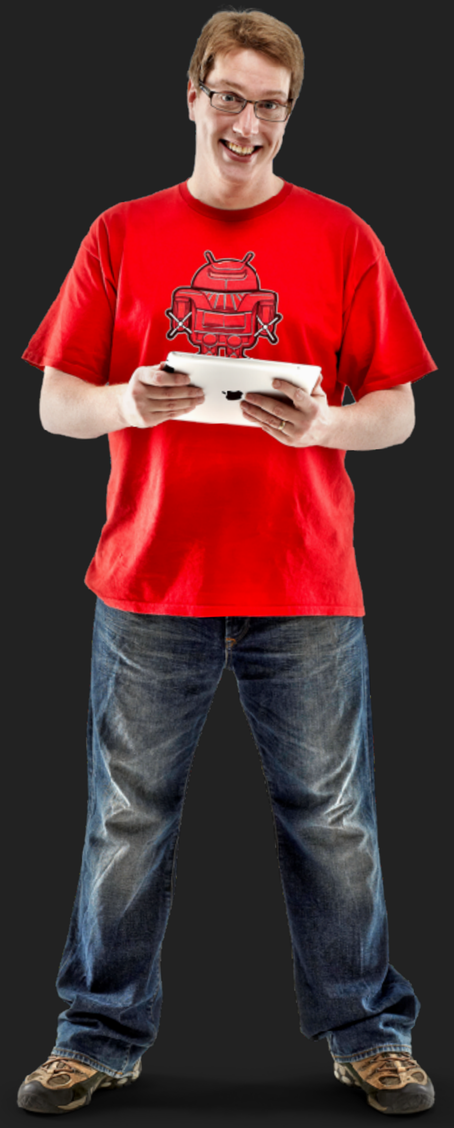
PAUL HUDSON – @twostraws

NUKE IT FROM ORBIT: How to Fix Massive View Controller Once and for All

https://www.hackingwithswift.com



HACKING WITH SWIFT

Swift

IN SIXTY SECONDS

2017
SWIFT COMMUNITY AWARDS
SWIFT
COMMUNITY
AWARDS

HACKING
WITH SWIFT
LEARN iOS DEVELOPMENT WITH REAL WORLD PROJECTS

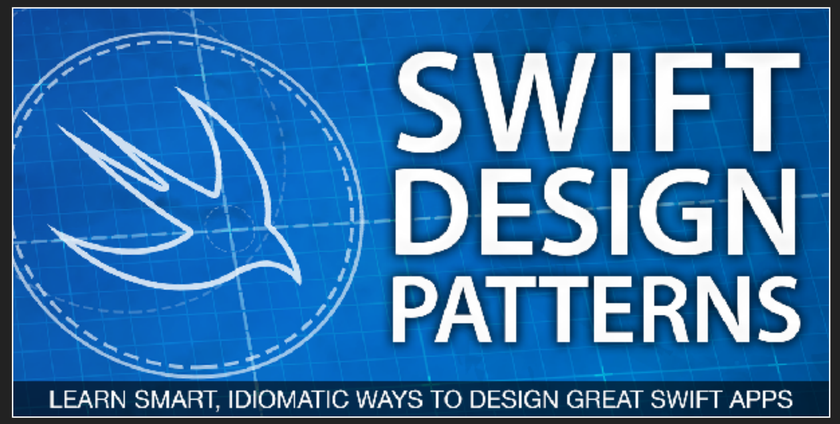
HACKING WITH
macOS
LEARN macOS DEVELOPMENT WITH REAL WORLD PROJECTS

HACKING WITH
watchOS
LEARN TO BUILD WATCHOS APPS WITH REAL-WORLD PROJECTS

HACKING WITH
tvOS
LEARN TO BUILD tvOS APPS WITH REAL-WORLD PROJECTS

PRO
SWIFT
BREAK OUT OF BEGINNER'S SWIFT WITH THIS HANDS-ON GUIDE

SERVER-SIDE
SWIFT
LEARN TO BUILD SWIFT WEBSITES WITH REAL-WORLD PROJECTS


SWIFT
DESIGN
PATTERNS
LEARN SMART, IDIOMATIC WAYS TO DESIGN GREAT SWIFT APPS

SWIFT
CODING CHALLENGES
FROM THE AUTHOR OF HACKING WITH SWIFT AND PRO SWIFT

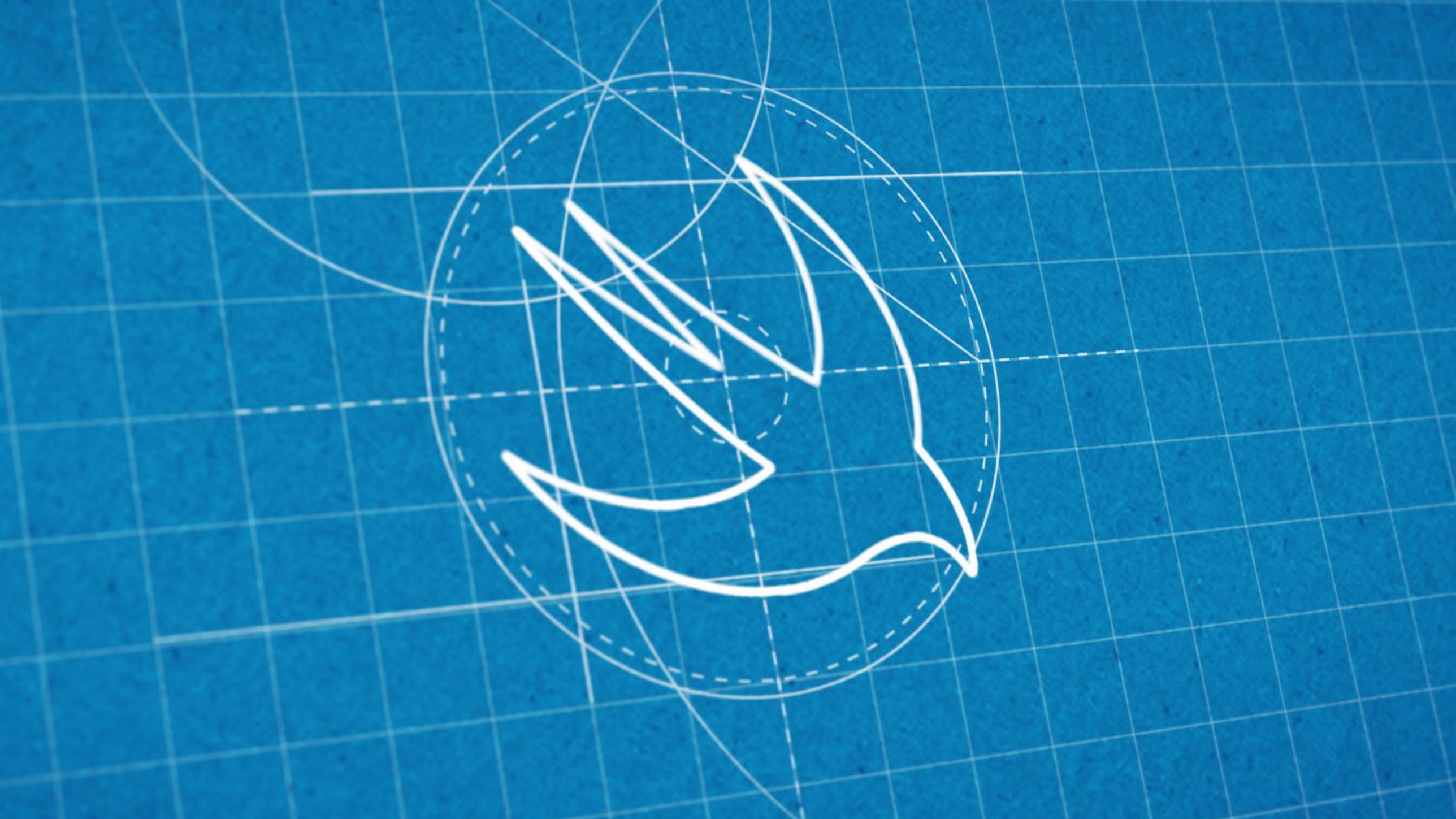
ADVANCED
iOS
VOLUME ONE
TAKE YOUR SKILLS FURTHER WITH ADVANCED PROJECTS

GET IN TOUCH!

- @twostraws on Twitter
- twostraws on GitHub
- twostraws on Reddit
- twostraws on StackOverflow
- paul@hackingwithswift.com

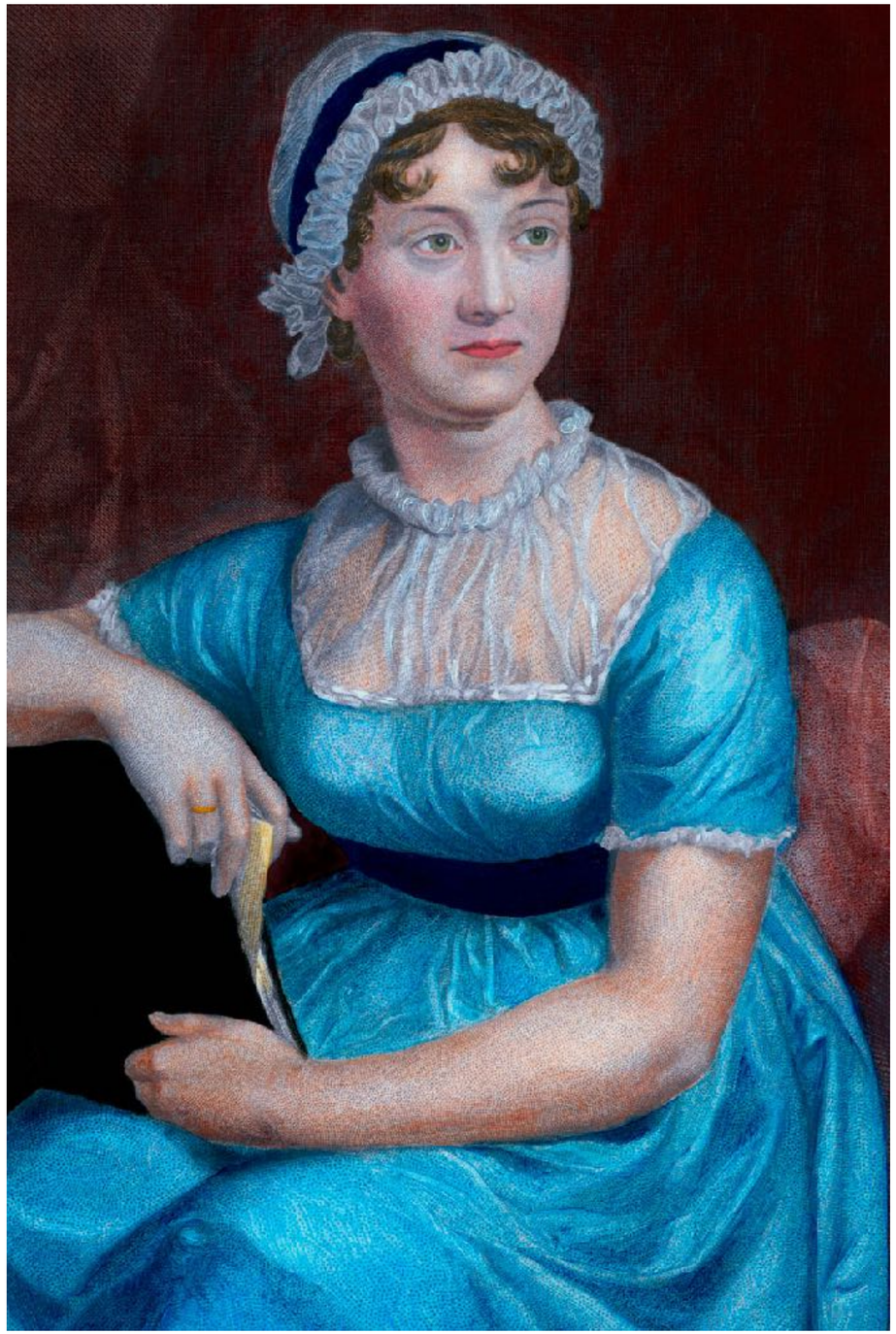
NUKE IT FROM ORBIT

How to fix massive view controllers
once and for all



SITE SAFETY

0 days since the last talk on iOS architectures



It is a truth universally acknowledged that every good iOS conference must be in want of a talk on app architectures.

Jane Austen

(Probably)

I can fix MVC and so can my wife!

THE PLAN

- What's the problem?
- What are the alternatives?
- How to fix MVC

- Lots of code to read

“I’m sorry, I can’t hear you over the sound of how awesome I am.”

CODE OR IT
DIDN'T
HAPPEN

THE PLAN

- What's the problem?
- What are the alternatives?
- How to fix MVC

- Lots of code to read

MVC

XCODE

WHAT'S THE PROBLEM?

iOScon
the conference for iOS
and Swift developers
organised by
skills matter
skillsmatter.com

MVC

MASSIVE VIEW CONTROLLER

Model

View

Controller

Model

View

Controller

Networking

Business logic

Data storage

Delegation

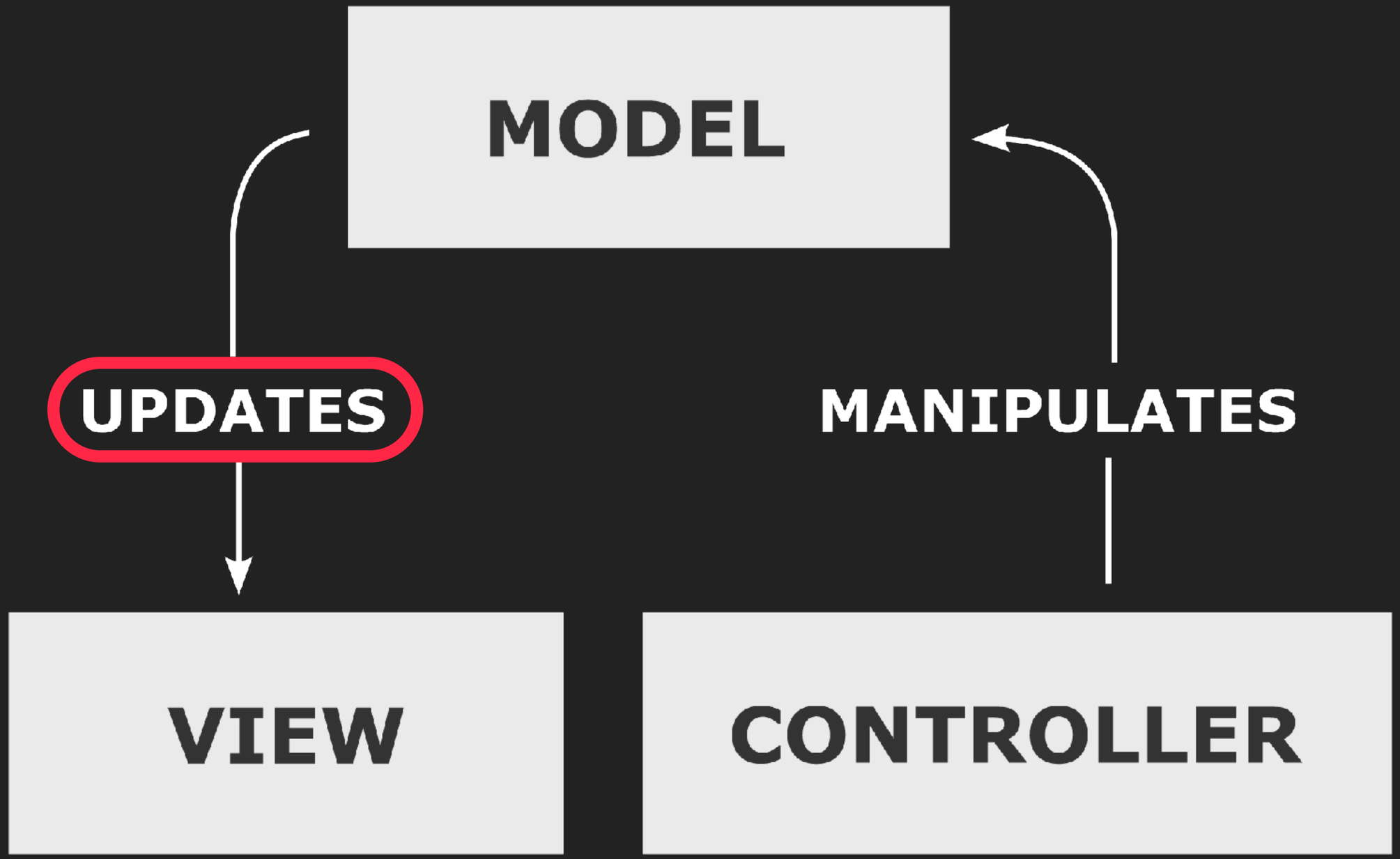
Kitchen Sink

Lifecycle Events

Segues

I/O

Animations


MODEL
UPDATES
MANIPULATES
VIEW
CONTROLLER

UIViewController

UIViewController

UIViewController

dotSwift
Paris 2018

UIViewController

dotSwift
Paris 2018

UIViewCon er

dot Swift
Paris 2018

UIViewCon er

Soroush Khanlou

@khanlou

“When you call something a Controller, it absolves you of the need to separate your concerns. Nothing is out of scope, since its purpose is to control things. Boundless, it begins absorbing responsibilities.”

cellForRowAt

Fetch model data

Format model data

Create table view cell

Lay out cell with formatted data

Render cell

Lay
ata

LOG IN:

USERNAME

PASSWORD

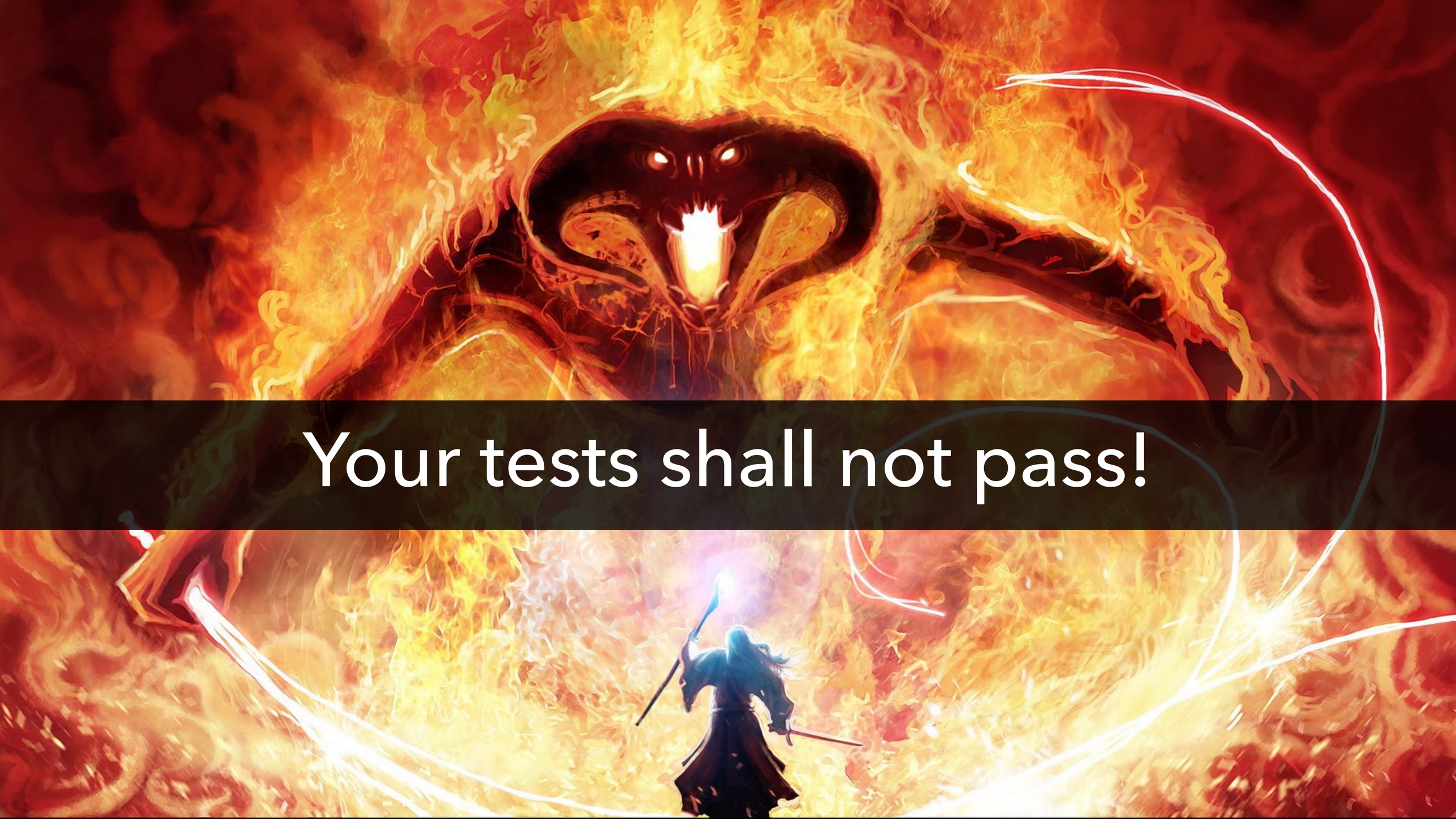
REMEMBER ME

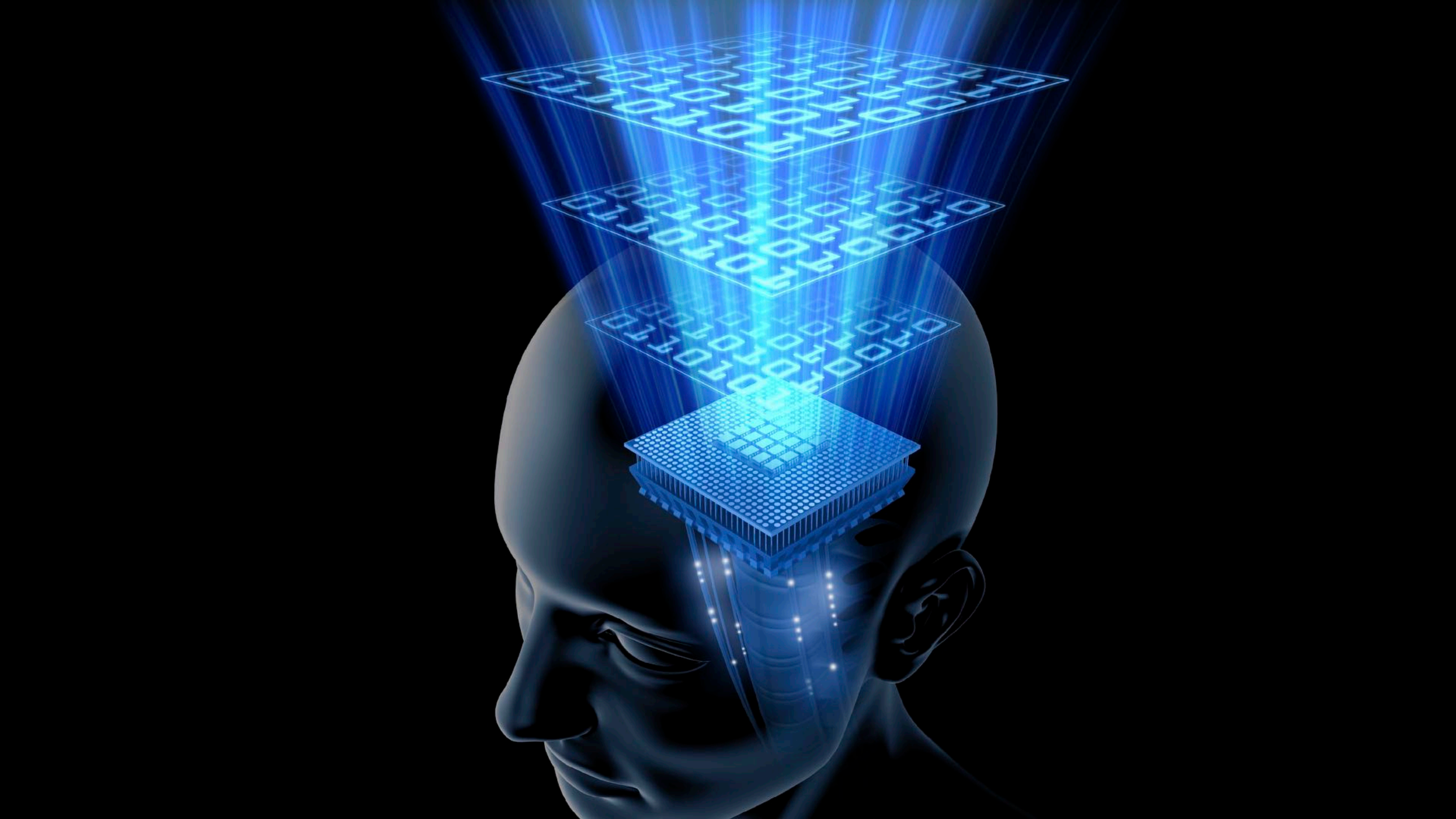
LOG IN

NOT A MEMBER?

SIGN UP

U MAD
BRO?


Your tests shall not pass!



WHAT ARE THE ALTERNATIVES?

MVC
VIP
MVVM
MVVM-C
VIPER

50 Points = 1 Coupon
100 Points = 2 Coupons
140
1
210
100
2

That’s not an architecture...
...THIS is an architecture!

Guilherme Rambo

@_inside

https://iosarchitecture.top

MVC
VIP
MVVM
MVVM-C
VIPER

Model

View

Controller

Networking

Business logic

Data storage

Delegation

Model

View

ViewModel

Networking

Business logic

Data storage

Delegation

Model

View

UIViewController

ViewModel

Networking

Business logic

Data storage

Delegation

~~import UIKit~~

ViewModel

Networking

Business logic

Data storage

Delegation

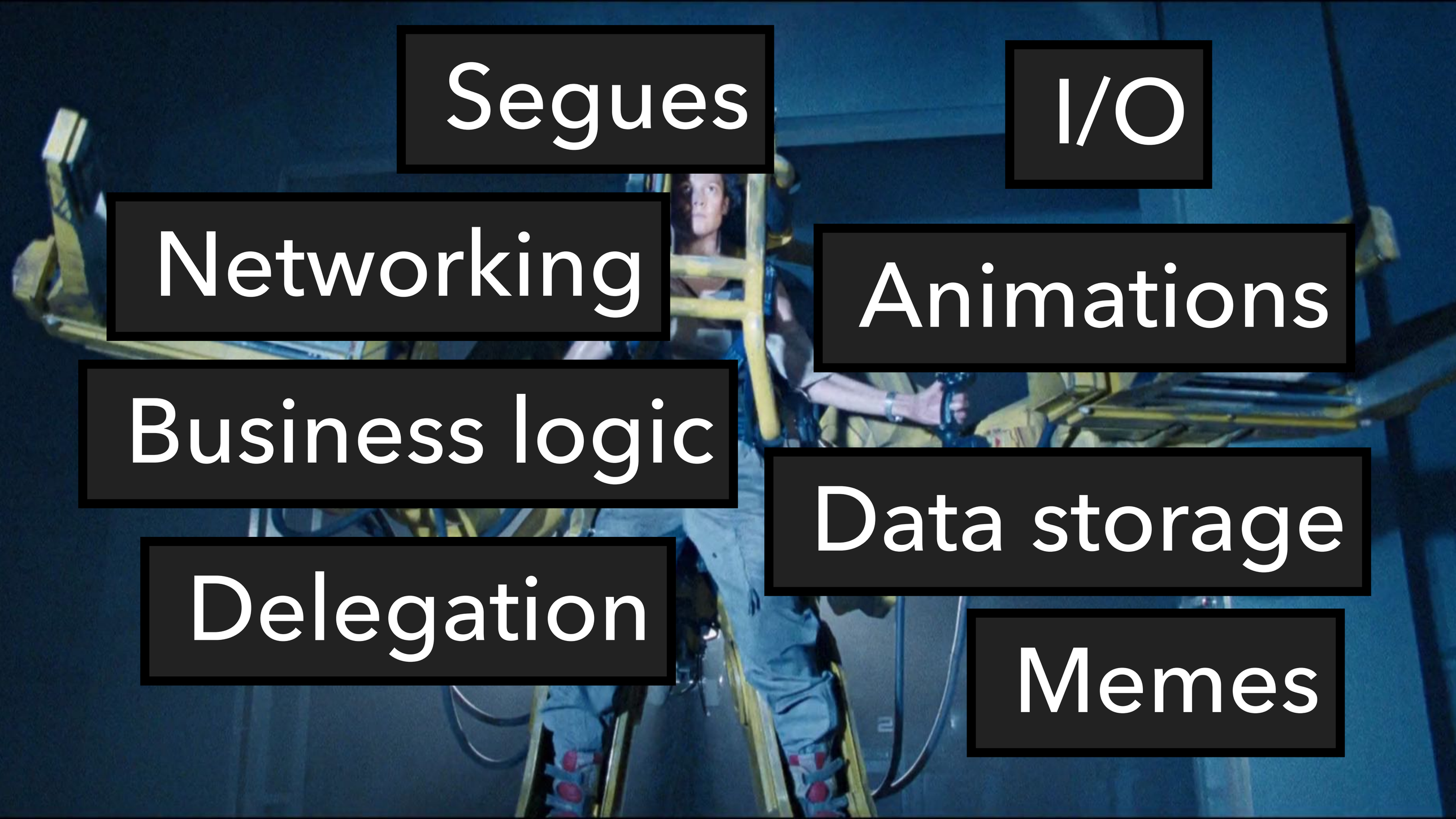
ViewModel

Networking

Business logic

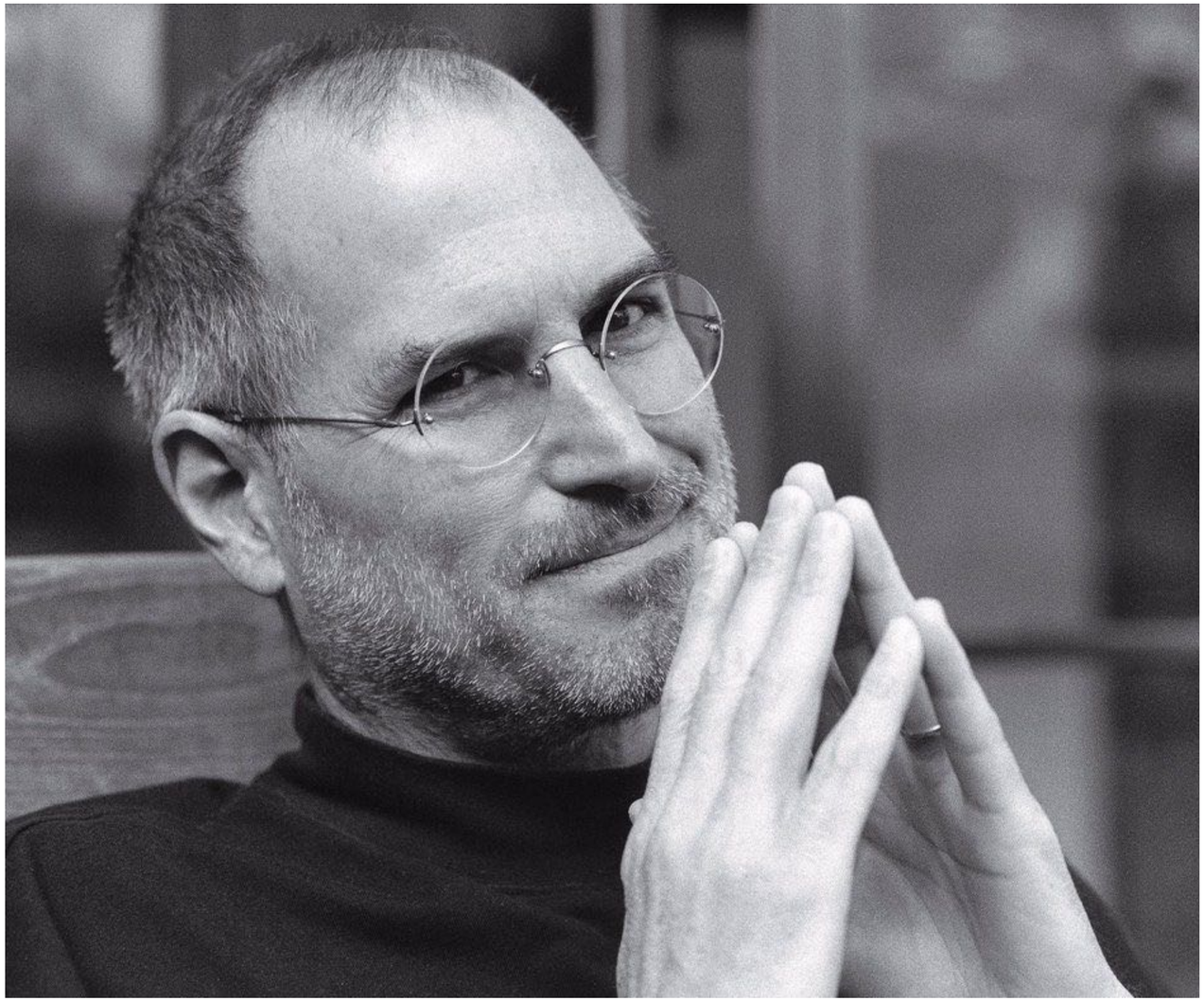
Data storage

Delegation


Segues
I/O
Networking
Animations
Business logic
Data storage
Delegation
Memes

- Person A: “No Scotsman puts sugar on his porridge.”
- Person B: “But my uncle Angus likes sugar with his porridge.”
- Person A: “Ah yes, but no *true* Scotsman puts sugar on his porridge.”

- Person A: “MVC architecture neatly separates models, views, and controllers.”
- Person B: “But all these apps put view code into the controller.”
- Person A: “Ah yes, but that’s not a *true* MVC architecture.”



Architecture not working for you?

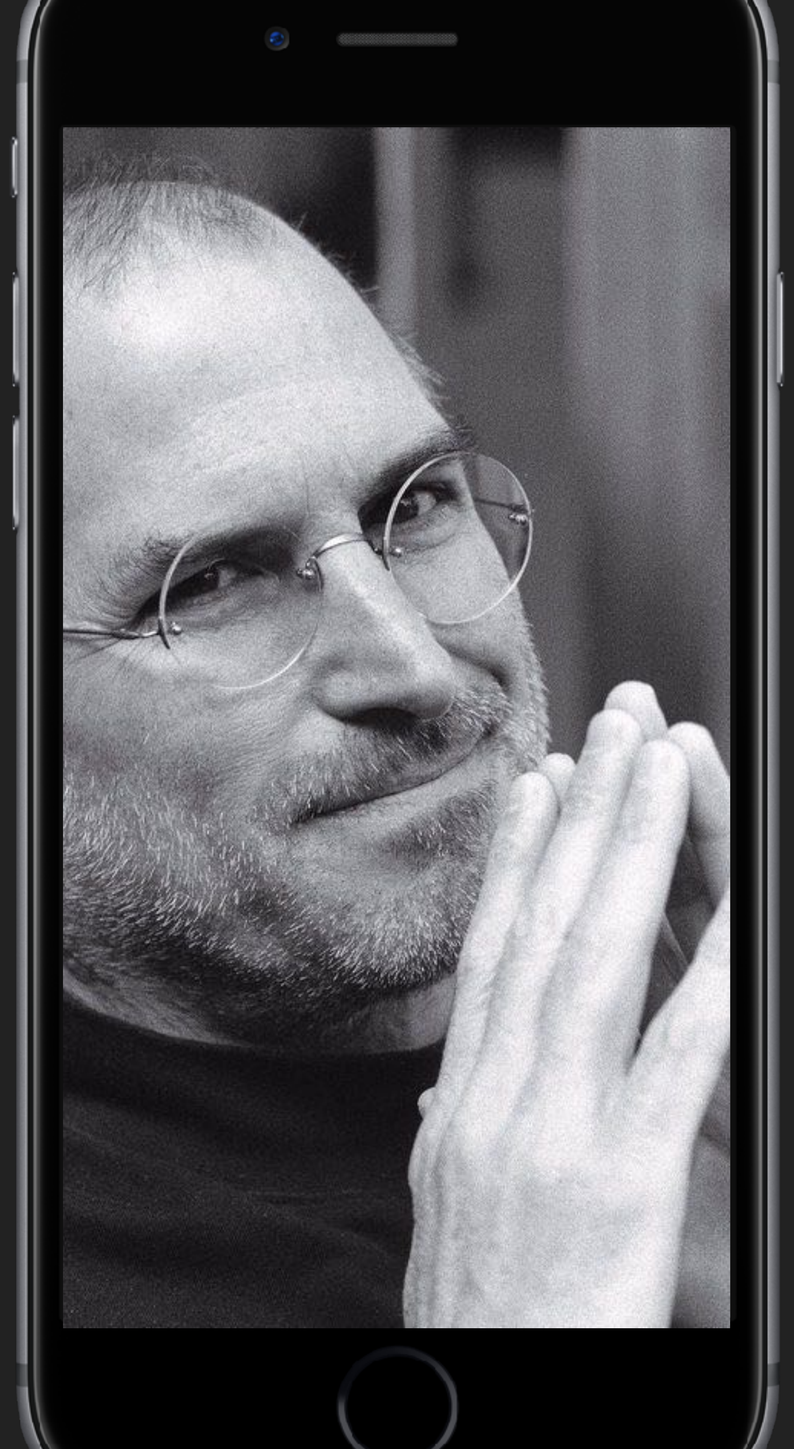
You're holding it wrong.

