



AdsMOGO

iOS SDK 插屏广告使用说明

V 1.4.5



尊敬的开发者朋友，欢迎您使用芒果移动广告优化平台！

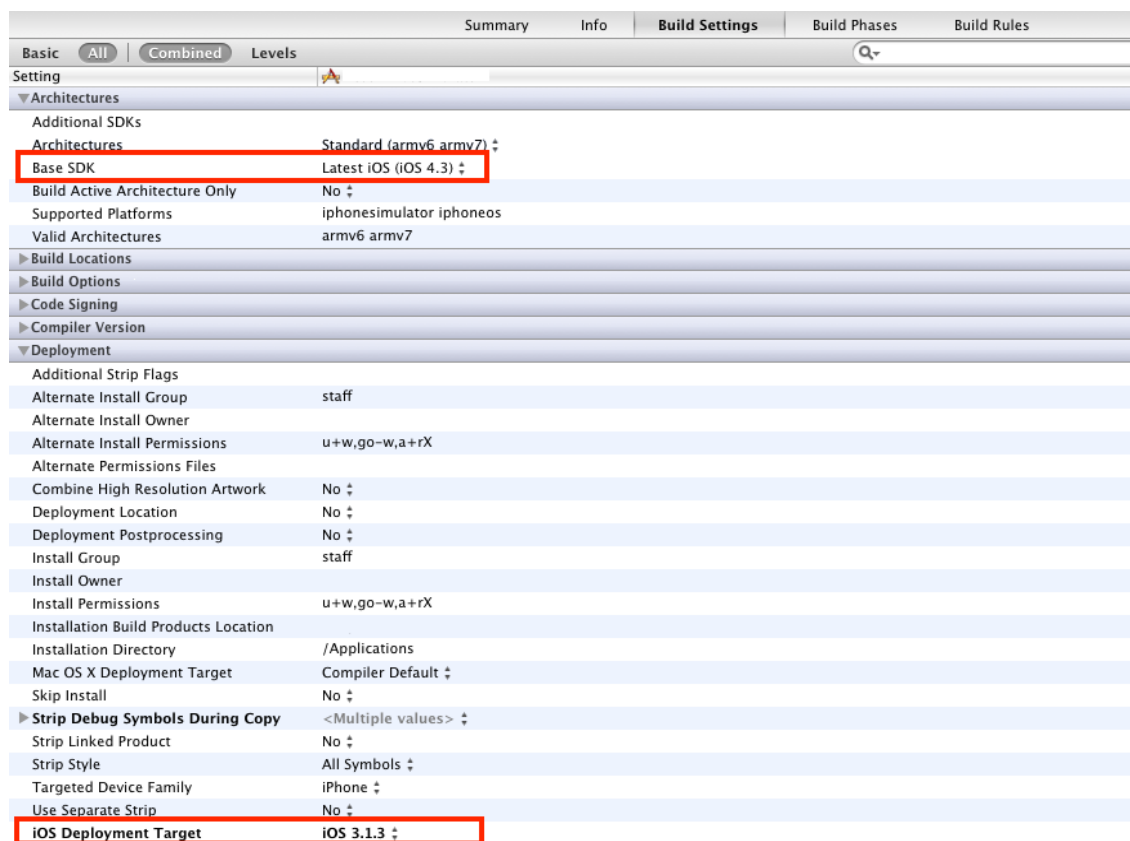
本文档为您详细介绍了AdsMOGO_iOS_SDK植入流程以及使用过程中的常见问题，希望对您使用芒果有所帮助！

一、iOS SDK版本设置

在使用AdsMOGO iOS SDK之前，请先确认您的Xcode版本为4.5以上。

虽然MOGO SDK 已经支持IOS6.0，但是存在一些平台仍然不支持，对于某些已支持的平台，建议用户将XCode升级至4.5以上使用，具体细节还需参考单一平台的说明文档。

注：由于很多广告平台 SDK升级至支持IOS6版本，需要添加AdSupport.framework、PassKit.framework等多个新类库，故必须将XCode升级至4.5才能使用。



