



AdsMOGO

iOS SDK 插屏+视频

植入流程及使用方法

## 目录

1.iOS SDK 植入流程 .....	1
1.1 iOS SDK 运行环境及支持的系统版本.....	1
1.2 在芒果后台注册 APP 获取芒果 ID 及添加芒果 SDK.....	1
1.2.1 注册 APP 获取芒果 ID .....	1
1.2.2 添加芒果 SDK.....	2
1.2.3 在 Xcode 工程中添加芒果 SDK 及相关使用注意事项.....	3
1.2.4 添加芒果静态库所需 framework.....	3
1.3 在三方平台注册 APP 并获取第三方平台 key 以及添加第三方平台 SDK.....	4
1.3.1 在三方平台注册 APP 并获取第三方平台 key.....	4
1.3.2 添加第三方平台 SDK.....	4
1.3.3 添加单一平台所需的 framework.....	5
1.3.4 关于在模拟器中调试的相关说明 .....	6
1.4 在程序中使用 AdsMOGO SDK 展示全插屏视频广告 .....	7
1.4.1 快速使用 .....	7
1.4.2 进阶使用.....	10
1.4.3 在应用中使用多个 AppKey.....	11
1.4.4 视频广告 .....	12
1.4.5 芒果日志控制.....	14
1.4.6 在 ARC 中使用芒果 SDK.....	15
1.4.7 配置离线文件 “mogoID.config” （可选） .....	15
2.芒果功能介绍 .....	17
2.1 全/插屏广告轮换时间.....	17
2.2 位置获取.....	17
3.常见问题解答 .....	18
4. 联系芒果.....	19

