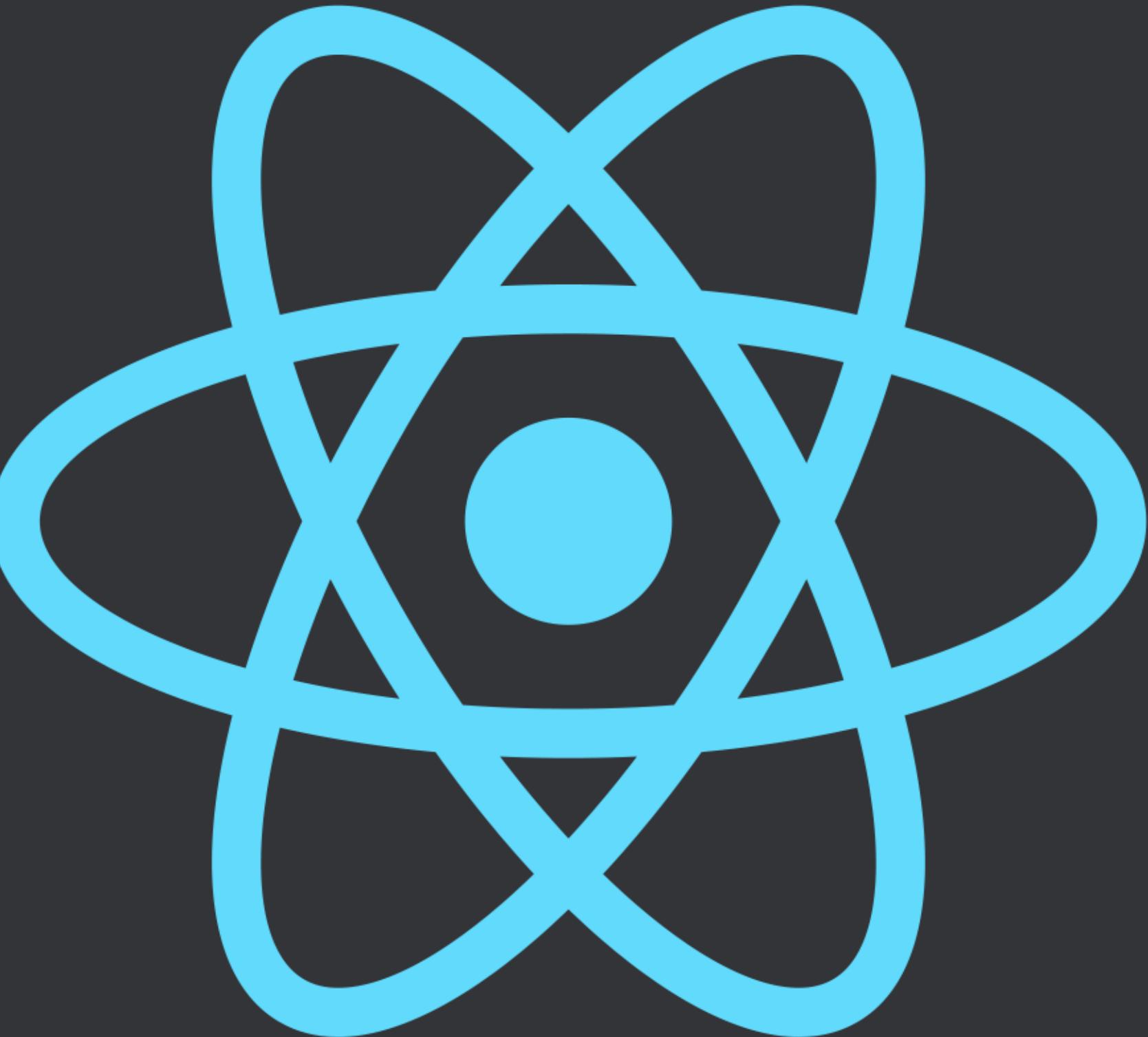


# React Native

**LEARN ONCE, WRITE ANYWHERE:  
BUILD MOBILE APPS WITH REACT**

# What is React Native?

- Native mobile application development
  - iOS (App Store deployable)
  - Android (Google Play Store deployable)
  - Not quite bare metal, but compiles to native code
- Everything is JavaScript (ES6)
- Something that kind of looks like CSS<sup>1</sup>



---

<sup>1</sup> If you love camelCase instead of kebab-case.

# Frequently Asked Questions

1. Is this really just a webview?

- No.

2. Do I have to live in Xcode/Android Studio?

- No, only to create deployable versions of the app.

3. Is this just a box of magic?

- Yes and No.

