

React Native

Learn once,write anywhere

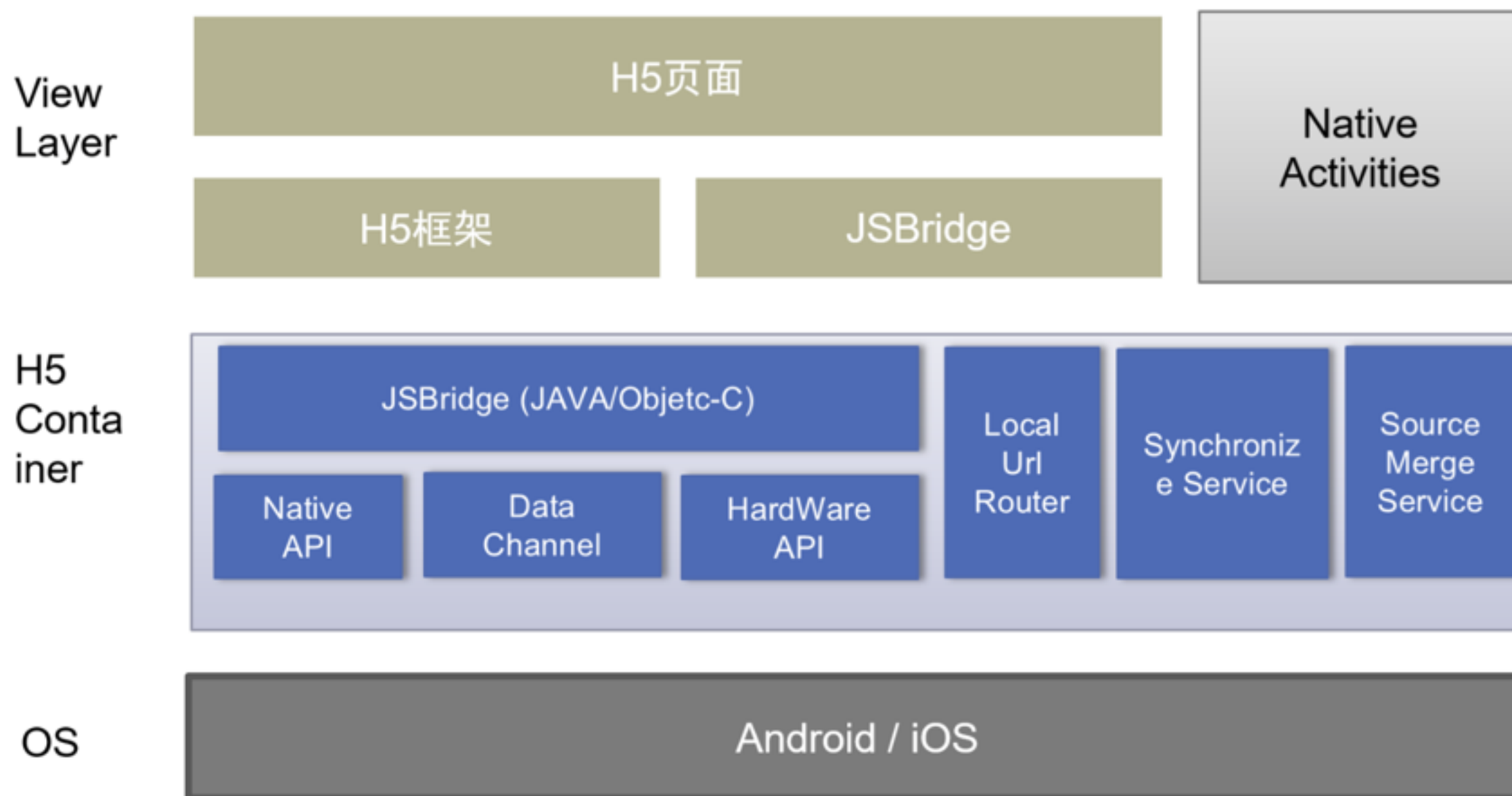


猫猫姐姐 出品
www.bingjie.me

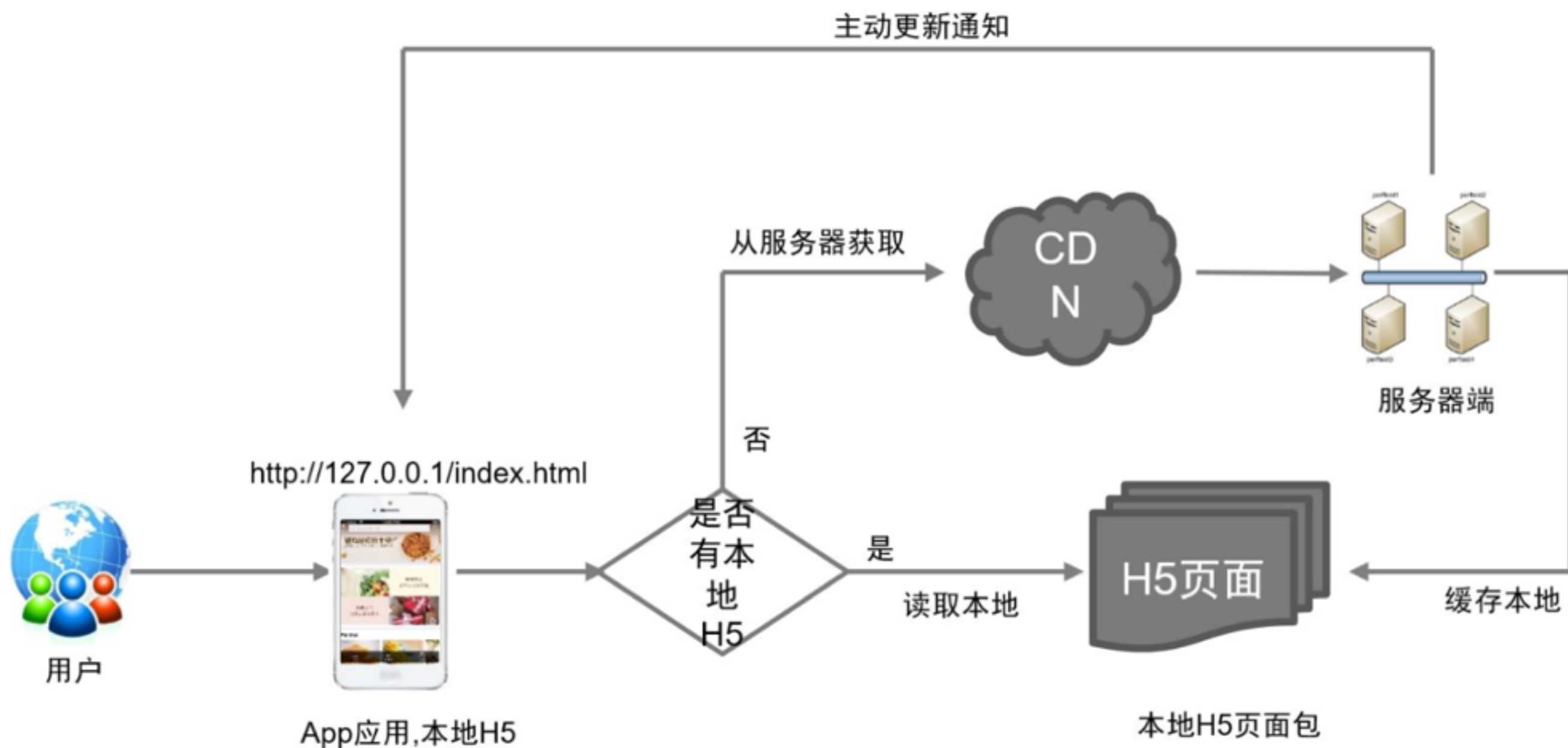
以往的app演化史



混合应用框架



h5本地缓存机制



What is the React?

