

Maximizing Apple Development Resources

Session 415

Nahir Khan
iOS Software Engineer

These are confidential sessions—please refrain from streaming, blogging, or taking pictures

What You'll Learn Today

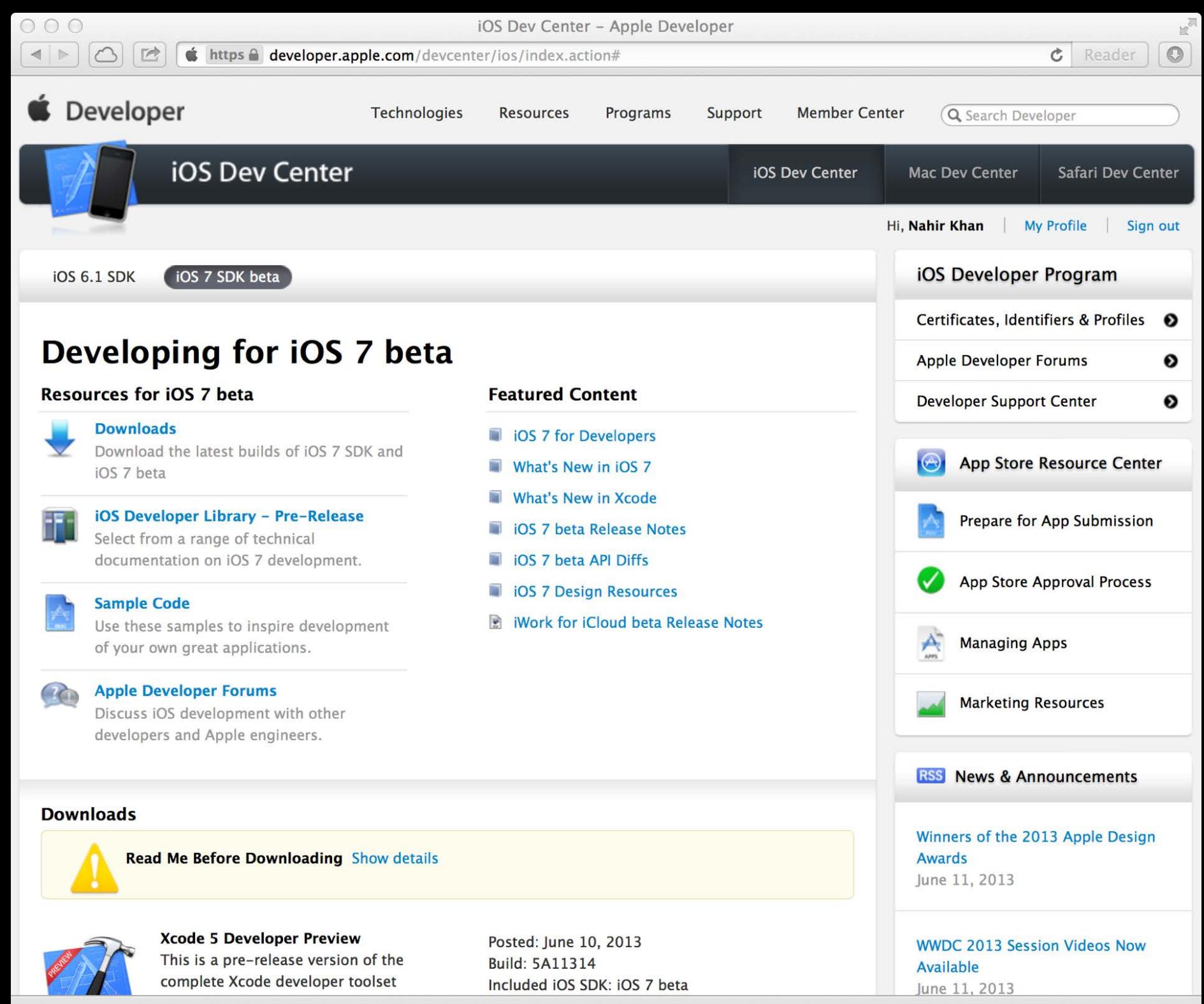
- Self help
 - Documentation
 - Developer Forums
- Working with Apple
 - Bug Reporting
 - Developer Technical Support (DTS)
- Deploying Your App
 - App Store
 - iTunes Connect

Self Help

Documentation

Documentation and More

- Also available from
 - developer.apple.com
- WWDC and Tech Talk videos
- Documentation
 - Getting Started
 - Developer Library
- Sample code and projects



Class Reference

- API reference
- Example code
- Common issues and FAQ

The screenshot shows the iOS Developer Library interface with the title "NSAttributedString Class Reference". The left sidebar includes a "Table of Contents" section with links to Overview, Adopted Protocols, Tasks, Instance Methods, Constants, and Revision History. The main content area displays the "NSAttributedString Class Reference" page, which includes sections for Inherits from (NSObject), Conforms to (NSCoding, NSCopying, NSMutableCopying, NSObject (NSObject)), Framework (/System/Library/Frameworks/Foundation.framework), Availability (Available in iOS 3.2 and later), Companion guide (Attributed String Programming Guide), Declared in (NSAttributedString.h), and Related sample code (CoreTextPageViewer, SimpleTextInput). Below this is the "Overview" section, which provides a detailed description of what an NSAttributedString object does and how it manages character strings and attributes. It also discusses the relationship between NSAttributedString and NSMutableAttributedString, and how attributed strings are used in the Core Text framework.

NSAttributedString Class Reference

Inherits from NSObject

Conforms to NSCoding, NSCopying, NSMutableCopying, NSObject (NSObject)

Framework /System/Library/Frameworks/Foundation.framework

Availability Available in iOS 3.2 and later.

Companion guide Attributed String Programming Guide

Declared in NSAttributedString.h

Related sample code CoreTextPageViewer, SimpleTextInput

Overview

An NSAttributedString object manages character strings and associated sets of attributes (for example, font and kerning) that apply to individual characters or ranges of characters in the string. An association of characters and their attributes is called an attributed string. The cluster's two public classes, NSAttributedString and NSMutableAttributedString, declare the programmatic interface for read-only attributed strings and modifiable attributed strings, respectively.

An attributed string identifies attributes by name, using an NSDictionary object to store a value under the given name. You can assign any attribute name/value pair you wish to a range of characters—it is up to your application to interpret custom attributes (see *Attributed String Programming Guide*). If you are using attributed strings with the Core Text framework, you can also use the attribute keys defined by that framework. In iOS, standard attribute keys are described in the "Constants" section of *NSAttributedString UIKit Additions Reference*. In OS X, standard attribute keys are described in the "Constants" section of *NSAttributedString Application Kit Additions Reference*.

You use attributed strings with any APIs that accept them, such as Core Text. In OS X, the Application Kit also a subclass of NSMutableAttributedString, called NSTextStorage, to provide the storage for the Application Kit's extended text-handling system. In iOS 6 and later you can use attributed strings to display formatted text in text views, text fields, and some other controls. Both AppKit and UIKit also define extensions to the basic attributed string interface that allows you to draw their contents in the current graphic context.

The default font for NSAttributedString objects is Helvetica 12-point, which may differ from the default system font for the platform. Thus, you might want to create new strings with non-default attributes suitable for your application. You can also use the NSParagraphStyle class and its subclass NSMutableParagraphStyle to encapsulate the paragraph or ruler attributes used by the NSAttributedString classes.

Be aware that comparisons of NSAttributedString objects using the `isEqual:` method look for exact equality. The comparison includes both a character-by-character string equality check and an equality check of all attributes. Such a comparison is not likely to yield a match if the string has many attributes, such as attachments, lists, and tables, for example.

Full Programming Guides

- Entire books worth of content

The screenshot shows a web browser displaying the "Text Programming Guide for iOS" from the "iOS Developer Library — Pre-Release". The page has a dark header with the Apple logo and a search icon. The main content area has a light gray background. On the left, there's a sidebar with a "Table of Contents" section containing links like "Introduction", "Text Handling in iOS", "Typographical Concepts", etc., and an "OBJECTIVE-C REFERENCE" section with links to "NSLayoutManager", "NSTextAttachment", etc. The main content area features a large heading "About Text Editing in iOS". Below it is a blue-bordered box containing an "Important" note about the document being preliminary and subject to change. Further down, there's a paragraph about iOS's text handling capabilities and a "Note" box at the bottom.

iOS Developer Library — Pre-Release

Text Programming Guide for iOS

PDF

Table of Contents

Introduction

Text Handling in iOS

Typographical Concepts

Managing Text Fields and Text Views

Displaying Web Content

Managing the Keyboard

Copy, Cut, and Paste Operations

Custom Views for Data Input

Displaying and Managing the Edit Menu

Drawing and Managing Text Revision History

OBJECTIVE-C REFERENCE

NSLayoutManager

NSLayoutManagerDelegate

NSString UIKit Additions Reference

NSTextAttachment

NSTextAttachmentContainer

About Text Editing in iOS

Important: This is a preliminary document for an API or technology in development. Although this document has been reviewed for technical accuracy, it is not final. This Apple confidential information is for use only by registered members of the applicable Apple Developer program. Apple is supplying this confidential information to help you plan for the adoption of the technologies and programming interfaces described herein. This information is subject to change, and software implemented according to this document should be tested with final operating system software and final documentation. Newer versions of this document may be provided with future seeds of the API or technology.

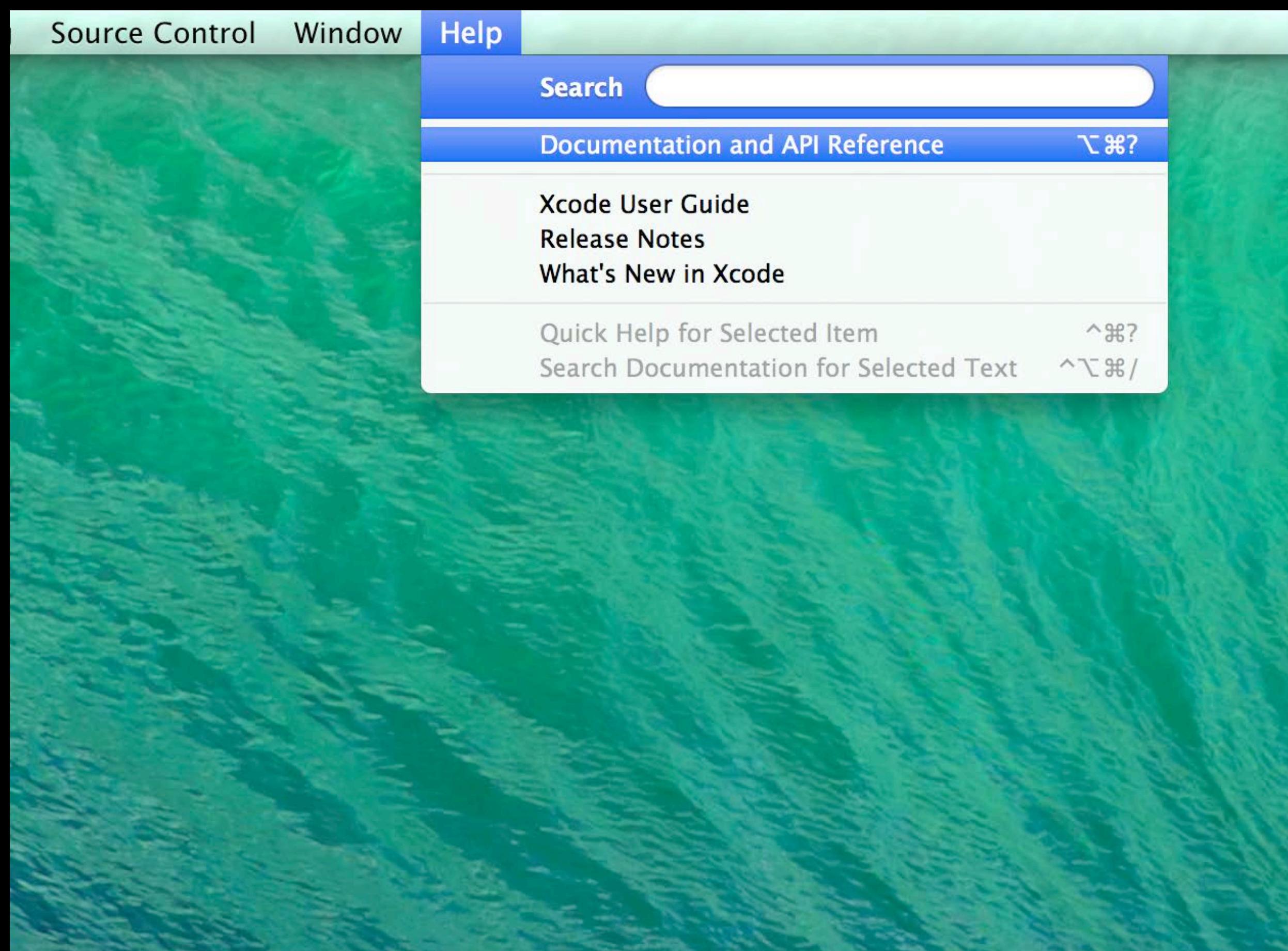
iOS gives you many ways to display text in your applications and let users edit that text. It also lets you display typographically formatted text and web content in your application's views. The resources at your disposal range from [framework](#) objects such as text views, text fields, and web views that allow your application to draw, lay out, and otherwise manage text.

With the classes in the UIKit framework, you can manage the edit menu (including adding custom items to it), implement custom input views, and copy, cut, and paste data within and between applications.

Note: This document was previously titled *Text, Web, and Editing Programming Guide for iOS*.

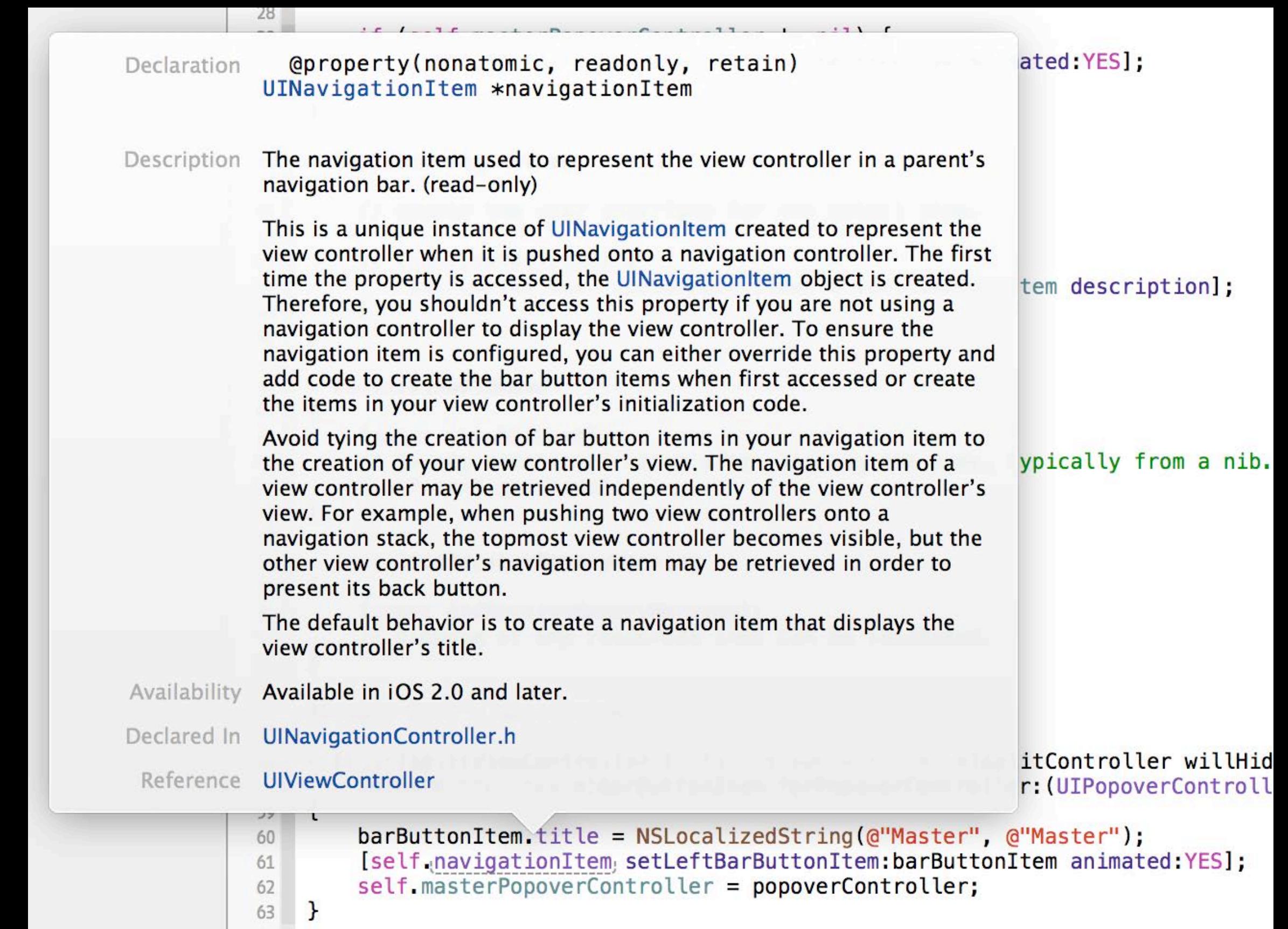
Documentation and API Reference

- Right from Xcode



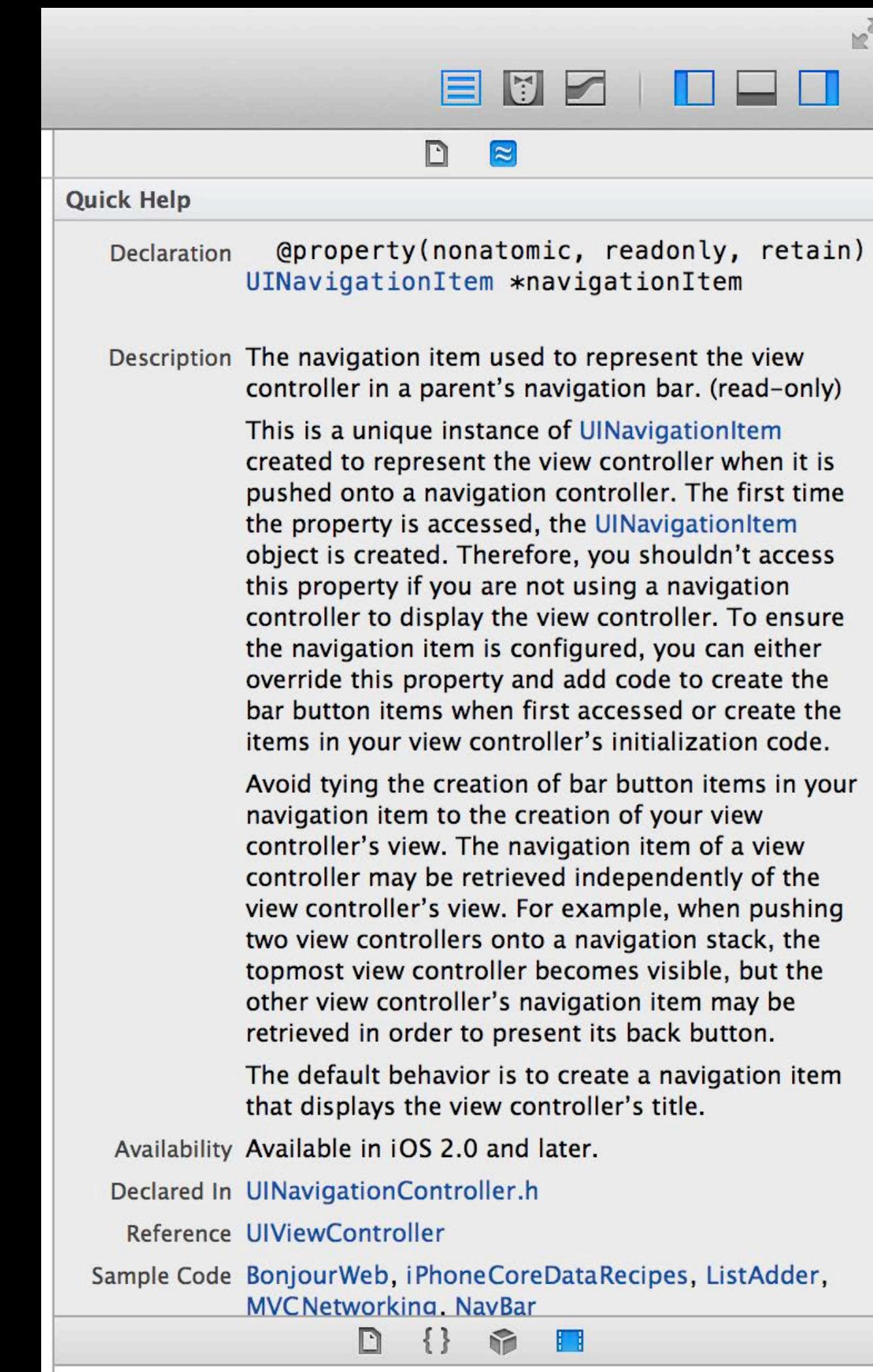
Symbol Definition

- Option-click on any symbol
- Quick description
- Access to header
- Click Reference to get full guide



Symbol Definition (Cont.)

- Select any symbol
- Quick access in sidebar to
 - Headers
 - API reference
 - Programming guides
 - Sample code



Feedback

- Bottom of every page
- Can report
 - Typos
 - Missing information
 - Confusing
 - Helpful

The screenshot shows a section of the Apple Developer Documentation. At the top, it asks if GC (Garbage Collection) is deprecated on the Mac. Below that, it provides a note about the deprecation of Garbage Collection in OS X Mountain Lion v10.8 and the recommendation to use ARC instead. A yellow box highlights the feedback section at the bottom.

Is GC (Garbage Collection) deprecated on the Mac?

Garbage collection is deprecated in OS X Mountain Lion v10.8, and will be removed in a future version of OS X. Automatic Reference Counting is the recommended replacement technology. To aid in migrating existing applications, the ARC migration tool in Xcode 4.3 and later supports migration of garbage collected OS X applications to ARC.

Note: For apps targeting the Mac App Store, Apple strongly recommends you replace garbage collection with ARC as soon as feasible, because Mac App Store guidelines (see *App Store Review Guidelines for Mac Apps*) prohibit the use of deprecated technologies.

Next

© 2012 Apple Inc. All Rights Reserved. (Last updated: 2012-07-17)

Did this document help you? **Yes** **It's good, but...** **Not helpful...**

Developer Forums

Developer Forums

- Visit the forums at:
 - devforums.apple.com

The screenshot shows the Apple Developer Forums homepage. At the top, there's a banner for "WWDC Discussions Open to All Registered Developers". Below it, the "iOS Development" section is visible, featuring a "Recent Posts" list with three entries:

- Re: Will downloading the iOS 7 beta on my iphone 5 hurt or mess up my phone - by Colin Schoen (less than a minute ago)
- Re: how to bold some parts of the text in UITextView? - by KMT (less than a minute ago)
- Re: iOS7 - iMessage activation - by Jake Edward Quinones (less than a minute ago)

On the right side, there's a sidebar with "Quick Links" to Documentation and Downloads (including links to the iOS Dev Center, Mac Dev Center, Safari Dev Center, App Store Resource Center, App Review Guidelines, iTunes Connect, and Licensing and Trademarks), Membership (Joining a Program, Account Summary, Membership Agreements), and Help and Support (Get Forums Help, Ask Questions, Forums Guidelines, Request Technical Support, Report Bugs / Suggest Features, Contact Support). The "Mac Development" section is also partially visible below the iOS one.

