### 风灵创景广告SDK iOS接入文档

|  |  |  |
| --- | --- | --- |
| 文档版本 | 修订说明 | 修订日期 |
| v1.0.0 | 开屏广告和横幅广告 | 2019.5.15 |
| V1.1 | 添加信息流广告、插屏广告、激励广告 | 2019.7.8 |
| V1.3 | 横幅支持信息流广告 | 2019.10.14 |
| V2.0 | 适配iOS14.5隐私 | 2021.02.20 |

目录

[1. iOS SDK接入 1](#_Toc64641152)

[2. 运行环境配置 1](#_Toc64641154)

[3. Xcode编译选项设置 1](#_Toc64641155)

[4. SDK接口类介绍与广告接入 1](#_Toc64641156)

[4.1 全局设置(FelinkAdSDK) 1](#_Toc64641161)

[4.2 开屏广告(FelinkAgSplash) 1](#_Toc64641162)

[4.3 横幅广告(FelinkAgBanner) 1](#_Toc64641163)

[4.5 信息流广告(FelinkAgNative) 1](#_Toc64641164)

[4.6 插屏广告(FelinkAgInterstitial) 1](#_Toc64641165)

[4.7激励视频广告(FelinkAgRewardVideo) 1](#_Toc64641166)

[5 Appstore 隐私数据获取说明 1](#_Toc64641167)

### iOS SDK接入

### pod方式接入，只需配置pod环境，在podfile文件中加入以下代码即可接入成功。

pod 'FelinkAdSDK'

更多关于pod方式的接入请参考[gitthub地址](https://github.com/lrbInfelink/FelinkAdSDK);

### 运行环境配置

* 1. 支持系统 iOS 8.X 及以上;
  2. SDK编译环境 Xcode 10, Base SDK 12.2;
  3. 支持架构：i386, x86-64, armv7, arm64

### Xcode编译选项设置

#### 添加权限

* 工程plist文件设置，点击右边的information Property List后边的 "+" 展开添加 App Transport Security Settings，先点击左侧展开箭头，再点右侧加号，Allow Arbitrary Loads 选项自动加入，修改值为 YES。

