

Reactive Web 应用 UI 架构的演进



DOM 操作

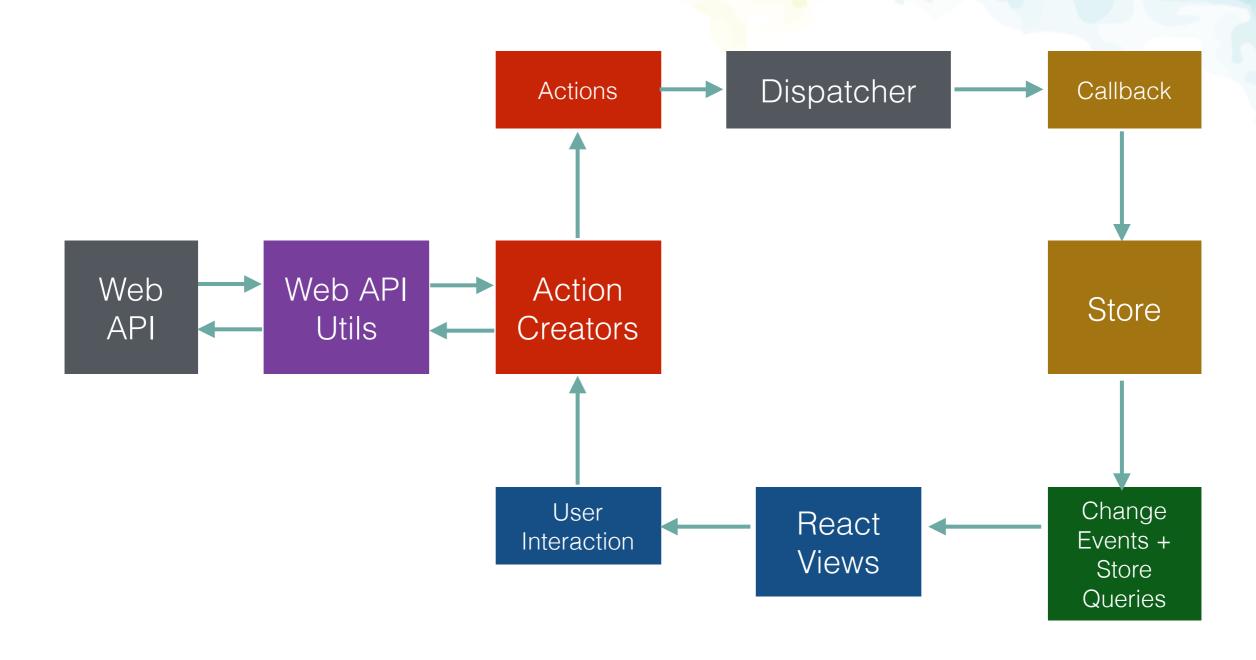
MV*

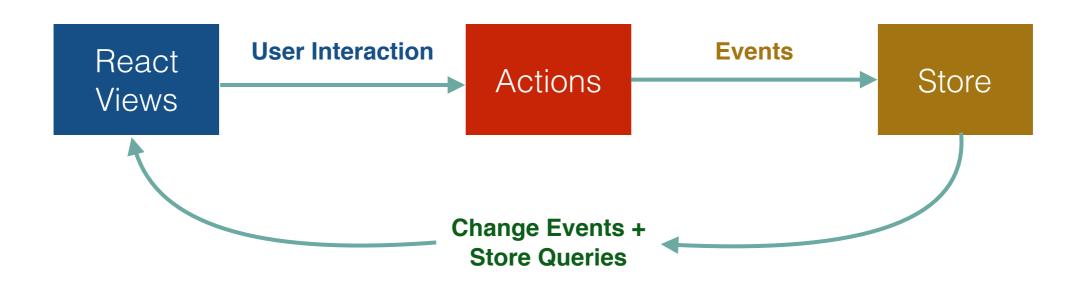
Flux

Data Driven Architecture

2008年

2015年







单向数据流

Change Events

React Views

Action

Store?







