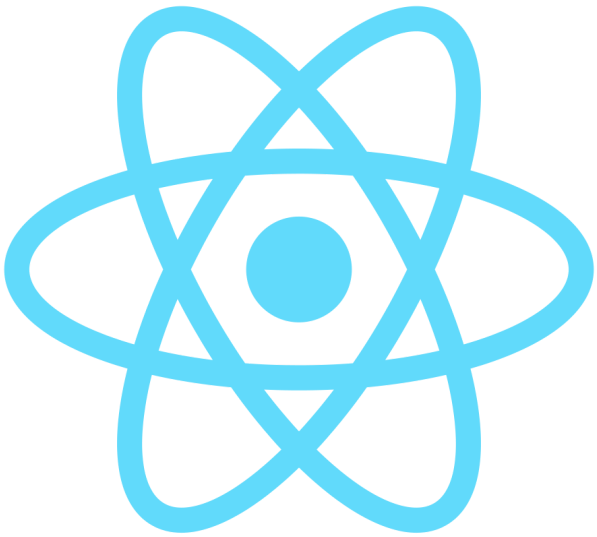

react
JA



INTRO TO REACT

David Dean

David Dean
Senior IT Developer at Florida Blue
@priceyvulture
https://github.com/flapdragon

PURPOSE

- Historical Context
- High Level Overview
- Where to Begin With React

Intended for developers at any level who are new to React


brooksource

A VERY BRIEF HISTORY OF WEB DEVELOPMENT

- Server Ages
- jQuery Empire
- Monolithic SPA Era
- Today

SERVER AGES: PROS

- All one technology
- Easier to understand/learn
- Easier to scale
- Better SEO

SERVER AGES: CONS

- Every action requires round trip to server and browser re-render
- Content is always re-rendered
- Limited to no JS
- Limited dynamic functionality
- Browsers not as standardized

JAVASCRIPT

- Released May 23, 1995
- Brendan Eich @ Netscape
- Written in 10 days

JQUERY EMPIRE: PROS

- Easy to learn
- Easy to integrate
- Plugins
- SEO

JQUERY EMPIRE: CONS

- Duplication of business logic
- Selector/Callback Hell
- Too many ways to do the same thing using the same tools
- Difficult to build large apps
- Difficult to scale/test
- Plugins (versions)

MONOLITHIC SPA ERA: PROS

- No more selectors
- Better separation of conerns
- Easier to scale
- Shift from DOM to data

MONOLITHIC SPA ERA: CONS

- Easy to build slow apps
- Too many ways to do the same thing using the same tools
- Frameworks don't always work well together
- Limited SEO

Today = (pros) => React

- "Easy to reason about" - Everyone ever
- Easy to test
- Even more separation of concerns
- Easy to integrate
- SEO

Today = (cons) => React

- Not opinionated enough - integration fatigue

REACT

It's the View

REACT

It's NOT the View

React challenges established best practices in MVC

Better to think of it as "Component-based" UI than MVC "View"

REACT
Core Concepts

- JSX
- Components
- One-way Data Flow
- Virtual DOM

JSX

Syntax extension to JavaScript

```
function Avatar(props) {
  return (
    <img className="Avatar"
      src={props.user.avatarUrl}
      alt={props.user.name}
    />
  );
}
```

JSX

JSX Prevents Injection/XSS Attacks

```
1  // user input
2  const title = response.potentiallyMaliciousInput;
3  // This is safe:
4  const element = <h1>{title}</h1>;
5
```

JSX

```jsx
// JSX
const element = (
  <h1 className="greeting">
    Hello, world!
  </h1>
);

```

```jsx
// Compiled to React.createElement() call
const element = React.createElement(
  'h1',
  {className: 'greeting'},
  'Hello, world!'
);

```

Components

UI Description and UI Logic are tightly coupled

```
function ActionLink() {
  function handleClick(e) {
    e.preventDefault();
    console.log('The link was clicked.');
  }

  return (
    <a href="#" onClick={handleClick}>
      Click me
    </a>
  );
}
```

