

CocoaPods

From Farm to Project

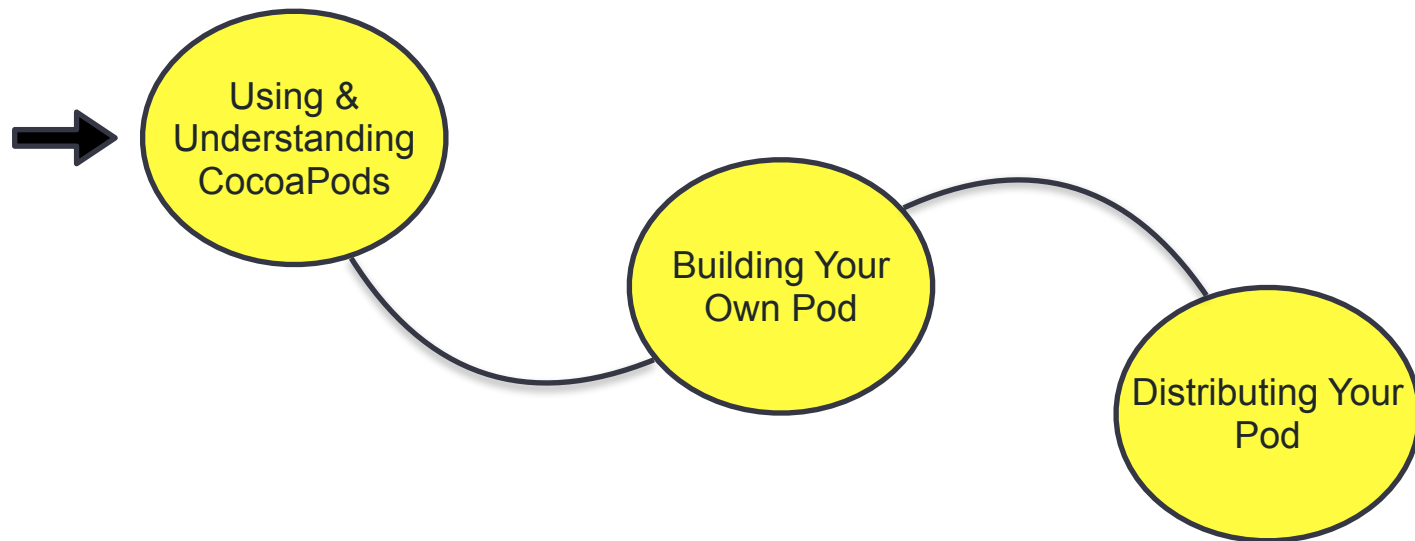


Jeriel Ng (@jerielng)

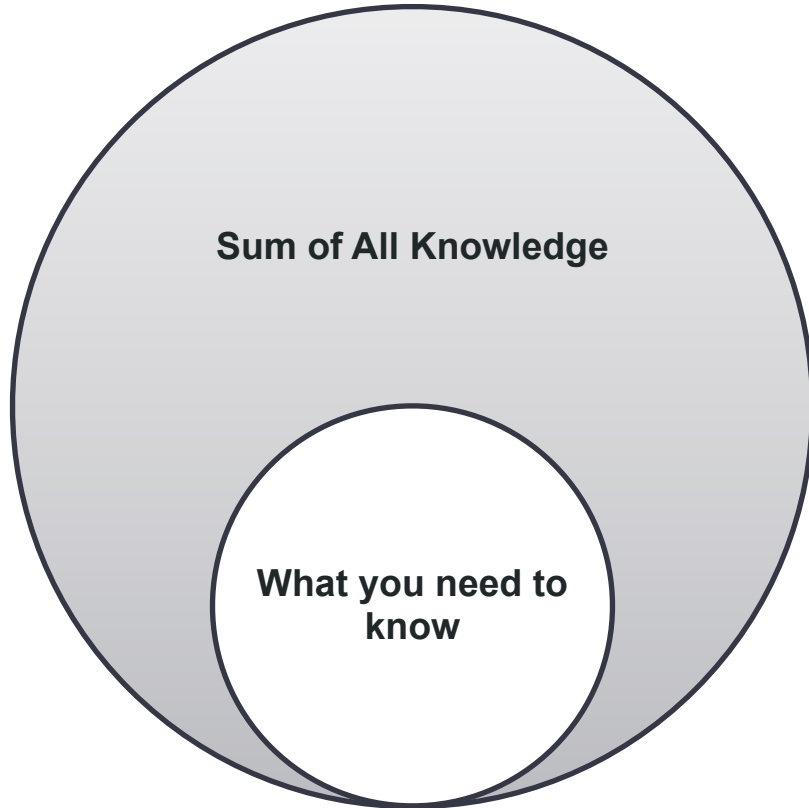
Mobile Software Engineer



Roadmap



A Disclaimer



Quick Notes

- When in doubt, refer to the documentation
- Use what you need. Learn the rest along the way

