

# 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

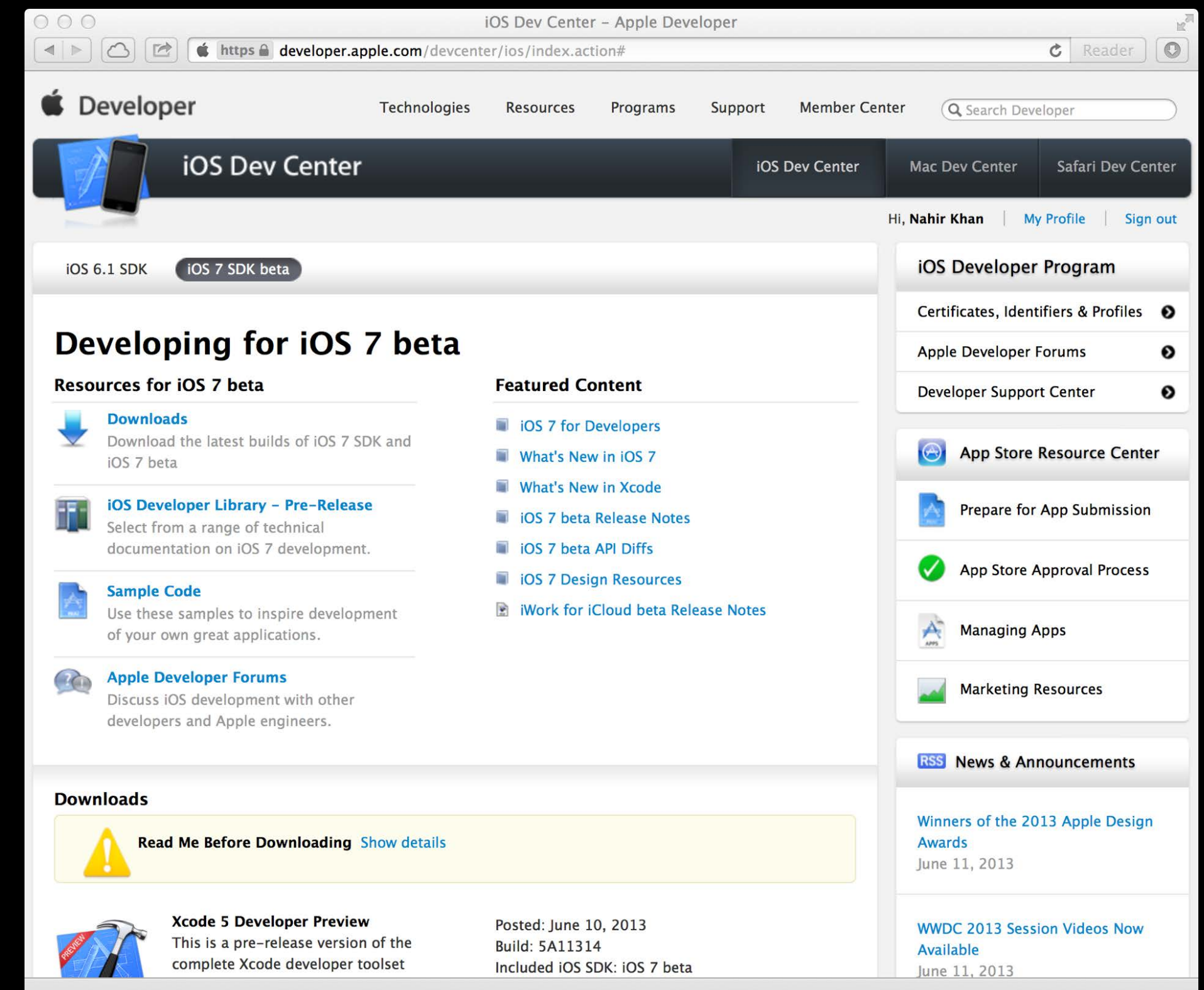
- Self help
  - Documentation
  - Developer Forums
- Working with Apple
  - Bug Reporting
  - Developer Technical Support (DTS)
- Deploying Your App
  - App Store
  - iTunes Connect

Self Help

Documentation

# Documentation and More

- Also available from
  - [developer.apple.com](https://developer.apple.com)
- WWDC and Tech Talk videos
- Documentation
  - Getting Started
  - Developer Library
- Sample code and projects



The screenshot displays the iOS Dev Center website for Apple Developer. The page is titled "iOS Dev Center" and features a navigation bar with links for Technologies, Resources, Programs, Support, and Member Center. The main content area is focused on "Developing for iOS 7 beta" and includes sections for "Resources for iOS 7 beta" (Downloads, iOS Developer Library - Pre-Release, Sample Code, Apple Developer Forums) and "Featured Content" (iOS 7 for Developers, What's New in iOS 7, iOS 7 beta Release Notes, iOS 7 beta API Diffs, iOS 7 Design Resources, iWork for iCloud beta Release Notes). A "Downloads" section at the bottom highlights the "Xcode 5 Developer Preview" with details: "This is a pre-release version of the complete Xcode developer toolset", "Posted: June 10, 2013", "Build: 5A11314", and "Included iOS SDK: iOS 7 beta". The right sidebar contains the "iOS Developer Program" menu, "App Store Resource Center", and "News & Announcements" section, which includes "Winners of the 2013 Apple Design Awards" and "WWDC 2013 Session Videos Now Available".



# Class Reference

- API reference
- Example code
- Common issues and FAQ

The screenshot displays the 'NSAttributedString Class Reference' page from the iOS Developer Library. The page is structured with a left-hand navigation menu and a main content area. The navigation menu includes 'Table of Contents', 'Overview', 'Adopted Protocols', 'Tasks', 'Instance Methods', 'Constants', and 'Revision History'. The main content area features a 'Jump To...' search bar and a 'Next' link. Below the search bar, the title 'NSAttributedString Class Reference' is prominently displayed. A table of class relationships follows, detailing inheritance, conformance, framework, availability, companion guide, declaration, and related sample code. The 'Overview' section provides a detailed description of the class, its methods, and its usage in various contexts, including text handling and formatting.

Inherits from	<a href="#">NSObject</a>
Conforms to	<a href="#">NSCoding</a> <a href="#">NSCopying</a> <a href="#">NSMutableCopying</a> <a href="#">NSObject (NSObject)</a>
Framework	<a href="#">/System/Library/Frameworks/Foundation.framework</a>
Availability	Available in iOS 3.2 and later.
Companion guide	<a href="#">Attributed String Programming Guide</a>
Declared in	<a href="#">NSAttributedString.h</a>
Related sample code	<a href="#">CoreTextPageViewer</a> <a href="#">SimpleTextInput</a>

### 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.

iOS Dev Center » iOS Developer Library » Framework » Core Services Layer » Foundation



# Full Programming Guides

- Entire books worth of content

The screenshot shows the 'Text Programming Guide for iOS' page in the iOS Developer Library. The page title is 'Text Programming Guide for iOS' and it includes a 'PDF' icon. The left sidebar contains a 'Table of Contents' with sections like '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', and 'Drawing and Managing Text Revision History'. Below this is an 'OBJECTIVE-C REFERENCE' section listing classes like 'NSLayoutManager', 'NSLayoutManagerDelegate', 'NSString UIKit Additions Reference', 'NSTextAttachment', and 'NSTextAttachmentContainer'. The main content area is titled 'About Text Editing in iOS' and features a 'Next' link. A prominent 'Important' note states that the document is preliminary and confidential. The main text explains that iOS provides various ways to display and edit text, mentioning framework objects like text views and text fields. A 'Note' at the bottom indicates the document was previously titled 'Text, Web, and Editing Programming Guide for iOS'.

iOS Developer Library — Pre-Release

Developer

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

Next

**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.

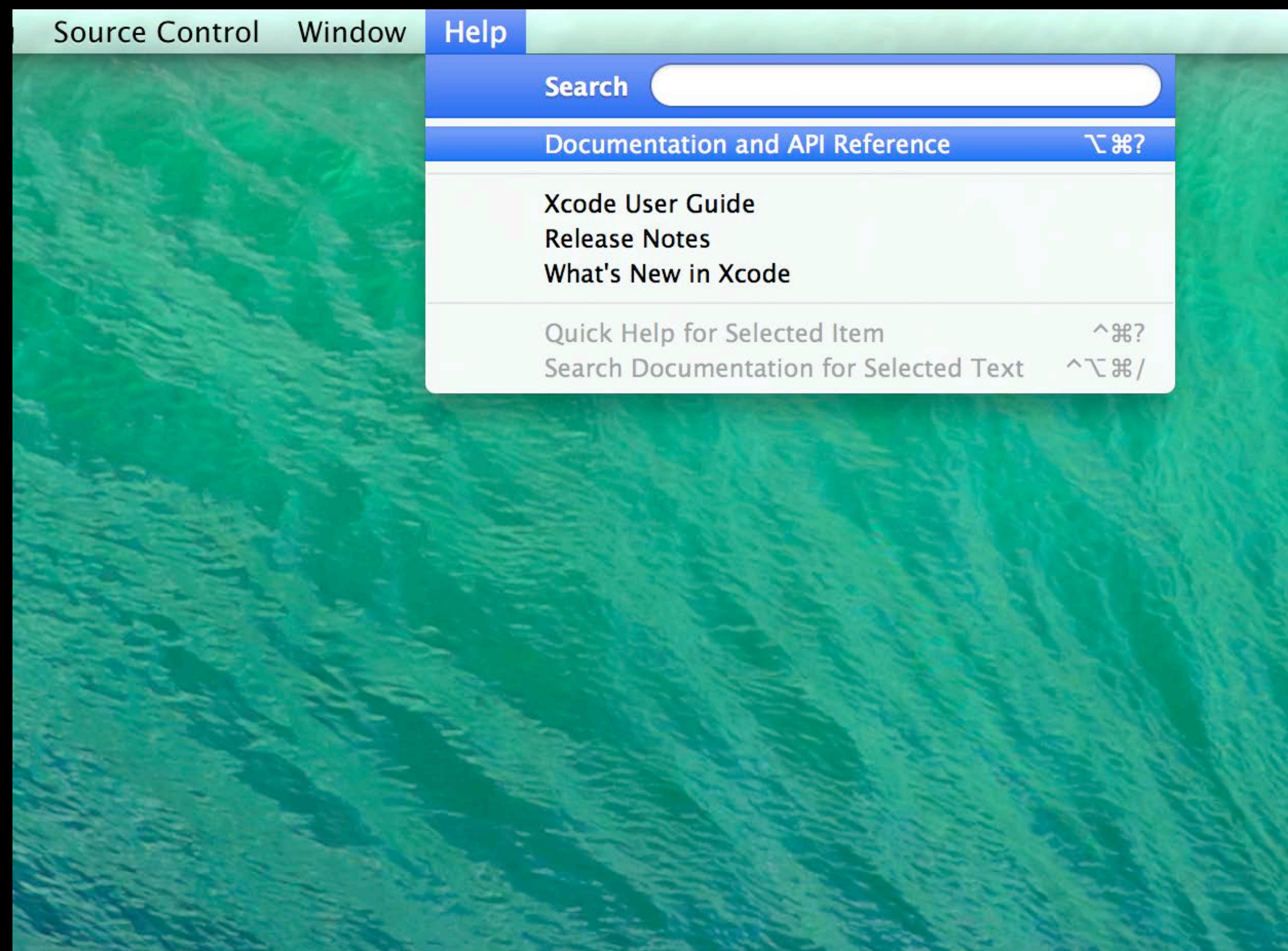
With the classes in the UIKit framework, you can manage the edit menu (including adding custom items to it), implement custom input views, and copy, cut, and paste data within and between applications.

**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

The screenshot shows a symbol definition popup for the `UINavigationController.navigationItem` property. The popup is overlaid on a code editor showing the implementation of `pushViewController:animated:` in `UINavigationController.m`. The popup contains the following information:

- Declaration:** `@property(n nonatomic, readonly, retain) UINavigationController *navigationItem`
- Description:** The navigation item used to represent the view controller in a parent's navigation bar. (read-only)  
This is a unique instance of `UINavigationController` created to represent the view controller when it is pushed onto a navigation controller. The first time the property is accessed, the `UINavigationController` object is created. Therefore, you shouldn't access this property if you are not using a navigation controller to display the view controller. To ensure the navigation item is configured, you can either override this property and add code to create the bar button items when first accessed or create the items in your view controller's initialization code.  
Avoid tying the creation of bar button items in your navigation item to the creation of your view controller's view. The navigation item of a view controller may be retrieved independently of the view controller's view. For example, when pushing two view controllers onto a navigation stack, the topmost view controller becomes visible, but the other view controller's navigation item may be retrieved in order to present its back button.  
The default behavior is to create a navigation item that displays the view controller's title.
- Availability:** Available in iOS 2.0 and later.
- Declared In:** `UINavigationController.h`
- Reference:** `UIViewController`

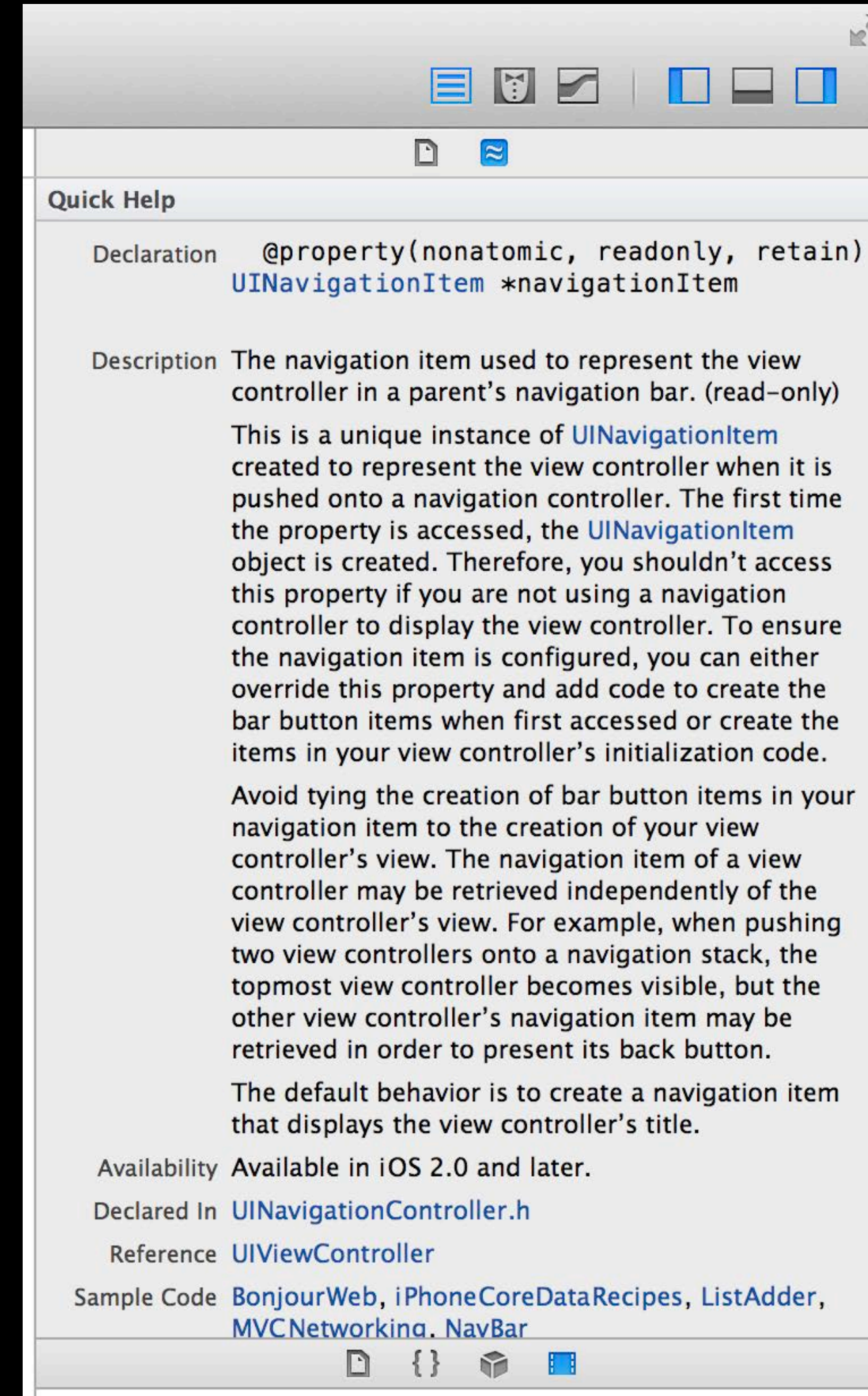
The background code editor shows the following code snippet:

```
60     barButtonItem.title = NSLocalizedString(@"Master", @"Master");
61     [self.navigationItem setLeftBarButtonItem:barButtonItem animated:YES];
62     self.masterPopoverController = popoverController;
63 }
```



# Symbol Definition (Cont.)

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



The screenshot shows the Xcode Quick Help window for the `UINavigationController.navigationItem` property. The window title is "Quick Help". The content is organized into sections: Declaration, Description, Availability, Declared In, Reference, and Sample Code. The Declaration section shows the property is declared as `@property(n nonatomic, readonly, retain) UINavigationController *navigationItem`. The Description section explains that this is a unique instance of `UINavigationController.navigationItem` created to represent the view controller when it is pushed onto a navigation controller. It also provides guidance on how to configure the navigation item and retrieve it. The Availability section states it is available in iOS 2.0 and later. The Declared In section points to `UINavigationController.h`. The Reference section points to `UIViewController`. The Sample Code section lists `BonjourWeb`, `iPhoneCoreDataRecipes`, `ListAdder`, `MVCNetworking`, and `NavBar`.

Quick Help

Declaration `@property(n nonatomic, readonly, retain) UINavigationController *navigationItem`

Description The navigation item used to represent the view controller in a parent's navigation bar. (read-only)

This is a unique instance of `UINavigationController.navigationItem` created to represent the view controller when it is pushed onto a navigation controller. The first time the property is accessed, the `UINavigationController.navigationItem` object is created. Therefore, you shouldn't access this property if you are not using a navigation controller to display the view controller. To ensure the navigation item is configured, you can either override this property and add code to create the bar button items when first accessed or create the items in your view controller's initialization code.

Avoid tying the creation of bar button items in your navigation item to the creation of your view controller's view. The navigation item of a view controller may be retrieved independently of the view controller's view. For example, when pushing two view controllers onto a navigation stack, the topmost view controller becomes visible, but the other view controller's navigation item may be retrieved in order to present its back button.

The default behavior is to create a navigation item that displays the view controller's title.

Availability Available in iOS 2.0 and later.

Declared In `UINavigationController.h`

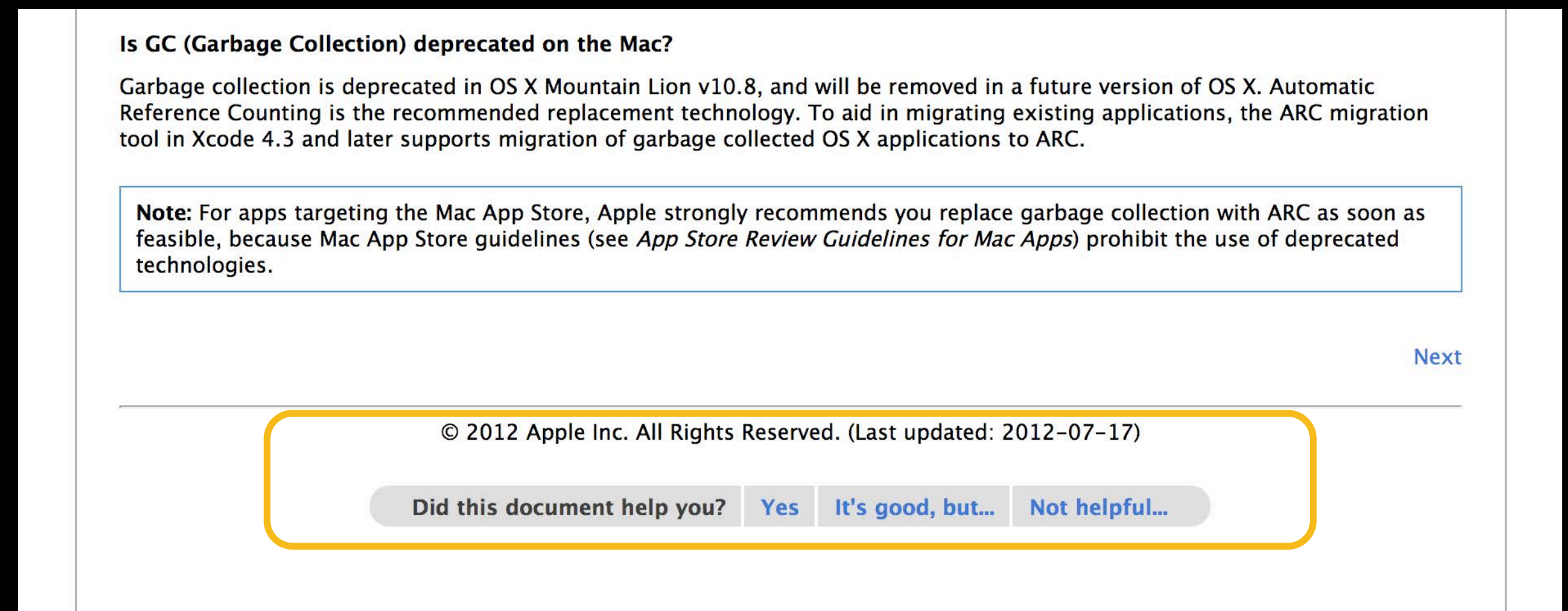
Reference `UIViewController`

Sample Code `BonjourWeb`, `iPhoneCoreDataRecipes`, `ListAdder`, `MVCNetworking`, `NavBar`



# Feedback

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



The screenshot shows the footer of a documentation page. At the top, there is a section titled "Is GC (Garbage Collection) deprecated on the Mac?" with a paragraph of text explaining that garbage collection is deprecated in OS X Mountain Lion v10.8 and will be removed in a future version. Below this is a note in a blue-bordered box: "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." To the right of the note is a "Next" link. Below the note is a copyright notice: "© 2012 Apple Inc. All Rights Reserved. (Last updated: 2012-07-17)". At the bottom is a feedback form with the question "Did this document help you?" and three buttons: "Yes", "It's good, but...", and "Not helpful...".

