おはよおう ございます

GOOD MORNING!

REAL WORLD MOCKING IN SWIFT

YOU WANT TO WRITE TESTS

BUT YOU DON'T WANT THEM TO MESS UP YOUR REAL STUFF

OR BE SLOW

```swift
let session = NSURLSession()
let url = NSURL(string: "http://www.tryswiftconf.com")!
let task = session.dataTaskWithURL(url) { (data, _, _) -> Void in
    if let data = data {
        let string = String(data: data, encoding: NSUTF8StringEncoding)
        print(string)
    }
}
task.resume()
```

OCMOCK CAN'T SAVE
YOU NOW

```
class HTTPClientTests: XCTestCase {
    var subject: HTTPClient!
    let session = MockURLSession()

    override func setUp() {
        super.setUp()
        subject = HTTPClient(session: session)
    }
```

```swift
class HTTPClientTests: XCTestCase {
    var subject: HTTPClient!
    let session = MockURLSession()

    override func setUp() {
        super.setUp()
        subject = HTTPClient(session: session)
    }

    func test_GET_RequestsTheURL() {
        let url = NSURL(string: "http://www.tryswiftconf.com")!

        subject.get(url) { (_, _) -> Void in }
```

```swift
class HTTPClientTests: XCTestCase {
    var subject: HTTPClient!
    let session = MockURLSession()

    override func setUp() {
        super.setUp()
        subject = HTTPClient(session: session)
    }

    func test_GET_RequestsTheURL() {
        let url = NSURL(string: "http://www.tryswiftconf.com")!

        subject.get(url) { (_, _) -> Void in }

        XCTAssert(session.lastURL === url)
    }
}
```

IT WILL TAKE A LOT OF

TIME

YOU WILL WONDER IF IT'S

WORTH IT

WHY USE MOCKS

» Make tests faster (like 1000s of times faster!) 🚄

» Increase coverage of test suite 🌍

» Make tests more robust 💪

TESTING

THE TIME

TO WRITE TESTS IS

NOW

MONACO
MONACO
MONACO
ÉTABLISSEMENTS
P. SOURIAU
ÉQUIPEMENT ÉLECTRIQUE
ACCESSOIRES
BOBINAGES
MAGNETOS, DYNAMOS
Essol
L'HUILE SUPÉRIEURE
VEEDOL
SHELL
Mobiloil
TOTAL

DEPENDENCY INJECTION

```
class VideoCaptureManager: NSObject {
    var userDefaults: UserDefaultsProtocol

    //MARK: - Initializers
    override init() {
        self.userDefaults = UserDefaults()
    }

    convenience init(userDefaults: UserDefaultsProtocol) {
        self.init()
        self.userDefaults = userDefaults
    }
}
```

“DEPENDENCY INJECTION MEANS GIVING AN OBJECT ITS INSTANCE VARIABLES. REALLY. THAT'S IT.”

James Shore

WHY NOT JUST USE A SINGLETON?

```
class VideoUploadManager {
    static let sharedInstance = VideoUploadManager()
}

class TimeMachineAPI {
    func uploadVideo(videoRecording: VideoRecording) {
        VideoUploadManager.sharedInstance.upload(videoRecording: self.videoRecording, completion: { (error) -> Void in
            if let error = error {
                Log.error(error.localizedDescription)
            }
        })
    }
}
```

WHY USE DEPENDENCY INJECTION

» Easy customization 🎨

» Clear ownership 🔐

» Testability 😍

CONSTRUCTOR INJECTION

TEST DOUBLES

TYPES OF TEST DOUBLES

» Stubs 😀

» Mocks 😕

» Partial mocks 😟

STUBS

“FAKES A RESPONSE TO METHOD CALLS OF AN OBJECT”

Unit Testing Tutorial: Mocking Objects

```
class StubTimeMachineAPI: TimeMachineAPI {
    var videoUrl = "https://www.youtube.com/watch?v=SQ8aRKG9660"

    func getVideoFor(year: Int) -> String {
        return videoUrl
    }
}
```

MOCKS

“LET YOU CHECK IF A METHOD CALL IS PERFORMED OR IF A PROPERTY IS SET”

Unit Testing Tutorial: Mocking Objects

```
class MockTimeMachine: TimeMachine {
    var timeTravelWasCalled = false

    mutating func timeTravelTo(year: Int) {
        timeTravelWasCalled = true
    }
}
```

PARTIAL MOCK

“ANY ACTUAL OBJECT WHICH HAS BEEN WRAPPED OR CHANGED”

Justin Searls

“TO PROVIDE ARTIFICIAL RESPONSES TO SOME METHODS BUT NOT OTHERS”

Justin Searls

PARTIAL MOCKS ARE AN
ANTI-PATTERN

WHAT'S WRONG WITH PARTIAL MOCKS?