MVVM > MVC

>

THE PLAN

- ~~What's the problem?~~
- ~~What are the alternatives?~~
- How to fix MVC
- Lots of code to read



THE PLAN

- ~~What's the problem?~~
- ~~What are the alternatives?~~
- How to fix MVC

- Lots of code to read

MVC

YOU'RE HOLDING IT WRONG

1

```
class MegaController: UIViewController,
UITableViewDataSource, UITableViewDelegate,
UIPickerViewDataSource, UIPickerViewDelegate,
UITextFieldDelegate, WKNavigationDelegate,
URLSessionDownloadDelegate {
```

“Model, view, and exactly one controller”

WKNavigationDelegate

```
func webView(_ webView: WKWebView, decidePolicyFor
navigationAction: WKNavigationAction, decisionHandler:
@escaping (WKNavigationActionPolicy) -> Void) {
  if let host = navigationAction.request.url?.host {
    if allowedSites.contains(where: host.contains) {
      decisionHandler(.allow); return
    }
  }
  decisionHandler(.cancel)
}
```

```
import Foundation
import WebKit


class ChildFriendlyWebDelegate: NSObject,
WKNavigationDelegate {
   var allowedSites = ["apple.com", "google.com"]
   // paste the delegate code here
}
```

MegaController

UITableViewDataSource

UITableViewDelegate

UIPickerViewDataSource

UIPickerViewDelegate

UITextFieldDelegate

WKNavigationDelegate

URLSessionDownloadDelegate

Not all controllers are view controllers!

```
let childMode =  ChildFriendlyWebDelegate()
myWebView.navigationDelegate = childMode

webView2.navigationDelegate = childMode

let unlocked =  UnrestrictedWebDelegate()
myWebView.navigationDelegate = unlocked
```

😀 View controller is smaller

😀 Delegate follows
single responsibility

😭 More testable?

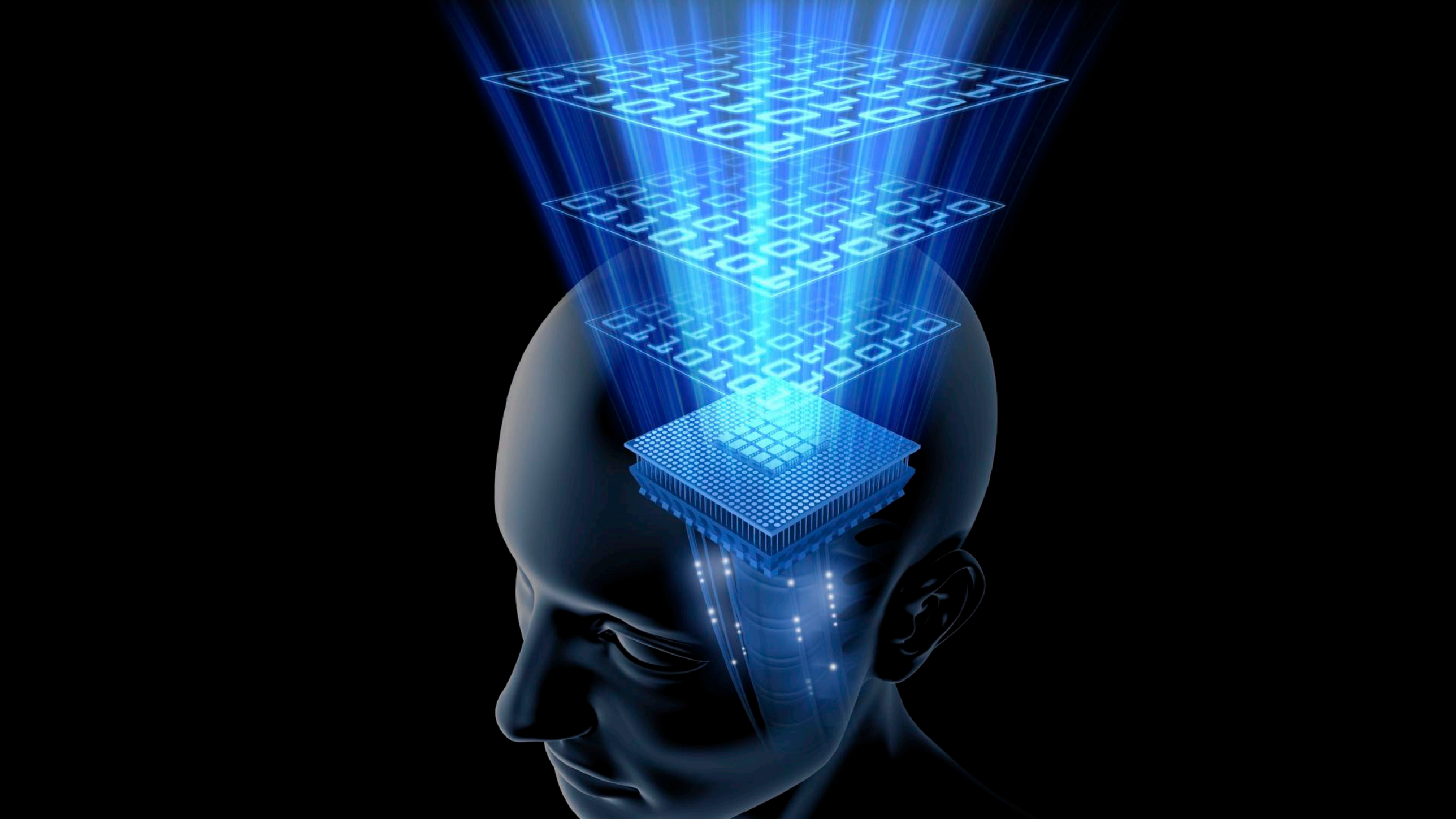
```
func isAllowed(url: URL?) -> Bool {
    guard let host = url?.host else { return false }
    if allowedSites.contains(where: host.contains) {
        return true
    }
    return false
}
```

```
func webView(_ webView: WKWebView, decidePolicyFor
navigationAction: WKNavigationAction, decisionHandler:
@escaping (WKNavigationActionPolicy) -> Void) {
  if isAllowed(url: navigationAction.request.url) {
    decisionHandler(.allow)
  } else {
    decisionHandler(.cancel)
  }
}
```

😀 View controller is smaller

😀 Delegate follows
single responsibility

😎 More testable



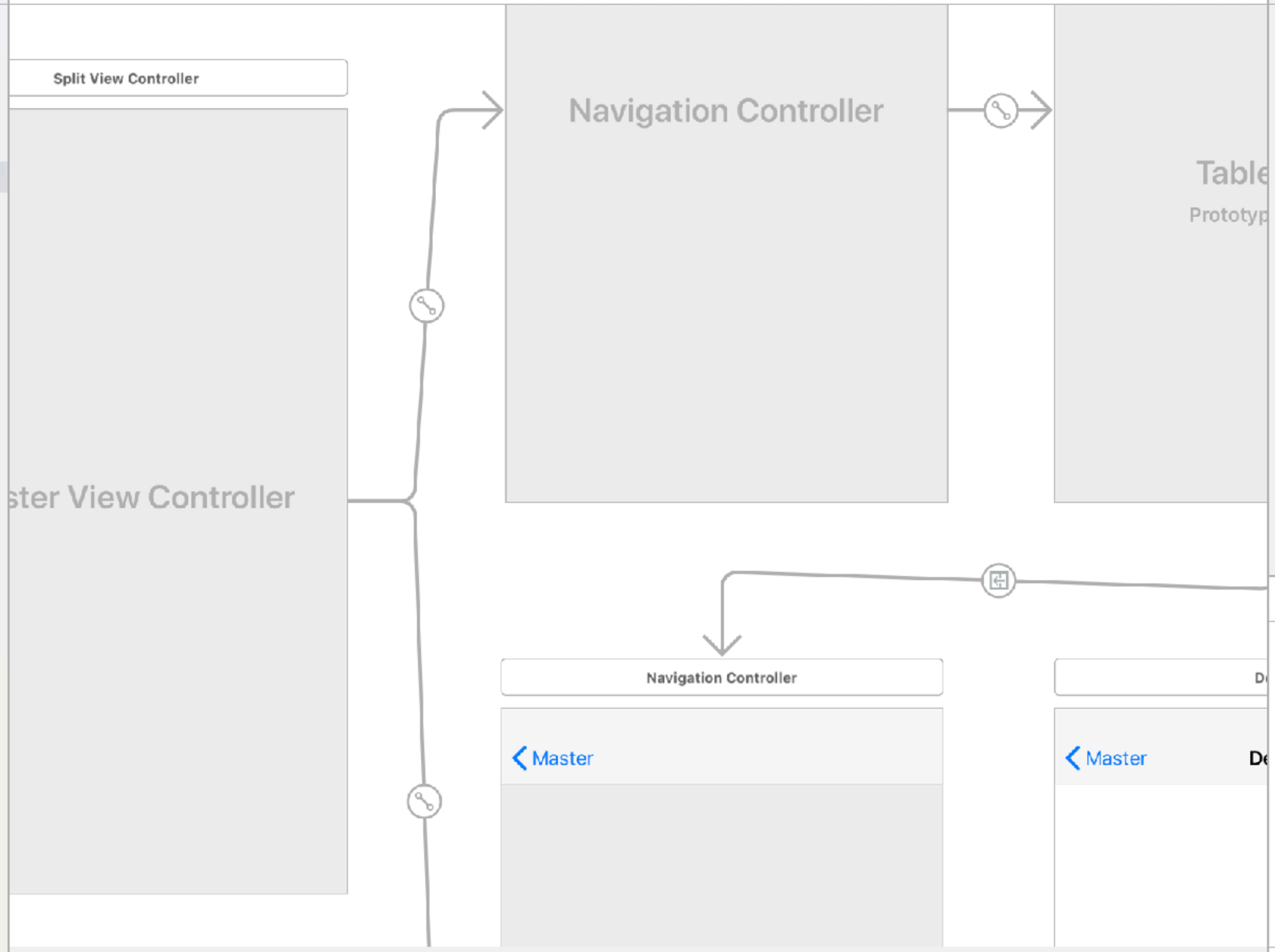
2

Xcode File Edit View Find Navigate Editor Product Debug Source Control Window Help Tue 16:15

IfYouCan...mpressed 〉 iPhone 8 Plus | IfYouCanReadThisIAmImpressed | Build **Succeeded** | Today at 16:06

IfYouCanReadThisIAmImpressed 〉 IfYouCanRea...mImpressed 〉 Main.storyboard 〉 Main.storyboard (Base) 〉 No Selection

- IfYouCanRead...sIAmImpressed
  - IfYouCanRea...AmImpressed
    - AppDelegate.swift
    - MasterView...ntroller.swift
    - DetailViewController.swift
    - Main.storyboard
    - Assets.xcassets
    - LaunchScreen.storyboard
    - Info.plist
  - IfYouCanRea...pressedTests
    - IfYouCanRe...dTests.swift
    - Info.plist
  - IfYouCanRea...ressedUITests
    - IfYouCanRe...ITests.swift
    - Info.plist
  - Products



No Selection

View Controller - A controller that manages a view.

Storyboard Reference - Provides a placeholder for a view controller in an external storyboard.

Navigation Controller - A controller that manages navigation through a hierarchy of views.

View as: iPhone 8 (wC hR) 75%

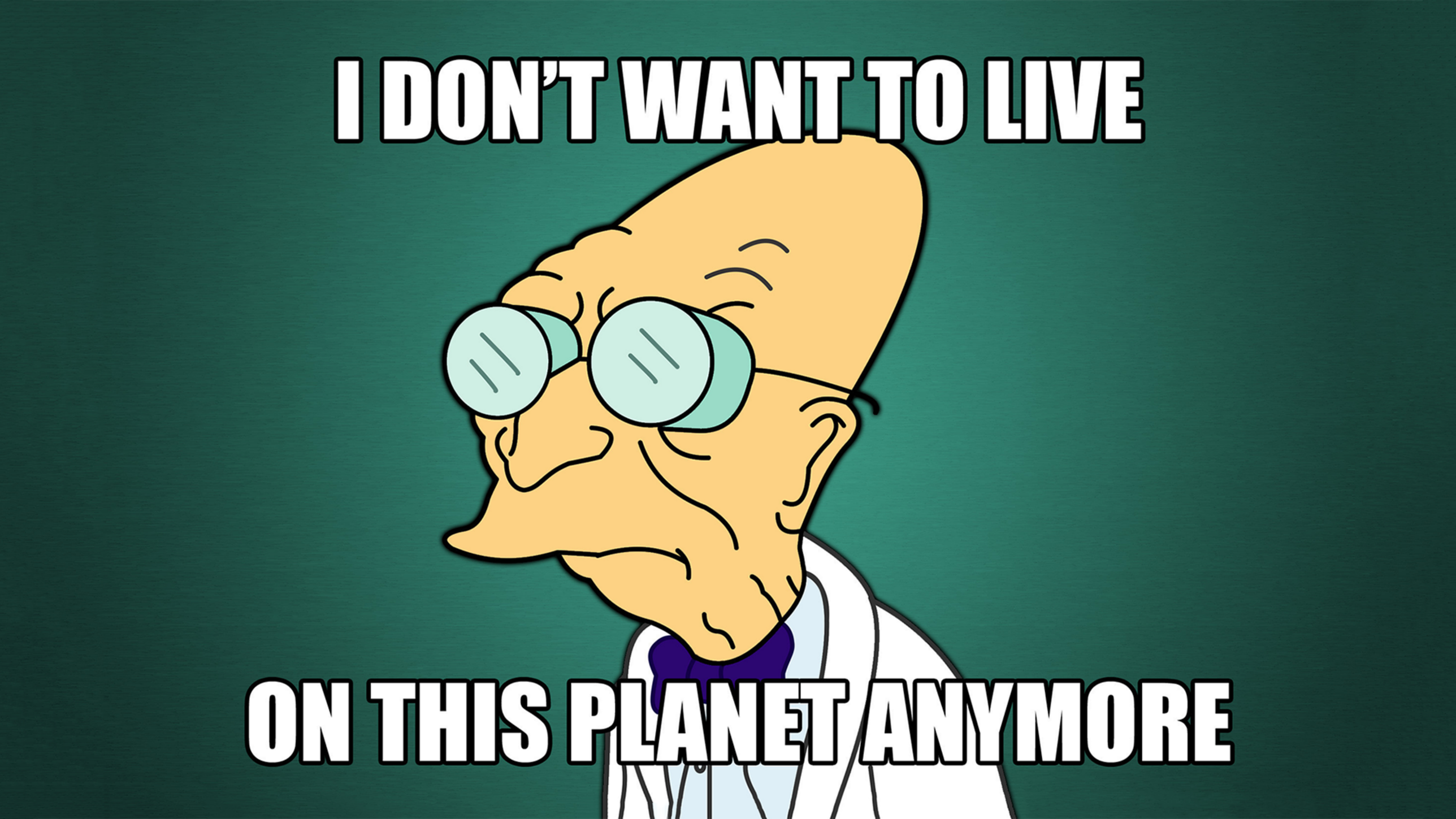
DO
NOT
WANT!

TMTOWTDI

```
func viewDidLoad() {
    backgroundColor = UIColor(white: 0.9, alpha: 1)

    let stack = UIStackView()
    stack.translatesAutoresizingMaskIntoConstraints = false
    stack.spacing = 10
    view.addSubview(stack)

    stack.topAnchor.constraint(equalTo:
    view.safeAreaLayoutGuide.topAnchor).isActive = true
```


I DON'T WANT TO LIVE
ON THIS PLANET ANYMORE

Coder refactors massive view controller with this one weird trick

Coder refactors massive view controller with this one weird trick

Coder refactors massive view controller with this one weird trick

Model

View

Controller

Dave DeLong
@DaveDeLong

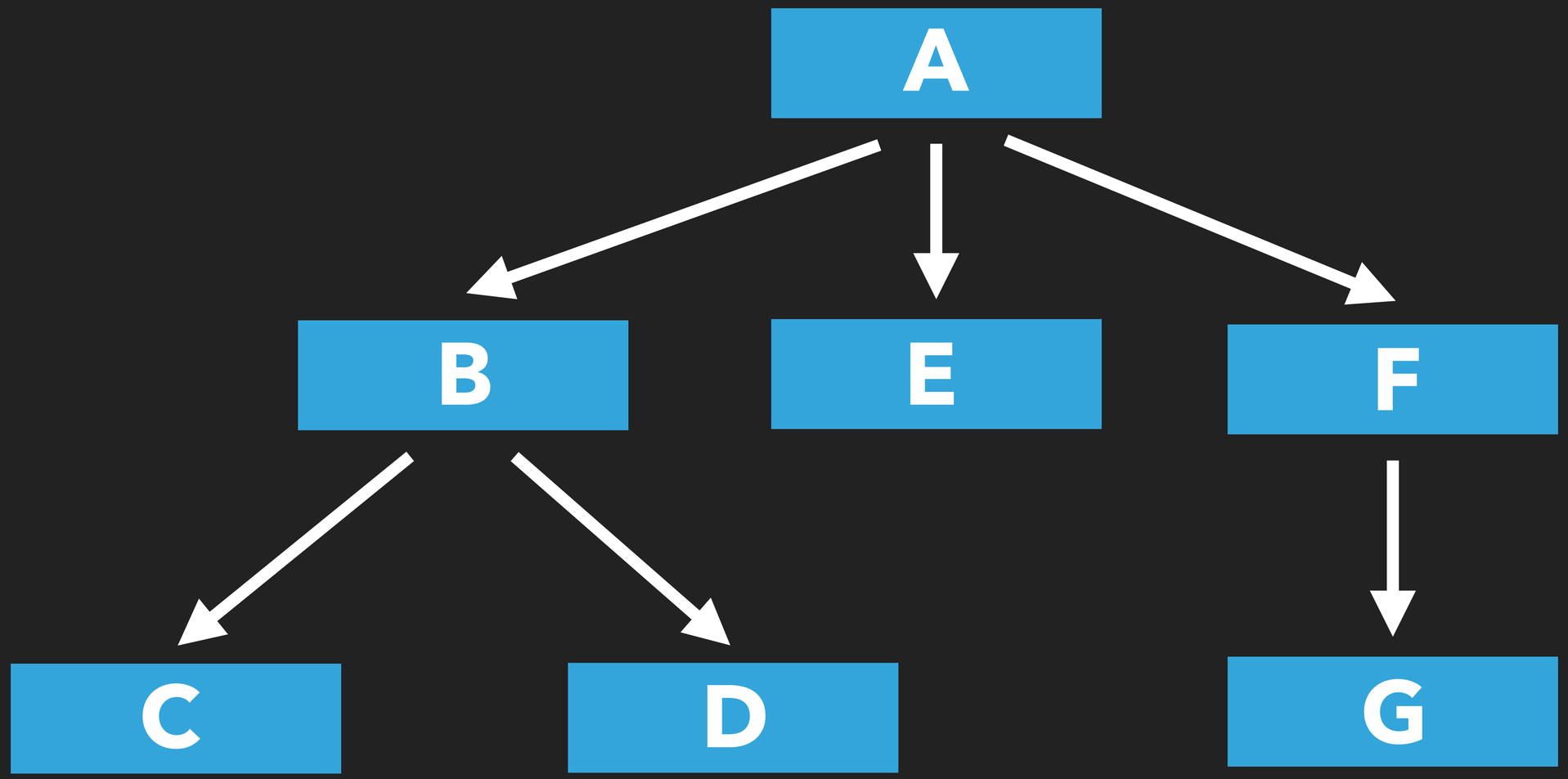
Views are small

Views are composeable

Views are reusable

```
class ViewController: UIViewController {
  var shareView = SharePromptView()

  override func loadView() {
    view = shareView
  }
}
```