"A JAVASCRIPT LIBRARY FOR BUILDING USER INTERFACES"

“——一个用用来构建 用用户界面面的 javascript 库”。

Why niu



极速的渲染性能



组件互相独立，关系隔离，可复用



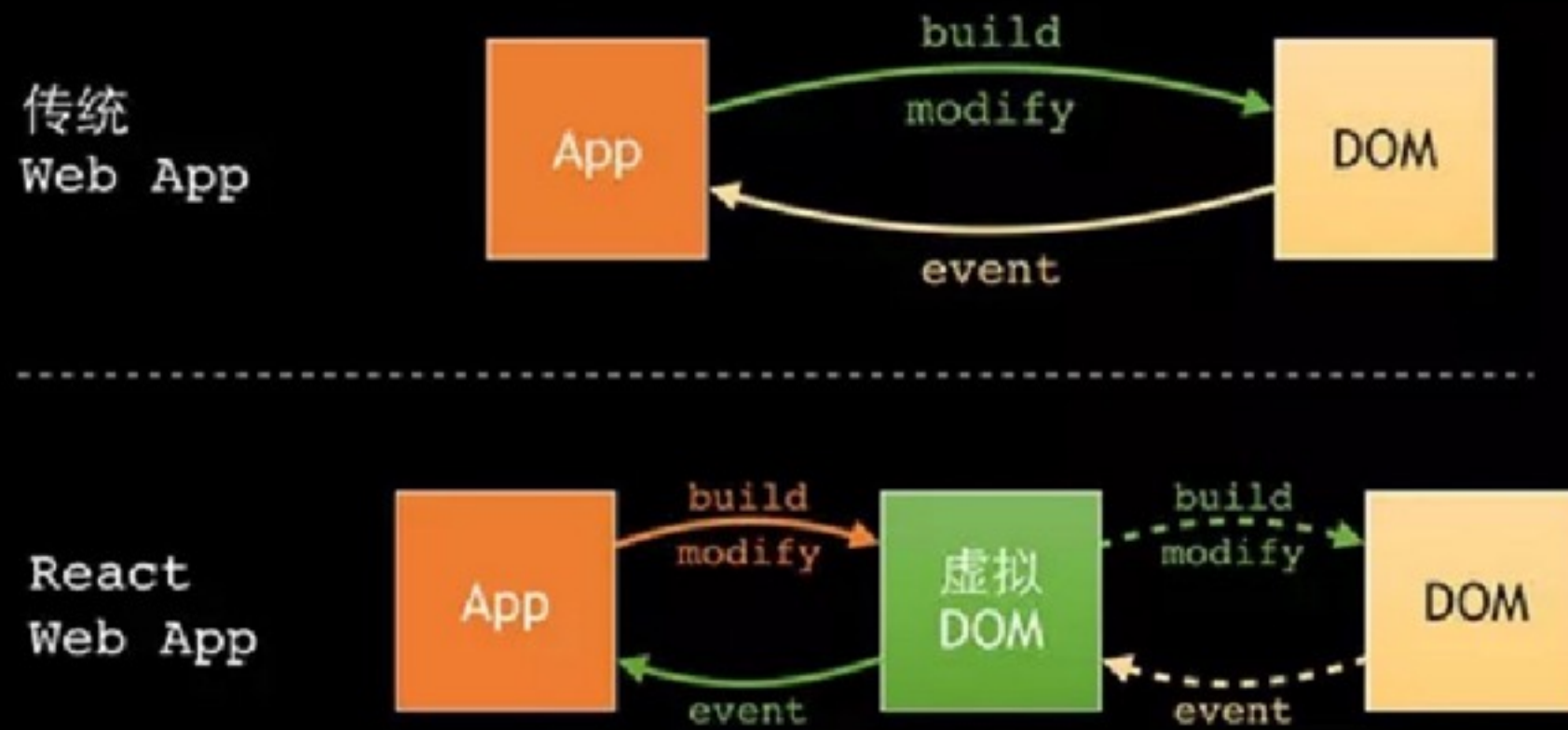
跨平台

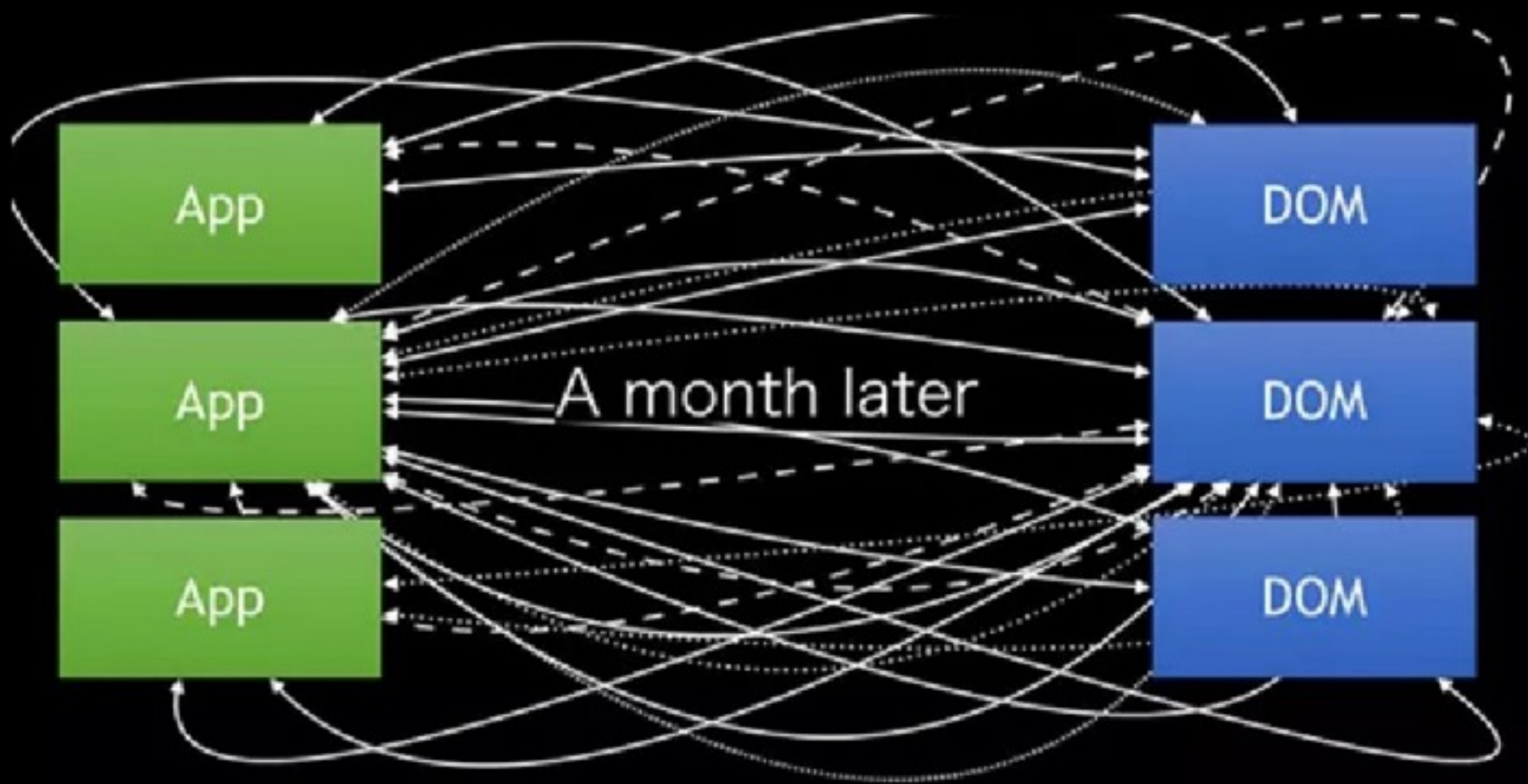


Facebook和Instagram帮我们做了充足的测试

