

# AppStore专用框架替换说明

由于线上2.2.5版本的框架可能导致AppStore审核不过，我们准备了一套专门用于上架AppStore用的替换框架，和一些涉及到的修改版SDK插件，用于AppStore过审。该版本框架和对应插件仅用于AppStore渠道上架，其他渠道还是继续使用之前线上的2.2.5版本框架和线上SDK。

目前认为IAP和PAY相关字段可能导致审核不通过，游戏也尽量少用这些字段  
替换步骤：

## 1. 下载专用框架，企业版暂时先到对接群下载

通用版框架：

C++: <https://pan.baidu.com/s/1ge8vE4V> 密码: r5hc

Unity: <https://pan.baidu.com/s/1dESl2mX> 密码: vbpd

JS: <https://pan.baidu.com/s/1eSrM378> 密码: ymw3 Creator也使用JS框架

Lua: <https://pan.baidu.com/s/1i4K2PY9> 密码: hne9

## 2. 替换专用版插件

AppStore:更新线上最新版本

微信推出了一个不包含支付功能的版本，意味着带支付功能的SDK也许有不过审的可能，这边也临时做了个不带支付的微信插件，有需要可以导入。如果要打越狱框架+渠道包，可以用下图右方按钮回复线上版本插件。

微信带支付: <https://pan.baidu.com/s/1cD7Jqu> 密码: md2r

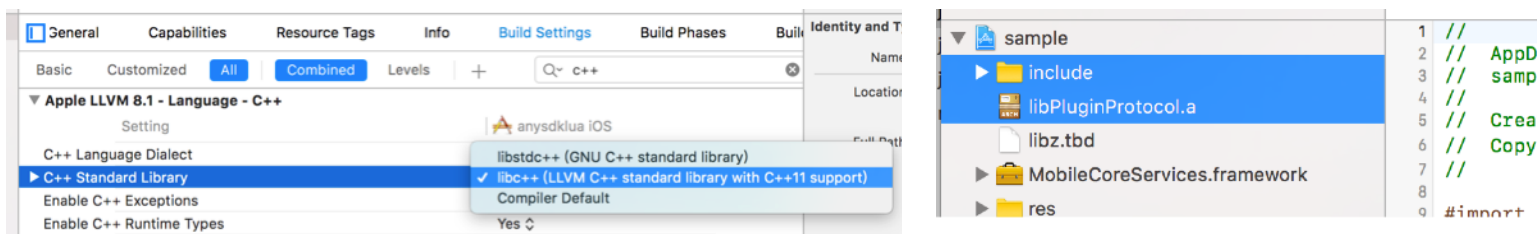
微信不带支付: <https://pan.baidu.com/s/1slExi0H> 密码: s4vq

替换插件后除Unity和Creator外，此时可以重新打包，打包前记得参考集成文档，删掉多余Target。



## 3. 替换打包后工程里的框架文件

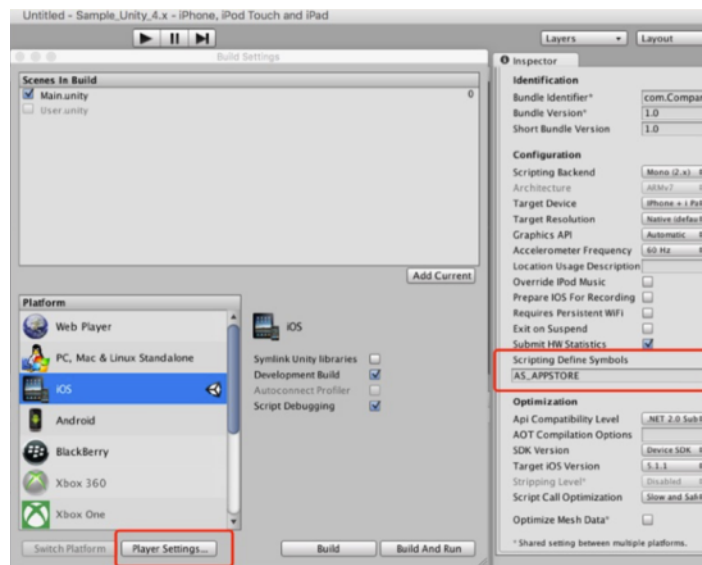
C++: 可在打包后的工程中，删除原先对应版本的include文件夹和libPluginProtocol.a，参考Xcode工程左图配置，找到对应文件夹里，替换版的include文件夹和libPluginProtocol.a分别拖入。不要直接拖入protocols文件夹，可能会找不到头文件的search path。