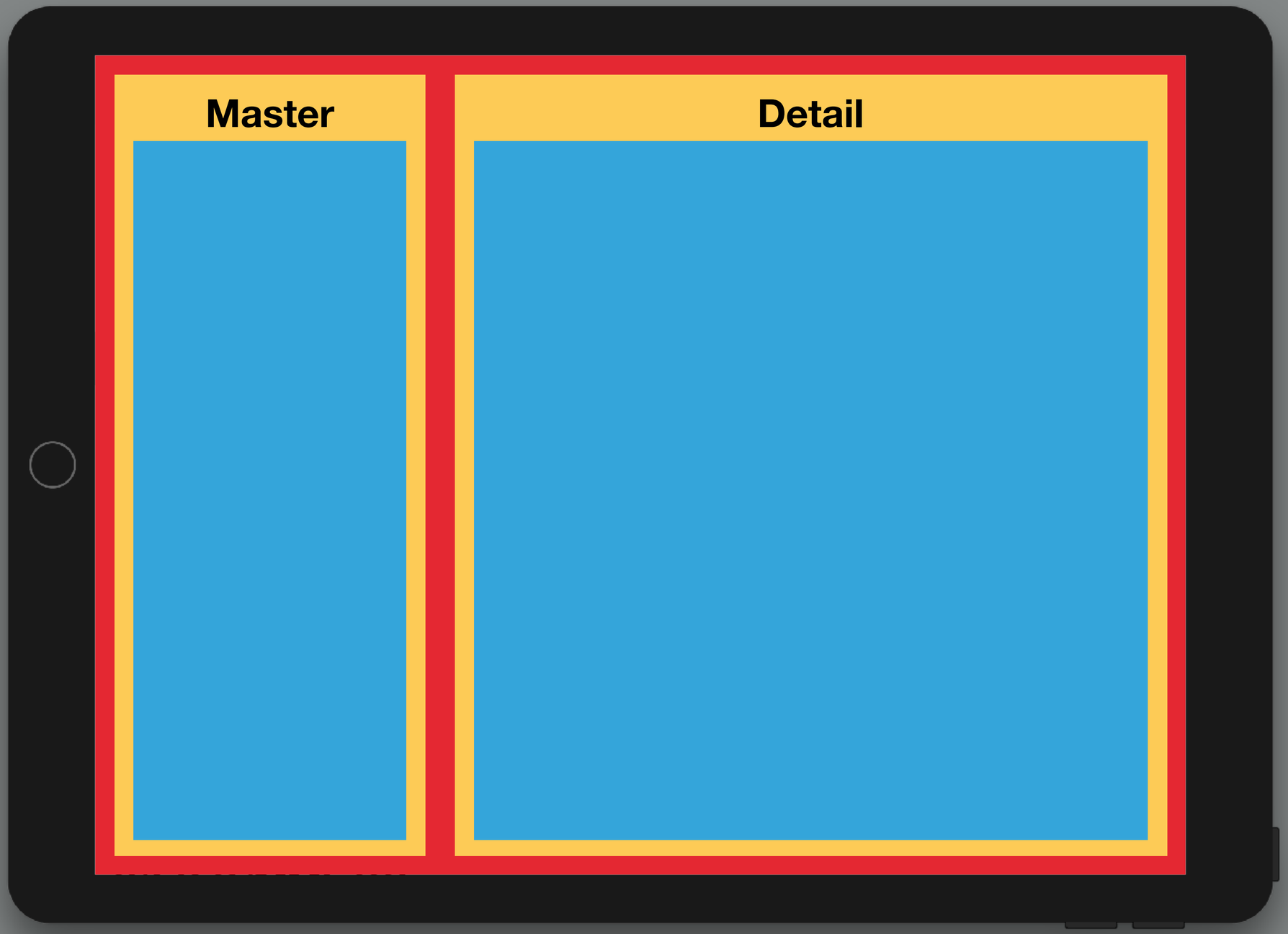

A
B
E
F
C
D
G

3

Wittgenstein’s Ladder

HACKING WITH SWIFT: PROJECT 1

“A view controller is best thought of as being one screen of information”


Master
Detail

WE NEED TO GO
DEEPER
memegenerator.net

1. Call **addChildViewController()** on your parent view controller, passing in your child.
2. Set the child's frame to whatever you need, if you're using frames.
3. Add the child's view to your main view, along with any Auto Layout constraints.
4. Call **didMove(toParentViewController:)** on the child, passing in your main view controller.

```
addChildViewController(vc)

vc.view.frame = someFrame

view.addSubview(vc.view)
vc.didMove(toParentViewController: self)
```

```
func add(_ vc: UIViewController, frame: CGRect? = nil) {
  addChildViewController(vc)

  if let frame = frame {
    vc.view.frame = frame
  }

  view.addSubview(vc.view)
  vc.didMove(toParentViewController: self)
}
```

```
@nonobjc extension UIViewController {
  func add(_ vc: UIViewController, frame: CGRect? = nil) {
    addChildViewController(vc)

    if let frame = frame {
      vc.view.frame = frame
    }

    view.addSubview(vc.view)
    vc.didMove(toParentViewController: self)
  }
}
```

```
@nonobjc extension UIViewController {
  func remove() {
    willMove(toParentViewController: nil)
    view.removeFromSuperview()
    removeFromParentViewController()
  }
}
```

Peter Steinberger

@steipete

Surprises with Swift Extensions

Adds a closure for doing Auto Layout!

Views are small

Views are composeable

Views are reusable

View controllers are small

View controllers are composeable

View controllers are reusable

4

You need smaller view controllers!

NUKE IT FROM ORBIT
It’s the only way
to be sure

I need smaller view controllers
I need smaller view controllers
I need smaller view controllers
I need smaller view controllers
I need smaller view controllers
I need to make fresh coffee

Model

View

Controller

Networking

Business logic

Data storage

Delegation

Networking

Data storage

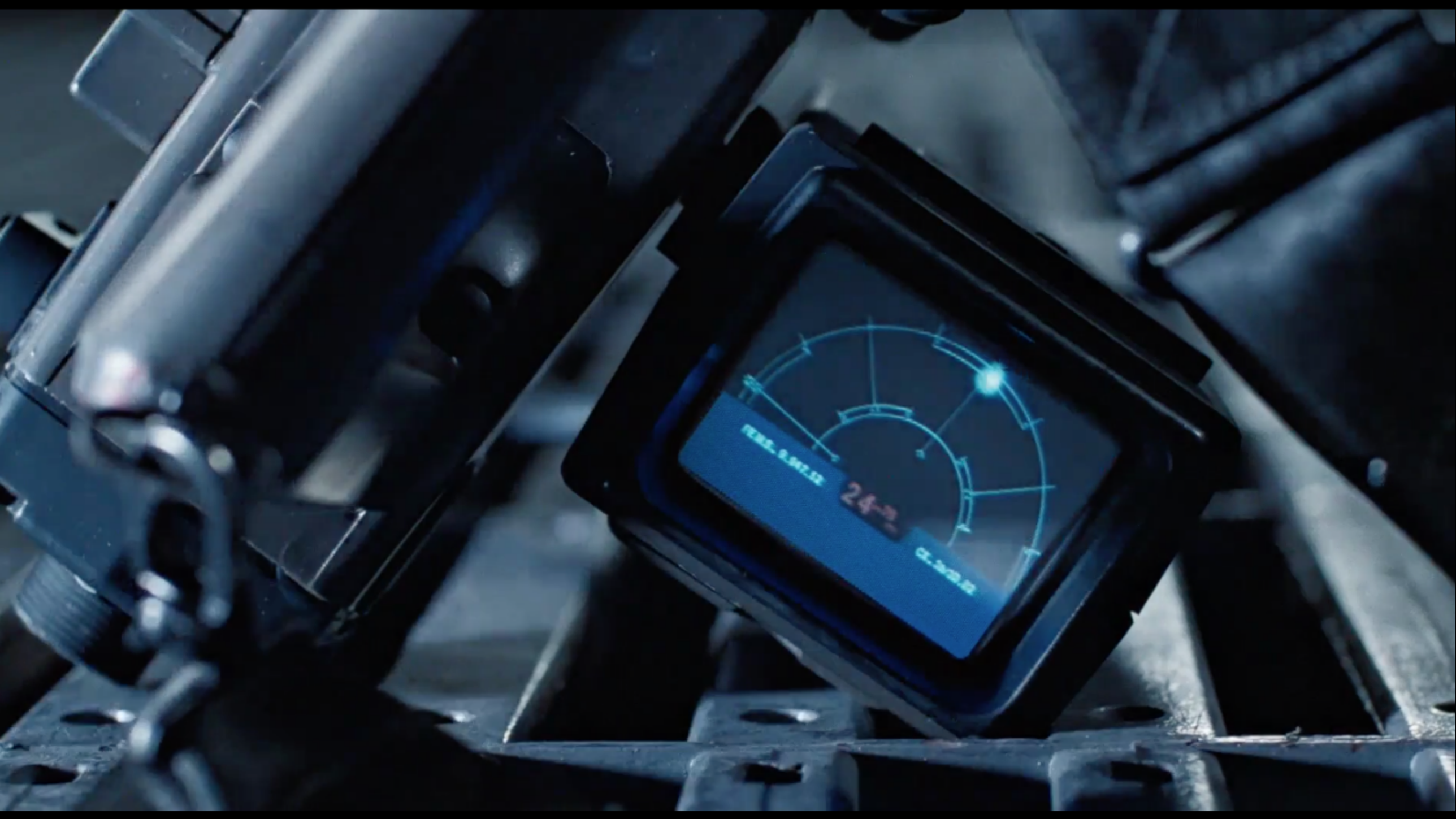
Model

View

Controller

Business logic

Delegation



APP DELEGATE

Networking

Data storage

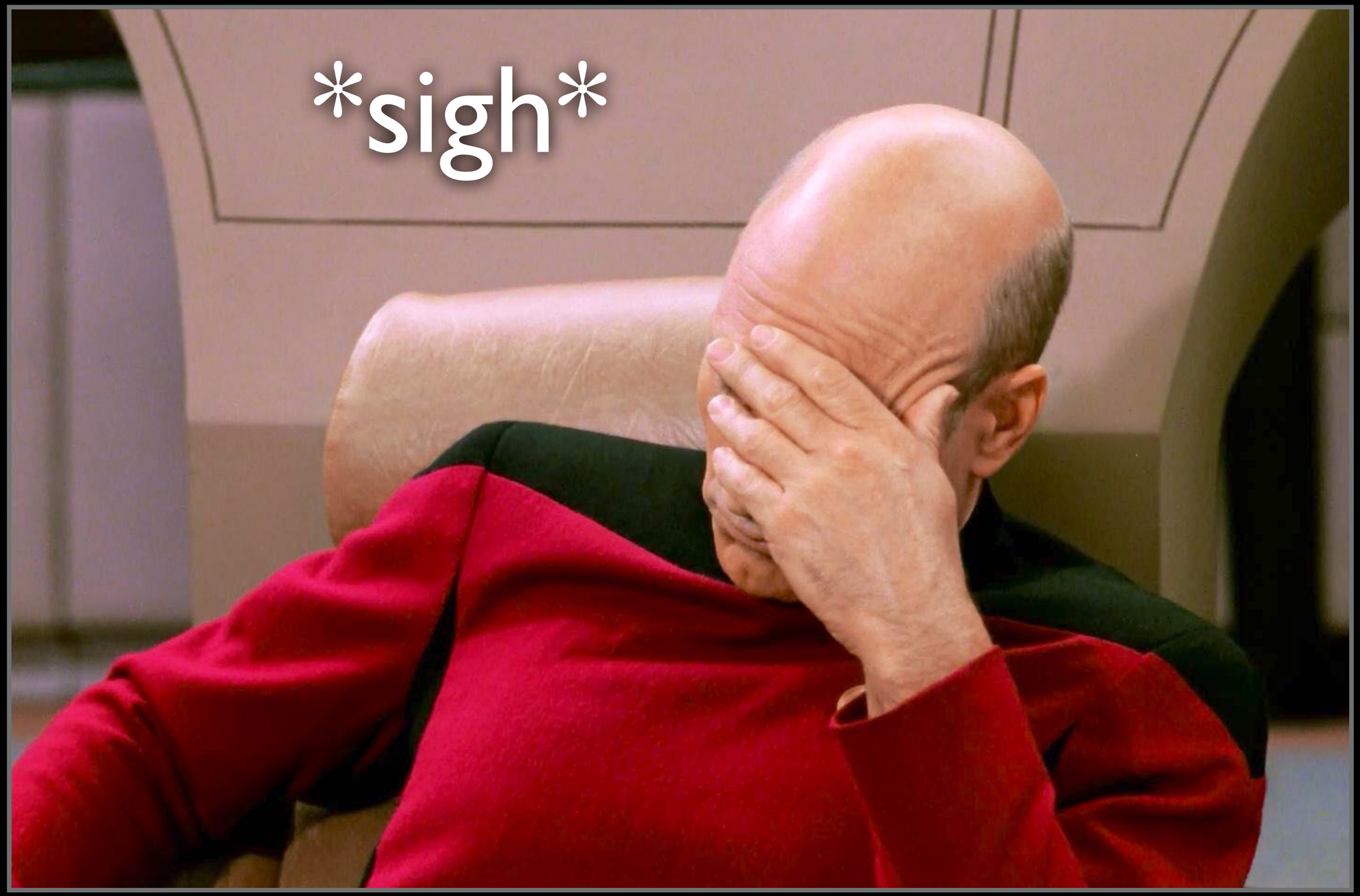
Model

View

Controller

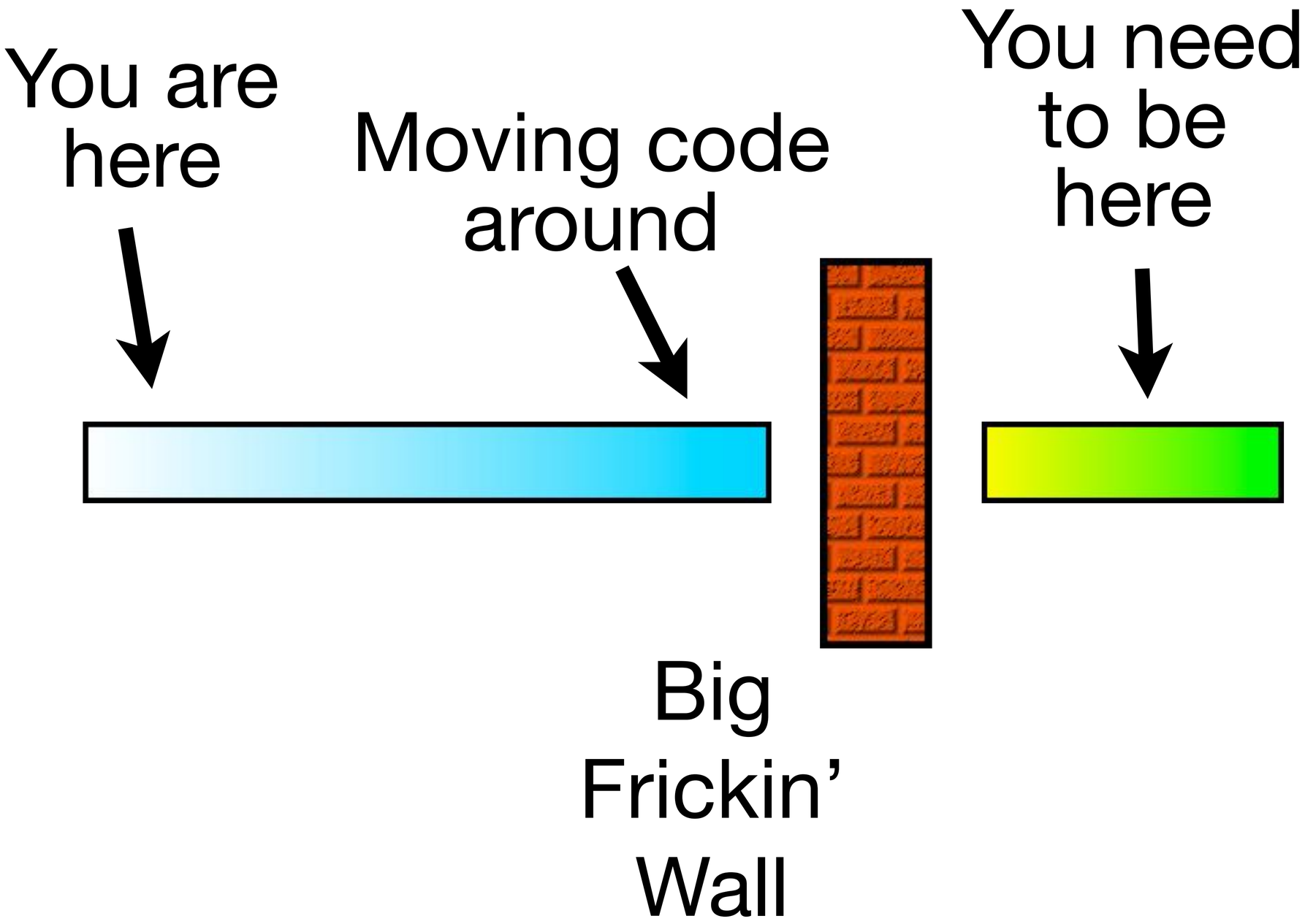
Business logic

Delegation


sigh

DO
NOT
WANT

WE
SHOULD
GET
KNEE
PADS


You are here
Moving code around
You need to be here
Big Frickin' Wall

UIView+Additions.swift

extension UIViewController { … }

Non-lifecycle code in AppDelegate

Non-lifecycle code in UIViewController

AppDelegate

didFinishLaunchingWithOptions

applicationDidEnterBackground

willContinueUserActivityWithType

performFetchWithCompletionHandler

applicationProtectedDataDidBecomeAvailable

AppDelegate

Handling launch and backgrounding

Application Continuity (“Handoff”)

Background fetching

Managing protected data

Opening from shared file or URL

UIViewController

viewDidLoad()

viewWillAppear()

viewWillDisappear()

traitCollectionDidChange()

didReceiveMemoryWarning()

applicationFinishedRestoringState()