» Challenging to set up 😫

» Decreases the comprehensibility of the test 😶

WHAT'S REAL?
WHAT'S FAKE?

MOCKING

MOCKING IN SWIFT VIA PROTOCOLS

“THIS KIND OF TESTING IS REALLY SIMILAR TO WHAT YOU GET WITH MOCKS, BUT IT’S SO MUCH BETTER.”

Protocol-Oriented Programming in Swift

“MOCKS ARE INHERENTLY FRAGILE.”

Protocol-Oriented Programming in Swift

“YOU HAVE TO COUPLE YOUR TESTING CODE”

Protocol-Oriented Programming in Swift

“TO THE IMPLEMENTATION DETAILS OF THE CODE UNDER TEST.”

Protocol-Oriented Programming in Swift

PLAYS WELL WITH STRUCTS AND CLASSES

PROTOCOLS HELP WHEN THERE'S INTERNAL DEPENDENCIES

THEY SAID DON'T MOCK TYPES YOU DON'T OWN

Mock Roles, not Objects

Steve Freeman, Nat Pryce, Tim Mackinnon, Joe Walnes
ThoughtWorks UK
Berkshire House, 168-173 High Holborn
London WC1V 7AA

{sfreeman, npryce, tmackinnon, jwalnes} @thoughtworks.com

WHY IT'S BAD TO MOCK TYPES YOU DON'T OWN

- Have to be sure that the behavior you implement in a mock matches the external library
- The external library could change, breaking your mock

MOCKING APPLE FRAMEWORK CLASSES

```
class UserDefaultsMock: UserDefaultsProtocol {

    private var objectData = [String : AnyObject]()

    func objectForKey(key: UserDefaultsKey) -> AnyObject? {
        return objectData[key.name]
    }

    func setObject(value: AnyObject?, forKey key: UserDefaultsKey) {
        objectData[key.name] = value
    }

    func removeObjectForKey(key: UserDefaultsKey) {
        objectData[key.name] = nil
    }

}
```

TIME TRAVELER TIMES

TEST THAT USES USERDEFAULTSMOCK

```
func testNotificationSettingsLoad() {
    let userDefaultsMock = UserDefaultsMock()
    mockUserDefaults.setObject("NEVER", forKey: "timeMachineBreakingNewsPushNotification")
    mockUserDefaults.setObject("NEVER", forKey: "timeMachineDailyDigestPushNotification")

    let dataProvider = PushNotificationsDataProvider(userDefaults: mockUserDefaults)
    let expectedFrequencies: [PushNotificationFrequency] = [.Never, .All, .All, .Never]

    XCTAssertEqual(expectedFrequencies, dataProvider.notificationSettings.values)

}
```

MOCKING NSNOTIFICATIONCENTER WAS NOT WORTH IT

» Complex 😕

» Injected throughout entire codebase 😰

» Limited functionality compared to real object 😬

» Solves a problem that doesn’t really exist 👻

THEY SAID DON'T MOCK VALUE TYPES

“WHEN YOU’RE DOING...SIMPLE DATA IN AND DATA OUT, YOU DON’T NEED MOCKING OR STUBBING.”

Andy Matuschak

“YOU CAN PASS A VALUE INTO A FUNCTION, THEN LOOK AT THE RESULTING VALUE.”

