

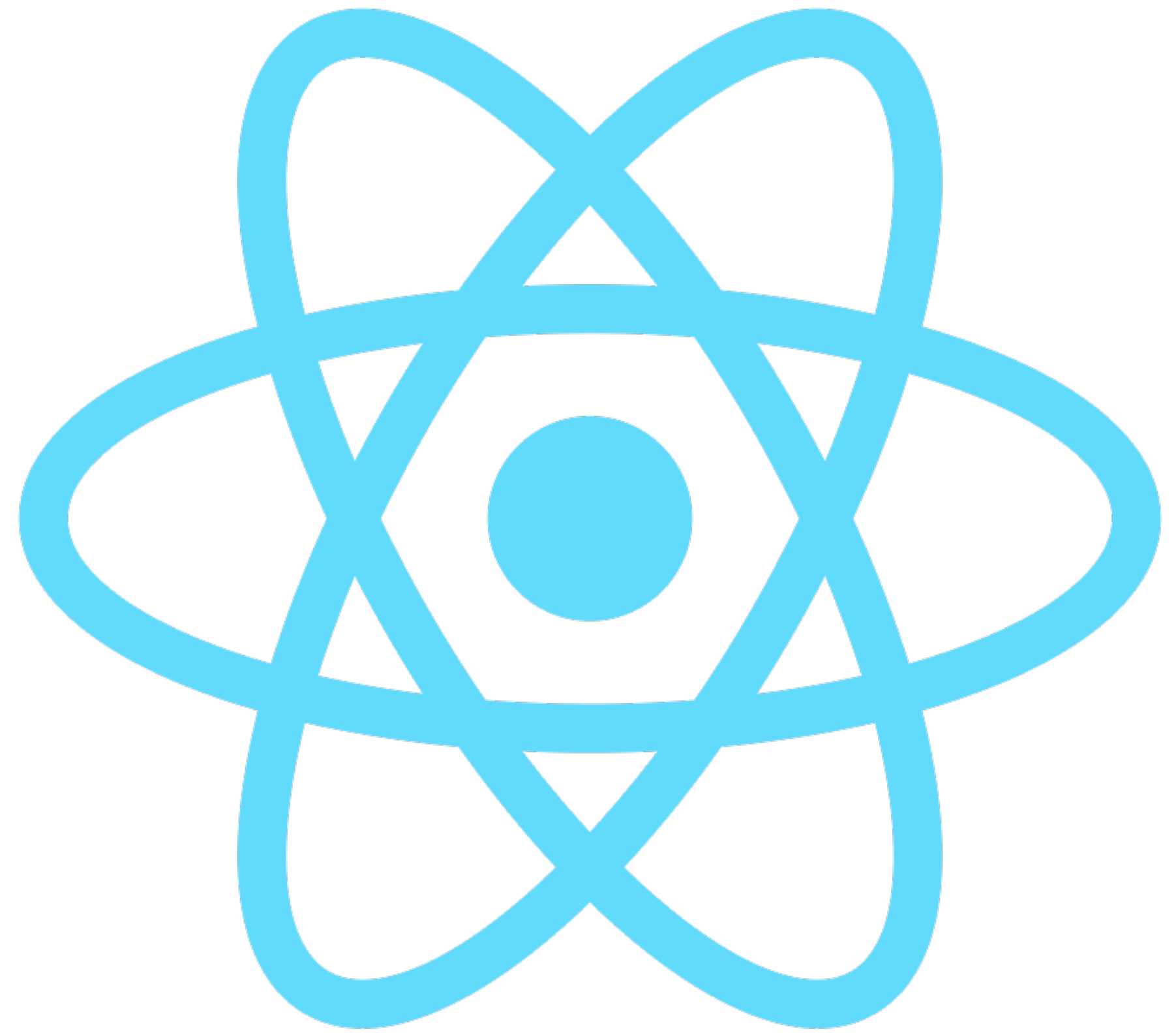
# React and Flux

@kevinold

[kevin@kevinold.com](mailto:kevin@kevinold.com)