Components

Stateless

```
function Dialog(props) {
  return (
    <FancyBorder color="blue">
      <h1 className="Dialog-title">
        {props.title}
      </h1>
      <p className="Dialog-message">
        {props.message}
      </p>
      {props.children}
    </FancyBorder>
  );
}
```

Components

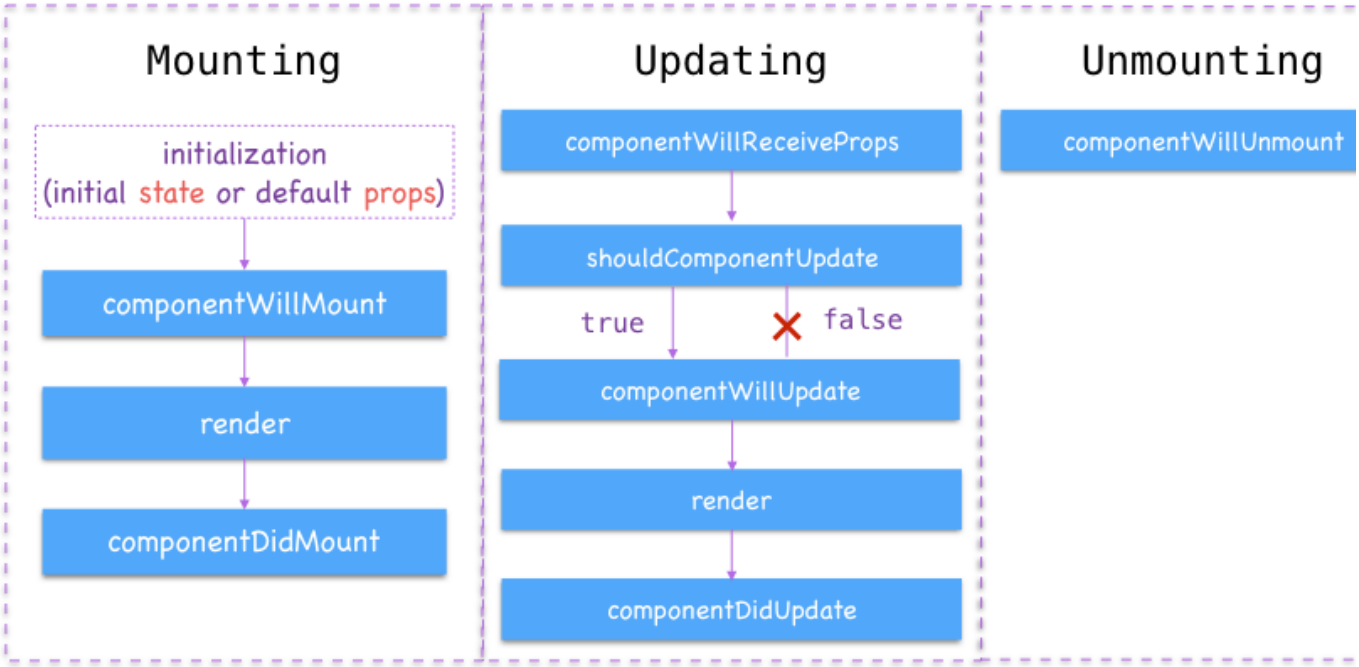
Stateful ...

```
class SignUpDialog extends React.Component {
  constructor(props) {
    super(props);
    this.handleChange = this.handleChange.bind(this);
    this.handleSignUp = this.handleSignUp.bind(this);
    this.state = {login: ''};
  }
  render() {
    return (
      <Dialog title="Mars Exploration Program" message="How should we refe
        <input value={this.state.login} onChange={this.handleChange} />
        <button onClick={this.handleSignUp}>Sign Me Up!</button>
      </Dialog>
    );
  }
  handleChange(e) {
    this.setState({login: e.target.value});
  }
  handleSignUp() {
    alert(`Welcome aboard, ${this.state.login}!`);
  }
}