# Getting Started

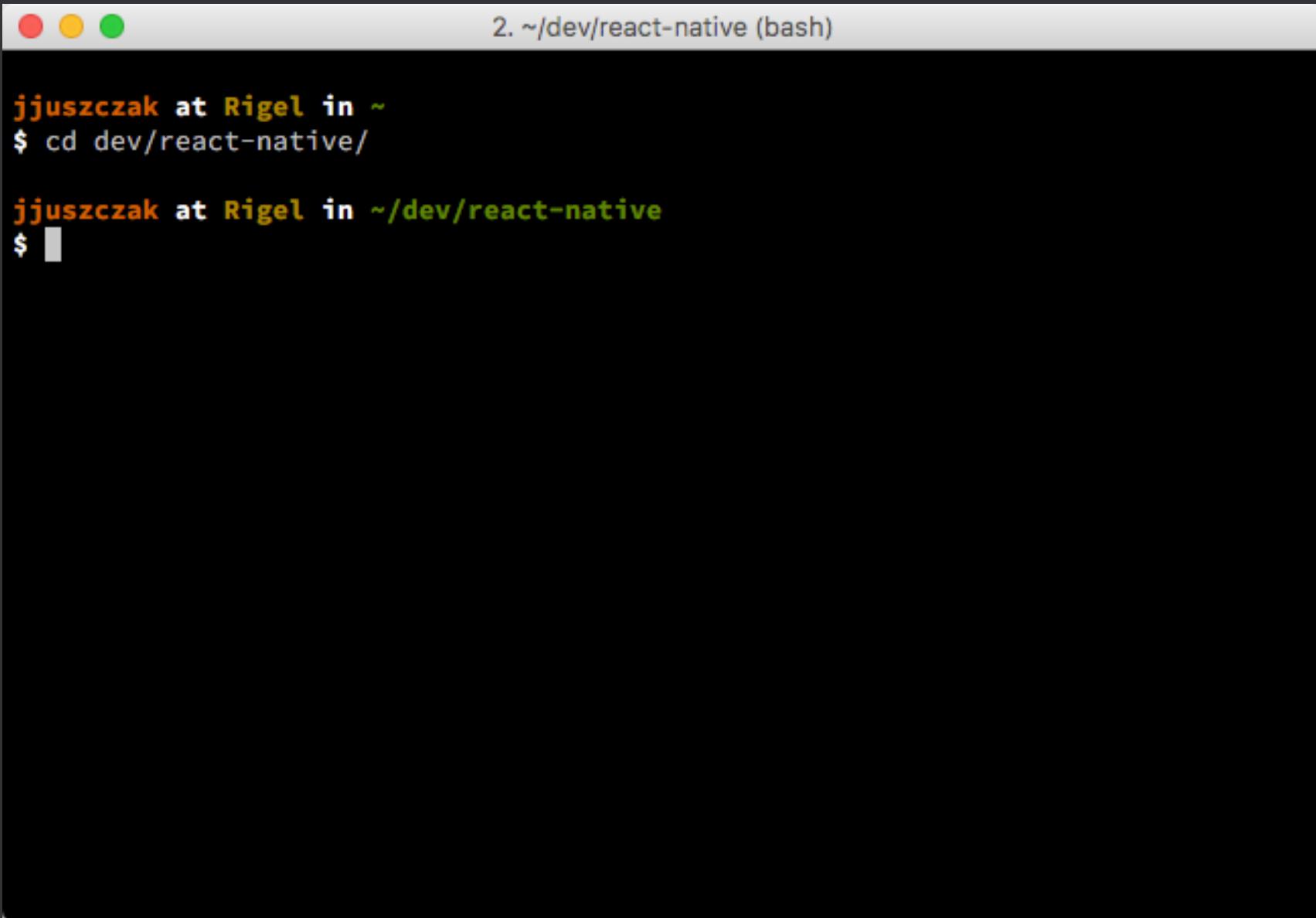
- MacOS (iOS and Android) or Windows (Android only)
- Xcode and/or Android Studio

MacOS with Xcode:

```
brew install node  
brew install watchman  
npm install -g react-native-cli
```

```
react-native init AwesomeProject  
cd AwesomeProject  
react-native run-ios
```

# Make it happen



```
jjuszczak at Rigel in ~
$ cd dev/react-native/

jjuszczak at Rigel in ~/dev/react-native
$ █
```

# What's in the box?

Hey, we all (*mostly*) understand JavaScript

```
!["React" isEqualToString:@"Objective-C"]
```

```
!"React".equals("Java")
```

```
"React" === "JavaScript";
```

# What's in the box?

## Tiny Library & API

- 34 Component Types (like `<Text>`, `<Button>`, and `<Image>`)
- 38 API methods (like `Alert`, `Clipboard`, and `Geolocation`)

