

THIRD PARTY FRAMEWORKS

Zeke Abuhoff

Lead iOS Instructor, General Assembly

THIRD PARTY FRAMEWORKS

LEARNING OBJECTIVES

- + Explain the utility of third party frameworks
- + Describe what a package manager does
- + Equip a project with cocoapods

THIRD PARTY FRAMEWORKS

FIRST PARTY FRAMEWORKS

So far, you've used several different **frameworks**. A framework is a module of code that encapsulates a particular set of functionality. Here are some you may have used:

UIKit

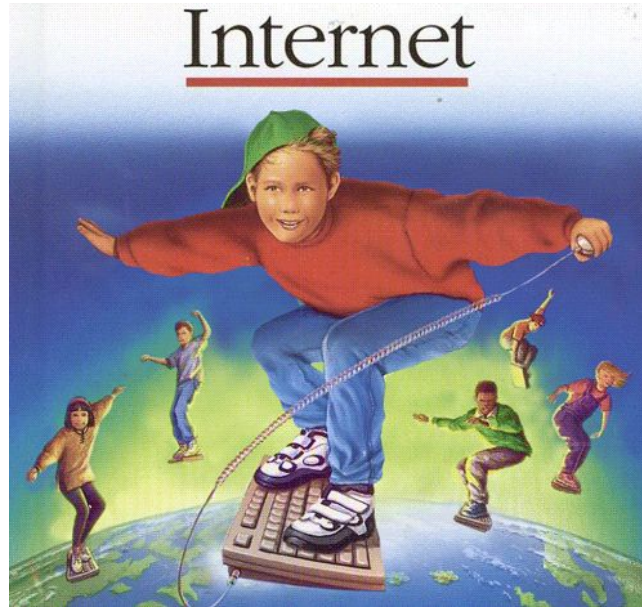
Foundation

CoreLocation

Those examples were made by Apple. Frameworks written by Apple come bundled with iOS and cover lots of basic stuff you may want your app to do.

But if what if you wanted your app to do something more than the basics?

THIRD PARTY FRAMEWORKS



TO THE INTERNET!

THIRD PARTY FRAMEWORKS

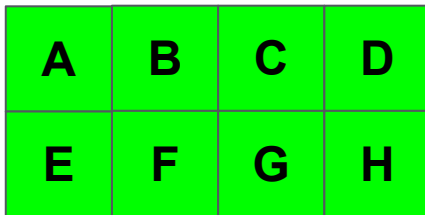
MODULAR CODE

Developers today often extol the value of **modularity**. This term refers to how self-contained a piece of software is - how independently it can do its job.

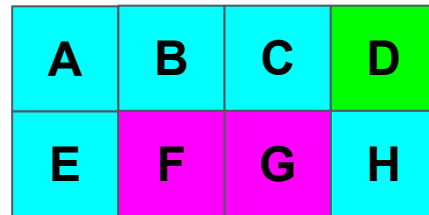
Oftentimes, developers deliberately make their code so modular in its design that it can be ported to other people's projects. In iOS, this takes the form of third party frameworks.



NOT MODULAR



MODULAR



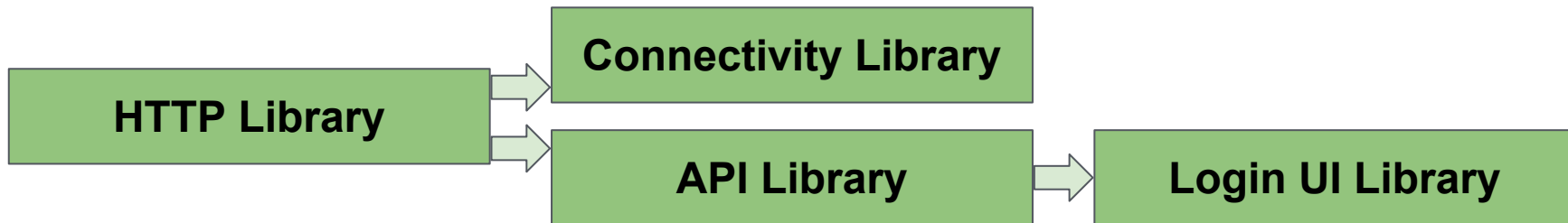
MODULAR

THIRD PARTY FRAMEWORKS

PACKAGE MANAGERS

While frameworks can be handled individually, it's typical in contemporary software development to use **package managers**.

Package managers are programs that download code libraries and manage these libraries' versions and **dependencies**. When one library relies on code from another library, that relationship is called a dependency.



THIRD PARTY FRAMEWORKS

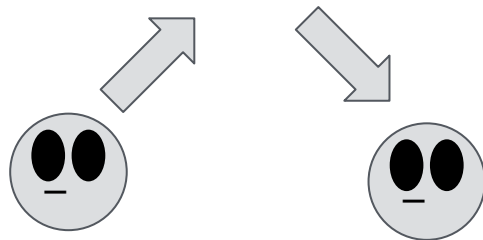
COCOAPODS

The most popular package manager for iOS frameworks is the service known as Cocoapods.

The logo for Cocoapods, featuring the word 'COCOAPODS' in a bold, sans-serif font. The 'COCO' part is white and set against a red rectangular background, while 'APODS' is red.

Developers package frameworks they've made as pods and submit them to be listed by the service.

Other developers then use the program Cocoapods to download the pods that are relevant for their own projects. Rather than reinventing something, Cocoapods lets you plug in what others have already made.

A smaller version of the Cocoapods logo, with 'COCO' in white on a red background and 'APODS' in red.

THIRD PARTY FRAMEWORKS

COCOPODS

Use cocoapods through the command line.

```
pod init
```

Creates Podfile

```
pod install
```

Sets up pods based on Podfile

```
pod update
```

Updates pods with latest changes

The Podfile is simply a text file where you list all the cocoapods you want in your project.

THIRD PARTY FRAMEWORKS

COCOPODS

Practice:

- 1) Create a new Xcode project.
- 2) Set up cocoapods in that project.
- 3) Install the pods SwiftyJSON and Alamofire in your project.

```
pod init
```

Creates Podfile

```
pod install
```

Sets up pods based on Podfile

```
pod update
```

Updates pods with latest changes

THIRD PARTY FRAMEWORKS

CARTHAGE

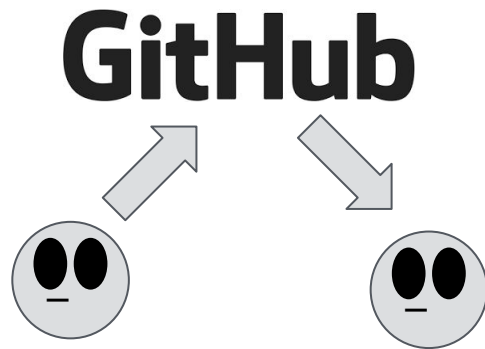
Another option for getting third party frameworks is Carthage.

Instead of listing all frameworks in one place, Carthage just grabs a framework from whatever url you give it (typically a Github repo).



This decentralized approach makes Carthage handy for setting up frameworks of your own. Also, CocoaPods configures an xcworkspace for you. Carthage doesn't alter your project directly.

Carthage isn't better or worse than CocoaPods - just different.



THIRD PARTY FRAMEWORKS

OPEN SOURCE

You've probably heard people refer to some projects being **open source**.

This means that the source code for a project is openly available to the public.

There are different licenses that lay out the exact terms, but the basic idea is that open source code can be freely used by people who didn't write it.

Services such as Cocoapods are built in the spirit of open source collaboration, but those licenses still matter. Make sure you know the legal ramifications of any code you're using that you didn't write.

