

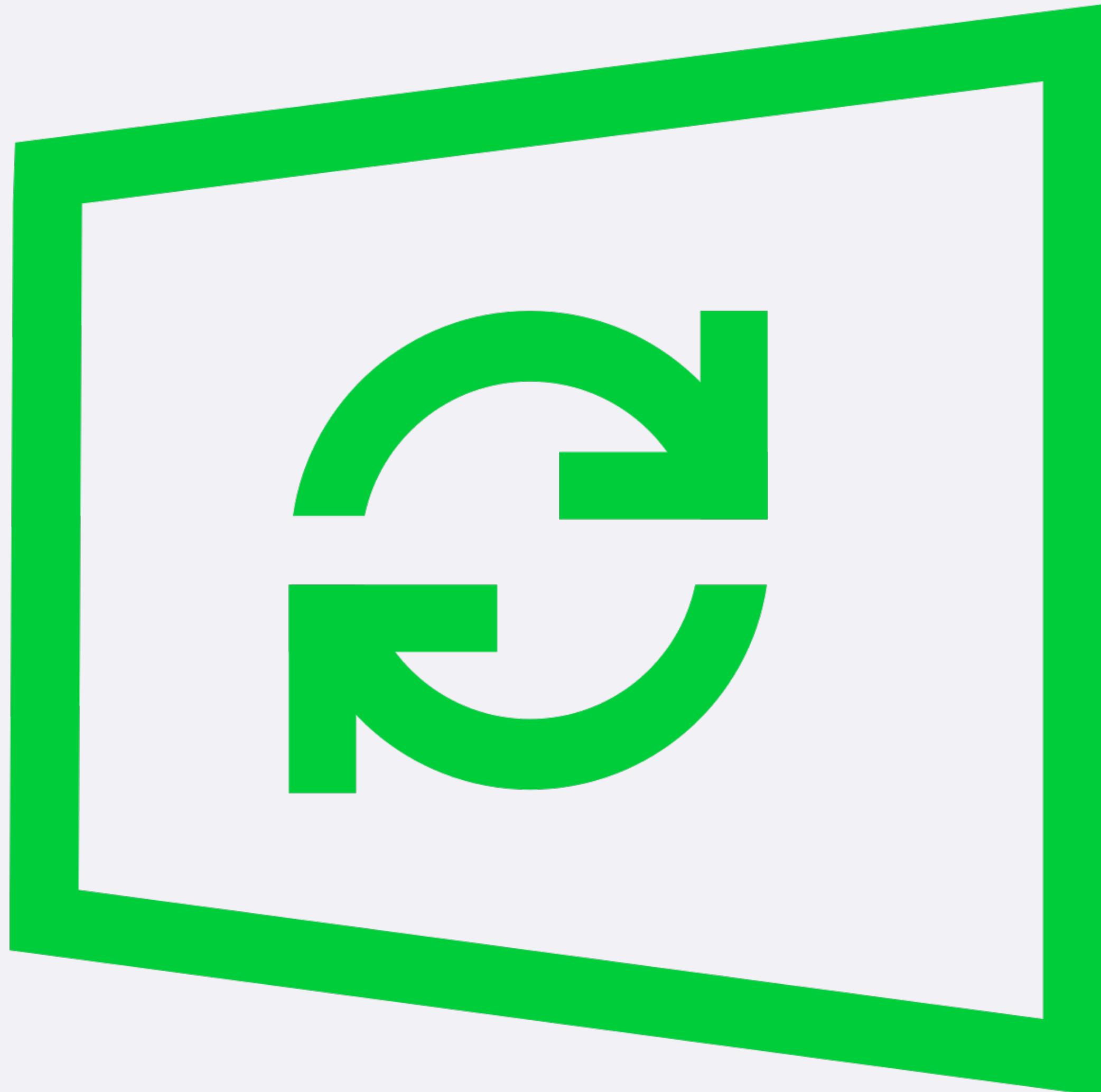
Package managers

Что такое Package manager?

