

Hi

Marco Sero

@marcosero



Microsoft

ios@Yammer



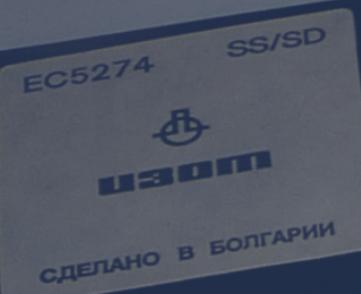
WORKING ON A LEGACY APP



DEFINITION OF LEGACY



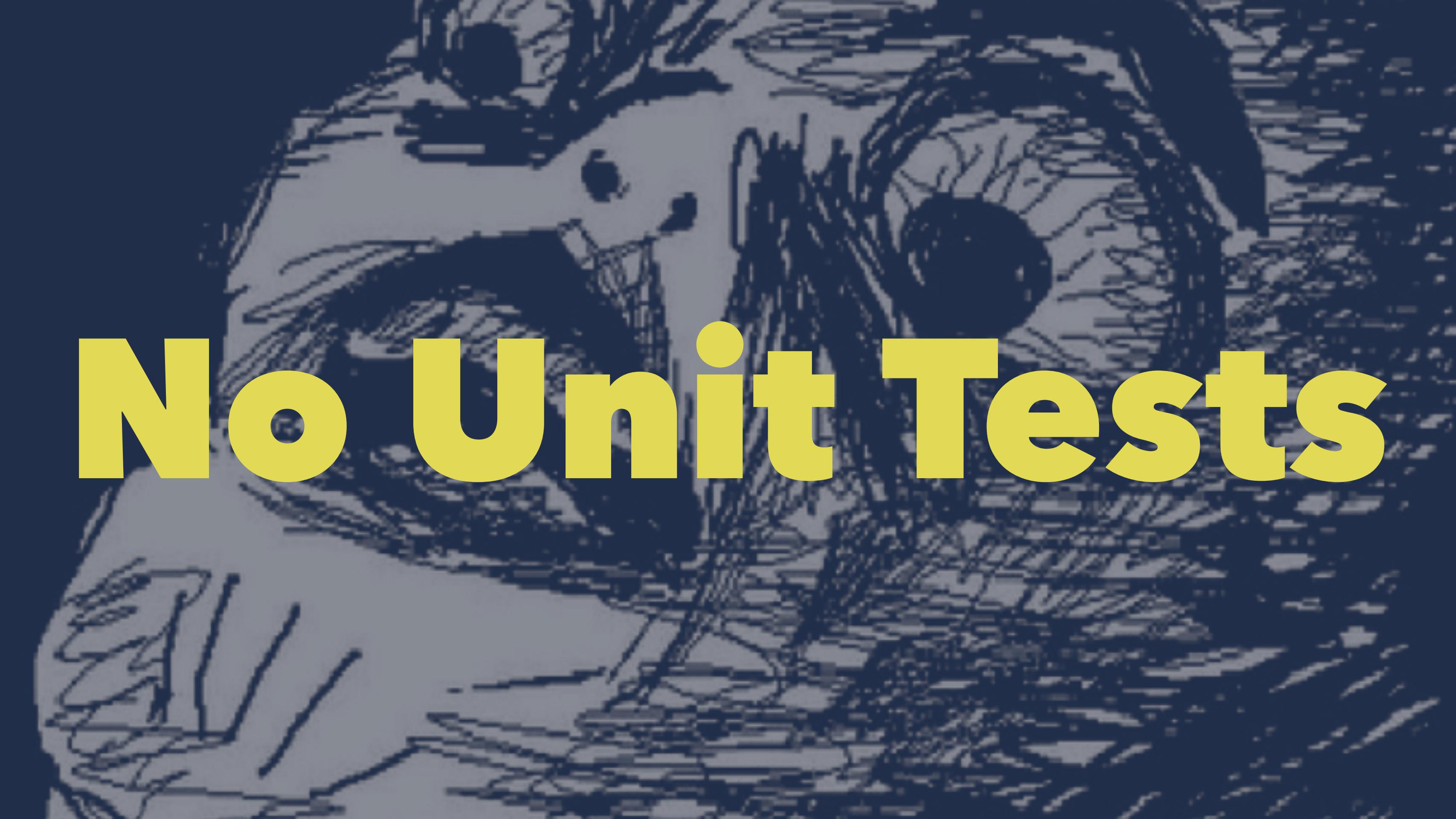
"Anything handed down from the past, as from an ancestor or predecessor"¹¹



“Obsolete.”



