

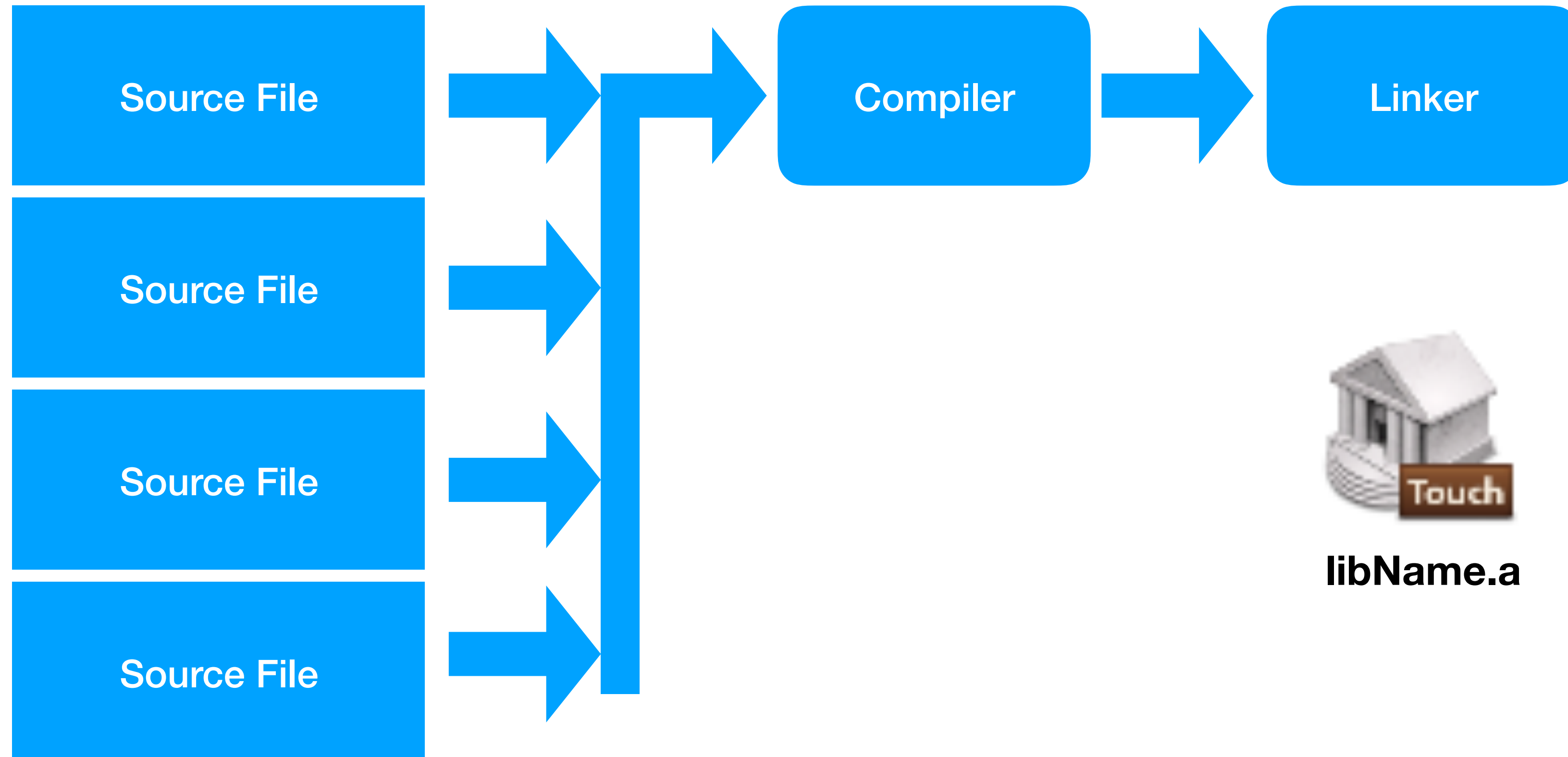
# Building Your Own Swift Frameworks

*Chris Woodard, Tampa Bay Cocoaheads*

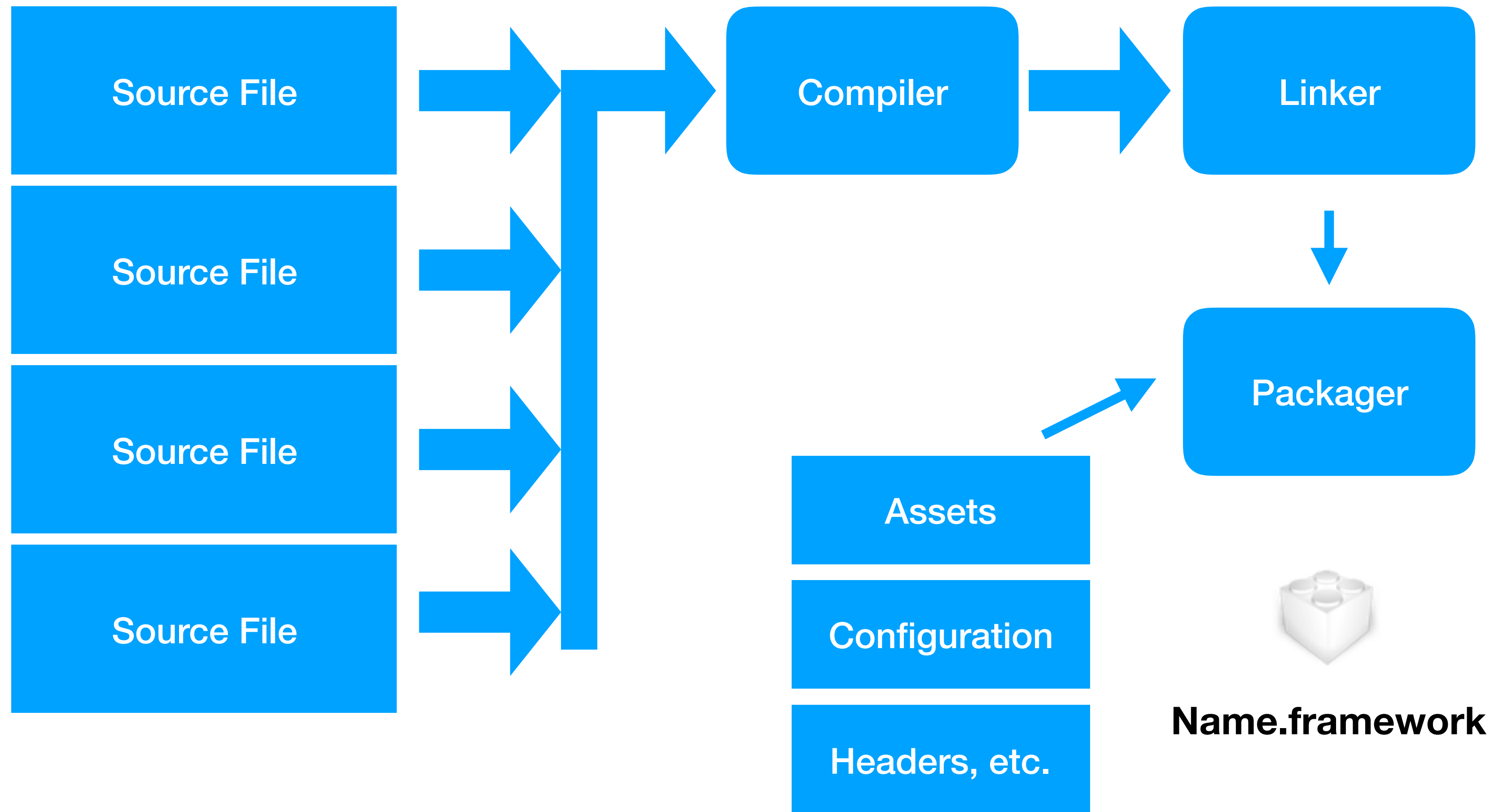
# Reusable Code

- Adding source code directly into your project
  - Updating, naming collisions
- Cocoapods / Alcatraz
  - Needing to update either one; may need to write your own pods
- Building a static library
- Building a framework

