

广点通联盟 IOS SDK

接入说明文档 V2.0

| | |
|--------------------------------------|----|
| 1. 概述 | 2 |
| 2. SDK 嵌入 | 2 |
| 2.1. 步骤 1: 添加 SDK 到工程中 | 2 |
| 2.2. 步骤 2: 为工程添加相应的 Frameworks | 3 |
| 2.3. 步骤 3: 更改导入静态库设置 | 4 |
| 3. 接入代码 | 5 |
| 3.1. Banner 广告接入代码 | 5 |
| 3.2. 插屏广告接入代码 | 8 |
| 4. 广告位创建流程 | 9 |
| 5. SDK 相关问题排查 | 10 |
| 6. 注意事项 | 10 |

1. 概述

本文档帮助 iOS 应用开发者在程序中快速植入广点通平台提供的广告。作为开发者，您只需要进行简单配置，就可以在您的应用中显示定制的广告。

关于 SDK 的具体使用方法，请仔细阅读下面的文档。

操作系统：Mac OS X 10.6.2 及以上版本

开发工具：Xcode 4.5 及以上版本

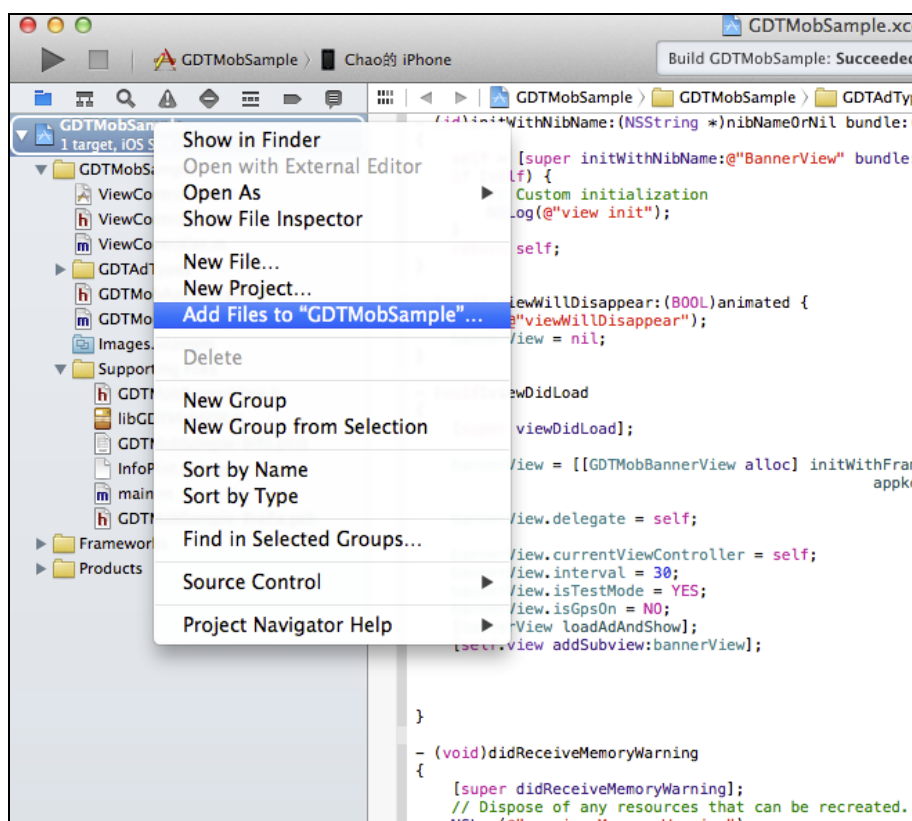
支持设备：iPhone/iPod Touch (iOS 4.3 及以上版本)