<key>NSAppTransportSecurity</key>

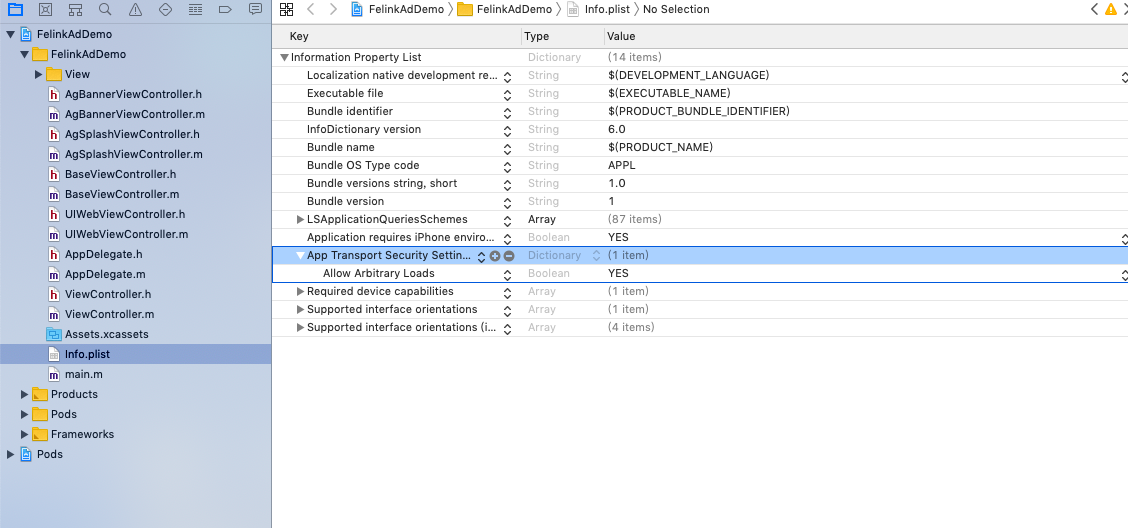
<dict>

<key>NSAllowsArbitraryLoads</key>

<true/>

</dict>

具体操作如图



* 工程plist文件设置，点击右边的information Property List后边的 "+" 展开

添加 LSApplicationQueriesSchemes。

<key>NSAppTransportSecurity</key>

<array>

<string>tbopen</string>

<string>tmall</string>

<string>itmall</string>

<string>baidumobads</string>

<string>yuedongapp</string>

<string>openapp.jdmobile</string>

<string>openApp.jdMobile</string>

<string>sinaweibo</string>

<string>sinaweibohd</string>

<string>sinaweibosso</string>

<string>sinaweibohdsso</string>

<string>weibosdk</string>

<string>weibosdk2.5</string>

<string>mqqOpensdkSSoLogin</string>

<string>mqqopensdkapiV2</string>

<string>mqqopensdkapiV3</string>

<string>wtloginmqq2</string>

<string>mqq</string>

<string>mqqapi</string>

<string>mqqOpensdkSSoLogin</string>

<string>mqqconnect</string>

<string>mqqopensdkdataline</string>

<string>mqqopensdkgrouptribeshare</string>

<string>mqqopensdkfriend</string>

<string>mqqopensdkapi</string>

<string>mqqopensdkapiV2</string>

<string>mqqopensdkapiV3</string>

<string>mqzoneopensdk</string>

<string>wtloginmqq</string>

<string>wtloginmqq2</string>

<string>mqqwpa</string>

<string>mqzone</string>

<string>mqzonev2</string>

<string>mqzoneshare</string>

<string>wtloginqzone</string>

<string>mqzonewx</string>

<string>mqzoneopensdkapiV2</string>

<string>mqzoneopensdkapi19</string>

<string>mqzoneopensdkapi</string>

<string>mqzoneopensdk</string>

<string>weixin</string>

<string>wechat</string>

<string>alipay</string>

<string>alipayshare</string>

<string>taobao</string>

<string>itaobao</string>

<string>dianping</string>

<string>jummei</string>

<string>jumeimall</string>

<string>meituan0000</string>

<string>iMeituan</string>