# Static Library



# Framework

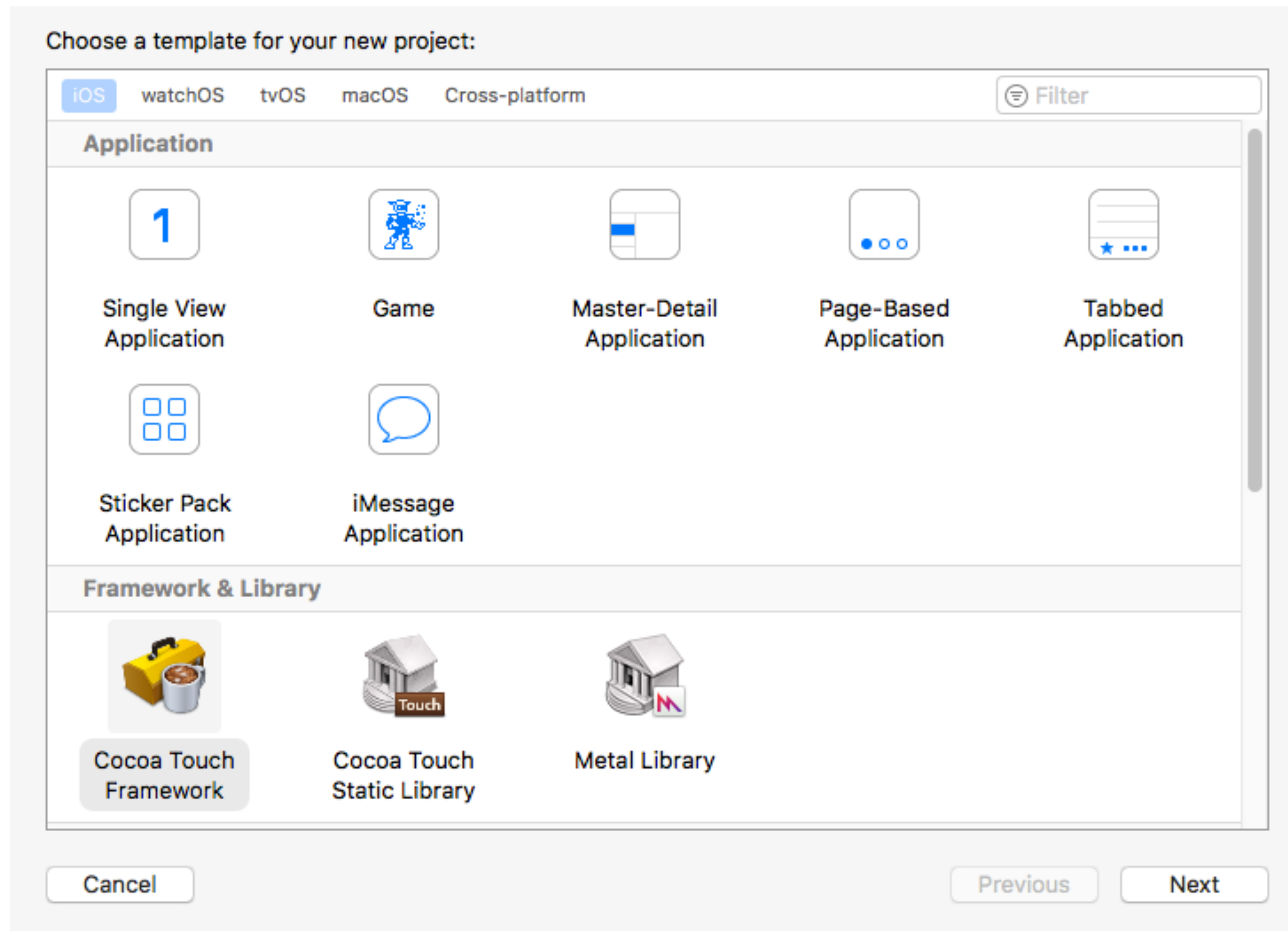




# Framework Anatomy

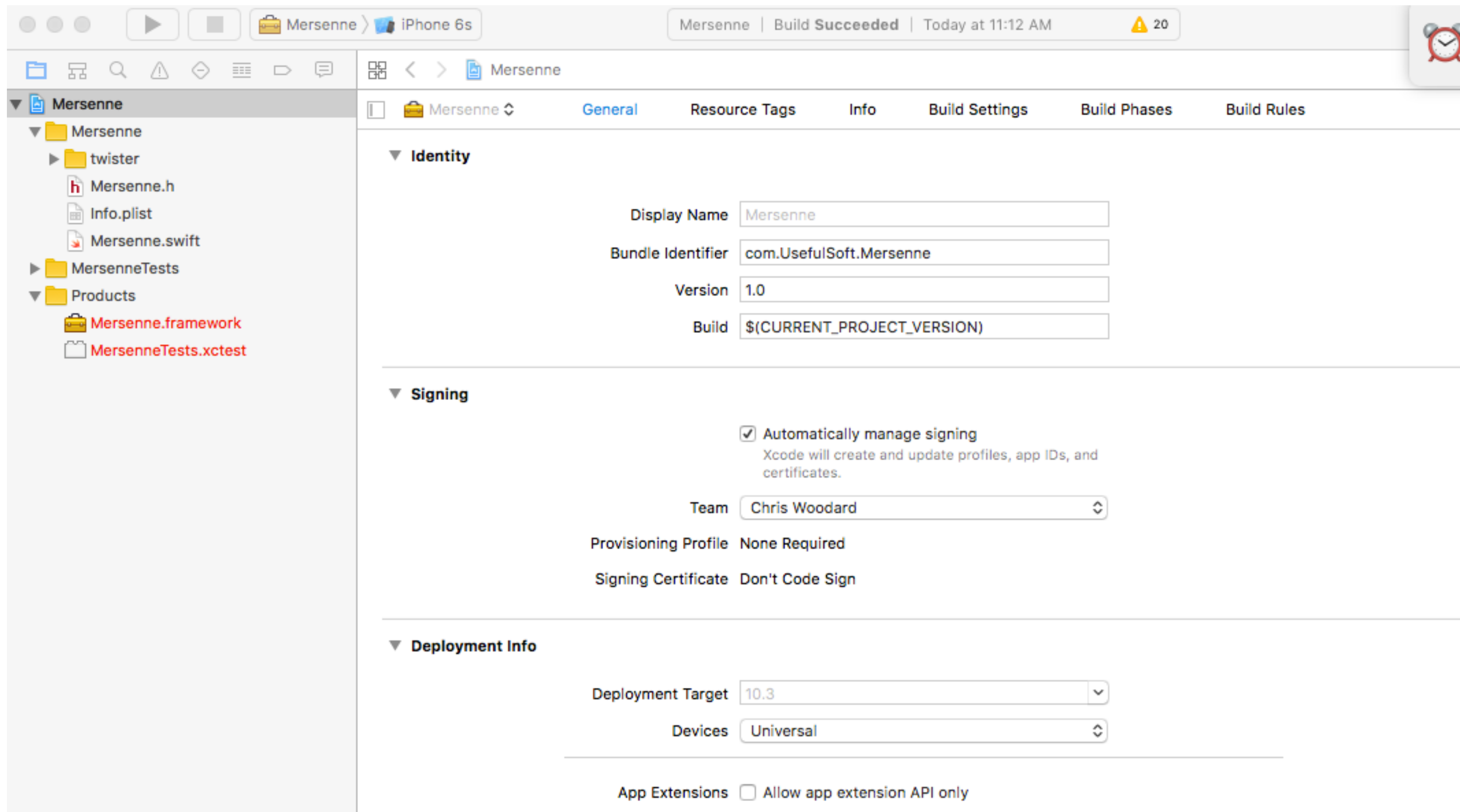
▼ Mersenne.framework	Today, 12:30 PM	--	Folder
▼ _CodeSignature	Today, 11:12 AM	--	Folder
CodeResources	Today, 11:12 AM	7 KB	TextEd...ument
▼ Frameworks	Today, 11:12 AM	--	Folder
libswiftCore.dylib	Today, 11:12 AM	10.8 MB	Dynam...Library
libswiftCoreGraphics.dylib	Today, 11:12 AM	309 KB	Dynam...Library
libswiftCoreImage.dylib	Today, 11:12 AM	77 KB	Dynam...Library
libswiftDarwin.dylib	Today, 11:12 AM	153 KB	Dynam...Library
libswiftDispatch.dylib	Today, 11:12 AM	693 KB	Dynam...Library
libswiftFoundation.dylib	Today, 11:12 AM	4.1 MB	Dynam...Library
libswiftObjectiveC.dylib	Today, 11:12 AM	119 KB	Dynam...Library
libswiftQuartzCore.dylib	Today, 11:12 AM	93 KB	Dynam...Library
libswiftSwiftOnoneSupport.dylib	Today, 11:12 AM	1.9 MB	Dynam...Library
libswiftUIKit.dylib	Today, 11:12 AM	198 KB	Dynam...Library
▼ Headers	Today, 11:12 AM	--	Folder
Mersenne-Swift.h	Today, 11:12 AM	5 KB	C Hea...Source
Mersenne.h	Today, 11:12 AM	520 bytes	C Hea...Source
▼ Modules	Today, 12:30 PM	--	Folder
▼ Mersenne.swiftmodule	Today, 11:12 AM	--	Folder
x86_64.swiftdoc	Today, 11:12 AM	348 bytes	Document
x86_64.swiftmodule	Today, 11:12 AM	13 KB	Document
module.modulemap	Today, 11:12 AM	154 bytes	LLVM...ule Map
libswiftRemoteMirror.dylib	May 11, 2017, 1:06 AM	724 KB	Dynam...Library
Info.plist	Today, 11:12 AM	743 bytes	Property List
Mersenne	Today, 11:12 AM	33 KB	Unix e...cutable