# 1.iOS SDK 植入流程

## 1.1 iOS SDK 运行环境及支持的系统版本

Xcode建议使用Xcode5；芒果的SDK支持iOS v4.3及以上系统；某些单一平台SDK支持iOS v6.0及以上系统。

## 1.2 在芒果后台注册 APP 获取芒果 ID 及添加芒果 SDK

### 1.2.1 注册 APP 获取芒果 ID

登陆芒果网站 [www.adsmogo.com](http://www.adsmogo.com) 后点击 App 管理 - 添加 App 如下图：



根据提示注册 App，注册完成后在第三步中显示您的 App 芒果 ID，如下图：



您还可以点击 App 管理从 App 列表找到您的 App，点击您的 App，在 App 配置界面左侧显示芒果 ID，如下图：



## 1.2.2 添加芒果 SDK

(1). 登录芒果后台，点击 <http://my.adsmogo.com/Apps/Download> 显示如下界面：

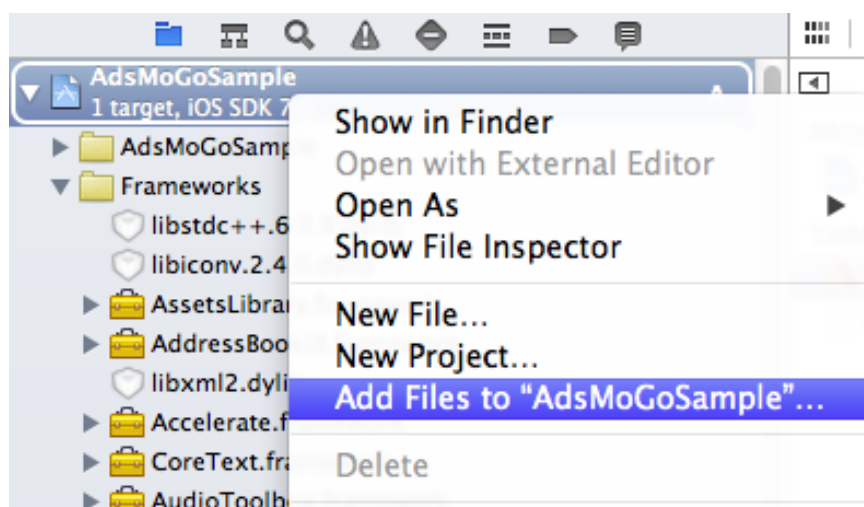


选择“芒果 iOS SDK” 点击下载芒果 iOS SDK。

(2). 解压 iOS SDK，目录文件文档说明：

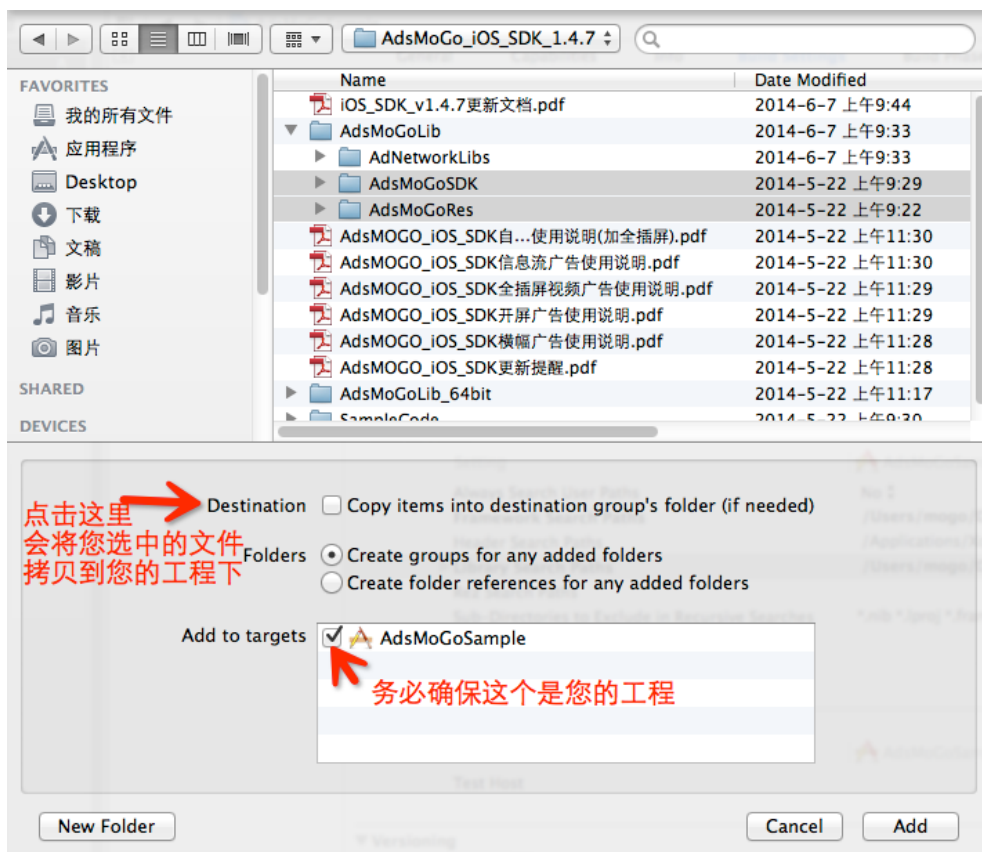
AdsMoGoLib	芒果 SDK + 芒果资源文件 + 各个第三方平台适配器及 SDK
AdsMoGoLib/AdsMoGoSDK	芒果 SDK
AdsMoGoLib/AdsMoGoRes	芒果资源文件
AdsMoGoLib/AdNetworkLibs	各个第三方平台适配器及 SDK
AdsMoGoLib_64bit	芒果 64 位 SDK + 部分平台 64 位 SDK

(3). 打开您的 Xcode 工程，右键点击您的 Xcode 工程 Add Files to xxx (xxx 为您的 Xcode 工程名称)，如下图：





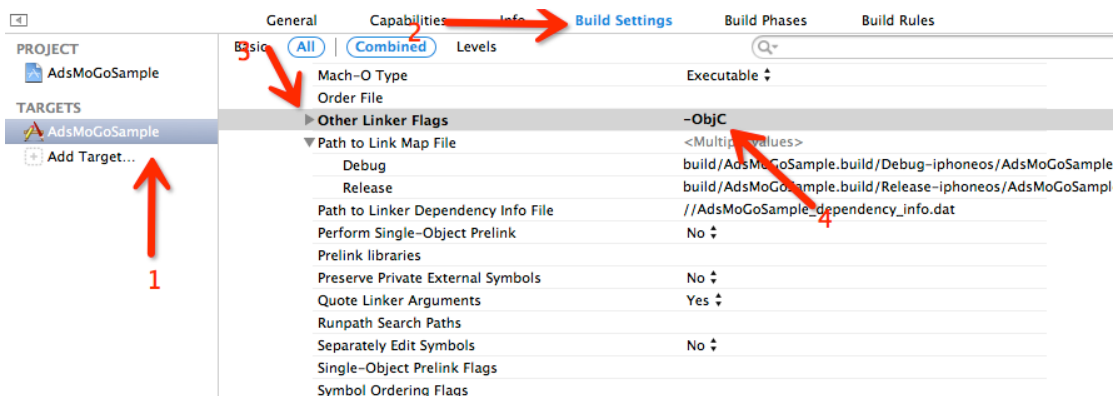
选择 AdsMoGoLib 下的 AdsMoGoSDK、AdsMoGoRes 这两个文件夹，如下图：



确保 Add To Targets 勾选相应的 target，点击 Add 将其添加到工程中。





















### 1.2.3 在 Xcode 工程中添加芒果 SDK 及相关使用注意事项

点击程序 Target 文件，选择 Build Settings 标签页，找到 Linking 下面的 Other Linker Flags，添加参数-ObjC。如下图（注意首字母和最后一个字母是大写字母，不添加这个运行会有崩溃）：



### 1.2.4 添加芒果静态库所需 framework

如下图所示：

	AudioToolbox.framework
	EventKit.framework
	AdSupport.framework
	EventKitUI.framework
	SystemConfiguration.framework
	QuartzCore.framework
	MessageUI.framework
	MapKit.framework
	CFNetwork.framework
	CoreData.framework
	AVFoundation.framework
	MediaPlayer.framework
	CoreLocation.framework
	CoreTelephony.framework
	StoreKit.framework
	CoreGraphics.framework
	Foundation.framework
	UIKit.framework
	libsqlite3.dylib
	libz.1.2.5.dylib

