

Expo App

Android

iOS



AppStore搜索Expo

React Native

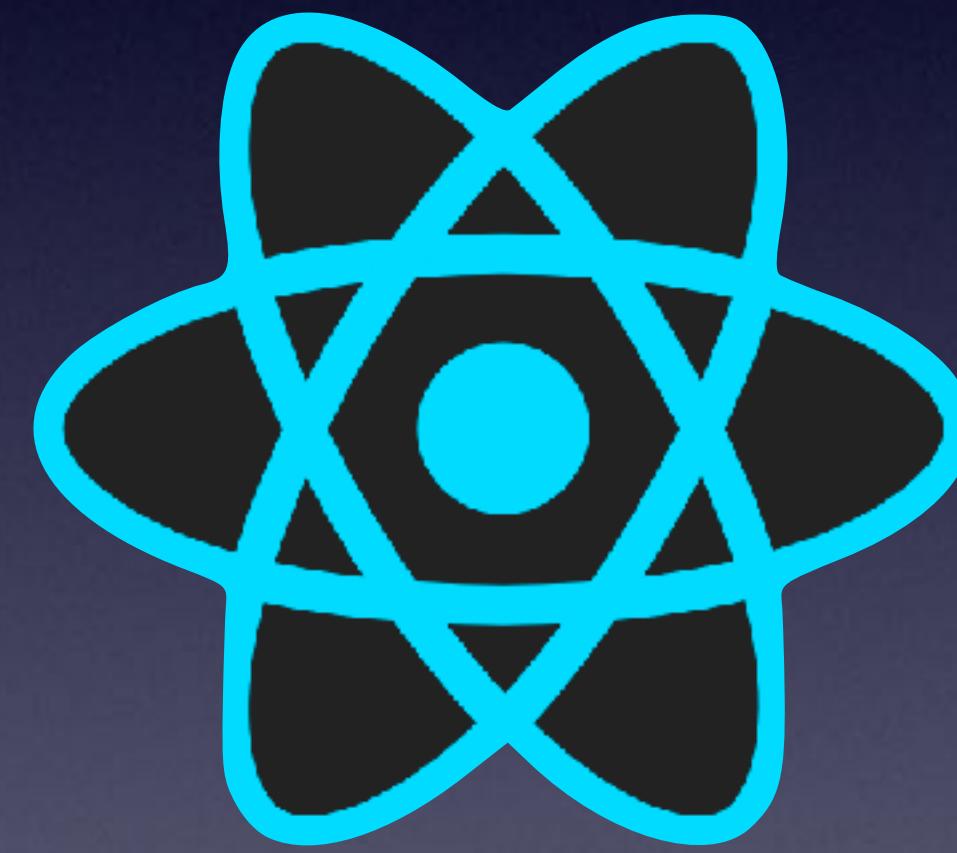
唐宗超



React.js

Learn Once, Write Anywhere

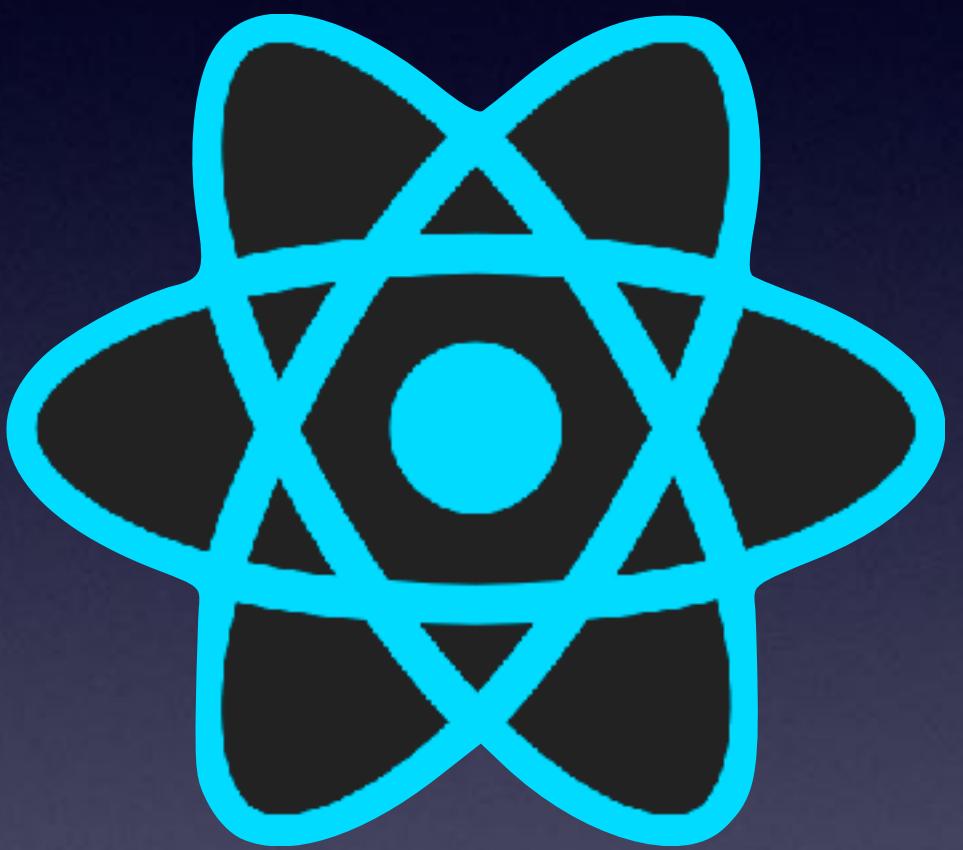
Learn Once, Write Anywhere



Write Once, Run Anywhere

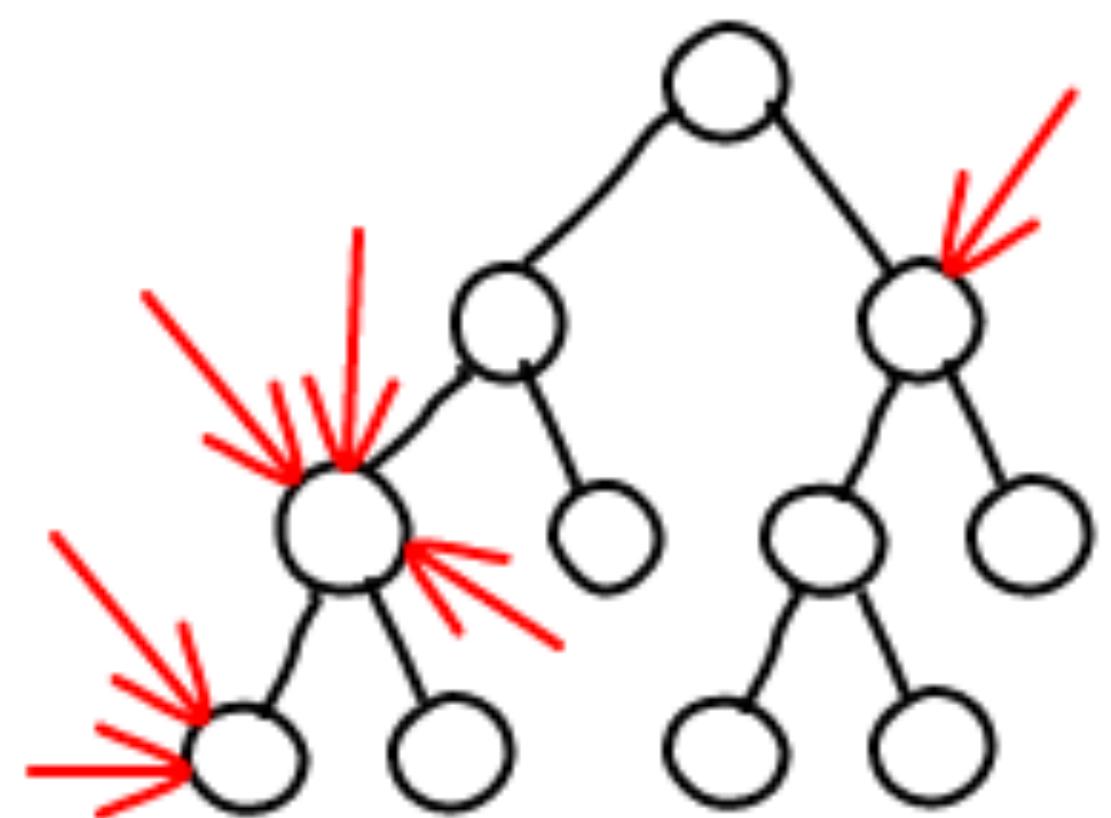


A JavaScript library
for building
user interface

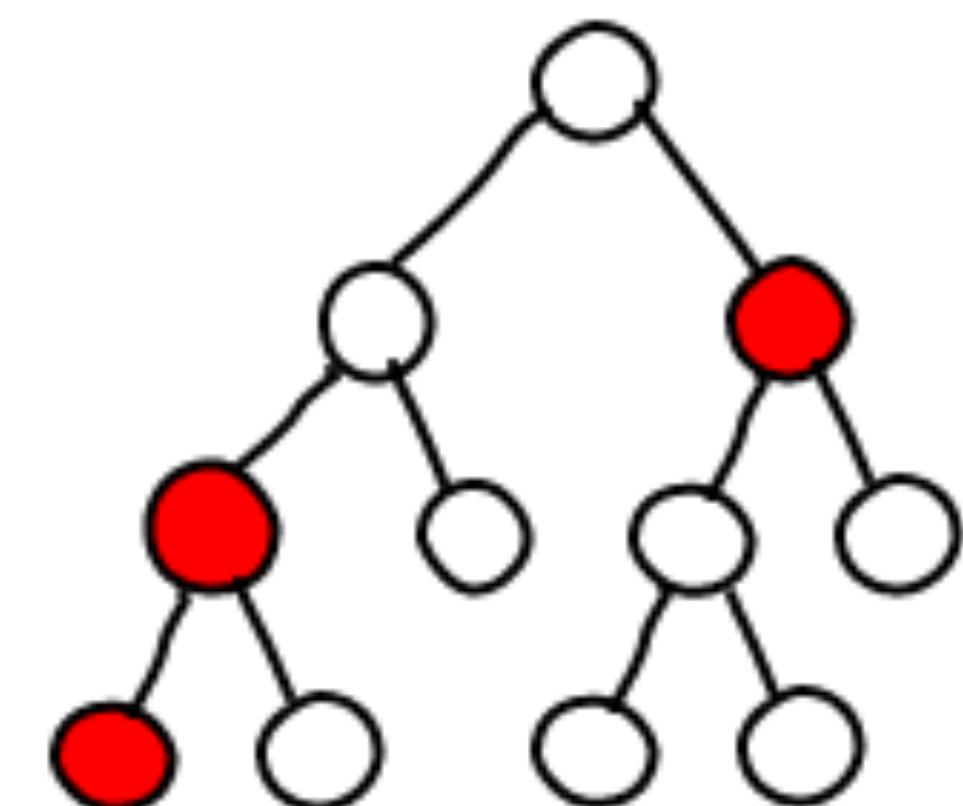


```
1 var React = require('react');
2 var ReactDOM = require('react-dom');
3
4 class MyComponent extends React.Component {
5   render() {
6     return <div>Hello World</div>;
7   }
8 }
9
10 ReactDOM.render(<MyComponent />, node);
```