CocoaPods vs. Other Package Managers



CocoaPods



Carthage



Swift Package Manager

Points to Consider

- Which one does your project already use?
- Which one supports the packages you need?
- Centralized or decentralized?

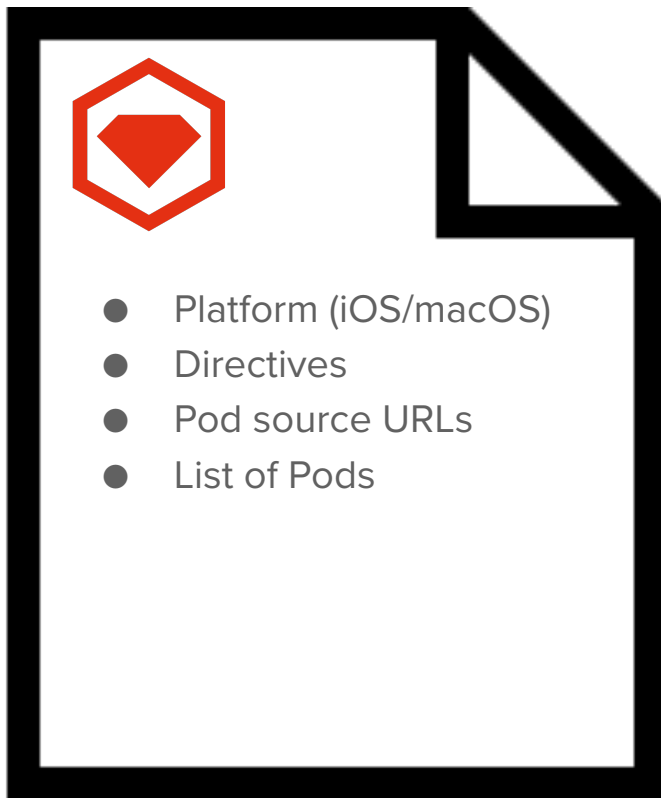
Breaking down the Podfile

Example Podfile

- [artsy/eidolon](#)



`pod init`



- Platform (iOS/macOS)
- Directives
- Pod source URLs
- List of Pods

Versioning your Pods

```
pod 'MyCoolPod', '1.0.0'  
pod 'MyCoolPod2', '>= 1.2.0'  
pod 'MyCoolPod3', '< 1.0'  
  
# Trying out different semantic versions  
pod 'MichaelPod', '~> 1.1.0' # Updates to any version up to 1.1.9  
pod 'MichaelPod', '~> 1.1'   # Updates to any version up to 1.9  
pod 'MichaelPod', '~> 1'     # Updates to any version up to whatever
```

Hooks: Pre- and post-install steps

pre_install/post_install

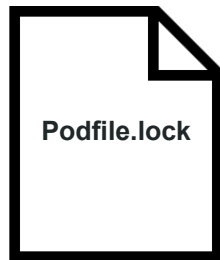
```
post_install do |installer|
  installer.pods_project.targets.each do |target|
    target.build_configurations.each do |config|
      config.build_settings['YOUR_BUILD_SETTING'] = 'whatever'
    end
  end
end
```

Common CocoaPods commands

```
pod install          # Installs any new changes to Podfile
pod update           # Updates dependencies to latest (possible) versions
pod deintegrate      # Removes CocoaPods and any caches from project
pod outdated         # Lists pods with new versions available
```


Analyzing the Podfile.lock

- Not to be confused with Podfile
- Should not be edited manually
- List of dependency and sub-dependency versions



A mismatch causes a build failure

✖ The sandbox is not in sync with the Podfile.lock. Run 'pod install' or update your CocoaPods installation.