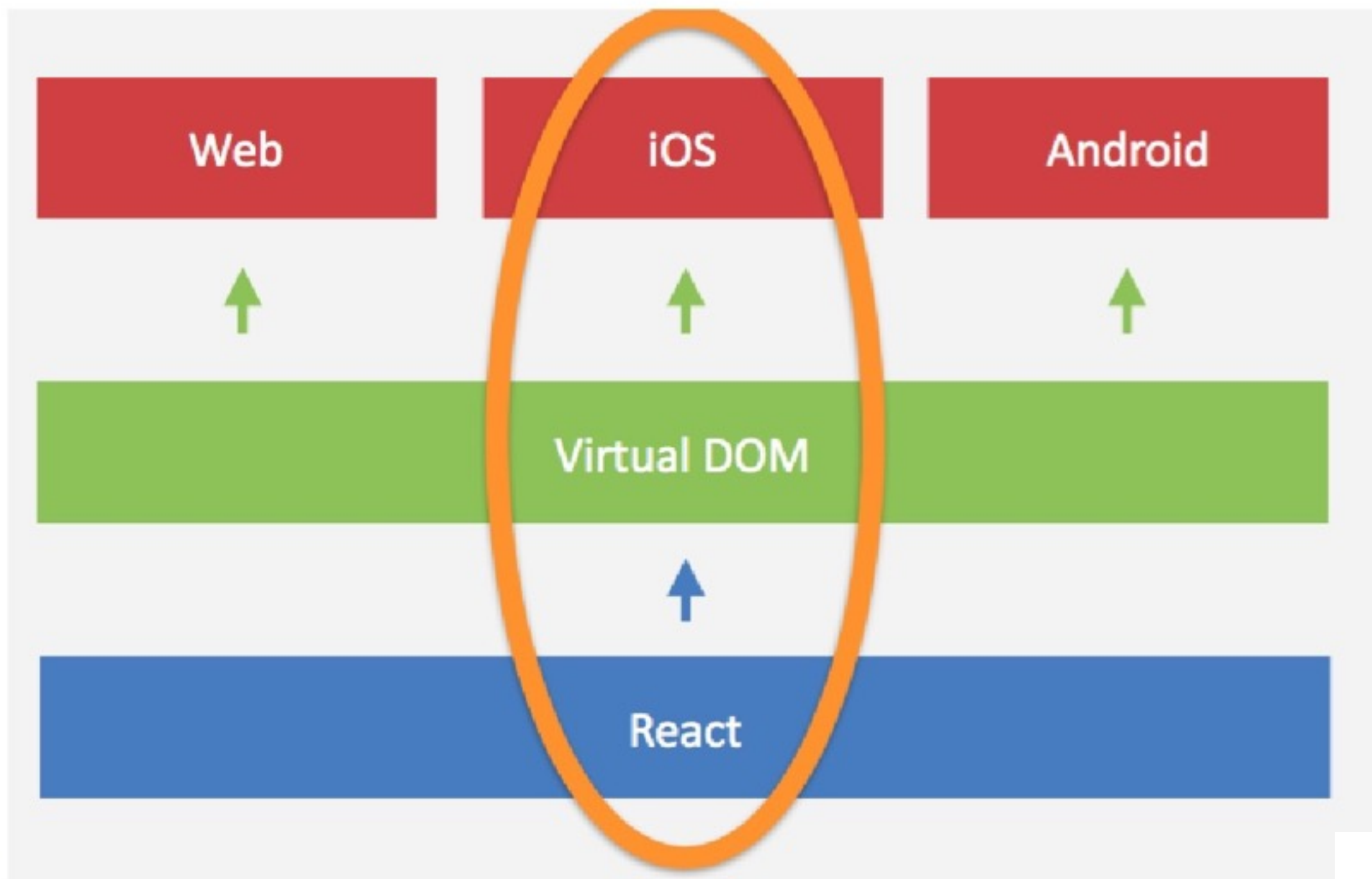
- Facebook MAU 13.9亿，移动端 7.45亿
- Instagram MAU 3亿

Virtual Dom

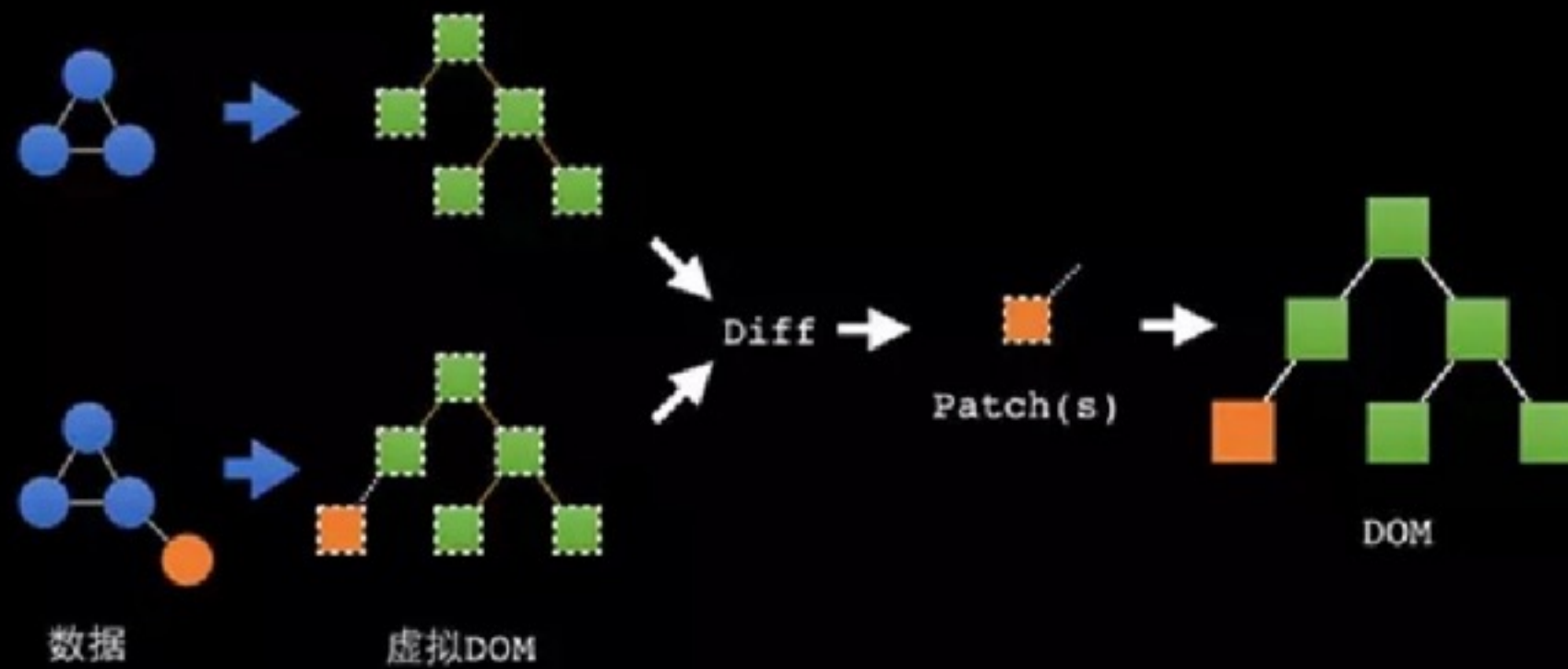




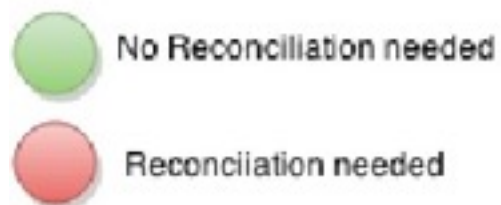
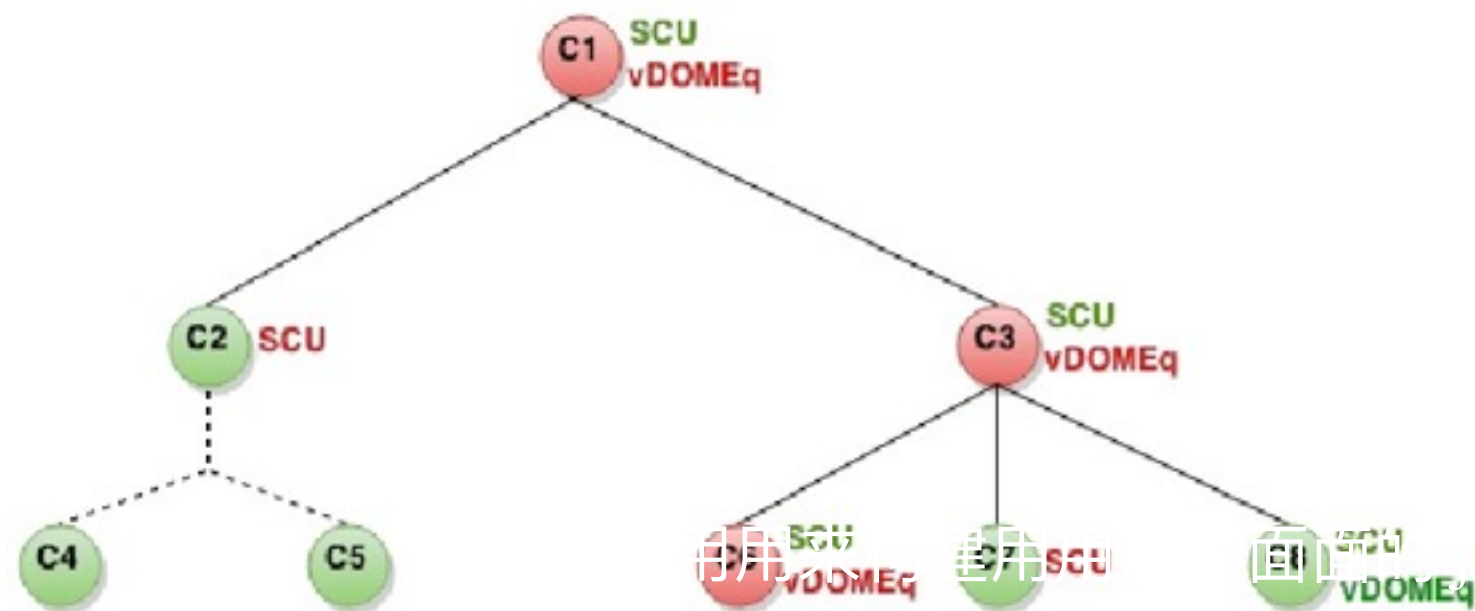
浏览器的工作原理



