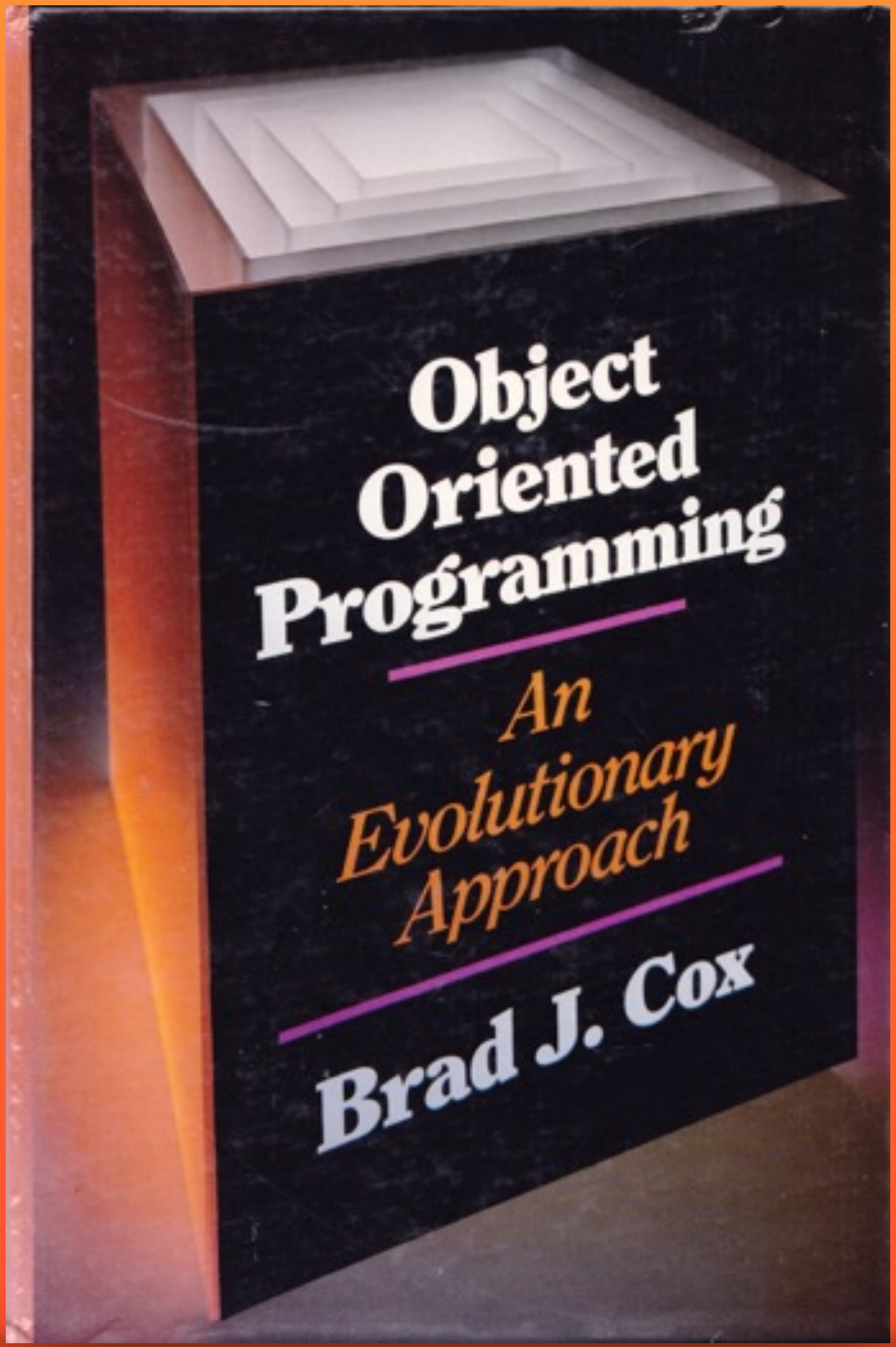
THE STATE OF SWIFT

KÁROLY LŐRENTEY

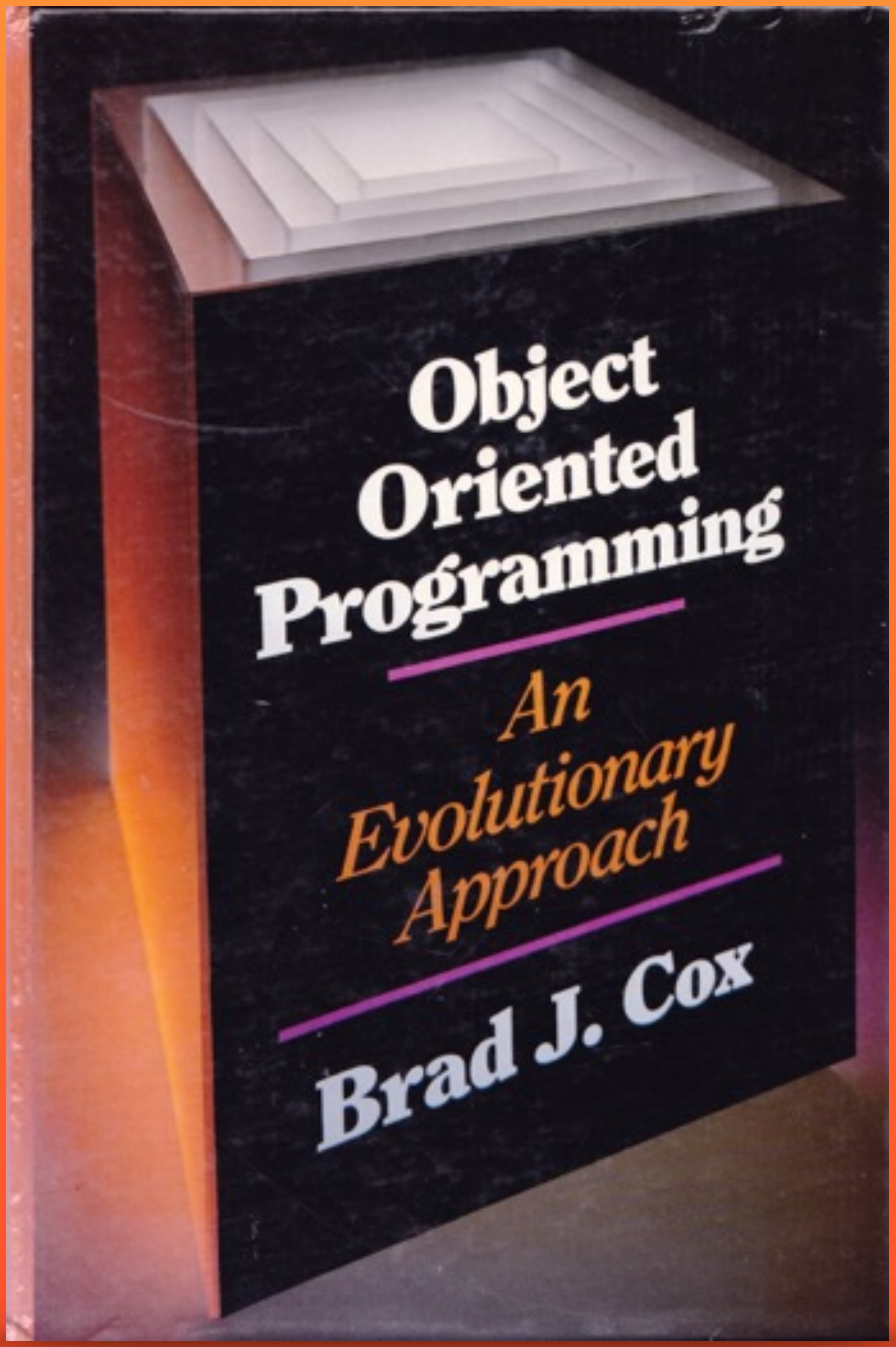
@lorentey

1986



Objective-C

1986



Object Oriented Programming

An Evolutionary Approach

Brad J. Cox

Objective-C

1989

NEXTStep 1.0

2014

Swift

```
var people = ["Dave", "Brian", "Alex", "A
let name = "Alex"
if let index = find(people, name) {
    println("\(name) is person \(index +
    delegate?.didFindPersonWithName(name,
