Agenda

1. Swift: state of the art

2. Code!

Why Swift

`[objC autorelease];`

What is Swift

The same

- LLVM compiler

- ARC

- Xcode

Not the same

- Without direct memory access

- Strictly typed

- Modern language features

Swift 1.x*

It is allowed to publish Swift apps in AppStore, but

- Bugs

- Runtime performance issues**

- Lack of mature native libraries

- Changes **continuously**

*this presentation is obsolete already!

Buggy as …

SourceKitService
Crashed
Crashlog generated
in ~/Library/Logs/
DiagnosticReports

Editor functionality
temporarily limited.

Changes continuously*

Xcode 6.1	...
Xcode 6.1 GM	Seed 1, Seed 2
Xcode 6.1 beta	Beta 1 - Beta 2
Xcode 6.0 beta	Beta 1 - Beta 7

*Oct, 2014

Changes continuously*

But you will be politely noticed

```
var sOld: String[] = ["1"] // outdated
var sNew: [String] = ["2"] // modern  🔴 Array types are now written with the brackets around the element type
```

*Oct, 2014

Changes continuously*

Apple Swift Blog

https://developer.apple.com/swift/blog/

Apple Swift dev forum

https://devforums.apple.com/community/tools/languages/swift

Great community resource about Swift evolution

https://github.com/ksm/SwiftInFlux

*Oct, 2014

What is a real project

- A LOT OF Objective-C code already written

- Third party libraries (including C/C++ ones)