Developer Forums

- Peer support
- Help with triaging your problem
- Help other developers
- Apple employees
- Public

The screenshot shows a forum post titled "README: About Mac Debugging" by user DrErnie. The post discusses general techniques and challenges with debugging on the Mac. Below the post, a reply from user Hakime SEDDIK says "Great, thanks a lot for that!".

Developer Forums

README: About Mac Debugging

May 1, 2013 5:29 PM

This is where you can discuss general techniques and challenges with debugging on the Mac, particularly problems that don't fit into other forums.

Related Forums

- Developer Tools -> Xcode -> [Debugging](#) (about using Xcode for Debugging)
- Developer Tools -> [Command-Line Development](#) (for command-line debugging tools)

Resources

- [Technical Note TN2124: Mac OS X Debugging Magic](#)
- [Technical Q&A QA1573: Debugging Process Startup](#)
- [Technical Note TN2063: Understanding and Debugging Kernel Panics](#)
- [Debugging – Technical Support – Apple Developer](#)

NOTE: This area was previously called 'Mac Development -> Developer Tools'. Going forward, please use the [Developer Tools](#) and [Xcode](#) areas for more general discussions of how to use Apple's developer tools.

1. Re: README: About Mac Debugging

In response to [DrErnie](#) on May 2, 2013 7:33 AM

Great, thanks a lot for that!

Quick Links

- Documentation and Downloads
- iOS Dev Center
- Mac Dev Center
- Safari Dev Center
- App Store Resource Center
- App Review Guidelines
- iTunes Connect
- Licensing and Trademarks

Membership

- Joining a Program
- Account Summary
- Membership Agreements

Help and Support

- Get Forums Help
- Ask Questions
- Forums Guidelines
- Request Technical Support
- Report Bugs / Suggest Features
- Contact Support

Example Post

Developer Forums

Why calls retain in setUp() method of iPhoneUnitTests de... [Quote](#) [Reply](#)

Apr 18, 2013 1:39 AM [Report](#) [Email Updates](#)

In Apple demo project "iPhoneUnitTests", CalculatorLogicTests.m Line 64 – 68:

```
- (void) setUp {
    NSLog(@"%@", self.name);
    calculator = [[[Calculator alloc] init] retain];
    STAssertNotNil(calculator, @"Cannot create Calculator instance");
}
```

We get an instance of Calculator named calculator by

```
[[[Calculator alloc] init] retain];
```

But according to Memory Management Guide, it shouldn't call retain or the retainCount of calculator becomes 2.

I wonder if it is right and why Apple does this here ?

Anyone knows?

Thanks.

Message was edited by brunic on 13-4-18 at 上午1:39

1. Re: Why calls retain in setUp0 method of iPhoneUnitTests demo? [Quote](#) [Reply](#)

In response to **brunic** on Apr 19, 2013 11:43 AM [Report](#) [Email Updates](#)

Seems like a bug. Looking at the latest version of the sample however, it seems this bug was fixed.

2. Re: Why calls retain in setUp() method of iPhoneUnitTests demo? [Quote](#) [Reply](#)

In response to **Rincewind** on Apr 24, 2013 7:19 PM [Report](#) [Email Updates](#)

Thanks, Rincewind.

Quick Links

Documentation and Downloads

- iOS Dev Center
- Mac Dev Center
- Safari Dev Center
- App Store Resource Center
- App Review Guidelines
- iTunes Connect
- Licensing and Trademarks

Membership

- Joining a Program
- Account Summary
- Membership Agreements

Help and Support

- Get Forums Help
- Ask Questions
- Forums Guidelines
- Request Technical Support
- Report Bugs / Suggest Features
- Contact Support

Example Post

Developer Forums

Why calls retain in setUp() method of iPhoneUnitTests de... [Quote](#) [Reply](#)

Apr 18, 2013 1:39 AM [Report](#) [Email Updates](#)

In Apple demo project "iPhoneUnitTests", CalculatorLogicTests.m Line 64 – 68:

```
- (void) setUp {
    NSLog(@"%@", self.name);
    calculator = [[[Calculator alloc] init] retain];
    STAssertNotNil(calculator, @"Cannot create Calculator instance");
}
```

We get an instance of Calculator named calculator by

```
[[[Calculator alloc] init] retain];
```

But according to Memory Management Guide, it shouldn't call retain or the retainCount of calculator becomes 2.

I wonder if it is right and why Apple does this here ?

Anyone knows?

Thanks.

Message was edited by brunic on 13-4-18 at 上午1:39

1. Re: Why calls retain in setUp0 method of iPhoneUnitTests demo? [Quote](#) [Reply](#)

In response to **brunic** on Apr 19, 2013 11:43 AM [Report](#) [Email Updates](#)

Seems like a bug. Looking at the latest version of the sample however, it seems this bug was fixed.

2. Re: Why calls retain in setUp() method of iPhoneUnitTests demo? [Quote](#) [Reply](#)

In response to **Rincewind** on Apr 24, 2013 7:19 PM [Report](#) [Email Updates](#)

Thanks, Rincewind.

Quick Links

Documentation and Downloads

- iOS Dev Center
- Mac Dev Center
- Safari Dev Center
- App Store Resource Center
- App Review Guidelines
- iTunes Connect
- Licensing and Trademarks

Membership

- Joining a Program
- Account Summary
- Membership Agreements

Help and Support

- Get Forums Help
- Ask Questions
- Forums Guidelines
- Request Technical Support
- Report Bugs / Suggest Features
- Contact Support

Example Post

The screenshot shows a forum post titled "Why calls retain in setUp() method of iPhoneUnitTests de...". The post was made by user **brunic** on April 18, 2013, at 1:39 AM. The code snippet in the post shows a call to `retain` on a newly allocated `Calculator` instance.

```
- (void) setUp {
    NSLog(@"%@", self.name);
    calculator = [[[Calculator alloc] init] retain];
    STAssertNotNil(calculator, @"Cannot create Calculator instance");
}
```

The post continues with a question about memory management and ends with a message edited on April 18, 2013, at 1:39 AM.

Below the original post, user **Rincewind** (highlighted with a yellow box) responded on April 19, 2013, at 11:43 AM, stating it seems like a bug. A second reply from **brunic** on April 24, 2013, at 7:19 PM, expresses thanks to Rincewind.

The right sidebar contains "Quick Links" to various Apple developer resources and "Help and Support" links.

Quick Links

- Documentation and Downloads
- iOS Dev Center
- Mac Dev Center
- Safari Dev Center
- App Store Resource Center
- App Review Guidelines
- iTunes Connect
- Licensing and Trademarks

Membership

- Joining a Program
- Account Summary
- Membership Agreements

Help and Support

- Get Forums Help
- Ask Questions
- Forums Guidelines
- Request Technical Support
- Report Bugs / Suggest Features
- Contact Support

Help from Apple

Bug Reporting

Tanya Gupta
iOS Software Engineer

Why should I file a bug report?
Help Us Help You

Should I File?

Should I File?

- The answer is always “Yes”, even if the issue:
 - Seems trivial

Should I File?

- The answer is always “Yes”, even if the issue:
 - Seems trivial
 - Happened once

Should I File?

- The answer is always “Yes”, even if the issue:
 - Seems trivial
 - Happened once
 - Seems obvious

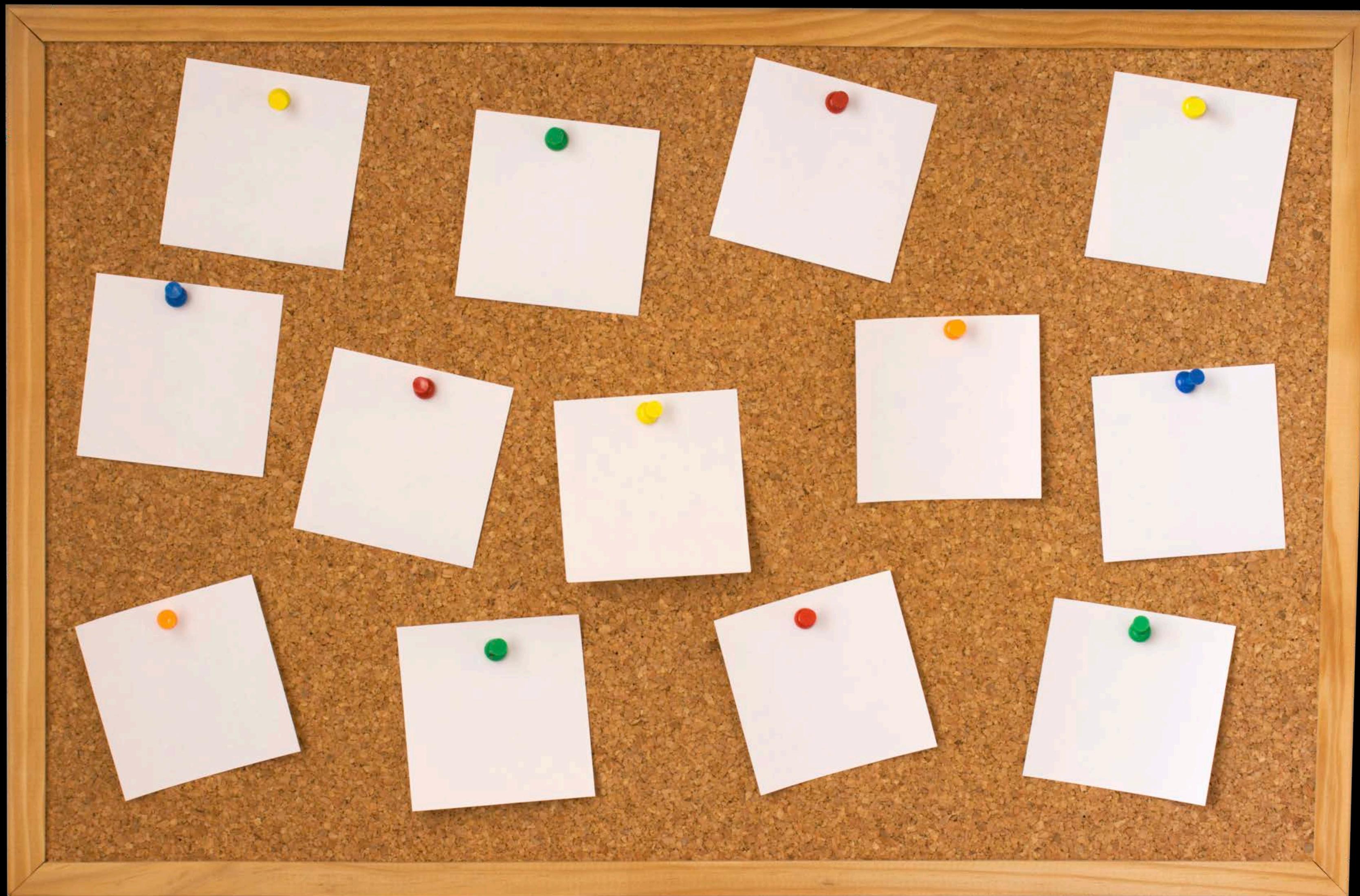
Should I File?

- The answer is always “Yes”, even if the issue:
 - Seems trivial
 - Happened once
 - Seems obvious
 - Someone else filed it

**“If an issue does not exist in
bug reporter, it does not exist.”**

Apple Engineering Mantra

Task List for Apple Engineering



“It’s after 5pm, and there’s no caffeine to be found on campus. This is a serious impediment to engineering performance.”

An example bug report

“It's after 5pm, and there's no canine to be found on campus. This is a serious impediment to engineering performance.”



An example bug report

“Enter elevator #11. Press the button for floor 3 on the panel on right side of the elevator doors. Only the left panel 3rd floor button lights up.”

An example bug report

“Recently, when I go to get some dinner in the cafeteria, the Odwalla cabinet is locked. I am forced to eat ice cream and Red Bull instead.”

An example bug report

How do I file a bug report? bugreport.apple.com

Problem Description:

Please provide the necessary information that will enable us to understand and reproduce this reported issue. Be sure to properly identify the product, severity, frequency and the current version of the product you are using. For example, to find the build number for the version for Mac OS X, select "About this Mac" from the Apple menu and click the version number listed (i.e. 10.x.x).

Note: If this report is a potential security issue, please select 'Security' as the classification.

Problem Report Title:

Product: **Version/Build Number:**

Classification: **Is It Reproducible?**

Problem Details:
Please use the [description format](#) required by Apple Engineering.

Summary:

Steps to Reproduce:

Expected Results:

Actual Results:

Regression:

Notes:

One Report = One Issue

- The new seed is completely broken

These are the problems I encountered with the new seed

- Problem 1: ...
- Problem 2: ...
- Problem 3: ...

What Do I File a Bug Report For?

What Do I File a Bug Report For?



Bugs

What Do I File a Bug Report For?



Bugs



Crashes, Hangs,
Data Loss

What Do I File a Bug Report For?



Bugs



Crashes, Hangs,
Data Loss



Performance
Issues

What Do I File a Bug Report For?



Bugs



Crashes, Hangs,
Data Loss



Performance
Issues



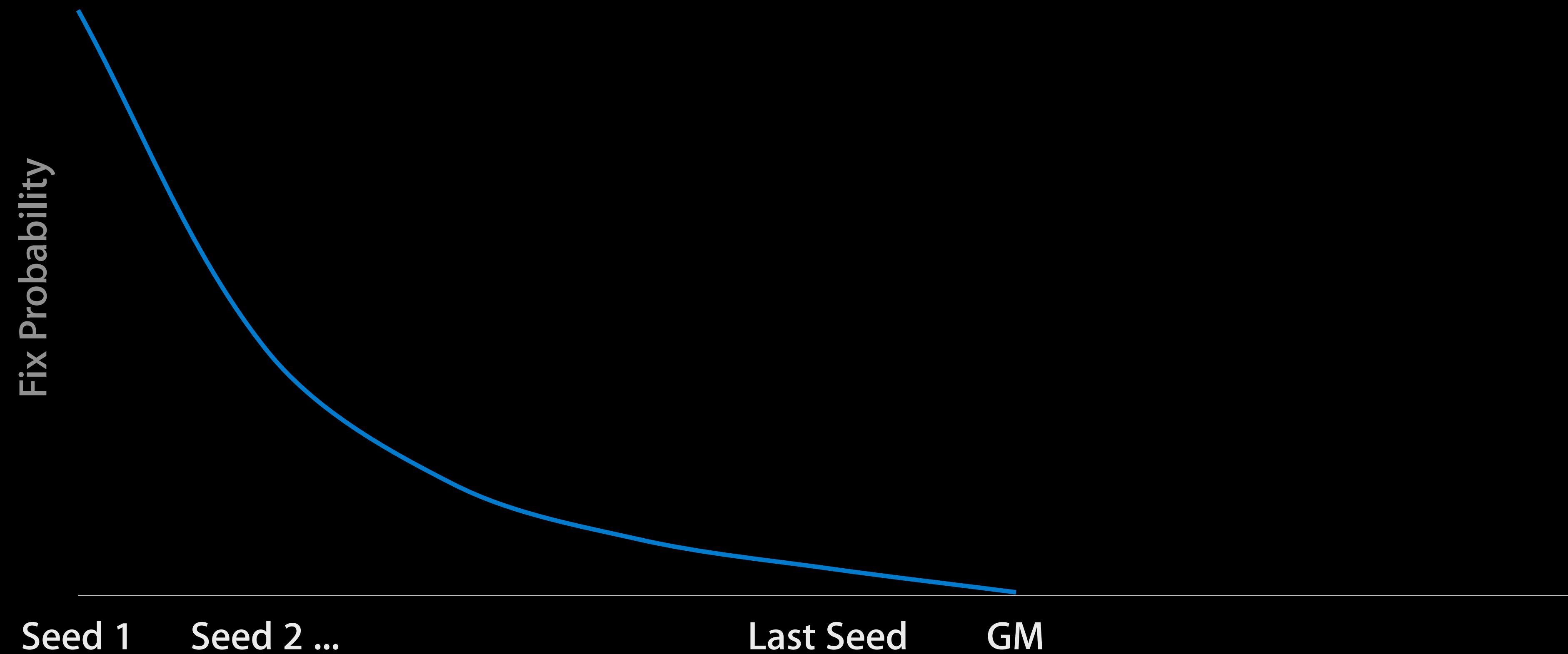
Enhancement
Requests

When do I file the bug report?

