

- [1.播放器](#)
- [2.设置渲染画面](#)
- [3.播放参数配置](#)
- [4.播放控制](#)
- [5.回调](#)

1.播放器

播放器管理器的对外核心类为 `PLVPlayerManager`，可以通过 SDK 对象的 `playerManager` 属性对其进行访问，`playerManager` 内部的 `mainMediaPlayer` 为视频播放的播放器。

2.设置渲染画面

播放器通过 `surface` 类型的 `XComponent` 组件进行渲染，`libraryname` 为 `plvplayer_xcomponent`，您可以通过 `startMainMediaPlayerWithComponent` 方法将渲染画面设置给播放器：

```
XComponent({
  id: `plvlw_video_xcomponent`,
  type: "surface",
  libraryname: "plvplayer_xcomponent"
}) {
}

.onLoad((component) => {
  // 设置渲染画面并开始播放
  this.sdk.playerManager.startMainMediaPlayerWithComponent(component!)
})
```

3.播放参数配置

通过调用 `mainMediaPlayer` 的 `setPlayerOption()` 方法设置播放参数

```
/**
 * 设置播放参数
 */
setPlayerOption(options: PLVMediaPlayerOption[])
```

`PLVMediaPlayerOptionEnum` 类中提供了一些常用的播放参数，您可以直接引用其中的常量，例如：

```
// 开启精准seek的参数
PLVMediaPlayerOptionEnum.ENABLE_ACCURATE_SEEK.value("1")
```

对于重复设置的参数，新设置的参数会覆盖旧的参数；如果想要清空参数，可以在 value 字段中传入空字符串

4.播放控制

播放器 `mainMediaPlayer` 提供了一系列的播放控制接口，例如：

```
/**
 * 开始播放
 */
start(): void;

/**
 * 暂停播放
 */
pause(): void;

/**
 * 跳转播放进度到指定位置
 * @param position 指定位置，单位：毫秒
 */
seek(position: number): void;
```

更多控制操作可以参考 `IPLVMediaPlayer` 以及它的父接口 `IPLVMediaPlayerControl`

5.回调

播放器的状态、事件回调可以通过回调注册中心进行监听，包括：

- `IPLVMediaPlayerBusinessListenerRegistry`：播放器业务回调注册中心
- `IPLVMediaPlayerEventListenerRegistry`：播放器事件回调注册中心
- `IPLVMediaPlayerStateListenerRegistry`：播放器状态回调注册中心

以监听播放/暂停状态为例，可以通过以下方式进行监听：

```
const playingState: MutableState<PLVMediaPlayerPlayingState> = player.getStateListenerRegistry().playingState;
const observer = playingState.observe((state: PLVMediaPlayerPlayingState) => {
  // 处理逻辑
  const isPlaying = state === PLVMediaPlayerPlayingState.PLAYING;
});
```