

Domob iOS SDK 4.x 使用手册

1 Demo 下载地址：

<https://github.com/Domob-SDK/domob-ios-ad-sdk-sample>

2 必要条件

1. 使用 Xcode 5 或更高版本
2. 运行环境为 iOS 4.3 或更高版本

3 从多盟官网获取 PublisherID & PlacementID

如果想要在 App 中使用多盟的广告，首先需要获得一个 Publisher ID 和相应的 PlacementID。请登陆多盟官网（<http://www.domob.cn/help/index.htm#tab1>），获取更多如何创建新用户以及新建应用的信息。

当新建应用并拥有一个PublisherID之后，添加广告位后即可获得广告位ID，如下图所示，详情请查看<http://www.domob.cn/news/announcement/20130503/503.htm>。

我的应用 > 我们是多盟星人

Publisher ID: 56OJy3pYuMZNTD1RgO

应用平台: iOS 程序

内容分类: 游戏-动作格斗

应用状态: 上传应用

广告位管理

点击此处
创建广告位

广告位名称	类型	状态	广告位ID	创建时间	操作
添加新的广告位					

广告位管理

广告位名称	类型	状态	广告位ID	创建时间	操作
首页横幅广告	inline广告位	运行	16TLw4GaAcxMc8shm5dR0Z0k	2013-05-02 20:08	暂停