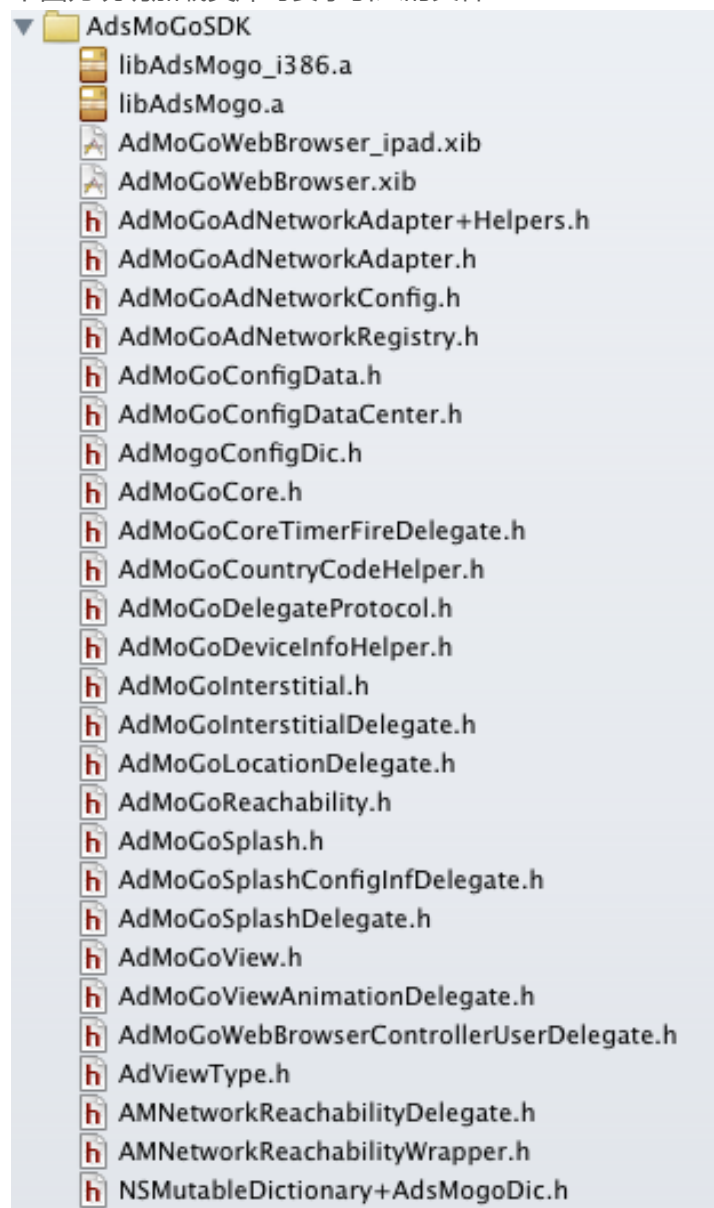
二、添加AdsMOGO SDK

在您设置完开发环境后，让我们开始植入AdsMOGO SDK，请按以下步骤操作。

Step 1：添加AdsMoGoSDK文件夹和AdsMoGoRes文件夹到工程。

（ 所在位置：\AdsMOGO_iOS_SDK_1.4.5\AdsMoGoLib\ ）

下图为说明加载类库时要求引入的文件：





Step 2 : MoGo库使用注意事项

在V1.2.6版本中，将原来的libAdMoGo.a这个包分成两个包，分别为：

libAdsMogo_i386.a——使用在模拟器上

libAdsMogo.a——使用在真实设备上

Step 3 : 添加Framework

UIKit.framework

Foundation.framework

CoreGraphics.framework

注：以上为程序默认添加的类库

CoreLocation.framework

注：要支持iOS 3.x，需要将其弱引用

CoreTelephony.framework

注：要支持iOS 3.x，需要将其弱引用

MapKit.framework

注：要支持iOS 3.x，需要将其弱引用

SystemConfiguration.framework

libsqlite3.dylib

注：弱引用类库，即点击程序的Target将指定类库的Role从Required设为Optional。

下图为引用类库的各个选项：

Summary	Info	Build Settings	Build Phases	Build Rules
<input type="text"/>				
▶ Copy Bundle Resources (60 items)				
▶ Compile Sources (49 items)				
▼ Link Binary With Libraries (34 items)				
Security.framework			Required	⬆ ⬇ ⬆
OpenAL.framework			Required	⬆ ⬇ ⬆
CoreTelephony.framework			Optional	⬆ ⬇ ⬆
OpenGL.framework			Required	⬆ ⬇ ⬆
EventKit.framework			Optional	⬆ ⬇ ⬆
CoreMotion.framework			Optional	⬆ ⬇ ⬆
AddressBookUI.framework			Required	⬆ ⬇ ⬆
Foundation.framework			Required	⬆ ⬇ ⬆
UIKit.framework			Required	⬆ ⬇ ⬆
CoreGraphics.framework			Required	⬆ ⬇ ⬆
