

eSDK ICP V200R001C10 接口参考 01(ECC, Windows, CS)

# eSDK ICP V200R001C10 接口参考 01(ECC, Windows, CS)

文档版本 01

发布日期 2017-09-01



#### 版权所有 © 华为技术有限公司 2017。 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默示的声明或保证。

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。

## 华为技术有限公司

地址: 深圳市龙岗区坂田华为总部办公楼 邮编: 518129

网址: <a href="http://www.huawei.com">http://www.huawei.com</a>
客户服务邮箱: <a href="mailto:support@huawei.com">support@huawei.com</a>

客户服务电话: 4008302118

# 目录

1	1
2 最新修定记录	2
3 数据类型	7
3.1 类	
3.1.1 Agent(会议成员)	
3.1.2 AgentInfo(座席信息)	g
3.1.3 AgentStatusDetails(座席状态)	g
3.1.4 AgentStatusOverview(座席状态总览)	10
3.1.5 CallListDetails(排队电话信息)	11
3.1.6 CallListDetailsChanged(同组呼叫队列状态改变信息)	12
3.1.7 DialContent(呼叫内容)	12
3.1.8 MeetingMessage(会议操作消息)	12
3.1.9 PhoneEventOperation(电话事件处理)	13
3.1.10 PhoneStatusItem(电话状态项)	13
3.1.11 Room(会议室)	14
3.1.12 WirelessDeviceInfo(无线设备信息)	17
3.1.13 SkillGroup(技能队列)	17
3.1.14 AccessCode(系统接入码配置信息)	17
3.1.15 RestReason(休息原因)	18
3.1.16 AgentCallInfo(当前通话信息)	
3.1.17 AgentSkillGroup(座席配置技能信息)	19
3.1.18 SkillGroupStatus(呼叫队列状态)	19
3.1.19 LostCall(丢话记录)	20
3.1.20 MaliciousCall(骚扰记录)	20
3.1.21 RedList(优先级名单)	20
3.1.22 BlackList(黑名单)	21
3.1.23 IvrInfo(IVR 流程信息)	21
3.1.24 LocalInfo(本地信息)	
3.1.25 TalkingGroup(集群群组信息)	
3.1.26 RadioGroup(无线集群群组信息)	
3.1.27 PageInfo(分页信息)	23
3.1.28 GroupUserState(集群群组用户状态信息)	2.4

CS)

CS)	目 录
3.1.29 GroupUserDetail(集群群组用户所有状态信息)	25
3.1.30 TalkingGroupState(集群群组状态信息)	26
3.1.31 MixedGroupUserState(混编集群群组用户状态信息)	27
3.1.32 SoundDeviceInfo(音频设备信息)	27
3.2 枚举	28
3.2.1 CallingPhoneScope(呼入电话范围)	28
3.2.2 DeviceCallType(设备呼叫类型)	28
3.2.3 DeviceModule(硬件通讯设备)	28
3.2.4 DeviceType(设备类型)	29
3.2.5 DeviceWorkMode(设备工作模式)	29
3.2.6 HoldMode(保持模式)	30
3.2.7 MeetingOperation(会议操作)	30
3.2.8 MeetingStatus(会议状态)	31
3.2.9 MeetingType(会议类型)	31
3.2.10 MemberType(成员类型)	31
3.2.11 OpFlags(操作标识)	32
3.2.12 OpStatus(操作状态)	33
3.2.13 PTTMode(PTT 模式)	34
3.2.14 PTTOperate(PTT 操作类型)	34
3.2.15 EventTypeEnum(事件类型)	35
3.2.16 UserType(用户类型)	35
3.2.17 PhoneUsedMode(话机使用模式)	35
3.2.18 PhoneStatus(话机状态)	36
3.2.19 TransferType(转移类型)	
3.2.20 AudioDeviceType(声卡设备类型)	37
4 IDevice(设备服务基本接口)	38
4.1 IDevice Methods: 运行管理功能	39
4.1.1 Initialize(启动软电话)	39
4.1.2 UnInitialize(关闭软电话)	
4.1.3 GetVeision(版本号查询)	42
4.1.4 LoginToUPortal(登录 UPortal)	42
4.1.5 LogoutFromUPortal(登出 UPortal)	
4.1.6 UnInitializeAll(回收所有资源)	44
4.2 IDevice Methods: 语音接续功能	44
4.2.1 Pickup(摘机)	
4.2.2 Hangup(挂机)	45
4.2.3 Dial(单呼)	
4.2.4 Hold(保持)	
4.2.5 ConsultCall(协商呼叫)	
4.2.6 EndConsultCall(结束协商呼叫)	
4.2.7 TransferToAgent(协商转移)	
4.2.8 SingleStepTransfer(非监督转移)	51

4.2.9 SingleStepSuccTransfer(呼叫成功转移).       52         4.2.10 QueryCurrentCallingScope(責当当前呼叫范围).       53         4.2.11 PollingCall(栓造呼叫).       53         4.2.12 DisconnectCall(根据 CallId 释放呼叫).       54         4.2.13 TransferToskillGroup(通话转队列).       55         4.2.14 Dialts(单呼拓展).       56         4.2.15 PollingCallEx(枪御呼叫拓展).       57         4.2.16 Afture(静音).       59         4.2.17 AgentSendDTMF(二次故号).       59         4.2.18 TransferTolVR(转到 IVR).       60         4.2.19 IndidRyCallId(根据 CallId 保持通话).       61         4.3 I Serlus(實置).       62         4.3 I Serlus(實置).       63         4.3.2 SetIdle(置用).       63         4.3.3 QueryAgentStatus(查询原序状态).       63         4.3.4 QueryAllAgentStatus(查询所有施所状态).       66         4.3.5 Rest(体息).       66         4.3.7 SetAutoEnterIde(後間通话后是否进入开闲状态).       66         4.3.8 SetWorkStop(设置是否进入开关班).       67         4.3.9 ExitACW(选出格理态).       67         4.3.10 QueryRestReasons(查询 VDN 体息原例).       68         4.3.11 QueryStillGroupStatus(查询是有是是名单位的表现例如如如如如如如如如如如如如如如如如如如如如如如如如如如如如如如如如如如如	<u>CS)</u>	目录
4.2.10 QueryCurrentCallingScope(食商当前呼叫花園)	4.2.9 SingleStepSuccTransfer(呼叫成功转移)	52
4.2.11 PollingCall(轮询呼叫)		
4.2.12 DisconnectCall(根据 Callid 释放呼叫)		
4.2.13 TransferToSkillGroup(通话转队列)		
4.2.14 DailEx(牟呼拓展). 56 4.2.15 PollingCallEx(発询呼叫拓展). 57 4.2.16 Mus(帝育). 59 4.2.17 AgentSendDTMF(二次接号). 59 4.2.18 TransferTofVR(特別 IVR). 60 4.2.19 HoldByCallId(根据 CallId 保持通话). 61 4.3 IDevice Methods: 状态管理类功能. 62 4.3.1 SetBusy(置忙). 62 4.3.1 SetBusy(置忙). 62 4.3.2 SetIdle(置何). 62 4.3.3 QueryAgentStatus(查询座席状态). 63 4.3.3 QueryAgentStatus(查询座席状态). 63 4.3.4 QueryAllAgentStatus(查询座席状态). 64 4.3.5 Rest(休息). 65 4.3.6 CancelRest(取消休息). 66 4.3.6 CancelRest(取消休息). 66 4.3.7 SetAuroEnterIdle(设置通话后是否进入开关班). 67 4.3.9 ExitACW(退出整理态). 67 4.3.10 QueryRestReasons(查询 VDN 休息原因). 68 4.3.11 QuerySkillGroupStatus(查询呼叫队列状态). 69 4.3.13 GetConfig(查型配置项). 69 4.3.13 GetConfig(查面配置项). 70 4.4 IDevice Methods: 其他查询类功能 4.4 1 QueryCallListByGroupId(停止查询某组呼叫队列). 72 4.4.4 QueryCallListByGroupId(停止查询某组呼叫队列). 72 4.4.4 QueryBlackList(查询是否是黑名单扩展). 74 4.4.5 QueryBlackList(查询是否是黑名单扩展). 74 4.4.6 QueryAllBlackList(查询是否是黑名单扩展). 74 4.4.7 QueryAllBlackList(查询后否是黑名单扩展). 74 4.4.8 QueryRedList(查询后否是黑名单扩展). 75 4.4.9 QueryAllRedList(查询所有黑名单扩展). 76 4.4.10 QueryAllRedList(查询所有系统接入码配置信息). 77 4.4.11 QueryAccessCodeList(查询所有系统接入码配置信息). 77 4.4.11 QueryLostCalls(查询表简见列信息). 77 4.4.11 QueryLostCalls(查询表简见列信息). 77 4.4.11 QueryLostCalls(查询表简记表). 78 4.4.13 QueryLostCalls(查询表简记表). 78 4.4.14 QueryCallInfo(查询去前记录数量). 79 4.4.14 QueryCallInfo(查询去前记录数量). 79 4.4.14 QueryLostCalls(CallsCount(查询表话记录数量). 79 4.4.14 QueryCallInfo(查询去前记录数量). 79 4.4.14 QueryLostCalls(CallsCount(查询表话记录数量). 79 4.4.14 QueryLostCalls(CallsCount(查询表话记录数量). 79 4.4.14 QueryCallInfo(查询去前记录数量). 79 4.4.14 QueryLostCalls(CallsCount(查询表话记录数量). 79 4.4.14 QueryCallInfo(查询去前记录数量). 79 4.4.14 QueryCallInfo(查询去前记录数量). 79 4.4.14 QueryCallInfo(查询去前记录数量). 79 4.4.14 QueryCallInfo(查询去前记录数量). 79		
4.2.15 PollingCallEx(轮询呼叫拓展)		
4.2.16 Mute(静音)       59         4.2.17 AgentSendDTMF(二次接号)       59         4.2.18 TransferTolVR(转到 IVR)       60         4.2.19 HoldByCallId(根据 CallId (保持通话)       61         4.3 IDevice Methods: 状态管理类功能       62         4.3.1 SetBusy(置忙)       62         4.3.2 SetIdle(置阴)       63         4.3.3 QueryAgentStatus(查询座席状态)       63         4.3.4 QueryAllAgentStatus(查询所有座席状态)       64         4.3.5 Rest(休息)       65         4.3.6 CancelRest(取消体息)       66         4.3.7 SetAutoEntertdle(设置通话后是否进入开关班)       67         4.3.9 ExitACW(退出整理态)       67         4.3.10 QueryRestReasons(查询 VDN 休息原因)       68         4.3.11 QuerySkillGroupStatus(查询呼叫队列状态)       69         4.3.12 SetConfig(设置配置项)       69         4.3.13 GetConfig(查询配置项)       70         4.4 IDevice Methods: 其他直询类功能       71         4.4.2 QueryCallListByGroupId(查询来组呼叫队列)       72         4.4.3 QueryBlackList(查询是否是黑名单)       73         4.4.4 QueryAllBlackList(查询是否是需名单)       74         4.4.5 QueryBlackList(查询是否是保生级名单)       74         4.4.6 QueryAllBlackList(查询是否是优先级名单)       76         4.4.9 QueryAllRedList(查询是否是优先级名单)       76         4.4.10 QueryScalliforoups(查询支能影列信息)       77		
4.2.18 TransferToIVR(转到IVR)		
4.2.19 HoldByCallId(根据 CallId 保持通话)	4.2.17 AgentSendDTMF(二次拨号)	59
4.2.19 HoldByCallId(根据 CallId 保持通话)	4.2.18 TransferToIVR(转到 IVR)	60
4.3.1 SetBusy(置忙).       62         4.3.2 SetIdle(置闲).       63         4.3.3 QueryAgentStatus(查询座席状态).       63         4.3.4 QueryAllAgentStatus(查询所有座席状态).       64         4.3.5 Rest(休息).       65         4.3.6 CancelRest(取消休息).       66         4.3.7 SetAutoEnterIdle(设置通话后是否进入示闲状态).       66         4.3.8 SetWorkStop(设置是否进入示厌焚).       67         4.3.9 ExitACW(退出整理态).       67         4.3.10 QueryRestReasons(查询 VDN 休息原因).       68         4.3.11 QuerySkillGroupStatus(查询呼叫队列状态).       69         4.3.12 SetConfig(设置配置项).       69         4.3.13 GetConfig(查询配置项).       70         4.4 IDevice Methods: 其他查询类功能.       71         4.4.1 QueryIVRInfo(查询图表现所有见的。       71         4.4.2 QueryCallListByGroupId(查询某组呼叫队列).       72         4.4.3 StopQueryCallListByGroupId(查询是看是黑名单).       73         4.4.4 QueryBlackList(查询是否是黑名单扩展).       74         4.4.5 QueryAllBlackList(查询所有黑名单扩展).       74         4.4.6 QueryAllBlackList(查询所有黑名单扩展).       76         4.4.9 QueryAllBlackList(查询所有然经入码配置信息).       77         4.4.10 QuerySkillGroups(查询我能队列信息).       77         4.4.11 QueryCostCalls(查询表话记录).       78         4.4.12 QueryLostCalls(查询去话记录).       79         4.4.14 QueryC		
4.3.2 SetIdle(置闲)       63         4.3.3 QueryAgentStatus(查询座席状态)       63         4.3.4 QueryAllAgentStatus(查询所有座席状态)       64         4.3.5 Rest(休息)       65         4.3.6 CancelRest(取消休息)       66         4.3.7 SetAutoEnterfdle(设置通话后是否进入示闲状态)       66         4.3.8 SetWorkStop(设置通话后是否进入开关班)       67         4.3.9 ExitACW(退出整理态)       67         4.3.10 QueryRestReasons(查询VDN 休息原因)       68         4.3.11 QuerySkillGroupStatus(查询呼叫队列状态)       69         4.3.12 SetConfig(设置配置项)       70         4.4 IDevice Methods: 其他查询类功能       71         4.4.1 QueryIVRInfo(查询 IVR)       71         4.4.2 QueryCallListByGroupId(停止查询某组呼叫队列)       72         4.4.3 StopQueryCallListByGroupId(停止查询某组呼叫队列)       72         4.4.4 QueryBlackListEx(查询是否是黑名单)       73         4.4.5 QueryBlackListEx(查询是否是黑名单扩展)       74         4.4.6 QueryAllBlackListEx(查询是否是黑名单扩展)       74         4.4.7 QueryAllBlackListEx(查询所有既名单扩展)       75         4.4.8 QueryBedList(查询是否是先统先级名单)       76         4.4.9 QueryAllRedList(查询所有既名单扩展)       76         4.4.10 QuerySkillGroups(查询技能队列信息)       77         4.4.11 QueryCostCodel.ist(查询方有优先级名单)       76         4.4.12 QueryLostCalls(查询去话记录数量)       79 <td< td=""><td>4.3 IDevice Methods: 状态管理类功能</td><td>62</td></td<>	4.3 IDevice Methods: 状态管理类功能	62
4.3.3 QueryAgentStatus(音询座席状态)       63         4.3.4 QueryAllAgentStatus(音询所有座席状态)       64         4.3.5 Rest(休息)       65         4.3.6 CancelRest(取消休息)       66         4.3.7 SetAutoEnterIdle(设置通话后是否进入示闲状态)       66         4.3.8 SetWorkStop(设置是否进入开关班)       67         4.3.9 ExitACW(退出整理态)       67         4.3.10 QueryRestReasons(查询 VDN 休息原因)       68         4.3.11 QuerySkillGroupStatus(查询呼叫队列状态)       69         4.3.12 SetConfig(设置配置项)       69         4.3.13 GetConfig(查询配置项)       70         4.4 IDevice Methods: 其他查询类功能       71         4.4.1 QueryIRInfo(查询 IVR)       71         4.4.2 QueryCallListByGroupId(牵填某组呼叫队列)       72         4.4.3 StopQueryCallListByGroupId(停止查询某组呼叫队列)       72         4.4.4 QueryBlackList(查询是否是黑名单)       73         4.4.5 QueryBlackList(查询是否是黑名单)       74         4.4.6 QueryBlackList(查询是否是黑名单)       74         4.4.7 QueryAllBlackList(查询是否是黑名单)       75         4.4.8 QueryRedList(查询是否是优先级名单)       76         4.4.9 QuerySlilGroups(查询表话记录为单)       76         4.4.10 QuerySkillGroups(查询表话记录为量)       77         4.4.11 QueryAccessCodeList(查询所有系统接入码配置信息)       77         4.4.12 QueryLostCallsCount(查询去话记录数量)       79         4.4.14	4.3.1 SetBusy(置忙)	62
4.3.4 QueryAllAgentStatus(音询所有座席状态).       64         4.3.5 Rest(休息).       65         4.3.6 CancelRest(取消休息).       66         4.3.7 SetAutoEnterIdle(设置通话后是否进入示闲状态).       66         4.3.8 SetWorkStop(设置是否进入开关班).       67         4.3.9 ExitACW(退出整理态).       67         4.3.10 QueryRestReasons(查询 VDN 休息原因).       68         4.3.11 QuerySkillGroupStatus(查询呼叫队列状态).       69         4.3.12 SetConfig(设置配置项).       69         4.3.13 GetConfig(查询配置项).       70         4.4 IDevice Methods: 其他查询类功能       71         4.4.1 QueryVRInfo(查询 IVR).       71         4.4.2 QueryCallListByGroupId(停止等询某组呼叫队列).       72         4.4.3 StopQueryCallListByGroupId(停止查询某组呼叫队列).       72         4.4.4 QueryBlackList(查询是否是黑名单扩展).       74         4.4.5 QueryBlackListEx(查询是否是黑名单扩展).       74         4.4.6 QueryAllBlackListEx(查询所有黑名单扩展).       75         4.4.7 QueryAllRedList(查询是否是优先级名单).       76         4.4.9 QueryAllRedList(查询所有优先级名单).       76         4.4.10 QuerySkillGroups(查询去话记录数量).       77         4.4.12 QueryLostCalls(查询丢话记录数量).       78         4.4.13 QueryLostCallsCount(查询丢话记录数量).       79         4.4.14 QueryCallInfo(查询当前通话信息).       80	4.3.2 SetIdle(置闲)	63
4.3.5 Rest(休息)       65         4.3.6 CancelRest(取消休息)       66         4.3.7 SetAutoEnterIdle(设置通话后是否进入示闲状态)       66         4.3.8 SetWorkStop(设置是否进入开关班)       67         4.3.9 ExitACW(退出整理态)       67         4.3.10 QueryRestReasons(查询 VDN 休息原因)       68         4.3.11 QuerySkillGroupStatus(查询呼叫队列状态)       69         4.3.12 SetConfig(设置配置项)       69         4.3.13 GetConfig(查询配置项)       70         4.4 IDevice Methods: 其他查询类功能       71         4.4.1 QueryIVRInfo(查询 IVR)       71         4.4.2 QueryCallListByGroupId(停止查询某组呼叫队列)       72         4.4.3 StopQueryCallListByGroupId(停止查询某组呼叫队列)       72         4.4.4 QueryBlackList(查询是否是黑名单)       73         4.4.5 QueryBlackList(查询是否是黑名单)       74         4.4.6 QueryAllBlackList(查询是否是黑名单)       74         4.4.7 QueryAllBlackList(查询是否是优先级名单)       76         4.4.9 QueryRedList(查询是否是优先级名单)       76         4.4.10 QuerySkillGroups(查询技能队列信息)       77         4.4.11 QueryAccessCodeList(查询所有系统接入码配置信息)       77         4.4.12 QueryLostCalls(查询去话记录数量)       78         4.4.13 QueryLostCallsCount(查询去话记录数量)       79         4.4.14 QueryCallInfo(查询当前通话信息)       80	4.3.3 QueryAgentStatus(查询座席状态)	63
4.3.6 CancelRest(取消休息)	4.3.4 QueryAllAgentStatus(查询所有座席状态)	64
4.3.7 SetAutoEnterIdle(设置通话后是否进入示闲状态).       66         4.3.8 SetWorkStop(设置是否进入开关班).       67         4.3.9 ExitACW(退出整理态).       67         4.3.10 QueryRestReasons(查询 VDN 休息原因).       68         4.3.11 QuerySkillGroupStatus(查询呼叫队列状态).       69         4.3.12 SetConfig(设置配置项).       70         4.4 IDevice Methods: 其他查询类功能.       71         4.4.1 QueryIVRInfo(查询 IVR).       71         4.4.2 QueryCallListByGroupId(查询某组呼叫队列).       72         4.4.3 StopQueryCallListByGroupId(停止查询某组呼叫队列).       72         4.4.4 QueryBlackList(查询是否是黑名单).       73         4.4.5 QueryBlackList医(查询是否是黑名单扩展).       74         4.4.6 QueryAllBlackList(查询所有黑名单扩展).       74         4.4.7 QueryAllBlackList(查询所有黑名单扩展).       75         4.4.8 QueryRedList(查询是否是优先级名单).       76         4.4.9 QueryAllRedList(查询方在优先级名单).       76         4.4.10 QuerySkillGroups(查询技能队列信息).       77         4.4.11 QueryAccessCodeList(查询所有系统接入码配置信息).       77         4.4.12 QueryLostCalls(查询丢话记录数量).       78         4.4.13 QueryLostCalls(查询当前通话信息).       80	4.3.5 Rest(休息)	65
4.3.8 SetWorkStop(设置是否进入开关班)       67         4.3.9 ExitACW(退出整理态)       67         4.3.10 QueryRestReasons(查询 VDN 休息原因)       68         4.3.11 QuerySkillGroupStatus(查询呼叫队列状态)       69         4.3.12 SetConfig(设置配置项)       69         4.3.13 GetConfig(查询配置项)       70         4.4 IDevice Methods: 其他查询类功能       71         4.4.1 QueryIVRInfo(查询 IVR)       71         4.4.2 QueryCallListByGroupId(查询某组呼叫队列)       72         4.4.3 StopQueryCallListByGroupId(停止查询某组呼叫队列)       72         4.4.4 QueryBlackList(查询是否是黑名单)       73         4.4.5 QueryBlackList医(查询是否是黑名单)       74         4.4.6 QueryAllBlackList(查询所有黑名单)       74         4.4.7 QueryAllBlackListEx(查询所有黑名单)       75         4.4.8 QueryRedList(查询是否是优先级名单)       76         4.4.9 QueryAllRedList(查询方信先级名单)       76         4.4.10 QuerySkillGroups(查询技能队列信息)       77         4.4.11 QueryAccessCodeList(查询所有系统接入码配置信息)       77         4.4.12 QueryLostCalls(查询丢话记录数量)       78         4.4.13 QueryLostCallsCount(查询丢话记录数量)       79         4.4.14 QueryCallInfo(查询当前通话信息)       80	4.3.6 CancelRest(取消休息)	66
4.3.9 ExitACW(退出整理态)       67         4.3.10 QueryRestReasons(查询 VDN 休息原因)       68         4.3.11 QuerySkillGroupStatus(查询呼叫队列状态)       69         4.3.12 SetConfig(设置配置项)       69         4.3.13 GetConfig(查询配置项)       70         4.4 IDevice Methods: 其他查询类功能       71         4.4.1 QueryCallListByGroupId(查询某组呼叫队列)       72         4.4.3 StopQueryCallListByGroupId(停止查询某组呼叫队列)       72         4.4.4 QueryBlackList(查询是否是黑名单)       73         4.5 QueryBlackListEx(查询是否是黑名单)       74         4.6 QueryAllBlackListEx(查询是否是黑名单)       74         4.7 QueryAllBlackList(查询是否是优先级名单)       75         4.4.8 QueryRedList(查询是否是优先级名单)       76         4.4.9 QueryAllRedList(查询技能队列信息)       77         4.4.10 QuerySkillGroups(查询技能队列信息)       77         4.4.11 QueryCostCalls(查询丢话记录)       78         4.4.12 QueryLostCalls(查询丢话记录数量)       79         4.4.14 QueryCallInfo(查询当前通话信息)       80	4.3.7 SetAutoEnterIdle(设置通话后是否进入示闲状态)	66
4.3.10 QueryRestReasons(查询 VDN 休息原因)       68         4.3.11 QuerySkillGroupStatus(查询呼叫队列状态)       69         4.3.12 SetConfig(设置配置项)       70         4.4 IDevice Methods: 其他查询类功能       71         4.4.1 QueryIVRInfo(查询 IVR)       71         4.4.2 QueryCallListByGroupId(查询某组呼叫队列)       72         4.4.3 StopQueryCallListByGroupId(停止查询某组呼叫队列)       72         4.4.4 QueryBlackList(查询是否是黑名单)       73         4.4.5 QueryBlackList(查询是否是黑名单扩展)       74         4.4.6 QueryAllBlackList(查询所有黑名单)       74         4.4.7 QueryAllBlackList(查询所有黑名单扩展)       75         4.4.8 QueryRedList(查询是否是优先级名单)       76         4.4.9 QueryAllRedList(查询所有优先级名单)       76         4.4.10 QuerySkillGroups(查询技能队列信息)       77         4.4.11 QueryAccessCodeList(查询所有系统接入码配置信息)       77         4.4.12 QueryLostCalls(查询丢话记录数量)       78         4.4.13 QueryLostCalls(查询丢话记录数量)       79         4.4.14 QueryCallInfo(查询当前通话信息)       80	4.3.8 SetWorkStop(设置是否进入开关班)	67
4.3.11 QuerySkillGroupStatus(查询呼叫队列状态)694.3.12 SetConfig(设置配置项)704.4 IDevice Methods: 其他查询类功能714.4.1 QueryIVRInfo(查询 IVR)714.4.2 QueryCallListByGroupId(查询某组呼叫队列)724.4.3 StopQueryCallListByGroupId(停止查询某组呼叫队列)724.4.4 QueryBlackList(查询是否是黑名单)734.4.5 QueryBlackListEx(查询是否是黑名单扩展)744.4.6 QueryAllBlackList(查询所有黑名单)744.7 QueryAllBlackListEx(查询所有黑名单)754.8 QueryRedList(查询是否是优先级名单)764.9 QueryAllRedList(查询所有优先级名单)764.10 QuerySkillGroups(查询技能队列信息)774.11 QueryAccessCodeList(查询所有系统接入码配置信息)774.12 QueryLostCalls(查询丢话记录)784.13 QueryLostCalls(查询当前通话信息)794.4.14 QueryCallInfo(查询当前通话信息)80	4.3.9 ExitACW(退出整理态)	67
4.3.12 SetConfig(设置配置项)	4.3.10 QueryRestReasons(查询 VDN 休息原因)	68
4.3.13 GetConfig(查询配置項)       70         4.4 IDevice Methods: 其他查询类功能       71         4.4.1 QueryIVRInfo(查询 IVR)       71         4.4.2 QueryCallListByGroupId(查询某组呼叫队列)       72         4.4.3 StopQueryCallListByGroupId(停止查询某组呼叫队列)       72         4.4.4 QueryBlackList(查询是否是黑名单)       73         4.5 QueryBlackList(查询是否是黑名单扩展)       74         4.6 QueryAllBlackList(查询所有黑名单扩展)       74         4.7 QueryAllBlackListEx(查询所有黑名单扩展)       75         4.8 QueryRedList(查询是否是优先级名单)       76         4.4.9 QueryAllRedList(查询所有优先级名单)       76         4.4.10 QuerySkillGroups(查询技能队列信息)       77         4.1.11 QueryAccessCodeList(查询所有系统接入码配置信息)       77         4.4.12 QueryLostCalls(查询丢话记录数量)       78         4.4.13 QueryLostCallsCount(查询丢话记录数量)       79         4.4.14 QueryCallInfo(查询当前通话信息)       80	4.3.11 QuerySkillGroupStatus(查询呼叫队列状态)	69
4.4 IDevice Methods: 其他查询类功能714.4.1 QueryIVRInfo(查询 IVR)714.4.2 QueryCallListByGroupId(查询某组呼叫队列)724.4.3 StopQueryCallListByGroupId(停止查询某组呼叫队列)724.4.4 QueryBlackList(查询是否是黑名单)734.4.5 QueryBlackListEx(查询是否是黑名单扩展)744.4.6 QueryAllBlackList(查询所有黑名单)744.4.7 QueryAllBlackListEx(查询所有黑名单扩展)754.4.8 QueryRedList(查询是否是优先级名单)764.4.9 QueryAllRedList(查询所有优先级名单)764.4.10 QuerySkillGroups(查询技能队列信息)774.4.11 QueryAccessCodeList(查询所有系统接入码配置信息)774.4.12 QueryLostCalls(查询丢话记录)784.4.13 QueryLostCalls(查询丢话记录数量)794.4.14 QueryCallInfo(查询当前通话信息)80	4.3.12 SetConfig(设置配置项)	69
4.4.1 QueryIVRInfo(查询 IVR)714.4.2 QueryCallListByGroupId(查询某组呼叫队列)724.4.3 StopQueryCallListByGroupId(停止查询某组呼叫队列)724.4.4 QueryBlackList(查询是否是黑名单)734.4.5 QueryBlackListEx(查询是否是黑名单扩展)744.4.6 QueryAllBlackList(查询所有黑名单)744.4.7 QueryAllBlackListEx(查询所有黑名单扩展)754.4.8 QueryRedList(查询是否是优先级名单)764.4.9 QueryAllRedList(查询所有优先级名单)764.4.10 QuerySkillGroups(查询技能队列信息)774.4.11 QueryAccessCodeList(查询所有系统接入码配置信息)774.4.12 QueryLostCalls(查询丢话记录)784.4.13 QueryLostCalls(查询丢话记录数量)794.4.14 QueryCallInfo(查询当前通话信息)80	4.3.13 GetConfig(查询配置项)	70
4.4.2 QueryCallListByGroupId(查询某组呼叫队列)724.4.3 StopQueryCallListByGroupId(停止查询某组呼叫队列)724.4.4 QueryBlackList(查询是否是黑名单)734.4.5 QueryBlackListEx(查询是否是黑名单扩展)744.4.6 QueryAllBlackList(查询所有黑名单)744.4.7 QueryAllBlackListEx(查询所有黑名单扩展)754.4.8 QueryRedList(查询是否是优先级名单)764.4.9 QueryAllRedList(查询所有优先级名单)764.4.10 QuerySkillGroups(查询技能队列信息)774.4.11 QueryAccessCodeList(查询所有系统接入码配置信息)774.4.12 QueryLostCalls(查询丢话记录)784.4.13 QueryLostCallsCount(查询丢话记录数量)794.4.14 QueryCallInfo(查询当前通话信息)80	4.4 IDevice Methods: 其他查询类功能	71
4.4.3 StopQueryCallListByGroupId(停止查询某组呼叫队列).724.4.4 QueryBlackList(查询是否是黑名单).734.4.5 QueryBlackListEx(查询是否是黑名单扩展).744.4.6 QueryAllBlackList(查询所有黑名单).744.4.7 QueryAllBlackListEx(查询所有黑名单扩展).754.4.8 QueryRedList(查询是否是优先级名单).764.4.9 QueryAllRedList(查询所有优先级名单).764.4.10 QuerySkillGroups(查询技能队列信息).774.4.11 QueryAccessCodeList(查询所有系统接入码配置信息).774.4.12 QueryLostCalls(查询丢话记录).784.4.13 QueryLostCallsCount(查询丢话记录数量).794.4.14 QueryCallInfo(查询当前通话信息).80	4.4.1 QueryIVRInfo(查询 IVR)	71
4.4.4 QueryBlackList(查询是否是黑名单)734.4.5 QueryBlackListEx(查询是否是黑名单扩展)744.4.6 QueryAllBlackList(查询所有黑名单)744.4.7 QueryAllBlackListEx(查询所有黑名单扩展)754.4.8 QueryRedList(查询是否是优先级名单)764.4.9 QueryAllRedList(查询所有优先级名单)764.4.10 QuerySkillGroups(查询技能队列信息)774.4.11 QueryAccessCodeList(查询所有系统接入码配置信息)774.4.12 QueryLostCalls(查询丢话记录)784.4.13 QueryLostCallsCount(查询丢话记录数量)794.4.14 QueryCallInfo(查询当前通话信息)80	4.4.2 QueryCallListByGroupId(查询某组呼叫队列)	72
4.4.5 QueryBlackListEx(查询是否是黑名单扩展)744.4.6 QueryAllBlackList(查询所有黑名单)744.4.7 QueryAllBlackListEx(查询所有黑名单扩展)754.4.8 QueryRedList(查询是否是优先级名单)764.4.9 QueryAllRedList(查询所有优先级名单)764.4.10 QuerySkillGroups(查询技能队列信息)774.4.11 QueryAccessCodeList(查询所有系统接入码配置信息)774.4.12 QueryLostCalls(查询丢话记录)784.4.13 QueryLostCallsCount(查询丢话记录数量)794.4.14 QueryCallInfo(查询当前通话信息)80	4.4.3 StopQueryCallListByGroupId(停止查询某组呼叫队列)	72
4.4.6 QueryAllBlackList(查询所有黑名单)	4.4.4 QueryBlackList(查询是否是黑名单)	73
4.4.7 QueryAllBlackListEx(查询所有黑名单扩展)	4.4.5 QueryBlackListEx(查询是否是黑名单扩展)	74
4.4.8 QueryRedList(查询是否是优先级名单)	4.4.6 QueryAllBlackList(查询所有黑名单)	74
4.4.9 QueryAllRedList(查询所有优先级名单)	4.4.7 QueryAllBlackListEx(查询所有黑名单扩展)	
4.4.10 QuerySkillGroups(查询技能队列信息)	4.4.8 QueryRedList(查询是否是优先级名单)	76
4.4.11 QueryAccessCodeList(查询所有系统接入码配置信息)774.4.12 QueryLostCalls(查询丢话记录)784.4.13 QueryLostCallsCount(查询丢话记录数量)794.4.14 QueryCallInfo(查询当前通话信息)80	4.4.9 QueryAllRedList(查询所有优先级名单)	76
4.4.12 QueryLostCalls(查询丢话记录)	4.4.10 QuerySkillGroups(查询技能队列信息)	77
4.4.13 QueryLostCallsCount(查询丢话记录数量)794.4.14 QueryCallInfo(查询当前通话信息)80	4.4.11 QueryAccessCodeList(查询所有系统接入码配置信息)	77
4.4.14 QueryCallInfo(查询当前通话信息)		
	4.4.13 QueryLostCallsCount(查询丢话记录数量)	
4.4.15 QueryAgentSkillGroups(查询座席技能信息)	4.4.14 QueryCallInfo(查询当前通话信息)	80
	4.4.15 QueryAgentSkillGroups(查询座席技能信息)	80
4.4.16 QueryMaliciousCalls(查询骚扰记录)	4.4.16 QueryMaliciousCalls(查询骚扰记录)	

CS)	目录
4.5 IDevice Methods: 电话会议类功能	82
4.5.1 CreateMeeting(创建会议)	
4.5.2 MakeMeeting(形成会议)	
4.5.3 SingleStepMeeting(单步会议)	
4.5.4 InviteMeetingMember(邀请会议成员)	
4.5.5 RemoveMeetingMember(移除会议成员)	
4.5.6 ExitMeeting(退出会议)	
4.5.7 TransferMeetingMaster(转移会议主席)	
4.5.8 EndMeeting(结束会议)	
4.5.9 QueryMeetingMembers(查询会议成员)	88
4.5.10 CreateMeetingEx(创建会议拓展)	89
4.5.11 ChangeMeetingMemberMode(修改与会者属性)	90
4.6 IDevice Methods: 座席管理类功能	91
4.6.1 Barge(强插)	91
4.6.2 ForceRelease(拦截)	92
4.6.3 Monitor(监听)	93
4.6.4 MonitorToBarge(监听转强插)	94
4.6.5 MonitorToForceRelease(监听转拦截)	95
4.6.6 Whisper(耳语)	96
4.6.7 SuperviseStop(停止质检)	97
4.6.8 SuperviseSwitch(质检切换)	97
4.6.9 AddToBlackList(设置黑名单)	98
4.6.10 AddToBlackListEx(设置黑名单扩展)	99
4.6.11 RemoveFromBlackList(取消黑名单)	100
4.6.12 RemoveAllBlackList(移除全部黑名单)	101
4.6.13 SelectPickup(选择应答)	101
4.6.14 SnatchPickup(代接)	102
4.6.15 ForceLogout(强制座席签出)	103
4.6.16 ChangeAgentPwd(修改座席密码)	
4.6.17 AddToRedList(设置优先级名单)	105
4.6.18 RemoveFromRedList(取消优先级名单)	105
4.6.19 RemoveAllRedList(移除全部优先级名单)	
4.6.20 SendMessage(座席发送便签)	106
4.6.21 SetAutoAnswer(设置是否自动应答)	
4.6.22 SetCallData(设置呼叫数据)	
4.6.23 QueryCallData(查询呼叫数据)	
4.6.24 SetCustomerLevel(设置呼叫的客户级别)	109
4.6.25 QueryCallIDOnAgent(查询座席的 CallId)	110
4.6.26 QueryAgentIP(查询座席 IP)	
4.6.27 ResetAgentSkillGroups(重置座席技能)	
4.6.28 SubscribeAgentEvent(订阅座席事件)	
4.6.29 StopSubscribeAgentEvent(停止订阅座席事件)	112

CS)	目录
4.7 IDevice Methods: 录音管理功能	113
4.7.1 PlayRecord(播放录音)	
4.7.2 PlayRecordPause(暂停播放)	
4.7.3 PlayRecordResume(继续播放)	
4.7.4 PlayRecordSeekTo(播放进度跳转)	
4.7.5 PlayRecordStop(停止播放)	
4.7.6 GetRecordLength(获取录音时长)	
4.7.7 DownloadRecordFile(下载录音文件)	
4.7.8 SetCustomizedRecordInfo(设置录音关联业务信息)	
4.8 IDevice Methods: 无线资源类功能	118
4.8.1 PTT(抢权)	118
4.8.2 QueryWirelessInfo(查询无线设备信息)	119
4.8.3 QueryAllWirelessDevice(查询所有无线设备信息)	120
4.8.4 QueryUsingWirelessDevice(查询当前使用的无线设备信息)	120
4.9 IDevice Methods: 短信服务类功能	121
4.9.1 SendSMS(发送短信)	121
4.9.2 ReplySMS(回复短信)	122
4.9.3 ReleaseCallExWithCCBIdx(座席释放指定短信用户呼叫)	123
4.9.4 TransferSMSToAgent(短信转移)	124
4.10 IDevice Methods: 邮件服务类功能	124
4.10.1 SendEmail(发送邮件)	125
4.10.2 ReplyEmail(回复邮件)	
4.10.3 QueryAgentEmailCfgInfo(查询座席邮件配置信息)	126
4.10.4 GetSendEmailIndex(获取发送邮件索引号)	127
4.10.5 ReleaseMailCall(停止邮件呼叫)	127
4.11 IDevice Methods: 视频通话类功能	128
4.11.1 PickupVCCall(接听视频电话)	
4.11.2 HangupVCCall(挂断视频电话)	128
4.12 IDevice Methods: 话机管理功能	
4.12.1 PhoneInitialize(初始化话机)	
4.12.2 PhoneUnInitialize(去初始化话机)	
4.12.3 PhoneAnswer(话机应答)	
4.12.4 PhoneRelease(话机释放)	
4.12.5 PhoneMute(话机静音)	
4.12.6 PhoneSwitchSoundDevice(话机呼叫切换声卡设备)	
4.12.7 PhoneGetSoundDevice(话机查询声卡设备信息)	
4.12.8 PhoneSetSoundDevice(话机设置声卡设备信息)	
4.12.9 PhoneGetVolume(话机查询设备音量)	
4.12.10 PhoneGetDeviceVolume(话机查询主、辅助设备音量)	
4.12.11 PhoneSetVolume(话机调整设备音量)	
4.12.12 PhoneSetDeviceVolume(话机调整辅助设备音量)	
4.12.13 PhoneGetVolumeGainByCall(话机查询指定呼叫音量增益)	141

CS)	目录
4.12.14 Db	142
4.12.14 PhoneSetVolumeGainByCall(话机设置指定呼叫音量)	
4.13 IDevice Methods:集群调度类功能	
4.13.1 TalkingGroupQuery(查询集群群组)	
4.13.2 TalkingGroupQueryState(查询集群群组状态)	
4.13.3 TalkingGroupAdd(新增集群群组)	
4.13.4 TalkingGroupModify(修改集群群组)	
4.13.5 TalkingGroupDelete(删除集群群组)	
4.13.6 RadioGroupQuery(查询无线集群群组)	
4.13.7 TalkingGroupSubscribe(订阅集群群组)	
4.13.8 TalkingGroupMonitor(Monitor)	
4.13.9 TalkingGroupSelect(Monitor/Select 切换)	
4.13.10 TalkingGroupPTT(抢权或放权)	
4.13.11 TalkingGroupGetCallInfo(查询集群群组呼叫信息)	
4.13.12 MixedGroupCallStart(发起混编集群群组呼叫)	
4.13.13 MixedGroupCallEnd(结束混编集群群组呼叫)	
4.13.14 MixedGroupChangeUserState(修改混编集群群组用户状态)	154
5 IDeviceEvents(设备服务回调事件)	156
5.1 IDeviceEvents Methods: 运行管理类事件	157
5.1.1 OnInitializeResult(启动软电话结果)	
5.1.2 OnUnInitializeResult(关闭软电话结果)	
5.1.3 OnLoginToUPortalResult(登录 UPortal 结果事件)	158
5.1.4 OnLogoutFromUPortal(登出 UPortal 结果事件)	158
5.1.5 OnDisConnection(设备掉线)	159
5.1.6 OnConnection(设备重连成功)	
5.1.7 OnError(系统异常)	160
5.1.8 OnUnInitializeAllResult(清空所有资源结果事件)	160
5.2 IDeviceEvents Methods: 语音接续类事件	161
5.2.1 OnNewCall(新来电信息)	161
5.2.2 OnAnswerCallResult(应答结果信息)	
5.2.3 OnHangUpResult(挂机返回)	162
5.2.4 OnOtherHangup(对方挂机事件)	163
5.2.5 OnAdvancedOtherHangUp(对方挂机事件)	163
5.2.6 OnDialResult(单呼返回)	164
5.2.7 OnHoldResult(保持结果)	
5.2.8 OnConsultCallResult(协商呼叫结果)	
5.2.9 OnEndConsultCallResult(结束协商呼叫结果)	165
5.2.10 OnTransferToAgentResult(协商转移结果)	166
5.2.11 OnSingleStepTransferResult(非监督转移结果)	
5.2.12 OnSingleStepSuccTransferResult(监督转移结果)	
5.2.13 OnTransferInform(呼叫转移信息)	
5.2.14 OnQueryCurrentCallingScopeResult(查询当前呼叫范围结果)	
5.2.15 OnPollingCallResult(轮询呼叫结果)	169

CS)	目录
5.2.16 OnDo Transfor(所則) 並結移車併)	170
5.2.16 OnBeTransfer(呼叫被转移事件)5.2.17 OnPhoneAlerting(座席电话振铃通知事件)	
5.2.18 OnPhoneAlertOtherHungup(座席电话振铃时对方挂机事件)	
5.2.19 OnPhoneHungup(座席电话挂机事件)	
5.2.21 OnTransferToSkillGroupResult(通话转移队列结果)	
5.2.23 OnPollingCallExResult(轮询呼叫拓展结果)	
5.2.24 OnMuteResult(静音结果)	
5.2.25 OnAgentSendDTMFResult(二次拨号结果)	
5.2.26 OnLongNoAnswer(久不应答通知事件)	
5.2.27 OnPhoneBusy(座席话机无法应答通知)	
5.2.28 OnPhoneOffhook(座席话机摘机事件)	
5.2.29 OnTransferToIVRResult(转到 IVR 结果)	
5.2.30 OnReturnFromIvr(呼叫从 IVR 返回)	
5.2.31 OnDummyCallArrive(早释电话通知)	
5.2.31 OnLinkageResultNotify(启动联动/停止联动结果)	
5.2.33 OnLinkagePhoneStatus(联动话机状态改变通知)	
5.2.34 OnLinkageCalloutNotify(联动话机呼出通知)	
5.2.35 OnHoldByCallIDResult(保持指定通话结果)	
5.3.1 OnSetBusyResult(置忙返回)	
5.3.2 OnSetIdleResult(置闲返回)	
5.3.3 OnAgentStatusChange(座席状态改变信息)	
5.3.4 OnQueryAgentStatusResult(查询座席状态结果)	
5.3.5 OnQueryAllAgentStatusResult(查询组内所有座席状态结果)	
5.3.6 OnCallListChange(组内呼叫队列状态改变信息)	
5.3.8 OnCancelRestResult(取消休息返回)     5.3.9 OnRestTimeOut(休息超时提醒)	
5.3.10 OnExitACWResult(退出整理态结果)	
5.3.11 OnMeetingPTTStatusChange(座席会议 PTT 状态改变信息)	
5.3.13 OnSetAutoEnterIdleResult(设置呼叫后是否进入示闲状态结果返回)	
5.3.14 OnSetWorkStopResult(设置是否进入关班状态结果返回)	
5.3.15 OnQueryRestReasonsResult(查询 VDN 休息原因返回结果)	
5.3.16 OnQuerySkillGroupStatusResult(查询呼叫队列状态返回)	
5.3.17 OnSetConfigResult(设置配置项结果)	
5.3.18 OnGetConfigResult(查询配置项结果).         5.4 IDeviceEvents Methods: 其他查询类事件.	
5.4.1 OnQueryIVRInfoResult(查询 IVR 结果)	
5.4.2 OnQueryCallListByGroupIdResult(查询组内呼叫队列结果)	191