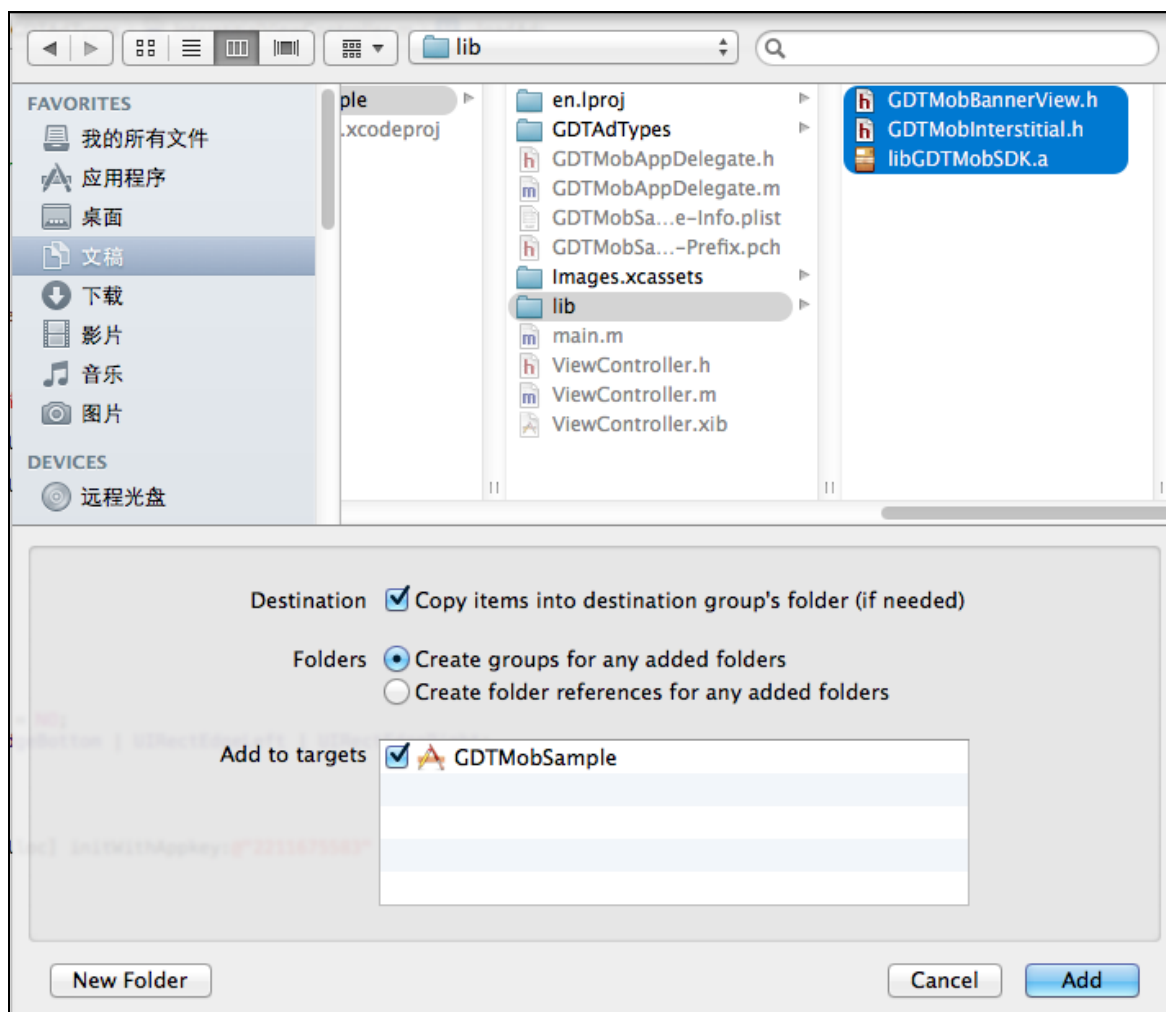
2. SDK 嵌入

2.1. 步骤 1：添加 SDK 到工程中

请在自己的工程导入libs文件夹下的SDK文件：

- libGDTMob.a
- GDTMobBannerView.h
- GDTMobInterstitial.h

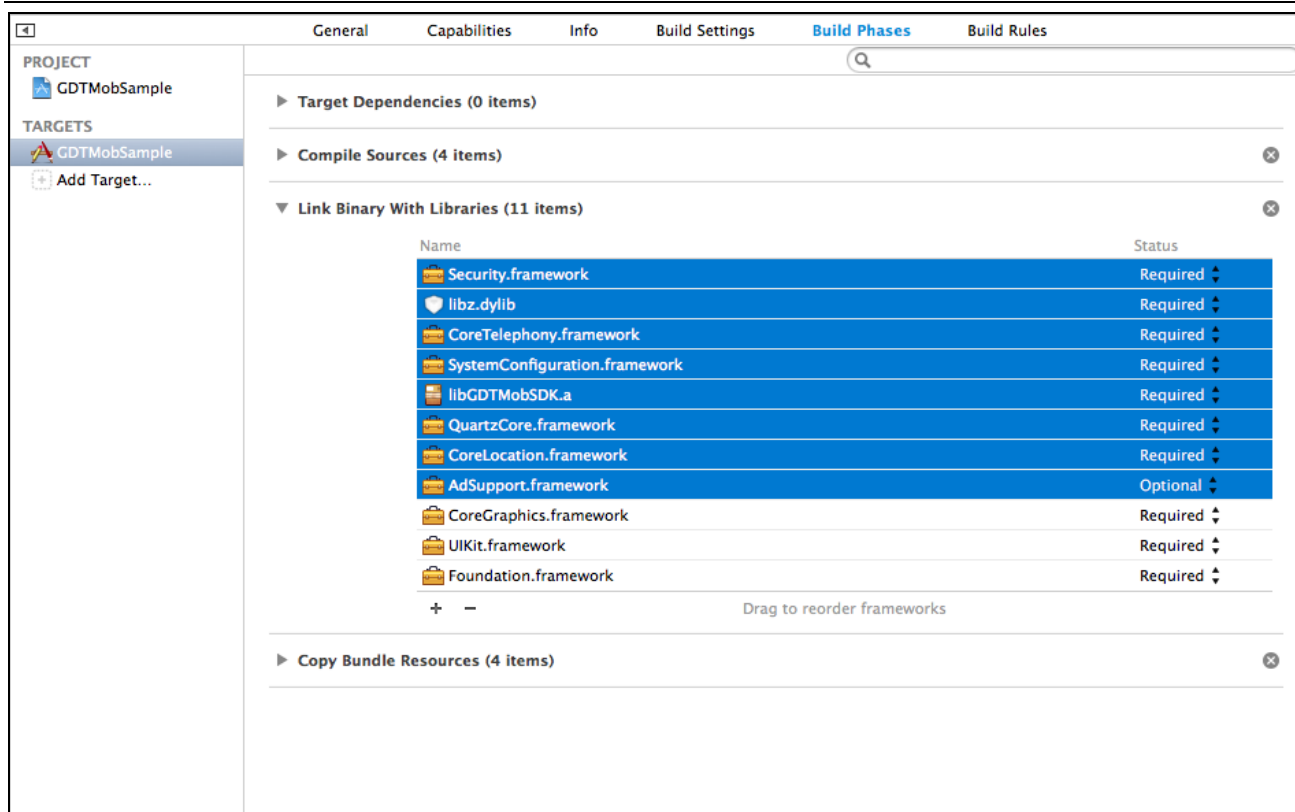




2.2. 步骤 2：为工程添加相应的 Frameworks

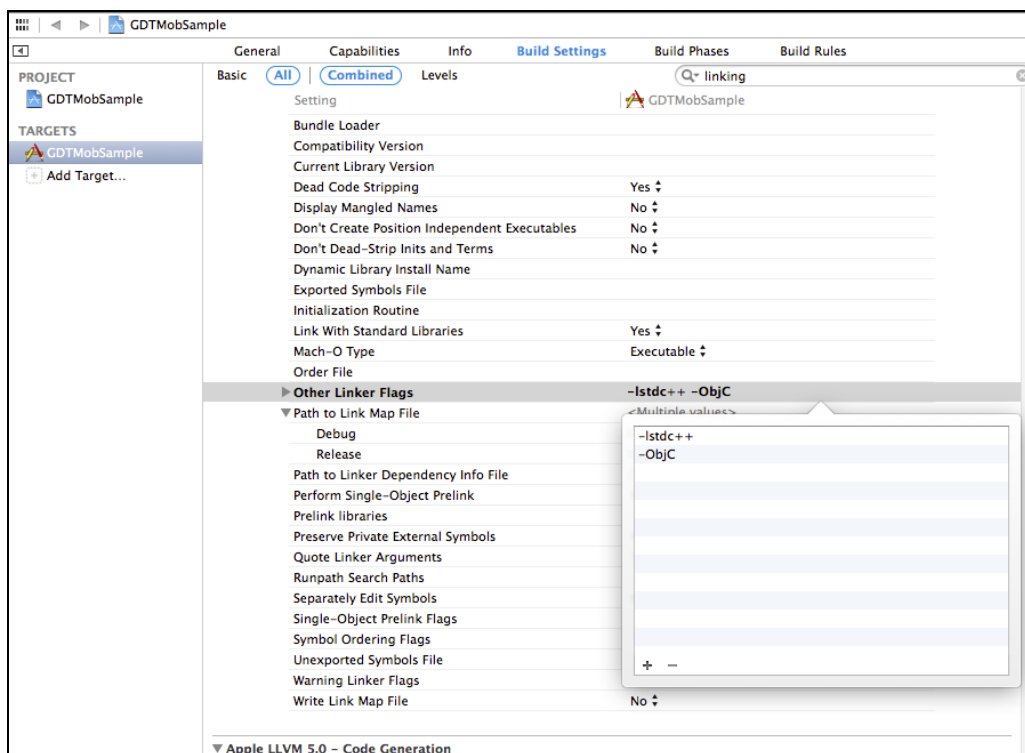
需要为项目添加如下 Frameworks

- AdSupport.framework 【设置为Optional】
- CoreLocation.framework
- QuartzCore.framework
- SystemConfiguration.framework
- CoreTelephony.framework
- libz.dylib
- Security.framework



2.3. 步骤 3：更改导入静态库设置

点击项目Target，选择Build Settings标签页，找到Linking下面的Other Linker Flags,分别添加两个参数：-ObjC, -lstdc++，如下图：



3. 接入代码

3.1. Banner 广告接入代码

（详细内容请参考压缩包中的 Sample 项目）

1. 在想要导入广告的 ViewController 头文件中导入头文件并声明实例：

```
#import <UIKit/UIKit.h>
#import "GDTMobBannerView.h" //导入GDTMobBannerView.h头文件

@interface BannerViewController : UIViewController<GDTMobBannerViewDelegate>
{
    GDTMobBannerView *_bannerView; //声明一个GDTMobBannerView的实例
}

@end
```