libsqlite3.dylib			Required	⬆ ⬇ ⬆
MessageUI.framework			Optional	⬆ ⬇ ⬆
CFNetwork.framework			Required	⬆ ⬇ ⬆
SystemConfiguration.framework			Required	⬆ ⬇ ⬆
QuartzCore.framework			Required	⬆ ⬇ ⬆
iAd.framework			Optional	⬆ ⬇ ⬆
AVFoundation.framework			Required	⬆ ⬇ ⬆
CoreLocation.framework			Required	⬆ ⬇ ⬆
AddressBook.framework			Required	⬆ ⬇ ⬆
AudioToolbox.framework			Required	⬆ ⬇ ⬆
MapKit.framework			Required	⬆ ⬇ ⬆
MediaPlayer.framework			Required	⬆ ⬇ ⬆

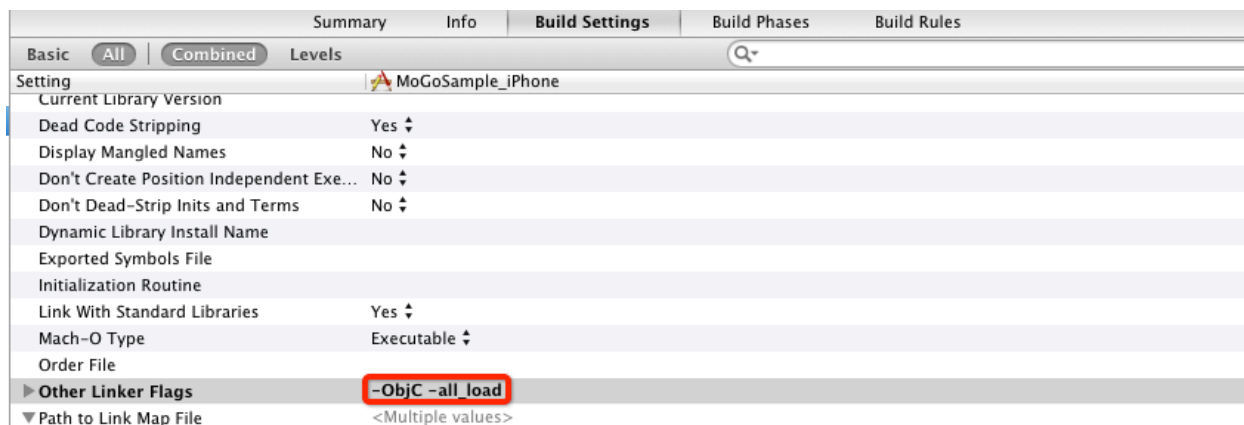
建议将如下类库都添加到工程中:

- libiconv.dylib
- libstdc++.6.0.9.dylib
- libz.dylib
- libsqlite3.0.dylib
- libxml2.dylib
- ▶ CoreFoundation.framework
- ▶ Security.framework
- ▶ CoreData.framework
- ▶ CoreMedia.framework
- ▶ CoreVideo.framework
- ▶ ImageIO.framework
- ▶ CoreMotion.framework
- ▶ CFNetwork.framework
- ▶ MobileCoreServices.framework
- ▶ iAd.framework
- ▶ OpenAL.framework
- ▶ AVFoundation.framework
- ▶ MediaPlayer.framework
- ▶ AudioToolbox.framework
- ▶ MapKit.framework
- ▶ SystemConfiguration.framework
- ▶ QuartzCore.framework
- ▶ MessageUI.framework
- ▶ CoreLocation.framework
- ▶ EventKit.framework
- ▶ CoreTelephony.framework
- ▶ UIKit.framework
- ▶ Foundation.framework
- ▶ CoreGraphics.framework

