

# State of state-management in React

Pavan Podila



@pavanpodila



# Concepts

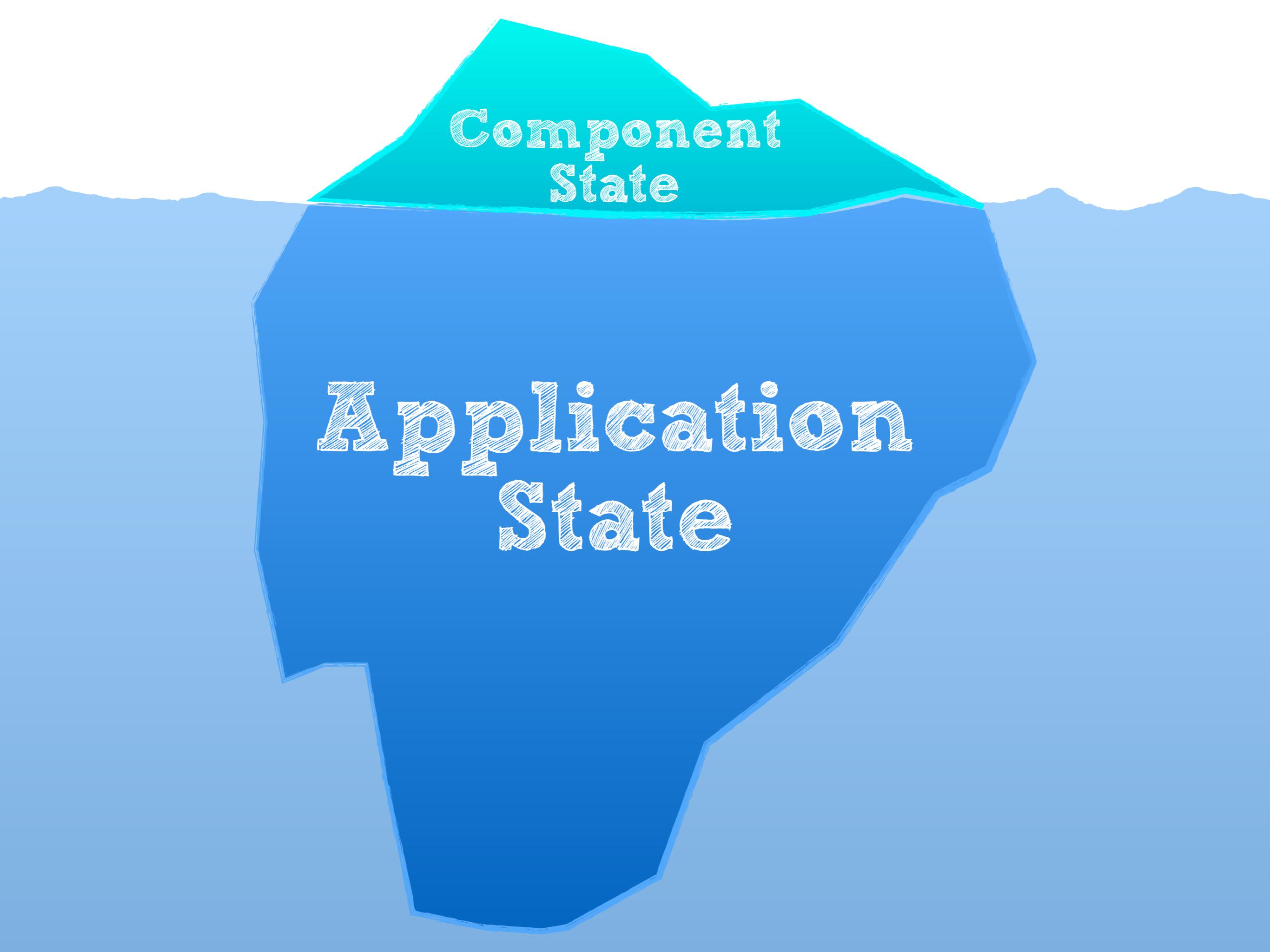
## Demo

### More Concepts

<Code />



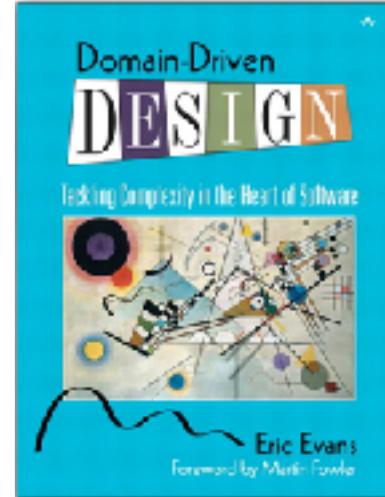
State



Component  
State

Application  
State

# Domain Driven Design



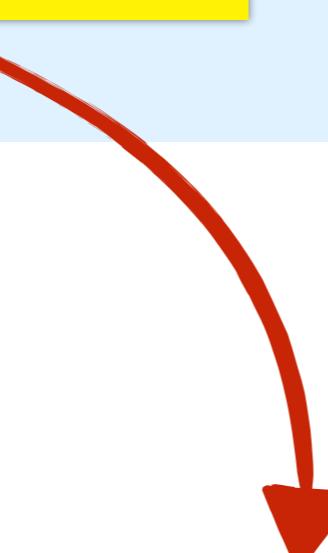
## Bounded Context

# Domain

```
class Contact {  
    firstName;  
    lastName;  
    email;  
  
    deliveryAddress;  
    billingAddress;  
    isDeliveryAddressSameAsBilling;  
}
```

# UI

```
function DeliveryAddress({isSameAsBilling}) {  
    return (  
        <div className="form-group">  
            <label>Same as Billing Address</label>  
            <Checkbox checked={isSameAsBilling}/>  
        </div>  
    );  
}
```