Unity: 在Unity工程中替换对应版本框架插件, 并且在Player Setting->ios配置信息中的Scripting Define Symbols中添加“AS\_APPSTORE”, 代码中添加:

```
#if AS_APPSTORE
using AnySDKIAP = anysdk.AnySDKYAP; //企业版
#endif
```

```
--- C#
#if AS_APPSTORE
using AnySDKIAP = anysdk.AnySDKYAP;
#endif
---
```



JS/Lua: 参考接入指南, 在打包后工程分别替换include文件夹, libPluginProtocol.a和绑定文件。注意点同C++框架。

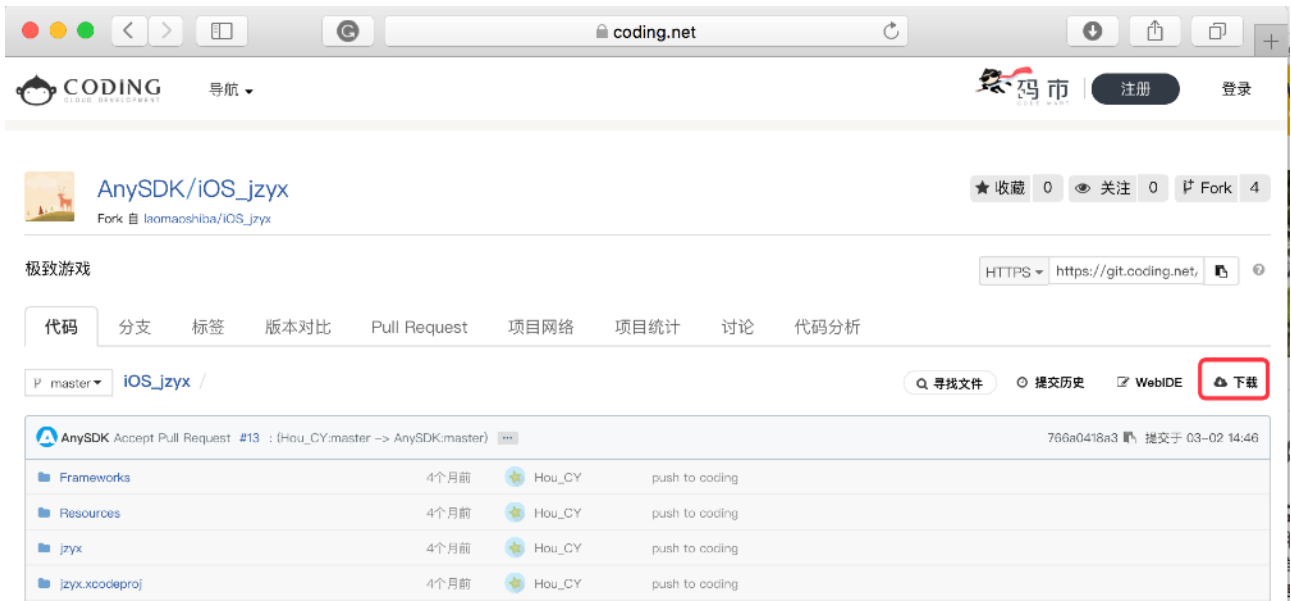
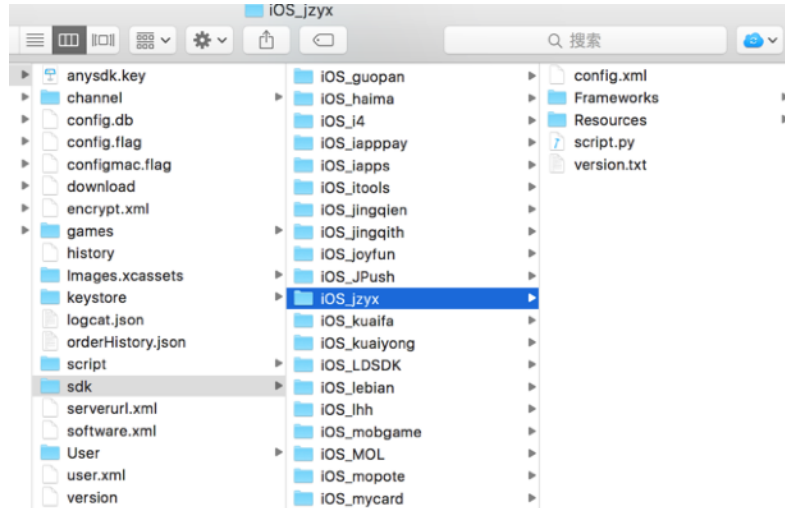
Creator:

1. 在打包后工程文件夹, 找到build/jsb-default/src/, 删掉jsb\_anysdk\_constants.jsc和jsb\_anysdk.jsc, 替换成框架目录下/src的jsb\_anysdk\_constants.js和jsb\_anysdk.js
2. 工程文件夹下/build/jsb-default/frameworks/cocos2d-x/external/ios/libs 替换libPluginProtocol.a, 注意点同C++框架
3. 找到CocosCreator安装目录/build/jsb-default/frameworks/cocos2d-x/external/ios/include/anysdk, 删掉原先的头文件, 拷贝替换头文件, 注意点同C++框架
4. 找到CocosCreator安装目录/build/jsb-default/frameworks/cocos2d-x/extensions/anysdk/js-bindings/ 替换绑定文件。1.5及之前版本, 用框架目录中3.11以上绑定文件, 1.6版本使用Creator1.6目录的绑定文件

## 4.如何自行修改插件

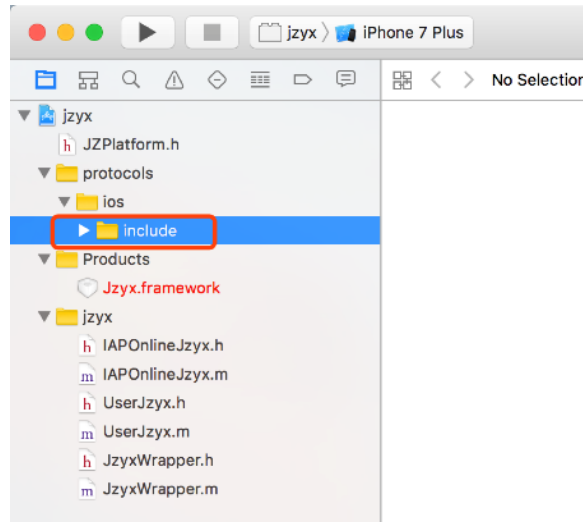
如果有涉及到以下情况：SDK插件涉及支付，而又需要用这个插件上线AppStore，可按以下方法自己修改

1. 获取插件工程，可使用自助开发功能获取。如果不使用自助开发功能，也可以直接在coding上下载到对应插件。先在/文稿/AnySDKFiles/config/sdk/目录下寻找对应的插件名称，例如iOS\_jzyx。

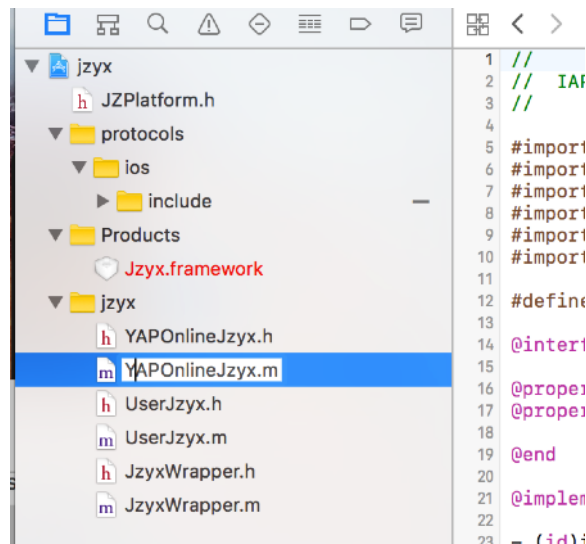


在 <http://coding.net/u/AnySDK/p/>路径后加上 iOS\_jzyx = [https://coding.net/u/AnySDK/p/iOS\\_jzyx](https://coding.net/u/AnySDK/p/iOS_jzyx) 访问coding页面，点击下载

2. 打开工程，替换工程中的include头文件，先删掉include文件夹链接，将替换版框架头文件include文件夹拖进工程。可下载C++框架使用里面的头文件。替换后的include文件夹里应该有InterfaceYAP.h, ProtocolYAP.h和YAPWrapper.h三个替换名称后的头文件。