Step 4：更改导入静态库设置

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-all_load -ObjC。

下图为Linking下面的Other Linker Flags，添加参数-ObjC。



三、添加各广告平台SDK

您已成功植入AdsMOGO SDK，现在让我们开始添加您需要的单一平台SDK。

Step 1：从芒果网站（或者从芒果SDK AdsMoGoLib目录下）获取单一平台SDK并添加到工程

（所在位置：<http://www.adsmogo.com/sdk?t=2>）

如图：

最新版本SDK下载

芒果SDK&各广告平台SDK

在这里，您可以下载到最新版本的芒果SDK，也可以下载到芒果支持的各单一移动广告平台的SDK。请您严格按照《芒果SDK使用说明》植入SDK，如果遇到任何问题请参考本页面左侧“芒果SDK下载说明”中的问题汇总。当然，您也可以直接联系芒果技术支持(QQ:1113272874)，芒果的技术支持团队将竭诚为您服务。

最新芒果SDK下载
其他移动广告平台SDK下载

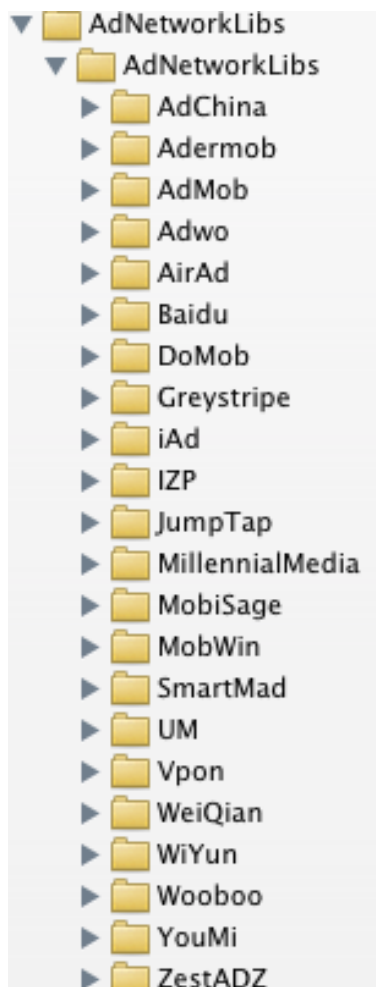
SDK下载说明

1. 芒果提供的各移动广告平台SDK都是经过芒果优化和测试通过的，以保证广告展示的稳定。建议您使用芒果SDK中包含的或从本页面下载的各广告平台SDK来配合芒果SDK。
2. 芒果建议您尽量将所有的广告平台SDK都添加到App中，这样方便您之后在后台调整广告平台。
3. 各广告平台需要添加framework，或者声明Activity，才能正常工作，请参

广告平台名称	iOS	Android
AdMob	v6.0.1 3.53M 下载	v6.0.0 0.31M 下载
Jumtap	v2.0 3.17M 下载	-
Millennial Media	v4.6 3.86M 下载	v4.2.6 1.12M 下载
Greystripe	v3.1.3 2.6M 下载	v1.6.1 0.46M 下载
MdotM	-	v3.0.0 0.02M 下载
InMobi 	v3.6.0 2.66M 下载	v2.0 1.14M 下载
ZestADZ	v1.2 0.4M 下载	v1.2 0.46M 下载
AdChina	v2.5.2 1.76M 下载	v2.4 1.45M 下载
微云	v3.0.0 1.14M 下载	v4.0.0 0.28M 下载
哇棒	v2.3 0.9M 下载	v2.3 0.27M 下载
有米	v4.0 7.06M 下载	v3.08 0.17M 下载
架势 	v2.7 1.33M 下载	v3.0 1.4M 下载
亿动智道	v2.0.5 7.85M 下载	v2.5.2 0.1M 下载