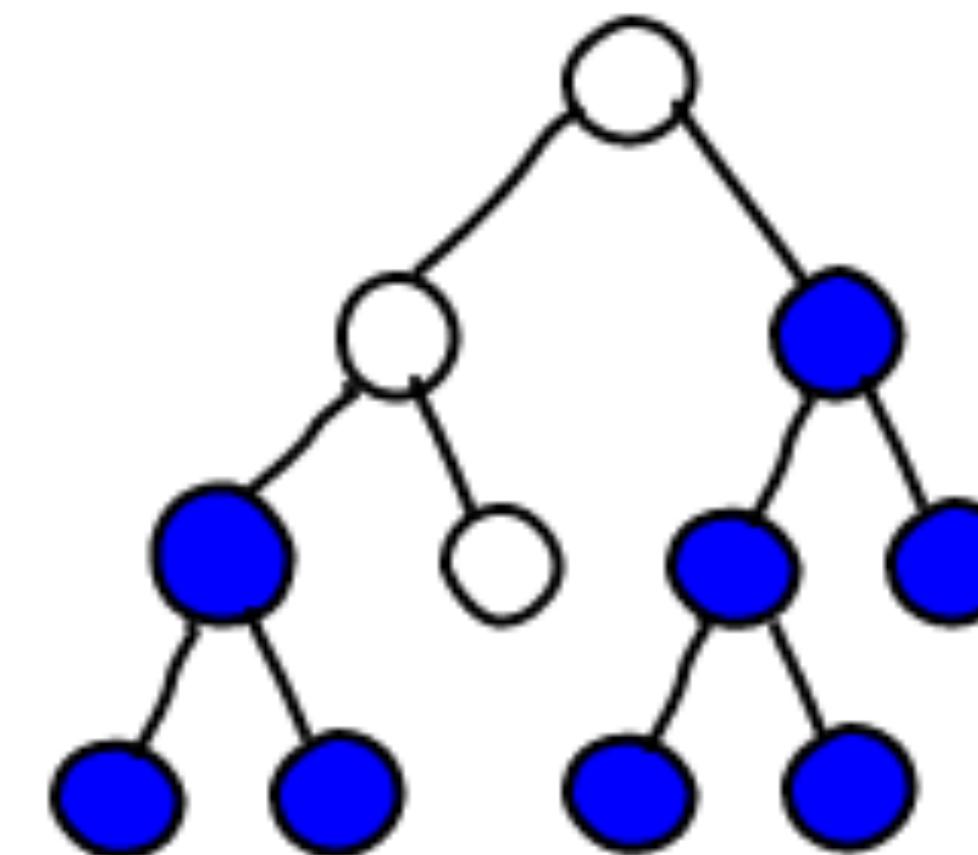
setState



Dirty

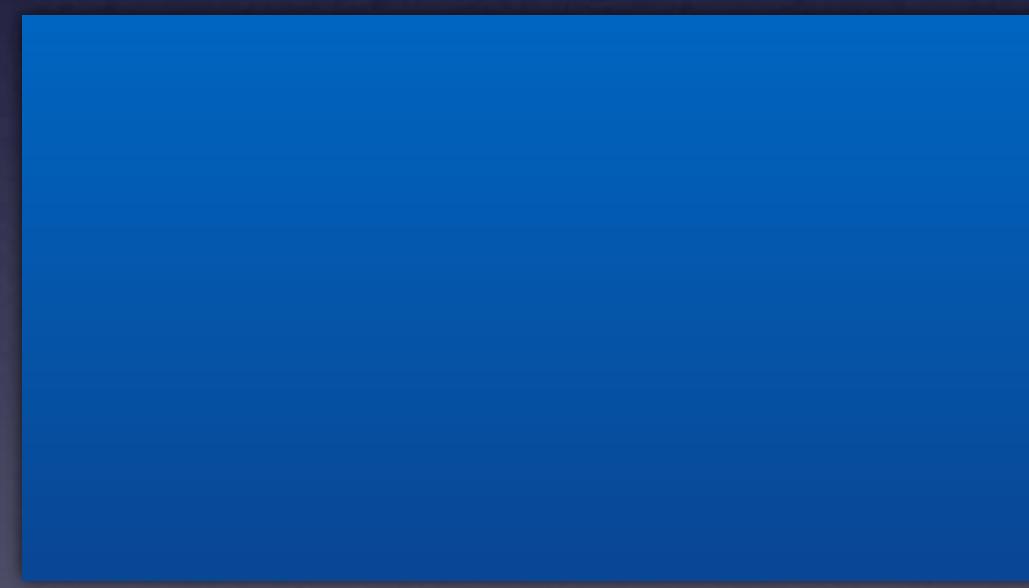


Re-rendered

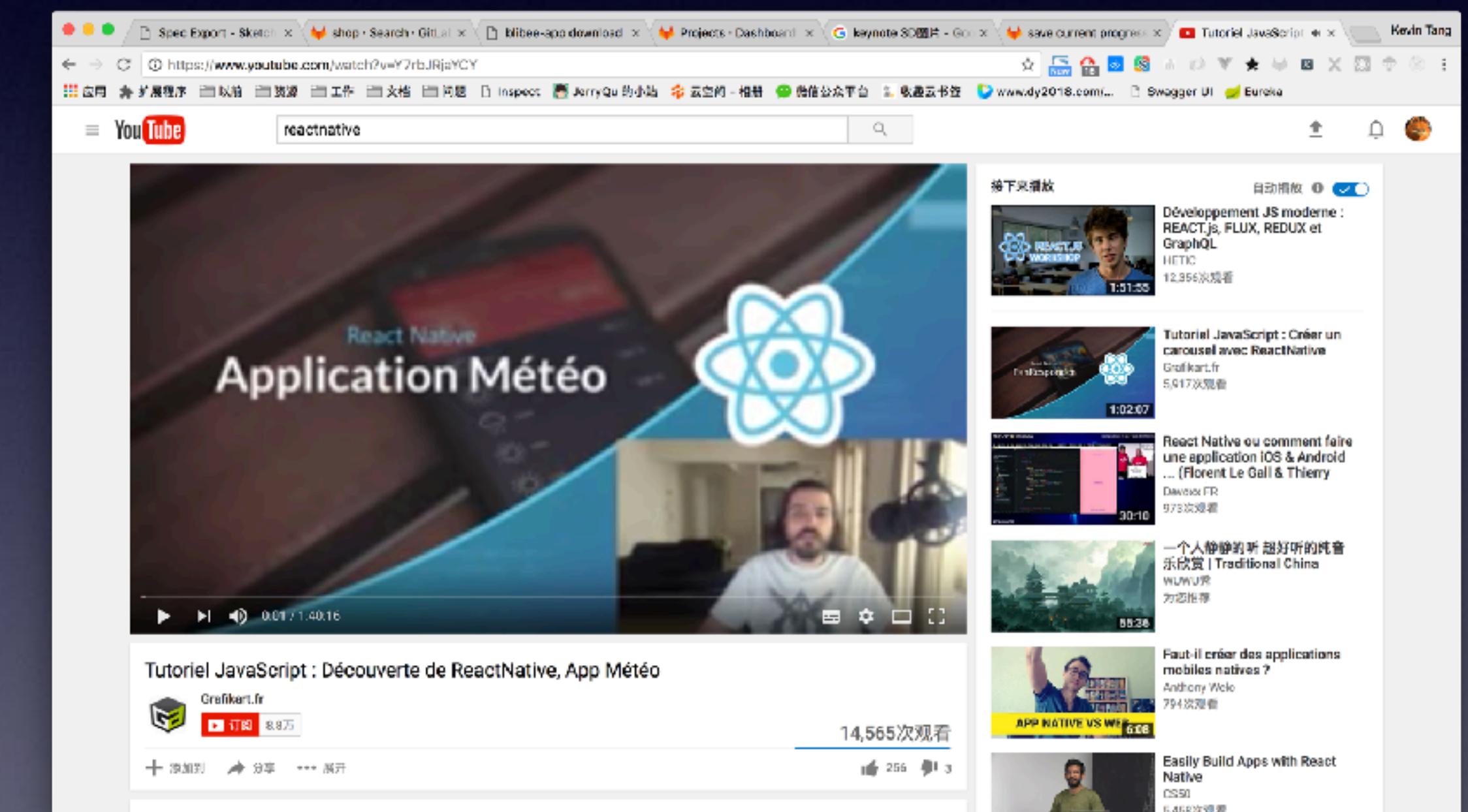