# Framework Mechanics





# Framework Mechanics



# Framework Mechanics

```
import Foundation
import twister

public class Mersenne {

    static var rn:Mersenne? = nil
    public static func shared() -> Mersenne? {
        if nil == rn {
            rn = Mersenne()
            init_genrand(808912898372948)
        }
        return rn
    }

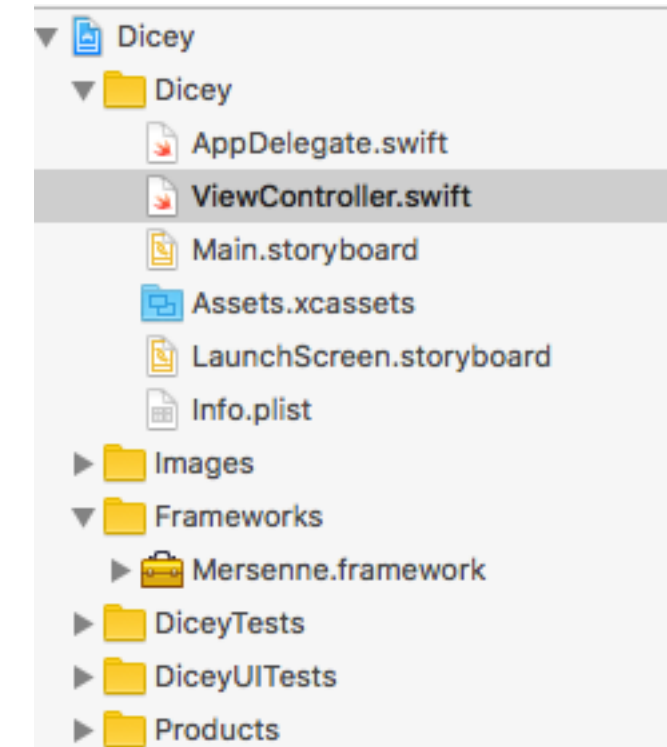
    public func roll(shakes:Int) -> [UInt] {

        for _ in 0...shakes {
            genrand_int32()
        }

        let die1 = 1 + genrand_int32()%6
        let die2 = 1 + genrand_int32()%6

        return [die1, die2]
    }
}
```

Framework code



```
import UIKit
import Mersenne

class ViewController: UIViewController {

    @IBAction func roll(_ sender: Any) {
        let m = Mersenne.shared()
        guard let rolled = m?.roll(shakes: 50)
        else {
            return
        }

        let left = rolled[0]
        let right = rolled[1]

        leftDice.image = UIImage(named: "\(left).png")
        rightDice.image = UIImage(named: "\(right).png")
    }

    @IBOutlet weak var leftDice: UIImageView!
    @IBOutlet weak var rightDice: UIImageView!

    override func viewDidLoad() {
        super.viewDidLoad()
        // Do any additional setup after loading the view,
        from a nib.
    }

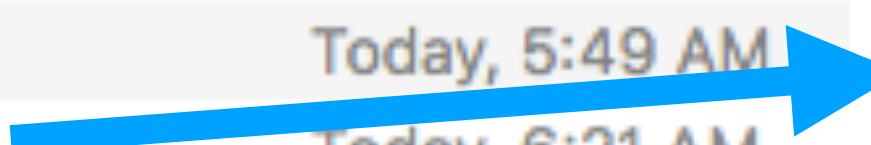
    override func didReceiveMemoryWarning() {
        super.didReceiveMemoryWarning()
        // Dispose of any resources that can be recreated.
    }
}
```

