



资讯

论坛

代码

工具

招聘

CVP



登录 | 注册

iOS开发 Swift App Store研究 产品设计 应用评测 游戏开发 苹果相关 WebApp 安卓相关 营销推广 业界动态 程序人生

Discover your IP Resource

Get IP news, blogs, white papers, videos & more all in one place now.



首页 > iOS开发

详述iOS国际化

2015-11-20 10:55 编辑: suiling 分类: iOS开发 来源: kk投稿

3 1660

iOS国际化 NSLocalizedString

招聘信息: cocos2dx手游客户端主程



本文是投稿文章, 作者: 龚凯

原文: [iOS国际化](#)

在真正将国际化实践前, 只知道通过NSLocalizedString方法将相应语言的字符串加载进来即可。但最近公司项目的新需求增加英文版本, 并支持应用内无死角切换~, 这才跳过各种坑实现了应用内切换语言, 并记录至此。

环境

系统环境: iOS7 - iOS9

开发环境: Swift2 & Xcode7

DEMO: [LocalDemo](#)

热门资讯

- 这些 iOS 面试基础题目, 你都深入了解吗?

点击量 17016
- 【译】17个提升iOS开发效率的必用工具

点击量 11064
- iOS开发——UI组件 (个人整理)

点击量 8769
- Objective-C 编码建议

点击量 6135
- 我已经写了48年代码了, 我感觉我还能写下

点击量 5906
- ViewController 瘦身另一种解决方案

点击量 5543
- 出大事了! A商要联合程序员们打淘宝分京东

点击量 5386
- 新手向: 五分钟搭建App设置页面_纯代码

点击量 5285
- 让我们来搞崩 Cocoa吧 (黑暗代码)

点击量 4647
- 来就来全套! 敏捷开发知识体系笔记

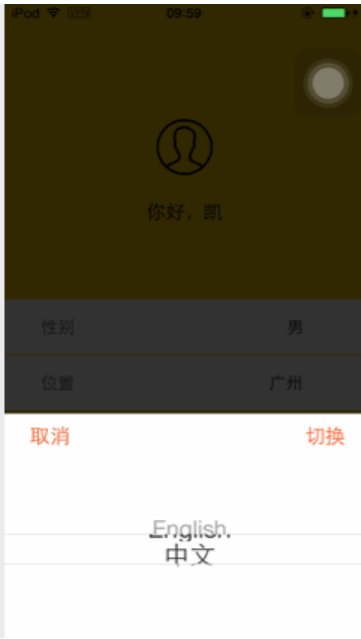
点击量 4581

综合评论

- 小码哥毕业的?

ai罗 评论了 源码推荐(11.23B): 类似今日头条/网易新闻的...
- 离线缓存的逻辑有点问题。应该先判断当前网络状态, 无网络或者服务端返回genning 评论了 iOS 数据库离线缓存思路和网络层封装...
- mark

whangchaojie 评论了 必看: 游戏开发者必知的21个免费学习资源...



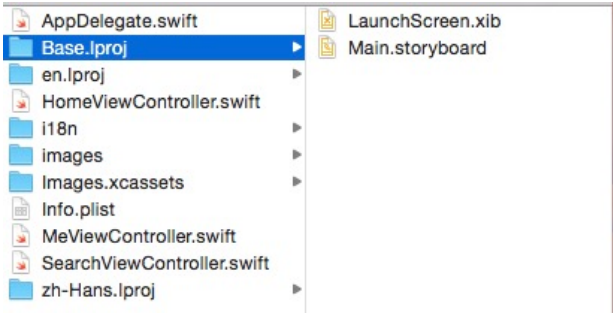
这个Demo的功能主要是切换语言后相应的界面文字&图片以及搜索引擎都会随语言变化。我们会围绕这个DEMO进行讲解，读者可以先下载这个Demo运行看下效果再往下

iOS国际化原理分析

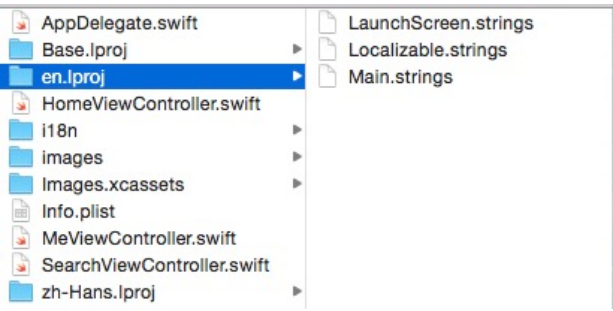
国际化其实都大同小异，其核心思想就是为每种语言单独定义一份资源。

iOS就是通过xxx.lproj目录来定义每个语言的资源，这里的资源可以是图片，文本，Storyboard，Xib等。我们可以看看LocalDemo源代码的物理目录结构

Base，暂时无需理会



English



mark 一下，你链接了文章基本上都看了
Aichiko 评论了 iOS开发——UI组件 (个人整理) ...

游戏可以的
苏小染 评论了 地狱边境：从一张草图到收入破2000万美元...