What happen after DIFF?

<div class="blueArea" />

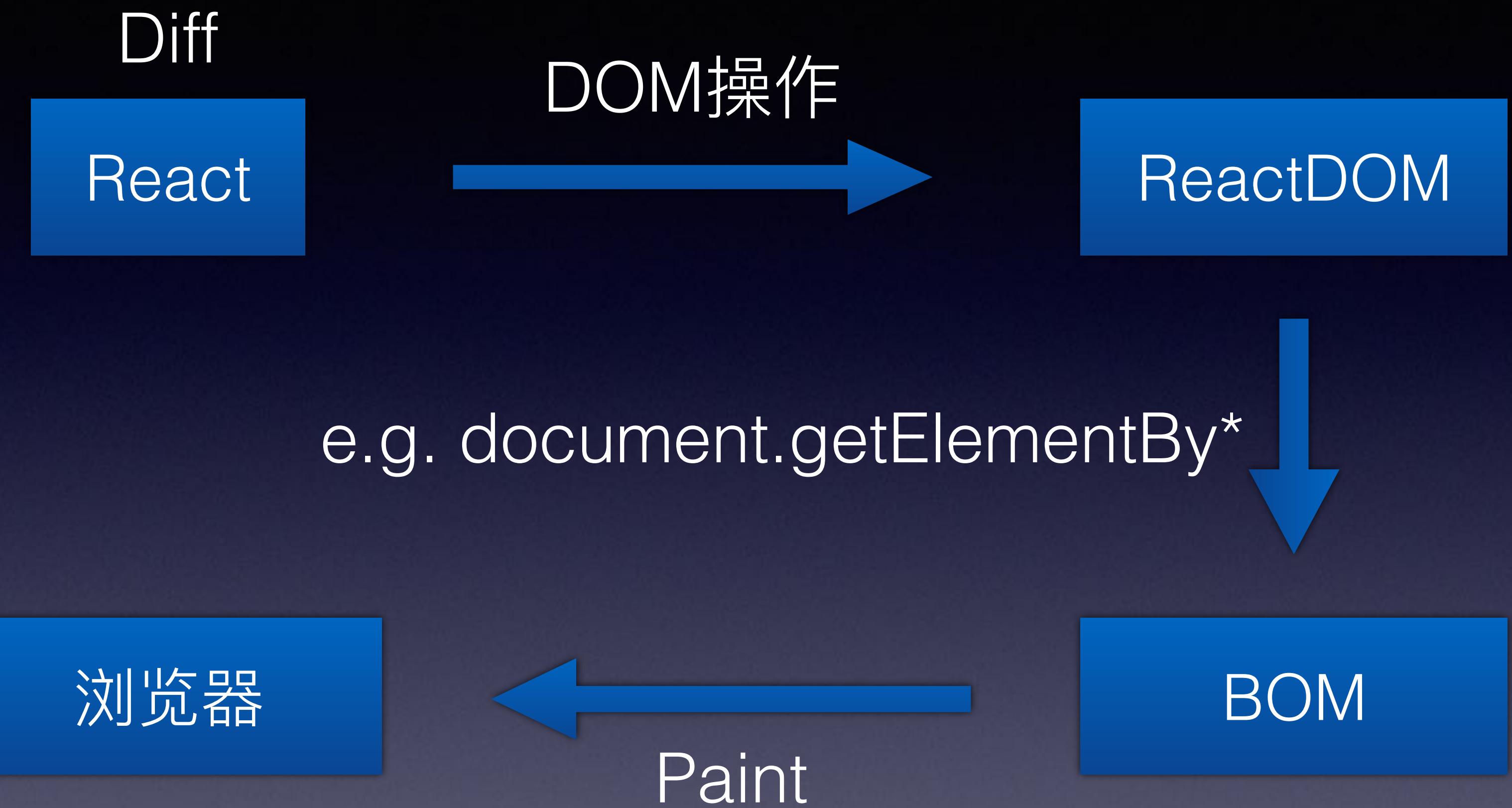


<video />



BOM (Browser Object Model)

