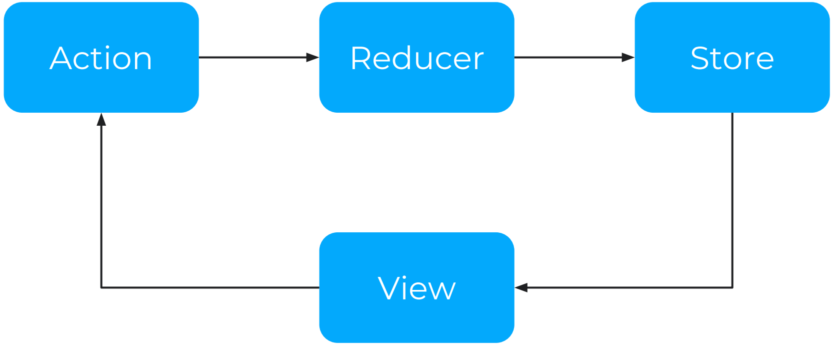
You might not need Redux; A little context

Dillon Mulroy | Software Engineer @ Visionist inc

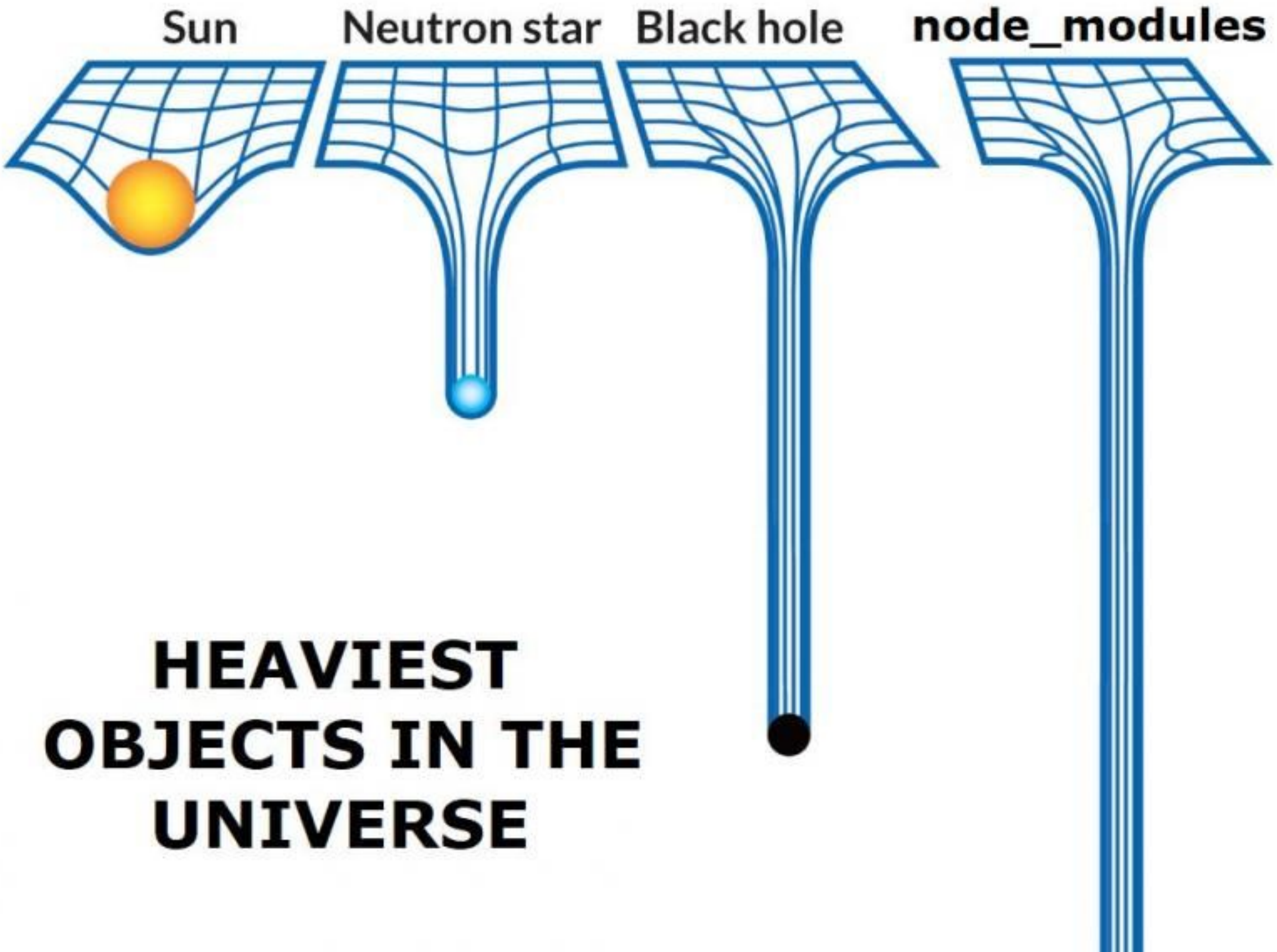
What is Redux?