CS)

CS)	目录
5.4.3 OnQueryBlackListResult(查询是否是黑名单结果)	191
5.4.4 OnQueryBlackListExResult(查询是否是黑名单结果扩展)	
5.4.5 OnQueryAllBlackListResult(查询所有黑名单结果)	
5.4.6 OnQueryAllBlackListExResult(查询所有黑名单结果扩展)	
5.4.7 OnQueryRedListResult(查询是否是优先级名单结果)	
5.4.8 OnQueryAllRedListResult(查询所有优先级名单结果)	
5.4.9 OnQuerySkillGroupsResult(查询系统的技能队列结果)	
5.4.10 OnQueryAccessCodeListResult(查询所有系统接入码配置信息返回)	
5.4.11 OnQueryLostCallsResult(查询丢话记录结果)	
5.4.12 OnQueryLostCallsCountResult(查询丢话记录数量结果)	
5.4.13 OnStopQueryCallListByGroupIdResult(停止查询呼叫队列结果)	
5.4.14 OnQueryCallInfoResult(查询当前通话信息结果)	198
5.4.15 OnQueryAgentSkillGroupsResult(查询座席技能信息结果返回)	199
5.4.16 OnQueryMaliciousCallsResult(查询骚扰记录结果返回)	
5.5 IDeviceEvents Methods: 电话会议类事件	
5.5.1 OnCreateMeetingResult(创建会议返回结果事件)	200
5.5.2 OnYieldAnswerCall(创建会议呼叫应答返回事件)	201
5.5.3 OnMakeMeetingResult(形成会议结果)	201
5.5.4 OnSingleStepMeetingResult(单步会议结果)	202
5.5.5 OnInviteMeetingMemberResult(邀请会议成员返回)	203
5.5.6 OnInvitedMeeting(被邀请加入会议返回)	203
5.5.7 OnRemoveMeetingMemberResult(移除会议成员结果)	204
5.5.8 OnLeaveMeeting(座席离开会议时返回)	204
5.5.9 OnExitMeetingResult(退出会议结果)	205
5.5.10 OnMeetingMemberExit(成员退出会议事件)	205
5.5.11 OnTransferMeetingMasterResult(转移会议主席结果)	206
5.5.12 OnBeTransferMaster(被任命为会议主席)	207
5.5.13 OnEndMeetingResult(结束会议结果)	207
5.5.14 OnQueryMeetingMembersResult(查询会议成员结果)	208
5.5.15 OnMeetingMemberAlerting(会议中邀请用户状态反馈)	209
5.5.16 OnMeetingMemberJoin(新会议成员加入事件)	209
5.5.17 OnCreateMeetingExResult(创建会议拓展)	210
5.5.18 OnChangeMeetingMemberModeResult(修改与会者属性结果)	210
5.5.19 OnMeetingMemberModeChange(与会者属性变更通知事件)	211
5.5.20 OnMeetingMasterChange(会议主席变更通知事件)	212
5.6 IDeviceEvents Methods: 座席管理类事件	212
5.6.1 OnBargeResult(强插结果)	212
5.6.2 OnBeBarged(被强插事件)	213
5.6.3 OnForceReleaseResult(拦截结果)	213
5.6.4 OnBeForceReleased(被拦截事件)	214
5.6.5 OnMonitorResult(监听结果)	214
5.6.6 OnMonitorToBargeResult(监听转强插结果)	215

CS)

CS)	目 录
5.6.7 OnMonitorToForceReleaseResult(监听转拦截结果)	215
5.6.8 OnAddToBlackListResult(设置黑名单结果)	216
5.6.9 OnAddToBlackListExResult(设置黑名单结果扩展)	217
5.6.10 OnRemoveFromBlackListResult(取消黑名单结果)	
5.6.11 OnRemoveAllBlackListResult(移除全部黑名单结果)	
5.6.12 OnSelectPickupResult(选择应答结果)	218
5.6.13 OnSnatchPickupResult(代接结果)	
5.6.14 OnSnatchPickUpAnswer(代接应答事件)	220
5.6.15 OnBeSnatchedPickup(被代接事件)	220
5.6.16 OnForceLogoutResult(强制座席签出结果)	220
5.6.17 OnBeForceLogout(座席被强制签出事件)	221
5.6.18 OnOtherBeForceRelease(对方被拦截事件)	221
5.6.19 OnChangeAgentPwdResult(修改座席密码结果)	
5.6.20 OnAddToRedListResult(设置优先级名单结果)	222
5.6.21 OnRemoveFromRedListResult(取消优先级名单结果)	223
5.6.22 OnRemoveAllRedListResult(移除全部优先级名单结果)	224
5.6.23 OnWhisperResult(耳语结果)	224
5.6.24 OnSuperviseStopResult(停止质检结果)	
5.6.25 OnSuperviseSwitchResult(质检切换结果)	225
5.6.26 OnBeWhispered(被耳语事件)	226
5.6.27 OnBeMonitored(被监听事件)	227
5.6.28 OnBeStopSupervise(停止质检事件)	227
5.6.29 OnSendMessageResult(发送便签结果返回)	
5.6.30 OnReceiveMessage(接收便签事件)	228
5.6.31 OnSetAutoAnswerResult(设置是否自动应答结果返回)	229
5.6.32 OnSetCallDataResult(设置呼叫随路数据结果返回)	229
5.6.33 OnQueryCallDataResult(获取呼叫随路数据结果返回)	230
5.6.34 OnSetCustomerLevelResult(设置呼叫的客户级别结果返回)	230
5.6.35 OnQueryCallIDOnAgentResult(查询座席的 CallId 结果返回)	231
5.6.36 OnQueryAgentIPResult(查询座席 IP 结果)	232
5.6.37 OnResetAgentSkillGroupsResult(重置座席技能队列结果)	
5.6.38 OnSubscribeAgentEventResult(订阅座席事件返回)	233
5.6.39 OnStopSubscribeAgentEventResult(取消订阅座席事件返回)	233
5.6.40 OnAgentEventNotify(座席事件通知)	234
5.7 IDeviceEvents Methods: 录音管理类事件	235
5.7.1 OnStartRecord(开始录音信息)	235
5.7.2 OnRecordFailure(录音失败)	235
5.7.3 OnPlayRecordResult(播放录音结果)	236
5.7.4 OnPlayStatus(录音播放状态)	236
5.7.5 OnPlayRecordPauseResult(暂停播放结果)	
5.7.6 OnPlayRecordResumeResult(继续播放结果)	237
5.7.7 OnPlayRecordSeekToResult(播放跳转结果)	238

CS)	目 录
5.7.8 OnPlayRecordStopResult(停止播放结果)	238
5.7.9 OnGetRecordLengthResult(获取录音时长结果)	
5.7.10 OnEndPlayRecord(放音结束)	
5.7.11 OnEndRecord(停止录音)	
5.7.12 OnDownloadRecordFileResult(下载录音文件结果)	
5.7.13 OnSetCustomizedRecordInfoResult(设置录音关联信息结果)	
5.8 IDeviceEvents Methods: 无线资源类事件	
5.8.1 OnPTTResult(PTT 操作结果)	
5.8.2 OnReportPTTOff(PTT 话权被释放通知)	
5.8.3 OnQueryWirelessInfoResult(查询无线设备信息结果)	
5.8.4 OnQueryAllWirelessDeviceResult(查询所有无线设备信息结果)	
5.8.5 OnQueryUsingWirelessDeviceResult(查询当前使用的无线设备信息结果)	
5.9 IDeviceEvents Methods: 短信服务类事件	
5.9.1 OnReceiveSMS(新短信接收事件)	
5.9.2 OnReceiveP2PSMS(短信接收事件)	
5.9.3 OnSendSMSResult(发送短信结果)	
5.9.4 OnReplySMSResult(回复短信结果)	
5.9.5 OnClientDataWithoutCallConfirm(座席主动发送短信后,返回用户短信接收确定)	
5.9.6 OnClientDataWithoutCallFail(座席发送短信,用户确认失败)	
5.9.7 OnReleaseCallExWithCCBIdxResult(座席释放指定短信用户呼叫结果)	
5.9.8 OnReceiveReleaseSMS(短信呼叫释放通知)	
5.9.9 OnTransferSMSToAgentResult(短信转移结果)	
5.10 IDeviceEvents Methods: 邮件服务类事件	
5.10.1 OnReceiveEmail(新邮件接收消息)	
5.10.2 OnSendEmailResult(发送邮件结果)	
5.10.3 OnReplyEmailResult(回复邮件结果)	
5.10.4 OnQueryAgentEmailCfgInfoResult(查询座席邮件配置信息结果)	
5.10.5 OnGetSendEmailIndexResult(获取发送邮件索引号结果)	
5.10.6 OnReleaseMailCall(停止邮件呼叫结果)	
5.10.7 OnAgentReceiveEmailRemind(催促座席尽快处理 Email 呼叫)	
5.11 IDeviceEvents Methods: 视频通话类事件	254
5.11.1 OnPickupVCCallResult(接听视频电话结果)	254
5.11.2 OnHangupVCCallResult(挂断视频电话结果)	255
5.11.3 OnVCPhoneStatusNotify(视频电话状态通知)	256
5.11.4 OnNewVCCall(新视频电话来电)	256
5.11.5 OnOtherVCCallHangup(视频通话对方挂断)	257
5.11.6 OnStartRecordVCCall(开始录制视频电话通知)	257
5.11.7 OnEndRecordVCCall(结束录制视频电话通知)	258
5.11.8 OnVCCallLongNoAnswer(视频电话久不应答通知)	258
5.11.9 OnVCCallRecordFail(录播失败通知)	258
5.12 IDeviceEvents Methods: 话机管理类事件	259
5.12.1 OnPhoneInitializeResult(初始化话机结果事件)	259

<u>CS</u> )	目录
5.12.2 OnPhoneUnInitializeResult(去初始化话机结果事件)	259
5.12.3 OnPhoneLinkageStatusNotify(话机联动状态上报事件)	
5.12.4 OnPhoneStatusChangeEvent(话机状态变更上报事件)	
5.12.5 OnPhoneAnswerRequestEx(话机请求应答事件)	
5.12.6 OnPhoneAnswerRequest(话机请求应答事件)	
5.12.7 OnPhoneAutoAnswerEvent(话机自动应答事件)	
5.12.8 OnPhoneAnswerResult(话机应答结果事件)	
5.12.9 OnPhoneReleaseResult(话机释放结果事件)	
5.12.10 OnPhoneReleaseEvent(话机呼叫释放事件)	
5.12.11 OnPhoneMuteResult(话机静音结果事件)	
5.12.12 OnPhoneSwitchSoundDeviceResult(话机呼叫切换声卡设备结果事件)	
5.12.13 OnPhoneGetSoundDeviceResult(话机查询声卡设备信息结果事件)	
5.12.14 OnPhoneSetSoundDeviceResult(话机设置声卡设备信息结果事件)	267
5.12.15 OnPhoneGetVolumeResult(话机查询设备音量结果事件)	268
5.12.16 OnPhoneGetDeviceVolumeResult(话机查询辅助设备音量结果事件)	269
5.12.17 OnPhoneSetVolumeResult(话机设置设备音量结果事件)	269
5.12.18 OnPhoneSetDeviceVolumeResult(话机设置设备音量结果事件)	270
5.12.19 OnPhoneGetVolumeGainByCallResult(话机查询指定呼叫音量结果事件)	271
5.12.20 OnPhoneSetVolumeGainByCallResult(话机设置指定呼叫音量结果事件)	272
5.13 IDeviceEvents Methods: 集群调类事件	272
5.13.1 OnTalkingGroupQueryResult(查询集群群组结果事件)	272
5.13.2 OnTalkingGroupQueryStateResult(查询集群群组状态结果事件)	273
5.13.3 OnTalkingGroupAddResult(新增集群群组结果事件)	274
5.13.4 OnTalkingGroupModifyResult(修改集群群组结果事件)	274
5.13.5 OnTalkingGroupDeleteResult(删除集群群组结果事件)	275
5.13.6 OnRadioGroupQueryResult(查询无线集群群组结果事件)	276
5.13.7 OnTalkingGroupSubscribeResult(订阅集群群组结果事件)	276
5.13.8 OnTalkingGroupUserStateChange(集群群组用户状态变更通知事件)	277
5.13.9 OnTalkingGroupMonitorResult(监听集群群组结果事件)	277
5.13.10 OnTalkingGroupSelectResult(Monitor/Select 切换结果事件)	278
5.13.11 OnTalkingGroupPTTResult(抢权或放权结果事件)	279
5.13.12 OnTalkingGroupGetCallInfoResult(查询集群群组呼叫信息结果事件)	280
5.13.13 OnMixedGroupCallStartResult(发起混编集群群组呼叫结果事件)	280
5.13.14 OnMixedGroupCallEndResult(结束混编集群群组呼叫结果事件)	281
5.13.15 OnMixedGroupChangeUserStateResult(修改混编集群群组用户状态结果事件)	282
6 错误码	283
6.1 ECC SDK 错误码	284
6.2 服务端错误码	288
6.3 Voice.ocx 接口错误码	292
6.4 TUP 错误码	294
7 术语	296

CS) 1 概述

**】** 概述

#### 目的

本文主要介绍eSDK ICP ECC SDK接口参考。介绍内容包括相关数据类型、接口功能、方法定义、参数描述和使用示例等。

### 版本配套

与本文档相配套产品版本如下所示。

产品名称	版本
eSDK ECC	V200R001C10
IPCC	V200R001C80SPC200

## 技术支持

如果您在开发过程中碰到任何问题,开发者社区提供了以下技术支持渠道:

- 可以在**华为开发者论坛**发帖求助。
- 可以在**DevCenter**系统中提单跟踪。
- 华为技术支持热线电话: 400-8828-000。
- 华为技术支持邮箱: esdk@huawei.com。

2 最新修定记录

# 2 最新修定记录

日期	修订版本	描述
2017-3-20	V200R001C10	V200R001C10版本文档发布。
		新增类:
		LocalInfo
		TalkingGroup
		RadioGroup
		PageInfo
		GroupUserState
		GroupUserDetail
		TalkingGroupState
		MixedGroupUserState
		SoundDeviceInfo
		新增枚举:
		• UserType
		● PhoneUsedMode
		PhoneStatus
		TransferType
		新增接口:
		● 话机管理功能
		● 集群调度类功能
		GetVeision
		LoginToUPortal
		LogoutFromUPortal
		• SetConfig
		GetConfig
		UnInitializeAll
		新增回调事件:

日期	修订版本	描述
		● 话机管理类事件
		● 集群调类事件
		OnLoginToUPortalResult
		OnLogoutFromUPortal
		OnUnInitializeAllResult
		新增错误码:
		● 服务端错误码
		● VOICE错误码
		● TUP错误码
		修改:
		● Initialize: 原先登入配置中,设置 MCP IP的地址,现需设置 AgentServer的URI地址。
		● 保活机制: AgentGateWay没有保活 机制,原OCX内部有保活机制。
		● Hold: 在联动场景下,可以通过外 置话机保持多路呼叫。
		● <b>Dial</b> :在联动场景下,当座席话机 的IPT呼叫保持后,座席可以发起外 呼请求。
		● 学习态:现Agent的学习态是指座席 签出所有接警技能,原OCX座席是 存在学习态这个状态的。
		● <b>AgentStatusDetails</b> : SDK中存在该对象,但已经不再使用。
		● AgentStatusOverview: SDK中存在 该对象,但已经不再使用。
		● Room: SDK中存在该对象,但已经 不再使用。
		● <b>SetWorkStop</b> : 由于座席状态已经没有学习态,开关班采用重置技能的方式。
		● ResetAgentSkillGroups: 座席只能 重置本座席的技能。
		● QueryCallInfo: 两种呼叫类型。当前通话为IPCC呼叫,则返回IPCC呼叫信息; 如果为IPT呼叫,则返回IPT呼叫信息。
2016-12-31	V200R001C00	V200R001C00版本文档发布

CS) 2 最新修定记录

日期	修订版本	描述
2016-10-13	IPCC	新增接口和事件:
	V200R001C80SPC1	AddToBlackListEx
	00	QueryBlackListEx
		QueryAllBlackListEx
		OnAddToBlackListExResult
		OnQueryBlackListExResult
		OnQueryAllBlackListExResult
		OnReceiveP2PSMS
		OnClientDataWithoutCallFail
		修改事件名称:
		OnVCCallRecordFail
2016-07-30	IPCC V200R001C80	新增接口和事件:
		OnLinkagePhoneStatus
		OnLinkageCalloutNotify
		OnLinkageResultNotify
		OnReturnFromIVR

2 最新修定记录

日期	修订版本	描述
2016-04-30	IPCC V200R001C60	新增接口和事件:
		Whisper
		SuperviseStop
		SuperviseSwitch
		Mute
		QueryCallInfo
		ChangeMeetingMemberMode
		OnBeWhispered
		OnBeMonitored
		OnBeStopSupervise
		<b>OnMeetingMasterChange</b>
		<b>OnMeetingMemberModeChange</b>
		SetAutoAnswer
		SetCallData
		QueryCallData
		SetCustomerLevel
		QueryCallIDOnAgent
		QueryMaliciousCalls
		OnLongNoAnswer
		OnPhoneBusy
		OnPhoneOffhook
		SendMessage
		OnReceiveMessage
		QueryAgentSkillGroups
		QuerySkillGroupStatus
		HangupVCCall
		OnPickupVCCallResult
		OnNewVCCall
		OnOtherVCCallHangup
		OnStartRecordVCCall
		OnEndRecordVCCall
		OnVCCallLongNoAnswer
		OnDummyCallArrive
		SubscribeAgentEvent
		StopSubscribeAgentEvent
		OnSubscribeAgentEventResult
		<b>OnStopSubscribeAgentEventResult</b>
		OnAgentEventNotify

CS) 2 最新修定记录

日期	修订版本	描述
		修改接口事件:
		OnCallListChange
		SingleStepMeeting
		OnQueryCallListByGroupIdResult
		Initialize
2015-10-22	eSpace CC	新增以下接口及其结果事件:
	V200R001C50SPC1 03	TransferToSkillGroup
		QueryLostCalls
		QueryAllRedList
		QueryAllBlackList
		QueryLostCallsCount
		QueryAccessCodeList
		PollingCallEx
		StopQueryCallListByGroupId
2015-06-30	eSpace CC V200R001C50SPC1 02	增加多媒体呼叫(短信)释放通知事件 OnReceiveReleaseSMS、录音接口参数 改为recordID.

**3** 数据类型

3.1 类

3.2 枚举

# 3.1 类

# 3.1.1 Agent(会议成员)

Namespace	Com.Huawei.ServiceModel
Syntax	public class Agent : IEquatable <agent></agent>

## 方法

方法名	描述
Equals	指示当前对象是否等于同一类型的另一 个对象

#### 方法定义:

public bool Equals(
 Agent other;

#### 参数描述:

参数	类型	描述
other	Agent	与此对象进行比较的对象

#### 返回值:

类型	含义
bool	● true: 相等
	● false: 不相等

属性名	类型	描述
MemberName	string	会议成员名称
MemberType	MemberType	成员类型
PhoneNumber	string	会议成员号码

属性名	类型	描述
Status	MeetingStatus	会议状态

# 3.1.2 AgentInfo(座席信息)

Namespace	Com.Huawei.ServiceModel
Syntax	public class AgentInfo

## 属性

属性名	类型	描述
AgentId	string	座席工号
GroupId	string	技能队列ID
AgentStatus	OpStatus	座席电话状态,电话状态不再有学习态
PhoneNumber	string	电话号码
Password	string	座席密码
UserName	string	用户名
WebVoipAccount	string	内置WebVoip帐号
WebVoipPassWord	string	内置WebVoip帐号密码
VCPhoneNumber	string	视频终端号码

# 3.1.3 AgentStatusDetails(座席状态)

Namespace	Com.Huawei.ServiceModel
Syntax	public class AgentStatusDetails

注: SDK中存在该对象,但已经不再使用。

属性名	类型	描述
AgentId	string	座席工号

属性名	类型	描述	
AgentStatus	int	座席状态:	
		● 0: 未知	
		● 1: 签入状态, 瞬态	
		● 2: 签出状态	
		● 3: 示忙状态	
		● 4: 示闲状态	
		● 5: 整理态	
		● 6: 未使用,同示闲状态	
		● 7: 通话态	
		● 8: 休息态	
Extension	string	座席话机号码	
ExtensionStatus	int	座席话机状态:	
		● 0: 挂机	
		● 1: 摘机	
		● 2: 未知状态	
GroupId	string	技能队列ID	

# 3.1.4 AgentStatusOverview(座席状态总览)

Namespace	Com.Huawei.ServiceModel
Syntax	public class AgentStatusOverview

注: SDK中存在该对象,但已经不再使用。

属性名	类型	描述
AgentId	string	座席工号

属性名	类型	描述	
AgentStatus	int	座席状态:	
		● 0: 未知	
		● 1: 签入状态, 瞬态	
		● 2: 签出状态	
		● 3: 示忙状态	
		● 4: 示闲状态	
		● 5: 整理态	
		● 6: 未使用,同示闲状态	
		● 7: 通话态	
		● 8: 休息态	
GroupId	string	技能队列ID	

# 3.1.5 CallListDetails(排队电话信息)

Namespace	Com.Huawei.ServiceModel
Syntax	public class CallListDetails

属性名	类型	描述
CalleeNumber	string	被叫号码
CalleeType	DeviceType	被叫类型
CallerNumber	string	主叫号码
CallerType	DeviceType	主叫类型
CallId	string	呼叫标识
CallTime	string	呼入时间
GroupId	string	技能队列ID
CustomerLevel	string	客户优先级

## 3.1.6 CallListDetailsChanged(同组呼叫队列状态改变信息)

Namespace	Com.Huawei.ServiceModel.CallListDetails Com.Huawei.ServiceModel
Syntax	public class CallListDetailsChanged : CallListDetails

## 属性

属性名	类型	描述
BeIsAddCall	bool	是否加入队列
		● true: 是
		● false: 否

# 3.1.7 DialContent(呼叫内容)

Namespace	Com.Huawei.ServiceModel.Agent
	Com.Huawei.ServiceModel
Syntax	public class DialContent : Agent

此类继承Agent(会议成员)。

## 属性

属性名	类型	描述
CallId	string	呼叫标识
CallType	DeviceCallType	呼叫类型: 常规、组呼、私密
DeviceId	string	无线标识
PhoneType	DeviceType	电话类型

# 3.1.8 MeetingMessage(会议操作消息)

Namespace	Com.Huawei.ServiceModel
Syntax	public class MeetingMessage

#### 属性

属性名	类型	描述
MeetingId	string	会议ID
MeetingOperation	MeetingOperatio n	会议操作
MeetingType	MeetingType	会议类型
Members	List <agent></agent>	全体成员
Source	Agent	操作来源
Target	Agent	操作目标

# 3.1.9 PhoneEventOperation(电话事件处理)

Namespace	Com.Huawei.ServiceModel
Syntax	public class PhoneEventOperation

注: SDK中存在该对象,但已经不再使用。

#### 属性

属性名	类型	描述
Result	bool	处理结果
		● true: 成功
		● false: 失败
Tel	string	电话号码

# 3.1.10 PhoneStatusItem(电话状态项)

Namespace	Com.Huawei.ServiceModel
Syntax	public class PhoneStatusItem

注: SDK中存在该对象,但已经不再使用。

## 属性

属性名	类型	描述
AgentId	string	座席工号
ChangedTime	DateTime	变化时间
PhoneNumber	string	电话号码
SeatId	string	席位号
Status	OpStatus	状态

# 3.1.11 Room(会议室)

Namespace	Com.Huawei.ServiceModel
Syntax	public class Room

注: SDK中存在该对象,但已经不再使用。

## 构造方法

构告器名	描述
Room	初始化会议室

Room: 初始化会议室。

#### 方法定义:

```
public Room(
    Agent master,
    MeetingType meetingType)
```

#### 参数描述:

参数	类型	描述
master	Agent	会议主席
meetingType	MeetingType	会议类型

## 方法

方法名	描述
ChangeMaster	修改会议主席
DeleteAgent	移除指定成员
InsertAgent	加入新成员
UpdateAgent	更新成员状态
UpdateMeetingId	更新会议ID

ChangeMaster: 修改会议主席。

#### 方法定义:

```
public void ChangeMaster(
Agent agent
)
```

#### 参数描述:

参数	类型	描述
agent	Agent	会议成员

DeleteAgent: 移除指定成员。

#### 方法定义:

```
public void DeleteAgent(
    Agent target
)
```

#### 参数描述:

参数	类型	描述
target	Agent	会议成员

InsertAgent: 加入新成员。

#### 方法定义:

```
public void InsertAgent(
    Agent parent,
    Agent target
)
```

#### 参数描述:

eSDK ICP V200R001C10 接口参考 01(ECC, Windows,

CS) 3 数据类型

参数	类型	描述
parent	Agent	上一级成员
target	Agent	新成员

UpdateAgent: 更新成员状态。

#### 方法定义:

```
public void UpdateAgent(
   Agent target,
   MeetingStatus status
```

#### 参数描述:

参数	类型	描述
target	Agent	会议成员
status	MeetingStatus	会议状态

UpdateMeetingId: 更新会议ID。

#### 方法定义:

```
public void UpdateMeetingId(
    string meetingId
```

#### 参数描述:

参数	类型	描述
meetingId	String	会议ID

属性名	类型	描述
Agents	List< <b>Agent</b> >	会议成员集
AgentsSchema	XDocument	会议成员集配置Schema
Master	Agent	会议主席
MeetingType	MeetingType	会议类型
RoomId	string	会议室ID

## 3.1.12 WirelessDeviceInfo(无线设备信息)

Namespace	Com.Huawei.ServiceModel
Syntax	public class WirelessDeviceInfo

## 属性

成员名	类型	描述
DeviceId	string	设备编号
DeviceMode	DeviceWorkMode	设备工作模式
DeviceName	string	设备名称
DeviceType	DeviceType	设备类型
Status	bool	设备状态
		● true: 可用
		● false: 不可用

# 3.1.13 SkillGroup(技能队列)

Namespace	Com.Huawei.ServiceModel
Syntax	public class SkillGroup

## 属性

属性名	类型	描述
GroupId	int	技能队列ID
GroupName	string	技能队列名称

# 3.1.14 AccessCode(系统接入码配置信息)

Namespace	Com.Huawei.ServiceModel
Syntax	public class AccessCode

## 属性

属性名	类型	描述
MediaType	short	媒体类型:  ■ 1: 文字交谈  ■ 2: 单击通话 说明
SystemAccessCode	string	● 5: 普通语音通话等 系统接入码
AccessCodeDesc	string	对此接入码的功能性描述

# 3.1.15 RestReason(休息原因)

Namespace	Com.Huawei.ServiceModel
Syntax	public class RestReason

## 属性

属性名	类型	描述
RestCode	int	休息原因码
RestReasonInfo	string	休息原因信息

# 3.1.16 AgentCallInfo(当前通话信息)

Namespace	Com.Huawei.ServiceModel
Syntax	public class AgentCallInfo

成员名	类型	描述
CallID	string	呼叫标识
CallerNumber	string	主叫号码
CalleeNumber	string	被叫号码

成员名	类型	描述
GroupName	string	技能队列名称
TrunkNo	int	中继号
LogonTime	int	呼叫总时长,包括等待应答时长和通话时长

# 3.1.17 AgentSkillGroup(座席配置技能信息)

Namespace	Com.Huawei.ServiceModel
Syntax	public class AgentSkillGroup

#### 属性

属性名	类型	描述
GroupId	int	技能队列ID
IsConfiged	bool	座席是否配置了该技能队列的技能
IsSignin	bool	座席是否签入了该技能队列的技能

# 3.1.18 SkillGroupStatus(呼叫队列状态)

Namespace	Com.Huawei.ServiceModel
Syntax	public class SkillGroupStatus

属性名`	类型	描述
AvaliableAgents	int	当前可用的座席数目,即处于示闲状态的座席数目
GroupId	int	技能队列ID
EstWaitTime	int	座席应答一个呼叫之前的估计等待时间
LogonAgents	int	当前签入到这个技能队列的座席数
QueueSize	int	当前等待在队列中的呼叫数目
LongestQueueTime	int	队列中最长的呼叫等待时间(秒)
GroupName	string	技能队列名称

# 3.1.19 LostCall(丢话记录)

Namespace	Com.Huawei.ServiceModel
Syntax	public class LostCall

#### 属性

属性名	类型	描述
CallId	string	呼叫标识
Caller	string	主叫号码
Callee	string	被叫号码
Begintime	DateTime	呼叫开始时间
Losttime	DateTime	呼叫结束时间

# 3.1.20 MaliciousCall(骚扰记录)

Namespace	Com.Huawei.ServiceModel
Syntax	public class MaliciousCall

## 属性

属性名	类型	描述
CallId	string	呼叫标识
Caller	string	主叫号码
Callee	string	被叫号码
CallTime	DateTime	呼叫时间

# 3.1.21 RedList(优先级名单)

Namespace	Com.Huawei.ServiceModel
Syntax	public class RedList

## 属性名

属性名	类型	描述
PhonePrefix	string	用户电话号码
Priority	int	优先级,数值越大优先级越高

# 3.1.22 BlackList(黑名单)

Namespace	Com.Huawei.ServiceModel
Syntax	public class BlackList

## 属性

属性名	类型	描述
PhoneNumber	string	用户电话号码
Duration	long	锁定时长
CreatedBy	string	创建者
CreateDate	DateTime	创建时间
FilterMode	short	<ul><li>黑名单模式,涉及黑名单扩展事件才会返回</li><li>● 1: 拒绝</li><li>● 2: 降级</li></ul>

# 3.1.23 IvrInfo(IVR 流程信息)

Namespace	Com.Huawei.ServiceModel
Syntax	public class IvrInfo

属性名	类型	描述
Id	int	自动流程ID
ServiceNo	int	自动流程处理的业务类型
InNo	string	流程接入码,最大字符长度24

属性名	类型	描述
Description	string	自动流程描述,最大字符长度200

# 3.1.24 LocalInfo(本地信息)

Namespace	Com.Huawei.ServiceModel
Syntax	public class LocalInfo

## 属性

属性名	类型	描述
LocalIp	string	本地IP
LocalSipPort	int	本地Sip端口,端口号范围为1024~65535
LocalAudioPort	int	本地音频端口,端口号范围为1024~65535

# 3.1.25 TalkingGroup(集群群组信息)

Namespace	Com.Huawei.ServiceModel
Syntax	public class TalkingGroup

属性名	类型	描述
talkGroupName	string	集群群组名称,长度不超过32位
talkGroupNumber	string	集群群组号码,由数字和大小写字母组成, 长度不超过32位,该字段为集群群组唯一标 识
talkGroupDesc	string	集群群组描述,长度不超过50位
talkGroupType	int	集群群组类型  ● 1: 普通群组  ● 2: 混编有线/无线集群群组

属性名	类型	描述
radioGroupNumber	List <string></string>	无线集群群组号码,由数字和大小写字母组成,长度不超过32位,数量不得超过4个 说明 当talkGroupType为2(混编有线/无线集群群组) 时,该参数非必填
dispatchNumber	List <string></string>	调度席号码,由数字和大小写字母组成,长 度不超过32位,数量1~20个
otherUserNumber	List <string></string>	其他用户号码,由数字和大小写字母组成,长度不超过32位,数量不得超过36个 说明 当talkGroupType为1(普通群组)时,该参数必须 为空
receiveOnly	string	调度席是否可抢权

# 3.1.26 RadioGroup(无线集群群组信息)

Namespace	Com.Huawei.ServiceModel
Syntax	public class RadioGroup

## 属性

属性名	类型	描述
radioTKGroupNumber	string	无线集群群组号码,该字段为无线集群 群组唯一标识
radioMode	string	无线集群群组制式
radioTKGroupType	string	无线集群群组类型
radioTKGroupDesc	string	无线集群群组描述

# 3.1.27 PageInfo(分页信息)

Namespace	Com.Huawei.ServiceModel
Syntax	public class PageInfo

3 数据类型

### 属性

属性名	类型	描述
pageNumber	string	当前页数
countPerPage	int	每页最大个数
hasNext	bool	是否有下一页
		● true: 有
		● false: 没有
currentCount	int	当前页个数
totalCount	int	所有数量

## 3.1.28 GroupUserState(集群群组用户状态信息)

Namespace	Com.Huawei.ServiceModel
Syntax	public class GroupUserState

## 属性

属性名	类型	描述
talkGroupNumber	string	集群群组号码
userNumber	string	操作状态变更的用户标识,包括无线集群 群组号码、调度席号码和其他用户号码
operationType	string	集群群组用户操作类型,参见 <b>表3-1</b>
operationState	string	集群群组用户操作状态,参见 <b>表3-2</b>

### 表 3-1 集群群组用户操作类型定义

集群群组用户操作类型	说明
subscribe	订阅资源操作
monitor	监听资源操作
select	选择资源操作
ptt	PTT资源操作,可以是集群群组用户发起的PTT操作或调度席针对无线集群群组的下行PTT操作
groupCall	集群群组组呼资源操作

集群群组用户操作类型	说明
mixedGroupCall	混编集群群组组呼资源操作

#### 表 3-2 集群群组用户操作状态定义

操作类型	集群群组操作状态	说明	
subscribe	notInOperation	未订阅状态	
	inOperation	订阅状态	
monitor	notInOperation	未监听状态	
	inOperation	监听状态	
	inProcess	正在监听处理过程中	
select	notInOperation 未选择状态		
	inOperation	选择状态	
ptt	notInOperation	未PTT状态	
	inOperation	PTT状态	
	inProcess	当前激活操作正在处理中	
	inCancelProcess	当前取消激活操作正在处理中	
	inDownPttOperation	当集群群组用户为无线集群时, PTT 下行激活	
	inDownPttProcess	当集群群组用户为无线集群时, PTT 下行请求处理中	
groupCall	notInOperation	未呼叫状态	
	inOperation	集群群组呼叫状态	
	inProcess	集群群组正在呼叫处理过程中	
mixedGroup	notInOperation	未加入会议	
CallState	muted	已加入会议但未授予话权	
	inProcess	正在加入会议过程中	
	unmuted	己加入会议且已授予话权	

## 3.1.29 GroupUserDetail(集群群组用户所有状态信息)

Namespace Com.Huawei.ServiceModel
-----------------------------------

3 数据类型

|--|

## 属性

属性名	类型	描述
talkGroupNumber	string	集群群组号码
userNumber	string	集群群组用户号码,唯一标识,包括无 线集群群组号码、调度席号码和其他用 户号码
subscribeState	string	用户当前对集群群组的订阅状态,参见 <b>集群群组用户操作状态定义</b>
monitorState	string	用户当前对集群群组的监听状态,参见 <b>集群群组用户操作状态定义</b>
selectState	string	用户当前对集群群组的选择状态,参见 <b>集群群组用户操作状态定义</b>
pttState	string	用户当前对集群群组的ptt状态,参见 <b>集</b> <b>群群组用户操作状态定义</b>
groupCallState	string	用户当前对集群群组的组呼状态,参见 <b>集群群组用户操作状态定义</b>
isInOtherGroupCall	bool	用户是否已经在其他集群群组呼叫中
mixedGroupCallState	string	用户当前对混编集群群组的组呼状态, 参见 <b>集群群组用户操作状态定义</b>

## 3.1.30 TalkingGroupState(集群群组状态信息)

Namespace	Com.Huawei.ServiceModel
Syntax	public class TalkingGroupState

## 属性

属性名	类型	描述
talkGroupNumber	string	集群群组号码
emergencyUser	string	紧急呼叫用户标识,包括无线集群群 组号码、调度席号码和其他用户号码
userStateList	List <groupuserdetail></groupuserdetail>	属于该集群群组的用户的所有状态信 息列表

CS) 3 数据类型

属性名	类型	描述
subscribeUsers	List <string></string>	订阅该集群群组的用户号码列表

## 3.1.31 MixedGroupUserState(混编集群群组用户状态信息)

Namespace	Com.Huawei.ServiceModel
Syntax	public class MixedGroupUserState

### 属性

属性名	类型	描述
userNumber	string	集群群组用户号码,唯一标识,包括无 线集群群组号码、调度席号码和其他用 户号码
userType	UserType	用户类型
state	string	用户状态  ● inProcess: 邀请用户加入混编集群群组呼叫
		● muted: 取消话权
		● unmuted: 授予话权
		● notInOperation: 将用户移出混编集群群组呼叫

## 3.1.32 SoundDeviceInfo(音频设备信息)

Namespace	Com.Huawei.ServiceModel
Syntax	public class SoundDeviceInfo

## 属性

属性名	类型	描述
deviceName	string	设备名称
deviceIndex	string	设备索引号

3 数据类型

## 3.2 枚举

## 3.2.1 CallingPhoneScope(呼入电话范围)

Namespace	Com.Huawei.ServiceModel
Syntax	public enum CallingPhoneScope

## 成员

成员名	描述
OutsideCalling	外线
InsideCalling	内线
Unknown	未知

## 3.2.2 DeviceCallType(设备呼叫类型)

Namespace	Com.Huawei.ServiceModel
Syntax	public enum DeviceCallType

## 成员

成员名	描述
ConventionCall	常规呼叫
GroupCall	组呼:呼叫群组号码
SecretCall	私密呼叫,即点呼

## 3.2.3 DeviceModule(硬件通讯设备)

Namespace	Com.Huawei.ServiceModel
Syntax	public enum DeviceModule

3 数据类型

### 成员

成员名	描述
NONE	无
CTI	路由平台服务器
IVR	IVR设备: Interactive Voice Response,即互动式语音应答
PBX	PBX设备: 专用电话设备中的一种交换实体,可接入公共电话交换网
TCU	TCU设备,该字段为保留字段,未被使用
MDR	MDR设备,该字段为保留字段,未被使用
Phone	软电话

## 3.2.4 DeviceType(设备类型)

Namespace	Com.Huawei.ServiceModel
Syntax	public enum DeviceType

## 成员

成员名	描述
Unknown	未知
Phone	普通电话
WLDigitalDevice	无线数字设备
WLSimulateDevice	无线模拟设备
SW	短波
Mobilephone	手机
Seat	座席
LTE	eLTE手台

## 3.2.5 DeviceWorkMode(设备工作模式)

Namespace	Com.Huawei.ServiceModel
Syntax	public enum DeviceWorkMode

CS) 3 数据类型

### 成员

成员名	描述
None	常规
TETRA	集群无线电

## 3.2.6 HoldMode(保持模式)

Namespace	Com.Huawei.ServiceModel
Syntax	public enum HoldMode

## 成员

成员名	描述
Hold	保持
HoldCancel	取消保持

## 3.2.7 MeetingOperation(会议操作)

Namespace	Com.Huawei.ServiceModel
Syntax	public enum MeetingOperation

成员名	描述
Create	创建
Invite	邀请
Accept	接受
Change	变更
Remove	移除
Finish	结束

3 数据类型

## 3.2.8 MeetingStatus(会议状态)

Namespace	Com.Huawei.ServiceModel
Syntax	public enum MeetingStatus

## 成员

成员名	描述
Loading	连接中
Meeting	会议中
Faulted	出错
Without	会议外

## 3.2.9 MeetingType(会议类型)

Namespace	Com.Huawei.ServiceModel
Syntax	public enum MeetingType

## 成员

成员名	描述
MultiTalking	多呼会议
CallThirdParty	三方会议

## 3.2.10 MemberType(成员类型)

Namespace	Com.Huawei.ServiceModel
Syntax	public enum MemberType

成员名	描述
Phone	普通电话

3	数	坭	米	刑
٦.	7+ V	114	$\rightarrow$	147

成员名	描述
Seat	普通座席
P25	无线集群
VHF	无线数字(高频)
UHF	无线数字(特高频)

## 3.2.11 OpFlags(操作标识)

Namespace	Com.Huawei.ServiceModel
Syntax	public enum OpFlags

注: SDK中存在该对象,但已经不再使用。

成员名	描述
Start	开始
Close	关闭
Pickup	摘机
Handup	挂机
SetBusy	置忙
SetIdle	置闲
Dial	单呼
DialCancel	取消单呼
Hold	呼叫保持
ConsultCall	协商呼叫
CallThirdParty	三方会议
EndDiscussing	结束会议
PTT	抢权
SetPTTManual	设置手动抢权
SoftPlayRecord	播放录音
SoftPlaySeekTo	播放进度跳转

3 数据类型

成员名	描述
SoftStopPlay	停止播放
SoftPlayPause	暂停播放
SoftPlayResume	继续播放
ForceRelease	拦截
Barge	强插
Monitor	监听
MonitorToBarge	监听转强插
MonitorToForceRelease	监听转拦截
AddToBlackList	添加黑名单
QueryAgentStatus	查询座席状态
MutliDial	多重呼叫
MakeMeeting	创建会议
ReleaseMeeting	释放会议
DialEx	呼叫拓展

## 3.2.12 OpStatus(操作状态)

Namespace	Com.Huawei.ServiceModel
Syntax	public enum OpStatus

成员名	描述
Closed	无
Fault	出错
Idle	空闲
Busy	忙碌
Talking	通话中
Ringing	来电中
Calling	去电中

CS) 3 数据类型

成员名	描述
Holding	保持中
Occupy	占用:座席质检时的状态
Rest	休息
Studying	学习
Consulting	协商中
Monitoring	监听中
MultiTalking	会议中
Barging	强插中

## 3.2.13 PTTMode(PTT 模式)

Namespace	Com.Huawei.ServiceModel
Syntax	public enum PTTMode

## 成员

成员名	描述
Manual	手动
Auto	自动

## 3.2.14 PTTOperate(PTT 操作类型)

Namespace	Com.Huawei.ServiceModel
Syntax	public enum PTTOperate

成员名	描述
PutUp	PTT抬起,即强权操作
PutDown	PTT按下,即放权操作

eSDK ICP V200R001C10 接口参考 01(ECC, Windows,

3 数据类型

## 3.2.15 EventTypeEnum(事件类型)

Namespace	Com.Huawei.ServiceModel
Syntax	public enum EventTypeEnum

## 成员

成员名	描述
OnLongNoAnswer	久不应答事件

## 3.2.16 UserType(用户类型)

Namespace	Com.Huawei.ServiceModel
Syntax	public enum UserType

## 成员

成员名	描述
RadioTrunk	无线集群群组用户
Dispatcher	调度席用户
OtherUser	其他类型用户

## 3.2.17 PhoneUsedMode(话机使用模式)

Namespace	Com.Huawei.ServiceModel
Syntax	public enum PhoneUsedMode

成员名	描述
PhoneEmbeddedMode	嵌入式内置软电话模式
PhoneLinkedMode	外置话机采用联动模式

3 数据类型

## 3.2.18 PhoneStatus(话机状态)

Namespace	Com.Huawei.ServiceModel
Syntax	public enum PhoneStatus

## 成员

成员名	描述
OnLine	在线
OffLine	离线
Ringing	振铃
OffHook	摘机
OnHook	挂机
Established	通话
Outing	呼出
Hold	保持
UnHold	取消保持
Mute	静音
Unmute	取消静音
BlindTransfer	盲转:即释放转,不考虑转出是否成功