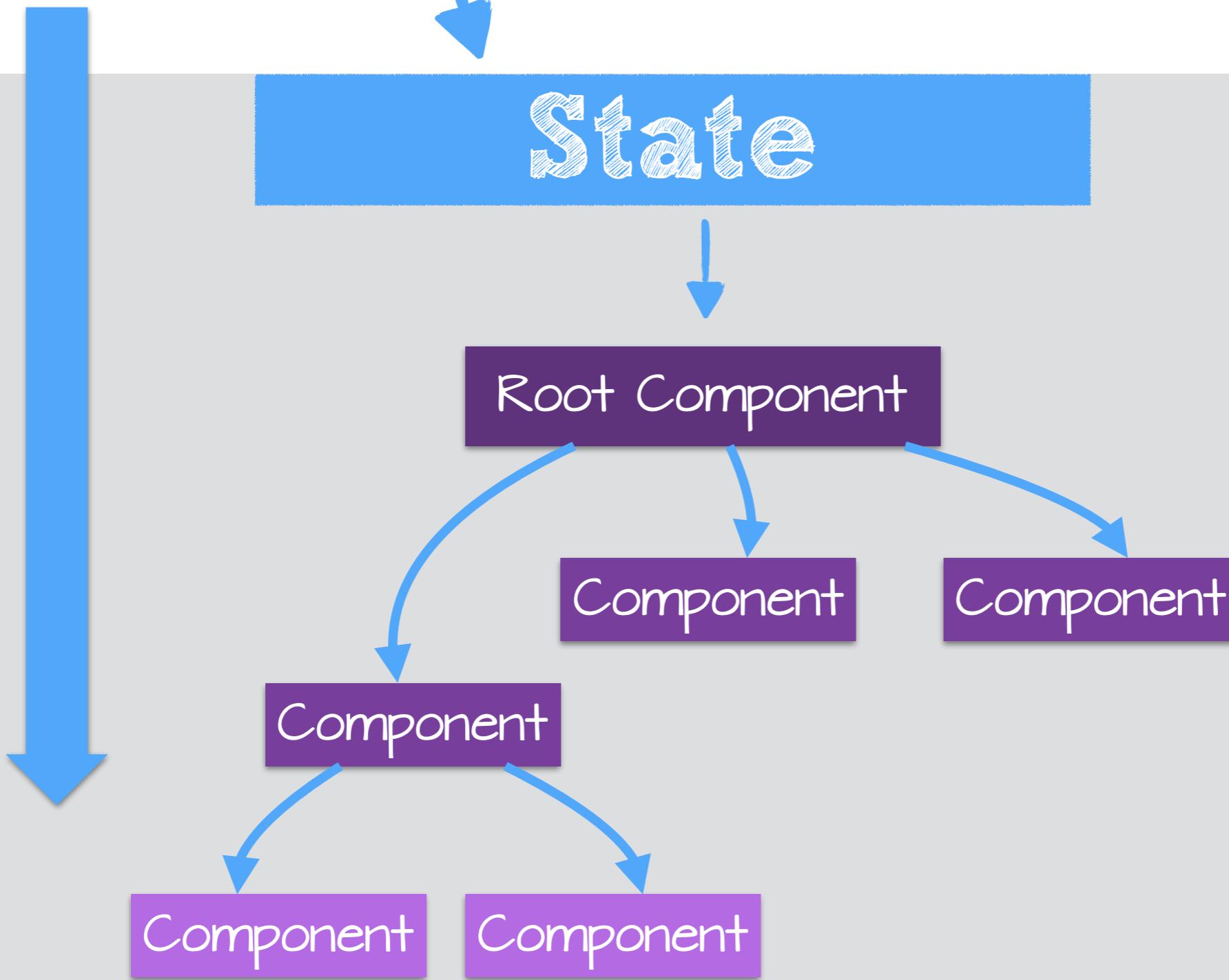
Server

Data



Client

State



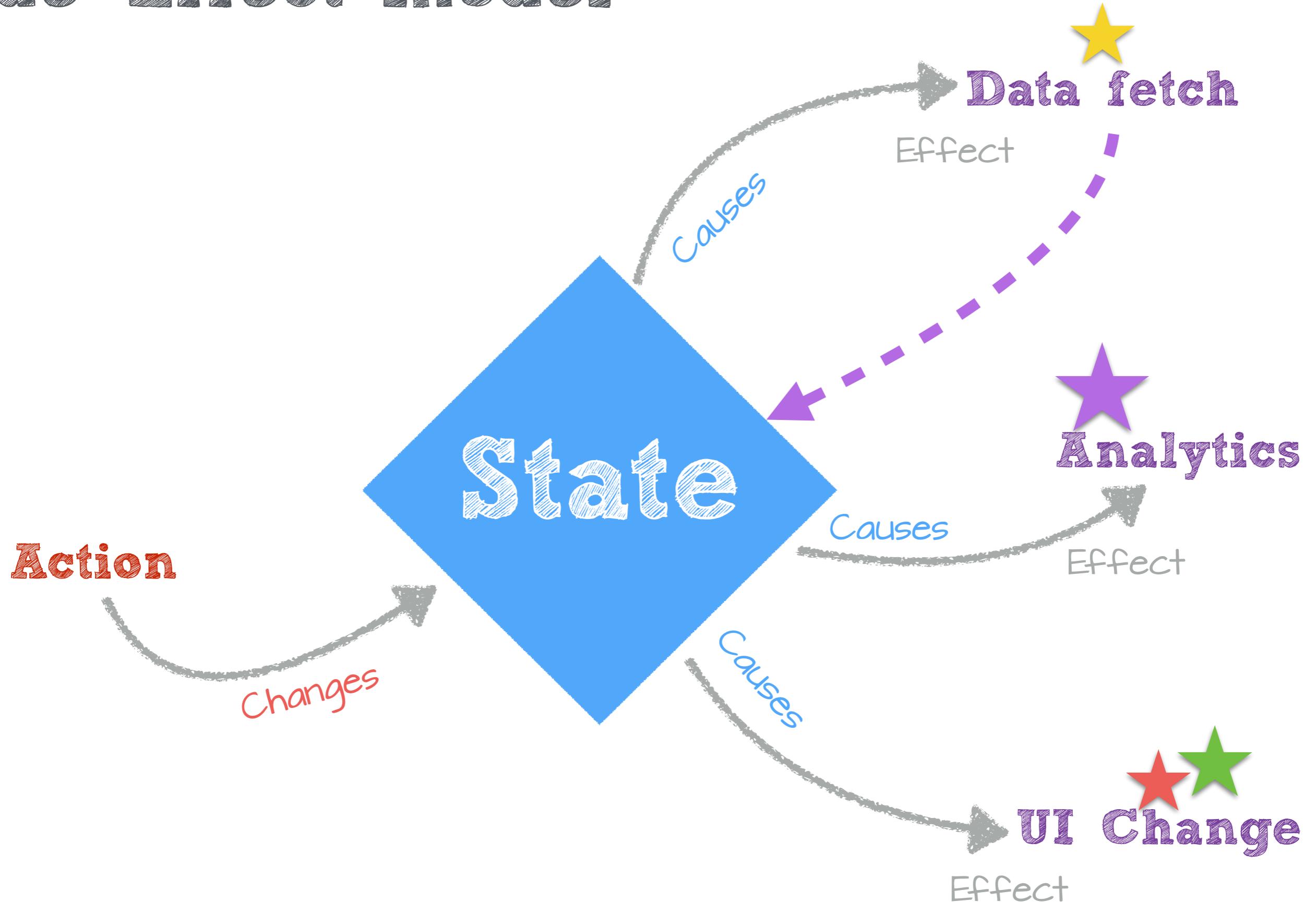
Events

Container

Presentation

# Side Effects

# Side-Effect model



Worker process

Routing

Animation

Analytics

UI Change

— Side Effects —

