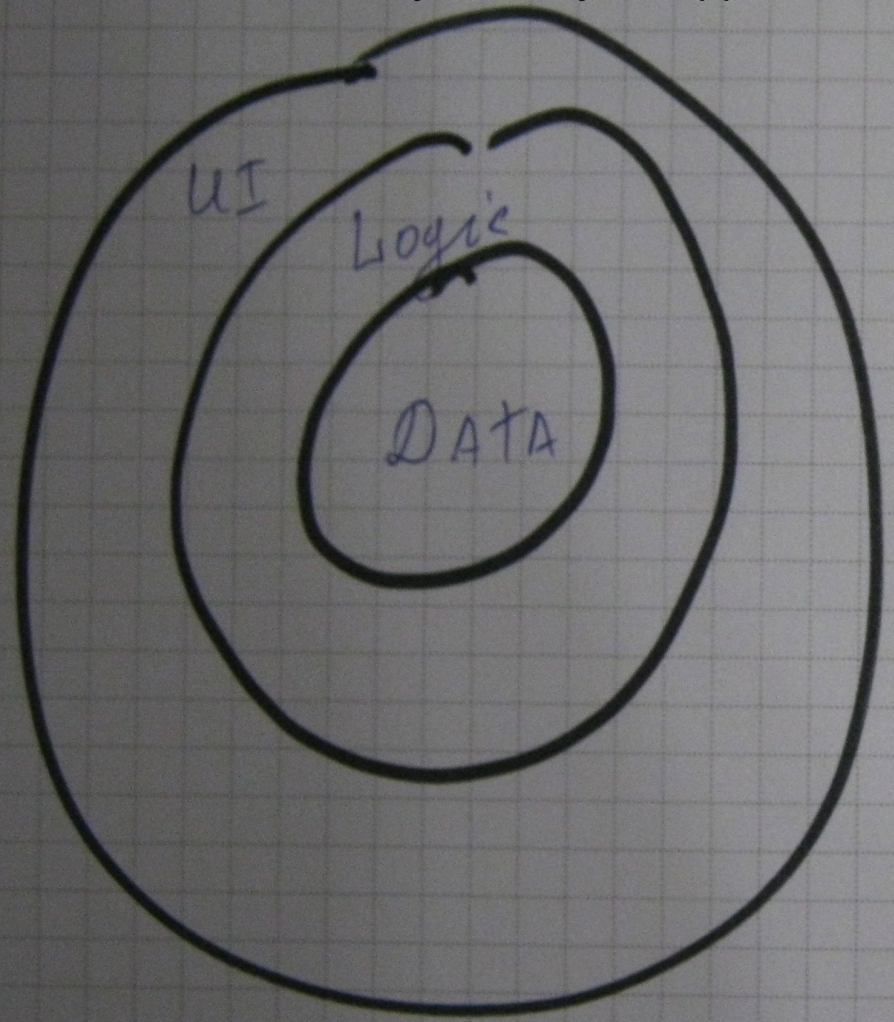
Building libraries for iOS
Going native

Alexander Dodatko
2014

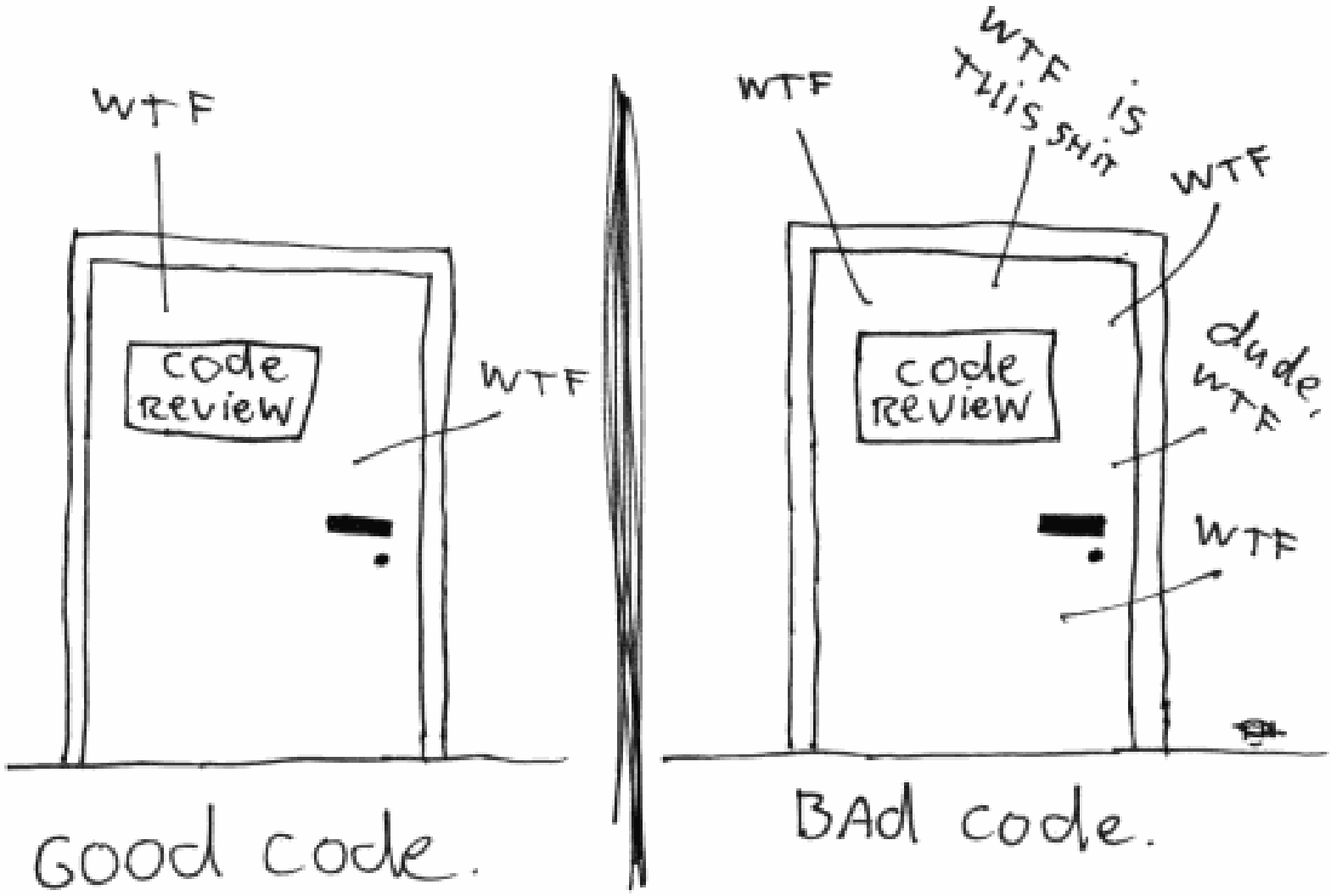
QR Code here

How you see your app



How the user sees your app

The ONLY VALID MEASUREMENT OF Code QUALITY: WTFs/MINUTE



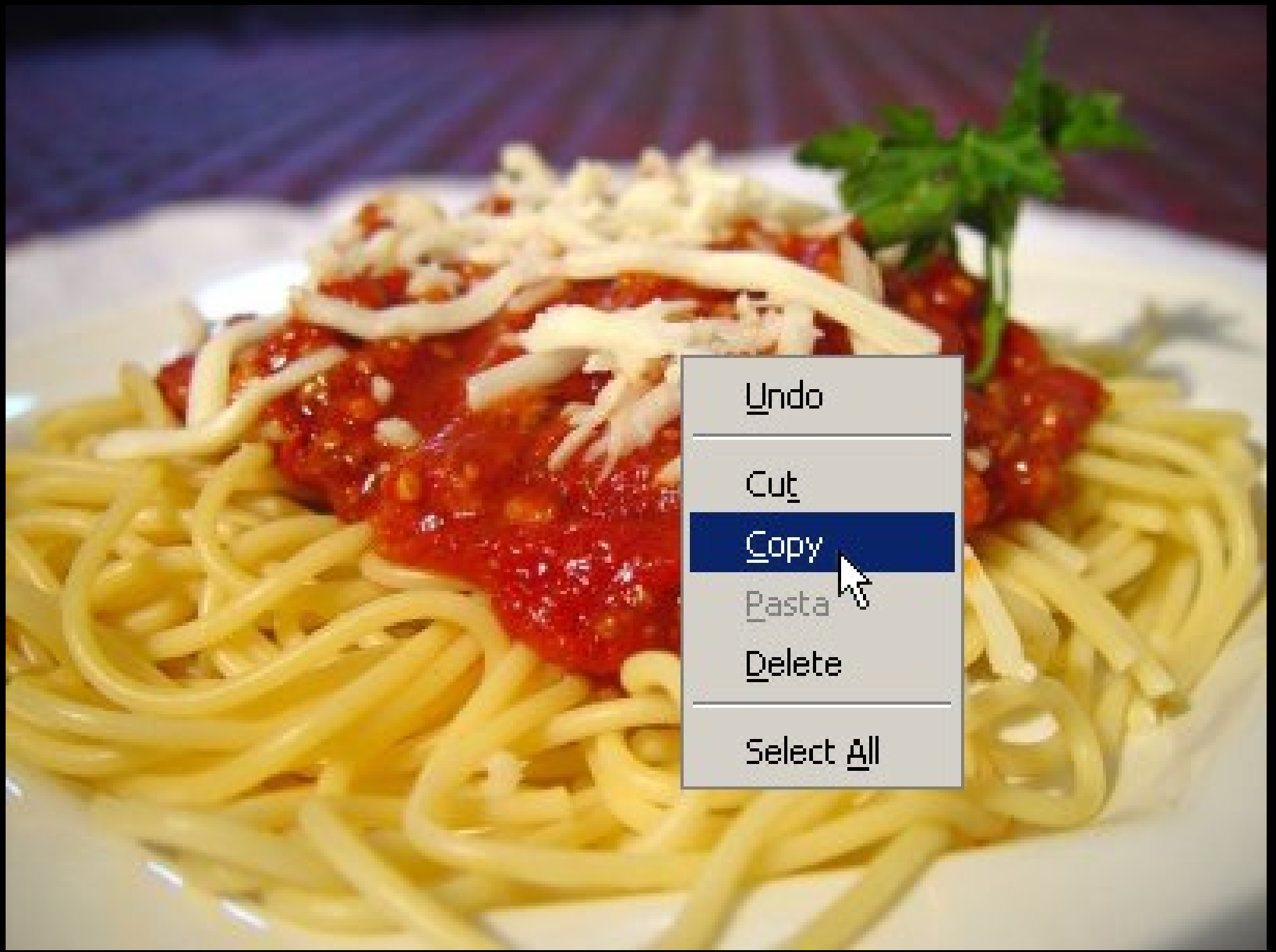
Code Reuse is Important


ASCENT
OF BOT

March 2008

http://engt.co/1pOLKH9

The fastest code is the code that reaches the market first


Undo
Cut
Copy
Pasta
Delete
Select All

Nobody Cares

August 2011

CocoaPods
@CocoaPods

Follow

@oliverfoggin Thanks! Just over one year old:
github.com/CocoaPods/Coco…

Reply Retweet Favorite ••• More

GitHub

CocoaPods

An Objective-C library dependency manager. Contribute to CocoaPods development by creating an account on GitHub.