注：您可以仅选择您想要支持的广告平台的SDK目录，从而能减小应用程序的大小。

下图为各广告平台的SDK:



Step 2：按使用说明要求把各广告平台需要的Framework加入工程

注：开启广告平台之前需要去对应广告平台网站申请广告位，并将广告位ID等参数填写到芒果网站“App管理”设定面板中。

四、在程序中使用AdsMOGO SDK显示广告

现在让我们看看在您的App代码中都需要添加哪些内容。

1. 引用广告的ViewController.h文件中添加AdMoGoDelegate、AdMoGoWebBrowserControllerUserDelegate的继承，同时定义AdMoGoView对象。在ViewController.m中实现AdMoGoDelegate、AdMoGoWebBrowserControllerUserDelegate中定义的方法。

比如要在TestAdViewController添加广告，可参考下面的代码片段：TestAdViewController.h（参考Sample Code/AdsMoGoSample/AdsMoGoSample/BannerAdViewController.h 9 ~ 16 行）

```
#import <UIKit/UIKit.h>
#import "AdMoGoDelegateProtocol.h"
#import "AdMoGoView.h"
#import "AdMoGoWebBrowserControllerUserDelegate.h"

@interface BannerAdViewController : UIViewController
<AdMoGoDelegate , AdMoGoWebBrowserControllerUserDelegate>{
    AdMoGoView *adView;
}
@property (nonatomic , retain) AdMoGoView *adView;
```

TestAdViewController.m

（参考Sample Code/ AdsMoGoSample/AdsMoGoSample /BannerAdViewController.m 11 ~ 12、77 ~ 79行）

```
@implementation BannerAdViewController
@synthesize adView;

- (UIViewController *)viewControllerForPresentingModalView{
    return self;
```

2. 获取并显示全屏 / 插屏广告

V1.4.4中全插屏新增两个初始化方法及对应的启动广告轮换的方法（注:原初始化方法中在获取配置后会自动开始轮换广告，新增初始化方法及启动广告轮换的方法见11页12页）。新增初始化方法，此方法只获取广告配置，需要使用新增启动广告轮换方法启动广告轮换。开发者可依据自己的需求决定是否使用这两个新增的初始化方法。

V1.3.1中的全插屏逻辑与之前有很大不同，建议开发者仔细阅读这部分。

V1.3.0 及以前版本中的全屏广告使用方法在v1.3.1中不再支持。

在TestAdViewController.m的viewDidLoad方法中添加如下代码创建并获取广告配置，获取配置后自动请求当前配置中的全屏广告之一。

（参考SampleCode/ AdsMoGoSample/AdsMoGoSample /FullAdViewController.m
39 ~ 50行）

```
- (void)viewDidLoad
{
    [super viewDidLoad];
    interstitial = [[AdMoGoInterstitial alloc]
        initWithAppKey:@"70745cc17a5341e3ba6fbe857703f6d7"
        isRefresh:YES
        adInterval:20
        adType:AdViewTypeFullScreen
        adMoGoViewDelegate:self];
}
```

以下AdMoGoInterstitialDelegate必须实现

```
/*
  返回广告rootViewController
*/
- (UIViewController *)viewControllerForPresentingInterstitialModalView{
    return self;
}
```

使用如下代码展示全屏广告

```
if (interstitial) {
    if ([interstitial isInterstitialReady]) {
        [interstitial present];
    }
}
```



请求到并展示一个平台后，在广告关闭后如果设置自动请求（isRefresh:YES）SDK依据设置的等待时间（adInterval:20），在等待时间后，请求下一平台插屏/全屏，如果没有设置自动请求（isRefresh:NO）不请求任何平台广告，展示结束。以下方法默认自动请求，等待时间为10秒。

```
interstitial = [[AdMoGoInterstitial alloc] initWithAppKey:@"  
70745cc17a5341e3ba6fbe857703f6d7"  
                adType:AdViewTypeFullScreen  
                adMoGoViewDelegate:self];
```