广告位名称: 详情页横幅广告

广告位类型: 请选择

- inline广告位
- 普通插屏广告位
- 开屏广告位
- 信息流顶部广告位
- 推广墙广告位

保存广告

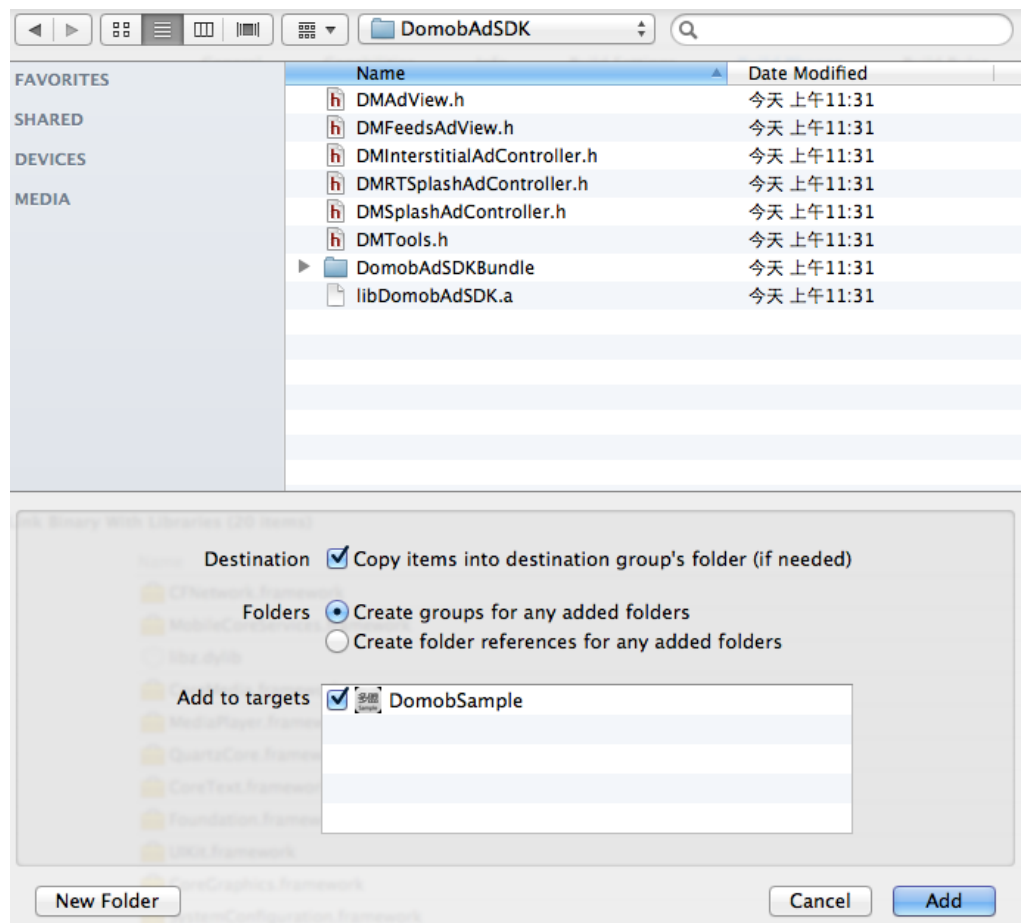
用此ID去配置SDK

选择广告位类型

4 将 SDK 添加到项目中

SDK 的发行版本中需要添加到项目中的文件包括一个静态库、4 个头文件以及一个 Bundle 文件。

1. 将上述文件，添加到您的项目中。建议直接将“DomobAdSDK”文件夹添加到项目中。
包含的内容，如下图：



2. 若要 SDK 正常工作，需要向项目中添加下面这些 Framework，如下图：
(红框中是 4.0 新增 framework，还请注意添加)

▼ Link Binary With Libraries (21 items)

Name	Status
CFNetwork.framework	Optional
MobileCoreServices.framework	Optional
libz.dylib	Optional
CoreMedia.framework	Optional
MediaPlayer.framework	Optional
QuartzCore.framework	Optional
CoreText.framework	Optional
Foundation.framework	Required
UIKit.framework	Required
CoreGraphics.framework	Required
SystemConfiguration.framework	Required
AdSupport.framework	Optional
CoreLocation.framework	Optional
StoreKit.framework	Optional
PassKit.framework	Optional
Social.framework	Optional
EventKit.framework	Optional
AVFoundation.framework	Optional
AudioToolbox.framework	Optional
libsqlite3.dylib	Optional
libDomobAdSDK.a	Required

5 广告视图的使用

5.1 Banner 广告的使用

（创建 Banner 广告位，请在官网“广告位管理”界面的“广告位类型设置中”请选择“inline 广告位”）

若要在应用中添加一个 Banner 广告，只需要简单的几步：

- 导入 DMAView.h 头文件。
- 声明一个 DMAView 的实例。
- 使您的 UIViewController 实现 DMAViewDelegate。
- 在您的 UIViewController 的 m 文件中，使用您从多盟官网获得的 PublisherId 和 PlacementId 以及您期望的广告尺寸，来初始化 DMAView。
- 设置 DMAView 的委派。
- 设置 DMAView 的 root view controller。
- 设置 DMAView 的 frame，把 DMAView 放置在您希望的位置。
- 将 DMAView 添加到父视图中。
- 最后，调用 loadAd 方法，开始请求广告。

下面的代码片段，简要的描述了这个过程：

```
#import "DMAView.h"

@interface DMBannerSampleViewController: UIViewController <DMAViewDelegate>
{
    DMAView *_dmAdView;
}
@end
```

```
@implementation DMBannerSampleViewController
```

```

- (id)initWithNibName:(NSString *)nibNameOrNil bundle:(NSBundle *)nibBundleOrNil
{
    self = [super initWithNibName:nibNameOrNil bundle:nibBundleOrNil];
    if (self)
    {
        self.title = NSLocalizedString(@"Banner", @"Banner");

        /* 创建广告视图，此处使用的是测试ID，请登陆多盟官网（www.domob.cn）获取新的ID */
        _dmAdView = [[DMAdView alloc] initWithPublisherId:@"560JyM1ouMGoULfJaL"
                                                    placementId:@"16TLwebvAchkAY6i0Md734jz"
                                                    size:DOMOB_AD_SIZE_320x50];

        // 设置广告视图的位置
        _dmAdView.frame = CGRectMake(0, 0,
                                      DOMOB_AD_SIZE_320x50.width,
                                      DOMOB_AD_SIZE_320x50.height);
    }
    return self;
}

- (void)viewDidLoad
{
    [super viewDidLoad];

    _dmAdView.delegate = self; // 设置 Delegate
    _dmAdView.rootViewController = self; // 设置 RootViewController
    [self.view addSubview:_dmAdView]; // 将广告视图添加到父视图中
    [_dmAdView loadAd]; // 开始加载广告
}

- (void)viewDidUnload
{
    [super viewDidUnload];
    [_dmAdView removeFromSuperview]; // 将广告视图从父视图中移除
}

- (void)dealloc
{
    _dmAdView.delegate = nil;
    _dmAdView.rootViewController = nil;
    [_dmAdView release];
    [super dealloc];
}
@end

