

# iOS 点播UI SDK

---

变更记录: <https://git2.baijiashilian.com/open-ios/BaijiaYun/-/blob/master/BaijiaYun-CHANGELOG.md>

git SDK 链接: <https://git2.baijiashilian.com/open-ios/BaijiaYun.git>

git 源码链接: <https://git2.baijiashilian.com/open-ios/BJVideoPlayerUI.git>

UI SDK是开源的, 支持在源码基础上自定义修改, 参考[点播 UI SDK自定义修改文档](#)

## 概述

---

[BJVideoPlayerCore](#) (百家云iOS 点播 Core SDK, 以下简称“Core SDK”), 是一款支持iOS **点播/回放**视频播放/下载的SDK, 不带UI, UI需要开发者自主集成。适用于iOS 11.0+。**集成 2.2.0 或以上版本的 SDK 要求 Xcode 的版本至少为 11.0。**

[BJVideoPlayerUI](#) (百家云iOS 点播 UI SDK, 以下简称“点播UI SDK”), 是基于BJVideoPlayerCore封装的提供通用**点播**播放UI的SDK。适用于iOS 11.0+。旧版本的 3.x 适用于 iOS 10.0+。

## Demo

---

**Demo**是集成了点播UI SDK的项目，以源码的形式详细展示了其中播放器的使用，点播和回放视频的下载等功能。开发者下载或者Git Clone本项目，经过简单的修改即可运行**Demo**。

Demo项目结构：

1. **BJPlayManagerUI**
2. |— **Podfile**
3. |— **Podfile.lock**
4. |— **Pods**
5. |— **BJPlayerManagerUI.xcworkspace**
6. |— **BJPlayerManagerUI**
7. |   |— **PUAppdelegate.h**
8. |   |— **PUAppdelegate.m**
9. |   |— **PUMainViewController.h**
10. |   |— **PUMainViewController.m** (**Demo**首页入口)
11. |   |— **playvideo** (点播视频模块)
12. |      |— **PUPlayViewController.h** (点播视频播放控制器)
13. |      |— **PUPlayViewController.m**
14. |   |— **playback** (回放视频模块)
15. |      |— **BJPViewController.h** (回放视频播放控制器)
16. |      |— **BJPViewController.m**
17. |   |— **download** (下载模块)
18. |      |— **localVideo** (本地视频播放模块)
19. |      |— **PMLocalViewViewController.h** (本地视频播放控制器)
20. |      |— **PMLocalViewViewController.m**
21. |      |— **BJLDownload**
22. |      |— **BJLDownloadViewController.h**(视频下载控制器)
23. |      |— **BJLDownloadViewController.m**
24. |   |— **setting** (播放设置模块)
25. |      |— **PUVideoOptionsViewController.h** (播放设置控制器)

```
26.     └─ PUVideoOptionsViewController.m
27.     └─ PUPlayOptionsManager.h
28.     └─ PUPlayOptionsManager.m (配置
Manager)
29.     └─ extension (扩展类)
30.     └─ UIAlertView+bjp.h
31.     └─ UIAlertView+bjp.m
32.     └─ NSString+md5.h
33.     └─ NSString+md5.m
34.     └─ UIWindow+PUMotion.h
35.     └─ UIWindow+PUMotion.m
36.     └─ UIViewController+PUUtil.h
37.     └─ UIViewController+PUUtil.m
38.     └─ BJPlayerManagerUI-Prefix.pch
39.     └─ BJPlayerManagerUI-Info.plist
40.     └─ Base.lproj
41.         └─ LaunchScreen.storyboard
42.         └─ Main.storyboard
43.     └─ Images.xcassets (图片资源)
44.     └─ main.m
```

## Demo运行

- 在 Example 的工程目录下执行 **pod install**
- 使用 Xcode 打开 Example 文件夹下的 BJPlayerManagerUI.xcworkspace 文件
- 将demo中的证书和bundleID更改为开发者自己的数据, 即可运行.

