



MOSDK iOS Unity集成说明

Prepared by: Amy Foster, Account Manager

March 1, 2015

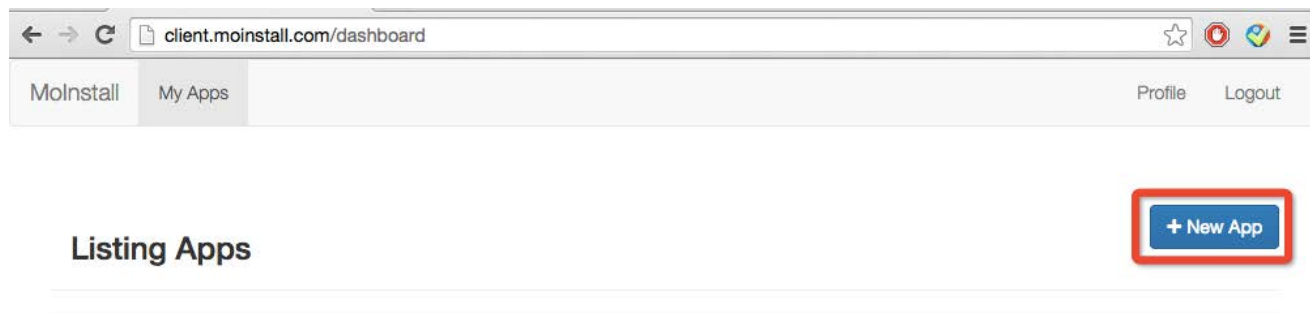
本framework目前支持armv7s, armv7, arm64, 以及i386 and x86_64 (模拟器调试)

Unity 版本需要在4.0以上

添加新游戏	2
下载SDK	3
集成SDK	3
导入MOSDK UNITY PLUGIN	3
设置MOSDK ID	4
在UNITY中调用广告显示	5
导出XCODE项目	5
在XCODE中导入MOSDK	5
模拟器 (Simulator) 特定修改	7

添加新游戏

- 成功登陆系统后，点击**My Apps**，再点击**+New App**



- 点击New App后会出现信息对话框。按照真实情况填写即可。对于iOS的App来说Store ID就是Apple ID。Orientation一项，Portrait是指竖屏，Landscape是指横屏。如果App两种都可以，则请选择Portrait。

New app

1

Platform	Apple
Store ID	12345678
Name	My Mobile Game
Orientation	Portrait

2

Create App

- 点击**Create App**后，系统会出现新游戏的信息。其中**ID**就是在集成MOSDK时必须的信息。新建的App会在右上角显示**Pending**字样。这是因为系统还在对新创的游戏进行审核。但这不影响SDK的集成