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)

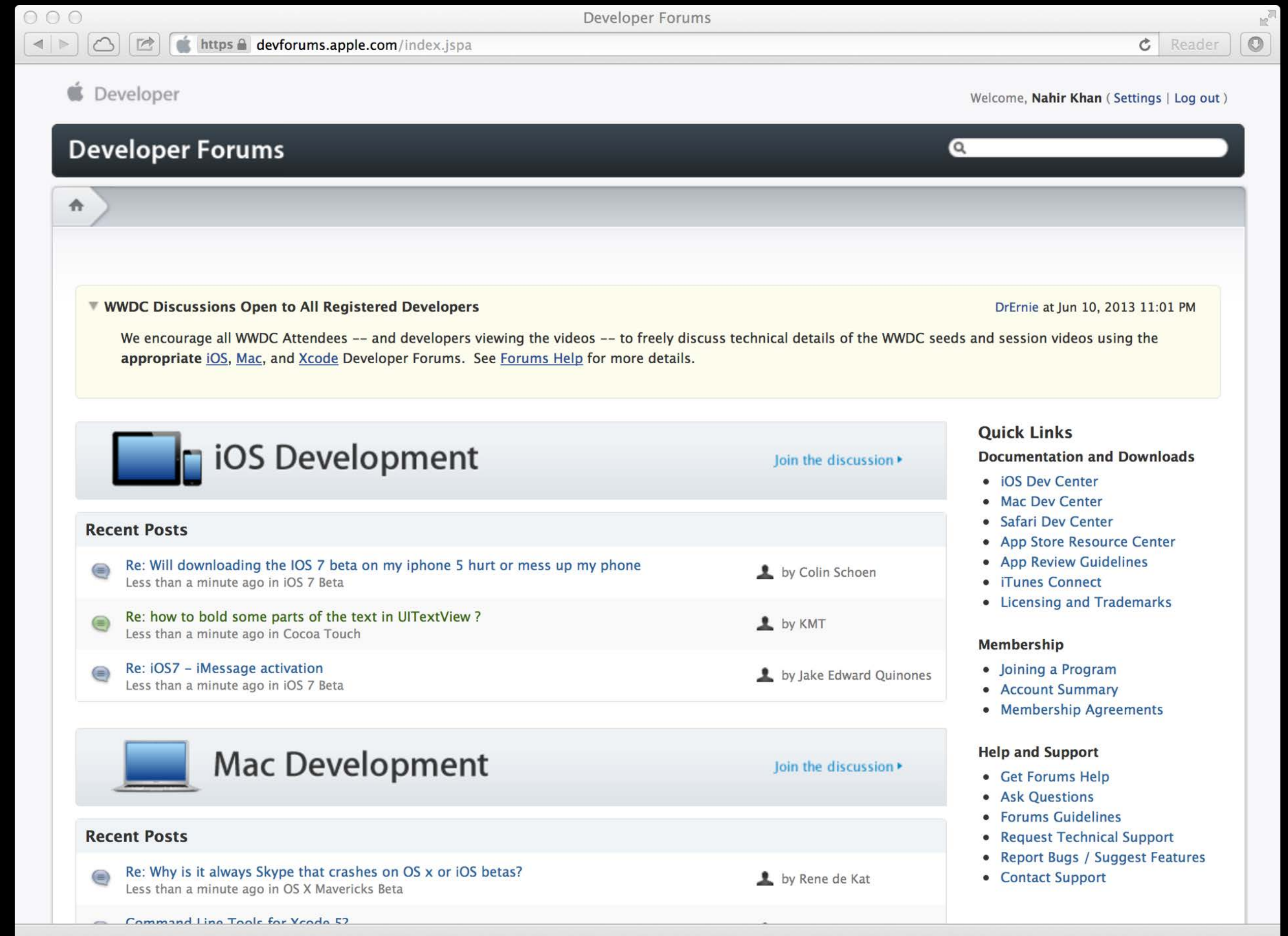
Did this document help you? Yes It's good, but... Not helpful...

# Developer Forums



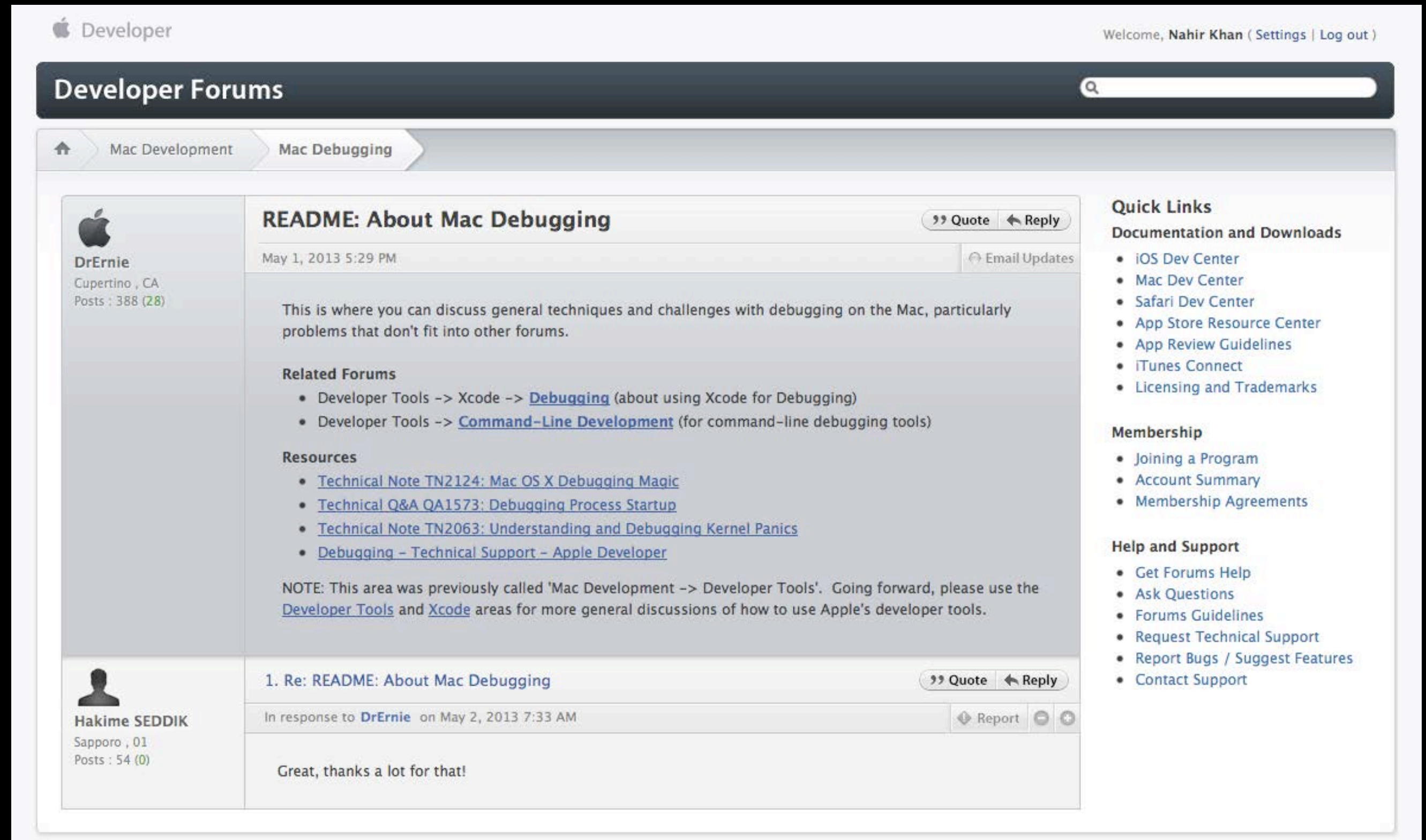
# Developer Forums

- Visit the forums at:
  - [devforums.apple.com](https://devforums.apple.com)



# Developer Forums

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



The screenshot shows the Apple Developer Forums interface. At the top, there's a navigation bar with the Apple logo and the word "Developer". On the right, it says "Welcome, Nahir Khan ( Settings | Log out )". Below this is a search bar. The main content area is titled "Developer Forums" and has a breadcrumb trail: "Mac Development" > "Mac Debugging".

The main post is titled "README: About Mac Debugging" and was posted by "DrErnie" on May 1, 2013 at 5:29 PM. The post content reads: "This is where you can discuss general techniques and challenges with debugging on the Mac, particularly problems that don't fit into other forums." It includes sections for "Related Forums" (Developer Tools -> Xcode -> Debugging, and Developer Tools -> Command-Line Development) and "Resources" (Technical Note TN2124: Mac OS X Debugging Magic, Technical Q&A QA1573: Debugging Process Startup, Technical Note TN2063: Understanding and Debugging Kernel Panics, and Debugging - Technical Support - Apple Developer). A note at the bottom states: "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."

Below the main post is a reply from "Hakime SEDDIK" on May 2, 2013 at 7:33 AM, which says "Great, thanks a lot for that!".

On the right side of the forum, there are several sections: "Quick Links" (Documentation and Downloads, Membership, Help and Support), "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), and "Help and Support" (Get Forums Help, Ask Questions, Forums Guidelines, Request Technical Support, Report Bugs / Suggest Features, Contact Support).




# Example Post

Developer Welcome, Nahir Khan ( Settings | Log out )

---

**Developer Forums** Q

Home > iOS Development > Performance Analysis and Debugging



**brunic**  
Beijing , 10  
Posts : 2 (0)

### Why calls retain in setUp() method of iPhoneUnitTests de...

Apr 18, 2013 1:39 AM Report | Email Updates

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

```
- (void) setUp {
    NSLog(@"%@ setUp", 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




**Rincewind**  
Posts : 10,585 (12)

1. Re: Why calls retain in setUp() method of iPhoneUnitTests demo? Quote | Reply

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

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



**brunic**  
Beijing , 10  
Posts : 2 (0)

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

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 Welcome, Nahir Khan ( Settings | Log out )

---

**Developer Forums** 🔍

🏠 iOS Development > Performance Analysis and Debugging



**brunic**  
Beijing , 10  
Posts : 2 (0)

### Why calls retain in setUp() method of iPhoneUnitTests de...

Apr 18, 2013 1:39 AM 🗉 Report 📧 Email Updates

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

```
- (void) setUp {
    NSLog(@"%@ setUp", 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




**Rincewind**  
Posts : 10,585 (12)

1. Re: Why calls retain in setUp() method of iPhoneUnitTests demo? 🗉 Quote 🗑 Reply

In response to [brunic](#) on Apr 19, 2013 11:43 AM 🗉 Report 🗑 🗑

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



**brunic**  
Beijing , 10  
Posts : 2 (0)

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 🗑 🗑

Thanks, Rincewind.



# Example Post

Apple Developer Welcome, Nahir Khan ( Settings | Log out )

