

NewMasterSdk v1.12








2017年12月12日

NewMasterSdk - 技术文档

接口说明、接入指南及常见问题

接入前注意事项:

此SDK已集成以下静态库:

| |
|---|
|  libcocos2d IOS.a |
|  libluacocos2d IOS.a |
|  libquick_libs IOS.a |
|  libsimulator IOS.a |
|  jcore-ios-1.1.6.a |
|  jpush-ios-3.0.7.a |
|  libDeviceUDID.a |

并且已知与以下框架或第三方库有冲突, 所以再接入前请确保您的工程未包含这些框架或第三方库:

Cocos2dx/Cocos2d

Jpush

Unity

React-native

关于版本号:

版本号要求使用两位小数的格式且要求使用1.10作为初始版本号。

| | |
|-------------------|---------------------|
| Display Name | TestProject |
| Bundle Identifier | com.company.project |
| Version | 1.10 |
| Build | 1 |

一、接口说明

1.本库包含两个文件

* NewMasterSdk.h 公开接口头文件

* libNewMasterSdk.a 静态库连接文件

2.本库的函数接口及说明

```
#define kJPushKey        @"jpush_key"  
#define kJPushChanel    @"jpush_channel"  
#define kCheckUrl        @"check_url"  
#define kIsDebugMode    @"isDebugMode"
```

```
void new_master_initialize(Class appDelegateClass, NSDictionary *info);
```

参数:

appDelegateClass: 应用程序代理类的Class对象, 一般传入[AppDelegate class].

info: 初始化参数, 为字典类型, 详细键值释义如下:

kJPushKey: 值为NSString*, 为激光推送(JPush)的key的字符串;

kJPushChanel: 值为NSString*, 为激光推送(JPush)的channel的字符串

kCheckUrl: 值为NSString*或NSArray<NSString*>*, 一个或多个检查过审信息及下载主包的**域名或IP地址**(例: abc.test1.com:8888). **注意: 为了确保能正常访问, 请务必配置多个域名或IP地址。**

kIsDebugMode: (**可选字段**) 值为BOOL, 填写**YES**为是否设置调试模式, 调试模式会输出Log, 不加这个字段或设置 **NO** 不显示输出。

说明:

本函数必须在所有代码执行前执行, 一般放在main.m或main.mm中main入口函数的**UIApplicationMain**调用之前, 如果是swift项目, 请删除**AppDelegate**的**@UIApplicationMain**注解, 并手动创建main.m或main.swift.

3.初始化示例代码

main.m 文件:

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

#import "NewMasterSdk.h"

int main(int argc, char * argv[]) {
    @autoreleasepool {
        NSDictionary *info =
            @{
                kJPushKey: @"4x99xxxx64c2b701b3x2xx9e",
                kJPushChanel: @"cp58",
                kCheckUrl: @[
                    @"test1.example.com:8888",
                    @"test1.example.com:8888",
                    @"test1.example.com:8888",
                ],
                kIsDebugMode:@YES,
            };

        // 这个初始化必须在所有初始化代码之前调用
        new_master_initialize([AppDelegate class], info);

        return UIApplicationMain(argc, argv, nil,
            NSStringFromClass([AppDelegate class]));
    }
}
```

二、接入指南

1.Object-C 项目

* 引入静态库文件和头文件, 在Xcode中选择“Add files to 'Your project name'...”, 将NewMasterSdk.h和libNewMasterSdk.a文件添加到你的工程目录中.

* 参考上一节中的示例代码, 修改main.m或main.mm文件.

* 添加Framework:

Foundation.framework

UIKit.framework

AdSupport.framework

AudioToolbox.framework

AVFoundation.framework

CFNetwork.framework

CoreAudio.framework

CoreFoundation.framework

CoreGraphics.framework

CoreImage.framework

CoreMedia.framework

CoreMotion.framework

CoreTelephony.framework

CoreText.framework

CoreVideo.framework

GameController.framework

OpenAL.framework

OpenGLESTexture.framework

QuartzCore.framework

MediaPlayer.framework

SystemConfiguration.framework

UserNotifications.framework -> optional

libc++.tbd

libiconv.tbd

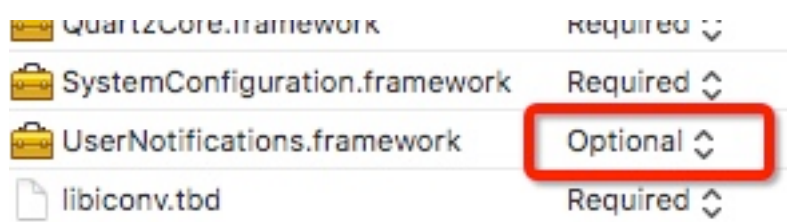
NEWMASTERSDK - 技术文档

libz.tbd

libresolv.tbd

注意:UserNotifications.framework是可选的, 否则iOS 10.0以下的系统运行

APP会崩溃!! 应该如下图设置:

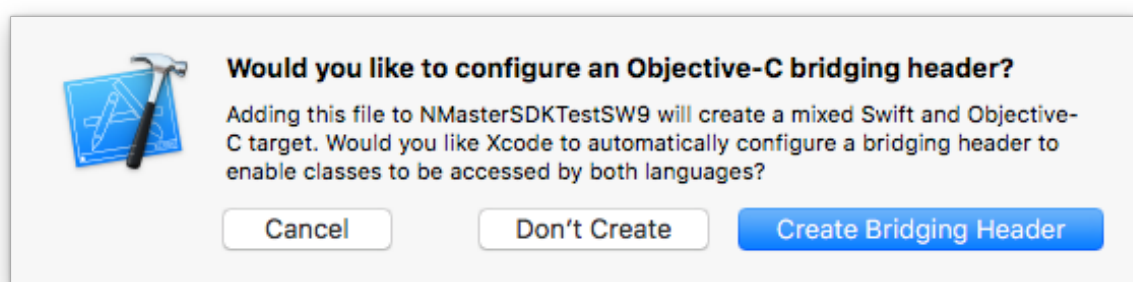


2.Swift 项目

* 移除AppDelegate的@UIApplicationMain注解