My Mobile Game

Back

Edit Reports Remove

Pending

ID: 7a1d-550267103238610007000000

Orientation: Portrait

Store ID: 12345678 Test Mode

Created at: March 13th, 2015 04:26

下载SDK

- 登陆 <http://client.moinstall.com>
- 登陆后在界面的顶端有SDK和说明文档的下载链接



Listing Apps

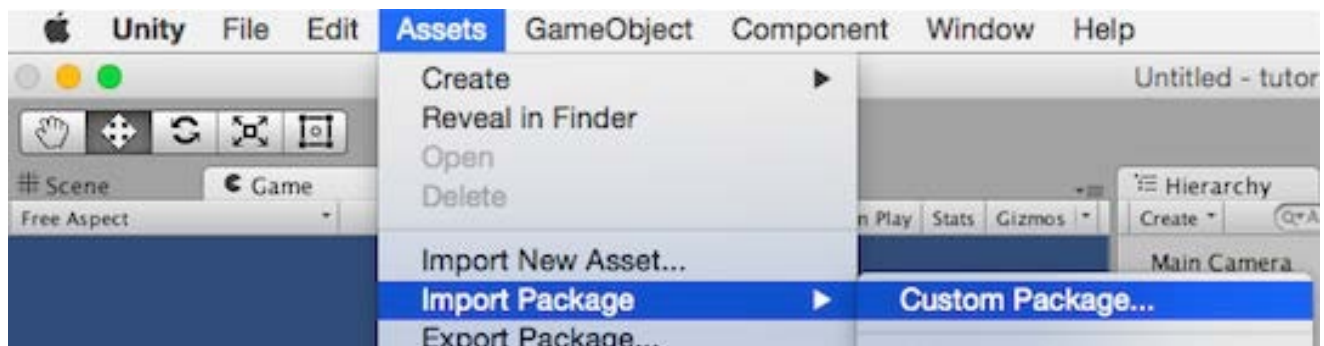


集成SDK

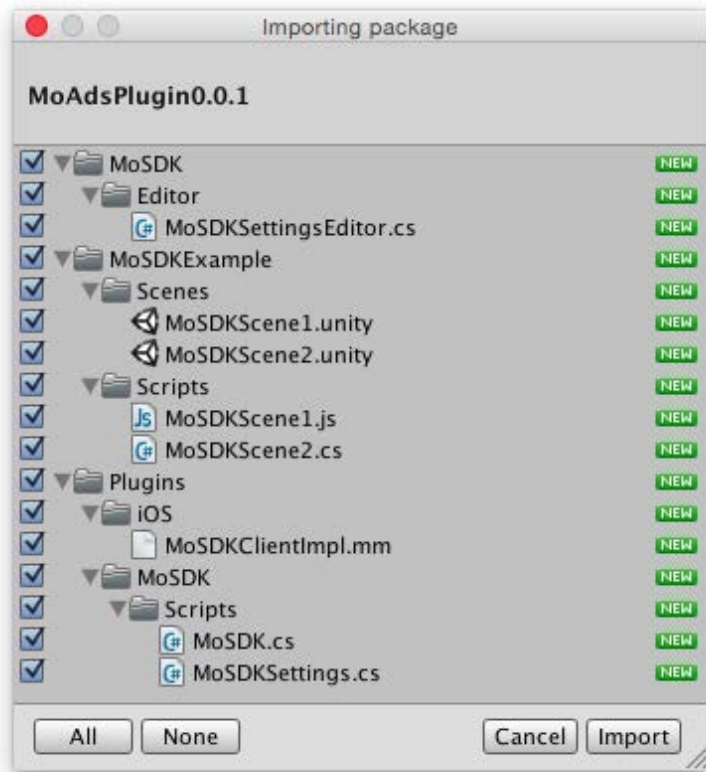
- 解压缩下载后的SDK文件moinstall-unity-sdk-master.zip

导入MOSDK Unity Plugin

- 在Unity中打开你的项目
- 选择菜单**Assets->Import Package->Custom Package**
- 导入刚才解压缩文件中的**MoAdsPlugin0.0.1.unitypackage**

