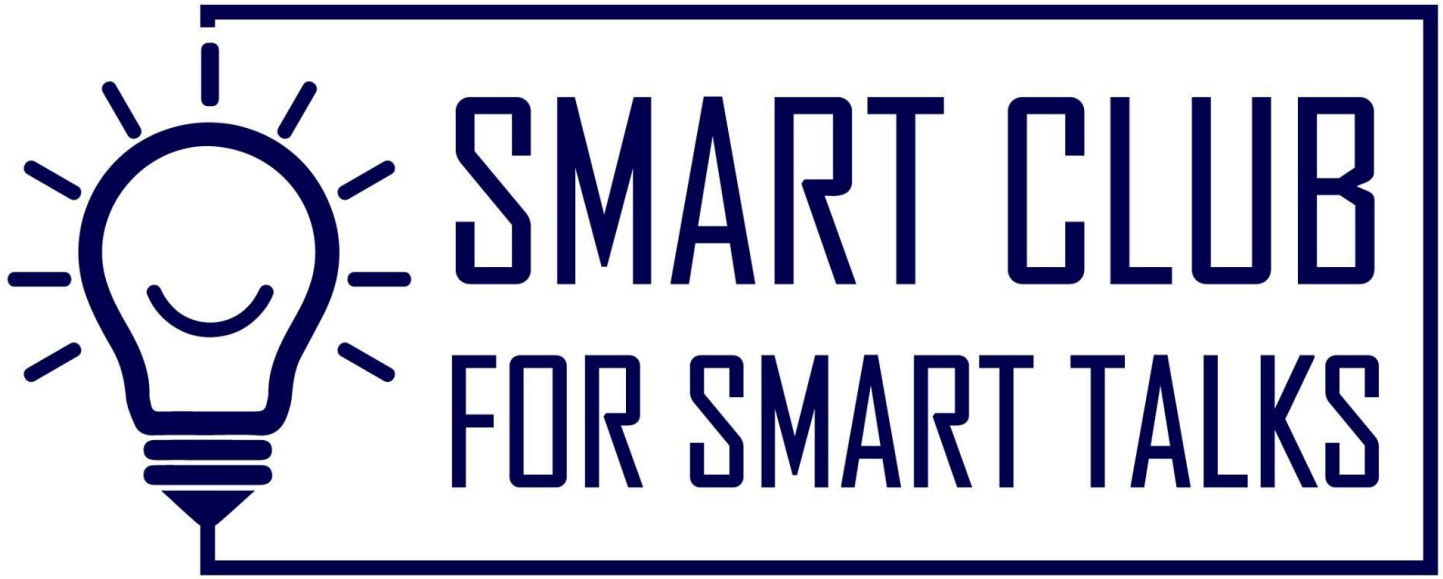
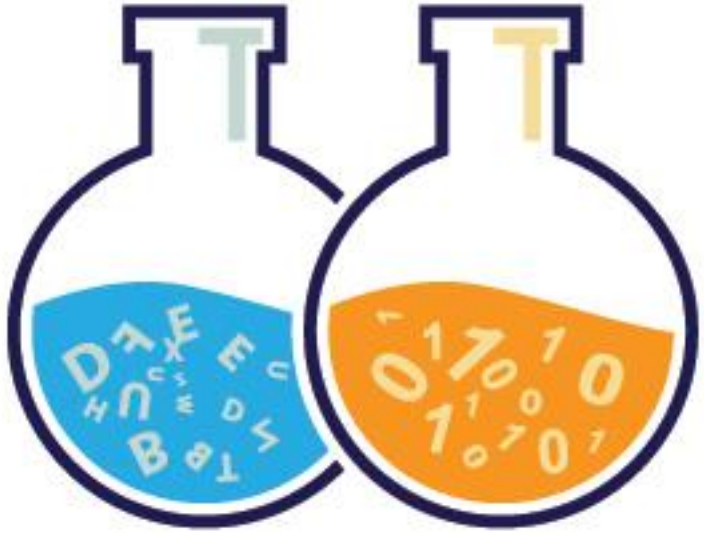