- “Redux is a predictable state container for JavaScript apps.” - redux.js.org
- Unidirectional data flow inspired by Elm and Flux
- State is only mutated by “dispatching” serializable actions.
- Reducers take actions and apply them to your state, returning a new copy.



Common Complaints

- Hard to learn
- Feels like a requirement to use React
- Boilerplate hell
- Lots of potential dependencies
    - redux-thunk/redux-saga
    - redux-logger
    - reselect
    - Immutable js
    - redux + react-redux



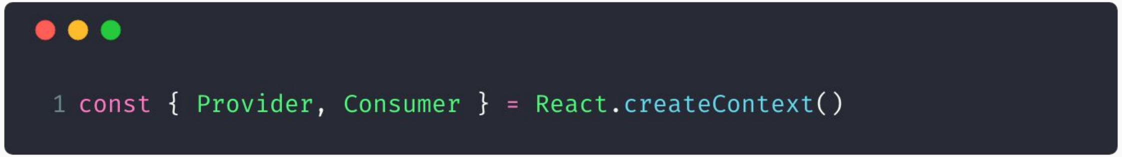
HEAVIEST OBJECTS IN THE UNIVERSE

Enter React 16.3

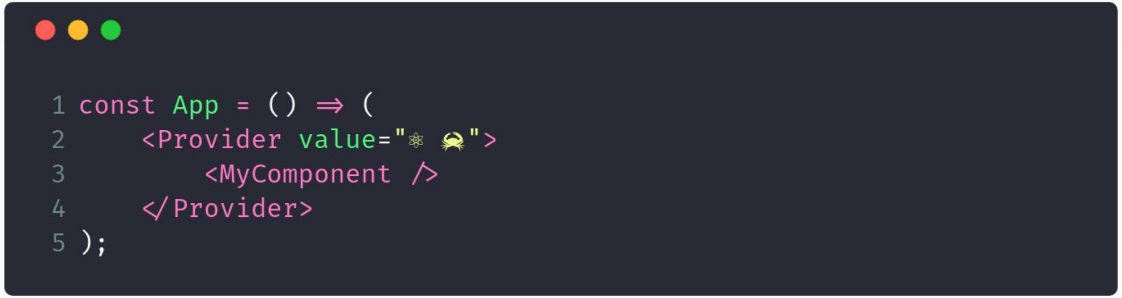
New Context API

Context API

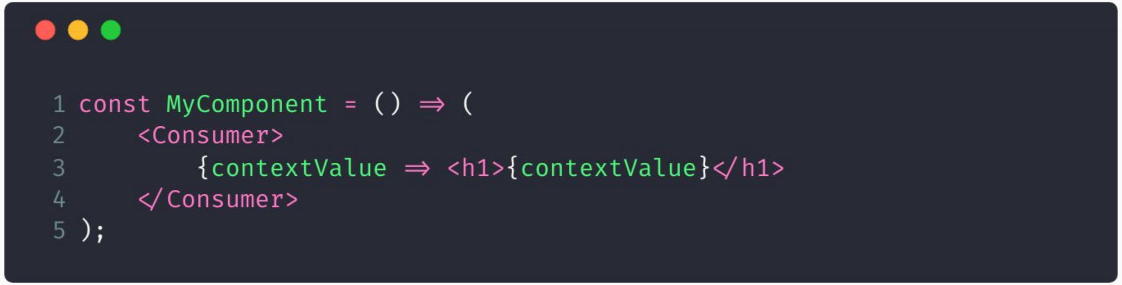
Context provides a way to pass data through the component tree without having to pass props down manually at every level.



```
const { Provider, Consumer } = React.createContext()
```