Andy Matuschak

“TRY MAKING YOUR REFERENCE TYPES IMMUTABLE”

Joe Groff

WHAT MAKES A GOOD MOCK

- Quick and easy to write
- Relatively short and does not contain tons of information you don’t need
- Legitimate reason to not use real object

I AM THOUGHTFUL ABOUT WHAT GETS MOCKED AND WHY

FELLOW TIME TRAVELERS

LET'S LOOK TOWARDS THE FUTURE

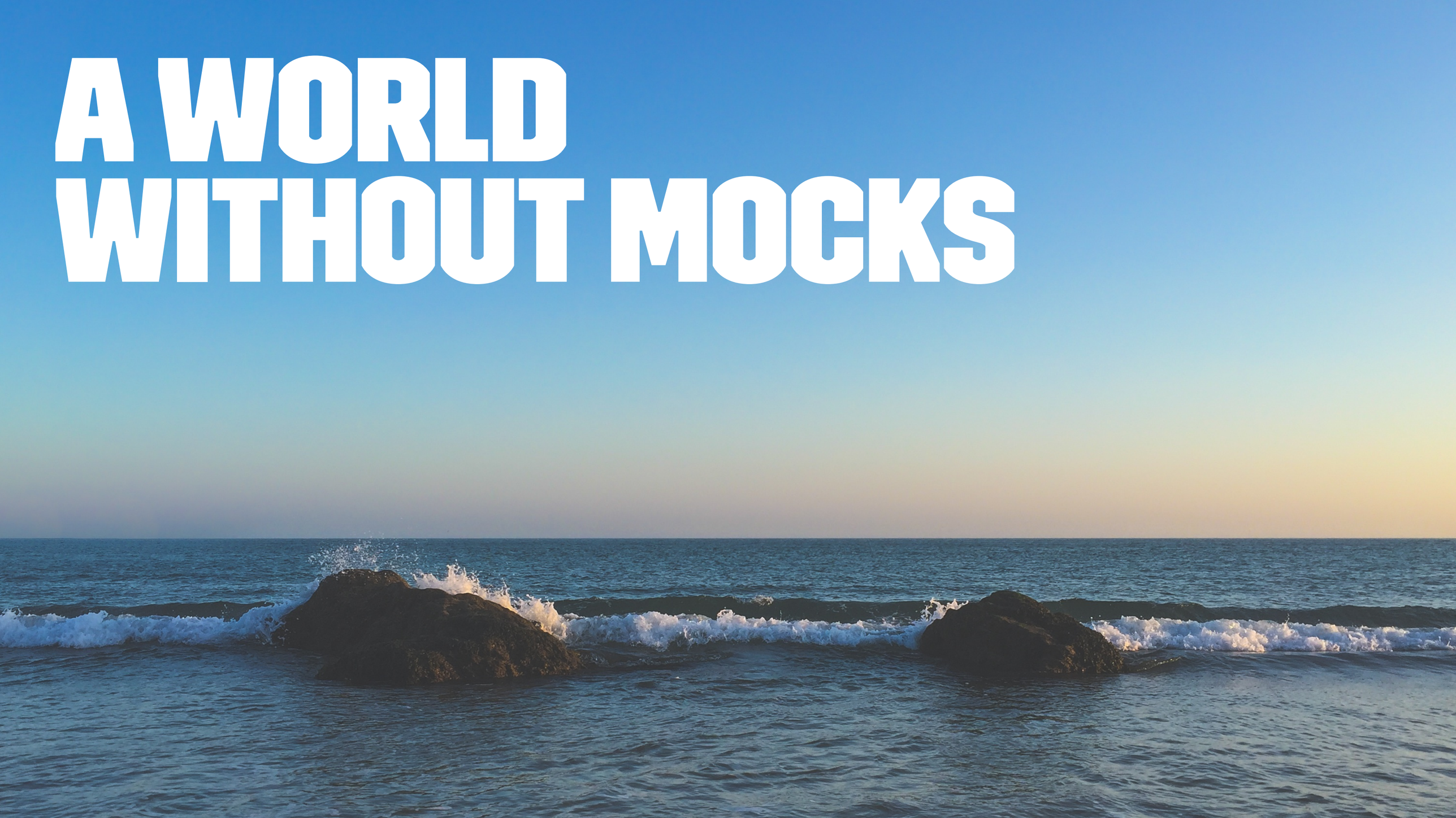
SWIFT

MOCKING FRAMEWORKS

DOBBY
MOCKFIVE
SWIFTMOCK

CUCKOO

A WORLD WITHOUT MOCKS



WRAP UP

- 💪 If you want to improve your codebase, write tests
- 💪 Mocks allow you to write great tests
- 💪 Create simple mocks that will last

WRITE MOCKS AND USE THEM FOR GREAT GOOD