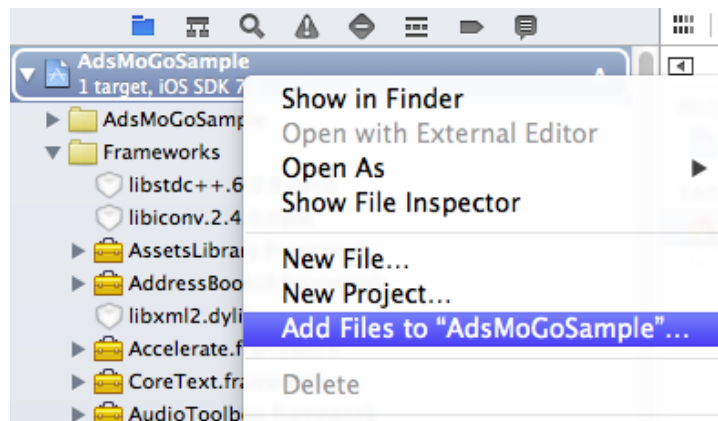
## 1.3 在三方平台注册 APP 并获取第三方平台 key 以及添加第三方平台 SDK

### 1.3.1 在三方平台注册 APP 并获取第三方平台 key

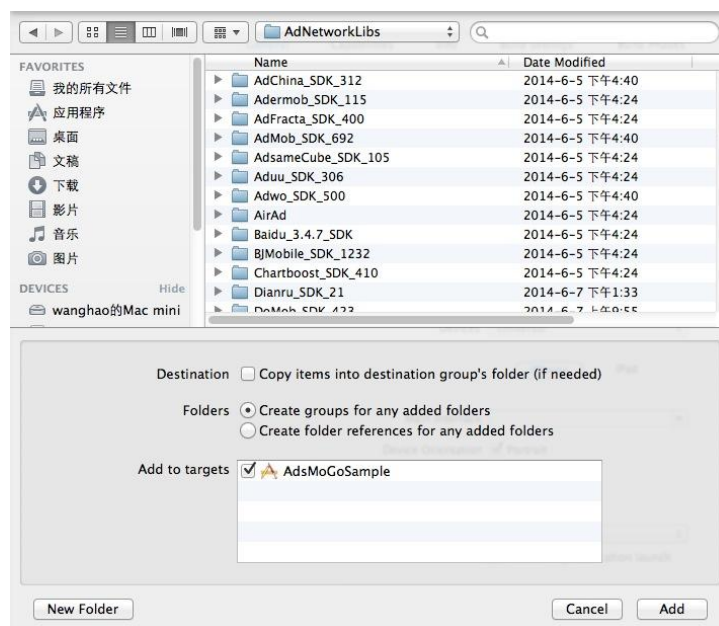
点击您注册的 App—广告平台管理，选择您需要的广告类型并注册、配置相关第三方广告平台参数。第三方广告平台的注册方法，请联系各个平台。

### 1.3.2 添加第三方平台 SDK

打开您的 Xcode 工程，右键点击您的 Xcode 工程，选择 Add Files to xxx (xxx 为您的工程名称)，如下图：



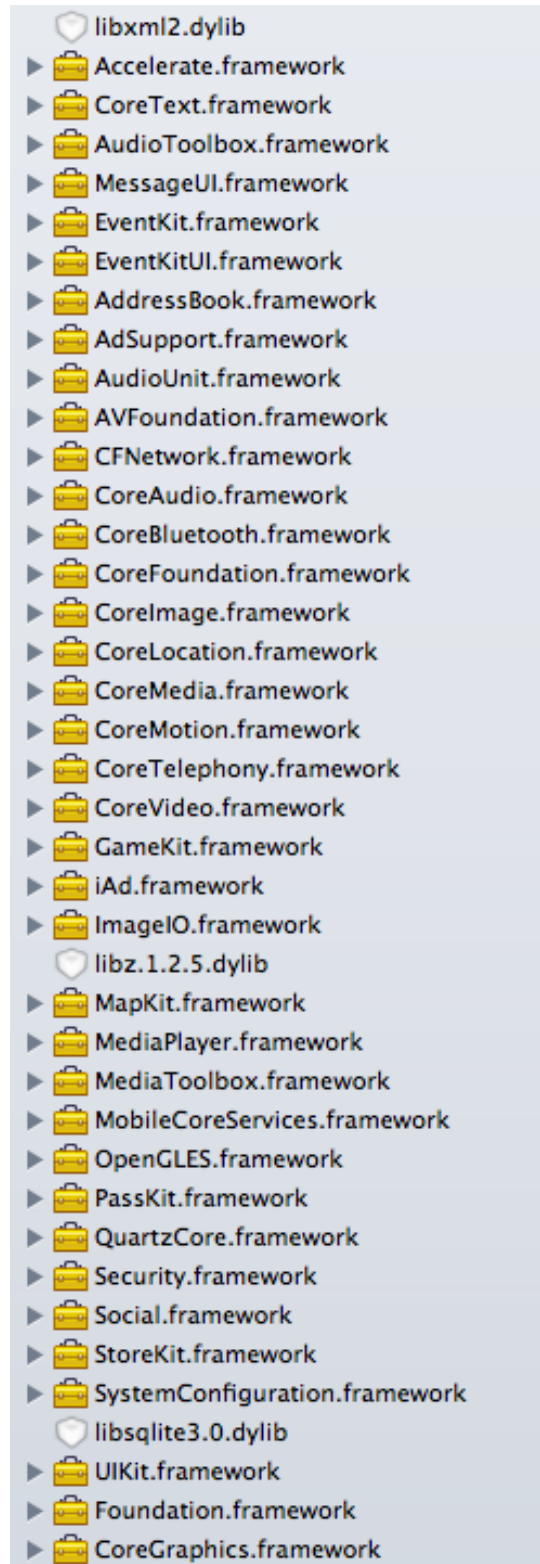
选择您所需的第三方平台 SDK（第三方 SDK 文件路径~/AdsMoGoLib/AdNetworkLibs）：