```

关于 DMAdView 的释放：

在这里强烈建议您，在释放 DMAdView 之前，将 DMAdView 的属性 delegate 和 rootViewController 设置为 nil。

设置可选参数：

还可以为 DMAdView 设置一些可选参数，来提高广告投放的精准性。这些参数的设置方法都定义在 DMAdview.h 中，如下：

```

// 设置地理位置信息
- (void)setLocation:(CLLocation *)location;

// 设置邮编
- (void)setPostcode:(NSString *)postcode;

// 设置关键字
- (void)setKeywords:(NSString *)keywords;

// 设置用户年龄
- (void)setUserBirthday:(NSString *)userBirthday;

// 设置用户性别
- (void)setUserGender:(DMUserGenderType)userGender;

```

实现 DMAbstractViewDelegate:

您可以通过实现 DMAbstractViewDelegate 中定义的方法，来跟踪广告生命周期中的各个阶段。所有这些方法也都定义在 DMAbstractView.h 中，如下：

```
@protocol DMAbstractViewDelegate <NSObject>
@optional
// 成功加载广告后，回调该方法
- (void)dmAbstractViewSuccessToLoadAd:(DMAbstractView *)adView;
// 加载广告失败后，回调该方法
- (void)dmAbstractViewFailToLoadAd:(DMAbstractView *)adView withError:(NSError *)error;

// 当将要呈现出 Modal View 时，回调该方法。如打开内置浏览器。
- (void)dmWillPresentModalViewFromAd:(DMAbstractView *)adView;
// 当呈现的 Modal View 被关闭后，回调该方法。如内置浏览器被关闭。
- (void)dmDidDismissModalViewFromAd:(DMAbstractView *)adView;
// 当因用户的操作（如点击下载类广告，需要跳转到Store），需要离开当前应用时，回调该方法
- (void)dmApplicationWillEnterBackgroundFromAd:(DMAbstractView *)adView;

@end
```

5.2 Flexible 广告的使用

（创建 Flexible 广告位，请在官网“广告位管理”界面的“广告位类型设置中”请选择“inline 广告位”）

Flexible 广告是多盟推出的自适应宽度的广告类型。

请使用特定的 size: FLEXIBLE_SIZE_PORTRAIT/FLEXIBLE_SIZE_LANDSCAPE

若要在应用中添加一个 Flexible 广告，只需要简单的几步：

- j. 导入 DMAbstractView.h 头文件。
- k. 声明一个 DMAbstractView 的实例。
- l. 使您的 UIViewController 实现 DMAbstractViewDelegate。
- m. 在您的 UIViewController 的 m 文件中，使用您从多盟官网获得的 PublisherId 和 PlacementId 以及您期望的广告尺寸，来初始化 DMAbstractView。
- n. 设置 DMAbstractView 的委派。
- o. 设置 DMAbstractView 的 root view controller。
- p. 设置 DMAbstractView 的 frame，把 DMAbstractView 放置在您希望的位置。
- q. 将 DMAbstractView 添加到父视图中。
- r. 最后，调用 loadAd 方法，开始请求广告。
- s. 在检测到设备旋转的时候请更改所创建实例的 frame。