## 3.2.19 TransferType(转移类型)

Namespace	Com.Huawei.ServiceModel
Syntax	public enum TransferType

## 成员名

成员名	描述
BlindTransfer	盲转: 即释放转,不考虑转出是否成功
ConsultTransfer	咨询转:即先查询转出目标座席状态,如果是示闲状态才转出。暂不支持

eSDK ICP V200R001C10 接口参考 01(ECC, Windows,

CS) 3 数据类型

## 3.2.20 AudioDeviceType(声卡设备类型)

Namespace	Com.Huawei.ServiceModel
Syntax	public enum AudioDeviceType

## 成员名

成员名	描述
MainDevice	主设备
AidDevice	辅助设备

# 4 IDevice(设备服务基本接口)

#### 所属类

Namespace	Com.Huawei.Device.Contract	
Syntax	public interface IDevice	
Implemented Class	Com.Huawei.Device.HuaweiDevice	

4.1 IDevice Methods: 运行管理功能

4.2 IDevice Methods: 语音接续功能

4.3 IDevice Methods: 状态管理类功能

4.4 IDevice Methods: 其他查询类功能

4.5 IDevice Methods: 电话会议类功能

4.6 IDevice Methods: 座席管理类功能

4.7 IDevice Methods: 录音管理功能

4.8 IDevice Methods: 无线资源类功能

4.9 IDevice Methods: 短信服务类功能

4.10 IDevice Methods:邮件服务类功能

4.11 IDevice Methods: 视频通话类功能

4.12 IDevice Methods: 话机管理功能

4.13 IDevice Methods: 集群调度类功能

## 4.1 IDevice Methods: 运行管理功能

## 4.1.1 Initialize(启动软电话)

## 接口描述

签入座席,签入后座席为示忙状态。该接口包含是否启动自动应答、是否启动整理态、是否强制签入、是否启动联动等功能。

### 注意事项

- 需要座席的软电话签入客户端(eSpace iAgent)成功后,才能使座席与软电话真正绑定成功。
- 如果座席签入为强制签入,签入时该座席在其它客户端已经签入,则其它客户端 座席会被强制签出。

## 方法定义

```
void Initialize(
   AgentInfo agentInfo,
   Dictionary<string, string> settings
```

### 参数描述

参数	类型	参数规则		描述
agentInfo	AgentInfo	AgentId	必选,长度 为1~5位数字	座席工号
		PhoneNumber	必选,长度 为1~24位数 字	电话号码
		PassWord	可选,长度 为1~32位字 符串	座席密码
	VCPhoneNu mber	可选,长度 为1~24位数 字	VC终端号码	
settings	Dictionary <string,< td=""><td>HuaweiCTISe rverIP</td><td>必选</td><td>主Agentgateway服务器IP地址。</td></string,<>	HuaweiCTISe rverIP	必选	主Agentgateway服务器IP地址。
string>	HuaweiBacku pCTIServerIP	可选	备Agentgateway服务器IP地址	
	HuaweiTimeo ut	可选,非负 整数	超时时长35~60(毫秒),超 出范围取边界上下值	
		HuaweiWebS erviceUrl	必选	主节点主Web 服务地址URL

参数	类型	参数规则		描述
	HuaweiBacku pWebService Url	可选	主节点备Web 服务地址URI	
		DRHuaweiWe bServiceUrl	可选	容灾节点主Web 服务地址 URL
		HuaweiDRHu aweiBackup WebServiceU rl	可选	容灾节点主Web 服务地址 URL
		HuaweiOutbo undPrefix	可选,长度 为1~24位字 符串	出局码
		HuaweiExtens ionPrefix	可选,长度 为1~24位字 符串	分机前缀
		HuaweiExtens ionLength	可选,长度 为1~20位数 字	分机长度
		HuaweiSetAu toAnswer	可选	是否自动应答 <ul> <li>true: 启用</li> <li>false: 不启用</li> </ul>
		HuaweiIsEna bleACW	可选	是否启用整理态 <ul> <li>true: 启用</li> <li>false: 不启用</li> </ul>
		HuaweiPhone Linkage	可选	座席话机工作模式  ● 0: 外置话机联动模式  ● 1: 普通模式  ● 2: 话机长通模式  说明 如果已经调用PhoneInitialize将 话机初始化为外置话机联动模 式或内置软电话模式,则不允

可选

## 返回值

#### IDevice Events. On Initialize Result

IsForceLogin

许设置为长通模式

● 0: 不强制登录● 1: 强制登录

是否强制登录

#### 使用示例

```
//c# code
//定义软电话接口参数,需读取xml配置文件初始化
private AgentInfo _agentInfo = null;
private Dictionary<string, string> _settings = null;

//获取调用接口的IDevice实例,后续接口都需用该实例
public IDevice device =
IDeviceProvider.GetIDevice(frmMainDeviceCallback.GetMainDeviceCallback(this)) as IDevice;

private void btnInitial_Click(object sender, EventArgs e)
{
    //启用软电话的两参数
    var agentInfo = this._agentInfo;
    var settings = this._settings;

    //调用启动软电话接口
    device.Initialize(agentInfo, settings);
}
```

## 4.1.2 UnInitialize(关闭软电话)

### 接口描述

关闭软电话,即座席签出。

第三方应用已经完成业务场景需关闭应用程序或者第三方应用需要签入另一个座席工号时(例如:质检座席与普通座席切换)需调用该接口签出已签入的座席。

## 注意事项

- 前提是座席已经签入。
- 座席话机必须处于挂机状态。

## 方法定义

void UnInitialize()

## 参数描述

无。

## 返回值

#### IDeviceEvents.OnUnInitializeResult

## 使用示例

```
//c# code
private void btnUnInitial_Click(object sender, EventArgs e)
{
    //关闭软电话
    device.UnInitialize();
}
```

## 4.1.3 GetVeision(版本号查询)

### 接口描述

查询当前SDK版本号。版本号名称例如: "V200R001C00", "V200R001C10",用户可根据获取到的版本号去查看相应的接口参考、开发指南等文档资料。

### 方法定义

string GetVersion()

### 参数描述

无。

#### 返回值

类型	描述	
string	SDK版本号,例如"V200R001C10"	

### 示例代码

//c# code //获取SDK版本号 string result = device.GetVersion();

## 4.1.4 LoginToUPortal(登录 UPortal)

#### 接口描述

登录UPortal系统,如果需要使用集群调度功能或话机同步登录功能,则必须先调用该接口进行初始化。

- 集群调度功能:集群调度由UPortal通过Restful接口对外提供,所以需要通过该接口初始化与UPortal之间的会话资源。初始化完该接口后,紧接着会初始化话机为内置软电话模式。
- 话机同步登录功能:使用外置话机,开启话机联动,并通过PC端同步登录登出话机。此时由于话机同步登录登出功能由UPortal提供,所以也需要调用该接口初始化与UPortal的会话资源。初始化完该接口后,如果只是使用IPCC线路的话机联动,则不需要再初始化话机资源,否则需要初始化话机为联动模式,开启IPT线路的联动能力。

## 注意事项

前提是当前未登录同一帐号。

## 方法定义

void LoginToUPortal(
 string serverAddressInfo,
 string userName,
 string password

string initNum

### 参数描述

参数	类型	参数规则	描述
serverAddressInfo	string	必选	服务器地址信息:包括协议类型(http/ https)、合法的IP地址与端口号、访问路 径。
			支持多个服务器地址 <b>说明</b>
			● 当UPortal的能力通过API Gateway统一对外 暴露时,格式举例为: https:// 10.10.10.10:443/v1/EC
			● 当与UPortal直接对接时,格式举例为: https://10.10.10.10:443
			● 多服务器地址遵循以下规则: rest和websocket主备服务器一致时格式举 例为: https://10.245.20.23:8443 https:// 10.245.20.25:8443
			rest和websocket主备服务器不一致时格式 举例为: https://10.245.20.23:8443 https:// 10.245.20.25:8443;10.245.20.24 10.245.20.29
userName	string	必选	调度席帐号
password	string	必选	帐号密码
initNum	string	可选	话机初始号码
			<b>说明</b> 如果话机初始号码不为空,则进行同步登录; 如果初始号码包含域名,则配置文件ECC.xml 上的域名配置huawei.com可为空。

## 返回值

#### ${\bf On Login To UP or tal Result}$

### 使用示例

```
//c# code
string uPortalURI = "https://10.175.0.248:8443"; //uPortal服务地址
string uPortalAccount = "admin"; //帐号
string uPortalPwd = "1qaz@WSX"; //密码
string initNum = """; //话机号码可为空
//登入uPortal
device.LoginToUPortal(uPortalURI, uPortalAccount, uPortalPwd, initNum);
```

## 4.1.5 LogoutFromUPortal(登出 UPortal)

## 接口描述

登出UPortal服务系统。一般在第三方应用程序退出之前调用,另外调用UnInitializeAll接口也可以实现这一功能。

eSDK ICP V200R001C10 接口参考 01(ECC, Windows, 4 IDevice(设备服务基本接口)

### 注意事项

当前已登录UPortal服务系统。

## 方法定义

void LogoutFromUPortal()

### 参数描述

无。

### 返回值

On Log out From UP ortal

### 使用示例

//c# code //登出uPortal服务 device.LogoutFromUPortal();

## 4.1.6 UnInitializeAll(回收所有资源)

### 接口描述

回收所有资源,即签出座席签入的所有服务,包括AgentGateway服务、CTI服务、 UPortal服务、软电话。

## 方法定义

void UnInitializeAll()

## 参数描述

无。

### 返回值

IDeviceEvents.OnUnInitializeAllResult

### 使用示例

//c# code //回收所有资源 device.UnInitializeAll();

## 4.2 IDevice Methods: 语音接续功能

## 4.2.1 Pickup(摘机)

### 接口描述

座席电话摘机。外置软电话通过接入码呼入,系统分配座席后,座席侧会收到 IDeviceEvents.OnNewCall事件,收到事件后需调用该接口接听电话。

## 注意事项

座席需处于示闲状态,才可收到来电。

### 方法定义

void Pickup()

### 参数描述

无。

#### 返回值

IDeviceEvents.OnAnswerCallResult

#### 使用示例

//c# code //座席电话摘机 device.Pickup();

## 4.2.2 Hangup(挂机)

#### 接口描述

挂机或取消线路未通的呼叫。

接口支持以下几种挂机场景:

- 协商中座席挂机:发起协商呼叫的座席主动挂断与被协商对象的呼叫(挂机后,仅是释放协商通话,原被保持通话未被释放)。
- 监听中座席挂机: 高级座席在监听其它座席时, 主动挂机。
- 强插中座席挂机:高级座席在强插入其它座席的通话中时,主动挂机。
- 取消线路未通的呼叫:在座席拨打电话过程中,呼叫目标未接听来电的情况,座席挂机取消这次呼叫。
- 普通挂机:除以上场景外挂机,全部挂机后,默认座席为强制示忙状态。

#### □ 说明

轮询呼叫中,座席挂断正在呼叫的号码,则会继续呼叫下一个号码。

## 注意事项

● 挂机后,如果对方为座席电话时,则座席侧会收到系统主动返回的 IDeviceEvents.OnOtherHangup事件。

4 IDevice(设备服务基本接口)

### 方法定义

void Hangup()

### 参数描述

无。

### 返回值

#### IDeviceEvents.OnHangUpResult

### 使用示例

//c# code //座席电话挂机 device.Hangup();

## 4.2.3 Dial(单呼)

### 接口描述

单呼包括对普通电话、无线数字设备、无线模拟设备、短波、手机发起常规呼叫、点呼、组呼。在联动场景下当座席话机的IPT呼叫保持后,座席可以发起外呼请求。

目前该接口支持以下场景:

- 1. 座席常规呼出:指座席向局外用户(固话、手机等)、内部话机(非座席)发起呼叫。
- 2. 座席内呼座席:指座席向内部其它座席(绑定的分机号)发起呼叫。
- 3. 座席呼叫eLTE: 指座席向LTE手台/集群群组发起呼叫。座席呼叫eLTE动态群组,动态群组中需要添加网关代理用户。
- 4. 座席呼叫Tetra: 指座席向车载Tetra手台/集群群组发起呼叫。

## 方法定义

```
void Dial(
DialContent dialContent
```

### 参数描述

参数	类型	参数规则		描述
dialContent	DialConten t	PhoneNumber	必选,长度为1~24位数 字	电话号码
		PhoneType	必选, <b>DeviceType</b>	设备类型
		CallType	必选,DeviceCallType	设备呼叫类型

### □说明

DialContent类继承于Agent类。

## 返回值

#### IDeviceEvents.OnDialResult.

## 使用说明

单呼时,针对被叫对象,主叫侧座席会收到系统主动触发的呼出状态返回事件: IDeviceEvents.OnCallOutStatusChange。

针对不同的呼叫场景,对应接口传入内容如下表所示。

场景类型	被叫类型	被叫号码	呼叫类型
座席常规呼出	<ul> <li>DialContent.Pho neType=DeviceT ype.Phone (普通电话)</li> <li>DialContent.Pho neType=DeviceT ype.Mobilephone (手机)</li> <li>DialContent.Pho neType=DeviceT ype.Unknown (其它)</li> </ul>	出局前缀(如果是外线)+业务侧传入的被叫号码; DialContent.PhoneNumber=业务侧传入的被叫号码。	DialContent.CallTyp e=DeviceCallType.C onventionCall(常 规呼叫)
座席内呼座席	接口内部已实现通过传入的DialContent.PhoneNumber查询判断是否为座席,无须从业务侧传入DialContent.PhoneType=DeviceType.Seat(座席)来判断	座席绑定的分机号码; DialContent.PhoneNumber=业务侧传入的分机号码。	DialContent.CallTyp e=DeviceCallType.C onventionCall(常 规呼叫)
座席呼叫eLTE	DialContent.PhoneT ype=DeviceType.LT E (eLTE)	LTE手台号码(点呼)/LTE群组号(组呼); DialContent.PhoneNumber=业务侧传入的LTE手台号码或LTE群组号。	● DialContent.Call Type=DeviceCall Type.SecretCall (私密呼叫,即 点呼) ● DialContent.Dev iceCallType=Dev iceCallType.Gro upCall (组呼)

场景类型	被叫类型	被叫号码	呼叫类型
座席呼叫Tetra	● DialContent.Pho neType=DeviceT ype.SW (短 波) ● DialContent.Pho neType=DeviceT ype.WLDigitalD evice (无线数 字设备) ● DialContent.Pho neType=DeviceT ype.WLSimulate Device (无线模 拟设备)	无线设备ID(即Tetra集群呼叫从资源池分配的网关号码); DialContent.PhoneNumber=业务侧传入的无线设备ID,只够各ID,只够该号码能够到一个空间,就够到一个空间,不线设备ID)。	● DialContent.Call Type=DeviceCall Type.SecretCall (私密呼叫,即点呼) ● DialContent.Call Type=DeviceCall Type.GroupCall (组呼)

## 使用示例

```
//c# code
private void btnDial_Click(object sender, EventArgs e)
{
    DialContent dialContent = new DialContent();
    //呼叫号码
    dialContent. PhoneNumber = this. textPhoneNumber. Text. Trim();
    //呼叫设备类型
    dialContent. PhoneType = DeviceType. Phone;
    //呼叫类型
    dialContent. CallType = DeviceCallType. ConventionCall;
    //单呼
    device. Dial(dialContent);
}
```

## 4.2.4 Hold(保持)

### 接口描述

保持/取消保持通话。

通话中的座席需紧急外呼其他电话时,需要保持当前通话,然后再单呼其它紧急的电话;座席接听来电后,如果无法解决来电问题时需要先保持当前通话,然后协商呼叫其它座席帮忙解决。外呼或协商呼叫结束后,座席可取消保持后恢复与被保持用户的通话。

## 注意事项

- 座席已经在通话状态。
- 非联动场景只能保持一路呼叫,联动场景可以通过外置话机保持多路呼叫。
- 保持的通话无法被挂断,需先调用该接口取消保持。

eSDK ICP V200R001C10 接口参考 01(ECC, Windows,

### 方法定义

```
void Hold(
HoldMode holdMode
)
```

### 参数描述

参数	类型	参数规则	描述
holdMode	HoldMode	必选	保持模式,枚举类型

### 返回值

#### IDeviceEvents.OnHoldResult

### 使用示例

```
//c# code
//保持通话
device. Hold(HoldMode. Hold);
//取消保持的通话
device. Hold(HoldMode. HoldCancel);
```

## 4.2.5 ConsultCall(协商呼叫)

## 接口描述

协商呼叫。座席A接听报警人来电后,如无法解答,需要协商呼叫另外的座席B来解答问题。协商呼叫成功后两座席形成通话,报警人电话被保持。

## 注意事项

- 发起协商呼叫座席的通话需要先被保持,被协商呼叫座席需置闲。
- 协商的对象为座席。
- 有线、无线都支持协商呼叫。

## 方法定义

```
void ConsultCall(
DialContent dialContent
)
```

## 参数描述

参数	类型	参数规则		描述
dialContent	DialContent	PhoneNumber	必选,长度为1~24位 数字	电话号码

## 返回值

#### IDeviceEvents.OnConsultCallResult

#### 使用示例

```
//c# code
private void btnConsult_Click(object sender, EventArgs e)
{
    DialContent dialContent = new DialContent();
    //呼叫号码需为座席绑定的号码
    dialContent.PhoneNumber = this.textPhoneNumber.Text.Trim();
    //呼叫设备类型
    dialContent.PhoneType = DeviceType.Phone;
    //呼叫类型
    dialContent.CallType = DeviceCallType.ConventionCall;
    //协商呼叫
    device.ConsultCall(dialContent);
}
```

## 4.2.6 EndConsultCall(结束协商呼叫)

#### 接口描述

结束协商呼叫请求。

座席A发起协商呼叫座席B,座席A咨询座席B能否解答报警人问题,如果座席B也不能解答或已经告知解答方案,则调用该接口结束协商呼叫。

#### 注意事项

座席发起协商呼叫成功。

## 方法定义

void EndConsultCall()

### 参数描述

无。

#### 返回值

#### IDeviceEvents.OnEndConsultCallResult

### 使用示例

```
//c# code
//结束协商呼叫
device.EndConsultCall();
```

## 4.2.7 TransferToAgent(协商转移)

### 接口描述

在电话协商过程中,将呼叫转移给被协商对象。

座席A发起协商呼叫座席B,座席A咨询座席B能否解答报警人问题,如果座席B能够解答,则调用该接口将报警人的电话转移过去。协商转移成功后形成座席B与报警人的通话,座席A的通话被挂断。

### 注意事项

- 座席已经协商呼叫成功。
- 被转移方会收到系统主动返回的IDeviceEvents.OnTransferInform事件。

### 方法定义

void TransferToAgent()

### 参数描述

无。

### 返回值

IDevice Events. On Transfer To Agent Result

#### 使用示例

```
//c# code
//协商转移
device.TransferToAgent();
```

## 4.2.8 SingleStepTransfer(非监督转移)

### 接口描述

座席电话通话过程中将呼叫转移给指定对象(电话)。

非监督转移为单步释放转,即座席在转出号码后便将原呼叫释放。不论转出是否成功,呼叫都将被转移到目标号码。

## 注意事项

- 座席的通话为非协商通话。
- 如转出到座席电话,转入方座席需要置闲。
- 如果当前通话为座席外呼通话时,不支持再转外部电话。

## 方法定义

```
void SingleStepTransfer(
   string sourceAgent,
   string targetNumber
)
```

### 参数描述

参数	类型	参数规则	描述
sourceAgent	string	可选,预留未使用	座席工号
targetNumber	string	必选,长度为1~24位 数字	目标号码:可以是内线或外线任 何号码

4 IDevice(设备服务基本接口)

### 返回值

#### IDeviceEvents.OnSingleStepTransferResult

## 使用示例

```
//c# code
private void btnBlindTransfer_Click(object sender, EventArgs e)
{
    //座席签入时保存AgentInfo对象
    string agentid=this._agentInfo.AgentID;
    //转移的电话号码
    string phoneNumber=this.txtPhoneNumber.Text.Trim();

    //非监督转移
    device.SingleStepTransfer(agentid, phoneNumber);
}
```

## 4.2.9 SingleStepSuccTransfer(呼叫成功转移)

### 接口描述

座席电话通话过程中,将呼叫转移给指定对象(电话)。

呼叫成功转移为单步成功转,即座席在转出后,待转出号码成功应答后,再将呼叫转 移到目标号码,否则转出失败,原通话保持不变。

## 注意事项

座席的通话需为非协商通话。

## 方法定义

```
void SingleStepSuccTransfer(
   string sourceAgent,
   string targetNumber
)
```

### 参数描述

参数	类型	参数规则	描述
sourceAgent	string	可选,预留未使用	座席工号
targetNumber	string	必选,长度为1~24位 数字	目标号码:可以是内线或外线 任何号码

### 返回值

#### IDevice Events. On Single Step Succ Transfer Result

### 使用示例

```
//c# code
private void btnSingleStepSuccTransfer_Click(object sender, EventArgs e)
{
//座席签入时保存AgentInfo对象
```

```
string agentid=this._agentInfo.AgentID;
//转移的目标电话号码
string phoneNumber=this.txtPhoneNumber.Text.Trim();

//呼叫成功转
device.SingleStepSuccTransfer(agentId, PhoneNumber);
}
```

## 4.2.10 QueryCurrentCallingScope(查询当前呼叫范围)

### 接口描述

查询当前呼入电话是外线还是内线。

### 方法定义

void QueryCurrentCallingScope()

### 参数描述

无。

#### 返回值

IDevice Events. On Query Current Calling Scope Result

## 使用示例

```
//c# code
//查询当前呼叫范围
device. QueryCurrentCallingScope();
```

## 4.2.11 PollingCall(轮询呼叫)

### 接口描述

对多个单呼对象轮询发起呼叫。

当座席号码需要依次单呼多个对象号码时,可调轮询呼叫接口。呼叫次序按照对象列 表排序。如果呼叫上一个对象失败,则会呼叫列表后面的对象。如果呼叫对象成功, 则不会再呼叫后面的对象。

## 注意事项

轮询号码呼出时,对每个号码的单呼,主叫侧座席会收到系统主动触发的呼出状态返回事件: IDeviceEvents.OnCallOutStatusChange。

## 方法定义

```
void PollingCall(
   List<DialContent> dialContents,
   int timeout
)
```

## 参数描述

参数	类型	参数规则		描述
dialContents	List <dialc ontent=""></dialc>	PhoneNumber	必选,长度为1~24 位数字	呼叫号码集合最 大支持10个号码
		PhoneType	必选, <b>DeviceType</b>	设备类型
timeout	int	必选, 10~60的整数		用户不接通的超 时时间(10~60), 单位:秒

#### 返回值

#### IDevice.OnPollingCallResult

### 使用示例

```
//c# code
private void btnPollingCall_Click(object sender, EventArgs e)
    //初始化被呼叫对象列表
   List<DialContent> diaContents=new List<DialContent>();
    //被呼叫对象
   DialContent diaContent1=new DiaContent();
   dialContent1.PhoneNumber = "70777";
    //呼叫设备类型
   dialContent1.PhoneType = DeviceType.Phone;
    //呼叫类型
   dialContent1.CallType = DeviceCallType.ConventionCall;
   DialContent diaContent2=new DiaContent();
   dialContent2.PhoneNumber = "70778";
   dialContent2.PhoneType = DeviceType.Phone;
   dialContent2.CallType = DeviceCallType.ConventionCall
   dialContents.Add(dialContent1);
   dialContents.Add(dialContent2);
    //呼叫超时时间
   int timeout = 60;
   device.PollingCall(dialContents, timeout);
```

## 4.2.12 DisconnectCall(根据 CallId 释放呼叫)

## 接口描述

座席根据CallId释放呼叫。座席外呼其它电话或电话呼入形成通话时都会生成呼叫的唯一标识(CallId),座席侧可根据这个呼叫标识释放这通电话。

## 注意事项

座席根据CallId只能释放本座席的呼叫。

4 IDevice(设备服务基本接口)

<u>(S)</u>

#### 方法定义

```
void DisconnectCall(
string CallId
)
```

### 参数描述

参数	类型	参数规则	描述
CallId	string	必选,长度为1~24位数字	呼叫标识

### 返回值

#### IDevice.OnDisconnectCallResult

## 使用示例

```
//c# code
public int DisconnectCall(CallID pCallID)

{
    //查询当前活动的呼叫
    string agentId = this._agentInfo.AgentID
    CallID callID = BusinessAdapter.GetBusinessInstance().QueryActiveCallId(agentId);

    //释放呼叫
    var result = AgentGatewayHelper.DropCallByCallID(callID.PCallID);
}
```

## 4.2.13 TransferToSkillGroup(通话转队列)

## 接口描述

在电话通话过程中(非协商、被协商以及会议中),将呼叫转移给指定技能队列。该转移分为释放转和成功转。

释放转:即座席在转出技能队列后即将原呼叫释放,不论转出是否成功,呼叫都将被转移到转出技能队列。

成功转:即座席在转出后,待转出技能队列成功后才将原呼叫释放。如果呼叫转给技能队列,队列有空闲座席,则通话直接分配给空闲座席;如果队列下没有空闲座席,则直接转到技能队列进行排队等待。

## 方法定义

```
void TransferToSkillGroup(
bool bSuccTrans,
string groupId
)
```

### 参数描述

参数	类型	参数规则	描述
bSuccTrans	bool	必选	转移类型 <ul><li>true: 成功转</li><li>false: 释放转</li></ul>
groupId	string	必选,长度小于25位数字	技能队列ID

#### 返回值

#### IDeviceEvents.OnTransferToSkillGroupResult

### 使用示例

```
//c# code
private void btnTransferToSkillGroup_Click(object sender, EventArgs e)
{
    //转移方式, true为成功转, false为释放转
    bool IsSuccTransfer = true;
    //技能队列ID, 可在Was平台上查看,也可调用查询技能队列信息接口获取
    string skillId = "4";
    //通话转队列
    device.TransferToSkillGroup(IsSuccTransfer, skillID);
}
```

## 4.2.14 DialEx(单呼拓展)

### 接口描述

单呼拓展包括对普通电话、无线数字设备、无线模拟设备、短波、手机发起常规呼叫、点呼、组呼(注:在联动场景下,当座席话机的IPT呼叫保持后,座席可以发起外呼请求。)。

目前该接口支持以下场景:

- 1. 座席常规呼出:指座席向局外用户(固话、手机等)、内部话机(非座席)发起呼叫。
- 2. 座席内呼座席:指座席向内部其它座席(绑定的分机号)发起呼叫。
- 3. 座席呼叫eLTE: 指座席向LTE手台/集群群组发起呼叫。座席呼叫eLTE动态群组,动态群组中需要添加网关代理用户。
- 4. 座席呼叫Tetra: 指座席向车载Tetra手台/集群群组发起呼叫。

## 方法定义

```
void DialEx(
    DialContent dialContent,
    string accessCode
)
```

#### 参数描述

参数	类型	参数规则	描述	
dialContent	DialCon	PhoneNumber	必选,长度为1~24位数字	电话号码
	tent	PhoneType	必选, <b>DeviceType</b>	设备类型
		CallType	必选,DeviceCallType	设备呼叫类型
accessCode	string	必选,长度为1~24位字符串		系统接入码

#### ∭说明

DialContent类继承于Agent类。

#### 返回值

#### IDeviceEvents.OnDialExResult

### 使用说明

单呼时,针对被叫对象,主叫侧座席会收到系统主动触发的呼出状态返回事件: IDeviceEvents.OnCallOutStatusChange。

针对不同的呼叫场景,对应接口传入内容参考Dial(单呼)的使用说明。

### 使用示例

```
//c# code
private void btnDialEx_Click(object sender, EventArgs e)
{
    DialContent dialContent = new DialContent();
    //呼叫号码
    dialContent. PhoneNumber = this. textPhoneNumber. Text. Trim();
    //呼叫设备类型
    dialContent. PhoneType = DeviceType. Phone;
    //呼叫类型
    dialContent. CallType = DeviceCallType. ConventionCall;
    //系统接入码,可以再WAS平台中查看
    string accessCode = "60013"
    //单呼扩展
    device. DialEx(dialContent, accessCode));
}
```

## 4.2.15 PollingCallEx(轮询呼叫拓展)

#### 接口描述

对多个单呼对象轮询发起呼叫。

应用场景: 当座席号码需要依次单呼多个对象号码时,可调轮询呼叫接口。呼叫次序按照对象列表排序。如果呼叫上一个对象失败,则会自动呼叫列表后面的对象。如果呼叫对象成功,则不会再呼叫后面的对象。

### 注意事项

轮询号码呼出时,对每个号码的单呼,主叫侧座席会收到系统主动触发的呼出状态返回事件: IDeviceEvents.OnCallOutStatusChange。

## 方法定义

```
void PollingCallEx(
   List<DialContent> dialContents,
   int timeout,
   string accessCode
)
```

### 参数描述

参数	类型	参数规则		描述
dialContents	List <dialc ontent=""></dialc>	PhoneNumber	必选,长度为1~24 位数字	呼叫号码集合最大 支持10个号码
		PhoneType	必选, <b>DeviceType</b>	设备类型
timeout	int	必选, 10-60的整数		用户不接通的超时 时间(10~60),单 位:秒
accessCode	string	必选,长度为1~24位数字		系统接入码

## 返回值

#### IDevice.OnPollingCallExResult

## 使用说明

```
//c# code
private void btnPollingCallEx_Click(object sender, EventArgs e)
    //初始化被呼叫对象列表
   List \(\text{DialContent}\) diaContents=new List \(\text{DialContent}\)();
    //被呼叫对象
   DialContent diaContent1=new DiaContent();
   dialContent1.PhoneNumber = "70777";
    //呼叫设备类型
   dialContent1. PhoneType = DeviceType. Phone;
    //呼叫类型
   dialContent1.CallType = DeviceCallType.ConventionCall;
   DialContent diaContent2=new DiaContent();
   dialContent2.PhoneNumber = "70778";
   dialContent2.PhoneType = DeviceType.Phone;
   dialContent2.CallType = DeviceCallType.ConventionCall
   dialContents.Add(dialContent1);
   dialContents.Add(dialContent2);
   //呼叫超时时间
    int timeout = 60;
    //系统接入码,可以再WAS平台中查看
   string accessCode = "60013"
   //单呼扩展
```

4 IDevice (设备服务基本接口)

CS'

device.PollingCallEx(dialContents, timeout, accessCode));

## 4.2.16 Mute(静音)

### 接口描述

通话静音/取消静音。在座席处理报警人电话时,如果需要和内部人员进行讨论,但又 不希望报警人听到,可调用静音接口实现。

## 注意事项

会议或三方通话中不能调用静音接口。

## 方法定义

```
void Mute(
bool isMute
```

### 参数描述

参数	类型	参数规则	描述
isMute	bool	必选	是否静音
			● true: 静音
			● false: 取消静音

#### 返回值

#### IDeviceEvents.OnMuteResult

### 使用示例

```
//c# code
//设置静音
device.Mute(true);
//取消静音
device.Mute(false);
```

## 4.2.17 AgentSendDTMF(二次拨号)

## 接口描述

二次拨号,该接口允许座席在外呼IVR/总机后输入进一步信息。

在座席呼叫IVR流程后,座席和IVR流程可能需要进行多次交互,例如在IVR流程中依次需要输入另外两个拨号信息"123"和"456",则需要调用该接口来输入,即二次拨号的电话号码依次为"123"和"456"。

#### 注意事项

这个接口多次调用时,前后间隔时间至少需要1秒。

## 方法定义

```
void AgentSendDTMF(
string DTMFDigits
)
```

## 参数描述

参数	类型	参数规则	描述
DTMFDigits	string	必选,最大长度为24 位字符串	二次拨号的号码

## 返回值

#### IDevice Events. On Agent Send DTMFR esult

## 使用示例

```
//c# code
private void btnAgentSendDTMF_Click(object sender, EventArgs e)
{
    //二次拨号信息
    List<string> phoneList=new List<string>();
    phoneList. Add("123");
    phoneList. Add("456");
    foreach(string number in phoneList)
    {
        //二次拨号
        device. AgentSendDTMF(number));
        //间隔1s
        Thread. Sleep(1000);
    }
}
```

# 4.2.18 TransferToIVR(转到 IVR)

#### 接口描述

将语音通话转移到IVR流程。通话转流程有两种转移方式:释放转和挂起转。

- 释放转:通话转IVR流程后,座席通话被释放。
- 挂起转:通话转IVR流程后,座席通话被挂起,当IVR流程执行结束后呼叫会自动 回到座席。

# 注意事项

● 挂起转IVR流程结束后,座席侧会触发OnReturnFromIvr事件。

```
void TransferToIVR(
bool bReturnTrans,
string AccessCode
)
```

参数	类型	参数规则	描述
bReleaseTrans	bool	必选	转移方式
			● true: 挂起转
			● false: 释放转
AccessCode	string	必选,长度小于的25 位数字	IVR流程接入码

#### 返回值

#### IDeviceEvents.OnTransferToIVRResult

#### 使用示例

```
//c# code
private void btnTransToIVR_Click(object sender, EventArgs e)
{
    //挂起转
    bool IsSuccTransfer = true;
    //系统接入码,WAS平台可以查询
    string accessCode = "60015";
    //呼叫转IVR
    var result = device.TransferToIVR(IsSuccTransfer, accessCode));
}
```

# 4.2.19 HoldByCallId(根据 CallId 保持通话)

#### 接口描述

根据CallId保持或者取消保持指定通话。

通话中的座席需紧急外呼其他电话时,需要保持当前通话,然后再单呼其它紧急的电话;座席接听来电后,如果无法解决来电问题时需要先保持当前通话,然后协商呼叫其它座席帮忙解决。外呼或协商呼叫结束后,座席可取消保持后恢复与被保持用户的通话。

# 注意事项

- 座席已经在通话状态。
- 非联动场景只能保持一路呼叫,联动场景可以通过外置话机保持多路呼叫。
- 保持的通话无法被挂断,需先调用该接口取消保持。

```
void HoldByCallId(
HoldMode holdMode,
string CallId
)
```

参数	类型	参数规则	描述
holdMode	HoldMode	必选	保持模式
CallId	string	必选	呼叫标识

### 返回值

#### IDeviceEvents.OnHoldByCallIdResult

## 使用示例

```
//c# code
//查询座席当前活动的呼叫
string agentId = this._agentInfo.AgentID
CallID callID = BusinessAdapter.GetBusinessInstance().QueryActiveCallId(agentId);
//保持通话
device.HoldByCallId(HoldMode.Hold, callID.PCallID);
//取消保持的通话
device.HoldByCallId(HoldMode.HoldCancel, callID.PCallID);
```

# 4.3 IDevice Methods: 状态管理类功能

# 4.3.1 SetBusy(置忙)

#### 接口描述

座席置忙,即进入示忙状态。

如果当前接/处警员有很多事务需要处理,不希望被新的报警电话所干扰,可以选择事后置忙功能。这样系统不会把报警电话分配到当前座席,接/处警员可以专心处理尚未 处理完的事务,而不会被报警电话所打断。

## 注意事项

座席置忙后,不可被分配来电,但能外呼电话。

# 方法定义

void SetBusy()

# 参数描述

无。

# 返回值

IDeviceEvents.OnSetBusyResult

#### 使用示例

//c# code //座席置忙 device.SetBusy();

# 4.3.2 SetIdle(置闲)

#### 接口描述

座席置闲,即座席进入示闲状态。

一般接警员、处警员处理完报警人电话后会进入示忙状态,可以选择置闲功能,使座席进入示闲状态,以便继续接听报警电话。

## 注意事项

座席置闲后,可被系统分配电话。

## 方法定义

void SetIdle()

## 参数描述

无。

## 返回值

IDeviceEvents.OnSetIdleResult

#### 使用示例

//c# code //座席示闲 device.SetIdle();

# 4.3.3 QueryAgentStatus(查询座席状态)

# 接口描述

查询某座席的电话线路状态。

当前座席外呼其它座席的电话号码时,如果一直呼叫失败,可以通过该接口查询一下被呼叫座席的状态是否为示忙状态。如果是,则需等待被呼叫座席置闲后,才能呼叫成功。

# 方法定义

void QueryAgentStatus( AgentInfo agentInfo

参数	类型	参数规则		描述
agentInfo	AgentInfo	AgentId	必选,长度1~5的数 字,范围1-59999	座席工号

## 返回值

#### IDeviceEvents.OnQueryAgentStatusResult

座席状态发生变化时,系统主动返回IDeviceEvents.OnAgentStatusChange事件。

## 使用示例

```
//c# code

//初始化座席信息对象

AgentInfo agentInfo = new AgentInfo();

agentInfo.AgentId = this.txtAgentId.Text.Trim();

//查询座席状态

device.QueryAgentStatus(agentInfo);
```

# 4.3.4 QueryAllAgentStatus(查询所有座席状态)

# 接口描述

查询当前VDN中所有座席的状态,可以通过查询的状态数据判断哪些座席在上岗,哪些座席在接电话,哪些座席在休息等;可以通过该数据做实时的数据统计,并反映当前VDN中座席的相关状态统计信息。

# 注意事项

无。

# 方法定义

void QueryAllAgentStatus()

# 参数描述

无。

# 返回值

#### IDevice Events. On Query All Agent Status Result

# 使用示例

```
//c# code
//查询所有座席状态
var result = device.QueryAllAgentStatus();
```

# 4.3.5 Rest(休息)

# 接口描述

座席申请进行休息操作。该接口调用成功只表明申请休息成功,并不代表座席已经进入休息态。只有当CTI上抛休息状态改变的时候才是真正进入休息态。

接/处警员短时间离开座席可以选择休息,座席休息后将不会再被分配来电。当接/处警员已回到座席后,可以取消休息继续工作。

# 注意事项

- 座席休息原因由业务平台自己定义,可通过QueryRestReasons接口获得。
- 调用接口前,座席需要置闲。
- 座席休息后,将不再被分配来电。

## 方法定义

```
void Rest(
   int time,
   int usCause
)
```

## 参数描述

参数	类型	参数规则	描述
time	int	必选,最大休息时长为3600秒	请求休息时长,单位为秒
usCause	int	必选,正整数	休息原因,由业务方平台自 己定义

# 返回值

#### IDeviceEvents.OnRestResult

# 使用示例

```
//c# code
private void BtnRest_Click(object sender, EventArgs e)
{
    //获取休息原因
    List<RestReason> restReasons = null;
    restReasons = BusinessAdapter.GetBusinessInstance().QueryRestReasons();
    //设置休息时间和休息原因
    int time = 300;
    int restCode = 1;
    if(restReasons!=null && restReasons.Count>0)
    {
        restCode = restReasons[0].RestCode;
    }
    //申请休息
    device.Rest(300, restCode));
}
```

# 4.3.6 CancelRest(取消休息)

## 接口描述

座席申请取消休息。当座席取消休息成功后会退出休息态,这个时候座席可以通过置 忙或置闲操作来改变自己的状态,从而控制是否可以被分配来电。

# 方法定义

void CancelRest()

## 参数描述

无

## 返回值

#### IDeviceEvents.OnCancelRestResult

### 使用示例

//c# code //取消休息态 device.CancelRest();

# 4.3.7 SetAutoEnterIdle(设置通话后是否进入示闲状态)

# 接口描述

调用该接口可以控制座席结束通话后的状态。如果设置为true,则表明通话结束后座席进入示闲状态:如果设置为false,则表明通话结束后座席进入示忙状态。

# 注意事项

- 需要在WAS上保持以下信息配置为默认状态。 即: "VDN配置"->"基本信息配置"->"在示忙状态下呼出或放音的处理方式"->"退出示忙"。
- 如果设置false,并且设置了启动整理态,则座席通话结束后进入的是整理态。

# 方法定义

void SetAutoEnterIdle(
 bool bIdle

# 参数描述

参数	类型	参数规则	描述
bIdle	bool	必选	通话后是否进入示闲状态
			● true: 进入示闲状态
			● false: 进入示忙状态

4 IDevice(设备服务基本接口)

# 返回值

#### IDeviceEvents.OnSetAutoEnterIdleResult

### 使用示例

```
//c# code
//座席通话结束后示闲
device.SetAutoEnterIdle(true);
//座席通话结束后示忙
device.SetAutoEnterIdle(false);
```

# 4.3.8 SetWorkStop(设置是否进入开关班)

## 接口描述

设置座席开关班。

- 当进入关班状态的时候,座席的所有技能都会被签出,但此时座席仍然可以接受 其他座席的协商呼叫和会议邀请。
- 当进入开班状态的时候,座席的所有技能都会被签入。

# 方法定义

```
void SetWorkStop(
bool bWork
)
```

# 参数描述

参数	类型	参数规则	描述
bWork	bool	必选	是否进入关班状态
			● true: 进入关班状态
			● false: 进入开班状态

#### 返回值

#### IDevice Events. On Set Work Stop Result

## 使用示例

```
//c# code
//进入开班状态
device. SetWorkStop(false);
//进入关班状态
device. SetWorkStop(true);
```

# 4.3.9 ExitACW(退出整理态)

# 接口描述

座席退出整理态。

CS)

如果设置座席启用整理态,则座席接听完通话后就会进入整理态。这时置忙和置闲都

4 IDevice(设备服务基本接口)

# 注意事项

- 已经设置启用整理态
- 未设置通话后进入示闲状态

不会成功, 需要先退出整理态后才能做相关操作。

## 方法定义

void ExitACW()

# 参数描述

无。

#### 返回值

IDeviceEvents.OnExitACWResult

### 使用示例

//c# code //退出整理态 device.ExitACW();

# 4.3.10 QueryRestReasons(查询 VDN 休息原因)

#### 接口描述

该接口用来查询当前VDN中配置的可选择的休息原因,在座席进行休息申请的时候需要填写休息原因,该接口为其提供数据。

## 方法定义

void QueryRestReasons()

# 参数描述

无。

#### 返回值

#### IDevice Events. On Query Rest Reasons Result

# 使用示例

```
//c# code
//获取休息原因
List<RestReason> restReasons = null;
restReasons = BusinessAdapter.GetBusinessInstance().QueryRestReasons();
```

#### CS)

# 4.3.11 QuerySkillGroupStatus(查询呼叫队列状态)

## 接口描述

查询呼叫队列状态。通过该接口可以获得某技能队列的呼叫信息(当前可用座席数、 当前签入到这个技能队列的座席数、当前等待在队列中的呼叫数目等)。

# 方法定义

```
void QuerySkillGroupStatus(
    string groupId
)
```

## 参数描述

参数	类型	参数规则	描述
groupId	string	必选,范围1~2000	技能队列ID

# 返回值

#### IDevice Events. On Query Skill Group Status Result

# 使用示例

```
//c# code
//参数groupId为技能队列ID
device.QuerySkillGroupStatus(groupId));
```

# 4.3.12 SetConfig(设置配置项)

## 接口描述

设置振铃音相关的配置项,具体配置项功能和生效机制参考配置项说明。

在SDK加载后调用该接口来设置话机振铃效果,通过CONFIG\_ALERTING\_ENABLE设置是否开启振铃音,通过CONFIG\_ALERTING\_FILE\_PATH设置振铃音文件地址。

# 注意事项

- Voice.ocx加载并初始化成功。
- 电话号码未注册并且处于待机状态。

```
string SetConfig(
strng cfgKey,
string cfgValue
)
```

参数	类型	参数规则	描述
cfgKey	string	必选	配置项key
cfgValue	string	必选	配置项值。

可设置的配置项如下:

配置项Key	配置项值	描述
CONFIG_ALERTING_FI LE_PATH	振铃音文件所在路 径	绝对路径,文件必须为64K的纯 PCM的wav文件。如果为空,则使 用默认振铃音;如果该绝对路径文 件不存在或不合法,则振铃时不放 音或放杂音。
CONFIG_ALERTING_E NABLE	振铃音播放开关  ● 0: disable  ● 1: enable	默认为1: 开启振铃音

## 返回值

#### IDeviceEvents.OnSetConfigResult

# 使用示例

```
//c# code

//配置项key

string cfgKey = "CONFIG_ALERTING_ENABLE";

//配置项值

string cfgValue = "0"

//关闭振铃音

device. SetConfig(cfgKey, cfgValue);
```

# 4.3.13 GetConfig(查询配置项)

# 接口描述

查询振铃音相关的配置项,具体参考配置项说明。

# 注意事项

● Voice.ocx加载并初始化成功。

```
string GetConfig(
    strng cfgKey
)
```

参数	类型	参数规则	描述
cfgKey	string	必选	配置项key

可查询的配置项如下:

配置项Key	配置项值	描述
CONFIG_ALERTING_FI LE_PATH	振铃音文件所在路 径	绝对路径。如果为空,则使用默认 振铃音;如果该绝对路径文件不存 在或不合法,则振铃时不放音或放 杂音
CONFIG_ALERTING_E NABLE	振铃音播放开关  ● 0: disable  ● 1: enable	默认为1: 开启振铃音

## 返回值

#### IDeviceEvents.OnGetConfigResult

# 使用示例

```
//c# code
//配置项key
string cfgKey = "CONFIG_ALERTING_ENABLE";
//关闭振铃音
device.GetConfig(cfgKey);
```

# 4.4 IDevice Methods: 其他查询类功能

# 4.4.1 QueryIVRInfo(查询 IVR)

## 接口描述

查询系统中配置的所有自动流程的描述信息。

在调用**TransferToIVR**接口实现呼叫转IVR流程的场景中,我们可以先调用该接口并通过返回事件**OnQueryIVRInfoResult**获得系统中所有IVR流程的描述信息,然后通过信息可明确呼叫转到哪个IVR流程。

# 方法定义

void QueryIVRInfo()

# 参数描述

无。

4 IDevice(设备服务基本接口)

# 返回值

#### IDeviceEvents.OnQueryIVRInfoResult

#### 使用示例

//c# code
device.QueryIVRInfo();

# 4.4.2 QueryCallListByGroupId(查询某组呼叫队列)

#### 接口描述

查询某组呼叫队列,通过该接口可获取某技能队列中排队的电话号码的信息,可实时了解队列的排队情况。

## 注意事项

同组内来电队列电话状态发生变化时,系统会主动返回 IDeviceEvents.OnCallListChange事件。

# 方法定义

void QueryCallListByGroupId(
 string groupId

## 参数描述

参数	类型	参数规则	描述
groupId	string	必选,长度小于25位的数字	技能队列ID

## 返回值

#### IDevice Events. On Query Call List By Group Id Result

### 使用示例

//c# code //参数groupId为技能队列ID device.QueryCallListByGroupId(groupId);

# 4.4.3 StopQueryCallListByGroupId(停止查询某组呼叫队列)

## 接口描述

停止查询呼叫队列。

当查询某组呼叫队列的接口QueryCallListByGroupId被调用后,每当队列发生变化时座席侧都会收到IDeviceEvents.OnCallListChange事件,调用该接口可停止排队队列变化事件的上抛。

eSDK ICP V200R001C10 接口参考 01(ECC, Windows,

## 注意事项

- 前提是查询某组呼叫队列接口QueryCallListByGroupId已经调用成功。
- 当输入参数为空时,会停止查询所有的技能队列。

# 方法定义

```
void StopQueryCallListByGroupId(
    string groupId
```

## 参数描述

参数	类型	参数规则	描述
groupId	string	必选,长度小于25位的数字	技能队列ID

## 返回值

#### IDevice Events. On Stop Query Call List By Group Id Result

## 使用示例

```
//c# code
//参数groupId为技能队列ID
device.StopQueryCallListByGroupId(groupId);
```

# 4.4.4 QueryBlackList(查询是否是黑名单)

# 接口描述

查询某号码是否是黑名单。

座席可调用AddToBlackList接口添加黑名单号码,当某号码被添加到黑名单后,该号码通过系统接入码呼入时会失败。因此,当电话呼叫系统接入码一直失败时,可查询该号码是否已被加入黑名单。

# 注意事项

查询结果从IDeviceEvents.OnQueryBlackListResult事件获取。

# 方法定义

```
void QueryBlackList(
    string phoneNumber
)
```

# 参数描述

参数	类型	参数规则	描述
phoneNumber	string	必选,长度为1~24位数字	电话号码

4 IDevice (设备服务基本接口)

CS

## 返回值

#### IDeviceEvents.OnQueryBlackListResult

#### 使用示例

//c# code //查询号码是否是黑名单 device.QueryBlackList(PhoneNumber);

# 4.4.5 QueryBlackListEx(查询是否是黑名单扩展)

#### 接口描述

查询某号码是否是黑名单。

座席可调用AddToBlackList接口添加黑名单号码,当某号码被添加到黑名单后,该号码通过系统接入码呼入时会失败。因此,当电话呼叫系统接入码一直失败时,可查询该号码是否已被加入黑名单。

# 注意事项

- 查询结果从IDeviceEvents.OnQueryBlackListExResult事件获取。
- 黑名单与黑名单扩展的区别在于返回结果参数多了一个filterMode字段。

# 方法定义

void QueryBlackListEx( string phoneNumber

# 参数描述

参数	类型	参数规则	描述
phoneNumber	string	必选,长度为1~24位数字	电话号码

# 返回值

#### IDevice Events. On Query Black List ExResult

## 使用示例

//c# code //查询号码是否是黑名单 device.QueryBlackListEx(PhoneNumber);

# 4.4.6 QueryAllBlackList(查询所有黑名单)

## 接口描述

查询所有黑名单。黑名单为存储在EDS服务器数据库里的一张电话号码表单,该表单中的号码呼叫系统接入码时,呼叫会被挂断,即该号码用户不能报警。

CS

## 注意事项

查询结果从IDeviceEvents.OnQueryAllBlackListResult事件获取

4 IDevice(设备服务基本接口)

## 方法定义

void QueryAllBlackList()

## 参数描述

无

#### 返回值

IDevice Events. On Query All Black List Result

#### 使用示例

//c# code //查询所有黑名单 device.QueryAllBlackList();

# 4.4.7 QueryAllBlackListEx(查询所有黑名单扩展)

# 接口描述

查询所有黑名单清单扩展。黑名单为存储在EDS服务器上的一张电话号码表单,该表单中的号码呼叫系统接入码时,呼叫会被挂断,即该号码用户不能报警。

#### 注意事项

查询结果从IDeviceEvents.OnQueryAllBlackListExResult事件获取

#### 方法定义

void QueryAllBlackListEx()

# 参数描述

无。

# 返回值

IDevice Events. On Query All Black List ExResult

#### 使用示例

//c# code //查询所有黑名单扩展 device. QueryAllBlackListEx();

4 IDevice(设备服务基本接口)

# 4.4.8 QueryRedList(查询是否是优先级名单)

## 接口描述

查询某电话号码优先级。

座席可调用AddToRedList接口添加优先级名单号码,当某号码被添加到优先级名单后,该号码通过系统接入码呼入时会被率先分配座席应答。

## 注意事项

查询结果从IDeviceEvents.OnQueryRedListResult事件获取

# 方法定义

```
void QueryRedList(
    string phonePrefix
)
```

## 参数描述

参数	类型	参数规则	描述	
phonePrefix	string	必选,长度为1~24位数字	被查座席的电话号码	

# 返回值

#### IDeviceEvents.OnQueryRedListResult

#### 使用示例

//c# code //查询某电话号码的优先级 device. QueryRedList (PhoneNumber);

# 4.4.9 QueryAllRedList(查询所有优先级名单)

# 接口描述

查询所有的优先级名单。

签入座席可以通过该接口查询EDS平台数据库中的所有具有优先级的电话号码,然后根据这个优先级的顺位进行外呼操作。

# 注意事项

查询结果从IDeviceEvents.OnQueryAllRedListResult事件获取

# 方法定义

void QueryAllRedList()

eSDK ICP V200R001C10 接口参考 01(ECC, Windows, CS) 4 IDevice (设备服务基本接口)

参数描述

无

## 返回值

#### IDevice Events. On Query All Red List Result

#### 使用示例

//c# code //查询所有优先级名单 device.QueryAllRedList();

# 4.4.10 QuerySkillGroups(查询技能队列信息)

# 接口描述

查询技能队列信息。该接口用于查询系统中配置的所有技能队列信息,主要包括技能 队列ID以及相关技能描述信息。

#### 注意事项

查询结果从IDeviceEvents.OnQuerySkillGroupsResult事件获取

#### 方法定义

void QuerySkillGroups()

#### 参数描述

无

#### 返回值

IDeviceEvents.OnQuerySkillGroupsResult

#### 使用示例

//c# code //查询技能队列信息 device.QuerySkillGroups();

# 4.4.11 QueryAccessCodeList(查询所有系统接入码配置信息)

#### 接口描述

查询所有系统接入码配置信息。

外置电话通过呼叫系统接入码,然后系统会分配空闲的座席应答外置电话的呼叫。接 入码主要包括呼叫接入码、文字交谈接入码等。

## 注意事项

查询结果从IDeviceEvents.OnQueryAccessCodeListResult事件获取

4 IDevice(设备服务基本接口)

# 方法定义

void QueryAccessCodeList()

# 参数描述

无

# 返回值

IDevice Events. On Query Access Code List Result

#### 使用示例

//c# code //查询系统接入码 device.QueryAccessCodeList();

# 4.4.12 QueryLostCalls(查询丢话记录)

# 接口描述

查询所有丢话记录。座席可以查询某一时间段内通话丢失记录,通过查询到的主被叫号码信息,可以了解座席自身所绑定电话号码的丢话记录。

## 注意事项

查询结果从IDeviceEvents.OnQueryLostCallsResult事件获取

# 方法定义

```
void QueryLostCalls(
   int startPosition,
   int endPosition,
   DateTime beginTime,
   DateTime endTime
```

#### 参数描述

参数	类型	参数规则	描述
startPosition	int	必选,正整数	开始位置。
endPosition	int	必选,正整数	结束位置,一次查询最多 500条。 <b>说明</b> endPosition-startPosition 需 大于0小于500
beginTime	DateTime	必选,有效时间,格式为: yyyy-MM-dd HH:mm:ss	查询某段时间内丢话记录 的开始时间。
endTime	DateTime	必选,有效时间,格式为: yyyy-MM-dd HH:mm:ss	查询某段时间内丢话记录 的结束时间。

## 返回值

#### IDevice Events. On Query Lost Calls Result

## 使用示例

```
//c# code
private void btnQueryLostCalls_Click(object sender, EventArgs e)
{
    //界面参数输入
    var beginTime = this.beginTimePicker.Value;
    var endTime = this.endTimePicker.Value;
    var startPosition= this.txtStartPosition.Text.Trim();
    var endPosition = this.txtEndPosition.Text.Trim();
    //需判断参数是否合理
    .....
    //查询丢话记录
    device.QueryLostCalls(startPosition, endPosition, beginTime, endTime);
}
```

# 4.4.13 QueryLostCallsCount(查询丢话记录数量)

### 接口描述

查询所有丢话记录数量。系统维护人员可以通过这个接口查询某一时间段内总的丢话记录数量,来判断系统是否存在问题。

# 注意事项

查询结果从IDeviceEvents.OnQueryLostCallsResult事件获取

# 方法定义

```
void QueryLostCallsCount(
   DateTime beginTime,
   DateTime endTime
)
```

# 参数描述

参数	类型	参数规则	描述
beginTime	DateTime	必选,有效时间,格式为: yyyy-MM-dd HH:mm:ss	查询某段时间内丢话记录数量的开始时间。
endTime	DateTime	必选,有效时间,格式为: yyyy-MM-dd HH:mm:ss	查询某段时间内丢话记录数量的结束时间。

# 返回值

#### IDevice Events. On Query Lost Calls Count Result

# 使用示例

```
//c# code
private void btnQueryCallsCount_Click(object sender, EventArgs e)
```

eSDK ICP V200R001C10 接口参考 01(ECC, Windows,

```
//设置查询时间参数
var beginTime = this.beginTimePicker.Value;
var endTime = this.endTimePicker.Value;
//参数校验
if (endTime < beginTime)
{
    MsgBox.Error("EndTime is small than BeginTime");
    return;
}
//查询时间段内总的丢话数量
device.QueryLostCallsCount(beginTime, endTime);</pre>
```

# 4.4.14 QueryCallInfo(查询当前通话信息)

### 接口描述

获取座席当前通话信息,主要包括呼叫标识、主被叫号码、通话时长等信息。座席可以通过呼叫标识执行挂断、保持当前通话等操作。

## 注意事项

- 如果当前通话为IPCC呼叫,则返回IPCC呼叫信息;如果为IPT呼叫,则返回IPT呼叫信息。
- IPCC保持呼叫不可查询, IPT保持呼叫可查询。

# 方法定义

void QueryCallInfo ()

# 参数描述

无。

#### 返回值

#### IDeviceEvents.OnQueryCallInfoResult

#### 使用示例

```
//c# code
//查询座席当前通话信息
device. QueryCallInfo();
```

# 4.4.15 QueryAgentSkillGroups(查询座席技能信息)

#### 接口描述

查询某座席的技能信息。

当座席执行短信、邮件等功能失败的时候,可以上WAS平台去查看自身所具有技能信息,也可通过该接口返回的技能信息来判断是否有相应的技能能力。签入座席可查询自身的技能信息,也可查询别的座席的技能信息。

CS)

# 注意事项

查询结果从IDeviceEvents.OnQueryAgentSkillGroupsResult事件获取

# 方法定义

```
void QueryAgentSkillGroups(
string agentId
)
```

4 IDevice(设备服务基本接口)

# 参数描述

参数	类型	参数规则	描述
agentId	string	必选,范围1~59999	座席工号

## 返回值

#### IDevice Events. On Query Agent Skill Groups Result

## 使用示例

```
//c# code
//签入座席查询指定座席的技能信息
//agentId为指定座席的id
device.QueryAgentSkillGroups(agentId));
```

# 4.4.16 QueryMaliciousCalls(查询骚扰记录)

# 接口描述

查询骚扰记录。骚扰记录所记载的是黑名单电话的呼叫记录,每当黑名单中的电话号 码呼叫一次系统接入码,虽然呼叫失败,但服务端都会记载一条这样的骚扰呼叫记录。

# 注意事项

查询结果从IDeviceEvents.OnQueryMaliciousCallsResult事件获取

# 方法定义

```
void QueryMaliciousCalls(
string PhoneNumber
)
```

# 参数描述

参数	类型	参数规则	描述
phoneNumber	string	必选,长度为1~24位数字	主叫电话号码。

#### <del>(()</del>

# 返回值

#### IDevice Events. On Query Malicious Calls Result

## 使用示例

```
//c# code
//查询某电话的骚扰记录
device. QueryMaliciousCalls(phoneNumber)
```

# 4.5 IDevice Methods: 电话会议类功能

# 4.5.1 CreateMeeting(创建会议)

# 接口描述

创建会议(多呼),座席电话为示闲状态时,座席对多个单呼对象(最多24个)进行同时呼叫。每个对象呼叫成功与否可通过会议成员在会议中的状态反映出来。

# 注意事项

- 创建会议的座席为会议主席,创建会议后也可将主席权转给其它座席。
- 多呼时,针对每个单呼对象(会议成员)系统向创建会议座席主动返回 IDeviceEvents.OnYieldAnswerCall事件。

# 方法定义

```
void CreateMeeting(
    List<DialContent> dialContents
)
```

# 参数描述

参数	类型	参数规则		描述
dialConte	List <dial< td=""><td>PhoneNumber</td><td>必选,长度为1~24位数字</td><td>呼叫号码集合</td></dial<>	PhoneNumber	必选,长度为1~24位数字	呼叫号码集合
nts Content>		PhoneType	必选, <b>DeviceType</b>	设备类型 参考 <b>使用说明</b>
		CallType	必选,DeviceCallType	设备呼叫类型 参考 <b>使用说明</b>

# 返回值

#### IDeviceEvents. OnCreateMeetingResult

# 使用示例

```
//c# code
private void btnCreateMeeting_Click(object sender, EventArgs e)
{
```

CS)

eSDK ICP V200R001C10 接口参考 01(ECC, Windows,

```
//初始化被呼叫对象列表
List Dial Content dia Contents = new List Dial Content ();
//被呼叫对象
Dial Content dia Content = "70777";
//呼叫设备类型
dial Content = Device Type. Phone;
//呼叫类型
dial Content = Device Call Type. Convention Call;

Dial Content dia Content = "70778";
dial Content = "70778";
dial Content = Device Type. Phone;
dial Content = Device Call Type. Convention Call

dial Content = Device Call Type. Convention Call

dial Content = Add (dial Content = Device Call Type. Convention Call

dial Content = Device Call Type. Convention Call
```

# 4.5.2 MakeMeeting(形成会议)

#### 接口描述

形成会议。场景如下:座席A将报警人的电话保持,并且协商呼叫座席B成功后。可调用该接口形成报警人、座席A、座席B三方的会议,这样能够更好的处理报警人问题。

#### 注意事项

发起形成会议成功后,发起座席为会议主席。

#### 方法定义

void MakeMeeting()

# 参数描述

无

#### 返回值

IDeviceEvents.OnMakeMeetingResult

## 使用示例

```
//c# code
device.MakeMeeting();
```

# 4.5.3 SingleStepMeeting(单步会议)

#### 接口描述

如果座席在通话过程中,可直接调用该接口呼叫第三方的普通电话或者座席电话形成单步会议(当第三方对象是无线类型时只支持呼叫eLTE号码)。

# 注意事项

发起单步会议的座席为会议主席。

# 方法定义

```
void SingleStepMeeting (
DialContent dialContent
)
```

#### 参数描述

参数	类型	参数规则	描述	
dialContent	DialConten t	PhoneNumber	必选,长度为1~24位 数字	电话号码
		PhoneType	必选, <b>DeviceType</b>	设备类型
				参考使用说明
		CallType	必选,DeviceCallType	设备呼叫类型
				参考使用说明

#### 返回值

#### IDeviceEvents.OnSingleStepMeetingResult

## 使用示例

```
//c# code
private void BtnSingleStepMeeting_Click(object sender, EventArgs e)
{
    //被单步会议对象
    DialContent diaContent=new DiaContent();
    dialContent. PhoneNumber = "70777";
    //呼叫设备类型
    dialContent. PhoneType = DeviceType. Phone;
    //呼叫类型
    dialContent. CallType = DeviceCallType. ConventionCall;
    device. SingleStepMeeting(dialContent);
}
```

# 4.5.4 InviteMeetingMember(邀请会议成员)

# 接口描述

邀请会议成员。创建会议、单步会议等操作成功后,会议成员可在返回的 IDeviceEvents.OnYieldAnswerCall事件中获取到会议ID,但只有会议主席可以根据会 议ID邀请其它成员加入到会议。

# 注意事项

多呼创建会议和协商形成会议都通过此接口来实现成员邀请,被邀请成员侧会收到系统主动返回的IDeviceEvents.OnInvitedMeeting事件。

```
void InviteMeetingMember(
string meetingId,
```

CS)

```
DialContent member
```

## 参数描述

参数	类型	参数规则		描述
meetingId	string	必选,数字		会议标识
member	DialConten t	PhoneNumber 必选,长度为1~24 位数字		会议成员电话号码
		PhoneType	必选,DeviceType	设备类型 参考 <b>使用说明</b>
		CallType	必选, DeviceCallType	设备呼叫类型 参考 <b>使用说明</b>

## 返回值

#### IDevice Events. On Invite Meeting Member Result

#### 使用示例

```
//c# code
private void btnAddMember_Click(object sender, EventArgs e)
{
    string MeetingId = "3308";
    //被邀请会议成员
    DialContent diaContent=new DiaContent();
    dialContent. PhoneNumber = "70777";
    //呼叫设备类型
    dialContent. PhoneType = DeviceType. Phone;
    //呼叫类型
    dialContent. CallType = DeviceCallType. ConventionCall;

//邀请会议成员
    device. InviteMeetingMember(MeetingId, diaContent);
}
```

# 4.5.5 RemoveMeetingMember(移除会议成员)

# 接口描述

移除会议成员。

创建会议、单步会议等操作成功后,会议成员可在返回的事件中获取到会议ID,但只有会议主席可以根据会议ID移除会议中的其它成员,例如,呼叫未成功的电话。

# 注意事项

- 会议主席不能被移除,只能退出会议。
- 被移除会议的成员侧会收到主动返回的IDeviceEvents.OnLeaveMeeting事件。

## 方法定义

```
void RemoveMeetingMember(
    string meetingId,
    DialContent dialContent
)
```

## 参数描述

参数	类型	参数规则		描述
meetingId	string	必选,数字		会议标识
dialContent	DialConten t	PhoneNumber	必选,长度为 1~24位数字	会议成员电话号码

## 返回值

#### IDevice Events. On Remove Meeting Member Result

#### 使用示例

```
//c# code
private void btnRemoveMember_Click(object sender, EventArgs e)
{
    string MeetingId = "3308";
    //被移除会议对象
    DialContent diaContent=new DiaContent();
    dialContent. PhoneNumber = "70777";
    //呼叫设备类型
    dialContent. PhoneType = DeviceType. Phone;
    //呼叫类型
    dialContent. CallType = DeviceCallType. ConventionCall;

//移除会议成员
    device. RemoveMeetingMember(MeetingId, diaContent);
}
```

# 4.5.6 ExitMeeting(退出会议)

# 接口描述

退出会议。创建会议、单步会议等操作成功后,每个会议成员可在返回的 IDeviceEvents.OnYieldAnswerCall事件中获取到会议ID,会议成员可以根据这个ID退 出会议(包括会议主席)。

# 注意事项

会议主席退出后会议将没有会议主席,因此对于会议中会议成员的相关操作就都不能执行了。

```
void ExitMeeting(
    string meetingId
)
```

4 IDevice(设备服务基本接口)

# 参数描述

参数	类型	参数规则	描述
meetingId	string	必选,数字	会议标识

## 返回值

#### IDeviceEvents.OnExitMeetingResult

## 使用示例

```
//c# code
string MettingId= "3308";
//退出会议
device.ExitMeeting(MettingId);
```

# 4.5.7 TransferMeetingMaster(转移会议主席)

# 接口描述

转移会议主席。

当会议主席需要退出会议时,一般需要将主席权限转给其它会议成员,否则后续对会 议成员的相关操作都将不能执行成功,导致会议不能正常的召开。

# 注意事项

新会议主席侧会收到主动返回的IDeviceEvents.OnBeTransferMaster事件。

# 方法定义

```
void TransferMeetingMaster(
    string meetingId,
    string oldMasterNumber,
    string newMasterNumber
)
```

## 参数描述

参数	类型	参数规则	描述
meetingId	string	必选,数字	会议标识
oldMasterNumber	string	必选,长度为1~24位数字	旧会议主席电话号码
newMasterNumber	string	必选,长度为1~24位数字	新会议主席电话号码

#### 返回值

#### IDevice Events. On Transfer Meeting Master Result

#### 使用示例

```
//c# code
private void btnTransferMeetingMaster_Click(object sender, EventArgs e)
{
    //会议ID
    string MettingId = "3308";
    //会议新旧主席号码
    string OldMasterNumber = this.txtOldMasterNumber.Text.Trim();
    string NewMasterNumber = this.txtNewMasterNumber.Text.Trim();
    //转移会议主席
    device.TransferMeetingMaster(MettingId, OldMasterNumber, NewMasterNumber);
}
```

# 4.5.8 EndMeeting(结束会议)

#### 接口描述

结束会议。创建会议、单步会议等操作成功后,会议成员可在返回的 IDeviceEvents.OnYieldAnswerCall事件中获取到会议ID,但只有会议主席可以根据会 议ID结束该会议。

# 方法定义

```
void EndMeeting(
    string meetingId
)
```

## 参数描述

参数	类型	参数规则	描述
meetingId	string	必选,数字	会议标识

#### 返回值

#### IDeviceEvents.OnEndMeetingResult

# 使用示例

```
//c# code
//会议主席根据会议ID结束会议
string MettingId = "3308"
device.ExitMeeting(MettingId);
```

# 4.5.9 QueryMeetingMembers(查询会议成员)

#### 接口描述

查询某个会议中所有会议成员。

创建会议、单步会议等操作成功后,会议成员可在返回的 IDeviceEvents.OnYieldAnswerCall事件中获取到会议ID,会议中的所有成员都可以根据会议ID查询当前会议中的成员。

# 方法定义

```
void QueryMeetingMembers(
string meetingId
)
```

## 参数描述

参数	类型	参数规则	描述
meetingId	string	必选	会议标识

## 返回值

#### IDevice Events. On Query Meeting Members Result

## 使用示例

```
//c# code
//会议标识返回的事件中获得
device.QueryMeetingMembers(mettingId);
```

# 4.5.10 CreateMeetingEx(创建会议拓展)

## 接口描述

创建会议(多呼),座席电话为示闲状态时,座席对多个单呼对象(最多24个)进行同时呼叫。每个对象呼叫成功与否可通过会议成员在会议中的状态反映出来。

# 注意事项

多呼时,针对每个被呼叫的会议成员,系统向创建会议的座席主动返回 IDeviceEvents.On Yield Answer Call 事件。

# 方法定义

```
void CreateMeetingEx(
   List<DialContent> dialContents,
   string accessCode
)
```

# 参数描述

参数	类型	参数规则	描述	
accessCode	string	必选,长度为1	系统接入码	
dialContents	List <dialco ntent&gt;</dialco 	PhoneNumber	呼叫号码集合	
		PhoneType	必选,DeviceType	设备类型 参考 <b>使用说明</b>

参数	类型	参数规则		描述
		CallType	必选,	设备呼叫类型
			DeviceCallType	参考 <b>使用说明</b>

### 返回值

CS)

#### IDeviceEvents. OnCreateMeetingExResult

## 使用示例

```
//c# code
private void btnCreateMeetingEx_Click(object sender, EventArgs e)
    //初始化被呼叫对象列表
    List<DialContent> diaContents=new List<DialContent>();
    //被呼叫对象
    {\tt DialContent\ diaContent1=new\ DiaContent();}
    dialContent1.PhoneNumber = "70777";
    //呼叫设备类型
    dialContent1.PhoneType = DeviceType.Phone;
    dialContent1.CallType = DeviceCallType.ConventionCall;
   DialContent diaContent2=new DiaContent();
dialContent2. PhoneNumber = "70778";
    dialContent2.PhoneType = DeviceType.Phone;
    dialContent2.CallType = DeviceCallType.ConventionCall
    dialContents.Add(dialContent1);
    dialContents.Add(dialContent2);
    //系统接入码可在平台配置中获取
    string accessCode = "60013"
    //创建会议拓展
    device.CreateMeetingEx(diaContents, accessCode))
```

# 4.5.11 ChangeMeetingMemberMode(修改与会者属性)

# 接口描述

会议主席修改会议成员属性。会议成员属性主要包括只能听、只能说、能听能说、不能听也不能说等。

# 注意事项

只有主席才能修改会议成员属性,当属性为1和3时,其他座席会议成员会自动返回 OnMeetingMemberModeChange事件。

```
void ChangeMeetingMemberMode(
   string meetingId,
   DialContent dialContent,
   int attribute
)
```

参数	类型	参数规则		描述
meetingId	string	必选,数字		会议标识
dialContent	DialCont ent	PhoneNumb er	必选,长度为 1~24位数字	会议成员电话号码
attribute	int	必选		与会者属性:  ● 1: 只能听,不能说  ● 2: 只能说,不能听  ● 3: 既能听,也能说  ● 4: 不能听,不能说

# 返回值

#### $IDevice Events.\ On Change Meeting Member Mode Result$

#### 使用示例

```
//c# code
private void btnChangeMemberMode_Click(object sender, EventArgs e)
{
    string MeetingID = "3308";
    //修改对象从会议成员中获取
    DialContent diaContent=new DiaContent();
    dialContent. PhoneNumber = "70777";
    //呼叫设备类型
    dialContent. PhoneType = DeviceType. Phone;
    //呼叫类型
    dialContent. CallType = DeviceCallType. ConventionCall;
    //会议成员能听能说
    int attribute = 3;

    device. ChangeMeetingMemberMode(MeetingID, dialContent, attribute);
}
```

# 4.6 IDevice Methods: 座席管理类功能

# 4.6.1 Barge(强插)

### 接口描述

强行插入指定座席的通话线路。质检座席进入占用态。

座席A为质检座席,座席B为普通接警员座席,当座席B与报警人通话的时候,质检座席可以调用该接口强行插入通话,插入成功后形成座席A、座席B、报警人的三方通话。

## 注意事项

● 调用该接口的座席必须具有质检员权限。

eSDK ICP V200R001C10 接口参考 01(ECC, Windows,

- 被强插座席必须是通话状态。
- 质检座席强插后,被强插座席侧会收到系统主动返回的IDeviceEvent.OnBeBarged 事件。

# 方法定义

```
void Barge(
   AgentInfo agentInfo
```

# 参数描述

参数	类型	参数规则	描述	
agentInfo	AgentInfo	AgentId	必选,范围: 1-59999	座席工号
		PhoneNumber	必选,长度为1~24位数 字	电话号码

#### 返回值

#### IDeviceEvents.OnBargeResult

## 使用示例

```
private void btnBarge Click(object sender, EventArgs e)
   AgentInfo agentInfo = new AgentInfo();
   //被强插座席工号
   agentInfo.AgentId = "20012";
   //被强插座席所绑定的电话号码
   agentInfo. PhoneNumber = "70777"
   device.Barge(agentInfo);
```

# 4.6.2 ForceRelease(拦截)

# 接口描述

拦截指定座席的通话线路。

座席A为质检座席,座席B为普通接警员座席,当座席B与报警人通话的时候,质检座 席可以调用该接口拦截此通话。拦截成功后座席B的通话被挂断,形成座席A与报警人 之间的通话。

# 注意事项

- 调用该接口的座席必须具有质检员权限。
- 被拦截座席必须是通话状态。
- 质检座席拦截后,被拦截座席会收到系统主动返回的 IDeviceEvent.OnBeForceReleased事件。

```
void ForceRelease(
AgentInfo agentInfo
)
```

参数	类型	参数规则		描述
agentInfo	AgentInfo	AgentId	必选,范围: 1-59999	座席工号
		PhoneNumber	必选,长度为1~24位数 字	电话号码

#### 返回值

#### IDeviceEvents.OnForceReleaseResult

#### 使用示例

```
//c# code
private void btnSnatch_Click(object sender, EventArgs e)
{
    AgentInfo agentInfo = new AgentInfo();
    //被拦截座席工号
    agentInfo. AgentId = "20012";
    //被拦截座席所绑定的电话号码
    agentInfo. PhoneNumber = "70777"
    //拦截
    device. ForceRelease(agentInfo);
}
```

# 4.6.3 Monitor(监听)

#### 接口描述

监听某一正在通话的座席的话务。

座席A为质检座席,座席B为普通接警员座席,当座席B与报警人通话的时候,质检座席可以调用该接口监听此通话,监听成功后座席A可以听到座席B与报警人的通话,而座席B和报警人听不到座席A的语音。

# 注意事项

- 调用该接口的座席必须具有质检员权限。
- 被监听座席必须是通话状态。
- 监听成功后,质检座席进入占用态。
- 质检座席监听后,被监听座席会收到系统主动返回的 IDeviceEvent.OnBeMonitored事件。

```
void Monitor(
AgentInfo agentInfo
)
```

参数	类型	参数规则	描述	
agentInfo	AgentInfo	AgentId	必选,范围: 1-59999	座席工号
		PhoneNumber	必选,长度为1~24位数 字	电话号码

## 返回值

#### IDeviceEvents.OnMonitorResult

## 使用示例

```
//c# code
private void btnMonitor_Click(object sender, EventArgs e)
{
    AgentInfo agentInfo = new AgentInfo();
    //被监听座席工号
    agentInfo. AgentId = "20012";
    //被强插座席所绑定的电话号码
    agentInfo. PhoneNumber = "70777"
    //监听
    device. Monitor(agentInfo);
}
```

# 4.6.4 MonitorToBarge(监听转强插)

## 接口描述

质检座席电话监听过程中, 强行插入通话线路。

质检座席A在监听普通座席B和报警人的通话过程中,如果需要座席B和报警人也能听到自己的声音时,例如下达指令、提出建议等可调用该接口。

# 使用说明

- 调用该接口的座席必须具有质检员权限。
- 被监听转强插座席必须在监听中。
- 监听转强插后,被强插座席侧会收到系统主动返回的**IDeviceEvent.OnBeBarged**事件。

# 方法定义

```
void MonitorToBarge(
    AgentInfo agentInfo
)
```

# 参数描述

参数	类型	参数规则	描述	
agentInfo	AgentInfo	AgentId	必选,范围: 1-59999	座席工号

参数	类型	参数规则		描述
		PhoneNumber	必选,长度为1~24位数 字	电话号码

# 返回值

#### IDevice Events. On Monitor To Barge Result

# 使用示例

```
//c# code
private void btnMonitorToBarge_Click(object sender, EventArgs e)
{
    AgentInfo agentInfo = new AgentInfo();
    //被强插座席工号
    agentInfo. AgentId = "20012";
    //被强插座席所绑定的电话号码
    agentInfo. PhoneNumber = "70777"
    //监听转强插
    device. MonitorToBarge(agentInfo);
}
```

# 4.6.5 MonitorToForceRelease(监听转拦截)

## 接口描述

质检座席电话监听过程中,强行拦截通话线路。

质检座席A在监听普通座席B和报警人的通话过程中,发现普通座席B的接警员在处理报警人来电时处理不当,可调用该接口将通话拦截过来。

# 注意事项

- 调用该接口的座席必须具有质检员权限。
- 被监听转拦截座席必须在监听中。
- 监听转拦截后,被拦截座席侧会收到系统主动返回的 IDeviceEvent.OnBeForceReleased事件。

# 方法定义

```
void MonitorToForceRelease(
   AgentInfo agentInfo
)
```

# 参数描述

参数	类型	参数规则	描述	
agentInfo	AgentInfo	AgentId	必选,范围: 1-59999	座席工号
		PhoneNumber	必选,长度为1~24位数 字	电话号码

#### \_\_\_\_

返回值

#### IDeviceEvents.OnMonitorToForceReleaseResult

#### 使用示例

```
//c# code
private void btnMonitorToSnatch_Click(object sender, EventArgs e)
{
    AgentInfo agentInfo = new AgentInfo();
    //被拦截座席工号
    agentInfo. AgentId = "20012";
    //被拦截座席所绑定的电话号码
    agentInfo. PhoneNumber = "70777"
    //监听转拦截
    device. MonitorToForceRelease(agentInfo);
}
```

# 4.6.6 Whisper(耳语)

#### 接口描述

质检座席向正在通话的座席耳语。

座席A为质检座席,座席B为普通接警员座席,当座席B与报警人通话的时候,质检座席可以调用该接口耳语此通话,耳语成功后座席A可以听到座席B与报警人的通话,座席B也能听到质检座席A的语音,但报警人听不到质检座席A的语音。

#### 注意事项

- 调用该接口的座席必须具有质检员权限。
- 被耳语座席必须在语音通话中。
- 质检座席耳语成功后,质检座席进入占用态。
- 质检座席耳语成功后,被耳语座席会收到系统主动返回的 IDeviceEvent.OnBeWhispered事件。

### 方法定义

```
void Whisper(
AgentInfo agentInfo
)
```

### 参数描述

参数	类型	参数规则		描述
agentInfo	AgentInfo	AgentId 必选,范围: 1-59999		座席工号
		PhoneNumber	必选,长度为1~24位数 字	电话号码

#### 返回值

#### IDeviceEvents.OnWhisperResult

#### 使用示例

```
//c# code
private void btnWhisper_Click(object sender, EventArgs e)
{
    AgentInfo agentInfo = new AgentInfo();
    //被耳语座席工号
    agentInfo. AgentId = "20012";
    //被耳语座席所绑定的电话号码
    agentInfo. PhoneNumber = "70777"
    //耳语
    device. Whisper(agentInfo);
}
```

# 4.6.7 SuperviseStop(停止质检)

#### 接口描述

质检座席和座席处于耳语、监听、强插状态中的某一种状态时,质检座席停止当前质检操作,停止质检成功后,质检座席退出质检,座席通话不受影响。

### 注意事项

- 调用该接口的座席必须具有质检员权限。
- 被质检座席必须在语音通话中,并且质检座席处于耳语、监听、强插状态其中的 某一种状态。
- 被质检座席会收到系统主动返回的IDeviceEvent.OnBeStopSupervise事件。

## 方法定义

void SuperviseStop()

# 参数描述

无

#### 返回值

IDeviceEvents.OnSuperviseStopResult

#### 使用示例

```
//c# code
//停止质检
device.SuperviseStop();
```

# 4.6.8 SuperviseSwitch(质检切换)

#### 接口描述

质检座席在监听、强插、耳语质检操作间进行切换。

- 质检座席监听状态,质检员能听到接警员与报警人的通话,而接警员和报警人听不到质检员的语音。
- 质检座席强插状态,质检员能听到接警员与报警人的通话,而接警员和报警人也 能听到质检员的语音。

● 质检座席耳语状态,质检员能听到接警员与报警人的通话,接警员也能听到质检员的语音,而报警人听不到质检员的语音。

#### 注意事项

- 调用该接口的座席必须具有质检员权限。
- 被质检座席必须在语音通话中,并且质检座席处于耳语、监听、强插状态其中的某一种状态。
- 如果切换到当前的质检状态,则直接返回错误。例如,当前为监听状态、再调用 该接口切到监听状态,则直接返回错误。
- 质检切换成功后,被质检座席会收到系统主动返回的对应事件 IDeviceEvent.OnBeMonitored、IDeviceEvent.OnBeBarged或者 IDeviceEvent.OnBeWhispered。

### 方法定义

```
void SuperviseSwitch(
   int operType
)
```

### 参数描述

参数	类型	参数规则	描述
operType	int	必选	质检操作类型
			● 0: 监听
			● 1: 强插
			● 3: 耳语

#### 返回值

#### IDeviceEvents.OnSuperviseSwitchResult

#### 使用示例

```
//c# code
//监听
device. SuperviseSwitch(0);
//强插
device. SuperviseSwitch(1);
//耳语
device. SuperviseSwitch(3);
```

# 4.6.9 AddToBlackList(设置黑名单)

### 接口描述

设置黑名单,即将某电话添加到EDS服务器数据库中的黑名单表中,被添加到该表的电话号码在锁定时长结束之前,呼叫系统接入码会失败,因此不会分配到座席。

4 IDevice(设备服务基本接口)

## 方法定义

```
void AddToBlackList(
    string phoneNumber,
    int lockTime
)
```

#### 参数描述

参数	类型	参数规则	描述
phoneNumber	string	必选,长度为1~24位数字	电话号码
lockTime	int	必选,范围: 5~999999999	锁定时长,单位: 秒

#### 返回值

#### IDeviceEvents.OnAddToBlackListResult

### 使用示例

```
//c# code
private void btnAddBlackList_Click(object sender, EventArgs e)
{
    //需添加的黑名单
    string PhoneNumber = "88888";
    //需锁定时长1小时
    int lockTime = 3000
    //设置黑名单
    device. AddToBlackList(PhoneNumber, lockTime));
}
```

# 4.6.10 AddToBlackListEx(设置黑名单扩展)

## 接口描述

设置黑名单扩展。即将某电话添加到EDS服务器数据库中的黑名单表中,被添加到该表的电话号码在锁定时长结束之前,呼叫系统接入码会失败,因此不会分配到座席。

# 方法定义

```
void AddToBlackListEx(
    string phoneNumber,
    int lockTime,
    FilterMode filterMode
)
```

### 参数描述

参数	类型	参数规则	描述
phoneNumber	string	必选,长度为1~24位数字	添加至黑名单的电话 号码
lockTime	int	必选,范围: 5~999999999	锁定时长,单位:秒

参数	类型	参数规则	描述
filterMode	int	必选	黑名单模式
			● 1: 拒绝
			● 2: 降级

#### IDeviceEvents.OnAddToBlackListExResult

### 使用示例

```
//c# code
private void btnAddToBlackListEx_Click(object sender, EventArgs e)

{
    //需添加的黑名单
    string PhoneNumber = "88888";
    //需锁定时长1小时
    int lockTime = 3000;
    int filterMode = 1;
    //设置黑名单
    device. AddToBlackListEx(PhoneNumber, lockTime, filterMode));
}
```

# 4.6.11 RemoveFromBlackList(取消黑名单)

## 接口描述

将指定号码从黑名单项中取消。即将某电话从EDS服务器数据库中的黑名单表中删除,删除后该电话号码可以通过系统接入码呼入,并分配空闲座席处理来电。

# 注意事项

● 如果phoneNumber传入null值,直接返回失败。

# 方法定义

```
void RemoveFromBlackList(
    string phoneNumber
)
```

## 参数描述

参数	类型	参数规则	描述
phoneNumber	string	必选,长度为1~24位数字	被取消黑名单的电话号码

### 返回值

#### IDeviceEvents.OnRemoveFromBlackListResult

#### 使用示例

```
//c# code
private void btnRemoveBlackList_Click(object sender, EventArgs e)

{
    //需取消的黑名单电话号码
    string phoneNumber = "88888";

    //取消黑名单
    device.RemoveFromBlackList(phoneNumber);
}
```

# 4.6.12 RemoveAllBlackList(移除全部黑名单)

#### 接口描述

移除黑名单中所有被屏蔽的电话号码。

#### 方法定义

void RemoveAllBlackList()

#### 参数描述

无。

#### 返回值

IDeviceEvents.OnRemoveAllBlackListResult

#### 使用示例

```
//c# code
device.RemoveAllBlackList();
```

# 4.6.13 SelectPickup(选择应答)

#### 接口描述

针对待接听的来电队列,选择应答。

座席A具有选择应答的能力,并且处于示忙状态,可通过QueryCallListByGroupId接口查询呼叫队列的信息,根据呼叫信息中的CallId信息可以调用该接口应答排队队列中的指定呼叫。

#### 注意事项

- 座席处于示忙状态。
- 座席具有选择应答的能力,可在WAS平台上查看。
- 待接听呼叫队列不为空。

## 方法定义

```
void SelectPickup(
DialContent dialContent
)
```

#### 参数描述

参数	类型	参数规则		描述
dialContent	DialConten t	CallId	必选,长度小于25位字符串	呼叫标识

#### 返回值

#### IDeviceEvents.OnSelectPickupResult

#### 使用示例

//c# code
DialContent dialContent = new DialContent();
//callId为通过QueryCallListByGroupId接口查询呼叫队列的信息
diaContent.CallId = callId;
//选择应答
device.SelectPickup(dialContent);

# 4.6.14 SnatchPickup(代接)

### 接口描述

代替接听某座席新来电。

座席A具有同组代答的权限,报警人通过呼叫系统接入码,系统分配座席B应答,并且座席B振铃状态中,这时座席A可调用代接接口将报警人的电话接过来处理。

## 注意事项

- 被代接的座席必须是振铃中状态。
- 代接座席具有同组代答权限。
- 如果新来电为内线电话,则不能被代接。
- 座席代接后,被代接座席侧会收到系统主动返回的 IDeviceEvent.OnBeSnatchedPickup事件。

# 方法定义

void SnatchPickup(AgentInfo agentInfo)

## 参数描述

参数	类型	参数规则		描述
agentInfo	AgentInfo	AgentId 必选,范围: 1-59999		座席工号
		PhoneNumber	必选,长度为1~24位数 字	电话号码

#### IDeviceEvents.OnSnatchPickupResult

#### 使用示例

```
//c# code
private void btnSnatchPickup_Click(object sender, EventArgs e)

{
    AgentInfo agentInfo = new AgentInfo();
    //被代接座席工号
    agentInfo. AgentId = "20012";
    //被代接座席所绑定的电话号码
    agentInfo. PhoneNumber = "70777"
    //代接
    device. SnatchPickup(agentInfo)
}
```

# 4.6.15 ForceLogout(强制座席签出)

### 接口描述

强制座席签出。质检座席可以强制签出某指定的座席,被强制签出的座席如需恢复业务工作,需重新调用Initialize接口签入。如果强制签出失败,则可能被强制签出座席没有处于示闲状态。

### 注意事项

- 被强制签出座席必须处于示闲状态,否则签出失败。
- 被强制签出的座席侧会收到系统主动触发的事件: IDeviceEvents.OnBeForceLogout。

# 方法定义

```
void ForceLogout(
AgentInfo agentInfo
)
```

# 参数描述

参数	类型	参数规则		描述
agentInfo	AgentInfo	AgentId	必选,范围: 1-59999	座席工号

# 返回值

#### IDevice Events. On Force Logout Result

# 使用示例

```
//c# code
private void btnForceLogout_Click(object sender, EventArgs e)
{
    AgentInfo agentInfo = new AgentInfo();
    //被强制退出座席工号
    agentInfo.AgentId = "20012";
    //强制退出
```

CS)

device. SnatchPickup(agentInfo)

# 4.6.16 ChangeAgentPwd(修改座席密码)

#### 接口描述

修改座席密码。

座席的初始密码是在WAS平台上配置(找到座席工号选择查看就可以配置)。第三方应用签入座席后,如果想让该座席号码不被别的应用使用,则可以修改座席的签入密码。

#### 方法定义

```
void ChangeAgentPwd (
string oldPassWord,
string newPassWord
)
```

#### 参数描述

参数	类型	参数规则	描述
oldPassWord	string	必选,长度不超过 32位字符	原密码
newPassWord	string	必选	新密码 <b>说明</b> 1、口令长度8-32位字符串。 2、口令必须包含如下至少3种字符的组合      至少一个小写字母     至少一个大写字母     至少一个数字     至少一个特殊字符: `~!@#\$%^&*()- _=+\ [{{}}];:",<.>/?