<string>imeituan</string>

<string>suning</string>

<string>vipshop</string>

<string>yhd</string>

<string>lightsky</string>

<string>shark.video</string>

<string>bobo</string>

<string>snssdk32</string>

<string>pptv</string>

<string>bilibili</string>

<string>kugouURL</string>

<string>gaeagj</string>

<string>qqnews</string>

<string>zhihu</string>

<string>doubanradio</string>

<string>sohunews</string>

<string>yanxuan</string>

<string>wccbyihaodian</string>

<string>dangdang</string>

<string>kaola</string>

<string>ctrip</string>

<string>kuaikanmanhua</string>

<string>gugutouchmanga</string>

<string>qrxs</string>

<string>lvmama</string>

<string>wbmain</string>

<string>jcnhers</string>

<string>shanbay</string>

<string>mailmaster</string>

<string>yddict</string>

<string>yixin</string>

<string>ydcourse</string>

<string>ntesopen</string>

<string>mogujie</string>

<string>YLVideos</string>

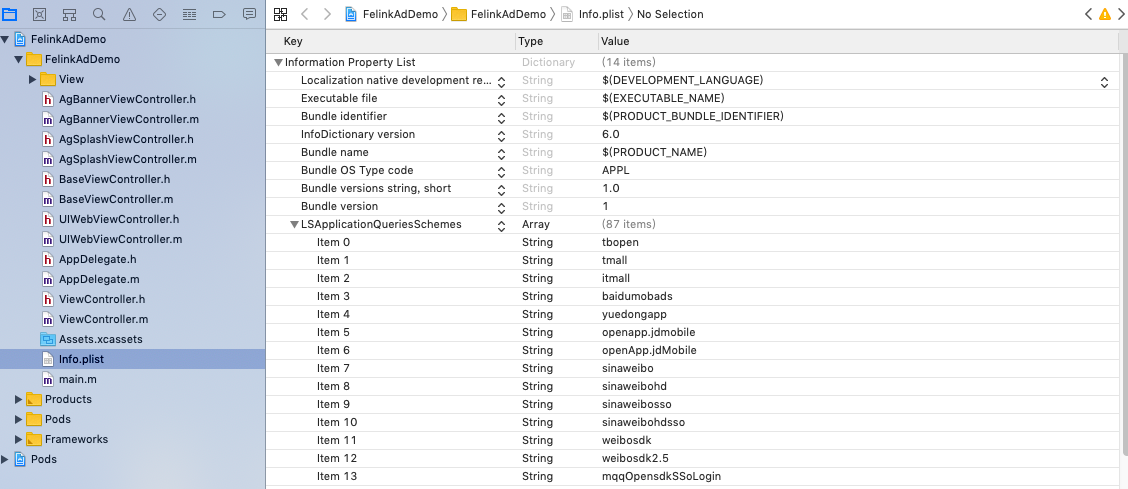
<string>VideoPaper</string>

<string>wdkhema</string>

<string>bytedance</string>

</array>

具体操作如图



* 工程plist文件设置，点击右边的information Property List后边的 "+" 展开

添加 SKAdNetworkItems。

<key>SKAdNetworkItems</key>

<array>

<dict>

<key>SKAdNetworkIdentifier</key>

<string>238da6jt44.skadnetwork</string>

</dict>

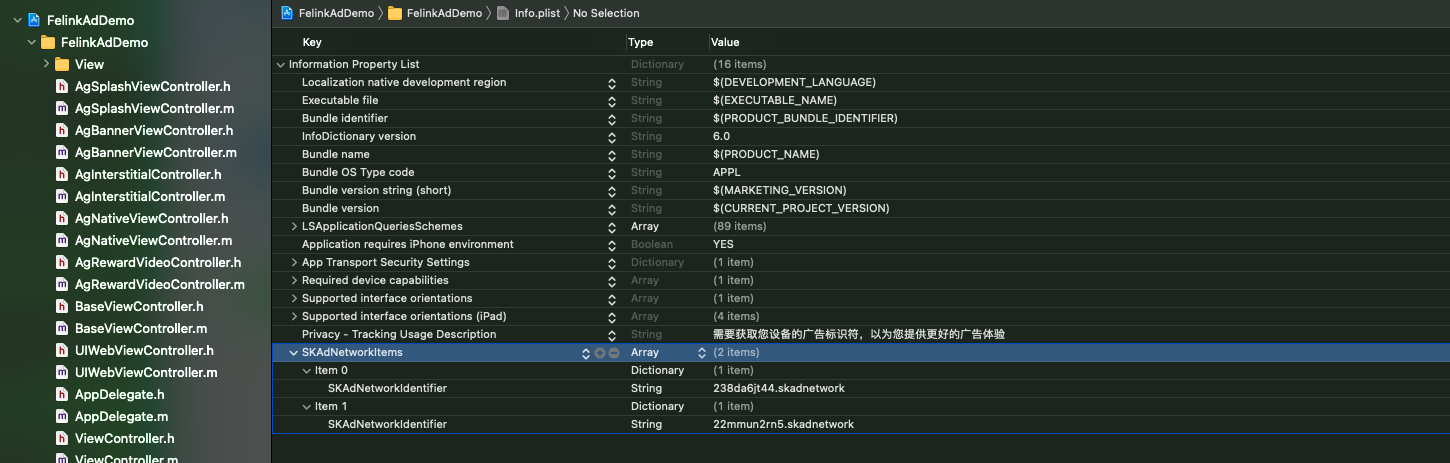
<dict>

<key>SKAdNetworkIdentifier</key>

<string>22mmun2rn5.skadnetwork</string>

</dict>

</array>

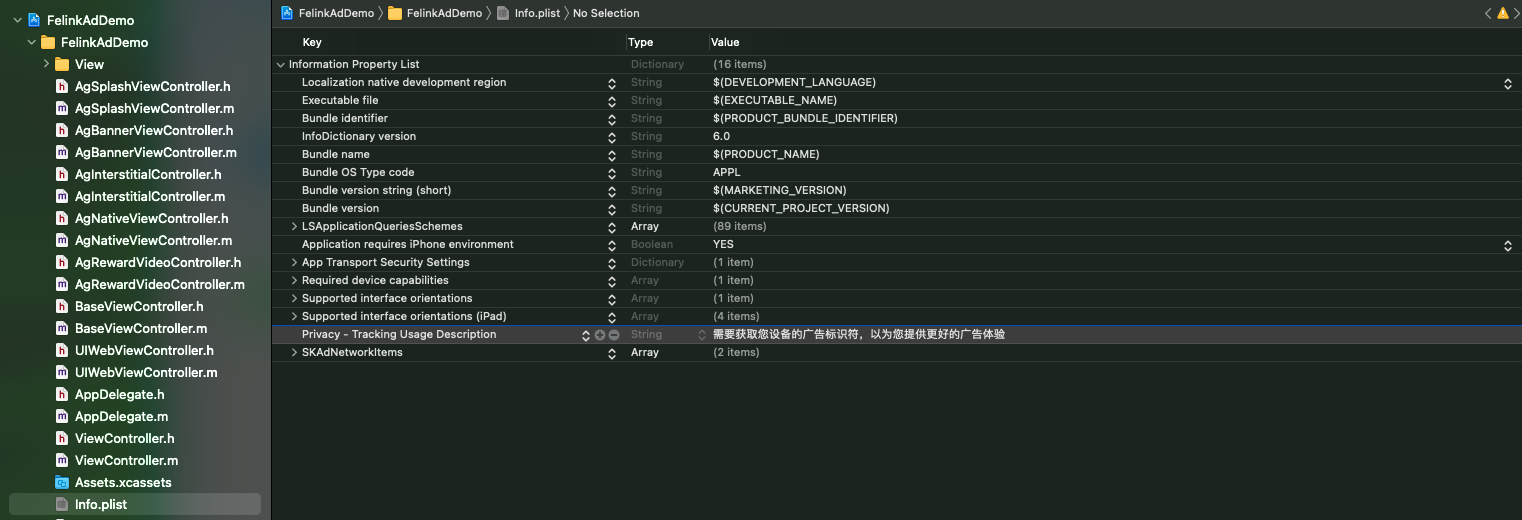


* 工程plist文件设置，点击右边的information Property List后边的 "+" 展开

添加 NSUserTrackingUsageDescription ,这个项是**可选的**，建议添加。

<key>NSUserTrackingUsageDescription</key>

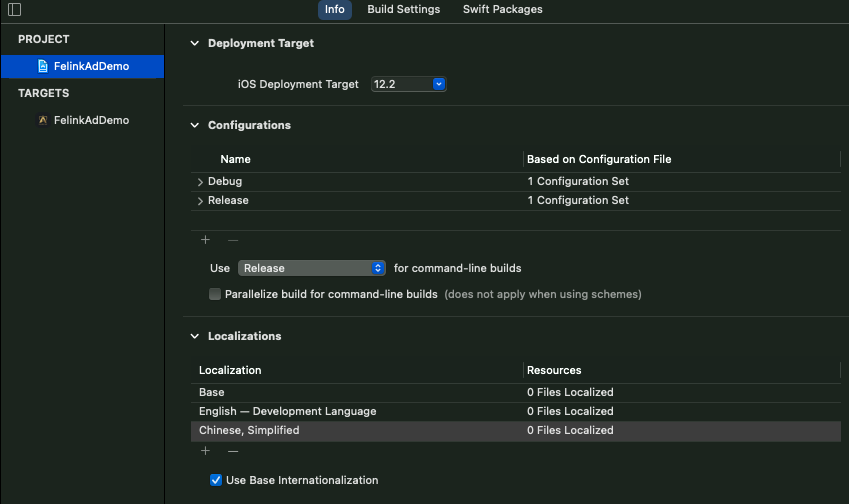
<string>需要获取您设备的广告标识符，以为您提供更好的广告体验</string>



* 本地化\国际化配置

注意 : 开发者必须在这里设置所支持的语言,否则会有语言显示的问题.

例如 : 支持中文 添加 Chines



### SDK接口类介绍与广告接入



### 全局设置(FelinkAdSDK)

FelinkAdSDK 类是整个 SDK 设置的入口和接口，可以设置 SDK 的一些全局信息，提供类方法获取设置结果。

