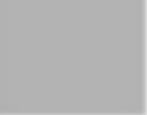
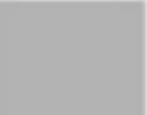


React和函数式编程

歌者科技 – 王楚安

Shopping Cart

Items in your cart

Product	Qty	Total	
	1	\$50.00	Remove
	1	\$150.00	Remove
	2	\$200.00	Remove

[Update cart](#)

[Continue shopping](#)

Subtotal

\$250.00

[Checkout](#)

一个案例

“The web is designed for documents,
not apps”

so we use javascript to operate it!

```
this.$el.find('#delete').on('click', function() {  
    // call addItem  
});
```

```
this.$el.find('#qty').on('change', function() {  
    // call changeQty  
});
```

让电脑忙一点，我们可以轻松一点

Compare Search terms ▾

react.js
Search term

angular.js
Search term

vue.js
Search term

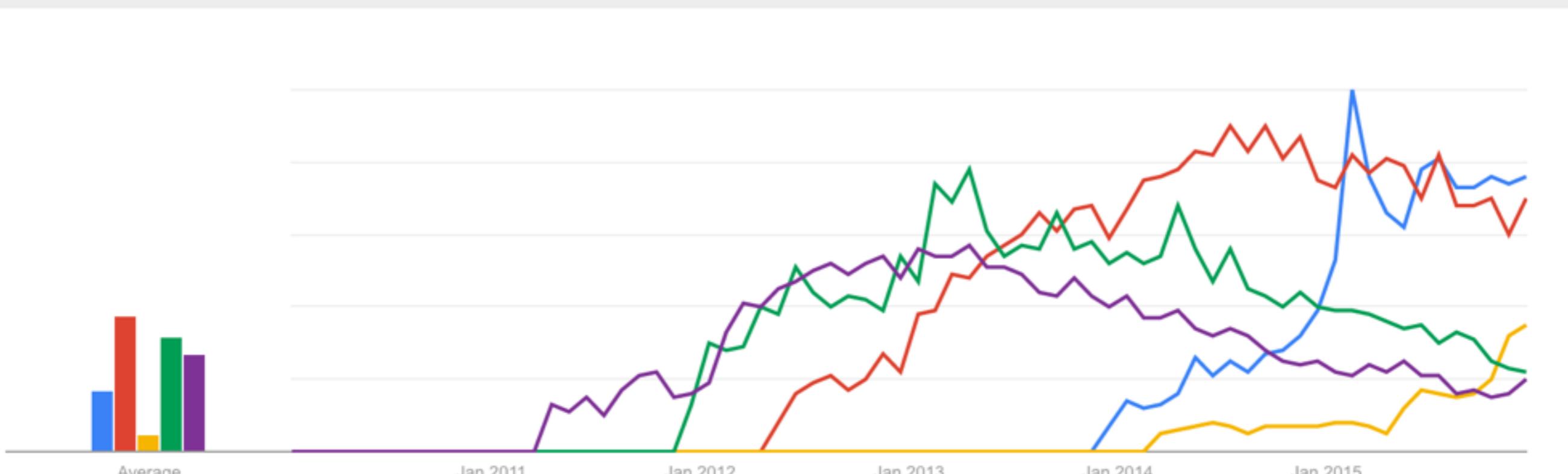
ember.js
Search term

knockout...
Search term

Interest over time ?

News headlines ?

Forecast ?



</>

```
var HelloMessage = React.createClass({  
  render: function() {  
    return <div>Hello {this.props.name}</div>;  
  }  
});  
  
ReactDOM.render(<HelloMessage name="John" />, mountNode);
```

初印象



React

Docs

Support

Download

Blog

GitHub

React Native

React

A JAVASCRIPT LIBRARY FOR BUILDING USER INTERFACES

Get Started

Download React v0.14.3

JUST THE UI

Lots of people use React as the V in MVC. Since React makes no assumptions about the rest of your technology stack, it's easy to try it out on a small feature in an existing project.

VIRTUAL DOM

React abstracts away the DOM from you, giving a simpler programming model and better performance. React can also render on the server using Node, and it can power native apps using [React Native](#).

DATA FLOW

React implements one-way reactive data flow which reduces boilerplate and is easier to reason about than traditional data binding.

