





用法：（例 AUDIO\_DATA\_SOURCE\_VOICEDISPOSE 做音效处理）

## 一. 注册回调类

```
1. AVAudioCtrl avAudioCtrl = qavsdk.getAVContext().getAudioCtrl();
```

```
2. avAudioCtrl.registAudioDataCallback(AudioDataSourceType.AUDIO_DATA_SOURCE_VOICEDISPOSE, registAudioDataCompleteCallback);
```

1. 之后每 20ms 音频引擎就会调用注册时类 registAudioDataCompleteCallback 里的回调接口：

```
2. protected int onComplete(AudioFrame audioframe, int srcType ) ,
```

3. 针对 AUDIO\_DATA\_SOURCE\_VOICEDISPOSE ， 回调接口里 audioframe 携带：

4. 1. 20ms 的原始麦克风数据： public byte [] data;

5. 2. 数据长度： public int dataLen;

6. 3. PCM 数据采样率： public int sampleRate;

7. 4. PCM 数据声道数： public int channelNum;

8. 5. PCM bit 数： public int bits;

二、回调操作处理人声，音频引擎会 20ms 执行一次回调函数 `onComplete(AudioFrame audioframe, int srcType )`

```
1. private AVAudioCtrl.RegistAudioDataCompleteCallback registAudioDataCompleteCallback = new AVAudioCtrl.RegistAudioDataCompleteCallback() {
2.     protected int onComplete(AudioFrame audioframe, int srcType ) {
3.         if (srcType==AudioDataSourceType.AUDIO_DATA_SOURCE_VOICEDISPOSE) {
4.             synchronized (obj)
5.             {
6.                 /*****
7.                 对 audioframe 中 byte [] data 人声原始数据进行处理
8.                 可用 soundtouch 组件对 data 数据进行变声，但要保证 20ms 可处理完毕
9.                 *****/
10.                return AVErrror.AV_OK;
11.            }

```

```
12.         }
```

```
13.     }
```

```
14. }
```

三、取消处理，反注册

```
1. avAudioCtrl.unregisterAudioDataCallback(AudioDataSourceType.AUDIO_DATA_SOURCE_VOICEDISPOSE);
```

特殊需求：

1) 场景：利用 VOICEDISPOSE 接口做 mix，但 mix 数据为 44.1K 双声道（内部默认抛出的人声数据为 48K 双声道）

可对 AUDIO\_DATA\_SOURCE\_VOICEDISPOSE 做采样率设置：

```
1. AudioFrameDesc audio_desc = avAudioCtrl.getAudioDataFormat(AudioDataSourceType.AUDIO_DATA_SOURCE_VOICEDISPOSE, audio_desc);
```

```
2. audio_desc.sampleRate = 44100;
```

```
3. avAudioCtrl.setAudioDataFormat(AudioDataSourceType.AUDIO_DATA_SOURCE_VOICEDISPOSE, audio_desc);
```

2) 场景 2：需要缩小为一半音量

1. `float fVolume = avAudioCtrl.getAudioDataVolume(AudioDataSourceType.AUDIO_DATA_SOURCE_VOICEDISPOSE);`
2. `fVolume = fVolume * 0.5;`
3. `avAudioCtrl.setAudioDataVolume(AudioDataSourceType.AUDIO_DATA_SOURCE_VOICEDISPOSE, fVolume );`