window screen
location navigator
history setTimeout



Diff

React

一个大胆的想法

浏览器

Diff
React

一个大胆的想法

浏览器

原生应用

3D应用

音乐

Windows应用

Mac应用

虚拟现实

React

ReactDOM

ReactNative

ReactCanvas

ReactThree

ReactART

ReactMusic

ReactNativeWindows

浏览器

原生应用

3D应用

音乐

Windows应用

Mac应用

虚拟现实

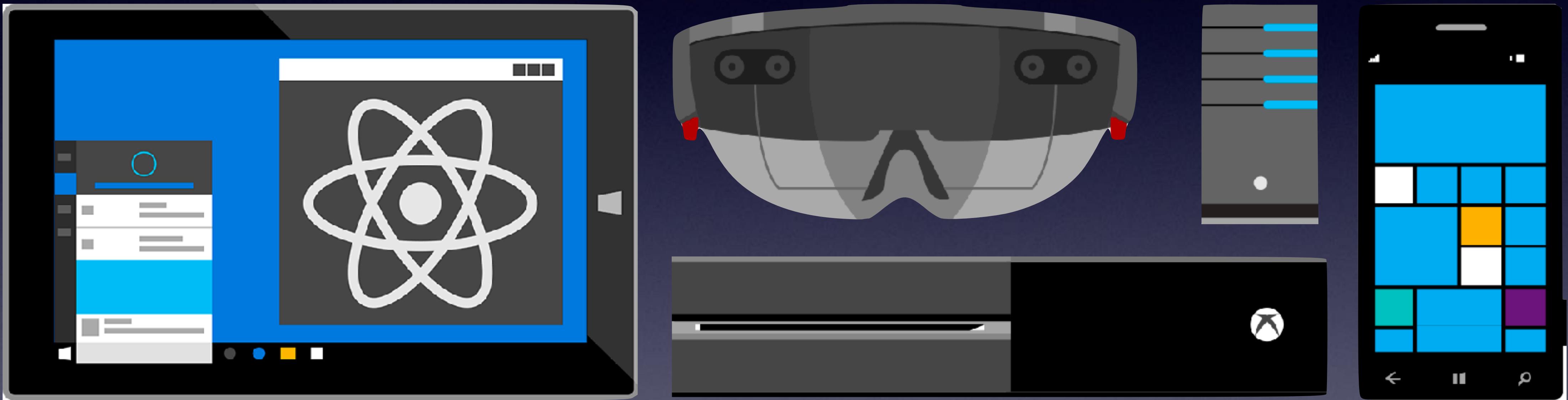
React.js + ReactDOM



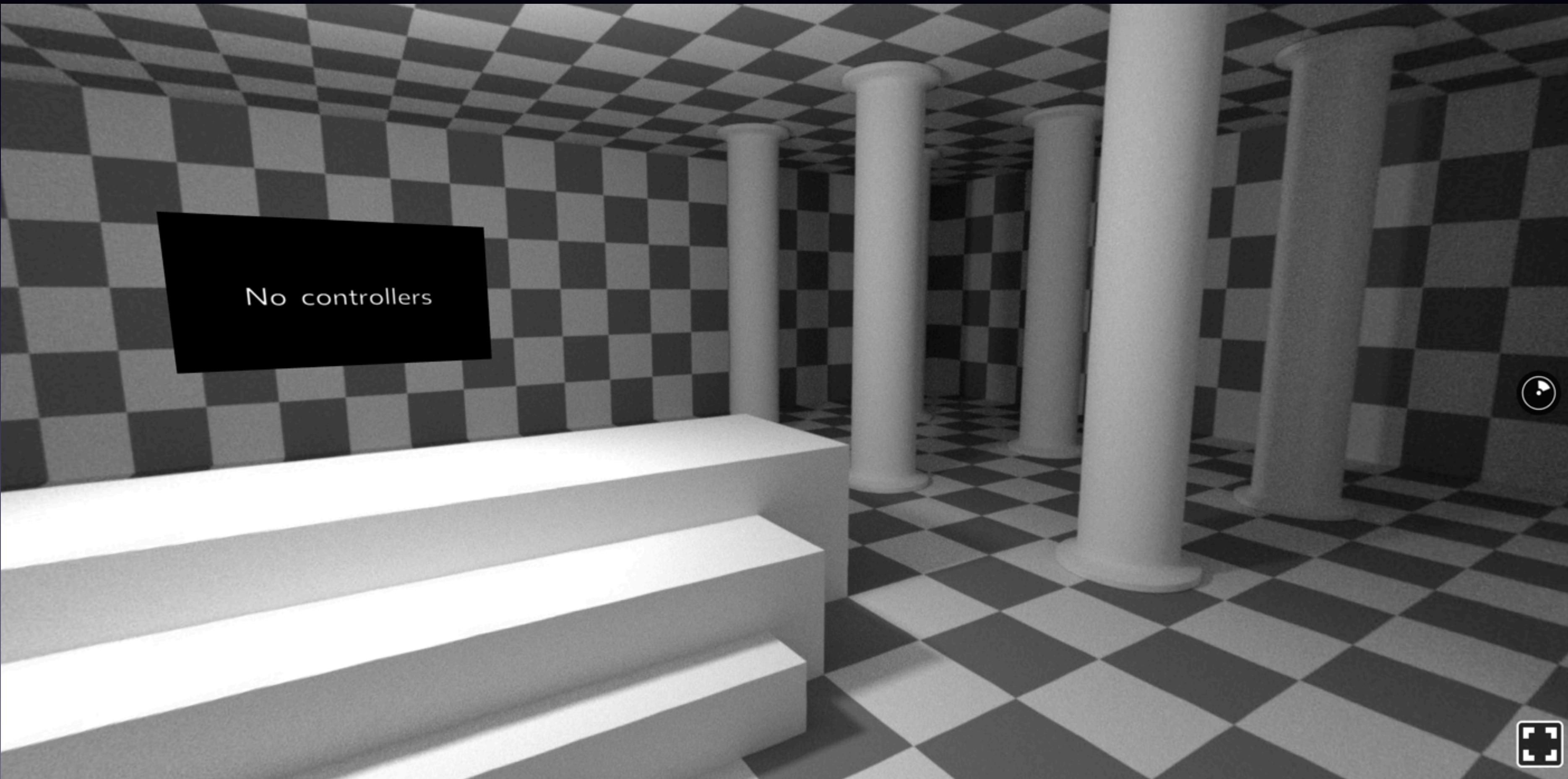
React.js + ReactNative