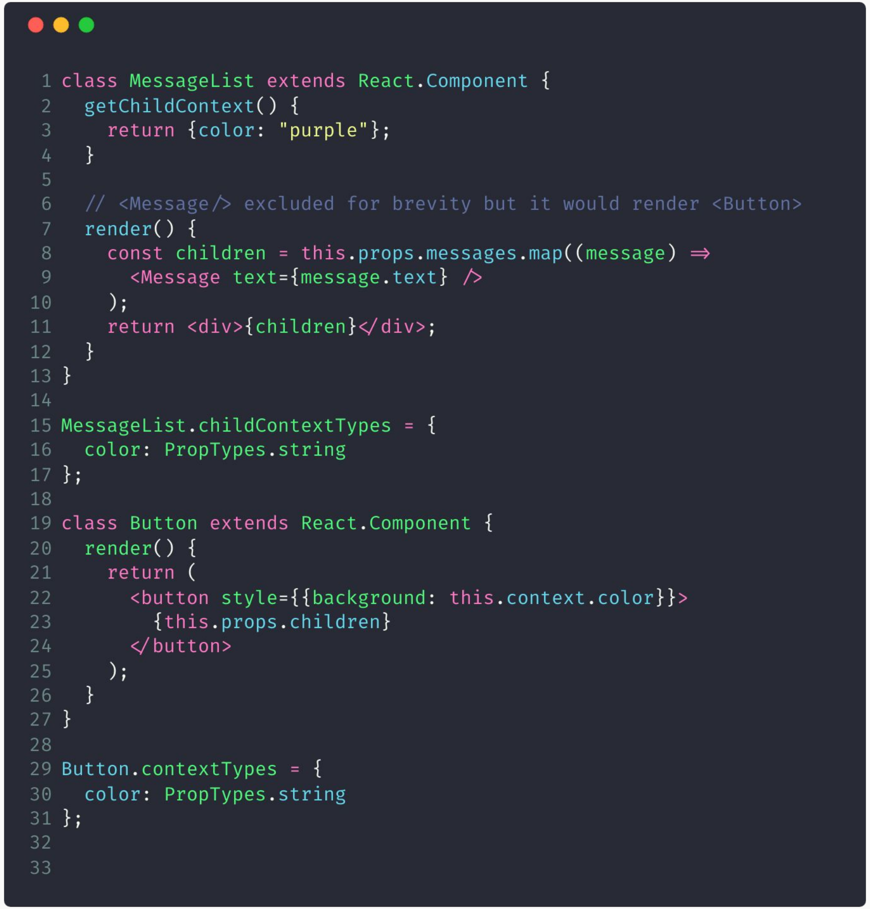
```
const App = () => (
    <Provider value="⚛ 🦀">
        <MyComponent />
    </Provider>
);
```



```
const MyComponent = () => (
    <Consumer>
        {contextValue => <h1>{contextValue}</h1>
    </Consumer>
);
```

<h1>⚛ 🦀</h1>

Legacy Context API*



```jsx
class MessageList extends React.Component {
  getChildContext() {
    return {color: "purple"};
  }

  // <Message/> excluded for brevity but it would render <Button>
  render() {
    const children = this.props.messages.map((message) =>
      <Message text={message.text} />
    );
    return <div>{children}</div>;
  }
}

MessageList.childContextTypes = {
  color: PropTypes.string
};

class Button extends React.Component {
  render() {
    return (
      <button style={{background: this.context.color}}>
        {this.props.children}
      </button>
    );
  }
}

Button.contextTypes = {
  color: PropTypes.string
};


```

* If you want your application to be stable, don’t use context. It is an experimental API and it is likely to break in future releases of React.