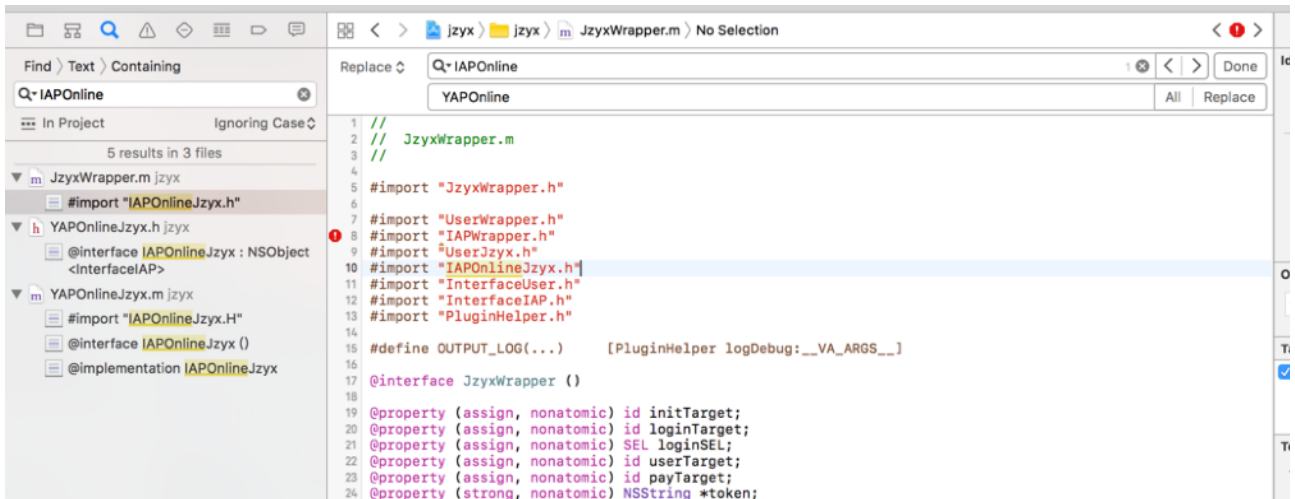


3. 修改文件名：IAPOnlineXXXX.h->YAPOnlineXXXX.h, IAPOnlineXXXX.m->YAPOnlineXXXX.m。直接在工程的文件名上点击修改。

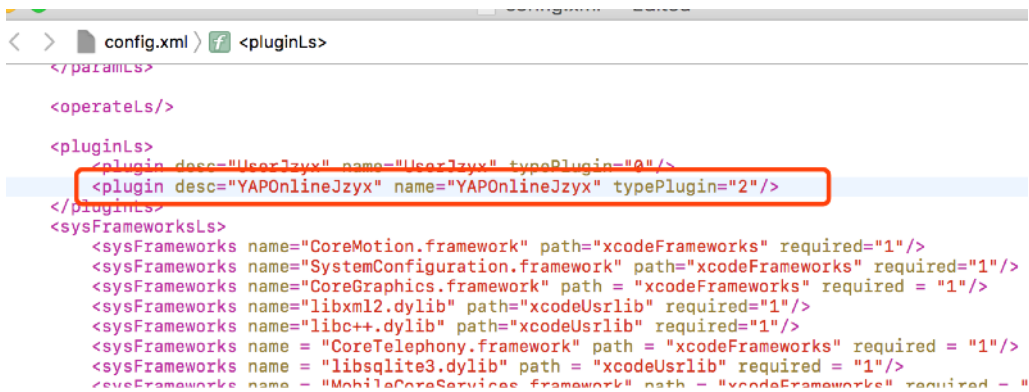


4. 全工程搜索替换字段：IAPOnline->YAPOnline, ProtocolIAP->ProtocolYAP, IAPWrapper->YAPWrapper, InterfaceIAP->InterfaceYAP。可用command+shift+F组合键全工程搜索。

5. 全工程搜索替换字段：PAYRESULT\_PAYEXTENSION->YAPRESULT\_YAPEXTENSION, PAYRESULT\_->YAPRESULT\_, getOrderId->getYapOrderId, onPayResult->onYapResult, payForProduct->yapForProduct, paylnSDK->yaplnSDK, payTarget->yapTarget。可用command+shift+F组合键全工程搜索。



6. 修改插件目录下config.xml，将里面的IAPOnline字段修改成YAPOnline



7. 如果使用自助开发，直接编译插件。如果是下载下来的工程，可以选择Generic iOS Device，编译插件工程，之后找到编译的.framework文件，替换原先插件目录下的/Framework/的同名工程文件。另外也需要替换修改后的config.xml文件。之后便可以尝试重新打包。