} else {
    println("Unable to find \(name) in t
}
```

2014

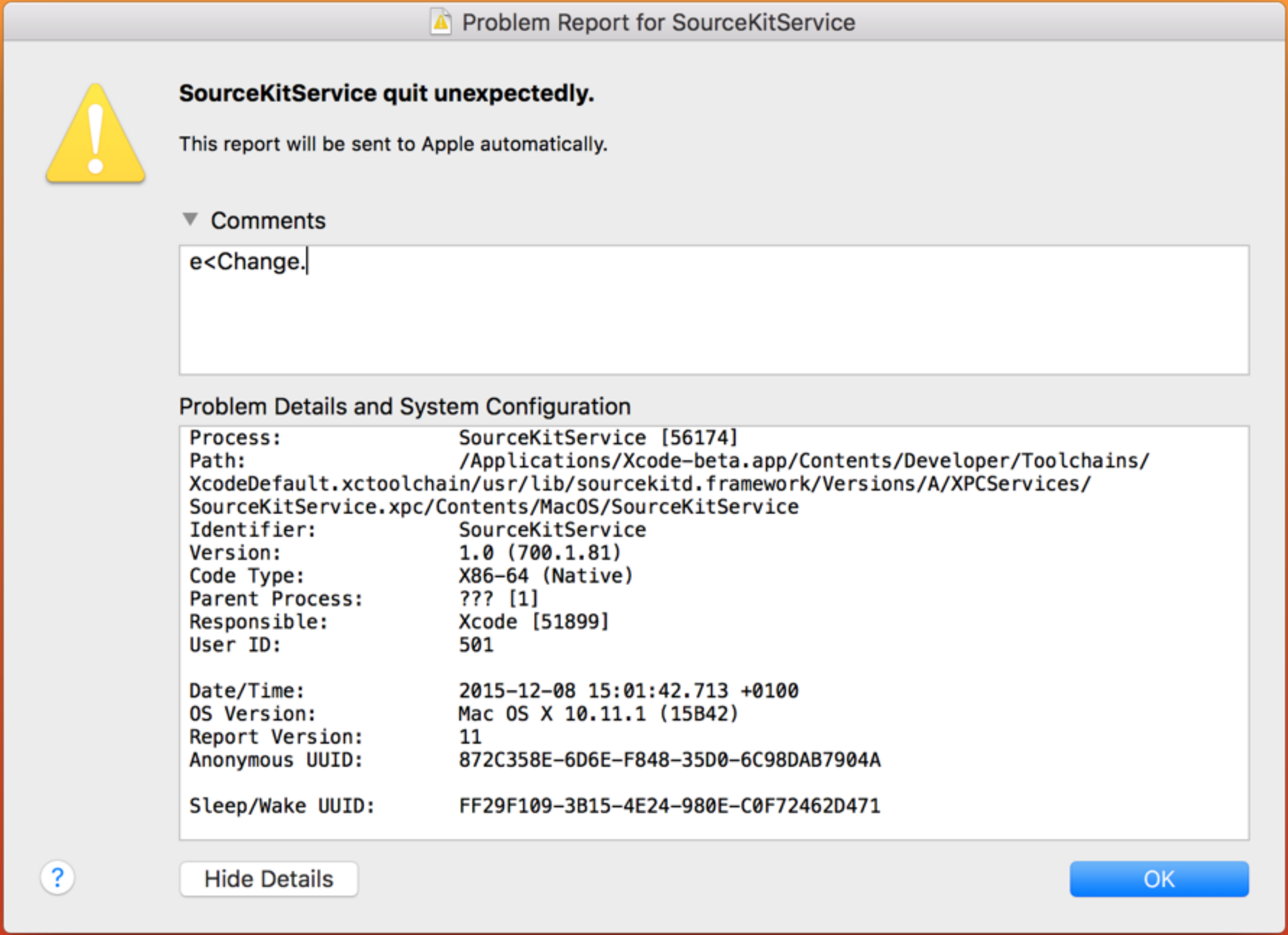
Swift

```
var people = ["Dave", "Brian", "Alex", "A
let name = "Alex"
if let index = find(people, name) {
    println("\(name) is person \(index +
    delegate?.didFindPersonWithName(name,
} else {
    println("Unable to find \(name) in th
}
```

2017

?

18 MONTHS



SourceKitService quit unexpectedly.

This report will be sent to Apple automatically.

