

# Static, Dynamic Framework 그리고 Encapsulation



kakaobank

안정민

# 소개

---

- 現 한국카카오은행 iOS 개발자 (16.10 ~ )
- 블로그 운영 : [minsone.github.io](https://minsone.github.io)

# 목차

---

- Framework
- Static, Dynamic Framework
- Framework Analytics Command
- Encapsulation
- QnA

# Framework

# Framework

Dynamic shared library, nib, 다국어 문자열, 이미지, 헤더 파일, 레퍼런스 문서들을 한 패키지로 묶어 계층화된 폴더에 모아놓은 것.

- Apple Document / Framework Programming Guide / What are Frameworks?

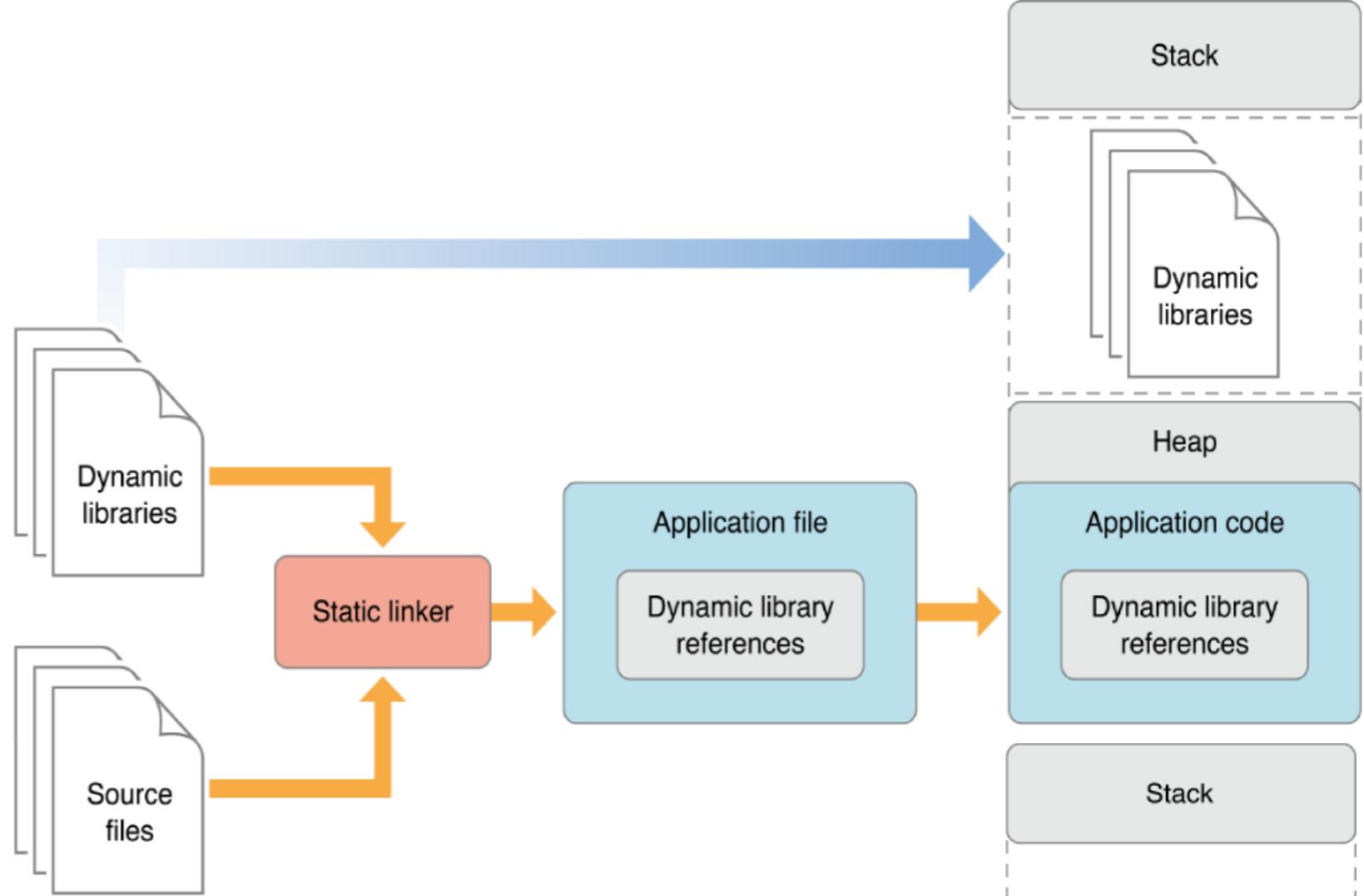


```
→ ~ tree /Library/Frameworks/PluginManager.framework  
/Library/Frameworks/PluginManager.framework  
└── PluginManager -> Versions/Current/PluginManager  
└── Resources -> Versions/Current/Resources  
└── Versions  
    ├── A  
    │   └── PluginManager  
    │   └── Resources  
    │       ├── English.lproj  
    │       │   └── InfoPlist.strings  
    │       ├── Info.plist  
    │       └── version.plist  
    │       └── CodeSignature  
    │           └── CodeResources  
    └── B  
        └── PluginManager  
        └── Resources  
            ├── English.lproj  
            │   └── InfoPlist.strings  
            ├── Info.plist  
            └── version.plist  
            └── CodeSignature  
                └── CodeResources  
    └── Current -> B
```