## 温馨提示

如果在Demo中发现了bug或者发现了SDK的bug, 在告知我公司开发人员时, 请将设备型号, 设备系统版本, 硬件版本, 产生

**bug**的操作流程一并告知我公司开发人员，必要时**需要提供日志**。

## BJVideoPlayerUI SDK

点播UI SDK是在Core SDK的基础上对通用点播播放功能的集成，如果用户有更丰富的需求，请参考点播UI SDK中接口的使用方式，在理解点播SDK机制的前提下，根据自己的需求进行自定义开发。最新4.x版本适用于iOS12.x以上版本，3.x适用于iOS10.0+。

### 通用功能简介

- 支持横竖屏切换
- 视频播放/暂停/快进/快退，改变清晰度和倍速播放
- 横屏播放支持锁屏功能
- 视频播放界面左侧支持滑动调节亮度，右侧支持滑动调节音量
- 视频播放界面支持切换CDN节点

点播UISDK源码目录结构如下：

```
1. BJVideoPlayerUI
2. |— BJVideoPlayerUI.h
3. |— BJVideoPlayerUI.m
4. |— BJPUMacro.h
5. |— Theme
6.   |— BJPUTheme
7. |— Util
8.   |— BJPUVideoOptions (播放配置)
9.   |— BJPUAppearance
10.  |— MBProgressHUD+bjp
11. |— View
12.   |— BJPUMediaControlView
```

- 13. |— BJPUMediaSettingView
- 14. |— BJPUSignInModel
- 15. |— BJPUReloadView
- 16. |— BJPUSignInAlertViewController
- 17. |— BJPUSliderView
- 18. |— Swift
- 19. |— BJPUVideoCatalogueCell
- 20. |— CommentInputView
- 21. |— CommentListCell
- 22. |— CommentListViewController
- 23. |— QuizViewController
- 24. |— QuizViewController+widget
- 25. |— TripartiteLayoutOutlinesViewController
- 26. |— ViewController
- 27. |— BJPUViewController
- 28. |— BJPUViewController+media
- 29. |— BJPUViewController+observe
- 30. |— BJPUViewController+protected
- 31. |— BJPUViewController+reachability
- 32. |— BJPUViewController+ui

## SDK集成

### Podfile配置

- 设置source

1. source  
`'https://github.com/CocoaPods/Specs.git'`
2. source `'https://git2.baijiashilian.com/open-  
ios/specs.git'`

- 引入 `BJVideoPlayerUI`

```
1. pod 'BaijiaYun/BJVideoPlayerUI', '~> 4.0'
```

## 全局设置

- 配置ATS
  - 需要在info.plist里面增加 `NSAllowsArbitraryLoads = true`
- 引入头文件 `#import <BJVideoPlayerUI/BJVideoPlayerUI.h>`
- 1.4.5 之后的版本支持设置专属域名前缀
  - 例如专属域名为 `demo123.at.baijiayun.com`，则前缀为 `demo123`，参考[专属域名说明](#)。

```
1. #import <BJVideoPlayerCore/BJVAppConfig.h>
2. ....
3. [[BJVAppConfig sharedInstance]
   setPrivateDomainPrefix:privateDomainPrefix];
```

- 注意:BJVideoPlayerCore 和 BJVRequestTokenDelegate

在使用点播/回放的播放和下载之前，必须先设置

`BJVideoPlayerCore` 的代理

`id<BJVRequestTokenDelegate>`

`tokenDelegate`，并实现 `tokenDelegate` 必要的两个方法：

```
1. /* 用于在播放/下载中，客户需要调用completion
   提供某一videoID对应的点播视频播放token */
2. - (void)requestTokenWithVideoID:(NSString *)videoID
```

```

3.         completion:(void (^)(NSString
* _Nullable token, NSError * _Nullable
error))completion;
4.
5. /* 用于在播放/下载中，客户需要调用completion
提供某一回放课程对应的视频播放token */
6. - (void)requestTokenWithClassID:(NSString
*)classID
7.         sessionID:(nullable NSString
*)sessionID
8.         completion:(void (^)(NSString
* _Nullable token, NSError * _Nullable
error))completion;

```

具体请参考Demo的 `PUAppDelegate.m` 文件中 `BJVRequestTokenDelegate` 的使用方法。

## 播放器使用

- 使用 `BJPUVideoOptions` 初始化 `BJPUViewController`

```

1. - (instancetype)initWithVideoOptions:
    (BJPUVideoOptions *)videoOptions;

```

- `BJPUVideoOptions` 的详细内容如下：

```

1. @property (nonatomic, assign)
    BJVPlayerType playerType; // 播放器类
    型
2. @property (nonatomic, assign) BOOL
    autoplay; // 自动播放
3. @property (nonatomic, assign) BOOL
    sliderDragEnabled; // 能否拖动进度条