#### 4.1.1 接口说明

目前接口提供以下几个类方法

+(void)initAppID:(NSString \*)appID; // 设置应用的 appID

+(void)setDebug:(BOOL)debug; // console上显示log信息

#### 4.1.2 使用

SDK 需要在 AppDelegate 的方法 - (BOOL)application:(UIApplication \*)application didFinishLaunchingWithOptions:(NSDictionary \*)launchOptions 里进行初始化

其中以下设置是 **必须** 的，应用相关 appID 设置：

[FelinkAdSDK initAppID:@"xxxxxx"];

更多使用方式可以参见 SDK Demo 工程

### 4.2 开屏广告(FelinkAgSplash)

* \*\*类型说明：\*\*开屏广告主要是 APP 启动时展示的全屏广告视图，开发只要按照接入标准就能够展示设计好的视图
* 注意:先调用showInWindow,再调用requestAd

#### 4.2.1 FelinkAgSplash接口说明

@interface FelinkAgSplash : NSObject

/\*\*

开屏启动的 状态回调

\*/

@property (nonatomic ,weak) id<FelinkAgSplashDelegate> delegate;

/\*\*

获取代码位id

\*/

@property (nonatomic,readonly,copy) NSString\* adPid;

/\*\*

设置广告View 背景颜色

\*/

@property (nonatomic ,strong) UIColor \*adBackgroundColor;

/\*\*

在bottomView != nil 时 底部 log 区域 高度；

gdt 还需要 bottomView高度不能为0

bottomHeight >= bottomView高度

\*/

@property (nonatomic ,assign) CGFloat bottomHeight;

/\*\*

初值

@param adPid 广告位

@param timeout 请求数据超时 建议3秒

\*/

-(instancetype)initWithAdPid:(NSString \*)adPid timeout:(NSTimeInterval)timeout;

/\*\*

请求广告

\*/

-(void)requestAd;

/\*\*

//展示开屏广告 或广告将展示 window 或 keyView

@param keyWindow window(必须的)

@param keyView 用来展示广告 可以nil, keyView 自动移除

@param bottomView 展示logo 可以nil

@param controller 来push广告 可以nil

\*/

- (void)showInWindow:(UIWindow \* \_Nonnull)keyWindow view:(UIView \*\_Nullable)keyView withBottomView:(UIView \* \_Nullable)bottomView controller:(UIViewController \* \_Nullable)controller;

@end

#### 4.2.2 FelinkAgSplash回调说明

@protocol FelinkAgSplashDelegate <NSObject>

@optional

//收到广告

- (void)felinkAdSplashDidReceiveAd:(FelinkAgSplash \*)ad;

//成功展示广告

- (void)felinkAdSplashSuccessPresentScreen:(FelinkAgSplash \*)ad;