下面的代码片段，简要的描述了这个过程：

iPhone 与 iPad 的操作是一致的，使用的 size 参数也一致。详细示例请参看 Demo，简单示例如下图：

（注意创建时的 SIZE,同时注意横竖屏切换的处理）

```

// 创建广告视图, 此处使用的是测试ID, 请登陆多盟官网 (www.domob.cn) 获取新的ID
// Creat advertisement view please get your own ID from domob website
_dmFlexibleAdView = [[DMAView alloc] initWithPublisherId:DM_PUBLISHER_ID
                                           placementId:DM_PLACEMENT_ID
                                           size:FLEXIBLE_SIZE_PORTRAIT];

// 设置广告视图的位置
// Set the frame of advertisement view
_dmFlexibleAdView.frame = CGRectMake(_adX, _adY,
                                     FLEXIBLE_SIZE_PORTRAIT.width,
                                     FLEXIBLE_SIZE_PORTRAIT.height);

_dmFlexibleAdView.delegate = self;
_dmFlexibleAdView.rootViewController = self; // set RootViewController
[self.view addSubview:_dmFlexibleAdView];
[_dmFlexibleAdView loadAd]; // start load advertisement

```

```

- (void)didRotateFromInterfaceOrientation:(UIInterfaceOrientation)fromInterfaceOrientation
{
    //针对横竖屏 设定对应的FlexibleView的frame
    //set right frame for FlexibleView
    if (UIInterfaceOrientationIsLandscape(fromInterfaceOrientation)){

        _dmFlexibleAdView.frame = CGRectMake(_dmFlexibleAdView.frame.origin.x,
                                              _dmFlexibleAdView.frame.origin.y,
                                              FLEXIBLE_SIZE_PORTRAIT.width,
                                              FLEXIBLE_SIZE_PORTRAIT.height);

    }else{

        _dmFlexibleAdView.frame = CGRectMake(_dmFlexibleAdView.frame.origin.x,
                                              _dmFlexibleAdView.frame.origin.y,
                                              FLEXIBLE_SIZE_LANDSCAPE.width,
                                              FLEXIBLE_SIZE_LANDSCAPE.height);

    }
}

```

关于 DMAView 的释放:

在这里强烈建议您, 在释放 DMAView 之前, 将 DMAView 的属性 delegate 和 rootViewController 设置为 nil。

设置可选参数:

还可以为 DMAView 设置一些可选参数, 来提高广告投放的精准性。这些参数的设置方法都定义在 DMAview.h 中, 如下:

```

// 设置地理位置信息
- (void)setLocation:(CLLocation *)location;

// 设置邮编
- (void)setPostcode:(NSString *)postcode;

// 设置关键字
- (void)setKeywords:(NSString *)keywords;

// 设置用户年龄
- (void)setUserBirthday:(NSString *)userBirthday;

// 设置用户性别
- (void)setUserGender:(DMUserGenderType)userGender;

```

实现 DMAViewDelegate:

您可以通过实现 DMAViewDelegate 中定义的方法, 来跟踪广告生命周期中的各个阶段。所有这些方法也都定义在 DMAView.h 中, 如下:

```

@protocol DMAViewDelegate <NSObject>
@optional

```

```

// 成功加载广告后，回调该方法
- (void)dmAdViewSuccessToLoadAd:(DmAdView *)adView;
// 加载广告失败后，回调该方法
- (void)dmAdViewFailToLoadAd:(DmAdView *)adView withError:(NSError *)error;

// 当将要呈现出 Modal View 时，回调该方法。如打开内置浏览器。
- (void)dmWillPresentModalViewFromAd:(DmAdView *)adView;
// 当呈现的 Modal View 被关闭后，回调该方法。如内置浏览器被关闭。
- (void)dmDidDismissModalViewFromAd:(DmAdView *)adView;
// 当因用户的操作（如点击下载类广告，需要跳转到Store），需要离开当前应用时，回调该方法
- (void)dmApplicationWillEnterBackgroundFromAd:(DmAdView *)adView;

@end

```

5.3 插屏广告的使用

插屏广告的使用，同样非常简单，只需要几步即可完成。

- 导入 DMInterstitialAdController.h 头文件。
- 声明一个 DMInterstitialAdController 的实例。
- 让您的 UIViewController 实现 DMInterstitialAdControllerDelegate 协议。
- 在您的 Controller 的 m 文件中，使用 PublisherId 和 PlacementId 以及负责呈现插屏广告的 UIViewController 来初始化一个 DMInterstitialAdController。
- 在合适的时候，调用 loadAd 方法，来预加载插屏广告。
- 在想要呈现插屏广告的时候，调用 present 方法，来呈现插屏广告。
- 在呈现完成后，需要重新通过 loadAd 方法，加载一条新广告，用于下次呈现。