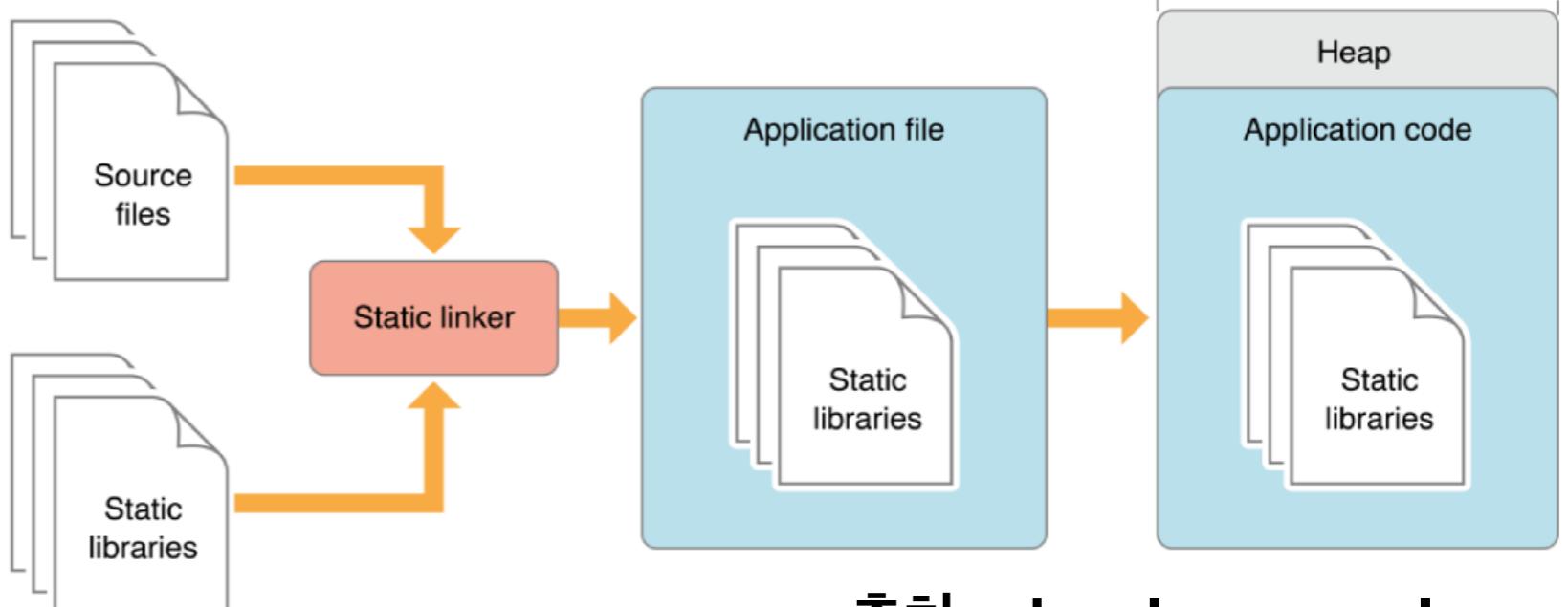
# **Dynamic Framework**

# **Static Framework**

## **Dynamic Framework**

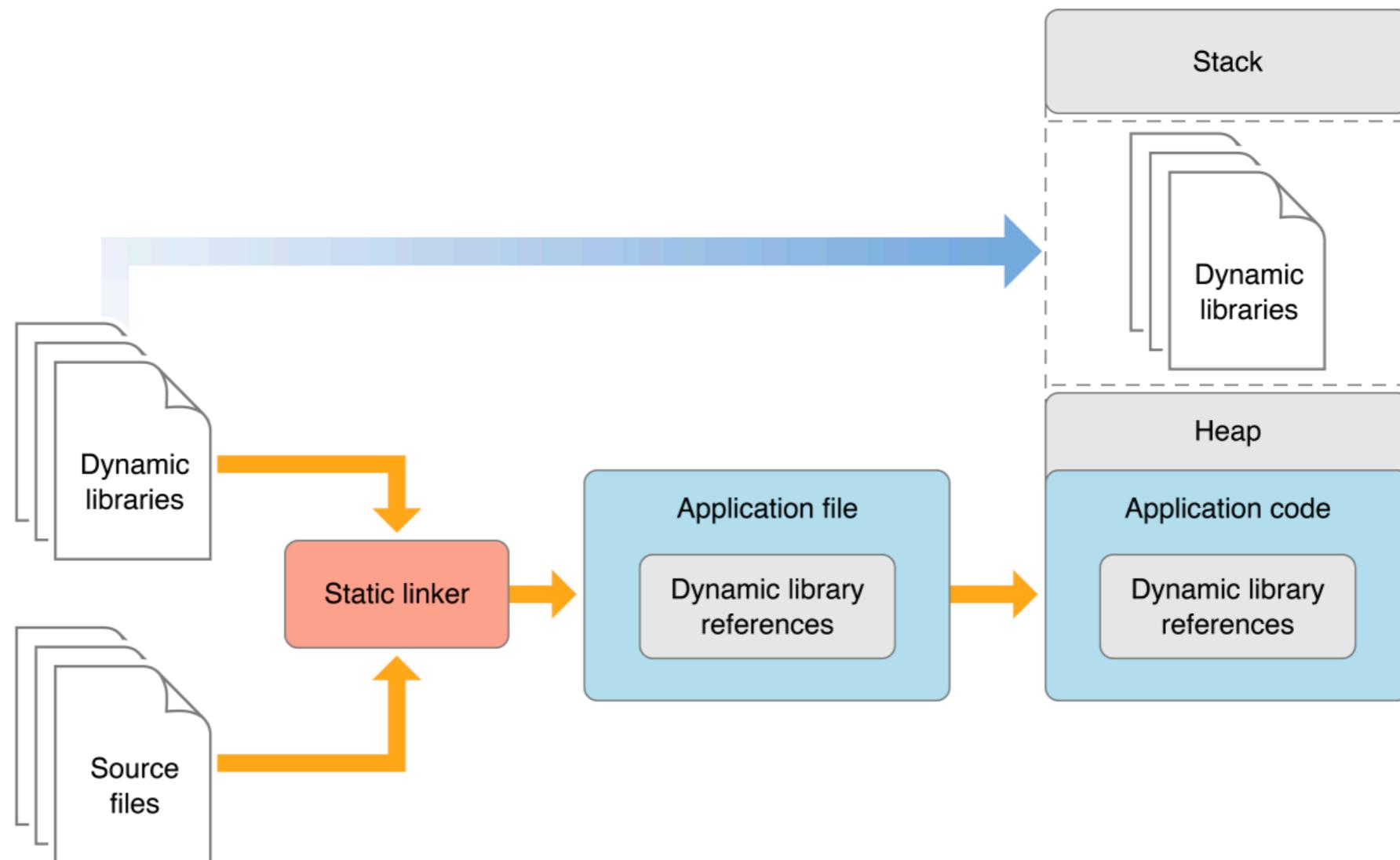


## **Static Framework**



# Dynamic Framework

- Dynamic Framework는 동시에 여러 프레임워크 또는 프로그램에서 동일한 코드 사본을 공유하고 사용
- 동적으로 연결되어 있으므로, 전체 빌드를 다시 하지 않아도 새로운 프레임워크 사용
- Static Linker를 통해 Dynamic Library Reference가 어플리케이션 코드에 들어가고 모듈 호출시 Stack에 있는 Library에 접근하여 사용



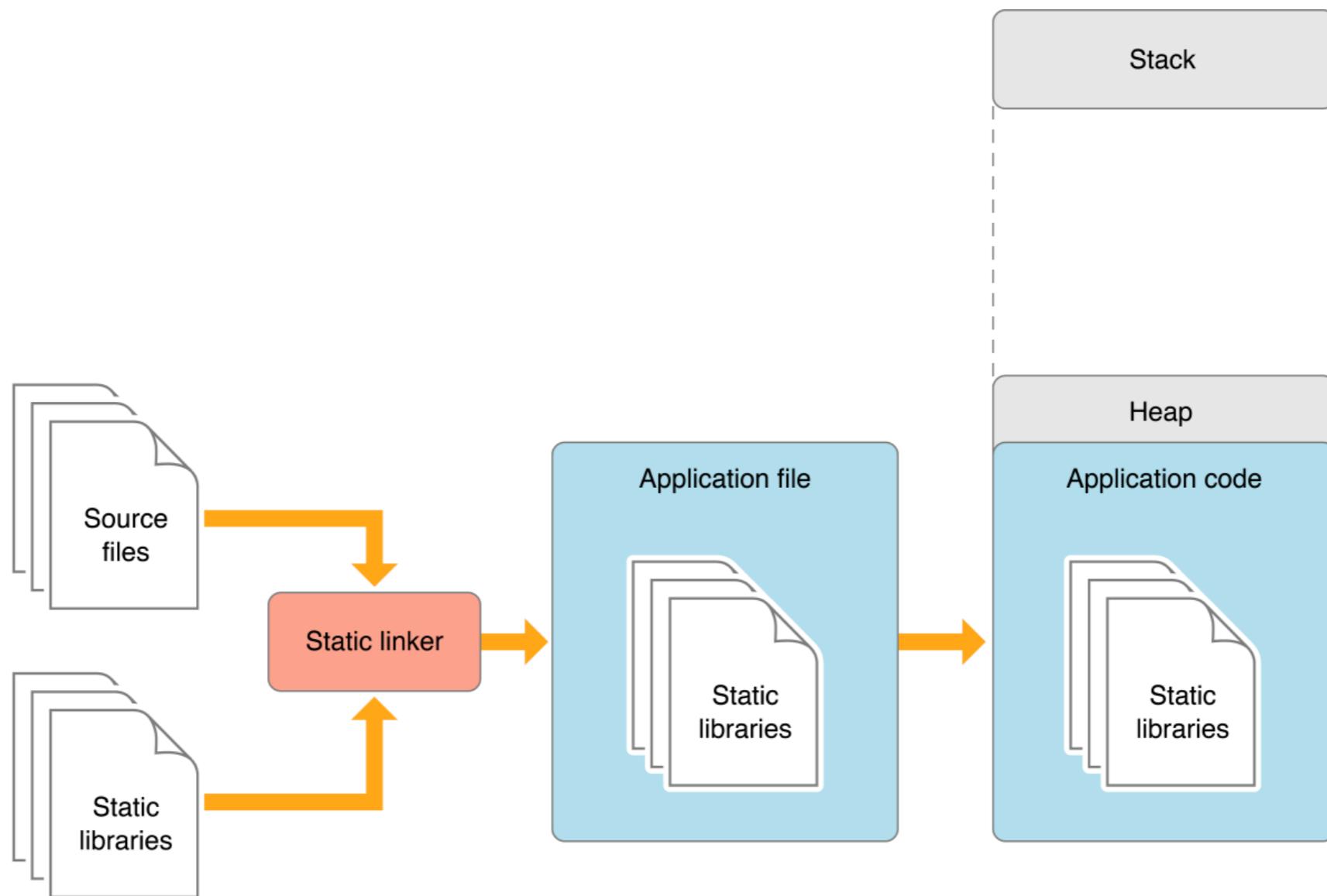
# Static Framework

Static Framework는 Static Linker를 통해 Static Library 코드가 어플리케이션 코드 내로 들어가 Heap 메모리에 상주.