Advanced Performance



shouldComponentUpdate



SCU shouldComponentUpdate?
SCU
vDOMEq are virtual DOMs equivalent?
vDOMEq

react-native 优点

- 引入web开发方式
- 编译速度加快
- 语言更统一
- 新的加载与显示逻辑
- 动态UI
- 热更新
- 跨平台

react-native 所存在的问题

Android 与 iOS 表现不同

Android端不尽如人意

ScrollView性能

子组件数量	加载时间(ms)	占用内存(MB)	绘制时间*(ms)
10	309	19.7	14.666
100	1170	21.9	15.016
1000	9461	26.5	15.025

* 注，这里的绘制时间，是在 *Tree View* 中获得的 *Draw* 时间。

这里有一个 **mViews**，用来保存所有的子 View，绑定 View 的时候只是简单用一个空的 View (RecyclableWrapperViewGroup) 包了一下。

这样一来，RecyclerView 完全没有什么起到复用的作用呀！
测试一下，确实也是这样，性能问题还是很严重。

这里我们也可以得到一个结论：

RN 中的 ListView 也不是我们想象的 ListView 该有的性能。

Q: RN和React.js是一个东西吗?

A: RN和React.js共用一些抽象层, 但具体有很多差异, 且目标平台不同:
RN目前只能开发iOS/Android App, 而React.js用于开发web页面。

Q: RN所支持的最低iOS和Android版本?

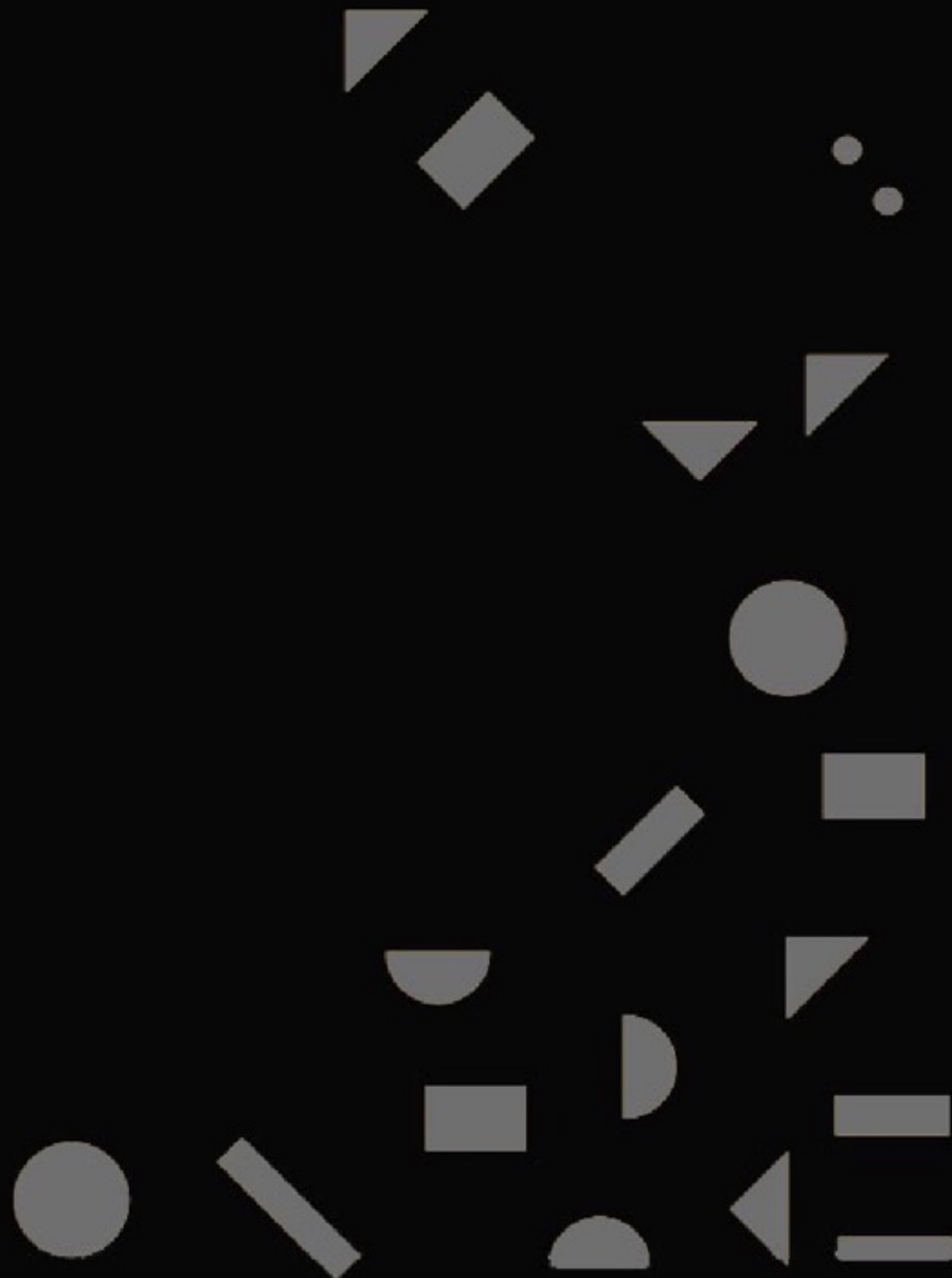
A: Android ≥ 4.1 (API 16)

iOS ≥ 7.0

Q: RN和cordova/phonegap是一个东西吗?

A: 不一样。RN不是一个webview (但包含了webview组件), 不能直接复用web页面代码。
RN的性能接近原生, 超过cordova/phonegap。

组件！



ActivityIndicatorIOS
DatePickerIOS
DrawerLayoutAndroid
Image
ListView
MapView
Navigator
Modal
NavigatorIOS
PickerIOS
ProgressBarAndroid
ProgressViewIOS
PullToRefreshViewAndroid
ScrollView

SegmentedControlIOS
SliderIOS
SwitchAndroid
SwitchIOS
TabBarIOS
TabBarIOS.Item
Text
TextInput
ToolBarAndroid
TouchableHighlight
TouchableNativeFeedback
TouchableOpacity
TouchableWithoutFeedback
View
ViewPagerAndroid
WebView

组件列表

component = data + jsx

```
var Hello = React.createClass({
  propTypes: {
    name: React.PropTypes.String.isRequired
  },
  getInitialState: function () {
    return {
      name: this.props.name
    };
  },
  render: function () {
    return (
      <ul>
        <li>hello {this.state.name}!</li>
      </ul>
    );
  }
});
```

props
属性, 数据源

state
来自props

JSX
通过state控制DOM结构变化

state与props

状态机

- React 框架将所有的UI视为一个简单的状态机，那么任意一个UI场景就是一个状态机中一种状态。根据决定状态的状态机变量的值，React框架渲染出状态机的当前状态——对于开发者来说，单个UI场景就被渲染出来了。随着状态机变量值的改变，UI状态机也在不停的改变状态，UI场景也随之不停的被重新渲染。这样一个过程可以轻松的做到数据与UI保持一致。

不要改变prop值

props 主要作用是提供数据来源，可以简单的理解为 props 就是构造函数的参数。state 唯一的作用是控制组件的表现，用来存放会随着交互变化状态，比如开关状态等。