确保 Add To Targets 勾选相应的 target，点击 Add 将需要的第三方平台 SDK 添加到工程中。

### 1.3.3 添加单一平台所需的 framework

如需要各个平台的 framework，请参照<芒果聚合各广告平台 framework 参考>文档：



#### 1.3.4 关于在模拟器中调试的相关说明

芒果建议使用真机调试。某些平台不向模拟器中投放广告；某些广告点击后是加载内置 Appstore，由于模拟器没有这个功能，所以会导致点击后没有点击效果。





## 1.4 在程序中使用 AdsMOGO SDK 展示全插屏视频广告

AdsMOGO SDK1.4.7之后，全插屏2.0。

### 1.4.1.快速使用

快速使用是指通过最少的代码即可正常请求展示全插屏广告。

#### (1).全插屏自动轮转

芒果SDK根据您在后台设置的全/插屏时间间隔自动为您请求广告，具体使用方式如下：

```
//导入AdMoGoInterstitialManager.h
#import "AdMoGoInterstitialManager.h"
@implementation AdsMogoInterstitialSampleViewController
- (void)viewDidLoad
{
    [super viewDidLoad];
    //设置AppKey
    [AdMoGoInterstitialManager setAppKey:@"your app key"];
    //初始化(必须先设置默认的AppKey才能通过此方法初始化SDK)
    [[AdMoGoInterstitialManagershareInstance] initWithDefaultInterstitial];
}
/**
 *进入展示时机
 *当应用需要展示全屏广告调用- (void)interstitialShow:(BOOL)isWait;
 *通知SDK进入展示时机,SDK会竭尽全力展示出广告,当然由于网络等问题不能立即展示
 *广告的,您可以通过参数isWait来控制授权SDK在获得广告后立即展示广告。
 */
- (void)showInterstitial{
    [[AdMoGoInterstitialManagershareInstance] interstitialShow:YES];
}
/**
 *退出展示时机
 *如果您之前进入了展示时机,并且isWait参数设置为YES,那么在需要取消等待广告展示的
 *时候调用方法- (void)interstitialCancel;来通知SDK
 */
- (void)cancelShow{
    [[AdMoGoInterstitialManagershareInstance] interstitialCancel];
}
```



注：您可以在任何位置使用此方式初始化全插屏并展示全插屏广告。上述代码是以 `UIViewController` 为例，您也可以在 `AppDelegate` 中用同样的方式启动 SDK。

(2). 全插屏手动轮换 (注: v2.0 版插屏手动轮换在 v1.4.9 及以后版本支持)

#### A. 手动轮换后台设置

请在 App 信息界面中，将全/插屏间隔时间设置为禁用，如下图所示：



#### B. 相关代码使用

使用 - (AdMoGoInterstitial \*)adMogoInterstitialByAppKey:(NSString \*)appKey  
isManualRefresh:(BOOL)isManualRefresh;

- (AdMoGoInterstitial \*)adMogoVideoInterstitialByAppKey:(NSString \*)appKey  
isManualRefresh:(BOOL)isManualRefresh; 方法初始化全插屏广告，并将 `isManualRefresh` 参数设置为 YES。然后调用 - (void)refreshAd 方法来刷新广告。具体示例代码如下：

```
//导入AdMoGoInterstitialManager.h
#import "AdMoGoInterstitialManager.h"
@implementation AdsMogoInterstitialSampleViewController
- (void)viewDidLoad
{
    [super viewDidLoad];
    //初始化全插屏
    AdMoGoInterstitial *interstitialIns = [[AdMoGoInterstitialManager
    sharedInstance] adMogoInterstitialByAppKey:self.appkey
    isManualRefresh:YES];
    interstitialIns.delegate = self;
}
/**
 *进入展示时机
 */
- (void)showInterstitial{
    AdMoGoInterstitial *ins = [[AdMoGoInterstitialManager
    sharedInstance] adMogoInterstitialByAppKey:self.appkey];
    [ins interstitialShow:YES]
}
/**
 *退出展示时机
 */
- (void)cancelShow{
    AdMoGoInterstitial *ins = [[AdMoGoInterstitialManager
    sharedInstance] adMogoInterstitialByAppKey:self.appkey];
    [ins interstitialCancel];
}

/**
 *轮换插屏广告 注意：此方法需要开发者根据app情况自行调用
 */
- (void)freshInterstitialAd{
    AdMoGoInterstitial *ins = [[AdMoGoInterstitialManager
    sharedInstance] adMogoInterstitialByAppKey:self.appkey];
    [ins refreshAd];
}
```