Library는 Framework가 아니라 Static Library가 복사된 곳 위치, 즉, Bundle의 위치는 Static Library가 복사된 위치해 있는 곳

# Static Framework

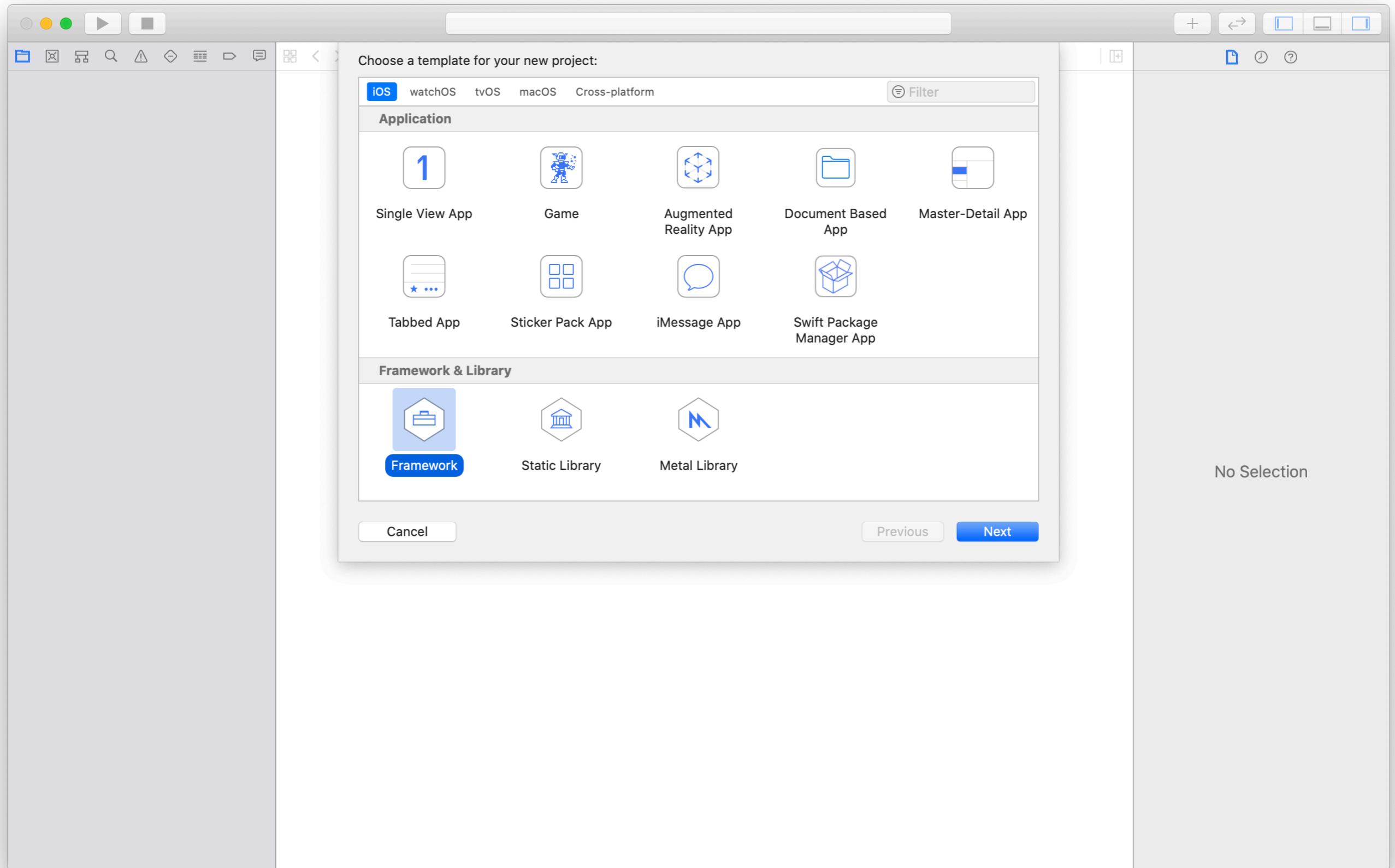
Library는 Framework가 아니라 Static Library가 복사된 곳 위치,  
즉, Bundle의 위치는 Static Library가 복사된 위치해 있는 곳