Краткий обзор

Создаем свой пакет

Зачем нужны пакетные менеджеры?



Зачем нужны пакетные менеджеры?

- Установка
- Обновление
- Настройка
- Удаление

Cocoapods

(COCOAPODS)

APP ABOUT GUIDES BLOG

SEARCH*



* Type here to search by name, author and summary.

Cocoapods

 CocoaPods / Specs

Watch ▾ 257 Star 5.5k Fork 8.3k

Code Pull requests 1 Actions Security Insights

The CocoaPods Master Repo <https://cocoapods.org>

433,558 commits 7 branches 0 packages 2 releases 24,256 contributors

Branch: master ▾ New pull request Create new file Upload files Find file Clone or download ▾

 didiaodanding and CocoaPodsBot [Add] LFTestPods 0.0.1 Latest commit 70b51d2 now

File	Description	Time
Scripts	Add attributes for Algolia search + do not redirect root YAML files	3 months ago
Specs	[Add] LFTestPods 0.0.1	now
.gitignore	Ignore _site for easier dev	5 months ago
.rubocop.yml	rewrite index generator and add deprecated podspec index	4 months ago
AlgoliaSearch.yml	Add attributes for Algolia search + do not redirect root YAML files	3 months ago
CocoaPods-version.yml	CocoaPods release 1.8.4	12 days ago
Gemfile	rewrite index generator and add deprecated podspec index	4 months ago
Gemfile.lock	rewrite index generator and add deprecated podspec index	4 months ago
README.md	[Readme] Remove Travis badge	6 years ago
netlify.toml	Netlify massaging	5 months ago

Cocoapods

```
$ pod spec create Peanut  
$ edit Peanut.podspec  
$ pod spec lint Peanut.podspec
```

Cocoapods

```
Pod::Spec.new do |spec|
    spec.name          = 'Reachability'
    spec.version       = '3.1.0'
    spec.license        = { :type => 'BSD' }
    spec.homepage      = 'https://github.com/tonymillion/Reachability'
    spec.authors        = { 'Tony Million' => 'tonymillion@gmail.com' }
    spec.summary        = 'ARC and GCD Compatible Reachability Class for iOS and OS X.'
    spec.source         = { :git => 'https://github.com/tonymillion/Reachability.git', :tag => 'v3.1.0' }
    spec.module_name    = 'Rich'
    spec.swift_version = '4.0'

    spec.ios.deployment_target = '9.0'
    spec.osx.deployment_target = '10.10'

    spec.source_files     = 'Reachability/common/*.swift'
    spec.ios.source_files = 'Reachability/ios/*.swift', 'Reachability/extensions/*.swift'
    spec.osx.source_files = 'Reachability/osx/*.swift'

    spec.framework        = 'SystemConfiguration'
    spec.ios.framework    = 'UIKit'
    spec.osx.framework    = 'AppKit'

    spec.dependency 'SomeOtherPod'
end
```

Cocoapods Trunk

- pod trunk register name1@domain.org 'Name' —description='specs'
- pod trunk push [NAME.podspec]
pod repo push REPO [NAME.podspec]
- pod trunk add-owner [NAME] name2@domain.org

Cocoapods integration

```
# 1
source 'https://github.com/CocoaPods/Specs.git'
# 2
platform :ios, '9.0'
# 3
use_frameworks!

# 4
target 'EasyToDo' do
    # 5
    pod 'Alamofire'
end
```