@property (nonatomic, readonly) Person *(^eat)(); @property halinuya_wjs 评论了 使用OC链式调用方式简化SpriteKit的动画调...

数据还有顺序的，而且更新服务器数据也是个问题，建议数据库只存储第一页
longdawei1988 评论了 iOS 数据库离线缓存思路和网络层封装...

为什么要用数据库做缓存
redoca 评论了 iOS 数据库离线缓存思路和网络层封装...

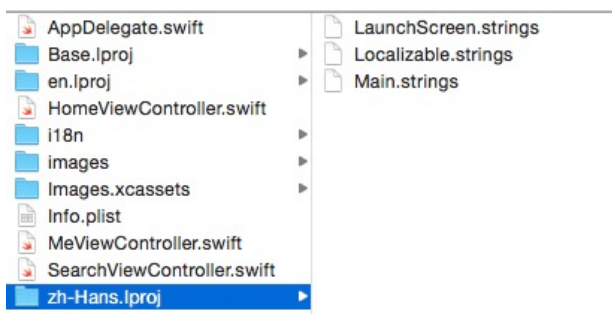
mark
liuyongchun 评论了 UIImagePickerController从拍...

网络请求前就向数据库读取数据，数据比网络上的旧怎么办？
zhangslnote 评论了 iOS 数据库离线缓存思路和网络层封装...

相关帖子

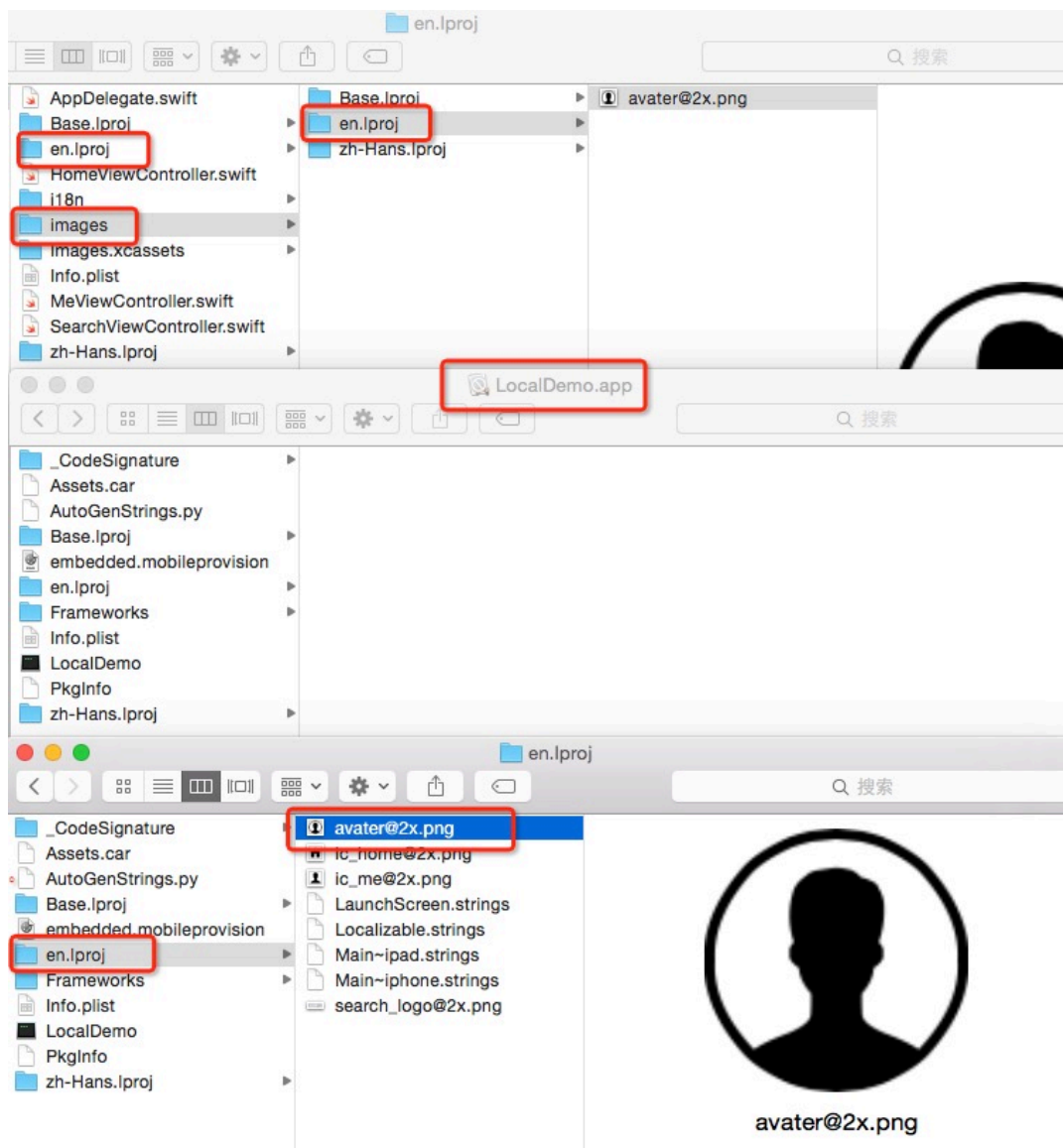
- view尺寸问题
- 关于AFNetworking
- 问题求助当隐藏状态栏的时候，导航栏也向上移了
- 如何在文本框的输入中只能输入70~100
- 现在用MacBook pro 13.3寸，8G内，256G配置会卡吗？
- 检测出来的一个内存泄漏的问题，一直没搞明白原因
- iOS中的数组和字典
- i车行——平行进口车交互平台上线!
- 关于YY直播