UI主要干了什么?
显示界面
获得交互

课程详情



1200人看过

益智 科学幻想机器人智能体验课程

¥180

¥280

全网最低价 差价双倍赔

所属机构



剑桥国际少儿英语

1.5km

沈阳市皇姑区松花江街岐山路

查看该机构更多课程 >

课程信息

课程周期：一学期，共24课时

课程时间：周六周日，任选一天

x Headers Preview Response Cookies Timing

```
▼ {errno: 0, error: null, ts: 1449851666177, id: 2796, shopId: 1  
  address: "皇姑区崇山西路1-10号1门 (市委党校正对面)"  
  allowCoupon: 0  
  ▼ banners: ["http://e.shouxiner.com/ins/pic/10273321.jpeg", "http://e.shouxiner.com/ins/pic/10273321.jpeg"  
    0: "http://e.shouxiner.com/ins/pic/10273321.jpeg"  
    1: "http://e.shouxiner.com/ins/pic/10273322.jpeg"  
    2: "http://e.shouxiner.com/ins/pic/10273323.jpeg"  
  cdate: "12节课,90分钟/节"  
  comments: []  
  content: "以校内教材为基础,针对数学基础较差的学生同步开班,在讲解校内  
  coupon: "0.00"  
  courseMode: "小班"  
  ctime: "每周一节课,周一至周五16:30--19:00,周六至周日10:00--19:00"  
  distance: 3692  
  errno: 0  
  error: null  
  expire: "2015-12-31"  
  id: 2796  
  instructions: ""  
  ▼ logo: "http://e.shouxiner.com/ins/logo/10273321.jpeg"  
    0: "http://e.shouxiner.com/ins/logo/10273321.jpeg"  
  pu: null  
  phones: "02431535913"  
  price: "600.00"  
  pricePromise: 1  
  priceSale: "500.00"  
  pv: 261  
  quota: 30  
  refundPolicy: 1  
  returnCash: "0.00"  
  saled: 20  
  shopId: 1027332  
  shopName: "沈阳诺亚教育培训学校"  
  shopTel: "02431535913"  
  start: "2015-08-31"  
  status: 1  
  ▼ subject: {type: 2, level: 2, id: 204, name: "奥数", parentId:  
    id: 204  
    level: 2}
```

纯函数：

给定一个输入，一定会
得到一个特定的输出

```
// f(x) = x + 1;  
var increment = (x) => {  
    return x + 1;  
}
```

```
// f(x) = 2^x * x^3 - x - 1;  
var complex = (x) => {  
    return Math.pow(2, x)  
        * Math.pow(x, 3)  
        - x - 1;  
}
```

