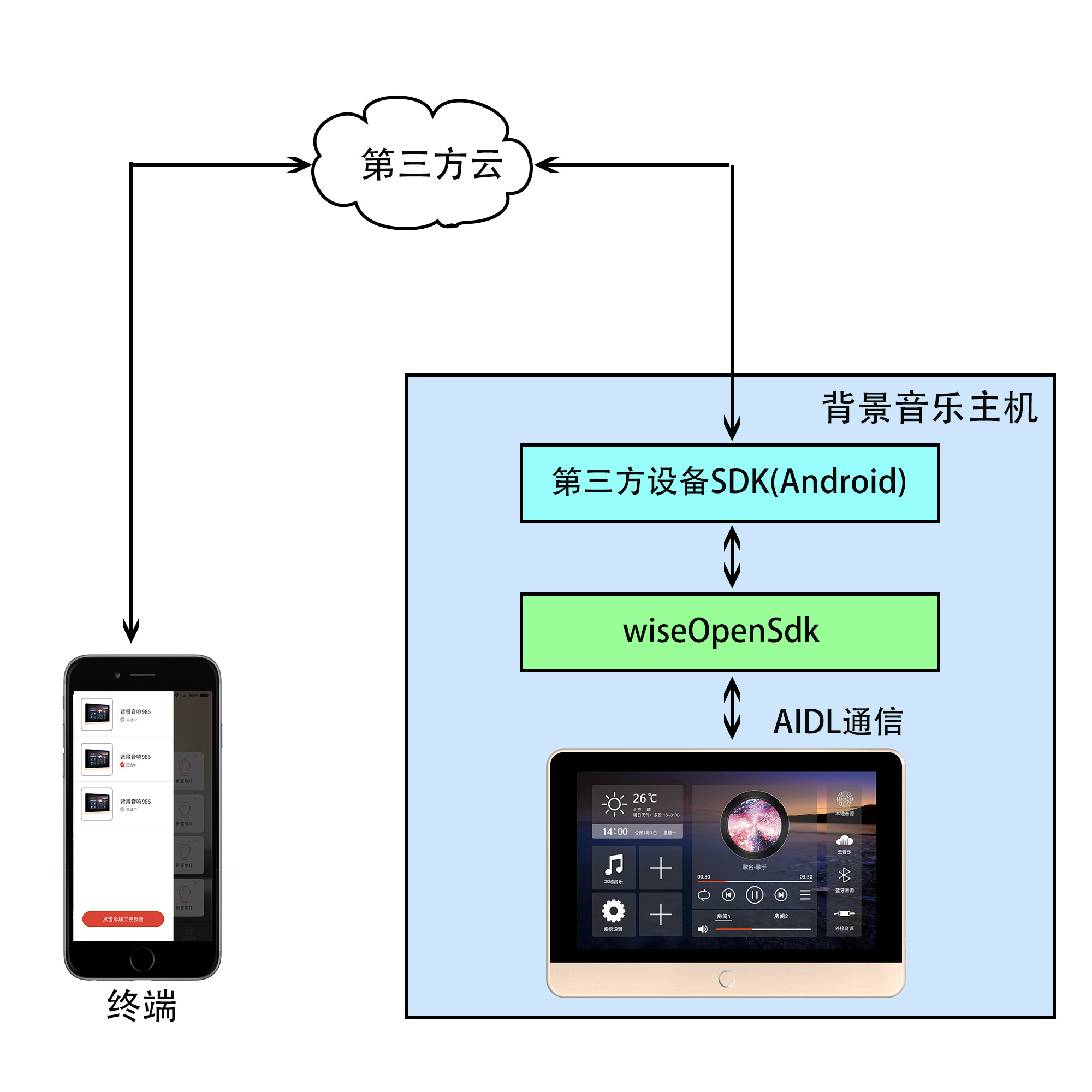
**WiseOpenSdk -** 安卓**SDK**对接说明

**1.1** 版本记录

|  |  |  |
| --- | --- | --- |
| 版本 | 日期 | 修改记录 |
| v 1.0.1 | 2019.10.22 | 初始版本 |
|  |  |  |
|  |  |  |