中文



每种语言都有自己的 **语言代码**.lproj文件夹，加载资源时只需要加载相应语言文件夹下的资源就OK，这一步可以系统为我们完成，也可以手动去做。

项目源代码中如果有多个不同目录的国际化资源，则会有产生多个xxx.lproj，但在编译打包后，会集中放在app的根目录中的xxx.lproj中，不信你看~



开始国际化

微博



CocoaChina

加关注

【iPhone 将于 2018 年用上三星的 OLED 屏?】据科技网站Ubergizmo报道，长期以来一直有传言称苹果将为iPhone换用OLED显示屏，而且三星也多次出现在iPhone OLED显示屏可能的供应商名单中。现在韩国媒体报道称，三星将向苹果供应用于2018年款iPhone的OLED显示屏。<http://t.cn/RU1Cj6o>



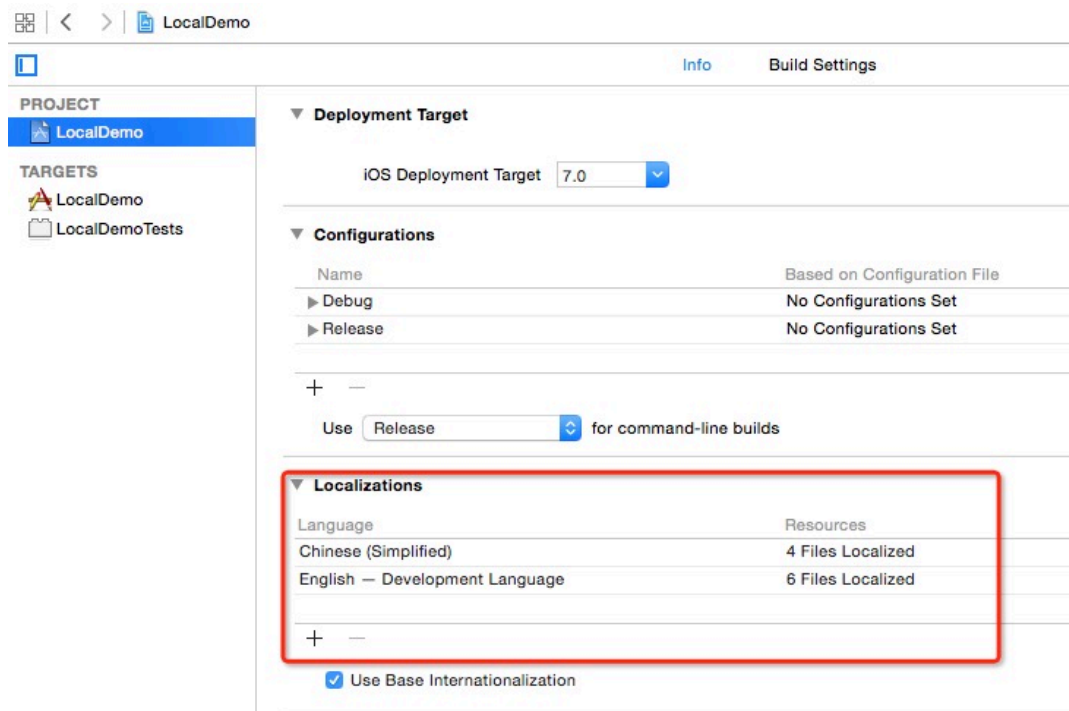
41分钟前

转发 | 评论

【iOS 开发 OpenGL 新手入门】说起 OpenGL，相信大不多数朋友都不会陌生，或多或少都有接触。本文不属于...