- Cocoapods

- CoreData

- Networking

- Unit testing

- … *<Your favorite>*

Legacy code in Swift
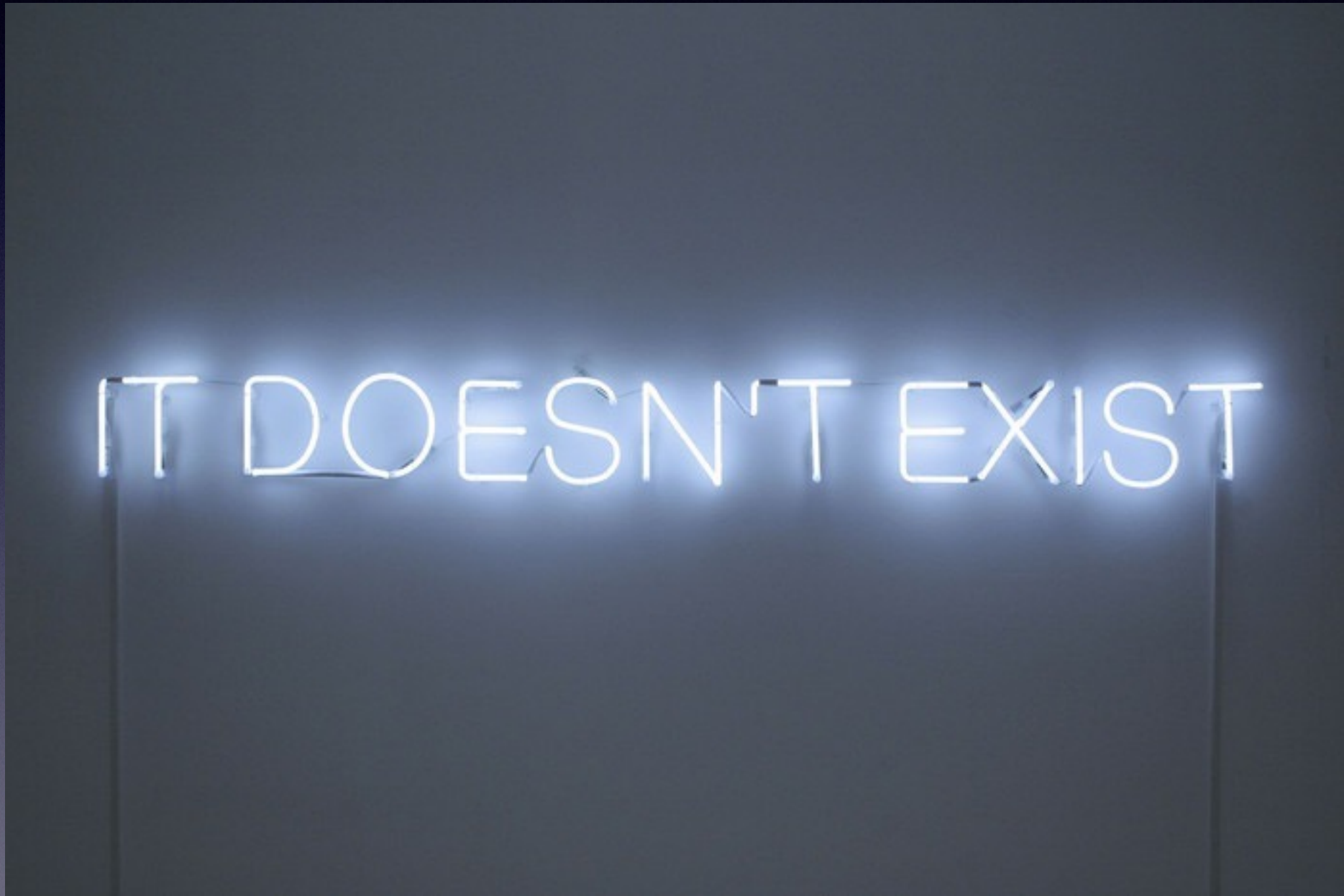

Objective-C modules in project - easy

Legacy code in Swift

C is available also

C++ not, as well as Objective-C++
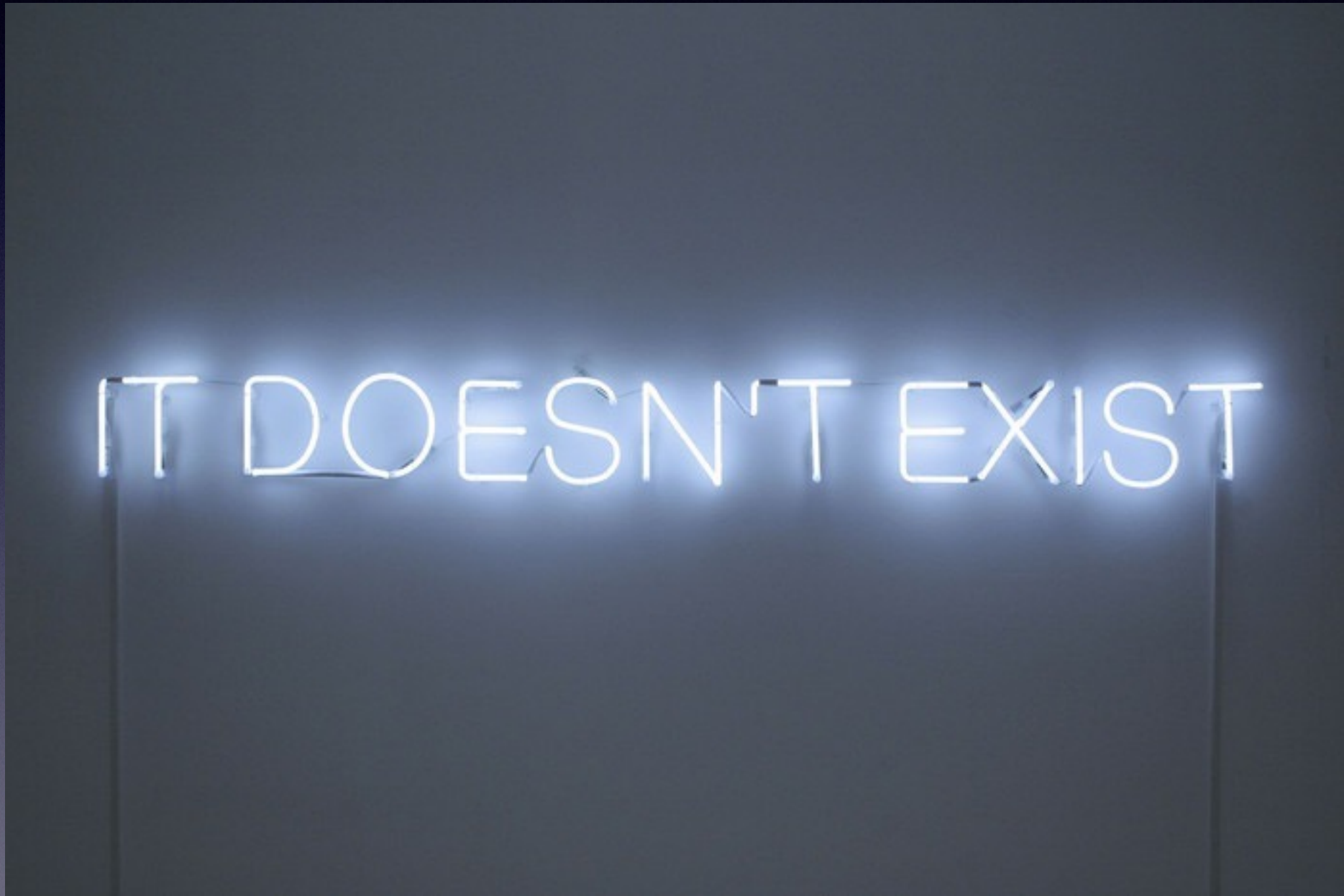
Swift own networking