Data Fetch

Alerts

Form Validation

Notifications

UI  
Development

Managing

State

and the

Side Effects

it causes



@pavanpodila

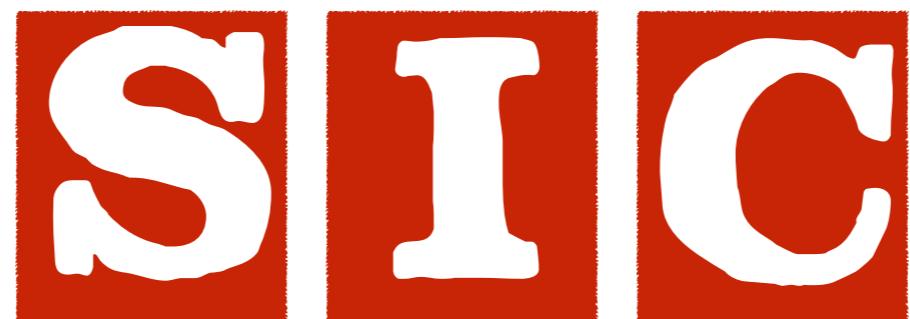
# Managing Complexity

State



Side  
Effects

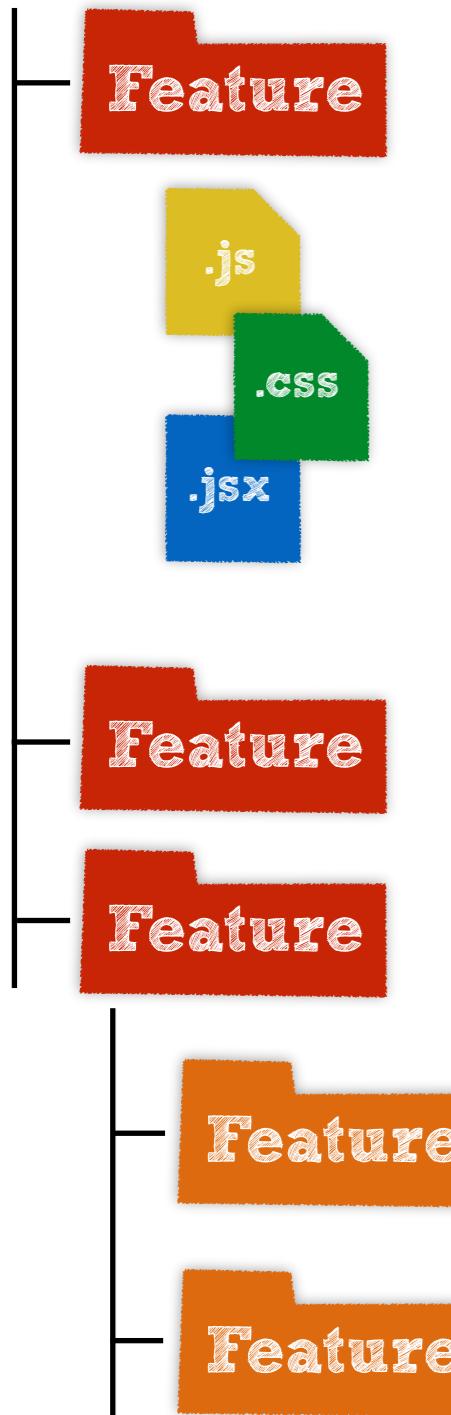