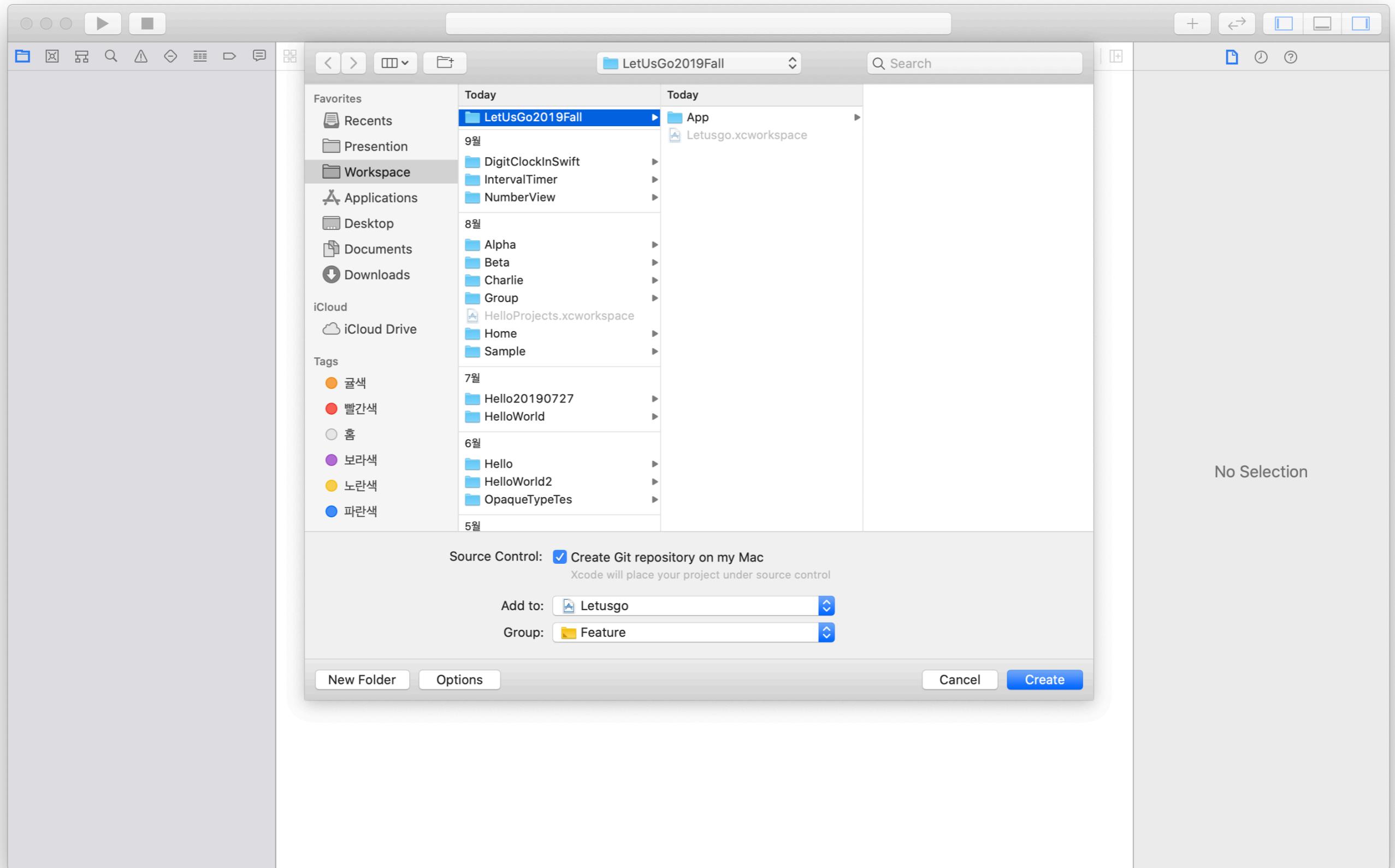


# **Dynamic 또는 Static Framework 만들기**

Project -> Targets -> Build Settings -> Mach-O

Dynamic 또는 Static Framework 설정





Letusgo: Ready | Today at 2:31 PM

Feature

General Signing & Capabilities Resource Tags Info Build Settings Build Phases Build Rules

PROJECT Feature

TARGETS Feature

Display Name Feature

Bundle Identifier kr.minsOne.Labs.Feature

Version 1.0

Build 1

Deployment Info

Target Device

iOS 13.0 ⓘ  iPhone  
 iPad  
 Mac (requires macOS 10.15)

App Extensions  Allow app extension API only

Frameworks and Libraries

Name	Platforms	Embed
Add frameworks and libraries here		

+ -

Development Assets

Add development assets here

+ - Add folders, groups, or individual assets for use during development

Identity and Type

Name Feature

Location Relative to Group

Full Path /Users/minsone/Workspace/LetUsGo2019Fall/Feature/Feature.xcodeproj

Project Document

Project Format Xcode 9.3-compatible

Organization minsone

Class Prefix

Text Settings

Indent Using Spaces

Widths Tab 4 Indent 4

Wrap lines

Letusgo: Ready | Today at 2:32 PM

Feature > Feature > Feature

General Signing & Capabilities Resource Tags Info Build Settings Build Phases Build Rules

**PROJECT**

Basic Customized All Combined Levels + Search

**TARGETS**

Feature

**Architectures**

Setting Feature

Additional SDKs Standard architectures - \$(ARCHS\_STANDARD) iOS

Architectures <Multiple values>

Base SDK iOS

Build Active Architecture Only Yes

Debug No

Release Supported Platforms iOS

Supported Platforms arm64 arm64e armv7 armv7s

Valid Architectures

**Assets**

Setting Feature

Asset Pack Manifest URL Prefix

Embed Asset Packs In Product Bundle No

Enable On Demand Resources No

On Demand Resources Initial Install Tags

On Demand Resources Prefetch Order

**Build Locations**

Setting Feature

Build Products Path build

Intermediate Build Files Path build

Precompiled Headers Cache Path build/SharedPrecompiledHeaders

**Build Options**

Setting Feature

Always Embed Swift Standard Libraries No - \$(EMBEDDED\_CONTENT\_CONTAINS\_SWIFT)

Build Libraries for Distribution No

Build Variants normal

Compiler for C/C++/Objective-C Default compiler (Apple Clang)

Debug Information Format <Multiple values>

Debug DWARF

Release DWARF with dSYM File

Enable Bitcode Yes

Enable Index-While-Building Functionality Default

Enable Previews No

Enable Testability <Multiple values>

**Identity and Type**

Name Feature

Location Relative to Group Feature/Feature.xcodeproj

Full Path /Users/minsone/Workspace/LetUsGo2019Fall/Feature/Feature.xcodeproj

**Project Document**

Project Format Xcode 9.3-compatible

Organization minsone

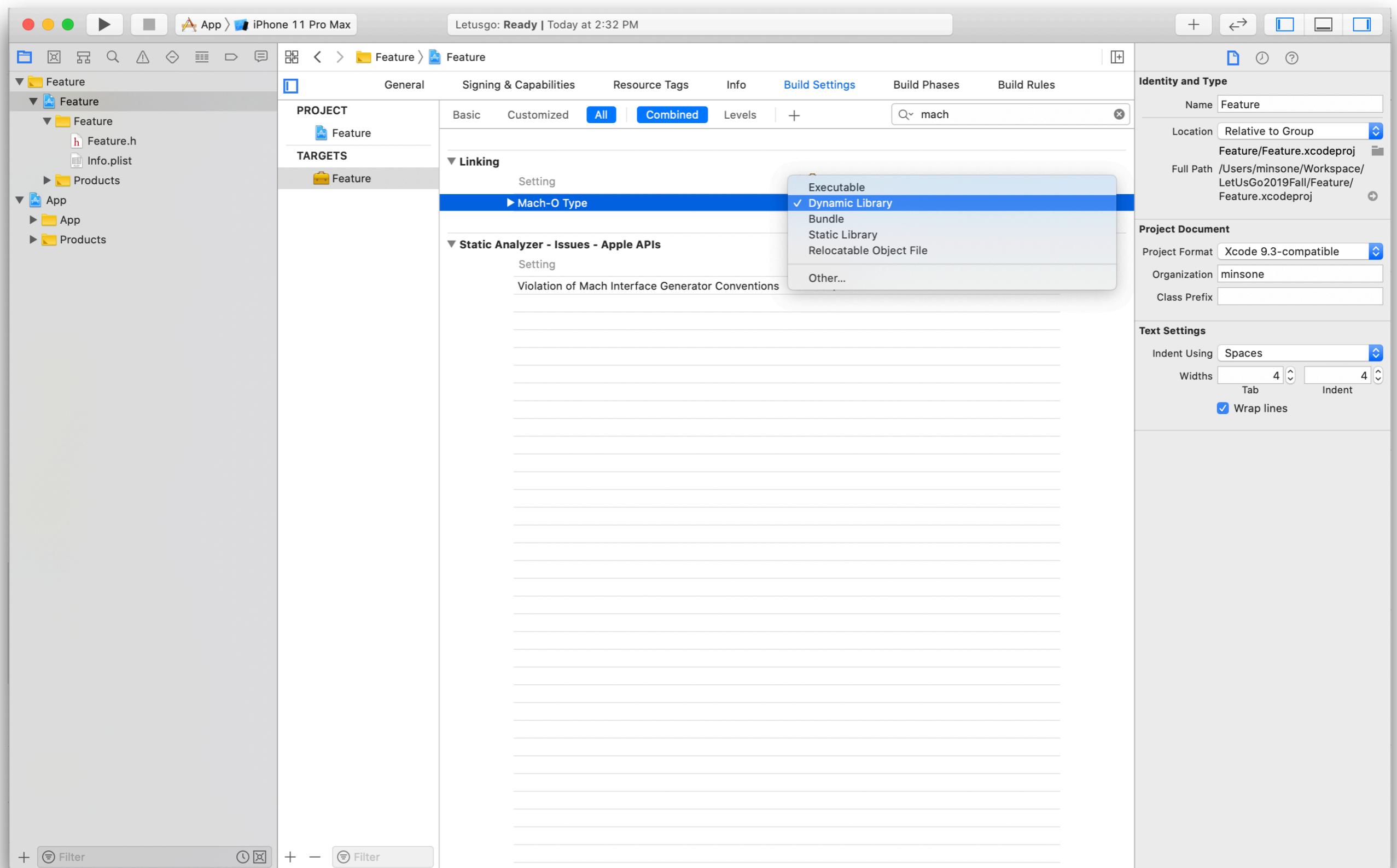
Class Prefix

**Text Settings**

Indent Using Spaces

Widths Tab Indent

Wrap lines



# **Framework Analytics Command**

# Framework Analytics Command(1)

file : 파일의 타입을 확인하는 명령어

## Static Framework

```
[→ ios git:(master) file Firebase.framework/Firebase]
Firebase.framework/Firebase: Mach-O universal binary with 4 architectures: [i386:Mach-O object i386] [x86_64:M
ach-O 64-bit object x86_64] [arm_v7:Mach-O object arm_v7] [arm64:Mach-O 64-bit object arm64]
Firebase.framework/Firebase (for architecture i386):      Mach-O object i386
Firebase.framework/Firebase (for architecture x86_64):    Mach-O 64-bit object x86_64
Firebase.framework/Firebase (for architecture armv7):     Mach-O object arm_v7
Firebase.framework/Firebase (for architecture arm64):     Mach-O 64-bit object arm64
```