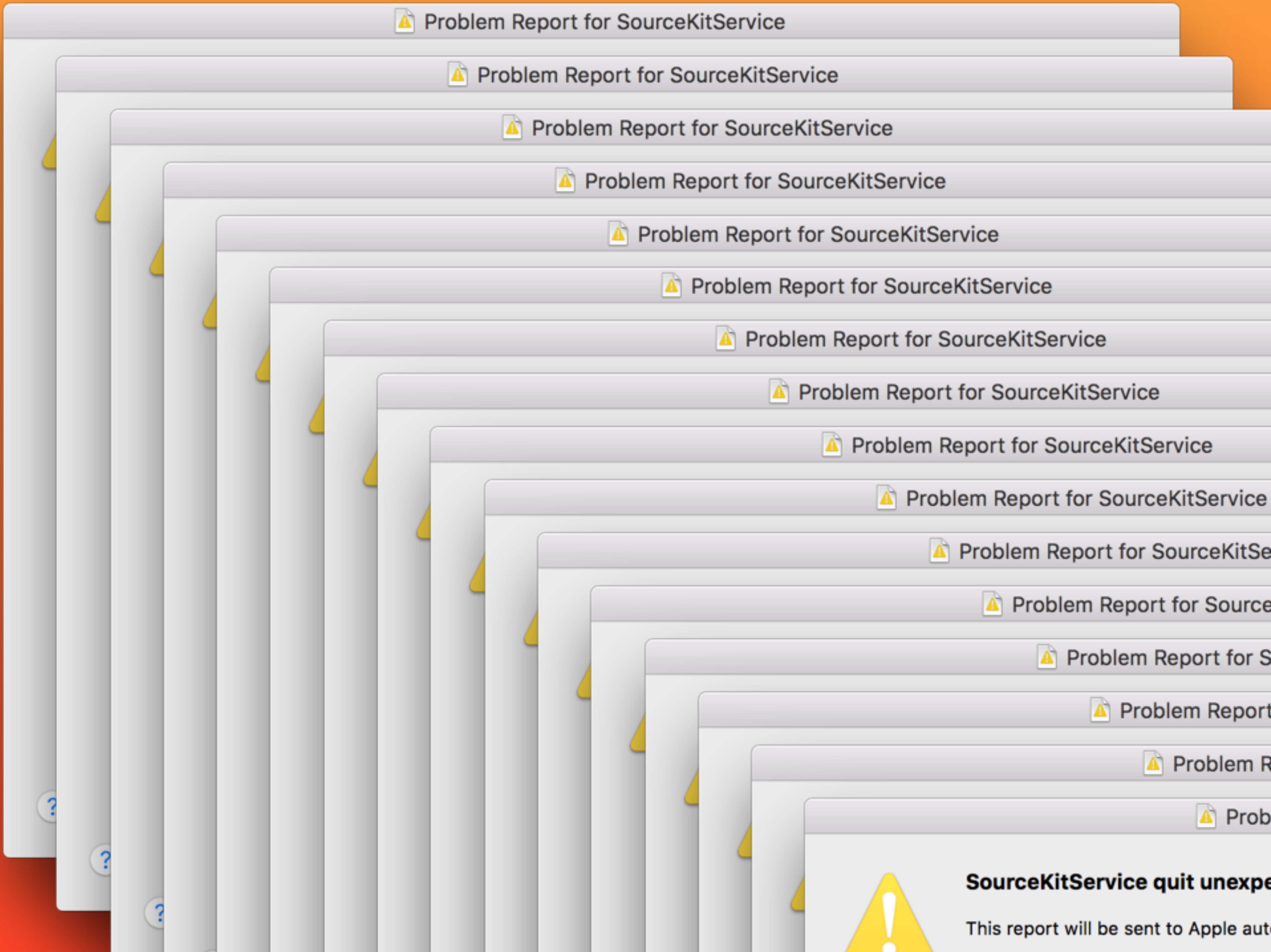
e<Change.

Problem Details and System Configuration

```
Process:               SourceKitService [56174]
Path:                  /Applications/Xcode-beta.app/Contents/Developer/Toolchains/
XcodeDefault.xctoolchain/usr/lib/sourcekitd.framework/Versions/A/XPCServices/
SourceKitService.xpc/Contents/MacOS/SourceKitService
Identifier:            SourceKitService
Version:               1.0 (700.1.81)
Code Type:             X86-64 (Native)
Parent Process:        ??? [1]
Responsible:           Xcode [51899]
User ID:               501

Date/Time:             2015-12-08 15:01:42.713 +0100
OS Version:            Mac OS X 10.11.1 (15B42)
Report Version:        11
Anonymous UUID:        872C358E-6D6E-F848-35D0-6C98DAB7904A

Sleep/Wake UUID:       FF29F109-3B15-4E24-980E-C0F72462D471
```



