

# 趣看云视频直播点播服务

## 视频播放器SDK iOS开发指南

□ □ □ 1.0

2016.05



## SDK 概述

本视频播放器 SDK 是趣看官方推出的跨平台的播放器软件开发工具包，其中 iOS SDK 为 iOS 开发者提供简单、快捷的接口，帮助开发者在 iOS 平台上快速开发播放器应用。

SDK 对于当前主流视频格式的网络视频流和本地视频文件都能够很好的支持，兼容性和稳定性以及功能上较之系统播放器都有很大的提高。

SDK 支持 iOS 7.0 及以上版本。

## SDK 功能说明

SDK 以方便开发者高效创建媒体播放应用为目标，具有以下的特色功能：

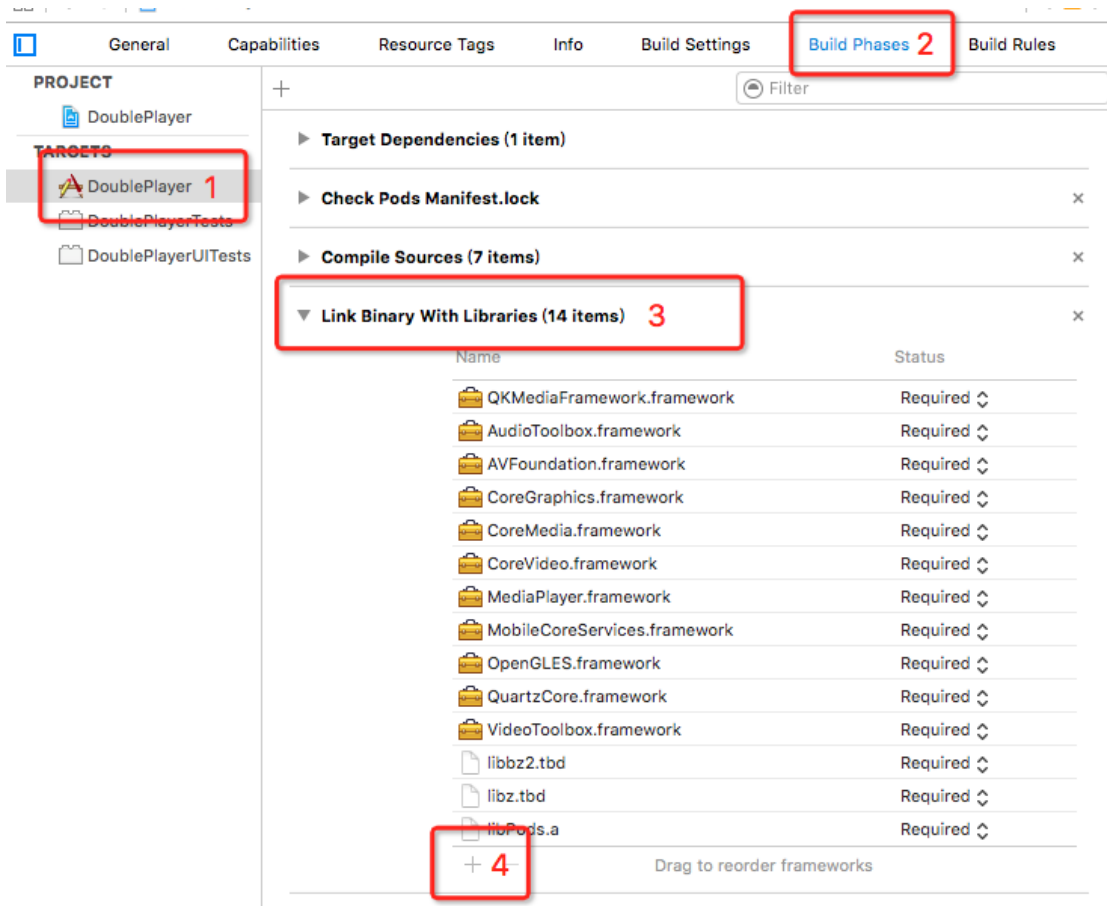
- **快速、灵活接入：**利用 iOS SDK 提供的 API 接口可以轻松接入播放器
- **流媒体格式支持：**支持 HTTP、RTMP、HDL 协议，支持 HLS 协议的 M3U8 文件
- **本地媒体格式支持：**支持 MP4、FLV、MP3 等格式
- **硬件解码支持：**iOS 8.0 及以上支持硬件解码，极大的降低了 CPU 的消耗