React

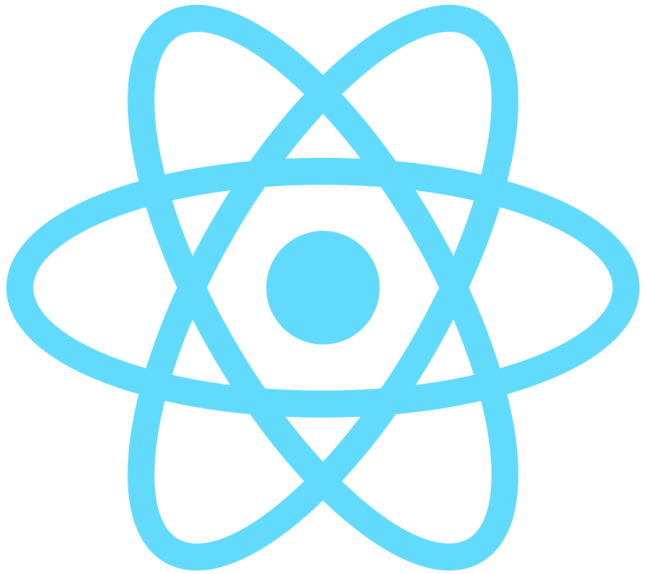
Скрытые угрозы

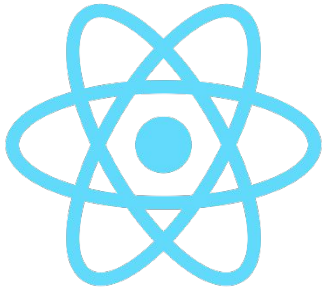


SmartJS.Academy WookieeLabs

xanf_ua

История мира JavaScript



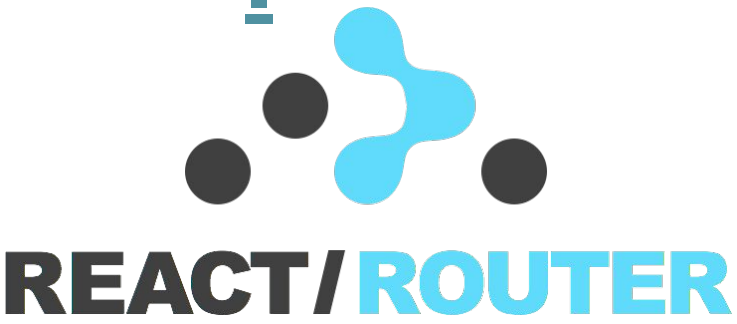


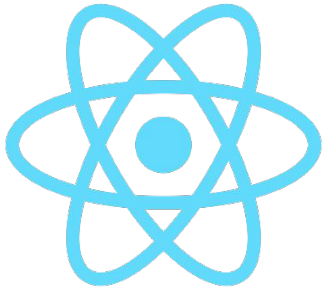
Экосистема

react-autocomplete

react-tabs

react-modal






Feature coverage

- Server-Side rendering
- Canvas / WebGL / etc.
- Mobile (React Native)
- JSX & Tools