- 可预测：好测试
- 没有副作用：一次只关心一件事



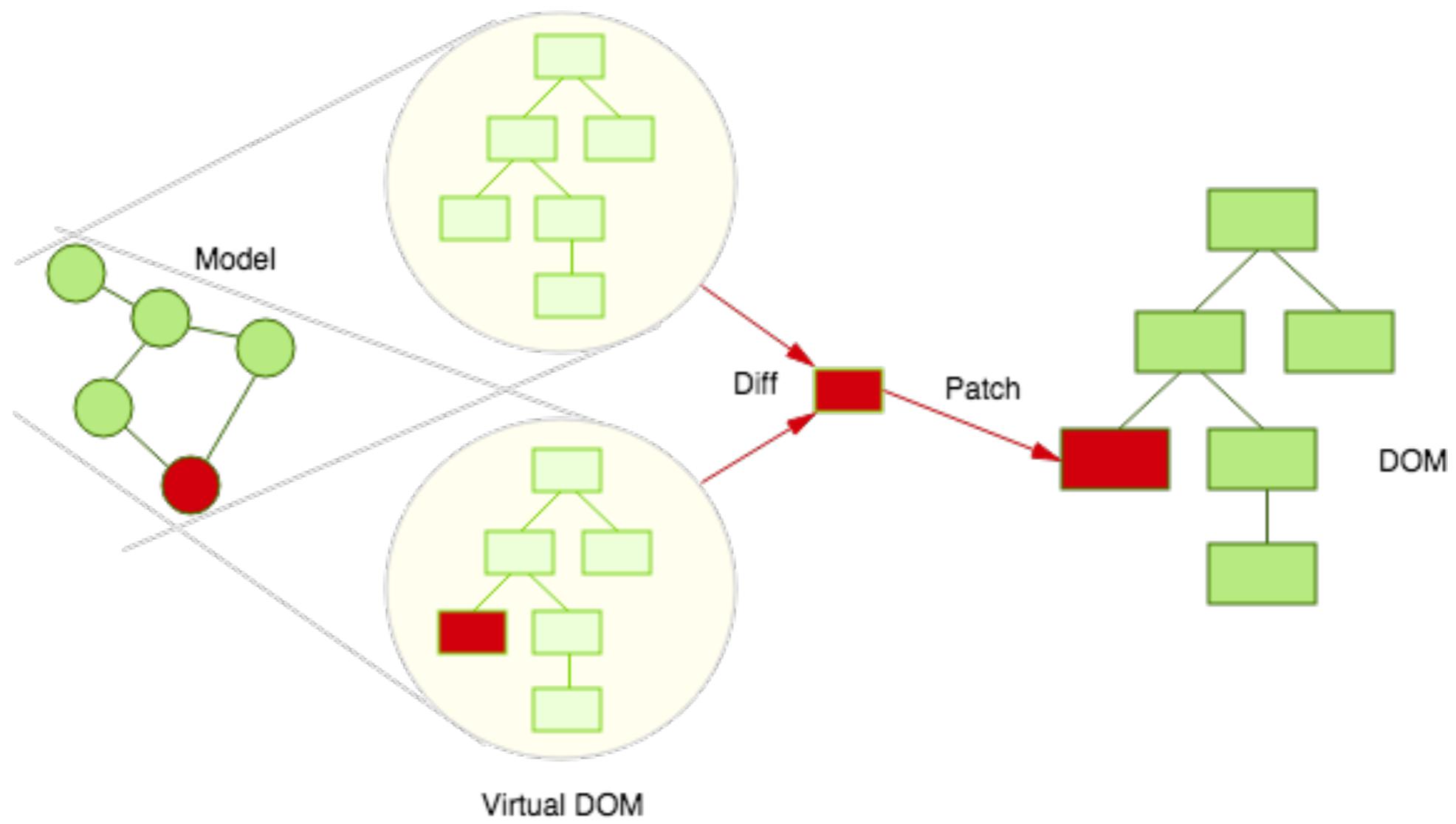
组件化
可重用
可替换
好维修

Web开发的环境很复杂：
数据会变，设计会变，用
户输入，元素被修改……

Web开发也曾经很简单
90年代：刷新

Re-Render Every Time:

所有的数据都是同步的
不需要数据绑定
不需要dirty checking
不需要操作元素



Virtual DOM 技术

高
效
率
不
脏
手
服
务
端
也
能
跑

展现层可以变： react-art react native

```
render: function() {
  return (
    <div>
      <h3>TODO</h3>
      <TodoList items={this.state.items} />
      <form onSubmit={this.handleSubmit}>
        <input onChange={this.onChange} value={this.state.text} />
        <button>'Add #' + (this.state.items.length + 1)</button>
      </form>
    </div>
  );
}
```