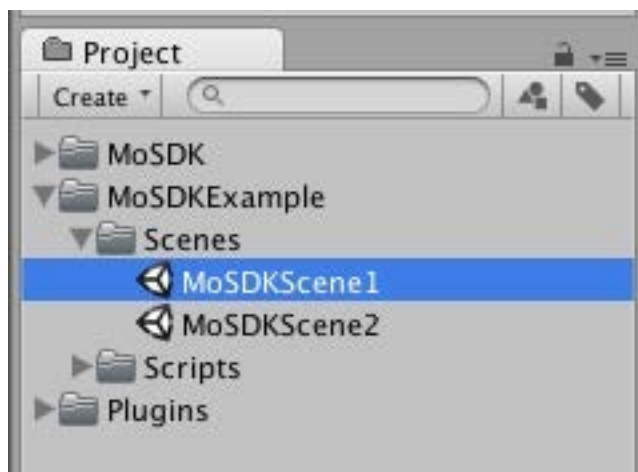


- 在导入对话框中确保所有文件都选中了，点击**Import**导入



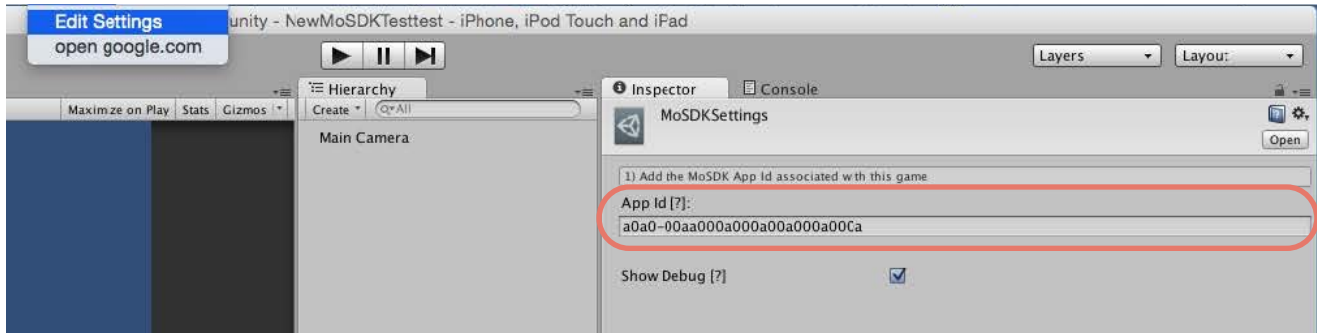
设置MOSDK ID

- 在项目视图中选中相应的场景节点



- 选中菜单MoSDK->Edit Settings

- 在inspector的窗口中填入先前在网站上创建的App的真实ID



在Unity中调用广告显示

因为MOSDK Unity Plugin中已经包含了相应静态函数的定义。在需要显示广告的地方只要插入ShowNextAd()函数就可以了。

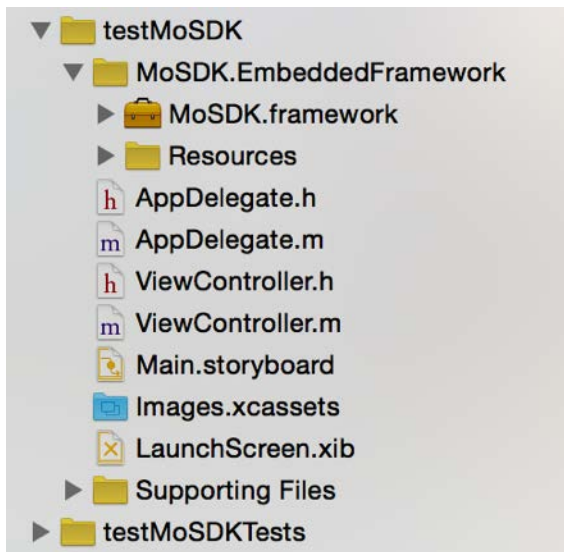
注意在Unity的模拟器中，MOSDK的函数是不会产生展示广告的功能的。要测试必须先导出xCode的项目，通过设备或者iOS模拟器来测试广告。