下面的代码片段，简单描述了这个过程：

```

@interface DMInterstitialSampleViewController: UIViewController
<DMInterstitialAdControllerDelegate>
{
    DMInterstitialAdController *_dmInterstitial;
}
@end

```

关于 DMInterstitialAdController 的释放：

在这里强烈建议您，在释放 DMInterstitialAdController 之前，将其 delegate 属性设置为 nil。

设置可选参数：

与 DmAdView 相同，也可以为 DMInterstitialAdController 设置一些可选参数，来提高广告投放的精准性。这些参数的设置方法都定义在 DMInterstitialAdController.h 中，如下：

```

// 设置地理位置信息
- (void)setLocation:(CLLocation *)location;

// 设置邮编
- (void)setPostcode:(NSString *)postcode;

// 设置关键字
- (void)setKeywords:(NSString *)keywords;

// 设置用户年龄
- (void)setUserBirthday:(NSString *)userBirthday;

// 设置用户性别

```



```
- (void)setUserGender:(DMUserGenderType)userGender;
```

实现 DMInterstitialAdControllerDelegate:

您可以通过实现 DMInterstitialAdControllerDelegate 中定义的方法，来跟踪广告生命周期中的各个阶段。所有这些方法也都定义在 DMInterstitialAdController.h 中，如下：

```
@protocol DMInterstitialAdControllerDelegate <DMAdViewDelegate>
@optional
// 当插屏广告被成功加载后，回调该方法
- (void)dmInterstitialSuccessToLoadAd:(DMInterstitialAdController *)dmInterstitial;
// 当插屏广告加载失败后，回调该方法
- (void)dmInterstitialFailToLoadAd:(DMInterstitialAdController *)dmInterstitial withError:(NSError *)err;

// 当插屏广告要被呈现出来前，回调该方法
- (void)dmInterstitialWillPresentScreen:(DMInterstitialAdController *)dmInterstitial;
// 当插屏广告被关闭后，回调该方法
- (void)dmInterstitialDidDismissScreen:(DMInterstitialAdController *)dmInterstitial;

// 当将要呈现出 Modal View 时，回调该方法。如打开内置浏览器。
- (void)dmInterstitialWillPresentModalView:(DMInterstitialAdController *)dmInterstitial;
// 当呈现的 Modal View 被关闭后，回调该方法。如内置浏览器被关闭。
- (void)dmInterstitialDidDismissModalView:(DMInterstitialAdController *)dmInterstitial;
// 当因用户的操作（如点击下载类广告，需要跳转到Store），需要离开当前应用时，回调该方法
- (void)dmInterstitialApplicationWillEnterBackground:(DMInterstitialAdController *)dmInterstitial;
@end
```

5.4 信息流广告的使用(具体使用请参考 Demo)

（信息流广告是针对 iphone/ipod 竖屏模式的下拉刷新定制）

信息流广告的使用，需要以下几步：

- 导入 DMFeedsAdView.h 头文件。
- 声明一个 DMFeedsAdView 的实例。
- 让您的 UIViewController 实现 DMFeedsAdViewDelegate 协议。
- 在您的 Controller 的 m 文件中，使用 PublisherId 和 PlacementId 以及负责呈现信息流广告的起始坐标 origin 来初始化一个 DMFeedsAdView，同时传递当前 controller,设置代理。
- 在合适的时候，调用 loadAd 方法，来预加载信息流广告。
- 在想要呈现信息流广告的时候，调用 present 方法，来呈现信息流广告。
- 在呈现完成后，需要重新通过 loadAd 方法，加载一条新广告，用于下次呈现。
- 注意：信息流只支持竖屏格式的展示，如果当前界面支持横屏 请做对应操作防止界面混乱。
 - 广告在播放时，如果由竖屏转为横屏，请调用 closeAd 关闭广告
 - 如果在横屏状态下拉界面，请阻止广告展现