View on web

4:36 AM - 2 Oct 2012

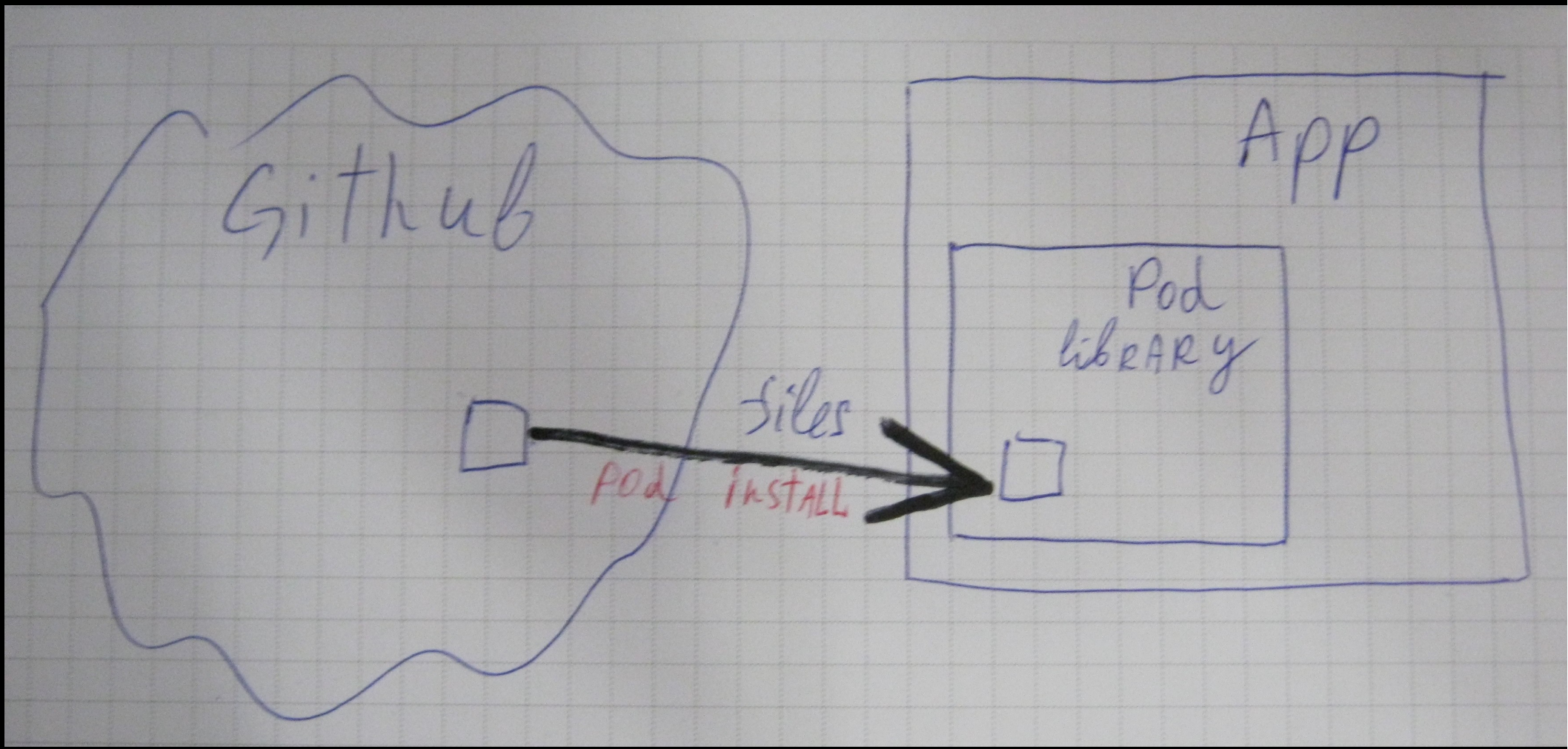
Flag media

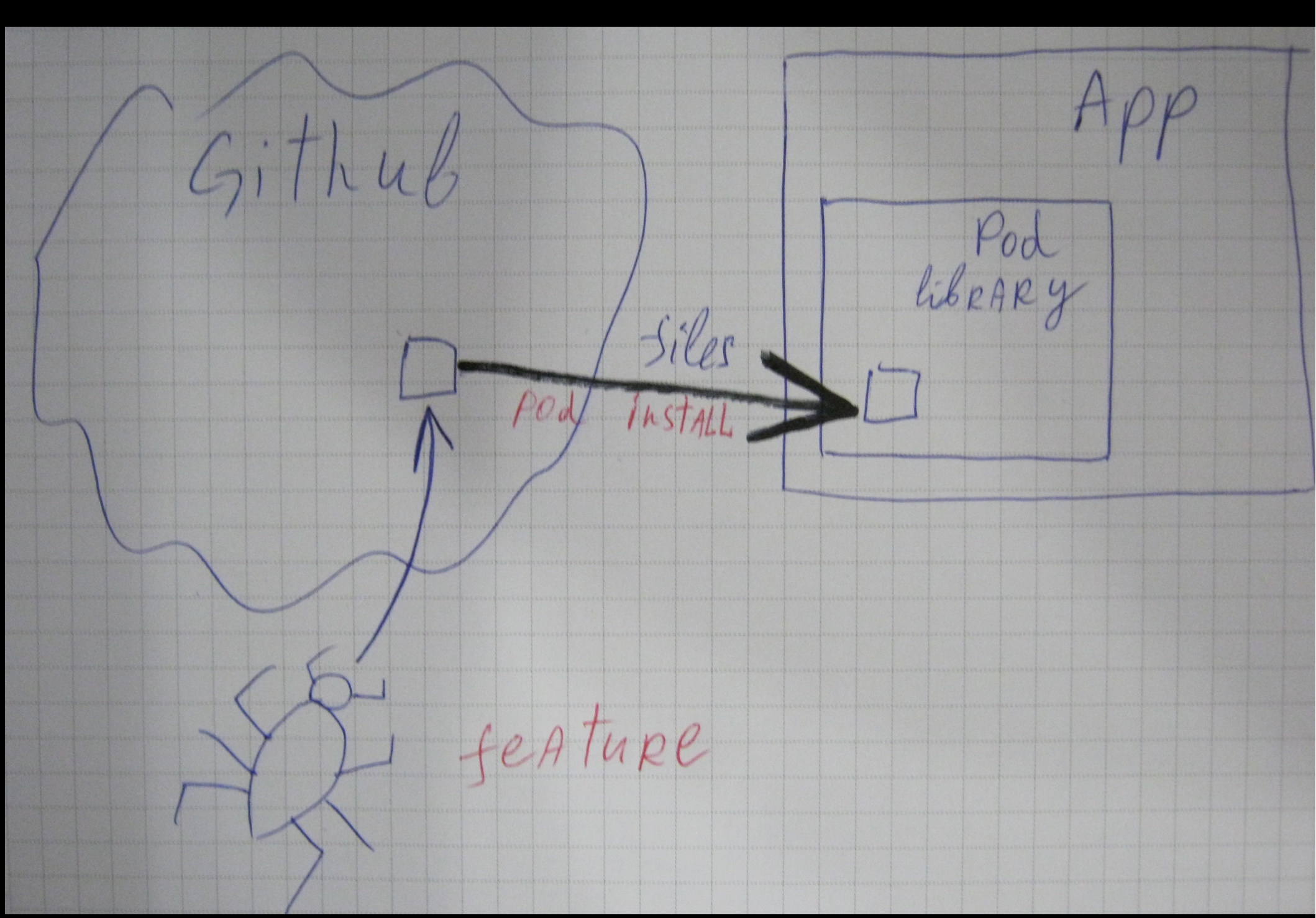
pod install SomeAwesomeLibXYZ

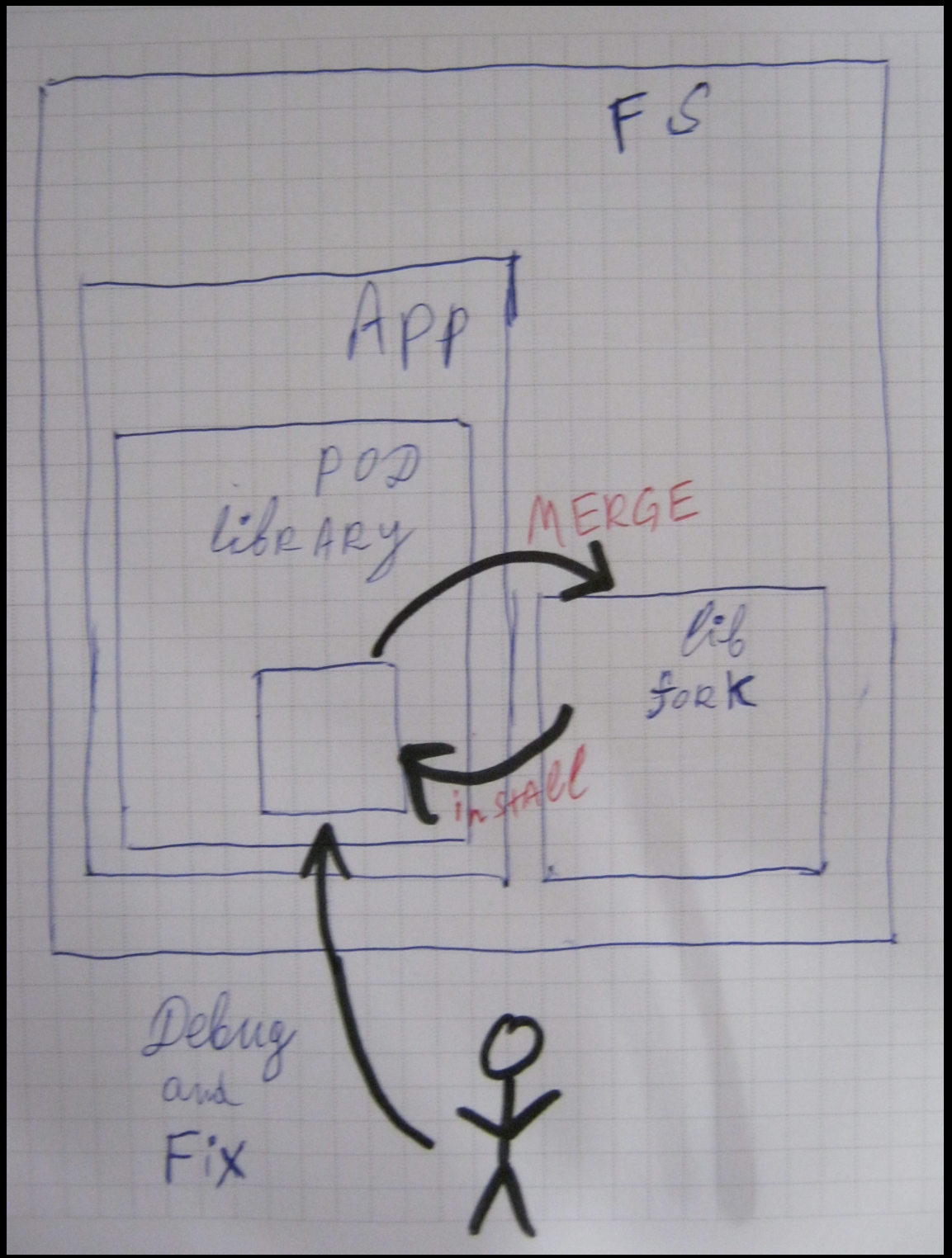
Except...


