

# 音频采集渲染 3A 处理一体库

## 一、简介

音频 3A 处理 (AEC、AGC、ANS) 是实时互动领域无法绕开的问题。其中 AEC 处理与音频的采集、渲染紧密相关，三者的合理耦合对 AEC 效果影响很大，这也对开发者提出了更高的要求。为了简化上层业务的工作量，我们将音频采集、渲染、3A 处理封装成一体库，对外提供非常易用的接口。该库有如下特点：

- A、AEC 尽量往底层靠，不受网络传输过程中的抖动、丢包等因素影响。
- B、提供 AEC 自动延时估计功能，适应不同的设备与场景。
- C、支持 8KHZ~48KHZ 采样率，单双声道（为了降低 CPU 开销，双声道合并为单声道处理，因此将丢失真实立体声效果）。
- D、仅 5 个 API 接口，方便集成。

## 二、API 说明

### 1、创建 SD3ACapRenderProcess 对象

```
void* SD3ACapRenderProcess_New(const char* outputPath, int outputLevel);
```

**参数：**

@param: outputPath 表示日志存放路径，支持相对路径和绝对路径，若目录不存在将自动创建

@param: outputLevel 表示日志输出的级别，只有等于或者高于该级别的日志输出到文件，取值范围参考 3A\_CAP\_RENDER\_LOG\_OUTPUT\_LEVEL

**返回值：**返回模块指针，为 NULL 则失败

### 2、销毁 SD3ACapRenderProcess 对象

```
void SD3ACapRenderProcess_Delete(void** pp3AProcess);
```

**说明：**使用者应该做好与其他 API 之间的互斥保护，调用本 API 后模块指针将被设置为 NULL。

**参数：**

@param pp3AProcess: 模块指针指针

### 3、开始启动 SD3ACapRenderProcess

```
BOOL SD3ACapRenderProcess_Start(void* p3AProcess, int nCapDeviceID, int nSampleRate, int nChannelNum, BOOL bEnableAec, BOOL bEnableAgc, BOOL bEnableAns, Output3AProcessedCaptureData pfOutput3ACallback, void* pObject);
```

说明：双向互动时，建议开启 AEC\ANS，关闭 AGC，防止对端渲染音量过大而影响 AEC 效果。

参数：

@param p3AProcess: 模块指针

@param nCapDeviceID: 采集音频设备 ID，-1 为默认设备

@param nSampleRate: 输入待处理数据采样率，比如 32000、44100

@param nChannelNum: 输入待处理数据声道数，比如 2、1

@param bEnableAec: 是否使能 AEC

@param bEnableAgc: 是否使能 AGC

@param bEnableAns: 是否使能 ANS

@param pfOutput3ACallback: 经过 3A 处理后的音频采集数据输出回调接口

@param pObject: 上述输出回调接口的透传指针

返回值：TRUE 成功，FALSE 则失败

### 4、停止 SD3ACapRenderProcess

```
void SD3ACapRenderProcess_Stop(void* p3AProcess);
```

参数：

@param p3AProcess: 模块指针

### 5、播放数据，要求采样率、声道数与创建时一致

```
int SD3ACapRenderProcess_Play(void* p3AProcess, unsigned char *pucData, int nLen);
```

参数：

@param p3AProcess: 模块指针

@param pucData: 待播放数据

@param nLen: 待播放数据大小

**返回值：**实际播放数据大小