**1.2** 交互流程图****

**1.3** 导入**SDK**包（以下**x.x.x**参数根据下载版本填入）

**SDK**下载链接：**https://pan.baidu.com/s/1og49LaQ5QE1no6v4SL3RMQ** 密码**:6aov**

将**“huaers\_open\_sdk-release-x.x.x.aar”**文件导入工程中，放到**libs**目录下，并添加配置

**build.grild**

**repositories {**

**flatDir {**

**dirs 'libs'**

**}**

**}**

**dependencies {**

**implementation(name: 'huaers\_open\_sdk-release-x.x.x', ext: ‘aar')**

**}**

**1.4** 添加权限

AndroidMainifest.xml添加以下权限

| 用途 | 权限 |
| --- | --- |
| 允许连接网络 | android.permission.INTERNET |
| 读取外部存储 | android.permission.READ\_EXTERNAL\_STORAGE |
| 写入外部存储 | android.permission.WRITE\_EXTERNAL\_STORAGE |
| 允许程序访问有关GSM网络信息 | android.permission.ACCESS\_NETWORK\_STATE |
| 允许程序访问Wi-Fi网络状态信息 | android.permission.ACCESS\_WIFI\_STATE |

音乐主机为安卓**8.1**版本，以下权限请手动申请

**android.permission.READ\_EXTERNAL\_STORAGE**

**android.permission.WRITE\_EXTERNAL\_STORAGE**

**1.5** 初始化

public class App extends Application {

    @Override public void onCreate() {

        super.onCreate();

        instance = this;

**YFLogUtils.init(true); //**可设置为**false**关闭打印

**BinderPool.getInstance().init(this); //SDK**初始化

    }

}

**1.6 SDK**接口功能介绍（回调函数）

**1.6.1** 关键类说明（回调函数）

|  |  |
| --- | --- |
| 类 | 说明 |
| IMusicCallbackListener | 获取歌曲播放信息 |
| IAudioSourceCallbackListener | 获取音源信息 |
| IPlayListCallbackListener | 本地歌曲列表 |
| IZoneCallbackListener | 分区信息 |
| IAsrCallbackListener | 语音信息 |

**1.6.2** 关键类接口使用说明（回调函数）

**1.6.2.1** 歌曲播放信息（**IMusicCallbackListener**）

获取歌曲播放信息，需先添加监听，监听与释放方法如下：

**//**监听

**PlayerCallBackManager.getInstance().addMusicCallbackListener(this);**

**//**释放

**PlayerCallBackManager.getInstance().removeMusicCallbackListener(this);**

//接口可实现方法

public interface IMusicCallbackListener {

void updatePlayStatus(int i)

void updatePlayPosition(long l)

void updateMedia(int i,MediaInfo mediaInfo)

void updatePlayMode(int i)

}

//方法说明

//更新当前播放状态

void updatePlayStatus(int i)

参数

\* i 歌曲播放状态 参数说明请下拉至第**8**页查看《各接口状态信息表格 **-** 播放状态》

//更新歌曲播放进度

void updatePlayPosition(long l)

参数

\* l 当前歌曲播放进度

//更新当前播放歌曲信息

void updateMedia(int i,MediaInfo mediaInfo)

参数

* i 当前播放序号
* mediaInfo 歌曲信息 参数说明请下拉至第**8**页查看《各接口状态信息表格 **-** 歌曲信息》

//更新当前播放模式

void updatePlayMode(int i)

参数

* i 当前播放模式 参数说明请下拉至第**9**页查看《各接口状态信息表格 **-** 播放模式》

**1.6.2.2** 音源信息（**IAudioSourceCallbackListener**）

获取音源信息，需先添加监听，监听与释放方法如下：

**//**监听

PlayerCallBackManager.getInstance().setAudioSourceCallbackListener(this);

**//**释放

PlayerCallBackManager.getInstance().setAudioSourceCallbackListener(null);

//接口可实现方法

public interface IAudioSourceCallbackListener{

void updateAudioSource(AudioSource audioSource)

}

//方法说明

//更新音源状态

void updateAudioSource(AudioSource audioSource)

