

# 媒体播放器

---

## 1 功能简介

媒体播放器组件提供播放音视频媒体文件的能力，并且支持将播放的媒体文件的音画数据推流出去。

### 1.1 应用场景

- 播放测试音频

可以使用媒体播放器播放测试音频，验证音频播放设备是否工作正常。

- 背景音乐

使用媒体播放器播放音乐，媒体播放器支持将播放出来的音频推流出去，让远端听到背景音乐。

- 在线 KTV

主唱使用媒体播放器播放 MV 或其它类型歌曲，主唱随音乐唱歌，通过简单的配置就能让观众欣赏主唱 K 歌。结合媒体次要信息，还可以实现歌词同步显示。

### 1.2 支持格式

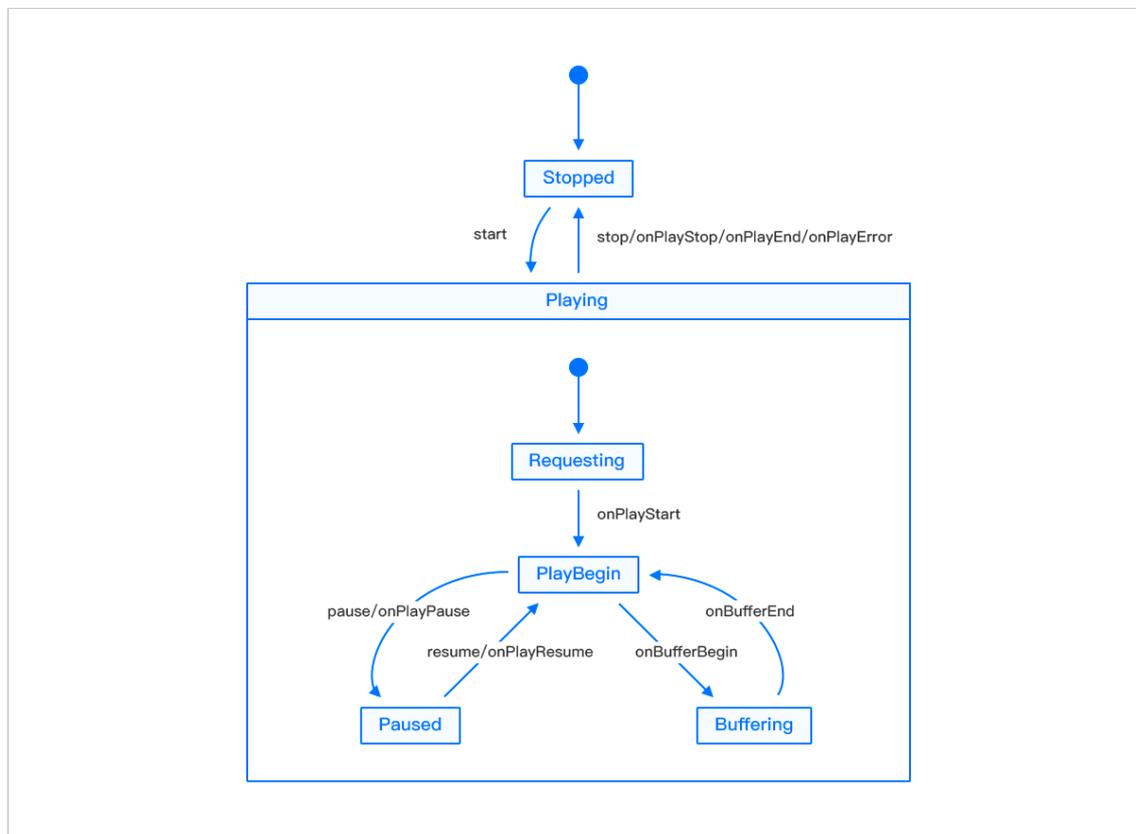
媒体播放器支持 MP3、MP4 格式的文件，并支持播放 iOS 媒体库中的音乐（[ipod-library://](#)）。除了本地媒体文件，媒体播放器还支持播放在线 MP3 与 MP4 格式的文件（[http://](#)）。

### 1.3 注意

媒体播放器功能需要使用特定的、包含 [ZegoMediaPlayer](#) 功能的 SDK，请下载 [ZegoLiveRoom MediaPlayer iOS](#)