注：

• 调用refreshAd方法时，请确保广告初始化已完成并且是手动刷新广告（isManualRefresh 为 YES，并且轮换时间为禁用），否则该方法不起作用。

如果需要在初始化后立刻请求广告，建议在广告初始化完成的回调中调用刷新广告的方法，刷新一次广告请求。

```
- (void)adMoGoInterstitialInitFinish{
    AdMoGoInterstitial *ins = [[AdMoGoInterstitialManager sharedInstance]
    adMoGoInterstitialByAppKey:self.appkey];
    [ins refreshAd];
}
```

• 如果在手动刷新广告状态下调用refreshAd请求广告轮换，但是还没有广告返回，又再次调用了refreshAd，此时不会再次请求广告轮换。直到广告请求失败或者广告请求成功展示完成关闭后，才能再次调用refreshAd请求广告轮换。

• 您可以在任何位置使用此方式初始化全插屏并展示全插屏广告。上述代码是以 UIViewController 为例，您也可以在 AppDelegate 中用同样的方式启动 SDK。

### 1.4.2 进阶使用

进阶使用方式是可以让您使用获得广告当前展示的状态，设置用于展示的默认 ViewController，使用多个 AppKey 等功能。

#### (1) .设置当前 ViewController

```
//设置AppKey
[AdMoGoInterstitialManagersetAppKey:@"your app key"];
//设置当前ViewController(可选,推荐),用于展示全屏广告
[AdMoGoInterstitialManager setRootViewController:self];
```

注：由于大部分广告平台需要基于当前的 ViewController 才能展示广告，那么如果您没有设置默认的 ViewController，SDK 会智能选择 ViewController 展示广告。如果您设置了此参数，SDK 会优先选择此参数用于广告的展示。

#### (2) .获取当前广告的通知

如果需要获取当前广告的状态，需要通过设置 delegate 来实现，具体方式如下：

首先，您需要在类中实现 “AdMoGoInterstitialDelegate” 协议；

然后，通过 + (void)setDefaultDelegate:(id)delegate; 设置代理对象：

```
//设置AppKey
[AdMoGoInterstitialManagersetAppKey:@"your app key"];
//设置Delegate(可选)
[AdMoGoInterstitialManagersetDefaultDelegate:self];
```

```
/*返回广告rootViewController*/
- (UIViewController *)viewControllerForPresentingInterstitialModalView;
/*全屏广告将要展示*/
- (void)adsMoGoInterstitialAdWillPresent;
/*全屏广告消失*/
- (void)adsMoGoInterstitialAdDidDismiss;
/**
 *广告过期
 *是否立即重新轮换广告
 *return YES-->立即轮换 NO-->等待下一次进入展示机会轮换
 */
- (BOOL)adsMogoInterstitialAdDidExpireAd;
/*全屏广告浏览器展示*/
- (void)adsMoGoWillPresentInterstitialAdModal;
/*全屏广告浏览器消失*/
- (void)adsMoGoDidDismissInterstitialAdModal;
/**
 *芒果全屏 关闭按钮被点击
 *return YES/NO 来确定是否可关闭 default YES
 */
- (BOOL)adsMogoInterstitialAdClosedButtonTap;
/*芒果广告关闭*/
- (void)adsMogoInterstitialAdClosed:(BOOL)isAutoClose;
- (BOOL)interstitialShouldAlertQAView:(UIAlertView *)alertView;
- (BOOL)isFullScreen;

/*芒果插屏初始化完成 */
- (void)adMoGoInterstitialInitFinish;
/*手动轮换下，广告轮空回调*/
- (void)adMoGoInterstitialInMaualfreshAllAdsFail;
```

### 1.4.3 在应用中使用多个 AppKey

上面的使用方式都是基于仅适用一个AppKey而言，如果您有使用多个AppKey的需求，您需要继续往下看，其中包括：



设置AppKey  
设置Delegate  
进入展示时机  
离开展示时机

具体使用示例:

```
/*//导入AdMoGoInterstitialManager.h
#import "AdMoGoInterstitialManager.h"
//导入AdMoGoInterstitial.h
#import "AdMoGoInterstitial.h"
@implementation AdsMogoInterstitialSampleViewController
- (void)viewDidLoad
{
    [super viewDidLoad];
    //设置AppKey 并初始化配置
    AdMoGoInterstitial *interstitialIns = [[AdMoGoInterstitialManagers sharedInstance]
    adMogoInterstitialByAppKey:@"your app key"];
    //设置delegate
    interstitialIns.delegate = self;
}
/**
 *进入展示时机
 *当应用需要展示全屏广告调用- (void)interstitialShow:(BOOL)isWait;
 *通知SDK进入展示时机,SDK会竭尽全力展示出广告,当然由于网络等问题不能立即展示
 *广告的,您可以通过参数isWait来控制授权SDK在获得到广告后立即展示广告。
 */
- (void)showInterstitial{
    [[[AdMoGoInterstitialManagers sharedInstance] adMogoInterstitialByAppKey:@"your
    app key"] interstitialShow:YES];
}
```