如果设置等待时间为0并且设置了刷新，那么关闭广告后会立刻请求下一平台；如果所有平台都没有请求到广告，并且设置自动，SDK在等待10秒后继续下一轮请求。

v1.4.4增加初始化方法

```
/*  
  初始化  
  默认轮换 没有轮换间隔时间  
  如果调用此初始化方法 必须使用 - (void)pollingInterstitialAd 启动广告轮换  
  
  ak:芒果ID  
  type:全屏 插屏  
  delegate:全屏回调  
*/  
-(id)noautopollinginitWithAppKey:(NSString *)ak  
    adType:(AdViewType)type  
    adMoGoViewDelegate:(id<AdMoGoInterstitialDelegate>) delegate;
```

```
/*  
  非自动轮换初始化  
  如果调用此初始化方法 必须使用 - (void)pollingInterstitialAd 启动广告轮换  
  ak:芒果ID  
  refreshed:是否刷新  
  adInterval:间隔时间（所有平台都请求失败，并且允许刷新，那么间隔这个时间后在请求全屏）  
  type:全屏 插屏  
  delegate:全屏回调  
*/  
-(id)noautopollinginitWithAppKey:(NSString *)ak  
    isRefresh:(BOOL)refreshed  
    adInterval:(int)adInterval  
    adType:(AdViewType)type  
    adMoGoViewDelegate:(id<AdMoGoInterstitialDelegate>) delegate;
```



如果使用1.4.4中增加的初始化方法，需要使用
pollingInterstitialAd启动广告轮换，启动后不用再次调用。

```
/*  
    非自动轮换初始化启动插屏轮换  
*/  
- (void)pollingInterstitialAd;
```



根据要实现AdMoGoInterstitialDelegate中的下列方法：
(参考Sample Code/ AdsMoGoSample/AdsMoGoSample/FullAdViewController.m
73 ~ 85 行)

```
/*
  返回广告rootViewController
*/
- (UIViewController *)viewControllerForPresentingInterstitialModalView{
    return self;
}
/*
  全屏广告开始请求
*/
- (void)adsMoGoInterstitialAdDidStart{
    NSLog(@"MOGO Full Screen Start");
}
/*
  全屏广告准备完毕
*/
- (void)adsMoGoInterstitialAdIsReady{
    NSLog(@"MOGO Full Screen IsReady");
}
/*
  全屏广告接收成功
*/
- (void)adsMoGoInterstitialAdReceivedRequest{
    NSLog(@"MOGO Full Screen Received");
}
/*
  全屏广告将要展示
*/
- (void)adsMoGoInterstitialAdWillPresent{
    NSLog(@"MOGO Full Screen Will Present");
}
/*
  全屏广告接收失败
*/
- (void)adsMoGoInterstitialAdFailedWithError:(NSError *) error{
    NSLog(@"MOGO Full Screen Failed");
}
```

```
/*
全屏广告消失
*/
- (void)adsMoGoInterstitialAdDidDismiss{
    NSLog(@"MOGO Full Screen Dismiss");
}

/*
全屏广告浏览器展示
*/
- (void)adsMoGoWillPresentInterstitialAdModal{
    NSLog(@"MOGO AdModal Screen Present");
}

/*
全屏广告浏览器消失
*/
- (void)adsMoGoDidDismissInterstitialAdModal{
    NSLog(@"MOGO AdModal Screen Dismiss");
}
```

注：芒果IOS SDKv1.3.9以下方法不再支持安沃平台。安沃平台插屏支持自动旋转。

```
-
(void)interstitialorientationChanged:(UIInterfaceOrientation)orientat
ion;
```

注：芒果IOS SDKv1.3.8 中手动请求全插屏广告

```
-(void)nofreshRequestInterstitial;
```

此方法使用条件：在初始化方法中，如果isRefresh为False时该方法调用有效。

此方法作用：当isRefresh为False且初始化后，全插屏广告请求成功关闭后或请求全部失败后，调用此方法再次请求插屏广告。



注：芒果IOS SDKv1.4.2 中新增关于自主、自售、交换、认领、自动优化广告关闭回调控制

```
/*
    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;

/*
    芒果自主、自售、交换、认领、自动优化广告广告关闭
    isAutoClose YES 自动关闭；NO 手动关闭
*/
- (void)adsMogoInterstitialAdClosed:(BOOL)isAutoClose
```