App code that uses the framework



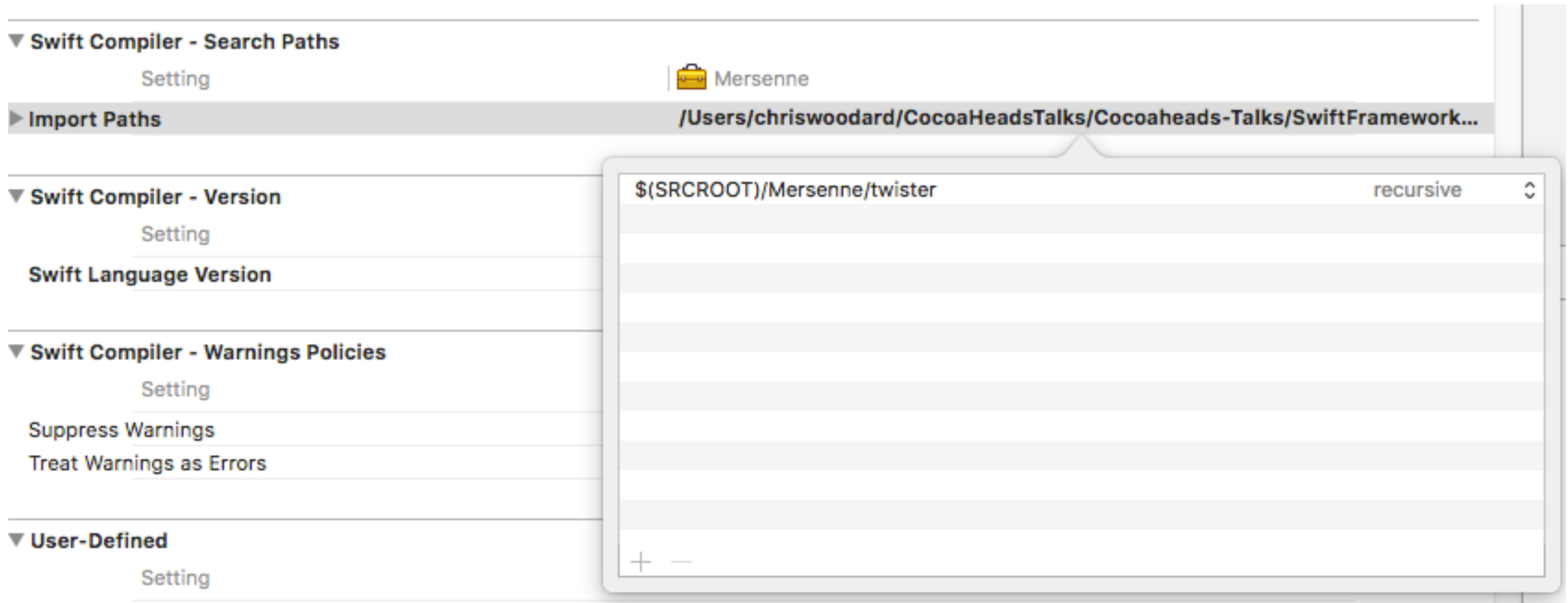
# Using C Source Code

▼	Folder	Mersenne	Today, 5:46 AM
▼	Folder	Mersenne	Today, 7:05 AM
▼	Folder	twister	Today, 6:21 AM
	Header File	Twister.h	Today, 5:36 AM
	Source File	Twister.c	Today, 5:49 AM
	File	module.modulemap	Today, 6:21 AM
	Header File	Mersenne.h	Today, 5:49 AM
	File	Info.plist	Today, 5:31 AM
	File	Mersenne.swift	Today, 7:05 AM
▶	Folder	MersenneTests	Today, 5:31 AM
	File	Mersenne.xcodeproj	Today, 7:03 AM