#### 1.4.4 视频广告

SDK 版本 1.4.7 之后视频广告聚合功能的使用也发生了变化,简化后的使用流程和使用方式与全屏非常相似。



## 初始化

### (1). 自动轮播初始化方式

芒果 SDK 根据您在后台设置的全/插屏时间间隔自动为您请求广告

```
//初始化视频广告
[[AdMoGoInterstitialManager sharedInstance]
adMogoVideoInterstitialByAppKey:@"your app key"];
```

### (2). 手动轮播初始化方式

手动轮播后台设置：

请在 App 信息界面中，将全/插屏时间间隔设置为禁用。

### 文字广告样式

背景颜色\*

#000000

文字颜色\*

#FFFFFF

横幅刷新时间

15秒

全/插屏时间间隔时间

禁用

过场效果

随机选择

效果增强

关闭

位置获取

☒

相关代码使用：

```
//初始化视频广告
[[AdMoGoInterstitialManager sharedInstance]
adMogoVideoInterstitialByAppKey:@"your app key" isManualRefresh:YES];
[AdMoGoInterstitialManagersetDefaultDelegate:self];

#pragma mark -
#pragma mark AdMoGoInterstitialDelegate
/**
 *广告初始化完成
 */
- (void) adMoGoInterstitialInitFinish {
    NSLog(@"视频初始化完成");
    //视频初始化未完成时，调用refreshAd将不起作用
    [[[AdMoGoInterstitialManagershareInstance]
adMogoVideoInterstitialByAppKey:@"your app key"]
refreshAd];
}
```

进入展示时机：

```
//进入展示时机
- (void)showInterstitial{
[[[AdMoGoInterstitialManagershareInstance]
adMogoVideoInterstitialByAppKey:@"your app key"]
interstitialShow:YES];
}
```

离开展示时机：

```
//离开展示时机
[[[AdMoGoInterstitialManagershareInstance]
adMogoVideoInterstitialByAppKey:@"your app key"]
interstitialCancel];
```

#### 1.4.5 芒果日志控制



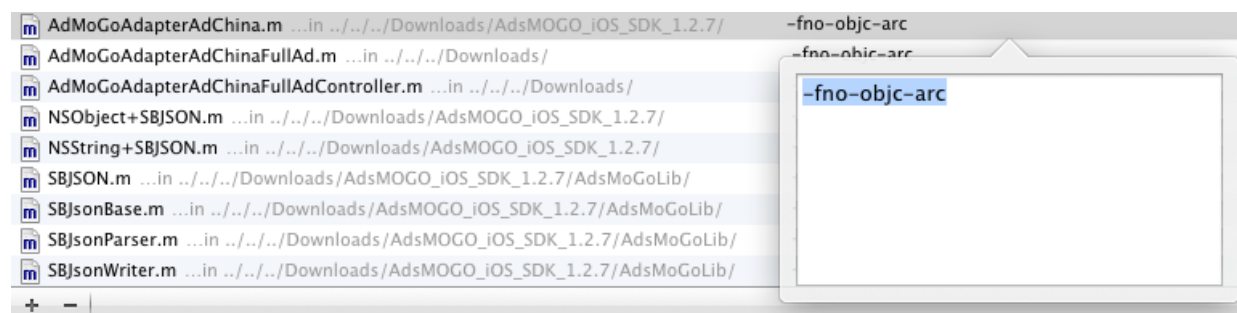
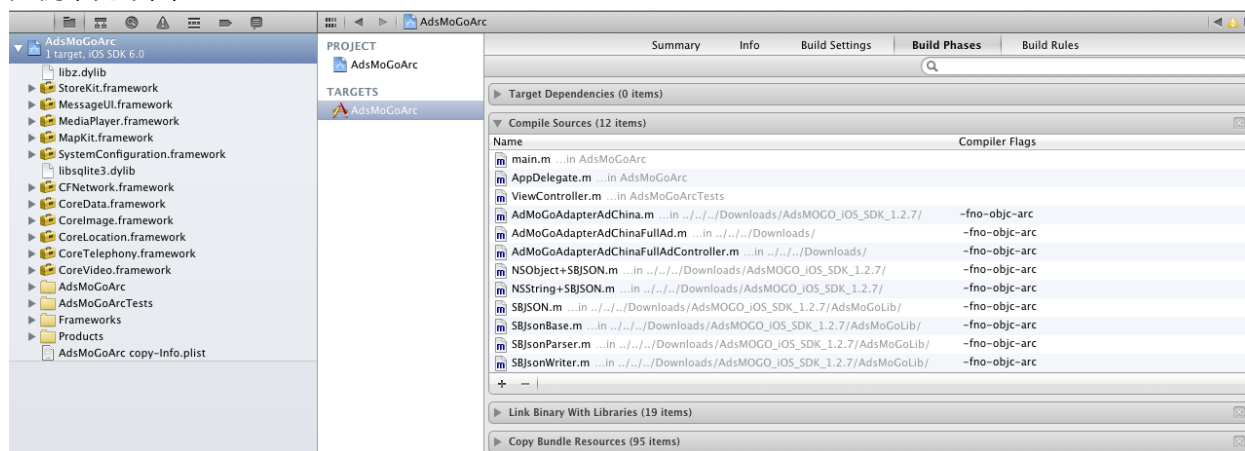