## 开发准备

- 下载视频播放器 iOS SDK
- 将下载的播放器 SDK 开发包解压，将静态库都导入工程。关于静态库的导入，以下为特被说明的例外情况：
  - 如果您同时在使用直播推流 SDK，则 libs 目录中的同名库文件只能导入一次。例如在导入直播推流 SDK 时包含了 librtmp.a，在导入播放器 SDK 时就不能再包含 librtmp.a 了。
  - 如果您同时在使用云信 NIMSDK，为了避免可能的静态库冲突，请不要导入 libs 中的 libyuv.a 和 libopenh264.a。

### 添加依赖框架

SDK 运行需要添加相关依赖框架，在 TARGETS -> Build Phases-> Link Binary With Libraries 中添加相关依赖框架。添加过程下图所示：



需要添加的所有依赖库：

- VideoToolbox.framework
- QuartzCore.framework
- OpenGL.framework
- MobileCoreServices.framework
- MediaPlayer.framework
- libz.dylib
- libbz2.dylib
- CoreVideo.framework
- CoreMedia.framework
- CoreGraphics.framework
- AVFoundation.framework
- AudioToolbox.framework

## 实现媒体播放功能

SDK 的 API 形式与 MediaPlayer 保持一致。方便开发者可以快速的接入，详细的流程可参见 demo。下面介绍基本的播放器开发过程：

## 设置播放属性

创建 QKFFoptions，在播放器初始化前配置相关参数。

```
QKFFoptions *options = [QKFFoptions optionsByDefault];
```

- 设置播放延迟

```
[options setOptionIntValue:250 forKey:@"max_delay"  
ofCategory:kQKFFoptionCategoryFormat];
```

- 设置硬编码，只支持 ios8 以上设备，1 使用硬编码，0 不使用。

```
[options setPlayerOptionIntValue:1 forKey:@"videotoolbox"];
```

## 创建播放器实例

创建 QKFFMoviePlayerController 实例，并进行相关初始化工作

```
self.player = [[QKFFMoviePlayerController alloc] initWithContentURL:self.url withOptions:options];
```

self.url 是待播放的网络视频流或本地文件的路径。

## 配置播放器相关参数

- 设置画面显示模式，默认按原始大小进行播放

```
self.player.scalingMode = MPMovieScalingModeNone;
```

- 设置视频文件初始化完成后是否自动播放，默认自动播放

```
self.player.shouldAutoplay = YES;
```

- 设置播放器切入后台后时暂停还是继续播放，默认暂停

```
[self.player setPauseInBackground:NO];
```

## 视频文件初始化

属性设置完成后，需调用 prepareToPlay 接口对视频文件进行初始化：

```
[self.player prepareToPlay];
```

初始化完成后 SDK 会发 QKMediaPlaybackIsPreparedToPlayDidChangeNotification 通知，并将 isPrepareToPlay 设置为 YES。

如果 shouldAutoplay 已经设置成 YES，则会自动调用 play 进行播放，如果 shouldAutoplay 设置成 NO，则需要用户调用 play 进行播放。

## 播放视频

若 shouldAutoplay 设置成 NO，则在调用 prepareToPlay 接口后，需要调用 play 进行播放。

```
[self.player play];
```

注意，在第一次调 play 接口之前一定要调 prepareToPlay 接口。

## 暂停播放

在播放过程中可以调用 pause 接口暂停播放：

```
[self.player pause];
```

暂停以后可调用 play 恢复播放。

## 退出播放器

退出播放需要调用 shutdown 接口：

```
[self.player shutdown];
```

如果没有退出播放界面，在重新播放之前（如播放过程中切换 URL）需要将创建的 view 释放掉：

```
[self.player.view removeFromSuperview];  
[self.player stop];  
[self.player shutdown];
```

## 指定时间点播放

在拖动进度条到某一时间点播放或者启动时从某一时间点播放可以调用 setCurrentPlaybackTime: 接口：

```
[self.player setCurrentPlaybackTime:currentPlayTime];
```

注意，该接口一定要在收到 `NELivePlayerDidPreparedToPlayNotification` 通知后或者当 `isPrepareToPlay` 值为 `YES` 时才可以调用。

## 截图功能

调用 `thumbnailImageAtCurrentTime` 接口来实现截图功能。

```
[self.player thumbnailImageAtCurrentTime];
```

截图结果以 `UIImage` 格式返回，开发者可根据自己的需求保存成 `jpg` 格式或 `png` 格式，并保存到指定位置。