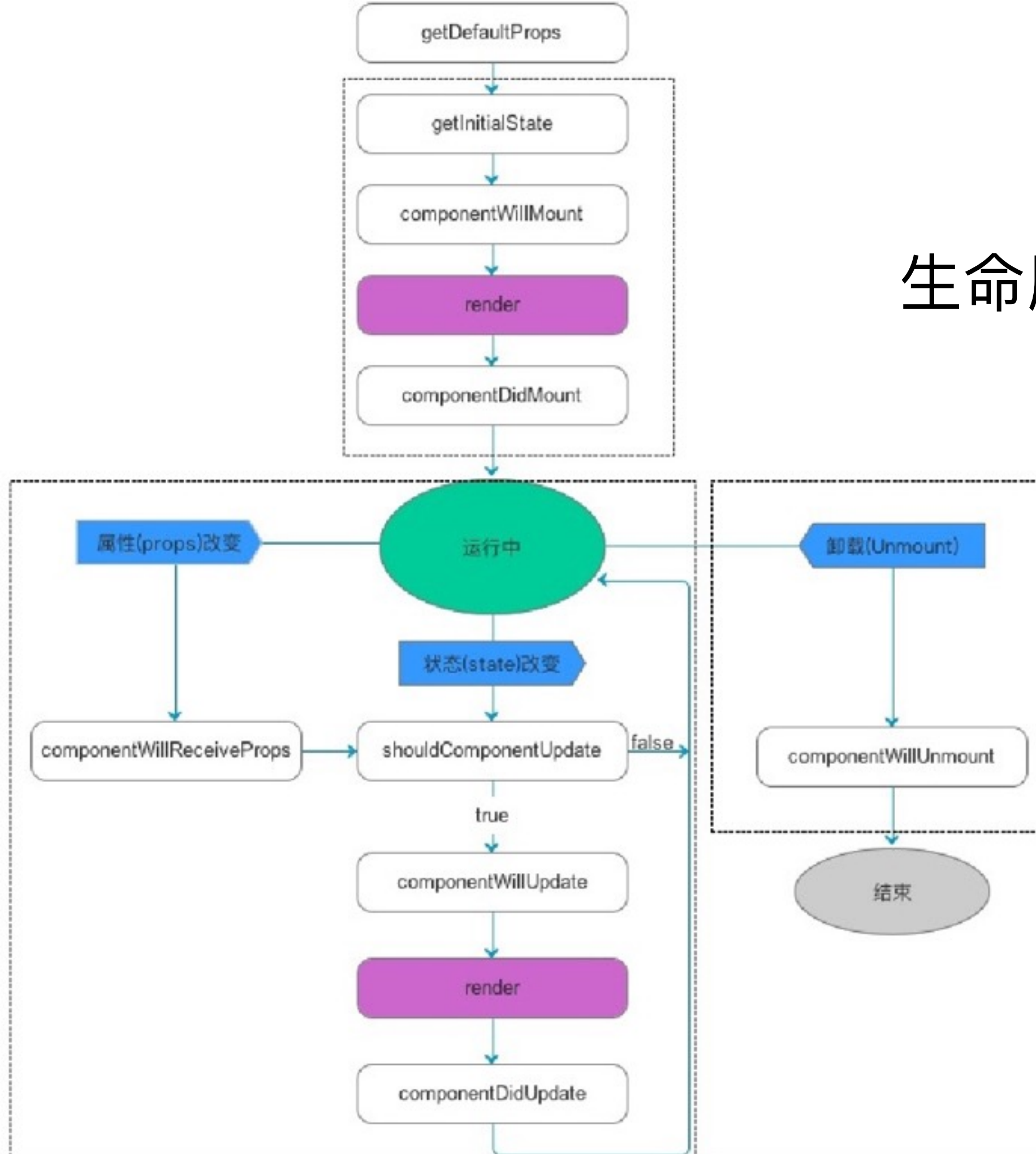
JSX 做的事情就是根据 state 和 props 中的值，结合一些视图层面的逻辑，输出对应的 DOM 结构。

传递prop

State Machines

状态机

生命周期



布局篇

宽度单位和像素密度

flexbox

水水平垂直居中

嵌套的网格



android framelayout

`alignItems` 和 `justifyContent`

水平居中



垂直居中



水平垂直居中



写——一个Project

——以android举例

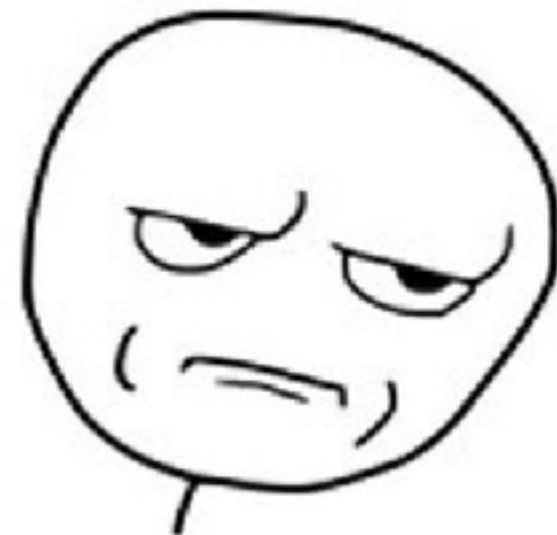
50%的人人死在了配置上。。

T_T

这其中，——一半死在了翻不了墙，翻得了太慢的卡了两天没动静

native开发死在配react上

react开发死在配native上



最小小安装子子集 (从前的)

necessary

homebrew
nvm→node
watchman
jdk
android sdk
simulator
gradle

choice

flow
android studio
webstorm
atom+nuclide
virtualbox+
genymotion

其他推荐插件

除了 Nuclide 之外，还可以根据你的需求安装其他的一些插件，这里推荐一些插件：

- react: React 的语法补全和智能重排；
- react-snippets: React 的代码段；
- highlight-selected: 高亮当前双击选中的标记；
- jshint: 检查 JavaScript 的语法，支持 JSX（需要在插件设置中开启 Support Linting JSX）；
- emmet: 用 emmet（Zen Coding）方式快速编写页面；
- save-session: 让 Atom 记住上一次打开的会话；
- browser-plus: 在 Atom 中内嵌一个浏览器窗口，方便页面调试（其实 Atom 本身就是一个浏览器）；
- minimap: 如果你对 Sublime Text 的 minimap 念念不忘；
- atomic-emacs: Emacs 键盘布局，适合 Emacs 用户使用；
- vim-mode: Vim 键盘布局，适合 Vim 用户使用。

reference

<http://react-native.cn/docs/getting-started.html#content>

<http://www.bingjie.me/2015/11/25/react-native-android.html>

从原生入口入口

MainActivity→index.android.js

AppDelegate→index.ios.js

常见错误等

Failed to find Build Tools revision 23.0.1

这个是gradle没编译通过，打开android studio去更新它。

下载太慢：

Npm install要把代理关掉并且设置

```
npm config set registry http://registry.cnpmjs.org
```

Cannot connect to the Server:

```
adb reverse tcp:8081 tcp:8081
```

这句话是用来连接JS服务器的

工具

Windows + Android

- 必须的软件
 - Chocolatey
 - Python2
 - Node
 - React-Native命令行
 - Android Studio
- SDK
- 环境变量

最终运行的时候是在命令行输入[react-native run-android](#)。

在设备上运行

- 确保打开了usb调试开关
- adb devices
 - 只能同时连接一个设备（包括虚拟机）
- Android 5.0及以上
 - 通过usb连接
- Android 5.0以下
 - 通过Wi-Fi连接
- 在电脑上查看编译后的服务器代码
 - `localhost:8081/index.android.bundle?platform=android`