# React

... is a  
JavaScript  
library for  
creating user  
interfaces



# Why React?

Simplify code

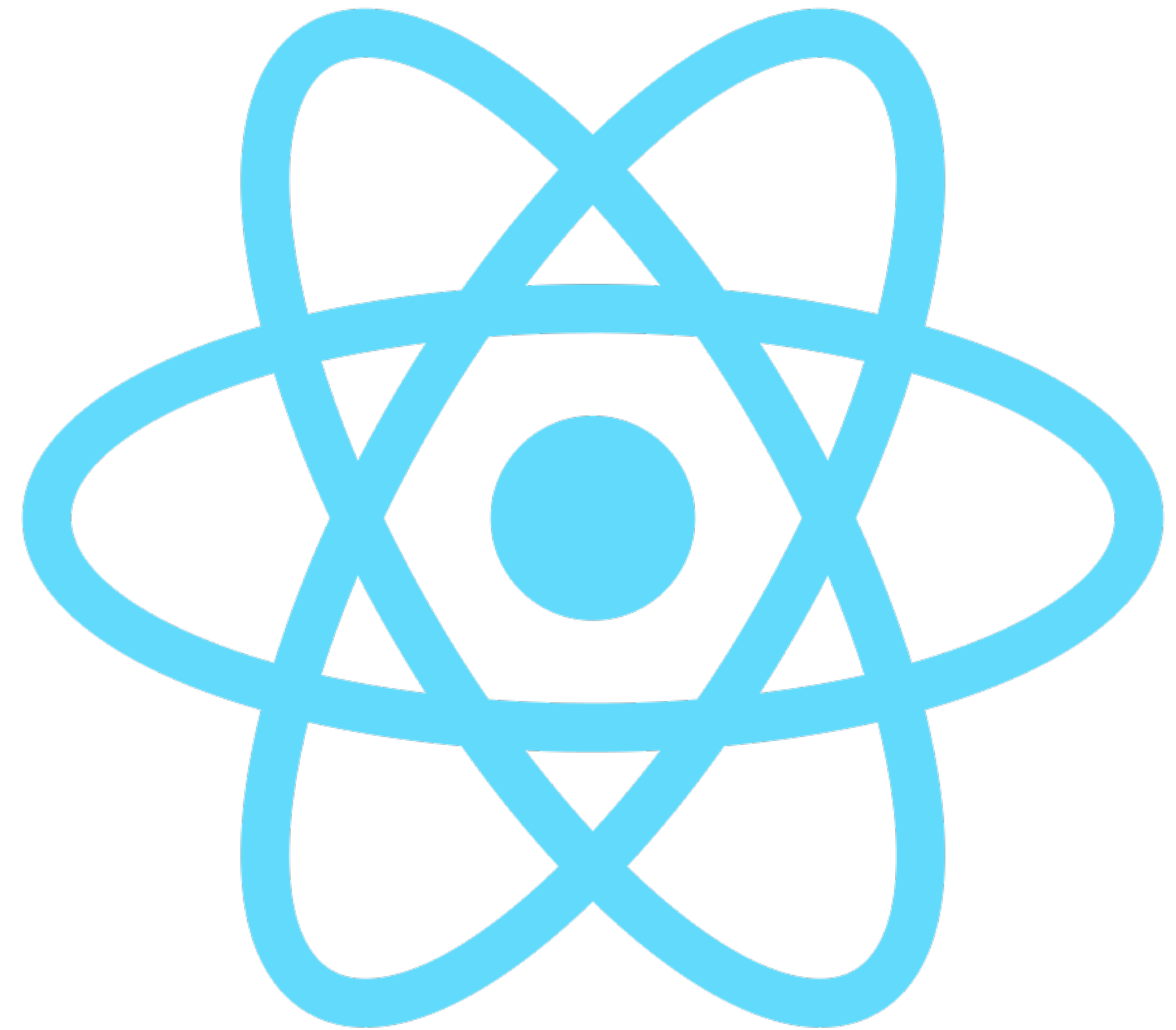
Build Reusable Components