## Dynamic Framework

```
[→ ios git:(master) file RxSwift.framework/RxSwift]
RxSwift.framework/RxSwift: Mach-O universal binary with 4 architectures: [i386:Mach-O dynamically linked share
d library i386] [x86_64] [arm_v7] [arm64]
RxSwift.framework/RxSwift (for architecture i386):        Mach-O dynamically linked shared library i386
RxSwift.framework/RxSwift (for architecture x86_64):      Mach-O 64-bit dynamically linked shared library x86_64
RxSwift.framework/RxSwift (for architecture armv7):       Mach-O dynamically linked shared library arm_v7
RxSwift.framework/RxSwift (for architecture arm64):       Mach-O 64-bit dynamically linked shared library arm64
```

# Framework Analytics Command(2)

strings : 파일에서 출력 가능한 문자열을 출력

```
→ iOS git:(master) strings RxSwift.framework/RxSwift
init
lock
unlock
isMainThread
dealloc
currentThread
threadDictionary
setObject:forKeyedSubscript:
hash
copyWithZone:
.cxx_destruct
objectForKeyedSubscript:
blockOperationWithBlock:
setQueuePriority:
addOperation:
stringFromDate:
setDateFormat:
load
setObject:forKey:
```

# Framework Analytics Command(3)

nm : 바이너리 파일을 검사하여 저장된 내용, 메타 정보를 표시

```
[→ ios git:(master) nm RxSwift.framework/RxSwift
0000000000c06dc S _$s10DisposeKey7RxSwift27SynchronizedUnsubscribeTypePT1
    U _$s10Foundation4DateV13distantFutureACvgZ
    U _$s10Foundation4DateV17timeIntervalSinceySdACF
    U _$s10Foundation4DateV18addingTimeIntervalyACSdF
    U _$s10Foundation4DateV19_bridgeToObjectiveCSo6NSDateCyF
000000000047740 T _$s10Foundation4DateV7RxSwiftE22addingDispatchIntervalyAC0F00f4TimeG00F
    U _$s10Foundation4DateV7compareySo18NSComparisonResultVACF
00000000000abc0 t _$s10Foundation4DateV9eventTime_7RxSwift5Event0y7ElementAE120bserverTypePQzG0C0tAeIRz1W0h
00000000000aef0 t _$s10Foundation4DateV9eventTime_7RxSwift5Event0y7ElementAE120bserverTypePQzG0C0tSgAeIRz1W0h
    U _$s10Foundation4DateVACycfc
    U _$s10Foundation4DateVMa
00000000009d680 t _$s10Foundation4DateVSgW0b
00000000000af70 t _$s10Foundation4DateVSgW0c
00000000009d5b0 t _$s10Foundation4DateVSgW0d
00000000000aff0 t _$s10Foundation4DateVSgW0h
00000000000b7228 S _$s12ReactiveBase7RxSwift0A10CompatiblePT1
00000000000b9920 S _$s15VirtualTimeUnit7RxSwift0aB13ConverterTypePT1
    U _$s18IntegerLiteralTypes013ExpressibleByaB0PT1
00000000000b9928 S _$s23VirtualTimeIntervalUnit7RxSwift0aB13ConverterTypePT1
    U _$s5IndexS1T1
0000000000bdb4c S _$s5Trait7RxSwift21PrimitiveSequenceTypePT1
00000000000b8628 S _$s5Value7RxSwift013InvocableWithA4TypePT1
000000000002d1b0 t _$s7Element7RxSwift120bserverTypePQy_SgAbCR_r0_lW0h
0000000000070a40 t _$s7Element7RxSwift120bserverTypePQzSbs5Error_pIegndzo_SgAbCRz1W0e
000000000008ded0 t _$s7Element7RxSwift120bserverTypePQzSgAbCRz1W0b
000000000001f750 t _$s7Element7RxSwift120bserverTypePQzSgAbCRz1W0c
000000000008ddd0 t _$s7Element7RxSwift120bserverTypePQzSgAbCRz1W0d
```

# Framework Analytics Command(4)

lipo : 바이너리에서 아키텍처를 지원하는 실행 파일을 추출 및 합침

```
→ RxSwift.framework git:(master) lipo -detailed_info RxSwift
Fat header in: RxSwift
fat_magic 0xcafebabe
nfat_arch 4
architecture i386
    cputype CPU_TYPE_I386
    cpusubtype CPU_SUBTYPE_I386_ALL
    offset 4096
    size 2779296
    align 2^12 (4096)
architecture x86_64
    cputype CPU_TYPE_X86_64
    cpusubtype CPU_SUBTYPE_X86_64_ALL
    offset 2785280
    size 2969360
    align 2^12 (4096)
architecture armv7
    cputype CPU_TYPE_ARM
    cpusubtype CPU_SUBTYPE_ARM_V7
    offset 5767168
    size 8846520
    align 2^14 (16384)
architecture arm64
    cputype CPU_TYPE_ARM64
```

```
→ RxSwift.framework git:(master) lipo -detailed_info RxSwift
Fat header in: RxSwift
fat_magic 0xcafebabe
nfat_arch 4
architecture i386
    cputype CPU_TYPE_I386
    cpusubtype CPU_SUBTYPE_I386_ALL
    offset 4096
    size 2779296
    align 2^12 (4096)
architecture x86_64
    cputype CPU_TYPE_X86_64
    cpusubtype CPU_SUBTYPE_X86_64_ALL
    offset 2785280
    size 2969360
    align 2^12 (4096)
architecture armv7
    cputype CPU_TYPE_ARM
    cpusubtype CPU_SUBTYPE_ARM_V7
    offset 5767168
    size 8846520
    align 2^14 (16384)
architecture arm64
    cputype CPU_TYPE_ARM64
    cpusubtype CPU_SUBTYPE_ARM64_ALL
    offset 14614528
    size 9315828
    align 2^14 (16384)
```

## Library를 Architecture 별로 추출

```
[→ RxSwift.framework git:(master) lipo RxSwift -thin i386 -output RxSwift_i386  
[→ RxSwift.framework git:(master) lipo RxSwift -thin x86_64 -output RxSwift_x86_64  
[→ RxSwift.framework git:(master) lipo RxSwift -thin armv7 -output RxSwift_armv7  
[→ RxSwift.framework git:(master) lipo RxSwift -thin arm64 -output RxSwift_arm64  
[→ RxSwift.framework git:(master) file RxSwift_i386 RxSwift_x86_64 RxSwift_armv7 RxSwift_arm64
```

RxSwift\_i386: Mach-O dynamically linked shared library i386

RxSwift\_x86\_64: Mach-O 64-bit dynamically linked shared library x86\_64

RxSwift\_armv7: Mach-O dynamically linked shared library arm\_v7

RxSwift\_arm64: Mach-O 64-bit dynamically linked shared library arm64

## Architecture 별로 나눠진 Library를 합침

```
[→ RxSwift.framework git:(master) lipo -create RxSwift_i386 RxSwift_x86_64 RxSwift_armv7 RxSwift_arm64  
-output RxSwift_Universal
```

```
[→ RxSwift.framework git:(master) file RxSwift_Universal ]
```