导出xCode项目

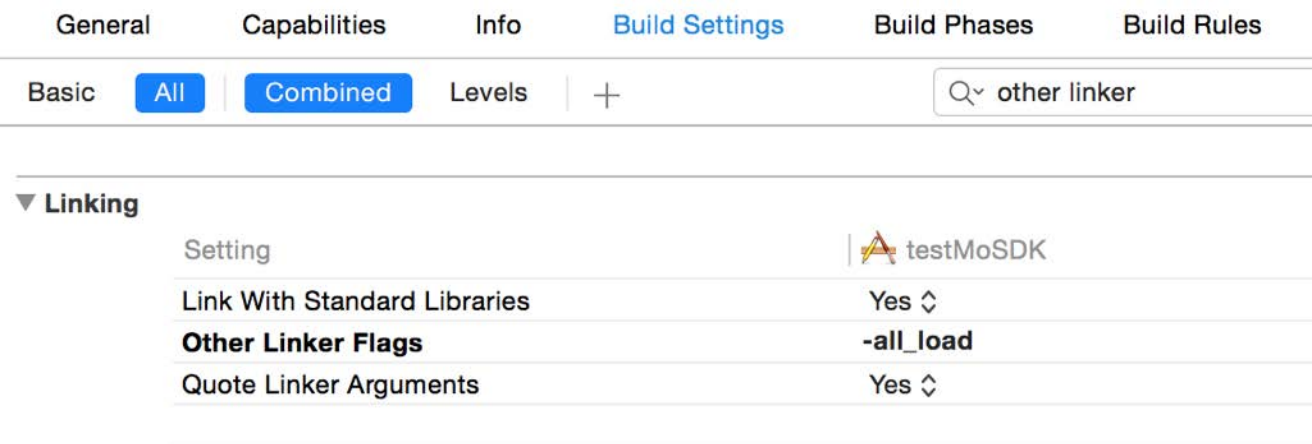
- 首先在File->Build Setting中将Platform切换到iOS
- 按照App的真实情况在Player Setting中做相应设置。包括是真机项目还是模拟器项目
- 点选相应的场景后导出xCode项目

在xCode中导入MOSDK

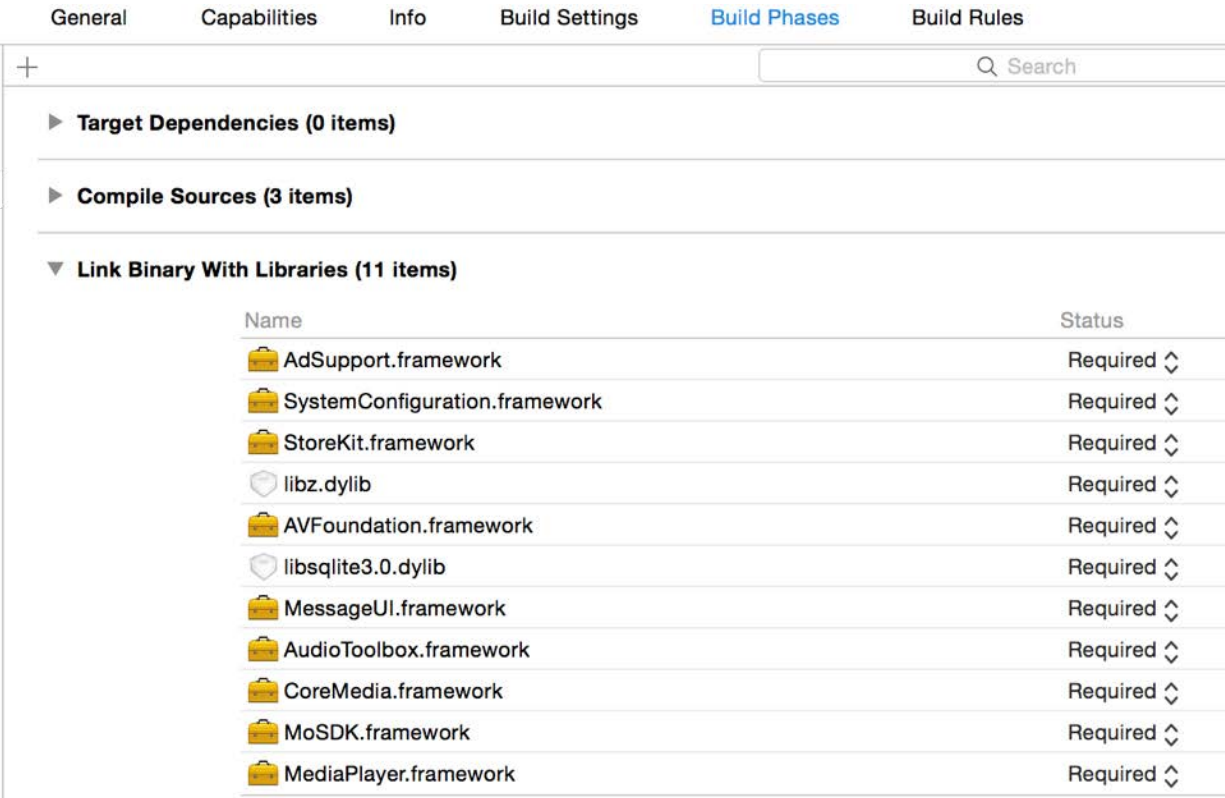
- 打开解压缩后的文件夹，把目录**MoSDK.EmbeddedFramework**拖拽加入到xcode开发项目中