DEFINITION OF LEGACY CODE



No Unit Tests



**Still used in
Production**

IT JUST WORKS*

*most of the time





**written in a language
that you don't
typically do new
development in**



Swift 1.2



old and complicated
history

A photograph of four children in a classroom setting. A girl in a blue sweater is on the left, looking down at a white laptop. Behind her, another girl in a pink sweater is also looking at the screen. To the right, two boys in green and blue shirts are looking intently. In the foreground, there's a green apple and a yellow cup holding several colorful pens or markers. A globe is visible in the background.

**Many
contributors...**

A black and white photograph of a man with dark hair and a mustache, wearing a dark suit jacket over a light-colored shirt. He is looking down and slightly to his left, holding a piece of paper in his hands. The background is dark and out of focus.

...That have all quit

Let's recap

- ✓ Still used in production 💥
- ✓ Old or deprecated language 🔥
- ✓ Without unit tests 😱
- ✓ Old history with many, many contributors... 
- ✓ ...most of which not around anymore 



Or whenever
you feel like this

A vibrant photograph of a group of people at a festive party. Confetti in shades of pink, blue, and yellow is falling from the top of the frame. In the foreground, a woman on the left wears a yellow party hat and a red dress, laughing joyfully. Next to her, another woman in a blue dress is also laughing. In the center, a man in a white shirt is smiling. To the right, a woman in a pink dress and a silver party hat is looking towards the camera with a wide smile. The background is dark, making the colorful confetti and the subjects stand out.

Not much
legacy
on iOS

A photograph of a group of people at a party. Several individuals are wearing festive party hats. Confetti is falling from above, creating a celebratory atmosphere. The background is dark, making the colorful confetti stand out.

iOS development is new(ish)

Short app lifetime

Easier to re-write

So why bother?

```
~/Yammer/app master ✓  
► git rev-list --max-parents=0 HEAD | tail -1 | xargs git l  
* db886d1 (7 years) <aa> first commit
```

```
~/Yammer/app master ✓  
► find Yammer Shared/YamKit -name "*.m" | wc -l  
694
```

```
~/Yammer/app master ✓  
► ( find Yammer Shared/YamKit -name "*.m" -print0 | xargs -0 cat ) | wc -l  
117399
```

```
~/Yammer/app master ✓  
► git log --format='%aN' | sort -u | wc -l  
79
```

BAD CODE



BAD CODE EVERYWHERE

JUST

STOP

PLEASE

STOPPING

COMPLAINING



**WHERE DO I
START?**

STATS

Identify the risk and start there

```
$ find YourApp -iname "*.h" |  
xargs grep -h 'interface' |  
cut -d ' ' -f 2 |  
while read class;  
do echo `grep -r "\b$class\b" YourApp --include "*.m" | wc -l` $class;  
done |  
sort -n -u
```

from Destroy All Software screencasts.

```
~/Yammer/app  master ✓
▶ find Yammer Shared/YamKit -iname "*.h" |\
  xargs grep -h 'interface' |\
  cut -d ' ' -f 2 |\
  while read class;\
  do echo `grep -r "\b$class\b" Yammer Shared/YamKit --include "*.m" | wc -l` $class;\
  done |\
  sort -n -u |\
  tail -n 22
204 YDGroup
209 UIImageView
224 NSMutableArray
231 YDNetwork
233 UILabel
250 YDMessage
281 YDThread
287 NSURL
288 YDDataStore
312 UITableView
354 YDUser
395 YSTreatmentManager
396 NSError
461 UIFont
569 UIImage
603 UIView
709 NSManagedObjectContext
773 NSNumber
790 NSDictionary
841 NSArray
1280 UIColor
2662 NSString
```

**leverage
git**

git blame -go





Xcode quit unexpectedly.

Click Reopen to open the application again. Click Report to see more detailed information and send a report to Apple.



Ignore

Report...

Reopen



		16 Apr 2013	i
	Cleared all files on sign out for clean restart	26 Mar 2014	
	Allow autorotation on login/signup for iPad, but turn it off for iPhone.	8 May 2013	
	make sure the login controller is only shown once, add messages to NotificationCell	26 Apr 2013	
nit];		5 Nov 2013	
gationBarClass:		7 Apr 2015	
	Added search bar view	26 Mar 2014	
.navigationBar];		10 Sep 2014	
		15 Oct 2013	
		14 Sep 2013	
		9 Jan 2014	
	improve 'show server field by default' hack so it works also on the iPhone	26 Apr 2013	
	User signin	26 Apr 2013	

git log -L 1,1:/your/file

A close-up photograph of a man with light brown hair and glasses, smiling broadly and pointing his right index finger directly at the viewer. He is wearing a dark-colored shirt. The background is dark and out of focus.

COMMIT MESSAGES



But I'm NOT giving up just yet.

MarcoSero authored on Dec 3, 2013



Not it doesn't.

MarcoSero authored on Dec 3, 2013



Kind of works

MarcoSero authored on Dec 3, 2013



But I'm NOT giving up just yet.