2. 在 ViewController 的实现文件中初始化并加载广告：

```
@implementation BannerViewController
```

```
- (id)initWithNibName:(NSString *)nibNameOrNil bundle:(NSBundle *)nibBundleOrNil
{
    self = [super initWithNibName:@"BannerView" bundle:nibBundleOrNil];
    if (self) {
        /*
         * 创建Banner广告View
         * "appkey" 指在 http://e.qq.com/dev/ 能看到的app唯一字符串
         * "placementId" 指在 http://e.qq.com/dev/ 生成的数字串，广告位id
         */
        _bannerView = [[GDTMobBannerView alloc] initWithFrame:CGRectMake(0, 0,
                                                                           GDTMOB_AD_SIZE_320x50.width,
                                                                           GDTMOB_AD_SIZE_320x50.height)
                       appkey:@"100720253"
                       placementId:@"2594074491383676351"];
    }
    return self;
}
```

```
- (void)viewDidLoad
{
    [super viewDidLoad];
    _bannerView.delegate = self; // 设置Delegate
    _bannerView.currentViewController = self; //设置当前的ViewController
    _bannerView.interval = 30; // 【可选】设置刷新频率;默认30秒
    _bannerView.isTestMode = NO; // 【可选】设置测试模式;默认NO
    _bannerView.isGpsOn = NO; // 【可选】开启GPS定位;默认关闭
    [self.view addSubview:_bannerView]; //添加到当前的view中
    [_bannerView loadAdAndShow]; //加载广告并展示
}

-(void)dealloc
{
    _bannerView.delegate = nil;
    _bannerView.currentViewController = nil;
    [_bannerView release];
    [super dealloc];
}
```

备注：

- 1) 在非 ARC 模式下，建议在 dealloc 方法中先将 delegate 和 currentViewController 至为空，然后再释放 bannerView。
- 2) 开启 GPS 定位可以获取地理位置信息，提高广告填充率，增加收益，但是如果与本身应用业务矛盾也有可能被苹果审核拒绝的风险。

3. 实现 GDTMobBannerDelegate 方法：【可选】

您可以通过实现 GDTMobBannerDelegate 中定义的方法，来跟踪广告生命周期中的各个阶段，这些方法定义如下：

```
// 请求广告条数据成功后调用
- (void)bannerViewDidLoad;
// 请求广告条数据失败后调用
- (void)bannerViewFailToReceived:(int)errCode;
// 全屏广告弹出时调用
- (void)bannerViewDidPresentScreen;
// 全屏广告关闭时调用
- (void)bannerViewDidDismissScreen;
// 应用进入后台时调用
- (void)bannerViewWillLeaveApplication;
```

3.2. 插屏广告接入代码

（详细内容请参考压缩包中的 Sample 项目）

1. 在想要导入广告的 ViewController 头文件中导入头文件并声明实例：

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

@interface InterstitialViewController : UIViewController<GDTMobInterstitialDelegate>
{
    GDTMobInterstitial *_interstitialObj;
}

@end
```

2. 在 ViewController 的实现文件中初始化并预加载广告，之后在合适的时机进行展现：

```
/**
 * 在适当的时候，初始化并调用loadAd方法进行预加载
 */
- (IBAction)loadAd:(id)sender {

    _interstitialObj = [[GDTMobInterstitial alloc] initWithAppkey:@"2211675583"
                                                                placementId:@"2954362461573316031"];

    _interstitialObj.delegate = self; //设置委托
    _interstitialObj.isGpsOn = NO;    //【可选】设置GPS开关

    //预加载广告
    [_interstitialObj loadAd];
}

/**
 * 在适当的时候，调用presentFromRootViewController来展现插屏广告
 */
- (IBAction)showAd:(id)sender {
    UIViewController *vc = [[[UIApplication sharedApplication] keyWindow]
                            rootViewController];

    [_interstitialObj presentFromRootViewController:vc];
}
```

备注：

- 1) 在一次展现完成后，需要再次调用 loadAd 方法来加载新的广告，用于下次展现
- 2) 关于释放，建议在释放 ViewController 之前将 _interstitialObj 的 delegate 属性设置为 nil

3. 实现 GDTMobInterstitialDelegate 方法：【可选】

您可以通过实现 GDTMobInterstitialDelegate 中定义的方法，来跟踪插屏广告生命周期中的各个阶段，这些方法定义如下：

```
// 广告预加载成功回调
- (void)interstitialSuccessToLoadAd:(GDTMobInterstitial *)interstitial;

// 广告预加载失败回调
- (void)interstitialFailToLoadAd:(GDTMobInterstitial *)interstitial;