React.js + ReactNative Windows

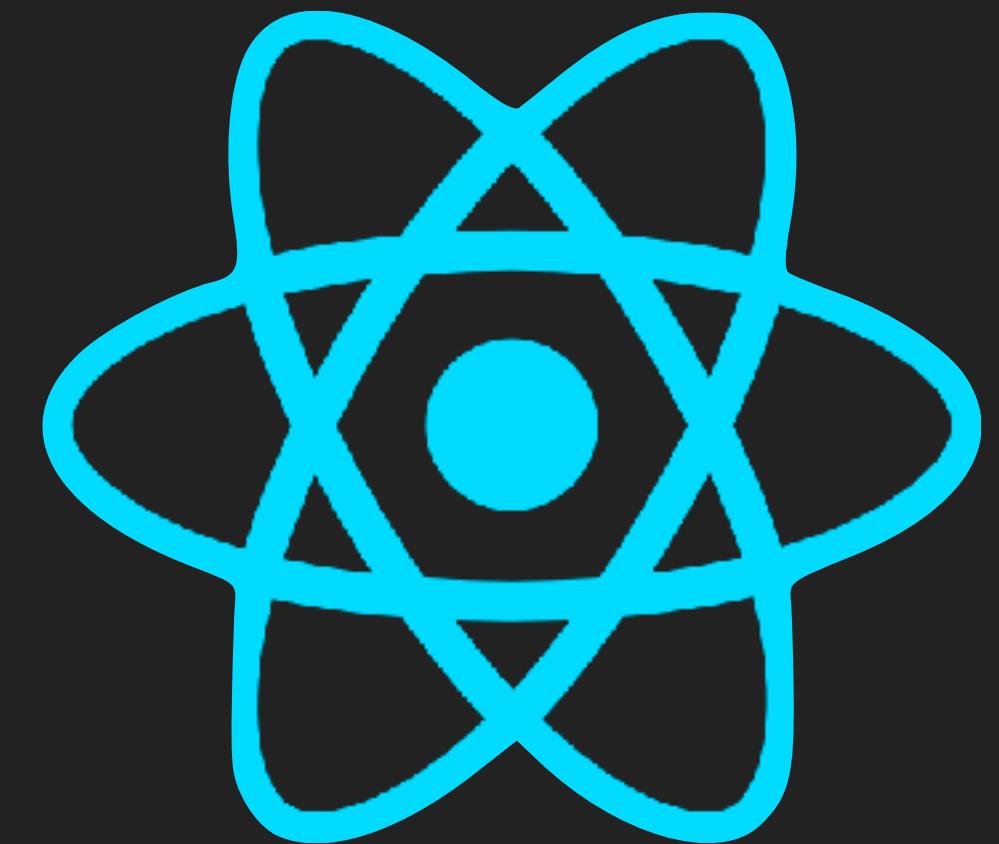


React.js + ReactVR

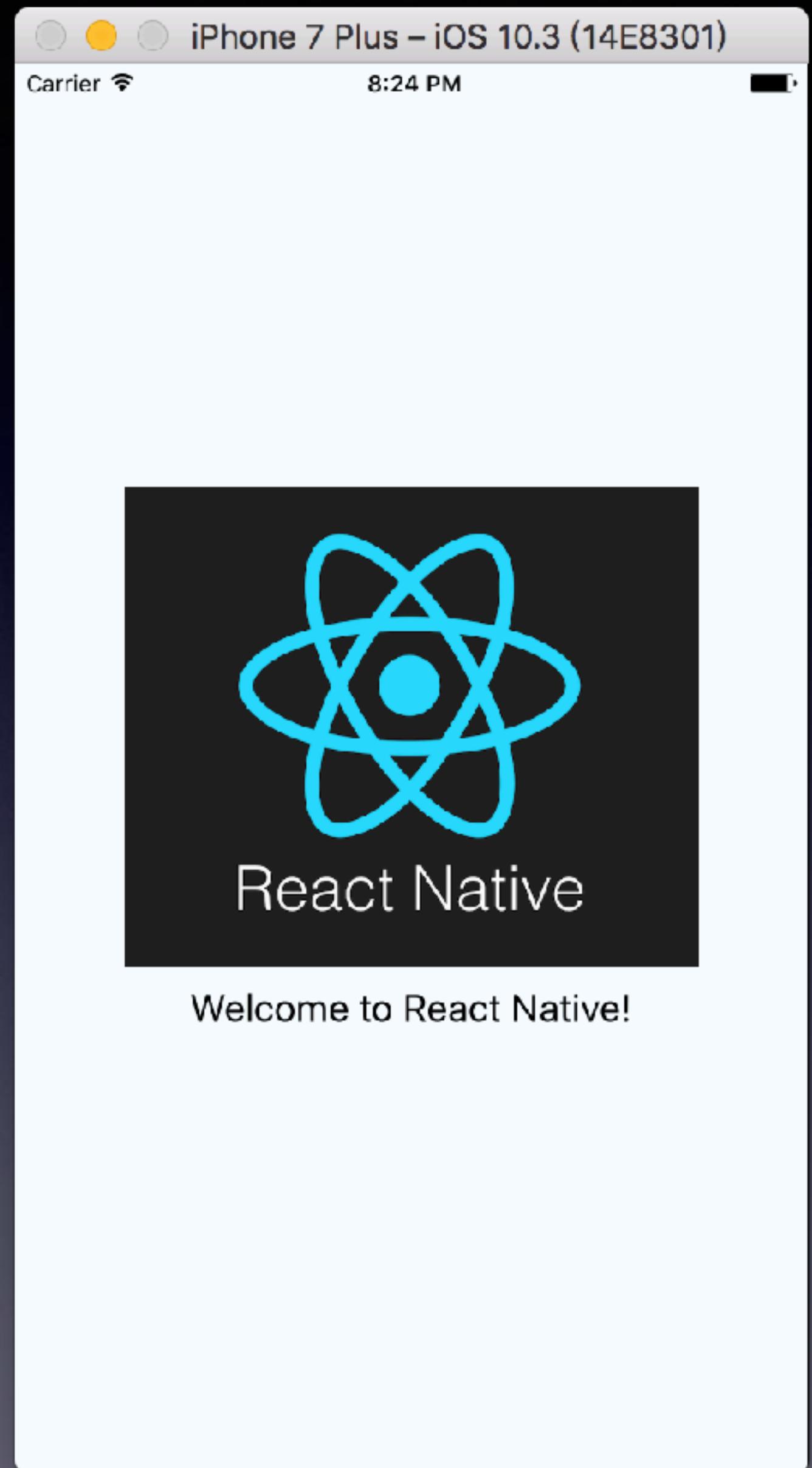


React Native

如何优雅地抢Native同学的饭碗



Scan with Expo



Native Development

Android

Java

Kotlin

iOS

Objective-C

Swift



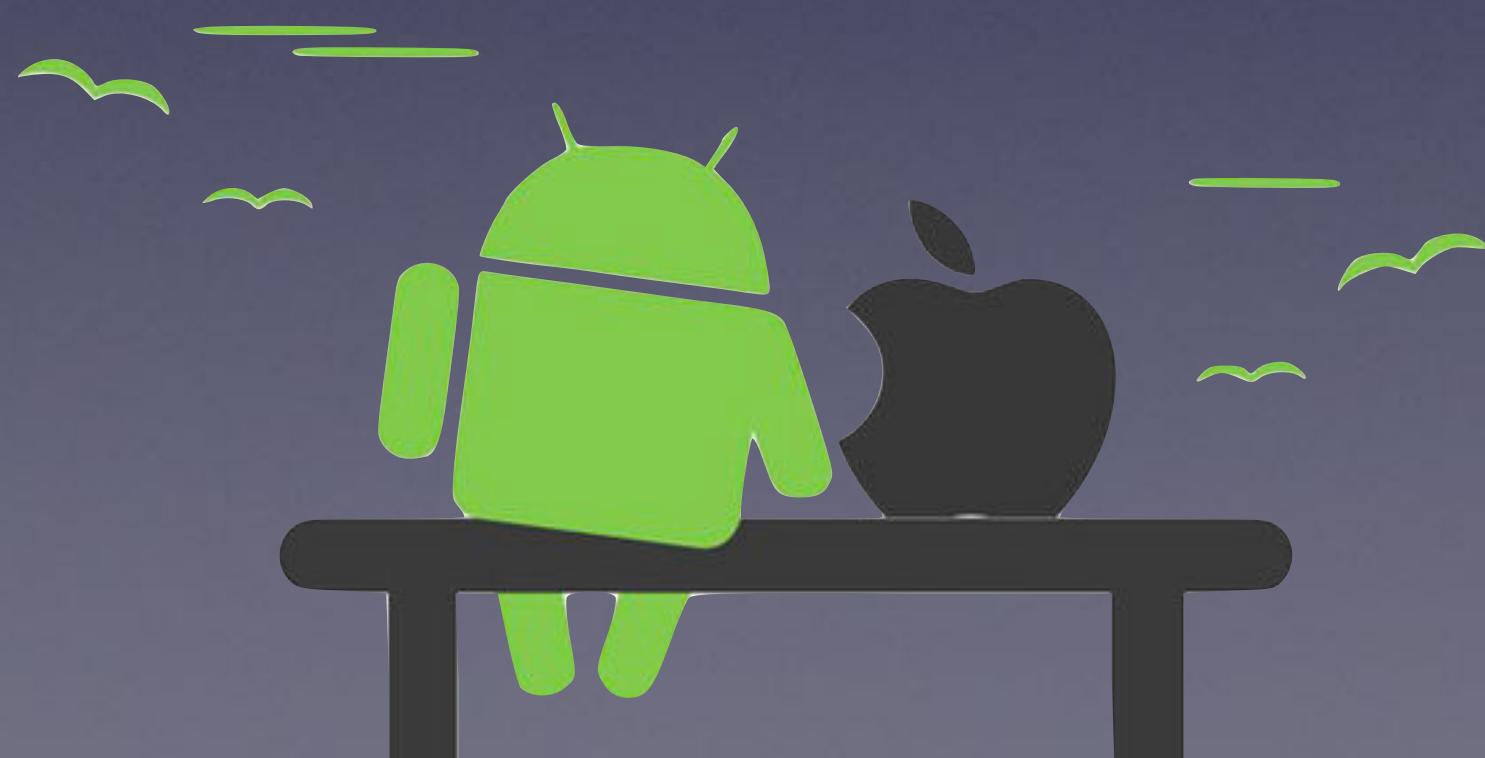
React Native Development

Javascript

ReactNative

Android

iOS



float: left?



滚动 transition 等等

position: fixed?