参数

* audioSource 音源状态 参数说明请下拉至第**10**页查看《各接口状态信息表格 **-** 音源状态》

**1.6.2.3** 本地歌曲列表（**IPlayListCallbackListener**）

获取本地歌曲列表，需先添加监听，监听与释放方法如下：

**//**监听

**PlayerCallBackManager.getInstance().addPlayListCallbackListener(this)**；

**//**释放

**PlayerCallBackManager.getInstance().removePlayListCallbackListener(this);**

//接口可实现方法

public interface IPlayListCallbackListener{

void updatePlayList(MediaPLayList mediaPLayList)

void updateCurPlayList(MediaPLayList curMediaPLayList)

void updateAllPlayList(List<MediaPLayList> list)

}

//方法说明

//某个播放列表的歌曲信息

void updatePlayList(MediaPLayList mediaPLayList)

参数

* mediaPLayList 歌曲列表信息 参数说明请下拉至第**9**页查看《各接口状态信息表格 **-** 歌曲列表信息》

//当前选择的播放列表

void updateCurPlayList(MediaPLayList curMediaPLayList)

参数

\* curMediaPLayList 参数说明同上

//现有的所有播放列表

void updateAllPlayList(List<MediaPLayList> list)

参数

* list MediaPlayList集合 参数说明同上

**1.6.2.3** 分区信息（**IZoneCallbackListener**）

获取分区信息，需先添加监听，监听与释放方法如下：

**//**监听

**PlayerCallBackManager.getInstance().addZoneCallbackListener(this);**

**//**释放

**PlayerCallBackManager.getInstance().removeZoneCallbackListener(this);**

//接口可实现方法

public interface IZoneCallbackListener{

void updateZone(Zone zone)

}

//更新分区状态

void updateZone(Zone zone)

参数

\*zone 分区信息 参数说明请下拉至第**9**页查看《各接口状态信息表格 **-** 分区信息》

**1.6.2.4** 语音信息（**IAsrCallbackListener**）

获取语音信息，需先添加监听，监听与释放方式如下：

**//**监听

**PlayerCallBackManager.getInstance().setAsrCallbackListener(this);**

**//**释放

**PlayerCallBackManager.getInstance().setAsrCallbackListener(null);**

//接口可实现方法

public interface IAsrCallbackListener{

void callbackAsrResult(String s)

void callbackStatus(int i)

}

//语义识别内容返回

void callbackAsrResult(String dialogId,String s)

参数

\* dialogId 对话唯一标示，判断当前交互是否存在

* s 说话内容。 例：你好小华，唤醒后，说播放音乐，此接口回调“播放音乐”
* 注：如果回调内容不需要，请调用以下方法，服务端默认超时**5S**，超时后将自动响应识别内容，超时后再调用以下方法返还结果，语音服务将不再处理，超时时间可通过方法自定义（方法说明请下拉至第**8**页查看《自定义超时时间》）
* 如果回调内容自己处理，不希望上传云平台，请返回空值
* 初始化
* **PlayerControlClient mPlayerControlClient = PlayerControlClient.getInstance();**
* 传入对话唯一标示（**dialogId**）与需要传给云平台的内容（**asrResult**）**mPlayerControlClient.returnAsrResult(String dialogId,String asrResult);**

**//**当前语音状态

void callbackStatus(int i)

参数

\* i 语音状态参数说明请下拉至第**9**页查看《各接口状态信息表格 **-** 语音状态》

**1.7 SDK**接口功能介绍（主动控制与获取数据功能）

**1.7.1** 初始化

//所有控制功能入口

public PlayerControlClient mPlayerControlClient = PlayerControlClient.getInstance();