//广告失败

- (void)felinkAdSplashFail:(FelinkAgSplash \*)ad error:(NSError \*)error;

// 广告被点击

- (void)felinkAdSplashClicked:(FelinkAgSplash \*)ad;

//结束广告

- (void)felinkAdSplashDidDismissScreen:(FelinkAgSplash \*)ad;

//结束广告详细页

- (void)felinkAdSplashDidDismissDetail:(FelinkAgSplash \*)ad;

// 广告跳到用户处理 若开发者处理改事件 返回YES；否则返回 NO，SDK会处理

- (BOOL)felinkAdSplashDirectClicked:(FelinkAgSplash \*)ad data:(NSDictionary \*)data;

//用户点击跳过广告 (并不所有广告跳过都被调用)

- (void)felinkAdSplashSkipped:(FelinkAgSplash \*)ad;

@end

#### 4.2.3 实例

- (BOOL)application:(UIApplication \*)application didFinishLaunchingWithOptions:(NSDictionary \*)launchOptions {

// Override point for customization after application launch.

[FelinkAdSDK initAppID:@"fb6e8f0b4507491c848eebf95dd56d06"];

self.window = [[UIWindow alloc] initWithFrame:[UIScreen mainScreen].bounds];

\_viewController = [[ViewController alloc] init];

\_window.rootViewController =[[UINavigationController alloc] initWithRootViewController:\_viewController];

[\_window makeKeyAndVisible];

\_loadingView = [[LoadingView alloc] initWithFrame:[UIScreen mainScreen].bounds];

[[UIApplication sharedApplication].keyWindow addSubview:\_loadingView];

\_felinkAgSplash = [[FelinkAgSplash alloc] initWithAdPid:@"100958" timeout:3];

[\_felinkAgSplash showInWindow:[UIApplication sharedApplication].keyWindow view:\_loadingView withBottomView:nil controller:\_viewController];

[\_felinkAgSplash requestAd];

return YES;

}

### 4.3 横幅广告(FelinkAgBanner)

* 类型说明：横幅广告主要是满足媒体多元化需求开发广告视图，开发只要按照接入标准就能够展示设计好的视图。注意:目前不支持广告轮播,如需要每次用户进入程序调request方法
* 注意:先调用showInView,再调用requestAd

#### 4.3.1 FelinkAgBanner接口说明

@interface FelinkAgBanner : NSObject

@property (nonatomic ,weak) id<FelinkAgBannerDelegate> delegate;

/\*\*

获取代码位id

\*/

@property (nonatomic,readonly,copy) NSString\* adPid;

/\*\*

\* banner 是否可以使用定位，默认YES

\*/

@property (nonatomic,assign) BOOL enableLocation;

/\*\*

\* banner 填充方式

\*/

@property (nonatomic,assign) FelinkAdBannerStyle bannerStyle;

/\*\*

\* bannerView 高度是否适应广告大小，默认YES

\*/

@property (nonatomic,assign) BOOL autoUpdateBannerSize;

/\*\*

\* banner 真实大小

\*/

@property (nonatomic,readonly,assign) CGSize bannerSize;

/\*\*

初值

@param adPid 广告位

@param timeout 请求数据超时 建议5秒

@param size 广告展示区域 推荐 (宽/高) 640:100 640:320 600:500

@return <#return value description#>

\*/

-(instancetype)initWithAdPid:(NSString \*)adPid timeout:(NSTimeInterval)timeout size:(CGSize)size;

/\*\*

请求广告

\*/

-(void)requestAd;

/\*\*

应用启动时展示开屏广告

@param bannerView <#keyView description#>

@param controller <#controller description#>

\*/

- (void)showInView:(UIView \*)bannerView controller:(UIViewController \*)controller;

@end

#### 4.3.2 FelinkAgBanner回调说明

@protocol FelinkAgBannerDelegate <NSObject>

@optional

//广告控件需要到屏幕上(使用广电通横幅广告必须实现)

- (void)felinkAdBannerNeedAddScreen:(FelinkAgBanner \*)ad;

//收到广告

- (void)felinkAdBannerDidReceiveAd:(FelinkAgBanner \*)ad;

//成功展示广告

- (void)felinkAdBannerSuccessPresentScreen:(FelinkAgBanner \*)ad;

//广告失败

- (void)felinkAdBannerFail:(FelinkAgBanner \*)ad error:(NSError \*)error;

// 广告被点击

- (void)felinkAdBannerClicked:(FelinkAgBanner \*)ad;

//结束广告

- (void)felinkAdBannerDidDismissScreen:(FelinkAgBanner \*)ad;

//结束广告详细页

- (void)felinkAdBannerDidDismissDetail:(FelinkAgBanner \*)ad;