As Early as Possible

Timing Is Key

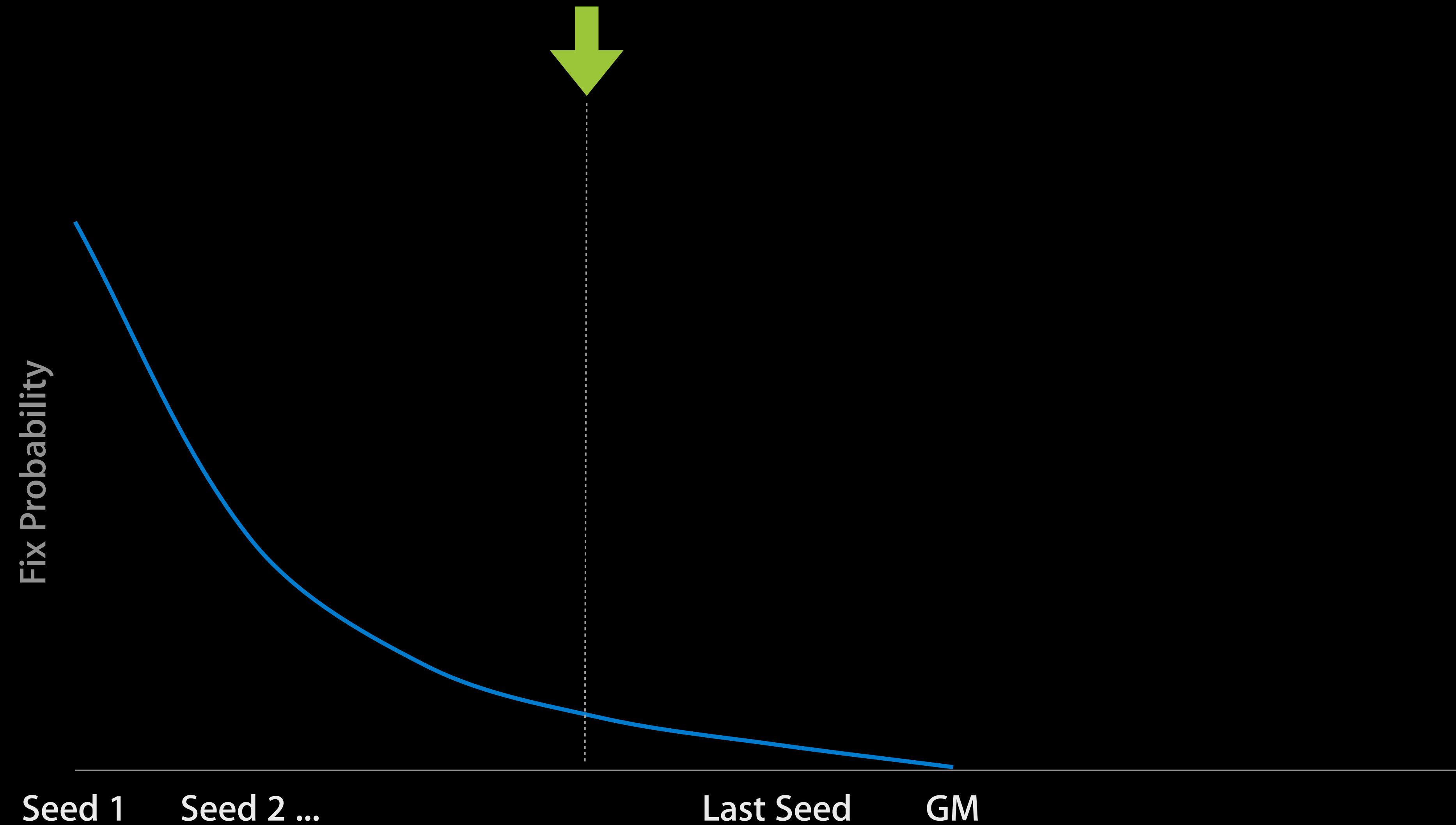
Timing Is Key



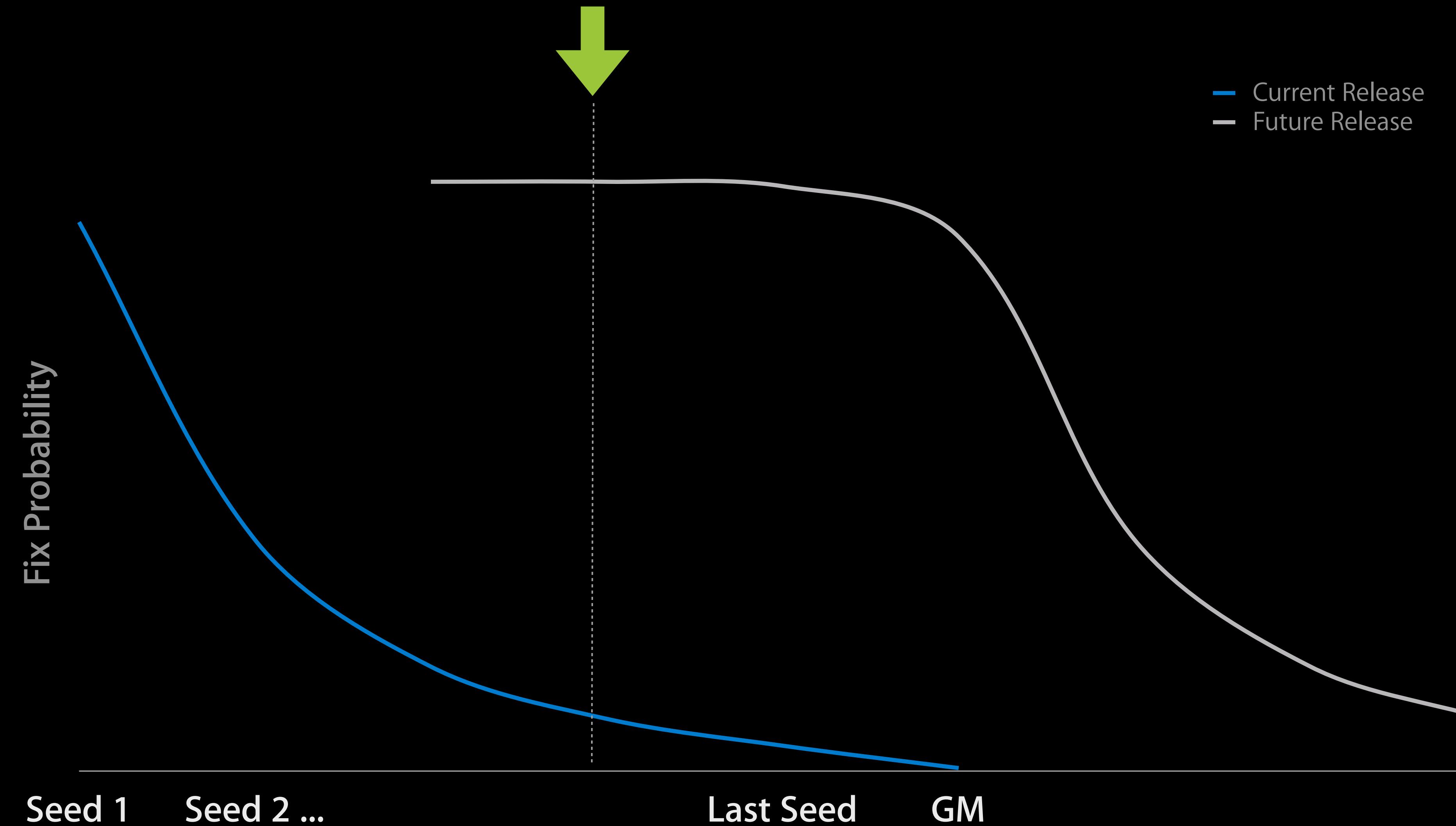
Timing Is Key



Timing Is Key



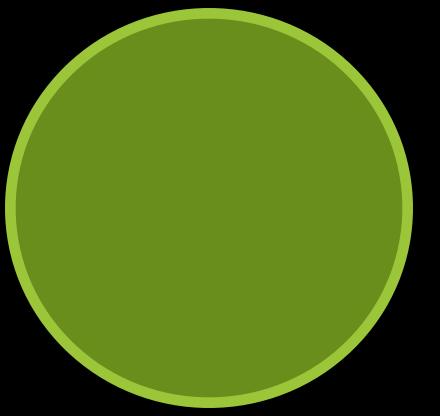
Timing Is Key



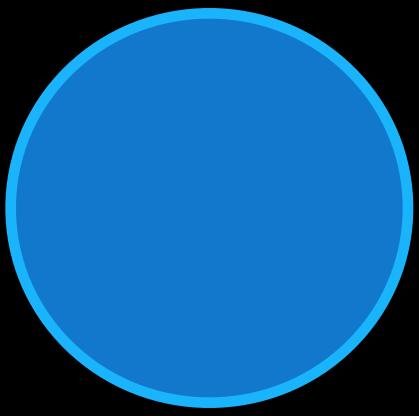
Life of the Bug

Analyze

You

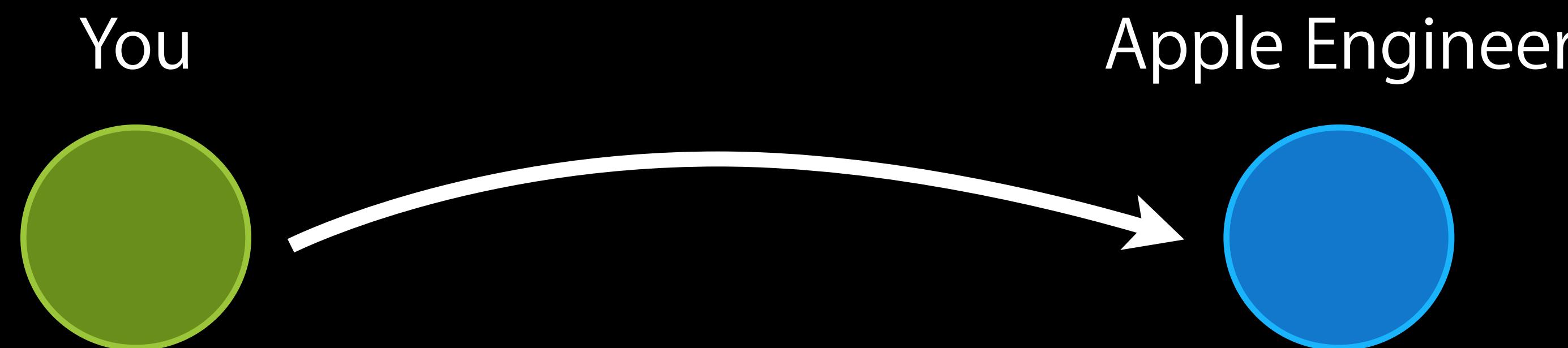


Apple Engineer



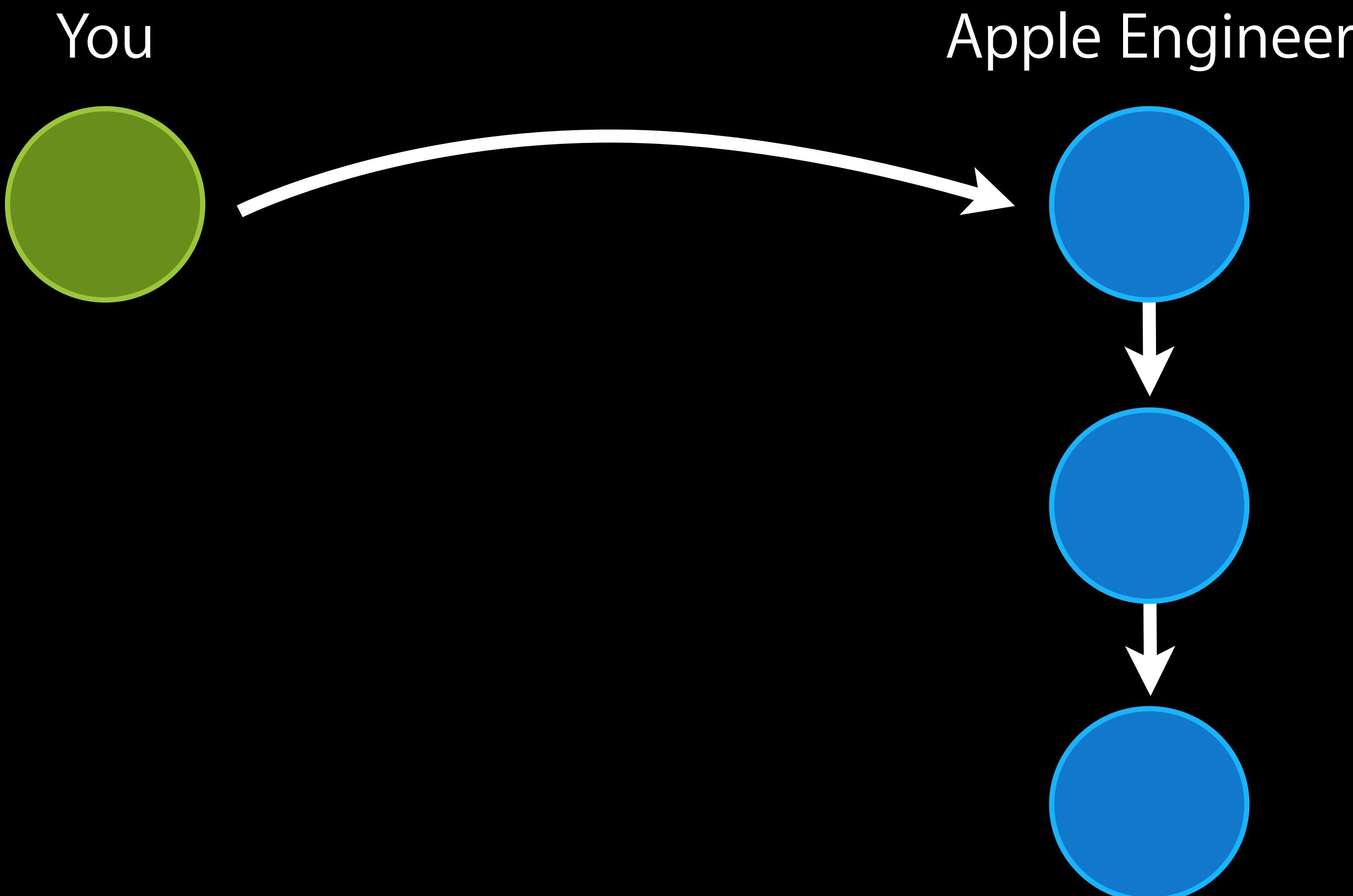
Life of the Bug

Analyze



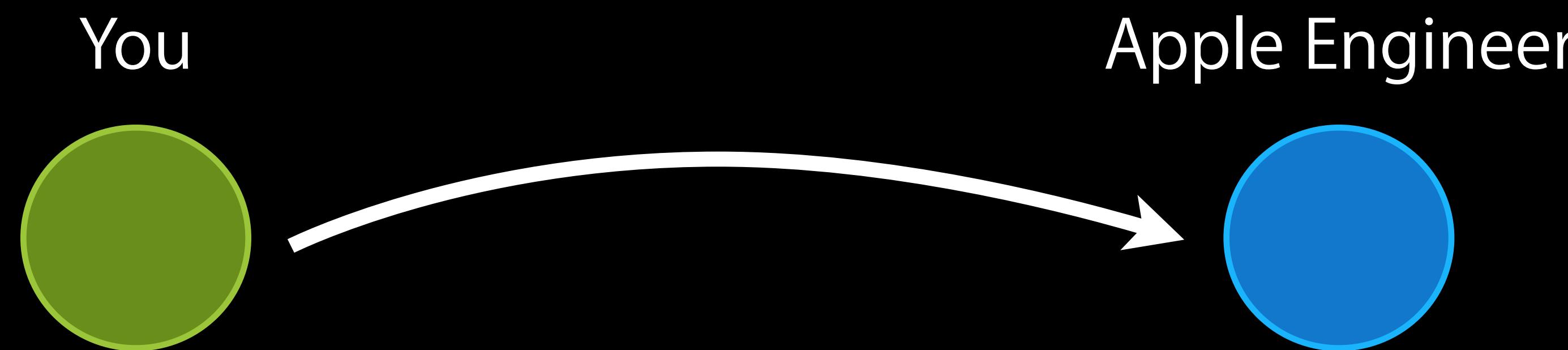
Life of the Bug

Analyze



Life of the Bug

Analyze



Life of the Bug

Software changed



Life of the Bug

Duplicate



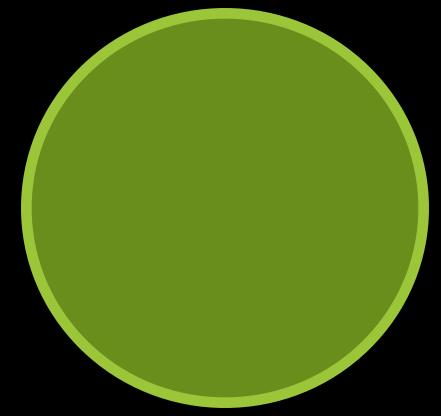
Duplicates Are Important

- Better information than the original
- Number of duplicates as seriousness indicator
- Request status on the original bug