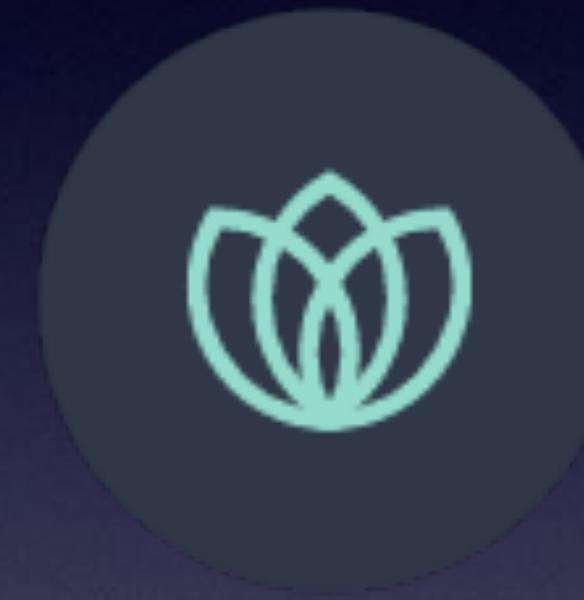
```
<div>test</div>  
<h1>test</div>
```



```
<div onClick="xxx" />
```



CSS



Yoga

A cross-platform layout engine

Size

React Native中的尺寸都是无单位的
表示的是与设备像素密度无关的逻辑像素点。

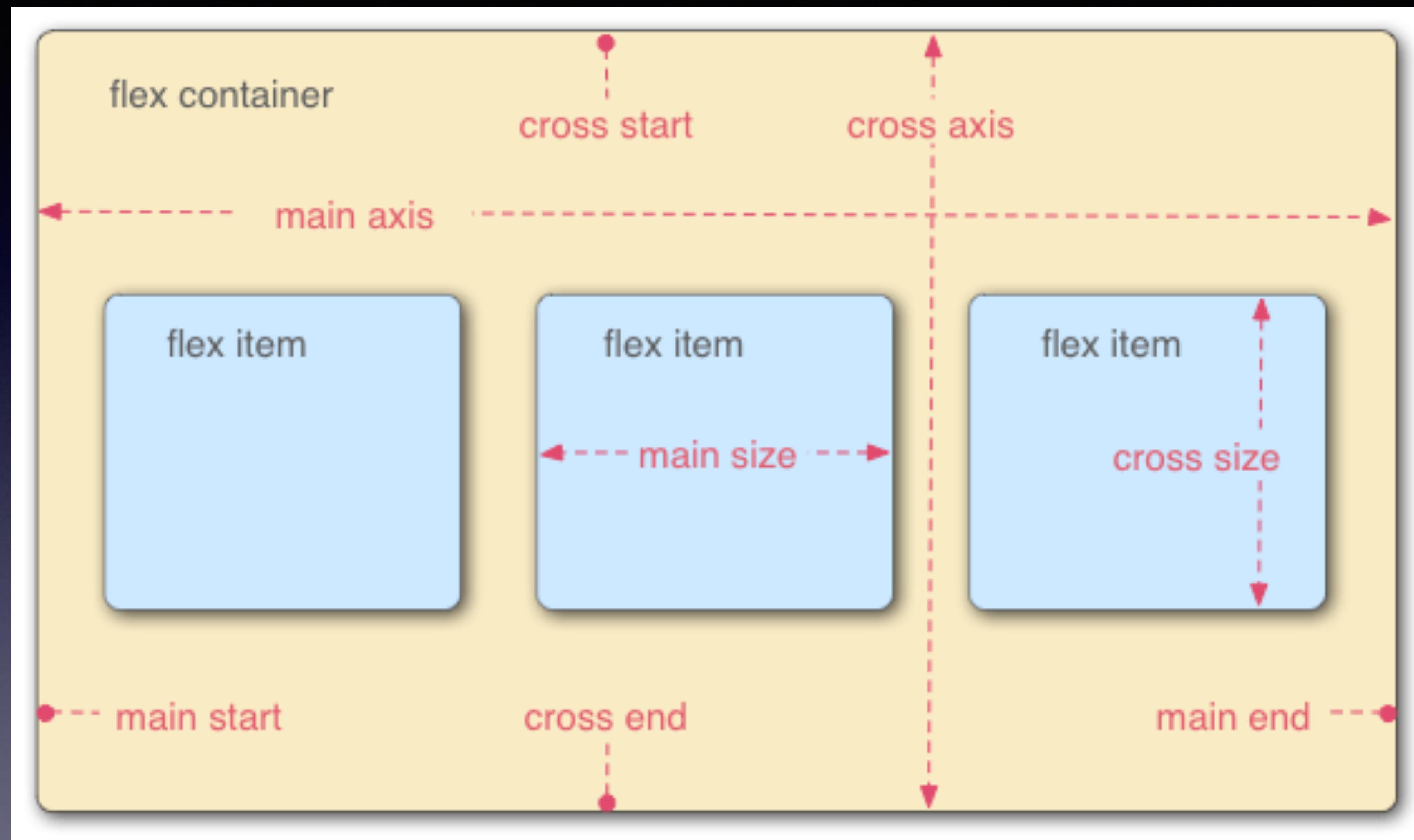
Network

Ajax?



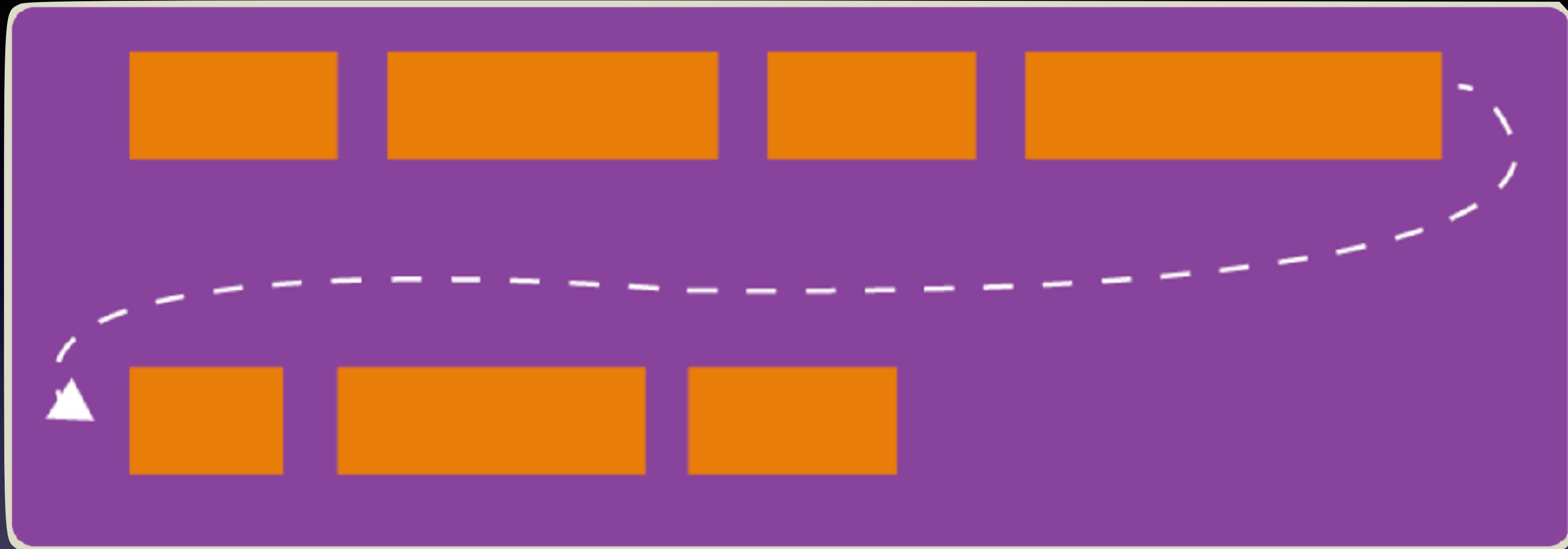
```
fetch('https://mywebsite.com/endpoint/', {  
  method: 'POST',  
  headers: {  
    'Accept': 'application/json',  
    'Content-Type': 'application/json',  
  },  
  body: JSON.stringify({  
    firstParam: 'yourValue',  
    secondParam: 'yourOtherValue',  
  })  
})
```