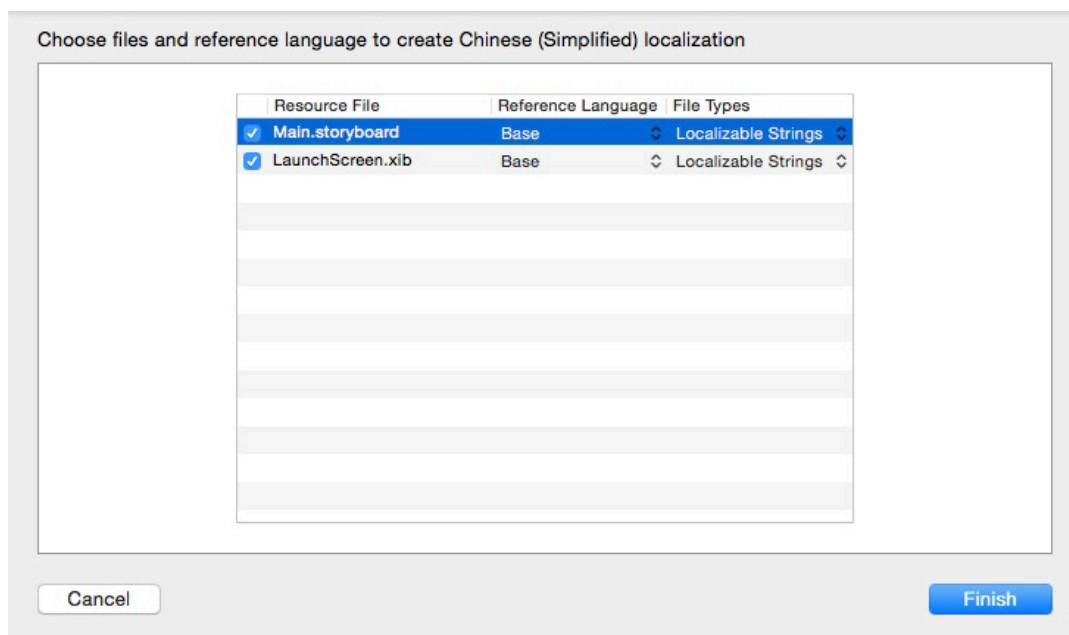


首先点击项目->PROJECT->Info->Localizations中添加要支持的语言。



此处Use Base Internationalization开启状态下，每个国际化资源文件会有个Base选项，主要针对String，Storyboard，Xib文件

在点击+ 添加相应语言时会弹出以下对话框，意思是为现有的资源添加语言文件，我们点击Finish就行了。



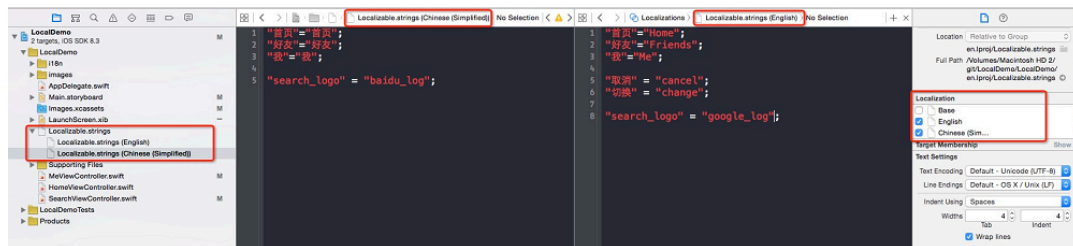
文本的国际化

主要针对代码中的字符串进行国际化，比如说一些消息，UI标题等。

我们通过一个Localizable.strings文件来存储每个语言的文本，它是iOS默认加载的文件，如果想用自定义名称命名，在使用NSLocalizedString方法时指定tableName为自定义名称就好了，但你的应用规模不是很大就不要分模块搞特殊

了。

每个资源文件如果想为一种语言添加支持，通过其属性面板中的Localization添加相应语言就行了，此时Localizable.strings处于可展开状态，子级有着相应语言的副本。我们把相应语言的文本放在副本里面就行了。



此处Base与前面提到过的开启Use Base Internationalization是有关联的，只有开启了全局Use Base Internationalization此处才会显示。那为什么这里没有勾选Base? Base作为一个基础模板，作用于Strings文件是没有太大意义的，另外去掉Base意味着在Base.lproj中少了一个strings文件，APP大小也有所下降，这点对于图片的Base更是如此。

在上图可以看到其实就是为每一套语言新建一份strings，其内容采用"key" = "value";的格式，注意有分号

我们在代码中这样写就行了

```
NSString *home = NSLocalizedString(@"首页",comment: "");
NSString *friends = NSLocalizedString(@"好友",comment: "");
NSString *me = NSLocalizedString(@"我",comment: "");
```

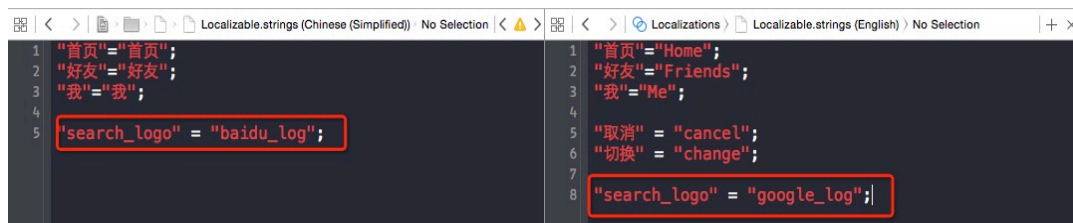
另外中文strings【Localizable.strings(Simplified)】可以不要的(可以理解为中文为APP的默认语言)，因为key就是value，当找不到相应的语言strings或value时会直接返回key。nice! 这样一来我们做文本的国际化就只要维护一个英文副本strings就OK了