MarcoSero authored on Dec 3, 2013



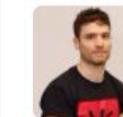
Not it doesn't.

MarcoSero authored on Dec 3, 2013



Kind of works

MarcoSero authored on Dec 3, 2013



Avoid request network if app just started ...

msero authored 5 days ago

This will avoid requesting the networks again when the user just opened the app or just logged in.



Change prefetching logic ...

msero authored 6 days ago

Prefetching the content when the app was becoming active was not enough as we were not covering 2 other cases:

- first login
- change of network

This should now cover all the different cases

Commits on Aug 11, 2015



Support transparent navigation bar in gallery ...

msero authored 5 days ago

Having a transparent/translucent navigation bar in the gallery, it's a lot of work.

First off, having a translucent nav bar changes the frames of all the scrollviews in the navigation controller and puts their content underneath the nav bar.

To avoid that we need to manually set edgesForExtendedLayout to .Bottom to make sure the scroll view doesn't go below the nav bar.

Also, the snapshot gets more complicated since we need to take before the gallery set the nav bar translucent.

Your
COMMIT MESSAGES
are your
LEGACY

(Maybe) DEAD
CO. D.E.

A photograph of a grey granite tombstone. A small, rectangular plaque is attached to the front of the stone. The letters 'BY' are visible on the plaque.

TOMBSTONES

```
- (void)possibleDeadCode  
{  
    // current date when tombstone was added  
    log_tombstone(@"2015-08-21");  
}
```

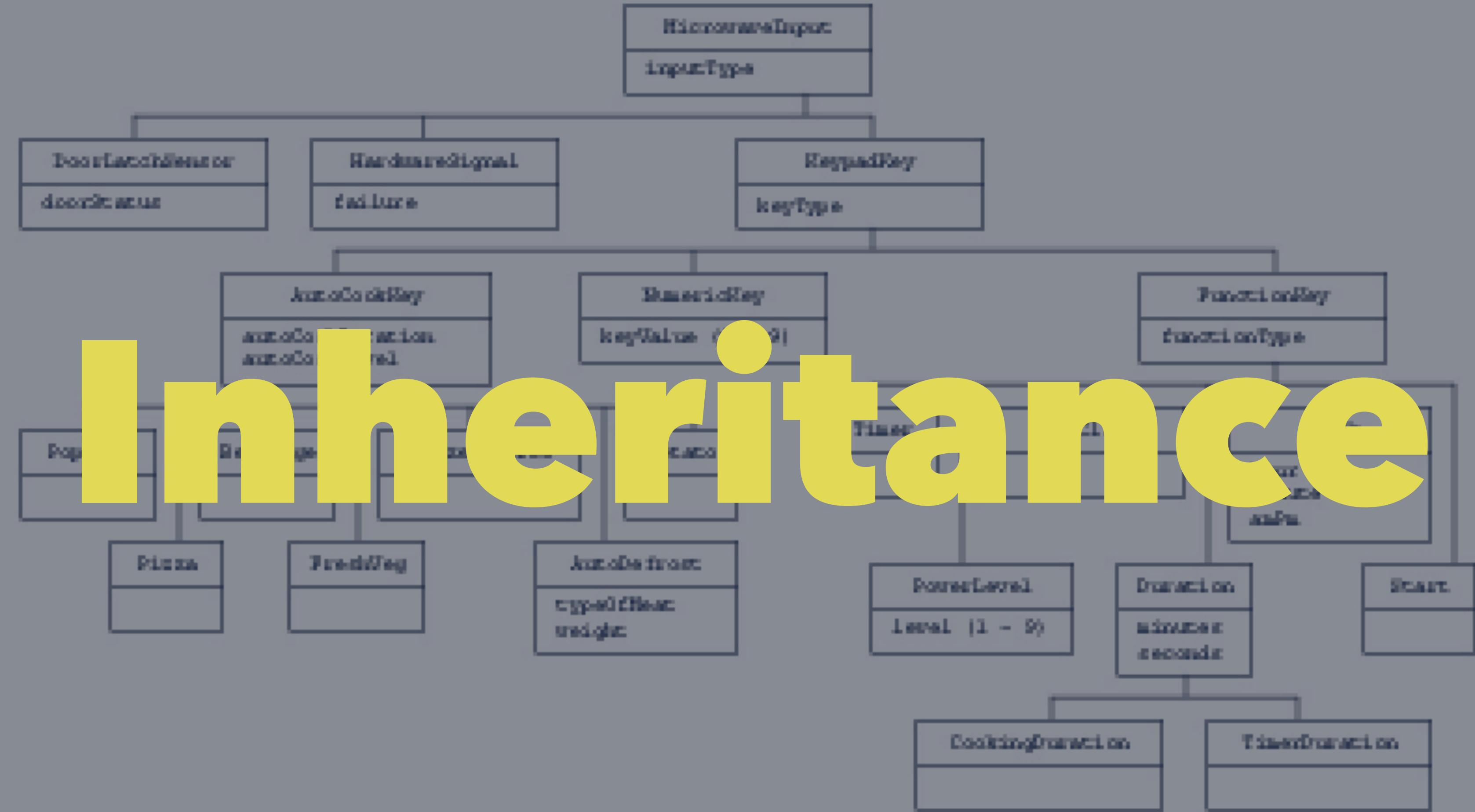
- Log found?
→ Code is alive 🙌
- Log not found?
→ Code is dead 💀

From David Schnepper's Ignite talk, "Isn't That Code Dead?"