**1.7.2** 各接口说明

void play(); //播放

void playIndex(int index); //播放指定位置歌曲

\* index 指定下标（不得超过歌曲总数量）

void pause(); //暂停

void playPause(); //播放暂停切换

void stop(); //停止

void previous(); //上一曲

void next(); //下一曲

void seekTo(long progress); //设置进度

\* progress 进度值（不得超过当前播放歌曲总时长）

long getCurPosition(); //获取进度

int getPlayStatus(); //获取播放状态

\* 返回参数说明请下拉至第**8**页查看《各接口状态信息表格 **-** 播放状态》

boolean isPlaying(); //是否在播放

\* true：正在播放。 false：未播放

MediaInfo getCurMedia(); //获取当前歌曲信息

\* 返回参数说明请下拉至第**8**页查看《各接口状态信息表格 **-** 歌曲信息》

int getPlayMode(); //获取当前播放模式

\* 返回参数说明请下拉至第**9**页查看《各接口状态信息表格 **-** 播放模式》

void setPlayMode(int playMode); //设置播放模式

\* playMode 播放模式 参数说明请下拉至第**8**页查看《各接口状态信息表格 **-** 播放模式》

void switchPlayMode(); //切换播放模式

List<MediaPLayList> getAllPlayLists(); //获取所有播放列表

\* 返回参数说明请下拉至第**9**页查看《各接口状态信息表格 **-** 歌曲列表信息》

MediaPLayList getCurPLayList(); //获取当前播放列表

\* 返回参数说明请下拉至第**9**页查看《各接口状态信息表格 **-** 歌曲列表信息》

void setCurPlayList(long curPlayListId); //设置当前播放列表

\* curPlayListId 传入参数说明请下拉至第**10**页查看《歌曲列表表格 **- id**（**long**）》

MediaPLayList getPlayList(long playListId); //获取指定播放列表

\* curPlayListId 传入参数说明请下拉至第**10**页查看《歌曲列表表格 **- id**（**long**）》

\* 返回参数说明请下拉至第**9**页查看《各接口状态信息表格 **-** 歌曲列表信息》

List<AudioSource> getAudioSourceList(); //获取所有音源

\* 返回参数说明请下拉至第**10**页查看《各接口状态信息表格 **-** 音源状态》

void setCurAudioSource(AudioSource source); //设置音源

\* source 传入参数说明请下拉至第**10**页查看《各接口状态信息表格 **-** 音源状态》

AudioSource getCurAudioSource(); //获取音源状态

\* 返回参数说明请下拉至第**10**页查看《各接口状态信息表格 **-** 音源状态》

void switchAudioSource(); //切换音源

Zone getZone(); //获取分区所有信息

\* 返回参数说明请下拉至第**9**页查看《各接口状态信息表格 **-** 分区信息》

void setZoneSwitch(int zoneNum, boolean zoneOpen); //分区开关控制

\* zoneNum 指定某一分区。 传入参数0～3，分别代表1~4分区

\* zoneOpen 控制分区开关。 传入true为打开。 传入false为关闭分区

void setZoneVolume(int zoneNum, int vol); //分区音量控制

\* zoneNum 指定某一分区。 传入参数0～3，分别代表1~4分区

\* vol 音量控制 传入0～15

void setZoneChannelVolume(int zoneNum, int zoneChannelNum, int vol);

//分区指定声道音量控制

\* zoneNum 指定某一分区 传入参数0～3，分别代表1~4分区

\* zoneChannelNum 指定某声道 传入参数0或1，0代表左声道，1代表右声道

\* vol 音量控制 传入0～15

void switchMute(); //切换静音状态

void mute(boolean mute); //设置静音状态

\* mute 设置指定状态。 传入true打开静音，false关闭静音

boolean isMute();