UIViewController

Updating views with model data

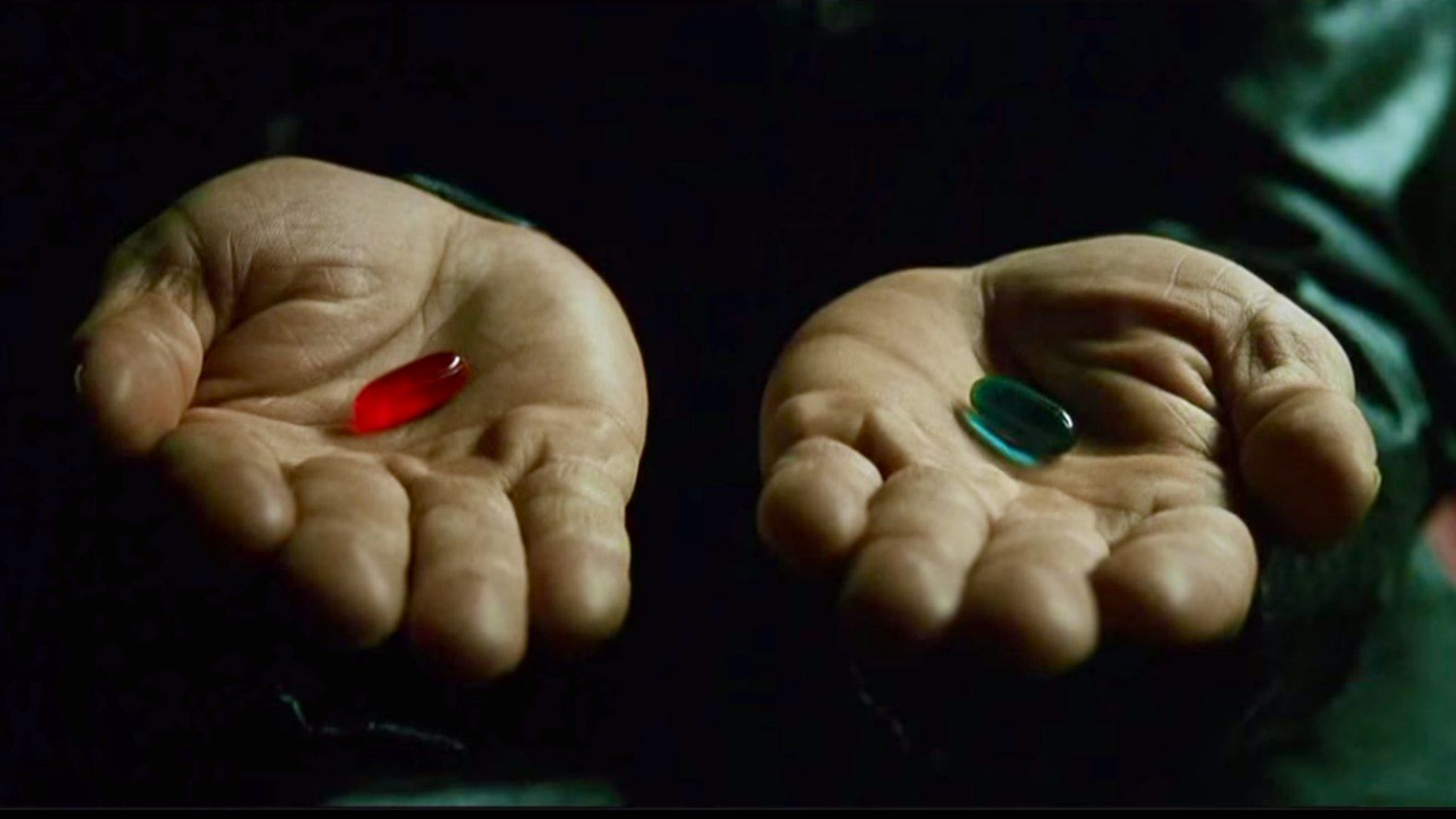
Responding to user input

Responding to UI changes

Saving and restoring UI state

FILING

FILING



UIViewController

Updating views with model data

Responding to user input

Responding to UI changes

Saving and restoring UI state

Data sources and delegates?

Networking?

Navigation?

UIViewController

Updating views with model data

Responding to user input

Responding to UI changes

Saving and restoring UI state

😀 Data sources and delegates

Networking?

Navigation?

UIViewController

Updating views with model data

Responding to user input

Responding to UI changes

Saving and restoring UI state

😀 Data sources and delegates

😷 Networking

Navigation?

UIViewController

Updating views with model data

Responding to user input

Responding to UI changes

Saving and restoring UI state

😀 Data sources and delegates

😷 Networking

Navigation?

UIViewController

Updating views with model data

Responding to user input

Responding to UI changes

Saving and restoring UI state

😀 Data sources and delegates

😷 Networking

😍 Navigation?

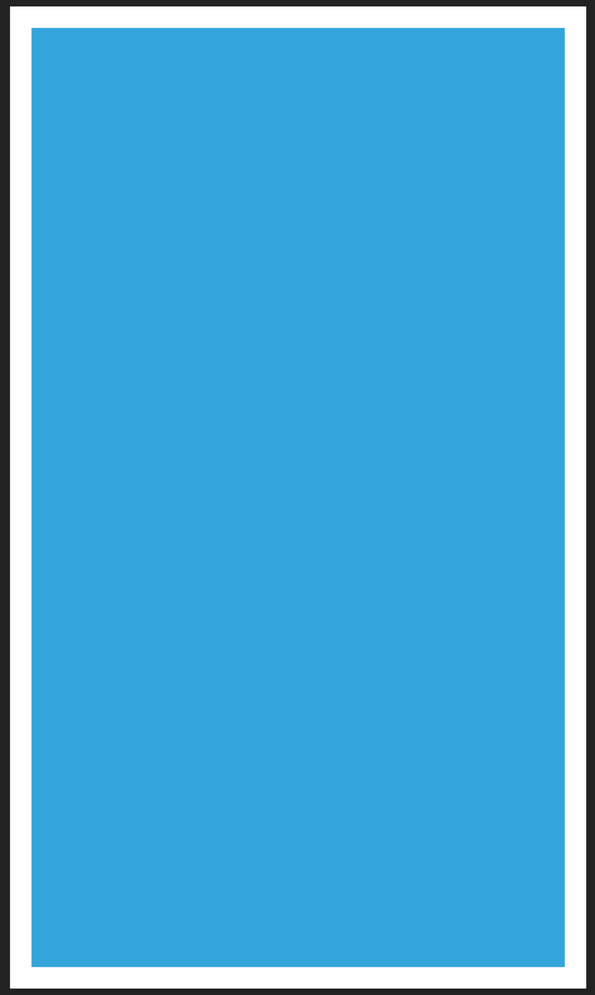
FLOW CONTROL

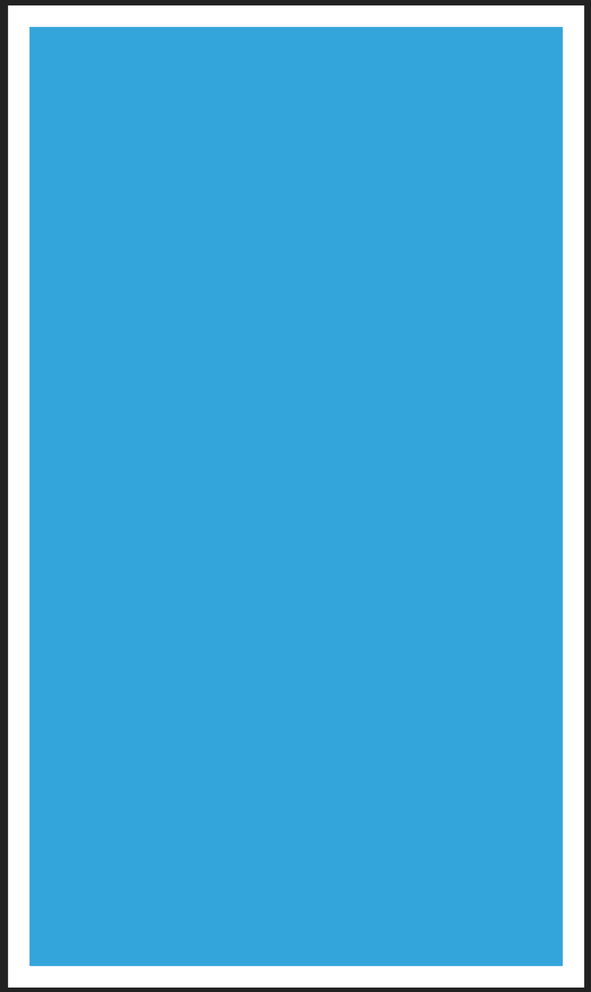
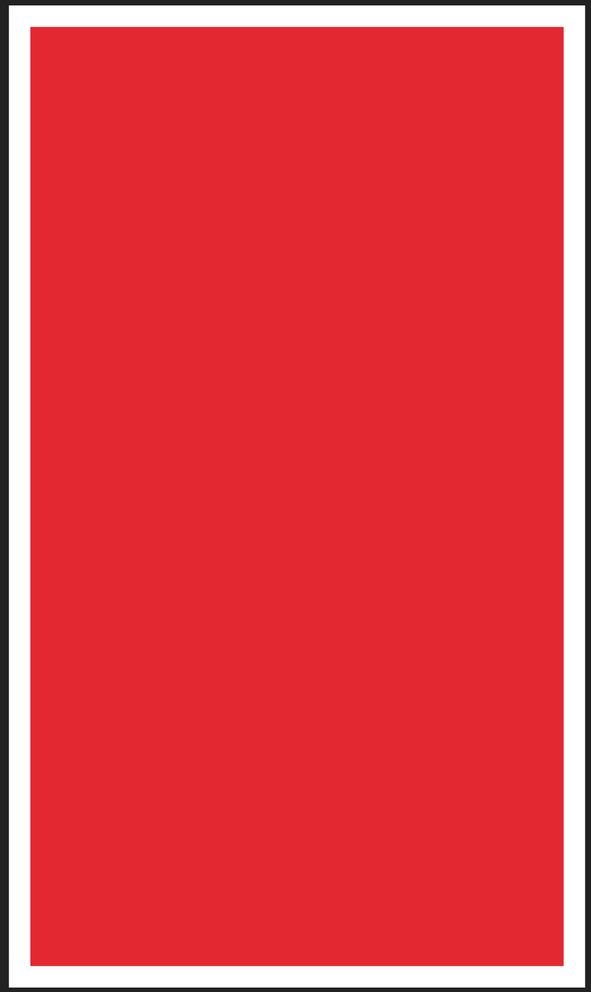
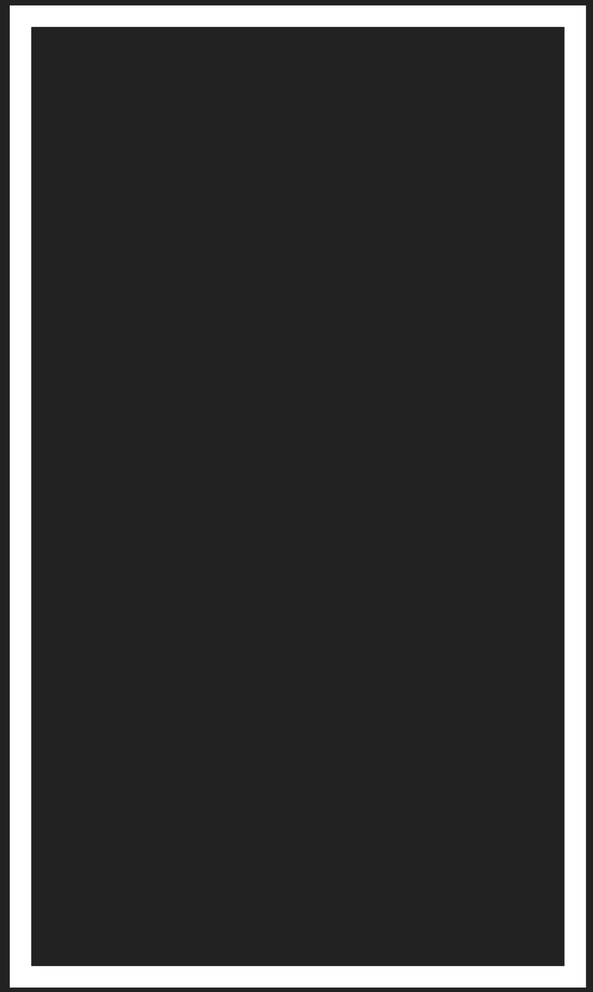
OperationQueue