## 1.4 媒体播放器业务流程

媒体播放器业务状态流转如下图所示：



1. 起始状态为 Stopped，调用了媒体开始播放接口后，进入 Playing 状态
2. 在 Playing 状态内部存在多个子状态
3. Requesting 表示提交播放请求，但还没有播放成功
4. PlayBegin 表示开始播放了
5. Paused 表示调用了暂停接口，播放暂停了
6. Buffering 表示正在缓冲
7. 当调用了 stop，或是收到 onPlayEnd、onPlayError 时，播放结束，进入 Stopped 状态。

## 2 下载示例源码

互动视频 Demo 源码下载地址：[互动视频示例 Demo\\_iOS](#)

[码云下载地址\(国内推荐\)](#)

[阿里云下载地址\(国内推荐\)](#)

[github下载地址](#)

示例 Demo 包含了基础专题和进阶专题，展示了如何使用 SDK API，开发者可参考其用法来实现自己的业务。示例 Demo 涵盖了 SDK 的大部分功能，开发者也可在示例 Demo 中查找 ZEGO SDK 更多的进阶功能，测试其功能和性能，以实现特定的需求。

媒体播放器专题源码请查看 [/src/LiveRoomPlayground-iOS/MediaPlayerUI](#) 目录下的源文件，该模块展示了使用媒体播放器播放音视频文件、并将他们推流的方法。使用媒体播放器组件，必须引入头文件 [zego-api-mediaplayer-oc.h](#)。

## 3 使用步骤

### 3.1 初始化播放器

- 接口原型：

```
/**
 初始化

  @param type @see MediaPlayerType
  @param index sdk提供多个播放器实例，通过index可以指定获取的是哪个播放器实例 @see Zego
  MediaPlayerIndex
  @return 播放器对象
 */
- (instancetype) initWithPlayerType:(MediaPlayerType) type playerIndex:(ZegoMediaPlayerIndex) index;
```

- 调用示例：

```
ZegoMediaPlayer *mediaPlayer1;
    ZegoMediaPlayer *mediaPlayer2;
    mediaPlayer1 = [[ZegoMediaPlayer alloc] initWithPlayerType:MediaPlayerTypeAux playerIndex:ZegoMediaPlayerIndexFirst];
    mediaPlayer2 = [[ZegoMediaPlayer alloc] initWithPlayerType:MediaPlayerTypePlayer playerIndex:ZegoMediaPlayerIndexSecond];
```

- 备注:

1. 在初始化播放器之后可以使用 `-setPlayerType:` 修改播放器类型。
2. 若不需要多实例媒体播放器可直接使用单实例媒体播放器，使用 `-initWithPlayerType:` 初始化媒体播放器。

### 3.2 设置播放器事件回调

设置播放器事件回调，就可以接收到开始播放，结束播放，播放错误等事件回调。

```
- (void)setEventWithIndexDelegate:(id<ZegoMediaPlayerEventWithIndexDelegate>)delegate;
```

回调代理协议请详细查看 [ZegoMediaPlayerEventWithIndexDelegate](#)

- 备注：若是单实例媒体播放器，使用 `-setDelegate:` 设置播放器事件回调监听。

### 3.3 设置视频渲染 View

当使用播放器播放视频资源时，可以使用 `-setView:` API 设置视频的显示视图。

```
/**  
 设置显示视频的view  
  
 @param view 播放的控件  
 */  
- (void)setView:(UIView *)view;
```

### 3.4 开始播放/停止播放

```
/**  
 开始播放  
  
 @param path 媒体文件的路径  
 @param repeat 是否重复播放  
 */  
- (void)start:(NSString *)path repeat:(BOOL)repeat;  
  
/**
```

```
停止播放
*/
- (void)stop;
```

1. 指定的播放文件可以是本地文件的路径，也可以是在线文件 url，比如 `http://your.domain.com/your-movie.mp4`；repeat 参数设置为 YES 时，表示开启循环播放。
2. 如果播放成功将回调 `onPlayStart` 事件，如果失败则回调 `onPlayError` 事件。

### 3.5 暂停播放/恢复播放

```
/**
 暂停播放
*/
- (void)pause;

/**
 恢复播放
*/
- (void)resume;
```

在开始播放后，可以调用 `pause` 暂停当前媒体播放，并在成功时回调 `onPlayPause` 事件；在预加载完成或暂停播放后，可以调用 `resume` 恢复播放，并在成功时回调 `onPlayResume` 事件。