Virtual DOM

Build Isomorphic apps

Eliminate querying / modifying  
DOM

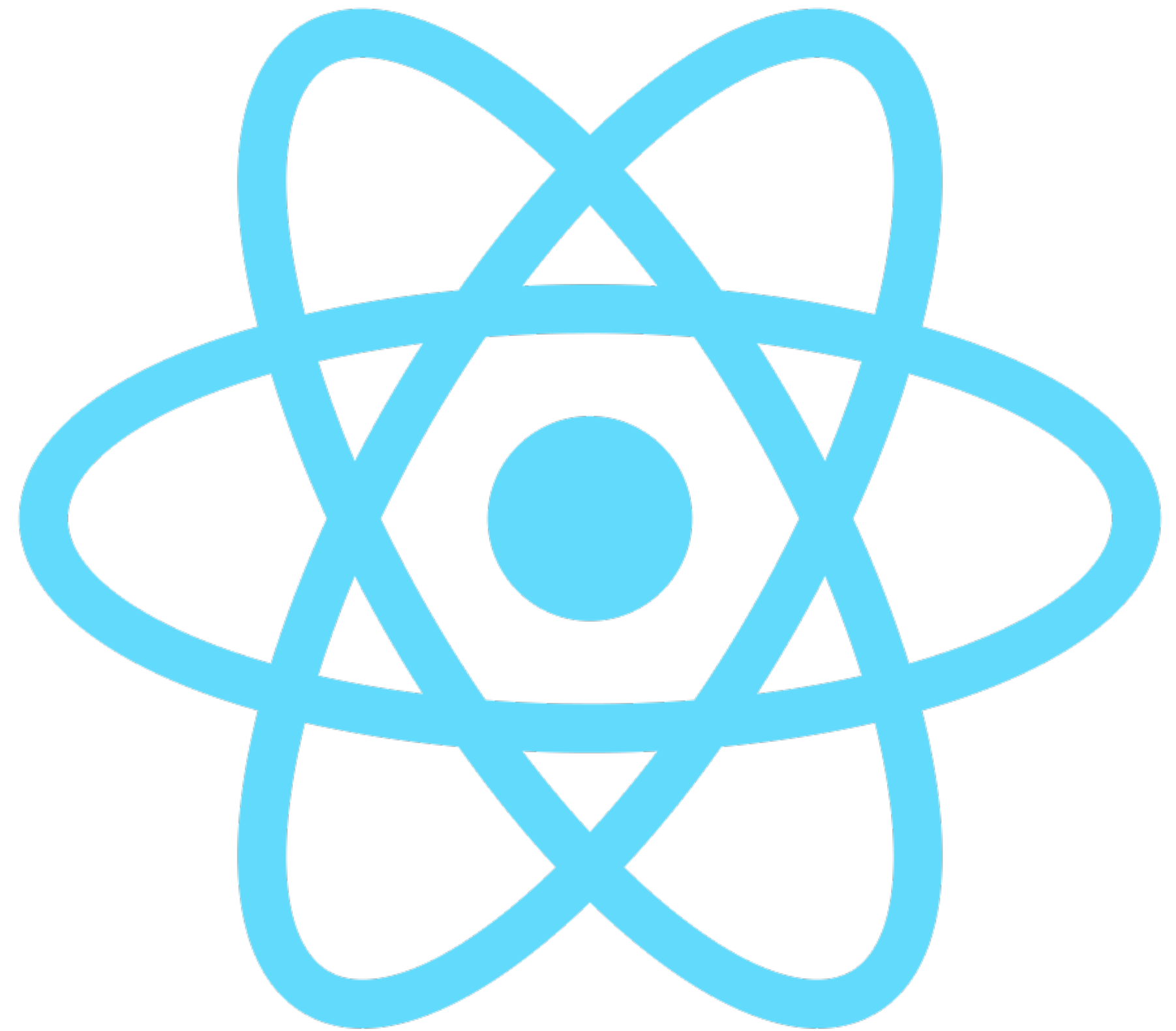
Small API / Easy to learn &  
remember