Organizing with the



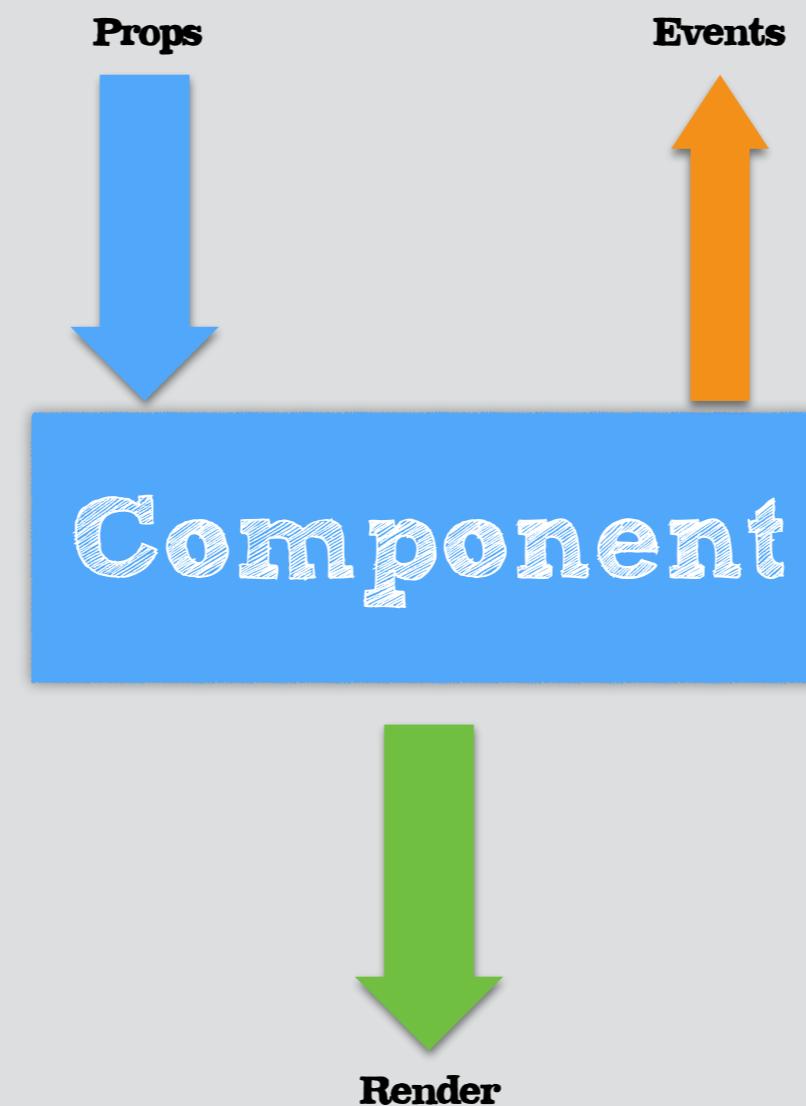
principle

# Structure

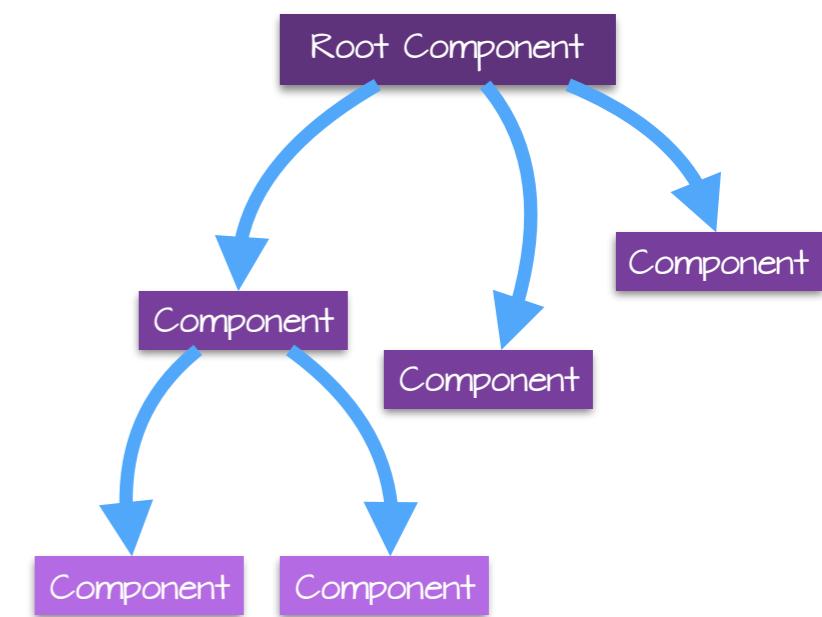
/src



# Isolation

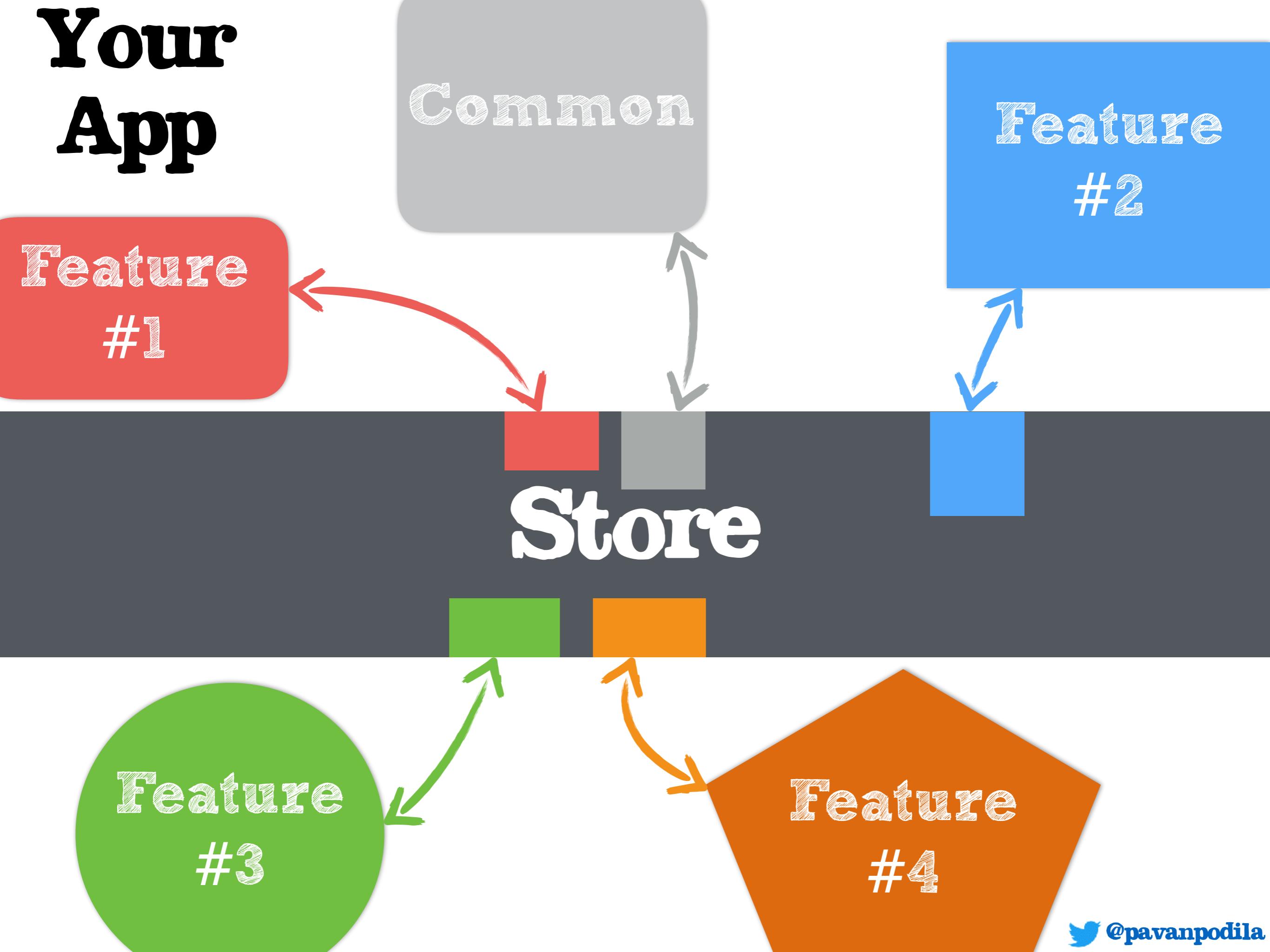


# Composition



SIC

# Your App



# The Demo



<https://github.com/pixelingene/react-state-mgmt>