图片的国际化

二种方案，通过原生支持与自定义命名

注意，新版Xcode中Images.xcassets不支持国际化（属性页面中没有Localization），Xcode5以前是支持的

- 方案一：自定义文本命名

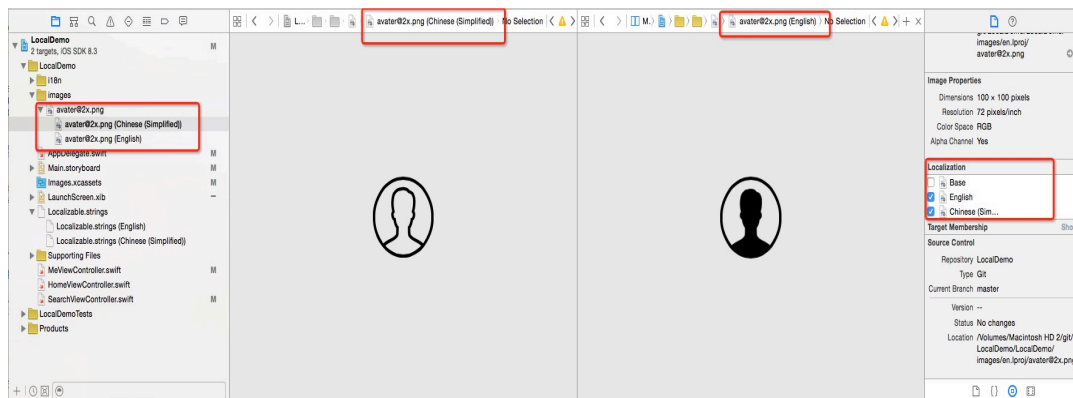


利用文本国际化的方式，在代码中调用

```
UIImage(named: NSLocalizedString(@"search_logo",comment: ""))
```

不推荐，一是因为做法太low了，工作量明显加大。二是不能在Storyboard或XIB中使用

- 方案二：原生支持



同上，Base副本去掉。另外需要注意的是，使用这种方式，在XIB或Storyboard中引用图片时如果只使用名称是实时显示不了的，一定要加上后缀名。如avatar.png