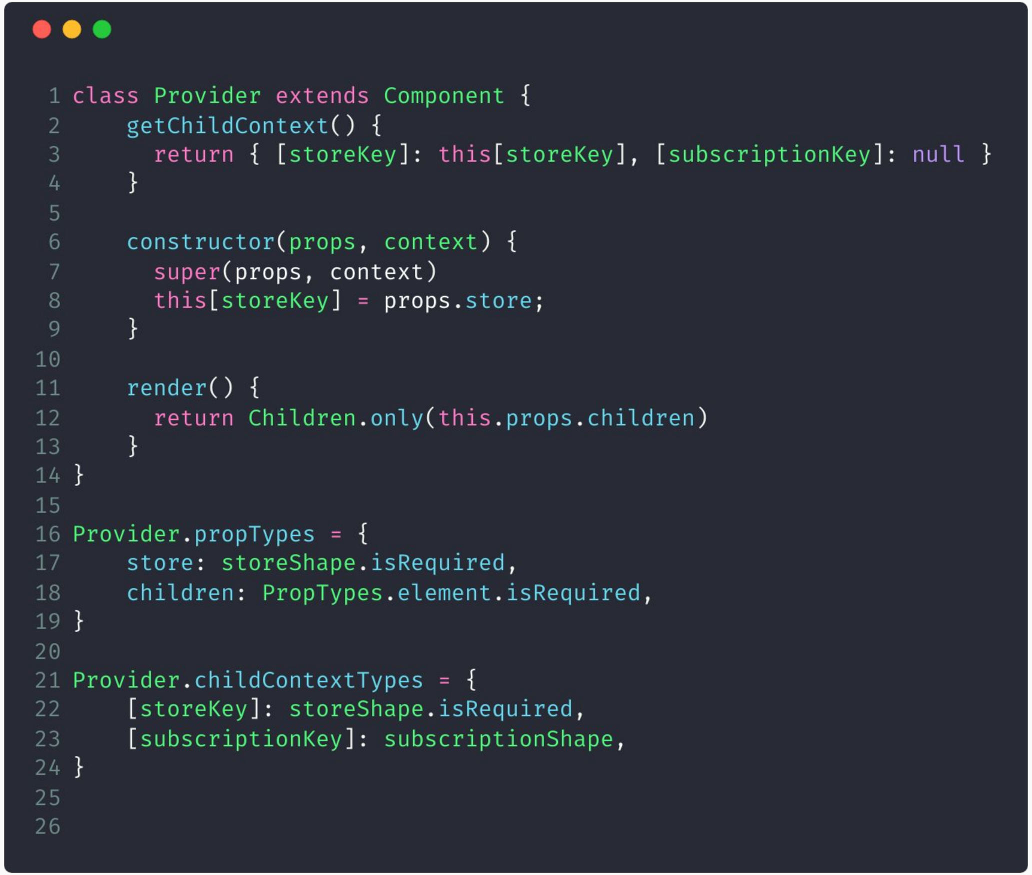
Demo

https://github.com/dmmulroy/react-md-context-talk

WHAT IF I TOLD YOU
REDUX USES THE CONTEXT API
imgflip.com

Spoiler Alert

(it does)



```
class Provider extends Component {
    getChildContext() {
      return { [storeKey]: this[storeKey], [subscriptionKey]: null }
    }

    constructor(props, context) {
      super(props, context)
      this[storeKey] = props.store;
    }

    render() {
      return Children.only(this.props.children)
    }
}

Provider.propTypes = {
    store: storeShape.isRequired,
    children: PropTypes.element.isRequired,
}

Provider.childContextTypes = {
    [storeKey]: storeShape.isRequired,
    [subscriptionKey]: subscriptionShape,
}
```

https://github.com/reduxjs/react-redux/blob/master/src/components/Provider.js

To summarize

- React 16.3 introduced a new top level API for Context
- Reach for the Context API when “lifting state” starts introducing multiple levels of prop drilling
- 0 dependencies
- You might not need Redux.

Find Me @

- github: https://github.com/dmmulroy
- twitter: https://twitter.com/dillon_mulroy
- gmail: dillon.mulroy@gmail.com