Mac + iOS

- 必须的软件
 - Homebrew
 - Node
 - React-Native命令行
 - Xcode
- Watchman(实时变更)
- Flow

python自带

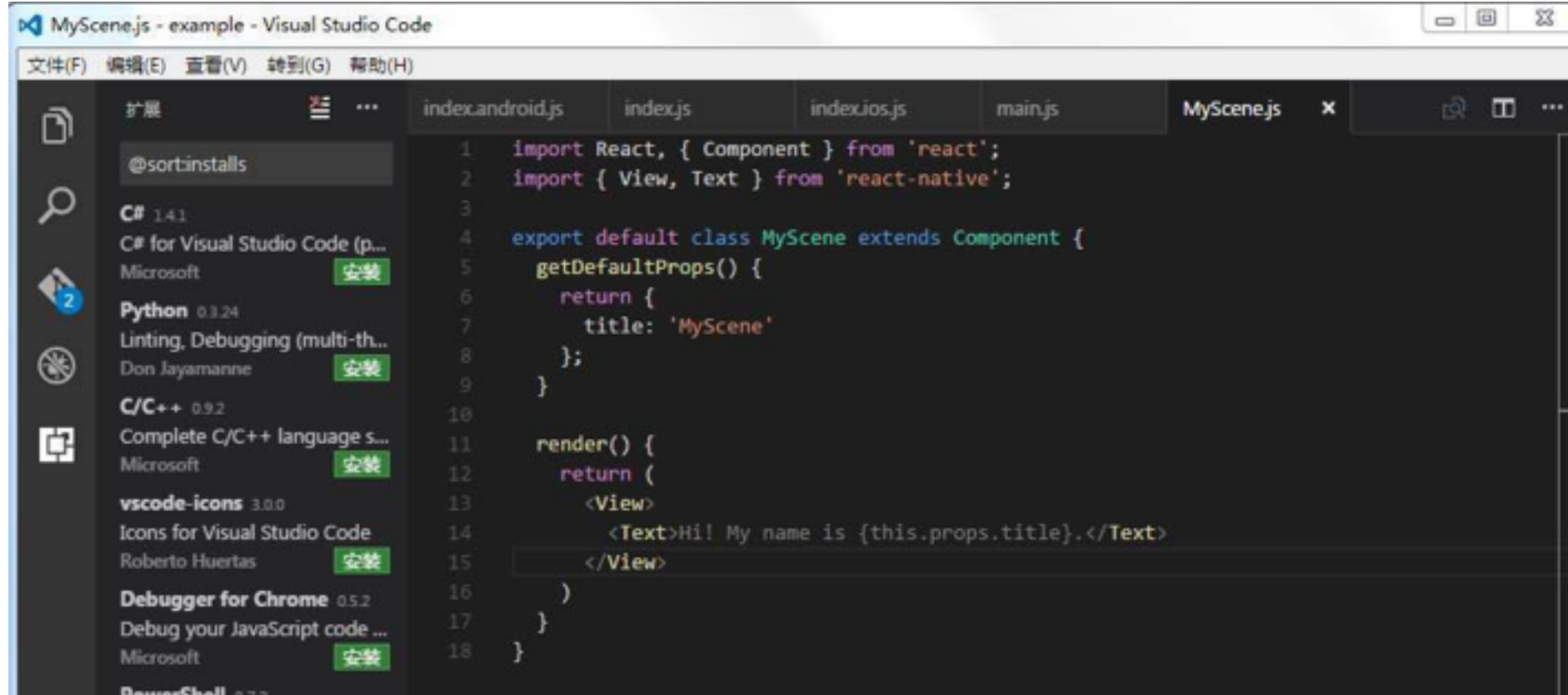
```
jsCodeLocation = [NSURL URLWithString:@"http://localhost:8081/index.ios.bundle?  
platform=ios&dev=true"];
```

VSCode

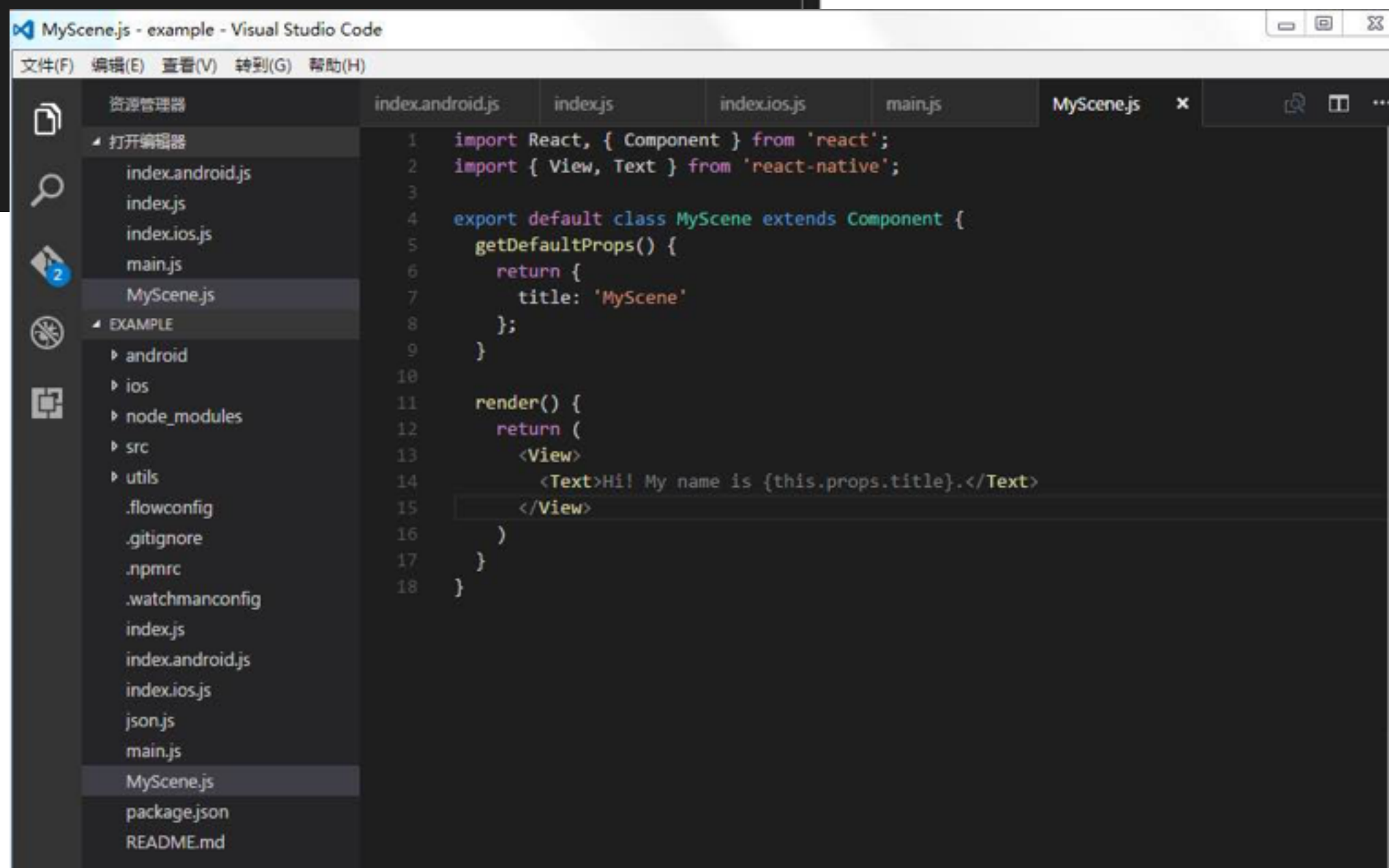
Visual Studio Code 是“巨硬”基于 Electron 开发的一款 **轻量级编辑器**，主打 web 开发，支持的语言包括：

- JavaScript
- C#
- JSON
- HTML
- Markdown
- TypeScript
- CSS, Sass, Less
- DockerFile
- Swift
- Groovy
- 不断增加中...