JSX

看起来是HTML，其实是JS：不需要再学一门模版语法了

Separation of Concern
Not Separation of
technology

Lifecycle:
Mounting and Unmounting

`getDefaultProps()`

`getInitialState()`

`componentWillMount()`

`render()`

`componentDidMount()`

`componentWillUnmount()`

Update:
New Props or State

`componentWillReceiveProps()*`

`shouldComponentUpdate()`

`componentWillUpdate()`

`render()`

DOM Mutations Complete

`componentDidUpdate()`

**Called only with new props*

数据往下传：显示界面
事件往上传：获得交互

单向数据流第一层：
数据到UI

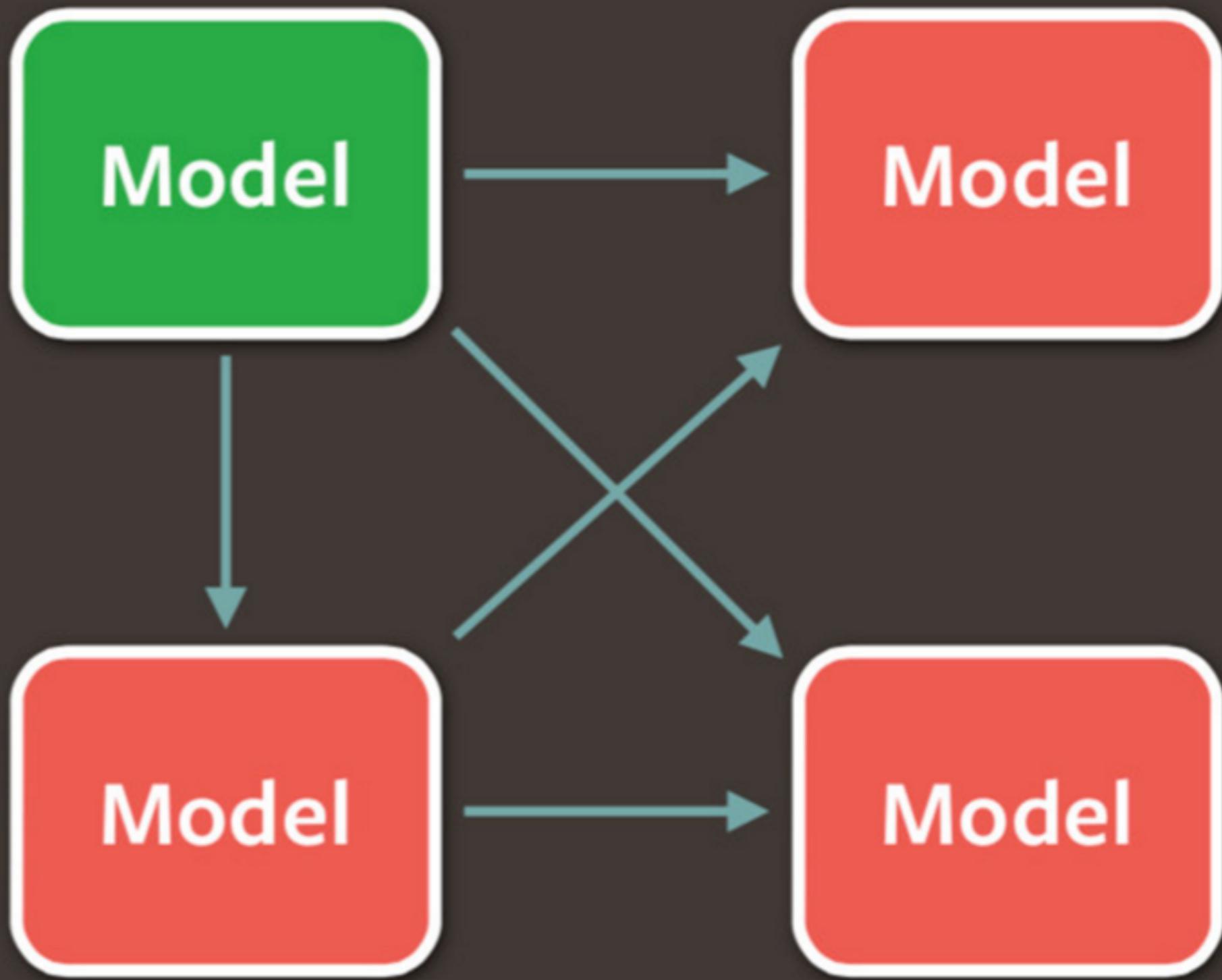
Hot Reloading

开发神器1

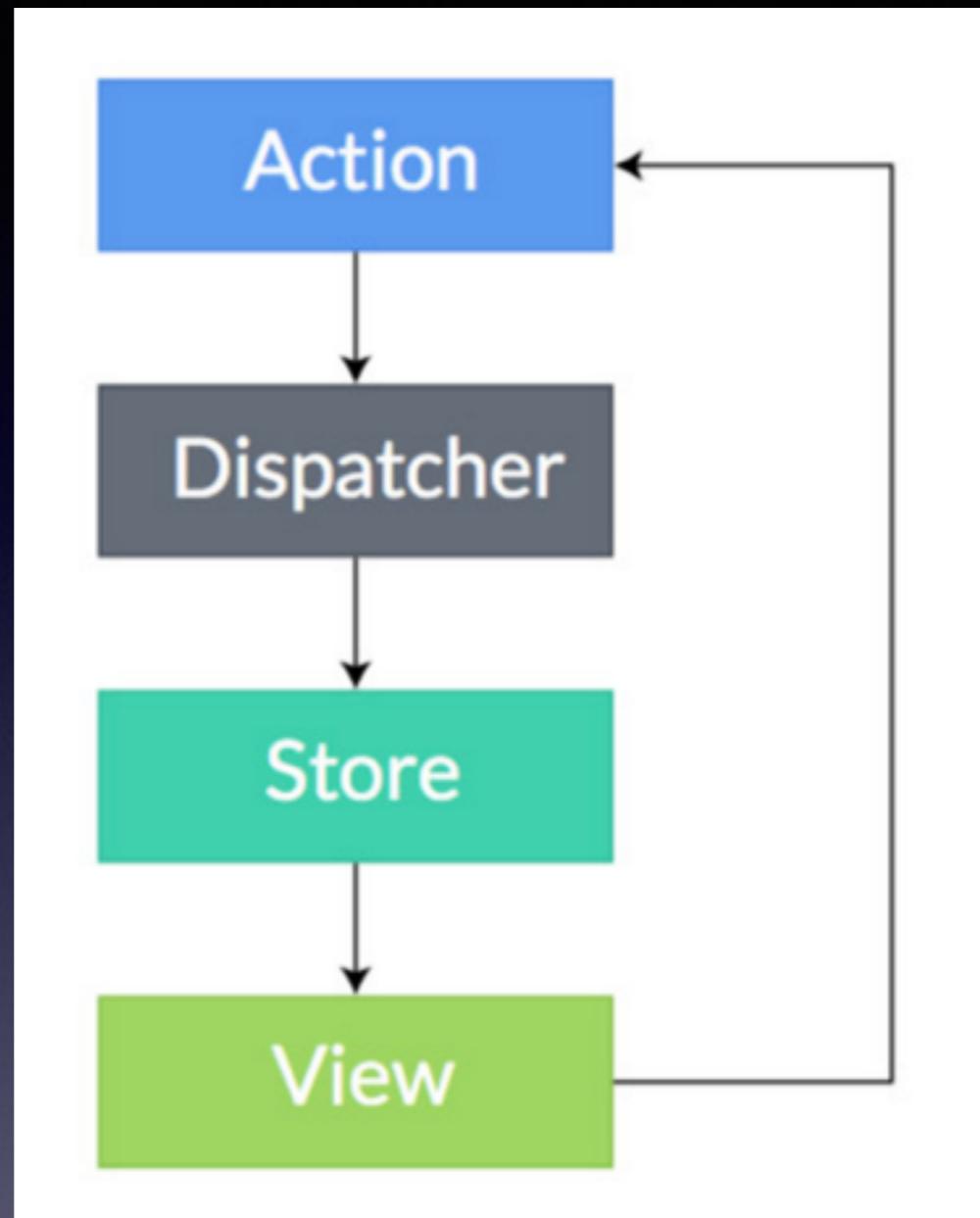
单向数据流第二层 架构