注：芒果IOS SDK v1.4.3中新增全屏\插屏回调 (V1.4.3中此回调支持平台亿赞普)

```
/*
    判断是否展示全屏。V1.4.3中支持亿赞普
    返回YES展示全屏广告。不实现或者返回NO展示插屏广告。
*/
- (BOOL)isFullScreen;
```

注：芒果IOS SDK v1.4.4中 与增加的初始化方法对应的一个回调方法表示初始化完成。

```
/*
    非自动轮询初始化完成
*/
-(void)noautopollinginitadfinished;
```




3. 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：



请保存您App的芒果ID，您需要将它复制到App的代码中

本应用的芒果ID：c255b3a879ca4b85a92592adc43786d0 [复制ID](#)

请保存您App的芒果ID，您需要将它复制到App的代码中。

为保证在断网或芒果服务器宕机等极端情况下，您的广告展示不受影响。芒果建议您在植入芒果SDK时同时植入一个“离线配置文件”。请在下方导出您的“离线配置文件”，并按照说明将其添加到App源代码工程中。

[离线配置文件下载 >>](#)

4. 芒果高级功能介绍

1) 地域优化

功能简介：

根据用户所处地理位置来判断并向其推送最合适的广告。此功能默认开启。

2) 点击优化

功能简介：

此功能聚合了来自芒果实验室的一系列广告优化逻辑，能够在理论上提升广告的点击数和点击率，进而提升您的广告收入。

比如，开启点击优化，用户点击广告处理一些事件后，返回展示广告的视图，立刻请求下一个广告，最大程度上节省了轮换广告的等待时间。关闭点击优化，用户点击广告后返回到展示广告视图，等到一轮的刷新时间到了才请求下一个广告。

3) 关闭广告

功能简介：

开启关闭广告，您的广告位右侧将会出现一个半透明的‘X’按钮，用户可以点击来暂时关闭此广告位。用户下次使用您的App时，广告位会恢复出现。

地域优化、点击优化、关闭广告这三项的位置开关如下图（红色



App信息

广告平台

SDK下载

4) 位置获取

推广工具

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, 既精确确定用户所在位置, 并根据这个位置推送合适的广告。此功能默认开启。

5) 过场效果

功能简介:

新加载广告时的轮换动画效果。所支持的效果选项如图:

复制ID

设备* iPhone [?]