Library
© dianne j. hook

Typical Workflow

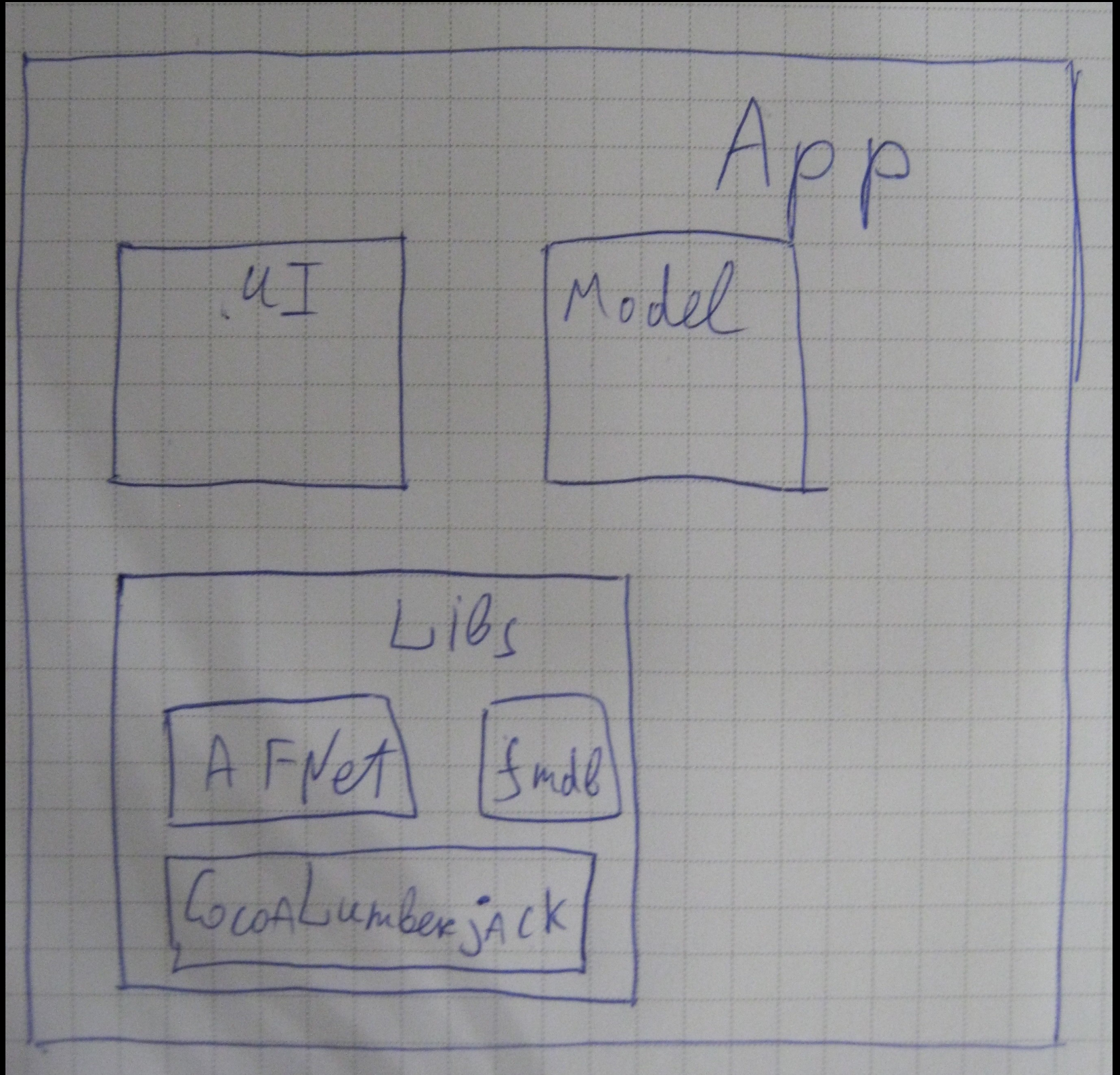

Github
App
Pod
library
files
pod install

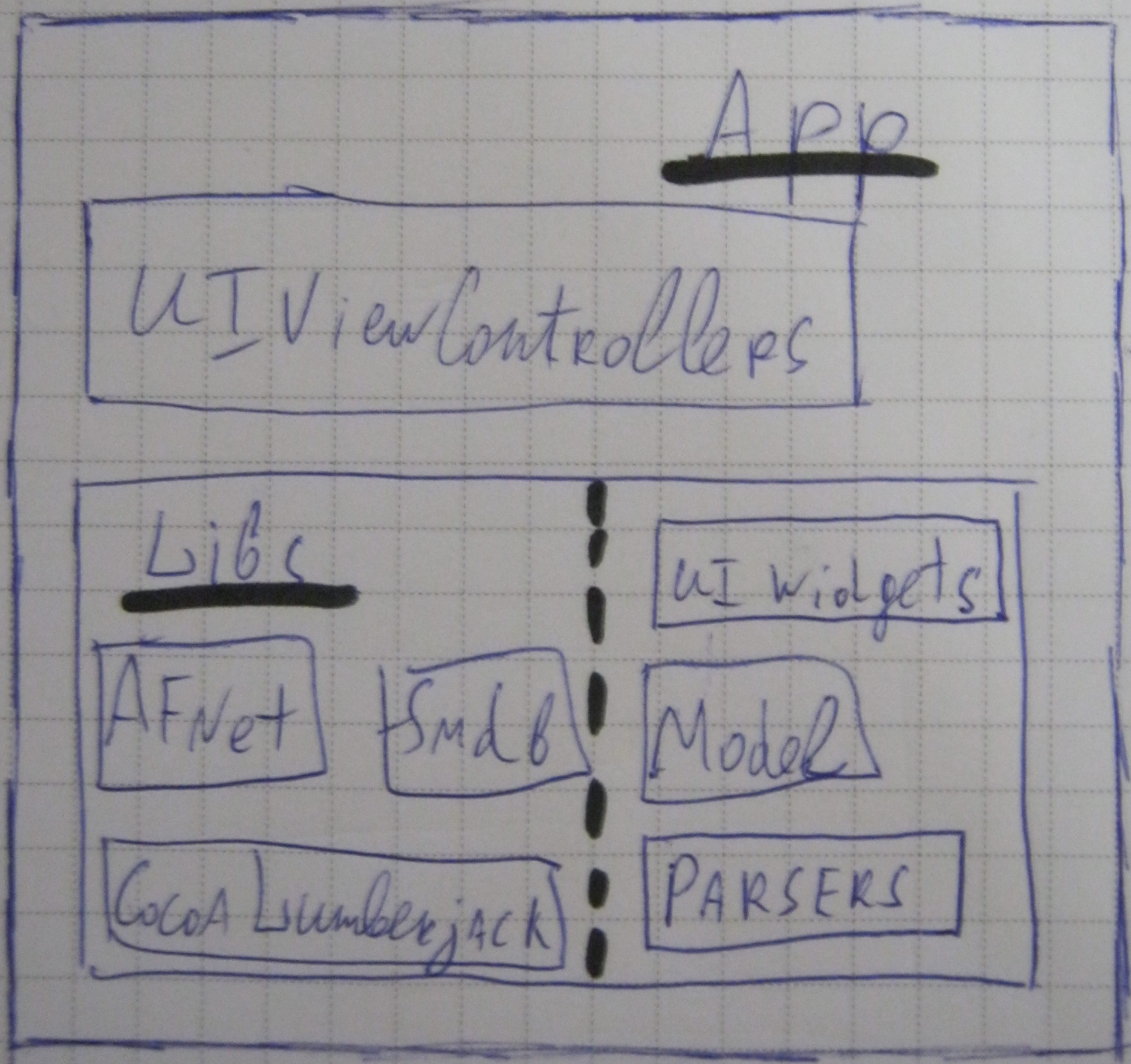

Github