# What's in the box?

## Interwebs

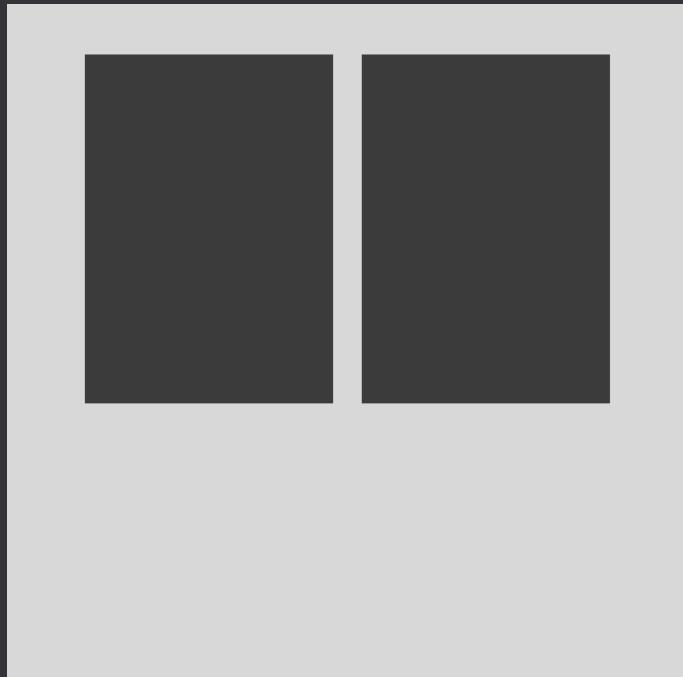
- Designed from the ground up for pulling data from APIs
- Gracefully deal with asynchronous request/response cycle

```
axios.get('https://rallycoding.herokuapp.com/api/music_albums')
  .then(response => this.setState({ albums: response.data }));
```

# What's in the box?

## Flexbox

2 Items



3 Items



4 Items



5 Items



# What's in the box?

**Redux**

...because Redux



# Advantages?

## React Native vs. Native Native

- "It's hard"
- Good luck building two (iOS, Android) versions of everything
- Just getting the boiler plate up is a PITA

# Advantages?

## React Native vs. Xamarin

- C#
- Steep learning curve
- VisualStudio (yeah, I know)

# Advantages?

## React Native vs. Cordova (PhoneGap)

- Not native (WebView all the things)
- Not very performant (because of above)
- No background processing

# Advantages?

## React Native vs. Ruby Motion

- Proprietary (not open source)
- Mostly just syntax wrappers, you still need to know what's going on under the hood
- Not a yuge community; might be hard to get help

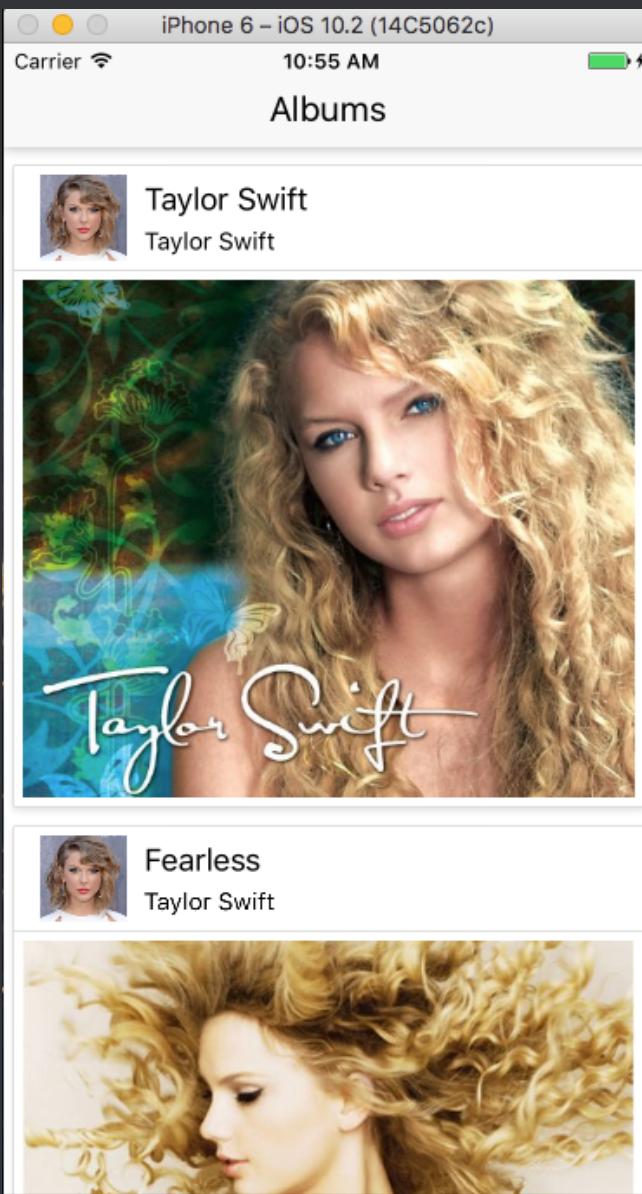
# Disadvantages?