Problem Report for SourceKitService

Problem Report for SourceKitService

Problem Report for SourceKitService

Problem Report for SourceKitService

Problem Report for SourceKitService

Problem Report for SourceKitService

Problem Report for SourceKitService

Problem Report for SourceKitService

Problem Report for SourceKitService

Problem Report for SourceKitService

Problem Report for SourceKitSe

Problem Report for Source

Problem Report for S

Problem Report

Problem R

Prob

SourceKitService quit unexpe

This report will be sent to Apple aut

Problem Report for SourceKitService

Problem Report for SourceKitService

Problem Report for SourceKitService

Problem Report for SourceKitService

ice

tService

urceKitService

or SourceKitSe

port for Source

m Report for S

roblem Report

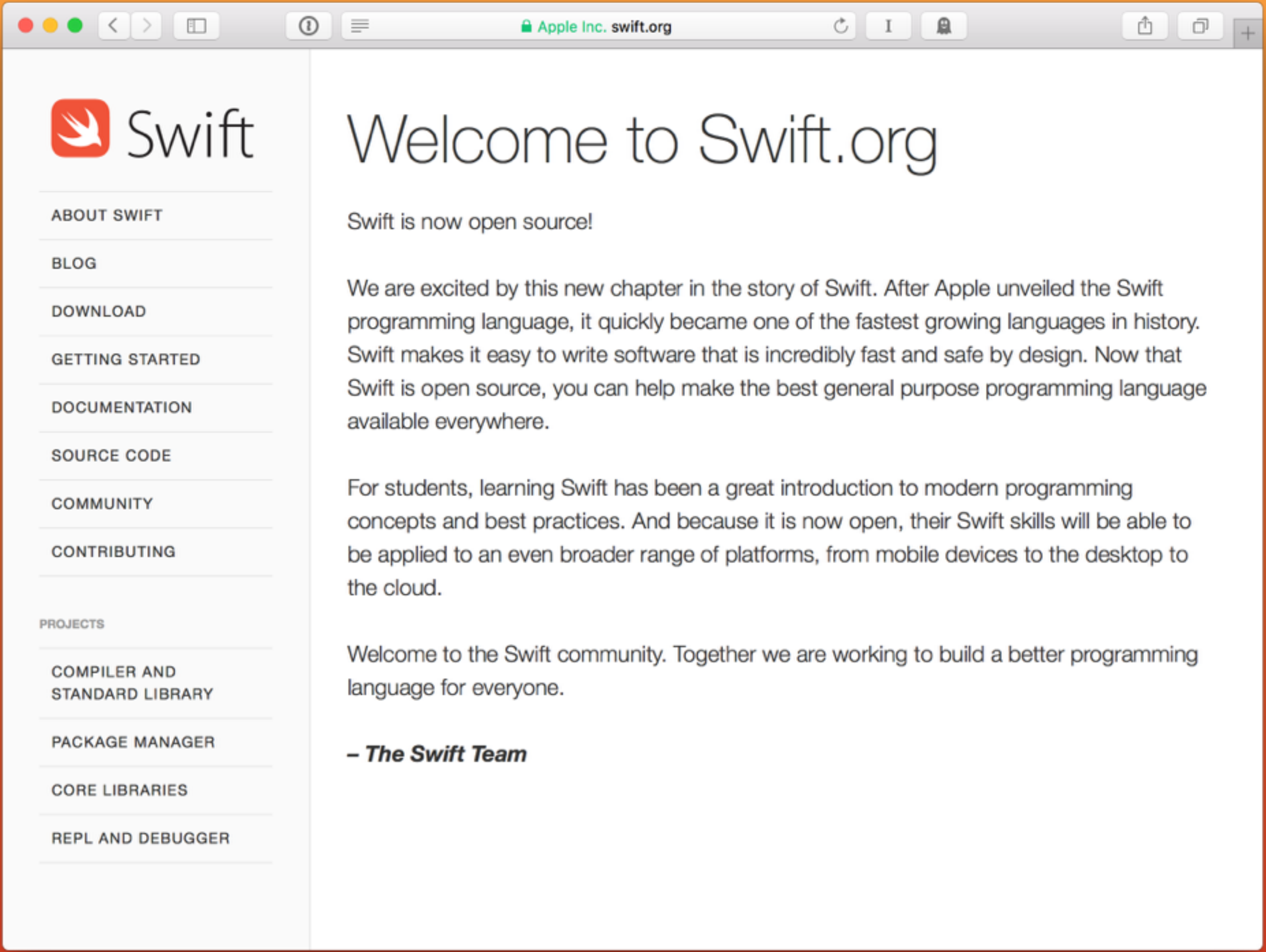
Problem R

Prob

SourceKitService quit unexpe

This report will be sent to Apple aut

http://gunshowcomic.com/648 © KC GREEN



Swift

- ABOUT SWIFT
- BLOG
- DOWNLOAD
- GETTING STARTED
- DOCUMENTATION
- SOURCE CODE
- COMMUNITY
- CONTRIBUTING

PROJECTS

- COMPILER AND STANDARD LIBRARY
- PACKAGE MANAGER
- CORE LIBRARIES
- REPL AND DEBUGGER

Welcome to Swift.org

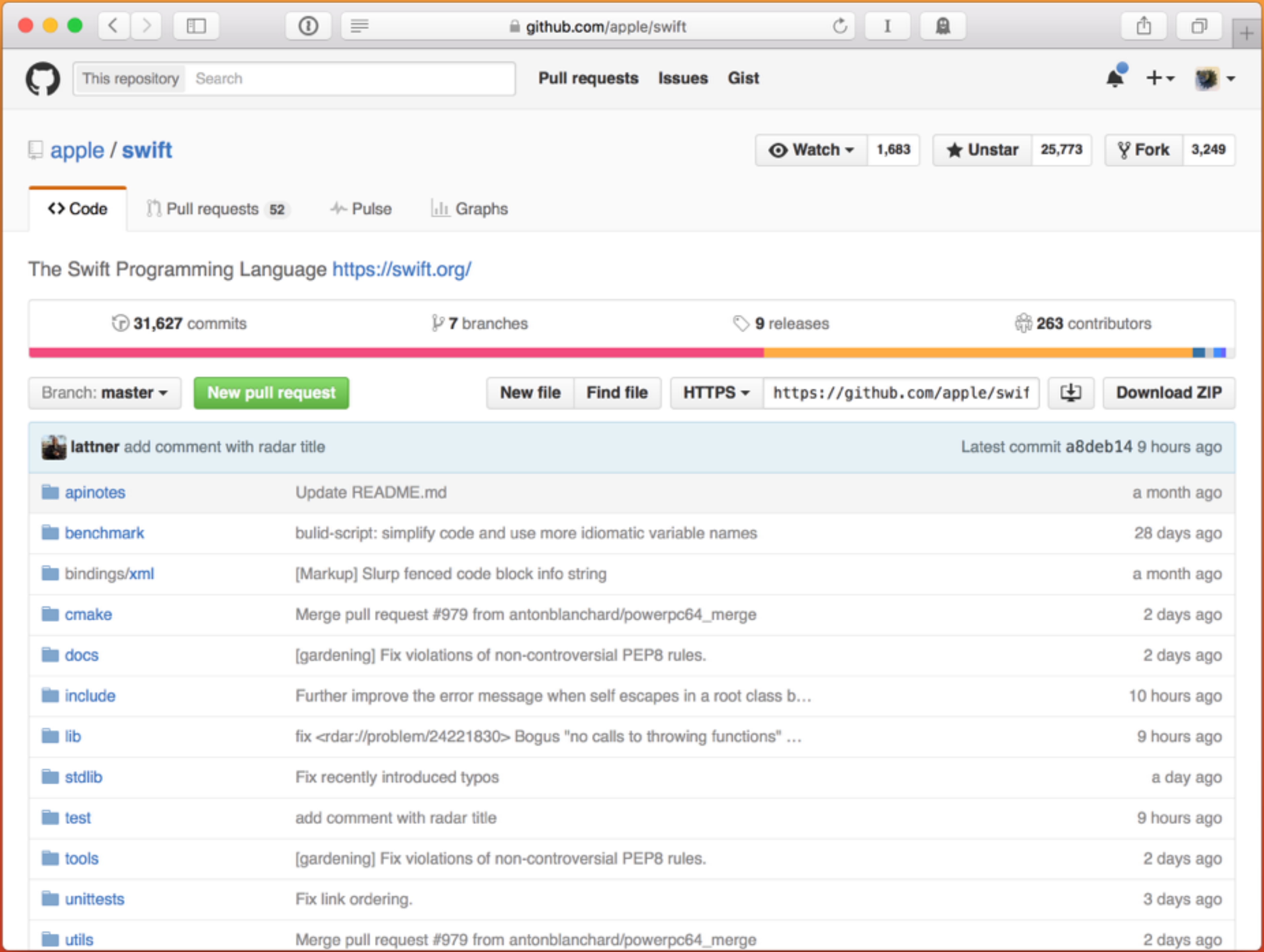
Swift is now open source!

We are excited by this new chapter in the story of Swift. After Apple unveiled the Swift programming language, it quickly became one of the fastest growing languages in history. Swift makes it easy to write software that is incredibly fast and safe by design. Now that Swift is open source, you can help make the best general purpose programming language available everywhere.