setState

# Redux

redux-~~xxx~~ middleware

redux-thunk

redux-batched-actions

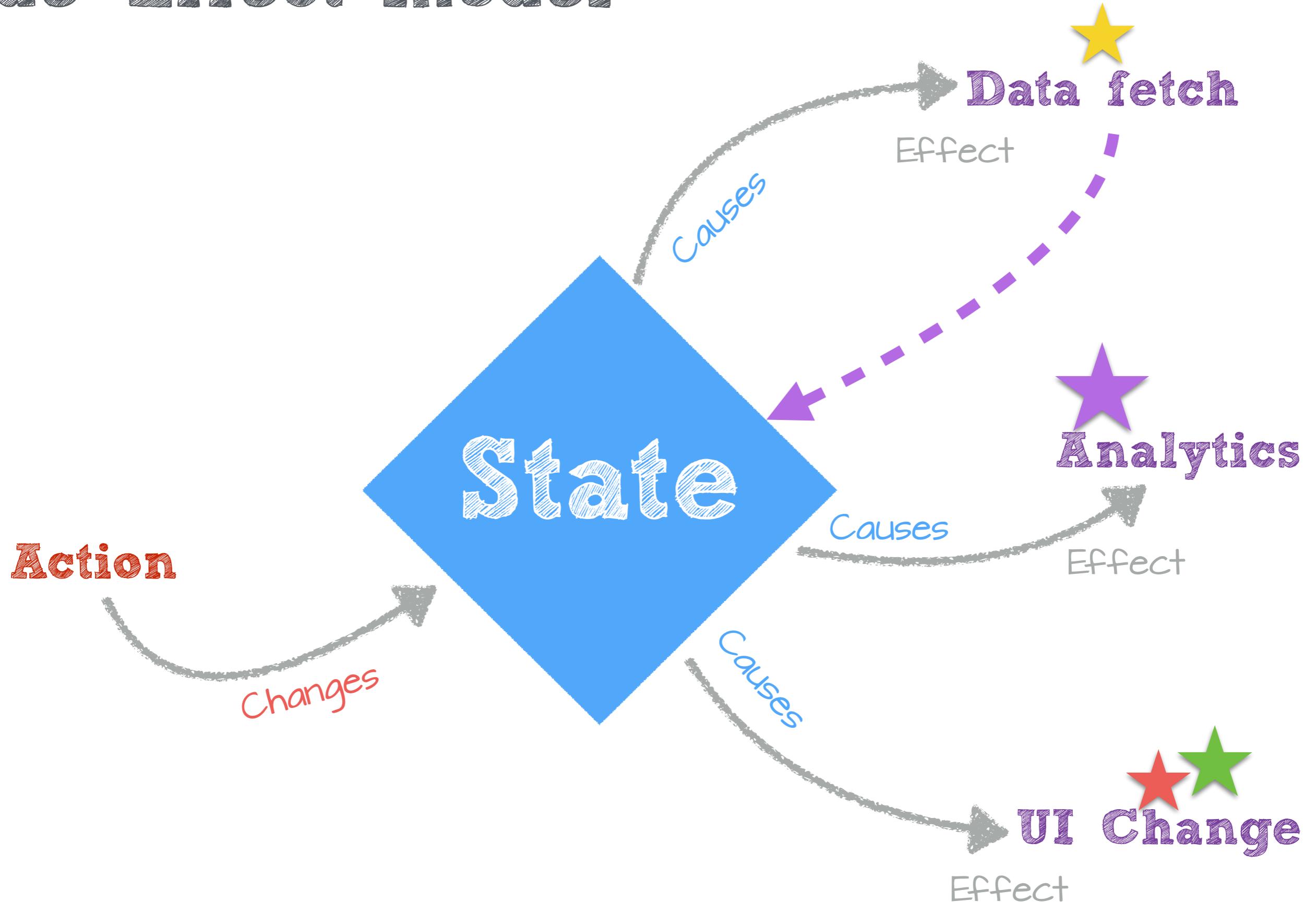
react-router-redux

redux-observable

redux-saga

# MobX

# Side-Effect model



React

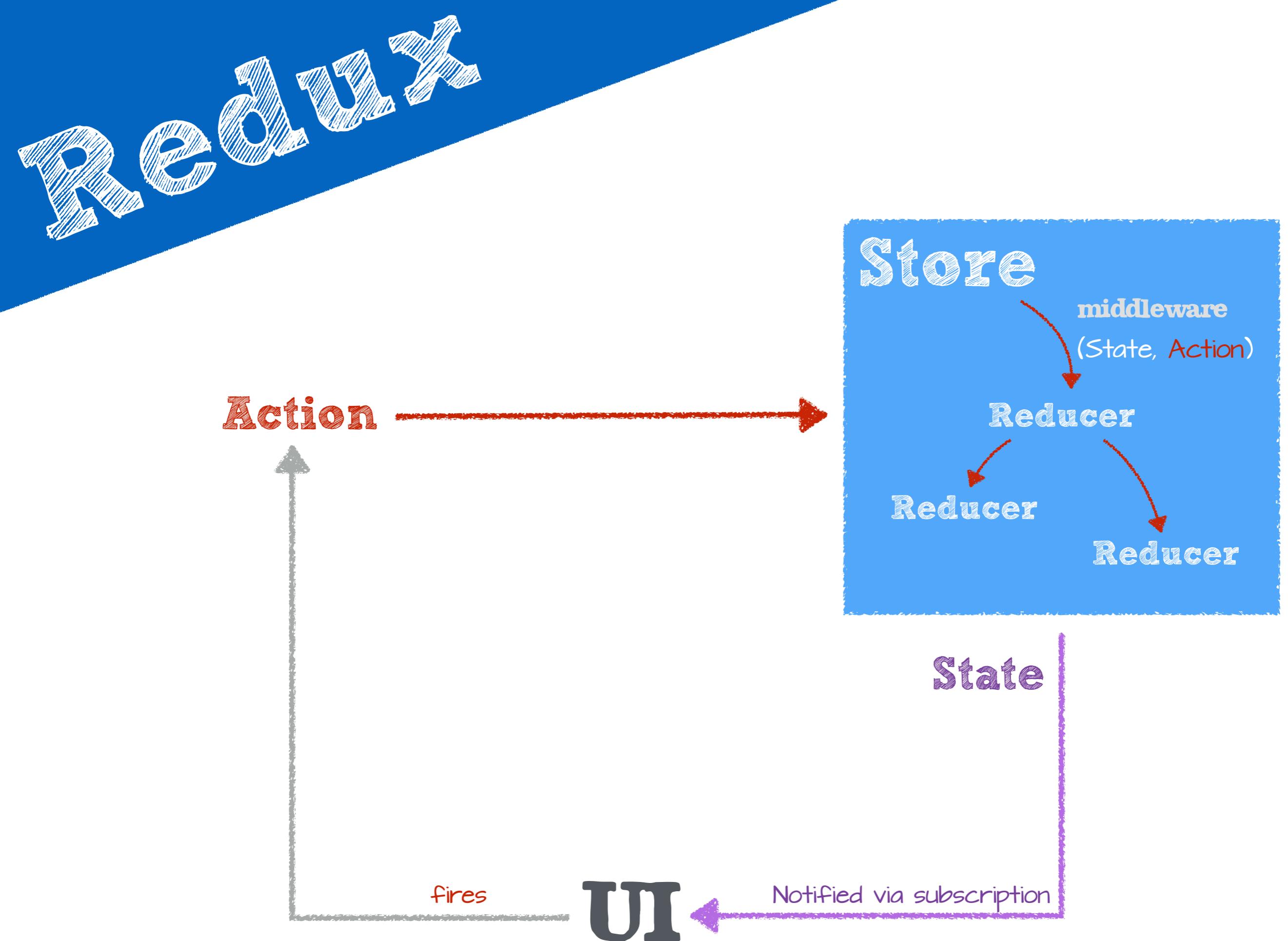