## Developer Forums

Home > iOS Development > Performance Analysis and Debugging

**brunic**  
Beijing, 10  
Posts: 2 (0)

### Why calls retain in setUp() method of iPhoneUnitTests de...

Apr 18, 2013 1:39 AM Report Email Updates

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

```
- (void) setUp {
    NSLog(@"%@ setUp", 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

**Rincewind**  
Posts: 10,585 (12)

1. Re: Why calls retain in setUp() method of iPhoneUnitTests demo? Quote Reply

In response to **brunic** on Apr 19, 2013 11:43 AM Report Like Dislike

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

**brunic**  
Beijing, 10  
Posts: 2 (0)

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 Like Dislike

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

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?

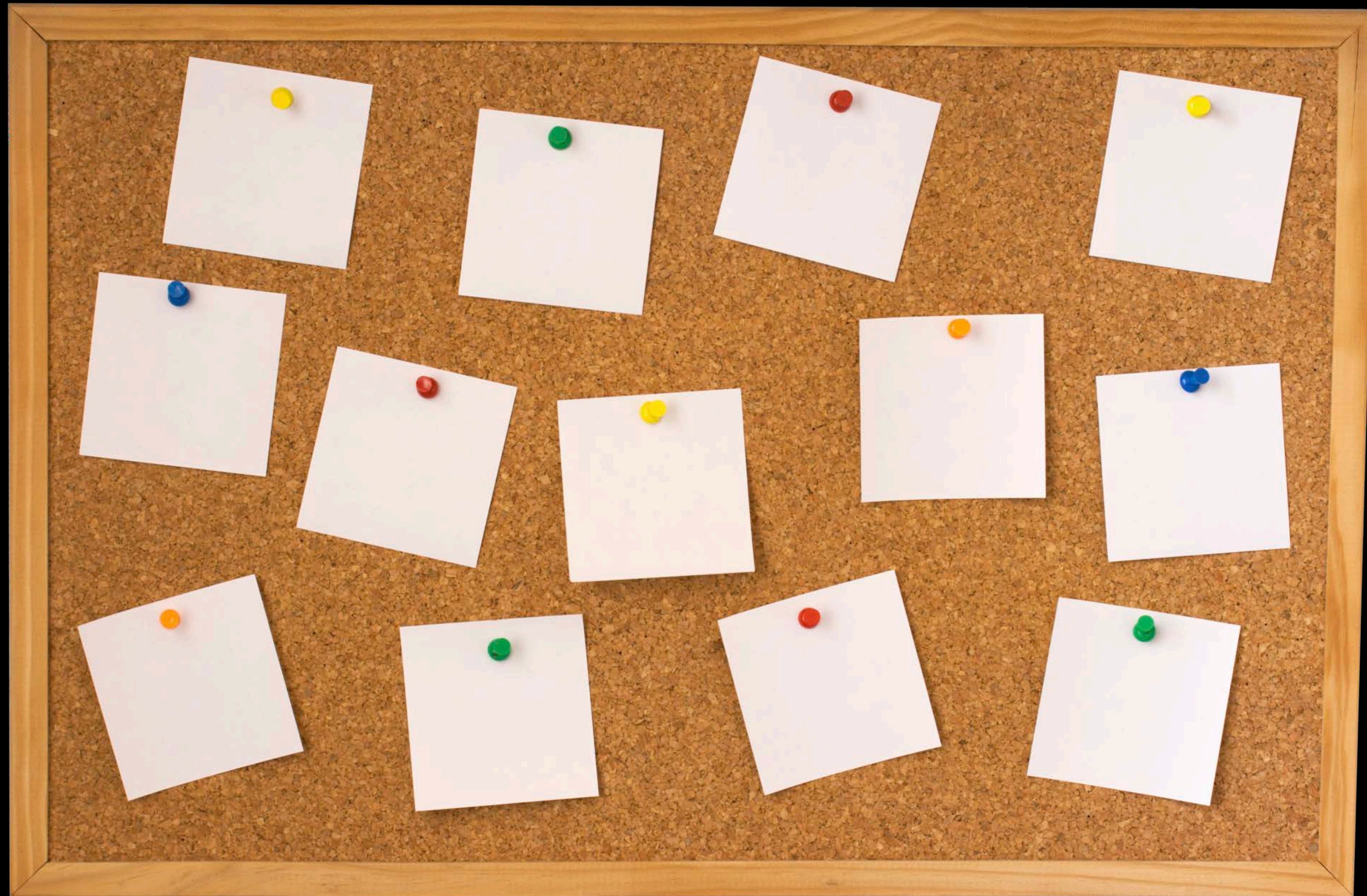
- The answer is always “Yes”, even if the issue:
  - Seems trivial
  - Happened once
  - Seems obvious
  - Someone else filed it

**“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 cat  
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

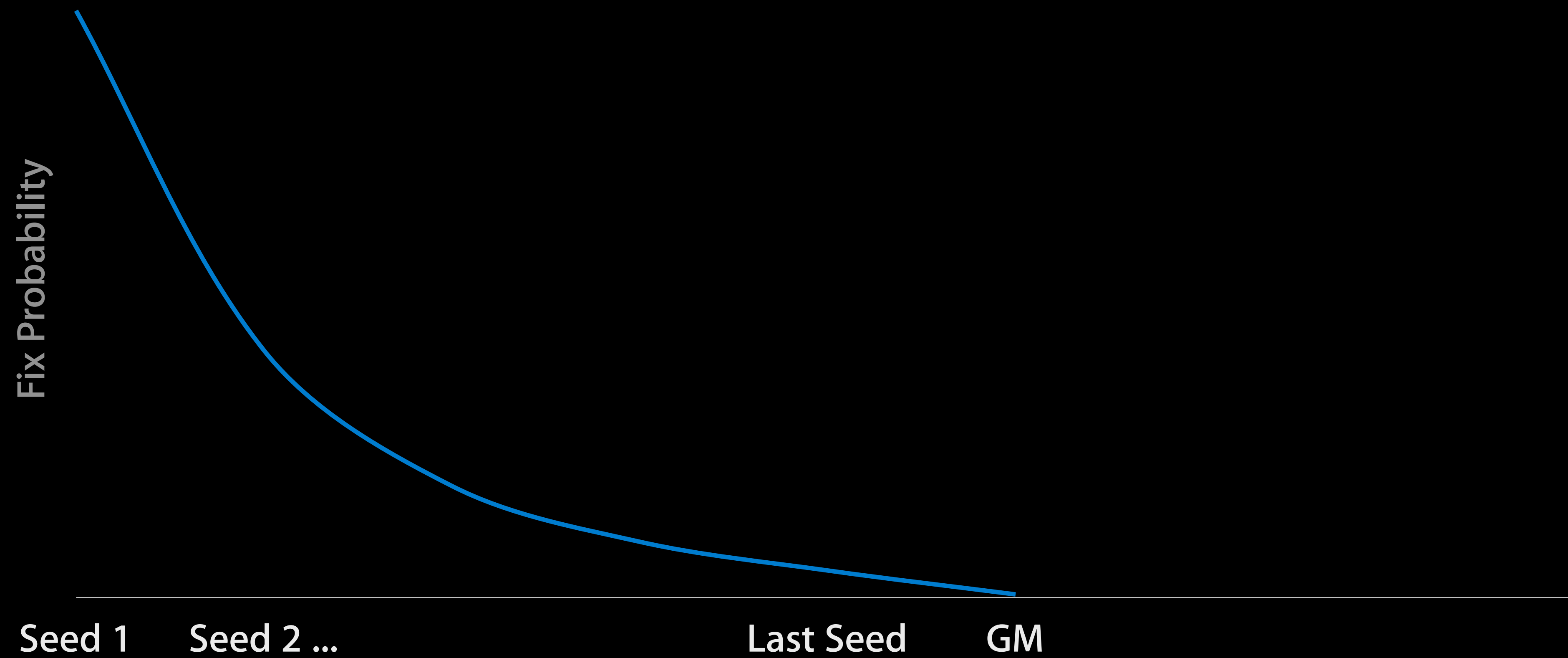
When do I file the bug report?

**As Early as Possible**

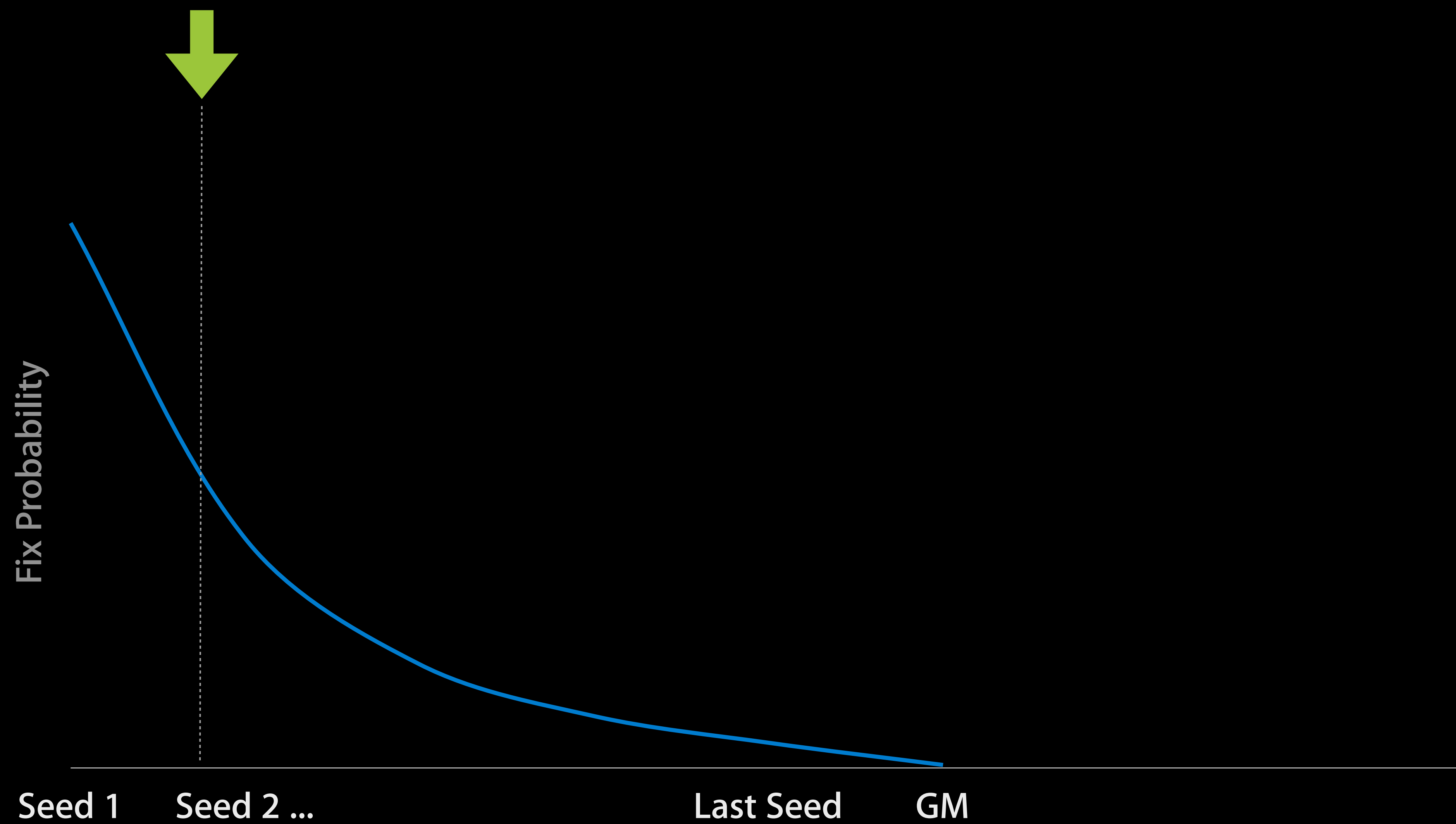
Timing Is Key



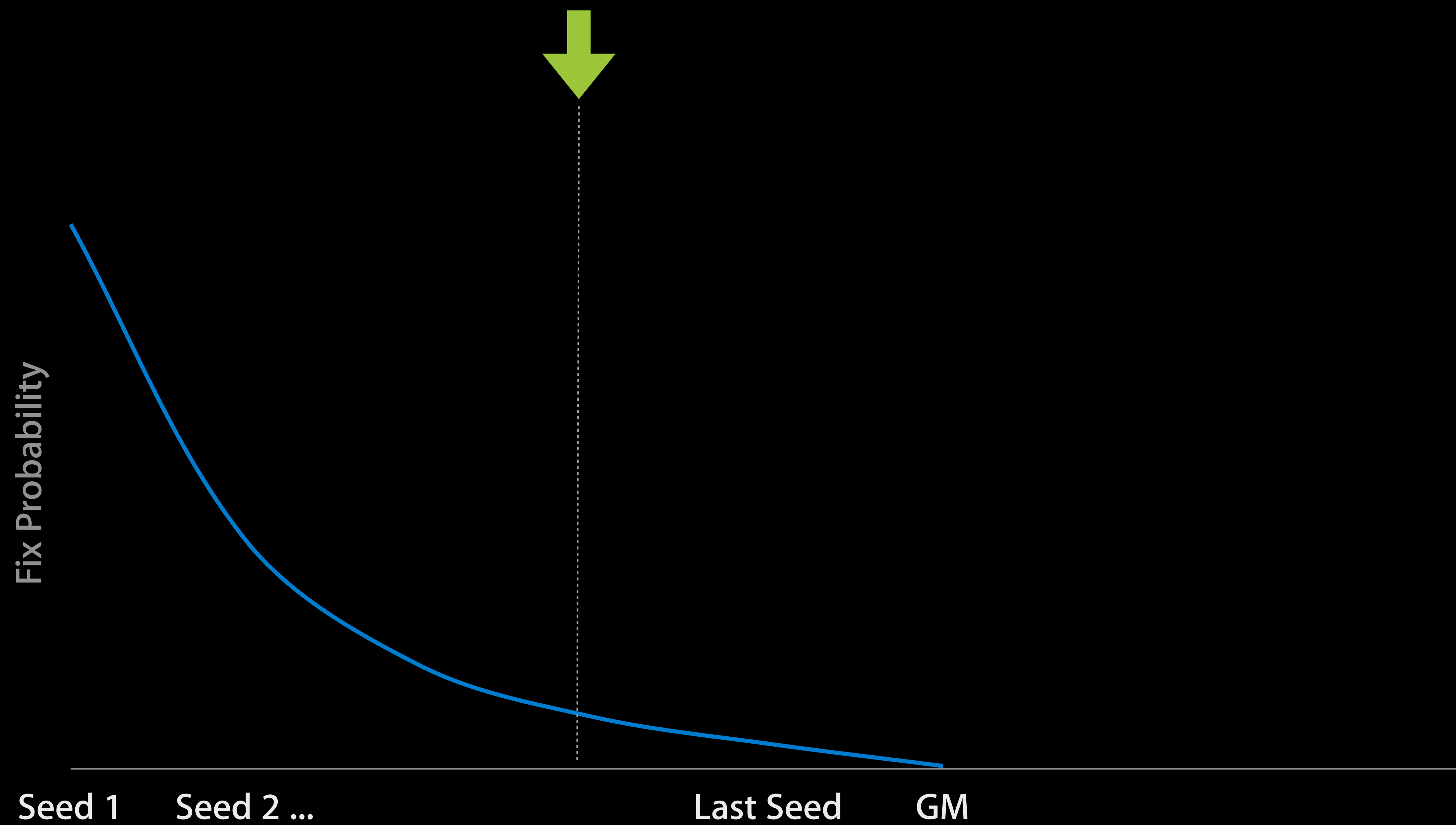
# Timing Is Key



# Timing Is Key

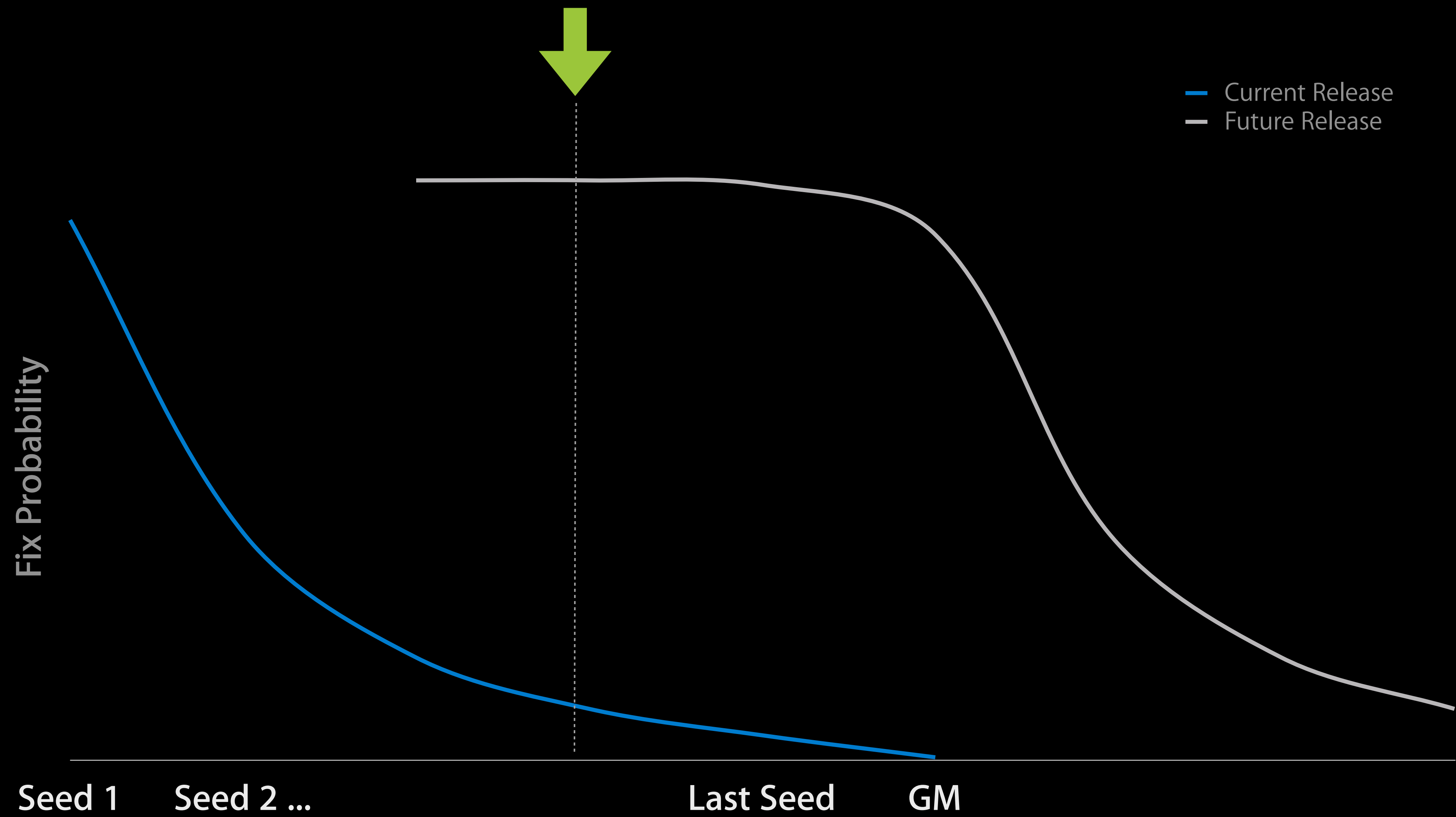


# Timing Is Key





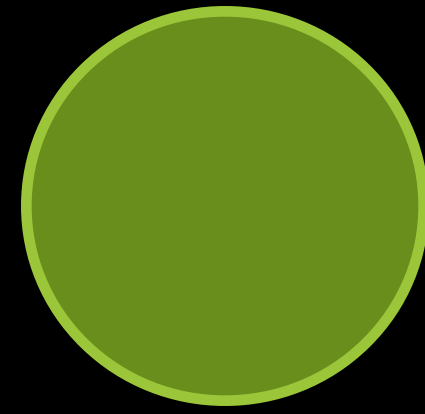
# Timing Is Key



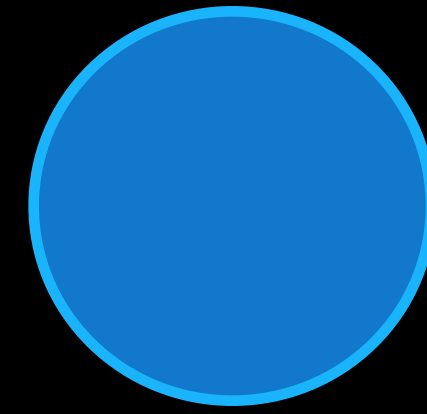
# Life of the Bug

Analyze

You

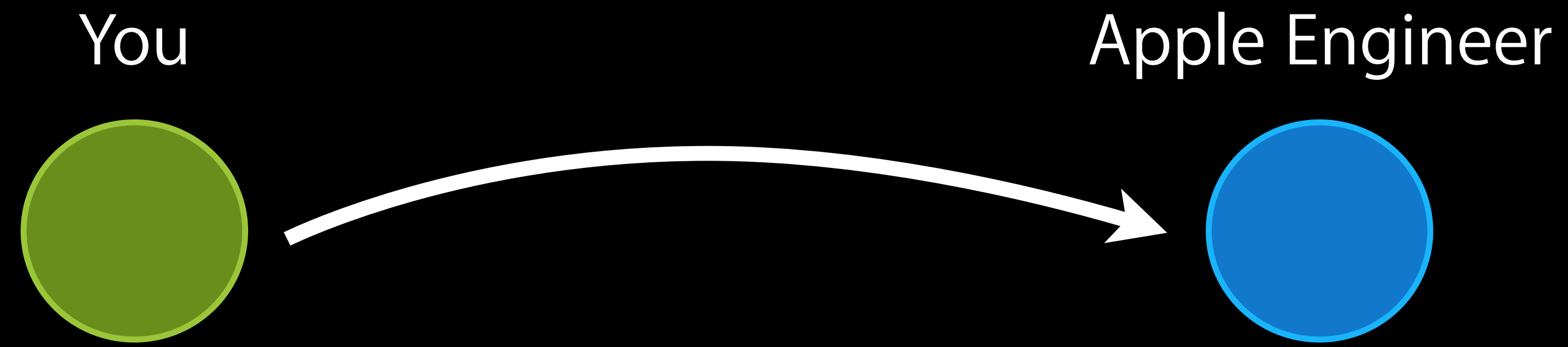


Apple Engineer



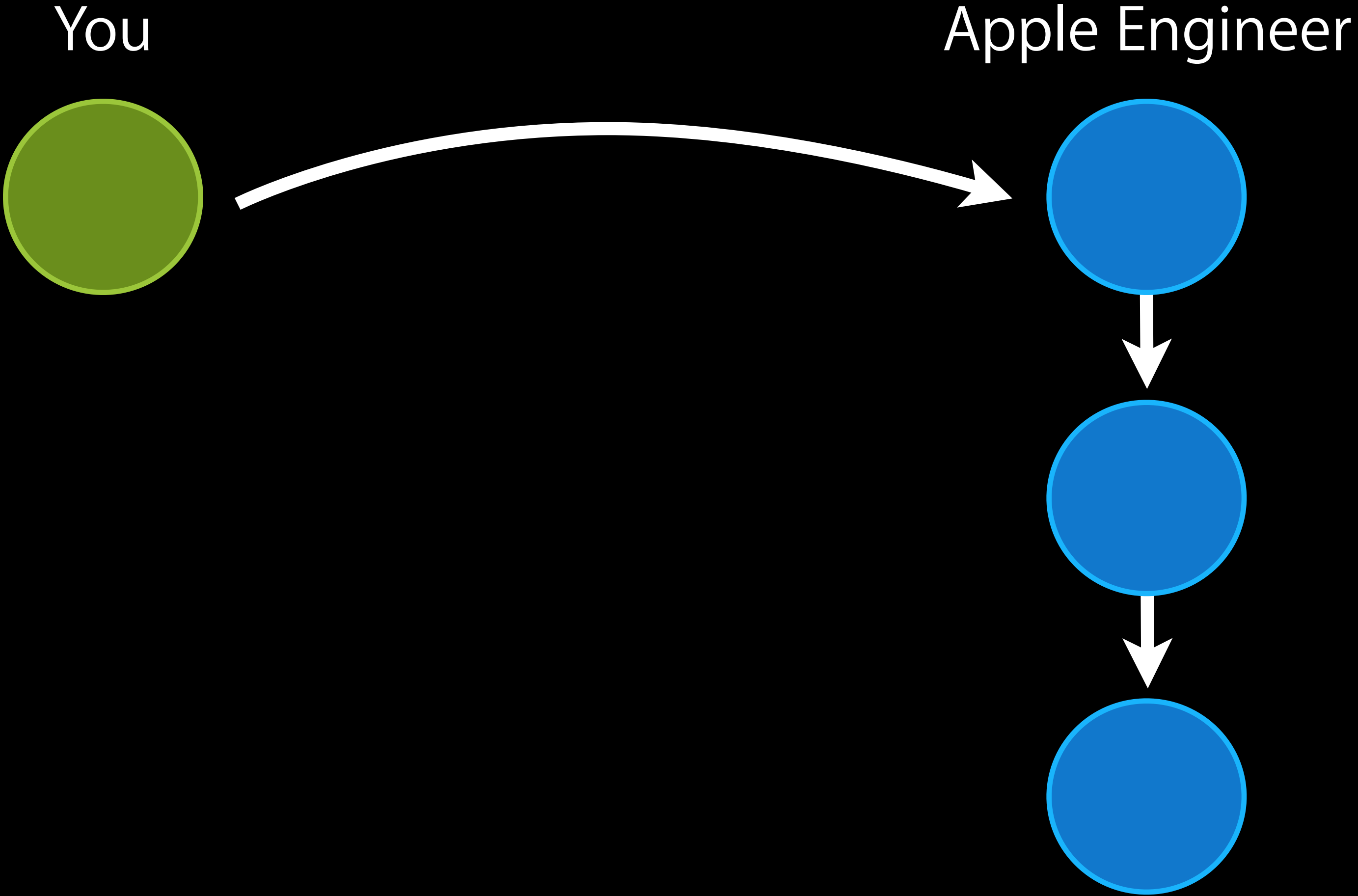