VS

Pods/



- Hangs out in Pods folder
- Compared against Podfile.lock
- Should not be edited manually

Podfile.lock - Using the wrong CocoaPods version

Podfile.lock

```
SwiftLint: 2c16e8fa99dac2e440dc0c70ea74d2532048f6cf  
WeddingModelService: fef2be6a5f2ad44b11f32c81cd57e064cb7bdbf2  
  
PODFILE CHECKSUM: e6ba4067bcd8e1761164ea41a416668b083d551f  
  
COCOAPODS: 1.11.3
```

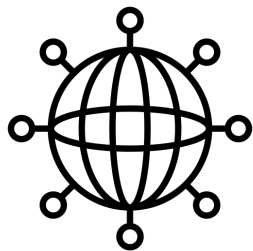
- Updated on `pod install` or `pod update`
- Should be standardized by team

Installing with an incompatible CocoaPods version can cause runtime errors

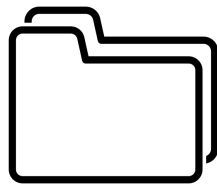
```
# Managing different versions of CocoaPods  
pod _1.10.0_ install  
pod _1.11.3_ install
```

Building Your Own

Where do Pods come from?



Specs Repo



Local Directory



Library Git Repo



A podspec defines how and what to install for the associated Pod

Building and understanding your podspec

- [Syntax Reference](#)
- Defines
 - Name, version, and other metadata
 - Relative path to Pod's source code
 - Pod dependencies

Example Podspec Files

- [Firebase](#)
- [AFNetworking](#)
- [SwiftLint](#)

A sample podspec

```
Pod::Spec.new do |s|
  s.name           = 'WeddingModelService'
  s.version        = '0.1.0'
  s.summary        = 'Library of model objects and networking utilities for The Wedding App.'
  s.homepage       = 'https://github.com/jerielng/WeddingModelService'
  s.license        = { :type => 'MIT', :file => 'LICENSE' }
  s.author         = { 'jerielng' => 'jeriel.ng@gmail.com' }
  s.source         = { :git => 'https://github.com/jerielng/WeddingModelService.git', :tag => s.version.to_s }

  s.ios.deployment_target = '13.0'

  s.source_files = 'WeddingModelService/Classes/**/*'

  # s.resource_bundles = {
  #   'WeddingModelService' => ['WeddingModelService/Assets/*.png']
  # }

  # s.public_header_files = 'Pod/Classes/**/*.h'

  s.static_framework = true

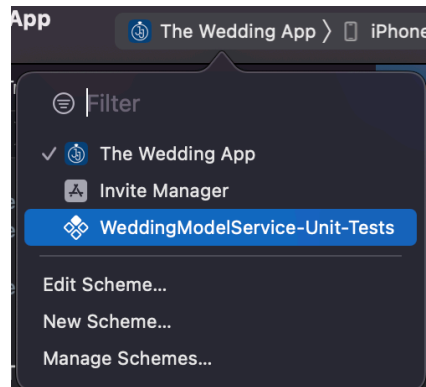
  s.dependency 'Firebase/Analytics', '6.34.0'
  s.dependency 'Firebase/Storage', '6.34.0'
  s.dependency 'Firebase/Firestore', '6.34.0'
  s.dependency 'FirebaseFirestoreSwift', '>= 0.4.0'
end
```

Adding tests to your podspec

- [Guide: Testing with CocoaPods](#)

```
# Podspec
Pod::Spec.new do |s|
  ...
  s.test_spec 'Tests' do |test_spec|
    test_spec.source_files = 'Tests/**/*.swift'
  end
  ...
end
```

```
# Podfile
pod 'MyCoolPod', :testspecs => ['Tests']
```