// 广告跳到用户处理 若开发者处理改事件 返回YES；否则返回 NO，SDK会处理

- (BOOL)felinkAdBannerDirectClicked:(FelinkAgBanner \*)ad data:(NSDictionary \*)data;

//用户关闭广告 若开发者处理改事件 返回YES；否则返回 NO，SDK会处理

- (BOOL)felinkAdBannerDidClose:(FelinkAgBanner \*)ad;

@end

#### 4.3.3 实例

\_banner = [[UIView alloc] initWithFrame:CGRectMake(0, 0, self.view.frame.size.width, self.view.frame.size.width \* 0.5)];

[self.view addSubview:\_banner];

\_felinkAgBanner = [[FelinkAgBanner alloc] initWithAdPid:self.aPid timeout:15.0 size:\_size];

\_felinkAgBanner.delegate = self;

[\_felinkAgBanner showInView:\_banner controller:self];

[\_felinkAgBanner requestAd];

### 4.5 信息流广告(FelinkAgNative)

* 类型说明： 信息流广告即普通 feed 流广告，是在feed流场景下的原生广告。
* 使用说明： 在SDK里只需要使用 FelinkAgNative就可以获取信息流广告。SDK 提供信息流广告的数据绑定、点击事件的上报，用户可自行定义信息流广告展示形态与布局

#### 4.5.1 FelinkAgNative接口说明

@interface FelinkAgNative : NSObject

@property (nonatomic, weak) id<FelinkAgNativeDelegate> delegate;

/\*\*

\* 获取代码位id

\*/

@property (nonatomic, readonly, copy) NSString \*adPid;

/\*\*

\* 广告正在请求中

\*/

@property (nonatomic, readonly, assign) BOOL adRequesting;

/\*\*

\*用于点击广告后跳转

\*/

@property (nonatomic, weak) UIViewController \*viewController;

/\*\*

初值

@param adPid 广告位

@param timeout 请求数据超时 建议5秒

@param size 广告展示区域 推荐 (宽/高) 320:50 320:160 300:250

@return <#return value description#>

\*/

- (instancetype)initWithAdPid:(NSString \*)adPid timeout:(NSTimeInterval)timeout size:(CGSize)size;

/\*\*

\* 推荐请求广告不超3条

\* 最多10条

\*/

-(void)requestAd:(NSInteger)adCount;

/\*\*

\* 先注册广告对象，否则广告展示、点击等无效

\*/

-(void)registerWithNativeObject:(FelinkAgNativeObject \*)nativeObject;

@end

#### 4.5.2 FelinkAgNative回调说明

@protocol FelinkAgNativeDelegate <NSObject>

//收到广告

- (void)felinkAdNativeDidReceiveAd:(FelinkAgNative \*)ad nativeObjects:(NSArray \*)nativeObjects;

@optional

//大图完成加载 该方法不一定被回调;同一个nativeObject，可能一次调用都没有，也可能多次被调用

- (void)felinkAdNativeDidLoadMainImage:(FelinkAgNative \*)ad nativeObject:(FelinkAgNativeObject \*)nativeObject;

//成功展示广告

- (void)felinkAdNativeSuccessPresentScreen:(FelinkAgNative \*)ad;

//广告失败

- (void)felinkAdNativeFail:(FelinkAgNative \*)ad error:(NSError \*)error;

// 广告被点击

- (void)felinkAdNativeClicked:(FelinkAgNative \*)ad;

//结束广告详细页

- (void)felinkAdNativeDidDismissDetail:(FelinkAgNative \*)ad;

// 广告跳到用户处理 若开发者处理该事件 返回YES；否则返回 NO，SDK会处理

- (BOOL)felinkAdNativeDirectClicked:(FelinkAgNative \*)ad data:(NSDictionary \*)data;

@end

#### 4.5.3 实例

\_felinkAgNative = [[FelinkAgNative alloc] initWithAdPid:self.aPid timeout:15.0 size:CGSizeMake([UIScreen mainScreen].bounds.size.width, 150)];

\_felinkAgNative.delegate = self;

\_felinkAgNative.viewController = self

[\_felinkAgNative requestAd:3];

### 4.6 插屏广告(FelinkAgInterstitial)

* 类型说明：原生插屏广告是为满足媒体多元化需求而开发的一种原生广告。
* 使用说明：SDK可提供数据绑定、点击事件的上报、响应回调。

#### 4.6.1 FelinkAgInterstitial接口说明

@interface FelinkAgInterstitial : NSObject

@property (nonatomic ,weak) id<FelinkAgInterstitialDelegate> delegate;

/\*\*

获取代码位id

\*/

@property (nonatomic,readonly,copy) NSString\* adPid;

/\*\*

\* 广告正在请求中

\*/

@property (nonatomic,readonly,assign) BOOL adRequesting;

/\*\*

初值

@param adPid 广告位

@param timeout 请求数据超时 建议3秒

\*/

-(instancetype)initWithAdPid:(NSString \*)adPid timeout:(NSTimeInterval)timeout;