For students, learning Swift has been a great introduction to modern programming concepts and best practices. And because it is now open, their Swift skills will be able to be applied to an even broader range of platforms, from mobile devices to the desktop to the cloud.

Welcome to the Swift community. Together we are working to build a better programming language for everyone.

– The Swift Team



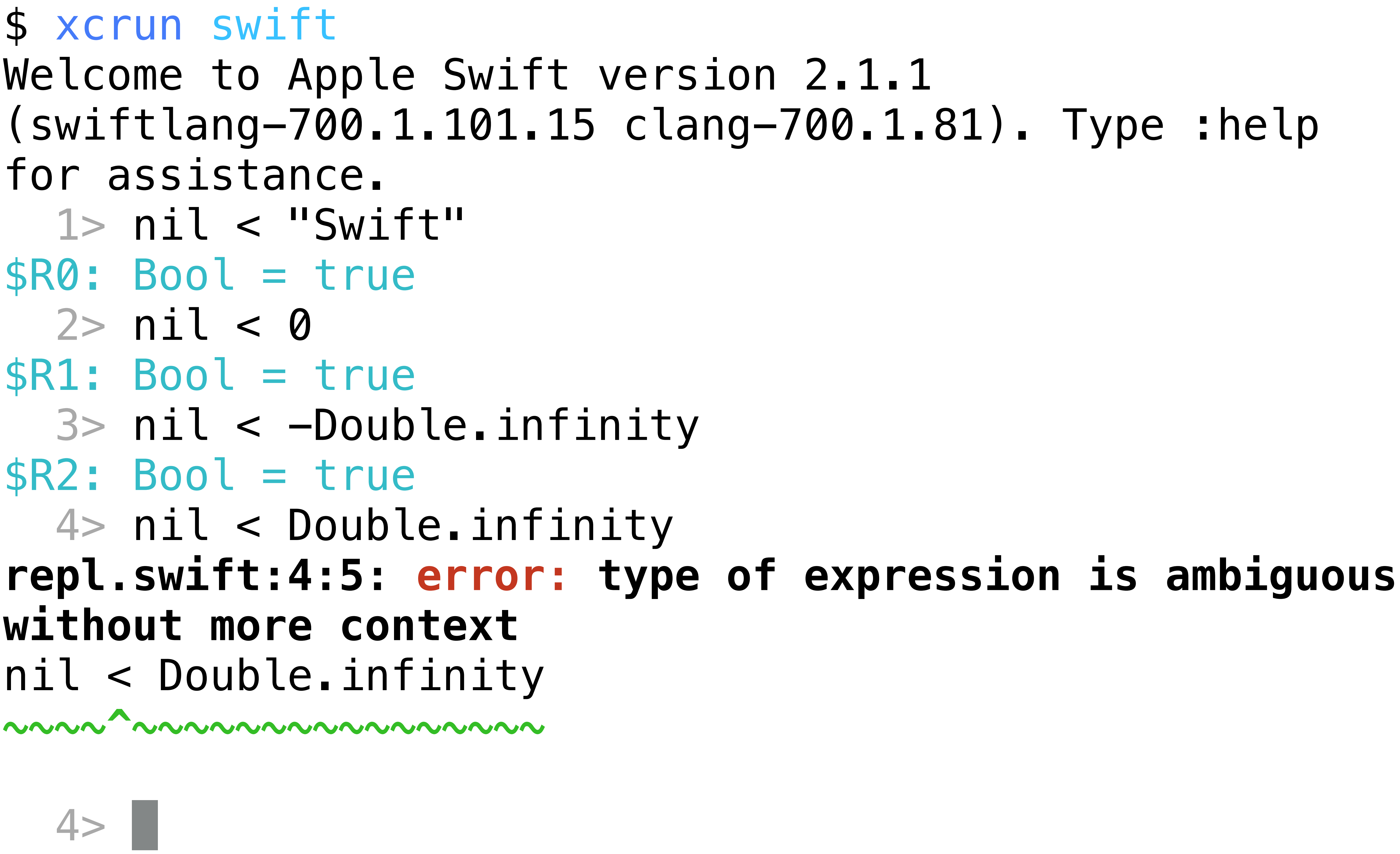
The Swift Programming Language https://swift.org/

31,627 commits
7 branches
9 releases
263 contributors

lattner add comment with radar title — Latest commit a8deb14 9 hours ago

apinotes	Update README.md	a month ago
benchmark	bulid-script: simplify code and use more idiomatic variable names	28 days ago
bindings/xml	[Markup] Slurp fenced code block info string	a month ago
cmake	Merge pull request #979 from antonblanchard/powerpc64_merge	2 days ago
docs	[gardening] Fix violations of non-controversial PEP8 rules.	2 days ago
include	Further improve the error message when self escapes in a root class b…	10 hours ago
lib	fix <rdar://problem/24221830> Bogus "no calls to throwing functions" …	9 hours ago
stdlib	Fix recently introduced typos	a day ago
test	add comment with radar title	9 hours ago
tools	[gardening] Fix violations of non-controversial PEP8 rules.	2 days ago
unittests	Fix link ordering.	3 days ago
utils	Merge pull request #979 from antonblanchard/powerpc64_merge	2 days ago

Wrinkles in the Type System



```
$ xcrun swift
Welcome to Apple Swift version 2.1.1
(swiftlang-700.1.101.15 clang-700.1.81). Type :help
for assistance.
  1> nil < "Swift"
$R0: Bool = true
  2> nil < 0
$R1: Bool = true
  3> nil < -Double.infinity
$R2: Bool = true
  4> nil < Double.infinity
repl.swift:4:5: error: type of expression is ambiguous
without more context
nil < Double.infinity
~~~~^~~~~~~~~~~~~~~~