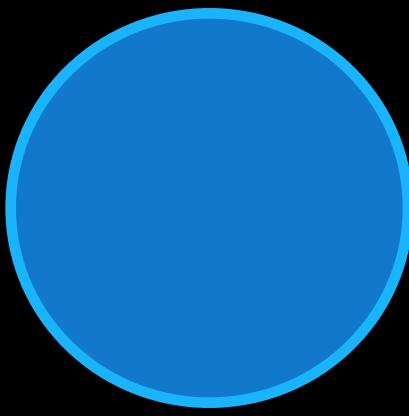
Life of the Bug

Insufficient information

You

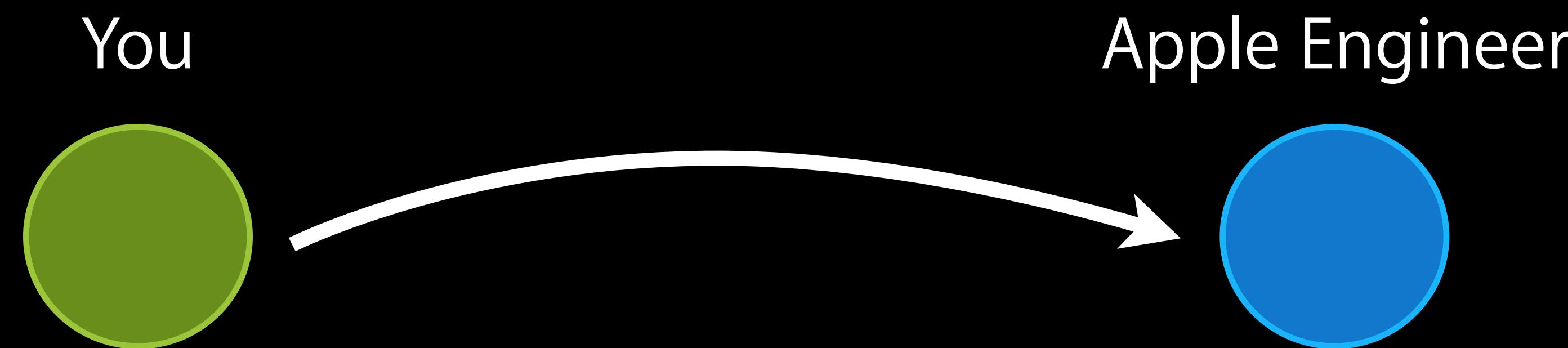


Apple Engineer



Life of the Bug

Insufficient information

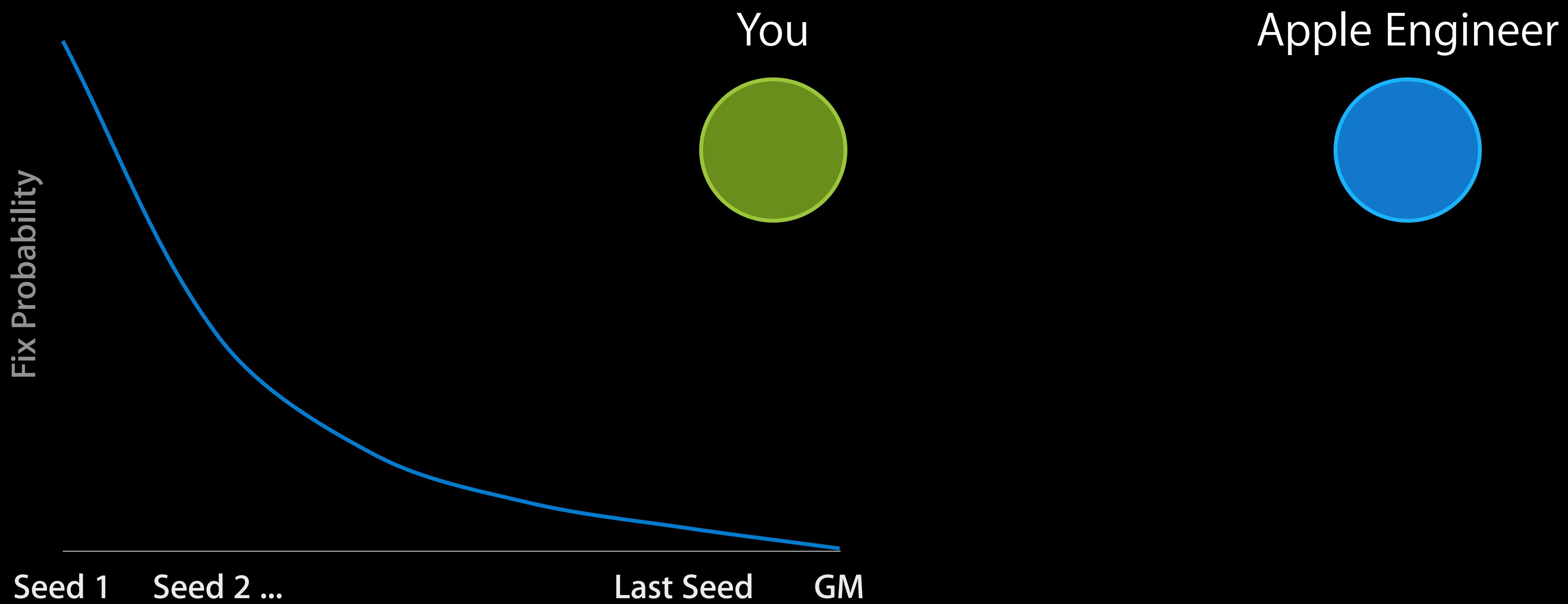


Life of the Bug

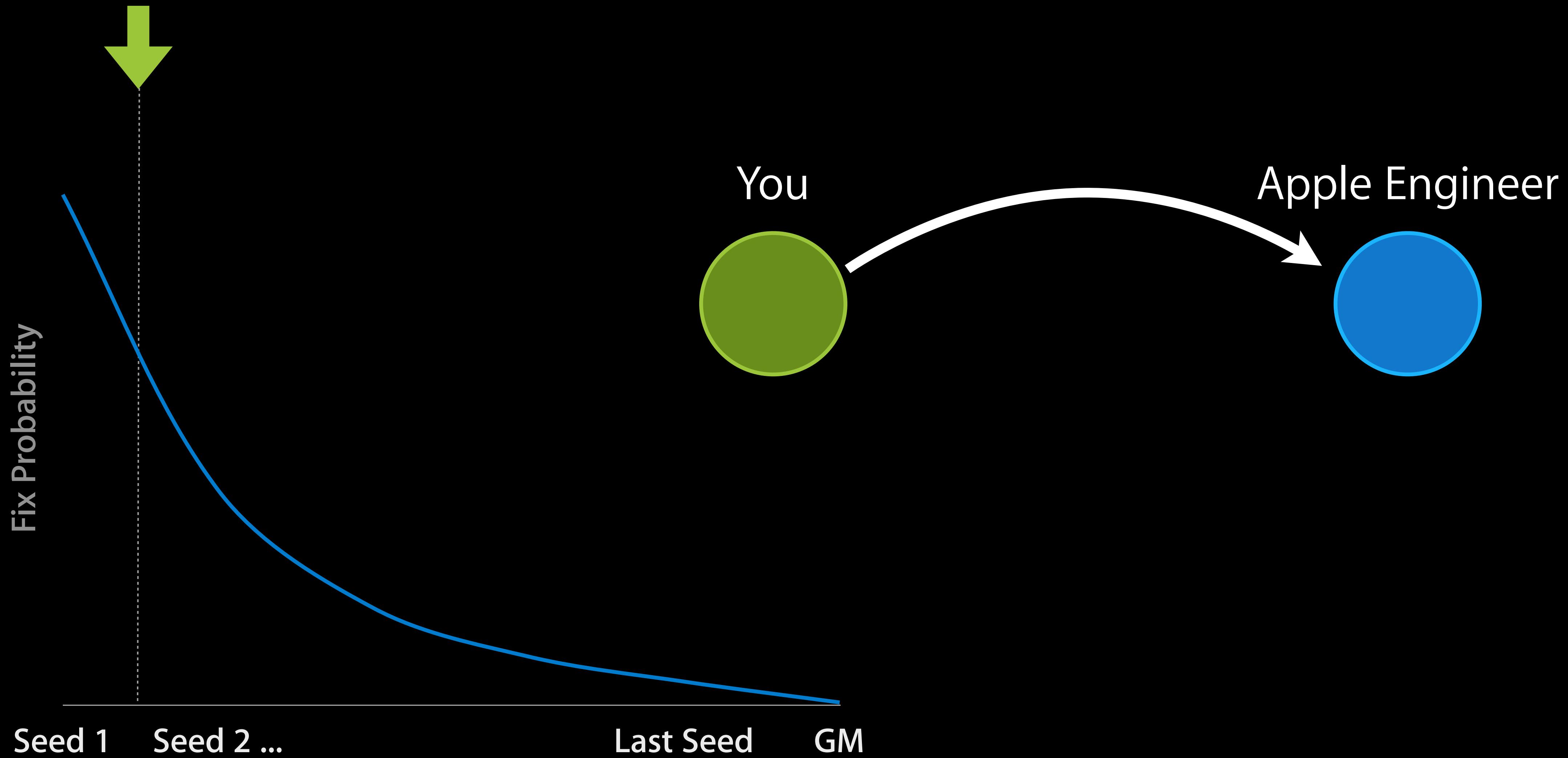
Insufficient information



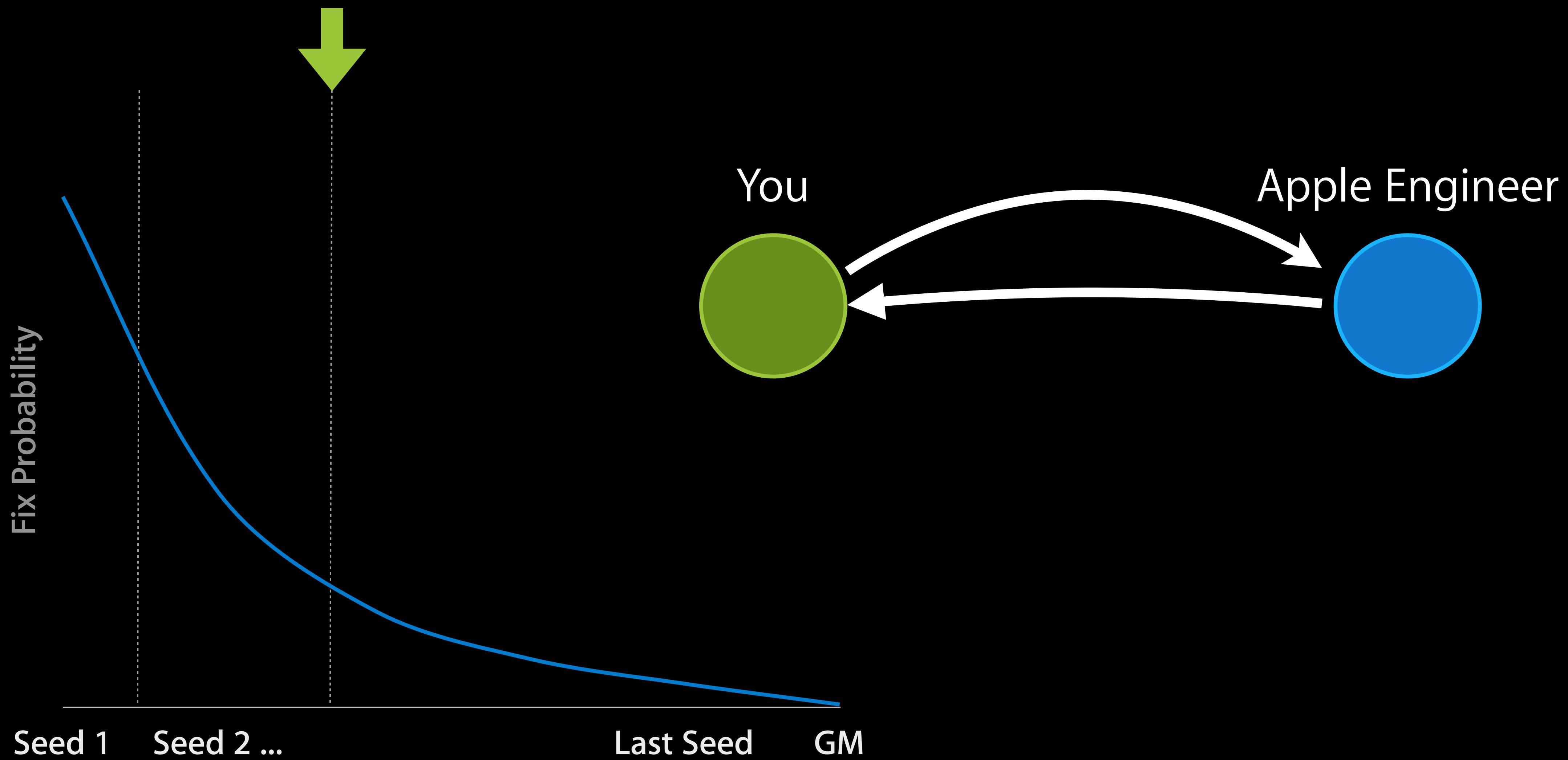
Insufficient Information Hazards



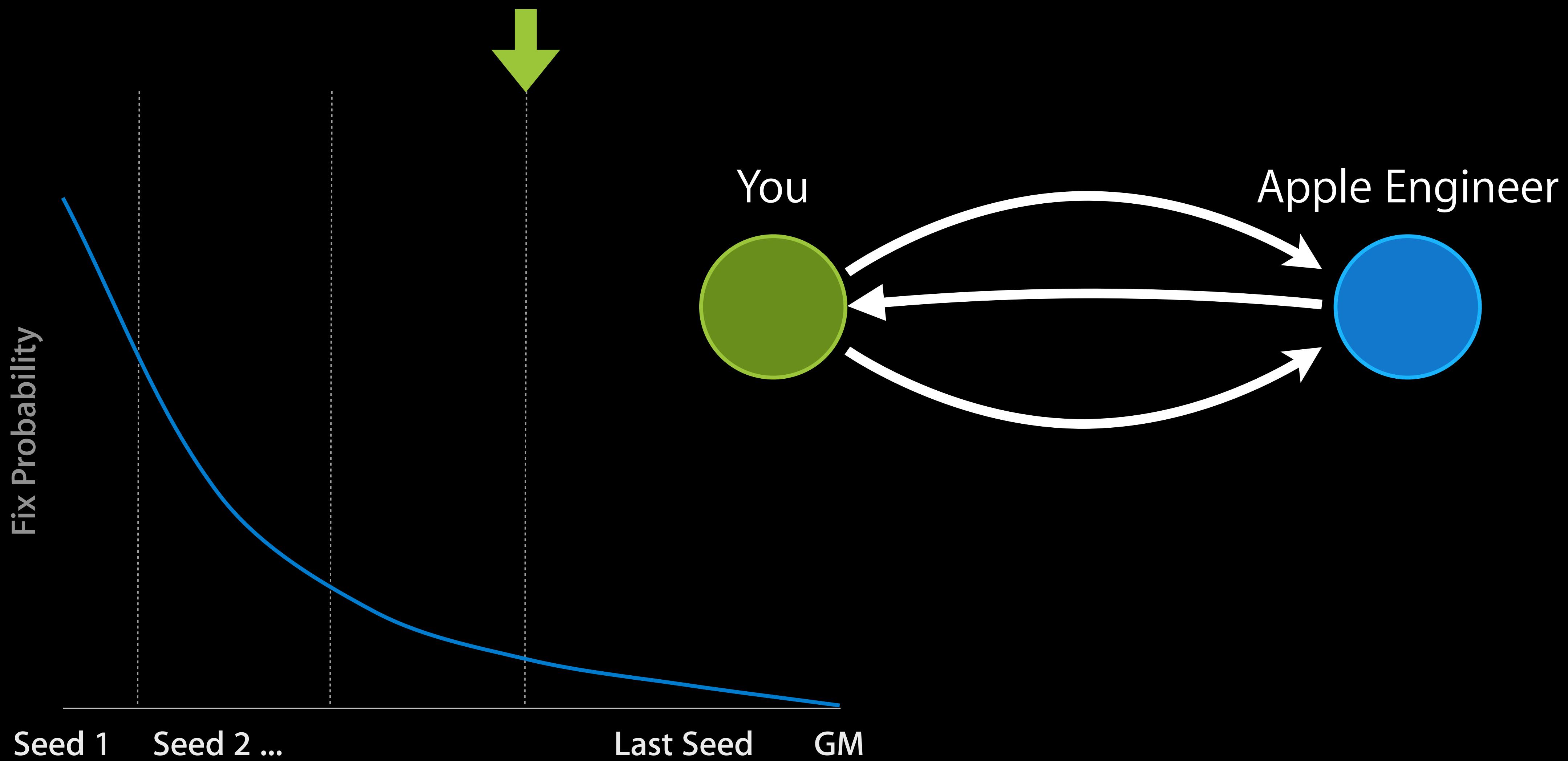
Insufficient Information Hazards



Insufficient Information Hazards



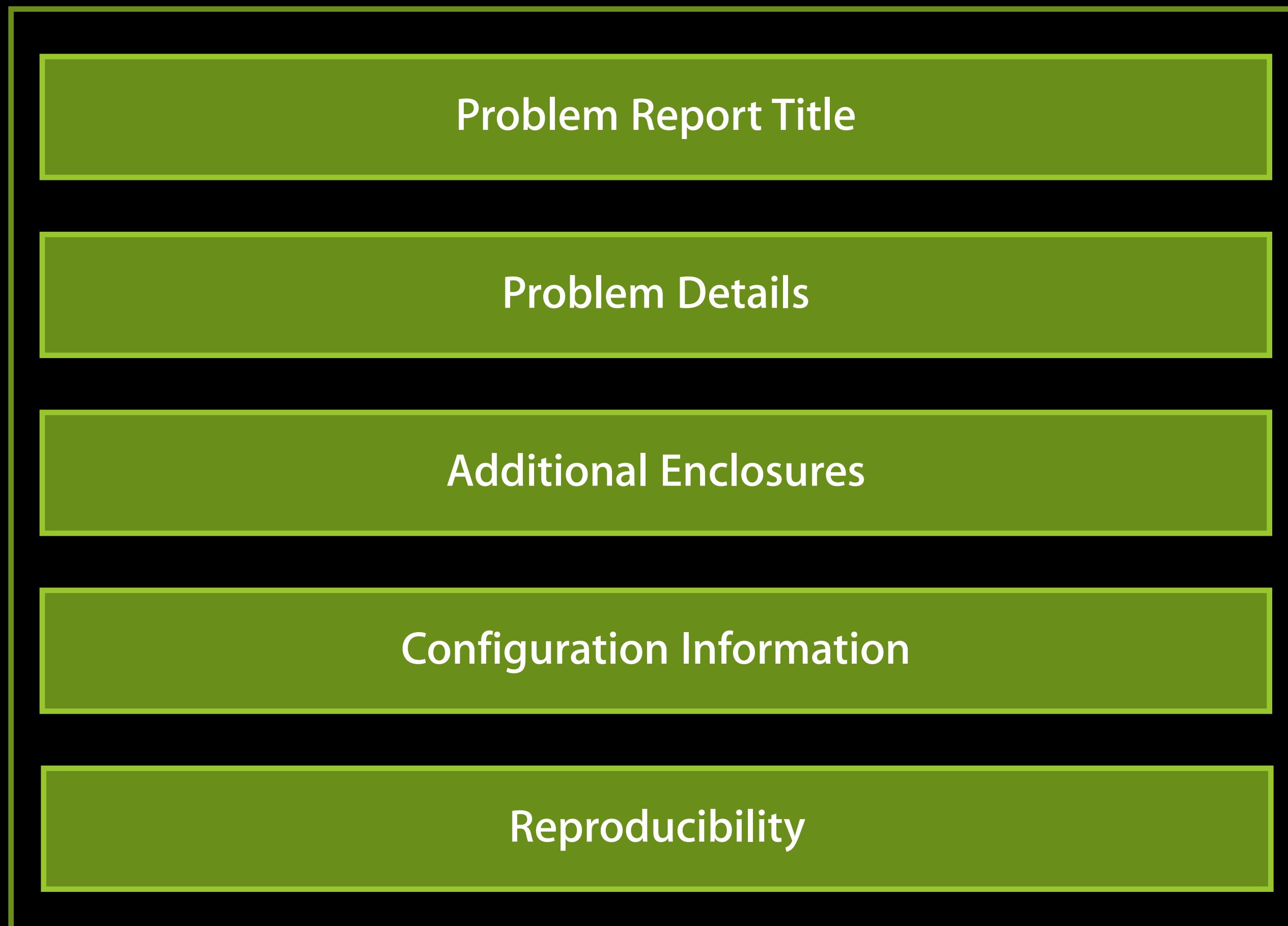
Insufficient Information Hazards



Write Actionable Reports

- Clear
- Concise
- Informative
- Constructive
- Specific

Components of a Bug Report



Components of a Bug Report



Problem Report Title

- Advertisement of your issue
- Clear and succinct
- Should stand on its own

An Example

An Example

Copy and paste is broken

An Example

Copy and ~~paste~~ is broken

Pasting an item stored using [UIPasteboard setImage:] is broken

An Example

Copy and ~~paste~~ is broken

Pasting an item stored using ~~[UIPasteboard setImage:]~~ is broken

Pasting an item stored using [UIPasteboard setImage:]
produces images with question marks

An Example

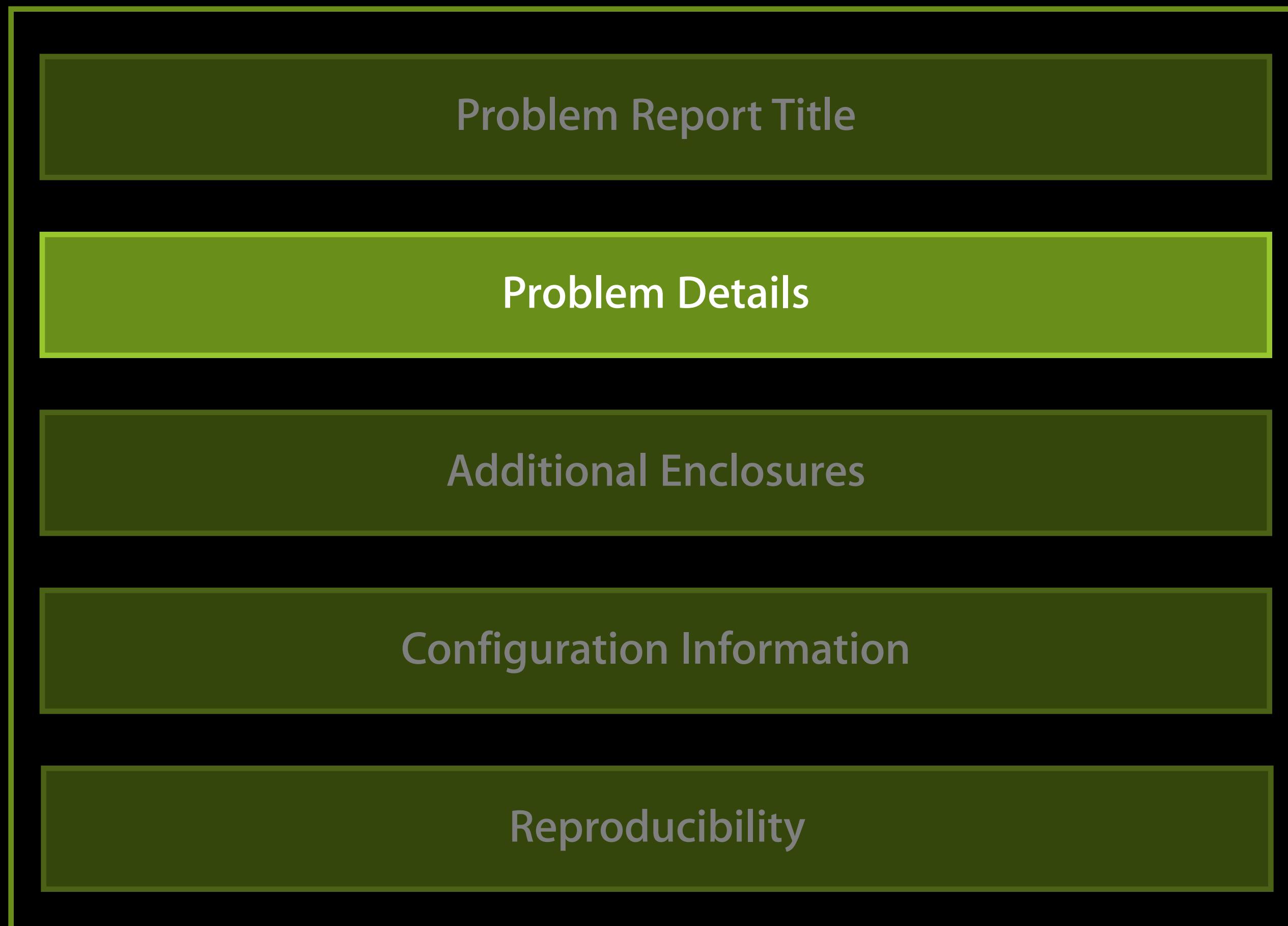
Copy and ~~p~~aste is broken

Pasting an item stored using [~~UI~~Pasteboard setImage:] is broken

Pasting an item stored using ~~[UI~~Pasteboard setImage:]
produces images ~~with~~ question marks

10A5355d: Pasting an item stored using [UIPasteboard setImage:]
produces images with question marks

Components of a Bug Report



Problem Details

Problem Details

Problem Summary

Steps To Reproduce

Expected Results

Actual Results

Regression

Problem Details

Problem Details

Problem Summary

Steps To Reproduce

Expected Results

Actual Results

Regression

Problem Summary

- Recap the title
- Include more descriptive summary information
- Avoid vague language (nothing happens, doesn't work, etc.)

An Example

- Summary:
 - Paste operation on item stored using [UIPasteboard setImage:] produces images with question marks in iOS 6 seed version 10A5355d. I have noticed this problem happen while pasting these images in Mail or Messages app

Problem Details

Problem Details

Problem Summary

Steps To Reproduce

Expected Results

Actual Results

Regression

An Example

An Example

- Steps to reproduce:

An Example

- Steps to reproduce:
 - Create a UIImage object:
 - `UIImage *image = [[UIImage imageNamed:@"test"]];`

An Example

- Steps to reproduce:
 - Create a UIImage object:
 - `UIImage *image = [[UIImage imageNamed:@"test"]];`
 - Write the UIImage object data to the pasteboard
 - `[[UIPasteboard generalPasteboard] setImage:image];`

An Example

- Steps to reproduce:
 - Create a UIImage object:
 - `UIImage *image = [[UIImage imageNamed:@"test"]];`
 - Write the UIImage object data to the pasteboard
 - `[[UIPasteboard generalPasteboard] setImage:image];`
 - Launch the Mail app

An Example

- Steps to reproduce:
 - Create a UIImage object:
 - `UIImage *image = [[UIImage imageNamed:@"test"]];`
 - Write the UIImage object data to the pasteboard
 - `[[UIPasteboard generalPasteboard] setImage:image];`
 - Launch the Mail app
 - Select the compose button to bring up the mail compose sheet

An Example

- Steps to reproduce:
 - Create a UIImage object:
 - `UIImage *image = [[UIImage imageNamed:@"test"]];`
 - Write the UIImage object data to the pasteboard
 - `[[UIPasteboard generalPasteboard] setImage:image];`
 - Launch the Mail app
 - Select the compose button to bring up the mail compose sheet
 - Tap and hold on message body of the mail compose sheet

An Example

- Steps to reproduce:
 - Create a UIImage object:
 - `UIImage *image = [[UIImage imageNamed:@"test"]];`
 - Write the UIImage object data to the pasteboard
 - `[[UIPasteboard generalPasteboard] setImage:image];`
 - Launch the Mail app
 - Select the compose button to bring up the mail compose sheet
 - Tap and hold on message body of the mail compose sheet
 - Select "Paste" from the callout that appears

Problem Details

Problem Details

Problem Summary

Steps To Reproduce

Expected Results

Actual Results

Regression

An Example

- Expected results:
 - I would expect the image that was copied using [UIPasteboard setImage:] to be pasted in the mail compose sheet
- Actual results:
 - The paste operation adds an image with a blue box containing a question mark (see attached screenshot)

Problem Details

Problem Details

Problem Summary

Steps To Reproduce

Expected Results

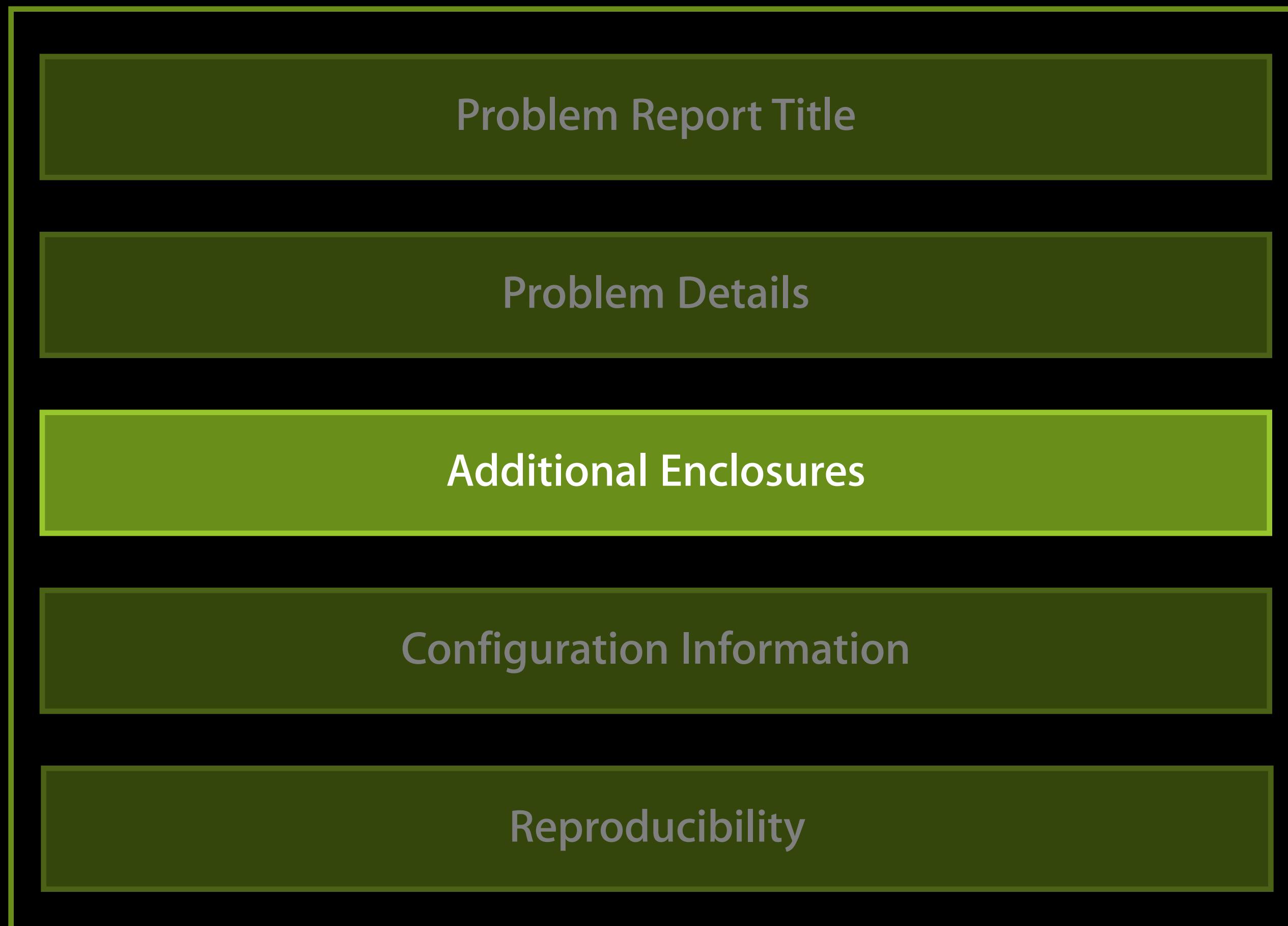
Actual Results

Regression

An Example

- Regression information:
 - This issue occurred on iOS 6 seed version 10A5355d.
 - This problem did not happen in iOS 5 and earlier
 - This problem does not occur if I use [UIPasteboard setData:] to write the image data to the pasteboard

Components of a Bug Report



Sample Project



Additional Information

- Information available at developer.apple.com/bugreporter/bugbestpractices

Additional Information

When you submit a bug report, it's important to include as much information as possible. Review the list below for instructions on obtaining additional logs and details you may need to provide with your bug report.

[iPhone, iPad, and iPod](#)

- [3rd Party Apps](#)
- [Apple Push Notifications](#)
- [Baseband Logs \(iPad\)](#)
- [Baseband Logs \(iPhone\)](#)
- [Battery Life Logs](#)
- [Bluetooth](#)
- [Calendar Data Export](#)
- [Contacts Data Export](#)
- [Device Console Logs](#)
- [iCloud Logs](#)
- [iOS Crash Logs](#)
- [iPod](#)
- [Mail Logs](#)
- [Stackshot Logs](#)
- [Sync Diagnostics Logs \(DA\)](#)
- [Test Case Requirements](#)
- [WiFi Logs](#)
- [Miscellaneous](#)

[Mac](#)

- [3rd Party Products](#)
- [Bluetooth](#)
- [Console Logs](#)
- [Crash Logs](#)
- [iCloud](#)
- [Mail](#)
- [Panic Reports](#)
- [Safari](#)
- [System Information Report](#)
- [Time Machine](#)
- [WiFi Logs](#)
- [Xcode](#)
- [Miscellaneous](#)