/\*\*

\* 请求广告

\*/

-(void)requestAd;

/\*\*

\* 展示广告，在调用之前 请先调用isAdValid

\*/

-(void)showFromViewController:(UIViewController \*)vc;

/\*\*

\* 插屏广告预加载是否完成

\*/

@property (nonatomic, readonly) BOOL isAdValid;

/\*

\* 收到广告

\* 1.获取直投定制信息

\*/

-(NSDictionary \*)getCurrentAdExtInfo;

@end

#### 4.6.2 FelinkAgInterstitial回调说明

@protocol FelinkAgInterstitialDelegate <NSObject>

//收到广告

- (void)felinkAgInterstitialDidReceiveAd:(FelinkAgInterstitial \*)ad;

//成功展示广告

- (void)felinkAgInterstitialSuccessPresentScreen:(FelinkAgInterstitial \*)ad;

//广告失败

- (void)felinkAgInterstitialFail:(FelinkAgInterstitial \*)ad error:(NSError \*)error;

// 广告被点击

- (void)felinkAgInterstitialClicked:(FelinkAgInterstitial \*)ad;

//结束广告

- (void)felinkAgInterstitialDidDismissScreen:(FelinkAgInterstitial \*)ad;

//结束广告详细页

- (void)felinkAgInterstitialDidDismissDetail:(FelinkAgInterstitial \*)ad;

// 广告跳到用户处理 若开发者处理该事件 返回YES；否则返回 NO，SDK会处理

- (BOOL)felinkAgInterstitialDirectClicked:(FelinkAgInterstitial \*)ad data:(NSDictionary \*)data;

//用户关闭广告

- (void)felinkAgInterstitialDidClose:(FelinkAgInterstitial \*)ad;

@end

#### 4.6.3 实例

\_felinkAgInterstitial = [[FelinkAgInterstitial alloc] initWithAdPid:self.aPid timeout:15.0];

\_felinkAgInterstitial.delegate = self;

[\_felinkAgInterstitial requestAd];

### 4.7激励视频广告(FelinkAgRewardVideo)

* 类型说明：激励视频广告是一种全新的广告形式，用户可选择观看视频广告以换取有价物，例如虚拟货币、应用内物品和独家内容等等；这类广告的长度为 15-30 秒，不可跳过，且广告的结束画面会显示结束页面，引导用户进行后续动作。
* 使用说明：SDK可提供数据绑定、点击事件的上报、响应回调。

#### 4.7.1 FelinkAgRewardVideo接口说明

@interface FelinkAgRewardVideo : NSObject

@property (nonatomic, weak) id<FelinkAgRewardVideoDelegate> delegate;

/\*\*

获取代码位id

\*/

@property (nonatomic, readonly, copy) NSString \*adPid;

/\*\*

\* 广告正在请求中

\*/

@property (nonatomic,readonly,assign) BOOL adRequesting;

/\*\*

初值

@param adPid 广告位

@param timeout 请求数据超时 建议5秒

@return <#return value description#>

\*/

- (instancetype)initWithAdPid:(NSString \*)adPid timeout:(NSTimeInterval)timeout;

/\*\*

请求广告

\*/

-(void)requestAd;

/\*\*

展示广告

\*/

-(void)showFromViewController:(UIViewController \*)vc;

/\*\*

\* 激励视频广告预加载是否完成,激励视频广告的展示存在有效期，单次检索后须在一定时间内展示在页面上

\*/

@property (nonatomic, readonly) BOOL isAdValid;

@end

#### 4.7.2 FelinkAgRewardVideo回调说明

@protocol FelinkAgRewardVideoDelegate <NSObject>

@optional

//收到广告

- (void)felinkAgRewardVideoDidReceiveAd:(FelinkAgRewardVideo \*)ad;

//广告失败

- (void)felinkAgRewardVideoFail:(FelinkAgRewardVideo \*)ad error:(NSError \*)error;

//视频播放页即将打开

- (void)felinkAgRewardVideoSuccessPresentScreen:(FelinkAgRewardVideo \*)ad;

//广告完成播放

- (void)felinkAgRewardVideoDidFinish:(FelinkAgRewardVideo \*)ad;

//用户点击下载/查看详情

- (void)felinkAgRewardVideoDidClick:(FelinkAgRewardVideo \*)ad;

//用户点击关闭

- (void)felinkAgRewardVideoDidClose:(FelinkAgRewardVideo \*)ad;

//用户获取奖励

- (void)felinkAgRewardServerRewardDidSucceed:(FelinkAgRewardVideo \*)ad reward:(FelinkAgRewardObject \*)reward;

@end

#### 4.7.3 实例

\_felinkAgRewardVideo = [[FelinkAgRewardVideo alloc] initWithAdPid:self.aPid timeout:15.0];