```

4. `@property (nonatomic, assign) BOOL playTimeRecordEnabled;` // 开启记忆播放
5. `@property (nonatomic, assign) BOOL encryptEnabled;` // 开启加密
6. `@property (nonatomic, assign) BOOL backgroundAudioEnabled;` // 开启后台播放，必须在工程的 `background modes` 中添加 `audio` 才会生效
7. `@property (nonatomic, assign) BOOL remoteControlEnabled;` // 开启远程控制
8. `@property (nonatomic, strong) UIImage *remoteControllImage;` // 远程控制图片
9. `@property (nonatomic, assign) NSTimeInterval initialPlayTime;` // 起始播放时间
10. `@property (nonatomic, assign) NSInteger maxWatchTime;` // 点播观看最大可拖拽时间，单位秒，时间点之前可以拖动，之后不可以拖动，取值大于0有效
11. `@property (nonatomic, assign) BOOL isLandscapeLayout;` // 默认布局是否是横屏，默认为 YES
12. `@property (nonatomic, assign) BOOL autoPlayNextAlbumVideo;` // 点播专辑是否自动播放下一个视频，默认 YES
13. `@property (nonatomic, assign) BOOL externalAlnumShow;` // 外部控制是否展示合集按钮
- 14.
15. `/* 开启画中画`
16. `note:` 当前的播放器类型为 `BJVPlayerType_AVPlayer` 且 `backgroundAudioEnabled == NO` 时，设置 `pictureInPictureEnabled` 才有效



```
17. note: 当设置 backgroundAudioEnabled 为  
    YES 时，内部会把 pictureInPictureEnabled  
    置为 NO  
18. */  
19. @property (nonatomic, assign) BOOL  
    pictureInPictureEnabled;  
20.  
21. /* 偏好清晰度列表  
22. #discussion 优先使用此列表中的清晰度播放  
    在线视频，优先级按数组元素顺序递减  
23. #discussion 列表元素为清晰度的标识字符  
    串，现有标识符：low（标清），high（高  
    清），superHD（超清），720p，1080p，  
    audio（纯音频），可根据实际情况动态扩展  
24. #discussion 此设置对播放本地视频无效  
25. */  
26. @property (nonatomic, strong)  
    NSArray<NSString*>  
    *preferredDefinitionList;  
27.  
28. /*  
29. 能否拖动进度条到未观看过的位置  
30.  
31. #discussion 开启之后，拖动进度条会弹框提  
    示  
32. #discussion 可自定义设置弹框的文字提示标  
    题，最大 16 字符  
33. #discussion 可自定义设置弹框的文字提示内  
    容，最大 48 字符  
34. #discussion 可自定义设置弹框的 icon 和按钮  
    文本，按钮文本最大 20 字符  
35. */  
36. @property (nonatomic, assign) BOOL  
    sliderAdvanceEnabled;  
37. @property (nonatomic, strong)  
    BJPUSkipPlayModel
```

```
*sliderDragStopPlayModel;
38.
39. /*
40. 试听时间，单位秒
41.
42. #discussion 开启之后，拖动进度条会弹框提示
43. #discussion 可自定义设置弹框的文字提示标题，最大 16 字符
44. #discussion 可自定义设置弹框的文字提示内容，最大 48 字符
45. #discussion 可自定义设置弹框的icon 和按钮文本，按钮文本最大 20 字符
46. */
47. @property (nonatomic, assign) NSInteger
    auditionTime;
48. @property (nonatomic, strong)
    BJPUStopPlayModel
    *auditionTimeStopPlayModel;
49.
50. /*
51. 自定义水印
52. #discussion 自定义优先级高于原配置项
53. */
54. @property (nonatomic, strong)
    BJVCustomWaterMarkModel
    *waterMarkModel;
55.
56. /* 第三方用户名和编号
57. #discussion 用于上报统计
58. */
59. @property (nonatomic, strong) NSString
    *userName;
60. @property (nonatomic, strong) NSString
    *userNumber;
```

- 根据需要设置block

```
1. /* 点播视频播放页面退出回调，nil 则默认直接退出，销毁页面 */
2. @property (nonatomic, copy) void (^
   _Nullable cancelCallback)(void);
3.
4. /* 视频画面锁定回调，回调可以设置画面锁之后是否允许设备屏幕旋转 */
5. @property (nonatomic, copy) void (^
   _Nullable screenLockCallback)(BOOL locked);
```