```

Components

Component Lifecycle



One-way Data Flow

- Flows down the component hierarchy
- "Props down" -> Immutable, to children
- "State up" -> Mutable, up to parent
- Paradigm -> Removes confusion

Challenge is understanding which component should own state

Virtual DOM

The Problem: the Actual DOM

The Document Object Model is a platform- and language-neutral interface that will allow programs and scripts to dynamically access and update the content, structure and style of documents

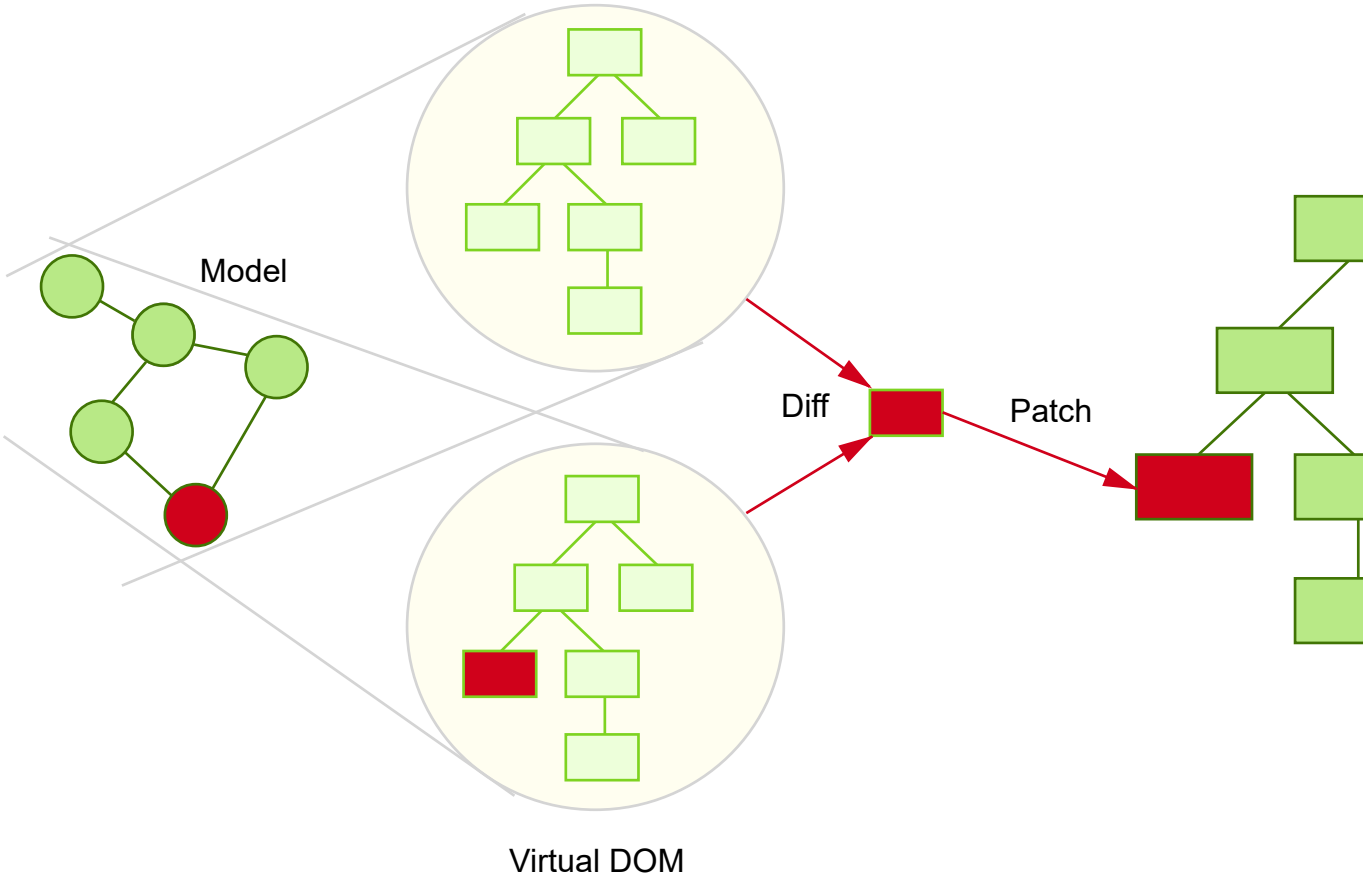
It's slow

Virtual DOM

The Solution

- Re-render everything on every update!
- Diff
- Compute changes
- Batch execution