Парадигма

Store

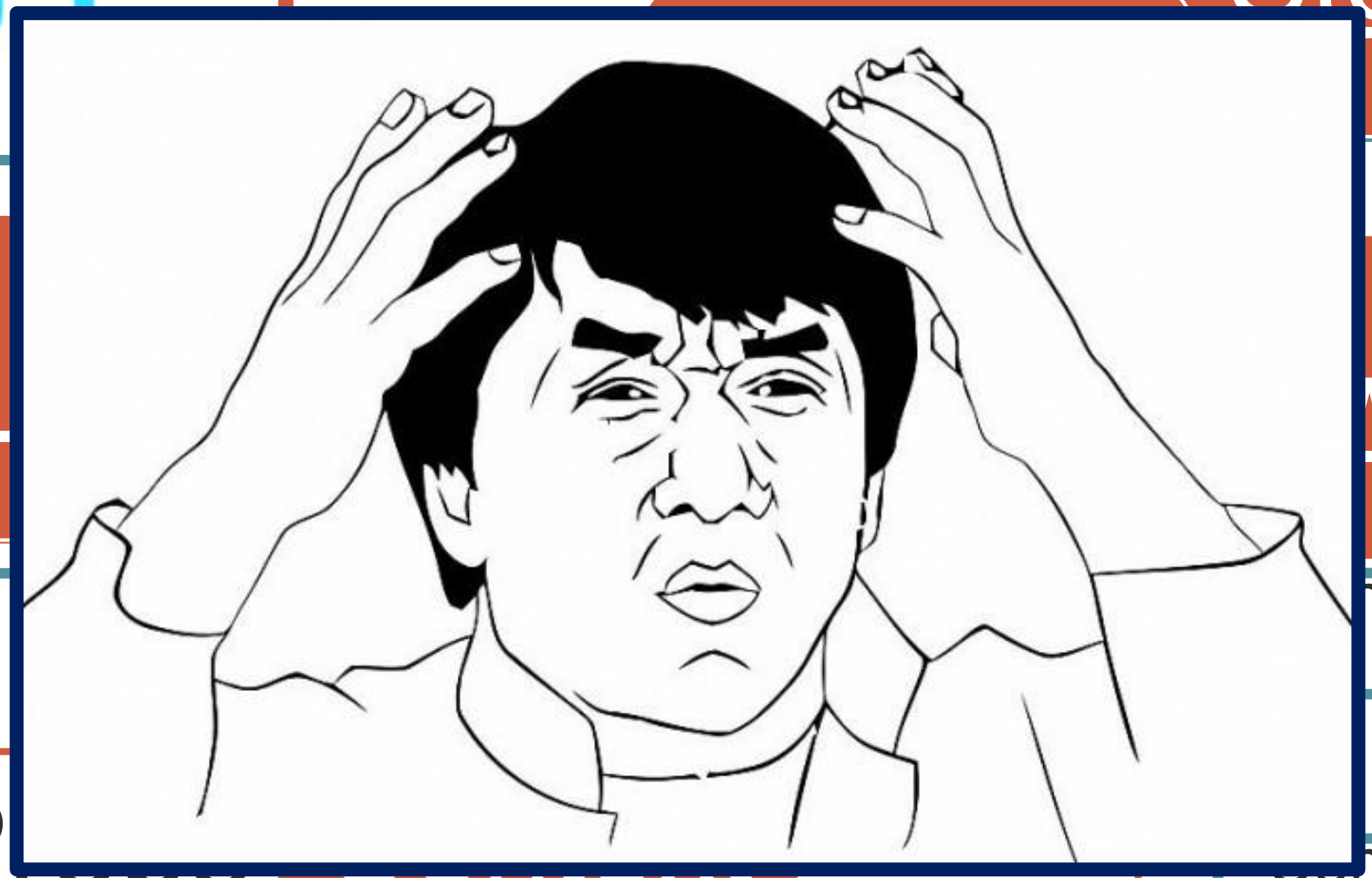
Contai

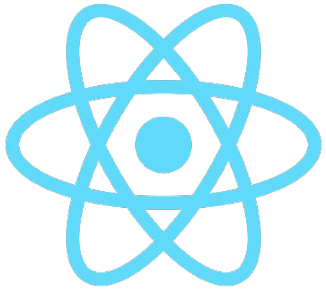
ware

mponent 6

Pr Context

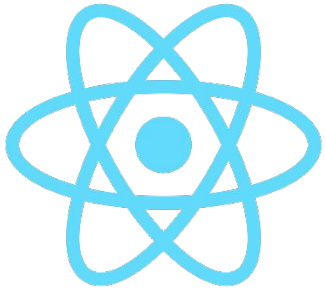
mponent 7





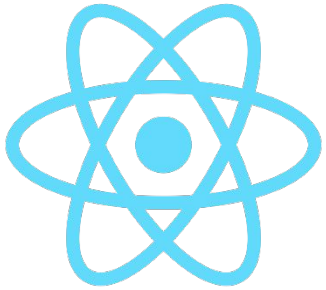

Дьявольски быстрый

- Virtual DOM
- shouldComponentUpdate




connect

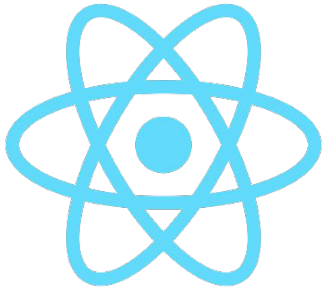
```
connect(
  ({ users, fav }) => ({ users, fav }),
  (dispatch) => ({ actions: {
    loadUsers, markFavorite, select, clear
  }})
)(FriendsList)
```




shouldComponentUpdate

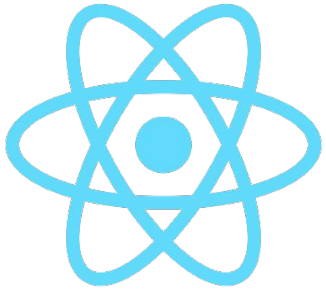
{ actions: { a: *function*, b: *function* }}

```
shouldComponentUpdate(nextProps, nextState) {
  return   !shallowE(this.props,  nextProps)  ||
           !shallowE(this.state,  nextState);
}
```




connect #2

```
connect(
{ users, fav }) => ({
  users,
  favoriteUsers: users.filter(u =>
    fav.contains(u.id)
  )
}
)(FriendsList)
```