App
Pod
library
files
pod install
feature


FS
App
POD
LIBRARY
MERGE
lib
fork
install
Debug
and
Fix

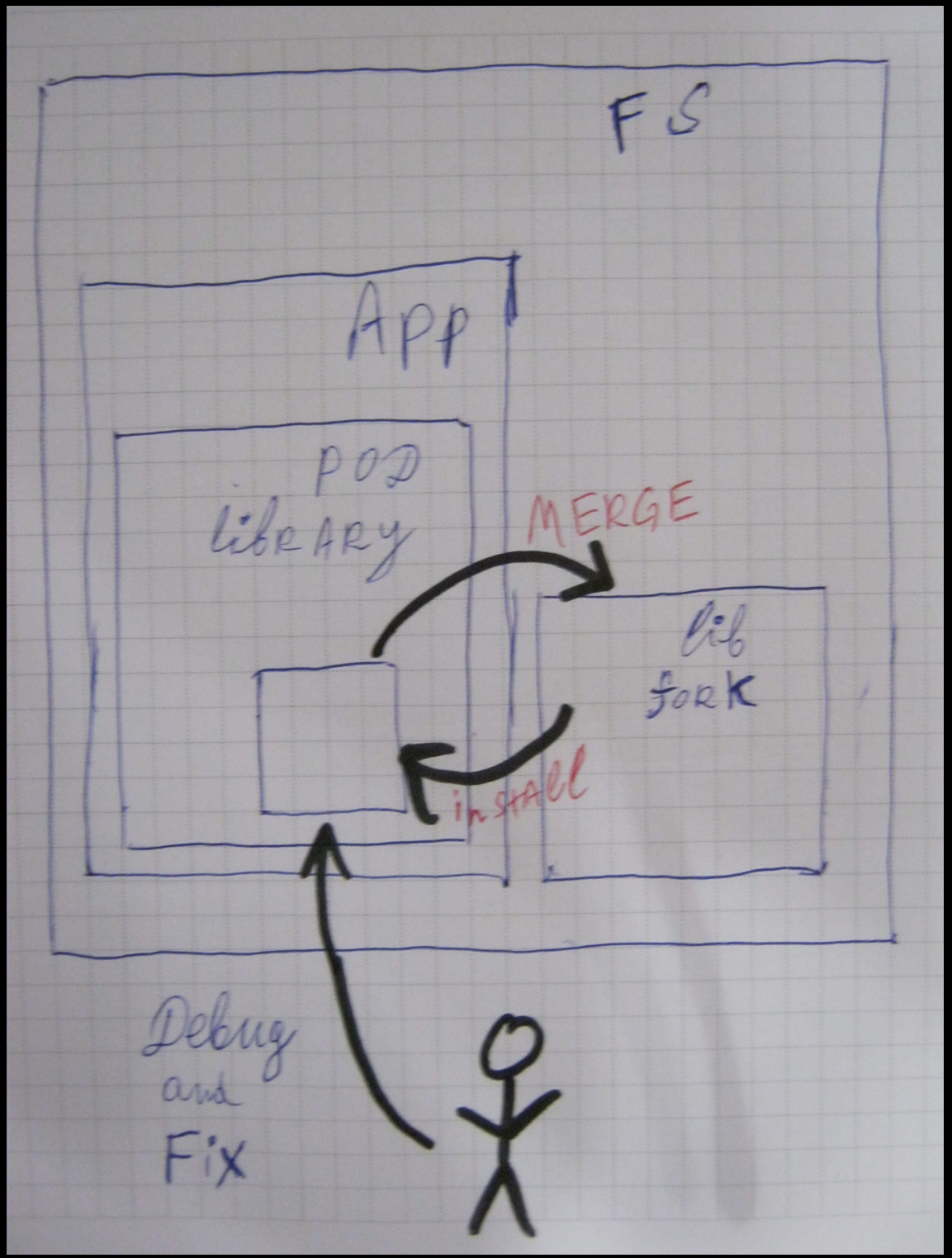
It is a Rare Case

Isn't it ?