  4> 
```

```
let s = "2"
let x = [1, 2, 3].filter { $0 != Int(s) }
```

```
Array<Optional<Int>>
```

```swift
let s = "2"
let x: [Int] = [1, 2, 3].filter { $0 != Int(s) }
```

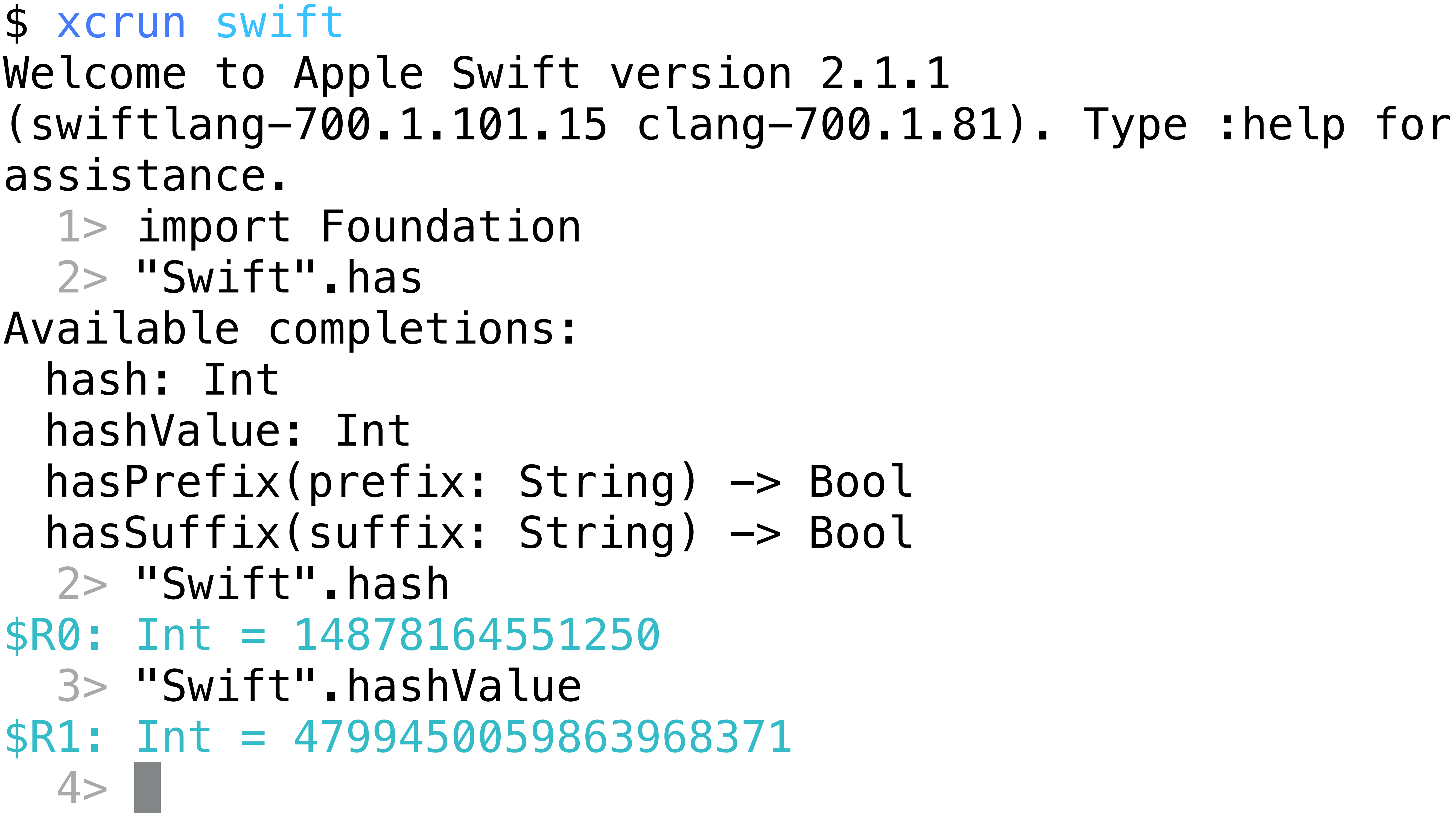
Array<Int>

Type system cleanup and documentation: Revisit and document the various subtyping and conversion rules in the type system, as well as their implementation in the compiler's type checker. The intent is to converge on a smaller, simpler type system that is more rigorously defined and more faithfully represented by the type checker.

Swift 3 Roadmap

https://github.com/apple/swift-evolution/blob/master/README.md

Cocoa vs Swift APIs



```
$ xcrun swift
Welcome to Apple Swift version 2.1.1
(swiftlang-700.1.101.15 clang-700.1.81). Type :help for
assistance.
  1> import Foundation
  2> "Swift".has
Available completions:
 hash: Int
 hashValue: Int
 hasPrefix(prefix: String) -> Bool
 hasSuffix(suffix: String) -> Bool
  2> "Swift".hash
$R0: Int = 14878164551250
  3> "Swift".hashValue
$R1: Int = 4799450059863968371
  4> 
```

API design guidelines: The way in which Swift is used in popular libraries has almost as much of an effect on the character of Swift code as the Swift language itself. The API design guidelines provide guidance for building great Swift APIs. For Swift 3.0, the Swift standard library and core libraries are being updated to match these guidelines, and Swift's Objective-C importer will automatically map from the Cocoa guidelines for Objective-C to the Swift API guidelines.

Swift 3 Roadmap

https://github.com/apple/swift-evolution/blob/master/README.md

```
class UIBezierPath : NSObject, NSCopying, NSCoding {
  convenience init(ovalInRect: CGRect)
  func moveToPoint(_: CGPoint)
  func addLineToPoint(_: CGPoint)
  func addCurveToPoint(_: CGPoint, controlPoint1: CGPoint,
                       controlPoint2: CGPoint)
  func addQuadCurveToPoint(_: CGPoint, controlPoint: CGPoint)
  func appendPath(_: UIBezierPath)
  func bezierPathByReversingPath() -> UIBezierPath
  func applyTransform(_: CGAffineTransform)
  var empty: Bool { get }
  func containsPoint(_: CGPoint) -> Bool
  func fillWithBlendMode(_: CGBlendMode, alpha: CGFloat)
  func strokeWithBlendMode(_: CGBlendMode, alpha: CGFloat)
  func copyWithZone(_: NSZone) -> AnyObject
  func encodeWithCoder(_: NSCoder)
}
```

https://github.com/apple/swift-evolution/blob/master/proposals/0005-objective-c-name-translation.md