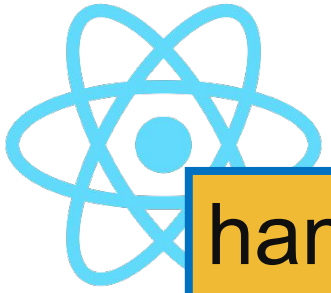



M: Memoization

reselect

```
export const favSelector =
  createSelector(
    state => state.users,
    state => state.fav,
    ( users, fav ) =>
      users.filter(
        u => fav.contains(u.id)
      )
  )
;
```

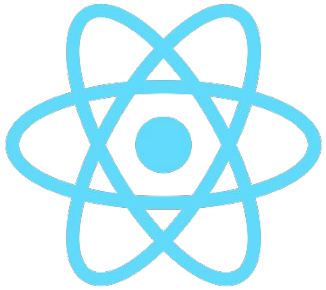
```
connect(state) => ({
  state.users,
  favoriteUsers:
    favSelector(state)
})
```



::bind

```
handleClick = ({target}) => {
  this.props.setChecked(target.value);
}
```

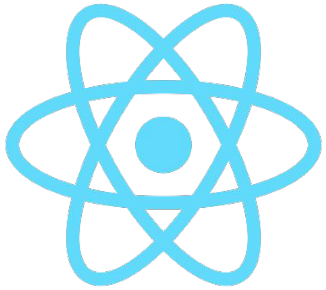
```
h
  t
}
render() {
  //...
  <input onClick="::this.handleClick" />
}
```




::bind #2

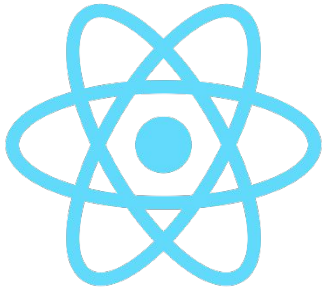
```
handleClick = name => value => {
  this.props.setFilter({[name]: value});
}
```

```
  < … onClick="this.handleClick.bind('user') >
  < … onClick="this.handleClick.bind('admin') >
```



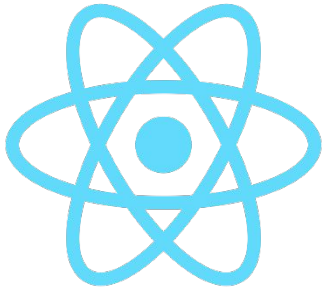

M: Memoization

- render не должен порождать новых сущностей
- новые сущности очень коварны
- [ ], { }
- тестируйте свои компоненты