Swift own networking
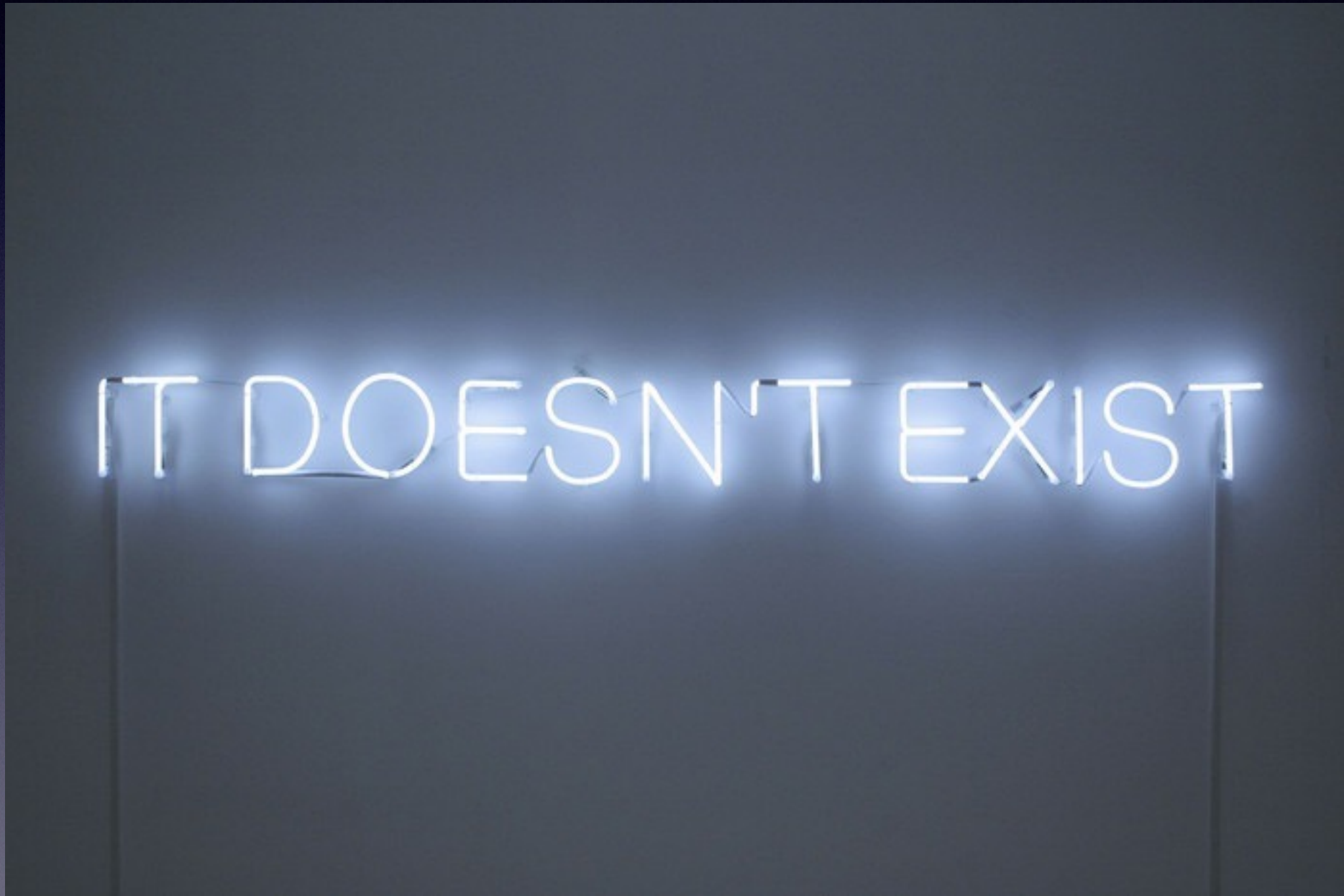

ALAMOFIRE

Elegant Networking in Swift

Swift own threading

Swift own threading

Swift own CoreData

Swift own CoreData

we still have magic **@objc** and **@NSManaged**

Swift performance

[-Onone]:

```
Swift:      0.678056695