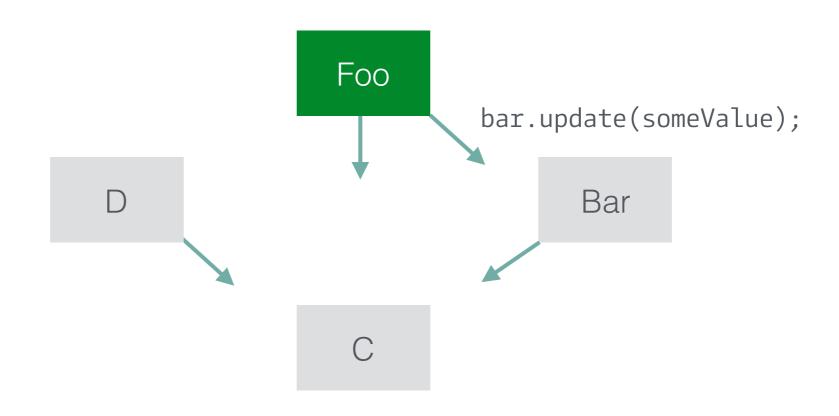
事件可被监听的组件 Observable Component

数据驱动开发 Data Driven Development



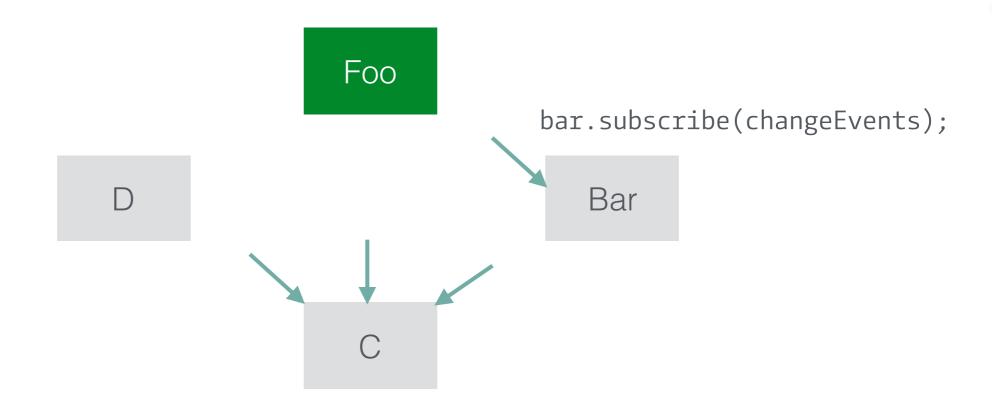
$$def A = B + C$$

Interactive Programming



模块X的状态的修改由其他模块触发

Reactive Programming



模块X的状态的修改由X管理





方法

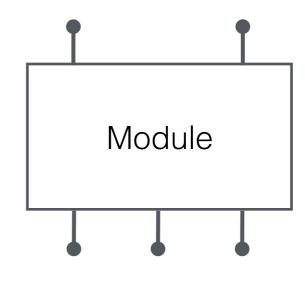
输入 INPUT



OUTPUT 输出

组件

输入 INPUT



OUTPUT 输出

系统 View C View B View A Action A Action B Action C Store B Store C Store A

多值操作

Array

res = resources .filter(q => q.age > 5) .map(q => q.name)

Observable

```
res = resources
.filter(q => q.age > 5)
.map(q => q.name)
```

res.subscribe(x =>

单值操作

Τ

$$var y = f(x)$$

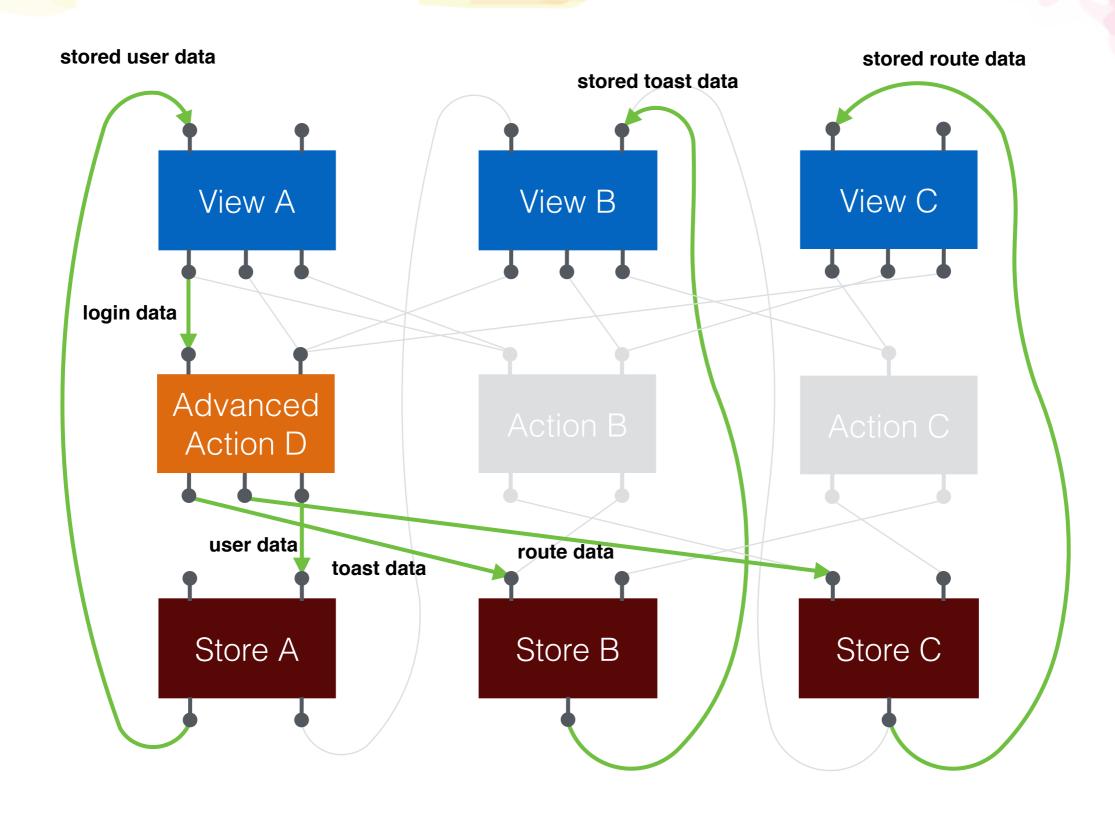
 $var z = g(x)$

Promise

同步

异步





Reactive Programming

https://en.wikipedia.org/wiki/Reactive_programming

Reactive Manifesto

www.reactivemanifesto.org

Reactive MVC and the Virtual DOM

http://futurice.com/blog/reactive-mvc-and-the-virtual-dom/

The Reactive Extensions for JavaScript

http://reactivex.io

Reactive React - Tiny and Pure Flux Architecture Implementation

https://github.com/winsonwq/RR