\_felinkAgRewardVideo.delegate = self;

[\_felinkAgRewardVideo requestAd];

### 5 Appstore 隐私数据获取说明

#### 5.1. 收集的数据类型

| 据类型 | 收集情况 | 备注 |
| --- | --- | --- |
| **联系信息 Contact Info** 姓名 Name 电子邮件地址 Email Address 电话号码 Phone Number 实际地址 Physical Address 其他用户联系信息 Other User Contact Info | 不收集 |  |
| **健康与健身 Health and Fitness** 健康 Health 健身 Fitness | 不收集 |  |
| **财务信息 Financial Info** 付款信息 Payment Info 信用信息 Credit Info 其他财务信息 Other Financial Info | 不收集 |  |
| **位置 Location** 精确位置 Precise Location 粗略位置 Coarse Location | 可选收集 不收集 | **广告商会获取地理位置信息用于广告投放与反作弊。** 应用被用户授予地理位置权限时，广告商会获取地理位置信息，用于广告定向与反作弊； 当应用不获取地理位置权限时，广告商不会主动获取地理位置权限及地理位置信息。 |
| **敏感信息 Sensitive Info** | 不收集 |  |
| **联系人 Contacts** | 不收集 |  |
| **用户内容 User Content** 电子邮件或短信内容 Emails or Text Messages 照片或视频 Photos or Videos 音频数据 Audio Data 游戏内容 Gameplay Content 客户支持 Customer Support 其他用户内容 Other User Content | 不收集 |  |
| **浏览历史记录 Browsing History** | 不收集 |  |
| **搜索历史记录 Search History** | 不收集 |  |
| **标识符 Identifiers** 用户ID User ID 设备ID Device ID | 可选收集 | **用户ID** 仅用于功能-奖励视频服务器端验证回调。开发者在广告请求期间向广告商发送他们的 用户名（基本上是一系列数字，广告商不知道它与该应用内的内部用户id或内部用户 名的关系。广告商只接收这些数据，不检查、不验证、不存储、不使用）。当用户完 全观看奖励视频时，广告商将把这个用户id发送回开发者的服务器。如果开发者不使用 此功能，则开发者不需要传递此类数据。 **设备ID** 当应用被用户授予广告追踪权限时，广告商将获取idfa用于广告归因与追踪。 |
| **购买项目 Purchases** 购物历史 Purchase History | 不收集 |  |
| **使用数据 Usage Data** 产品交互 Product Interaction 广告数据 Advertising Data 其他使用数据 Other Usage Data | 不收集 收集 不收集 | 广告商广告将统计下列广告数据，以用于广告主统计投放结果。 展示 、点击 、转化 |
| **诊断 Diagnostics** 崩溃数据 Crash Data 性能数据 Performance Data 其他诊断数据 Other Diagnostic Data | 收集 收集 不收集 | **崩溃数据** 广告商将收集广告商SDK带来的崩溃信息，以此来优化代码缺陷。 **性能数据** 广告商将收集SDK运行过程中性能数据，以优化广告商SDK的性能。 |
| **其他数据 Other Data** 其他数据类型 Other Data Types | 收集 | 技术上我们还会收集一些设备信息（例如，设备型号、操作系统及版本、时区、网络类型等） |

#### 5.2. 数据使用目的

苹果明示了一系列的数据使用目的，需要开发者根据App中数据的使用情况进行披露。如下表所示，广告商依据自身功能，列举了数据的使用目的，以供参考。

**基于媒体与广告商的合作目的，广告商收集数据的主要目的是用于“第三方广告”、“APP功能”（如流量变现的功能）；  
其他使用目的基于媒体自己的使用情况而定。**

| 使用目的 | 基于接入广告商的功能，选择数据的使用目的 |
| --- | --- |
| 第三方广告 Third-Party Advertising | **是** |
| 开发者广告或营销内容 Developer’s Advertising or Marketing | 如您的App使用数据用于显示应用中的第一方广告（包括与第三方共享以显示广告）或其他营销目的**是**，请选择 **是** |
| 分析 Analytics | 广告数据会用于分析，请选择 **是** |
| 产品个性化 Product Personalization | 如您的App将数据用于任何产品个性化功能，请选择 **是** |
| App 功能 App Functionality | **是** |
| 其他用途 Other Purposes | 如您的App出于其他目的使用数据，请选择 **是** |

#### 5.3. 与用户身份的关联性

通过广告商收集的数据类型通常连接到诸如设备ID（如果可用）之类的标识符，或者与其他标识符（如配置时由客户发布的ID）相连。  
因此，在询问数据是否链接到身份时，应用程序开发人员应选择 **是**。

#### 5.4. 关于用户追踪

根据苹果对追踪的定义，当应用被用户授予广告追踪权限时，广告商将获取IDFA用于广告归因与追踪。