App
UI
Model
Libs
AFNet
fmdb
CocoaLumberjack


App
UIViewControllers
Libs
UI widgets
AFNet
FSmdb
Model
CocoaLumberjack
PARSERS

--no-integrate


FS
App
POD
LIBRARY
MERGE
lib
fork
install
Debug
and
Fix

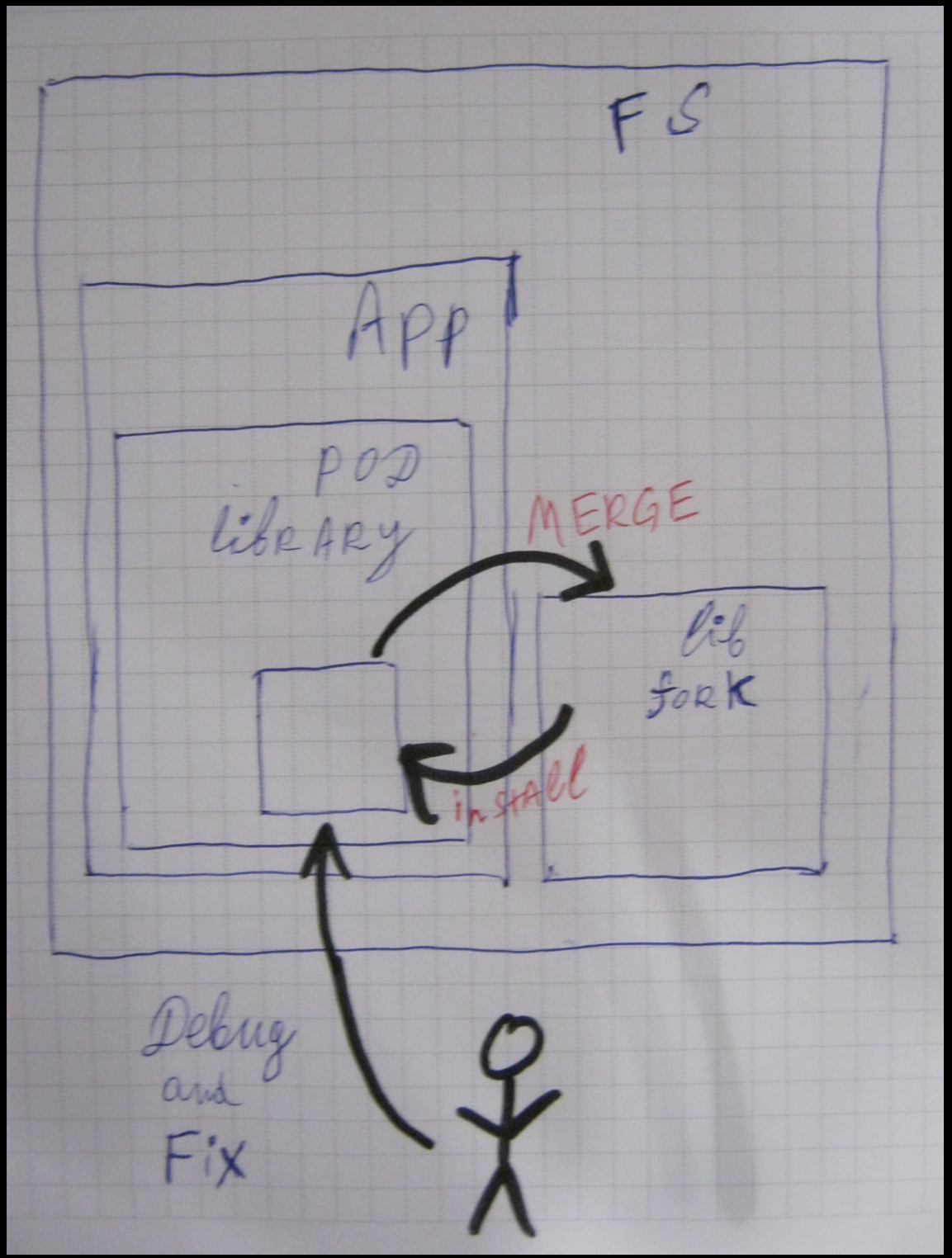
Modular Architecture Benefits

Code Reuse

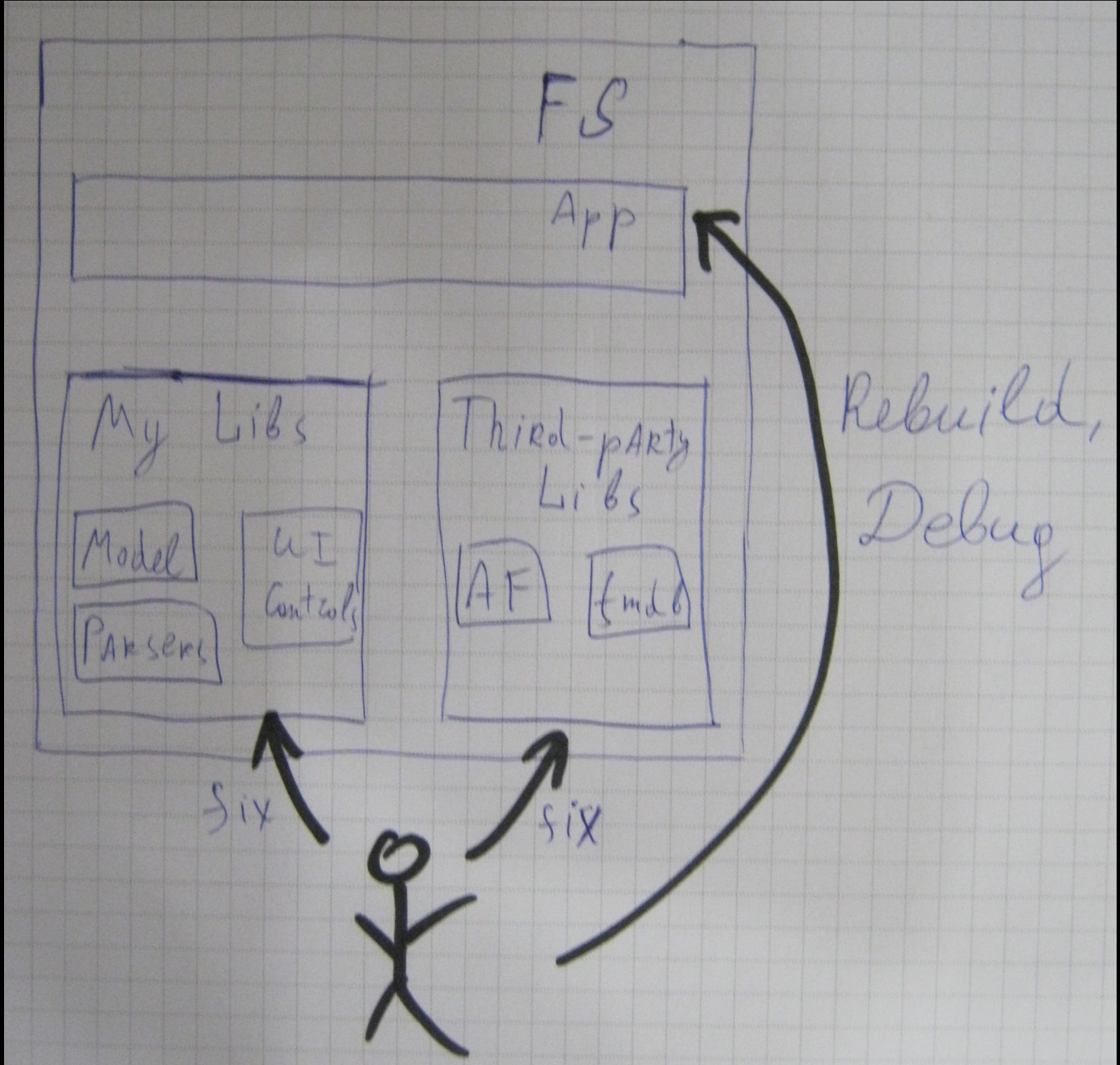
Test Coverage

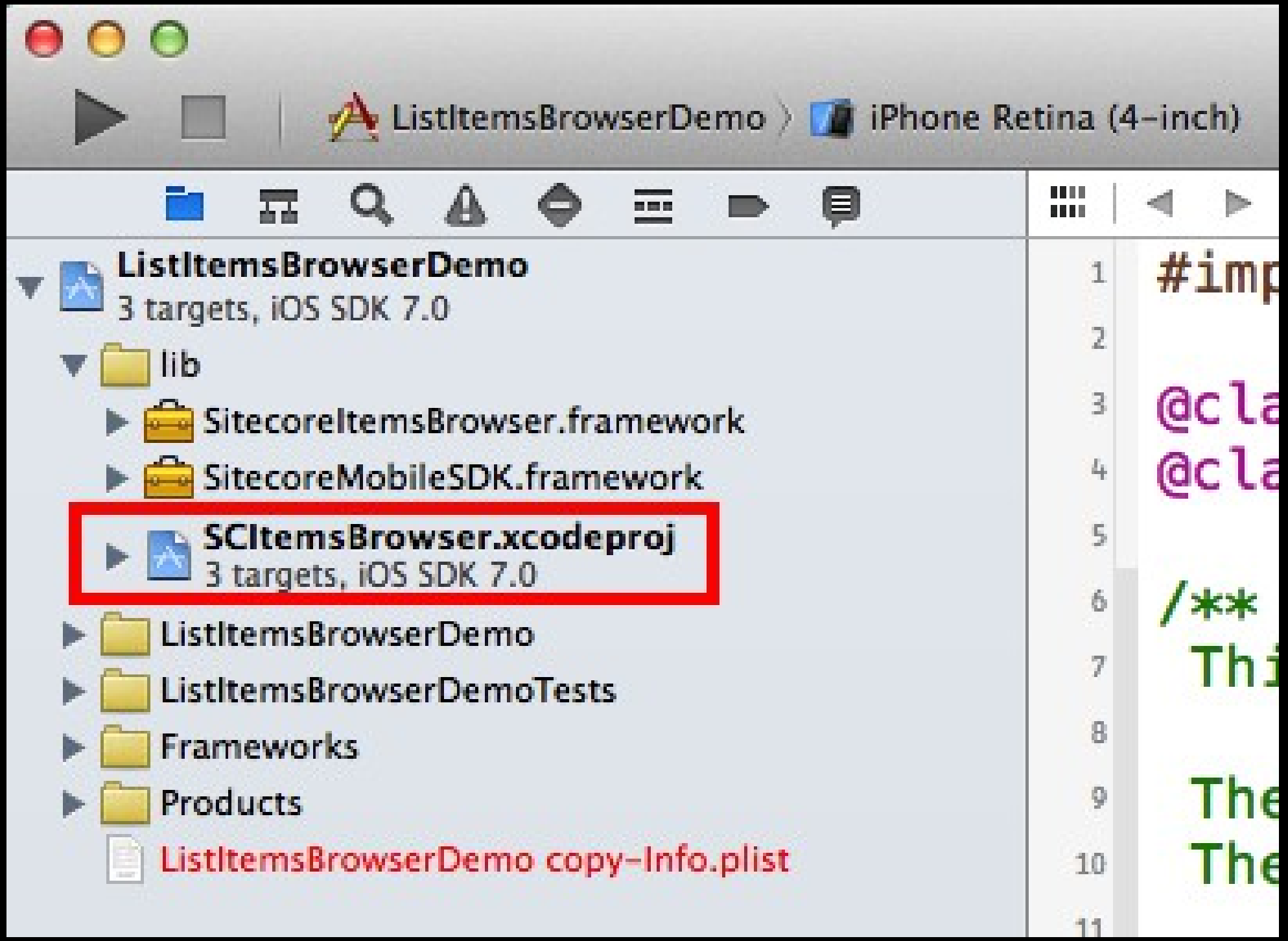
Easier to Apply Changes


App
UIViewControllers
Frequent Changes
Libs
AFNet
Smdb
CocoA Lumberjack
UI widgets
Model
PARSERS


FS
App
POD
LIBRARY
MERGE
lib
fork
install
Debug
and
Fix

Going Native


FS
App
My Libs
Model
Parsers
UI Controls
Third-party Libs
AF
fmdb
Rebuild, Debug
six
six


ListItemsBrowserDemo
iPhone Retina (4-inch)
ListItemsBrowserDemo
3 targets, iOS SDK 7.0
lib
SitecoreItemsBrowser.framework
SitecoreMobileSDK.framework
SCItemsBrowser.xcodeproj
3 targets, iOS SDK 7.0
ListItemsBrowserDemo
ListItemsBrowserDemoTests
Frameworks
Products
ListItemsBrowserDemo copy-Info.plist
#imp
@cla
@cla
/**
Thi
The
The