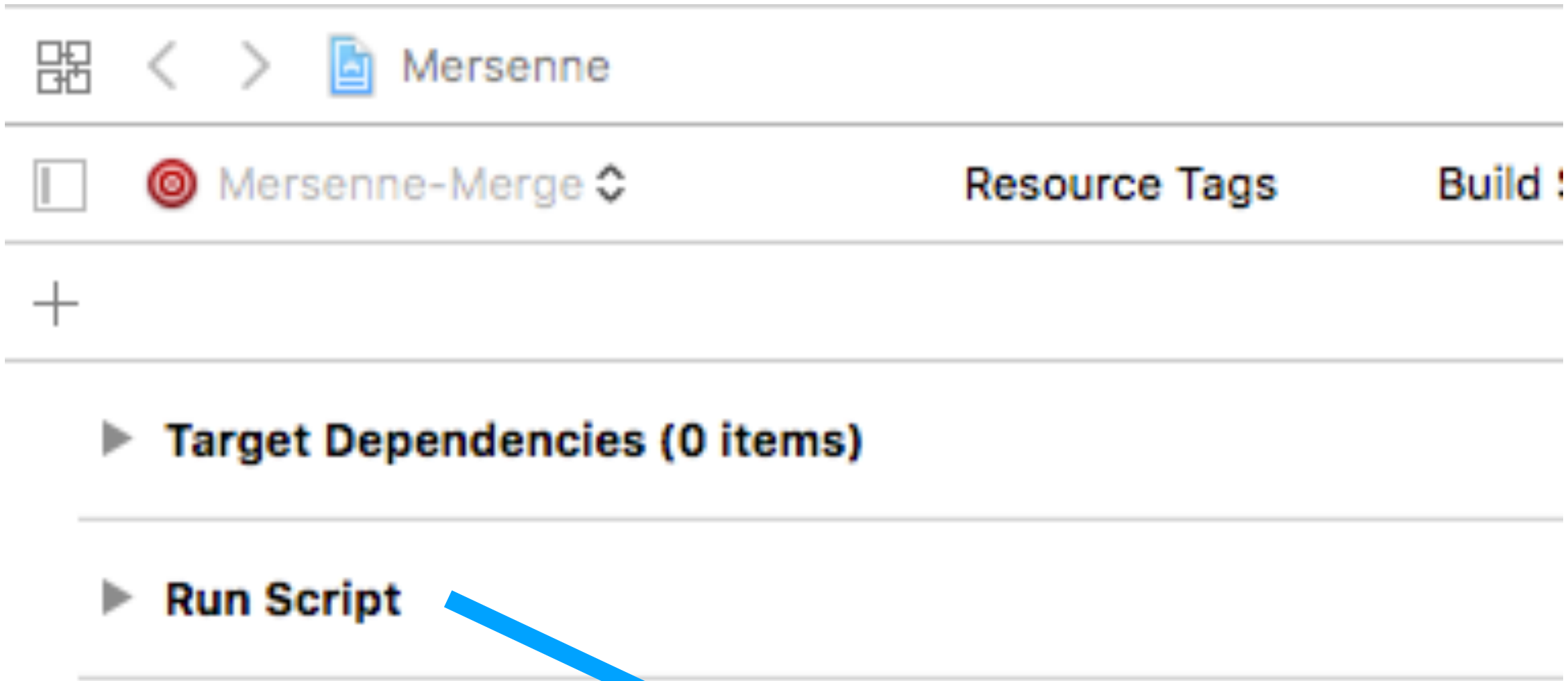
```
module twister [system] [extern_c] {  
    header "Twister.h"  
    export *  
}
```

# Using C Source Code





# Building for Simulator and Device



## # 7

```
# Replace the framework executable within the framework with
# a new version created by merging the device and simulator
# frameworks' executables with lipo.
```

```
lipo -create -output "${HOME}/Desktop/${FRAMEWORK_NAME}.framework/${FRAMEWORK_NAME}" \
"${SRCROOT}/build/Debug-iphonios/${FRAMEWORK_NAME}.framework/${FRAMEWORK_NAME}" \
"${SRCROOT}/build/Debug-iphonesimulator/${FRAMEWORK_NAME}.framework/${FRAMEWORK_NAME}"
```

**Demo**

# Resources

- <https://medium.com/swift-and-ios-writing/using-a-c-library-inside-a-swift-framework-d041d7b701d9>
- <https://www.raywenderlich.com/126365/ios-frameworks-tutorial>
- <http://arsenkin.com/ios-universal-framework.html>