Action

`setState()`

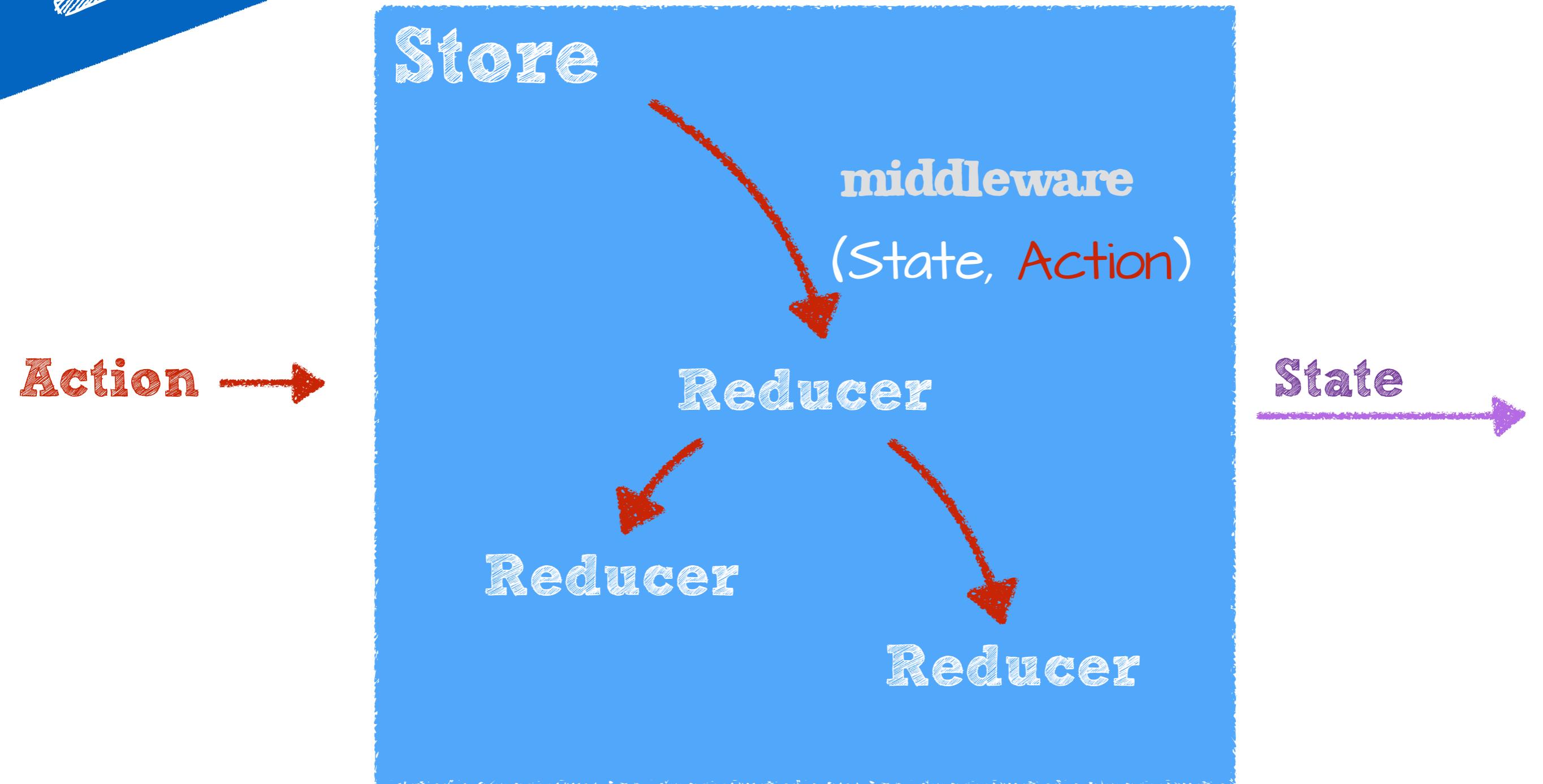


Effect

`render()`



# redux



Side Effects

in

# Redux

→ **redux-thunk**

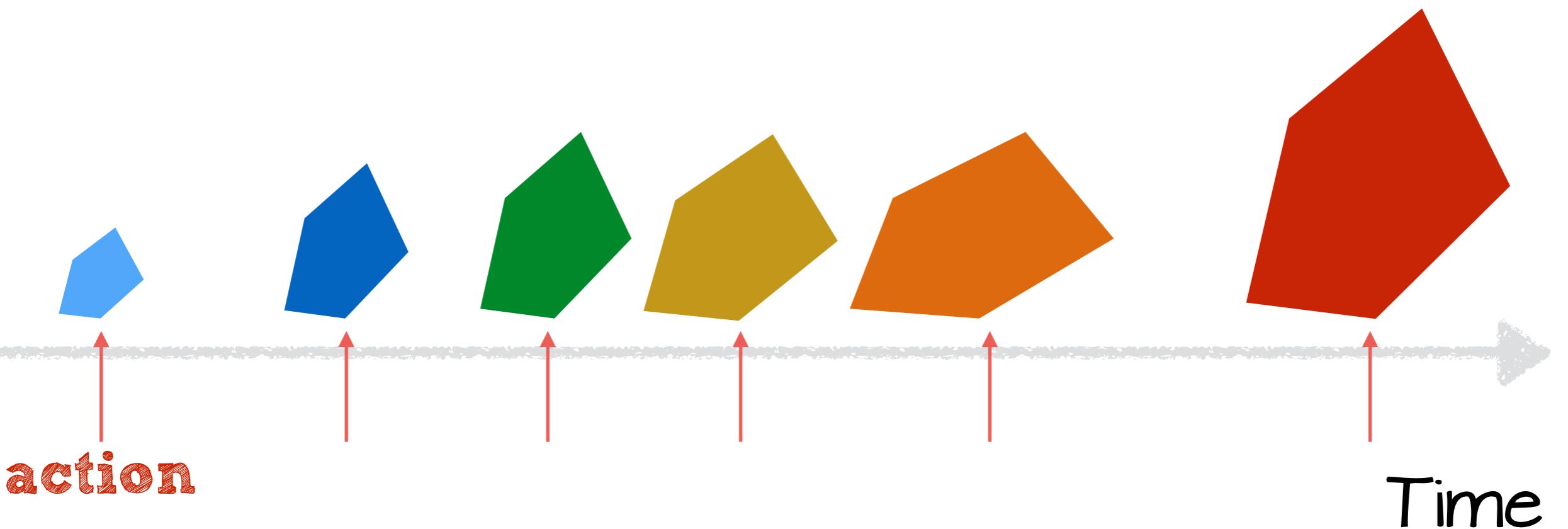
→ **redux-batched-actions**

redux-\* middlewares

→ **redux-saga**