# Life of the Bug

Analyze



# Life of the Bug

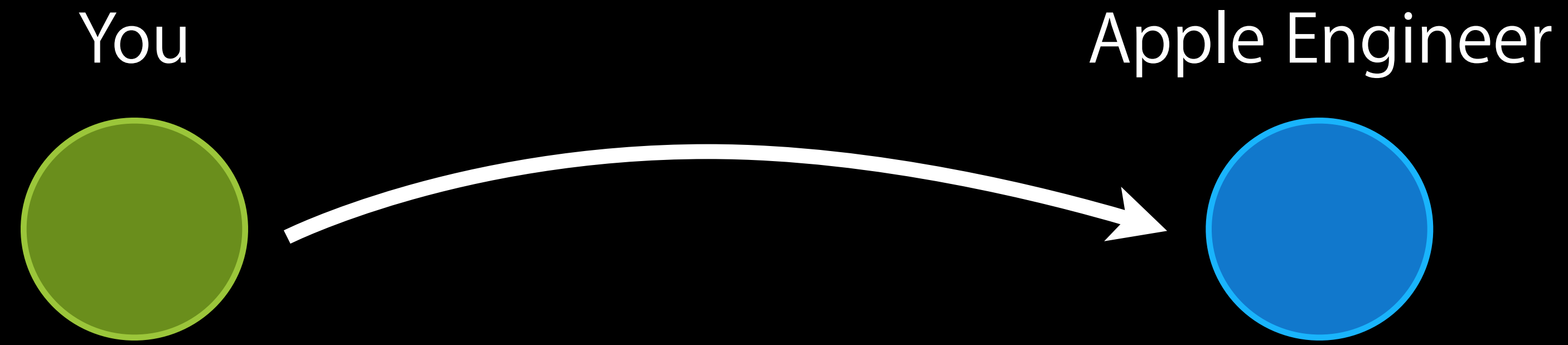
Analyze





# Life of the Bug

Analyze



# Life of the Bug

Software changed



# Life of the Bug

Duplicate



# Duplicates Are Important

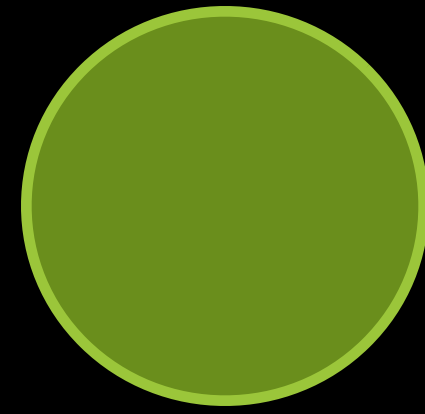
- Better information than the original
- Number of duplicates as seriousness indicator
- Request status on the original bug



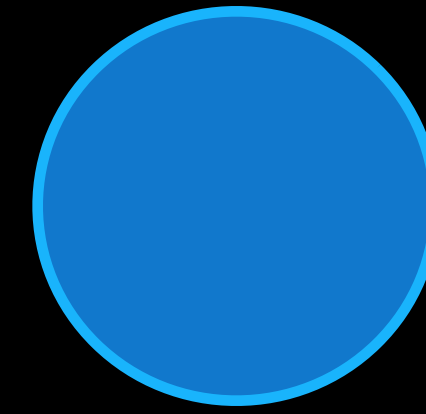
# Life of the Bug

Insufficient information

You

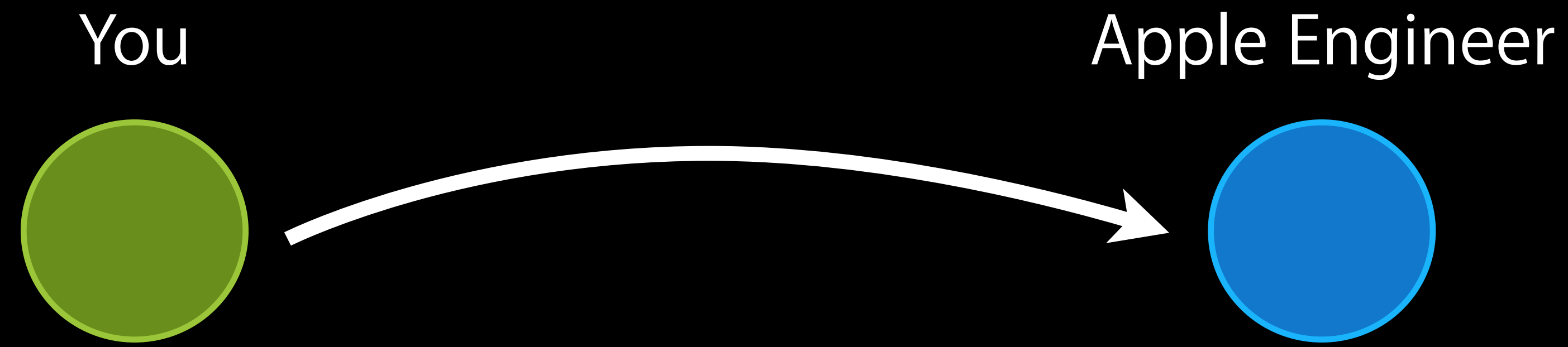


Apple Engineer



# Life of the Bug

Insufficient information

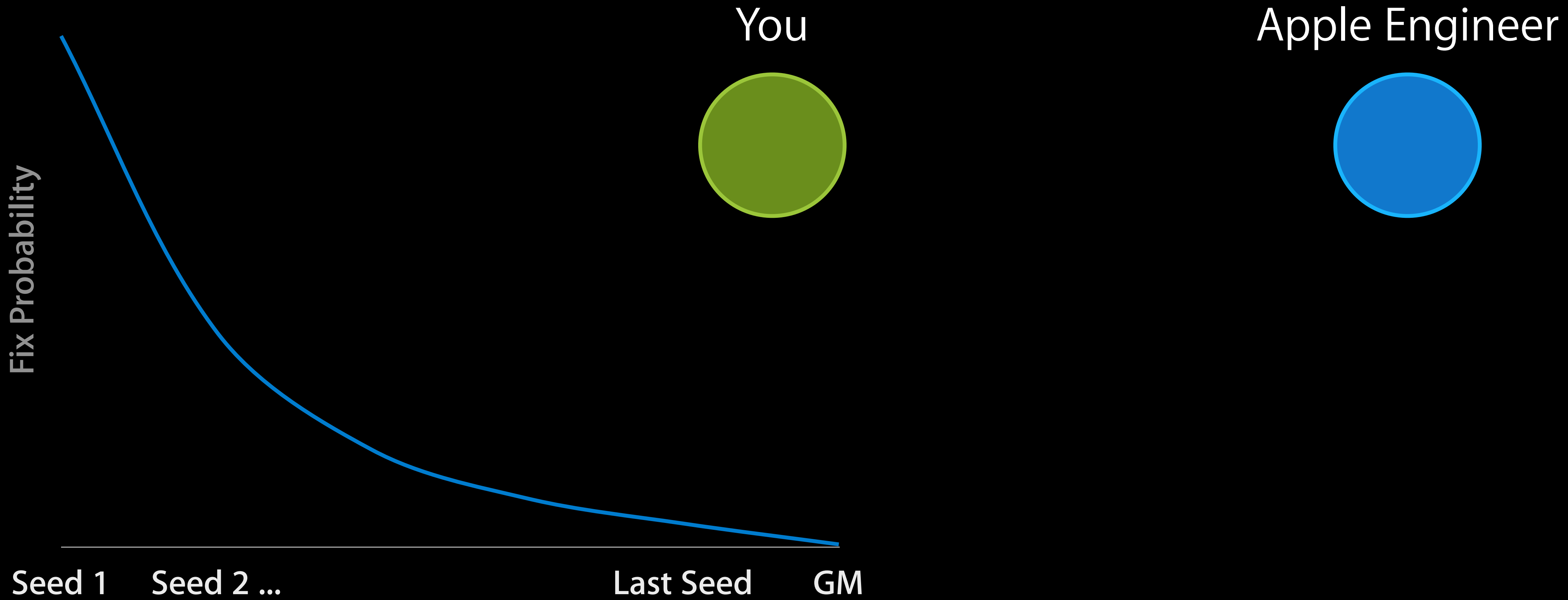


# Life of the Bug

Insufficient information

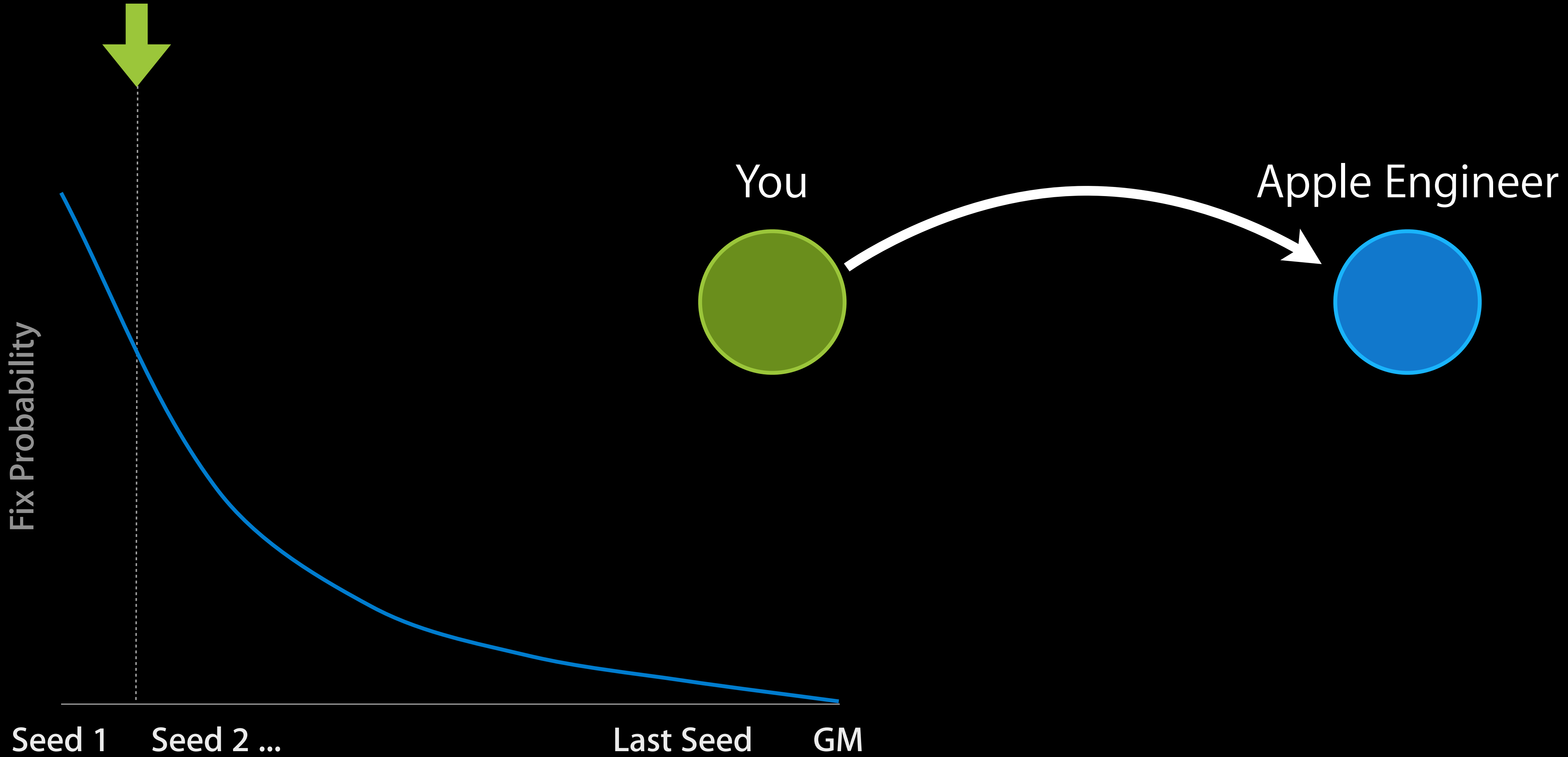


# Insufficient Information Hazards

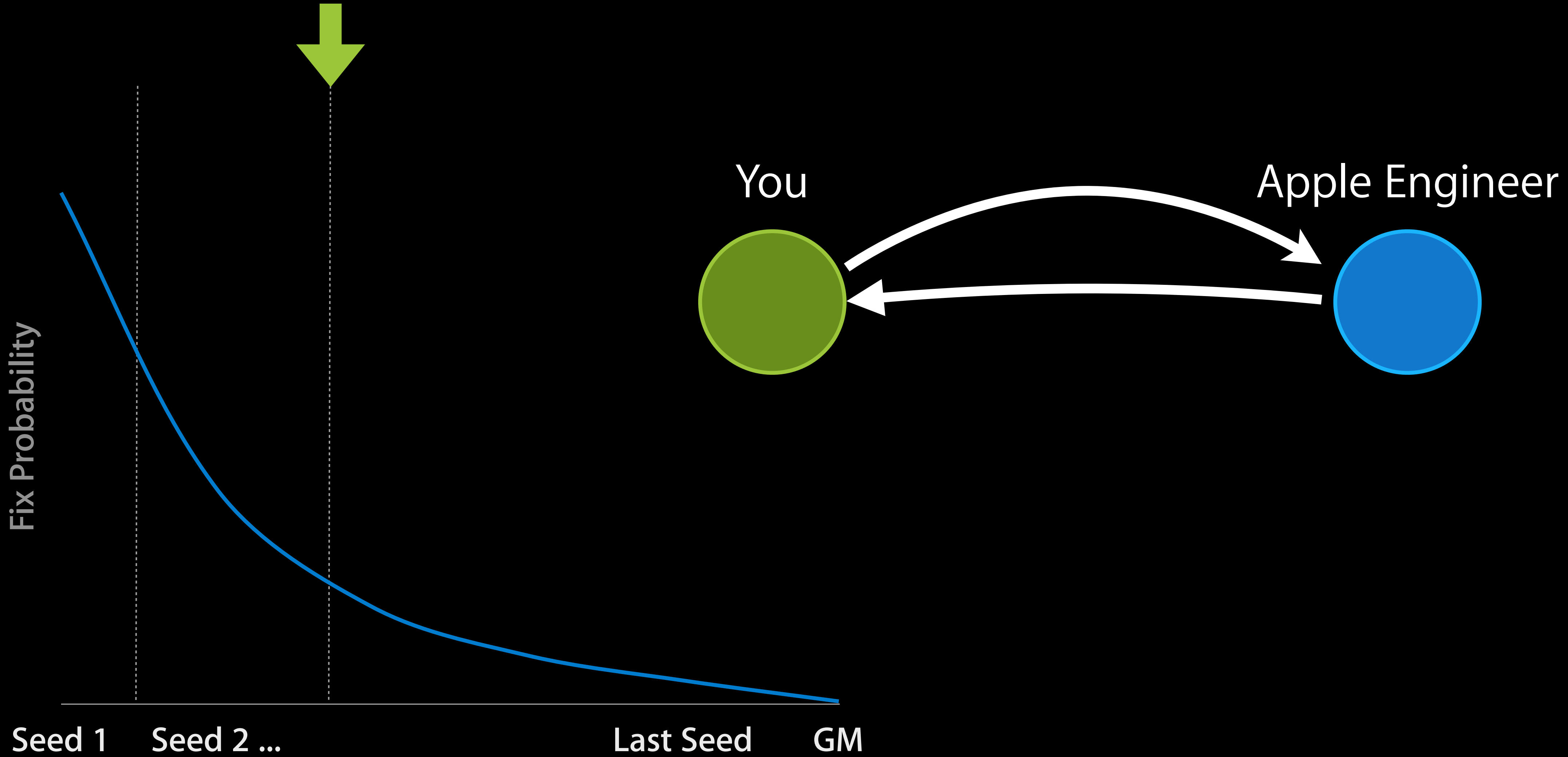




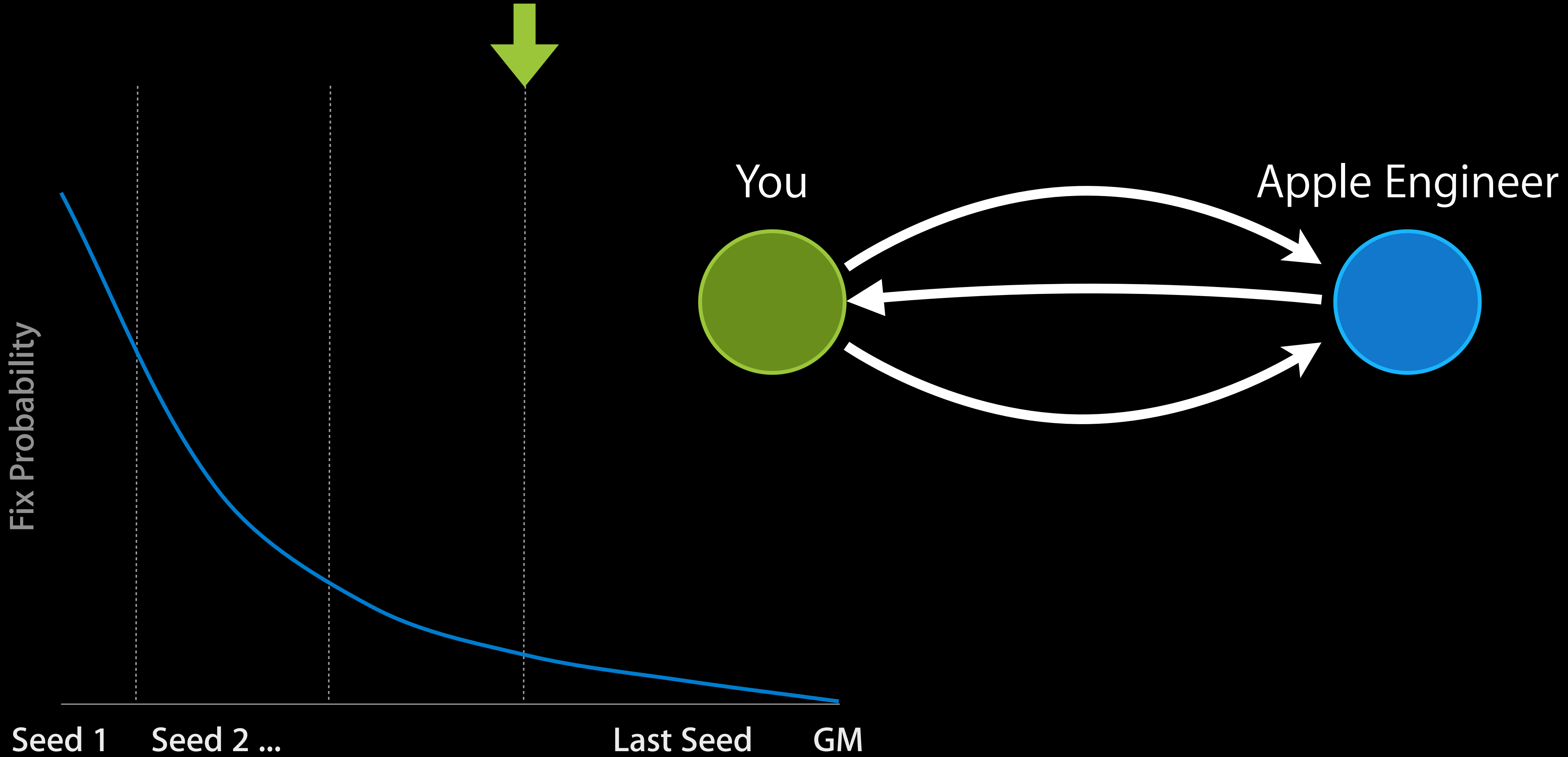
# Insufficient Information Hazards



# Insufficient Information Hazards



# Insufficient Information Hazards



# Write Actionable Reports

- Clear
- Concise
- Informative
- Constructive
- Specific



# Components of a Bug Report

Problem Report Title

Problem Details

Additional Enclosures

Configuration Information

Reproducibility

# Components of a Bug Report

Problem Report Title

Problem Details

Additional Enclosures

Configuration Information

Reproducibility

# 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 paste is broken

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

Pasting an item stored using [UIPasteboard 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 Report Title

Problem Details

Additional Enclosures

Configuration Information

Reproducibility

# 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

Problem Report Title