Flexbox

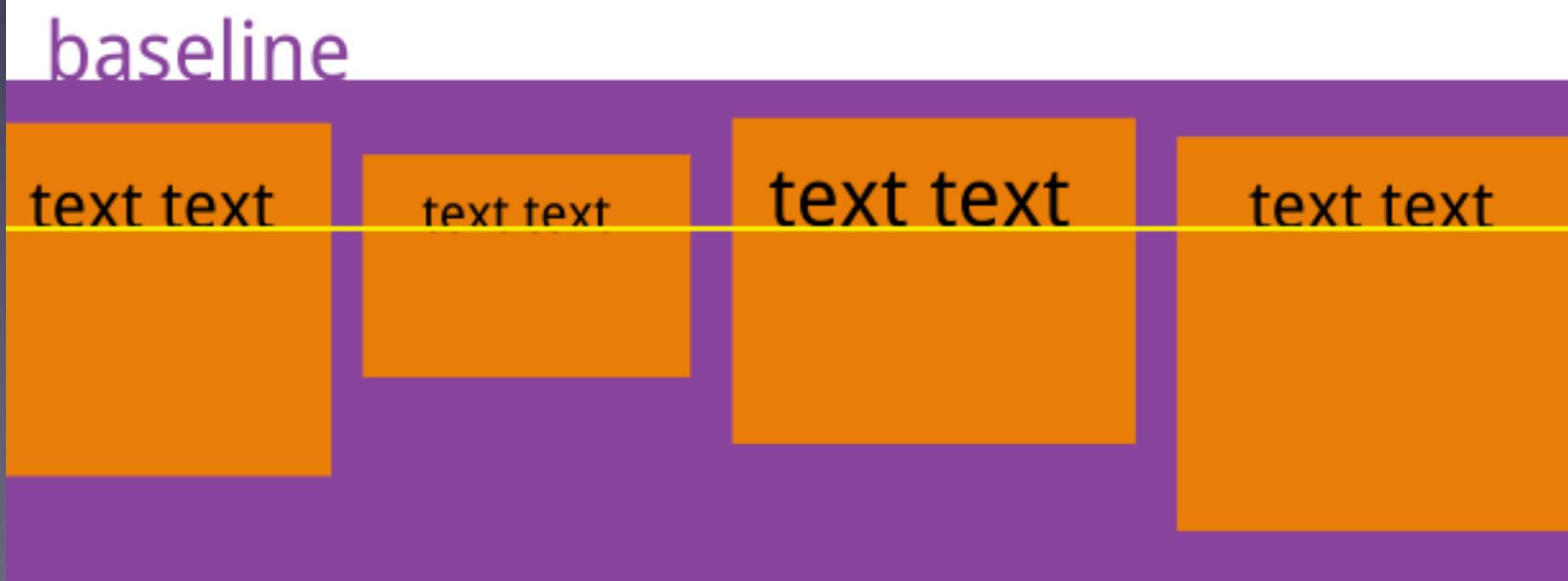
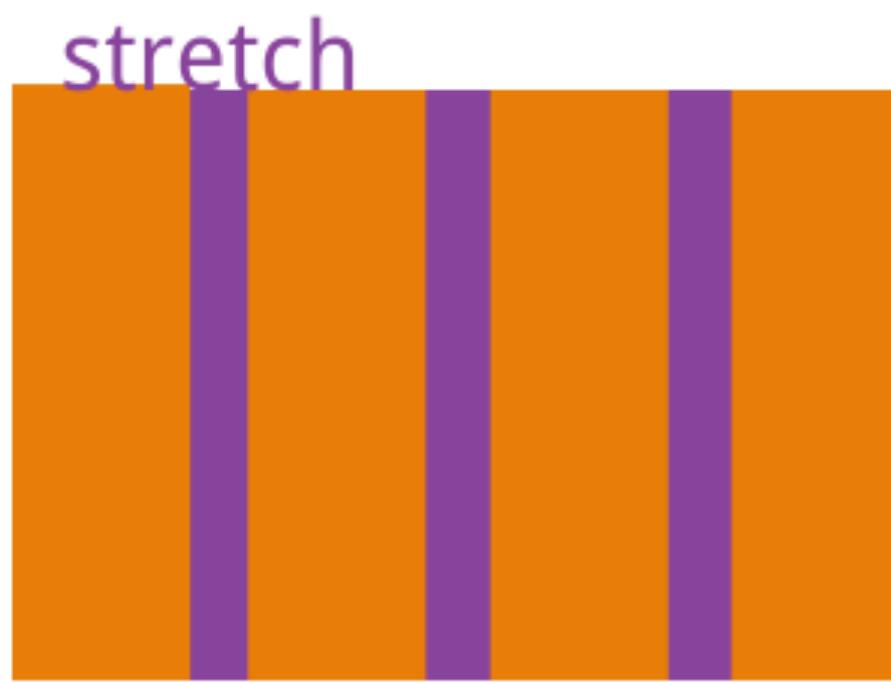
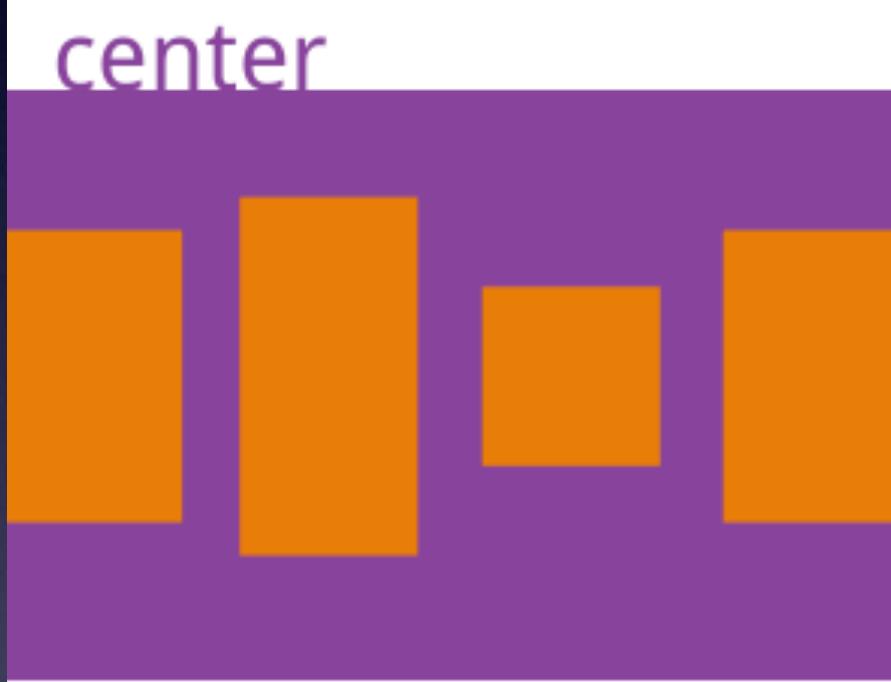
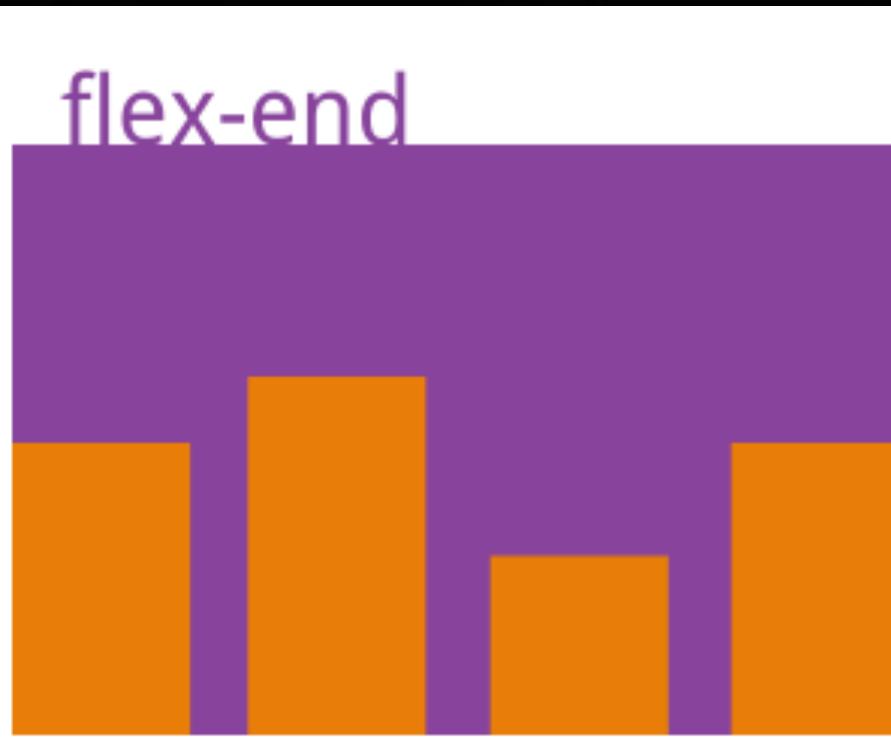
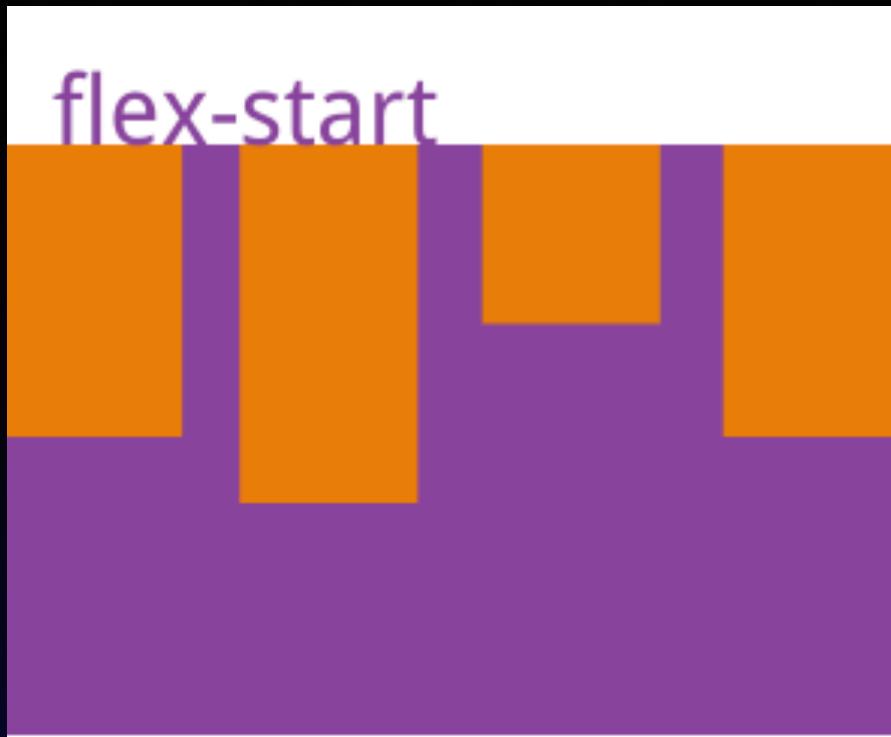


```
.box {  
  flex-direction: row | row-reverse | column | column-reverse;  
}
```





```
.box{  
  flex-wrap: nowrap | wrap | wrap-reverse;  
}
```