MV^{*} 的世界

MVC MVP MVVM



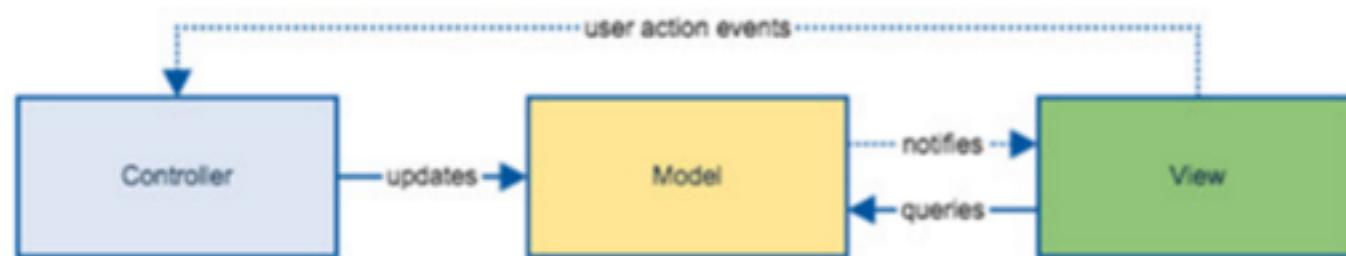




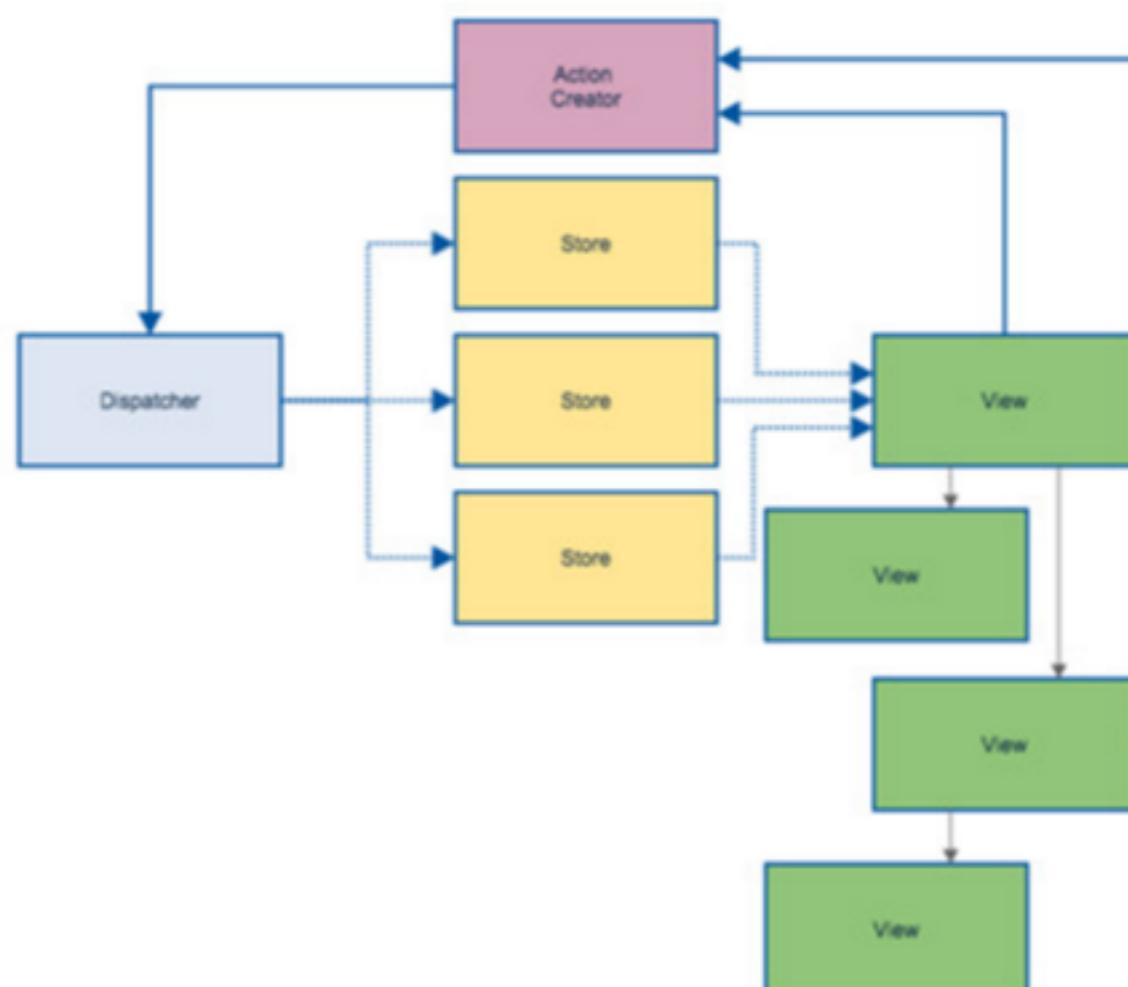
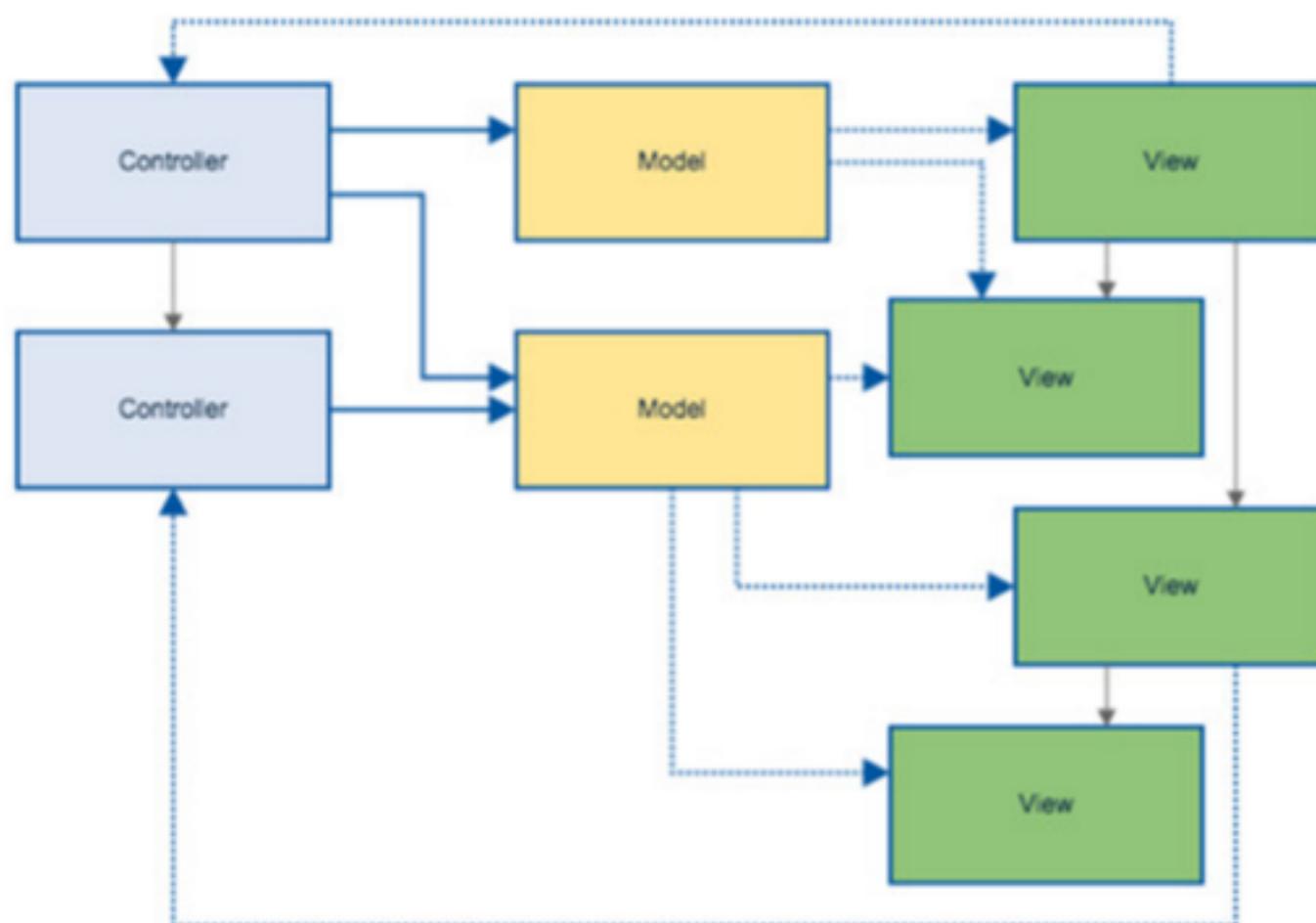
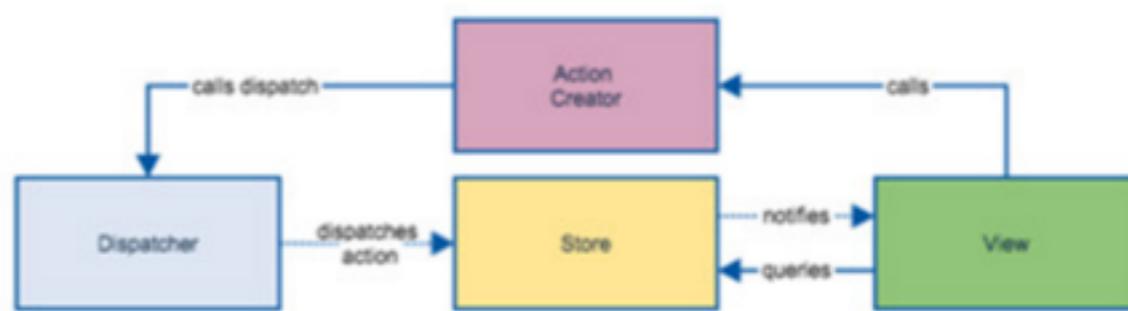
Flux