在你的工程 AppDelegate 中添加头文件#import "AdMoGoLogCenter.h", 添加如下代码开启日志打印:

```
[[AdMoGoLogCentershareInstance] setLogLevelFlag:AdMoGoLogTemp];
```

### 1.4.6 在 ARC 中使用芒果 SDK

如果在 ARC 工程中使用芒果 SDK, 需要进行如下配置。在 target 里的 build phases 中, 找到 compile sources 将使用的平台的适配器.m 文件 (名称为 AdMoGoAdapterxxxx.m 的文件, 其中 xxxx 是平台名) 和平台使用的类库非 ARC 的.m 后面加上: -fno-objc-arc。以 AdChina (易传媒) 为例, 见下图:



AdMoGoAdapterAdChina.m	...in ../Downloads/AdsMOGO_iOS_SDK_1.2.7/	-fno-objc-arc
AdMoGoAdapterAdChinaFullAd.m	...in ../Downloads/	-fno-objc-arc
AdMoGoAdapterAdChinaFullAdController.m	...in ../Downloads/	-fno-objc-arc
NSObject+SBJSON.m	...in ../Downloads/AdsMOGO_iOS_SDK_1.2.7/	-fno-objc-arc
NSString+SBJSON.m	...in ../Downloads/AdsMOGO_iOS_SDK_1.2.7/	-fno-objc-arc
SBJSON.m	...in ../Downloads/AdsMOGO_iOS_SDK_1.2.7/AdMoGoLib/	-fno-objc-arc
SBJsonBase.m	...in ../Downloads/AdsMOGO_iOS_SDK_1.2.7/AdMoGoLib/	-fno-objc-arc
SBJsonParser.m	...in ../Downloads/AdsMOGO_iOS_SDK_1.2.7/AdMoGoLib/	-fno-objc-arc
SBJsonWriter.m	...in ../Downloads/AdsMOGO_iOS_SDK_1.2.7/AdMoGoLib/	-fno-objc-arc

### 1.4.7 配置离线文件 “mogoID.config” (可选)

功能简介:



在芒果网站配置好广告平台后，可以下载对应的广告配置离线文件，导入到工程中和芒果 SDK 配合使用。当新用户第一次使用您的应用，手机与芒果服务器信息交互异常，无法获取广告配置时，即可调用离线文件，以便实现正常展示广告。

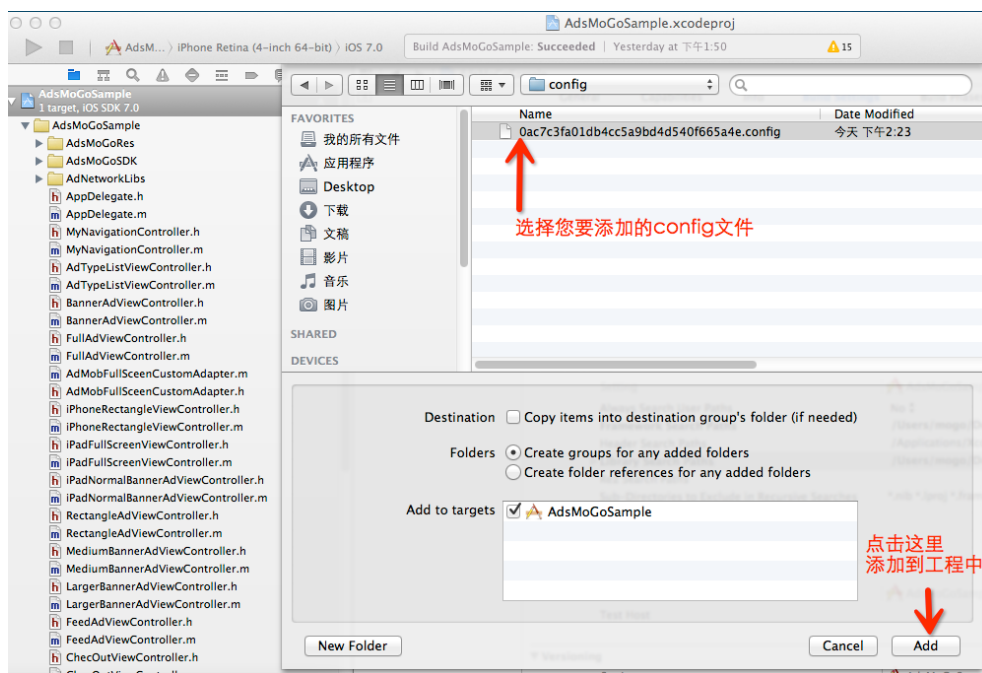
使用方法：

配置完广告平台后，选择SDK下载，点击[离线配置文件下载]，下载的文件格式为.config，如下图所示：



右键点击您的 Xcode 工程选择 Add Files to xxx (xxx 为您的 xcode 工程名称)。

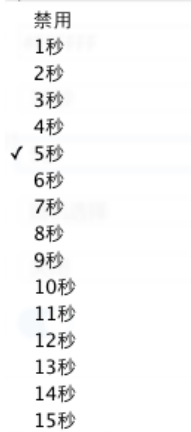
找到您下载的 xxx.config 文件，确保 AddToTargets 勾选相应的 target 点击 Add 添加 xxx.config 到 xcode 工程中。如下图：



## 2. 芒果功能介绍

### 2.1 全/插屏广告轮换时间

功能简介：全/插屏广告在广告关闭后等待下次请求时间间隔，默认 5 秒。



A vertical dropdown menu with the following options: 禁用, 1秒, 2秒, 3秒, 4秒, 5秒 (selected with a checkmark), 6秒, 7秒, 8秒, 9秒, 10秒, 11秒, 12秒, 13秒, 14秒, 15秒.