Cocoapods integration

```
pod install
```

Cocoapods

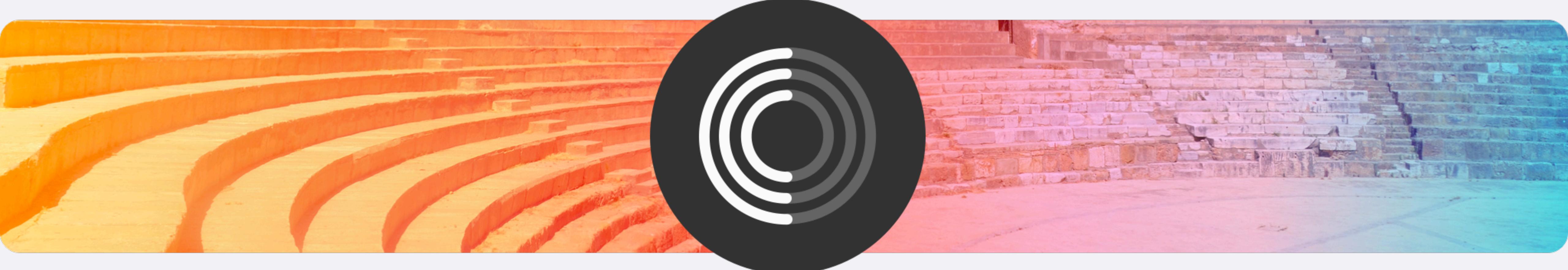


Демо

Cocoapods Disadvantages

- Долгая первая установка
- Все зависимости компилируются вместе с проектом
- Нужно изменить исходный проект