RxSwift\_Universal: Mach-O universal binary with 4 architectures: [i386:Mach-O dynamically linked shared library i386] [x86\_64] [arm\_v7] [arm64]

RxSwift\_Universal (for architecture i386): Mach-O dynamically linked shared library i386

RxSwift\_Universal (for architecture x86\_64): Mach-O 64-bit dynamically linked shared library x86\_64

RxSwift\_Universal (for architecture armv7): Mach-O dynamically linked shared library arm\_v7

RxSwift\_Universal (for architecture arm64): Mach-O 64-bit dynamically linked shared library arm64

```
[→ RxSwift.framework git:(master) file RxSwift ]
```

RxSwift: Mach-O universal binary with 4 architectures: [i386:Mach-O dynamically linked shared library i386] [x86\_64] [arm\_v7] [arm64]

RxSwift (for architecture i386): Mach-O dynamically linked shared library i386

RxSwift (for architecture x86\_64): Mach-O 64-bit dynamically linked shared library x86\_64

RxSwift (for architecture armv7): Mach-O dynamically linked shared library arm\_v7

RxSwift (for architecture arm64): Mach-O 64-bit dynamically linked shared library arm64

# Framework Analytics Command(5)

otool: Object 파일 표시 툴, 각종 정보를 보여줌.

## Dynamic Framework

```
[→ RxCocoa.framework git:(master) otool -L RxCocoa
RxCocoa:
@rpath/RxCocoa.framework/RxCocoa (compatibility version 1.0.0, current version 1.0.0)
/System/Library/Frameworks/Foundation.framework/Foundation (compatibility version 300.0.0, current version 1665.15.0)
/usr/lib/libobjc.A.dylib (compatibility version 1.0.0, current version 228.0.0)
/usr/lib/libSystem.B.dylib (compatibility version 1.0.0, current version 1281.0.0)
/System/Library/Frameworks/CoreFoundation.framework/CoreFoundation (compatibility version 150.0.0, current version 1665.15.0)
/System/Library/Frameworks/QuartzCore.framework/QuartzCore (compatibility version 1.2.0, current version 1.11.0)
/System/Library/Frameworks/UIKit.framework/UIKit (compatibility version 1.0.0, current version 61000.0.0)
@rpath/RxRelay.framework/RxRelay (compatibility version 1.0.0, current version 1.0.0)
@rpath/RxSwift.framework/RxSwift (compatibility version 1.0.0, current version 1.0.0)
@rpath/libswiftCore.dylib (compatibility version 1.0.0, current version 1100.2.255)
@rpath/libswiftCoreGraphics.dylib (compatibility version 1.0.0, current version 0.0.0)
@rpath/libswiftDispatch.dylib (compatibility version 1.0.0, current version 0.0.0)
@rpath/libswiftFoundation.dylib (compatibility version 1.0.0, current version 0.0.0)
@rpath/libswiftObjectiveC.dylib (compatibility version 1.0.0, current version 0.0.0)
@rpath/libswiftUIKit.dylib (compatibility version 1.0.0, current version 0.0.0)
```

# Dynamic Framework

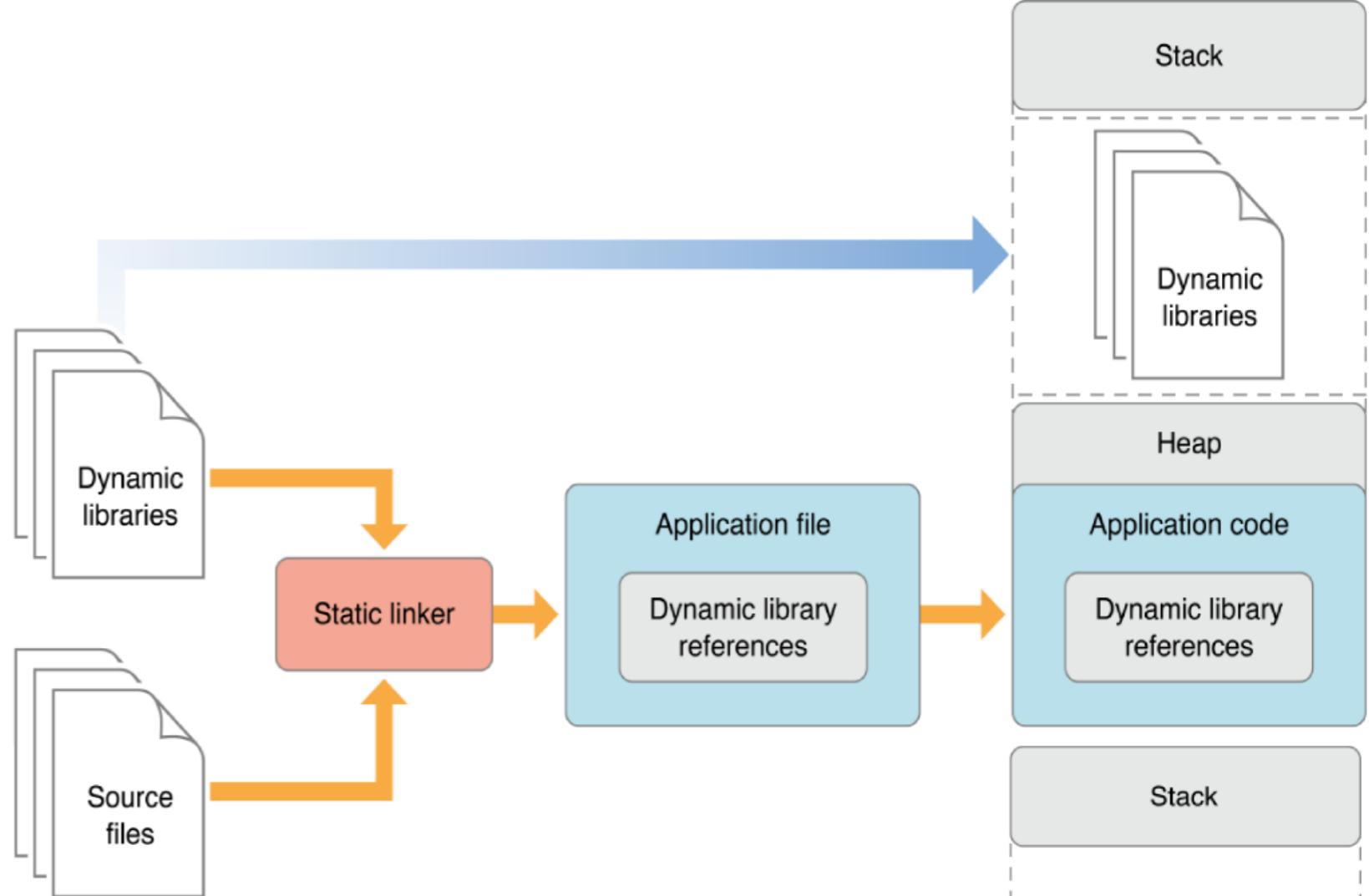
```
[→ RxCocoa.framework git:(master) otool -L RxCocoa
RxCocoa:
    @rpath/RxCocoa.framework/RxCocoa (compatibility version 1.0.0, current version 1.0.0)
    /System/Library/Frameworks/Foundation.framework/Foundation (compatibility version 300.0.0, current version 1665.15.0)
    /usr/lib/libobjc.A.dylib (compatibility version 1.0.0, current version 228.0.0)
    /usr/lib/libSystem.B.dylib (compatibility version 1.0.0, current version 1281.0.0)
    /System/Library/Frameworks/CoreFoundation.framework/CoreFoundation (compatibility version 150.0.0, current version 1665.15.0)
    /System/Library/Frameworks/QuartzCore.framework/QuartzCore (compatibility version 1.2.0, current version 1.11.0)
    /System/Library/Frameworks/UIKit.framework/UIKit (compatibility version 1.0.0, current version 61000.0.0)
    @rpath/RxRelay.framework/RxRelay (compatibility version 1.0.0, current version 1.0.0)
    @rpath/RxSwift.framework/RxSwift (compatibility version 1.0.0, current version 1.0.0)
    @rpath/libswiftCore.dylib (compatibility version 1.0.0, current version 1100.2.255)
    @rpath/libswiftCoreGraphics.dylib (compatibility version 1.0.0, current version 0.0.0)
    @rpath/libswiftDispatch.dylib (compatibility version 1.0.0, current version 0.0.0)
    @rpath/libswiftFoundation.dylib (compatibility version 1.0.0, current version 0.0.0)
    @rpath/libswiftObjectiveC.dylib (compatibility version 1.0.0, current version 0.0.0)
    @rpath/libswiftUIKit.dylib (compatibility version 1.0.0, current version 0.0.0)
```