Manipulating the virtual DOM is fast


Model
Diff
Patch
Virtual DOM

HOW TO USE REACT

- Large application
- Small piece that fits into larger app

WHO USES REACT

- Facebook
- Netflix
- AirBnB
- Instagram
- Lyft
- New York Times
- Reddit
- Twitter
- Uber

WHY DO I LOVE REACT?

- Modularity -> Choice
- Separation of Concerns -> Knowledge
- JavaScript
- One-way Data Flow
- Immutability

LOCAL DEVELOPMENT ENVIRONMENT

- IDE/Code Editor
- Build Tools
- Version Control System

NODE

Node is central to everything
Node -> NPM
Node -> Electron -> Atom/VS Code
Node -> Webpack/Grunt/Gulp

IDEs/Code Editors

- GitHub Atom
- JetBrains IntelliJ IDEA
- JetBrains WebStorm
- Microsoft Visual Studio Code
- Vim

IDEs/Code Editors: PLUGINS

- Framework (react...)
- Build (language-babel...)
- Linting (linter, linter-eslint...)
- Versioning (Git, SVN...)
- Productivity (Emmet!, Snippets...)
- Testing (mocha-test-runner...)
- Pair Programming (atom-pair...)

Build Tools

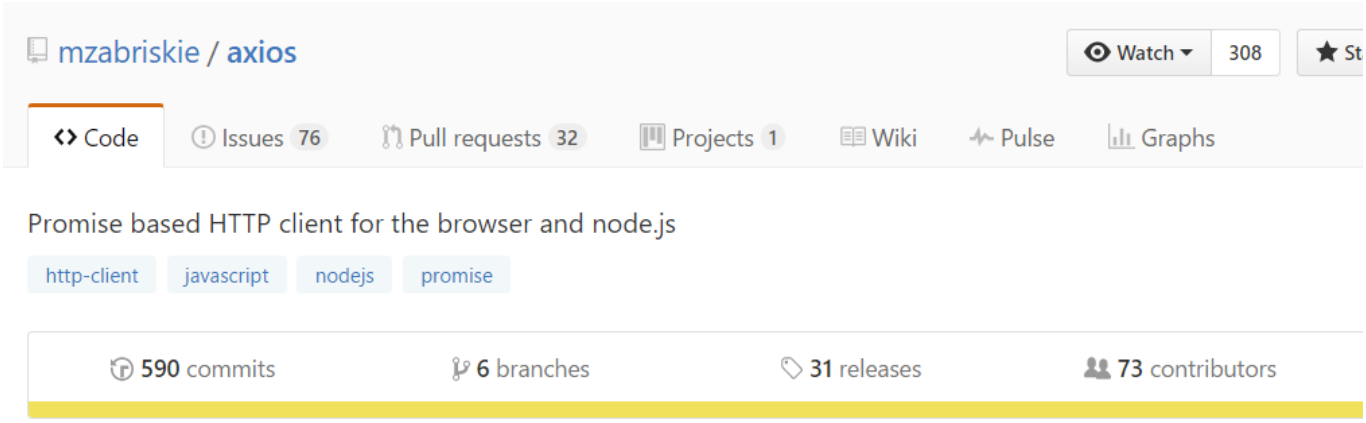
- Webpack
- Babel
- Linters -> Syntax
- Editor Config -> Code Style