```
class UIBezierPath : Object, Copying, Coding {
  convenience init(ovalIn: CGRect)
  func moveTo(_: CGPoint)
  func addLineTo(_: CGPoint)
  func addCurveTo(_: CGPoint, controlPoint1: CGPoint,
                  controlPoint2: CGPoint)
  func addQuadCurveTo(_: CGPoint, controlPoint: CGPoint)
  func append(_: UIBezierPath)
  func reversing() -> UIBezierPath
  func apply(_: CGAffineTransform)
  var isEmpty: Bool { get }
  func contains(_: CGPoint) -> Bool
  func fillWith(_: CGBlendMode, alpha: CGFloat)
  func strokeWith(_: CGBlendMode, alpha: CGFloat)
  func copy(_: Zone = nil) -> AnyObject
  func encodeWith(_: Coder)
}
```

https://github.com/apple/swift-evolution/blob/master/proposals/0005-objective-c-name-translation.md

NSObject

Object

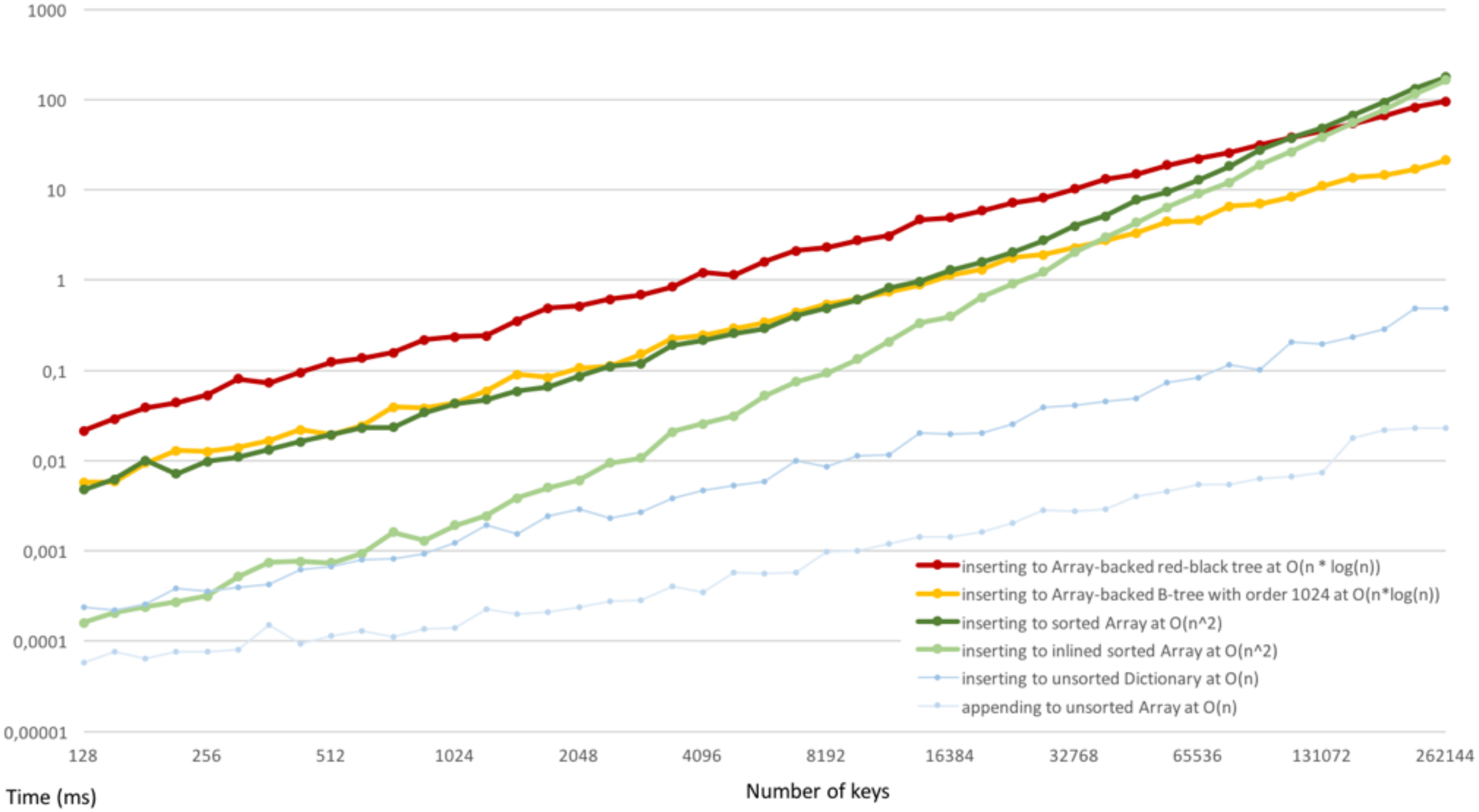
Collection Types with Copy on Write Semantics

COW

```
var a = Array(0 ..< 1_000_000)
a[42] = 23

let b = a
a[42] = 23
```

Inserting random keys into some Swift collections



Generics

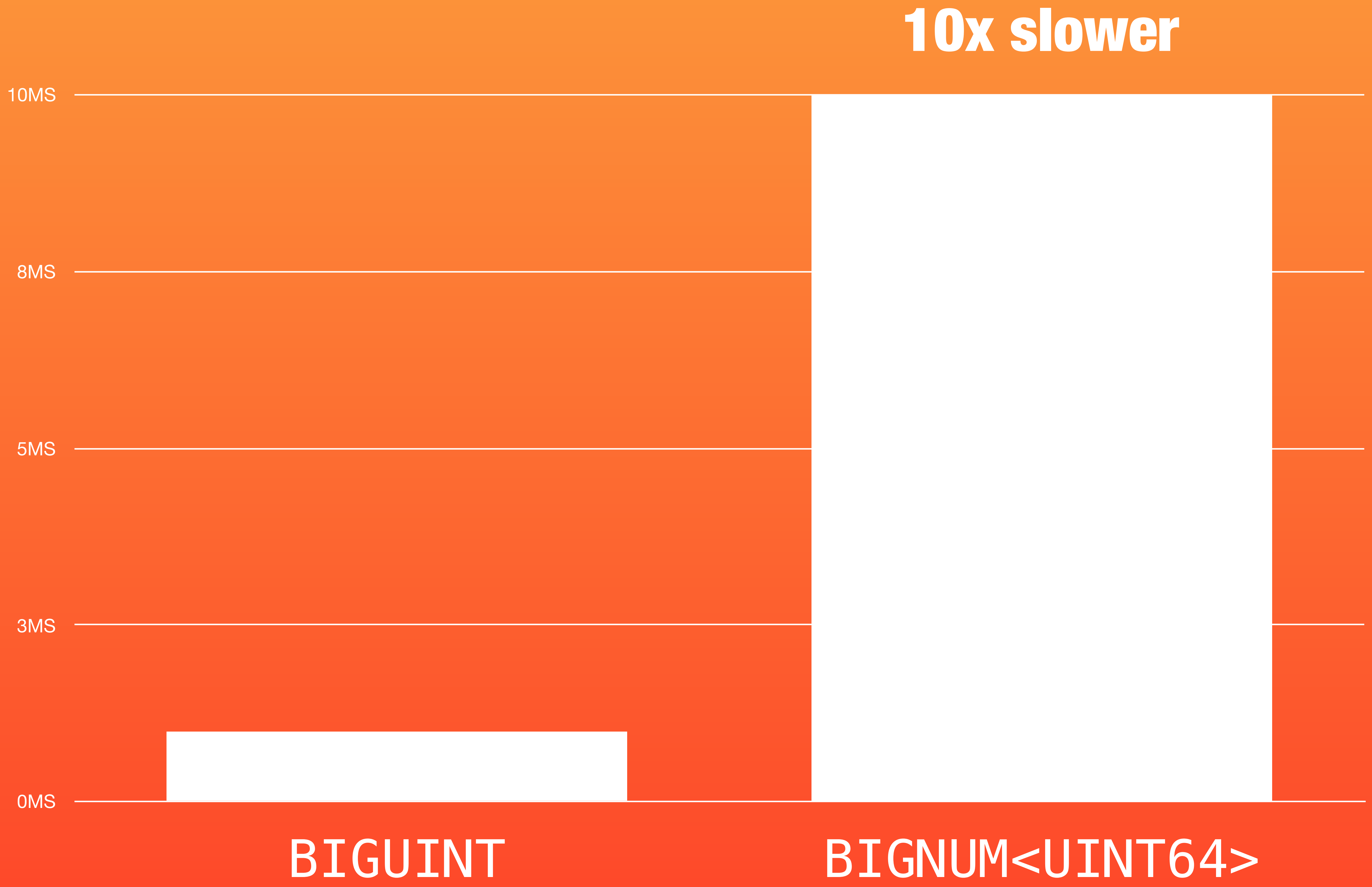
```
public struct BigUInt {
  public typealias Digit = UInt64
  internal var digits: [Digit]