使用方式不变，iOS会自动找相应语言(xxx.lproj)下的图片

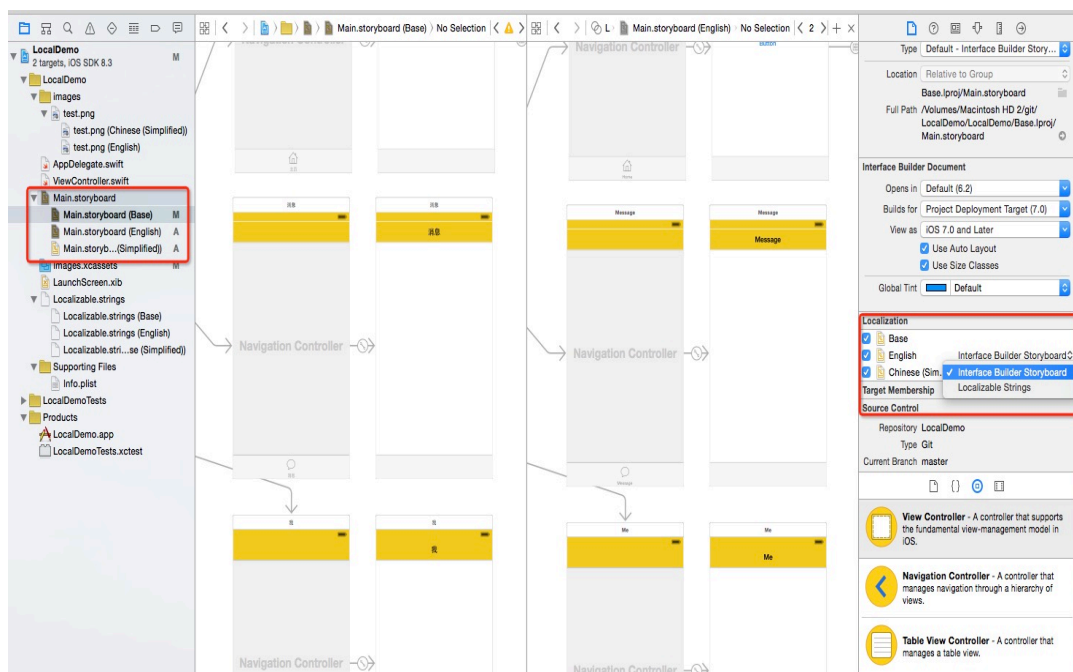
```
UIImage(named: "avater")
```

对于图片的放置，正确姿态应该是需要国际化的图片放在自定义Group里面，不需要国际化的图片放在 Images.xcassets

Storyboard&XIB的国际化

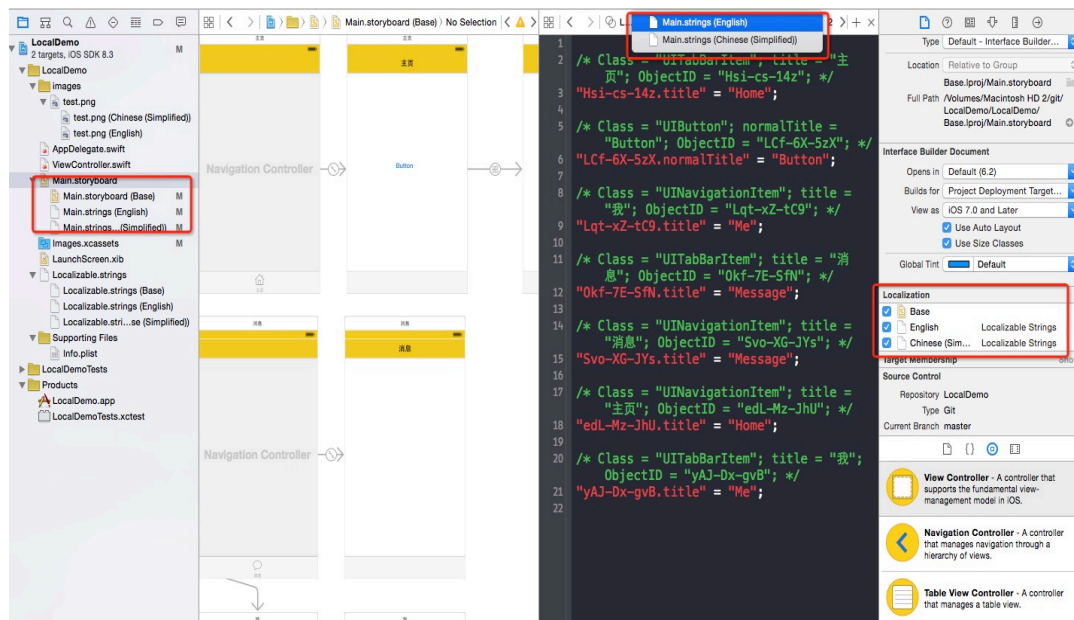
前面的两种资源国际化比较简单，但Storyboard国际化就稍微麻烦了点。同样它也有二种方案

- 方案一：每种语言定制一套Storyboard



在上图我们可以看到，每种语言都可以切换为strings或Storyboard（默认为strings）。如果选用Interface Builder Storyboard方案，那么每种语言都有一套相应的Storyboard，各个语言Storyboard间的界面改动不关联。

- 方案二：基于基础的Base StoryBoard以及每种语言一套strings



基于一个基础的Storyboard，可以看作是一个基础的模板，Storyboard里面所有的文本类资源(如UILabel的text)都会被放在相应语言的strings里面。此时我们为Storyboard里的字符类资源作国际化只需要编辑相应语言的strings就行了

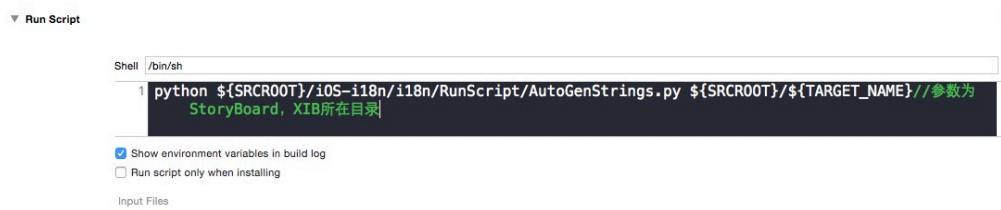
首选方案二。因为采用方案一，意味着你每改动一个界面元素就得去相应语言Storyboard一一改动，那跟为每个语言新起一个项目是一样的道理。但是采用方案二，我们只需改动Base Storyboard就行了。

注意，方案二中相应语言的strings一旦生成后，Base Storyboard有任何编辑都不会影响到strings，这就意味着如果我们删除或添加了一个UILabel的text，strings也不能同步改动。

还好，Xcode为我们提供了ibtool工具来生成Storyboard的strings文件。

```
ibtool Main.storyboard --generate-strings-file ./NewTemp.strings
```

但是ibtool生成的strings文件是BaseStoryboard的strings(默认语言的strings)，且会把我们原来的strings替换掉。所以我们要做的就是把新生成的strings与旧的strings进行冲突处理(新的附加上，删除掉的注释掉)，这一切可以用这个pytho脚本来实现，见AutoGenStrings.py。然后我们将借助Xcode 中 Run Script来运行这段脚本。这样每次Build时都会保证语言strings与Base Storyboard保持一致。



应用内切换语言

应用启动时，首先会读取NSUserDefaults中的key为AppleLanguages的内容，该key返回一个String数组，存储着APP支持的语言列表，数组的第一项为APP当前默认的语言。

在安装后第一次打开APP时，会自动初始化该key为当前系统的语言编码，如简体中文就是zh-Hans。

```
//获取APP当前语言
```

```
(NSUserDefaults.standardUserDefaults().valueForKey("AppleLanguages") as! Array)[0]
```

