

录制混流

1 功能简介

本地服务端录制 SDK 支持将房间内的所有用户的音视频流混合成一个画面再录制，集成 SDK 后，即可开始使用 SDK 的混流录制功能。

2 使用步骤

2.1 设置回调监听

在初始化之前需要先继承实现 `ICallback`，并设置回调监听。

- 接口原型：

```
ZEGO_API bool SetCallback(ICallback* pCB)
```

- 参数：

`pCB`：回调对象指针。

- 备注：

在初始化 SDK 前可以根据需要调用如下接口：

[SetLogDirAndSize](#) 设置日志文件路径和大小

[SetUseTestEnv](#) 设置是否使用测试环境

2.2 初始化 SDK

使用获取的 ZEGO AppID、AppSign 来初始化 SDK，不初始化 SDK 将导致 SDK 相关功能异常。若没有 AppID、AppSign，请在 [即构管理控制台](#) 申请；如何申请请查看 [获取 AppID](#)

和 AppSign 指引

- 接口原型:

```
ZEGO_API bool InitSDK(unsigned int uiAppID, unsigned char* pBufAppSignature, int nSignatureSize)
```

- 参数:

uiAppID: ZEGO 派发的数字 ID, 各个开发者的唯一标识。

pBufAppSignature: ZEGO 派发的签名, 用来校验对应 AppID 的合法性。

nSignatureSize: 签名长度 (字节)

- 异步回调接口:

```
class ICallback
{
public:
    virtual void OnInitSDK(int errorCode) {}
}
```

errorCode == 0 表示初始化成功。

2.3 登录房间

在初始化成功之后, 录制之前需要先登录房间。

- 接口原型:

```
ZEGO_API bool LoginRoom(const char* pszRoomID)
```

- 参数:

pszRoomID: 房间 ID, 正在推流的房间 ID。

- 异步回调接口:

```

class ICallback
{
public:
    virtual void OnLoginRoom(int errorCode, const char *pszRoomID) =
0;
}

```

2.4 开始录制

1. 此接口必须在初始化成功后调用。
2. 建议在登录房间后并收到 [OnStreamUpdate](#) 流信息更新回调后，根据增加流信息来开启录制。
3. 可在开始录制之前调用 [GetMaxRecordCount](#) 获取 SDK 支持的最大同时录制流数，根据其值处理同时录制的流数量。

- 接口原型：

```
ZEGO_API bool StartRecordMixStream(ZegoMixStreamRecordConfig* pConfig)
```

- 参数：

pConfig：混流录制配置，请参考 [ZegoMixStreamRecordConfig](#)。

- 异步回调接口：

```

class ICallback
{
public:
    virtual void OnStreamRecordBegin(const char* pszStreamID, const c
har *pszPathAndName) = 0;
    virtual void OnMixStreamRecordUpdate(const char** ppszStreamID, i
nt nStreamCount) = 0;
}

```

录制启动成功会收到 [OnStreamRecordBegin](#) 回调，成功拉到有音视频数据的流时会收到 [OnMixStreamRecordUpdate](#) 回调。

- 备注：

若需要同时录制单流和混流，在开始录制之前调用 [EnableSingleAndMixRecordMode](#) 启用单流混流同时录制模式；**此种模式比较消耗系统资源，且只能先启动混流录制，然后再启动单流录制；结束混流录制时也会同时结束所有的单流录制。**此种模式下，SDK 有一个限制，混流录制会混入所有单流录制正在录制的流，即使没有将单流录制中的某单流作为混流的输入流，若此单流有音频，这个音频也会被混流录制给录制下来。

2.5 更新混流配置

当需要增加输入流、原输入流变更或者混流输出配置变更时，重新构造混流配置并调用更新混流配置接口。

- 接口原型：

```
ZEGO_API bool UpdateInputStreamConfig(ZegoStreamConfig* pInputStream, int nInputStreamCount)
```

- 异步回调接口：

```
class ICallback
{
public:
    virtual void OnMixStreamRecordUpdate(const char** ppszStreamID, int nStreamCount) = 0;
}
```

混流成功后会返回包含参与混流录制的单流 ID 列表的回调，只包括成功拉到数据的流 ID。**注意：参与混流录制的流发生变化比如开始录制、录制中流发生异常等都会收到此回调。**

2.6 停止录制

- 接口原型：

```
ZEGO_API bool StopRecordMixStream()
```

- 异步回调接口：

```

class ICallback
{
public:
    virtual void OnStreamRecordEnd(const char* pszStreamID, const cha
r *pszPathAndName, RecordEndReason reason) = 0;
}

```

- 备注：

建议在不需要再录制的时候，做以下处理：

1. 调用 [LogoutRoom](#) 退出房间。
2. 调用 `LIVEROOM::SetCallback(nullptr)`；去掉回调监听。
3. 调用 [UnInitSDK](#) 反初始化 SDK。

3 API 参考列表

方法	描述
SetUseTestEnv	设置是否使用测试环境，开发阶段可使用测试环境测试
SetCallback	设置回调监听
InitSDK	初始化 SDK
LoginRoom	登录房间
GetMaxRecordCount	获取 SDK 支持的最大同时录制流数
EnableSingleAndMixRecordMode	设置是否启用单流混流同时录制模式，启用该模式比较消耗系统资源
StartRecordMixStream	开始混流录制
UpdateInputStreamConfig	更新混流录制输入流配置
StopRecordMixStream	停止混流录制
SetMuxerOutType	设置录制数据输出方式

SetLogDirAndSize	设置 SDK log 路径和单个 log 文件大小
SetConfig	设置配置信息
SetMuxerCacheSize	设置文件落地前的缓存大小(64KB~1MB)
GetSDKVersion	获取 SDK 版本
UploadLog	上传 log 文件到 ZEGO 后台
UnInitSDK	反初始化 SDK
LogoutRoom	退出房间