C:          0.000973914
```

[-O]:

```
Swift:      0.001158492
C:          0.001192406
```

[-Ounchecked]:

```
Swift:      0.000827764
C:          0.001078914
```

share | improve this answer   edited Oct 9 at 3:18   answered Jun 8 at 1:36
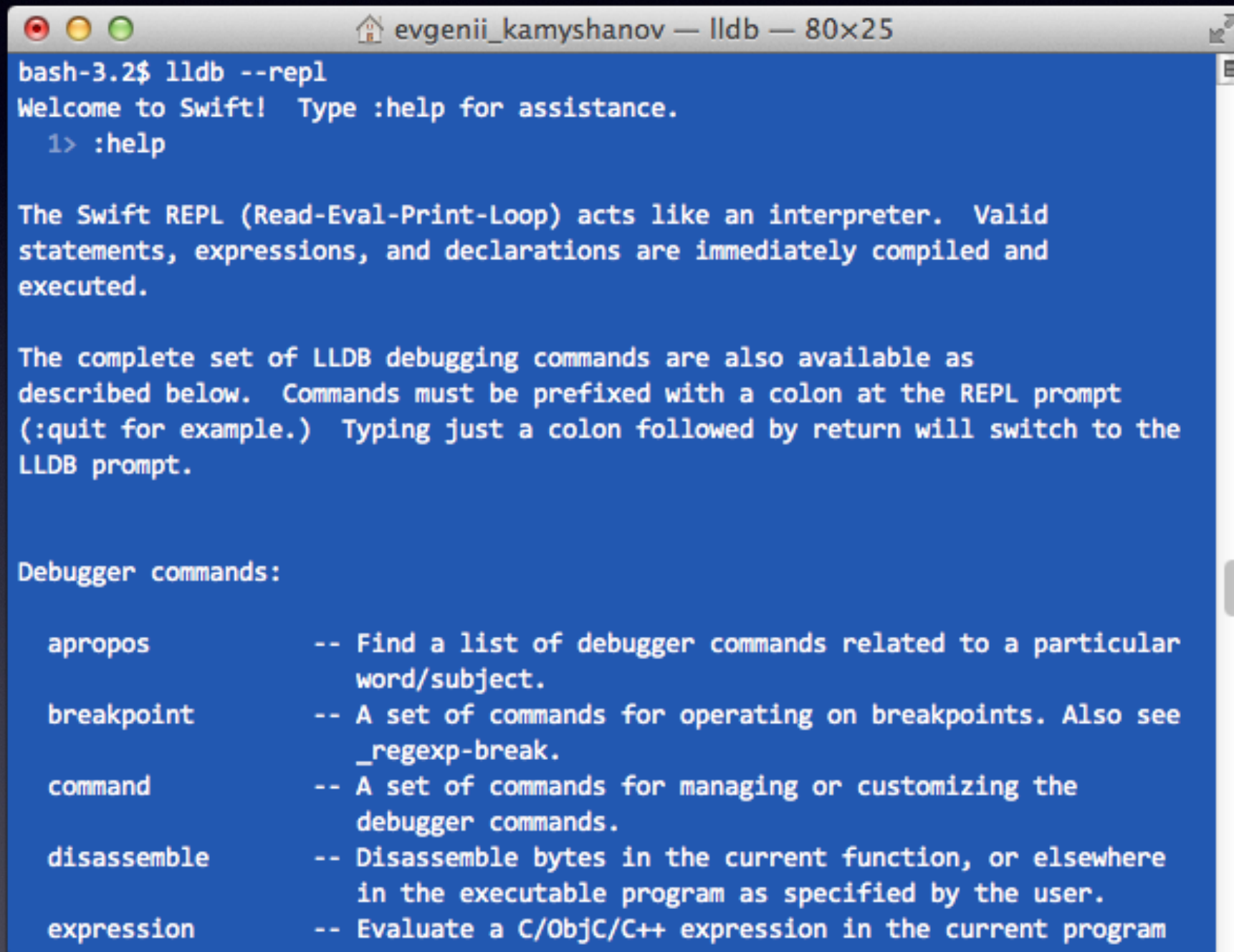
Joseph Mark
2,206 ●2 ●6 ●17