- 添加播放器view到对应的页面上

```
1. self.playerUIVC = [[BJPUViewController
   alloc] initWithVideoOptions:
   [PUPlayOptionsManager getVideoOptions]];
2. [self
   addChildViewController:self.playerUIVC];
3. [self.playerUIVC
   didMoveToParentViewController:self];
4.
5. __weak typeof(self) weakSelf = self;
6. /* 退出回调 */
7. [self.playerUIVC setCancelCallback:^(
8.     __strong __typeof__(weakSelf) strongSelf
   = weakSelf;
9.     [strongSelf.navigationController
   pushViewControllerAnimated:YES];
10. }];
11.
12. /* 锁屏回调 */
13. [self.playerUIVC
   setScreenLockCallback:^(BOOL locked) {
```

```

14.     PUAppDelegate *appDelegate =
        (PUAppDelegate *)[UIApplication
        sharedApplication].delegate;
15.     appDelegate.allowRotation = !locked;
16.   }];
17.
18.   [self.view
        addSubview:self.playerUIVC.view];

```

- 播放视频

```

1.  /* 播放在线视频 */
2.  [self.playerUIVC playWithVid:_vid
        token:_token];
3.
4.  /* 播放本地视频BJVDownloadItem */
5.  [self.playerUIVC
        playWithDownloadItem:downloadItem];

```

## 点播评论

SDK内部根据配置会控制点播评论是否展示，外部对接只需要设置 `BJVCommentManager` 的 `userToken`。

建议在每次播放之前确认是否有 `userToken`，如果没有再次设置。

具体 `userToken` 的获取参考[获取点播评论token接口](#)

```

1. @interface BJVCommentManager: NSObject
2. @property (nonatomic, readonly, nullable)
        NSString *userToken;
3. + (instancetype)sharedInstance;
4. - (void)updateUserToken:(NSString *)userToken;

```

# 定制UI元素(BJPUTheme.h)

```
1.
2. /* 默认状态下文本颜色 */
3. + (void)setDefaultTextColor:(UIColor *)color;
4. /* 选中状态的文本颜色 */
5. + (void)setHighlightTextColor:(UIColor *)color;
6. /* 设置topBar/bottomBar/definitionView/RateView
   的背景 */
7. + (void)setBrandColor:(UIColor *)color;
8.
9. /* small image */
10. /* logo */
11. + (void)setLogoImage:(UIImage *)image;
12. /* backButtonImage */
13. + (void)setBackButtonImage:(UIImage *)image;
14. /* PlayButton */
15. + (void)setPlayButtonImage:(UIImage *)image;
16. /* pauseButton */
17. + (void)setPauseButtonImage:(UIImage *)image;
18. /* stopButton */
19. + (void)setStopButtonImage:(UIImage *)image;
20. /* nextButton */
21. + (void)setNextButtonImage:(UIImage *)image;
22. /* ProgressSlider */
23. + (void)setProgressSliderImage:(UIImage
   *)image;
24. /* ScaleButton */
25. + (void)setScaleButtonImage:(UIImage *)image;
26. /* ForwardImage */
27. + (void)setForwardImage:(UIImage *)image;
28. /* BackwardImage */
29. + (void)setBackwardImage:(UIImage *)image;
```

# 代码示例

参见[Demo](#)，Demo只是示范SDK中接口的使用方式，并不保证使用方式最符合用户的需求，请用户在充分理解SDK机制的前提下，根据自己的情况处理数据。对照抄SDKDemo代码引起的性能问题和其他非SDK bug，不负任何责任。

## 常见问题

- 为什么接入了 `BJPUViewController` 之后，点击全屏，无法全屏
  - 首先，看APP项目的设置，检查 `General` -> `Device Orientation` 中是否打开了Landscape Left, Landscape Right。
  - 其次，确认在代码中，window、root view controller、navigation controller、tabbar controller 等VC中有没有特殊设置，是否支持旋转和支持的方向。
  - 最后，如果APP其他界面代码不能支持横屏，可以仅在使用 `BJPUViewController` 时允许横屏，其他时候限制使用竖屏。
  - UI demo 更新了两种弹出UISDK的方式，可以自定义弹出横竖屏，具体实现可参考SDK demo
- UI SDK的功能和界面无法满足现有需求
  - UI SDK是开源的，支持在源码基础上自定义修改，参考[点播 UI SDK自定义修改文档](#)



下载为pdf格式