.16.5-src-x64\hwmsdk-electron\node_modules\electron-rebuild\lib\src\cli.js:11  
4:9
```


原因分析:

python环境变量没有配置好。

解决方案:

1. 系统中配置python环境变量
2. 或者重新安装python，选择自定义安装，勾选 Add Python to PATH 和 install for all user users

11.16 【 Electron 】 windows 平台编译.node 文件时遇 Could not find any Visual Studio installation to use 的 错误提示

客户案例:

windows平台，在编译.node文件时提示Could not find any Visual Studio installation to use。

```
An unhandled error occurred inside electron-rebuild
node-gyp failed to rebuild 'D:\jsuisdk\hwmsdk-electron-70.16.5-src\hwmsdk-electron'.
For more information, rerun with the DEBUG environment variable set to "electron-rebuild".
Error: Could not find any Visual Studio installation to use

Error: node-gyp failed to rebuild 'D:\jsuisdk\hwmsdk-electron-70.16.5-src\hwmsdk-electron'.
For more information, rerun with the DEBUG environment variable set to "electron-rebuild".
Error: Could not find any Visual Studio installation to use

    at NodeGyp.rebuildModule (D:\jsuisdk\hwmsdk-electron-70.16.5-src\hwmsdk-electron\node_modules\electron-rebuild\lib\src\module-type\node-gyp.js:117:19)
    at processTicksAndRejections (internal/process/task_queues.js:93:5)
    at async ModuleRebuilder.rebuildNodeGypModule (D:\jsuisdk\hwmsdk-electron-70.16.5-src\hwmsdk-electron\node_modules\electron-rebuild\lib\src\module-rebuilder.js:94:9)
    at async ModuleRebuilder.rebuild (D:\jsuisdk\hwmsdk-electron-70.16.5-src\hwmsdk-electron\node_modules\electron-rebuild\lib\src\module-rebuilder.js:124:14)
    at async Rebuilder.rebuildModuleAt (D:\jsuisdk\hwmsdk-electron-70.16.5-src\hwmsdk-electron\node_modules\electron-rebuild\lib\src\rebuild.js:145:13)
    at async Rebuilder.rebuild (D:\jsuisdk\hwmsdk-electron-70.16.5-src\hwmsdk-electron\node_modules\electron-rebuild\lib\src\rebuild.js:108:17)
    at async D:\jsuisdk\hwmsdk-electron-70.16.5-src\hwmsdk-electron\node_modules\electron-rebuild\lib\src\cli.js:154:9
```

原因分析:

Visual Studio没安装好或者环境变量GYP_MSVS_VERSION和安装的Visual Studio版本不配套

解决方案:

重新安装Visual Studio或者修改环境变量GYP_MSVS_VERSION为对应的Visual Studio的版本

12 专有名词解释

12.1 帐号相关名词

- **华为云帐号**
华为云上的租户身份标志，用于在华为云上购买华为云会议服务，或者使用华为云提供的工具（如API Explorer等）。帐号鉴权在华为云的IAM上完成。
- **华为云会议帐号**
华为云会议上的用户身份标志。帐号鉴权在华为云会议的USG上完成。华为云会议帐号分为3类：
 - **华为云会议SP帐号**
华为云会议SP帐号，用于多租户场景下SP管理员对多个企业的管理，包括企业发放、企业资源分配等。
 - **华为云会议企业管理员帐号**
华为云会议企业管理员帐号，用于企业管理员对企业的管理，包括企业部门管理、通讯录管理、云会议室管理等。
 - **华为云会议普通用户帐号**
华为云会议普通帐号，用于会议用户登录App（或者个人Portal）、创建会议、加入会议等。

12.2 会议 ID 相关名词

会议ID相关的名词包括：

- **会议ID (Conference ID)**：在创建并发资源会议时由系统生成的会议ID；或者在创建云会议室资源会议并且选择随机会议ID时由系统生成的会议ID。用户用这个ID加入会议。
- **云会议室会议ID (VMR Conference ID)**：属于某个云会议室或者某个帐号的个人固定会议ID。如果创建的会议ID是固定的，用户用这个ID加入会议。
- **会议UUID (Conference UUID)**：会议开始后，由系统生成的UUID。如果同一个会议ID在预约时间到期前（或者主持人结束会议前），召开多次，会生成多个会议UUID。这个ID用户不可见。

- **云会议室ID (VMR ID)**：标志某个云会议室或者个人固定会议ID的唯一ID。创建云会议室资源的会议或者并发资源的个人固定会议ID的会议时，需要在创会接口中填入这个ID。这个ID用户不可见。

这几个ID之间的关系如下图所示：

图 12-1 会议相关 ID 之间的关系