### 3.6 反初始化播放器

和初始化接口对应，在不需要播放器时，应该调用反初始化播放器接口。

```
/**
 释放播放器
*/
- (void)uninit;
```

## 4 进阶：将播放器播放的音视频推流出去

`ZegoMediaPlayer` 支持将音视频都推流出去，以下分别介绍播放器的音频、视频的推流方式。  
推流参考文档：[快速集成-推流](#)

## 4.1 将音频推流出去

在创建播放器 `-initWithPlayerType:` 或修改播放器类型 `-setPlayerType:` 设置播放器为 `MediaPlayerTypeAux` 类型，就可以实现将播放器的音频混音到推流。

## 4.2 将视频推流出去

要将播放器的视频推流出去，需要使用视频外部采集方式将视频数据混入推流中。具体使用步骤为：

1. 设置播放器播放视频帧回调
2. 给 SDK 设置视频外部采集工厂
3. 开始推流
4. 给 SDK 塞播放器视频帧数据

### 设置播放器播放视频帧回调

通过方法 `-setVideoPlayDelegate:format:` 设置播放器播放视频帧回调和回调视频帧的格式，方法说明如下：

```
/**
 * 设置视频帧数据回调
 *
 * @param delegate 回调
 * @param format 需要返回的视频帧数据格式，@see ZegoMediaPlayerVideoPixelFormat
 */
- (void)setVideoPlayDelegate:(id<ZegoMediaPlayerVideoPlayDelegate>)delegate format:(ZegoMediaPlayerVideoPixelFormat)format;
```

`ZegoMediaPlayerVideoPixelFormat` 现支持的视频帧数据格式如下：

```
typedef NS_ENUM(NSUInteger, ZegoMediaPlayerVideoPixelFormat) {
    ZegoMediaPlayerVideoPixelFormatUnknown = 0,
    ZegoMediaPlayerVideoPixelFormatI420 = 1,
    ZegoMediaPlayerVideoPixelFormatNV12 = 2,
```

```

ZegoMediaPlayerVideoPixelFormatNV21    = 3,
ZegoMediaPlayerVideoPixelFormatBGRA32  = 4,
ZegoMediaPlayerVideoPixelFormatRGBA32  = 5,
ZegoMediaPlayerVideoPixelFormatARGB32  = 6,
ZegoMediaPlayerVideoPixelFormatABGR32  = 7,
};

```

在回调对象中实现 `ZegoMediaPlayerVideoPlayDelegate` 中的方法，其中 `-onPlayVideoData:size:format:` 回调的是 RGB 类的数据，`-onPlayVideoData2:size:format:` 回调的是 YUV 类的数据，说明如下：

```

/**
 视频帧数据回调接口
 当格式为ARGB32/ABGR32/RGBA32/BGRA32，数据通过onPlayVideoData回调。
 当格式为I420/NV12/NV21，数据通过onPlayVideoData2回调。
 其他非法格式都判定为I420
 */
@protocol ZegoMediaPlayerVideoPlayDelegate <NSObject>

@optional

/**
 视频帧数据回调，格式为ARGB32/ABGR32/RGBA32/BGRA32

 @param data 视频帧原始数据
 @param size 视频帧原始数据大小
 @param format 视频帧原始数据格式
 @note 同步回调，请不要在回调中处理数据或做其他耗时操作
 */
- (void)onPlayVideoData:(const char *)data size:(int)size format:(struct ZegoMediaPlayerVideoDataFormat)format;

/**
 视频帧数据回调，格式为I420/NV12/NV21

 @param data 视频帧原始数据
 @param size 视频帧原始数据大小
 @param format 视频帧原始数据格式
 @note 同步回调，请不要在回调中处理数据或做其他耗时操作
 */
- (void)onPlayVideoData2:(const char **)data size:(int *)size format:(struct ZegoMediaPlayerVideoDataFormat)format;

@end

```

回调的设置和回调方法的实现可以参考 Demo 中的实现。

## 给 SDK 设置视频外部采集工厂

在 Init SDK 前，设置 SDK 使用视频外部采集。示例代码如下：

```
[ZegoLiveRoomApi setVideoCaptureFactory:self.externalVideoCaptureFactory];
```

视频外部采集，详见[视频外部采集](#)，或参考 Demo 的[视频外部采集](#)专题代码。

## 开始推流

必须先开始推流，才能将播放器的音视频数据推流出去。推流参考文档：[快速集成-推流](#)