Building large applications with data that changes  
over time

Simple

Declarative



# What we'll cover

JSX

Components

Props

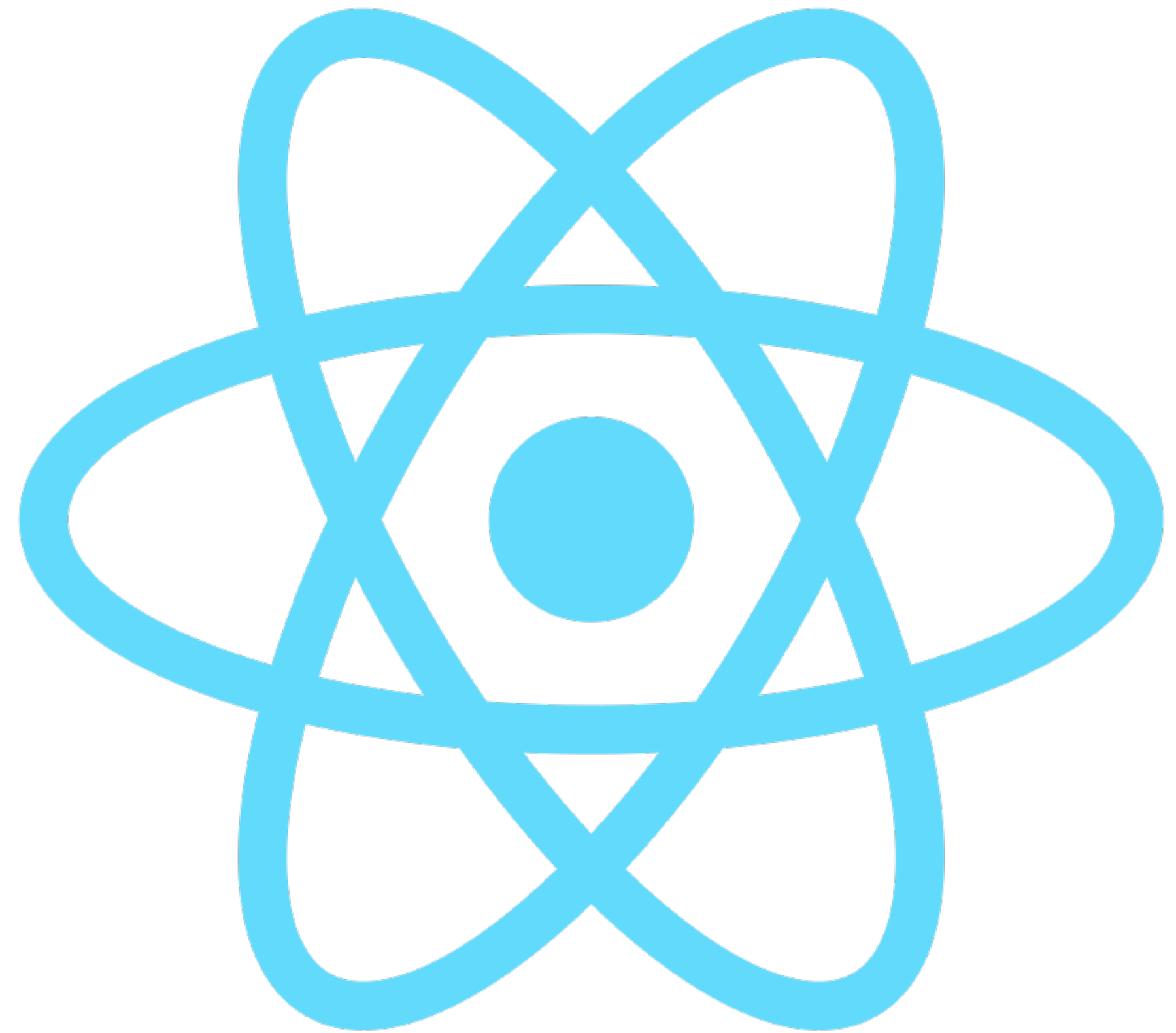