具体见：<http://i5ting.github.io/vsc/>

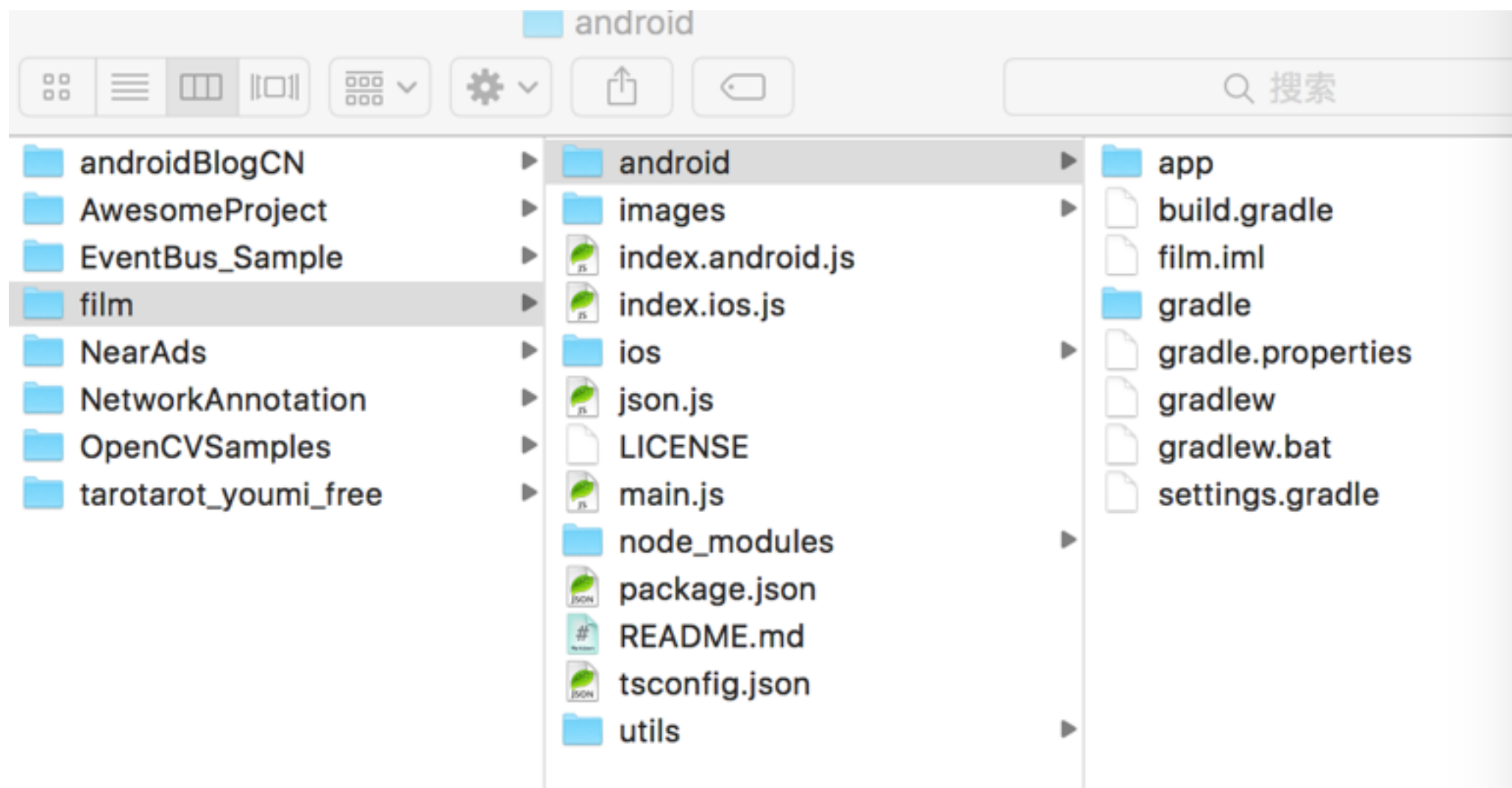


左边几个按钮



底下蓝色按钮

路径

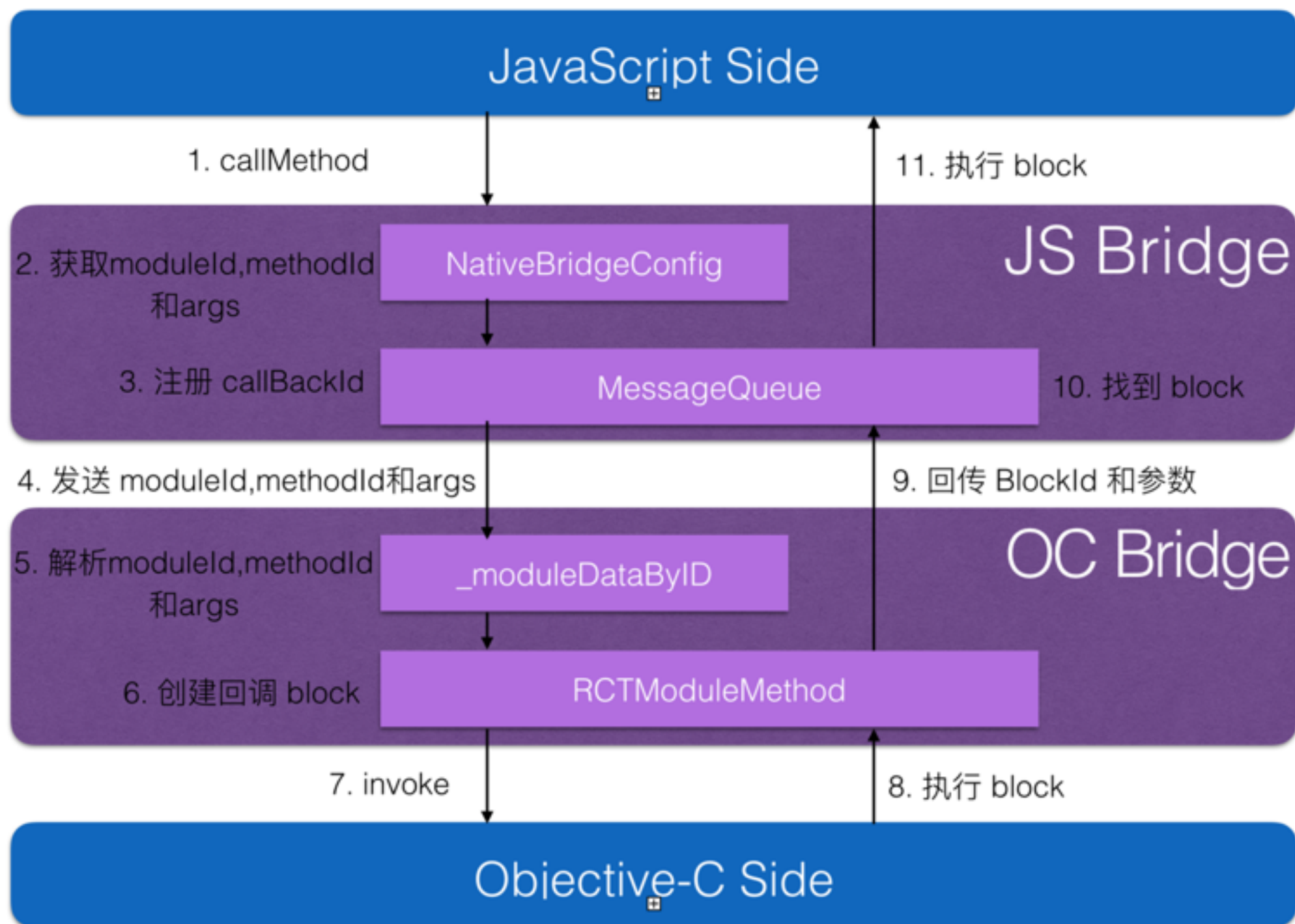


运行命令在什么底下

npm install过程主要是node modules特别大

闭包

与回调的原理



JSX

语法进阶

JS vs JSX

Code

```
// 输入 (JSX):  
var content = <Container>{window.isLoggedIn ? <Nav /> : <Login />}</Container>;  
// 输出 (JS):  
var content = React.createElement(  
  Container,  
  null,  
  window.isLoggedIn ? React.createElement(Nav) : React.createElement(Login)  
);
```

“use strict” 开启严格模式

只在 ECMAScript 5 时需要，现在6 已经不需要了

React Element

react中最主要的变量类型就是ReactElement。
它有四种属性，分别是type, props, key和ref。
这个类既没有方法也没有原型。
你可以通过createElement来创建一个对象：

```
var root = React.createElement( 'div' );
```

就是说右边表达式返回值不用写 ReactElement，也不能直接使用这个类。

```
var child = React.createElement('li', null, 'Text Content');
var root = React.createElement('ul', { className: 'my-list' }, child);
ReactDOM.render(root, document.getElementById('example'));
```

JS语法是将React当作一个类，createElement当作一个静态方法来调用的。

这个方法内有三个参数。

第一个参数是新建的DOM类型，第二个参数是传递给这个DOM的属性值，第三个参数是该DOM的子DOM。

(所以DOM的结构是如此层层连通的)

```
var root = <ul className="my-list">
    <li>Text Content</li>
</ul>;
ReactDOM.render(root, document.getElementById('example'));
```

这个是JSX的不同写法。

React Component

```
import React, { Component } from 'react';
import { View, Text } from 'react-native';

export default class MyScene extends Component {
  getDefaultProps() {
    return {
      title: 'MyScene'
    };
  }

  render() {
    return (
      <View>
        <Text>Hi! My name is {this.props.title}</Text>
      </View>
    );
  }
}
```

```
var MyComponent = React.createClass({  
  render: function() {  
    ...  
  }  
});
```

When this constructor is invoked it is expected to return an object with at least a **render** method on it. This object is referred to as a `ReactComponent`.

```
var component = new MyComponent(props); // never do this
```