那么我们要实现语言切换改变AppleLanguages的值即可，但是这里有一个坑，因为苹果没提供给我们直接修改APP默认语言的API，我们只能通过NSUserDefaults手动去操作，且AppleLanguages的值改变后APP得重新启动后才会生效（才会读取相应语言的lproj中的资源，意味着就算你改了，资源还是加载的APP启动时lproj中的资源），猜测应该是框架层在第一次加载时对AppleLanguages的值进行了内存缓冲

```
//设置APP当前语言
var def = NSUserDefaults.standardUserDefaults()
def.setValue(["zh-Hans"], forKey:"AppleLanguages")
def.synchronize()
```

那么问题来了，如何做到改变AppleLanguages的值就加载相应语言的lproj资源？

其实，APP中的Storyboard的加载，图片与字符串的加载都是在NSBundle.mainBundle()上操作的，那么我们只要在语言切换后把NSBundle.mainBundle()替换成当前语言的bundle就行了，这样系统通过NSBundle.mainBundle()去加载资源时实则是加载的当前语言bundle中的资源

lproj目录可以用一个NSBundle表示：

```
import Foundation

/**
 * 当调用onLanguage后替换掉mainBundle为当前语言的bundle
 */
private let _bundle:UnsafePointer= unsafeBitCast(0,UnsafePointer.self)

class BundleEx: NSBundle {
    override func localizedStringForKey(key: String, value: String?, table tableName: String?) -> S
        if let bundle = languageBundle() {
            return bundle.localizedStringForKey(key, value: value, table: tableName)
        }else{
            return super.localizedStringForKey(key, value: value, table: tableName)
        }
}

extension NSBundle{
    private struct Static {
        static var onceToken : dispatch_once_t = 0
    }
    func onLanguage(){
        //替换NSBundle.mainBundle()为自定义的BundleEx
        dispatch_once(&Static.onceToken) {
            object_setClass(NSBundle.mainBundle(), BundleEx.self)
        }
    }

    //当前语言的bundle
    func languageBundle()->NSBundle?{
```



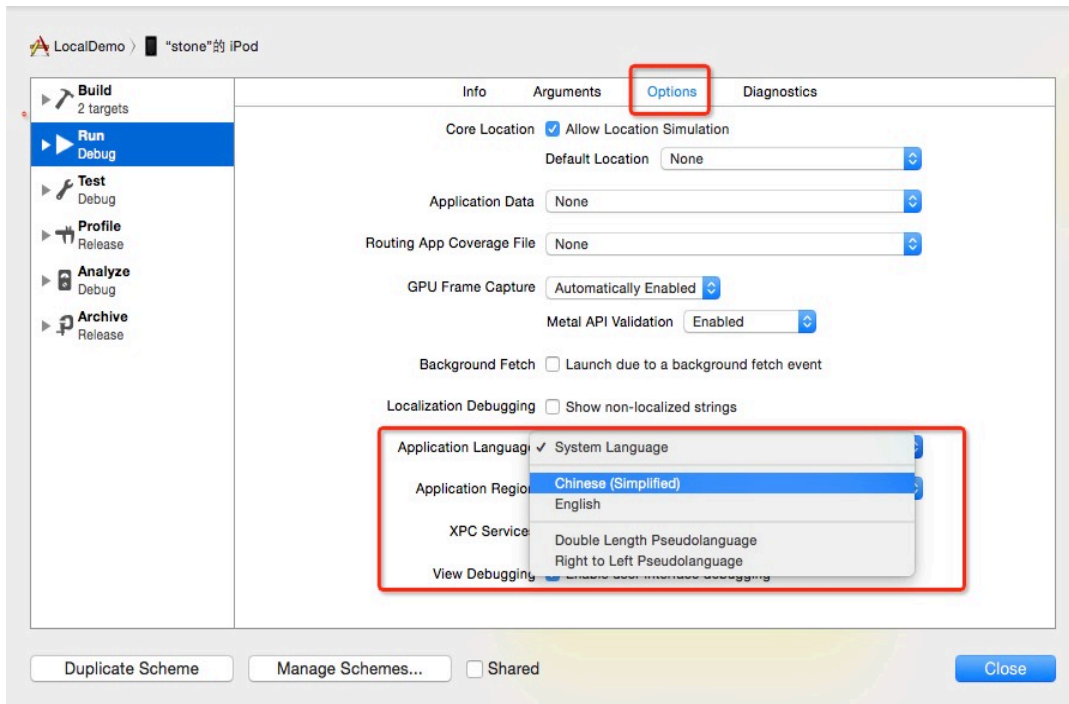
```
return Language.standardLanguage().currentLanguageBundle  
}  
}
```