The background image shows a dense urban landscape with numerous skyscrapers. In the lower-left foreground, a large, multi-story building is being demolished. A massive, billowing cloud of dust and debris is visible, rising from the base of the building and spreading across several stories. The scene captures a moment of significant urban change and destruction.

CONTINUOUS DESTRUCTION

inheritance





Marco Sero

@marcosero

Most of the bugs I've been fixing in the past months were due to inheritance.

```
@implementation RootViewController
- (void)doSomething:(NSString *)aString { }
@end
```

```
@implementation RootViewController_iPad
- (void)doSomething:(NSString *)aString {
    NSLog(@"%@", iPad, aString);
}
@end
```

```
@implementation RootViewController_iPhone
- (void)doSomething:(NSString *)aString {
    NSLog(@"%@", iPhone, aString);
}
@end
```

```
RootViewController *vc = [[RootViewController_iPad alloc] init];
[vc doSomething:@"cool"]; // prints "cool iPad"
```

```
@implementation RootViewController
- (void)doSomething:(NSString *)aString { }
@end
```

```
@implementation RootViewController_iPad
- (void)doSomething {
    NSLog(@"%@", @"iPad");
}
@end
```

```
@implementation RootViewController_iPhone
- (void)doSomething:(NSString *)aString {
    NSLog(@"%@", @"iPhone", aString);
}
@end
```

```
RootViewController *vc = [[RootViewController_iPad alloc] init];
[vc doSomething:@"cool"]; // prints nothing
```

```
class RootViewController: UIViewController {
    func doSomething(aString: String) { }
}

class RootViewController_iPad: RootViewController {
    override func doSomething(aString: String) {
        print(aString + "iPad")
    }
}

class RootViewController_iPhone: RootViewController {
    override func doSomething(aString: String) {
        print(aString + "iPhone")
    }
}
```

```
class RootViewController: UIViewController {
    func doSomething(aString: String) { }
}

class RootViewController_iPad: RootViewController {
    override func doSomething() { // COMPILE ERROR
        print("iPad")
    }
}

class RootViewController_iPhone: RootViewController {
    override func doSomething(aString: String) {
        print(aString + "iPhone")
    }
}
```

"I've got your
back"

- Chris Lattner on Swift's Safety



THIRD-PARTY DEPENDENCIES

**AVOID
THEM**

(if possible)

Build a small, but non-trivial, Rails app. [...] Go away for six months. Come back and update all of your dependencies. Your app no longer works.

– Gary Bernhardt

*Build a small, but non-trivial, iOS app. [...] Go away for six months.
Come back and update all of your dependencies. Your app no
longer works.*

– Marco Sero

A photograph of a man in a dark suit jacket and a striped shirt, holding a book titled "QUINN" in front of his face. He is standing in what appears to be a library or bookstore. In the background, a woman in a red and black plaid dress is visible. The overall mood is mysterious and dramatic.

FORKS

Recap:

- Start identifying the risk
- Leverage Git
- Spend some time writing good commit messages
- Continuous destruction of dead code
- Avoid subclasses and touch them with care
- Third party dependencies have a cost and private forks are almost always a bad idea



Gary Bernhardt

@garybernhardt



Following

instead of "software engineer", consider
"technical debt broker"

RETWEETS

436

FAVOURITES

410



6:26 p.m. - 4 Aug 2015



...

A photograph showing several people in yellow safety vests with "FRIENDS BRP AT WORK" printed on the back, working together to rake fallen autumn leaves into large piles in a park setting. One person in the foreground is using a hand rake, while others are using leaf blowers. The scene is outdoors with trees and a paved path in the background.

BE A GOOD CITIZEN

HAVE FUN



Thank you.

@marcosero

- *Working Effectively with Legacy Code* by **Michael Feathers**
- *Utter Disregard for Git Commit History* by **Zach Holman**
<http://zachholman.com/posts/git-commit-history/>
- *Destroy All Software* screencast by **Gary Bernhardt**
<https://www.destroyallsoftware.com/screencasts>
- *Isn't That Code Dead?* by **David Schnepper**
<https://www.youtube.com/watch?v=29UXzfQWOhQ>