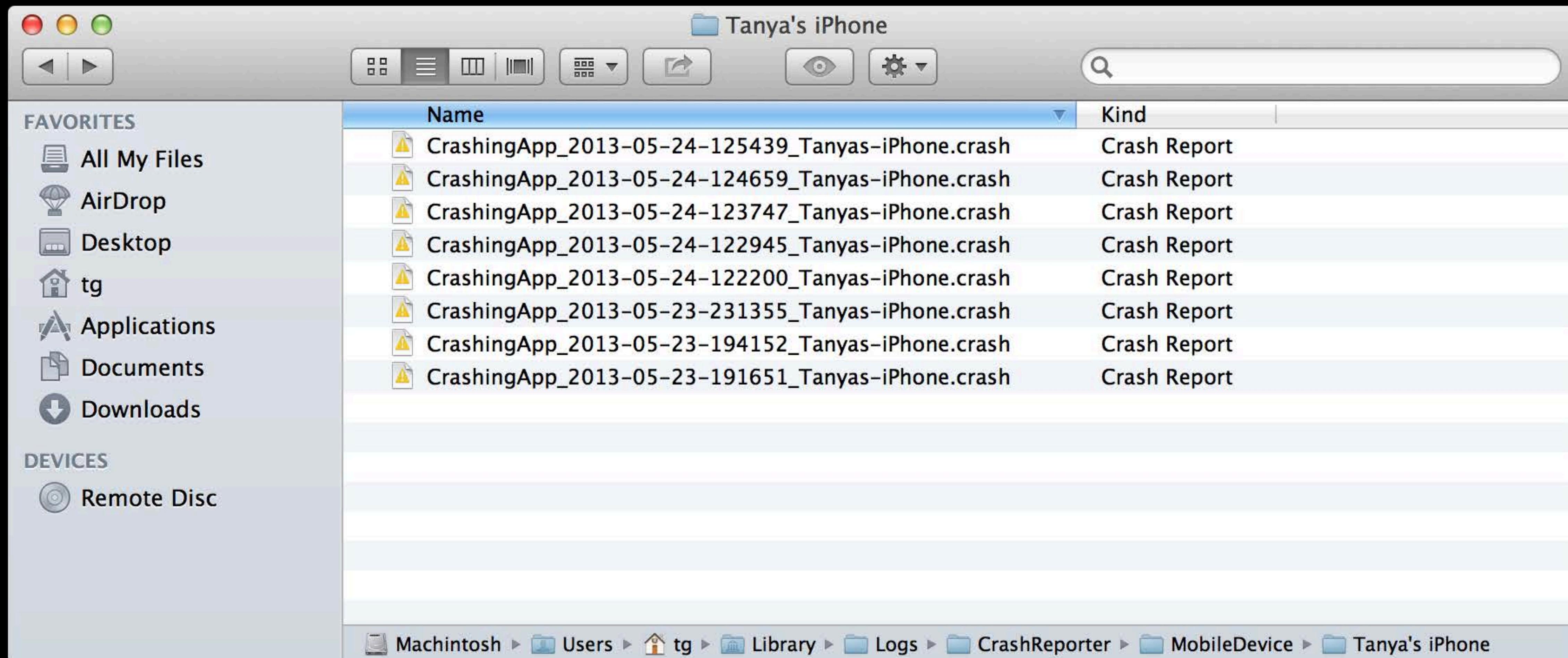
Crash Logs for iOS

Step 1: Transfer the logs

- Connect your device
- Sync the device in iTunes

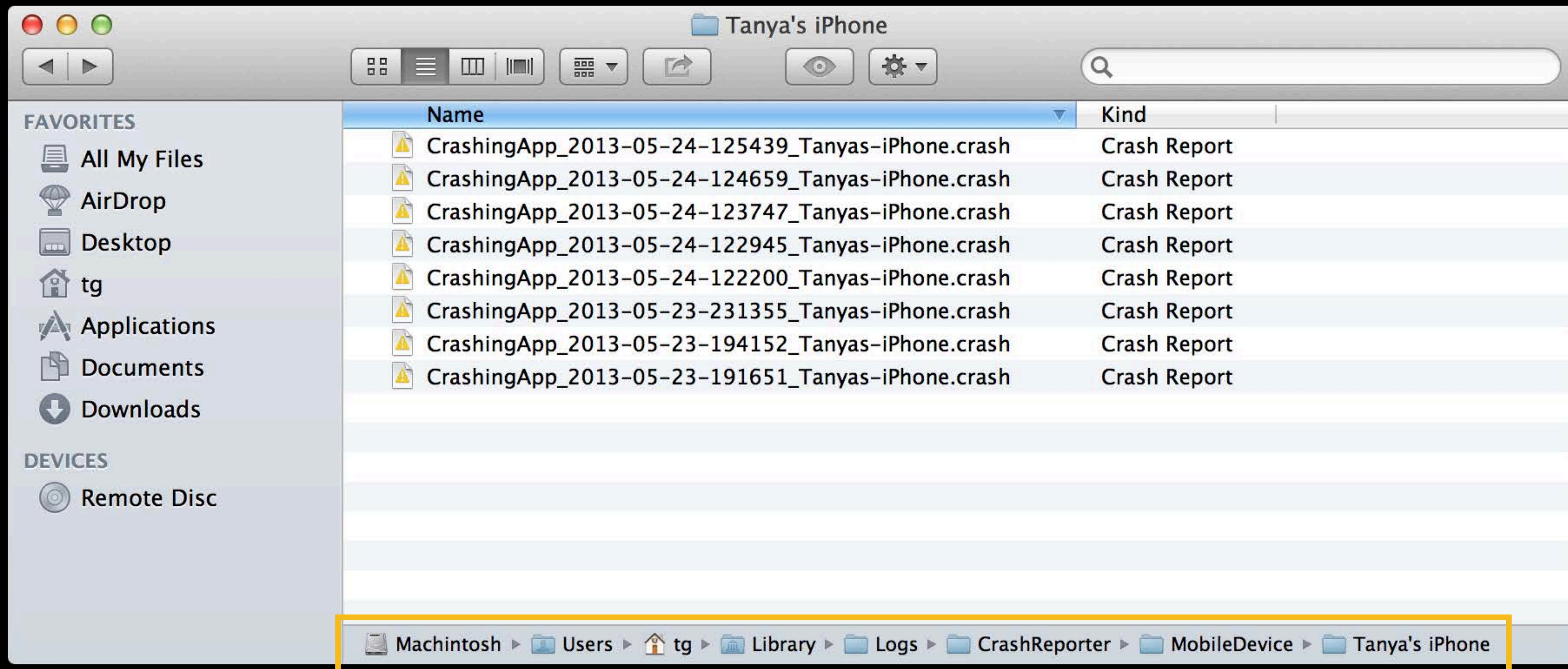
Crash Logs for iOS

Step 2: Locate the log file



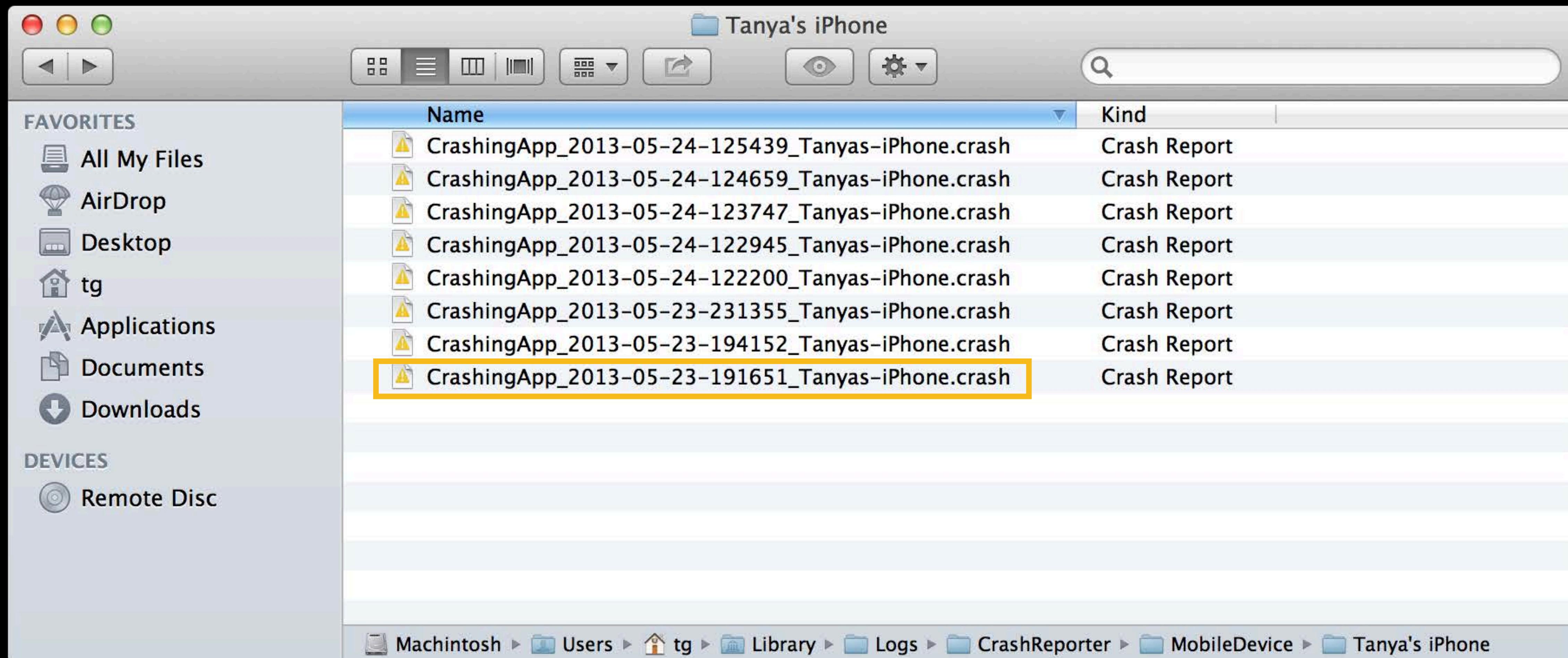
Crash Logs for iOS

Step 2: Locate the log file



Crash Logs for iOS

Step 2: Locate the log file

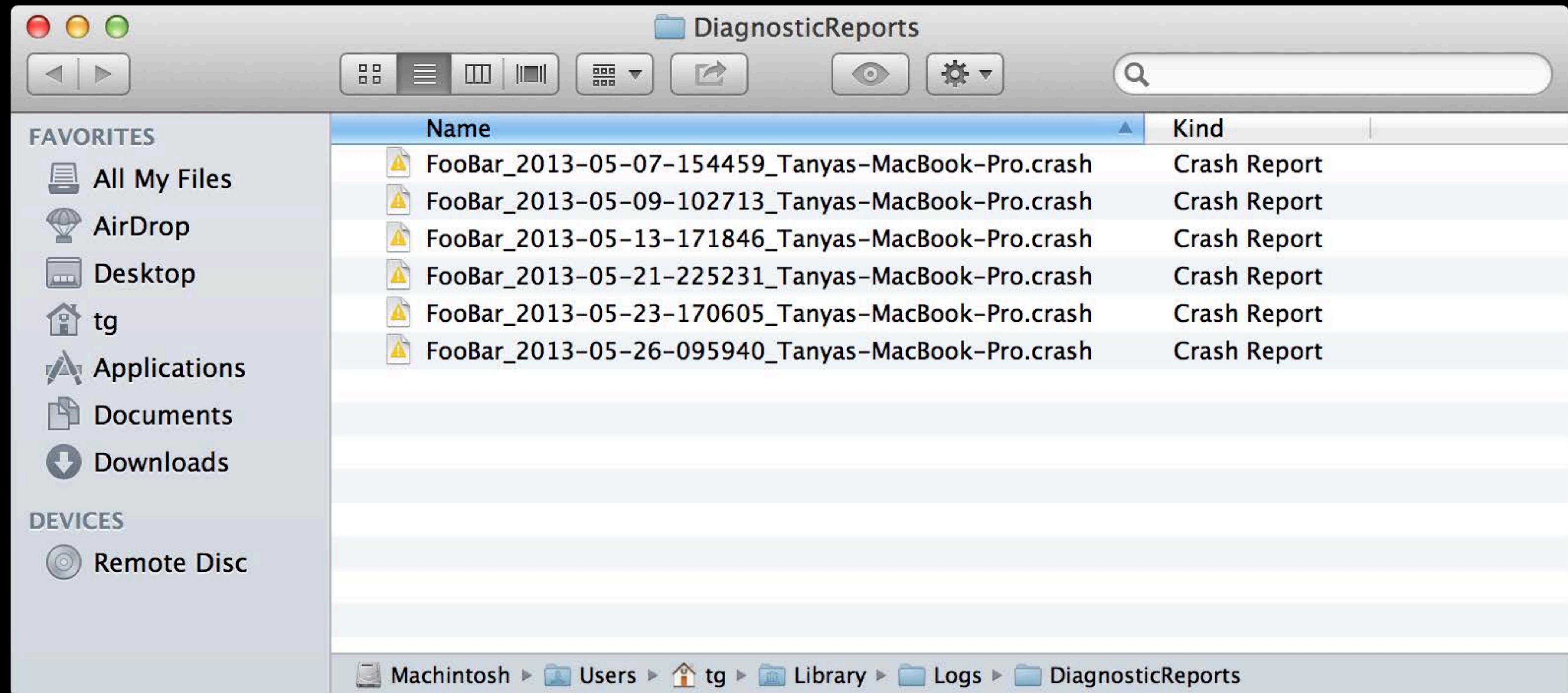


Crash Logs for iOS

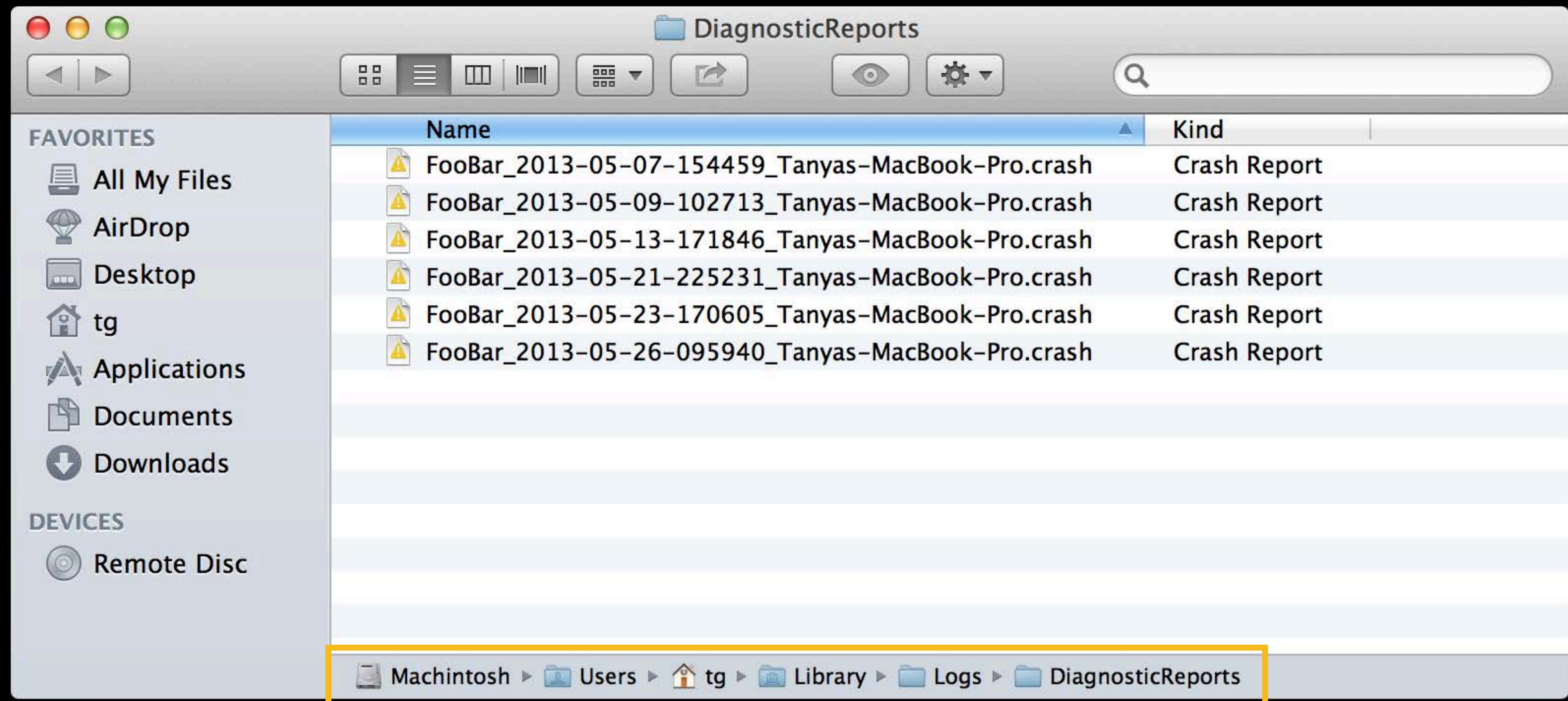
Step 2: Locate the log file

- On Windows, look under:
 - C:\Users\[Your_User_Name]\AppData\Roaming\Apple Computer\Logs\CrashReporter\MobileDevice\[Your_Device_Name]