MVC and Flux

MVC



Flux



Redux

newState = reducer(previousState, action)

Redux三原则

Single source of truth

State is read-only

Mutations are written as pure functions

[API](#)[fromJS\(\)](#)[is\(\)](#)[List](#)[Map](#)[OrderedMap](#)[Set](#)[OrderedSet](#)[Stack](#)[Range\(\)](#)[Repeat\(\)](#)[Record](#)[Seq](#)[KeyedSeq](#)[IndexedSeq](#)[SetSeq](#)[Iterable](#)[KeyedIterable](#)

Immutable data encourages pure functions (data-in, data-out) and lends itself to much simpler application development and enabling techniques from functional programming such as lazy evaluation.

While designed to bring these powerful functional concepts to JavaScript, it presents an Object-Oriented API familiar to Javascript engineers and closely mirroring that of Array, Map, and Set. It is easy and efficient to convert to and from plain Javascript types.

Note: all examples are presented in [ES6](#). To run in all browsers, they need to be translated to ES3. For example:

```
// ES6
foo.map(x => x * x);
// ES3
foo.map(function (x) { return x * x; });
```

API

[fromJS\(\)](#)

Deeply converts plain JS objects and arrays to Immutable Maps and Lists.

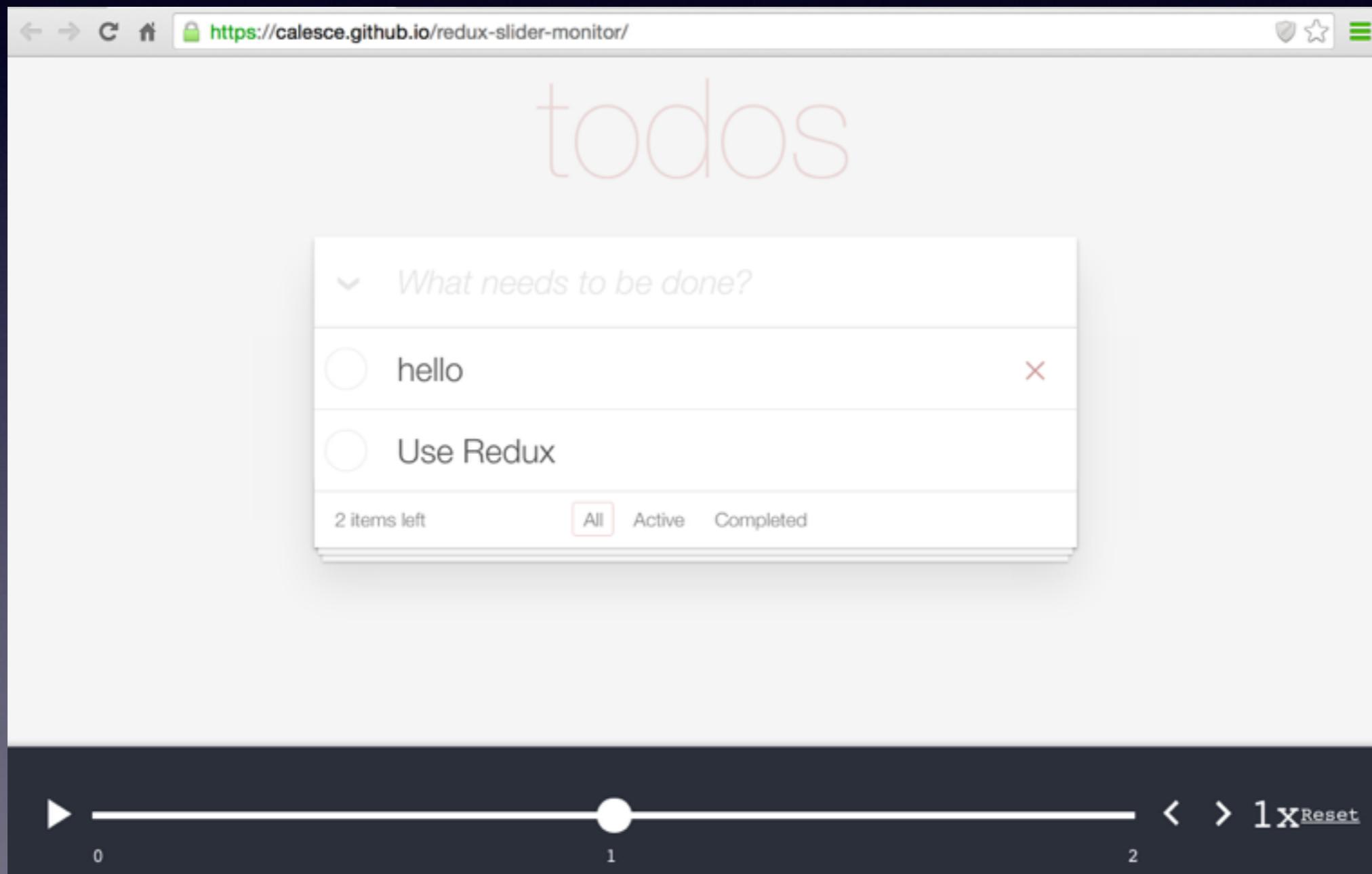
[is\(\)](#)

每次要发生修改，就产
生一个新的值

开发一个应用用到了几个变量？

Time Travel Debugging

开发神器2



更进一步

都是纯JS函数了，那跑在前端、后端后一样了
isomorphic javascript

把render放在后端

server side rendering

后端渲染给前端，前端拿到html之后接管

把store放在后端

并用socket.io把更新推送到前端

= 实时应用



歌者-王楚安

云南 昆明



扫一扫上面的二维码图案，加我微信