## 返回值

#### IDeviceEvents.OnChangeAgentPwdResult

#### 使用示例

```
//c# code
private void btnChangeAgentPwd_Click(object sender, EventArgs e)
{
    //从界面获取参数
    string oldPassWord = this.txtOldPassword.Text.Trim();
    string newPassWord = this.txtNewPassword.Text.Trim();
    //修改密码
    device.ChangeAgentPwd(oldPassWord, newPassWord);
}
```

#### \_\_\_\_

# 4.6.17 AddToRedList(设置优先级名单)

#### 接口描述

设置优先级名单,优先级越高报警电话越优先。座席调用该接口向数据库中的优先级 名单添加电话号码,成功后,该号码通过系统接入码呼入时将会被率先分配座席应 答。

### 方法定义

```
void AddToRedList(
    string phonePrefix,
    int priority
)
```

#### 参数描述

参数	类型	参数规则	描述
phonePrefix	string	必选,长度为1~24位数字	电话号码
priority	int	必选,范围: 1~14	优先级别(数字越大优先级 越高)

#### 返回值

#### IDeviceEvents.OnAddToRedListResult

## 使用示例

```
//c# code
//设置优先级名单
string PhoneNumber = "70777";
int Priority = 14; //最高优先级
device. AddToRedList(PhoneNumber, Priority);
```

# 4.6.18 RemoveFromRedList(取消优先级名单)

## 接口描述

将指定号码,从优先级名单中取消。当确认某电话号码不再需要高等级的优先级时,可调用该接口将其从数据库的优先级名单表中移除。

# 注意事项

如果phoneNumber传入null值,直接返回失败。

# 方法定义

```
void RemoveFromRedList(
    string phonePrefix
)
```

70)

#### 参数描述

参数	类型	参数规则	描述
phonePrefix	string	必选,长度为1~24位数字	被移除优先级名单的电话号码

4 IDevice(设备服务基本接口)

#### 返回值

#### IDeviceEvents.OnRemoveFromRedListResult

#### 使用示例

//c# code //将号码移除出优先级名单 device.RemoveFromRedList(PhoneNumber);

# 4.6.19 RemoveAllRedList(移除全部优先级名单)

#### 接口描述

移除优先级名单中所有号码。随着业务场景的变更, 当数据库中的优先级名单数据出现很大误差性或有用性问题的时候,可以先备份数据后,调用该接口移除所有名单数据,然后重新添加数据。

#### 方法定义

void RemoveAllRedList()

### 参数描述

无

#### 返回值

#### IDeviceEvents.OnRemoveAllRedListResult

### 使用示例

//c# code
device.RemoveAllRedList();

# 4.6.20 SendMessage(座席发送便签)

## 接口描述

座席发送便签。接警员座席A在处理报警人来电过程中,如果有问题需找其它座席协商的时候,除了协商呼叫方式之外,也可调用发送便签接口进行消息互通。

### 注意事项

- 座席机器的"区域和语言"设置要保持一致,否则发送的便签会出现乱码。
- 目标座席会收到OnReceiveMessage回调事件。

CS

#### 方法定义

```
void SendMessage(
string agentId,
string msgContent
)
```

### 参数描述

参数	类型	参数规则	描述
agentId	string	必选,范围: 1~59999	目标座席工号
msgContent	string	必选,不超过169个字节	便签消息内容

#### 返回值

#### IDeviceEvents.OnSendMessageResult

#### 使用示例

```
//c# code
private void btnSendMessage_Click(object sender, EventArgs e)
{
    //目标座席工号
    string agentId = this.txtAgentId.Text.Trim();
    //便签消息
    string message = this.txtMessage.Text.Trim();
    //调用SDK接口发送便签
    device.SendMessage(agentId, message);
}
```

# 4.6.21 SetAutoAnswer(设置是否自动应答)

## 接口描述

设置是否自动应答。座席设置自动应答后,当外电话或内电话到达该座席时,座席所 绑定的电话接听后,座席侧不在需要调用摘机接口就能自动形成通话,否则座席侧需 要调用摘机接口才能形成通话。

# 注意事项

当作为会议成员的时候该接口不适用。

# 方法定义

```
void SetAutoAnswer(
  bool isAutoAnswer
)
```

#### 参数描述

参数	类型	参数规则	描述	
isAutoAnswer	bool	必选	是否自动应答	
			● true: 自动应答	
			● false: 不自动应答	

#### 返回值

#### IDeviceEvents.OnSetAutoAnswerResult

#### 使用示例

//c# code //设置自动应答 device.SetAutoAnswer(true);

# 4.6.22 SetCallData(设置呼叫数据)

### 接口描述

设置呼叫随路数据。座席A呼叫座席B时,可调用该接口附带呼叫数据来描述本次通话的主题、内容、目的等信息,座席B可以通过调用QueryCallData接口和其返回的事件OnQueryCallDataResult获取到呼叫数据。

# 方法定义

```
void SetCallData(
string callData
```

## 参数描述

参数	类型	参数规则	描述
callData	string	必选,不超过1023个字节	呼叫随路数据

## 返回值

#### IDeviceEvents.OnSetCallDataResult

### 使用示例

//c# code //callData主要描述呼叫的主要内容 device.SetCallData(callData); eSDK ICP V200R001C10 接口参考 01(ECC, Windows,

# 4.6.23 QueryCallData(查询呼叫数据)

#### 接口描述

获取呼叫数据。座席A呼叫座席B,可调用SetCallData接口附带呼叫数据来描述本次通话的主题、内容、目的等信息,座席B可以通过调用该接口和其返回的事件 OnQueryCallDataResult获取到呼叫数据。

### 方法定义

void QueryCallData()

#### 参数描述

无

#### 返回值

IDevice Events. On Query Call Data Result

#### 使用示例

//c# code
device.QueryCallData();

# 4.6.24 SetCustomerLevel(设置呼叫的客户级别)

## 接口描述

设置呼叫的客户级别。

当座席在处理来电的时候,当发现来电方是一个潜在的重要客户时,就可以调用该接口设置和提高客户的级别,并存入平台数据库形成重要的信息资源。

# 方法定义

```
void SetCustomerLevel(
string phoneNumber,
int level
```

# 参数描述

参数	类型	参数规则	描述
phoneNumber	string	必选,长度为1~24位数字	当前通话的电话号码
level	int	必选,范围: 1~14	客户级别(数字越大级别越高)

### 返回值

IDeviceEvents.OnSetCustomerLevelResult

#### 使用示例

```
//c# code
string phoneNumber = "70777";
int customerLevel = 14 //客户级别
device.SetCustomerLevel(phoneNumber, customerLevel);
```

# 4.6.25 QueryCallIDOnAgent(查询座席的 CallId)

#### 接口描述

查询指定座席的CallId。

此接口不支持查询座席会议中的CallId,如果被查询座席没有通话或者在会议中,返回结果为空。

## 方法定义

```
void QueryCallIDOnAgent(
string agentId
)
```

### 参数描述

参数	类型	参数规则	描述
agentId	string	必选,范围: 1~59999	座席工号

#### 返回值

#### IDevice Events. On Query Call IDOn Agent Result

## 使用示例

```
//c# code
//agentId被查座席工号
device.QueryCallIDOnAgent(agentId);
```

# 4.6.26 QueryAgentIP(查询座席 IP)

### 接口描述

查询某座席的IP地址,用于实时屏检。如果座席所在的PC有多个IP4类型的IP,则优先返回和CTI IP 同一网段的IP。

# 方法定义

```
void QueryAgentIP(
AgentInfo agentInfo
)
```

4 IDevice(设备服务基本接口)

#### 参数描述

参数	类型	参数规则		描述
agentInfo	AgentInfo	AgentId	必选,范围: 1-59999	座席工号

#### 返回值

#### IDeviceEvents.OnQueryAgentIPResult

#### 使用示例

```
//c# code
//参数agentInfo, 只需对AgentId赋值即可
AgentInfo agentInfo = new AgentInfo();
agentInfo.AgentId = "20012";
device.QueryAgentIP(agentInfo);
```

# 4.6.27 ResetAgentSkillGroups(重置座席技能)

## 接口描述

座席签入成功后, 可以重置签入座席的技能列表。

应用场景:座席签入后可查询平台技能队列信息和查询本座席的技能队列信息,通过比较,如果座席需添加新的平台技能队列,可调用该接口重置座席技能队列。

## 注意事项

● 座席只能重置本座席的技能

# 方法定义

```
void ResetAgentSkillGroups(
string agentId,
List<int> groupIds
)
```

### 参数描述

参数	类型	参数规则	描述
agentId	string	必选,范围: 1-59999	座席工号
groupIds	List <int></int>	必选,正整数集合(最多2000个)	技能列表ID集合

### 返回值

#### IDevice Events. On Reset Agent Skill Groups Result

## 使用示例

```
//c# code
private void btnResetAgentSkill_Click(object sender, EventArgs e)
```

CS)

```
{
    string agentId = "20012";//本座席工号
    List<int> groupIds = new List<int>();//技能ID集合
    groupIds.Add(1);
    groupIds.Add(2);
    groupIds.Add(5);
    //重置座席技能
    device.ResetAgentSkillGroups(agentId, groupIds);
}
```

4 IDevice(设备服务基本接口)

# 4.6.28 SubscribeAgentEvent(订阅座席事件)

#### 接口描述

订阅座席事件,例如座席A订阅座席B的某事件后,座席B的该事件会通知到座席A;座席B签出后,订阅关系自动取消。

### 注意事项

目前只支持订阅久不应答事件OnLongNoAnswer。

## 方法定义

```
void SubscribeAgentEvent(
    string agentId,
    string eventType
)
```

#### 参数描述

参数	类型	参数规则	描述
agentId	string	必选,范围: 1-59999	座席工号
eventType	string	必选, <b>EventTypeEnum</b> 枚举, 字符串长度不超过100	事件类型

#### 返回值

#### IDeviceEvents.OnSubscribeAgentEventResult

## 使用示例

```
//c# code
//agentId被订阅座席工号
//eventType为订阅的事件类型
eventType = "OnLongNoAnswer"
device. SubscribeAgentEvent (agentId, eventType);
```

# 4.6.29 StopSubscribeAgentEvent(停止订阅座席事件)

#### 接口描述

取消订阅座席事件,例如座席A取消订阅座席B的某事件后,原先的订阅关系就不存在了,座席B的该事件不再通知到座席A。

#### 注意事项

目前只支持久不应答事件OnLongNoAnswer。

#### 方法定义

```
void StopSubscribeAgentEvent(
    string agentid,
    string eventType
)
```

#### 参数描述

参数	类型	参数规则	描述
agentid	string	必选,范围: 1-59999	座席工号
eventType	string	必选, <b>EventTypeEnum</b> 枚举,字符 串长度不超过100	事件类型

#### 返回值

#### IDevice Events. On Stop Subscribe Agent Event Result

#### 使用示例

```
//c# code
//agentId被取消订阅座席工号
//eventType为订阅的事件类型
eventType = "OnLongNoAnswer"
device. StopSubscribeAgentEvent(agentId, eventType);
```

# 4.7 IDevice Methods: 录音管理功能

# 4.7.1 PlayRecord(播放录音)

## 接口描述

播放录音。在平台已经配置对呼叫进行录音功能后,每通呼叫建立时座席侧都会收到 OnStartRecord开始录音的事件,从中可以获得录音文件地址信息,另外,录音文件地 址也可在录音文件服务器上取得,然后调用播放接口就可播放。

## 注意事项

- 需要在WAS平台上配置呼叫是否开启录音的功能。
- 只有当录音完成后才能根据录音文件地址播放该文件,不能边录边播。
- 录音播放时,系统主动向上层应用返回IDeviceEvents.OnPlayStatus事件。

## 方法定义

```
void PlayRecord(
string recordId
)
```

#### 参数描述

参数	类型	参数规则	描述
recordId	string	必选,字符串长度不超过129	录音文件地址

#### 返回值

#### IDeviceEvents.OnPlayRecordResult

#### 使用示例

```
//c# code
private void btnPlayRecord_Click(object sender, EventArgs e)

{
    //录音文件地址,可在录音文件服务器上取
    string RecordId = this.txtRecordId.Text.Trim();
    //播放录音
    device.PlayRecord(RecordId));
}
```

# 4.7.2 PlayRecordPause(暂停播放)

#### 接口描述

暂停播放。录音文件在播放过程中如需要做暂停处理可调用该接口,录音暂停播放后可调用PlayRecordResume接口继续播放录音。

### 方法定义

void PlayRecordPause()

#### 参数描述

无。

## 返回值

#### IDeviceEvents.OnPlayRecordPauseResult

### 使用示例

```
//c# code
device.PlayRecordStop();
```

# 4.7.3 PlayRecordResume(继续播放)

## 接口描述

取消暂停后继续播放。

录音文件在播放过程中被暂停后,需继续播放可调用该接口继续播放录音;不需要可调用PlayRecordStop接口停止播放录音。

eSDK ICP V200R001C10 接口参考 01(ECC, Windows,

#### 方法定义

void PlayRecordResume()

### 参数描述

无

### 返回值

#### IDevice Events. On Play Record Resume Result

#### 使用示例

//c# code //继续播放录音 device.PlayRecordResume();

# 4.7.4 PlayRecordSeekTo(播放进度跳转)

#### 接口描述

播放进度跳转。录音文件在播放过程中,如需要跳转到某一指定的位置开始播放,则可调用该接口,跳转可以向前跳也可以向后跳。

#### □说明

当前录音文件已经播放到25s位置,如果需要后退到10s的位置重新开始播放,则跳转参数应输入10000;如果需要前进到30s的位置开始播放,则跳转参数应输入30000。

### 注意事项

- 跳转的时长不能超过录音的总时长。
- 跳转的位置是相对于开始播放录音的时间位置计算,例如跳转时间为10000,则跳转到整个录音的10s位置开始播放。

## 方法定义

void PlayRecordSeekTo(
 int nStep

#### 参数描述

参数	类型	参数规则	描述
nStep	int	必选,非负整数	播放进度,单位:毫秒 说明 参数小于等于录音的总毫秒数,当 超过录音长度则不跳转

#### 返回值

#### IDeviceEvents.OnPlayRecordSeekToResult

4 IDevice (设备服务基本接口)

# 使用示例

//c# code //nStep录音跳转位置ms device.PlayRecordSeekTo(nStep);

# 4.7.5 PlayRecordStop(停止播放)

#### 接口描述

停止播放。录音文件在播放、暂停过程中都可以调用该接口停止播放录音。

#### 方法定义

void PlayRecordStop()

#### 参数描述

无

#### 返回值

#### IDevice Events. On Play Record Stop Result

#### 使用示例

//c# code
device.PlayRecordStop();

# 4.7.6 GetRecordLength(获取录音时长)

#### 接口描述

获取指定录音文件的播放时间长度。获取的录音时长主要作用是对**PlayRecordSeekTo**录音跳转接口的参数设置起到一个参考的作用。

# 方法定义

```
void GetRecordLength(
    string recordId
)
```

### 参数描述

参数	类型	参数规则	描述
recordId	string	必选,字符串长度不超过129	录音文件地址

#### 返回值

#### IDevice Events. On Get Record Length Result

#### 使用示例

//c# code
device.GetRecordLength(recordId);

# 4.7.7 DownloadRecordFile(下载录音文件)

#### 接口描述

录音下载。从服务端下载录音文件到指定目录。音频文件下载格式支持wav格式,也可封装为zip格式下载。

#### 注意事项

文件保存全路径中的文件名后缀需与下载格式保持一致。

### 方法定义

```
void DownloadRecordFile (
    string recordId,
    string localFileName,
    int format,
    string zipPassword
)
```

#### 参数描述

参数	类型	参数规则	描述
recordId	string	必选,字符串长 度不超过129	录音文件地址
localFileName	string	必选	文件保存全路径 <b>说明</b> 字符串长度不超过129, 并以".zip" 或".wav"结尾, 如: "C:\ipcc\c.zip"
format	int	必选	文件下载格式  ● 0: zip格式,文件被压缩为zip格式,但压缩包内还是wav格式  ● 1: wav格式,不压缩
zipPassword	string	可选,保留参数	压缩密码,暂时无效

# 返回值

#### On Download Record File Result

## 使用示例

```
//c# code
//录音文件地址
string fileDownloadPos = this.txtRecordId.Text.Trim();
//保存地址
string fileStorePos = this.txtSavePath.Text.Trim();
//文件格式
```

eSDK ICP V200R001C10 接口参考 01(ECC, Windows,

int format = 1; //下载文件 device.DownloadRecordFile(fileDownloadPos, fileStorePos, formate);

# 4.7.8 SetCustomizedRecordInfo(设置录音关联业务信息)

#### 接口描述

设置录音关联业务信息。

获取录音文件地址后,可以调用该接口向文件添加关联的业务信息,这将便于录音文件的分类和筛选。

### 方法定义

```
void SetCustomizedRecordInfo(
    string recordId ,
    sting data
```

#### 参数描述

参数	类型	参数规则	描述
recordId	string	必选,字符串长度不超过129	录音文件地址.
data	string	必选,Json字符串,具体内容由 CAD和录音管理系统协商	需要反写的录音关联 信息

#### 返回值

#### On Set Customized Record Info Result

#### 使用示例

```
//c# code
//获取录音文件地址后,可为录音文件添加关联信息
device.SetCustomizedRecordInfo(recordId, data)
```

# 4.8 IDevice Methods: 无线资源类功能

# 4.8.1 PTT(抢权)

#### 接口描述

Push To Talk操作。用于座席与无线设备通话。

座席(主席)可以对会议中的群组进行抢权/放权操作。座席(主席)抢权成功后,会议中的所有座席/普通用户说话都能听到彼此,集群用户也能够听到。

#### 注意事项

- 抢权/放权事件,需要通知到会场中的所有座席。
- 目前只支持座席抢权、放权功能。

### 方法定义

```
void PTT(
PTTOperate pttOperate,
string pttLevel
)
```

#### 参数描述

参数	类型	参数规则	描述
pttOperate	PTTOperate	必选	PTT操作类型
pttLevel	string	可选,默认值为1, 范围: 1~15的数字	PTT优先级,按照LTE的定义,优 先级为1~15,数字越小优先级越 高,ECC平台仅进行透明传输

#### 返回值

#### IDeviceEvents.OnPTTResult

### 使用示例

```
//c# code
//放权
device.PTT(PTTOperate.PutDown, pptlevel);
//抢权
device.PTT(PTTOperate.PutUp, pptlevel);
```

# 4.8.2 QueryWirelessInfo(查询无线设备信息)

## 接口描述

查询某个无线设备信息。主要适用于配置在EDS中的窄带设备,在外呼窄带设备时可以先查询然后选择进行外呼。

# 注意事项

无线设备标识从QueryAllWirelessDevice接口的返回参数中获取。

# 方法定义

```
void QueryWirelessInfo(
string deviceId
)
```

 $(\mathbf{S})$ 

参数描述

参数	类型	参数规则	描述
deviceId	string	必选,长度不超过36位的字符串	无线设备标识

4 IDevice (设备服务基本接口)

#### 返回值

#### IDevice Events. On Query Wireless Info Result

#### 使用示例

//c# code

//WirelessDeviceId从QueryAllWirelessDevice接口的返回参数中获取device.QueryWirelessInfo(WirelessDeviceId);

# 4.8.3 QueryAllWirelessDevice(查询所有无线设备信息)

#### 接口描述

查询所有无线设备信息。主要适用于配置在EDS中的窄带设备,在外呼窄带设备时可以先查询然后选择进行外呼。

#### 方法定义

void QueryAllWirelessDevice()

#### 参数描述

无。

#### 返回值

#### IDevice Events. On Query All Wireless Device Result

#### 使用示例

//c# code

device. QueryAllWirelessDevice();

# 4.8.4 QueryUsingWirelessDevice(查询当前使用的无线设备信息)

#### 接口描述

查询系统当前使用的无线设备信息。主要适用于配置在EDS中的窄带设备,在外呼窄带设备时可以先查询然后选择进行外呼。

## 方法定义

void QueryUsingWirelessDevice()

### 参数描述

无。

#### IDevice Events. On Query Using Wireless Device Result

#### 使用示例

//c# code
device.QueryUsingWirelessDevice();

# 4.9 IDevice Methods: 短信服务类功能

# 4.9.1 SendSMS(发送短信)

#### 接口描述

座席主动对外发送短信。此接口也具有群发短信的功能。接/处警系统调用该接口发送短信,输入要发送的短信内容,接收方号码,短信平台IP、端口等信息。

#### 注意事项

- 呼叫中心系统完成与运营商网络的对接。
- 接/处警员已成功签入,并具有SMS的技能。

## 方法定义

```
void SendSMS(
string deviceId,
string phoneNumber,
string note,
string mciIp,
string mciPort,
int ulSeqNo
```

## 参数描述

参数	类型	参数规则	描述
deviceId	string	必选,可转换为正整数的字 符串	短信技能ID
phoneNumber	string	必选,长度为1~24位数字	短信接收者号码 说明 群发短信时号码之间用";"隔 离
note	string	必选,最大长度为140个汉 字、英文或者数字	短信文本内容
mciIp	string	必选,格式例如: "172.22.19.98"	MCI IP地址 说明 如果IP有效,则使用该参数, 否则忽略该参数
mciPort	string	必选,例如: 80	MCI 端口

1	٦	C	1
•		. `	

参数	类型	参数规则	描述
ulSeqNo	int	必选, 非负整数	序列编号

#### IDeviceEvents.OnSendSMSResult

#### 使用示例

```
//c# code
private void btnSMS_Click(object sender, EventArgs e)
{
    //从应用程序界面获取参数
    string deviceId = txtDeviceId.Text.Trim();
    string phoneNumber = txtPhoneNumber.Text.Trim();
    string note = txtNote.Text.Trim();
    string mciIp = txtMciIp.Text.Trim();
    string mciPort = txtMciPort.Text.Trim();
    string txtUlSeqNo = txtulSeqNo.Text.Trim();
    /*参数校验*/
    //发送短信
    device.SendSMS(deviceId, phoneNumber, note, mciIp, mciPort, ulSeqNo));
}
```

# 4.9.2 ReplySMS(回复短信)

#### 接口描述

座席对收到的短信呼叫进行回复。

报警人在发现警情时可以通过发送短信的方式报警,呼叫中心系统将话务分配到指定座席,座席通过调用该接口回复短信给报警人来处理警情。

## 注意事项

- 呼叫中心系统完成与运营商网络的对接。
- 接/处警员已成功签入,并且有具有SMS技能。
- 新短信到达座席时,系统会主动触发事件: IdeviceEvents.OnReceiveSMS。

### 方法定义

```
void ReplySMS(
   int CCBIdx,
   string mediaStreamInfo,
   int ulSeqNo
)
```

### 参数描述

参数	类型	参数规则	描述
CCBIdx	int	必选, 正整数	短信呼叫的内部索引号
mediaStreamInfo	string	必选,最大长度为140位字符	短信内容

٦	

参数	类型	参数规则	描述
ulSeqNo	int	必选, 非负整数	序列编号

#### IDeviceEvents.OnReplySMSResult

#### 使用示例

```
//c# code
public void ReplySMS(int CCBIdx, string mediaStreamInfo, int ulSeqNo)
{
    CCBIdex = 1;//第一个报警人来电为1
    string replySMS = txtSendInfo.Text.Trim();
    string txtUlSeqNo = txtulSeqNo.Text.Trim();
    int ulSeqNo = int.TryParse(txtUlSeqNo, out ulSeqNo);
    //回复短信
    device.ReplySMS(CCBIdx, replySMS, ulSeqNo);
}
```

# 4.9.3 ReleaseCallExWithCCBIdx(座席释放指定短信用户呼叫)

#### 接口描述

座席释放指定短信用户呼叫。

报警人在发现警情时可以通过发送短信的方式报警,呼叫中心系统将话务分配到指定座席。当有多个报警人短信报警被分配到同一个座席时,可通过**OnReceiveSMS**事件中的CCBIdx短信呼叫的内部索引号来区分,处理完报警后可通过该参数来释放不同的报警人呼叫。

### 注意事项

- 呼叫中心系统完成与运营商网络的对接。
- 接/处警员已成功签入,并且有具有SMS技能。

# 方法定义

```
Void ReleaseCallExWithCCBIdx(
   int CCBIdx
)
```

# 参数描述

参数	类型	参数规则	描述
CCBIdx	int	必选, 正整数	短信呼叫的内部索引号

#### 返回值

#### IDevice Events. On Release Call ExWith CCBIdx Result

#### 使用示例

```
//c# code
//释放短信内部索引号为1的呼叫
device. ReleaseCallExWithCCBIdx(1);
```

# 4.9.4 TransferSMSToAgent(短信转移)

#### 接口描述

座席处理多个短信呼叫,可选择将其中一个呼叫转移给另一个座席,转移哪一个呼叫由短信呼叫的内部索引号CCBIdx来决定。

#### 注意事项

目标座席必须为示闲状态。

### 方法定义

```
void TransferSMSToAgent (
  int CCBIdx,
  string targetAgent
)
```

#### 参数描述

参数	类型	参数规则	描述
CCBIdx	int	必选,正整数	短信呼叫的内部索引号
targetAgent	string	必选,范围: 1-59999	目标座席工号

### 返回值

#### IDevice Events. On Transfer SMSToAgent Result

#### 使用示例

```
//c# code
//将第一个报警人的呼叫转給座席20012
targetAgent = "20012"
device.TransferSMSToAgent(1, targetAgent);
```

# 4.10 IDevice Methods:邮件服务类功能

使用邮件相关功能, 需要安装客户端组件。

登录**Support IPCC专区**,单击"软件->V200R001C80SPC600",下载CRSV300R006C87SPC600\_PUBLIC\_ICD.zip,解压缩文件,获取Client\_update\_pack.zip,运行其中脚本和控件。

# 4.10.1 SendEmail(发送邮件)

#### 接口描述

座席主动对外发送邮件。通过**IDevice.GetSendEmailIndex**接口获取到邮件索引号,通过**WAS**平台查询邮件技能**ID**后,调用该接口向指定收件人发送邮件。

#### 注意事项

座席已成功签入,并且有具有邮件技能。

### 方法定义

```
void SendEmail(
string receiverAddress,
string skillID,
int index
)
```

#### 参数描述

参数	类型	参数规则	描述
receiverAddress	string	必选	收件人邮件地址,目前只能是 一个收件人
skillID	string	必选, 非负整数	邮件技能ID
index	int	必选,最大值为4字 节正整数	发送的邮件索引号,必须是4个字节数字,由 GetSendEmailIndex接口获取

### 返回值

#### IDeviceEvents.OnSendEmailResult

# 使用示例

```
//c# code
private void btnSendEmail_Click(object sender, EventArgs e)
{
    string receiverAddress = this.txtReceiverAddress.Text.Trim();
    string skillID = "7";//邮件技能Id可在Was平台查看
    string index = mailIndex;//由方法获取发送邮件索引号获得
    //发送邮件
    device.SendEmail(receiverAddress, skillID, index);
}
```

# 4.10.2 ReplyEmail(回复邮件)

### 接口描述

回复邮件。座席利用该接口对收到的邮件呼叫进行回复,回复邮件时邮件呼叫不能释放,否则回复失败。对于座席而言的邮件回复或主动发送,邮件控件中所传To值都是指向收件人邮箱地址,From指发件人邮箱地址。

CS)

#### 注意事项

- 呼叫中心系统完成与运营商网络的对接。
- 座席已成功签入,并且有具有邮件技能。
- 安装客户端组件:

登录**Support IPCC专区**,单击"软件>V200R001C80SPC600",下载CRSV300R006C87SPC600\_PUBLIC\_ICD.zip,解压缩文件,获取Client\_update\_pack.zip,运行其中脚本和控件。如果ECC SDK使用AgentGateway的服务端接口,就不需要通过CTI的Windows安装包安装客户端组件了,只需要注册邮件控件即可。

4 IDevice(设备服务基本接口)

#### 方法定义

void ReplyEmail()

#### 参数描述

无。

#### 返回值

IDeviceEvents.OnReplyEmailResult

#### 使用示例

//c# code
device.ReplyEmail();

# 4.10.3 QueryAgentEmailCfgInfo(查询座席邮件配置信息)

#### 接口描述

查询当前座席的Email配置参数信息,通过返回的结果事件获取,主要包括邮件的发送方和接收方邮箱地址、SMTP和POP3服务器地址。

#### 方法定义

void QueryAgentEmailCfgInfo()

## 参数描述

无。

#### 返回值

IDevice Events. On Query Agent Email Cfg Info Result

#### 使用示例

//c# code
device.QueryAgentEmailCfgInfo()

# 4.10.4 GetSendEmailIndex(获取发送邮件索引号)

#### 接口描述

座席在发送邮件时,获取要发送的邮件索引号。该接口需要在发送邮件内容前调用,并通过IDeviceEvents.OnGetSendEmailIndexResult结果事件获得。

### 方法定义

void GetSendEmailIndex()

#### 参数描述

无。

#### 返回值

IDeviceEvents.OnGetSendEmailIndexResult

#### 使用示例

//c# code
device.GetSendEmailIndex();

# 4.10.5 ReleaseMailCall(停止邮件呼叫)

#### 接口描述

停止邮件呼叫。

座席完成接警事务的处理后,调用该接口释放当前邮件呼叫。ECC SDK向呼叫中心平台发起释放某邮件呼叫的请求,通过IDeviceEvents.OnReleaseMailCall获取返回结果,通过获取结果可以执行停止呼叫后的相关业务操作,如关闭发送邮件窗口等。

#### 注意事项:

- 呼叫中心系统完成与运营商网络的对接。
- 座席已成功签入,并且具有邮件技能。
- 安装客户端组件:

登录**Support IPCC专区**,单击"软件>V200R001C80SPC600",下载CRSV300R006C87SPC600\_PUBLIC\_ICD.zip,解压缩文件,获取Client update pack.zip,运行其中脚本和控件。

如果ECC SDK使用AgentGateway的服务端接口,就不需要通过CTI的Windows安装包安装客户端组件了,只需要注册邮件控件即可。

# 方法定义

void ReleaseMailCall()

### 参数描述

无。

4 IDevice (设备服务基本接口)

CS'

#### 返回值

#### IDeviceEvents.OnReleaseMailCall

#### 使用示例

//c# code
device.ReleaseMailCall();

# 4.11 IDevice Methods: 视频通话类功能

# 4.11.1 PickupVCCall(接听视频电话)

### 接口描述

座席接听视频电话。

应用场景:移动端APP上发起匿名呼叫,座席客户端收到振铃事件 OnVCPhoneStatusNotify,座席调用本接口接听后,会收到OnNewVCCall事件。

#### 注意事项

- 座席处于示闲状态。
- 座席己配置关联TE软终端。
- 座席关联的TE软终端已签入。

#### 方法定义

void PickupVCCall()

#### 参数描述

无。

#### 返回值

IDeviceEvents.OnPickupVCCallResult

#### 使用示例

//c# code device.PickupVCCall();//device为接口类IDevice实例

# 4.11.2 HangupVCCall(挂断视频电话)

#### 接口描述

座席挂断视频电话。

应用场景:移动端APP上发起匿名呼叫,座席接听该呼叫进入视频通话,座席调用本接口挂断通话,并收到视频电话状态通知事件OnVCPhoneStatusNotify。

#### 注意事项

- 座席已配置关联TE软终端。
- 座席关联的TE软终端已签入。
- 座席处于视频电话中。
- 座席在视频电话来电振铃时,不能挂断该通话。

#### 方法定义

void HangupVCCall()

#### 参数描述

无。

#### 返回值

IDeviceEvents.OnHangupVCCallResult

#### 使用示例

//c# code device.HangupVCCall();//device为接口类IDevice实例

# 4.12 IDevice Methods: 话机管理功能

# 4.12.1 PhoneInitialize(初始化话机)

#### 接口描述

初始化话机相关配置信息,并根据话机的不同配置完成帐号注册流程。通常情况下,若用户需实现IPT线路话机相关功能,那么进行登录操作时,需要调用该接口,按话机使用模式进行初始化。

### 注意事项

- Voice.ocx加载并初始化成功
- 当前话机未进行过初始化

### 方法定义

void PhoneInitialize(
PhoneUsedMode phoneMode,
string phoneNum,
string phonePassword,
string sipServerInfo,
string audioDeviceInfo,
LocalInfo localInfo,
string settings

## 参数描述

参数	类型	参数规则	描述
phoneMode	PhoneUse dMode	必选	话机使用模式
phoneNum	string	必选,不超过 24位的数字	内置软电话号码
phonePassword	string	可选	内置软电话密码
sipServerInfo	string	必选	SIP服务器地址信息,可以填写1个 主地址和最多3个备用地址,格式为 IP1:Port1 IP2:Port2,端口范围为 1024~65535
audioDeviceInfo	string	可选	声卡设备信息,格式为Json格式字符串。参考表4-1
localInfo	LocalInfo	可选	本地信息
settings	string	可选	非常用配置项,格式为Json字符串,参考表4-2

## 表 4-1 声卡设备信息 Json 字符串格式说明

参数	类型	描述
mainSpeakerDeviceName	string	主声卡扬声器设备所在声卡名称, 可为空,为空时将选择默认声卡设 备
mainMicrophoneDeviceNa me	string	主声卡麦克风设备所在声卡名称, 可为空,为空时将选择默认声卡设 备
isAidEnable	bool	是否需要使用辅助声卡
		● true: 是
		● false: 否
		<b>说明</b> 如果需要辅助声卡,则主声卡只设置
aidSpeakerDeviceName	string	辅助声卡扬声器设备名称,可为 空,启用辅助声卡时不可为空
aidMicrophoneDeviceName	string	辅助声卡麦克风设备名称,可为 空,启用辅助声卡时不可为空

参数	类型	描述
mainIp	string	主声卡所在客户端地址信息,默认 为127.0.0.1,可为空
mainControlPort	int	主声卡控制信令端口,默认为 19200,可为空
mainMediaPort	int	主声卡媒体端口,默认为19221, 可为空
aidIp	string	辅助声卡所在客户端地址信息,默 认为127.0.0.1,可为空
aidMediaPort	int	辅助声卡媒体流端口,默认为 19301,可为空

#### 表 4-2 非常用配置项 Json 格式说明

参数	类型	描述
sipDomain	string	SIP服务器域名信息,默认为 example.com
isAutoAnswerIPCC	bool	是否对IPCC线路呼叫进行自动应答 处理。默认为false • true:自动应答 • false:手动应答
isAutoAnswerIPT	bool	是否对IPT线路呼叫进行自动应答 处理。默认为false • true: 自动应答 • false: 手动应答
encryptType	string	加密类型
phoneNum	string	当开启IPT联动时,此处需要传入 话机号码。可为空。

## 返回值

#### On Phone Initialize Result

当开启联动时,还返回结果事件OnPhoneLinkageStatusNotify。

#### 使用示例

```
//c# code
//内置软电话模式
var phoneUsedMode = PhoneUsedMode.PhoneEmbeddedMode;
//非常用配置项
SettingsInfo settings = new SettingsInfo();
settings.isAutoAnswerIPCC = false;
settings.isAutoAnswerIPT = false;
settings.sipDomain = "example.com";
settings.phoneNum = "";
//本地信息
LocalInfo localInfo = new LocalInfo();
localInfo.LocalAudioPort = 19100;
localInfo.LocalIp = 10.135.0.164;
localInfo.LocalSipPort = 19090;
//声卡设备信息
string aidDeviceInfo = "{\"isAidEnable\":false,\"mainSpeakerDeviceName\":\"扬声器 (Citrix HDX
Audio)\", \"mainMicrophoneDeviceName\":\"麦克风 (Citrix HDX Audio)\"}";
//将对象转换成Json字符串
JavaScriptSerializer jss = new JavaScriptSerializer();
var settingsData = jss.Serialize(settings);
//调用初始化话机接口, device为接口类IDevice实例
device. PhoneInitialize (phoneUsedMode, "4501", "1qaz@WSX", "10.175.0.247:5060", aidDeviceInfo,
localInfo, settingsData));
```

# 4.12.2 PhoneUnInitialize(去初始化话机)

#### 接口描述

在程序退出之前, 去初始化话机, 释放话机资源。

应用场景: 若调用该接口时,有进行中的通话,则通话将被挂断,且不会上报话机呼叫释放事件OnPhoneReleaseEvent。

#### 注意事项

- Voice.ocx加载并初始化成功
- 已经初始化话机

#### 方法定义

void PhoneUnInitialize()

## 参数描述

无。

## 返回值

#### **OnPhoneUnInitializeResult**

## 使用示例

```
//c# code
//调用去初始化话机接口,device为接口类IDevice实例
device.PhoneUnInitialize();
```

# 4.12.3 PhoneAnswer(话机应答)

#### 接口描述

当话机收到振铃事件后,调用该接口进行应答,支持联动场景和内置软电话非联动场景,联动场景下,可选择调用该接口或在联动话机上进行应答。

## 注意事项

- Voice.ocx加载并初始化成功
- 已经初始化话机
- 收到话机振铃状态上报事件

## 方法定义

```
void PhoneAnswer(
string callSessionId
```

## 参数描述

参数	类型	参数规则	描述
callSessionId	string	必选,范围: 0~4294967295	呼叫唯一标识

#### 返回值

#### **OnPhoneAnswerResult**

#### 使用示例

```
//c# code
//调用话机应答接口,device为接口类IDevice实例
device.PhoneAnswer("14684865313");
```

# 4.12.4 PhoneRelease(话机释放)

## 接口描述

释放话机呼叫,用于挂断振铃、通话、保持等状态下的呼叫,支持联动场景和内置软电话非联动场景,联动场景下,可选择调用该接口或在联动话机上进行释放。

调用该接口成功后,除OnPhoneReleaseResult事件外,还将触发OnPhoneReleaseEvent事件。

## 注意事项

- Voice.ocx加载并初始化成功
- 已经初始化话机
- 当前正在呼叫处理中

## 方法定义

```
void PhoneRelease(
string callSessionId
)
```

## 参数描述

参数	类型	参数规则	描述
callSessionId	string	必选,范围: 0~4294967295	呼叫唯一标识

#### 返回值

#### **OnPhoneReleaseResult**

#### 使用示例

```
//c# code
//调用话机释放接口, device为接口类IDevice实例
device.PhoneRelease("14684865313");
```

# 4.12.5 PhoneMute(话机静音)

## 接口描述

设置话机音麦克风设备静音或取消静音,即静音后对端将听不到己端声音,支持联动场景和内置话机非联动场景,联动场景下可与联动话机交互操作。

## 注意事项

- Voice.ocx加载并初始化成功。
- 已经初始化话机。
- 被静音呼叫当前正在通话中。

## 方法定义

```
void PhoneMute(
string callSessionId,
bool isMute
)
```

参数	类型	参数规则	描述
callSessionId	string	必选,范围: 0~4294967295	呼叫唯一标识
isMute	bool	必选	是否静音
			● true: 静音
			● false: 取消静音

## 返回值

#### **OnPhoneMuteResult**

#### 使用示例

```
//c# code
//调用话机静音接口,device为接口类IDevice实例
device.PhoneMute("14684865313", true);
```

## 4.12.6 PhoneSwitchSoundDevice(话机呼叫切换声卡设备)

## 接口描述

把指定呼叫切换到辅助音频设备上,或者从辅助音频设备切回主设备,仅支持内置软电话场景。

- 激活辅助声卡后,接听普通IPT线路通话;
- 集群调度场景下,选择某个Monitor状态下的集群群组。

## 注意事项

- Voice.ocx加载并初始化成功。
- 已经初始化话机。
- 当前已经激活辅助声卡。
- 被切换呼叫正在通话中。
- 辅助声卡上只能存在一路通话。如果存在主声卡上多路呼叫的场景,则用户需要 一个通话列表,用户自己选择将哪一路通话切换到辅助声卡上。

## 方法定义

```
void PhoneSwitchSoundDevice(
string callSessionId,
bool isToAidDevice
```

## 参数描述

参数	类型	参数规则	描述
callSessionId	string	必选,范围: 0~4294967295	呼叫唯一标识
isToAidDevice	bool	必选	是否为切换到辅助设备 <ul><li>true: 辅助设备</li><li>false: 主设备</li></ul>

## 返回值

#### **OnPhoneSwitchSoundDeviceResult**

#### 使用示例

```
//c# code
//调用话机呼叫切换声卡设备接口,将呼叫切换到辅助设备上
//device为接口类IDevice实例
device. PhoneSwitchSoundDevice ("14684865313", true);
```

# 4.12.7 PhoneGetSoundDevice(话机查询声卡设备信息)

#### 接口描述

获取音频设备信息,只支持内置软电话场景。

调用本接口不需要事先初始化话机,通过本接口查询到的声卡设备信息,可用于**话机** 初始化和话机设置声卡设备信息。

## 注意事项

Voice.ocx加载并初始化成功。

#### 方法定义

```
void PhoneGetSoundDevice(
   bool isInput
)
```

#### 参数描述

参数	类型	参数规则	描述	
isInput	bool	必选	必选 是否为输入设备	
			● true: 输入设备	
			● false: 输出设备	

#### 返回值

#### **OnPhoneGetSoundDeviceResult**

#### 使用示例

```
//c# code
//调用话机查询声卡设备信息接口,查询输入设备(麦克风设备)信息
//device为接口类IDevice实例
device. PhoneGetSoundDevice(true);
```

# 4.12.8 PhoneSetSoundDevice(话机设置声卡设备信息)

#### 接口描述

设置主进程的音频设备信息,只支持内置软电话场景。

调用本接口不需要事先初始化话机,接口所用声卡设备信息可通过**话机查询声卡设备信息**接口查询。

建议一次使用中,主进程音频设备一经设置便不再修改。

eSDK ICP V200R001C10 接口参考 01(ECC, Windows,

初始化话机接口中对主声卡设备信息设置的效果与本接口一致。

## 注意事项

Voice.ocx加载并初始化成功。

## 方法定义

```
void PhoneSetSoundDevice(
bool isInput,
string deviceIndex
```

#### 参数描述

参数	类型	参数规则	描述
isInput	bool	必选 是否为输入设备	
			● true: 输入设备
			● false: 输出设备
deviceIndex	string	必选	设备索引号,可通过 <b>话机查询声卡设备</b> 信息接口获取。

## 返回值

#### **OnPhoneSetSoundDeviceResult**

## 使用示例

```
//c# code
//device为接口类IDevice实例
device.PhoneSetSoundDevice (true,"1");
```

# 4.12.9 PhoneGetVolume(话机查询设备音量)

## 接口描述

话机查询当前音频设备音量。

调用本接口不需要事先初始化话机。

单声卡模式下获取主声卡设备音量;双声卡模式下获取辅助声卡设备音量。

无论当前是否在通话中,无论是否已激活辅助声卡,均可调用本接口进行设备音量查询。

## 注意事项

Voice.ocx加载并初始化成功。

#### 方法定义

```
void PhoneGetVolume(
   bool isInput
)
```

## 参数描述

参数	类型	参数规则	描述
isInput	bool	必选	是否为输入设备  ● true: 输入设备
			● false: 输出设备

#### 返回值

#### **OnPhoneGetVolumeResult**

#### 使用示例

```
//c# code
//调用话机查询设备音量接口,查询输入设备(麦克风设备)音量
//device为接口类IDevice实例
device. PhoneGetVolume(true);
```

## 4.12.10 PhoneGetDeviceVolume(话机查询主、辅助设备音量)

## 接口描述

话机查询当前声卡设备音量。

调用本接口不需要事先初始化话机。

可以根据传入的deviceType来指定获取主(辅助)声卡音量。

无论当前是否在通话中,无论是否已激活辅助声卡(未激活辅助声卡只能查询主声卡设备音量),均可调用本接口进行设备音量查询。

### 注意事项

Voice.ocx加载并初始化成功。

```
void PhoneGetDeviceVolume(
   bool isInput,
   AudioDeviceType deviceType
)
```