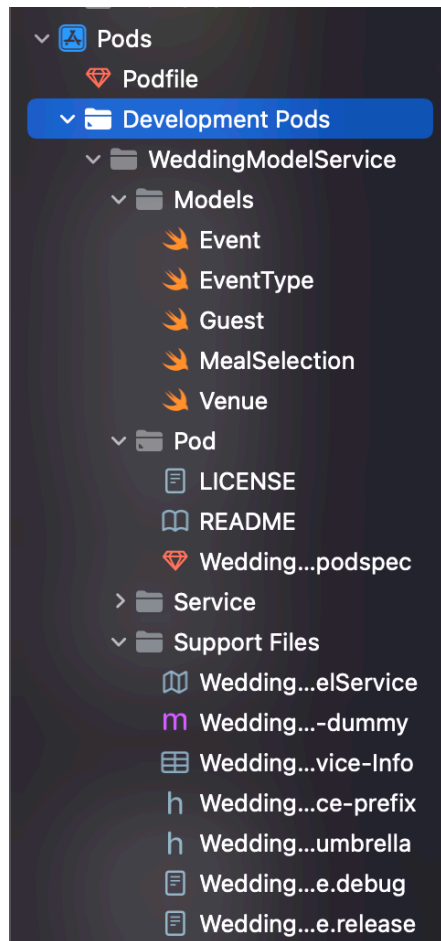
Distribution Methods

Developing a local pod

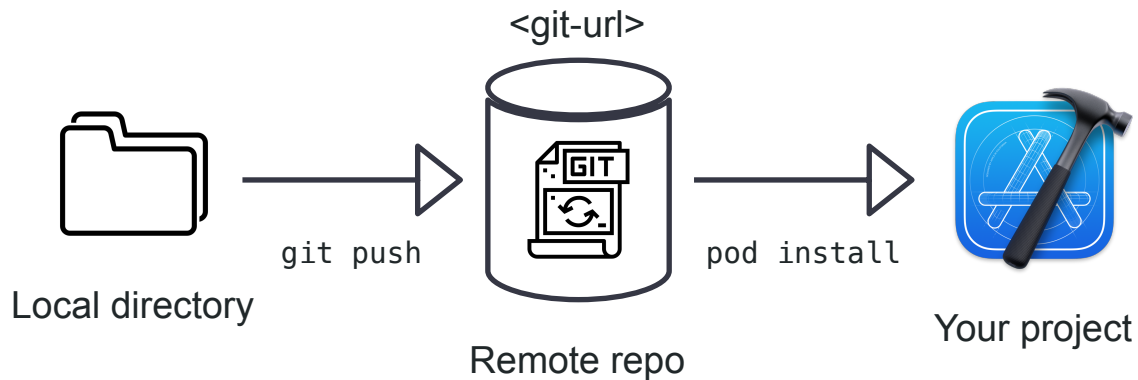
- Development Pods
 - Built from a local path or Git submodule
 - Allows you to make changes without losing them on pod update

Podfile

```
pod 'MyCoolPod', :path => './<path-to-dir>'
```

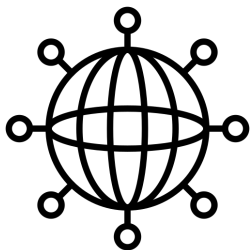


Hosting a pod on a Git remote



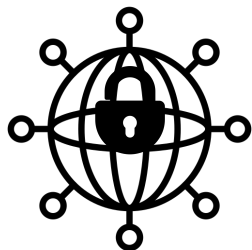
```
pod 'MyCoolPod', :git => '<git-url>', :commit => '<hash>'
pod 'MyCoolPod', :git => '<git-url>', :branch => '<branch-name>'
pod 'MyCoolPod', :git => '<git-url>', :tag => '<tag>'
```

Specs! Public and Private



Trunk (Public)

<https://cdn.cocoapods.org/>



Your Private Repo

```
pod repo list
```

```
jerieIng-specs
- Type: git (main)
- URL:  git@github.com:jerieIng/Specs.git
- Path: /Users/jerieIng/.cocoapods/repos/jerieIng-specs

trunk
- Type: CDN
- URL:  https://cdn.cocoapods.org/
- Path: /Users/jerieIng/.cocoapods/repos/trunk

2 repos
```

```
pod repo <command>
```

Publishing your pod

```
# Publish publicly to trunk
```

```
# or
```

```
# Private specs repo
```

```
# 1. Create a Git repository
```

```
# 2. Create a reference on your local machine  
pod repo add <your-repo> <git-url>
```

```
# 3. Push your podspec  
pod repo push <your-repo> <spec>.podspec
```

```
Updating the `jerieIng-specs' repo
```

```
Adding the spec to the `jerieIng-specs' repo
```

```
- [Add] WeddingModelService (0.1.0)
```

```
Pushing the `jerieIng-specs' repo
```

Why create a Pod?

- Allows you to share code across projects
- Trains you to write standalone code
- Logical separation of product capabilities

Further Resources

- [CocoaPods Guides](#)
- [Cocoapods Ruby Documentation](#)
- [objc - CocoaPods Under The Hood](#)

Questions