stack**overflow**

Swift tools

- Xcode 6

- Playground

- REPL (Read-Eval-Print-Loop)

bash$ xcrun swift
bash$ lldb --repl


```
bash-3.2$ lldb --repl
Welcome to Swift!  Type :help for assistance.
  1> :help

The Swift REPL (Read-Eval-Print-Loop) acts like an interpreter.  Valid
statements, expressions, and declarations are immediately compiled and
executed.

The complete set of LLDB debugging commands are also available as
described below.  Commands must be prefixed with a colon at the REPL prompt
(:quit for example.)  Typing just a colon followed by return will switch to the
LLDB prompt.


Debugger commands:

  apropos          -- Find a list of debugger commands related to a particular
                      word/subject.
  breakpoint       -- A set of commands for operating on breakpoints. Also see
                      _regexp-break.
  command          -- A set of commands for managing or customizing the
                      debugger commands.
  disassemble      -- Disassemble bytes in the current function, or elsewhere
                      in the executable program as specified by the user.
  expression       -- Evaluate a C/ObjC/C++ expression in the current program
```

Code!

- Swift app

- with Objective-C classes imported

- with REST service calling

- with CoreData caching

- with some UI

Swift

Applied

All the code is available on Github:

https://github.com/ptiz/RealLifeApp.git

Thank You

Evgenii Kamyshanov

SSE, EPAM Systems

✉ evgenii_kamyshanov@epam.com

in Evgeny Kamyshanov

Ⓢ evgeny.kamyshanov

Links

- Swift InFlux: https://github.com/ksm/SwiftInFlux

- Swift change log: https://developer.apple.com/library/ios/documentation/Swift/Conceptual/Swift_Programming_Language/RevisionHistory.html

- Swift on DevForums: https://devforums.apple.com/community/tools/languages/swift

- Cocoapods for Swift: https://github.com/CocoaPods/CocoaPods/issues/2272

- Swift performance: http://stackoverflow.com/questions/24101718/swift-performance-sorting-arrays

- Alamofire: https://github.com/Alamofire/Alamofire

- CoreData using Swift: https://github.com/KoNEW/CoreDataTest

- ifacegen: https://github.com/ptiz/ifacegen