Problem Details

Additional Enclosures

Configuration Information

Reproducibility

# Sample Project





# Additional Information

- Information available at [developer.apple.com/bugreporter/bugbestpractices](https://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

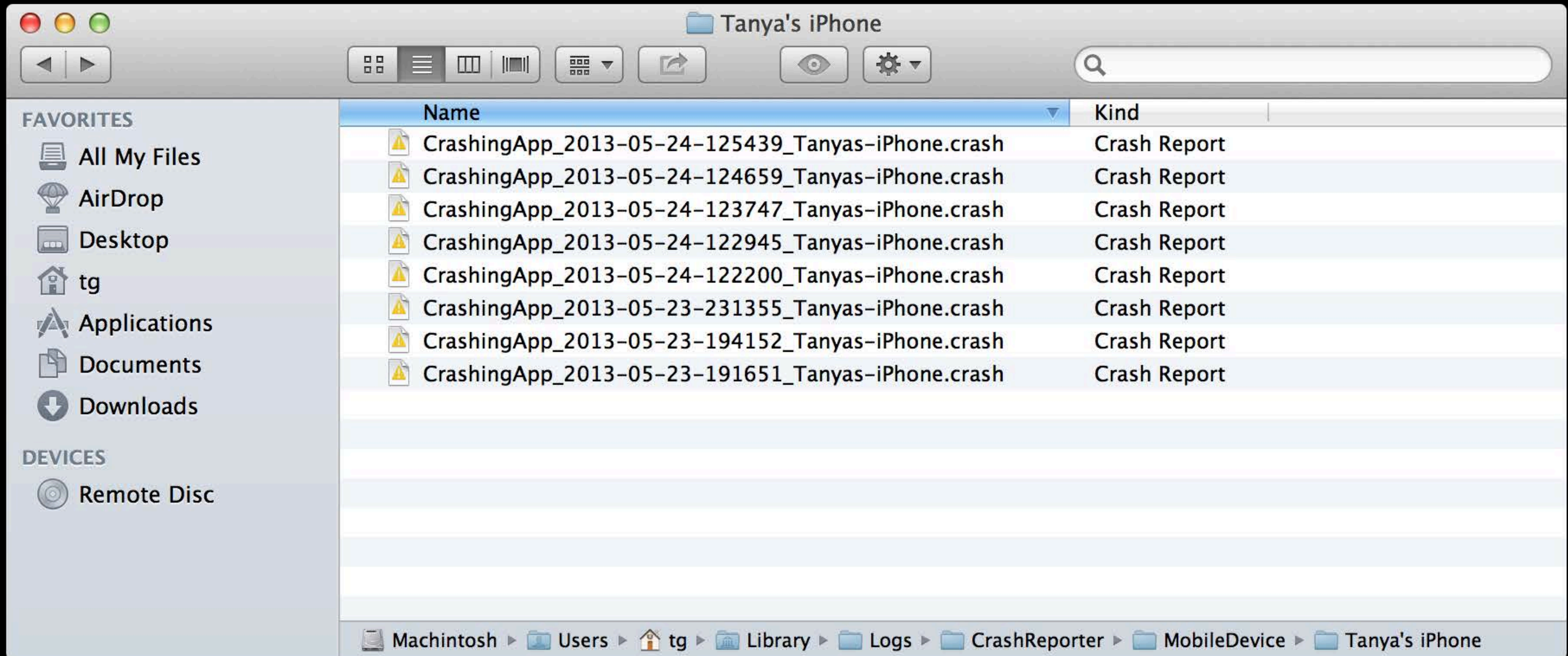
## Step 1: Transfer the logs

- Connect your device
- Sync the device in iTunes



# Crash Logs for iOS

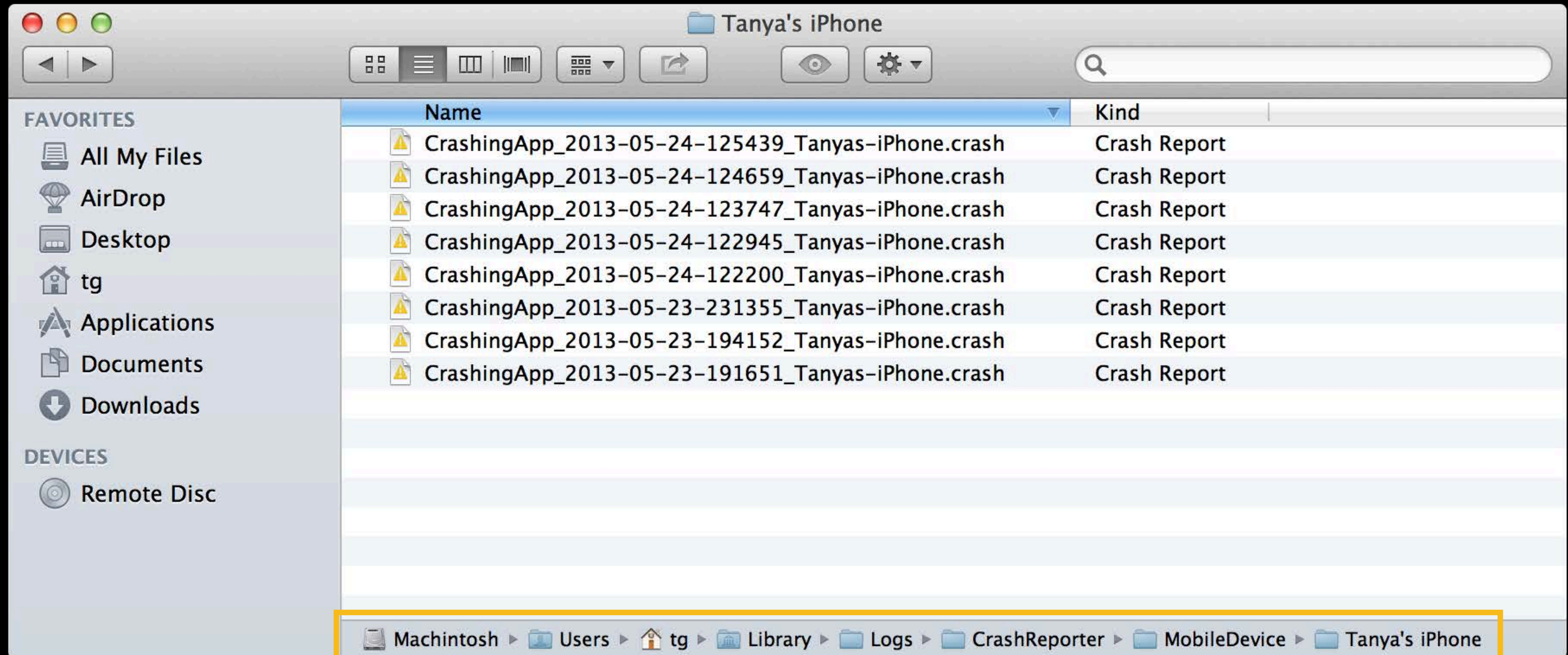
## Step 2: Locate the log file





# Crash Logs for iOS

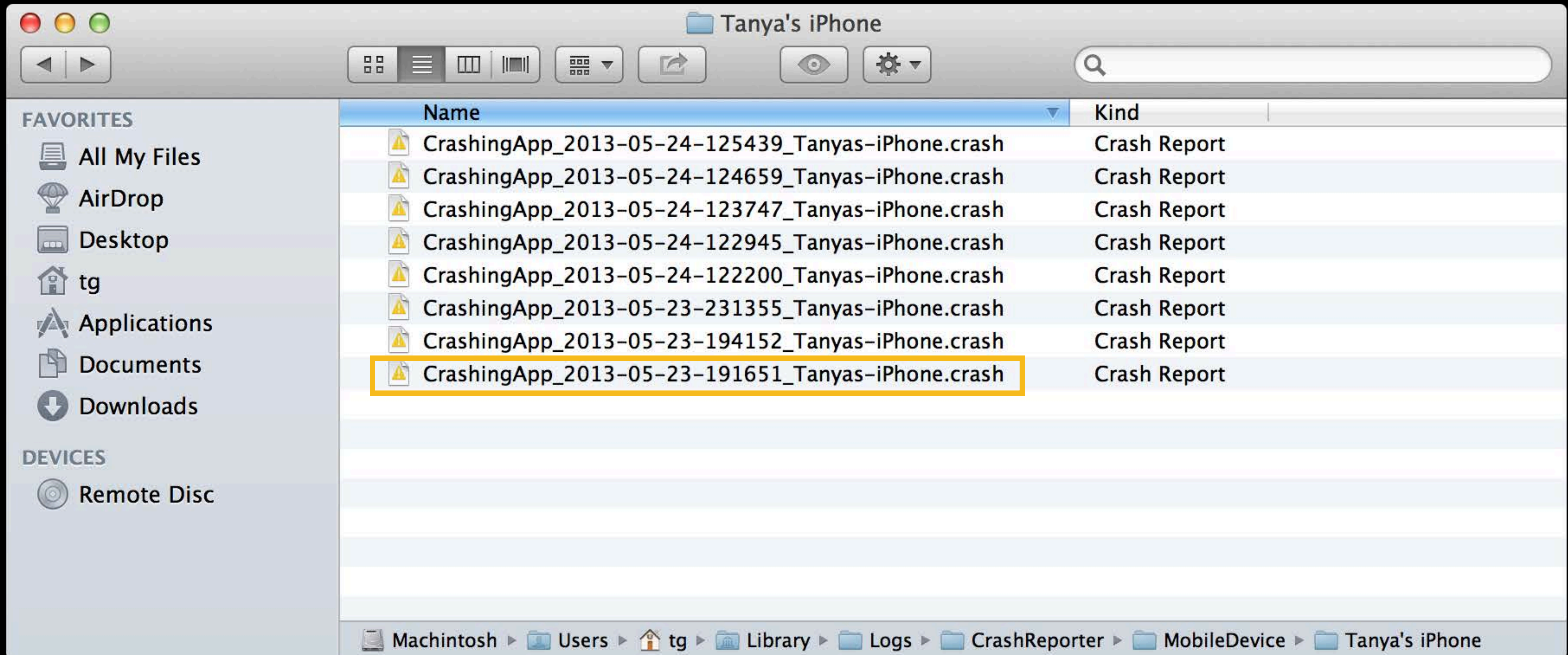
## Step 2: Locate the log file





# Crash Logs for iOS

## Step 2: Locate the log file



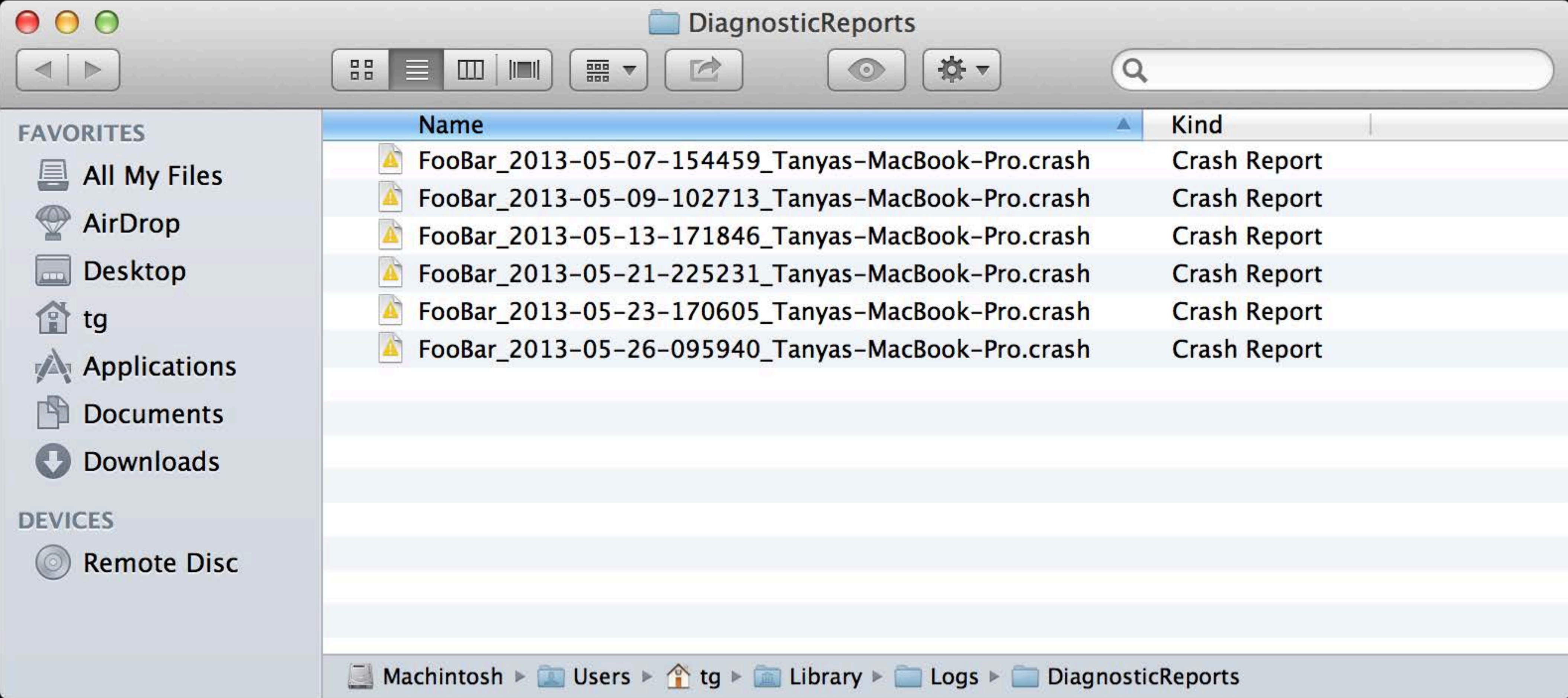
# Crash Logs for iOS

## 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]

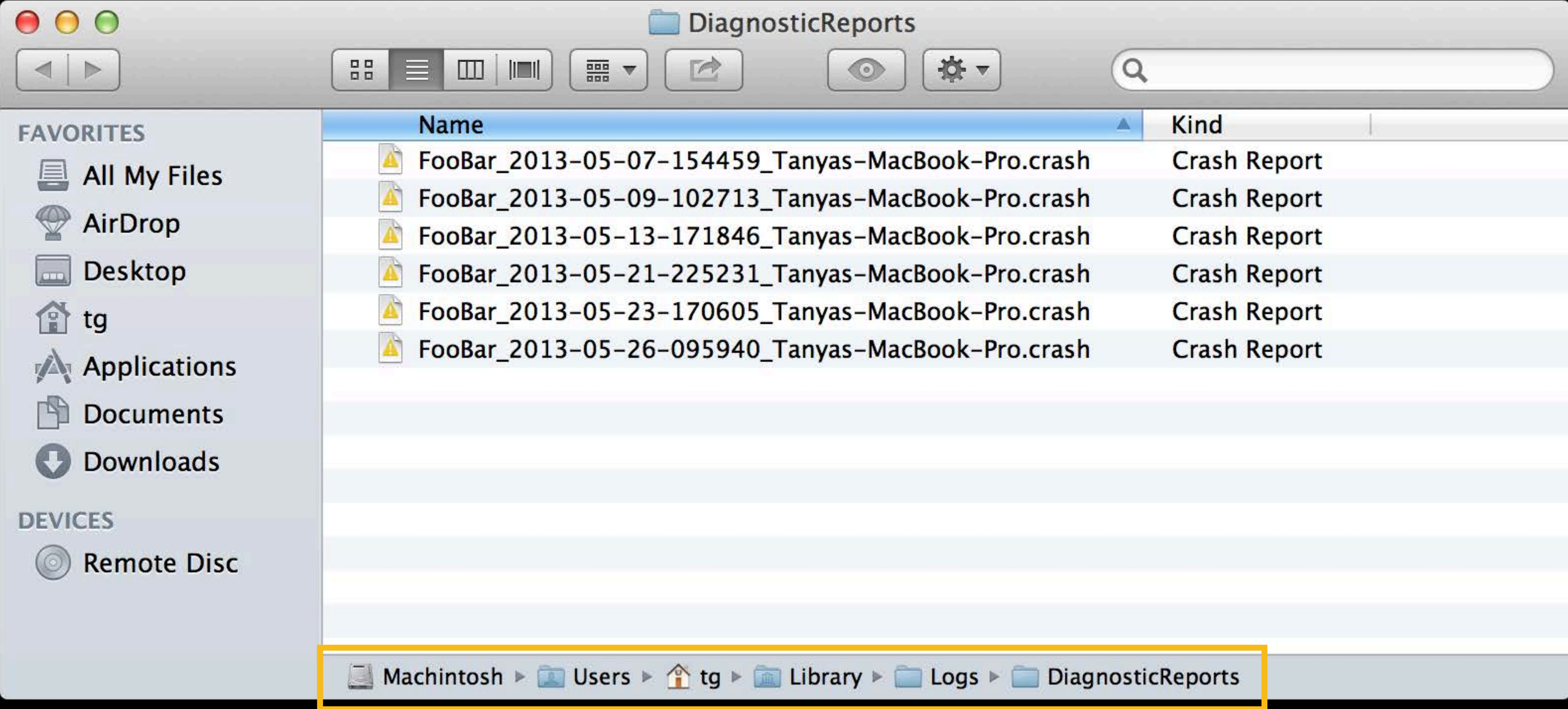


# Crash Logs for OS X





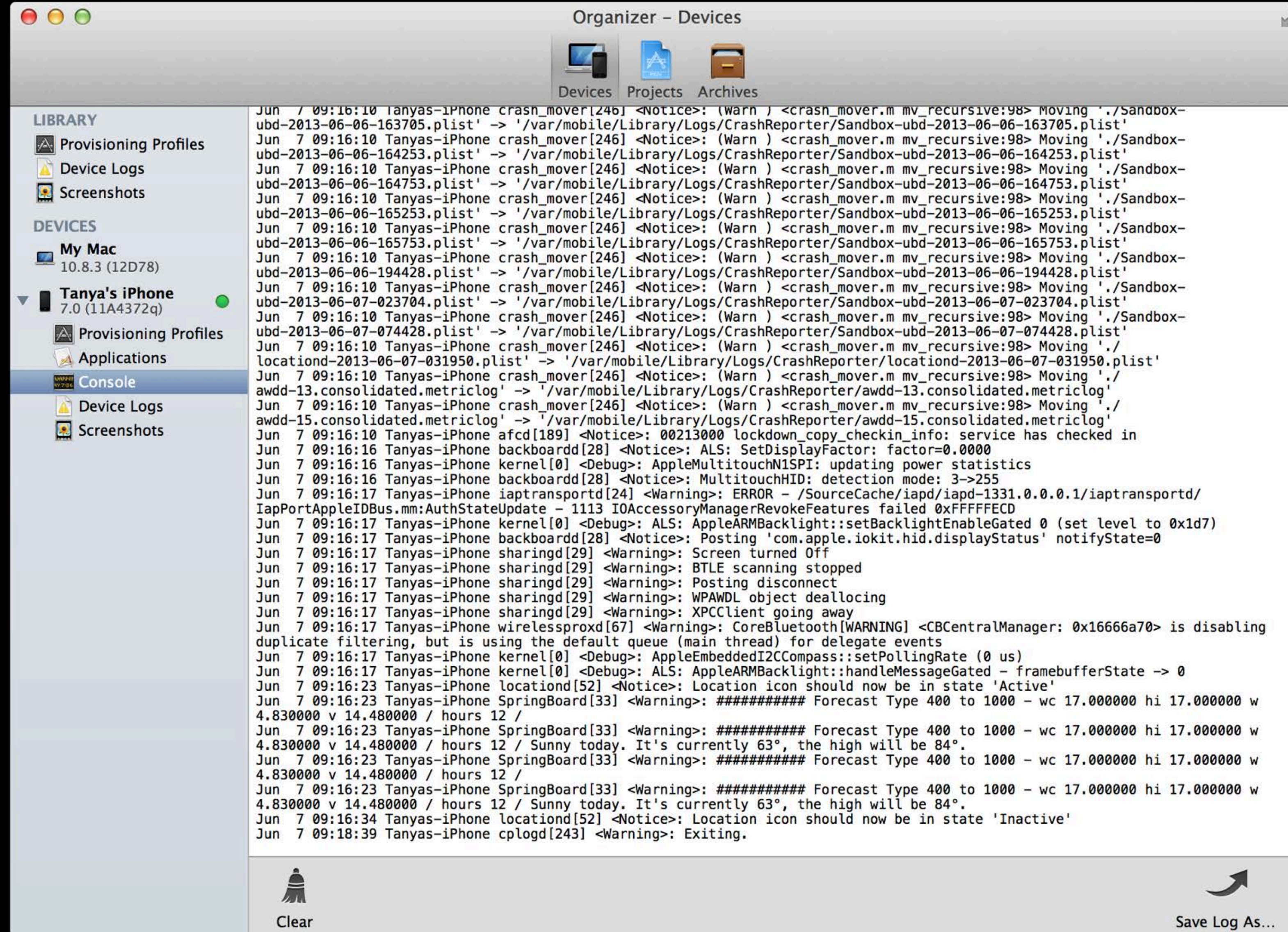
# Crash Logs for OS X





# Console Logs for iOS

## Xcode organizer



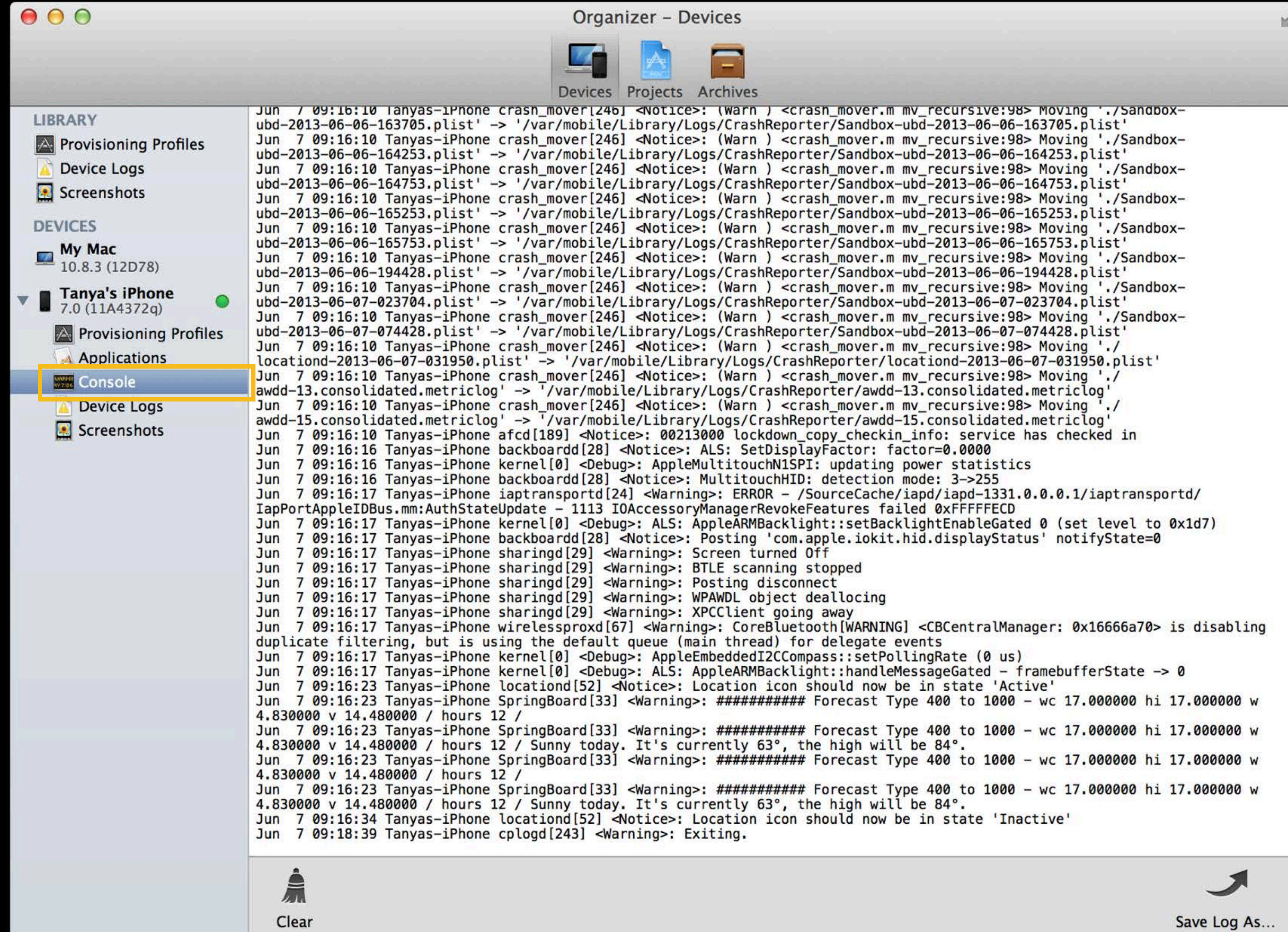
The screenshot shows the Xcode Organizer - Devices window. The left sidebar is divided into 'LIBRARY' and 'DEVICES'. Under 'LIBRARY', there are 'Provisioning Profiles', 'Device Logs', and 'Screenshots'. Under 'DEVICES', there are 'My Mac' (10.8.3 (12D78)) and 'Tanya's iPhone' (7.0 (11A4372q)). The 'Console' view for 'Tanya's iPhone' is selected, displaying a list of system logs. The logs include various messages such as file moves, warnings, and debug information from different system components like 'crash\_mover', 'backboardd', 'kernel', 'sharingd', and 'SpringBoard'. At the bottom of the console, there are 'Clear' and 'Save Log As...' buttons.