# Static Framework

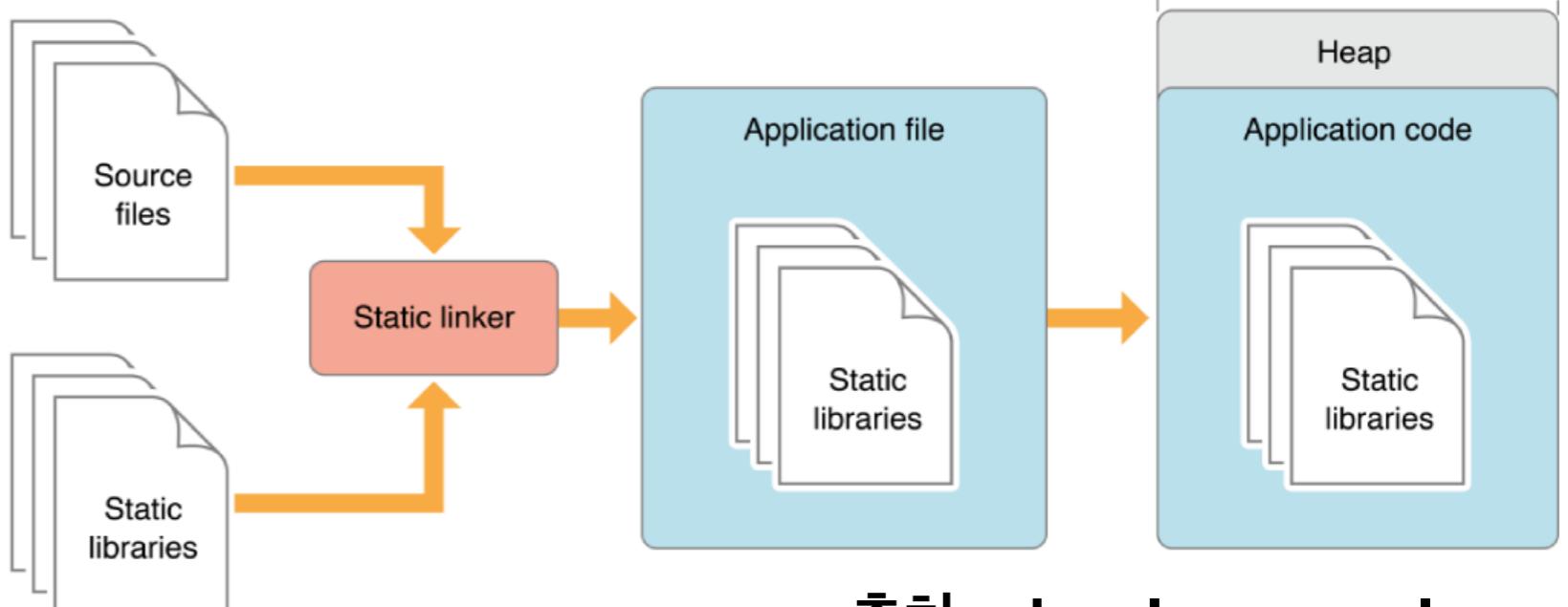
```
[→ ios git:(master) ✘ otool -L FirebaseAnalytics.framework/FirebaseAnalytics
Archive : FirebaseAnalytics.framework/FirebaseAnalytics
FirebaseAnalytics.framework/FirebaseAnalytics(FIRAIIdentifiers_16c07d77c69044b43f7b9de70da25f9f.o):
FirebaseAnalytics.framework/FirebaseAnalytics(FIRAClusiveUserProperty_991dd1dfefbe571de6dbf327969af9a0.o):
FirebaseAnalytics.framework/FirebaseAnalytics(FIRAClusiveUserPropertyController_faf82aa1c60b80614f617fe3e96f08fe.o):
FirebaseAnalytics.framework/FirebaseAnalytics(FIRAEEvent_c32f39c4e1681f460e696def59992d96.o):
FirebaseAnalytics.framework/FirebaseAnalytics(FIRAUUserAttribute_66224d31710f67eef7b8c0e9f1c05115.o):
FirebaseAnalytics.framework/FirebaseAnalytics(FIRAValue_2e0ad7f420162fa95ff8e12af195aa2b.o):
FirebaseAnalytics.framework/FirebaseAnalytics(FIRAAAdExposureReporter_9f9da43a0e4ed4258a1c1de8d08084f3.o):
FirebaseAnalytics.framework/FirebaseAnalytics(FIRAConstants_ff4f35cf6d81297e28401868f46f7b4f.o):
FirebaseAnalytics.framework/FirebaseAnalytics(FIRAMeasurement_7113499ff6c634053dac5bf06844d505.o):
FirebaseAnalytics.framework/FirebaseAnalytics(FIRASessionReporter_fedb19668cf4c09fe19f4dd3dd423569.o):
FirebaseAnalytics.framework/FirebaseAnalytics(FIRAnalytics_ca9cdf99cfbae2f51d040b73dd242e94.o):
FirebaseAnalytics.framework/FirebaseAnalytics(FIRAScreenViewReporter_ded399434f21648531c172048caa1206.o):
```

# Encapsulation

## **Dynamic Framework**



## **Static Framework**



# **3rd Party Static Framework Encapsulation**

Firebase, Crashlytics 등의 Framework는  
우리가 만든 Dynamic Framework로 감싸 이용.