Crash Logs for OS X

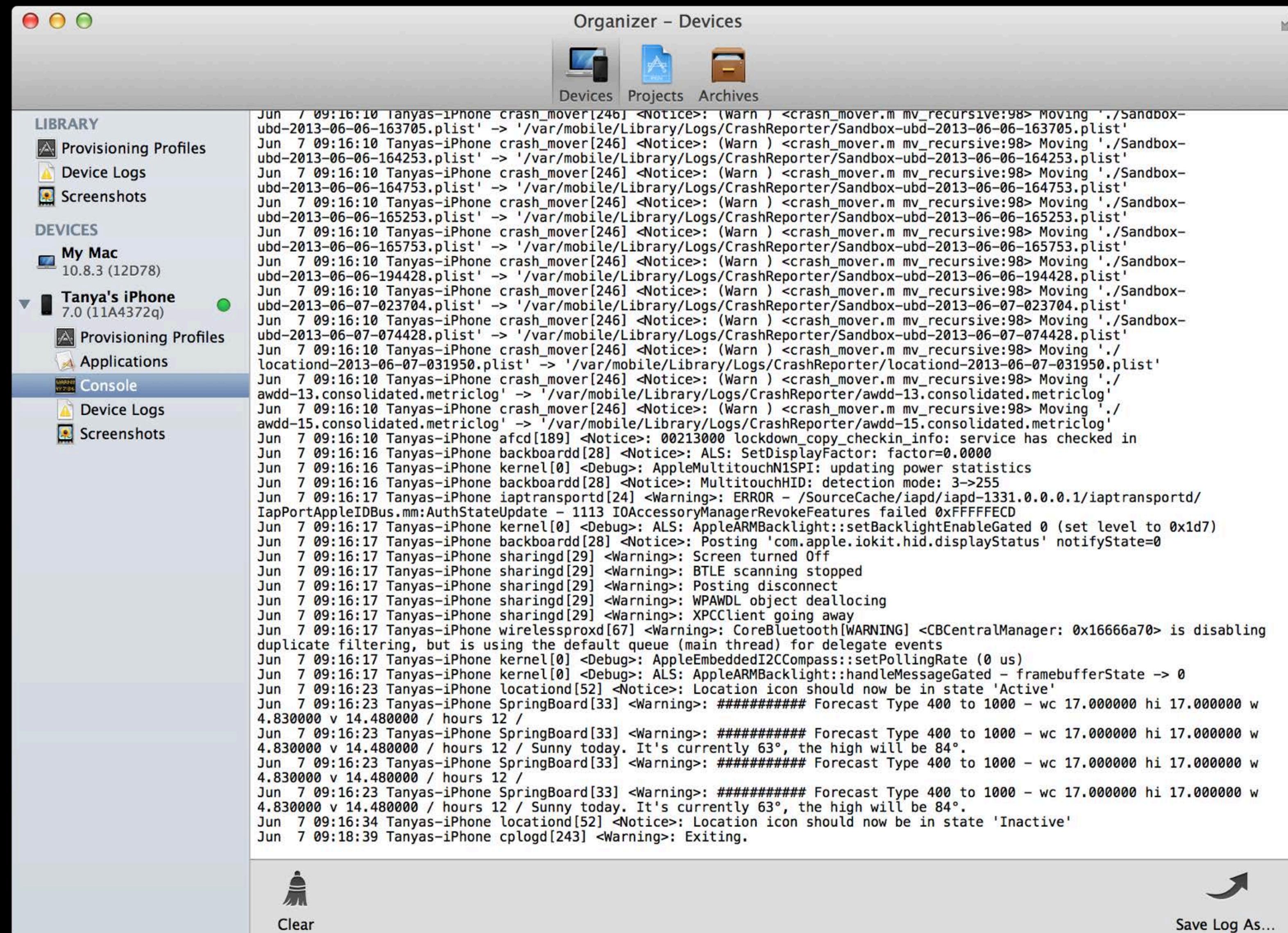


Crash Logs for OS X



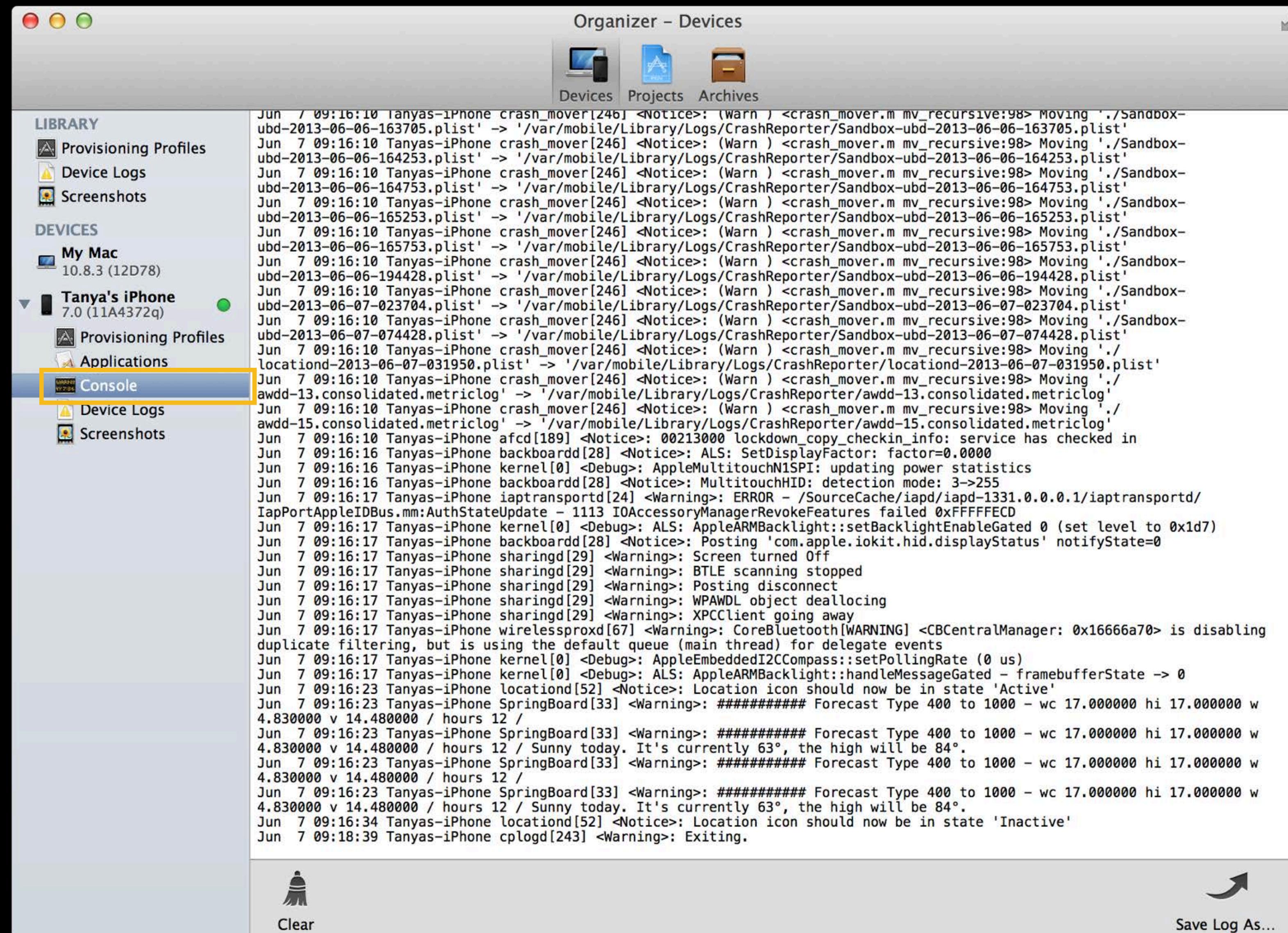
Console Logs for iOS

Xcode organizer



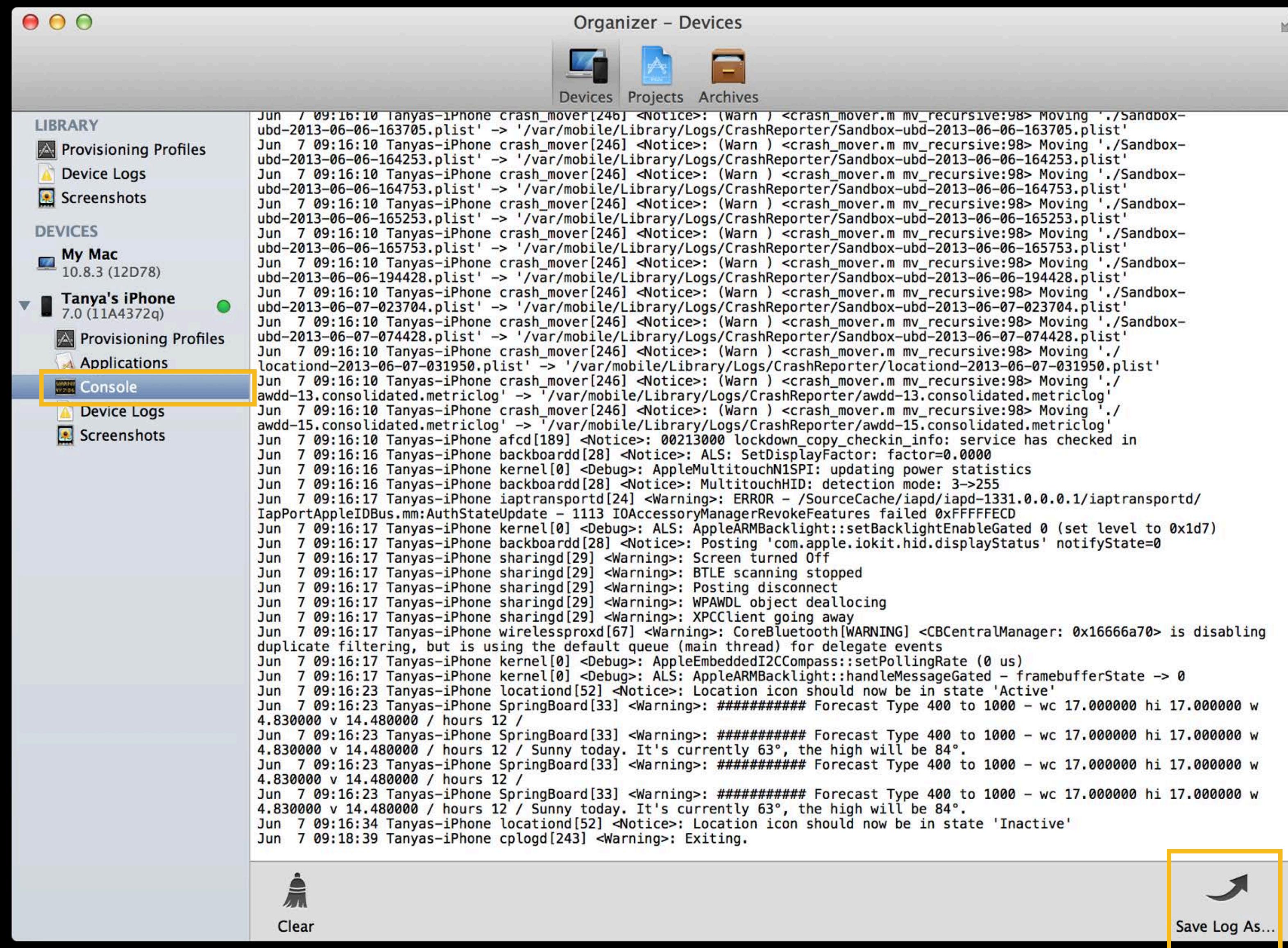
Console Logs for iOS

Xcode organizer



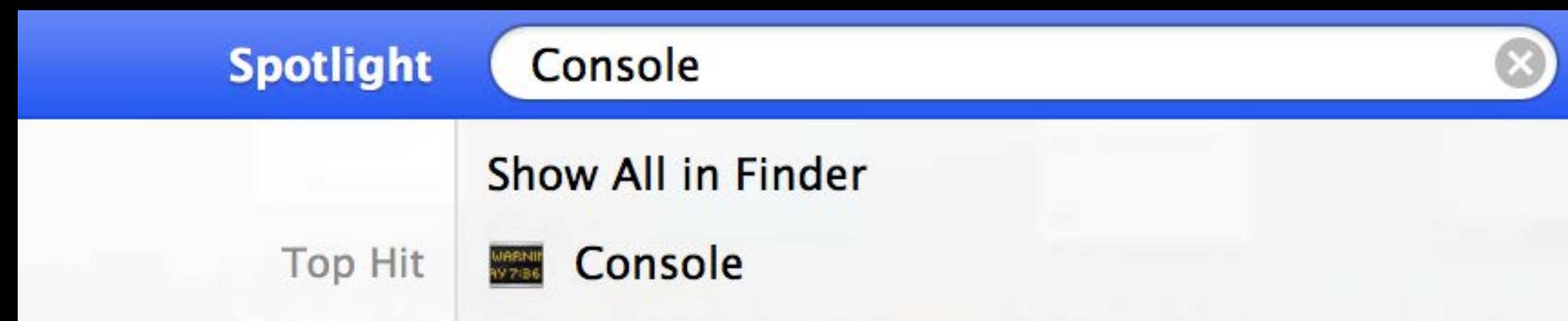
Console Logs for iOS

Xcode organizer



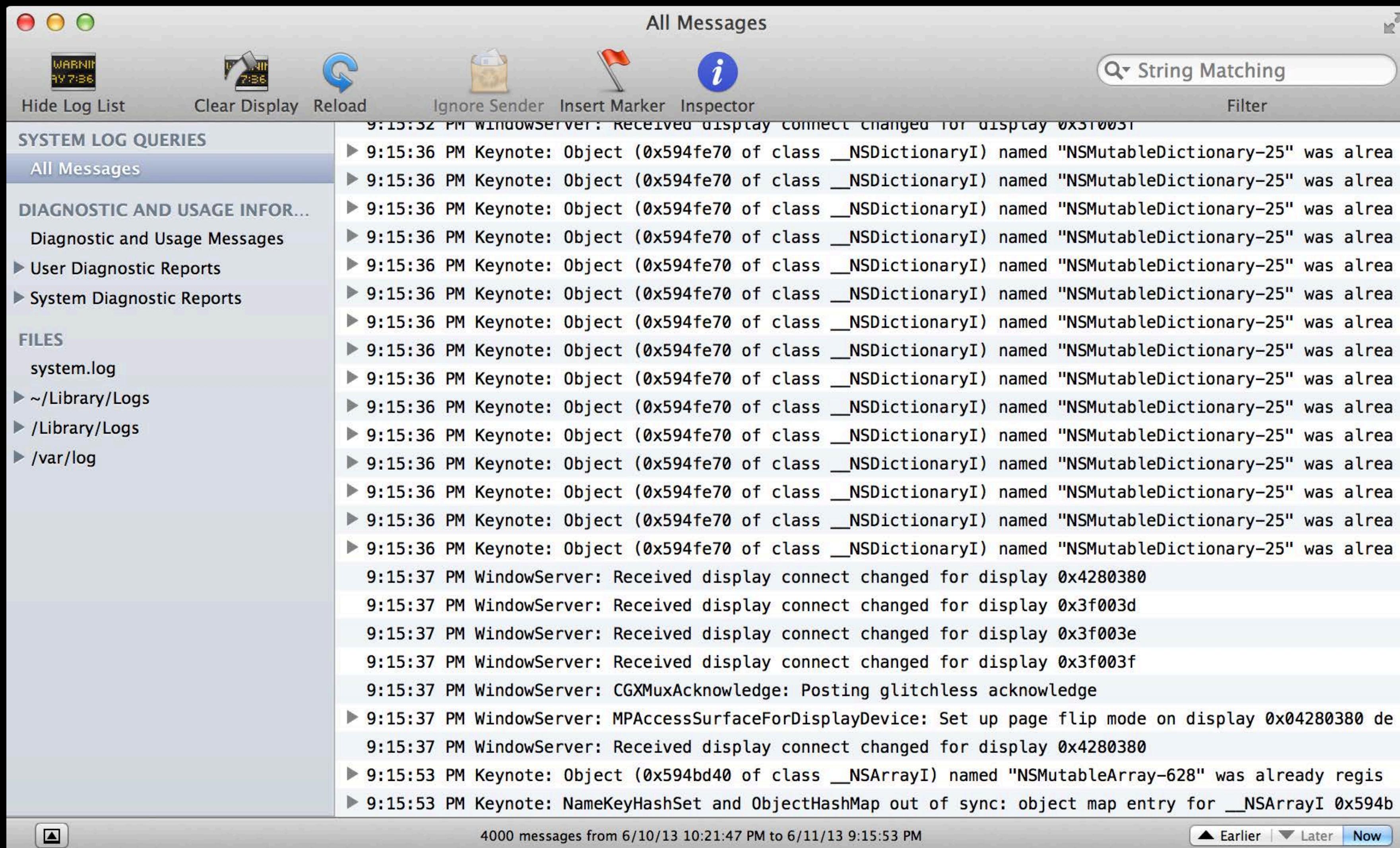
Console Logs for OS X

Step 1: Launch Console.app



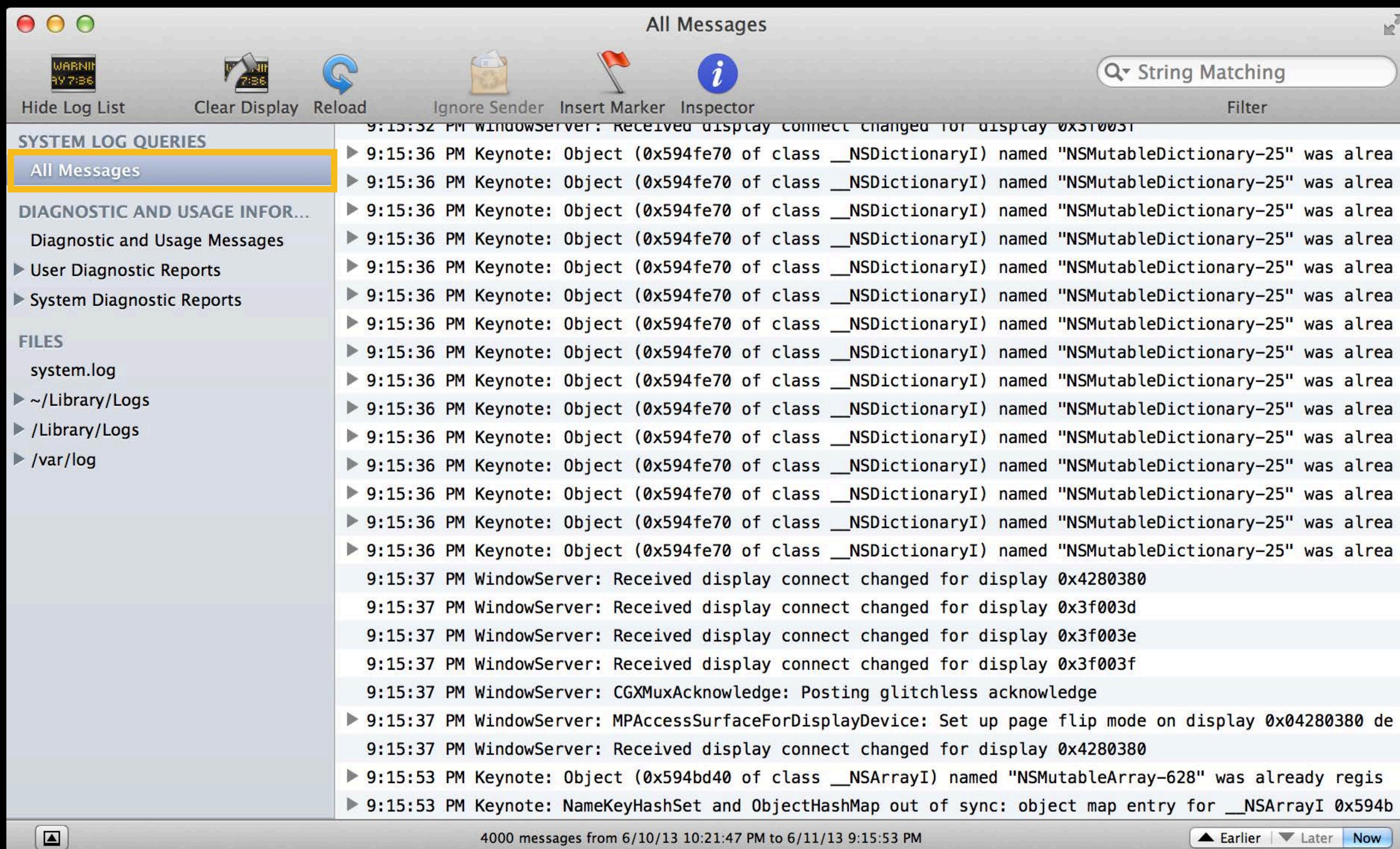
Console Logs for OS X

Step 2: Select All Messages



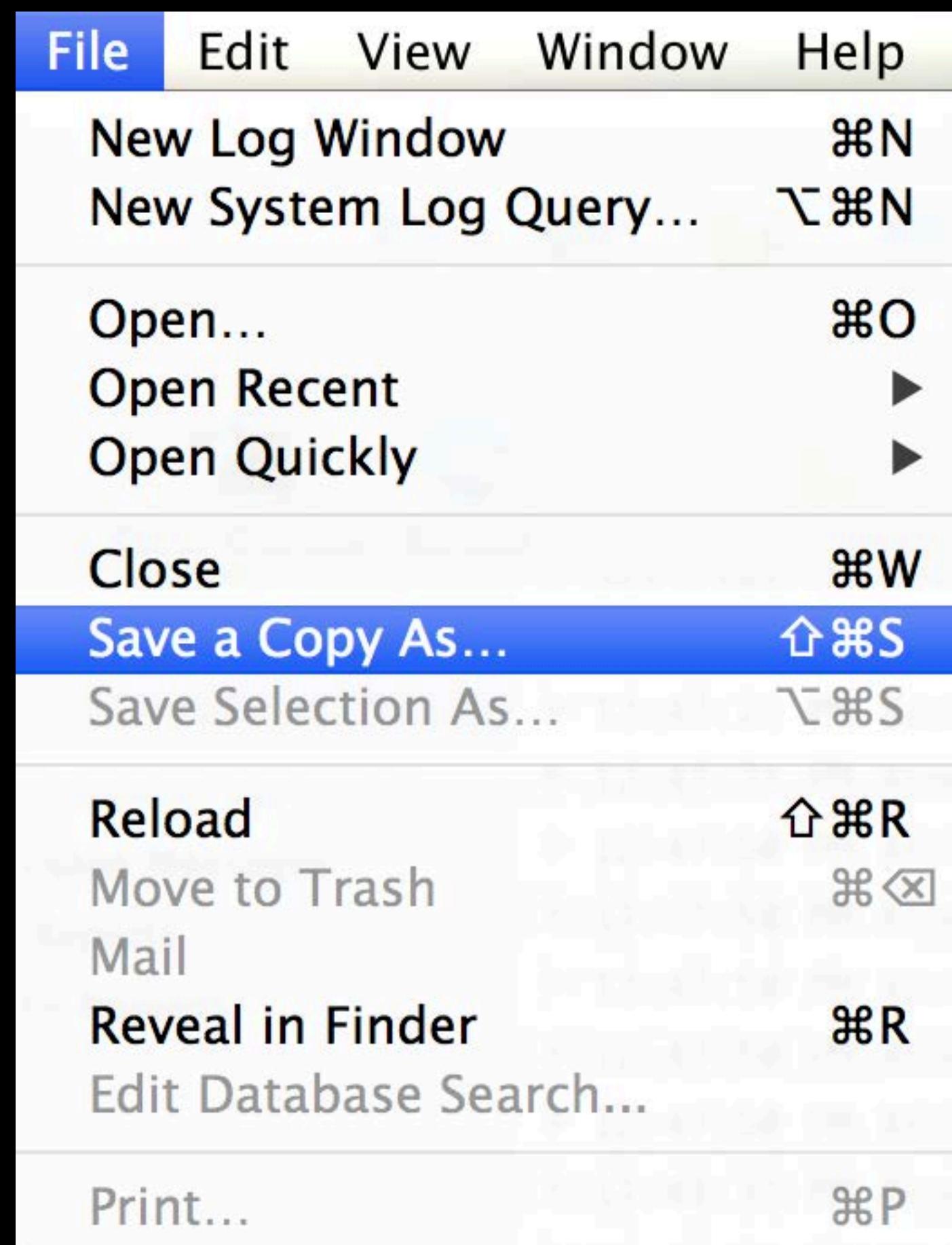
Console Logs for OS X

Step 2: Select All Messages



Console Logs for OS X

Step 3: Save the log file



Screenshots on iOS

- Press and release the home and power button
- Saved to the Photos app on your iOS device



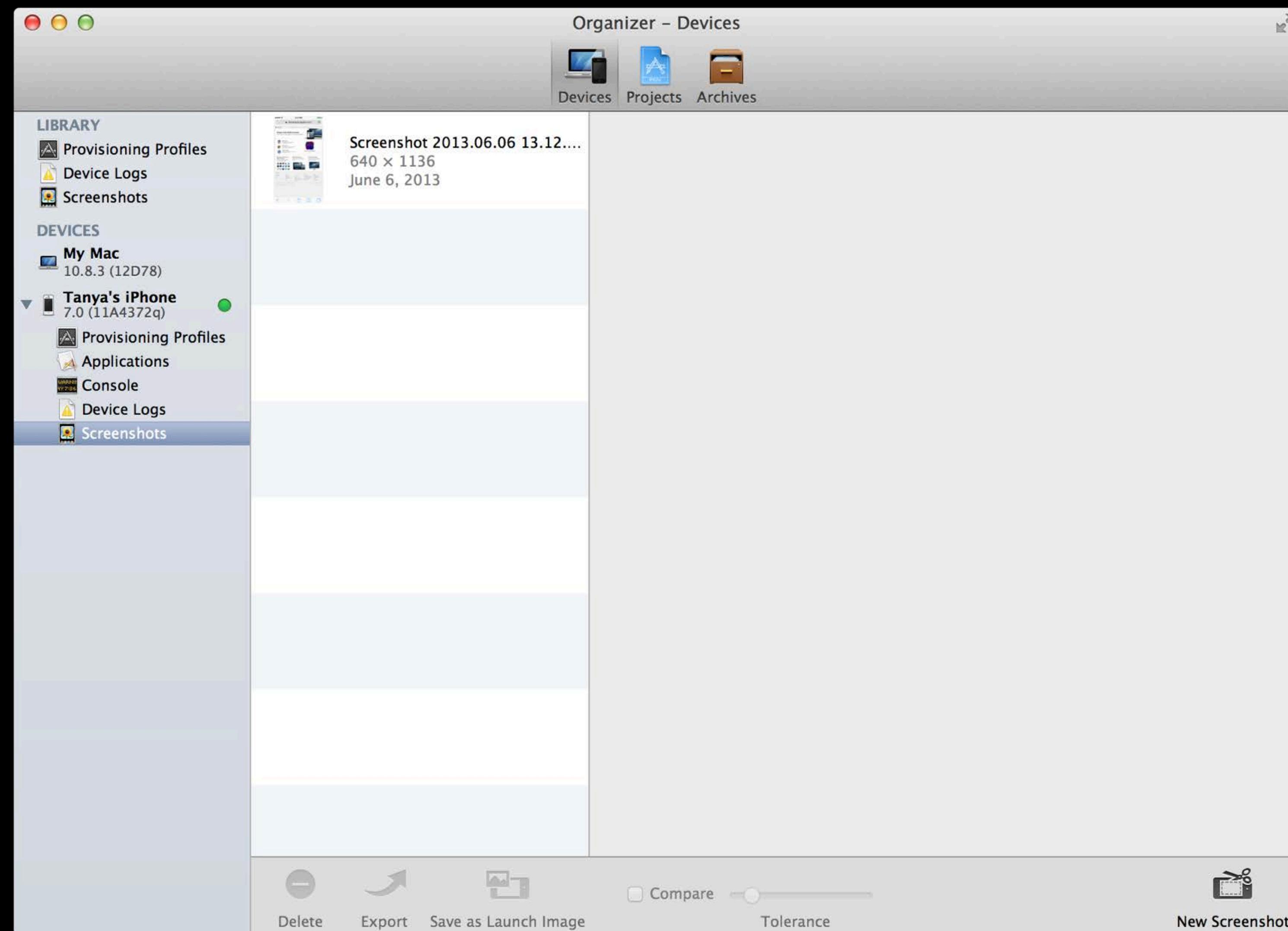
Screenshots on iOS

- Press and release the home and power button
- Saved to the Photos app on your iOS device