```
Jun 7 09:16:10 Tanyas-iPhone crash_mover[246] <Notice>: (warn ) <crash_mover.m mv_recursive:98> Moving './Sandbox-ubd-2013-06-06-163705.plist' -> '/var/mobile/Library/Logs/CrashReporter/Sandbox-ubd-2013-06-06-163705.plist'
Jun 7 09:16:10 Tanyas-iPhone crash_mover[246] <Notice>: (warn ) <crash_mover.m mv_recursive:98> Moving './Sandbox-ubd-2013-06-06-164253.plist' -> '/var/mobile/Library/Logs/CrashReporter/Sandbox-ubd-2013-06-06-164253.plist'
Jun 7 09:16:10 Tanyas-iPhone crash_mover[246] <Notice>: (warn ) <crash_mover.m mv_recursive:98> Moving './Sandbox-ubd-2013-06-06-164753.plist' -> '/var/mobile/Library/Logs/CrashReporter/Sandbox-ubd-2013-06-06-164753.plist'
Jun 7 09:16:10 Tanyas-iPhone crash_mover[246] <Notice>: (warn ) <crash_mover.m mv_recursive:98> Moving './Sandbox-ubd-2013-06-06-165253.plist' -> '/var/mobile/Library/Logs/CrashReporter/Sandbox-ubd-2013-06-06-165253.plist'
Jun 7 09:16:10 Tanyas-iPhone crash_mover[246] <Notice>: (warn ) <crash_mover.m mv_recursive:98> Moving './Sandbox-ubd-2013-06-06-165753.plist' -> '/var/mobile/Library/Logs/CrashReporter/Sandbox-ubd-2013-06-06-165753.plist'
Jun 7 09:16:10 Tanyas-iPhone crash_mover[246] <Notice>: (warn ) <crash_mover.m mv_recursive:98> Moving './Sandbox-ubd-2013-06-06-194428.plist' -> '/var/mobile/Library/Logs/CrashReporter/Sandbox-ubd-2013-06-06-194428.plist'
Jun 7 09:16:10 Tanyas-iPhone crash_mover[246] <Notice>: (warn ) <crash_mover.m mv_recursive:98> Moving './Sandbox-ubd-2013-06-07-023704.plist' -> '/var/mobile/Library/Logs/CrashReporter/Sandbox-ubd-2013-06-07-023704.plist'
Jun 7 09:16:10 Tanyas-iPhone crash_mover[246] <Notice>: (warn ) <crash_mover.m mv_recursive:98> Moving './Sandbox-ubd-2013-06-07-074428.plist' -> '/var/mobile/Library/Logs/CrashReporter/Sandbox-ubd-2013-06-07-074428.plist'
Jun 7 09:16:10 Tanyas-iPhone crash_mover[246] <Notice>: (warn ) <crash_mover.m mv_recursive:98> Moving './locationd-2013-06-07-031950.plist' -> '/var/mobile/Library/Logs/CrashReporter/locationd-2013-06-07-031950.plist'
Jun 7 09:16:10 Tanyas-iPhone crash_mover[246] <Notice>: (warn ) <crash_mover.m mv_recursive:98> Moving './awdd-13.consolidated.metriclog' -> '/var/mobile/Library/Logs/CrashReporter/awdd-13.consolidated.metriclog'
Jun 7 09:16:10 Tanyas-iPhone crash_mover[246] <Notice>: (warn ) <crash_mover.m mv_recursive:98> Moving './awdd-15.consolidated.metriclog' -> '/var/mobile/Library/Logs/CrashReporter/awdd-15.consolidated.metriclog'
Jun 7 09:16:10 Tanyas-iPhone afcd[189] <Notice>: 00213000 lockdown_copy_checkin_info: service has checked in
Jun 7 09:16:16 Tanyas-iPhone backboardd[28] <Notice>: ALS: SetDisplayFactor: factor=0.0000
Jun 7 09:16:16 Tanyas-iPhone kernel[0] <Debug>: AppleMultitouchN1SPI: updating power statistics
Jun 7 09:16:16 Tanyas-iPhone backboardd[28] <Notice>: MultitouchHID: detection mode: 3->255
Jun 7 09:16:17 Tanyas-iPhone iaptransportd[24] <Warning>: ERROR - /SourceCache/iapd/iapd-1331.0.0.0.1/iaptransportd/IapPortAppleIDBus.mm:AuthStateUpdate - 1113 IOAccessoryManagerRevokeFeatures failed 0xFFFFFECF
Jun 7 09:16:17 Tanyas-iPhone kernel[0] <Debug>: ALS: AppleARMBacklight::setBacklightEnableGated 0 (set level to 0x1d7)
Jun 7 09:16:17 Tanyas-iPhone backboardd[28] <Notice>: Posting 'com.apple.iokit.hid.displayStatus' notifyState=0
Jun 7 09:16:17 Tanyas-iPhone sharingd[29] <Warning>: Screen turned Off
Jun 7 09:16:17 Tanyas-iPhone sharingd[29] <Warning>: BTLE scanning stopped
Jun 7 09:16:17 Tanyas-iPhone sharingd[29] <Warning>: Posting disconnect
Jun 7 09:16:17 Tanyas-iPhone sharingd[29] <Warning>: WPAWDL object deallocing
Jun 7 09:16:17 Tanyas-iPhone sharingd[29] <Warning>: XPCClient going away
Jun 7 09:16:17 Tanyas-iPhone wirelessproxd[67] <Warning>: CoreBluetooth[WARNING] <CBCentralManager: 0x16666a70> is disabling duplicate filtering, but is using the default queue (main thread) for delegate events
Jun 7 09:16:17 Tanyas-iPhone kernel[0] <Debug>: AppleEmbeddedI2CCompass::setPollingRate (0 us)
Jun 7 09:16:17 Tanyas-iPhone kernel[0] <Debug>: ALS: AppleARMBacklight::handleMessageGated - framebufferState -> 0
Jun 7 09:16:23 Tanyas-iPhone locationd[52] <Notice>: Location icon should now be in state 'Active'
Jun 7 09:16:23 Tanyas-iPhone SpringBoard[33] <Warning>: ##### Forecast Type 400 to 1000 - wc 17.000000 hi 17.000000 w 4.830000 v 14.480000 / hours 12 /
Jun 7 09:16:23 Tanyas-iPhone SpringBoard[33] <Warning>: ##### Forecast Type 400 to 1000 - wc 17.000000 hi 17.000000 w 4.830000 v 14.480000 / hours 12 / Sunny today. It's currently 63°, the high will be 84°.
Jun 7 09:16:23 Tanyas-iPhone SpringBoard[33] <Warning>: ##### Forecast Type 400 to 1000 - wc 17.000000 hi 17.000000 w 4.830000 v 14.480000 / hours 12 /
Jun 7 09:16:23 Tanyas-iPhone SpringBoard[33] <Warning>: ##### Forecast Type 400 to 1000 - wc 17.000000 hi 17.000000 w 4.830000 v 14.480000 / hours 12 / Sunny today. It's currently 63°, the high will be 84°.
Jun 7 09:16:34 Tanyas-iPhone locationd[52] <Notice>: Location icon should now be in state 'Inactive'
Jun 7 09:18:39 Tanyas-iPhone cplogd[243] <Warning>: Exiting.
```



# Console Logs for iOS

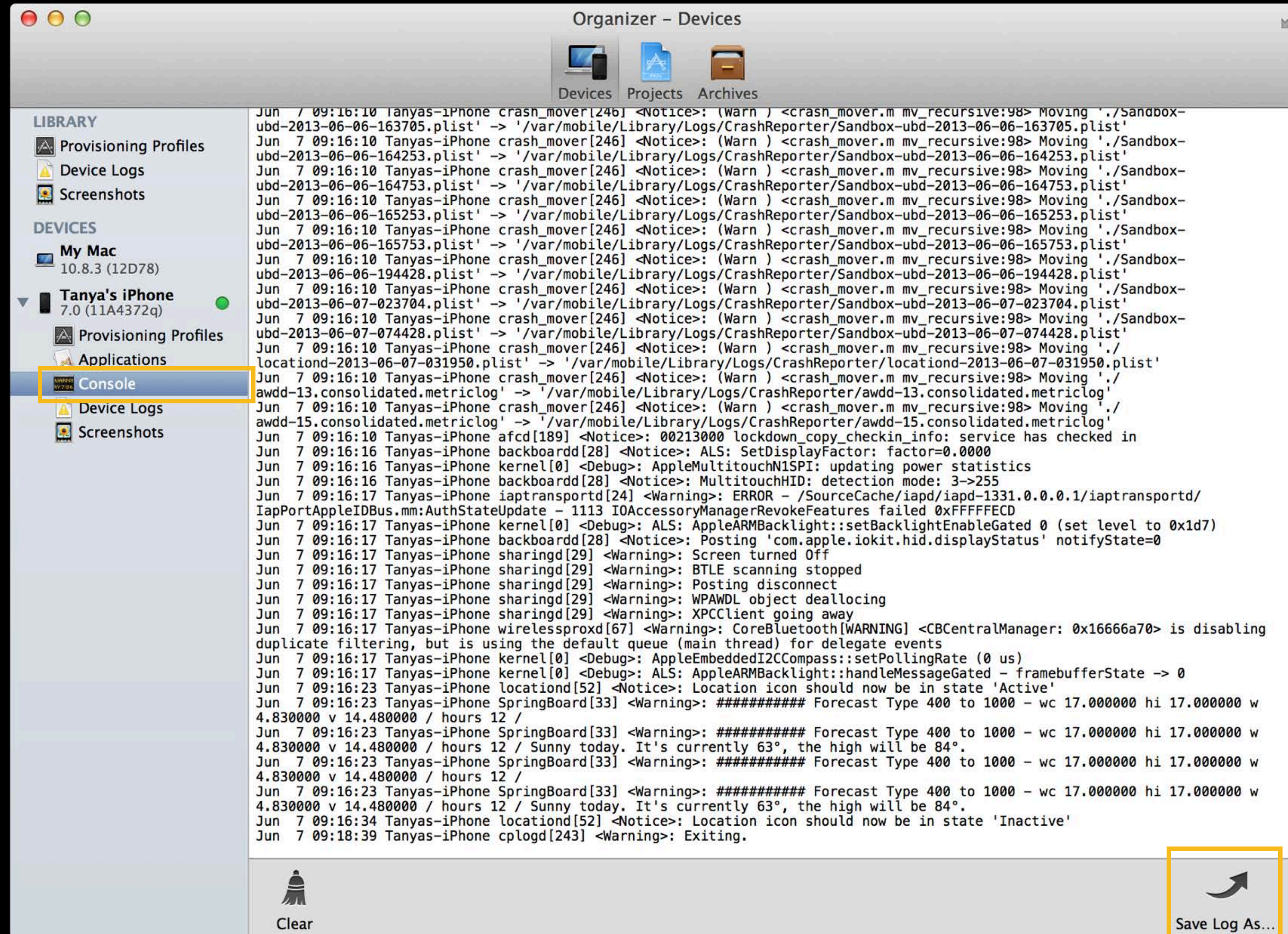
## Xcode organizer





# Console Logs for iOS

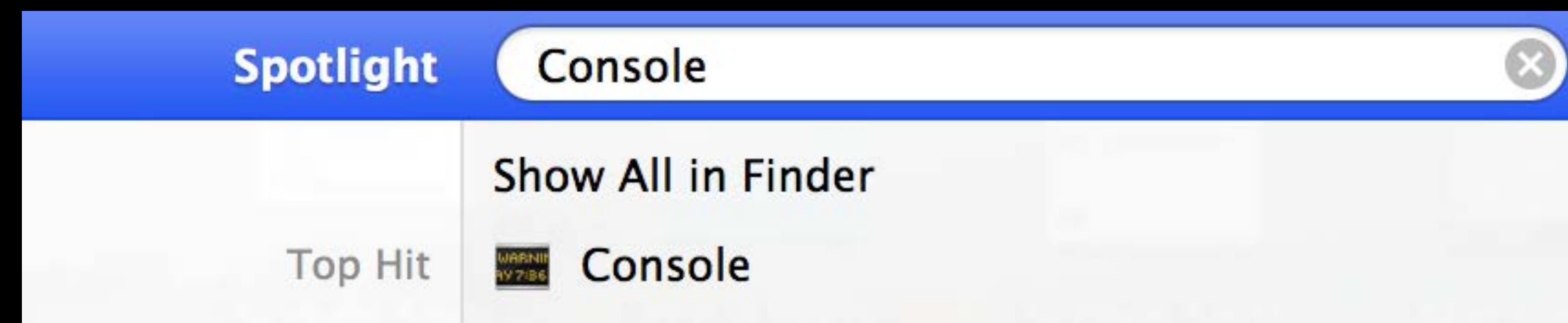
## Xcode organizer





# Console Logs for OS X

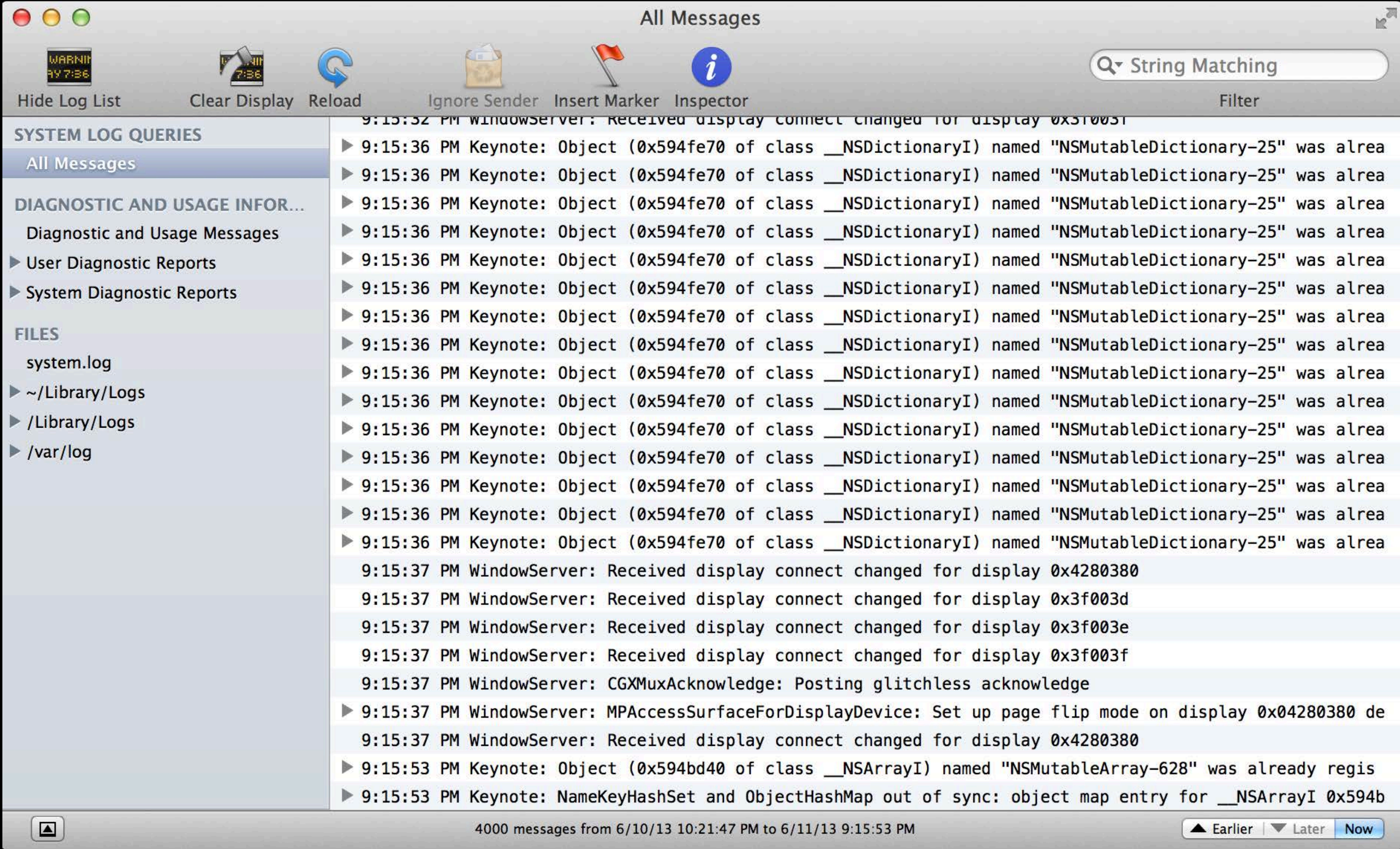
## Step 1: Launch Console.app





# Console Logs for OS X

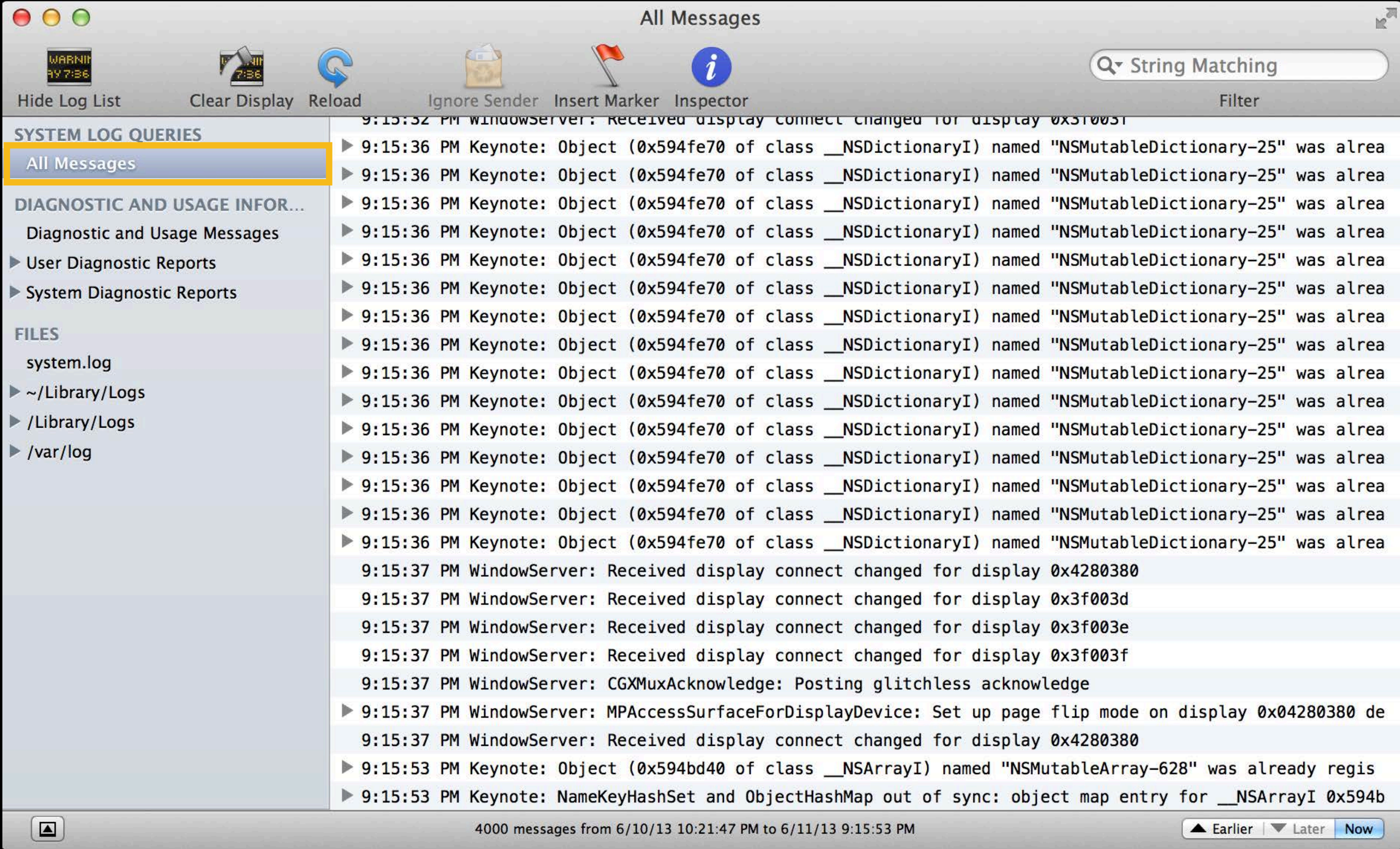
## Step 2: Select All Messages





# Console Logs for OS X

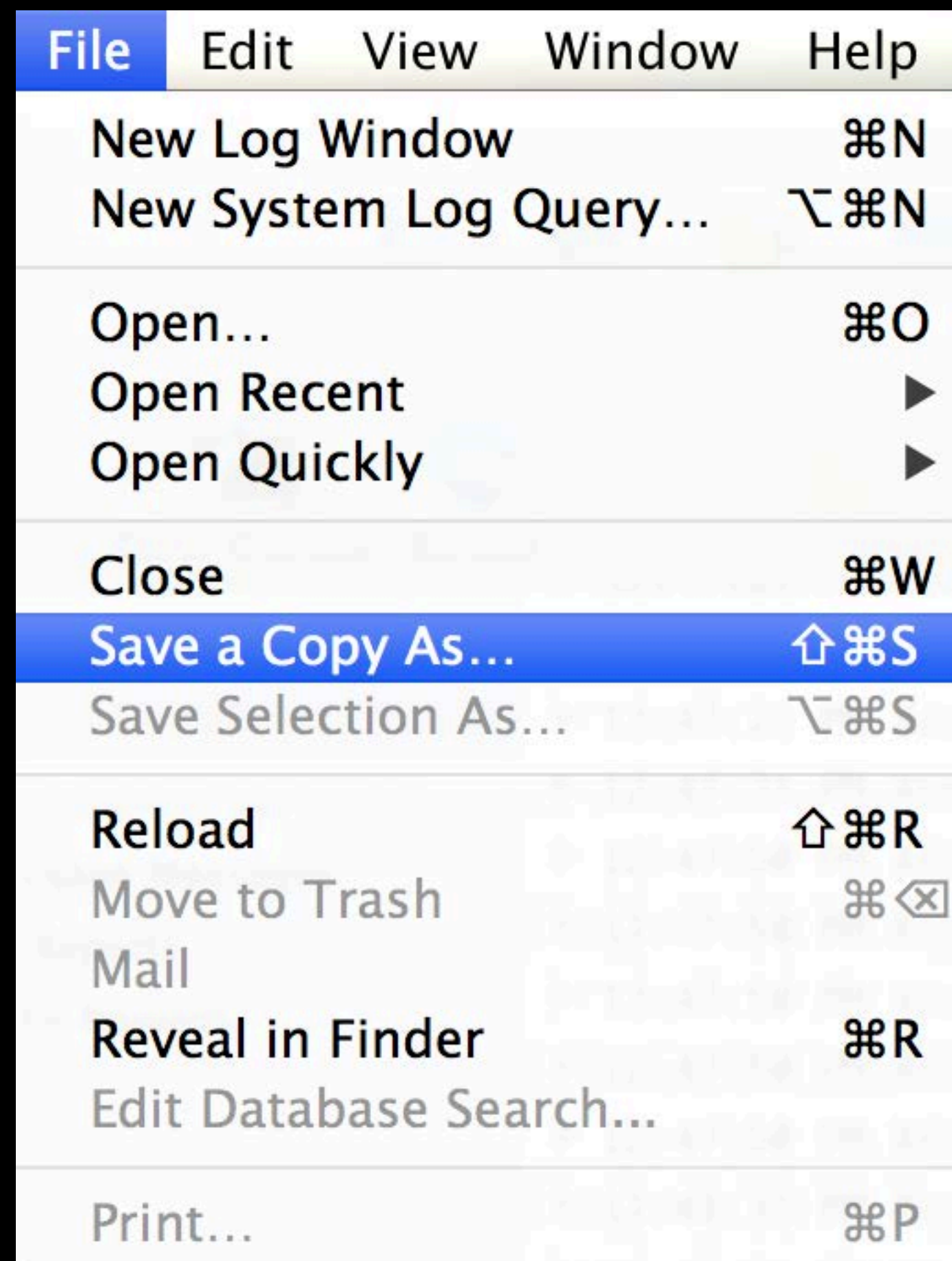
## Step 2: Select All Messages





# 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





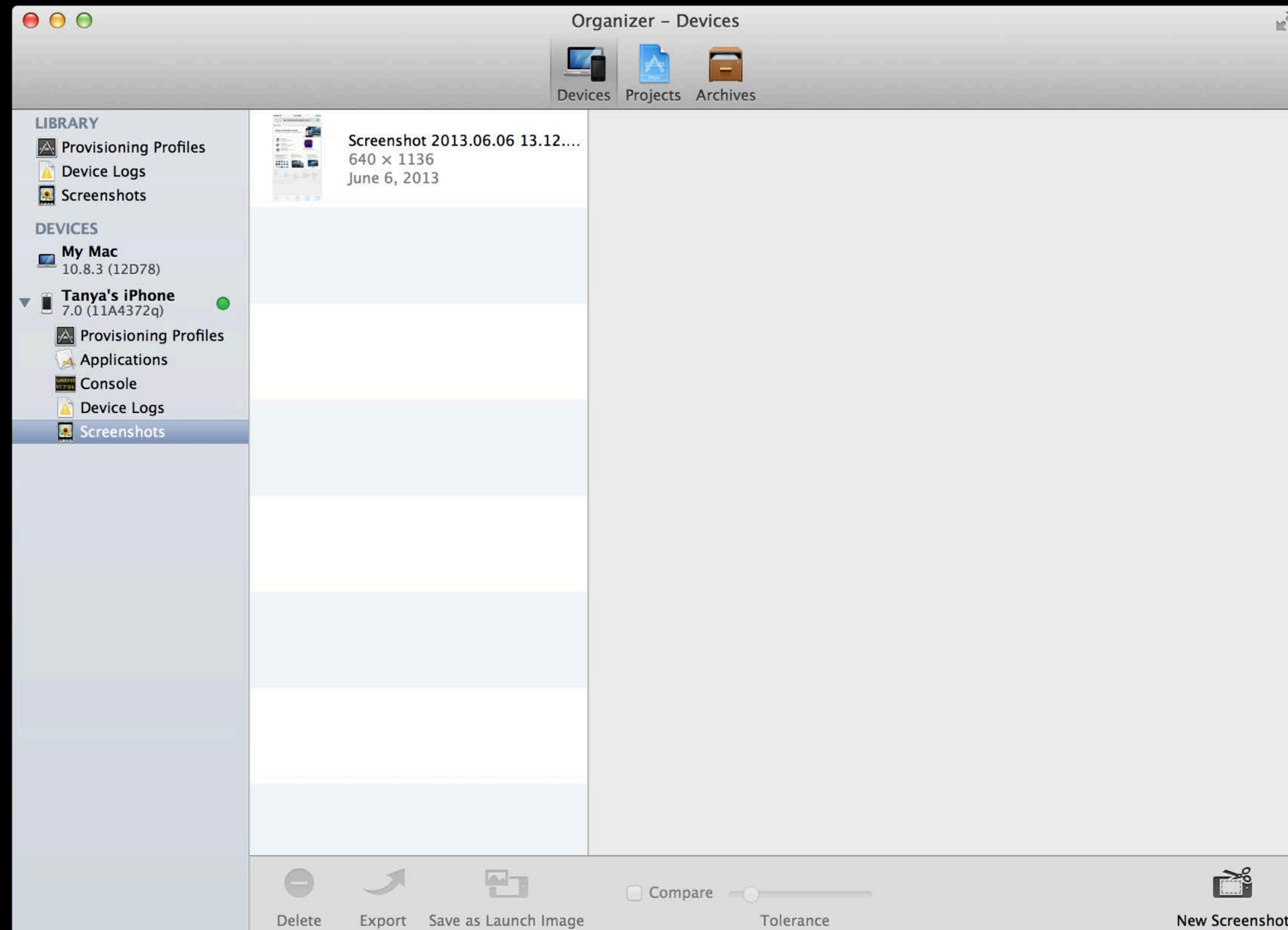
# Screenshots on iOS

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



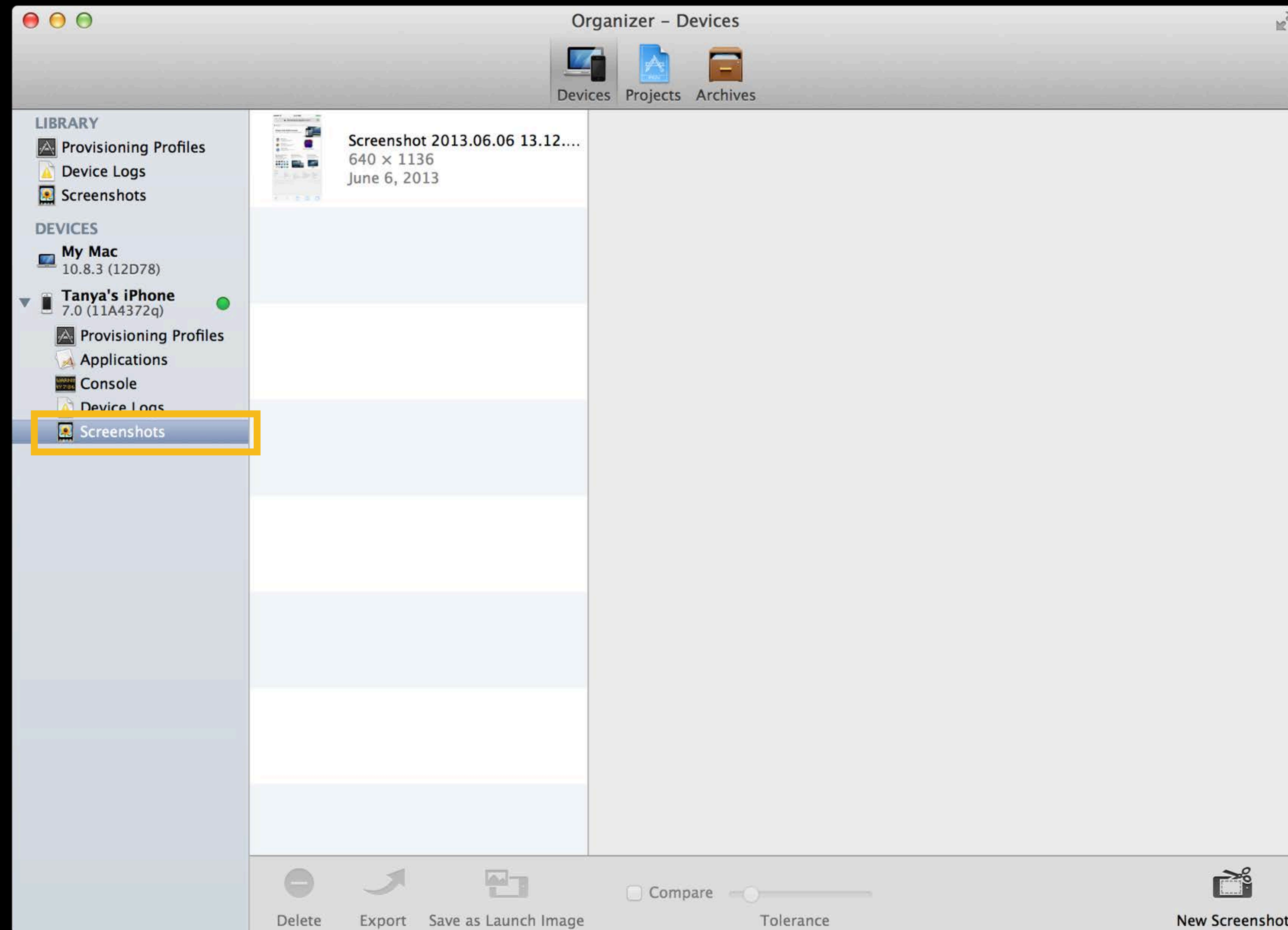
# Screenshots on iOS

## Xcode Organizer



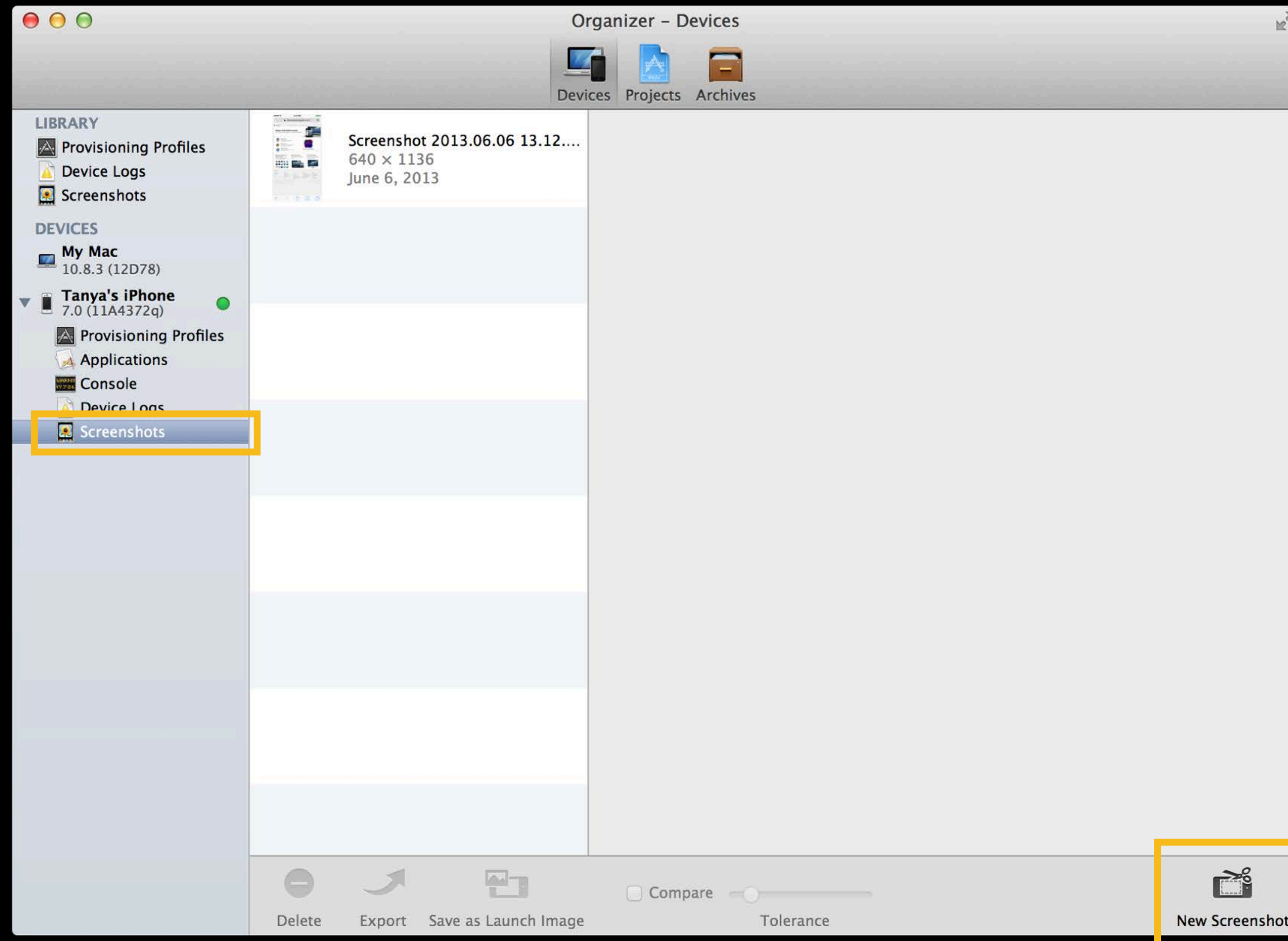
# Screenshots on iOS

## 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