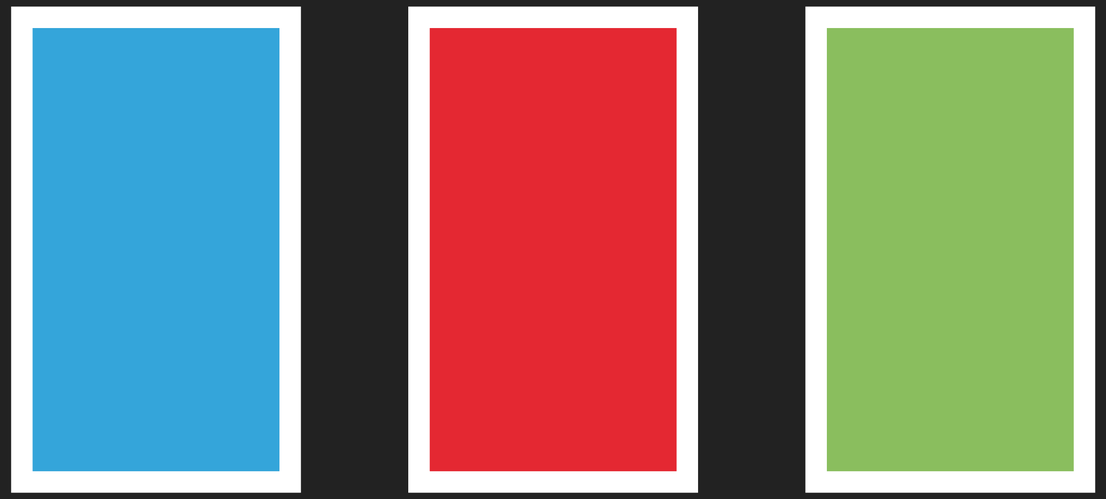
Host VC

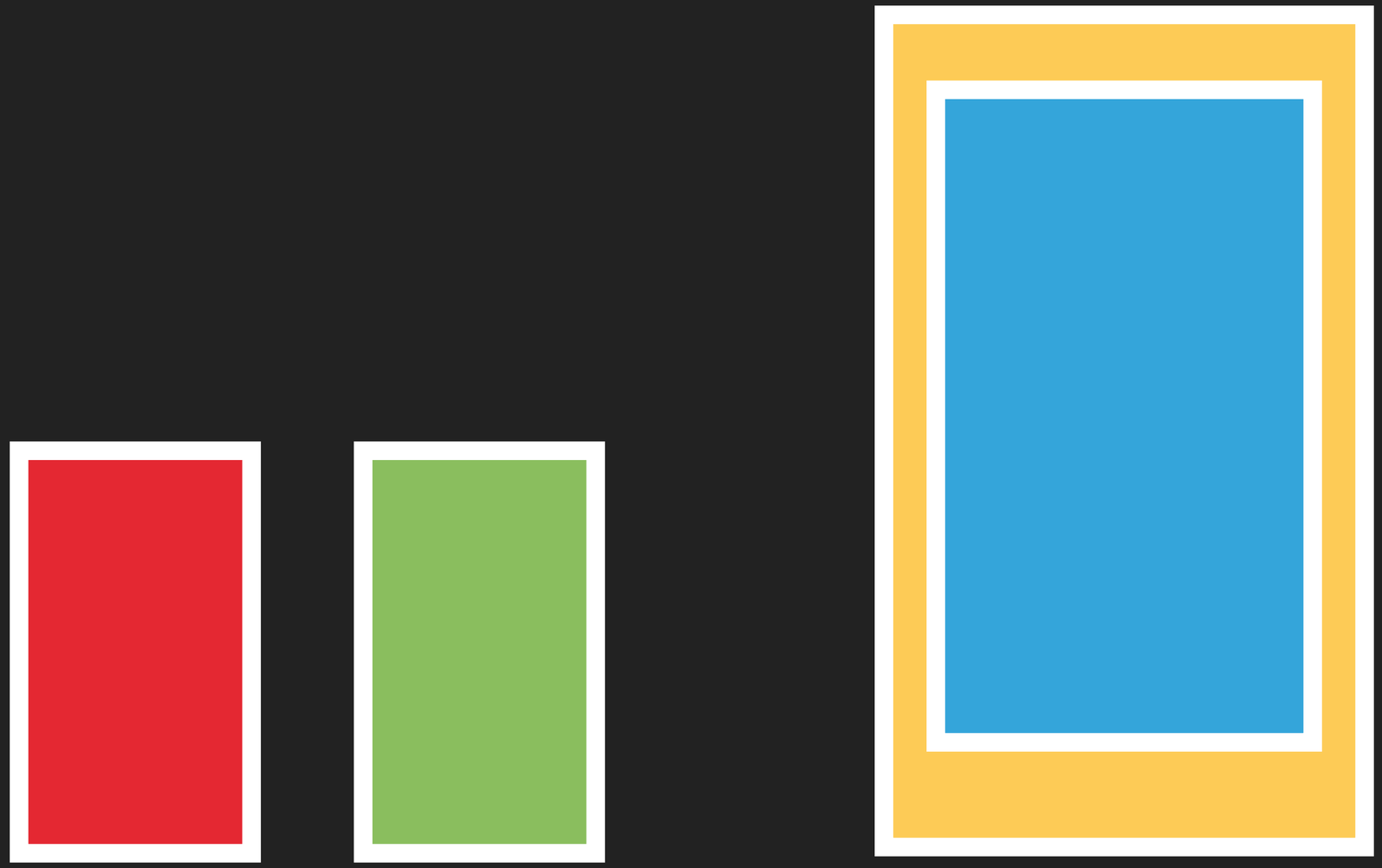
Coordinators

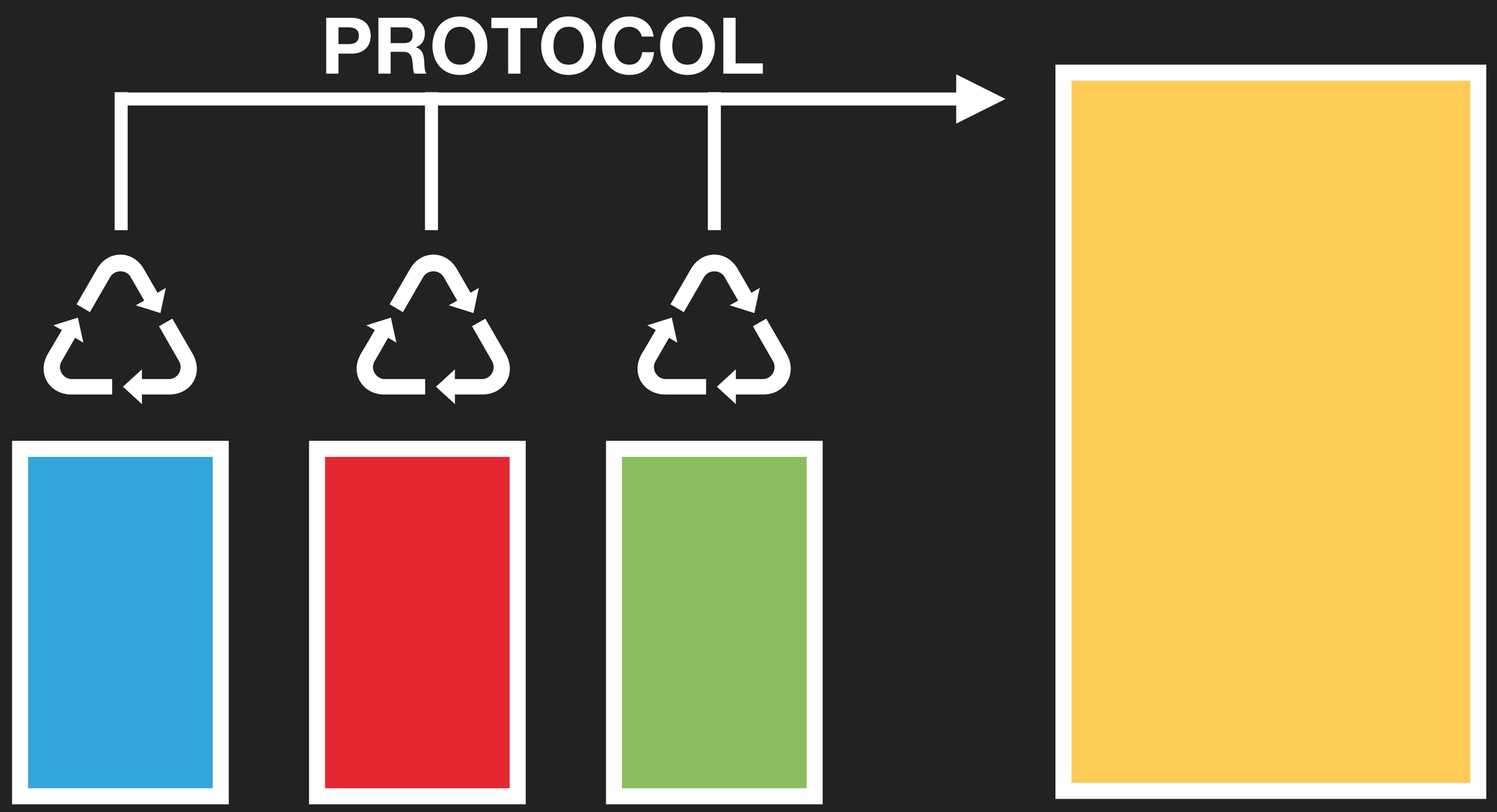
Host View Controllers

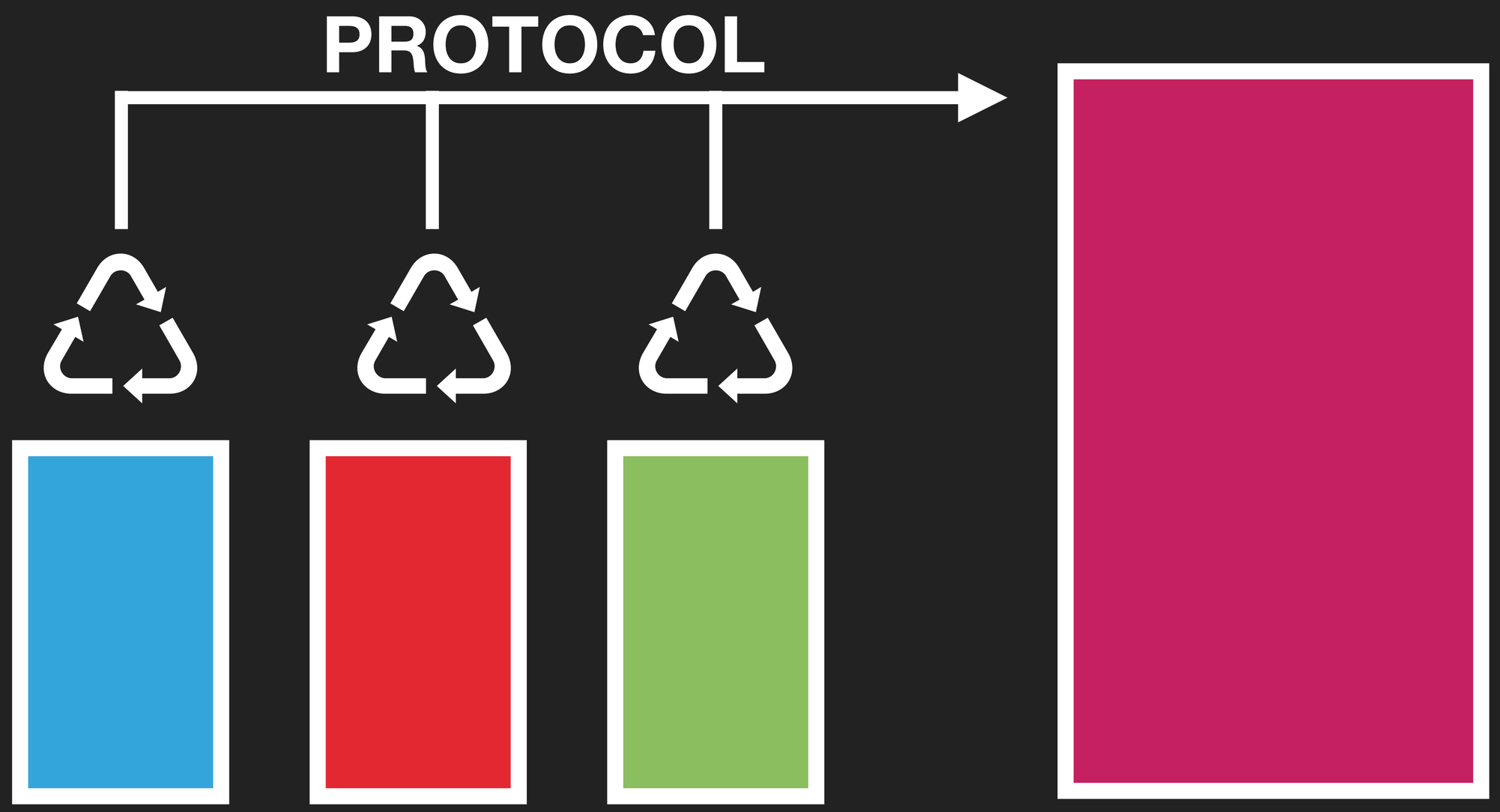







PROTOCOL


PROTOCOL

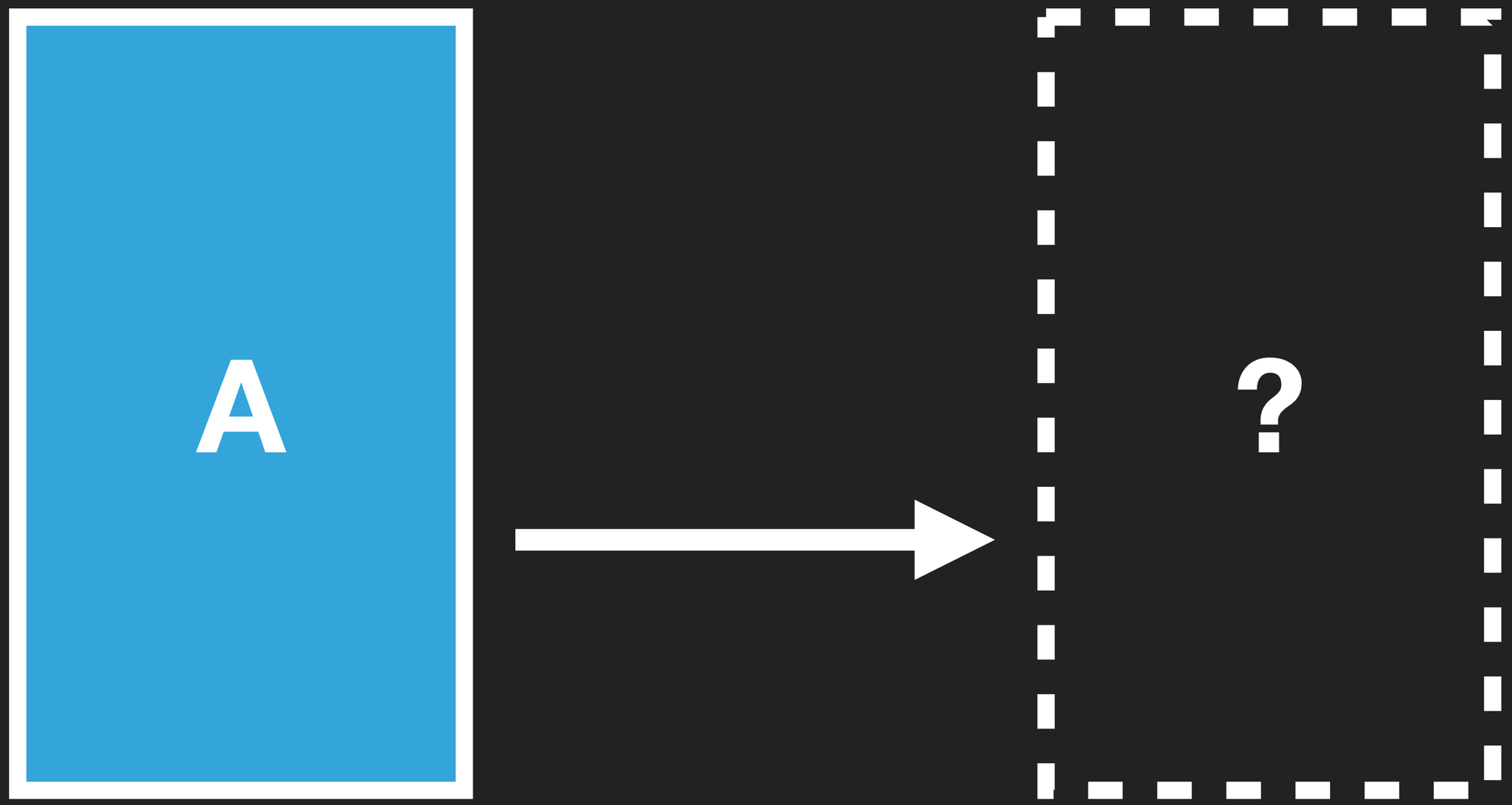
View controllers are small

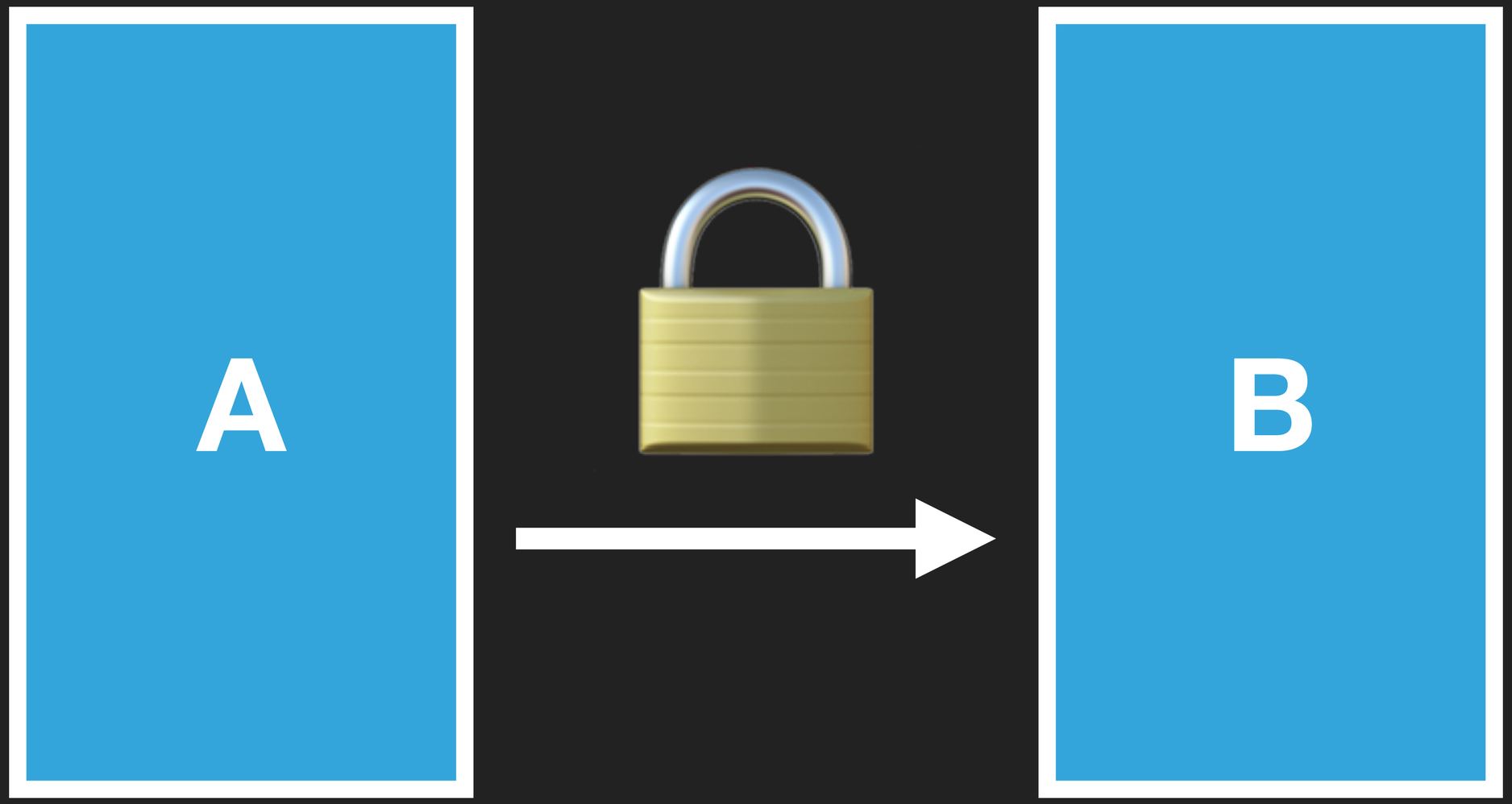
View controllers are composeable

View controllers are reusable

Coordinators

```
navigationController?.pushViewController(
    someVC, animated: true
)
```


A
?


A
B

Xcode File Edit View Find Navigate Editor Product Debug Source Control Window Help Tue 16:15

IfYouCan...mpressed 〉 iPhone 8 Plus | IfYouCanReadThisIAmImpressed | Build Succeeded | Today at 16:06

IfYouCanReadThisIAmImpressed 〉 IfYouCanRea...mImpressed 〉 Main.storyboard 〉 Main.storyboard (Base) 〉 No Selection

- IfYouCanRead...sIAmImpressed
  - IfYouCanRea...AmImpressed
    - AppDelegate.swift
    - MasterView...ntroller.swift
    - DetailViewController.swift
    - Main.storyboard
    - Assets.xcassets
    - LaunchScreen.storyboard
    - Info.plist
  - IfYouCanRea...pressedTests
    - IfYouCanRe...dTests.swift
    - Info.plist
  - IfYouCanRea...ressedUITests
    - IfYouCanRe...ITests.swift
    - Info.plist
  - Products