Carthage



Carthage

brew install carthage

Cartfile

```
github "Alamofire/Alamofire"
```

Installation

carthage update

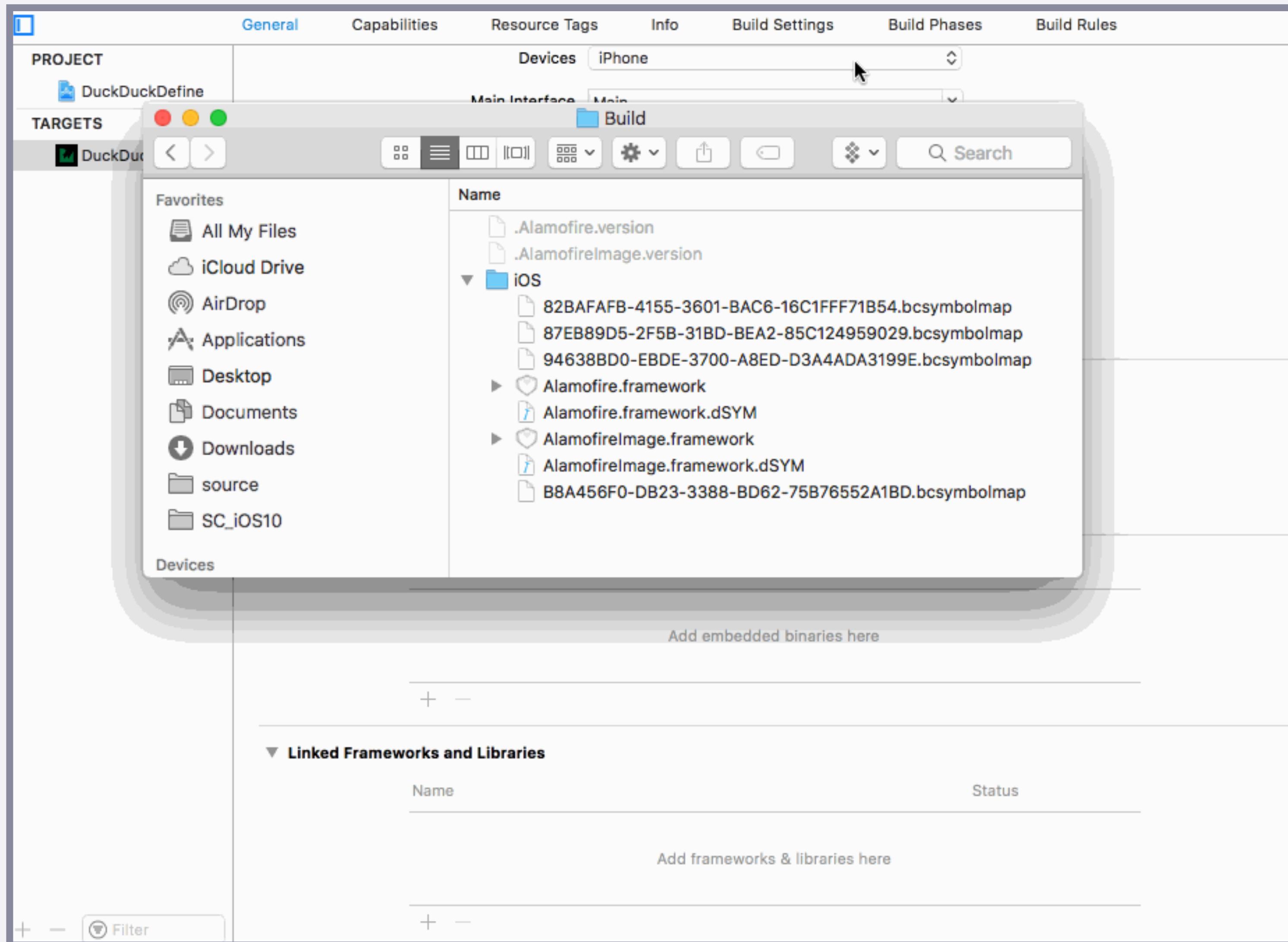
Installation



Installation

```
carthage update --platform iOS
```

Installation



Installation

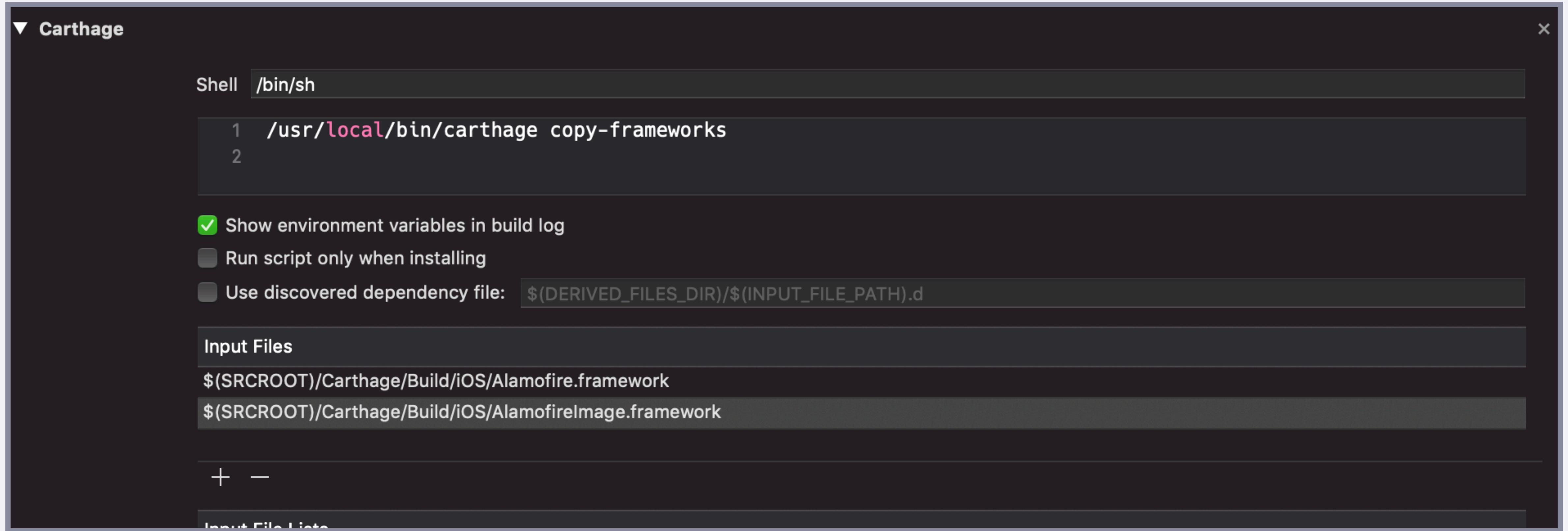
▼ Carthage x

Shell `/bin/sh`

```
1 /usr/local/bin/carthage copy-frameworks
2
```

Show environment variables in build log

Installation



The screenshot shows the Xcode interface with the "Carthage" build phase selected. The "Script" tab is active, displaying the command:

```
1 /usr/local/bin/carthage copy-frameworks
```

Below the script, there are three configuration options:

- Show environment variables in build log
- Run script only when installing
- Use discovered dependency file: `$(DERIVED_FILES_DIR)/$(INPUT_FILE_PATH).d`

The "Input Files" section lists the frameworks being copied:

- `$(SRCROOT)/Carthage/Build/iOS/Alamofire.framework`
- `$(SRCROOT)/Carthage/Build/iOS/AlamofireImage.framework`

At the bottom, there are "+" and "-" buttons for managing input files, and a "Input File List" button.

Carthage Disadvantages

- Долгая и сложная настройка проекта
- Долгий и сложный процесс поддержки Carthage
- Меньше проектов доступно