参数	类型	参数规则	描述
isInput	bool	必选	是否为输入设备 <ul><li>true: 输入设备</li><li>false: 输出设备</li></ul>
deviceType	AudioDeviceType	必选	声卡设备类型

#### 返回值

#### **OnPhoneGetDeviceVolumeResult**

## 使用示例

//c# code //调用话机查询主、辅助设备音量接口,查询辅助声卡输入设备音量 //device为接口类IDevice实例 device.PhoneGetDeviceVolume(true, AudioDeviceType. AidDevice);

# 4.12.11 PhoneSetVolume(话机调整设备音量)

## 接口描述

调整指定设备音量。

调用本接口不需要事先初始化话机。

单声卡模式下设置主声卡设备音量;双声卡模式下设置辅助声卡设备音量。

无论当前是否在通话中,无论是否已激活辅助声卡,均可调用本接口进行设备音量调整。

## 注意事项

Voice.ocx加载并初始化成功。

## 方法定义

void PhoneSetVolume(
 bool isInput,
 int volumeValue

参数	类型	参数规则	描述
isInput	bool	必选	是否为输入设备
			● true: 输入设备
			● false: 输出设备

4 IDevice(设备服务基本接口)

参数	类型	参数规则	描述
volumeValue	int	必选,范围: 0~100	音量大小

### 返回值

#### **OnPhoneSetVolumeResult**

#### 使用示例

//c# code //调用话机调整设备音量接口,设置输入设备音量为50 //device为接口类IDevice实例 device.PhoneSetVolume(true, 50);

# 4.12.12 PhoneSetDeviceVolume(话机调整辅助设备音量)

## 接口描述

调整辅助设备音量。

调用本接口不需要事先初始化话机。

可以根据传入的deviceType来指定设置主(辅助)声卡音量。

无论当前是否在通话中,无论是否已激活辅助声卡(未激活辅助声卡只能调整主声卡设备音量),均可调用本接口进行设备音量调整。

## 注意事项

Voice.ocx加载并初始化成功。

## 方法定义

void PhoneSetDeviceVolume(
 bool isInput,
 int volumeValue,
 AudioDeviceType deviceType

参数	类型	参数规则	描述
isInput	bool	必选	是否为输入设备 <ul><li>true: 输入设备</li><li>false: 输出设备</li></ul>
volumeValue	int	必选,范围: 0~100	音量大小
deviceType	AudioDevic eType	必选	声卡设备类型

4 IDevice(设备服务基本接口)

#### 返回值

#### **OnPhoneSetDeviceVolumeResult**

## 使用示例

```
//c# code
//调用话机调整辅助设备音量接口,设置辅助声卡输入设备音量为50
//device为接口类IDevice实例
device.PhoneSetDeviceVolume(true, 50, AudioDeviceType.AidDevice);
```

# 4.12.13 PhoneGetVolumeGainByCall(话机查询指定呼叫音量增益)

## 接口描述

查询指定呼叫音量的当前增益值,以便进行调整。

应用场景:查询到的呼叫音量增益用于在设备音量一定的情况下,对具体某一路呼叫进行单独的音量调整,可用于多路呼叫混音的情况下突出某一路呼叫。

#### 注意事项

- Voice.ocx加载并初始化成功。
- 已经初始化话机。
- 当前存在处理中的呼叫。

## 方法定义

```
void PhoneGetVolumeGainByCall(
   string callSessionId,
   bool isInput
)
```

## 参数描述

参数	类型	参数规则	描述
callSessionId	string	必选,范围: 0~4294967295	呼叫唯一标识
isInput	bool	必选	是否为输入设备
			● true: 输入设备
			● false: 输出设备

## 返回值

#### On Phone Get Volume Gain By Call Result

## 使用示例

```
//c# code
//调用话机查询指定呼叫音量增益接口,获取该呼叫输入设备音量增益值
//device为接口类IDevice实例
device.PhoneGetVolumeGainByCall("14684865313", true);
```

#### CS)

# 4.12.14 PhoneSetVolumeGainByCall(话机设置指定呼叫音量)

#### 接口描述

调整指定呼叫的当前音量增益大小。呼叫音量增益用于在设备音量一定的情况下,对具体某一路呼叫进行单独的音量调整,可用于多路呼叫混音的情况下突出某一路呼叫。

## 注意事项

- Voice.ocx加载并初始化成功。
- 已经初始化话机。
- 当前存在通话。

## 方法定义

```
void PhoneSetVolumeGainByCall(
    string callSessionId,
    bool isInput,
    int gainValue
```

## 参数描述

参数	类型	参数规则	描述
callSessionId	string	必选,范围: 0~4294967295	呼叫唯一标识
isInput	bool	必选	是否为输入设备
gainValue	int	必选,范围: 0~10	音量增益大小,音量增 益值越大,音量越高

## 返回值

#### **OnPhoneSetVolumeGainByCallResult**

## 使用示例

```
//c# code
//调用话机设置指定呼叫音量增益接口,设置该呼叫输入设备音量增益值为5
//device为接口类IDevice实例
device.PhoneSetVolumeGainByCall("14684865313", true, 5);
```

# 4.13 IDevice Methods: 集群调度类功能

4 IDevice(设备服务基本接口)

#### CS'

# 4.13.1 TalkingGroupQuery(查询集群群组)

## 接口描述

查询集群群组信息,可以通过集群群组号,查询指定集群群组详细信息;也可以通过输入分页信息,查询所有集群群组信息,支持分页查询。

## 注意事项

当前已经进行过集群群组功能初始化,即已成功登录UPortal。

## 方法定义

```
void TalkingGroupQuery(
  int queryType,
  Int pageSize,
  string content
)
```

#### 参数描述

参数	类型	参数规则	描述
queryType	int	必选,1或2的整数	查询类型。 <ul><li>● 1: 指定分页进行查询</li><li>● 2: 指定集群群组进行查询</li></ul>
pageSize	int	必选	每页显示的集群群组数量。 <b>说明</b> 当queryType的值为1时,该参数的值有效 且取值应大于等于2
content	string	必选	查询内容 <b>说明</b> 当查询类型为1时,为指定页的编号,即 PageNumber 当查询类型为2时,为指定集群群组唯一 标识,即TalkingGroupId

## 返回值

#### **OnTalkingGroupQueryResult**

## 使用示例

```
//c# code
//分页查询,每页记录数5,查询第1页所有记录,device为接口类IDevice实例
device. TalkingGroupQuery(1, 5, "1");
//指定集群群组查询
device. TalkingGroupQuery(2, 1, "9996300");
```

# 4.13.2 TalkingGroupQueryState(查询集群群组状态)

## 接口描述

查询集群群组状态信息,可以查询指定集群群组详细状态信息,包含该群组的所有用户对该群组的操作(订阅、监听、选择等)状态。

#### 注意事项

当前已经进行过集群群组功能初始化,即已成功登录UPortal。

## 方法定义

```
void TalkingGroupQueryState(
string groupNumber
```

#### 参数描述

参数	类型	参数规则	描述
groupNumber	string	必选	集群群组号码

## 返回值

#### **OnTalkingGroupQueryStateResult**

## 使用示例

```
//c# code
device. TalkingGroupQueryState("9996300");//device为接口类IDevice实例
```

## 4.13.3 TalkingGroupAdd(新增集群群组)

## 接口描述

调度席新增一条集群群组信息。新增前,请参考**TalkingGroup**中对各项参数的描述,确保每一项参数正确填写,同时,确保没有添加过具有相同集群群组名称或号码的集群群组。

## 注意事项

当前已经进行过集群群组功能初始化,即已成功登录UPortal。

```
void TalkingGroupAdd(
TalkingGroup groupInfo
)
```

参数	类型	参数规则	描述
groupInfo	TalkingGroup	必选	需要新增的集群群组信息。

#### 返回值

#### **OnTalkingGroupAddResult**

#### 使用示例

```
//c# code
//创建TalkingGroup类实例
TalkingGroup talkingGroup = new TalkingGroup();
//集群群组名称
talkingGroup.talkGroupName = "test";
//集群群组号码
talkingGroup.talkGroupNumber = "9996300";
//集群群组类型,普通群组
talkingGroup.talkGroupType = 1;
//无线集群群组号码
string radioGroupNumber = "70001,70002";
talkingGroup.radioGroupNumber = new List<string>(radioGroupNumber.Split(',');
//调度席号码
string dispatchNumber = "4501, 4502";
talkingGroup.dispatchNumber = new List<string>(dispatchNumber.Split(',');
talkingGroup.otherUserNumber = new List<string>();
talkingGroup.recordFileAddr = "";
talkingGroup.talkGroupDesc = "";
//调用接口新增一个普通集群群组, device为接口类IDevice实例
device.TalkingGroupAdd(talkingGroup);
```

# 4.13.4 TalkingGroupModify(修改集群群组)

#### 接口描述

调度席修改集群群组信息。通过调用**TalkingGroupQuery**接口查询集群群组信息,在对集群群主信息修改前,请参考**TalkingGroup**中对各项参数的描述,确保每一项参数正确填写,同时,集群群组号码和群组类型不能修改。

## 注意事项

- 当前已经进行过集群群组功能初始化,即已成功登录UPortal。
- 当前存在需要操作的集群群组。

```
void TalkingGroupModify(
TalkingGroup groupInfo
)
```

参数	类型	参数规则	描述
groupInfo	TalkingGrou p	必选	需要修改的集群群组信息,不支持修改 集群群组号码和群组类型

## 返回值

#### **OnTalkingGroupModifyResult**

## 使用示例

```
//c# code
//创建TalkingGroup类实例
TalkingGroup talkingGroup = new TalkingGroup();
//集群群组名称
talkingGroup.talkGroupName = "test";
//集群群组号码
talkingGroup.talkGroupNumber = "9996300";
//集群群组类型,普通群组
talkingGroup.talkGroupType = 1;
//无线集群群组号码
string radioGroupNumber = "70001,70002";
talkingGroup.radioGroupNumber = new List<string>(radioGroupNumber.Split(',');
//调度席号码
string dispatchNumber = "4501, 4502";
talkingGroup.dispatchNumber = new List<string>(dispatchNumber.Split(',');
talkingGroup.otherUserNumber = new List<string>();
talkingGroup.recordFileAddr = "";
talkingGroup.talkGroupDesc = "";
device. TalkingGroupModify(talkingGroup);//调用接口修改一个普通集群群组,device为接口类IDevice实例
```

# 4.13.5 TalkingGroupDelete(删除集群群组)

## 接口描述

调度席删除一条集群群组信息。

应用场景: 若调度席正在监听某个集群群组时,该集群群组被删除,那么前者的监听呼叫将被挂断,其他类似的操作也会中断。

## 注意事项

- 当前已经进行过集群群组功能初始化,即已成功登录UPortal。
- 当前存在需要操作的集群群组。

```
void TalkingGroupDelete(
    string groupNum
)
```

参数	类型	参数规则	描述
groupNum	string	必选	集群群组号码

#### 返回值

#### On Talking Group Delete Result.

#### 使用示例

//c# code //调用接口删除一个集群群组,device为接口类IDevice实例 device.TalkingGroupDelete("9996300");

# 4.13.6 RadioGroupQuery(查询无线集群群组)

## 接口描述

查询无线集群群组信息,现阶段支持分页查询,根据查询结果可完成创建、修改集群群组等操作,该接口支持同时多次并发调用。

## 注意事项

当前已经进行过集群群组功能初始化,即已成功登录UPortal。

## 方法定义

```
void RadioGroupQuery(
   int queryType,
   int pageSize,
   string content
```

参数	类型	参数规则	描述
queryType	int	必选,1或2的整数	查询类型。
			● 1: 指定分页进行查询
			● 2: 指定集群群组进行查询
pageSize	int	必选	每页显示的集群群组数量。
			<b>说明</b> 当queryType的值为1时,该参数的值有效 且取值应大于等于2

CS)

eSDK ICP V200R001C10 接口参考 01(ECC, Windows,

参数	类型	参数规则	描述
content	string	必选	查询内容
			<b>说明</b> 当查询类型为1时,为指定页的编号,即 PageNumber
			当查询类型为2时,为指定集群群组号 码,即TalkingGroupId

#### 返回值

#### On Radio Group Query Result.

#### 使用示例

//c# code //分页查询,每页记录数5,查询第1页所有记录,device为接口类IDevice实例 device.RadioGroupQuery(1, 5, "1");

# 4.13.7 TalkingGroupSubscribe(订阅集群群组)

## 接口描述

调度席可以通过该接口订阅指定集群群组,以便时刻关注该集群群组的相关信息,以及进行监听该集群群组等后续操作。混编集群群组不支持订阅操作。

订阅成功后,当被订阅的集群群组用户状态发生变更时,调度席能通过 OnTalkingGroupUserStateChange事件,及时收到该状态变更推送消息。

调度席也可以通过该接口取消订阅指定集群群组。

## 注意事项

- 当前已经进行过集群群组功能初始化,即已成功登录UPortal。
- 当前调度席为对应集群群组的调度席成员。

## 方法定义

```
void TalkingGroupSubscribe(
   int expire,
   string groupNum
```

参数	类型	参数规则	描述
expire	int	必选,范围: 0~7200	订阅时长,单位秒 说明 时间为0时表示取消订阅
groupNum	string	必选	集群群组号码

#### 返回值

#### **OnTalkingGroupSubscribeResult**

#### 使用示例

//c# code //调用接口订阅集群群组1小时,device为接口类IDevice实例 device.TalkingGroupSubscribe(3600, "9996300");

## 4.13.8 TalkingGroupMonitor(Monitor)

#### 接口描述

调度席可以通过该接口监听指定集群群组或停止监听指定集群群组,监听过程为集群群组到调度席的半双工通话。混编集群群组不支持监听操作。

#### 注意事项

- 当前已经进行过集群群组功能初始化,即已成功登录UPortal。
- 调度席绑定话机需完成内置软电话模式下的话机初始化并激活辅助声卡。
- 调度席已经订阅该集群群组。
- 调度席调用该接口发起监听或停止监听。
  - 如果发起监听成功,则调度席所绑定的话机将收到对应集群群组来电振铃并自动应答,此时调度席成功创建并加入集群群组会议,但是只能听不能说。
  - 如果调度席从监听状态退出,则调度席将会退出该会议,与该会议的呼叫连接释放。
  - 调度席同时监听多个集群群组时,对应呼叫在本地主声卡扬声器设备上混音播放。

## 方法定义

```
void TalkingGroupMonitor(
   string groupNum,
   bool isMonitor,
   bool isAutoSubscribe
```

参数	类型	参数规则	描述
groupNum	string	必选	集群群组号码
isMonitor	bool	必选	开始监听还是停止监听  ● true: 开始监听  ● false: 停止监听
isAutoSubsc ribe	bool	可选,默认值为true	监听时是否自动处理订阅  ● true: 监听时自动处理订阅  ● false: 监听时不自动处理订阅

#### 返回值

#### **OnTalkingGroupMonitorResult**

#### 使用示例

//c# code //开始监听集群群组,并自动处理订阅,device为接口类IDevice实例 device.TalkingGroupMonitor("9996300", true, true);

# 4.13.9 TalkingGroupSelect(Monitor/Select 切换)

#### 接口描述

调度席可以通过该接口设置群主通话状态,从Monitor切换为Select,进行全双工通话;从Select切换为Monitor,进行半双工通话。如果调度席要在集群群组中抢权,则必须先Select才能进行抢权或放权动作。

## 注意事项

- 当前已经进行过集群群组功能初始化,即已成功登录UPortal。
- 调度席绑定话机需完成内置软电话模式下的话机初始化并激活辅助声卡。
- 调度席已经订阅指定集群群组。
- 调度席已经监听指定集群群组。
- 调度席调用该接口进行Monitor/Select切换。 当从Monitor切换为Select后,所监听的呼叫将会从主声卡切换到辅助声卡上。 当从Select切换回Monitor后,所监听的呼叫将会从辅助声卡切换回主声卡。

## 方法定义

```
void TalkingGroupSelect(
string groupNum,
bool isSelect
```

## 参数描述

参数	类型	参数规则	描述
groupNum	string	必选	集群群组号码
isSelect	bool	必选	开始或停止Select
			● true: 开始
			● false: 停止

## 返回值

#### **OnTalkingGroupSelectResult**

#### 使用示例

//c# code //调用接口Select集群群组, device为接口类IDevice实例 device.TalkingGroupSelect("9996300", true);

# 4.13.10 TalkingGroupPTT(抢权或放权)

## 接口描述

调度席可以通过该接口进行抢权或放权操作。如果调度席抢权成功,则可以进行抢权发言,此时呼叫为调度席到集群群组的半双工通话。

#### 注意事项

- 当前已经进行过集群群组功能初始化,即已成功登录UPortal。
- 调度席绑定话机需完成内置软电话模式下的话机初始化并激活辅助声卡。
- 调度席已经订阅指定集群群组。
- 调度席已经监听指定集群群组。
- 若TalkingGroup(集群群组信息)的receiveOnly字段为true,则调度席不可抢权。

## 方法定义

```
void TalkingGroupPTT(
string groupNum,
bool isPTT
```

## 参数描述

参数	类型	参数规则	描述
groupNum	string	必选	集群群组号码
isPTT	bool	必选	抢权还是放权
			● true: 抢权
			● false: 放权

## 返回值

#### OnTalkingGroupPTTResult

## 使用示例

```
//c# code
//调用接口对集群群组进行放权,device为接口类IDevice实例
device.TalkingGroupPTT("9996300", false);
```

# 4.13.11 TalkingGroupGetCallInfo(查询集群群组呼叫信息)

## 接口描述

调度席可以通过该接口查询集群群组呼叫信息和混合集群群组的呼叫信息,只有正在通话中的集群群组呼叫信息才能查询到。

调度席可以通过该接口获取当前集群群组监听呼叫或混合集群群组呼叫的呼叫唯一标识,以便进行调整该路呼叫的音量等后续操作。

#### 注意事项

- 当前已经进行过集群群组功能初始化,即已成功登录UPortal。
- 调度席已经订阅指定集群群组(混合集群群组不需要)。
- 调度席已经监听指定集群群组(混合集群群组不需要)或发起了混合集群群组呼叫。

#### 方法定义

```
void TalkingGroupGetCallInfo(
    string groupNum
```

#### 参数描述

参数	类型	参数规则	描述
groupNum	string	必选	集群群组号码

#### 返回值

#### $On Talking Group Get Call In fo Result {\tt .}$

## 使用示例

```
//c# code
//调用接口查询通话中的集群群组呼叫信息,device为接口类IDevice实例
device.TalkingGroupGetCallInfo("9996300");
```

# 4.13.12 MixedGroupCallStart(发起混编集群群组呼叫)

#### 接口描述

调度席发起临时混编集群群组呼叫,此时混编群组内的成员都会被邀请加入临时会议中,加入成功后,默认只能听不能说。在发起呼叫时可以邀请该临时混编集群群组之外用户加入。

## 注意事项

- 当前已经进行过集群群组功能初始化,即已成功**登录UPortal**。
- 调度席绑定话机需完成内置软电话模式下的话机初始化。

● 当前已经创建了临时混编集群群组。

#### 方法定义

```
void MixedGroupCallStart(
    string groupNum,
    List<MixedGroupUserState> userStateList
)
```

#### 参数描述

参数	类型	参数规则	描述
groupNum	string	必选	集群群组号码
userStateList	List <mixedgrou pUserState&gt;</mixedgrou 	可选	非该临时混编集群群组用户状 态列表

#### 返回值

#### $On Mixed Group Call Start Result {\tt \circ}$

#### 使用示例

```
//c# code
//创建混编集群群组用户状态列表
List<MixedGroupUserState> usersList= new List<MixedGroupUserState>();
//调用接口发起混编集群群组呼叫,不邀请组外用户,device为接口类IDevice实例
device.MixedGroupCallStart(~9996300~, usersList);
```

## 4.13.13 MixedGroupCallEnd(结束混编集群群组呼叫)

#### 接口描述

调度席可以结束临时混编集群群组呼叫,此时临时会议将终止,所有入会成员的呼叫将自动释放。

调用该接口成功后,除OnMixedGroupCallEndResult事件外,还将触发OnPhoneReleaseEvent事件。

## 注意事项

- 已经加载SDK。
- 当前已经进行过集群群组功能初始化,即已成功登录UPortal。
- 调度席绑定话机需完成内置软电话模式下的话机初始化。
- 已经发起混编集群群组呼叫。

```
void MixedGroupCallEnd(
string groupNum
```

参数	类型	参数规则	描述
groupNum	string	必选	集群群组号码

## 返回值

#### OnMixedGroupCallEndResult

#### 使用示例

//c# code //调用接口结束混编集群群组呼叫,device为接口类IDevice实例 device.MixedGroupCallEnd(~9996300~);

# 4.13.14 MixedGroupChangeUserState(修改混编集群群组用户状态)

#### 接口描述

调度席可以修改混编集群群组用户状态,以便在混编集群群组呼叫进行中时对会议成员进行操作,如批量邀请成员进入通话,或者批量移除人员,或者进行话权授予或回收等。

呼叫进行中,调度席也可以通过该接口临时批量邀请或者移除集群群组外成员。

## 注意事项

- 当前已经进行过集群群组功能初始化,即已成功登录UPortal。
- 调度席绑定话机需完成内置软电话模式下的话机初始化。
- 已经发起混编集群群组呼叫。
- 只能修改同一集群的多个用户的状态。

## 方法定义

```
void MixedGroupChangeUserState(
string groupNum,
List<MixedGroupUserState> userStateList
```

## 参数描述

参数	类型	参数规则	描述
groupNum	string	必选	集群群组号码
userStateList	List <mixedgroup UserState&gt;</mixedgroup 	可选	非该临时混编集群群组用户状态 列表

#### 返回值

On Mixed Group Change User State Result.

4 IDevice(设备服务基本接口)

## 使用示例

CS)

```
//c# code
//创建混编集群群组列表实例
List<MixedGroupUserState> users = new List<MixedGroupUserState>();

//将Json字符串转换类型对象实力
JavaScriptSerializer jss = new JavaScriptSerializer();

//添加集群群组用户信息
string user1 = "{\"userNumber\":\"512\",\"userType\":1,\"state\":\"notInOperation\"}";
users. Add(jss. Deserialize<MixedGroupUserState>(user1));

string user2 = "{\"userNumber\":\"514\",\"userType\":2,\"state\":\"notInOperation\"}";
users. Add(jss. Deserialize<MixedGroupUserState>(user2));

//调用接口修改对应用户状态,device为接口类IDevice实例
device. MixedGroupChangeUserState (frm. GroupId, users);
```

# 5 IDeviceEvents(设备服务回调事件)

#### 所属类

Namespace	Com.Huawei.Device.Contract
Syntax	public interface IDeviceEvents

#### □ 说明

不同资源的事件,没有必然的事件上报先后顺序,上层业务不应依赖不同资源的事件上报顺序来 处理业务逻辑。

- 5.1 IDeviceEvents Methods: 运行管理类事件
- 5.2 IDeviceEvents Methods: 语音接续类事件
- 5.3 IDeviceEvents Methods: 状态管理类事件
- 5.4 IDeviceEvents Methods: 其他查询类事件
- 5.5 IDeviceEvents Methods: 电话会议类事件
- 5.6 IDeviceEvents Methods: 座席管理类事件
- 5.7 IDeviceEvents Methods: 录音管理类事件
- 5.8 IDeviceEvents Methods: 无线资源类事件
- 5.9 IDeviceEvents Methods: 短信服务类事件
- 5.10 IDeviceEvents Methods: 邮件服务类事件
- 5.11 IDeviceEvents Methods: 视频通话类事件
- 5.12 IDeviceEvents Methods: 话机管理类事件
- 5.13 IDeviceEvents Methods: 集群调类事件

# 5.1 IDeviceEvents Methods: 运行管理类事件

# 5.1.1 OnInitializeResult(启动软电话结果)

## 事件描述

座席调用启动软电话接口后,上报的结果事件。

## 方法定义

```
void OnInitializeResult(
    string callId,
    AgentInfo agentInfo,
    Dictionary<string, string> settings,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
agentInfo	AgentInfo	座席基本信息
settings	Dictionary <string, string=""></string,>	PBX配置信息
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.1.2 OnUnInitializeResult(关闭软电话结果)

## 事件描述

座席调用关闭软电话接口后,上报的结果事件。

```
void OnUnInitializeResult(
    string callId,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.1.3 OnLoginToUPortalResult(登录 UPortal 结果事件)

## 事件描述

座席调用登录UPortal接口后,上报的结果事件。

#### 方法定义

```
void OnLoginToUPortalResult(
    string callId,
    bool flag,
    int errorCode
)
```

## 参数描述

参数	类型	描述	
callId	string	接口调用序号	
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败	
errorCode	int	错误码	

# 5.1.4 OnLogoutFromUPortal(登出 UPortal 结果事件)

## 事件描述

座席调用登出UPortal接口后,上报的结果事件。

```
void OnLogoutfromUPortalResult (
string callId,
```

eSDK ICP V200R001C10 接口参考 01(ECC, Windows,

bool flag, int errorCode

#### 参数描述

CS)

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.1.5 OnDisConnection(设备掉线)

#### 事件描述

客户端和CTI平台断连通知。此回调事件已经废弃,不建议调用。

## 使用说明

设备掉线后,需要通知上层应用。

## 方法定义

```
void OnDisConnection(
string callId,
DeviceModule mode
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
mode	DeviceModule	设备类型

# 5.1.6 OnConnection(设备重连成功)

#### 事件描述

客户端和CTI平台重连成功通知,该事件由设备侧主动推送至上层应用。在设备掉线后,如果设备重连成功后,需要通知上层。

此回调事件已经废弃,不建议使用。

## 方法定义

```
void OnConnection(
string callId,
DeviceModule mode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
mode	DeviceModule	设备类型

# 5.1.7 OnError(系统异常)

#### 事件描述

系统异常通知,该事件由设备侧主动推送至上层应用。

## 方法定义

```
void OnError(
    string callId,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
errorCode	int	错误码

# 5.1.8 OnUnInitializeAllResult(清空所有资源结果事件)

## 事件描述

座席调用清空所有资源接口后,上报的结果事件。

```
void OnUnInitializeAllResult(
    string callId,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.2 IDeviceEvents Methods: 语音接续类事件

# 5.2.1 OnNewCall(新来电信息)

#### 事件描述

新来电信息通知,该事件由设备侧主动推送至上层应用。

## 方法定义

```
void OnNewCall(
   string callId,
   DeviceType callerType,
   string callerNumber,
   DeviceType calleeType,
   string calleeNumber
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
callerType	DeviceType	主叫设备类型
callerNumber	string	主叫号码
calleeType	DeviceType	被叫设备类型
calleeNumber	string	被叫号码

# 5.2.2 OnAnswerCallResult(应答结果信息)

#### 事件描述

座席话机应答来电后, 上报的结果事件。

## 方法定义

```
void OnAnswerCallResult(
string callId,
DialContent dialContent,
bool flag,
int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
dialContent	DialContent	电话信息
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.2.3 OnHangUpResult(挂机返回)

## 事件描述

座席话机主动挂断通话后, 上报的结果事件

## 方法定义

```
void OnHangUpResult(
string callId,
bool flag,
int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

eSDK ICP V200R001C10 接口参考 01(ECC, Windows,

# 5.2.4 OnOtherHangup(对方挂机事件)

#### 事件描述

对方挂机通知,该事件由设备侧主动推送至上层应用。

如果主动挂机座席侧在监听、强插场景中挂断通话时,则被质检座席侧不会触发该事件,而是触发IDeviceEvents.OnAdvancedOtherHangUp。

## 方法定义

```
void OnOtherHangup(
string callId,
DialContent dialContent
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
dialContent	DialContent	对方呼叫信息

# 5.2.5 OnAdvancedOtherHangUp(对方挂机事件)

## 事件描述

对方挂机通知,该事件由设备侧主动推送至上层应用。只有主动挂机侧座席在监听、 强插场景中挂断通话时,才会在被质检座席侧会触发该事件。

## 方法定义

```
void OnAdvancedOtherHangUp(
string callId,
OpStatus phoneStatus,
DialContent dialContent)
```

参数	类型	描述
callId	string	接口调用序号
phoneStatus	OpStatus	对方座席状态
dialContent	DialContent	对方呼叫信息

# 5.2.6 OnDialResult(单呼返回)

#### 事件描述

座席调用 单呼接口后,上报的结果事件。

## 方法定义

```
void OnDialResult(
string callId,
DialContent dialContent,
bool flag,
int errorCode
)
```

## 参数描述

参数	类型	描述
callId	string	接口调用序号
dialContent	DialContent	呼叫信息
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.2.7 OnHoldResult(保持结果)

## 事件描述

座席调用保持接口后,上报的结果事件。

## 方法定义

```
void OnHoldResult(
string callId,
HoldMode holdMode,
bool flag,
int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号

参数	类型	描述
holdMode	HoldMode	保持模式
		● Hold: 保持
		● HoldCancel: 取消保持
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.2.8 OnConsultCallResult(协商呼叫结果)

## 事件描述

座席调用协商呼叫接口后,上报的结果事件。

## 方法定义

```
void OnConsultCallResult(
string callId,
DialContent dialContent,
bool flag,
int errorCode
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
dialContent	DialContent	呼叫信息
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.2.9 OnEndConsultCallResult(结束协商呼叫结果)

## 事件描述

座席调用结束协商呼叫接口后,上报的结果事件。

## 方法定义

```
void OnEndConsultCallResult(
   string callId,
   bool flag,
   int errorCode
)
```

## 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.2.10 OnTransferToAgentResult(协商转移结果)

## 事件描述

座席调用协商转移接口后,上报的结果事件。

## 方法定义

```
void OnTransferToAgentResult(
    string callId,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识 ● true: 成功 ● false: 失败
errorCode	int	错误码

# 5.2.11 OnSingleStepTransferResult(非监督转移结果)

## 事件描述

座席调用**非监督转移**接口后,上报的结果事件。

## 方法定义

```
void OnSingleStepTransferResult(
    string callId,
    string sourceAgent,
    string targetNumber,
    bool flag,
    int iErrorCode
)
```

## 参数描述

参数	类型	描述	
callId	string	接口调用序号	
sourceAgent	string	发起转移的座席工号	
targetNumber	string	转移目标的电话号码	
flag	bool 接口调用结果标识		
		● true: 成功	
		● false: 失败	
iErrorCode	int	错误码	

# 5.2.12 OnSingleStepSuccTransferResult(监督转移结果)

## 事件描述

座席调用监督转移接口后,上报的结果事件。

## 方法定义

```
void OnSingleStepSuccTransferResult(
   string callId,
   string sourceAgent,
   string targetNumber,
   bool flag,
   int iErrorCode
)
```

参数	类型	描述
callId	string	接口调用序号

参数	类型	描述
sourceAgent	string	发起转移的座席工号
targetNumber	string	转移目标的电话号码
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
iErrorCode	int	错误码

# 5.2.13 OnTransferInform(呼叫转移信息)

### 事件描述

呼叫转移时,转移目标方收到系统返回的转移信息事件。该事件由设备侧主动推送至上层应用。

#### 方法定义

```
void OnTransferInform(
   string callId,
   string dAgentId,
   string sAgentId,
   string nCallId,
   string oCallId,
   DeviceType callType,
   string callNumber,
   bool isOutPhone
```

参数	类型	描述
callId	string	接口调用序号
dAgentId	string	目标座席工号
sAgentId	string	原座席工号
nCallId	string	新呼叫标识
oCallId	string	原呼叫标识
callType	DeviceType	主叫类型
callNumber	string	主叫号码
isOutPhone	bool	是否是外线电话
		● true: 是
		● false: 不是

# 5.2.14 OnQueryCurrentCallingScopeResult(查询当前呼叫范围结果)

#### 事件描述

座席调用查询当前呼叫范围接口后,上报的结果事件。

#### 方法定义

```
void OnQueryCurrentCallingScopeResult(
    string callId,
    CallingPhoneScope phoneScope,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	bool	接口调用序号
phoneScope	CallingPhoneScope	呼入电话范围
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.2.15 OnPollingCallResult(轮询呼叫结果)

### 事件描述

座席调用轮询呼叫接口后,上报的结果事件。

## 方法定义

```
void OnPollingCallResult(
    string callId,
    DialContent dialContent,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
dialContent	DialContent	呼叫信息

eSDK ICP V200R001C10	接口参考 01(ECC, Windows,
----------------------	-----------------------

参数	类型	描述
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.2.16 OnBeTransfer(呼叫被转移事件)

#### 事件描述

座席的通话被对端座席转移给第三方座席时,本端座席收到系统返回的事件。该事件由系统主动向上层应用返回。

这个回调事件已废弃,不推荐使用。

#### 方法定义

```
void OnBeTransfer(
string callId,
AgentInfo sourceAgent,
AgentInfo targetAgent)
```

### 参数描述

参数	类型	描述
callId	string	接口调用序号
sourceAgent	AgentInfo	转出座席信息
targetAgent	AgentInfo	转入座席信息

# 5.2.17 OnPhoneAlerting(座席电话振铃通知事件)

### 事件描述

座席电话振铃信息通知,该事件由设备侧主动推送至上层应用。

```
void OnPhoneAlerting(
   string callId,
   DeviceType callerType,
   string callerNumber,
   DeviceType calleeType,
   string calleeNumber
)
```

参数	类型	描述
callId	string	接口调用序号
callerType	DeviceType	主叫设备类型
callerNumber	string	主叫号码
calleeType	DeviceType	被叫设备类型
calleeNumber	string	被叫号码

# 5.2.18 OnPhoneAlertOtherHungup(座席电话振铃时对方挂机事件)

#### 事件描述

座席侧电话振铃时对方挂机事件,该事件由设备侧主动推送至上层应用。

#### 方法定义

```
void OnPhoneAlertOtherHungup(
string callId,
DialContent dialContent
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
dialContent	DialContent	对方呼叫信息

# 5.2.19 OnPhoneHungup(座席电话挂机事件)

### 事件描述

座席侧电话主动挂机后返回事件,该事件由设备侧主动推送至上层应用。

```
void OnPhoneHungup (
string callId,
DialContent dialContent
)
```

eSDK ICP V200R001C10 接口参考 01(ECC, Windows,

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
dialContent	DialContent	对方呼叫信息

# 5.2.20 OnDisconnectCallResult(根据 CallId 释放呼叫返回结果)

### 事件描述

座席调用根据CallId释放呼叫接口后,上报的结果事件。

#### 方法定义

```
void OnDisconnectCallResult(
   string callId,
   bool flag,
   int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.2.21 OnTransferToSkillGroupResult(通话转移队列结果)

### 事件描述

座席调用通话转移队列接口后,上报的结果事件。

```
void OnTransferToSkillGroupResult(
    string callId,
   bool flag,
   int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.2.22 OnDialExResult(单呼拓展结果)

#### 事件描述

座席调用 单呼拓展接口后,上报的结果事件。

#### 方法定义

```
void OnDialExResult(
    string callId,
    DialContent dialContent,
    string accessCode,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
dialContent	DialContent	呼叫信息
accessCode	string	系统接入码
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.2.23 OnPollingCallExResult(轮询呼叫拓展结果)

#### 事件描述

座席调用轮询呼叫拓展接口后,上报的结果事件。

```
void OnPollingCallExResult(
    string callId,
    DialContent dialContent,
    string accessCode,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
dialContent	DialContent	呼叫信息
accessCode	string	系统接入码
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.2.24 OnMuteResult(静音结果)

### 事件描述

座席调用静音接口后,上报的结果事件。

### 方法定义

```
void OnMuteResult(
   string callId,
   bool isMute,
   bool flag,
   int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
isMute	bool	静音标识
		● true: 静音。
		● false: 取消静音

参数	类型	描述
flag	bool	接口调用结果标识 ● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.2.25 OnAgentSendDTMFResult(二次拨号结果)

#### 事件描述

座席调用二次拨号接口后,上报的结果事件。

#### 方法定义

```
void OnAgentSendDTMFResult(
   string callId,
   bool flag,
   int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.2.26 OnLongNoAnswer(久不应答通知事件)

### 事件描述

座席侧对来电久不应答后上报该事件,该事件由设备侧主动推送至上层应用。

```
void OnLongNoAnswer(
string callId,
DialContent dialContent
)
```

参数	类型	描述
callId	string	接口调用序号
dialContent	DialContent	对方呼叫信息

# 5.2.27 OnPhoneBusy(座席话机无法应答通知)

#### 事件描述

座席侧话机无法应答通知事件,该事件由设备侧主动推送至上层应用。

#### 方法定义

```
void OnPhoneBusy(
string callId,
DialContent dialContent
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
dialContent	DialContent	对方呼叫信息

# 5.2.28 OnPhoneOffhook(座席话机摘机事件)

#### 事件描述

座席话机摘机后返回该事件,该事件由设备侧主动推送至上层应用。

#### 方法定义

void OnPhoneOffHook()

## 参数描述

无。

# 5.2.29 OnTransferToIVRResult(转到 IVR 结果)

#### 事件描述

座席调用转到IVR接口后,上报的结果事件。

```
void OnTransferToIVRResult(
string callid,
bool flag,
int errorCode
)
```

#### 参数描述

参数	类型	描述
callid	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.2.30 OnReturnFromIvr(呼叫从 IVR 返回)

### 事件描述

座席直接调用或者以挂起转方式调用**TransToIVR**成功后,座席被挂起并等待呼叫从IVR返回。当呼叫从IVR返回时触发此事件,此时座席话路已经接通。

#### 方法定义

void OnReturnFromIvr()

#### 参数描述

无。

# 5.2.31 OnDummyCallArrive(早释电话通知)

### 事件描述

早释电话通知到座席时, 上报的结果事件。

```
void OnDummyCallArrive(
string callId,
string CallNo,
string dummyCallData
)
```

参数	类型	描述
callId	string	接口调用序号
CallNo	string	主叫号码
dummyCallData	string	虚拟呼叫数据

# 5.2.32 OnLinkageResultNotify(启动联动/停止联动结果)

### 事件描述

启动联动或停止联动时触发此事件。

#### 方法定义

```
void OnLinkageResultNotify(
    string callId,
    int result
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
result	int	启动联动或停止联动结果  ● 0: 启动联动成功  ● 1: 启动联动失败  ● 2: 停止联动成功  ● 3: 停止联动失败  ● 4: 初始值

# 5.2.33 OnLinkagePhoneStatus(联动话机状态改变通知)

## 事件描述

当联动的话机状态改变的时候座席收到该通知事件

```
void OnLinkagePhoneStatus(
int status
)
```

参数	类型	描述
status	int	联动话机状态
		● 0:表示话机摘机
		● 1:表示话机挂机
		● 2:表示话机通话中
		● 3:表示话机上线态
		● 4:表示示话机离线
		● 5:表示话机振铃中

# 5.2.34 OnLinkageCalloutNotify(联动话机呼出通知)

### 事件描述

当联动的话机主动呼出的时候座席收到该通知事件。

#### 方法定义

```
void OnLinkageCalloutNotify(
    int ulTime,
    int usDsn,
    int ucHandle,
    int ucServer,
    int iCallerType,
    int iCalleeType,
    string Caller,
    string Called
)
```

参数	类型	描述
ulTime	int	予保留,未使用
usDsn	int	予保留,未使用
ucHandle	int	予保留,未使用
ucServer	int	予保留,未使用
iCallerType	int	予保留,未使用
iCalleeType	int	予保留,未使用
Caller	string	主叫号码
Called	string	被叫号码

# 5.2.35 OnHoldByCallIDResult(保持指定通话结果)

#### 事件描述

座席调用保持指定通话接口后,上报的结果事件。

#### 方法定义

```
void OnHoldByCallIDResult(
string callID,
HoldMode holdMode,
bool flag,
int Errorcode
)
```

#### 参数描述

参数	类型	描述
CallID	string	接口调用序号
holdMode	HoldMode	保持模式
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.3 IDeviceEvents Methods: 状态管理类事件

# 5.3.1 OnSetBusyResult(置忙返回)

#### 事件描述

座席调用置忙接口后,上报的结果事件。

```
void OnSetBusyResult(
    string callId,
    OpStatus status,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
status	OpStatus	座席状态
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.3.2 OnSetIdleResult(置闲返回)

### 事件描述

座席调用置闲接口后,上报的结果事件。

### 方法定义

```
void OnSetIdleResult(
    string callId,
    OpStatus status,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
status	OpStatus	座席状态
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.3.3 OnAgentStatusChange(座席状态改变信息)

#### 事件描述

座席状态信息改变时,主动向上层应用推送该事件。

```
void OnAgentStatusChange(
string callId,
AgentInfo agentInfo
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
agentInfo	AgentInfo	座席信息

# 5.3.4 OnQueryAgentStatusResult(查询座席状态结果)

### 事件描述

座席调用查询座席状态接口后,上报的结果事件。

#### 方法定义

```
void OnQueryAgentStatusResult(
    string callId,
    AgentInfo agentInfo,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
agentInfo	AgentInfo	座席信息
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.3.5 OnQueryAllAgentStatusResult(查询组内所有座席状态结果)

### 事件描述

座席调用查询组内所有座席状态接口后,上报的结果事件。

```
void OnQueryAllAgentStatusResult(
    string callId,
    List<AgentInfo> agentInfos,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
agentInfos	List <agentinfo></agentinfo>	座席信息集合
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.3.6 OnCallListChange(组内呼叫队列状态改变信息)

### 事件描述

同组呼叫队列状态信息改变时,该事件由设备侧主动向上层应用推送。

## 方法定义

```
void OnCallListChange(
string callId,
CallListDetailsChanged callList
)
```

### 参数描述

参数	类型	描述
callId	string	接口调用序号
callList	CallListDetailsChanged	呼叫队列信息

# 5.3.7 OnRestResult(休息返回)

#### 事件描述

座席调用休息接口后,上报的结果事件。

```
void OnRestResult(
    string callId,
    OpStatus status,
    bool flag,
    int errorCode
)
```

### 参数描述

参数	类型	描述
callId	string	接口调用序号
status	OpStatus	座席状态
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.3.8 OnCancelRestResult(取消休息返回)

### 事件描述

座席调用取消休息接口后,上报的结果事件。

## 方法定义

```
void OnCancelRestResult(
    string callId,
    OpStatus status,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
status	OpStatus	座席状态
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.3.9 OnRestTimeOut(休息超时提醒)

#### 事件描述

座席休息超时后, 主动返回该事件到上层应用。

#### 方法定义

```
void OnRestTimeOut(
    string callId,
    int restTime
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
restTime	int	座席请求休息时长,单位为秒

# 5.3.10 OnExitACWResult(退出整理态结果)

### 事件描述

座席调用退出整理态接口后,上报的结果事件。

## 方法定义

```
void OnExitACWResult(
string callId,
bool flag,
int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.3.11 OnMeetingPTTStatusChange(座席会议 PTT 状态改变信息)

### 事件描述

会议中当前座席PTT状态(按下/抬起)发生改变时的通知事件,该事件由系统主动返回给上层应用。

#### 方法定义

```
void OnMeetingPTTStatusChange(
    string callId,
    string number,
    string agentId,
    PTTOperate pttOperate
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
number	string	PTT状态改变的会议成员电话号码
agentId	string	收到该事件的当前座席工号
pttOperate	PTTOperate	PTT操作类型
		● PutUp: PTT抬起
		● PutDown: PTT按下

# 5.3.12 OnCallOutStatusChange(呼出状态返回)

#### 事件描述

座席发起呼叫时,系统主动向主叫侧返回被叫侧呼叫状态信息。

#### 方法定义

```
void OnCallOutStatusChange(
DialContent dialContent,
OpStatus status)
```

参数	类型	描述
dialContent	DialContent	被叫呼叫信息
status	OpStatus	被叫状态

# 5.3.13 OnSetAutoEnterIdleResult(设置呼叫后是否进入示闲状态结果返回)

#### 事件描述

座席调用设置呼叫后是否进入示闲状态接口后,上报的结果事件。

#### 方法定义

```
void OnSetAutoEnterIdleResult(
    string callId,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识 <ul><li>true: 成功</li><li>false: 失败</li></ul>
errorCode	int	错误码