### 2.2 位置获取

功能简介：

开启位置获取，可以获得更精确的地域优化推送的广告。可精确确定用户所在位置，并根据这个位置推送最合适的广告。此功能默认开启。

全/插屏广告轮换时间、位置获取功能在芒果网站上的位置如下图：



The screenshot shows the '易传媒sdk测试ipad' configuration page. On the left is a sidebar with 'App信息' selected. The main area has two sections: 'App信息' and '文字广告样式'. In the 'App信息' section, 'App名称' is '易传媒sdk测试ipad', '设备' is 'iPad [?]', and 'App类型' is '图书'. In the '文字广告样式' section, '背景颜色' is '#000000', '文字颜色' is '#FFFFFF', '横幅刷新时间' is '30秒', '全/插屏间隔时间' is '5秒' (highlighted with a red box), '过场效果' is '随机选择', '效果增强' is '关闭', and '位置获取' is '开启' (highlighted with a red box). A preview of the ad is shown on the right.

## 3.常见问题解答

### 3.1 编译报错，报的错误为

Undefined symbols for architecture  $\times\times\times$ :

"\_OBJC\_CLASS\_\$\_ $\times\times\times\times\times$ ", referenced from:

objc-class-ref in  $\times\times\times\times\times.a(\times\times\times\times.o)$

答：缺少芒果或单一平台需要的框架。若红色 $\times\times\times$ 部分为 libAdsMogo\_i386.a 或 libAdsMogo.a 或 libAdsMogo\_x8664bit.a 或 libAdsMogo\_64bit.a，则代表缺少芒果 SDK 需要的框架，请参照 1.2.4 部分的相关说明添加相应的框架。若红色 $\times\times\times$ 部分为单一平台的静态库，则代表缺少相应平台的框架，请参照 1.3.3 部分添加该单一平台需要的框架。

### 3.2 编译报错，报的错误为

duplicate symbol  $\times\times\times$  in:

$/\times\times/\times\times/\times\times/\times\times/\times\times.o$

$/\times\times/\times\times/\times\times/\times\times/\times\times.o$

答：此问题为文件或类命名冲突导致的。其中 $/\times\times/\times\times/\times\times/\times\times/\times\times.o$  为引起冲突的文件或类的路径。

### 3.3 编译报错，报的错误为

ld: library not found for  $-\times\times\times$

答：此问题是 SDK 中的静态库路径错误引起的，其中 $\times\times\times$ 为路径错误的平台 SDK。请检查 Library Search Paths 下是否有该平台的 SDK 路径，并检查路径是否正确，若无该路径或路径错误，请添加正确的路径。若无法确认正确的路径，请将该平台 SDK 从工程中去掉，然后重新加进来。

### 3.4 工程可运行，但插屏、视频广告不展示

- 答：(1). 请检查初始化方法中的芒果 id 和广告类型是否与后台配置
- (2). 请检查是否添加了后台配置的单一平台的 SDK（若使用原生整合的平台，则无需添加单一平台的 SDK）
- (3). 请确认在芒果后台填写的单一平台的 key 是否在单一平台通过审核。若未审核通过，请确认单一平台会提供测试广告
- (4). 请检查您的设备是否为模拟器或 iPod Touch，有些平台不往模拟器或 iPod Touch 上投放广告，建议使用 iPhone 或者 iPad 进行测试
- (5). 请检查您设备的网络是否正常
- (6). 请检查您是否在芒果后台配置了全插屏、视频广告

### 3.5 工程一运行，即崩溃，崩溃日志如下



```
-[NSDictionaryMobjectForKey:andSuffix:]:unrecognized selector sent to instance xxx
```

```
***Terminating app due to uncaught exception  
'NSInvalidArgumentException', reason: '-  
[NSDictionaryMobjectForKey:andSuffix:]:unrecognized selector sent to  
instance xxx
```

答：请检查 Other Linker Flags 中是否正确的添加了-ObjC。（注：OC 为大写，bj 为小写）

### 3.6 模拟器中点击广告没有响应

答：有些广告是点击后跳转到内置 AppStore 的，模拟器没有内置 AppStore 的功能

## 4. 联系芒果

如果您有任何问题或疑问，请及时联系芒果客服，我们将在第一时间做出回应。

Email : help@adsmogo.com

QQ : 1346883530/1976719507

TEL : 400-6088-215

芒果技术论坛经验交流版块：<http://bbs.adsmogo.com/showforum-5.aspx>