Split View Controller

ster View Controller

Navigation Controller

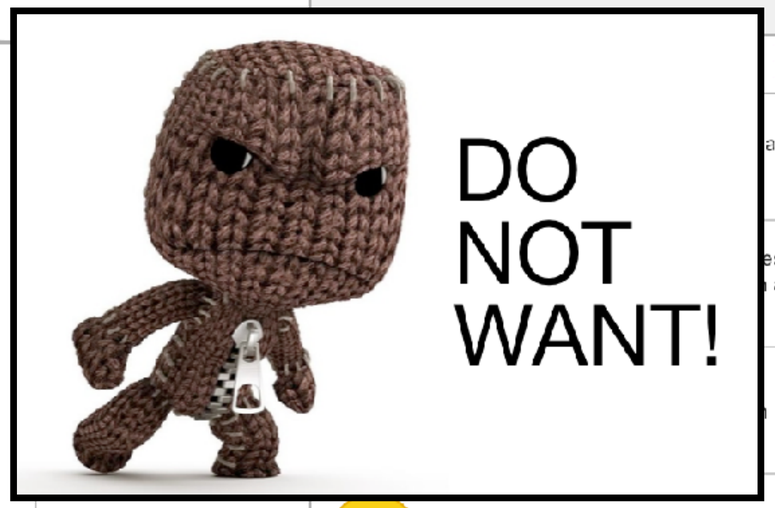
Table

Prototyp

No Selection

Navigation Controller

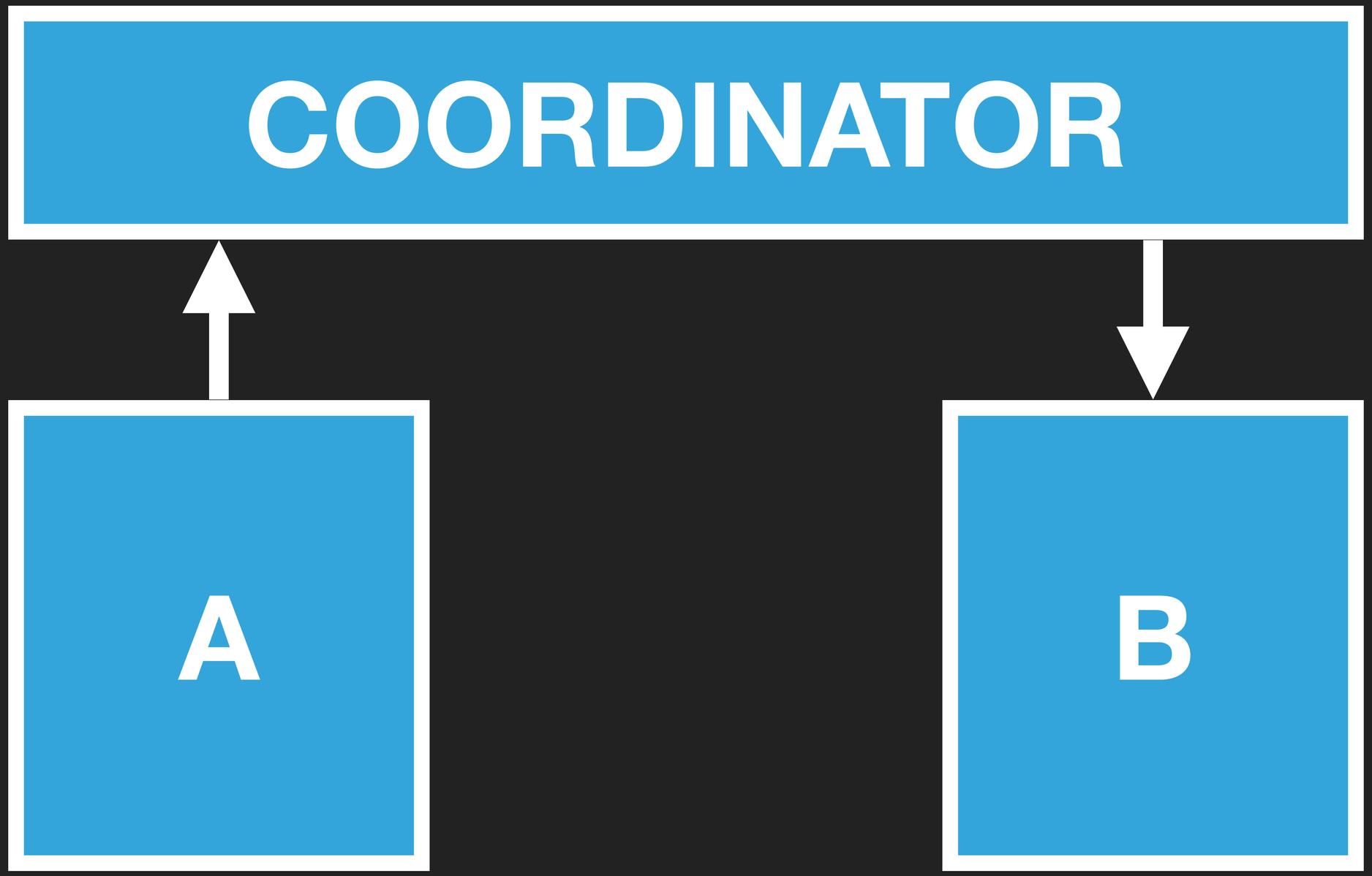
Master

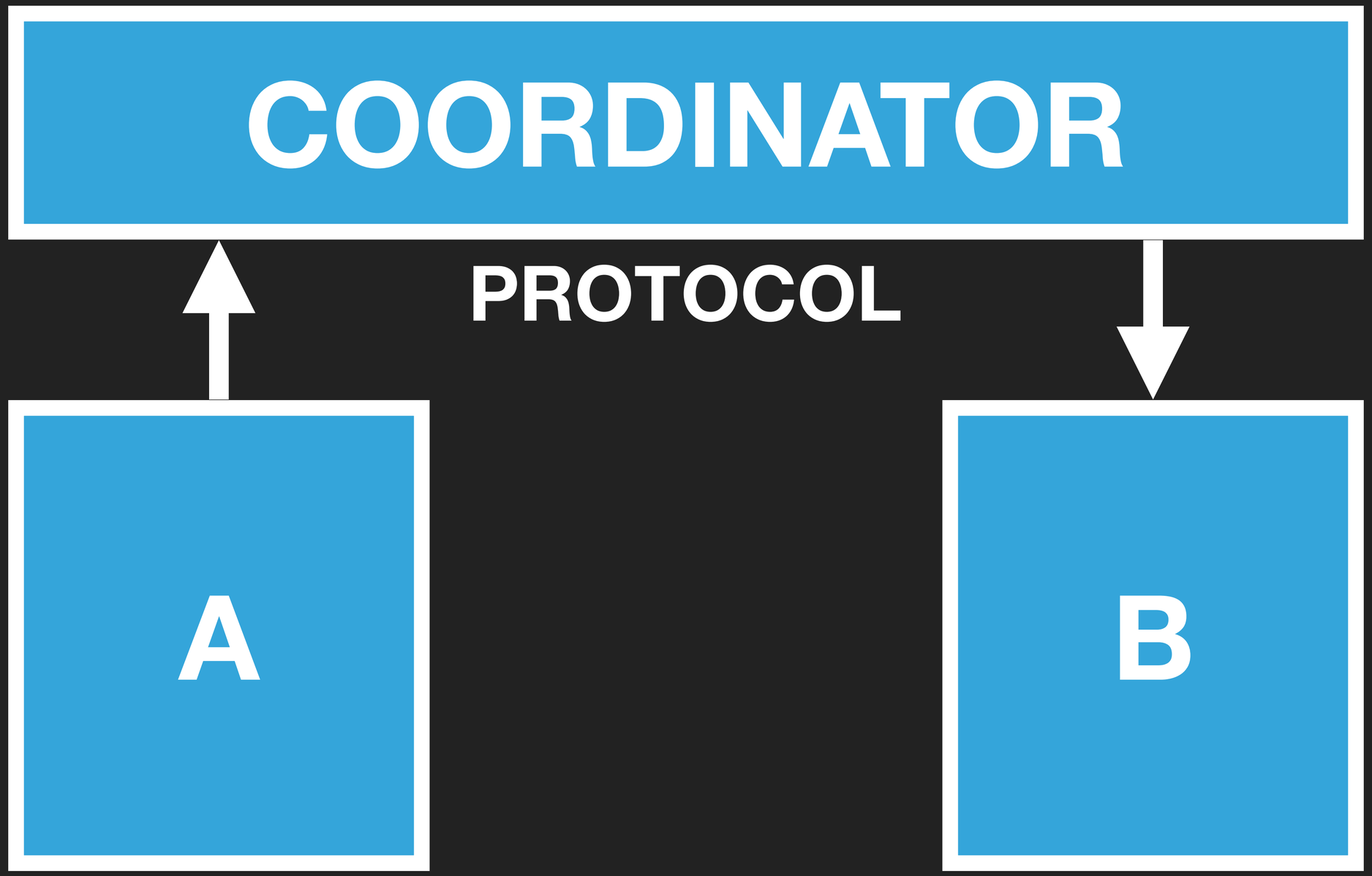


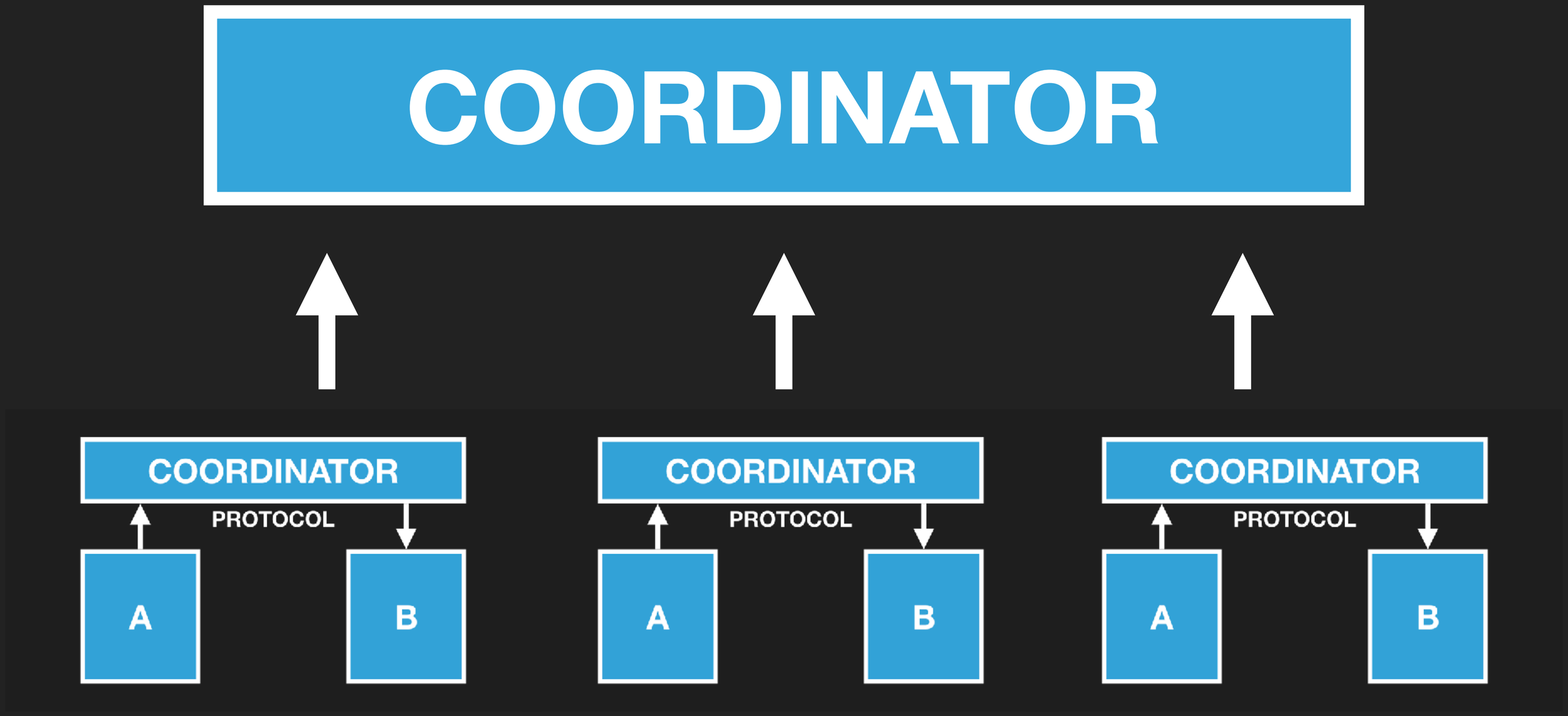
View as: iPhone 8 (wC hR) 75%

A

B


COORDINATOR
A
B


COORDINATOR
PROTOCOL
A
B


COORDINATOR
COORDINATOR
PROTOCOL
A
B
COORDINATOR
PROTOCOL
A
B
COORDINATOR
PROTOCOL
A
B

COUPLING IS THE ENEMY

OperationQueue

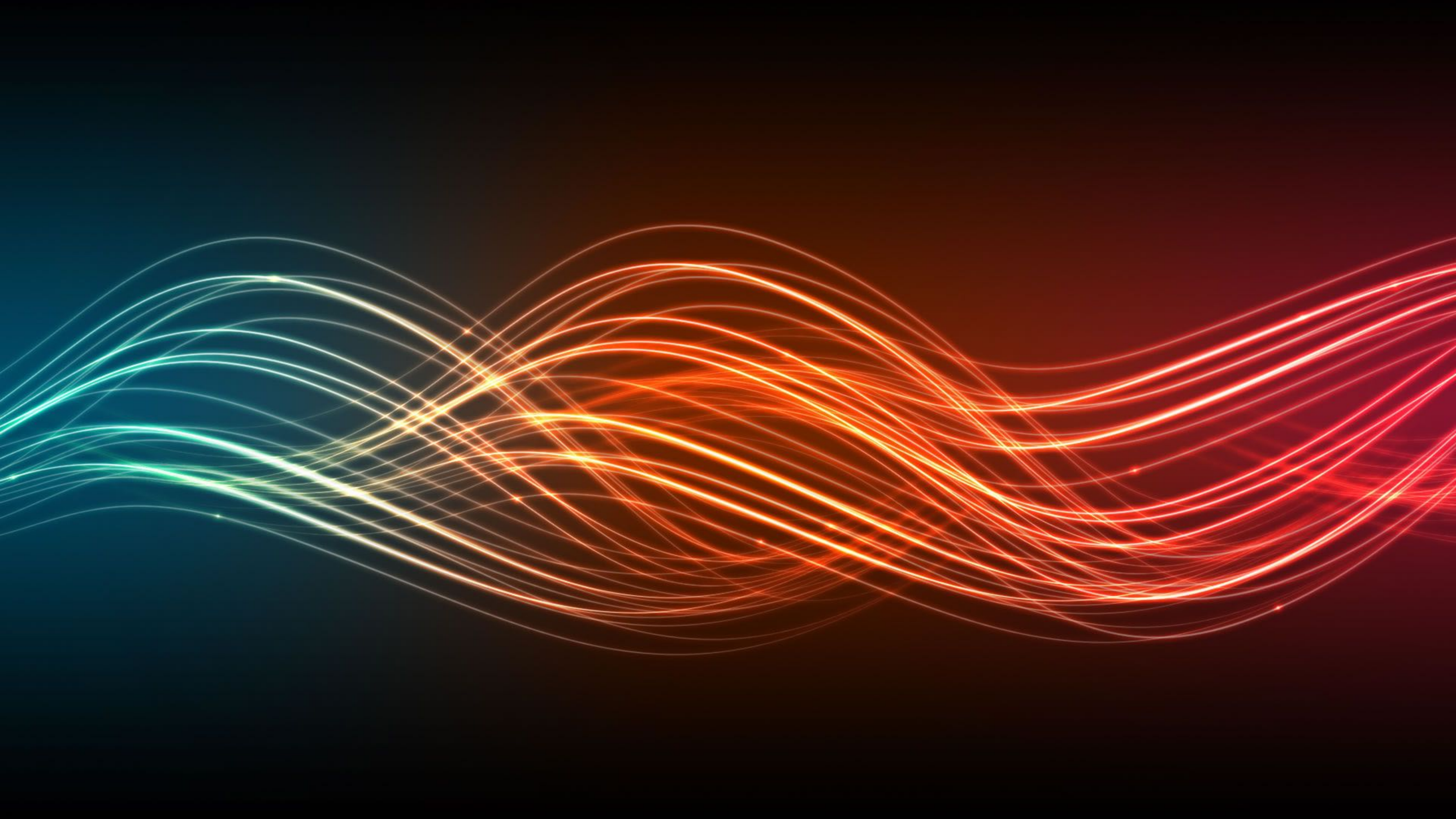
```
let queue = OperationQueue()
for headline in headlines {
    queue.addOperation {
        generateThumbnail(for: headline)
    }
}
queue.waitUntilAllOperationsAreFinished()
```

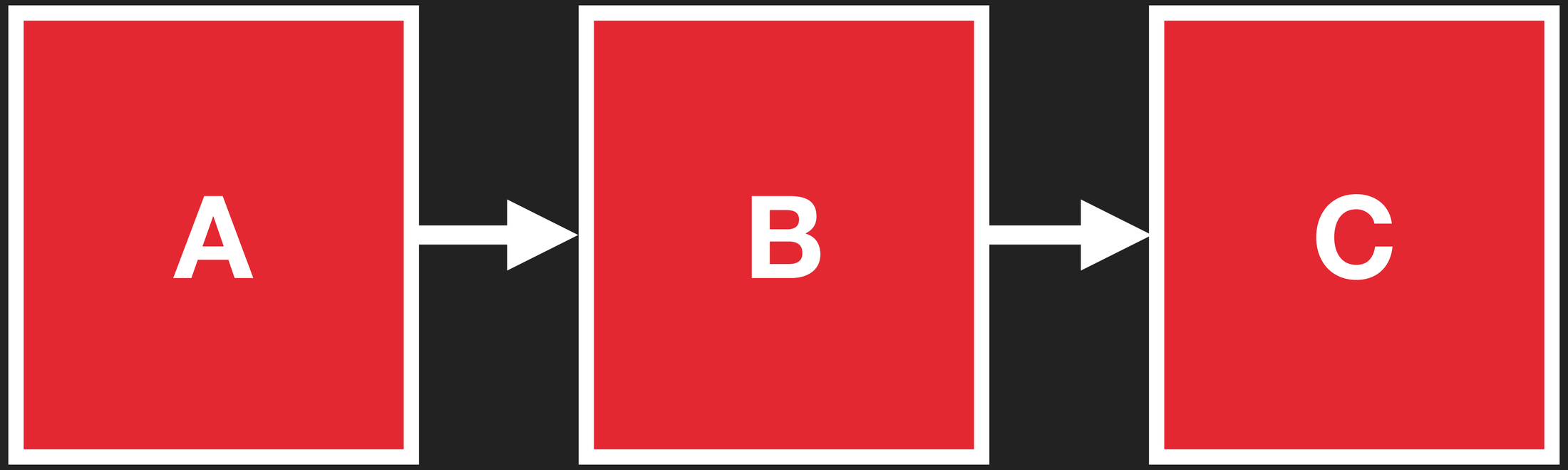
```
let queue = OperationQueue()
for headline in headlines {
    queue.addOperation {
        generateThumbnail(for: headline)
    }
}
queue.waitUntilAllOperationsAreFinished()
```

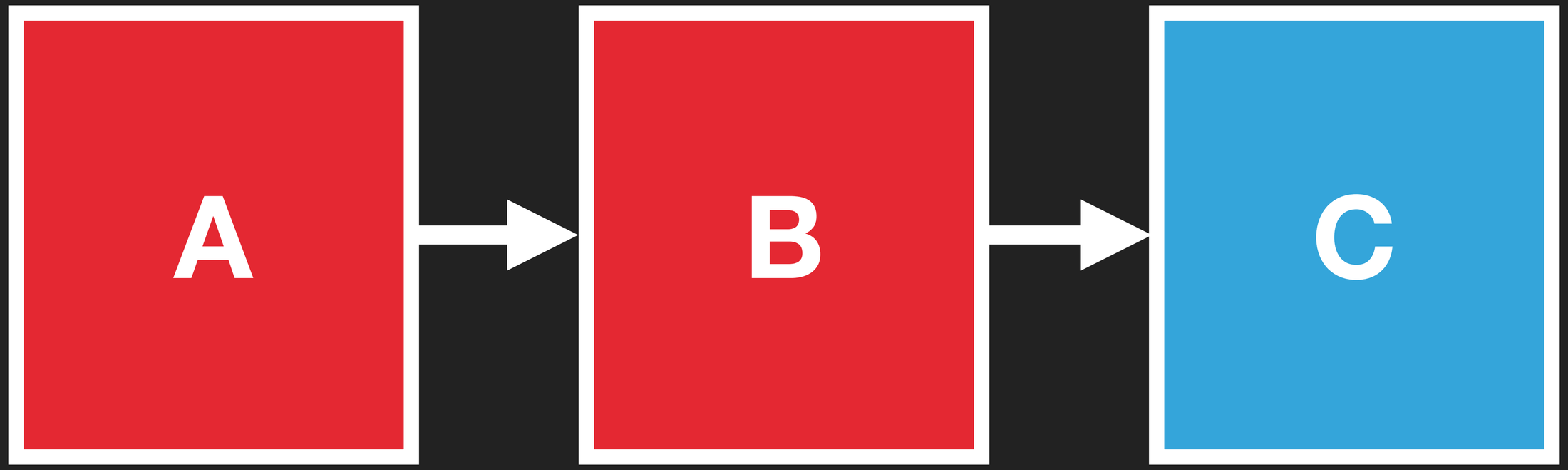
```
let queue = OperationQueue()
for headline in headlines {
    queue.addOperation {
        generateThumbnail(for: headline)
    }
}
queue.waitUntilAllOperationsAreFinished()
```