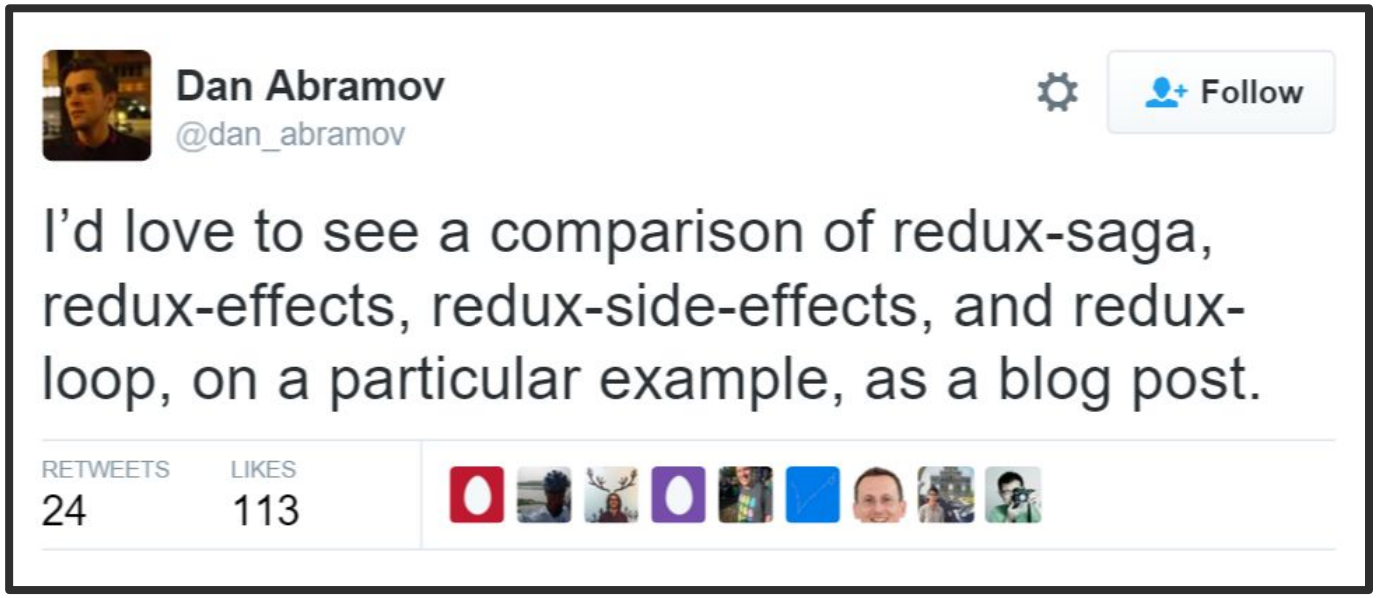
redux

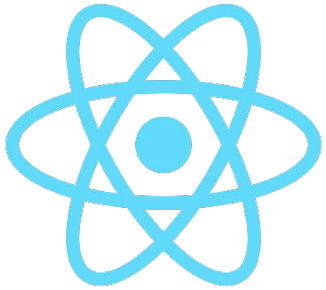
- action creators
- reducers
- middlewares




?: ???

- ❑ redux-sagas?
- ❑ redux-thunk?
- ❑ redux-side-effects?
- ❑ redux-effects?

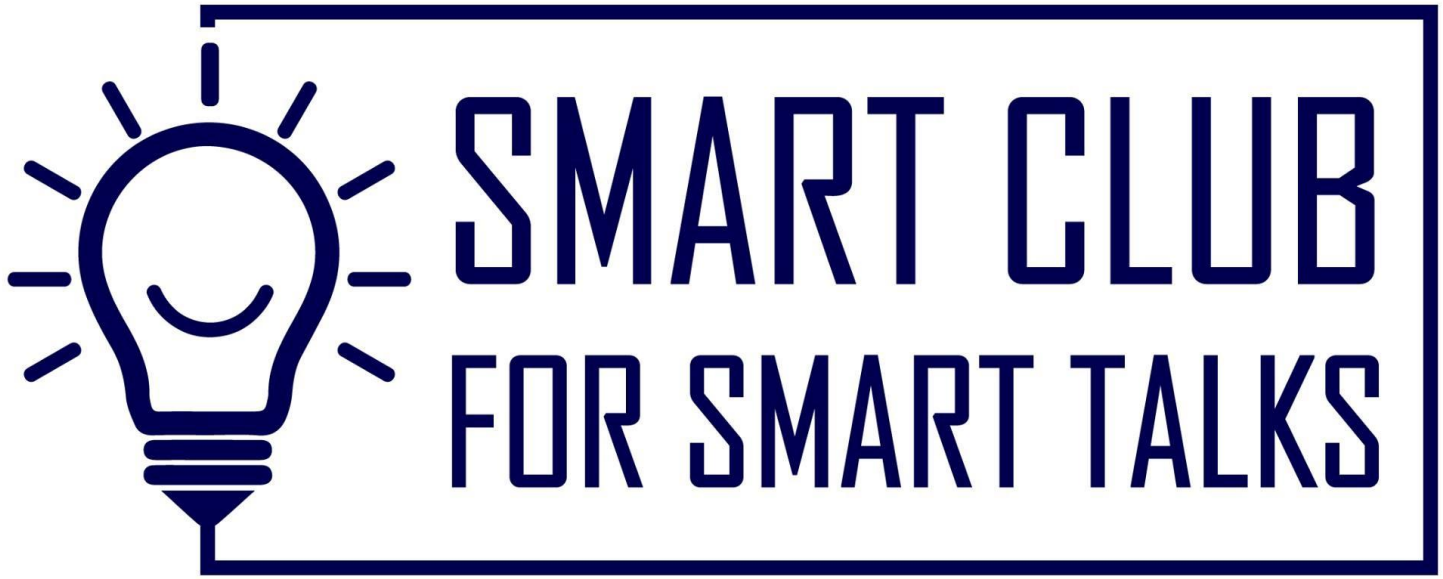





Слежение

```
const Perf = require('react-addons-perf');
Perf.start();
Perf.end();
Perf.printWasted();
```

(index)	Owner > component	Wasted time (ms)	Instances
0	"<root> > App"	15.317999947001226	37
1	"App > Box"	6.479999952716753	37
2	"Box > Box2"	1.9630000460892916	37



Вопросы?