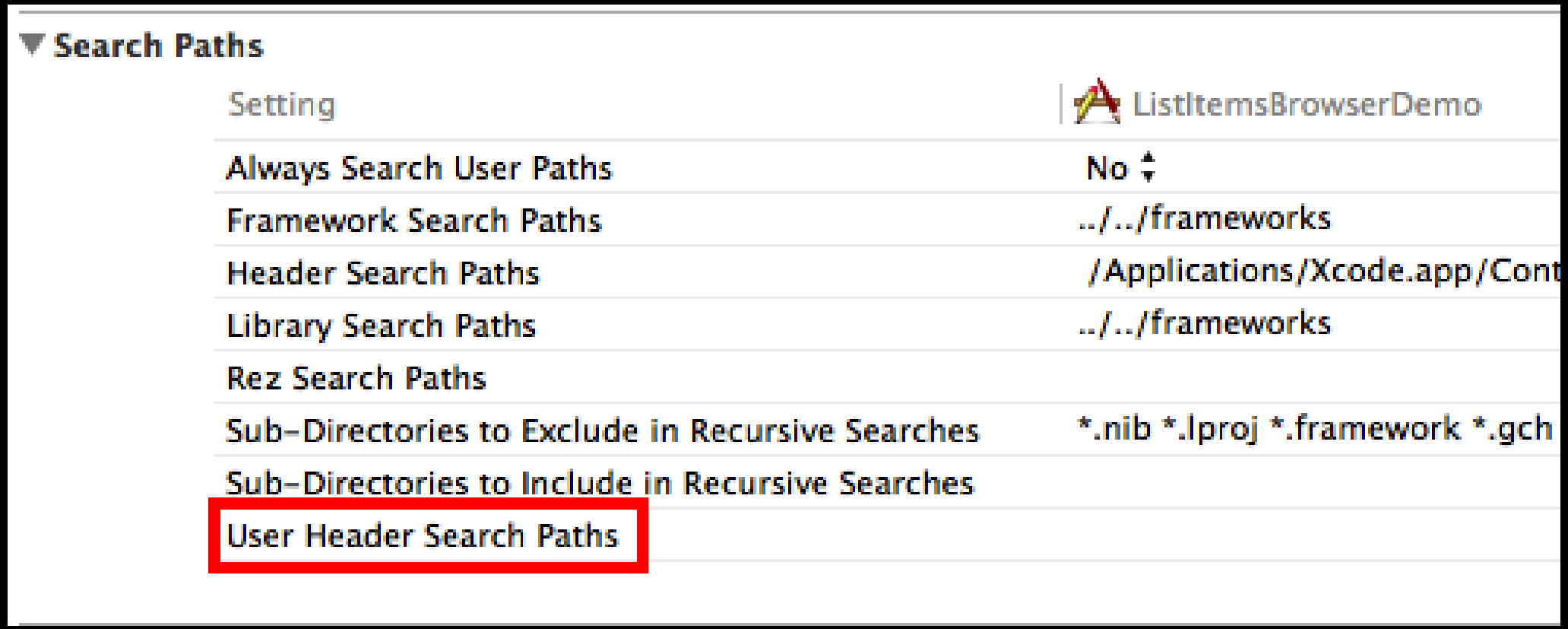
Not Just Library Target

```
#import <AFNetworking/AFNetworking.h>
```

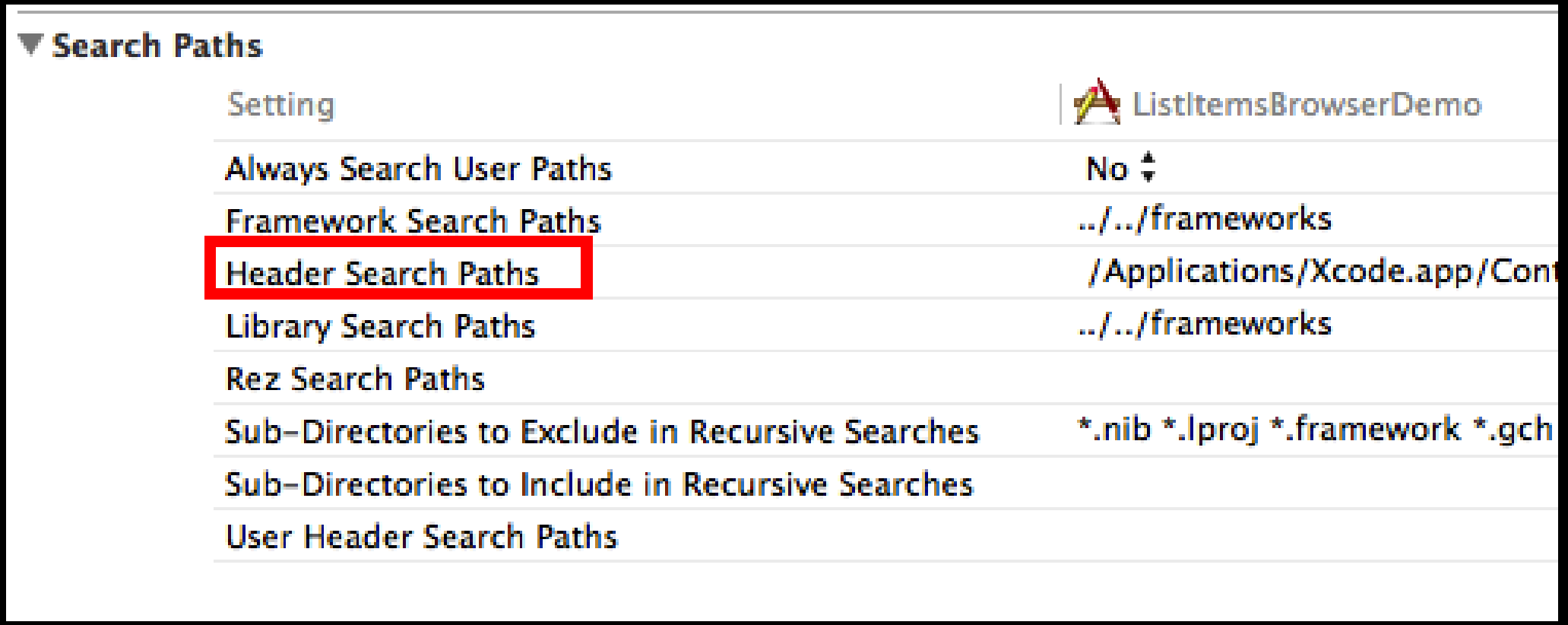
VS

```
#import "AFHTTPRequestOperation.h"
```

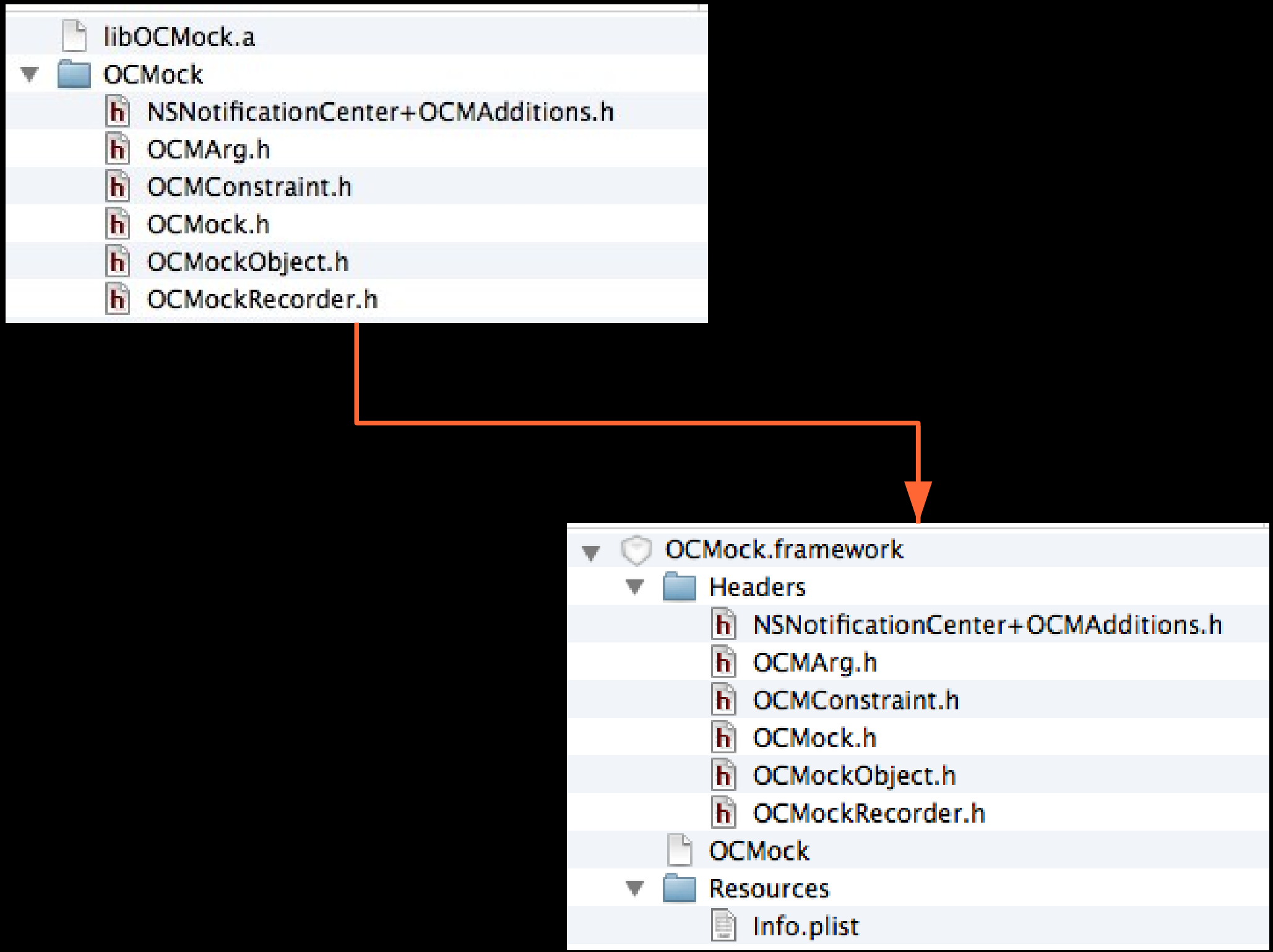
#import "AFHTTPRequestOperation.h"