- 进入项目目的Build Settings，设置Other Linker Flags 为“-all_load”。如果是在模拟器上运行则使用“-ObjC”标志



- 进入项目目的Build Phases 里里的Link Binary With Libraries，依次添加如下所需的系统框架(如已存在，无无须重复添加):



- 编译执行即可

模拟器 (SIMULATOR) 特定修改

- 在源码中找到RegisterMonoModules.cpp文件
- 锁定其中包含mono_dl_register_symbol的那行代码

```
5 extern "C"
6 {
7     typedef void* gpointer;
8     typedef int gboolean;
9     #if !(TARGET_IPHONE_SIMULATOR)
10     const char*      UnityIPhoneRuntimeVersion = "4.6.0f3";
11     void             mono_dl_register_symbol (const char* name, void *addr);
12     extern int       mono_ficall_flag;
13     void             mono_aot_register_module(gpointer *aot_info);
```

- 把那行代码移动到#if !(TARGET_IPHONE_SIMULATOR)之前 (如下图)

```
5 extern "C"
6 {
7     typedef void* gpointer;
8     typedef int gboolean;
9     void             mono_dl_register_symbol (const char* name, void *addr);
10    #if !(TARGET_IPHONE_SIMULATOR)
11     const char*      UnityIPhoneRuntimeVersion = "4.6.0f3";
```

- 找到包含_StartWithAppId 和_ShowNextAd

```
47 mono_aot_register_module(mono_aot_module_UnityEngine_UI_info);
48 mono_aot_register_module(mono_aot_module_UnityEngine_info);
49 mono_aot_register_module(mono_aot_module_mscorlib_info);
50
51 mono_dl_register_symbol("_StartWithAppId", (void*)&_StartWithAppId);
52 mono_dl_register_symbol("_ShowNextAd", (void*)&_ShowNextAd);
53 mono_dl_register_symbol("UnityNSObject_RetainObject", (void*)&UnityNSObject_RetainObject);
54 mono_dl_register_symbol("UnityNSObject_ReleaseObject", (void*)&UnityNSObject_ReleaseObject);
55 mono_dl_register_symbol("UnityNSError_Code", (void*)&UnityNSError_Code);
56 mono_dl_register_symbol("UnityNSError_Description", (void*)&UnityNSError_Description);
57 mono_dl_register_symbol("UnityNSError_Reason", (void*)&UnityNSError_Reason);
```

- 把那行代码移动到 #if !(TARGET_IPHONE_SIMULATOR)之前 (如下图)

```
void RegisterMonoModules()
{
    gEnableGyroscope = false;
    mono_dl_register_symbol("_StartWithAppId", (void*)&_StartWithAppId);
    mono_dl_register_symbol("_ShowNextAd", (void*)&_ShowNextAd);
    #if !(TARGET_IPHONE_SIMULATOR)
        mono_aot_only = true;
        mono_ficall_flag = false;
```

- 保存代码后运行程序