
WEICHUANG
React
Weiss Field Watch (White Dial)
$950.00
Canvas & Suede Backpack
$165.00
2 ITEMS IN YOUR CART
Marschall Pack
QTY: 1
$359.00
EDC Kit 2
QTY: 1
$229.00
Handmade Picking Baskets
$129.00
$45.00
SHOP NOW
CONDITION

第一节 脚手架

React搭建脚手架

1、安装node　　node –v (Node >= 6)　npm –v(5.2.0+)

2、安装react 脚手架

(sudo) npm install -g create-react-app

3、create-react-app命令来创建react项目

create-react-app react-app

cd react-app 进入项目目录

npm start 启动项目

第二节 Redux

唯创网讯
WEICHUANGWANGXUN

Redux

Redux 是 JavaScript 状态容器，提供可预测化的状态管理

首先明确一点，**Redux** 是一个有用的架构，但不是非用不可。事实上，大多数情况，你可以不用它，只用 **React** 就够了

有人说："如果你不知道是否需要 Redux，那就是不需要它。"

Redux 的创造者 **Dan Abramov**又补充说：

"只有遇到 React 实在解决不了的问题，你才需要 Redux 。"

- 用户的使用方式复杂
- 不同身份的用户有不同的使用方式（比如普通用户和管理员）
- 多个用户之间可以协作
- 与服务器大量交互，或者使用了WebSocket
- View要从多个来源获取数据

Redux

- 官网地址： **https://redux.js.org/**

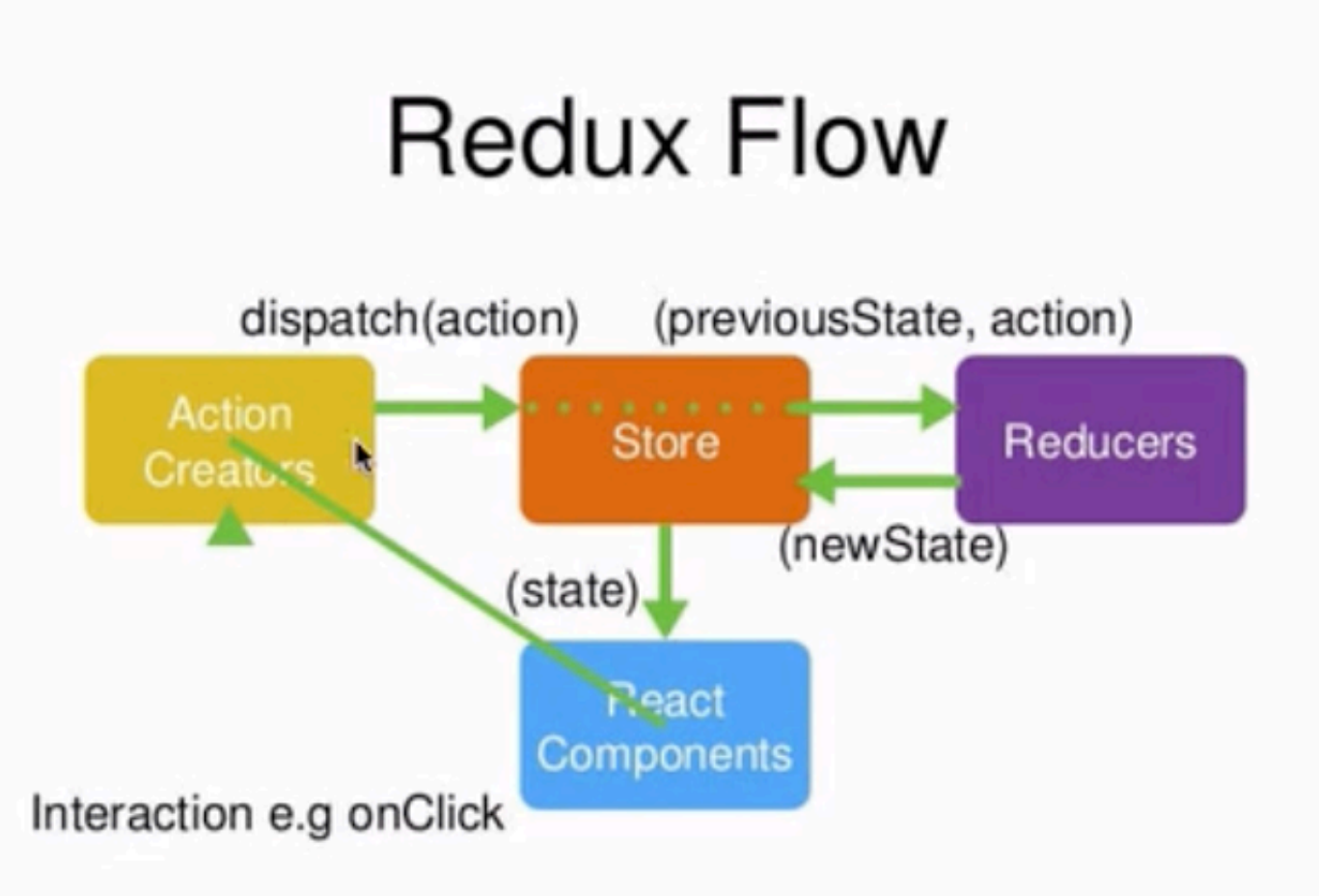
- 中文网址： **http://www.redux.org.cn/**

- 安装：

  (sudo) npm install redux react-redux

Redux

应用中所有的 **state** 都以一个对象树的形式储存在一个单一的 ***store*** 中。惟一改变 **state** 的办法是触发 ***action***，一个描述发生什么的对象。为了描述 **action** 如何改变 **state** 树，你需要编写 ***reducers***。



store

- 引入react-redux 提供的provider组件,可以让容器组件拿到state（state来自store）

<Provider store= {store}> </Provider>

- store保存数据的地方

```
import { createStore } from 'redux';

const initialState = 0;
//创建store  存储数据的
const store = createStore(()=>{

},initialState);

export default store;
```

Action

- Action本质上是 JavaScript 普通对象。我们约定，action 内必须使用一个字符串类型的 type 字段来表示将要执行的动作。多数情况下，type 会被定义成字符串常量。当应用规模越来越大时，建议使用单独的模块或文件来存放 action。

```
const addTodo = (text) => {
    return {
        type:'ADD_TODO',
        text
    }
}
export default addTodo;
```

Reducer

- Store 收到 Action 以后，必须给出一个新的 State，这样 View 才会发生变化。这种 State 的计算过程就叫做 Reducer。
- reducer 就是一个纯函数，接收旧的 state 和 action，返回新的 state。
  - combineReducers() 所做的只是生成一个函数，这个函数来调用你的一系列 reducer，每个 reducer 根据它们的 key 来筛选出 state 中的一部分数据并处理，然后这个生成的函数再将所有 reducer 的结果合并成一个大的对象

```
import { combineReducers } from 'redux';
const count = (state = 0,action)=> {
    switch (action.type){
        case 'ADD':
            return state+1;
        default:
            return state;
    }
}
const reducer = combineReducers({
    count
})
export default reducer;
```

获取state

- Connect(react-redux提供的)

  store 里能直接通过 store.dispatch() 调用 dispatch() 方法调用action，但是多数情况下你会使用 react-redux提供的 connect() 帮助器来调用。

- mapStateToProps 将state转化成props属性

```
const mapStateToProps = state => ({
    todos:state.todos
})
```

- mapStateToProps 将state转化成props属性

```
const  mapDispatchToProps = (dispatch) => {
    return {
      add: () => dispatch(add())
    };
}
```


Thank you
谢 谢 观 看