

CocoaPods

2015年1月23日 星期五 下午2:33

零、CocoaPods 是什么？

CocoaPods 是开发 OS X 和 iOS 应用程序的一个第三方库的依赖管理工具。利用 CocoaPods，可以定义自己的依赖关系 (称作 pods)，并且随着时间的变化，以及在整个开发环境中对第三方库的版本管理非常方便。

CocoaPods 背后的理念主要体现在两个方面。首先，在工程中引入第三方代码会涉及到许多内容。针对 Objective-C 初级开发者来说，工程文件的配置会让人很沮丧。在配置 build phases 和 linker flags 过程中，会引起许多人为因素的错误。CocoaPods 简化了这一切，它能够自动配置编译选项。

其次，通过 CocoaPods，可以很方便的查找到新的第三方库。当然，这并不是说你可以简单的将别人提供的库拿来拼凑成一个应用程序。它的真正作用是让你能够找到真正好用的库，以此来缩短我们的开发周期和提升软件的质量。

CocoaPods 的原理，它是将所有的依赖库都放到另一个名为 Pods 项目中，然后让主项目依赖 Pods 项目，这样，源码管理工作都从主项目移到了 Pods 项目中

1、Pods 项目最终会编译成一个名为 libPods.a 的文件，主项目只需要依赖这个 .a 文件即可。

2、对于资源文件，CocoaPods 提供了一个名为 Pods-resources.sh 的 bash 脚本，该脚本在每次项目编译的时候都会执行，将第三方库的各种资源文件复制到目标目录中。

3、CocoaPods 通过一个名为 Pods.xcconfig 的文件来在编译时设置所有的依赖和参数。

一、CocoaPods 安装：

Mac 下都自带 ruby，使用 ruby 的 gem 命令即可下载安装：

》安装 CocoaPods

```
sudo gem install cocoapods
```

》设置 pod 仓库

```
pod setup
```

》测试

```
pod --version
```

》卸载CocoaPods

```
sudo gem uninstall cocoapods
```

注意：

》如果gem太老可能会有问题，所以安装之前最好升级一下gem

```
sudo gem update --system
```

》ruby的软件源rubygems.org使用的亚马逊的云服务被墙了，所以需要更新一下ruby的源，如下代码将官方的ruby源替换成国内淘宝的源

```
gem sources --remove https://rubygems.org/
```

```
gem sources -a http://ruby.taobao.org/
```

```
gem sources -l
```

》pod setup在执行时，会输出Setting up CocoaPods master repo，但是会等待比较久的时间。这步其实是Cocoapods在将它的信息下载到~/.cocoapods目录下，如果你等太久，可以试着cd到那个目录，用du -sh *来查看下载进度。你也可以参考本文接下来的使用cocoapods的镜像索引一节的内容来提高下载速度。

》使用CocoaPods的镜像索引

所有的项目的Podspec文件都托管在<https://github.com/CocoaPods/Specs>。第一次执行pod setup时，CocoaPods会将这些podspec索引文件更新到本地的

~/.cocoapods/目录下，这个索引文件比较大，有100M左右。所以第一次更新时非常慢。

为了提高下载速度，可以将文件托管地址从github替换为国内oschina，这样会快很多。如下操作就可以将github替换为国内oschina

```
pod repo remove master
```

```
pod repo add master http://git.oschina.net/akuandev/Specs.git
```

```
pod repo update
```

二、CocoaPods使用：

使用时需要新建一个名为Podfile的文件

>1.利用vim创建Podfile。注意：Podfile文件应该和你的工程文件.xcodeproj在同一个目录下

```
vim Podfile
```

>2.将依赖的库名字依次列在文件中

```
platform :ios
```

```
pod 'JSONKit', '~> 1.4'
```

```
pod 'Reachability', '~> 3.0.0'
```

```
pod 'AFNetworking', '2.0.0'  
pod 'RegexKitLite'  
>3.保存并退出  
按下esc，然后输入:wq  
>4.利用CocoPods下载类库  
pod install
```

注意：

1.利用CocoPods管理类库后，以后打开项目就用xxx.xcworkspace打开，而不是之前的.xcodeproj文件

2.每次更改了Podfile文件，你需要重新执行一次pod update命令。

3.关于.gitignore

当你执行pod install之后，除了Podfile外，CocoaPods还会生成一个名为Podfile.lock的文件，你不应该把这个文件加入到.gitignore中。因为Podfile.lock会锁定当前各依赖库的版本，之后如果多次执行pod install不会更改版本，要pod update才会改Podfile.lock了。这样多人协作的时候，可以防止第三方库升级时造成大家各自的第三方库版本不一致。

4. CocoaPods在执行pod install和pod update时，会默认先更新一次CocoPods的spec仓库索引。使用--no-repo-update参数可以禁止其做索引更新操作

```
pod install --no-repo-update
```

```
pod update --no-repo-update
```

三、查找第三方库

你如果不知道cocoaPods管理的库中，是否有你想要的库，那么你可以通过pod search命令进行查找，以下是我用pod search json查找到的所有可用的库：

```
pod search AFNetworking
```

四：参考文献

用CocoaPods做iOS程序的依赖管理：

<http://www.devtang.com/blog/2014/05/25/use-cocoapod-to-manage-ios-lib-dependency/>

CocoaPods最佳实践：<http://weibo.com/p/1001603800875490492754>

极速化 CocoaPods：<http://www.cocoachina.com/ios/20150123/11002.html>

深入理解CocoaPods：<http://objccn.io/issue-6-4/>