iTunes ID

请输入本App

App类型

--请选择--

添加

App类型

书籍 X

从右滑过
从左滑过
从右翻页
从左翻页
淡入屏幕
向上翻卷
向下翻卷
✓ 随机选择
无

字广告样式

色* #000000

色* #FFFFFF

刷新时间

30秒

过场效果

随机选择

位置获取

OFF

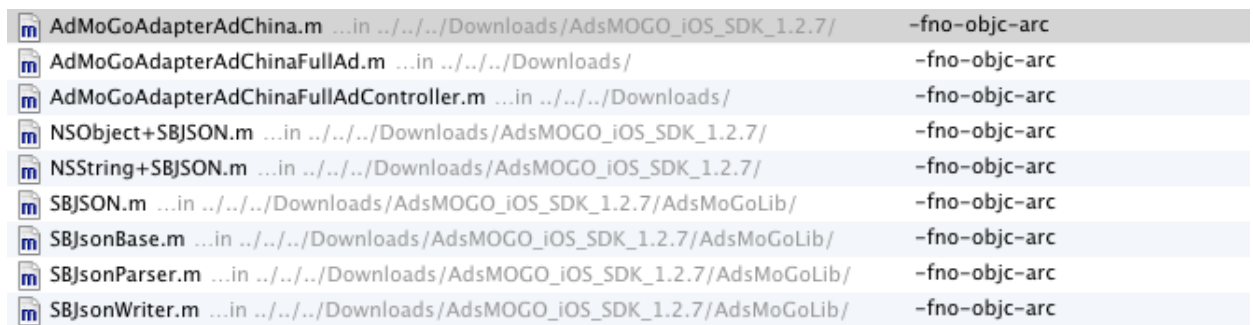
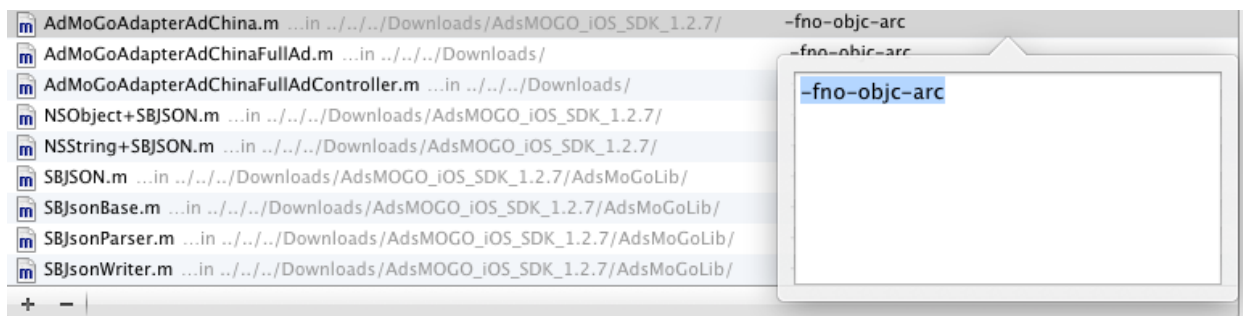
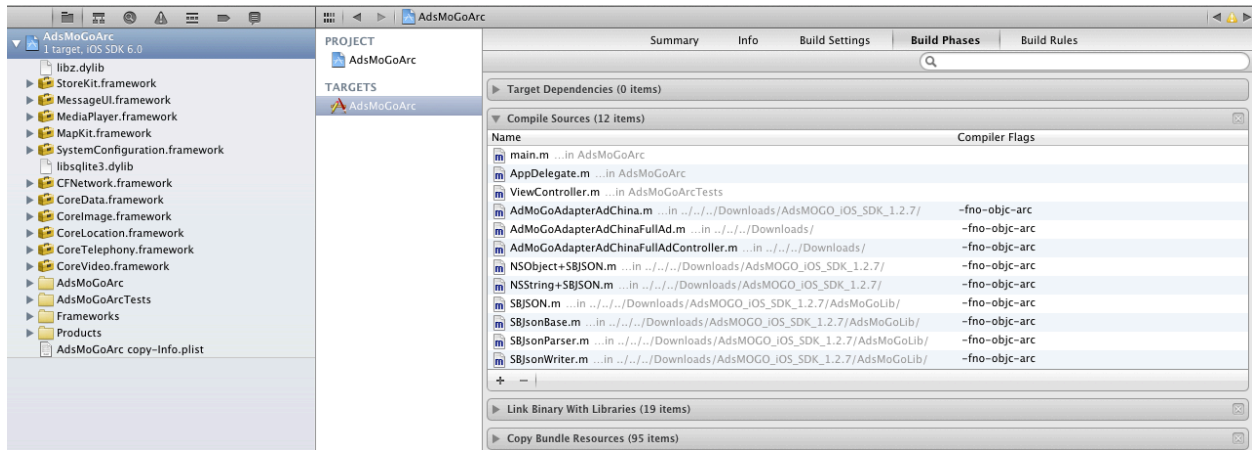
位置获取、过场效果这两项功能位置如下



6) 在ARC配置中使用芒果SDK

如果在一个 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 (这样的文件 AdMoGoAdapterxxxx.m 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:





7) Vpon平台自定义区域

如果需要自行设置vpon Platform属性 可以在{appName}-Prefix.pch 文件中宏定义
vpon_custom_platform

如：#define vpon_custom_platform @"TW"。

注：定义值为国家代码双字母简写 目前vpon只支持 CN 和 TW 的设置，如果不定义 则交由
adsMogo® SDK根据真实环境处理。

五、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>