以上Language是 iOS-i18n 开源库的一部分，我把项目中国际化部分封装了下，有兴趣的童鞋可以去看看

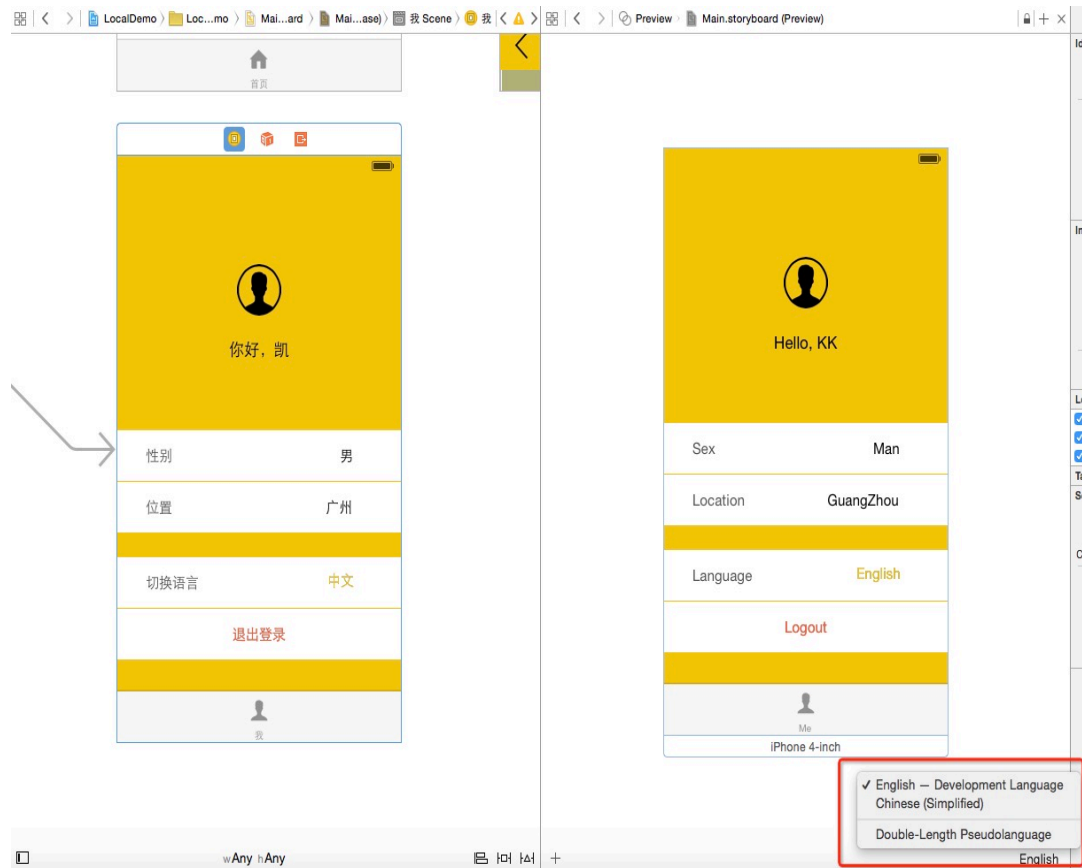
其他

- 设置运行语言环境

有时我们第一次安装APP时不想默认跟随系统，那么可以通过Xcode的scheme来指定特定语言



Storyboard实时预览，直接上图~



- IB中UIImageView国际化无效

解决办法就是为UIImageView扩展一个方法，然后通过IB中的User Defined Runtime Attributes把imageName传进去

```
extension UIImageView{
    var locale:String{
        get{
            return ""
        }
        set(newlocale){
            self.image = localizedImage(newlocale)
        }
    }
}
```

- IB中UITextView国际化无效

解决办法和UIImageView类似，扩展一个方法，然后把self.text做为key去strings文件中拿相应语言的value

```
extension UITextView{
    var locale:Bool{
        get{
            return true
        }
        set(newlocale){
            self.text = localized(self.text)
        }
    }
}
```

```
}  
  
}  
  
}
```

- LaunchScreen.xib的国际化

很遗憾，到目前为止，还不支持LaunchScreen.xib的国际化，我们只能通过自定义一个LaunchViewController来完成此需求，但也有些不足，就是应用启动时会黑屏一段时间，所以建议启动页面不要弄国际化。

参考：

[iOS国际化——通过脚本使storyboard翻译自增](#)

[Working with Localization](#)

[How to force NSLocalizedString to use a specific language](#)



微信扫一扫

订阅每日移动开发及APP推广热点资讯

公众号：CocoaChina

[我要投稿](#)

[收藏文章](#)

分享到：

8

上一篇：源码推荐(11.18)：全屏滑动返回，主题模式(一键换肤)

下一篇：iOS hybrid App 的实现原理及性能监测

相关资讯



想知道如何  Mac 提速？

立即开始

MacKeeper

我来说两句



您还没有登录！请 [登录](#) 或 [注册](#)

所有评论 (3)



elliotxin

2015-11-23 02:04:16

希望作者能出个OC版的，SWIFT看着费劲。

👍 0 💬 0 回复



何天雄

2015-11-22 02:27:59

挺有意思的。

👍 0 💬 0 回复



也许_____

2015-11-21 04:40:45

工程编译错误 python: can't open file '/Users/myMac/Downloads/LocalDemo-master/iOS-i18n/i18n/RunScript/AutoGenStrings.py': [Errno 2] No such file or directory

👍 0 💬 0 回复

[关于我们](#) [商务合作](#) [联系我们](#) [合作伙伴](#)

北京触控科技有限公司版权所有

©2015 Chukong Technologies, Inc.

京ICP备 11006519号

京ICP证 100954号

京公网安备11010502020289



京网文[2012]0426-138号