→ **redux-observable**

# State changes

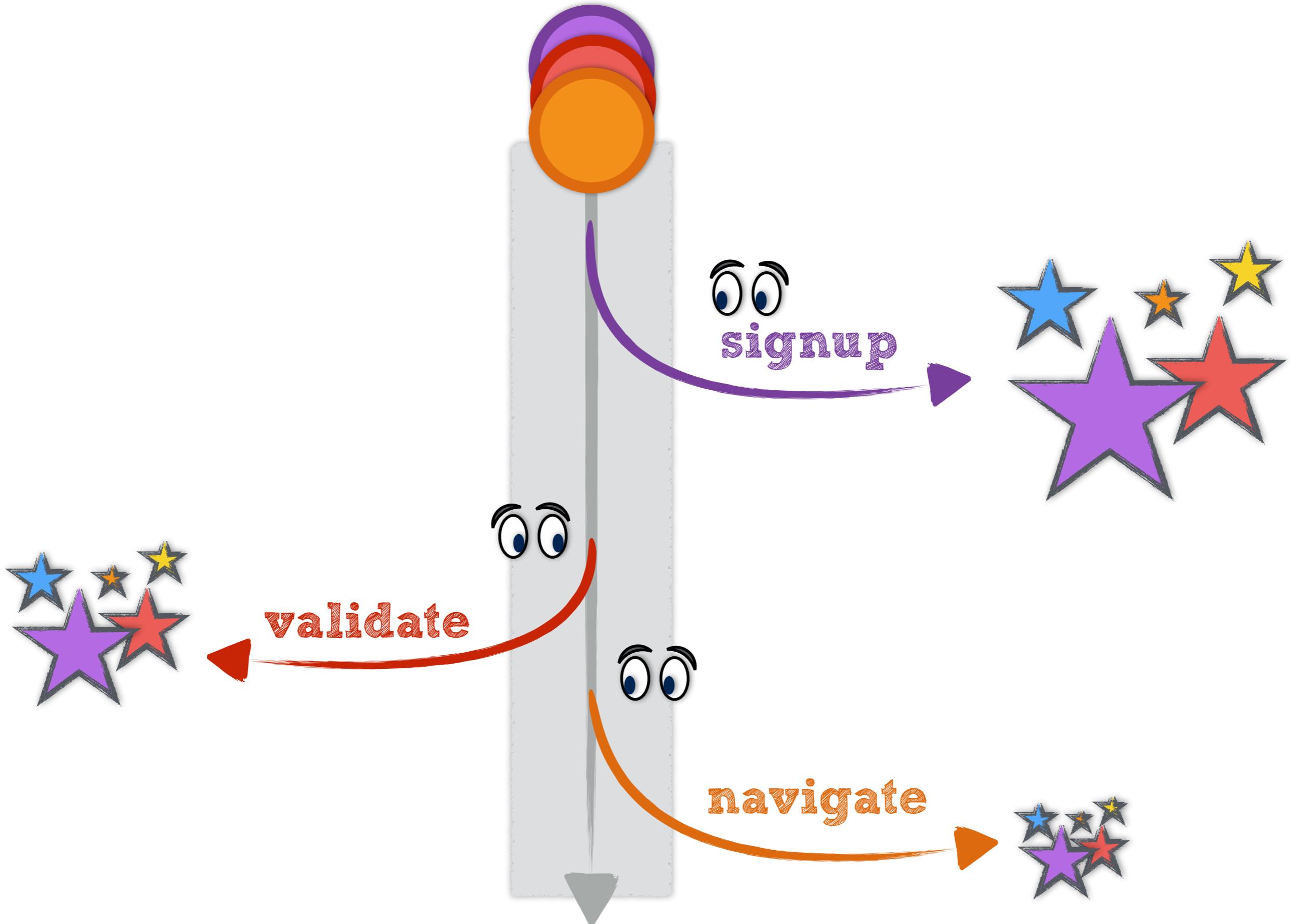


Monitor  
the



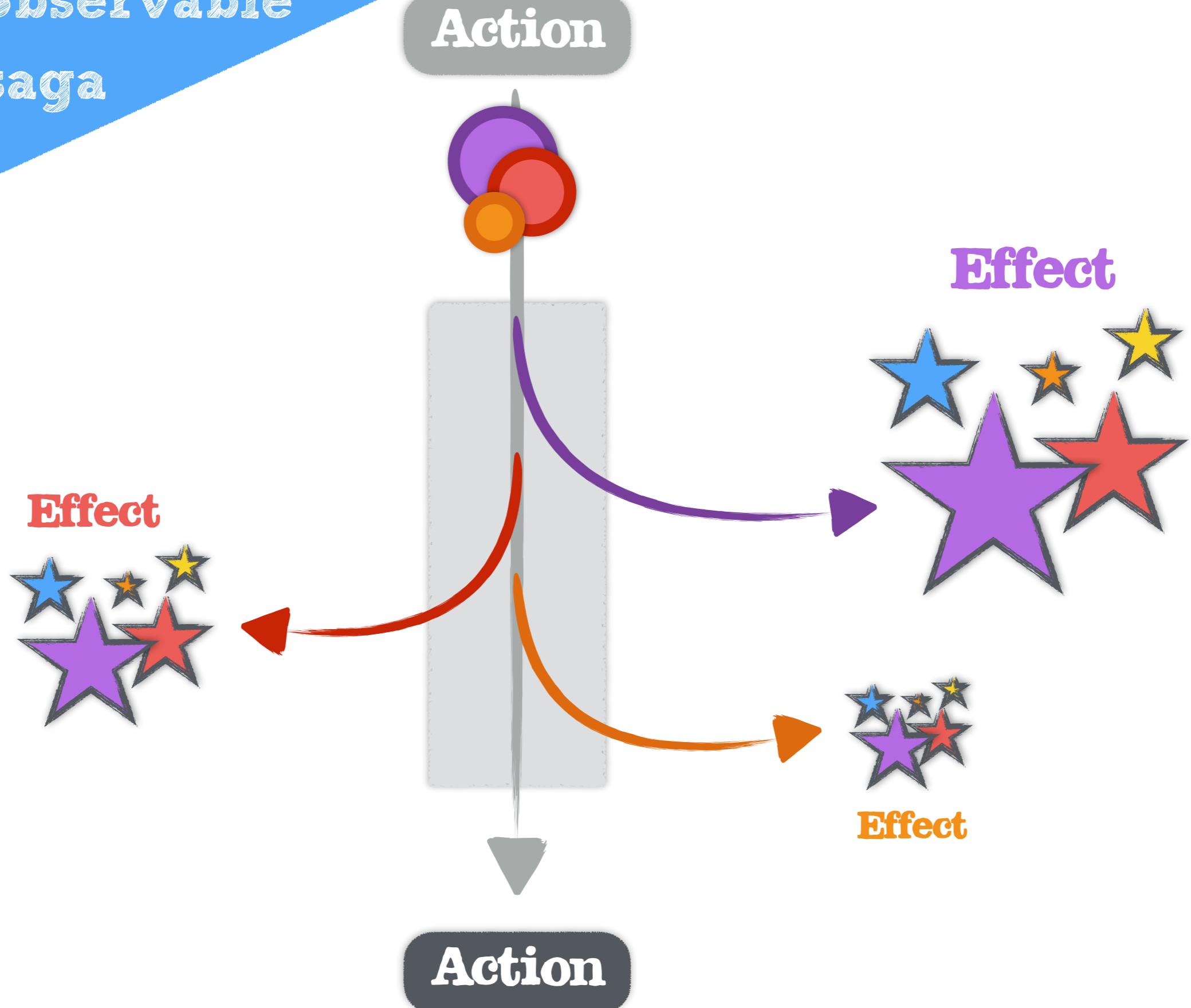
Action

# Action



# redux-observable

# redux-saga

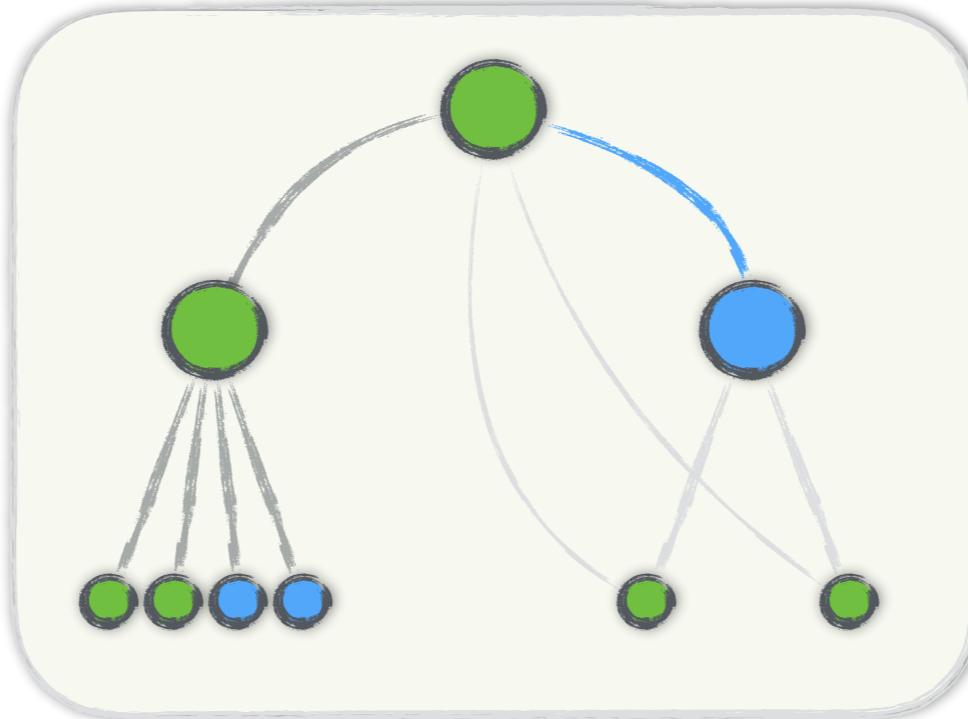


# MobX

## observables

action