  public mutating func addDigitInPlace(d: Digit) {
    var carry: Digit = d
    var i = 0
    while carry > 0 {
      let (d, c) = Digit.addWithOverflow(self[i], carry)
      self[i] = d
      carry = (c ? 1 : 0)
      i += 1
    }
  }
}
```

```swift
public struct BigNum<Digit: DigitProtocol> {

    internal var digits: [Digit]

    public mutating func addDigitInPlace(d: Digit) {
        var carry: Digit = d
        var i = 0
        while carry > 0 {
            let (d, c) = Digit.addWithOverflow(self[i], carry)
            self[i] = d
            carry = (c ? 1 : 0)
            i += 1
        }
    }
}
public typealias BigUInt = BigNum<UInt64>
```


10x slower
10MS
8MS
5MS
3MS
0MS
BIGUINT
BIGNUM<UINT64>

18 MONTHS

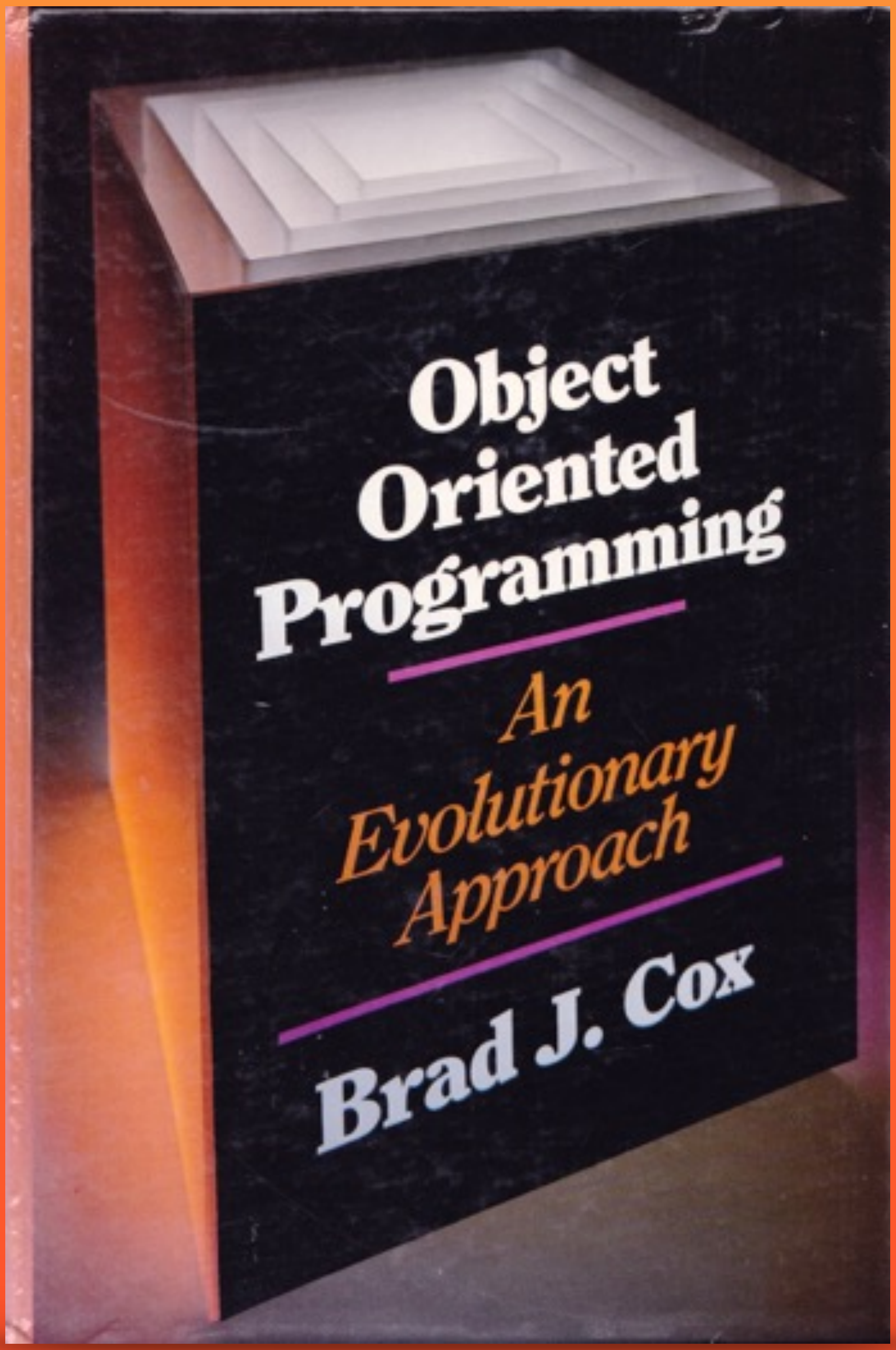
Swift 3

Late 2016

Stable ABI

Resilient ABI

1986



Object Oriented Programming

An Evolutionary Approach

Brad J. Cox

Objective-C

1989

NEXTStep 1.0

2014

2017

Swift

```
var people = ["Dave", "Brian", "Alex", "A
let name = "Alex"
if let index = find(people, name) {
    println("\(name) is person \(index +
    delegate?.didFindPersonWithName(name,
} else {
    println("Unable to find \(name) in th
}
```

?

THANK YOU!

You can follow me on Twitter if you like: @lorentey