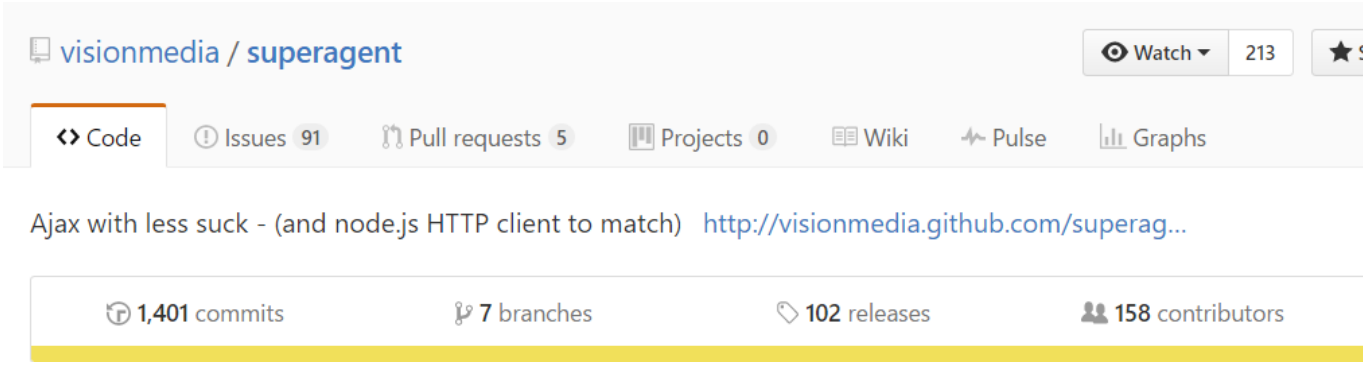
Browser Plugins

- React Developer Tools
- Redux DevTools

Version Control System (Git)

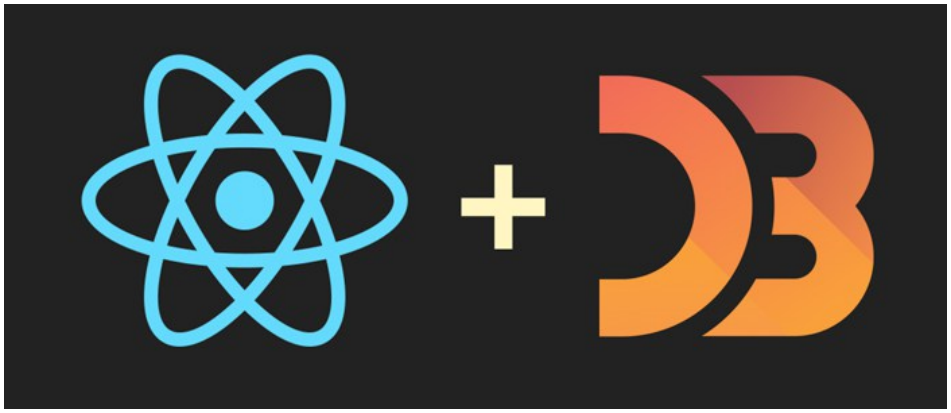
- Try Git
- Git Flow
- Git Commit Message Style


mzabriskie / axios
Watch 308
Code
Issues 76
Pull requests 32
Projects 1
Wiki
Pulse
Graphs
Promise based HTTP client for the browser and node.js
http-client
javascript
nodejs
promise
590 commits
6 branches
31 releases
73 contributors


visionmedia / superagent
Watch 213
Code
Issues 91
Pull requests 5
Projects 0
Wiki
Pulse
Graphs
Ajax with less suck - (and node.js HTTP client to match) http://visionmedia.github.com/superag...
1,401 commits
7 branches
102 releases
158 contributors

BRING YOUR FORMS INTO THE 21ST CENTURY WITH REDUX FORM

- Jonathan Stewart

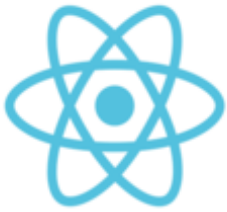


REACT + D3

- David Dean

Jax Node

www.meetup.com/Jax-Node-js-UG



React Native Jax

www.meetup.com/React-Native-Jax

THANK YOU :)

Linting errors are a direct reflection on your personality.

- David Dean