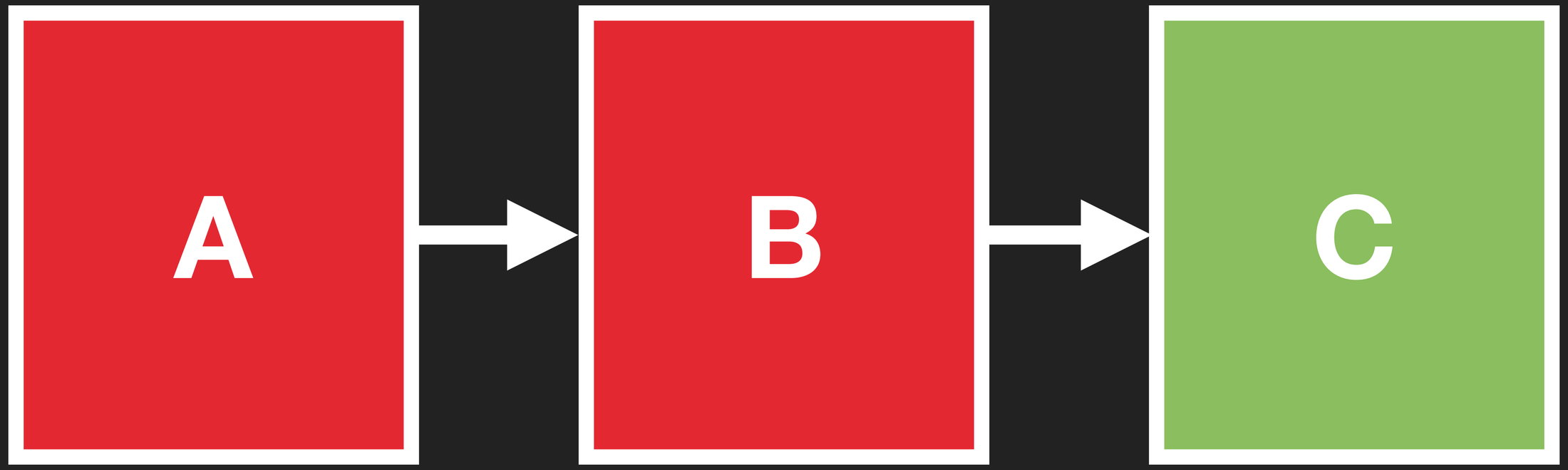
```
let queue = OperationQueue()
for headline in headlines {
    queue.addOperation {
        generateThumbnail(for: headline)
    }
}
queue.waitUntilAllOperationsAreFinished()
```

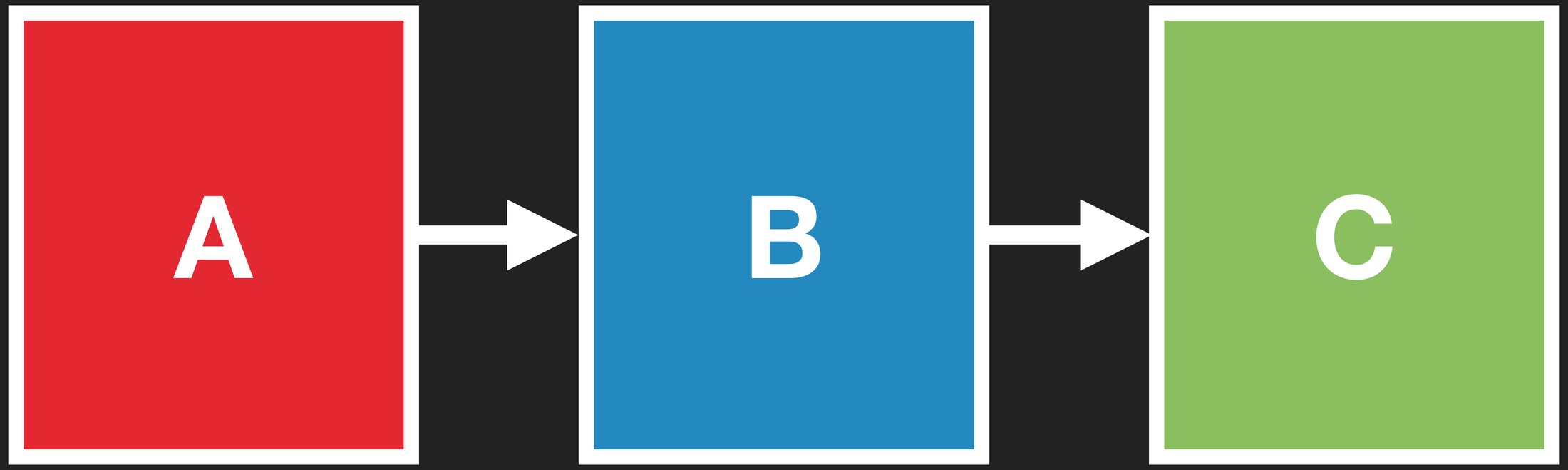
```
secondOperation.addDependency(first)
```

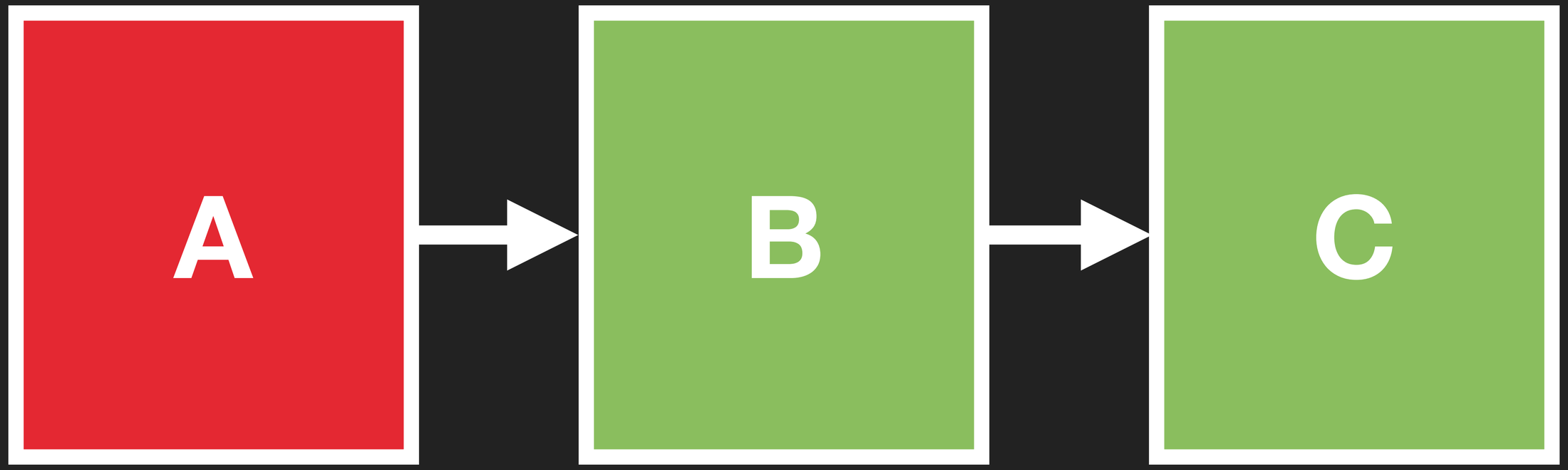


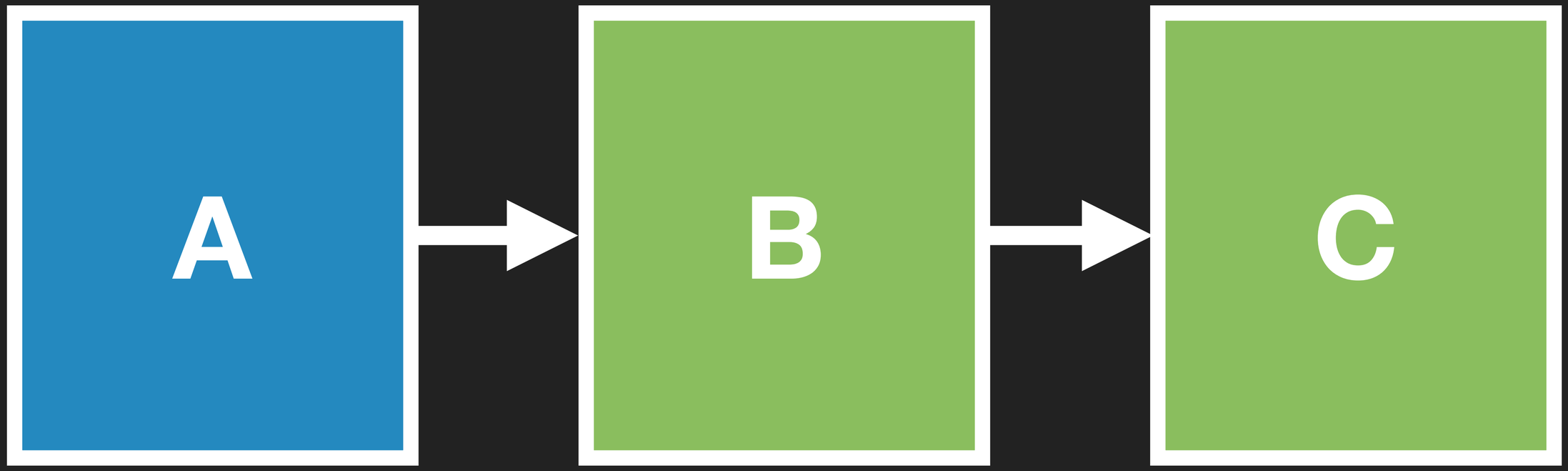

A
B
C

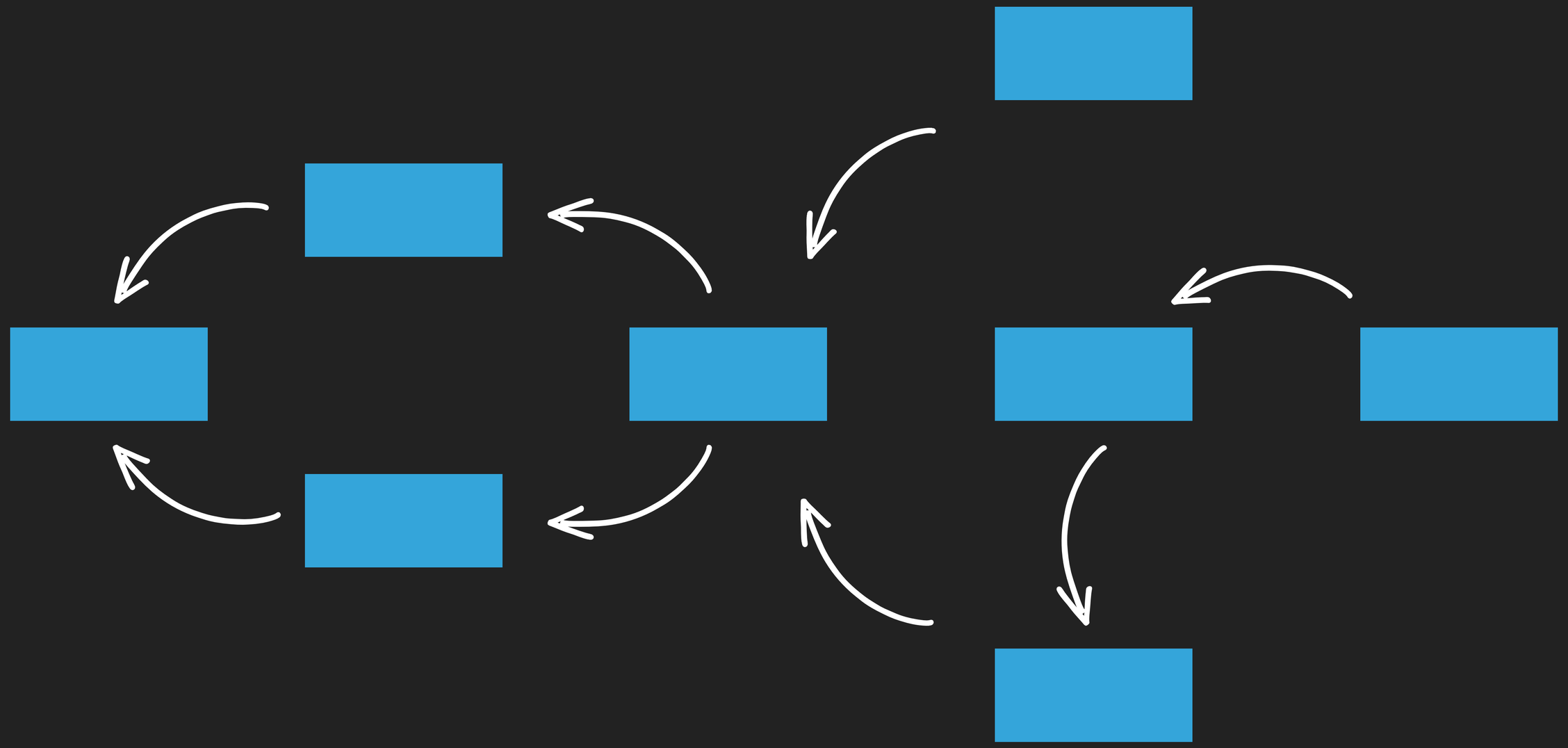

A
B
C


A
B
C


A
B
C


A
B
C


A
B
C



016
910
016

WTF

SUMMARY

WHAT'S THE PROBLEM?

- Massive view controller is real.
- We start off on the wrong foot thanks to UIViewController.
- VCs absorb many responsibilities.
- UI always makes testing harder.
  - Try to avoid **import UIKit**

WHAT ARE THE ALTERNATIVES?

- There are many – and that's OK.
- Choose for smart reasons: does it solve problems better than MVC?
- MVVM works great, but is "overkill" for simpler apps.
- MVC is usually the simplest.

HOW TO FIX MVC

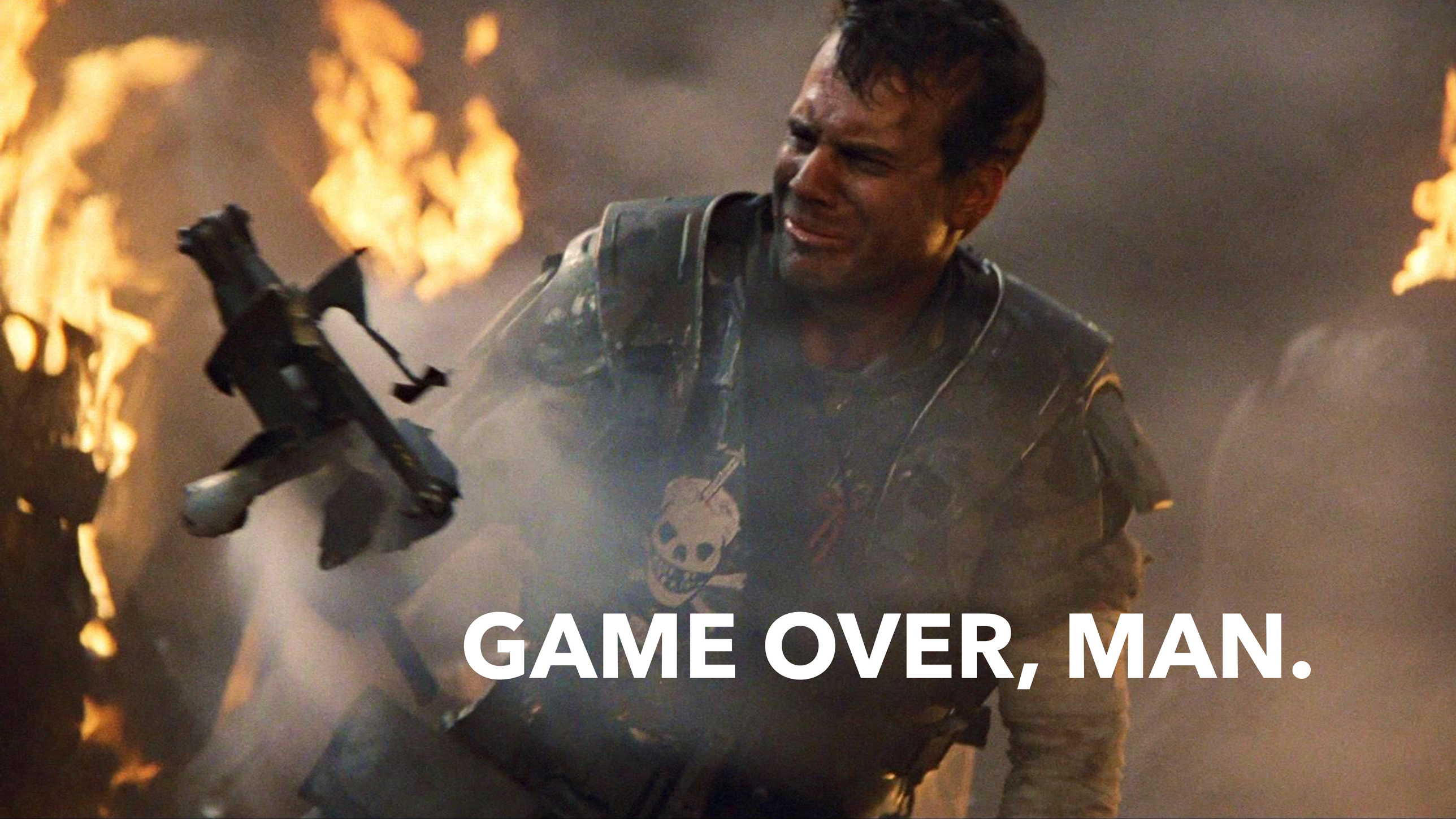
- Get data sources and delegates into separate controllers.
- Compose views into new views.
- Use view controller containment.
- Think where code should go.
- ~~Start a fight over networking code.~~

The Fresh Taste of Butter, the Healthier Way
I Can't Believe It's
MVC
Original
Health Facts
No Trans Fat*
70% Less Saturated Fat than Butter
A Blend of Oils Naturally Rich in Omega 3 ALA**
58% VEGETABLE OIL SPREAD
NET WT 15 OZ (425g)
No MVVM in sight

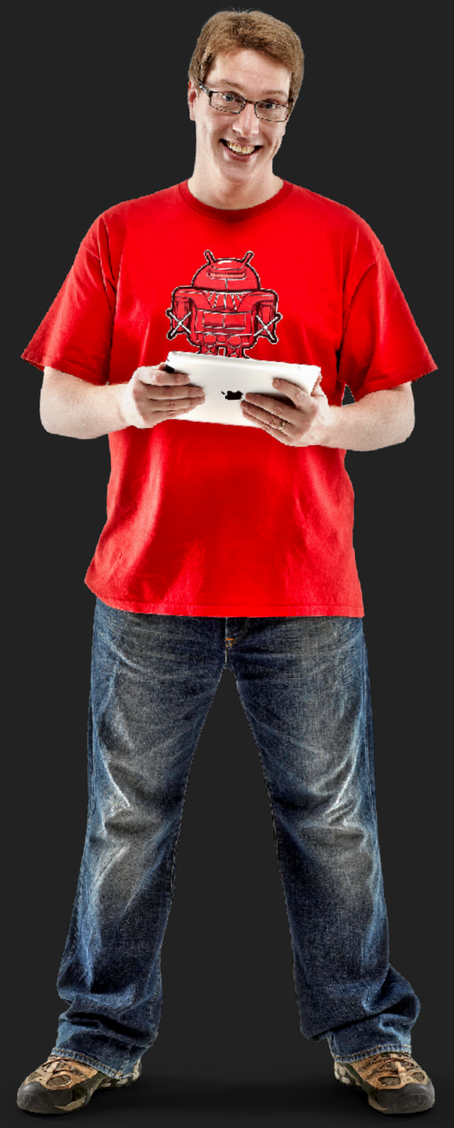
khanlou.com

davedelong.com

@twostraws


GAME OVER, MAN.

PAUL HUDSON – @twostraws

NUKE IT FROM ORBIT: How to Fix Massive View Controller Once and for All

https://www.hackingwithswift.com