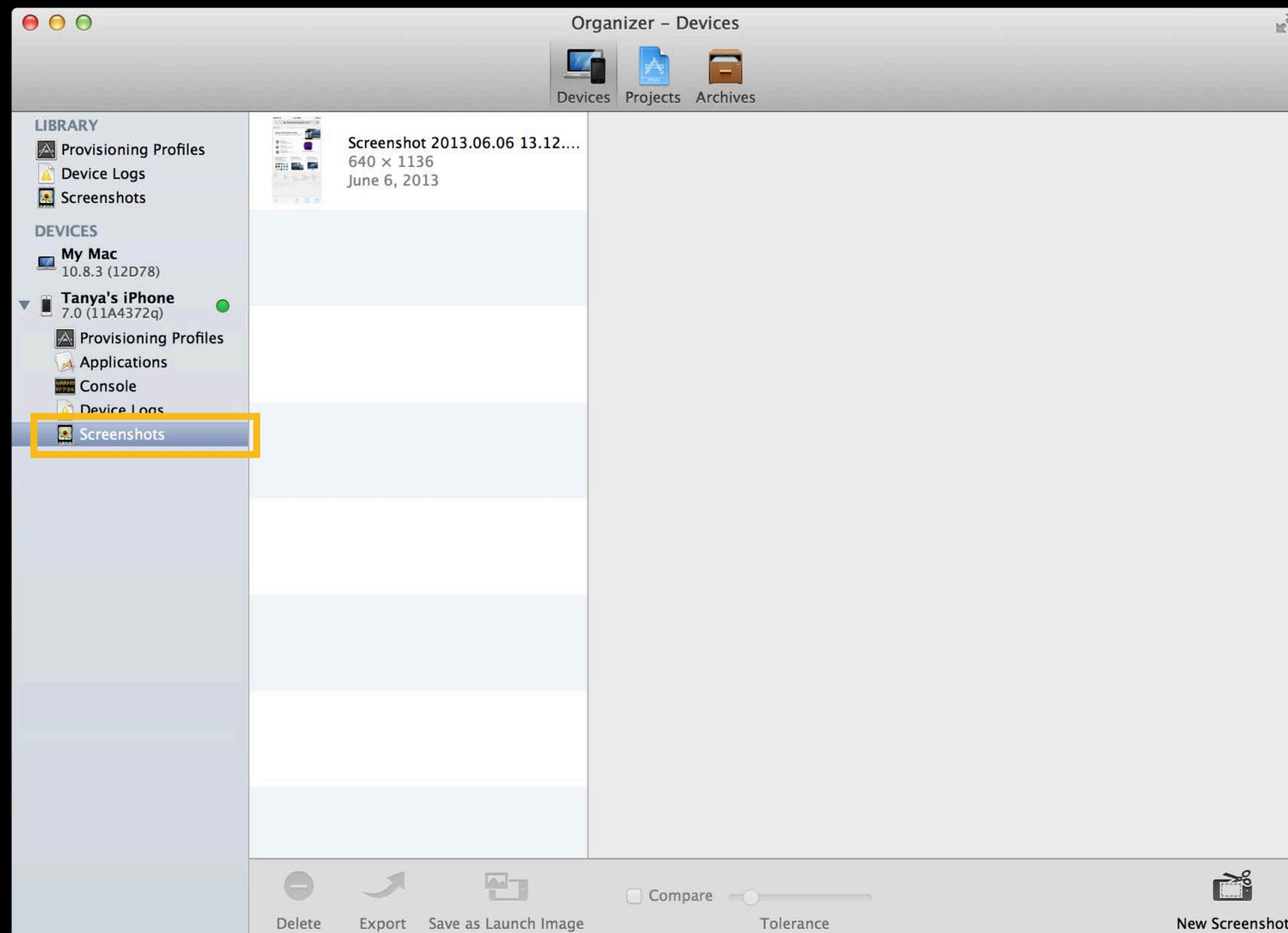
Screenshots on iOS

Xcode Organizer



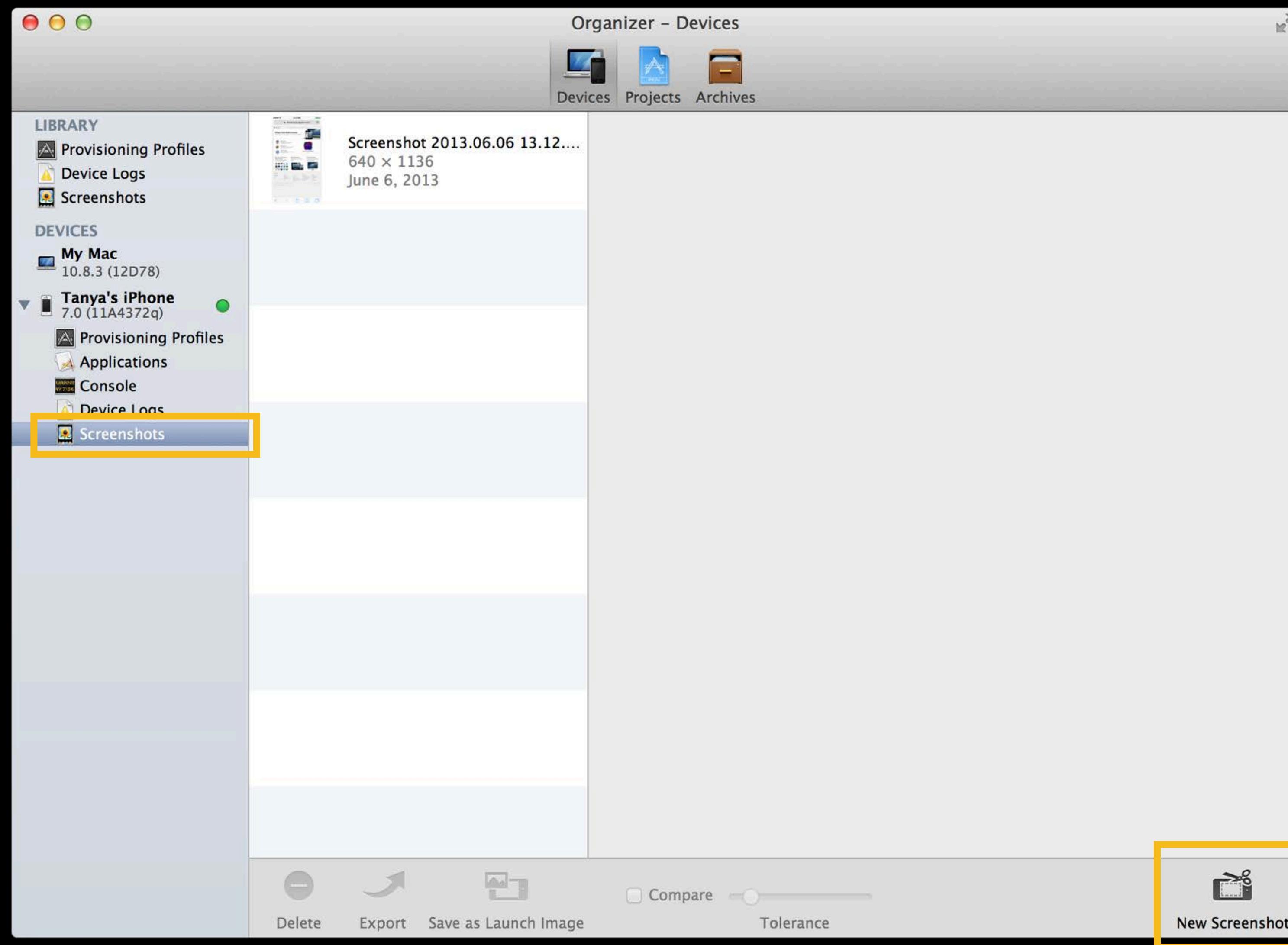
Screenshots on iOS

Xcode Organizer



Screenshots on iOS

Xcode Organizer



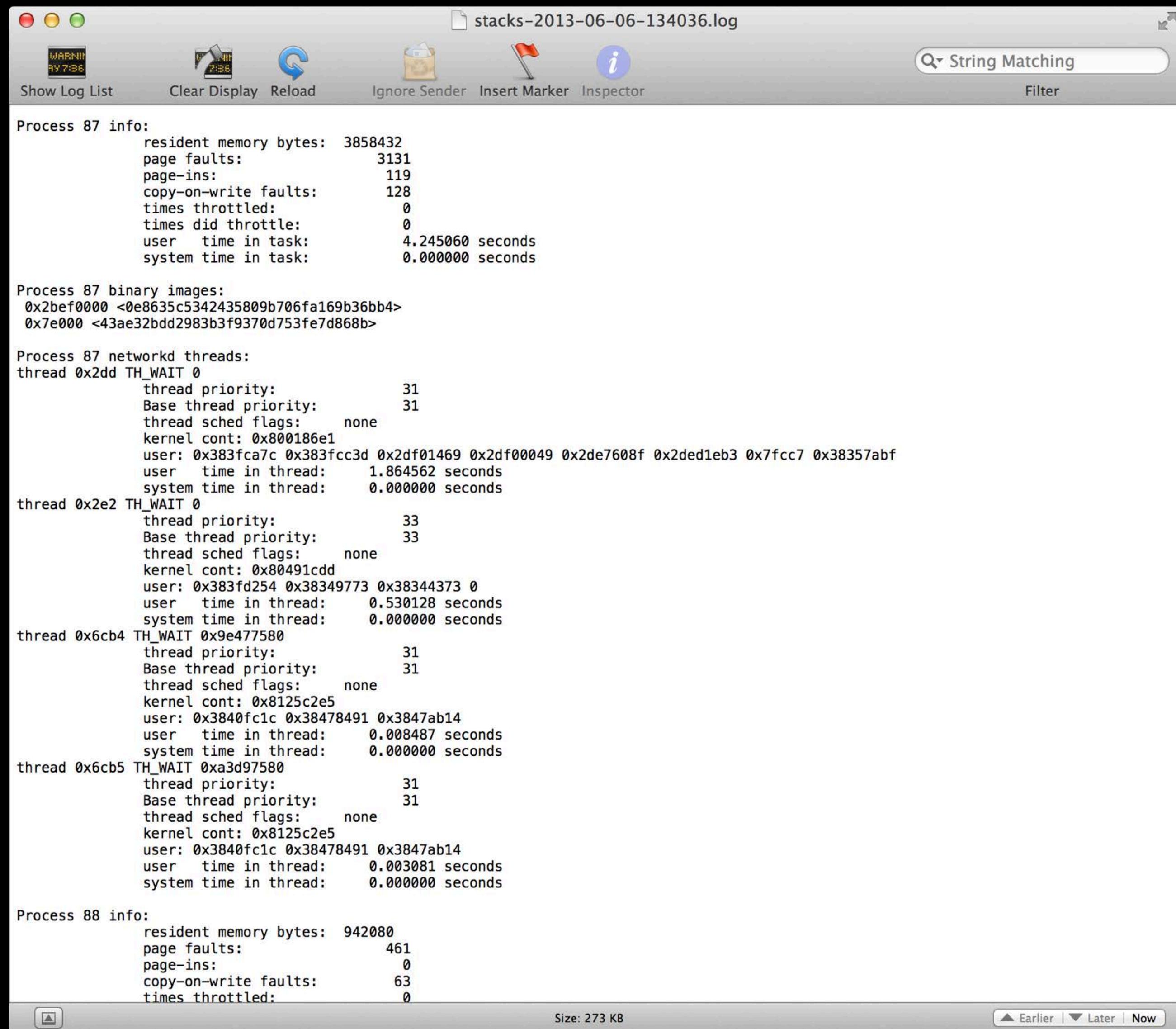
Screenshots on OS X

- Save a picture of the entire screen or screens
 - Press Command+Shift and then 3
- Save a picture of a selected area of the screen
 - Press Command+Shift and then 4
- Save a picture of a specific window
 - Press Command+Shift, then 4 and then spacebar

Screen Recordings on OS X

- To capture a screen recording, use QuickTime Player
 - Select File→New Screen Recording
 - Select the record button

Stackshot Logs for iOS



The screenshot shows the Stackshot application window displaying log files. The title bar reads "stacks-2013-06-06-134036.log". The menu bar includes "File", "Edit", "Log", "Process", "Threads", "Memory", "CPU", "Disk", "Network", "Filesystem", "Security", "Performance", "System", "Help", and "About". The toolbar contains icons for "Show Log List", "Clear Display", "Reload", "Ignore Sender", "Insert Marker", "Inspector", "String Matching", and "Filter".

The main pane displays log entries for Process 87 and Process 88.

Process 87 info:

```
resident memory bytes: 3858432
page faults: 3131
page-ins: 119
copy-on-write faults: 128
times throttled: 0
times did throttle: 0
user time in task: 4.245060 seconds
system time in task: 0.000000 seconds
```

Process 87 binary images:

```
0x2bef0000 <0e8635c5342435809b706fa169b36bb4>
0x7e000 <43ae32bdd2983b3f9370d753fe7d868b>
```

Process 87 networkd threads:

```
thread 0x2dd TH_WAIT 0
    thread priority: 31
    Base thread priority: 31
    thread sched flags: none
    kernel cont: 0x800186e1
    user: 0x383fc7c 0x383fcc3d 0x2df01469 0x2df00049 0x2de7608f 0x2ded1eb3 0x7fcc7 0x38357abf
    user time in thread: 1.864562 seconds
    system time in thread: 0.000000 seconds

thread 0x2e2 TH_WAIT 0
    thread priority: 33
    Base thread priority: 33
    thread sched flags: none
    kernel cont: 0x80491cd
    user: 0x383fd254 0x38349773 0x38344373 0
    user time in thread: 0.530128 seconds
    system time in thread: 0.000000 seconds

thread 0x6cb4 TH_WAIT 0x9e477580
    thread priority: 31
    Base thread priority: 31
    thread sched flags: none
    kernel cont: 0x8125c2e5
    user: 0x3840fc1c 0x38478491 0x3847ab14
    user time in thread: 0.008487 seconds
    system time in thread: 0.000000 seconds

thread 0x6cb5 TH_WAIT 0xa3d97580
    thread priority: 31
    Base thread priority: 31
    thread sched flags: none
    kernel cont: 0x8125c2e5
    user: 0x3840fc1c 0x38478491 0x3847ab14
    user time in thread: 0.003081 seconds
    system time in thread: 0.000000 seconds
```

Process 88 info:

```
resident memory bytes: 942080
page faults: 461
page-ins: 0
copy-on-write faults: 63
times throttled: 0
```

At the bottom, there are navigation buttons for "Earlier", "Later", and "Now", and a status bar indicating "Size: 273 KB".

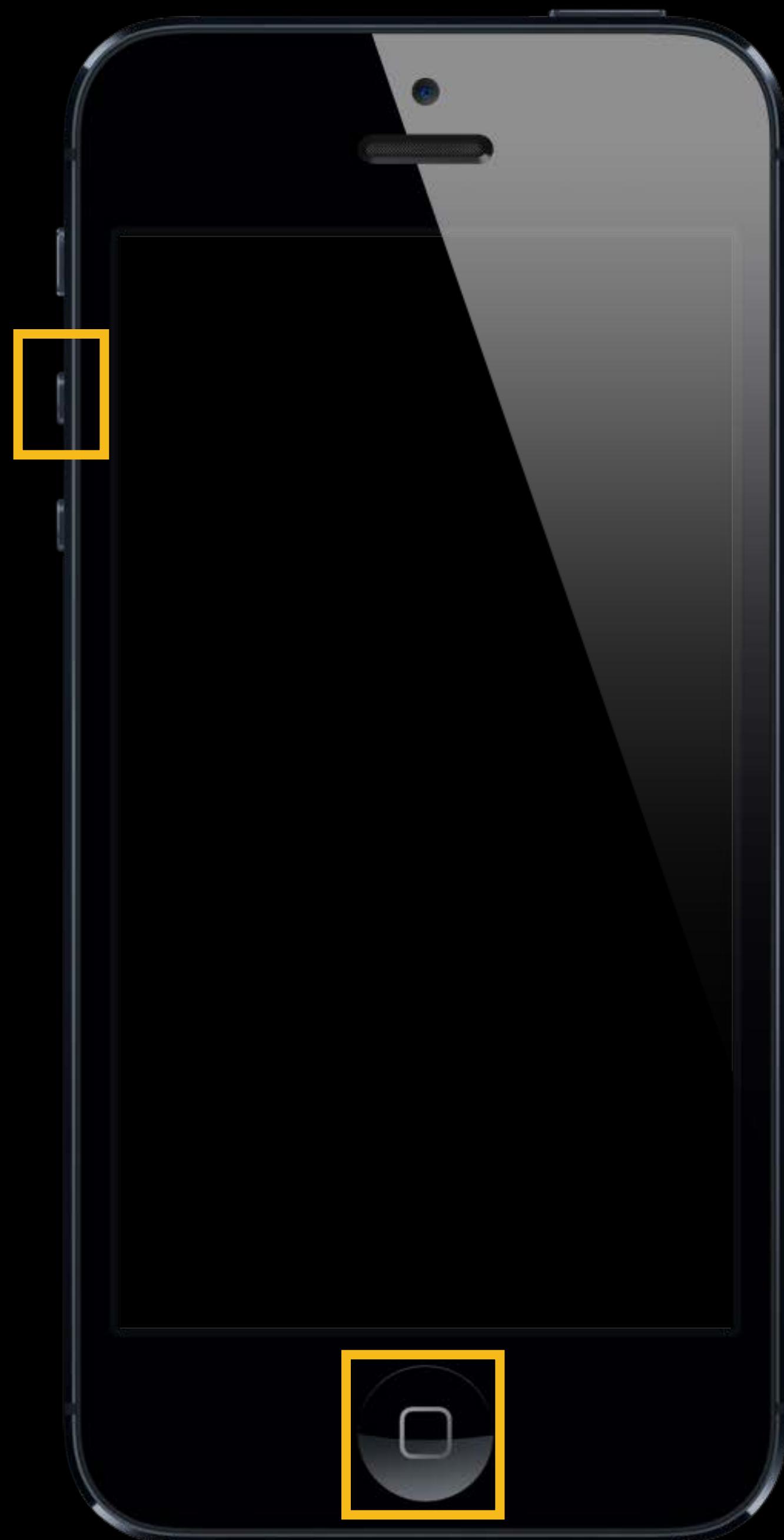
Stackshot Logs for iOS

- Press home and one of the volume buttons together
- Same steps as a crash log for retrieval

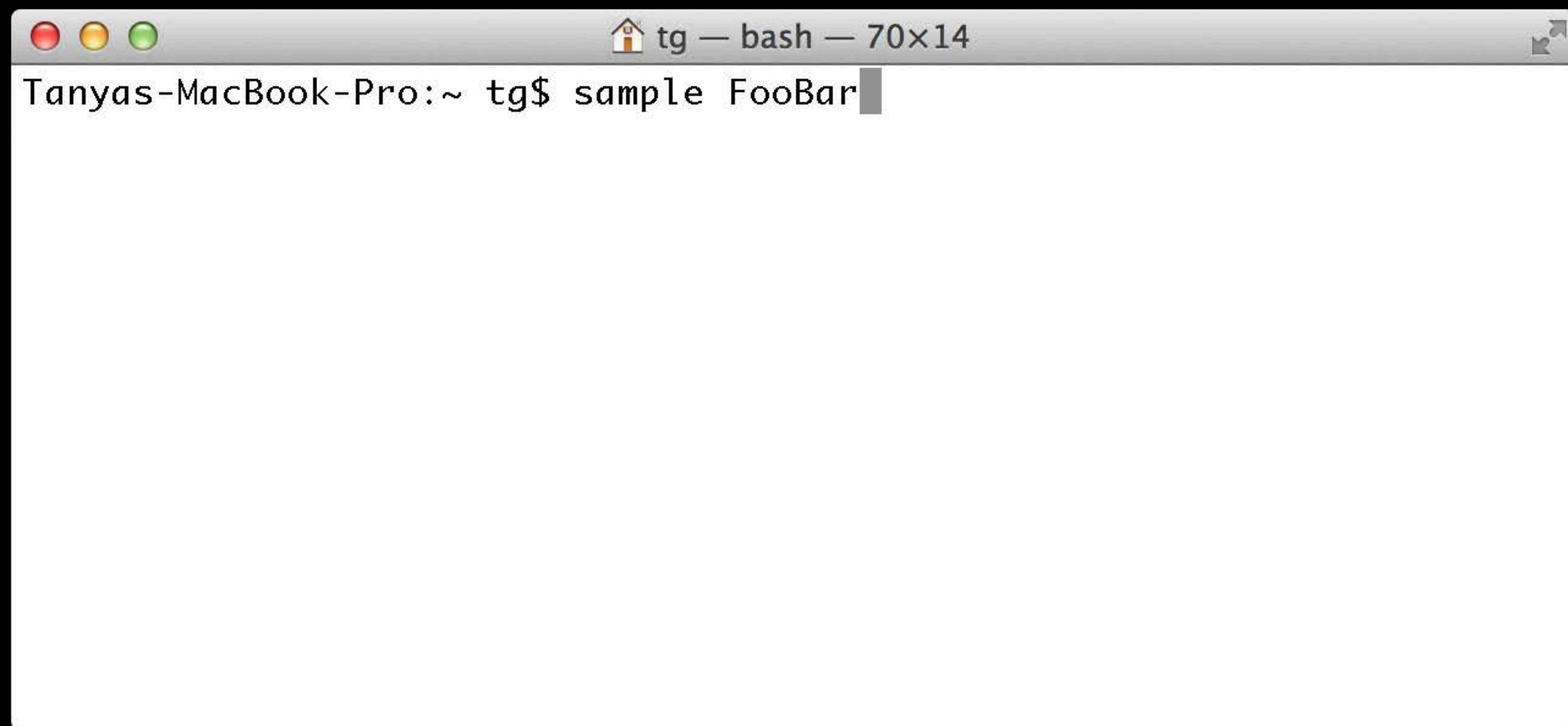


Stackshot Logs for iOS

- Press home and one of the volume buttons together
- Same steps as a crash log for retrieval

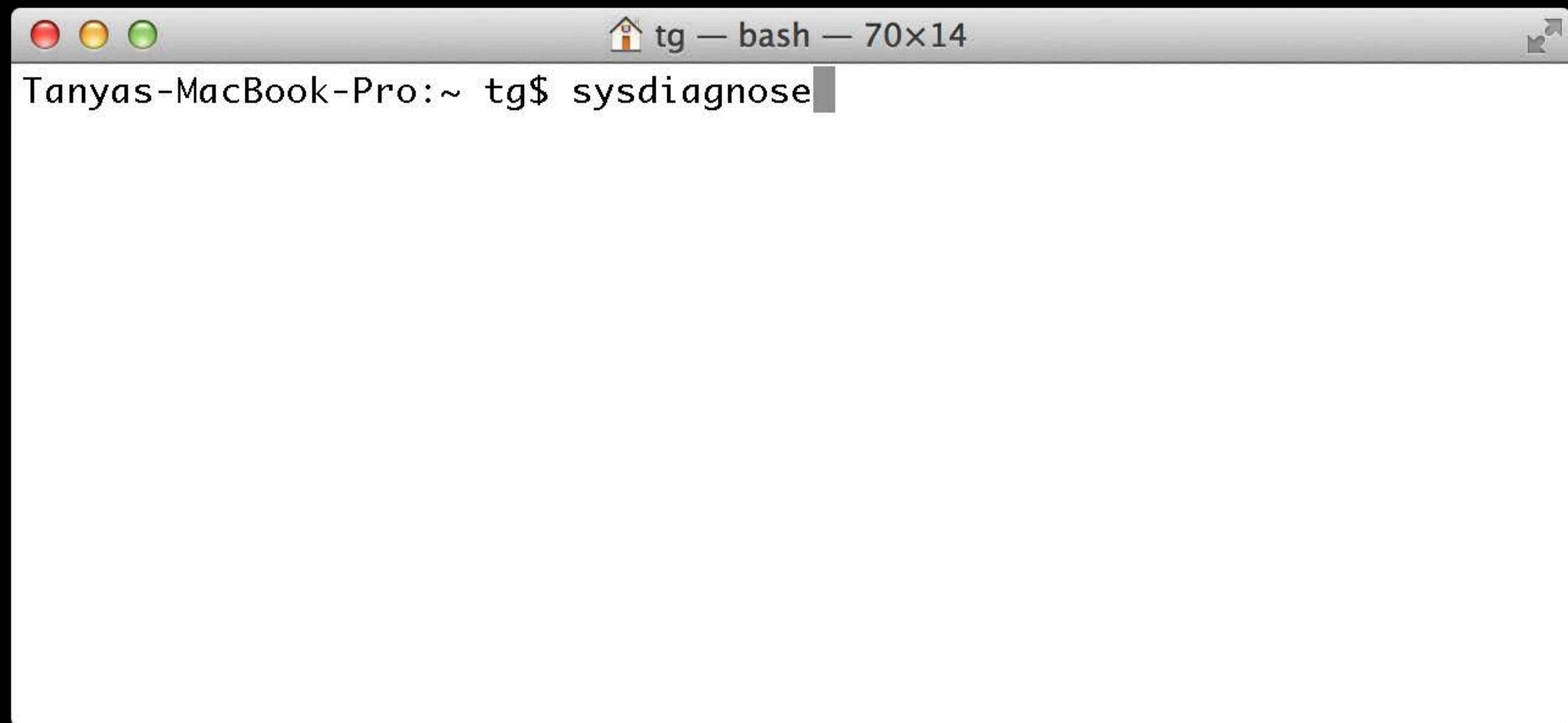


Sample Logs for OS X



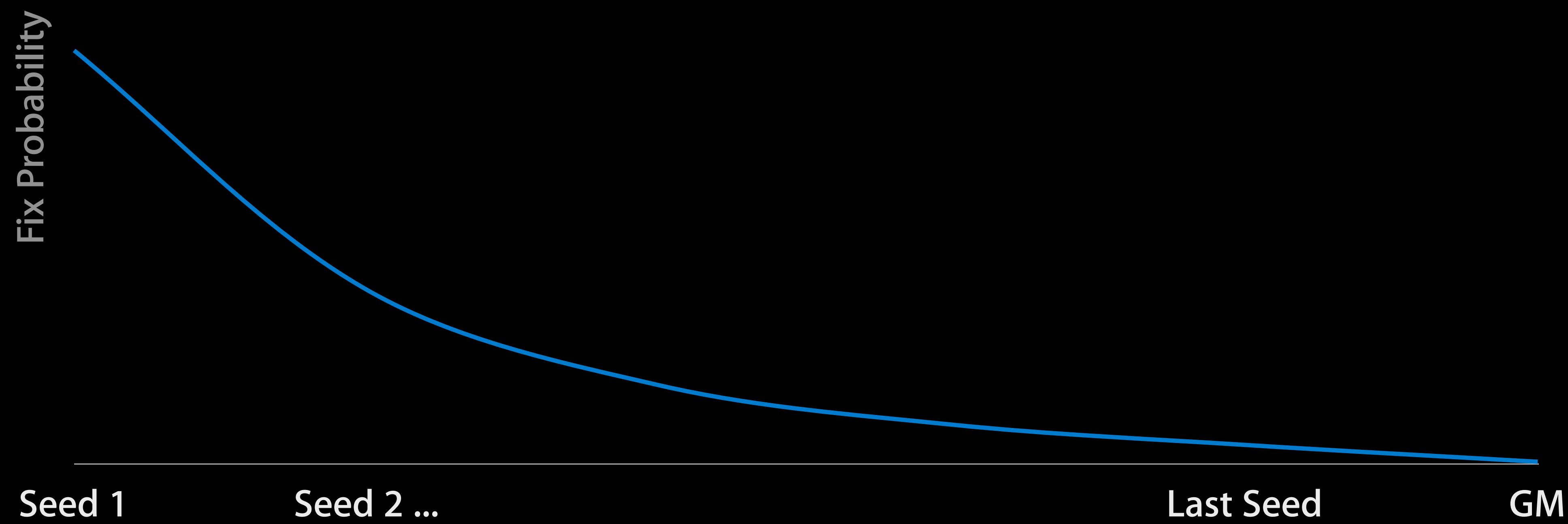
A screenshot of a Mac OS X terminal window. The window title bar reads "tg — bash — 70x14". The main pane displays the command "Tanyas-MacBook-Pro:~ tg\$ sample FooBar" followed by a blank white space, indicating that the command has been entered but no output has been displayed.