```
13  
14 //@UIApplicationMain  
15 class AppDelegate: UIResponder, UIApplicationDelegate {  
16
```

* 参考上一节中的示例代码, 新增main.m或main.mm文件, 注意: 在新增main.m文件时xcode会弹出如下提示, 这里选择点击”Create Bridging Header”, 点击后会在你的工程目录下会生成一个”你的工程名称-Bridging-Header.h” (例: 我的工程名为MySwiftProject, 则生成的文件名为MySwiftProject-Bridging-Header.h)



* 更改main.m文件中的

#import “AppDelegate.h” 为 #import “你的工程名称-Swift.h”

（例：我的工程名为MySwiftProject，则需要修改为MySwiftProject-Swift.h）或
直接使用Build Settings 中下列设置中的文件名称。

| ▼ Swift Compiler - General | |
|---|---|
| Setting | NMasterSDKTestSW8 |
| Install Objective-C Compatibility Header | Yes ↕ |
| Objective-C Bridging Header | NMasterSDKTestSW8/NMasterSDKTestSW8-Bridging-Header.h |
| ▶ Objective-C Generated Interface Header Name | NMasterSDKTestSW8-Swift.h |
| Precompile Bridging Header | Yes ↕ |
| Reflection Metadata Level | All ↕ |

* 另外如同Object-C项目一样需要引入静态库文件和头文件, 以及添加Framework

3. 工程设置

* 请确保General此部分设置与下图一致

▼ Deployment Info

Deployment Target 9.0

Devices Universal

Main Interface Main

Device Orientation ☒ Portrait
☐ Upside Down
☒ Landscape Left
☒ Landscape Right

Status Bar Style Default

☐ Hide status bar
☒ Requires full screen

* Capabilities 确保打开Push Notifications 和 Background Modes，并勾选
Background Modes 下Remote notifications 复选框

▼ Push Notifications ON

Steps: ☒ Add the Push Notifications feature to your App ID.
☒ Add the Push Notifications entitlement to your entitlements file

▼ Background Modes ON

Modes: ☐ Audio, AirPlay, and Picture in Picture
☐ Location updates
☐ Voice over IP
☐ Newsstand downloads
☐ External accessory communication
☐ Uses Bluetooth LE accessories
☐ Acts as a Bluetooth LE accessory
☐ Background fetch
☒ Remote notifications

* 在info.plist文件中添加下列内容

```
<key>UIRequiredDeviceCapabilities</key>
<array>
    <string>armv7</string>
</array>
<key>NSCalendarsUsageDescription</key>
<string>我们获取日历信息以便提供服务</string>
<key>NSCameraUsageDescription</key>
<string>我们需要打开相机以便提供服务</string>
<key>NSLocationUsageDescription</key>
<string>我们需要获取位置信息以便提供服务</string>
<key>NSLocationWhenInUseUsageDescription</key>
<string>我们需要获取位置信息以便提供服务</string>
<key>NSMicrophoneUsageDescription</key>
<string>我们打开麦克风以便提供服务</string>
<key>NSPhotoLibraryAddUsageDescription</key>
<string>我们需要写入相册信息以便提供服务</string>
<key>NSPhotoLibraryUsageDescription</key>
<string>i wanna use you photo library</string>
<key>LSRequiresIPhoneOS</key>
<true/>
<key>NSAppTransportSecurity</key>
<dict>
    <key>NSAllowsArbitraryLoads</key>
    <true/>
</dict>
```

三、常见问题

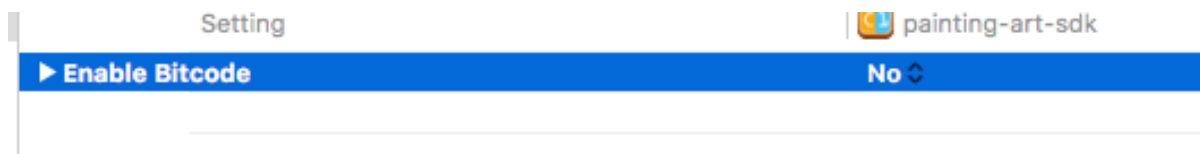
1.bitcode报错

错误信息:

```
'xxx' does not contain bitcode. You must rebuild it with bitcode enabled (Xcode
setting ENABLE_BITCODE), obtain an updated library from the vendor, or disable
bitcode for this target. for architecture armv7
```

解决方案:

Build Settings -> Enable Bitcode -> No

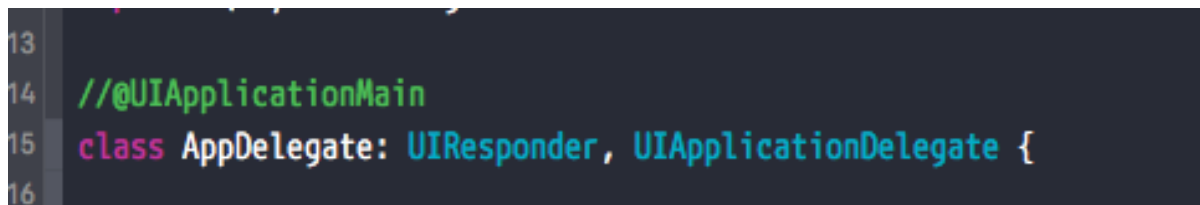


2. duplicate symbol _main

```
duplicate symbol _main in:  
  .../main.o  
  .../AppDelegate.o  
ld: 1 duplicate symbol for architecture arm64
```

错误信息:

解决方案:



移除AppDelegate的@UIApplicationMain注解

3. 启动黑屏

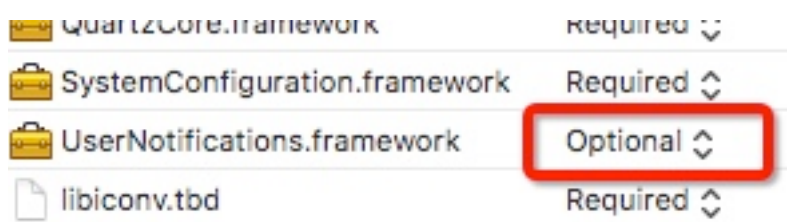
错误信息:

```
dyld: Library not loaded: /System/Library/Frameworks/  
UserNotifications.framework/UserNotifications  
Referenced from: .../xxxxx.app/xxxxx  
Reason: image not found
```

启动时黑屏, 并打印:

解决方案:

设置UserNotifications.framework为Optional

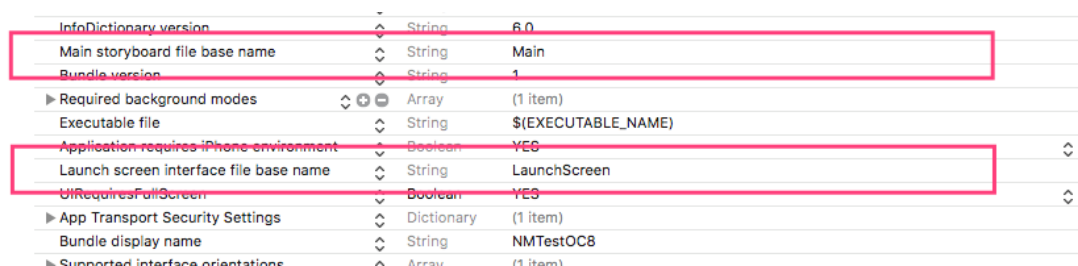


4. 崩溃并打印以下信息

Assertion failure in +[UIStoryboard storyboardWithName:bundle:], /BuildRoot/Library/Caches/com.apple.xbs/Sources/UIKit/UIKit-3600.6.22/UIStoryboard.m:88
2017-12-13 15:52:20.540146 RechargeScanner[712:158813] *** Terminating app due to uncaught exception 'NSInternalInconsistencyException', reason: 'Invalid parameter not satisfying: [name length] > 0'

解决方案:

检查info.plist中的 Main storyboard file base name 和 Launch screen interface file base name 是否为空, 如果未空请删除掉这两项。



5. 下载解压完成后黑屏

这是因为服务器需要需要几分钟来更新缓存, 稍等几分钟后重启应用即可正常显示。