# 5.3.14 OnSetWorkStopResult(设置是否进入关班状态结果返回)

#### 事件描述

座席调用设置是否进入关班状态接口后,上报的结果事件。

#### 方法定义

```
void OnSetWorkStopResult(
    string callId,
    bool bWork,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
bWork	bool	开关班状态
		● true: 关班
		● false: 开班

参数	类型	描述
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.3.15 OnQueryRestReasonsResult(查询 VDN 休息原因返回结果)

#### 事件描述

座席调用**查询VDN休息原因**接口后,上报的结果事件。

#### 方法定义

```
void OnQueryRestReasonsResult(
    string callId,
    List<RestReason> listRestReason,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
listRestReason	List <restreason></restreason>	VDN休息原因集合
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.3.16 OnQuerySkillGroupStatusResult(查询呼叫队列状态返回)

#### 事件描述

座席调用查询呼叫队列状态接口后,上报的结果事件。

```
void OnQuerySkillGroupStatusResult(
string callId,
string groupId,
SkillGroupStatus skillGroupStatus,
bool flag,
```

```
int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
groupId	string	技能队列ID
skillGroupStatus	SkillGroupStatus	技能队列状态
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.3.17 OnSetConfigResult(设置配置项结果)

### 事件描述

座席调用**设置配置项**接口后,上报的结果事件。

### 方法定义

```
Void OnSetConfigResult(
string callid,
string cfgKey,
string cfgValue,
bool flag,
int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
cfgKey	string	配置项key
cfgValue	string	配置项值
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.3.18 OnGetConfigResult(查询配置项结果)

#### 事件描述

座席调用查询配置项接口后,上报的结果事件。

#### 方法定义

```
Void OnGetConfigResult(
string callid,
string cfgKey,
string cfgValue,
bool flag,
int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
cfgKey	string	配置项key
cfgValue	string	配置项值
flag	bool	接口调用结果标识  ● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.4 IDeviceEvents Methods: 其他查询类事件

# 5.4.1 OnQueryIVRInfoResult(查询 IVR 结果)

### 事件描述

座席调用查询IVR接口后,上报的结果事件。

```
void OnQueryIVRInfoResult(
   List<IvrInfo> IvrInfos,
   string callid,
   bool flag,
   int errorCode
)
```

参数	类型	描述
IvrInfos	List <ivrinfo></ivrinfo>	IVR信息的集合
callid	string	呼叫标识
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.4.2 OnQueryCallListByGroupIdResult(查询组内呼叫队列结果)

### 事件描述

座席调用查询组内呼叫队列接口后,上报的结果事件。

#### 方法定义

```
void OnQueryCallListByGroupIdResult(
    string callId,
    List<CallListDetails> callListDetails,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
callListDetails	List <calllistdetails></calllistdetails>	排队电话列表
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.4.3 OnQueryBlackListResult(查询是否是黑名单结果)

### 事件描述

座席调用查询是否是黑名单接口后,上报的结果事件。

```
void OnQueryBlackListResult(
    string callId,
    string phoneNumber,
    DateTime createTime,
    int Locktime,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
phoneNumber	string	电话号码
createTime	DateTime	创建时间
Locktime	int	锁定时长
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.4.4 OnQueryBlackListExResult(查询是否是黑名单结果扩展)

#### 事件描述

座席调用查询是否是黑名单扩展接口后,上报的结果事件。

## 方法定义

```
void OnQueryBlackListExResult(
    string callId,
    string phoneNumber,
    DateTime createTime,
    int Locktime,
    short filterMode,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
phoneNumber	string	电话号码

eSDK ICP V200R001C10 接口参考 01(ECC, Windows,	
CS)	

参数	类型	描述
createTime	DateTime	创建时间
Locktime	int	锁定时长
filterMode	short	<ul><li>黑名单模式</li><li>● 1: 拒绝</li><li>● 2: 降级</li></ul>
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.4.5 OnQueryAllBlackListResult(查询所有黑名单结果)

### 事件描述

座席调用**查询所有黑名单**接口后,上报的结果事件。

#### 方法定义

```
void OnQueryAllBlackListResult(
    string callId,
    List<BlackList> blackList,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
blackList	List <blacklist></blacklist>	黑名单列表
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.4.6 OnQueryAllBlackListExResult(查询所有黑名单结果扩展)

#### 事件描述

座席调用查询所有黑名单扩展接口后,上报的结果事件。

#### 方法定义

```
void OnQueryAllBlackListExResult(
    string callId,
    List<BlackList> blackList,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
blackList	List <blacklist></blacklist>	黑名单列表
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.4.7 OnQueryRedListResult(查询是否是优先级名单结果)

### 事件描述

座席调用查询是否是优先级名单接口后,上报的结果事件。

#### 方法定义

```
void OnQueryRedListResult(
    string callId,
    string phoneNumber,
    int priority,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
phoneNumber	string	电话号码

参数	类型	描述
priority	int	优先级别,数值越大级别越高
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.4.8 OnQueryAllRedListResult(查询所有优先级名单结果)

### 事件描述

座席调用查询所有优先级名单接口后,上报的结果事件。

#### 方法定义

```
void OnQueryAllRedListResult(
    string callId,
    List<RedList> redList,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
redList	List <redlist></redlist>	优先级名单列表
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.4.9 OnQuerySkillGroupsResult(查询系统的技能队列结果)

#### 事件描述

座席调用**查询系统的技能队列**接口后,上报的结果事件。

### 方法定义

void OnQuerySkillGroupsResult(
 string callId,

```
List<SkillGroup> SkillGroups,
bool flag,
int errorCode
```

参数	类型	描述
callId	string	接口调用序号
SkillGroups	List< SkillGroup>	技能列表
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.4.10 OnQueryAccessCodeListResult(查询所有系统接入码配置信息返回)

### 事件描述

座席调用**查询所有系统接入码配置信息**接口后,上报的结果事件。

### 方法定义

```
void OnQueryAccessCodeListResult(
    string callId,
    List<AccessCode> accessCodeList,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
accessCodeList	List< AccessCode>	系统接入码配置信息列表
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.4.11 OnQueryLostCallsResult(查询丢话记录结果)

#### 事件描述

座席调用查询丢话记录接口后,上报的结果事件。

#### 方法定义

```
void OnQueryLostCallsResult(
    string callId,
    List<LostCall> lostCalls,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
lostCalls	List <lostcall></lostcall>	丢话记录列表
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.4.12 OnQueryLostCallsCountResult(查询丢话记录数量结果)

### 事件描述

座席调用查询丢话记录数量接口后,上报的结果事件。

### 方法定义

```
void OnQueryLostCallsCountResult(
    string callId,
    int callsCount,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
callsCount	int	丢话记录数量

参数	类型	描述
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.4.13 OnStopQueryCallListByGroupIdResult(停止查询呼叫队列结果)

#### 事件描述

座席调用停止查询呼叫队列接口后,上报的结果事件。

#### 方法定义

```
void OnStopQueryCallListByGroupIdResult(
    string callId,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.4.14 OnQueryCallInfoResult(查询当前通话信息结果)

#### 事件描述

座席调用查询当前通话信息接口后,上报的结果事件。

```
void OnQueryCallInfoResult(
    string callId,
    List<AgentCallInfo> agentCallInfos,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
agentCallInfos	List <agentcallinfo></agentcallinfo>	当前通话信息集合
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.4.15 OnQueryAgentSkillGroupsResult(查询座席技能信息结果返回)

#### 事件描述

座席调用**查询座席技能信息**接口后,上报的结果事件。

#### 方法定义

```
void OnQueryAgentSkillGroupsResult(
    string callId,
    List<AgentSkillGroup> AgentSkillGroups,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
AgentSkillGroups	List <agentskillgroup></agentskillgroup>	座席配置的技能队列信息列表
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.4.16 OnQueryMaliciousCallsResult(查询骚扰记录结果返回)

#### 事件描述

座席调用查询骚扰记录接口后,上报的结果事件。

```
void OnQueryMaliciousCallsResult(
string callId,
List<MaliciousCall> maliciousCalls,
bool flag,
int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
maliciousCalls	List <maliciouscall></maliciouscall>	查询到的骚扰记录集合
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.5 IDeviceEvents Methods: 电话会议类事件

# 5.5.1 OnCreateMeetingResult(创建会议返回结果事件)

### 事件描述

创建会议返回结果事件。该事件在调用创建会议后返回,后续成员状态修改再通过 IDeviceEvent.OnYieldAnswerCall来实现

## 方法定义

```
void OnCreateMeetingResult(
    string callId,
    string meetingId,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
meetingId	string	会议标识

eSDK ICP V200R001C10 接口参考 01(ECC, Windows,
CS)

参数	类型	描述
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.5.2 OnYieldAnswerCall(创建会议呼叫应答返回事件)

### 事件描述

创建会议后每个呼叫对象(会议成员)应答返回事件。该事件由设备侧主动推送至上 层应用。

#### 方法定义

```
void OnYieldAnswerCall(
    string callId,
    string meetingId,
    DialContent dialContent,
    bool flag,
    int errorCode
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
meetingId	string	会议标识
dialContent	DialContent	当前单呼对象信息
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.5.3 OnMakeMeetingResult(形成会议结果)

### 事件描述

座席调用形成会议接口后,上报的结果事件。

## 方法定义

void OnMakeMeetingResult( string callId,

```
string meetingId,
bool flag,
int errorCode
```

参数	类型	描述
callId	string	接口调用序号
meetingId	string	会议标识
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.5.4 OnSingleStepMeetingResult(单步会议结果)

### 事件描述

座席调用单步会议接口后,上报的结果事件。

### 方法定义

```
void OnSingleStepMeetingResult(
   string callId,
   string mettingId,
   bool flag,
   int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
mettingId	string	会议标识
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.5.5 OnInviteMeetingMemberResult(邀请会议成员返回)

#### 事件描述

座席调用 邀请会议成员接口后,上报的结果事件。

#### 方法定义

```
void OnInviteMeetingMemberResult(
   string callId,
   string meetingId,
   DialContent dialContent,
   bool flag,
   int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
meetingId	string	会议标识
dialContent	DialContent	会议成员信息
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.5.6 OnInvitedMeeting(被邀请加入会议返回)

### 事件描述

成员被邀请加入会议时,设备侧主动向被邀请座席侧返回该事件。

## 方法定义

```
void OnInvitedMeeting(
   string callId,
   string meetingId,
   DialContent dialContent,
   bool flag,
   int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号

参数	类型	描述
meetingId	string	会议标识
dialContent	DialContent	当前要加入会议的成员信息
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.5.7 OnRemoveMeetingMemberResult(移除会议成员结果)

#### 事件描述

座席调用移除会议成员接口后,上报的结果事件。

#### 方法定义

```
void OnRemoveMeetingMemberResult(
    string callId,
    string meetingId,
    DialContent dialContent,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
meetingId	string	会议标识
dialContent	DialContent	被移除的会议成员信息
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.5.8 OnLeaveMeeting(座席离开会议时返回)

#### 事件描述

座席离开会议时,设备侧主动向上层应用返回该事件。

```
void OnLeaveMeeting(
   string callId,
   string meetingId,
   DialContent dialContent
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
meetingId	string	会议标识
dialContent	DialContent	离开会议的成员信息

# 5.5.9 OnExitMeetingResult(退出会议结果)

### 事件描述

座席调用退出会议接口后,上报的结果事件。

#### 方法定义

```
void OnExitMeetingResult(
    string callId,
    string meetingId,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
meetingId	string	会议标识
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.5.10 OnMeetingMemberExit(成员退出会议事件)

#### 事件描述

会议中有成员退出会议时,系统对其它成员座席主动返回该事件。

```
void OnMeetingMemberExit(
    string callId,
    string meetingId,
    DialContent dialContent
)
```

### 参数描述

参数	类型	描述
callId	String	接口调用序号
meetingId	String	会议标识
dialContent	DialContent	退出会议的成员信息

# 5.5.11 OnTransferMeetingMasterResult(转移会议主席结果)

### 事件描述

座席调用**转移会议主席**接口后,上报的结果事件。

#### 方法定义

```
void OnTransferMeetingMasterResult(
    string callId,
    string meetingId,
    string oldMasterNumber,
    string newMasterNumber,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
meetingId	string	会议标识
oldMasterNumber	string	原会议主席电话号码
newMasterNumber	string	新会议主席电话号码
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.5.12 OnBeTransferMaster(被任命为会议主席)

#### 事件描述

会议成员被任命为新会议主席时,设备侧主动向上层应用返回该事件。

#### 方法定义

```
void OnBeTransferMaster(
    string callId,
    string meetingId,
    DialContent oldMaster,
    DialContent newMaster,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
meetingId	string	会议标识
oldMaster	DialContent	旧会议主席呼叫信息
newMaster	DialContent	新会议主席呼叫信息
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

### 5.5.13 OnEndMeetingResult(结束会议结果)

#### 事件描述

座席调用结束会议接口后,上报的结果事件。

```
void OnEndMeetingResult(
    string callId,
    string meetingId,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
meetingId	string	会议标识
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.5.14 OnQueryMeetingMembersResult(查询会议成员结果)

#### 事件描述

座席调用查询会议成员接口后,上报的结果事件。

#### 方法定义

```
void OnQueryMeetingMembersResult(
    string callId,
    string meetingId,
    string chairManPhone,
    List<DialContent> members,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
meetingId	string	会议标识
chairManPhone	string	会议主席电话号码
members	List <dialcontent></dialcontent>	会议成员呼叫信息列表
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.5.15 OnMeetingMemberAlerting(会议中邀请用户状态反馈)

#### 事件描述

会议中邀请用户状态反馈。此接口已经废弃,不建议使用。

#### 方法定义

```
void OnMeetingMemberAlerting (
string callId,
string meetingId,
DialContent dialContent
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
meetingId	string	会议标识
dialContent	DialContent	被叫电话内容

# 5.5.16 OnMeetingMemberJoin(新会议成员加入事件)

### 事件描述

会议中有成员加入会议时,系统对其它成员座席主动返回该事件。

### 方法定义

```
void OnMeetingMemberJoin (
string callId,
string meetingId,
DialContent dialContent
)
```

参数	类型	描述
callId	string	接口调用序号
meetingId	string	会议标识
dialContent	DialContent	加入会议的成员信息

### 5.5.17 OnCreateMeetingExResult(创建会议拓展)

#### 事件描述

创建会议拓展返回,该事件在调用创建会议拓展后返回,后续成员状态修改再通过 IDeviceEvent.OnYieldAnswerCall来实现

#### 方法定义

```
void OnCreateMeetingExResult(
    string callId,
    string meetingId,
    string accessCode,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
meetingId	string	会议标识
accessCode	string	系统接入码
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.5.18 OnChangeMeetingMemberModeResult(修改与会者属性结果)

### 事件描述

座席调用修改与会者属性接口后,上报的结果事件。

```
void OnChangeMeetingMemberModeResult(
    string callId,
    string meetingId,
    DialContent dialContent,
    int attribute,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
meetingId	string	会议标识
dialContent	DialContent	被修改的会议成员信息
attribute	int	被修改的与会者属性  ● 1: 只能听,不能说  ● 2: 只能说,不能听  ● 3: 既能听,也能说  ● 4: 不能听,不能说
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.5.19 OnMeetingMemberModeChange(与会者属性变更通知事件)

### 事件描述

会议中有成员属性变更时,系统对其它成员座席主动返回该事件。

### 方法定义

```
OnMeetingMemberModeChange(
string callId,
string meetingId,
DialContent dialContent,
int attribute
```

参数	类型	描述
callId	string	接口调用序号
meetingId	string	会议标识
DialContent	DialContent	属性变更的成员信息

eSDK ICP V200R001C10 接口参考	7 01(ECC,	Windows
CS)		

参数	类型	描述
attribute	int	与会者属性
		● 1: 只能听,不能说。
		● 2: 只能说,不能听。
		● 3: 既能听,也能说。
		● 4: 不能听,不能说。

# 5.5.20 OnMeetingMasterChange(会议主席变更通知事件)

#### 事件描述

会议中会议主席发生变更时,系统对其它成员座席主动返回该事件。

#### 方法定义

```
void OnMeetingMasterChange(
   string callId,
    string meetingId,
    DialContent newMaster
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
meetingId	string	会议标识
newMaster	DialContent	新会议主席信息

# 5.6 IDeviceEvents Methods: 座席管理类事件

# 5.6.1 OnBargeResult(强插结果)

#### 事件描述

质检座席调用强插接口强插普通座席时,质检座席侧主动上报该事件。

```
void OnBargeResult(
    string callId,
    AgentInfo agentInfo,
    bool flag,
    int\ error Code
```

参数	类型	描述
callId	string	接口调用序号
agentInfo	AgentInfo	目标座席信息
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

### 5.6.2 OnBeBarged(被强插事件)

#### 事件描述

质检座席调用强插接口强插普通座席时,普通座席侧主动上报该事件。

#### 方法定义

```
void OnBeBarged(
string callId,
AgentInfo agentInfo,
AgentInfo fromAgentInfo
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
agentInfo	AgentInfo	被强插座席信息
fromAgentInfo	AgentInfo	发起强插的座席信息

# 5.6.3 OnForceReleaseResult(拦截结果)

### 事件描述

质检座席调用 拦截接口拦截普通座席时,质检座席侧主动上报该事件。

```
void OnForceReleaseResult(
    string callId,
    AgentInfo agentInfo,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
agentInfo	AgentInfo	被拦截座席信息
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

### 5.6.4 OnBeForceReleased(被拦截事件)

#### 事件描述

质检座席调用 拦截接口拦截普通座席时,普通座席侧主动上报该事件。

#### 方法定义

```
void OnBeForceReleased(
string callId,
AgentInfo agentInfo,
AgentInfo fromAgentInfo
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
agentInfo	AgentInfo	被拦截座席信息
fromAgentInfo	AgentInfo	发起拦截的座席信息

# 5.6.5 OnMonitorResult(监听结果)

### 事件描述

质检座席调用 监听接口监听普通座席时,质检座席侧主动上报该事件。

```
void OnMonitorResult(
    string callId,
    AgentInfo agentInfo,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
agentInfo	AgentInfo	目标座席信息
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.6.6 OnMonitorToBargeResult(监听转强插结果)

### 事件描述

质检座席调用 监听转强插接口时,质检座席侧主动上报该事件。

#### 方法定义

```
void OnMonitorToBargeResult(
    string callId,
    AgentInfo agentInfo,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
agentInfo	AgentInfo	目标座席信息
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.6.7 OnMonitorToForceReleaseResult(监听转拦截结果)

#### 事件描述

质检座席调用**监听转拦截**接口时,质检座席侧主动上报该事件。

```
void OnMonitorToForceReleaseResult(
   string callId,
   AgentInfo agentInfo,
   bool flag,
   int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
agentInfo	AgentInfo	目标座席信息
flag	bool	接口调用结果标识 <ul><li>true: 成功</li><li>false: 失败</li></ul>
errorCode	int	错误码

# 5.6.8 OnAddToBlackListResult(设置黑名单结果)

### 事件描述

座席调用设置黑名单接口后,上报的结果事件。

### 方法定义

```
void OnAddToBlackListResult(
    string callId,
    string phoneNumber,
    int lockTime,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
phoneNumber	string	电话号码
lockTime	int	锁定时长,单位:秒
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

### 5.6.9 OnAddToBlackListExResult(设置黑名单结果扩展)

#### 事件描述

座席调用设置黑名单扩展接口后,上报的结果事件。

#### 方法定义

```
void OnAddToBlackListExResult(
    string callId,
    string phoneNumber,
    int lockTime,
    short filterMode,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
phoneNumber	string	电话号码
lockTime	int	锁定时长,单位:秒
filterMode	short	<ul><li>黑名单模式</li><li>● 1: 拒绝</li><li>● 2: 降级</li></ul>
flag	bool	接口调用结果标识  • true: 成功  • false: 失败
errorCode	int	错误码

### 5.6.10 OnRemoveFromBlackListResult(取消黑名单结果)

#### 事件描述

座席调用取消黑名单接口后,上报的结果事件。

```
void OnRemoveFromBlackListResult(
   string callId,
   string phoneNumber,
   bool flag,
```

int errorCode

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
phoneNumber	string	电话号码
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.6.11 OnRemoveAllBlackListResult(移除全部黑名单结果)

### 事件描述

座席调用移除全部黑名单接口后,上报的结果事件。

#### 方法定义

```
void OnRemoveAllBlackListResult(
   string callId,
   bool flag,
   int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  • true: 成功  • false: 失败
errorCode	int	错误码

# 5.6.12 OnSelectPickupResult(选择应答结果)

#### 事件描述

质检座席调用选择应答接口后,上报的结果事件。

```
void OnSelectPickupResult(
    string callId,
    DialContent dialContent,
    bool flag,
    int errorCode
)
```

### 参数描述

参数	类型	描述
callId	string	接口调用序号
dialContent	DialContent	呼叫内容
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.6.13 OnSnatchPickupResult(代接结果)

### 事件描述

质检座席调用代接接口后,上报的结果事件。

### 方法定义

```
void OnSnatchPickupResult(
    string callId,
    AgentInfo agentInfo,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
agentInfo	AgentInfo	代接座席信息
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

### 5.6.14 OnSnatchPickUpAnswer(代接应答事件)

#### 事件描述

当前座席的来电呼叫被质检座席代接应答后,系统主动返动该事件给当前座席。

#### 方法定义

```
void OnSnatchPickUpAnswer(
string callId,
DialContent dialContent,
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
dialContent	DialContent	代接呼叫内容

# 5.6.15 OnBeSnatchedPickup(被代接事件)

#### 事件描述

质检座席调用代接接口后,被代接座席侧主动上报该事件。

#### 方法定义

```
void OnBeSnatchedPickup(
   string callId,
   AgentInfo agentInfo,
   AgentInfo fromAgentInfo
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
agentInfo	AgentInfo	被代接座席信息
fromAgentInfo	AgentInfo	发起代接的座席信息

### 5.6.16 OnForceLogoutResult(强制座席签出结果)

#### 事件描述

质检座席调用**强制座席签出**接口强制签出其它普通座席后,质检座席侧主动上报该结果事件。

```
void OnForceLogoutResult(
   string callId,
   AgentInfo agentInfo,
   bool flag,
   int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
agentInfo	AgentInfo	被强制退出的座席信息
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.6.17 OnBeForceLogout(座席被强制签出事件)

#### 事件描述

质检座席调用**强制座席签出**接口强制签出其它普通座席后,普通座席侧主动上报该结果事件。

### 方法定义

```
void OnBeForceLogout(
string callId,
AgentInfo agentInfo,
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
agentInfo	AgentInfo	被强制退出的座席信息

### 5.6.18 OnOtherBeForceRelease(对方被拦截事件)

#### 事件描述

本端座席收到对端座席被第三方拦截事件,由系统主动返回给上层应用。这个回调事件已经废弃,不推荐使用。

```
void OnOtherBeForceRelease(
   string callId,
   AgentInfo otherAgentInfo,
   AgentInfo sponsorAgentInfo
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
otherAgentInfo	AgentInfo	对端座席信息
sponsorAgentInfo	AgentInfo	拦截对端座席的第三方座 席信息

# 5.6.19 OnChangeAgentPwdResult(修改座席密码结果)

#### 事件描述

座席调用修改座席密码接口后,上报的结果事件。

#### 方法定义

```
void OnChangeAgentPwdResult(
    string callId,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

### 5.6.20 OnAddToRedListResult(设置优先级名单结果)

#### 事件描述

座席调用设置优先级名单接口后,上报的结果事件。

```
void OnAddToRedListResult(
    string callId,
    string phoneNumber,
    int priority,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
phoneNumber	string	电话号码
priority	int	名单优先级,数值越大级别越高
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.6.21 OnRemoveFromRedListResult(取消优先级名单结果)

### 事件描述

座席调用取消优先级名单接口后,上报的结果事件。

#### 方法定义

```
void OnRemoveFromRedListResult(
    string callId,
    string phoneNumber,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
phoneNumber	string	电话号码
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.6.22 OnRemoveAllRedListResult(移除全部优先级名单结果)

### 事件描述

座席调用移除全部优先级名单接口后,上报的结果事件。

#### 方法定义

```
void OnRemoveAllRedListResult(
    string callId,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.6.23 OnWhisperResult(耳语结果)

#### 事件描述

质检座席调用耳语接口时,质检座席侧主动上报该事件。

### 方法定义

```
void OnWhisperResult(
   string callId,
   string agentId,
   bool flag,
   int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
agentId	string	被耳语座席工号

参数	类型	描述
flag	bool	接口调用结果标识 ● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.6.24 OnSuperviseStopResult(停止质检结果)

#### 事件描述

质检座席调用停止质检接口时,质检座席侧主动上报该事件。

#### 方法定义

```
void OnSuperviseStopResult(
    string callId,
    string agentId,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
agentId	string	被质检座席工号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.6.25 OnSuperviseSwitchResult(质检切换结果)

#### 事件描述

质检座席调用质检切换接口时,质检座席侧主动上报该事件。

```
void OnSuperviseSwitchResult(
   string callId,
   string agentId,
   int operType,
```

bool flag, int errorCode

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
agentId	string	被质检座席工号
operType	int	质检操作类型 <ul><li>0: 监听</li><li>1: 强插</li><li>3: 耳语</li></ul>
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.6.26 OnBeWhispered(被耳语事件)

### 事件描述

质检座席调用耳语接口耳语普通座席时,普通座席侧主动上报该事件。

### 方法定义

```
void OnBeWhispered(
    string callId,
    AgentInfo agentInfo,
    AgentInfo fromAgentInfo
```

参数	类型	描述
callId	string	接口调用序号
agentInfo	AgentInfo	被耳语座席信息
fromAgentInfo	AgentInfo	发起耳语的质检座席信息

# 5.6.27 OnBeMonitored(被监听事件)

#### 事件描述

质检座席调用 监听接口监听普通座席时,普通座席侧主动上报该事件。

#### 方法定义

```
void OnBeMonitored(
string callId,
AgentInfo agentInfo,
AgentInfo fromAgentInfo
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
agentInfo	AgentInfo	被监听座席信息
fromAgentInfo	AgentInfo	发起监听的座席信息

### 5.6.28 OnBeStopSupervise(停止质检事件)

#### 事件描述

质检座席调用停止质检接口时,普通座席侧主动上报该事件。

### 方法定义

```
void OnBeStopSupervise(
   string callId,
   AgentInfo agentInfo,
   AgentInfo fromAgentInfo,
   OperType operType
)
```

参数	类型	描述
callId	string	接口调用序号
agentInfo	AgentInfo	被质检座席信息
fromAgentInfo	AgentInfo	质检座席信息
operType	int	质检操作类型 <ul><li>0: 监听</li><li>1: 插入</li><li>3: 耳语</li></ul>

# 5.6.29 OnSendMessageResult(发送便签结果返回)

#### 事件描述

座席调用发送便签接口后,上报的结果事件。

#### 方法定义

```
void OnSendMessageResult(
    string callId,
    string agentID,
    string msgContent,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
agentID	string	目标座席工号
msgContent	string	便签消息
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.6.30 OnReceiveMessage(接收便签事件)

#### 事件描述

当座席收到其他座席发送的便签事件时,主动向上抛出该事件。

### 方法定义

```
void OnReceiveMessage(
    string agentId,
    string msgContent
)
```

参数	类型	描述
agentId	string	发送方座席工号
msgContent	string	便签消息内容

### 5.6.31 OnSetAutoAnswerResult(设置是否自动应答结果返回)

#### 事件描述

座席调用**设置是否自动应答**接口后,上报的结果事件。

#### 方法定义

```
void OnSetAutoAnswerResult(
string callId,
bool isAutoAnswer,
bool flag,
int errorCode
)
```

### 参数描述

参数	类型	描述
callId	string	接口调用序号
isAutoAnswer	bool	是否自动应答
		● true: 是
		● false: 否
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.6.32 OnSetCallDataResult(设置呼叫随路数据结果返回)

#### 事件描述

座席调用**设置呼叫随路数据**接口后,上报的结果事件。

### 方法定义

```
void OnSetCallDataResult(
string callId,
string callData,
bool flag,
int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号

参数	类型	描述
callData	string	待设置的呼叫随路数据
flag	bool	接口调用结果标识  ● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.6.33 OnQueryCallDataResult(获取呼叫随路数据结果返回)

#### 事件描述

座席调用获取呼叫随路数据接口后,上报的结果事件。

#### 方法定义

```
void OnQueryCallDataResult(
    string callId,
    string callData,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
callData	string	获取到的呼叫随路数据
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.6.34 OnSetCustomerLevelResult(设置呼叫的客户级别结果返回)

#### 事件描述

座席调用设置呼叫的客户级别接口后,上报的结果事件。

### 方法定义

void OnSetCustomerLevelResult(
 string callId,
 string phoneNumber,

CS

```
int level,
bool flag,
int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
phoneNumber	string	当前通话的电话号码
level	int	客户级别,数值越大级别越高
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.6.35 OnQueryCallIDOnAgentResult(查询座席的 CallId 结果返回)

#### 事件描述

座席调用查询座席的CallId接口后,上报的结果事件。

#### 方法定义

```
void OnQueryCallIDOnAgentResult(
   string callId,
   string agentId,
   string callIDs,
   bool flag,
   int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
agentId	string	座席工号
callIDs	string	查询到的座席CallId集合
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.6.36 OnQueryAgentIPResult(查询座席 IP 结果)

#### 事件描述

座席调用**查询座席IP**接口后,上报的结果事件。

#### 方法定义

```
void OnQueryAgentIPResult(
    string agentId,
    string ip,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
agentId	string	座席工号
ip	string	座席签入CTI的IP
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.6.37 OnResetAgentSkillGroupsResult(重置座席技能队列结果)

### 事件描述

座席调用**重置座席技能队列**接口后,上报的结果事件。

### 方法定义

```
void OnResetAgentSkillGroupsResult(
string callId,
bool flag,
int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号

参数	类型	描述
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.6.38 OnSubscribeAgentEventResult(订阅座席事件返回)

#### 事件描述

座席调用**订阅座席事件**接口订阅其它座席事件后,上报的结果事件。 目前只支持久不应答事件。

#### 方法定义

```
void OnSubscribeAgentEventResult(
   string callId,
   string agentid,
   string eventType,
   bool flag,
   int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
agentid	string	座席工号
eventType	string	参考事件类型EventTypeEnum
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.6.39 OnStopSubscribeAgentEventResult(取消订阅座席事件返回)

#### 事件描述

座席调用**取消订阅座席事件**接口取消订阅其它座席事件后,上报的结果事件。 目前只支持久不应答事件。

```
void OnStopSubscribeAgentEventResult(
    string callId,
    string agentid,
    string eventType,
    bool flag,
    int errorCode
)
```

### 参数描述

参数	类型	描述
callId	string	接口调用序号
agentid	string	座席工号
eventType	string	参考事件类型EventTypeEnum
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.6.40 OnAgentEventNotify(座席事件通知)

### 事件描述

座席事件通知, 主动返回。

目前只支持久不应答事件。

### 方法定义

```
void OnAgentEventNotify(
   string agentid,
   string eventType,
   string eventInfo
)
```

参数	类型	描述
agentid	string	座席工号
eventType	string	参考事件类型EventTypeEnum
eventInfo	string	事件参数信息

### 5.7 IDeviceEvents Methods: 录音管理类事件

### 5.7.1 OnStartRecord(开始录音信息)

#### 事件描述

座席通话时,设备侧主动向上层应用返回开始录音信息的事件。

#### 方法定义

```
void OnStartRecord(
    string callId,
    string recordId,
    string fileName,
    List<string> numbers
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
recordId	string	录音ID
fileName	string	录音文件路径信息
numbers	List <string></string>	录音文件包含电话号码集合

### 5.7.2 OnRecordFailure(录音失败)

#### 事件描述

座席通话时,设备侧主动向上层应用返回录音失败事件。

#### 方法定义

```
void OnRecordFailure(
   string callId,
   string recordId,
   string errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
recordId	string	录音ID

参数	类型	描述
errorCode	string	错误码

# 5.7.3 OnPlayRecordResult(播放录音结果)

### 事件描述

座席调用播放录音接口后,上报的结果事件。

#### 方法定义

```
void OnPlayRecordResult(
    string callId,
    string recordId,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
recordId	string	录音ID
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.7.4 OnPlayStatus(录音播放状态)

### 事件描述

播放录音时,设备侧主动向上层应用返回录音播放状态事件。

```
void OnPlayStatus(
    string callId,
    int nTotalLength,
    int nPlayLength,
    bool status
)
```

参数	类型	描述
callId	string	接口调用序号
nTotalLength	int	录音总长度
nPlayLength	int	录音播放长度
status	bool	播放状态
		● true: 正常播放
		● false: 非正常播放

### 5.7.5 OnPlayRecordPauseResult(暂停播放结果)

#### 事件描述

座席调用**暂停播放**接口后,上报的结果事件。

#### 方法定义

```
void OnPlayRecordPauseResult(
string callId,
bool flag,
int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.7.6 OnPlayRecordResumeResult(继续播放结果)

#### 事件描述

座席调用继续播放接口后,上报的结果事件。

### 方法定义

void OnPlayRecordResumeResult(
 string callId,

```
bool flag,
int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.7.7 OnPlayRecordSeekToResult(播放跳转结果)

#### 事件描述

座席调用播放跳转接口后,上报的结果事件。

#### 方法定义

```
void OnPlayRecordSeekToResult(
string callId,
bool flag,
int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.7.8 OnPlayRecordStopResult(停止播放结果)

### 事件描述

座席调用**停止播放**接口后,上报的结果事件。

#### 方法定义

void OnPlayRecordStopResult(
 string callId,

```
bool flag,
int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.7.9 OnGetRecordLengthResult(获取录音时长结果)

### 事件描述

座席调用 获取录音时长接口后,上报的结果事件。

#### 方法定义

```
void OnGetRecordLengthResult(
   string callId,
   int length,
   bool flag,
   int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
length	int	时长 (ms)
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.7.10 OnEndPlayRecord(放音结束)

#### 事件描述

当放音到文件末尾或座席停止放音时, 主动向上层应用返回该事件。

```
void OnEndPlayRecord(
string callId,
string recordId
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
recordId	string	录音ID

# 5.7.11 OnEndRecord(停止录音)

#### 事件描述

座席通话结束后, 主动返回停止录音返回事件。

## 方法定义

```
void OnEndRecord(
    string callId,
    string recordId,
    List<string> numbers
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
recordId	string	录音ID
numbers	List <string></string>	录音文件包含电话号码集合

# 5.7.12 OnDownloadRecordFileResult(下载录音文件结果)

#### 事件描述

座席调用下载录音文件接口后,上报的结果事件。

```
void OnDownloadRecordFileResult (
    string callId,
    string localFileName,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
localFileName	string	文件保存全路径。如: "C:\ipcc\c.zip"
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.7.13 OnSetCustomizedRecordInfoResult(设置录音关联信息结果)

## 事件描述

座席调用**设置录音关联信息**接口后,上报的结果事件。

## 方法定义

```
void OnSetCustomizedRecordInfoResult(
    string callId,
    int num,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
num	int	设置关联信息的字符数量
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.8 IDeviceEvents Methods: 无线资源类事件

# 5.8.1 OnPTTResult(PTT 操作结果)

#### 事件描述

座席调用PTT接口后,上报的结果事件。

#### 方法定义

```
void OnPTTResult(
    string callId,
    PTTOperate pttOperate,
    string pttLevel,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
pttOperate	PTTOperate	PTT操作类型  ● PutDown: 按下  ● PutUp: 抬起
pttLevel	string	PTT优先级,按照LTE的定义,优先级 为1~15,数字越小优先级越高,ECC 平台仅进行透明传输
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.8.2 OnReportPTTOff(PTT 话权被释放通知)

## 事件描述

PTT被抢权通知消息通知。该回调事件已经废弃,不建议使用。

```
void OnReportPTTOff (
   string callId,
   string number,
   PTTOperate pttOperate,
   string reason
)
```

参数	类型	描述
callId	string	接口调用序号
number	string	电话号码
pttOperate	PTTOperate	PTT操作类型
		● PutDown: 按下
		● PutUp: 抬起
reason	string	原因

# 5.8.3 OnQueryWirelessInfoResult(查询无线设备信息结果)

#### 事件描述

座席调用查询无线设备信息接口后,上报的结果事件。

#### 方法定义

```
void OnQueryWirelessInfoResult(
    string callId,
    WirelessDeviceInfo wirelessDeviceInfo,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
wirelessDeviceInfo	WirelessDeviceInfo	无线设备信息
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.8.4 OnQueryAllWirelessDeviceResult(查询所有无线设备信息结果)

#### 事件描述

座席调用查询所有无线设备信息接口后,上报的结果事件。

```
void OnQueryAllWirelessDeviceResult(
   string callId,
   List<WirelessDeviceInfo> wirelessDeviceInfos,
   bool flag,
   int errorCode
)
```

## 参数描述

参数	类型	描述
callId	string	接口调用序号
wirelessDeviceInfos	List <wirelessdeviceinfo></wirelessdeviceinfo>	无线设备信息集合
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.8.5 OnQueryUsingWirelessDeviceResult(查询当前使用的无线设备信息结果)

#### 事件描述

座席调用**查询当前使用的无线设备信息结果**接口后,上报的结果事件。

## 方法定义

```
void OnQueryUsingWirelessDeviceResult(
    string callId,
    WirelessDeviceInfo wirelessDeviceInfo,
    bool flag,
    int errorCode
)
```

类型	描述
string	接口调用序号
WirelessDeviceInfo	无线设备信息
bool	接口调用结果标识
	● true: 成功 ● false: 失败
int	错误码
	string WirelessDeviceInfo

# 5.9 IDeviceEvents Methods: 短信服务类事件

# 5.9.1 OnReceiveSMS(新短信接收事件)

## 事件描述

新短信消息到达座席侧接收通知事件,由设备侧主动推送过来。

#### 方法定义

```
void OnReceiveSMS(
string callId,
string mediaType,
int CCBIdx,
string callNumber,
string receiveTime,
string note
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
mediaType	string	呼叫的媒体类型  ● 1: 文字交谈  ● 2: 单击通话 <b>说明</b> 匿名通话,用户不需輸入任何信息。
CCBIdx	int	● 5: 普通语音通话等 短信呼叫的内部索引号
callNumber	string	短信发送者电话号码
receiveTime	string	接收短信时间
note	string	短信文本内容

# 5.9.2 OnReceiveP2PSMS(短信接收事件)

#### 事件描述

短信消息到达座席侧通知事件,由设备侧主动推送过来。

## 方法定义

void OnReceiveP2PSMS( string callNumber, string receiveTime,

```
string note
)
```

参数	类型	描述
callNumber	string	短信发送者电话号码
receiveTime	string	接收时间
note	string	短信文本内容

# 5.9.3 OnSendSMSResult(发送短信结果)

#### 事件描述

座席调用发送短信接口后,上报的结果事件。

## 方法定义

```
void OnSendSMSResult(
    string callId,
    string phoneNumber,
    string note,
    int ulSeqNo,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
phoneNumber	string	短信接收者电话号码
note	string	发送的短信文本内容
ulSeqNo	int	序列编号
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

#### <u>CS</u>

# 5.9.4 OnReplySMSResult(回复短信结果)

#### 事件描述

座席调用回复短信接口后,上报的结果事件。

#### 方法定义

```
void OnReplySMSResult(
string callId,
string phoneNumber,
string note,
int ulSeqNo,
bool flag,
int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
phoneNumber	string	回复短信座席电话号码
note	string	回复短信内容
ulSeqNo	int	序列编号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.9.5 OnClientDataWithoutCallConfirm(座席主动发送短信后,返回用户短信接收确定)

#### 事件描述

座席主动发送短信后,返回用户短信接收确认事件。

```
void OnClientDataWithoutCallConfirm(
    string callId,
    int confirmerType,
    string confirmer,
    int ulDataSeqNo
)
```

参数	类型	描述
callId	string	接口调用序号
confirmerType	int	保留字段,默认值为0
confirmer	string	接收确认方用户号码
ulDataSeqNo	int	序列编号

# 5.9.6 OnClientDataWithoutCallFail(座席发送短信,用户确认失败)

## 事件描述

座席主动发送短信后,返回用户短信接收失败确认事件。

#### 方法定义

```
void OnClientDataWithoutCallFail(
   string callId,
   int confirmerType,
   string confirmer,
   int ulDataSeqNo
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
confirmerType	int	保留字段,默认值为0
confirmer	string	接收确认方用户号码
ulDataSeqNo	int	序列编号

# 5.9.7 OnReleaseCallExWithCCBIdxResult(座席释放指定短信用户呼叫结果)

#### 事件描述

座席调用 座席释放指定短信用户呼叫接口后,上报的结果事件。

```
void OnReleaseCallExWithCCBIdxResult(
string callId,
int mediaType,
int CCBIdx,
```

bool flag, int errorCode

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
mediaType	int	呼叫的媒体类型  ● 1: 文字交谈  ● 2: 单击通话 说明    匿名通话,用户不需输入任何信
		<ul><li>息。</li><li>● 5: 普通语音通话等</li></ul>
CCBIdx	int	短信呼叫的内部索引号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.9.8 OnReceiveReleaseSMS(短信呼叫释放通知)

## 事件描述

当座席收到短信呼叫释放通知时,触发此事件。

## 方法定义

void OnReceiveReleaseSMS( string callId, string mediaType, int CCBIdx

参数	类型	描述
callId	string	接口调用序号

参数	类型	描述
mediaType	string	呼叫的媒体类型
		● 1: 文字交谈
		● 2: 单击通话
		<b>说明</b> 匿名通话,用户不需输入任何信 息。
		● 5: 普通语音通话等
CCBIdx	int	短信呼叫的内部索引号

# 5.9.9 OnTransferSMSToAgentResult(短信转移结果)

#### 事件描述

座席调用短信转移接口将短信会话转移给其它座席后收到的返回结果。

#### 方法定义

```
void OnTransferSMSToAgentResult(
string callId,
bool flag,
int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.10 IDeviceEvents Methods:邮件服务类事件

# 5.10.1 OnReceiveEmail(新邮件接收消息)

#### 事件描述

新邮件消息到达座席侧通知事件,由设备侧主动推送过来。

```
void OnReceiveEmail(
   string bHandle,
   string emailCallId,
   string receiveTime
)
```

#### 参数描述

参数	类型	描述
bHandle	string	标识
emailCallId	string	邮件呼叫ID
receiveTime	string	邮件呼叫到达时间,注:不是指 收到邮件内容的时间

# 5.10.2 OnSendEmailResult(发送邮件结果)

# 事件描述

座席调用发送邮件接口后,上报的结果事件。

## 方法定义

```
void OnSendEmailResult(
    string callId,
    string index,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
index	string	发送邮件的索引号,必须是4个字节数字
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.10.3 OnReplyEmailResult(回复邮件结果)

#### 事件描述

座席调用回复邮件接口后,上报的结果事件。

#### 方法定义

```
void OnReplyEmailResult(
string callId,
bool flag,
int errorCode
)
```

### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.10.4 OnQueryAgentEmailCfgInfoResult(查询座席邮件配置信息结果)

#### 事件描述

座席调用查询座席邮件配置信息接口后,上报的结果事件。

## 方法定义

```
void OnQueryAgentEmailCfgInfoResult(
   string callId,
   string pop3Svr,
   string smtpSvr,
   string receiveEmailAddr,
   string sendEmailAddr,
   string emailUserName,
   string emailPassword,
   bool flag,
   int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号

errorCode

错误码

参数	类型	描述
pop3Svr	string	POP3服务器地址
smtpSvr	string	SMTP服务器地址
receiveEmailAddr	string	座席私有接收邮箱地址
sendEmailAddr	string	座席私有发送邮箱地址
emailUserName	string	邮箱用户名
emailPassword	string	邮箱密码
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败

# 5.10.5 OnGetSendEmailIndexResult(获取发送邮件索引号结果)

int

## 事件描述

座席调用获取发送邮件索引号接口后,上报的结果事件。

## 方法定义

```
void OnGetSendEmailIndexResult(
    string callId,
    int emailIndex,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
emailIndex	int	发送邮件的索引号,必须是4个字节数字
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.10.6 OnReleaseMailCall(停止邮件呼叫结果)

#### 事件描述

座席调用停止邮件呼叫接口后,上报的结果事件。

#### 方法定义