\* 返回true代表静音打开，false代表静音关闭

void playTts(TTS tts); //语音播报

\* tts 需要播报内容 传入参数说明请下拉至第**9**页查看《各接口状态信息表格 **-** 语音播报》

void stopTts(); //停止播报

void startDialog(); //开始对话

void stopDialog(); //结束对话

void setDialogTimeOutTime(int timeOutTime); //自定义超时时间

\* timeOutTime 超时时间，单位s

Int getDialogTimeOutTime(); //获取超时时间

\* 返回当前超时时间

|  |  |  |  |
| --- | --- | --- | --- |
| **各接口状态信息表格** | | | |
| **总类名称（类型）** | **状态名称** | **参数** | **说明（类型）** |
| 播放状态（int） | MediaPlayState.STOP | 3 | 停止(int) |
| MediaPlayState.PAUSE | 2 | 暂停(int) |
| MediaPlayState.PLAY | 1 | 播放(int) |
| 歌曲信息  (MediaInfo) |  | duration | 时长（long） |
|  | artist | 专辑（String） |
|  | name | 歌名（String） |
|  | albumName | 专辑名（String） |
|  | path | 播放路径（String） |
|  | imageUrl | 图片路径（String） |
| 播放模式（int） | MediaPlayMode. ALL\_PLAY\_MODE | 5 | 全部循环（int） |
| MediaPlayMode. SINGLE\_PLAY\_MODE | 6 | 单曲循环（int） |
| MediaPlayMode. RANDOM\_PLAY\_MODE | 7 | 随机播放（int） |
| MediaPlayMode. ORDER\_PLAY\_MODE | 8 | 顺序播放（int） |
| 音源状态  (AudioSource)  （**id**与**name**参数说明请查第**9**页《音源参数表格》） |  | id | 音源ID（int） |
|  | name | 音源名称（String） |
| 歌曲列表信息  (MediaPLayList)  （**id**与**listName**参数说明请查看第**10**页《歌曲列表表格》） |  | id | 列表ID（long） |
|  | listName | 列表名称（String） |
|  | mediaInfoList | 歌曲信息集合（List<MediaInfo>）  MediaInfo == 此表格中歌曲信息 |
| 分区信息  （Zone）  （**zoneBean**参数说明请查看第**10**页《分区状态表格》） |  | volMax | 最大音量（int） |
|  | zoneBean | 所有分区状态集合  （List<ZoneBean>） |
| 语音状态（int） | AsrTtsState.VOICE\_STATE\_TTS\_START | 1001 | TTS播报开始（int） |
| AsrTtsState.VOICE\_STATE\_TTS\_FINISH | 1002 | TTS播报结束（int） |
| AsrTtsState.VOICE\_STATE\_SESSION\_START | 10001 | 语音对话开始（int） |
| AsrTtsState.VOICE\_STATE\_ASR\_START | 10002 | 开始录音（int） |
| AsrTtsState.VOICE\_STATE\_ASR\_END | 10003 | 结束录音（int） |
| AsrTtsState.VOICE\_STATE\_ASR\_ANALYSIS | 10004 | 开始识别语义（int） |
| AsrTtsState.VOICE\_STATE\_SESSION\_END | 10005 | 语音对接结束（int） |
| 语音播报（TTS） |  | ttsText | 播报内容（String） |
|  | interrupt | 是否打断播报（boolean） |

|  |  |  |
| --- | --- | --- |
| **音源参数表格** | | |
| **id（int）** | **name（String）** | **参数说明** |
| 0 | 本地音源 | 本地音源 |
| 1 | 外部音源 | 外部音源 |
| 3 | 蓝牙 | 蓝牙音源 |
| 7 | 在线音源 | 云音乐 |

|  |  |  |
| --- | --- | --- |
| **歌曲列表表格** | | |
| **id（long）** | **listName（String）** | **参数说明** |
| 25 | LOCAL\_LIST | 本地 |
| 26 | HIS\_LIST | 最近历史播放 |
| 27 | LOVE\_LIST | 收藏 |
| 1 | ALL\_LIST | 全部音乐 |
| 2 | INTERNAL\_LIST | 内存音乐 |
| 3 | EXTERNAL\_LIST | SD卡 |
| 4 | USB\_LIST | USB |
| 17 | SCENE\_LIGHT\_MUSIC | 轻音乐 |
| 18 | SCENE\_POPULAR | 流行 |
| 19 | SCENE\_CLASSICAL | 古典 |
| 20 | SCENE\_ROCK | 摇滚 |
| 21 | SCENE\_RECEIVE | 会客 |
| 22 | SCENE\_REPAST | 就餐 |
| 23 | SCENE\_ENTERTAINMENT | 娱乐 |
| 24 | SCENE\_RECREATION | 休闲 |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **分区状态表格** | | |
| **参数名称** | **参数类型** | **参数说明** |
| zoneName | String | 分区名称 |
| zoneNum | int | 分区号 |
| vol | int | 分区音量 |
| zoneOpen | boolean | 分区是否打开 |
| zoneMathCode | boolean | 话筒对码是否打开 |
| zoneChannel | List<ZoneChannelBean> | 声道状态（**ZoneChannelBean**参数说明请查看第**11**页《声道状态表格》） |

|  |  |  |
| --- | --- | --- |
| **声道状态表格** | | |
| **参数名称** | **参数类型** | **参数说明** |
| channelName | String | 声道名称 |
| channelNum | int | 声道位号  （左声道0，有声道1） |
| vol | int | 声道音量 |