Scroll

Click

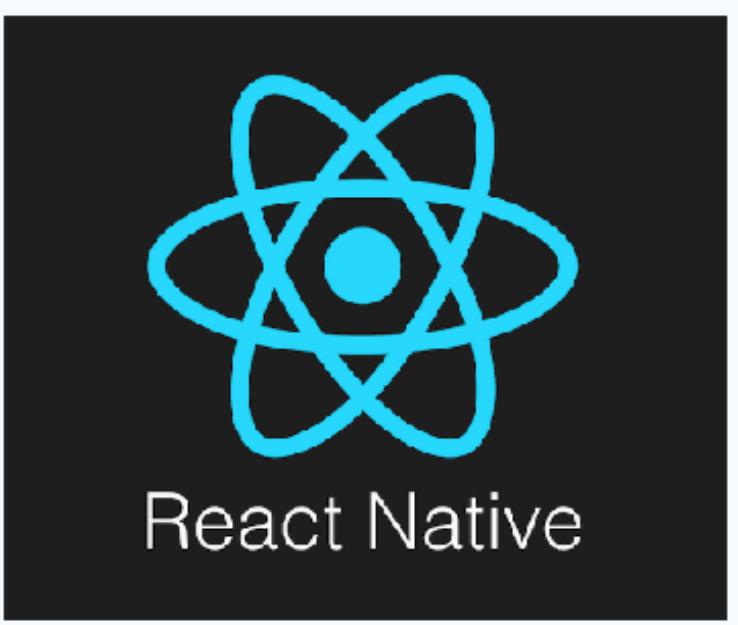
```
<div onClick="onPress">  
</div>
```

```
<Touchable* onPress="onPress">  
  <View>  
  </View>  
</Touchable*>
```

How to Code?

```
1 class Test extend React.Component {  
2   render() {  
3     <div class="body">  
4         
5       <div>Welcome to React Native!</div>  
6     </div>  
7   }  
8 }  
9  
10  
11  
12
```

```
1 .body {  
2   width: 100%;  
3   height: 100%;  
4   background-color: #F5FCFF;  
5 }  
6  
7 .image {  
8   font-size: 20px;  
9   text-align: center;  
10  margin: 10px;  
11 }  
12
```



Welcome to React Native!

Homework

封装一个Button组件， 要求使用TouchableOpacity
并且要求有一个可配置的props， 可以对onPress进行debounce

```
1 <Button
2   style={styles.xxx}
3   onPress={this.handlePress}
4   debounceDelay={300}
5   text="确定"
6 />
```

ReactNative实现右侧界面



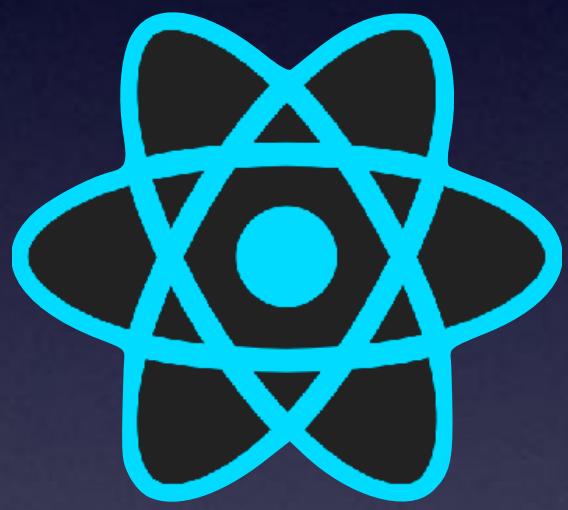
最后再说两句.....



Redux



GraphQL

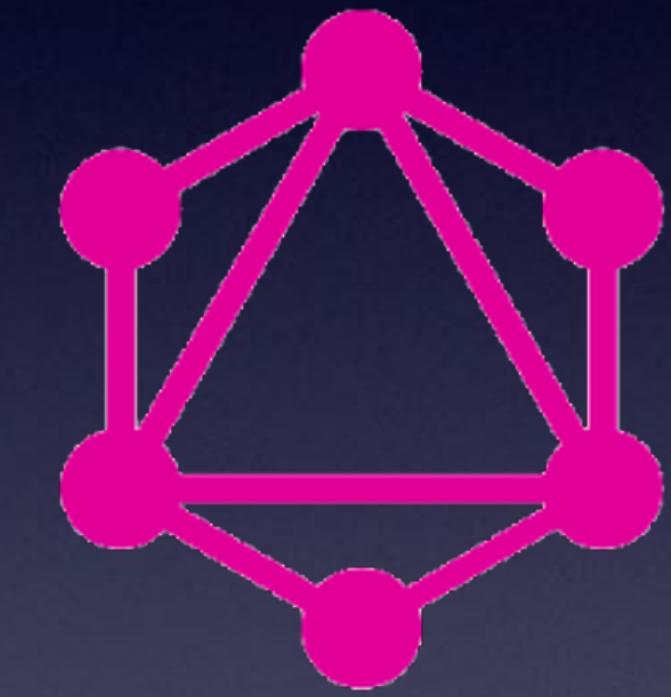


React

Immutable



Relay



GraphQL

谢谢 接下来是
Q&A

如果你们还醒着
我讲完