sysdiagnose for OS X

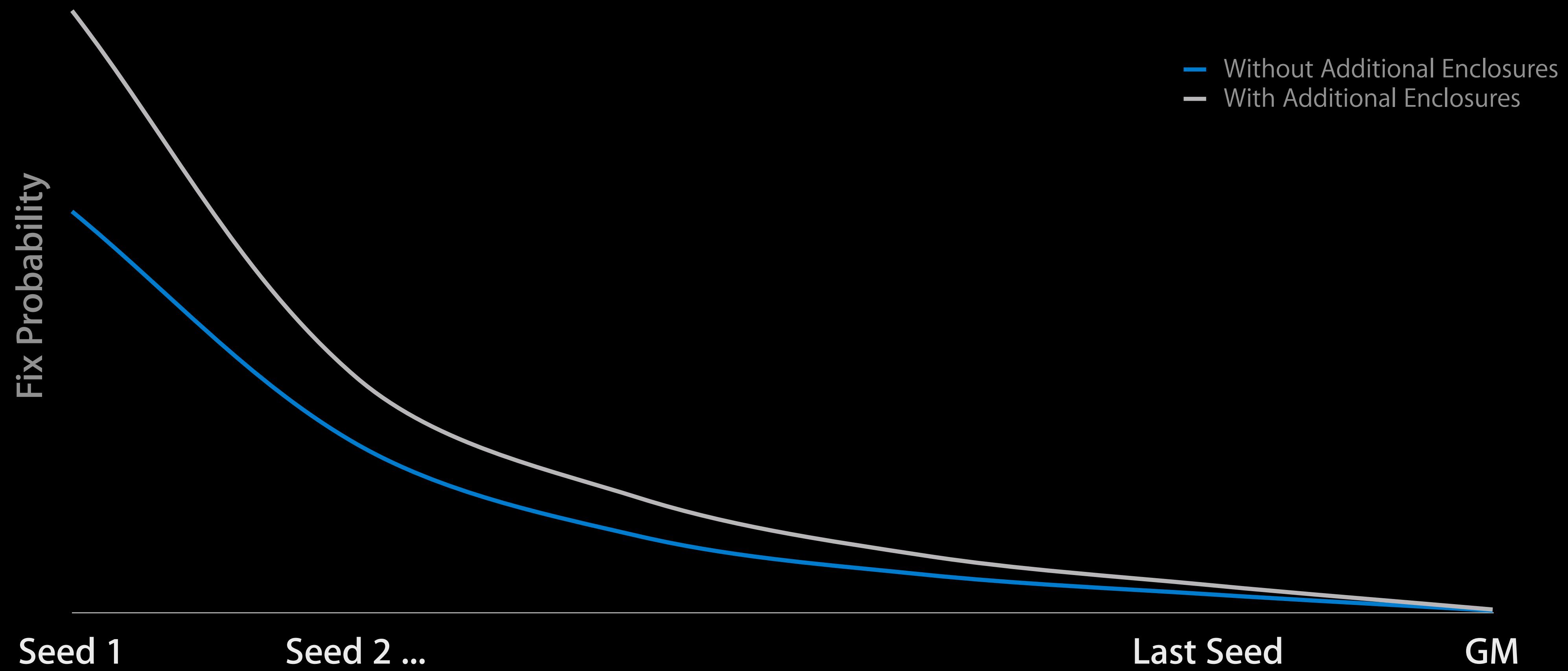


Benefit of Adding More Information

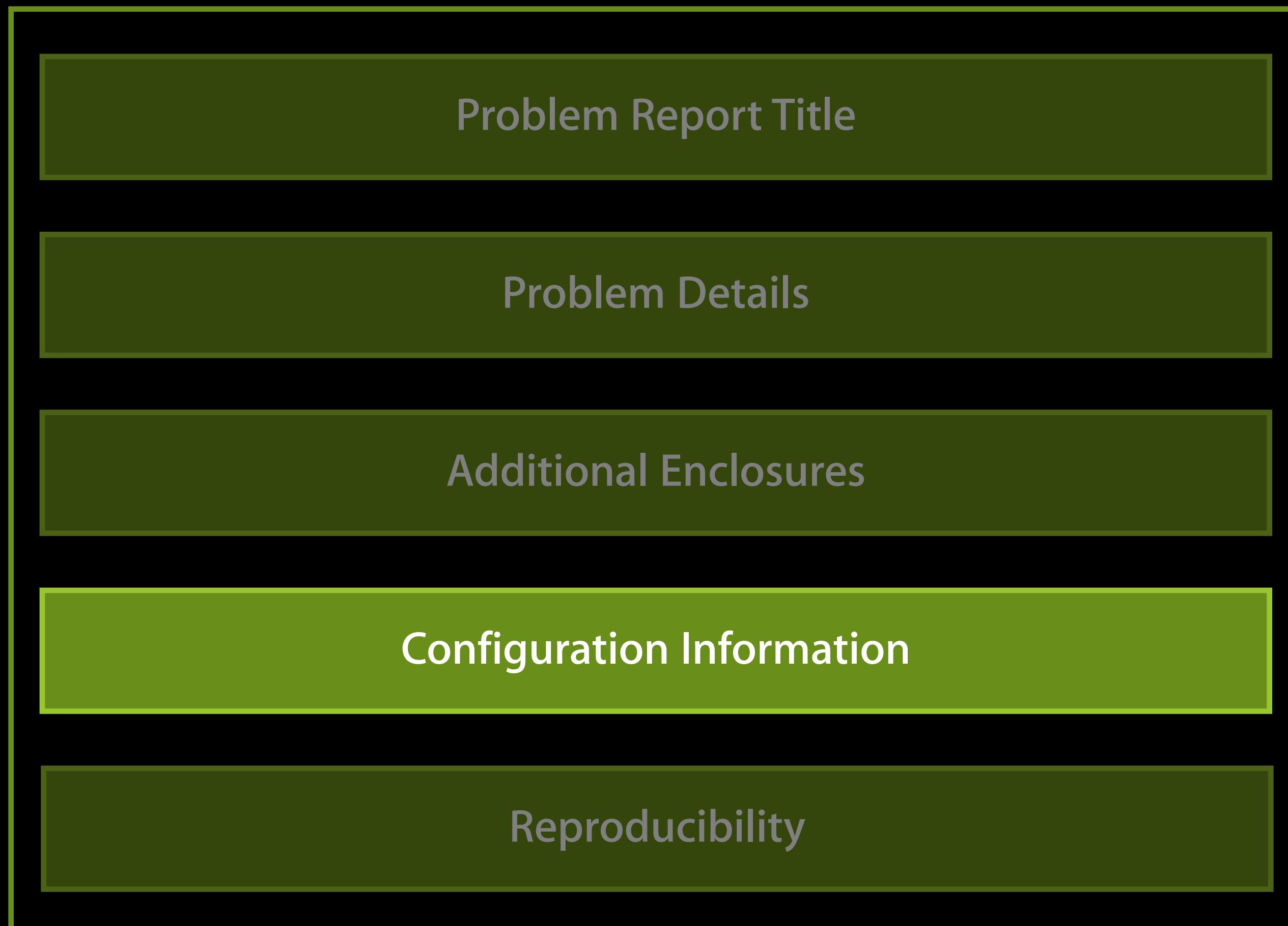
Benefit of Adding More Information



Benefit of Adding More Information



Components of a Bug Report



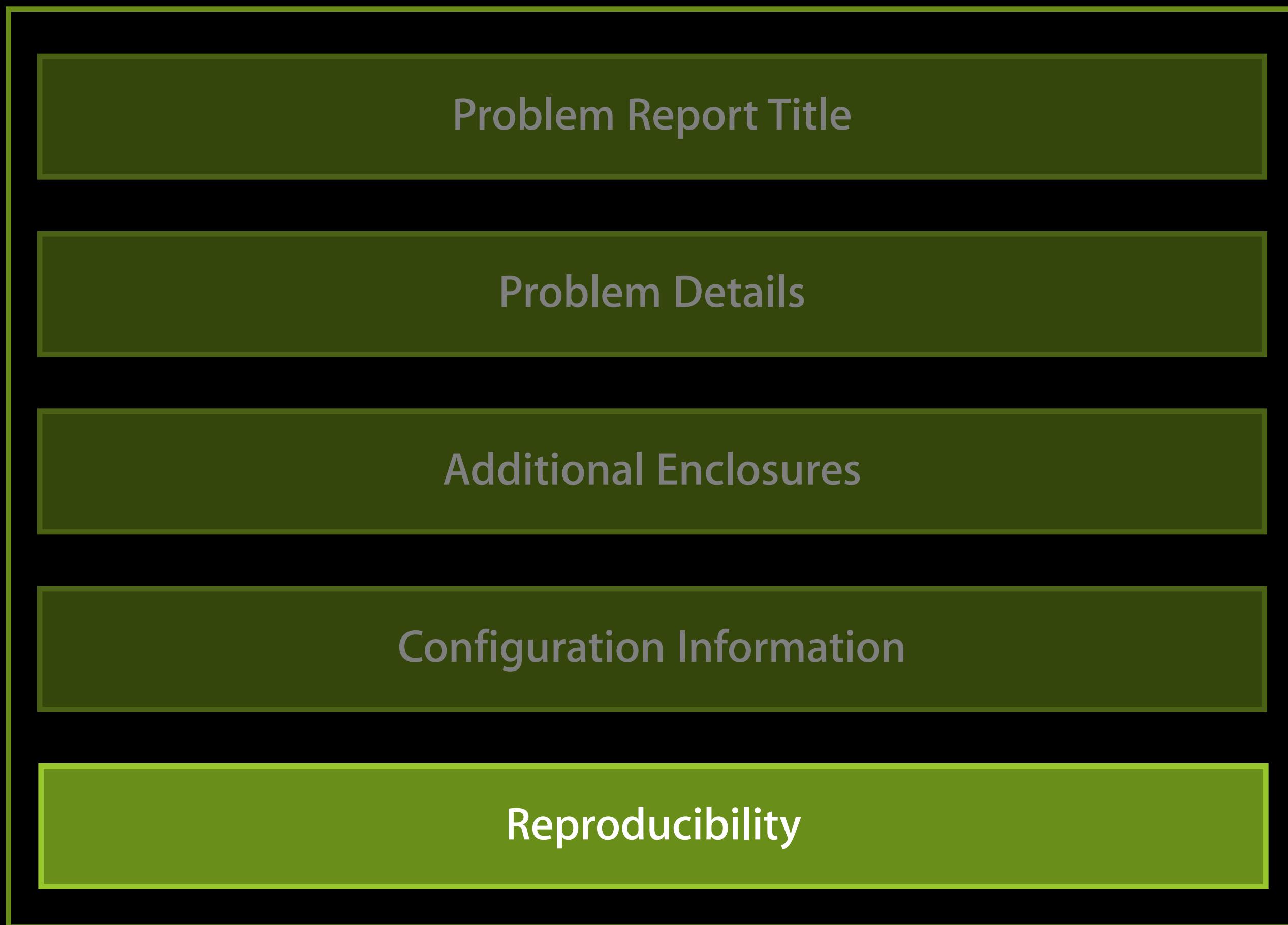
Configuration Information

- iOS:
 - Hardware model and version (iPod touch, 5th generation)
 - Software version
 - Available in Version field in Settings app under General→About
 - Other related hardware (Bluetooth headsets, automobiles)

Configuration Information

- OS X:
 - System information report
 - More info at developer.apple.com/bug-reporting/mac/system

Components of a Bug Report



What We Learned

- Importance of bug reports
- Timing is key
- Good bug filing practices

Developer Technical Support

Ted Jucevic
DTS Manager

What We Do

- One : One support
 - Public APIs
 - Xcode
 - MFi developers
- Implementation guidance
 - Not “how do I write an app?”
- Troubleshooting

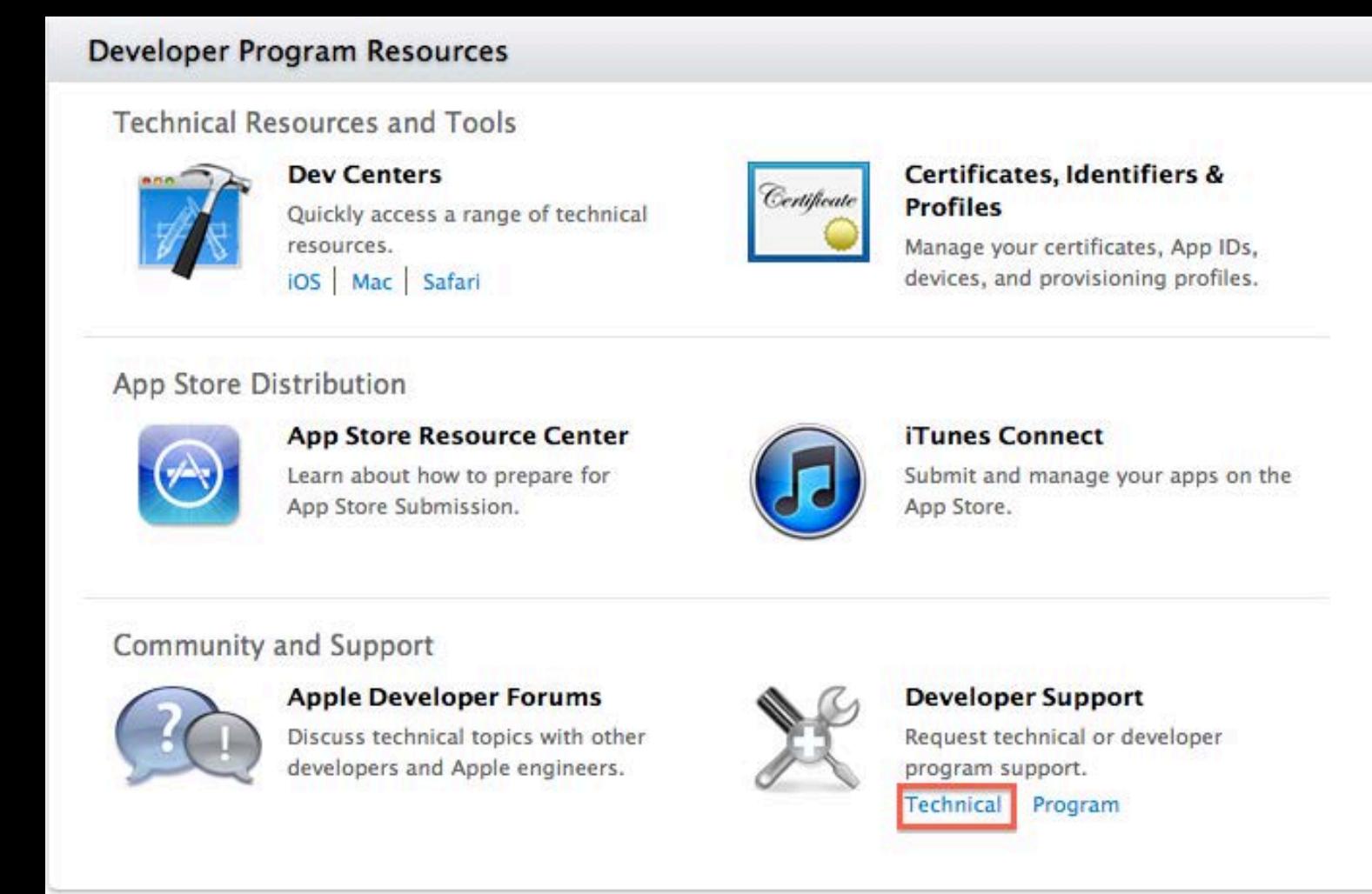
When to Contact DTS

- You've read though the docs
- You've read the forums
- You've written the test code
- Question not suitable for forums

Working with DTS

Requesting Help

- developer.apple.com/membercenter
 - Technical Support
- Fill out the web form
- Responses through email



Information We May Need

- Xcode sample project
- Clear and concise steps
- Results of your troubleshooting
- Symbolicated Crash Logs
- Console Logs
- Bug report numbers

Common Questions Answered

iOS Answers

TN2239

iOS Debugging Magic

TN2151

Understanding and Analyzing iOS Application Crash Reports

QA1747

Debugging Deployed iOS Apps

TN2285

Testing iOS App Updates

QA1764

How to reproduce bugs reported against App Store submissions

OS X Answers

TN2124

Mac OS X Debugging Magic

QA1778

How to reproduce bugs reported against Mac App Store

Deploying Your App

App Store

Overview

- Available at:
 - developer.apple.com/appstore
- Learn about
 - Approval process
 - Guidelines
- Marketing assets
 - Downloads and guidelines

The screenshot shows the 'App Store Resource Center' page. At the top, there's a navigation bar with links for Technologies, Resources, Programs, Support, and Member Center. Below the header, the title 'App Store Resource Center' is displayed, followed by a subtitle: 'Find details on everything you need to know about distributing your app on the App Store and Mac App Store — from how to prepare for app submission to managing your app once it's been posted.' The main content area is divided into several sections: 'Essential Guidelines' (with links to iTunes Connect Developer Guide, App Review Guidelines, App Store Marketing Guidelines, and Mac App Store Marketing Guidelines); 'Apple Product Images and Badges' (with links to iOS Devices and App Store Badges, and Mac Computers and Mac App Store Badges); 'Contact Us' (with links to App Review Questions and Status, Submit a Rejection Appeal, and Request an Expedited App Review); and four large cards at the bottom: 'Develop Custom B2B Apps' (describing selling custom B2B apps to business customers), 'iTunes Connect Mobile' (describing app submission updates and sales/trend data on mobile devices), 'Promote Your Apps' (describing using App Store badges and product images), and 'iOS Developer Opportunities' (describing revenue creation through iAd, App Store, and Apple Pay).

Contact Form

- Available at:
 - developer.apple.com/appstore/contact

The screenshot shows a web page titled "Contact the App Review Team" from the Apple Developer website. At the top, there's a navigation bar with links for Technologies, Resources, Programs, Support, and Member Center, along with a search bar and a user profile for "Nahir Khan". Below the navigation is a sub-header "App Store Resource Center". The main content area has a heading "Contact the App Review Team". Underneath it, there's a section titled "Topic" with a placeholder "Start by choosing a topic." A dropdown menu titled "I would like to:" is open, showing several options: "get the status of my app" (which is highlighted in blue), "get clarification on an app rejection", "appeal an app rejection", "request an expedited app review", "provide requested information", "ask a technical question about my app", "ask about using iTunes Connect", and "ask about something else". At the bottom of the page, there's a "More Ways to Get Help" section with a note: "To ask a question about using iTunes Connect or get code-level technical assistance with your app, view our main [contact us](#) page to get started." There are also icons for a wrench and a question mark, and a link to the "Contact us" page. The footer includes links to "Apple Developer", "App Store Resource Center", and "Contact the App Review Team".

Contact Form

- Use the contact form to:
 - Get the status of your app
 - Get clarification of an app rejection
 - Appeal an app rejection
 - Request an expedited app review
 - Provide requested information

iTunes Connect

iTunes Connect

- Everything about managing your app

The screenshot shows the iTunes Connect homepage with a grid of icons and descriptions:

- Sales and Trends**: View and download your sales and trends information.
- Contracts, Tax, and Banking**: Manage your contracts, tax, and banking information.
- Payments and Financial Reports**: View and download your earnings, payments, and financial reports.
- Manage Users**: Add, view, and manage iTunes Connect users and In-App Purchase test accounts.
- Manage Your Apps**: Add, view, and manage your App Store apps.
- iAd Network**: View ad performance and manage the ads that appear in your apps.
- Catalog Reports**: Request catalog reports for your App Store content.
- Developer Forums**: Find solutions and share tips with Apple developers from around the world.
- Contact Us**: Find answers or submit a question to an App Store representative.

At the bottom, there are links to "Access the Developer Guide." and "FAQs".

iTunes Connect FAQ

- Great starting out point

The screenshot shows the iTunes Connect interface with a dark background. At the top left is the "iTunes Connect" logo. On the right, there is a user profile icon labeled "nahir_khan ▾". Below the header, a horizontal navigation bar contains the text "Frequently Asked Questions". The main content area is organized into two columns of three items each. The first column includes "Getting Started" (general questions about using iTunes Connect), "Payments and Financial Reports" (information about payments and financial reports), and "Manage Users" (instructions for managing users). The second column includes "Banking and Tax" (tax form requirements and bank account setup), "App Store" (information about the App Store and customer download issues), and "Promo Codes" (instructions for requesting and using promo codes).

Getting Started	Banking and Tax
General questions including: how to use iTunes Connect and login and account issues.	Get information relating to tax form requirements, assistance completing tax forms and setting up your bank account on iTunes Connect.
Payments and Financial Reports	App Store
Get more information about payments as well as obtaining and understanding your financial reports.	Learn more about the App Store, how your applications are organized and displayed, and how to address reported customer download issues
Manage Users	Promo Codes
Learn how to add and edit your iTunes Connect users	Understand what it means to request promo codes and how they can be used
Manage Your Apps	
Learn how to manage and edit your apps using iTunes Connect	

Contact Us Page

- Drill down menu
- Where to get help

iTunes Connect Contact Us

nahir_khan ▾

Contact Us

Welcome to the iTunes Contact Us module. Please select the most relevant options from the dropdown menu(s) to help us precisely identify your issue. You may click on the "Start Over" button at any point within Contact Us to return you to the first Contact Us screen.

1. Please choose from the following topics:

2. Please choose from the following topics:

3. Please choose from the following topics:

Frequently Asked Questions:

Where do I send completed Canadian tax form(s)?
Original completed and signed Canadian tax forms should be sent to:

Apple Canada, Inc.
c/o Apple Inc.
MS: 198-2RA
12545 Riata Vista Circle
Austin, TX 78727-6524 USA

If this has not addressed your issue, [contact us](#).

Related FAQs:

- Am I required to complete the Canadian tax forms?
- Can I fax or email you my Canadian tax form(s)?
- I live in Quebec. Can I just complete the Quebec tax form?

[Start Over](#) [Done](#)

[Home](#) | [FAQs](#) | [Contact Us](#) | [Sign Out](#)
Copyright © 2013 Apple Inc. All rights reserved. [Terms of Service](#) | [Privacy Policy](#)

More Information

- For iOS Developers:
 - developer.apple.com/library/ios/iTunesConnectGuide
- For Mac Developers
 - developer.apple.com/library/mac/iTunesConnectGuide

Summary

- Different developer resources and when to use them
 - Self help
 - Working with Apple
 - App deployment
- Importance of bug reports
 - Behind the scenes
 - Filing good bug reports

More Information

Developer Forums:
devforums.apple.com

App Review Contact Form:
developer.apple.com/appstore

Developer Technical Support:
developer.apple.com/support/technical

Bug Reporter:
bugreport.apple.com

More Information on Bug Reporting:
developer.apple.com/bug-reporting

More Information

TN2239 - iOS Debugging Magic

developer.apple.com/library/ios/#technotes/tn2239/_index.html

TN2151 - Understanding and Analyzing iOS Application Crash Reports

developer.apple.com/library/ios/#technotes/tn2151/_index.html

QA1747 - Debugging Deployed iOS Apps

developer.apple.com/library/ios/#qa/qa1747/_index.html

TN2285 - Testing iOS App Updates

developer.apple.com/library/ios/#technotes/tn2285/_index.html

More Information

QA1764 - How to reproduce bugs reported against App Store submissions

developer.apple.com/library/ios/#qa/qa1764/_index.html

TN2124 - Mac OS X Debugging Magic

developer.apple.com/library/mac/#technotes/tn2124/_index.html

QA1778 - How to reproduce bugs reported against Mac App Store submissions

developer.apple.com/library/mac/#qa/qa1778/_index.html