```
stacks-2013-06-06-134036.log
String Matching
Filter

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: 0x383fca7c 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: 0x80491cdd
  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

Size: 273 KB
Earlier | Later | Now
```

# Stackshot Logs for iOS

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



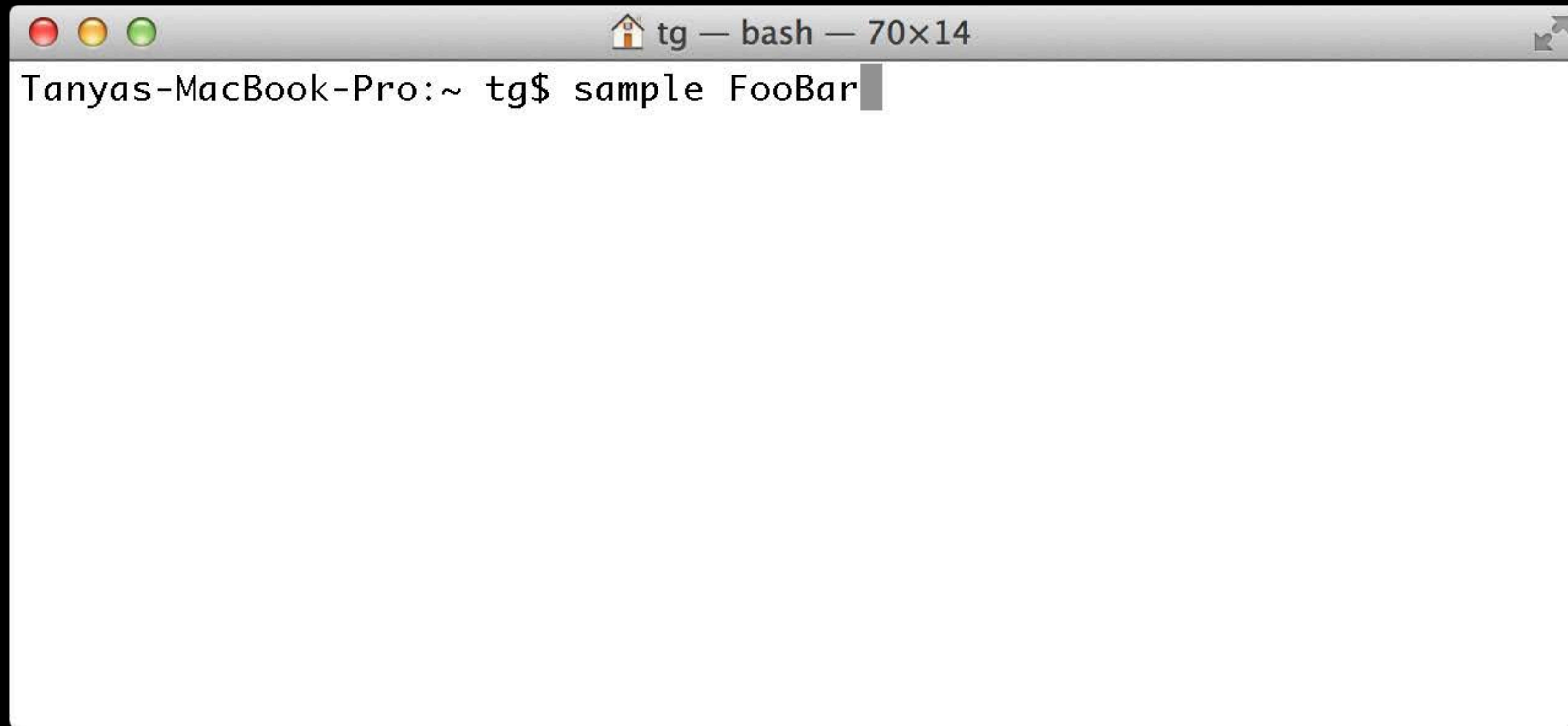


# Stackshot Logs for iOS

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

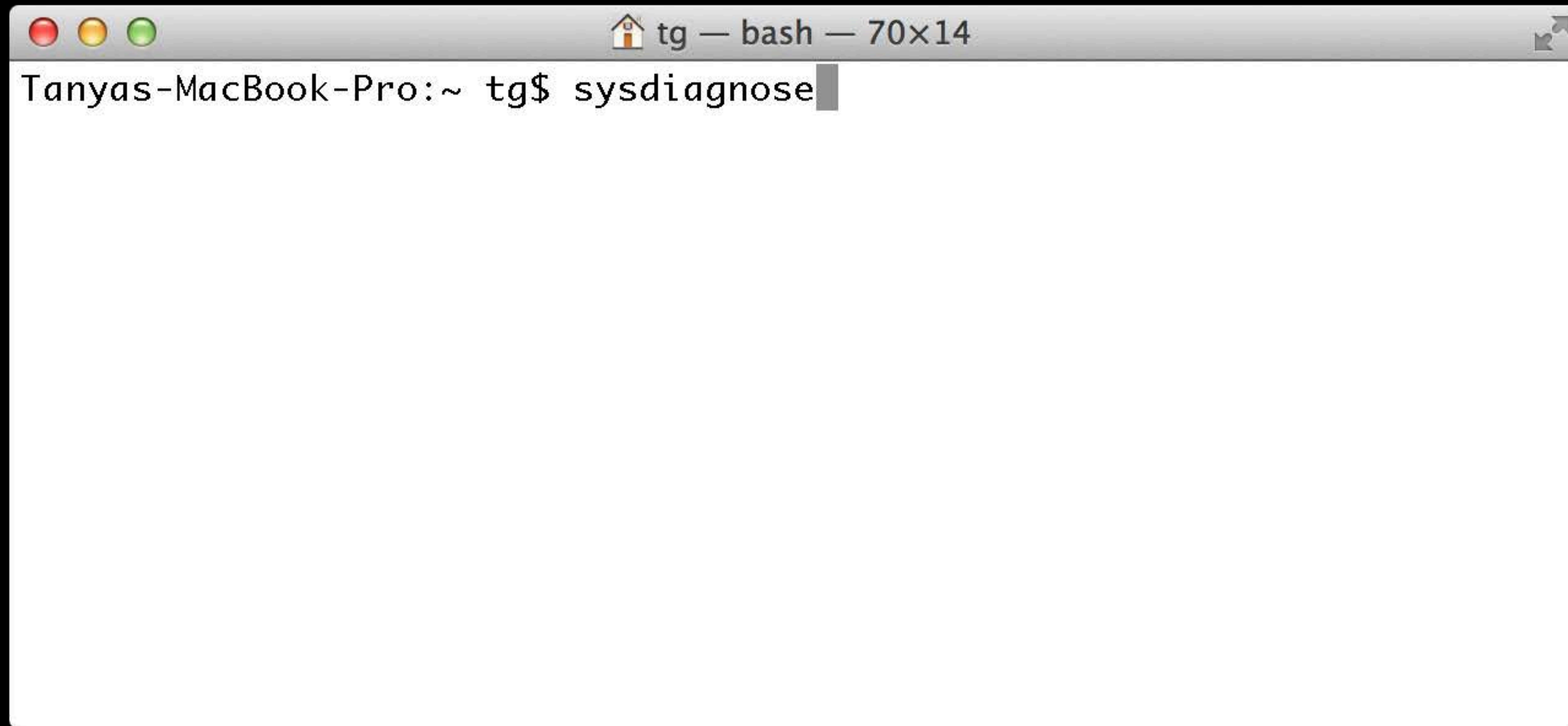


# Sample Logs for OS X

A screenshot of a macOS terminal window. The window title bar shows a home icon, the text 'tg — bash — 70x14', and window control buttons. The terminal content shows the prompt 'Tanyas-MacBook-Pro:~ tg\$' followed by the command 'sample FooBar' and a cursor.