// 插屏广告将要展示回调
- (void)interstitialWillPresentScreen:(GDTMobInterstitial *)interstitial;

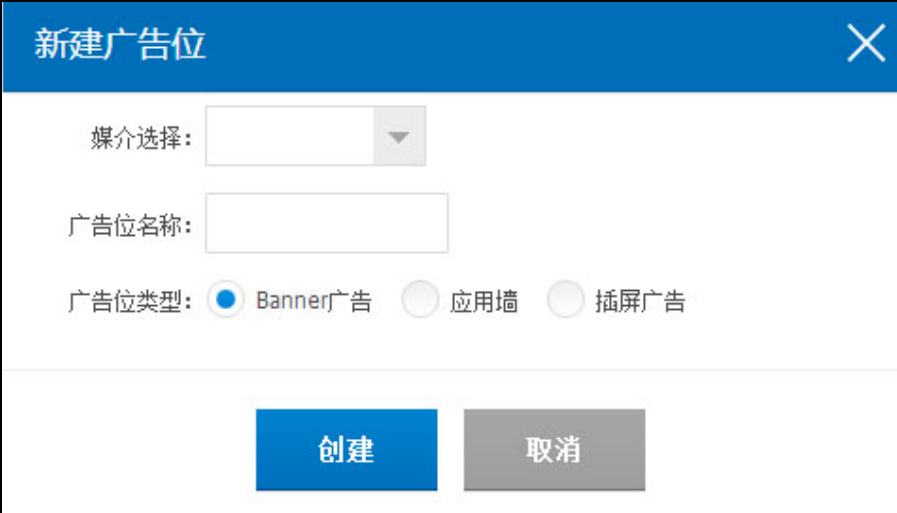
// 插屏广告视图展示成功回调
- (void)interstitialDidPresentScreen:(GDTMobInterstitial *)interstitial;

// 插屏广告展示结束回调
- (void)interstitialDidDismissScreen:(GDTMobInterstitial *)interstitial;

// 应用进入后台时回调
- (void)interstitialApplicationWillEnterBackground:(GDTMobInterstitial *)interstitial;
```

4. 广告位创建流程

在您的应用通过审核后，可以在相关页面选择“创建广告位”。
创建广告位界面如下图所示：



新建广告位

媒介选择:

广告位名称:

广告位类型: ☒ Banner广告 ☐ 应用墙 ☐ 插屏广告

步骤说明：

- 1.在创建广告位之前，确保您已经下载最新版 SDK 并阅读完说明文档，了解如何在您的应用中使用广告位 ID。
- 2.填写需要嵌入广告的应用名称。注意此处只可选择已通过审核的应用。
- 3.设置广告位的名称及格式。广告位名称用于区分应用内设置的各个广告位，方便您在后台进行统一管理和查看数据报表；目前提供选择的广告位格式为 Banner 广告。
- 4.点击创建按钮以后即获得对应该广告位的 ID 号。
- 5.成功得到广告位 ID 以后，您可以在应用代码内使用该 ID 进行广告联调测试（见“接入代码”部分）。
- 6.通过测试后去掉测试代码部分（isTestMode），即可发布带有联盟广告位的 IOS 应用。

5. SDK 相关问题排查

如果根据正常的注册流程仍然无法在嵌入广点通 sdk 的 app 中看到广告，可以尝试着看控制台输出。

广点通 sdk 默认设置下会对广告流程中的关键步骤打印出 log。

LogTag : **[GDTMob::GDTMob]**
Text : **code:**[返回码]. **msg :**[信息]

广点通在打印广告关键信息时会带上**[GDTMob::GDTMob]** 的标记,其他带有 GDTMob 标记的 log 与广告信息无关。

6. 注意事项

在以下情况下使用 Banner 广告请将 interval 刷新频率设为 0，即不自动刷新。

- a) 使用广告聚合平台控制刷新频率时。
- b) 使用代码定时新建 GDTMobBannerView 来控制刷新频率时。