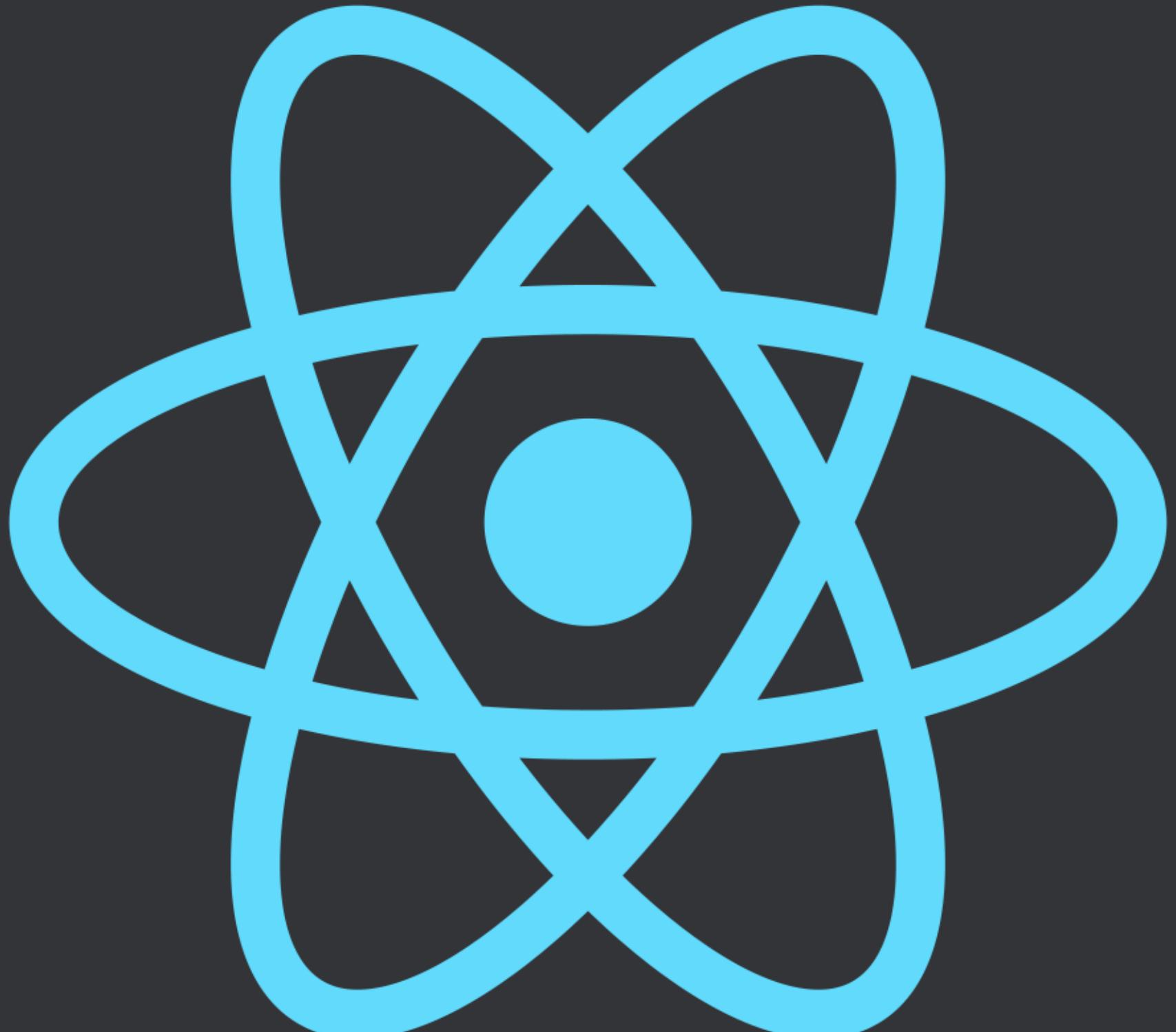
- Complex stuff is still hard, especially if there is no native React Native component<sup>2</sup>
- API is pretty stable, but new enough that something may break the ecosystem
- Compiler is pretty black box; if something doesn't work, good luck finding out why

---

<sup>2</sup> You can still write native modules in Xcode or Android Studio if React Native can't do what you want. I wouldn't try writing any OpenGL games in React.

# Taylor [not] Swift





# More please...

Where can I get more of this sweet, sweet React Native?

- React Native website: Obviously
- JS.coach: Kind of like Ruby Toolbox for React and React Native components
- The Complete React Native and Redux Course (from Udemy): Very good (if a bit overly detailed) course

# Thanks!