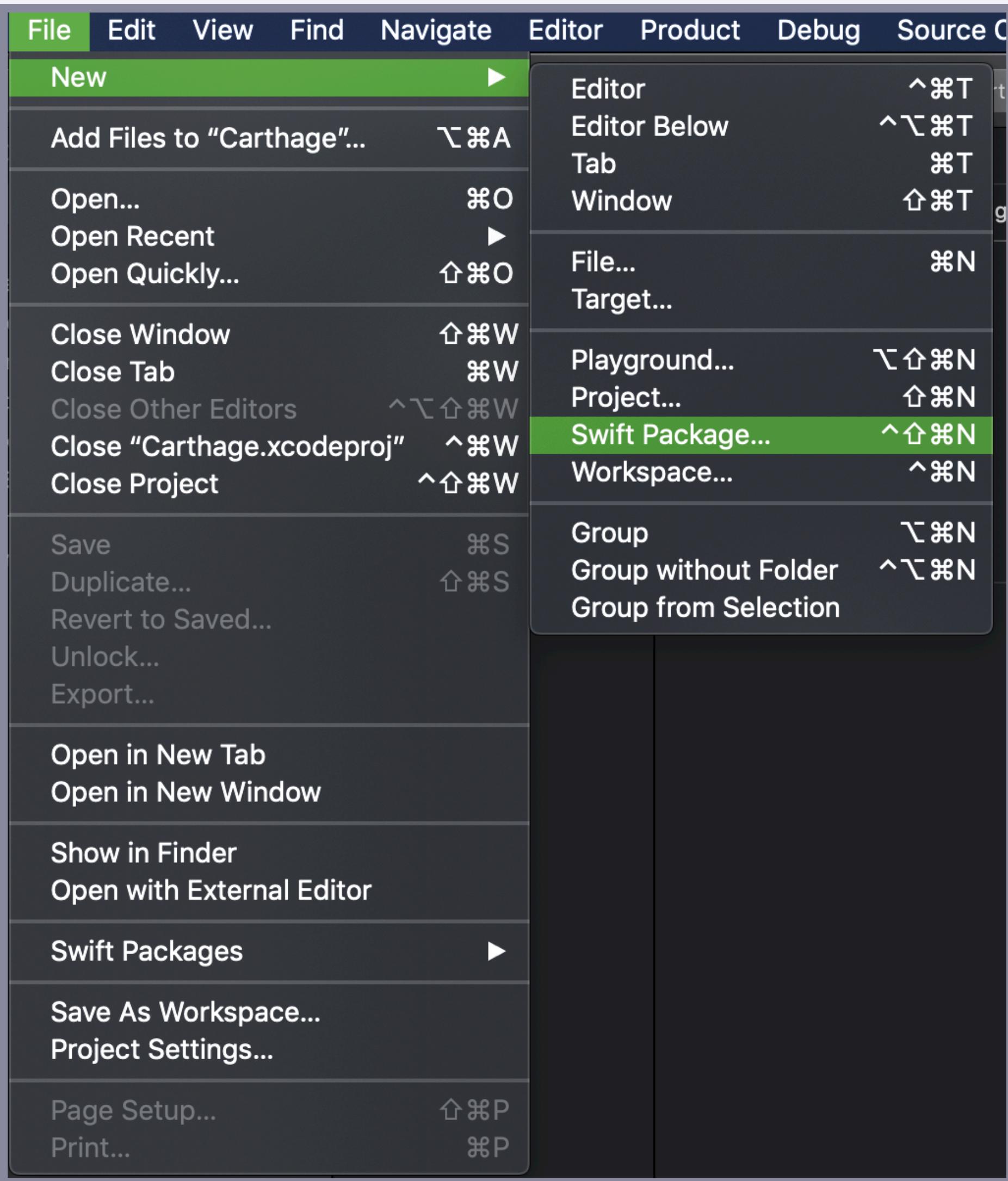
Swift Package Manager



Swift Package Manager

```
$ mkdir TodoKit  
$ cd TodoKit  
$ swift package init
```

Swift Package Manager



Swift Package Manager

```
// swift-tools-version:5.1

import PackageDescription

let package = Package(
    name: "TodoKit",
    products: [
        // The external product of our package is an importable
        // library that has the same name as the package itself:
        .library(
            name: "TodoKit",
            targets: ["TodoKit"]
        )
    ],
    targets: [
        // Our package contains two targets, one for our library
        // code, and one for our tests:
        .target(name: "TodoKit"),
        .testTarget(
            name: "TodoKitTests",
            dependencies: ["TodoKit"]
        )
    ]
)
```

Swift Package Manager

```
let package = Package(  
    ...  
    dependencies: [  
        // Here we define our package's external dependencies  
        // and from where they can be fetched:  
        .package(  
            url: "https://github.com/johnsundell/files.git",  
            from: "4.0.0"  
        )  
    ],  
    targets: [  
        .target(  
            name: "TodoKit",  
            // Here we add our new dependency to our main target,  
            // which lets us import it within that target's code:  
            dependencies: ["Files"]  
        ),  
        .testTarget(  
            name: "TodoKitTests",  
            dependencies: ["TodoKit"]  
        )  
    ]  
)
```

Swift Package Manager

```
.package(  
    url: "https://github.com/johnsundell/files.git",  
    .exact("4.0.0")  
)
```

Swift Package Manager

```
// Depending on a branch (master in this case):
.package(
    url: "https://github.com/johnsundell/files.git",
    .branch("master")
)

// Depending on an exact commit:
.package(
    url: "https://github.com/johnsundell/files.git",
    .revision("0e0c6aca147add5d5750ecb7810837ef4fd10fc2")
)
```

Демо

SPM Disadvantages

- Xcode 11

Почитать

1. <https://www.codementor.io/blog/swift-package-manager-5f85eqvygj>
2. https://developer.apple.com/documentation/swift_packages/package
3. <https://cocoapods.org>
4. <https://github.com/Carthage/Carthage>
5. <https://www.raywenderlich.com/416-carthage-tutorial-getting-started>
6. <https://www.swiftbysundell.com/articles/managing-dependencies-using-the-swift-package-manager/>
7. <https://www.ralfbert.de/ios/swift-package-manager-for-ios-projects/>

Домашнее задание

Создать свой пакет для carthage

<https://github.com/Carthage/Carthage#supporting-carthage-for-your-framework>