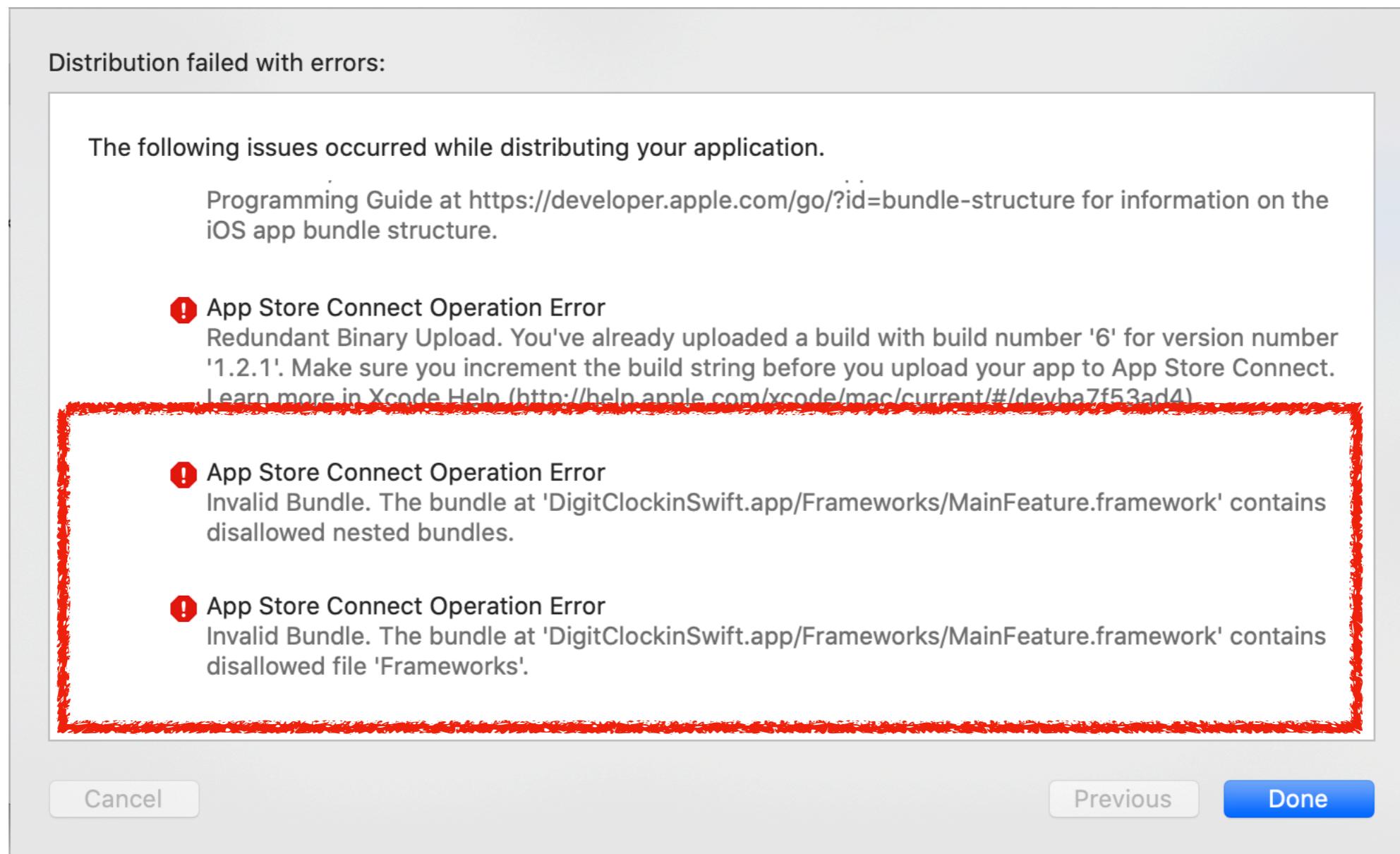
# **3rd Party Dynamic Framework Encapsulation**

Dynamic Framework를 Linking하면 Library Reference만 알기 때문에 어플리케이션이 3rd Party Dynamic Framework를 가지고 있어야 함.

만약 Framework 안에 Framework가 있는 경우, AppStore에 바이너리 제출이 안됨.

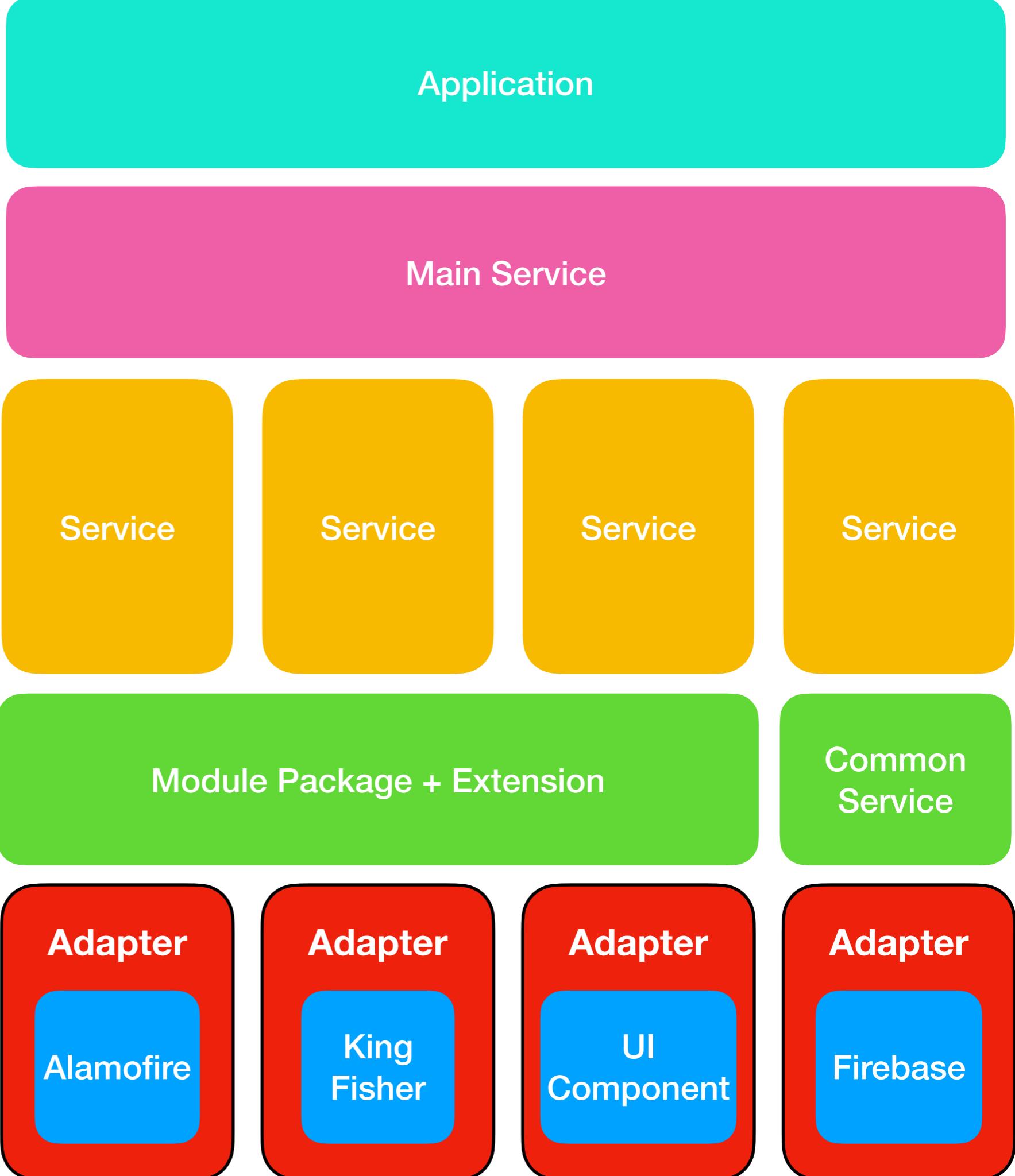
# 3rd Party Dynamic Framework Encapsulation

만약 Framework 안에 Framework가 있는 경우, AppStore에 바이너리 제출이 안됨.



# let us: Go! 2019 Spring

## A Framework Driven Development



# 정리

---

- Dynamic과 Static을 구분하여 잘 사용하자
- 깨끗한 코드를 개발하기 위해선 패키지를 이용하여  
격리화
- 잘 설계된 프로젝트 구조가 미래의 나를 구원함.

# QnA

# Reference

---

- [Apple : Framework Programming Guide](#)
- [Apple : Mach-O Programming Topics](#)
- [Apple : Dynamic Library Programming Topics](#)
- [Google : iOS Static Framework](#)
- [RUNTASTIC : BASIC OVERVIEW OF STATIC AND DYNAMIC FRAMEWORKS ON IOS](#)
- [Github : grap/cocoapods-pod-merge](#)
- [Github : markjarecki/ModularFlowArchitecture](#)
- [Building a dynamic modular iOS architecture](#)
- [iOS Frameworks Part 1: A Treacherous Voyage Through Murky Waters](#)
- [Optimize your iOS projects creating binaries Frameworks](#)
- [Big Nerd Ranch - It Looks Like You Are Trying to Use a Framework](#)
- [iOS에서 프레임워크와 라이브러리의 차이점을 알아보자!](#)
- [Creating and Distributing an iOS Binary Framework](#)
- [Resource Bundles & Static Library in iOS](#)
- [Deep dive into Swift frameworks](#)
- [Fat Framework 만들기](#)