#import <AFNetworking/AFNetworking.h>



Static Framework for iOS


libOCMock.a
OCMock
NSNotificationCenter+OCMAdditions.h
OCMArg.h
OCMConstraint.h
OCMock.h
OCMockObject.h
OCMockRecorder.h
OCMock.framework
Headers
NSNotificationCenter+OCMAdditions.h
OCMArg.h
OCMConstraint.h
OCMock.h
OCMockObject.h
OCMockRecorder.h
OCMock
Resources
Info.plist

Drag & Drop

Framework Search Path
is updated by Xcode

```
<plist version="1.0">
<dict>
    <key>CFBundleDevelopmentRegion</key>
    <string>English</string>
     <key>CFBundleIdentifier</key>
     <string>org.ocmock</string>
    <key>CFBundleInfoDictionaryVersion</key>
    <string>6.0</string>
     <key>CFBundlePackageType</key>
     <string>FMWK</string>
    <key>CFBundleSignature</key>
    <string>????</string>
     <key>CFBundleVersion</key>
     <string>2.2.3</string>
</dict>
</plist>
```

One More Thing

```
// Pure C headers ho here
#include <math.h>

#ifdef __cplusplus
    // Pure C++ headers ho here
    #include <vector>
#endif

#ifdef __OBJC__
    // Objective-C headers ho here
    #import <Foundation/Foundation.h>

    #ifdef __cplusplus
        #import "MyObjectiveCppClass.h"
    #endif
#endif
```

But I do not Need C++

Print Message for Low Level Error

```
NSDictionary* errorMessages;
NSLog( @"%@", errorMessages[ @( errorCode ) ]  );
```

C++ ===> No Boxing

```
std::map< NSInteger, NSString* > errorMessages;
NSLog( @"%@", errorMessages[ errorCode ] );
```

Scoped Guard

```
void bad(const char* p)
{
    FILE* fh = fopen(p,"r"); // acquire


    // use f
    if ( someCondition )
    {
        // Oops! File handle leaks
        return;
    }

    fclose( fh ); // release
}
```

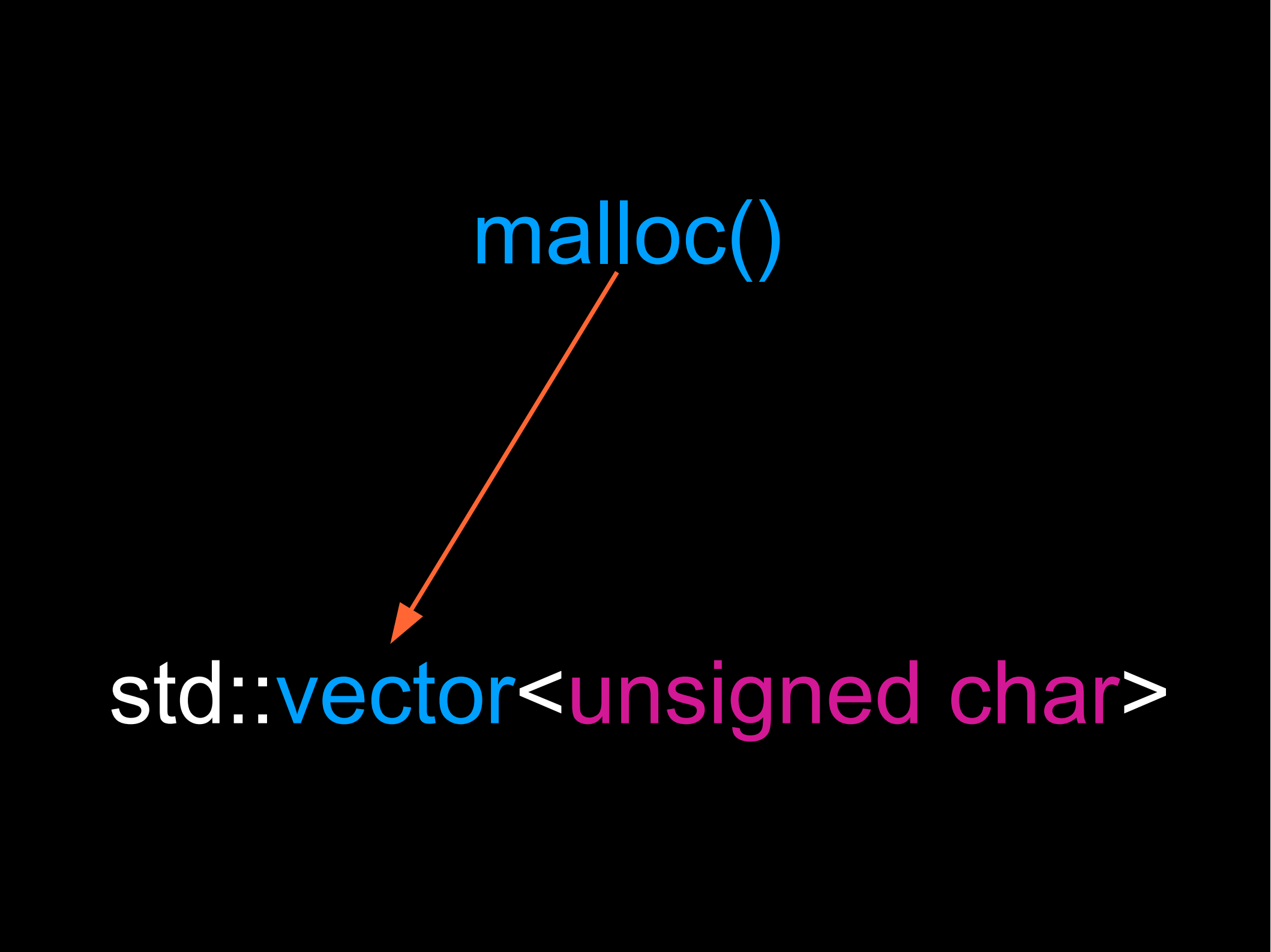
```
void good(const char* p)
{
    FILE* fh = fopen(p,"r"); // acquire

    // the block to perform cleanup actions
    GuardCallbackBlock releaseBlock_ = ^void( void )
    {
        fclose( fh );
    };

    // creating a guard
    ObjcScopedGuard guard( releaseBlock_ );

    if ( someCondition )
    {
        // Now the scoped guard will release the resource
        return;
    }
}
```

Exception-Safe Resource Deallocation


malloc()
std::vector<unsigned char>

Native Rulezzz

Alexander Dodatko

@dodikk88