```
void OnReleaseMailCall(
string callId,
bool flag,
int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.10.7 OnAgentReceiveEmailRemind(催促座席尽快处理 Email 呼叫)

#### 事件描述

座席有邮件到达提醒时, 座席侧主动上报该事件。

#### 方法定义

void OnAgentReceiveEmailRemind()

## 参数描述

无

# 5.11 IDeviceEvents Methods: 视频通话类事件

# 5.11.1 OnPickupVCCallResult(接听视频电话结果)

#### 事件描述

座席调用接听视频电话接口后,上报的结果事件。

```
void OnPickupVCCallResult(
    string callId,
    DialContent dialContent,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
dialContent	DialContent	电话信息
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.11.2 OnHangupVCCallResult(挂断视频电话结果)

## 事件描述

座席调用挂断视频电话接口后,上报的结果事件。

## 方法定义

```
void OnHangupVCCallResult(
string callId,
bool flag,
int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.11.3 OnVCPhoneStatusNotify(视频电话状态通知)

#### 事件描述

视频电话状态变化时,座席侧主动上报该事件。

#### 方法定义

```
void OnVCPhoneStatusNotify(
    string status
)
```

#### 参数描述

参数	类型	描述	
status	string	视频电话状态	
		● 0: 振铃	
		● 1: 摘机	
		● 2: 挂断	

# 5.11.4 OnNewVCCall(新视频电话来电)

## 事件描述

座席侧有新视频电话时, 主动上报该事件。

## 方法定义

```
void OnNewVCCall(
   string callid,
   DeviceType callerType,
   string callerNumber,
   DeviceType calleeType,
   string calleeNumber
)
```

参数	类型	描述
callid	string	接口调用序号
callerType	DeviceType	主叫设备类型
callerNumber	string	主叫号码
calleeType	DeviceType	被叫设备类型
calleeNumber	string	被叫号码

# 5.11.5 OnOtherVCCallHangup(视频通话对方挂断)

#### 事件描述

座席视频通话时,对方挂断时被动收到该事件。

### 方法定义

```
void OnOtherVCCallHangup(
string callId,
DialContent dialContent)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
dialContent	DialContent	对方呼叫信息

# 5.11.6 OnStartRecordVCCall(开始录制视频电话通知)

## 事件描述

座席视频通话时,设备侧主动向上层应用返回该事件。

## 方法定义

```
void OnStartRecordVCCall (
    string callId,
    String conferenceId,
    string recordId,
    string fileName,
    List<string> numbers
)
```

参数	类型	描述
callId	string	接口调用序号
conferenceId	string	会议标识
recordId	string	录音文件唯一ID
fileName	string	录音文件路径信息
numbers	List <string></string>	录音文件包含电话号码集 合

# 5.11.7 OnEndRecordVCCall(结束录制视频电话通知)

#### 事件描述

座席视频通话结束后, 主动返回停止录音返回事件。

#### 方法定义

```
void OnEndRecordVCCall (
    string callId,
    string conferenceId,
    string recordId,
    string fileName,
    List<string> numbers
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
conferenceId	string	会议标识
recordId	string	录音文件唯一ID
fileName	string	录音文件路径信息
numbers	List <string></string>	录音文件包含电话号码集 合

# 5.11.8 OnVCCallLongNoAnswer(视频电话久不应答通知)

#### 事件描述

座席侧有新视频电话时,如果座席久不应答则会上报该事件。

## 方法定义

void OnVCCallLongNoAnswer()

## 参数描述

无。

# 5.11.9 OnVCCallRecordFail(录播失败通知)

#### 事件描述

座席侧有新视频电话,如果座席录制视频失败则返回该事件。

```
void OnVCCallRecordFail(
string errorCode
)
```

#### 参数描述

参数	类型	描述
errorCode	string	错误码(常量-1)

# 5.12 IDeviceEvents Methods: 话机管理类事件

# 5.12.1 OnPhoneInitializeResult(初始化话机结果事件)

#### 事件描述

调用初始化话机接口后,上报的结果事件。

## 方法定义

```
void OnPhoneInitializeResult (
string callId,
PhoneUsedMode phoneMode,
bool flag,
int errorCode
```

## 参数描述

参数	类型	描述
callId	string	接口调用序号
phoneMode	PhoneUsedMode	话机使用模式
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.12.2 OnPhoneUnInitializeResult(去初始化话机结果事件)

#### 事件描述

调用去初始化话机接口后,上报的结果事件。

```
void OnPhoneUnInitializeResult (
string callId,
bool flag,
int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.12.3 OnPhoneLinkageStatusNotify(话机联动状态上报事件)

#### 事件描述

当话机开启联动或关闭联动时,上报该事件。

## 方法定义

```
void OnPhoneLinkageStatusNotify (
bool isEnabled,
int reason
)
```

#### 参数描述

参数	类型	描述
isEnabled	bool	联动是否开启
		● true: 开启联动
		● false: 美闭联动
reason	int	联动功能关闭原因

# 5.12.4 OnPhoneStatusChangeEvent(话机状态变更上报事件)

#### 事件描述

当话机状态发生变更时,上报该事件,暂不支持联动场景下的IPCC线路状态上报。

```
void OnPhoneStatusChangeEvent (
   string callSessionId,
   PhoneStatus status,
   string number,
   int callType
)
```

#### 参数描述

参数	类型	描述
callSessionId	string	呼叫标识
status	PhoneStatus	话机状态
Number	string	对端号码, 可为空
callType	int	当前呼叫类型
		● 0: IPT线路呼叫
		● 1: 内线呼叫
		● 4: IPCC线路呼叫
		<b>说明</b> 在上报摘机、挂机、振铃和通话时, callType为预留字段,为空值。

# 5.12.5 OnPhoneAnswerRequestEx(话机请求应答事件)

## 事件描述

请求应答话机呼叫,支持联动场景和内置软电话非联动场景。

## 方法定义

```
void OnPhoneAnswerRequestEx (
    string callSessionId,
    string callerNumber,
    string calleeNumber,
    bool isEmergencyCall,
    int callType
)
```

参数	类型	描述
callSessionId	string	呼叫标识
callerNumber	string	主叫号码
calleeNumber	string	被叫号码

参数	类型	描述
isEmergencyCall	bool	是否为紧急呼叫
		● true: 紧急呼叫
		● false: 非紧急呼叫
callType	int	当前呼叫类型,可为空
		● 0: IPT线路呼叫
		● 1: 内线呼叫
		● 4: IPCC线路呼叫

# 5.12.6 OnPhoneAnswerRequest(话机请求应答事件)

#### 事件描述

请求应答话机呼叫, 支持联动场景和内置软电话非联动场景。

#### 方法定义

```
void OnPhoneAnswerRequestEx (
   string callSessionId,
   string callerNumber,
   string calleeNumber,
   bool isEmergencyCall,
)
```

#### 参数描述

参数	类型	描述
callSessionId	string	呼叫标识
callerNumber	string	主叫号码
calleeNumber	string	被叫号码
isEmergencyCall	bool	是否为紧急呼叫

# 5.12.7 OnPhoneAutoAnswerEvent(话机自动应答事件)

#### 事件描述

调用初始化话机接口并设置自动应答功能,座席接听来电成功后上报的结果事件。

```
void OnPhoneAutoAnswerEvent (
    string callSessionId,
    string callerNumber,
    string calleeNumber,
    bool isEmergencyCall,
    int callType
)
```

#### 参数描述

参数	类型	描述
callSessionId	string	呼叫标识
callerNumber	string	主叫号码
calleeNumber	string	被叫号码
isEmergencyCall	bool	是否为紧急呼叫 <ul><li>true: 紧急呼叫</li><li>false: 非紧急呼叫</li></ul>
callType	int	当前呼叫类型,可为空

# 5.12.8 OnPhoneAnswerResult(话机应答结果事件)

#### 事件描述

调用话机应答接口后,上报的结果事件。

## 方法定义

```
void OnPhoneAnswerResult (
    string callId,
    string callSessionId,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
callSessionId	string	呼叫标识

参数	类型	描述
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.12.9 OnPhoneReleaseResult(话机释放结果事件)

## 事件描述

调用话机释放接口后,上报的结果事件。

#### 方法定义

```
void OnPhoneReleaseResult (
string callId,
string callSessionId,
bool flag,
int errorCode
)
```

## 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码
callSessionId	sttring	呼叫唯一标识

# 5.12.10 OnPhoneReleaseEvent(话机呼叫释放事件)

#### 事件描述

当对方主动挂断,或后台拆线导致呼叫释放时,会上报该事件。

```
void OnPhoneReleaseEvent (
string callSessionId,
string caller,
string callee,
```

```
int reason
```

参数	类型	描述
callSessionId	string	呼叫标识
caller	string	主叫号码
callee	string	被叫号码
reason	int	错误码

# 5.12.11 OnPhoneMuteResult(话机静音结果事件)

#### 事件描述

调用话机静音接口后,上报的结果事件。

#### 方法定义

```
void OnPhoneMuteResult (
string callId,
string callSessionId,
bool isMute,
bool flag,
int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码
callSessionId	string	呼叫唯一标识
isMute	bool	是否静音
		● true: 静音。
		● false: 取消静音。

# 5.12.12 OnPhoneSwitchSoundDeviceResult(话机呼叫切换声卡设备结果事件)

#### 事件描述

调用话机呼叫切换声卡设备接口后,上报的结果事件。

#### 方法定义

```
void OnPhoneSwitchSoundDeviceResult (
    string callId,
    string callSessionId,
    bool isToAidDevice,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码
callSessionId	string	呼叫唯一标识
isToAidDevice	bool	是否为切换到辅助设备
		● true: 辅助设备
		● false: 主设备

# 5.12.13 OnPhoneGetSoundDeviceResult(话机查询声卡设备信息结果事件)

## 事件描述

调用话机查询声卡设备信息接口后,上报的结果事件。

```
void OnPhoneGetSoundDeviceResult (
    string callId,
    bool isInput,
    int deviceNum,
    List<SoundDeviceInfo> deviceList,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
deviceNum	int	音频设备数量
errorCode	int	错误码
isInput	bool	是否为输入设备 <ul><li>true: 输入设备</li><li>false: 输出设备</li></ul>
deviceList	List <sounddevice Info&gt;</sounddevice 	音频设备信息列表。

# 5.12.14 OnPhoneSetSoundDeviceResult(话机设置声卡设备信息结果事件)

## 事件描述

调用话机设置声卡设备信息接口后,上报的结果事件。

## 方法定义

```
void OnPhoneSetSoundDeviceResult (
    string callId,
    bool isInput,
    string deviceIndex,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

参数	类型	描述
isInput	bool	是否为输入设备 <ul><li>true: 输入设备</li><li>false: 输出设备</li></ul>
deviceIndex	string	音频设备索引号。

# 5.12.15 OnPhoneGetVolumeResult(话机查询设备音量结果事件)

## 事件描述

调用话机查询设备音量接口后,上报的结果事件。

### 方法定义

```
void OnPhoneGetVolumeResult (
    string callId,
    bool isInput,
    int volumeValue,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
isInput	bool	是否为输入设备 <ul><li>true: 输入设备</li><li>false: 输出设备</li></ul>
volumeValue	int	音量大小
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.12.16 OnPhoneGetDeviceVolumeResult(话机查询辅助设备音量结果事件)

#### 事件描述

调用话机查询辅助设备音量接口后,上报的结果事件。

#### 方法定义

```
void OnPhoneGetDeviceVolumeResult (
    string callId,
    bool isInput,
    int volumeValue,
    AudioDeviceType deviceType,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
isInput	bool	是否为输入设备 <ul><li>true: 输入设备</li><li>false: 输出设备</li></ul>
volumeValue	int	音量大小
deviceType	AudioDeviceType	声卡设备类型
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

## 5.12.17 OnPhoneSetVolumeResult(话机设置设备音量结果事件)

## 事件描述

调用话机设置设备音量接口后,上报的结果事件。

```
void OnPhoneSetVolumeResult (
   string callId,
   bool isInput,
   int volumeValue,
   bool flag,
   int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
isInput	bool	是否为输入设备 <ul><li>true: 输入设备</li><li>false: 输出设备</li></ul>
volumeValue	int	音量大小
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.12.18 OnPhoneSetDeviceVolumeResult(话机设置设备音量结果事件)

#### 事件描述

调用话机设置辅助设备音量接口后,上报的结果事件。

## 方法定义

```
void OnPhoneSetDeviceVolumeResult (
    string callId,
    bool isInput,
    int volumeValue,
    AudioDeviceType deviceType,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
isInput	bool	是否为输入设备 <ul><li>true: 输入设备</li><li>false: 输出设备</li></ul>
volumeValue	int	音量大小
deviceType	AudioDeviceType	声卡设备类型

参数	类型	描述
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码

# 5.12.19 OnPhoneGetVolumeGainByCallResult(话机查询指定呼叫音量结果事件)

## 事件描述

调用话机查询指定呼叫音量接口后,上报的结果事件。

## 方法定义

```
void OnPhoneGetVolumeGainByCallResult (
    string callId,
    string callSessionId,
    bool isInput,
    int gainValue,
    bool flag,
    int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
callSessionId	string	呼叫唯一标识
isInput	bool	是否为输入设备 <ul><li>true: 输入设备</li><li>false: 输出设备</li></ul>
gainValue	int	音量增益大小
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.12.20 OnPhoneSetVolumeGainByCallResult(话机设置指定呼叫音量结果事件)

#### 事件描述

调用话机设置指定呼叫音量增益接口后,上报的结果事件。

#### 方法定义

```
void OnPhoneSetVolumeGainByCallResult (
    string callId,
    string callSessionId,
    bool isInput,
    int gainValue,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
callSessionId	string	呼叫唯一标识
isInput	bool	是否为输入设备 <ul><li>true: 输入设备</li><li>false: 输出设备</li></ul>
gainValue	int	音量增益大小
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码

# 5.13 IDeviceEvents Methods: 集群调类事件

# 5.13.1 OnTalkingGroupQueryResult(查询集群群组结果事件)

## 事件描述

调用查询集群群组接口后,上报的结果事件。

```
void OnTalkingGroupQueryResult (
string callId,
```

```
int queryType,
List<TalkingGroup> groupList,
PageInfo pageInfo,
bool flag,
int errorCode
```

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码
queryType	int	查询类型  ● 1: 指定分页进行查询  ● 2: 指定集群群组进行查询
groupList	List <talkinggrou p&gt;</talkinggrou 	集群群组信息列表,可为空 说明 当queryType为2:指定集群群组进行查 询时,该参数必填
pageInfo	PageInfo	分页信息,可为空 <b>说明</b> 当queryType为1:指定分页进行查询 时,该参数必填

# 5.13.2 OnTalkingGroupQueryStateResult(查询集群群组状态结果事件)

#### 事件描述

调用查询集群群组状态接口后,上报的结果事件。

```
void OnTalkingGroupQueryStateResult (
   string callId,
   TalkingGroupState groupState,
   bool flag,
   int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码
groupState	TalkingGroupStat e	集群群组状态信息

# 5.13.3 OnTalkingGroupAddResult(新增集群群组结果事件)

#### 事件描述

调用**新增集群群组**接口后,上报的结果事件。

#### 方法定义

```
void OnTalkingGroupAddResult (
    string callId,
    TalkingGroup groupInfo,
    bool flag,
    int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码
groupInfo	TalkingGroup	所操作的集群群组信息

# 5.13.4 OnTalkingGroupModifyResult(修改集群群组结果事件)

#### 事件描述

调用修改集群群组接口后,上报的结果事件。

```
void OnTalkingGroupModifyResult (
string callId,
TalkingGroup groupInfo,
bool flag,
int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码
groupInfo	TalkingGroup	所操作的集群群组信息

# 5.13.5 OnTalkingGroupDeleteResult(删除集群群组结果事件)

#### 事件描述

调用删除集群群组接口后,上报的结果事件。

## 方法定义

```
void OnTalkingGroupDeleteResult (
string callId,
string groupNum,
bool flag,
int errorCode
)
```

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码
groupNum	string	集群群组号码,可为空

## 5.13.6 OnRadioGroupQueryResult(查询无线集群群组结果事件)

### 事件描述

调用查询无线集群群组接口后,上报的结果事件。

### 方法定义

```
void OnRadioGroupQueryResult (
    string callId,
    int queryType,
    List<RadioGroup> groupList,
    PageInfo pageInfo,
    bool flag,
    int errorCode
)
```

### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码
queryType	int	查询类型
		● 1: 指定分页进行查询。
		● 2: 指定集群群组进行查询。
groupList	List <radiogroup< td=""><td>集群群组信息列表,可为空</td></radiogroup<>	集群群组信息列表,可为空
	>	说明 ************************************
		当queryType为2:指定集群群组进行查 询时,该参数必填
pageInfo	PageInfo	分页信息,可为空
		<b>说明</b> 当queryType为1:指定分页进行查询 时,该参数必填

### 5.13.7 OnTalkingGroupSubscribeResult(订阅集群群组结果事件)

### 事件描述

调用订阅集群群组接口后,上报的结果事件。

### 方法定义

```
void OnTalkingGroupSubscribeResult (
string callId,
```

```
int expire,
string groupNum,
bool flag,
int errorCode
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码
expire	int	订阅时间
		说明
		● 当时间为0时,停止订阅。
		● 当时间大于0时,开始订阅。
groupNum	string	集群群组号码,可为空

## 5.13.8 OnTalkingGroupUserStateChange(集群群组用户状态变更通知事件)

### 事件描述

当集群群组中的某个用户状态发生变更时,推送该事件给该集群群组所有订阅者。

#### 方法定义

```
void OnTalkingGroupUserStateChange (
    GroupUserState groupUserState
)
```

#### 参数描述

参数	类型	描述
groupUserState	GroupUserState	集群群组用户状态信息

### 5.13.9 OnTalkingGroupMonitorResult(监听集群群组结果事件)

#### 事件描述

调用监听集群群组接口后,上报的结果事件。

### 方法定义

```
void OnTalkingGroupMonitorResult (
    string callId,
    string groupNum,
    bool isMonitor,
    bool isRecvOnly,
    bool flag,
    int errorCode
)
```

### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码
groupNum	string	集群群组号码
isMonitor	bool	开始监听还是停止监听
		● true: 开始监听
		● false: 停止监听
		说明
		•
isRecvOnly	bool	是否为半双工
		● true: 半双工
		● false: 全双工
		<b>说明</b> 当开始监听时,该值表示是否为半双工(只收不 发),即不需要进行抢权操作

## 5.13.10 OnTalkingGroupSelectResult(Monitor/Select 切换结果事件)

### 事件描述

调用Monitor/Select切换接口后,上报的结果事件。

### 方法定义

```
void OnTalkingGroupSelectResult (
   string callId,
   string groupNum,
   bool isSelect,
   bool flag,
```

CS)

```
int errorCode
```

### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识
		● true: 成功
		● false: 失败
errorCode	int	错误码
groupNum	string	集群群组号码
isSelect	bool	开始Select还是停止Select
		● true: 开始
		● false: 停止

### 5.13.11 OnTalkingGroupPTTResult(抢权或放权结果事件)

### 事件描述

调用抢权或放权接口后,上报的结果事件。

### 方法定义

```
void OnTalkingGroupPTTResult (
   string callId,
   string groupNum,
   bool isPTT,
   bool flag,
   int errorCode
)
```

### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	string	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	bool	错误码
groupNum	bool	集群群组号码

参数	类型	描述
isPTT	int	抢权还是放权
		● true: 抢权
		● false: 放权

## 5.13.12 OnTalkingGroupGetCallInfoResult(查询集群群组呼叫信息结果事件)

### 事件描述

调用查询集群群组呼叫信息接口后,上报的结果事件。

### 方法定义

```
void OnTalkingGroupGetCallInfoResult (
    string callId,
    string groupNum,
    string callSessionId,
    bool flag,
    int errorCode
)
```

### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码
groupNum	string	集群群组号码
callSessionId	string	该集群群组呼叫的会话唯一标识

## 5.13.13 OnMixedGroupCallStartResult(发起混编集群群组呼叫结果事件)

#### 事件描述

调用发起混编集群群组呼叫接口后,上报的结果事件。

### 方法定义

```
void OnMixedGroupCallStartResult (
   string callId,
   string groupNum,
   bool flag,
   int errorCode
)
```

### 参数描述

参数	类型	描述
callId	stirng	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码
groupNum	string	集群群组号码

## 5.13.14 OnMixedGroupCallEndResult(结束混编集群群组呼叫结果事件)

### 事件描述

调用结束混编集群群组呼叫接口后,上报的结果事件。

### 方法定义

```
void OnMixedGroupCallEndResult (
   string callId,
   string groupNum,
   bool flag,
   int errorCode
)
```

#### 参数描述

参数	类型	描述
callId	string	接口调用序号
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码
groupNum	string	集群群组号码

## 5.13.15 OnMixedGroupChangeUserStateResult(修改混编集群群组用户状态结果事件)

### 事件描述

调用修改混编集群群组用户状态接口后,上报的结果事件。

### 方法定义

```
void OnMixedGroupChangeUserStateResult (
   string callId,
   string groupNum,
   List<MixedGroupUserState> userStateList,
   bool flag,
   int errorCode
)
```

### 参数描述

参数	类型	描述
callId	string	接口调用序号
groupNum	string	集群群组号码
flag	bool	接口调用结果标识  ● true: 成功  ● false: 失败
errorCode	int	错误码
userStateList	List <mixedgroup UserState&gt;</mixedgroup 	集群群组用户状态列表

6 错误码

**6** 错误码

- 6.1 ECC SDK错误码
- 6.2 服务端错误码
- 6.3 Voice.ocx接口错误码
- 6.4 TUP错误码

## 6.1 ECC SDK 错误码

错误码值	描述
0	正确返回
-1	状态错误
1	一般性错误
2	配置文件不存在
3	加载配置文件出现错误
11	路由失败
13	强制呼出错误
14	没有该主叫
15	控制块错误或参数错误
16	DEVICE_ID不存在
17	CALLID不存在
18	没有呼损待取
19	没有该技能树节点
21	连接已断
22	没有代理
99	APC卡没有初始化
100	内部错误
101	参数错误
102	通讯层暂时忙
103	相应资源不可用或故障
104	API调用超时
105	没有预拨号呼叫
106	太多线程(大于50)同时调用API
107	空间不够
108	一般性错误
109	连接已断
110	没有向CCS登录
111	权限错误

错误码值	描述
112	用户信息文件错误
113	用户信息文件没有找到
114	系统用户数已满
115	用户已经存在
116	用户没有找到
117	文件打开错误
118	控制块状态错误
119	控制块HANDLE不匹配
120	无效话务员
121	无效的呼叫
122	无效的队列
123	话务员状态错误
124	资源连接错误
125	取保持错误
126	保持错误
127	接续超时
128	占用话务员错误
129	调用空指针
130	质检等待SCP或OC超时
131	该工号的SPY已经登录
132	用户名或密码错误
133	CCS正在接收数据,不能发送配置文件
134	初始化文件被其他的进程加锁
135	数据包失序
136	操作中内存错误
137	座席录音终止
138	座席放音终止
146	用户无响应
147	用户无应答
149	用户拒绝

错误码值	描述
155	目的地不可达
159	座席已经加锁
202	话务员未签入
207	路由到话务员失败
259	资源不足
260	Modem
261	传真
262	传真或Modem
263	自动应答机
531	用户挂机
537	被叫忙
538	空号或错号
548	首次登录必须修改密码
549	手机不在服务区
550	密码修改时间短于规定的最短有效期
551	配置被其他用户锁定
601	应答前用户挂机
602	排队机无资源
603	无应答拆线
604	呼叫拒绝
605	交换设备忙
606	排队机超时
701	排队中用户主动挂机
702	排队超时释放
2253	座席已经加锁
2282	在CALLID中服务器类型错误
2354	密码不符合强制加密策略
2382	登录失败,工号被锁定
2383	修改密码失败,与最近修改过的密码相同
2384	修改密码失败,密码小于最小长度

错误码值	描述	
2385	修改密码失败,密码包含用户名	
2386	密码过期	
2410	修改密码失败,密码和用户名的逆序相同	
2412	修改密码失败,密码中包含连续3位相同数字	
3498	找不到对应的jau控制块	
21002	传入参数非法	
25014	已存在此质检员对该话务员的任务	
25018	无相应权限	
20601	转移到技能队列时,参数错误	
25212	占用质检员失败	
60000	参数错误	
60001	未初始化	
60002	无效的email地址	
60003	无效的TokenID	
60004	无效的IPv4地址	
60005	无效的Port	
60006	文件不存在	
60007	系统错误	
60008	不满足密码复杂度	
60009	代表通道错误	
60010	数据库异常	
60011	直呼时抢权失败	
60012	会议中无线数量超过1	
60013	被呼叫座席不处于示闲态或者学习态	
60014	SDK 连接EDS双机失败	
60015	密码长度不符合规范,密码必须8-32位	
60100	会议成员不存在	
60101	会议成员已存在	
61001	表示建立呼叫失败	
61002	挂机	

错误码值	描述
61003	表明被叫号码不存在
61004	表示排队机无法为该呼叫分配中继线路或收号器
61005	表示在规定的时间内未收到正常的应答消息
61006	表示主叫号码或被叫号码错误
61007	表示被叫号码的电话占线
61008	表示用户没有足够的权限来建立呼叫
61009	有效黑名单数量超出限制
61010	红名单数量超出限制
61020	监控座席数不能超过100个
62000	控件ocx一般错误
40000001	用户未登录
40000002	用户无质检录音/质检放音权限
40000003	入参不合法
40000004	录音下载失败之文件名不合法
40000005	录音下载失败之没有配置映射盘符
4000006	录音下载失败之录音文件不存在
4000007	录音下载失败之获取文件根目录失败
40000008	录音下载失败之文件下载打包时出现写入异常
4000009	录音下载失败之空间不足
40000010	文件流关闭异常
40000011	录音文件格式不支持
40000012	录音格式系统参数未配置
40000013	录音下载失败之不能下载其他vdn下录音

### 6.2 服务端错误码

错误码	中文描述
-1	未知错误
0	操作成功
1	操作失败

错误码	中文描述
1200	操作数据库失败
100001	参数为空
100002	参数非法
100003	参数长度大于最大限制
100004	参数长度小于最小限制
100005	参数值大于最大限制
100006	参数值小于最小限制
100007	参数不在合法参数集合
100008	参数值非法
100009	存在非数字的字符
100010	存在非法邮件地址
100011	存在非法参数集合
100012	参数已存在
100013	参数不存在
100014	参数不能被删除
100015	会议资源不足
300000	参数为空
300001	参数非法
300002	内部错误
300003	重复的无线集群群组
300004	超出最大配置数量
300005	操作数据库失败
300006	无效集群群组名称,名称中含有存在非法字符
300007	无效集群群组号码, 名称中含有存在非法字符
300008	无效集群群组类型
300009	无效无线集群群组号码
300010	无效调度席号码
300011	无效其他用户号码
300012	录音文件存储相对地址超过最大长度
300013	群组描述超过最大支持长度

错误码	中文描述
300014	号码为空
300015	无效会议Id
300016	无效与会人号码
300017	无效与会人角色
300018	无效与会人类型
300019	无线集群群组被集群群组引用
300020	重复的集群群组
300021	预定会议失败
300022	邀请入会失败
300023	到达集群群组monitor数量上限
300024	到达集群群组monitor数量上限
300025	已经处于Monitor状态
300026	集群群组未配置
300027	无线集群群组类型
300028	无线集群群组未配置
300029	调度席未配置
300030	不在Monitor状态
300031	取消monitor操作失败
300032	取消monitor操作失败
300033	无效无线集群模式
300034	配置的群组个数达到上限
300035	记录为空
300036	调度席号码个数无效
300037	无线集群群组号码个数无效
300038	其他用户号码个数无效
300039	无线集群群组,调度席号码,其他用户号码包含重复的记录
300040	超过最大用户数目
300041	公共参数表中的index字段值无效
300042	集群群组用户帐号不存在

300043	错误码	中文描述
300045       集群群组用户未配置         300046       无效集群群组用户或密码         300047       号码长度超过32个字符         300048       号码中包含非法字符,号码仅0-9、A~F或a~f等字符         300053       配置的集群群组个数达到上限         300054       调度席己处于PTT状态         300055       调度席不在PTT状态         300056       PTT操作正在处理         300057       PTT操作需要先Select         300058       PTT操作需要先Select         300059       话权冲突         300060       群组中无有效集群群组         300061       PTT群组信息无效         300062       Monitor操作超时         300063       PTT操作超时         300064       集群群组用户帐号已被引用         300065       查询用户状态信息失败         300066       下行PTT群组信息无效         300067       PTT请求被拒绝         300068       内部通信错误         300070       DeSelect群组信息无效         300071       调度席已经在Select状态         300073       调度席已经在Select状态         300074       更新用户状态信息失败	300043	无效集群群组用户
300046         无效集群群组用户或密码           300047         号码长度超过32个字符           300048         号码中包含非法字符, 号码仅0~9、A~F或a~f等字符           300053         配置的集群群组个数达到上限           300054         调度席已处于PTT状态           300055         调度席不在PTT状态           300056         PTT操作正在处理           300057         PTT操作需要先Select           300058         PTT操作需要先Select           300059         适权冲突           300059         适权冲突           300060         群组中无有效集群群组           300061         PTT群组信息无效           300062         Monitor操作超时           300063         PTT操作超时           300064         集群群组用户帐号已被引用           300065         查询用户状态信息失败           300066         下行PTT群组信息无效           300067         PTT请求被拒绝           300068         内部通信错误           300070         DeSelect群组信息无效           300071         调度席已经在Select状态           300073         调度席已经在Select状态           300074         更新用户状态信息失败	300044	集群群组用户已存在
300047	300045	集群群组用户未配置
300048	300046	无效集群群组用户或密码
300053   配置的集群群组个数达到上限   300054   调度席已处于PTT状态   300055   调度席不在PTT状态   300056   PTT操作正在处理   300057   PTT操作需要先monitor   300058   PTT操作需要先Select   300059   话权冲突   300057   PTT群组信息无效   300060   群组中无有效集群群组   300060   PTT群组信息无效   300061   PTT群组信息无效   300062   Monitor操作超时   300063   PTT操作超时   300064   集群群组用户帐号已被引用   300065   查询用户状态信息失败   300066   下行PTT群组信息无效   300067   PTT请求被拒绝   300068   内部通信错误   300069   Select群组信息无效   300070   DeSelect群组信息无效   300071   调度席不在Select状态   300072   调度席已达到Select数量上限   200074   更新用户状态信息失败   200074   更新用户状态信息失败   200074   更新用户状态信息失败   200074   200073   300074   更新用户状态信息失败   200075   300074   200076   200076   200077	300047	号码长度超过32个字符
300054       调度席己处于PTT状态         300055       调度席不在PTT状态         300056       PTT操作正在处理         300057       PTT操作需要先monitor         300058       PTT操作需要先Select         300059       话权冲突         300050       PTT群组信息无效         300060       群组中无有效集群群组         300061       PTT群组信息无效         300062       Monitor操作超时         300063       PTT操作超时         300064       集群群组用户帐号已被引用         300065       查询用户状态信息失败         300066       下行PTT群组信息无效         300067       PTT请求被拒绝         300068       内部通信错误         300070       DeSelect群组信息无效         300071       调度席不在Select状态         300072       调度席已经在Select状态         300073       调度席已达到Select数量上限         300074       更新用户状态信息失败	300048	号码中包含非法字符,号码仅0~9、A~F或a~f等字符
300055       调度席不在PTT状态         300056       PTT操作正在处理         300057       PTT操作需要先monitor         300058       PTT操作需要先Select         300059       话权冲突         300057       PTT群组信息无效         300060       群组中无有效集群群组         300061       PTT群组信息无效         300062       Monitor操作超时         300063       PTT操作超时         300064       集群群组用户帐号已被引用         300065       查询用户状态信息失败         300066       下行PTT群组信息无效         300067       PTT请求被拒绝         300068       内部通信错误         300070       DeSelect群组信息无效         300071       调度席不在Select状态         300072       调度席已经在Select状态         300073       调度席已达到Select数量上限         300074       更新用户状态信息失败	300053	配置的集群群组个数达到上限
300056       PTT操作正在处理         300057       PTT操作需要先monitor         300058       PTT操作需要先Select         300059       话权冲突         300057       PTT群组信息无效         300060       群组中无有效集群群组         300061       PTT群组信息无效         300062       Monitor操作超时         300063       PTT操作超时         300064       集群群组用户帐号已被引用         300065       查询用户状态信息失败         300066       下行PTT群组信息无效         300067       PTT请求被拒绝         300068       内部通信错误         300069       Select群组信息无效         300070       DeSelect群组信息无效         300071       调度席已经在Select状态         300072       调度席已经在Select状态         300073       调度席已达到Select数量上限         300074       更新用户状态信息失败	300054	调度席已处于PTT状态
300057   PTT操作需要先monitor   300058   PTT操作需要先Select   300059   话权冲突   300057   PTT群组信息无效   300060   群组中无有效集群群组   300061   PTT群组信息无效   300062   Monitor操作超时   300063   PTT操作超时   300064   集群组用户帐号已被引用   300065   查询用户状态信息失败   7行PTT群组信息无效   300066   下行PTT群组信息无效   300067   PTT请求被拒绝   300068   内部通信错误   300069   Select群组信息无效   300070   DeSelect群组信息无效   300071   调度席不在Select状态   300072   调度席已经在Select状态   300073   调度席已达到Select数量上限   更新用户状态信息失败   更新用户状态信息失败   更新用户状态信息失败	300055	调度席不在PTT状态
300058       PTT操作需要先Select         300059       话权冲突         300057       PTT群组信息无效         300060       群组中无有效集群群组         300061       PTT群组信息无效         300062       Monitor操作超时         300063       PTT操作超时         300064       集群群组用户帐号已被引用         300065       查询用户状态信息失败         300066       下行PTT群组信息无效         300067       PTT请求被拒绝         300068       内部通信错误         300070       DeSelect群组信息无效         300071       调度席不在Select状态         300072       调度席已经在Select状态         300073       调度席已达到Select数量上限         300074       更新用户状态信息失败	300056	PTT操作正在处理
300059       话权冲突         300057       PTT群组信息无效         300060       群组中无有效集群群组         300061       PTT群组信息无效         300062       Monitor操作超时         300063       PTT操作超时         300064       集群群组用户帐号已被引用         300065       查询用户状态信息失败         300066       下行PTT群组信息无效         300067       PTT请求被拒绝         300068       内部通信错误         300069       Select群组信息无效         300070       DeSelect群组信息无效         300071       调度席不在Select状态         300072       调度席已经在Select状态         300073       调度席已达到Select数量上限         300074       更新用户状态信息失败	300057	PTT操作需要先monitor
300057       PTT群组信息无效         300060       群组中无有效集群群组         300061       PTT群组信息无效         300062       Monitor操作超时         300063       PTT操作超时         300064       集群群组用户帐号已被引用         300065       查询用户状态信息失败         300066       下行PTT群组信息无效         300067       PTT请求被拒绝         300068       内部通信错误         300069       Select群组信息无效         300070       DeSelect群组信息无效         300071       调度席不在Select状态         300072       调度席已经在Select状态         300073       调度席已达到Select数量上限         300074       更新用户状态信息失败	300058	PTT操作需要先Select
300060       群组中无有效集群群组         300061       PTT群组信息无效         300062       Monitor操作超时         300063       PTT操作超时         300064       集群群组用户帐号已被引用         300065       查询用户状态信息失败         300066       下行PTT群组信息无效         300067       PTT请求被拒绝         300068       内部通信错误         300069       Select群组信息无效         300070       DeSelect群组信息无效         300071       调度席不在Select状态         300072       调度席已经在Select状态         300073       调度席已达到Select数量上限         300074       更新用户状态信息失败	300059	话权冲突
300061       PTT群组信息无效         300062       Monitor操作超时         300063       PTT操作超时         300064       集群群组用户帐号已被引用         300065       查询用户状态信息失败         300066       下行PTT群组信息无效         300067       PTT请求被拒绝         300068       内部通信错误         300069       Select群组信息无效         300070       DeSelect群组信息无效         300071       调度席不在Select状态         300072       调度席已经在Select状态         300073       调度席已达到Select数量上限         300074       更新用户状态信息失败	300057	PTT群组信息无效
300062       Monitor操作超时         300063       PTT操作超时         300064       集群群组用户帐号已被引用         300065       查询用户状态信息失败         300066       下行PTT群组信息无效         300067       PTT请求被拒绝         300068       内部通信错误         300069       Select群组信息无效         300070       DeSelect群组信息无效         300071       调度席不在Select状态         300072       调度席已经在Select状态         300073       调度席已达到Select数量上限         300074       更新用户状态信息失败	300060	群组中无有效集群群组
300063       PTT操作超时         300064       集群群组用户帐号已被引用         300065       查询用户状态信息失败         300066       下行PTT群组信息无效         300067       PTT请求被拒绝         300068       内部通信错误         300069       Select群组信息无效         300070       DeSelect群组信息无效         300071       调度席不在Select状态         300072       调度席已经在Select状态         300073       调度席已达到Select数量上限         300074       更新用户状态信息失败	300061	PTT群组信息无效
300064       集群群组用户帐号已被引用         300065       查询用户状态信息失败         300066       下行PTT群组信息无效         300067       PTT请求被拒绝         300068       内部通信错误         300069       Select群组信息无效         300070       DeSelect群组信息无效         300071       调度席不在Select状态         300072       调度席已经在Select状态         300073       调度席已达到Select数量上限         300074       更新用户状态信息失败	300062	Monitor操作超时
300065       查询用户状态信息失败         300066       下行PTT群组信息无效         300067       PTT请求被拒绝         300068       内部通信错误         300069       Select群组信息无效         300070       DeSelect群组信息无效         300071       调度席不在Select状态         300072       调度席已经在Select状态         300073       调度席已达到Select数量上限         300074       更新用户状态信息失败	300063	PTT操作超时
300066       下行PTT群组信息无效         300067       PTT请求被拒绝         300068       内部通信错误         300069       Select群组信息无效         300070       DeSelect群组信息无效         300071       调度席不在Select状态         300072       调度席已经在Select状态         300073       调度席已达到Select数量上限         300074       更新用户状态信息失败	300064	集群群组用户帐号已被引用
300067       PTT请求被拒绝         300068       内部通信错误         300069       Select群组信息无效         300070       DeSelect群组信息无效         300071       调度席不在Select状态         300072       调度席已经在Select状态         300073       调度席已达到Select数量上限         300074       更新用户状态信息失败	300065	查询用户状态信息失败
300068       内部通信错误         300069       Select群组信息无效         300070       DeSelect群组信息无效         300071       调度席不在Select状态         300072       调度席已经在Select状态         300073       调度席已达到Select数量上限         300074       更新用户状态信息失败	300066	下行PTT群组信息无效
300069       Select群组信息无效         300070       DeSelect群组信息无效         300071       调度席不在Select状态         300072       调度席已经在Select状态         300073       调度席已达到Select数量上限         300074       更新用户状态信息失败	300067	PTT请求被拒绝
300070       DeSelect群组信息无效         300071       调度席不在Select状态         300072       调度席已经在Select状态         300073       调度席已达到Select数量上限         300074       更新用户状态信息失败	300068	内部通信错误
300071       调度席不在Select状态         300072       调度席已经在Select状态         300073       调度席已达到Select数量上限         300074       更新用户状态信息失败	300069	Select群组信息无效
300072       调度席已经在Select状态         300073       调度席已达到Select数量上限         300074       更新用户状态信息失败	300070	DeSelect群组信息无效
300073	300071	调度席不在Select状态
300074 更新用户状态信息失败	300072	调度席已经在Select状态
	300073	调度席已达到Select数量上限
第一个 集群群组已达到Subscribe数量上限	300074	更新用户状态信息失败
	300075	集群群组已达到Subscribe数量上限

错误码	中文描述
300076	用户已在此状态
300077	混编群组已处于呼叫状态
300078	非主席用户无此操作权限
300079	集群群组类型不能被修改
300080	用户不在订阅状态
300081	普通集群群组不能配置其他用户号码
300082	该群组类型不是混编集群群组
300083	混编集群群组不支持此操作
300084	调度席达到最大订阅数
300085	录音文件不合法
300086	主机处理数据失败
300087	调度席帐号未开户
300088	鉴权失败
300089	录音文件地址有效字符为0~9、A~Z、a~z、/或空字符串。 当参数不为空时,必须以0~9、A~Z、a~z开头,以/字符结 尾
300090	混编群组正在呼叫中,请在呼叫结束后操作
300091	Monitor状态下不允许取消订阅
300092	PTT状态下不允许取消订阅
300093	一个无线集群群组只能配置在一个普通集群群组中
300094	ReceiveOnly属性为True的集群群组中,调度席不能进行 PTT操作
300095	集群群组已PTT其他集群群组

### 6.3 Voice.ocx 接口错误码

错误码	描述
79999	默认错误
70000	输入参数错误
70001	未初始化
70002	无效的email地址

错误码	描述
70003	无效的TokenID
70004	无效的IPv4地址
70005	无效的Port
70006	文件不存在
70007	系统错误
70008	不满足密码复杂度
70009	代表通道错误、已经强权(已经抢权, 不允许改变通道)
70010	数据库异常
70011	sip消息RequestlineReject,抢权失败
70012	会议中无线数量超过1
70013	对方状态错误(不是处于示闲态或者学 习态)
70014	SDK 连接EDS双机失败
70015	密码长度不符合规范
70016	操作超时
70017	帐户名/密码错误
70019	获取主声卡设备失败
70020	获取声卡设备失败
70021	软电话注册失败
70100	会议成员不存在
70101	会议成员已存在
70102	不能移除正在呼叫中的成员
70103	本地会议不存在
70200	用户重复登录UPortal
70201	UPortal服务器未响应
70202	UPortal服务器连接失败
70203	用户token值为空
70204	不允许抢权
70205	用户不存在
70210	WebSocket通道未开启

错误码	描述
70211	所查询集群群组号不在monitor或组呼状 态中
72000	ocx其他错误

### 6.4 TUP 错误码

错误码	描述
0	Success
1	Voice Ctrl error: Voice control uninitialized
2	Voice Ctrl error: Point of param is null
3	Voice Ctrl error: Param is empty
4	Voice Ctrl error: Param is not a valid value
5	Required information is not configured.
6	Voice Ctrl error: State wrong
7	Voice Ctrl error: Set tup config value failed
8	Voice Ctrl error: Set tup config value failed
9	Voice Ctrl error: System unkown error
1000	Call tup api failed
1001	TUP call general error
1002	TUP call parameter error
1003	TUP call memory alloc error
1004	TUP call system error
1005	TUP call send message error
1006	TUP call system configuration error
1007	TUP call physical network error
1008	TUP call network access failed
1009	TUP call create timer error
1010	TUP call state error
1011	TUP call some other operation has not finished error
1012	TUP call request main control to call error

错误码	描述
1013	TUP call apply audio media resource error
1014	TUP call additional service is on processing error
1015	TUP call record call operation error
1016	TUP call media process error
1017	TUP call exist max call number error
1018	TUP call sip account id not exist error
1019	TUP call call id not exist error
1020	TUP call registering error
1021	TUP call invoke sip register interface error
1022	TUP call invoke sip unregister interface error
1023	TUP call set account info error
1024	TUP call set sip module error
1025	TUP call notification error
1026	TUP call sip server information error
1027	TUP call account information error
1028	TUP call SIPC error
1029	TUP call unregister error
1030	TUP call invoke sip subscribe interface error
1031	TUP call undo subscribe error
1032	TUP call subscribing error
1057	TUP call network error
1058	TUP call service conflict error
1078	TUP call tls root cert error.

7 术语

# **7** <sub>术语</sub>

术语	全称	含义
调度席		调度席即集群群组资源的使用者和集群群组操作的 发起者。调度席可绑定软终端和IP话机,对集群群 组发起的呼叫相关操作需调度席绑定软终端并注册 软终端号码。
VDN	Virtual Directory Number	VDN 是呼叫中心平台内一种虚拟设备的编号,它代表了某一类的应用。在呼叫中心平台,VDN用以指代虚拟呼叫中心。
CTI	Computer Telephony Integration	路由平台服务器
IVR	Interactive Voice Response	互动式语音应答
PBX	Private Branch Exchange	专用电话设备中的一种交换实体,可接入公共电话 交换网,通常供某一企业使用,基于计算机的数字 电话交换机。
WAS	Web Application Server	Web应用服务器