下面的代码片段，简单描述了这个过程：

```
#import <UIKit/UIKit.h>
#import "DMFeedsAdView.h"

@interface DMFeedsAdViewController : UIViewController<DMFeedsAdViewDelegate>
{
    DMFeedsAdView *_feedsView;
}
@property (nonatomic, retain) DMFeedsAdView *feedsView;
@end
```



```

/**feeds ad view */
_feedsView = [[DMFeedsAdView alloc] initWithPublisherId:DM_PUBLISHER_ID
                                              placementId:DM_PLACEMENT_ID
                                              origin:CGPointMake(0, 20)];

_feedsView.rootViewController = self;
_feedsView.delegate = self;

[_feedsView loadAd];

```

我们建议对起始位置做 ios7 的适配，如果有 statusBar 不要忘了下移 20 像素高度。

```

- (void)reloadTableViewDataSource{

    // should be calling your tableviews data source model to reload
    // put here just for demo
    _reloading = YES;

    /** 竖屏情况开始展现广告
    present feeds ad view
    please choose the orientation you support
    UIDeviceOrientationPortrait/UIDeviceOrientationPortraitUpsideDown */

    if ([[UIApplication sharedApplication] statusBarOrientation]
        == UIInterfaceOrientationPortrait)
    {
        [_feedsView present];
    }

}

```

竖屏状态用户下拉的时候调用 present 方法（我们只支持竖屏，所以请选择您所支持的竖屏模式进行判定后进行相应的 present 操作）。

鉴于显示广告时程序界面依然可操作，所以我们提供了关闭 Feeds ad view 的方法 closeAd。需要开发者在出现视图切换时手动关闭广告。（例：用户点击 cell 进入下一个界面, 用户从竖屏转向横屏操作）

```

- (void)didRotateFromInterfaceOrientation:(UIInterfaceOrientation)fromInterfaceOrientation
{
    //close the ad view by yourself
    [_feedsView closeAd];
}

```

同时我们建议在视图消失时实现加载下一条广告

```

- (void)dmFeedsDidDismissScreen:(DMFeedsAdView *)dmFeeds
{
    [UIView beginAnimations:nil context:NULL];
    [UIView setAnimationDuration:.3];
    [self.myTable setFrame:CGRectMake(origin.x,
                                      origin.y,
                                      self.myTable.frame.size.width,
                                      self.myTable.frame.size.height)];

    [UIView commitAnimations];
    //load ad when the ad view dismiss
    [_feedsView loadAd];
}

```

关于 DMFeedsAdView 的释放:

在这里强烈建议您, 在释放 DMFeedsAdView 之前, 将其 delegate 属性设置为 nil。

设置可选参数:

与 DMAView 相同, 也可以为 DMFeedsAdView 设置一些可选参数, 来提高广告投放的精准性。这些参数的设置方法都定义在 DMFeedsAdView.h 中,

如下:

```

// 设置地理位置信息
- (void)setLocation:(CLLocation *)location;

// 设置邮编
- (void)setPostcode:(NSString *)postcode;

// 设置关键字
- (void)setKeywords:(NSString *)keywords;

// 设置用户年龄
- (void)setUserBirthday:(NSString *)userBirthday;

// 设置用户性别
- (void)setUserGender:(DMUserGenderType)userGender;

```

实现 DMFeedsAdViewDelegate:

您可以通过实现 DMFeedsAdViewDelegate 中定义的方法, 来跟踪广告生命周期中的各个阶段。所有这些方法也都定义在 DMFeedsAdView.h 中, 如下:

```

@protocol DMFeedsAdViewDelegate
@optional
// Sent when an ad request success to load an ad
- (void)dmFeedsSuccessToLoadAd:(DMFeedsAdView *)dmFeeds;
// Sent when an ad request fail to load an ad
- (void)dmFeedsFailToLoadAd:(DMFeedsAdView *)dmFeeds withError:(NSError *)err;
// Sent when the feeds ad is clicked
- (void)dmFeedsDidClicked:(DMFeedsAdView *)dmFeeds;

// Sent just before presenting the user a modal view
- (void)dmFeedsWillPresentModalView:(DMFeedsAdView *)dmFeeds;
// Sent just after dismissing the modal view
- (void)dmFeedsDidDismissModalView:(DMFeedsAdView *)dmFeeds;
// Sent just before the application will background or terminate because the user's action
// (Such as the user clicked on an ad that will launch App Store).
- (void)dmFeedsApplicationWillEnterBackground:(DMFeedsAdView *)dmFeeds;

// Sent just before presenting an feeds ad view
- (void)dmFeedsWillPresentScreen:(DMFeedsAdView *)dmFeeds;
// Sent just after dismissing a feeds ad view
- (void)dmFeedsDidDismissScreen:(DMFeedsAdView *)dmFeeds;
@end

```

5.5 开屏广告的使用

开屏广告的使用场景是在应用刚刚开启时，在 SDK 内分为**缓存开屏**和**实时开屏**两种。缓存开屏的加载机制为，本次加载，缓存本次的广告应答以及相关资源，下次开启时展现。而实时开屏在每次开启时都会实时请求广告并加载和缓存资源，在超时时间前加载完成即自动展现。详细使用方法可见 Sample 中的 App 委派类的 `application:didFinishLaunchingWithOptions:` 方法中使用的示例代码。

	广告请求流程	优点	缺点
缓存开屏	当次缓存，下次展现	无需等待广告loading,体验好。	填充→展现流失率较高
实时开屏	当次加载，当次展现，超时放弃（超时时限开发者可设）	填充→展现流失率较低。	用户需要等待

如果您想让开屏广告与您的启动图片融合的更好，获得更好的用户体验，请参考 Sample 项目中的做法。

设置可选参数：

与其他广告形式相同，也可以为 `DMSplashAdController` 设置一些可选参数，来提高广告投放的精准性。这些参数的设置方法都定义在 `DMSplashAdController.h` 中，如下：

```
// 设置地理位置信息
- (void)setLocation:(CLLocation *)location;

// 设置邮编
- (void)setPostcode:(NSString *)postcode;

// 设置关键字
- (void)setKeywords:(NSString *)keywords;

// 设置用户年龄
- (void)setUserBirthday:(NSString *)userBirthday;

// 设置用户性别
- (void)setUserGender:(DMUserGenderType)userGender;
```

实现 `DMSplashAdControllerDelegate`：

您可以通过实现 `DMSplashAdControllerDelegate` 中定义的方法，来跟踪广告生命周期中的各个阶段。所有这些方法也都定义在 `DMSplashAdController.h` 中，如下：

```
@protocol DMSplashAdControllerDelegate
@optional
// 当开屏广告加载成功后，回调该方法
- (void)dmSplashAdSuccessToLoadAd:(DMSplashAdController *)dmSplashAd;
// 当开屏广告加载失败后，回调该方法
- (void)dmSplashAdFailToLoadAd:(DMSplashAdController *)dmSplashAd withError:(NSError *)err;

// 当插屏广告要被呈现出来前，回调该方法
- (void)dmSplashAdWillPresentScreen:(DMSplashAdController *)dmSplashAd;
// 当插屏广告被关闭后，回调该方法
- (void)dmSplashAdDidDismissScreen:(DMSplashAdController *)dmSplashAd;
@end
```

6 开发者工具的使用

6.1 评价提醒

只需简单几步，即可实现好评提醒功能：

```
// 检查更新提醒，此处使用的是测试ID，请登陆多盟官网（www.domob.cn）获取新的ID
DMTools *_dmTools = [[DMTools alloc] initWithPublisherId:@"560JyM1ouMGoULfJaL"];
[_dmTools checkRateInfo];
[_dmTools release];
```

若需更多帮助，请登陆多盟官网（<http://www.domob.cn/help/index.htm>）。

7 其他

更详细的使用方法，请参考发行文件中的 Sample 程序。

如还有疑问，欢迎随时发邮件到 support@domob.cn 获得更多帮助。