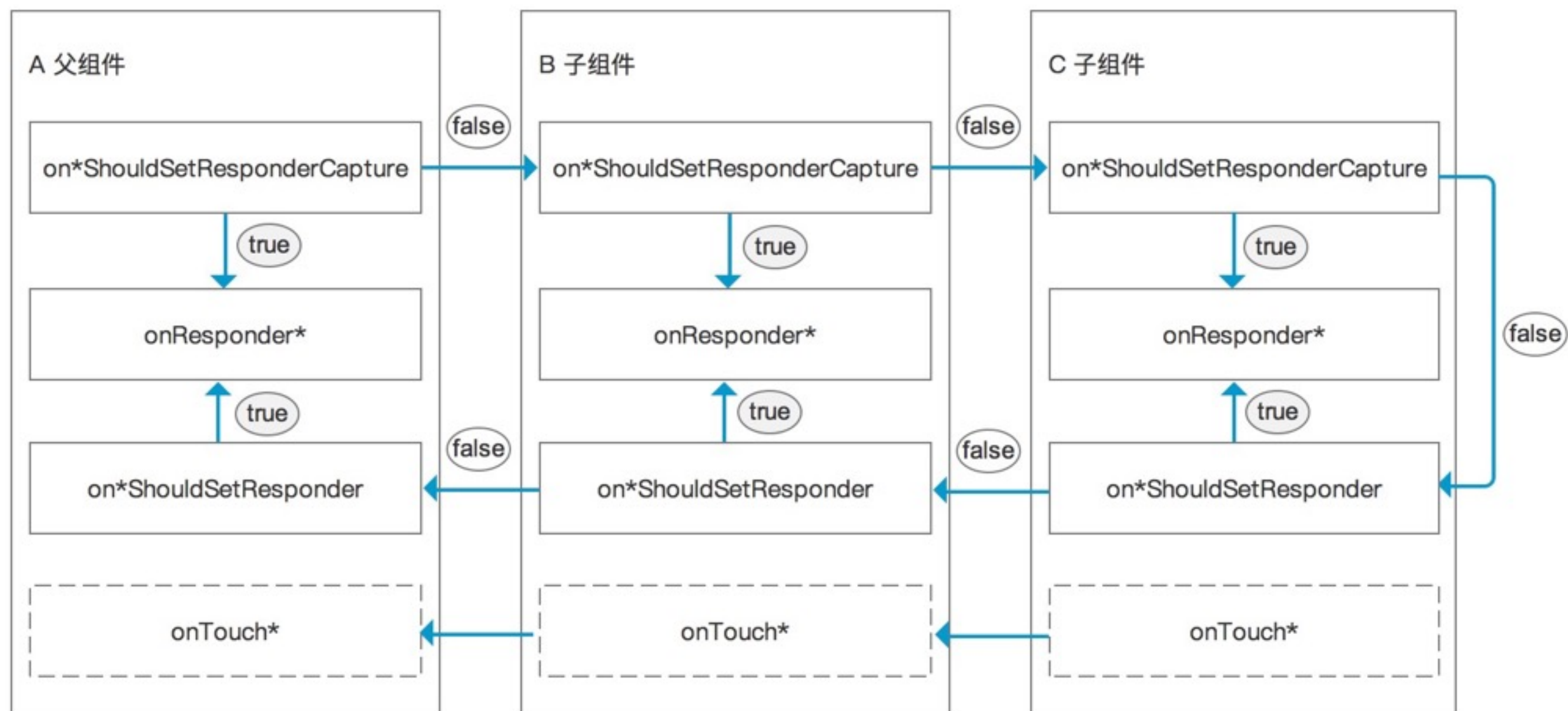
不需要使用构造函数。React会自动引入组件。Components

```
var element = React.createElement(MyComponent);
```

OR using JSX:

```
var element = <MyComponent />;
```


详解触摸事件



发布

```

public class MainActivity extends Activity {

    private static final String JSBUNDLE_FILE = "ReactNativeDevBundle.js";

    private static void copyFile(InputStream in, OutputStream out) throws IOException {
        byte[] buffer = new byte[1024];
        int read;
        while((read = in.read(buffer)) != -1){
            out.write(buffer,0,read);
        }
    }

    private void prepareJSBundle() {
        File targetFile = new File(getFilesDir(),JSBUNDLE_FILE);
        if (!targetFile.exists()) {
            try {
                OutputStream out = new FileOutputStream(targetFile);
                InputStream in = getAssets().open(JSBUNDLE_FILE);

                copyFile(in,out);
                out.close();
                in.close();
            } catch (FileNotFoundException e) {
                e.printStackTrace();
            } catch (IOException e) {
                e.printStackTrace();
            }
        }
    }

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        prepareJSBundle();
        ...
    }
}

```

在线更新

第一次打开 React Native 应用的时候，
会连接 JS Server 下载一个 ReactNativeDevBundle.js 文件，然后放到应用数据的 files 目录下，就能运行这个 JS 文件了

只要通过网络下载并替换这个 JS 文件，就可以实现 APP 的更新。
不需要下载 APK 包更新，也不需要市场发布，
只要后台上线一个新的 JS，客户端就能立即更新了。
这就绕过了应用市场，解决了应用更新困难的问题，修复
BUG 一秒上线，新 Feature 一秒到达用户

在RN中，
JS工程师解决的问题是：
[将基本组件拼装成可用用的React组件]
Native工程师解决的问题是：
[提供核心组件，足够的扩展性、灵活性和性能]

猫眼电影


iPhone 6s - iPhone 6s / iOS 9.1 (13B5110e)

Carrier 3:27 AM


猫眼电影

< 返回


分歧者3：忠诚世界




美国队长3 3D/IMAX 3D/IMAX
动作,科幻,冒险
铁人打美队, 群雄来站队
今天28家影院放映380场
8.9分




百鸟朝凤 2D
历史,音乐,剧情
两代手艺人, 唢呐感情深
今天25家影院放映129场
9.4分




超脑48小时 2D/3D
悬疑,动作,剧情
探员遭意外, 罪犯被委派
今天27家影院放映148场
8.3分



愤怒的小鸟 3D
动画,奇幻,冒险
绿猪来偷蛋, 鸟儿群奋战
本周五上映
116916人想看



北京遇上西雅图之不二情书 2D
喜剧,爱情,剧情
异国心未远, 书信寄情缘
今天26家影院放映116场



分歧者3：忠诚世界 2D
56460人想看
科幻,动作,冒险
美国/120分钟
本周五上映

立即购票

影片讲述在推翻了派系制度之后，翠丝（谢琳·伍德蕾 饰）与老四（提奥·詹姆斯 饰）带领众人翻越了芝加哥围墙来到全新世界。然而看似美好的新世界却隐藏着杀机，翠丝与老四之间的信任也遭受到前所未有的考验。二人能否从阴谋中脱身并守住彼此？

演员表
谢琳·伍德蕾 提奥·詹姆斯 李美琪 娜奥米·沃茨 迈尔斯·安塞尔·埃尔格特 杰夫·丹尼尔斯 阿克塔维亚·斯宾瑟 卡斯加德 佐伊·克拉维兹 乔尼·维斯顿 姆拉特·安图姆·黛尔 null Zero Kazama 布拉德利·鲍文 阿维森特 Brandon Tabassi null 德韦恩·博伊德 Scott Poythress null 斯科特·莱德贝特 尼克·马德里克 布莱恩·冈萨雷斯 克莱·杰弗里斯 艾米·帕里什 谢比·汤森 null null Brian Neal 简·希淇

短评


iPhone 6s - iPhone 6s / iOS 9.1 (13B5110e)

Carrier 3:42 AM

猫眼电影

< 返回

分歧者3：忠诚世界



分歧者3：忠诚世界 2D
56460人想看
科幻,动作,冒险
美国/120分钟
本周五上映

立即购票

影片讲述在推翻了派系制度之后，翠丝（谢琳·伍德蕾 饰）与老四（提奥·詹姆斯 饰）带领众人翻越了芝加哥围墙来到全新世界。然而看似美好的新世界却隐藏着杀机，翠丝与老四之间的信任也遭受到前所未有的考验。二人能否从阴谋中脱身并守住彼此？

演员表
谢琳·伍德蕾 提奥·詹姆斯 李美琪 娜奥米·沃茨 迈尔斯·安塞尔·埃尔格特 杰夫·丹尼尔斯 阿克塔维亚·斯宾瑟 卡斯加德 佐伊·克拉维兹 乔尼·维斯顿 姆拉特·安图姆·黛尔 null Zero Kazama 布拉德利·鲍文 阿维森特 Brandon Tabassi null 德韦恩·博伊德 Scott Poythress null 斯科特·莱德贝特 尼克·马德里克 布莱恩·冈萨雷斯 克莱·杰弗里斯 艾米·帕里什 谢比·汤森 null null Brian Neal 简·希淇

短评

iPhone 6s - iPhone 6s / iOS 9.1 (13B5110e)

Carrier 3:28 AM

思明区

17.5今典影城(中华城店) 座
思明区思明南路195号中华城5楼（近镇海路）

博纳国际影城(磐基店) 座
思明区嘉禾路199号磐基中心4层

金逸国际影城(明发IMAX店) 座
思明区明发商业广场B区二楼百安居后侧

金逸影城(明发店) 座
思明区嘉禾路莲坂明发商业广场东区381号3楼

金逸影城(名汇店) 座
思明区霞溪路28号名汇广场3楼

金逸影城(文艺店) 座
思明区体育路95号文化艺术中心科技馆一楼

思明电影院 座

影片 影院 我 影片 影院 我 影片 影院 我

Q&A