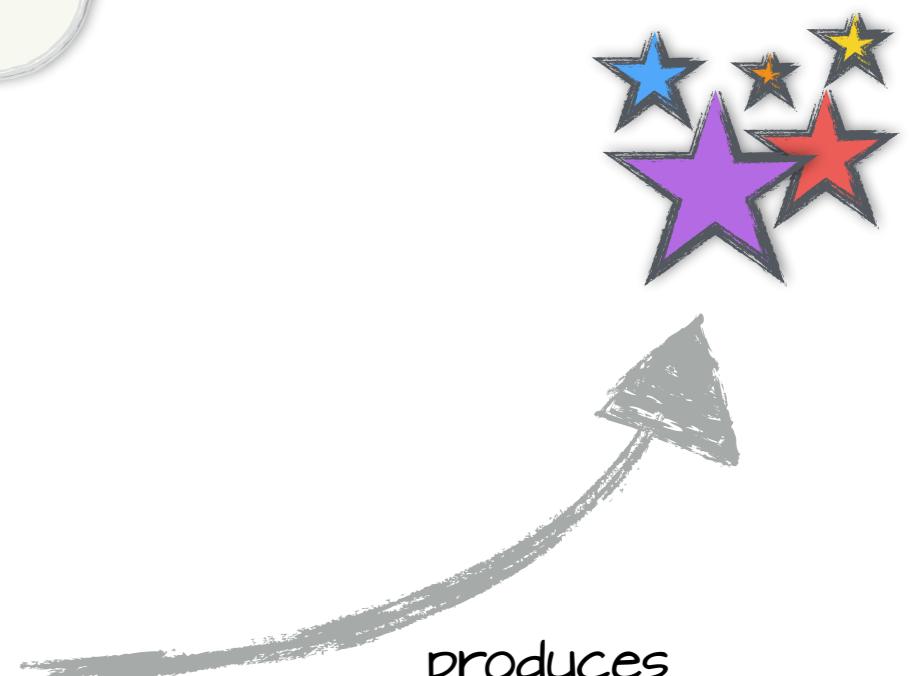
mutates



tracks



**when()**  
**autorun()**  
**reaction()**



Let's see this  
in  
Code

State of  
state-management  
in  
React

Redux  
Redux Saga  
Redux Observable  
MobX

**Pavan Podila**

@pavanpodila