## 给 SDK 塞播放器视频帧数据

根据设置的回调视频帧数据格式不同，在相应的回调方法中处理视频帧，然后将数据塞给 SDK，实现将播放器的视频推流出去。其中 `-onPlayVideoData:size:format:` 回调的是 RGB 类的的数据，`-onPlayVideoData2:size:format:` 回调的是 YUV 类的的数据。

在示例 Demo 中，回调的数据处理成 `CVPixelBufferRef` 类型，然后通过视频外部采集工厂塞给 SDK。Demo 代码参考如下：

```
// 回调 RGB 类的的数据
- (void)onPlayVideoData:(const char *)data size:(int)size format:(struct ZegoMediaPlayerVideoDataFormat)format {
    // 注意：不要在另外的线程处理 data，因为 data 可能会被释放
    Weakify(self);
    [self.playerVideoHandler convertRGBCategoryDataToPixelBufferWithVideoData:data size:size format:format completion:^(ZGMediaPlayerVideoDataToPixelBufferConverter * _Nonnull converter, CVPixelBufferRef _Nonnull buffer, CMTime timestamp) {
        Strongify(self);
        [self postMediaPlayerVideoFrameData:buffer presentationTimeStamp:timestamp format:format];
    }];
}

// 回调 YUV 类的的数据
- (void)onPlayVideoData2:(const char **)data size:(int *)size format:(struct ZegoMediaPlayerVideoDataFormat)format {
    // 注意：不要在另外的线程处理 data，因为 data 可能会被释放
    Weakify(self);
    [self.playerVideoHandler convertYUVCategoryDataToPixelBufferWithVideoData:data size:size format:format completion:^(ZGMediaPlayerVideoDataToPixelBufferConverter * _Nonnull
```

```

ull converter, CVPixelBufferRef _Nonnull buffer, CMTimestamp timestamp) {
    Strongify(self);
    [self postMediaPlayerVideoFrameData:buffer presentationTimeStamp:timestamp for
mat:format];
    }];
}

```

按照以上流程下来，即可实现将 `ZegoMediaPlayer` 的视频数据的推流出去。

## 5 错误码

播放器 `onPlayError` 返回的错误码如下：

错误码	说明
ErrorCode = -1	文件格式不支持
ErrorCode = -2	配置文件获取失败，路径不存在
ErrorCode = -3	文件无法解码
ErrorCode = -4	文件中没有可播放的音视频流
ErrorCode = -5	文件解析过程中出现错误
ErrorCode = -6	文件播放过程中出现丢帧，视频解码性能不够。 <b>提示信息，不影响播放流程</b>

## 6 Q&A

Q1: 播放器得到的音频混入推流后，采集音频数据回调

( `[ZegoLiveApiAudioRecordDelegate - onAudioRecord:sampleRate:numOfChannels:bitDepth:type:]` ) 包含播放器的音频吗？

答：包含。

Q2: 可以播放网络视频吗?

答: 支持 HTTP 协议的 mp3/mp4 网络资源。

Q3: 支持哪些格式的文件?

答: MP3、MP4、iTunes 媒体库文件。

Q4: 使用播放器混音后, 还能使用原有的混音接口吗?

答: 播放器混音推流并不影响原有混音接口功能, 二者可以同时并存。

Q5: 播放器调用了 `start` 方法以后没有开始播放?

答: `load` 播放器以后只能通过 `resume` 方法恢复, `load` 以后调用 `start` 方法会被跳过。如果想要调用 `start` 方法有效的话, 不能调用 `load` 方法。

Q6: 为什么收不到 `onPlayError` 回调?

答: 在 `mediaPlayer` 初始化以后设置 `mediaPlayer delegate`。

Q7: 播放结束以后怎么怎么再次播放, 不是让视频自动播放?

答: 调用 `start` 方法。

Q8: 为什么收不到播放进度 `onProcessInterval` 回调?

答: 需要通过 `setProcessInterval` 设置时间间隔。

Q9: 我调用了 `seek` 方法以后是不是必须 `resume` ?

答: `seek` 以后是自动播放的, 无需再次调用 `resume`。

Q10: 同一个实例能不能 `load` 两次 url, 还是说 `load` 第二个 url 之前必须先 `stop`?

答: 同一个实例需要 `stop` 之后才能再次 `load`, 不能连续 `load` 两次。