```
Tanyas-MacBook-Pro:~ tg$ sample FooBar
```

# sysdiagnose for OS X



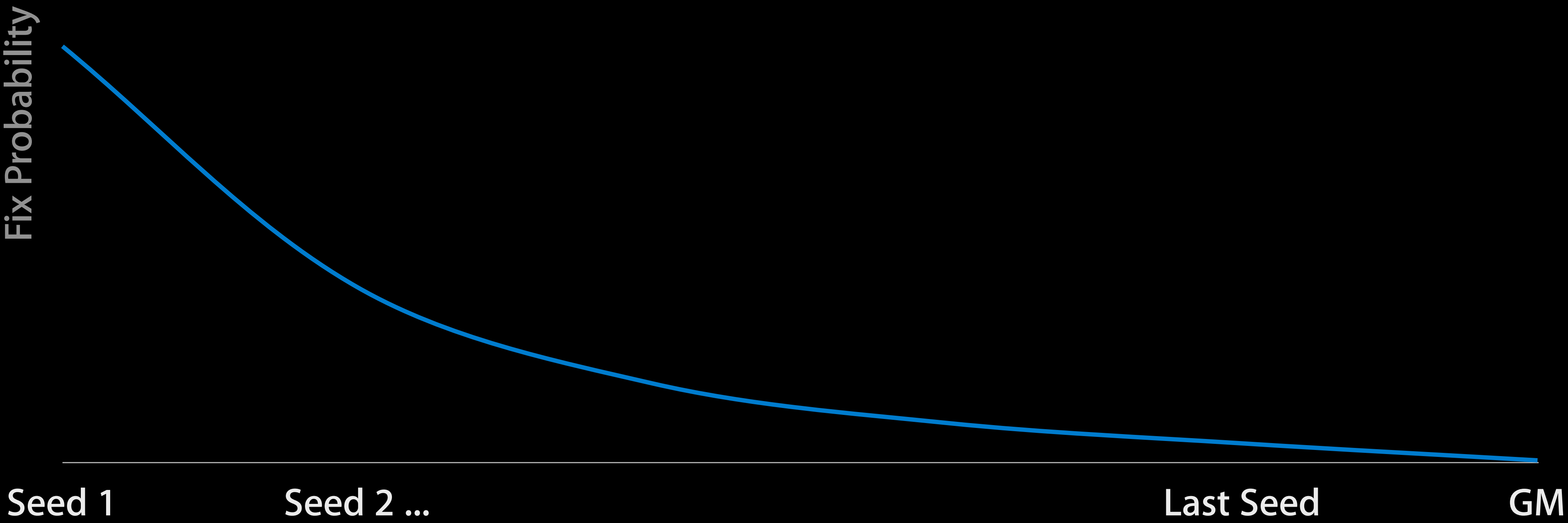
```
Tanyas-MacBook-Pro:~ tg$ sysdiagnose
```

The image shows a terminal window on a Mac. The title bar at the top reads "tg — bash — 70x14". The main content of the window shows the command prompt "Tanyas-MacBook-Pro:~ tg\$" followed by the command "sysdiagnose" being typed. A cursor is visible at the end of the command.

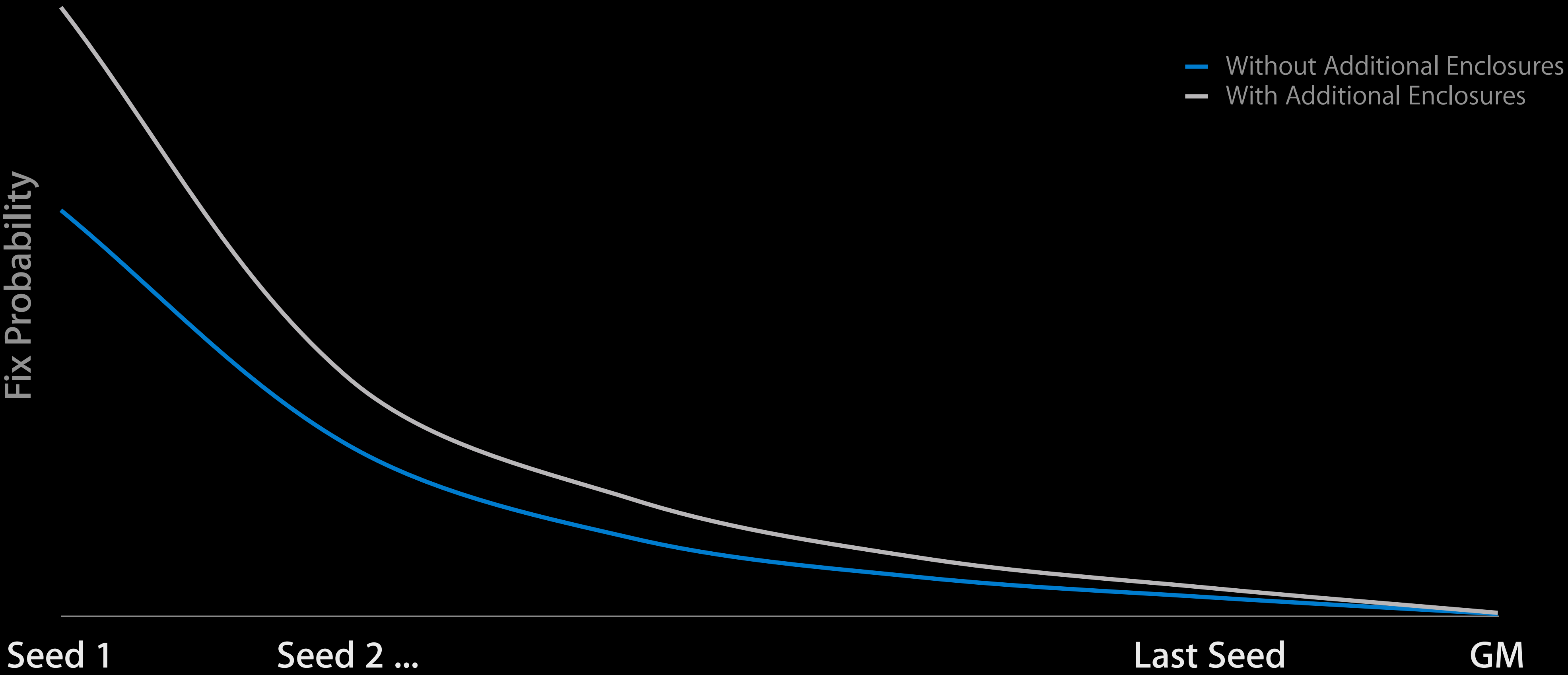
# Benefit of Adding More Information



# Benefit of Adding More Information



# Benefit of Adding More Information



# Components of a Bug Report

Problem Report Title

Problem Details

Additional Enclosures

Configuration Information

Reproducibility

# 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](https://developer.apple.com/bug-reporting/mac/system)

# Components of a Bug Report

Problem Report Title

Problem Details

Additional Enclosures

Configuration Information

Reproducibility

# 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

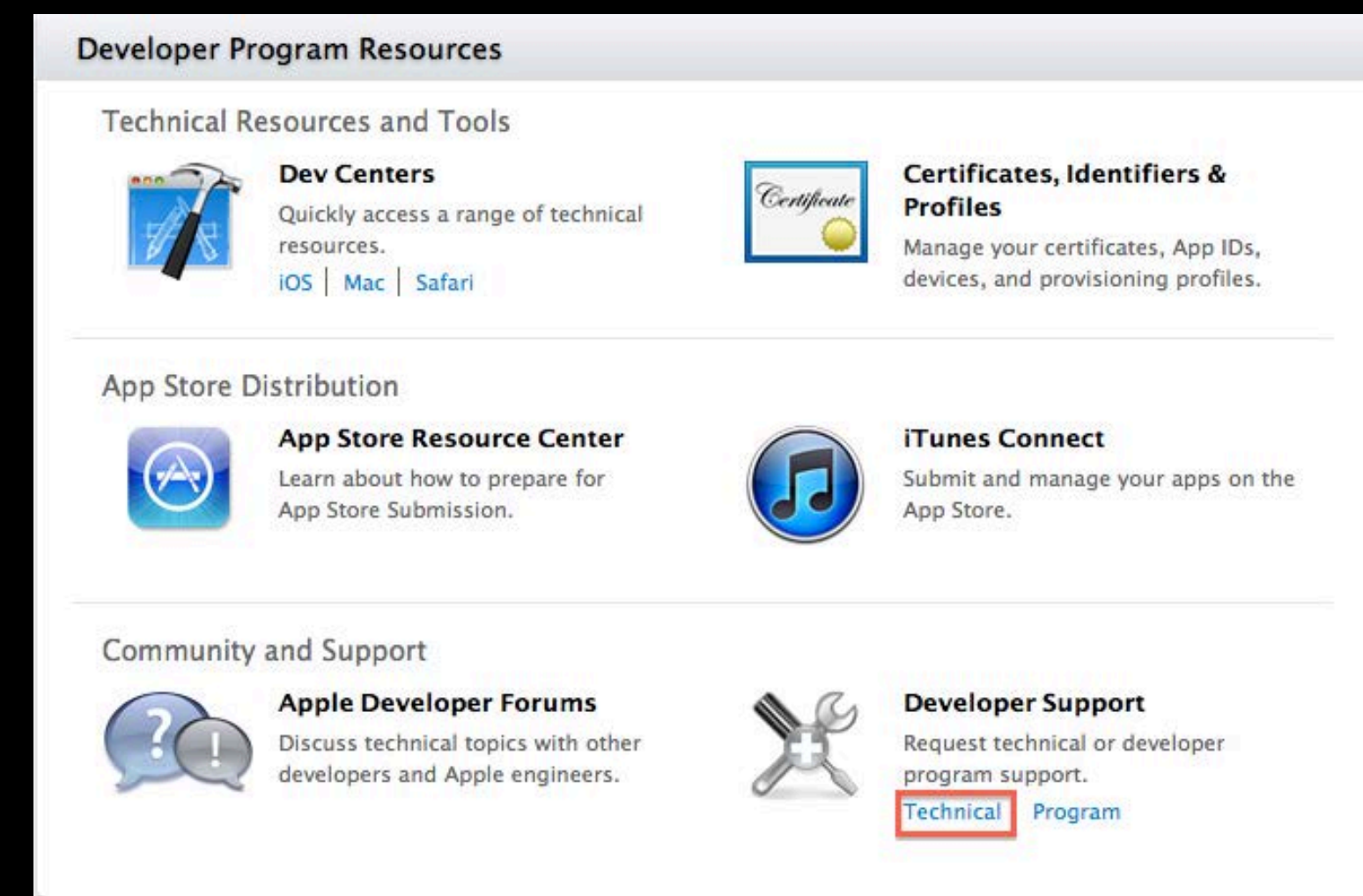
# When to Contact DTS

- You've read through 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](https://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

<b>TN2239</b>	iOS Debugging Magic
<b>TN2151</b>	Understanding and Analyzing iOS Application Crash Reports
<b>QA1747</b>	Debugging Deployed iOS Apps
<b>TN2285</b>	Testing iOS App Updates
<b>QA1764</b>	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](https://developer.apple.com/appstore)
- Learn about
  - Approval process
  - Guidelines
- Marketing assets
  - Downloads and guidelines

The screenshot shows the Apple Developer App Store Resource Center. The page is titled "App Store Resource Center" and includes a navigation bar with links for Technologies, Resources, Programs, Support, and Member Center. The main content area is divided into several sections:

- App Store**: Includes links for Prepare for App Submission, Approval Process, Review Guidelines, Managing Your Apps, and Marketing Resources and Downloads.
- Mac App Store**: Includes links for Prepare for App Submission, Approval Process, Review Guidelines, Managing Your Mac Apps, and Marketing Resources and Downloads.
- Essential Guidelines**: Includes links for iTunes Connect Developer Guide, App Review Guidelines, App Store Marketing Guidelines, and Mac App Store Marketing Guidelines.
- Apple Product Images and Badges**: Includes links for iOS Devices and App Store Badges, and Mac Computers and Mac App Store Badges.
- Contact Us**: Includes links for App Review Questions and Status, Submit a Rejection Appeal, and Request an Expedited App Review.

At the bottom of the page, there are four additional sections:

- Develop Custom B2B Apps**: Sell custom B2B apps directly to your business customers who have a Volume Purchase Program account.
- iTunes Connect Mobile**: Get app submission updates and view your sales and trend data on iPhone, iPad, or iPod touch.
- Promote Your Apps**: Use the new App Store badges and Apple product images to promote your apps on the App Store.
- iOS Developer Opportunities**: Create new or additional revenue from your iOS app by leveraging a range of easy to implement opportunities.

# Contact Form

- Available at:
  - [developer.apple.com/appstore/contact](https://developer.apple.com/appstore/contact)

The screenshot displays the Apple Developer website's 'Contact the App Review Team' page. At the top, the navigation bar includes 'Developer', 'Technologies', 'Resources', 'Programs', 'Support', and 'Member Center'. The main heading is 'App Store Resource Center' with a user profile 'Nahir Khan'. The central section is titled 'Contact the App Review Team'. Under the 'Topic' heading, there is a text input field with the placeholder 'Start by choosing a topic.' and a dropdown menu labeled 'I would like to:'. The dropdown menu is open, showing a list of topics: 'get the status of my app' (highlighted), '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'. Below this, the 'More Ways to Get Help' section provides instructions on how to ask questions and includes icons for technical assistance, a question mark, and a user profile. The footer shows a breadcrumb trail: 'Developer > App Store Resource Center > Contact the App Review Team'.



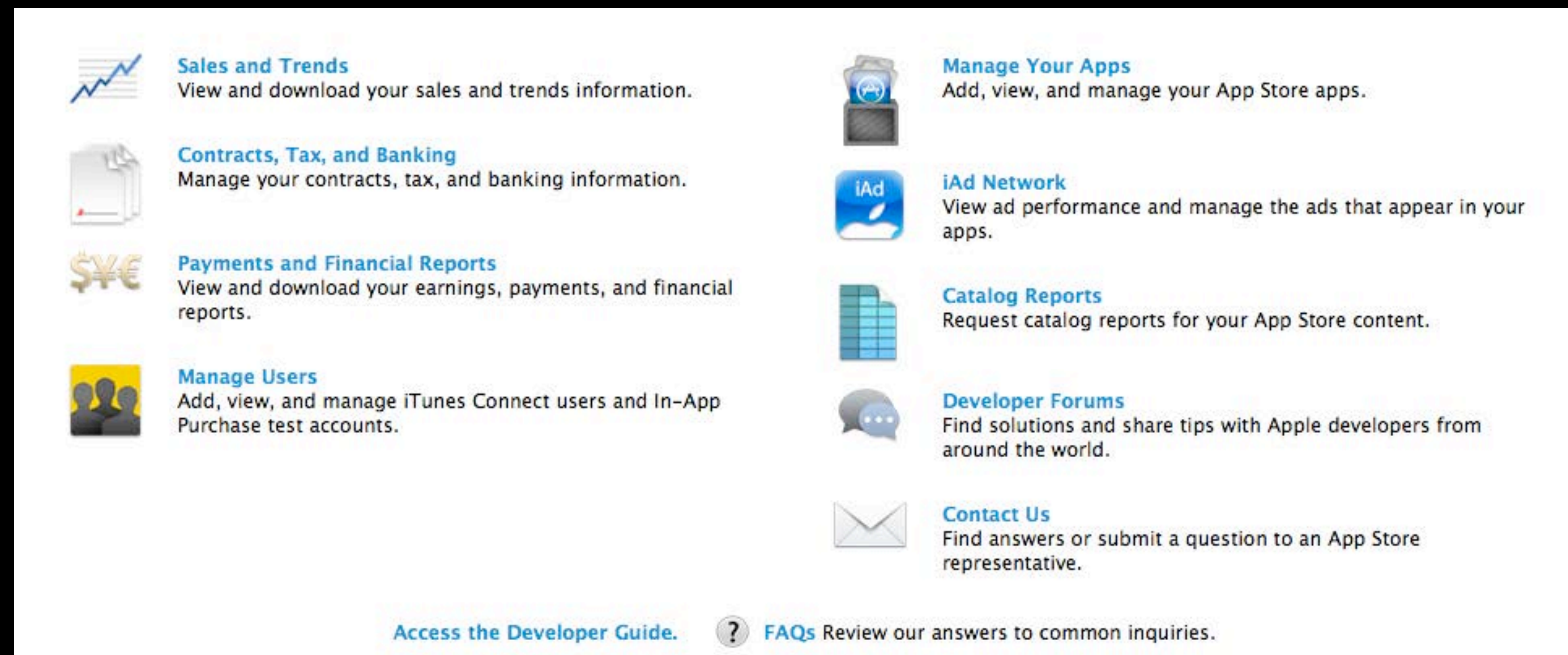
# Contact Form

- 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



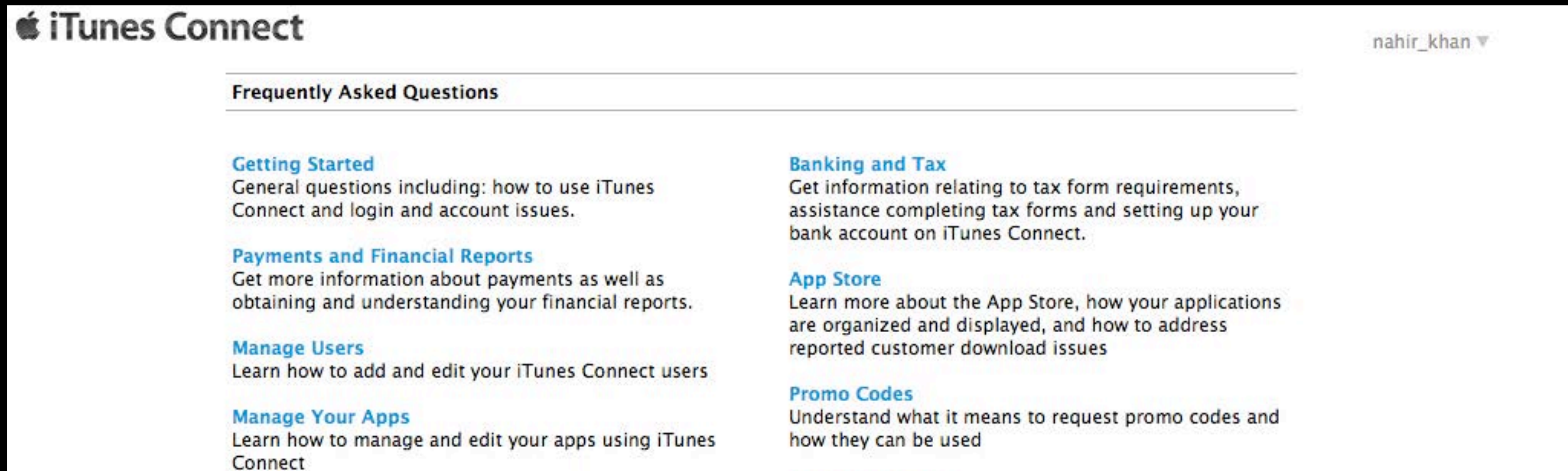
The screenshot displays the iTunes Connect dashboard with the following sections:

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

At the bottom, there are two links: [Access the Developer Guide.](#) and [FAQs](#) Review our answers to common inquiries.

# iTunes Connect FAQ

- Great starting out point



The screenshot shows the iTunes Connect website's FAQ section. At the top left is the Apple logo followed by "iTunes Connect". At the top right is the user name "nahir\_khan" with a dropdown arrow. Below the header is a horizontal line, followed by the section title "Frequently Asked Questions" and another horizontal line. The content is organized into two columns. The left column contains four items: "Getting Started" (General questions including: how to use iTunes Connect and login and account issues.), "Payments and Financial Reports" (Get more information about payments as well as obtaining and understanding your financial reports.), "Manage Users" (Learn how to add and edit your iTunes Connect users), and "Manage Your Apps" (Learn how to manage and edit your apps using iTunes Connect). The right column contains two items: "Banking and Tax" (Get information relating to tax form requirements, assistance completing tax forms and setting up your bank account on iTunes Connect.) and "App Store" (Learn more about the App Store, how your applications are organized and displayed, and how to address reported customer download issues). Below the "App Store" item is the "Promo Codes" item (Understand what it means to request promo codes and how they can be used).

**Apple iTunes Connect** nahir\_khan ▾

---

**Frequently Asked Questions**

---

**Getting Started**  
General questions including: how to use iTunes Connect and login and account issues.

**Payments and Financial Reports**  
Get more information about payments as well as obtaining and understanding your financial reports.

**Manage Users**  
Learn how to add and edit your iTunes Connect users

**Manage Your Apps**  
Learn how to manage and edit your apps using iTunes Connect

**Banking and Tax**  
Get information relating to tax form requirements, assistance completing tax forms and setting up your bank account on iTunes Connect.

**App Store**  
Learn more about the App Store, how your applications are organized and displayed, and how to address reported customer download issues

**Promo Codes**  
Understand what it means to request promo codes and how they can be used



# Contact Us Page

- Drill down menu
- Where to get help

The screenshot shows the iTunes Connect 'Contact Us' page. At the top left is the 'iTunes Connect Contact Us' header, and at the top right is the user name 'nahir\_khan'. The main content area is titled 'Contact Us' and contains a welcome message: '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.'

Below the welcome message is a three-step drill-down menu:

1. Please choose from the following topics: Banking and Tax
2. Please choose from the following topics: Canada Tax Forms
3. Please choose from the following topics: Where to Send Forms

Underneath the menu is a 'Frequently Asked Questions' section. The first question is 'Where do I send completed Canadian tax form(s)?' with the answer: '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'. A link 'contact us.' is provided for further assistance.

Below the FAQ is a 'Related FAQs' section with three links:

- 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?

At the bottom right of the main content area are two buttons: 'Start Over' and 'Done'. The footer contains navigation links: 'Home | FAQs | Contact Us | Sign Out' and copyright information: 'Copyright © 2013 Apple Inc. All rights reserved. Terms of Service | Privacy Policy'.

# More Information

- For iOS Developers:
  - [developer.apple.com/library/ios/iTunesConnectGuide](https://developer.apple.com/library/ios/iTunesConnectGuide)
- For Mac Developers
  - [developer.apple.com/library/mac/iTunesConnectGuide](https://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](https://devforums.apple.com)

## App Review Contact Form:

[developer.apple.com/appstore](https://developer.apple.com/appstore)

## Developer Technical Support:

[developer.apple.com/support/technical](https://developer.apple.com/support/technical)

## Bug Reporter:

[bugreport.apple.com](https://bugreport.apple.com)

## More Information on Bug Reporting:

[developer.apple.com/bug-reporting](https://developer.apple.com/bug-reporting)



# More Information

## **TN2239 - iOS Debugging Magic**

[developer.apple.com/library/ios/#technotes/tn2239/\\_index.html](https://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](https://developer.apple.com/library/ios/#technotes/tn2151/_index.html)

## **QA1747 - Debugging Deployed iOS Apps**

[developer.apple.com/library/ios/#qa/qa1747/\\_index.html](https://developer.apple.com/library/ios/#qa/qa1747/_index.html)

## **TN2285 - Testing iOS App Updates**

[developer.apple.com/library/ios/#technotes/tn2285/\\_index.html](https://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](https://developer.apple.com/library/ios/#qa/qa1764/_index.html)

**TN2124 - Mac OS X Debugging Magic**

[developer.apple.com/library/mac/#technotes/tn2124/\\_index.html](https://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](https://developer.apple.com/library/mac/#qa/qa1778/_index.html)

 WWDC2013