State

Component Lifecycle

App using only React

App using React + Flux

Tips

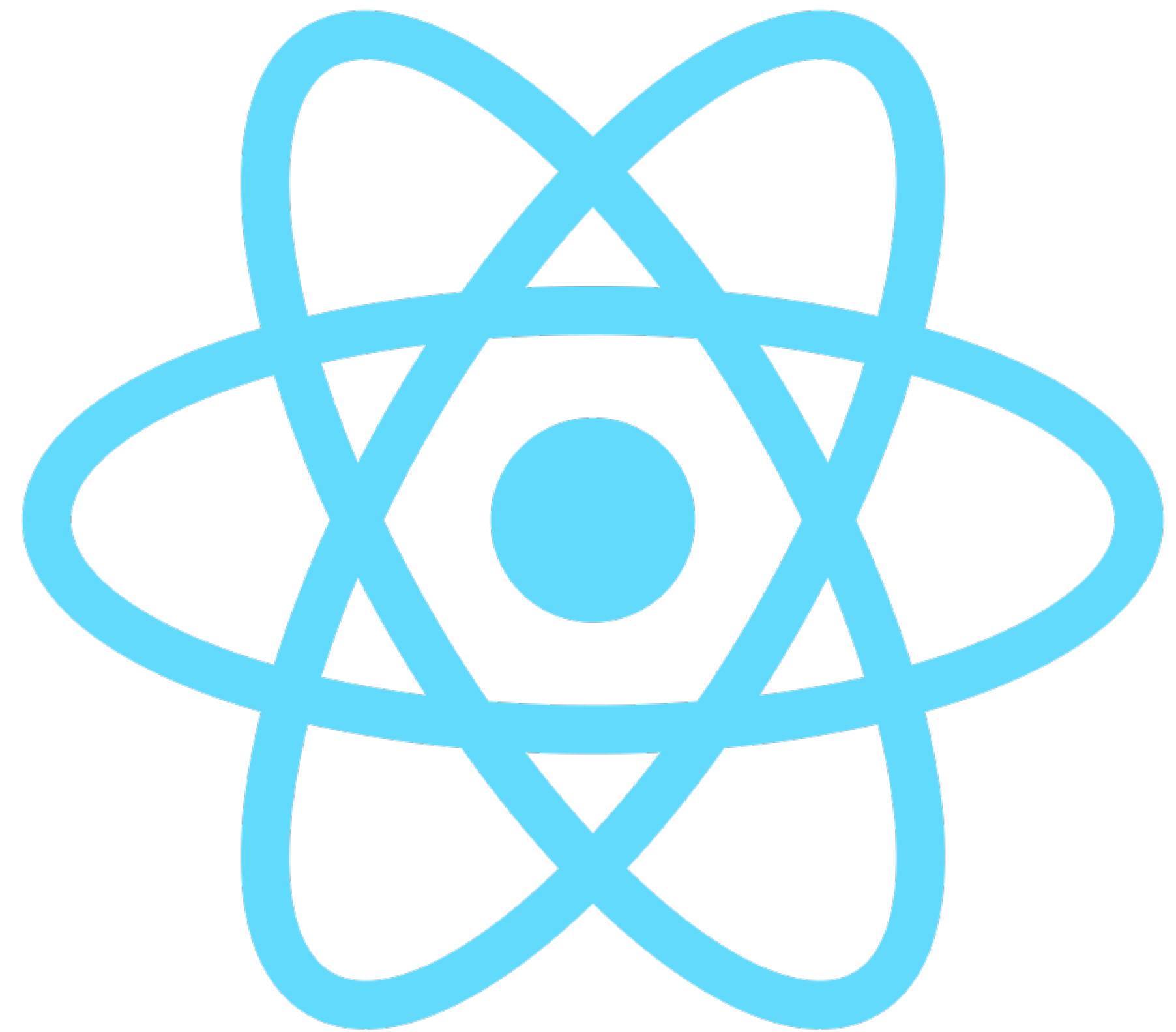


# About the code snippets

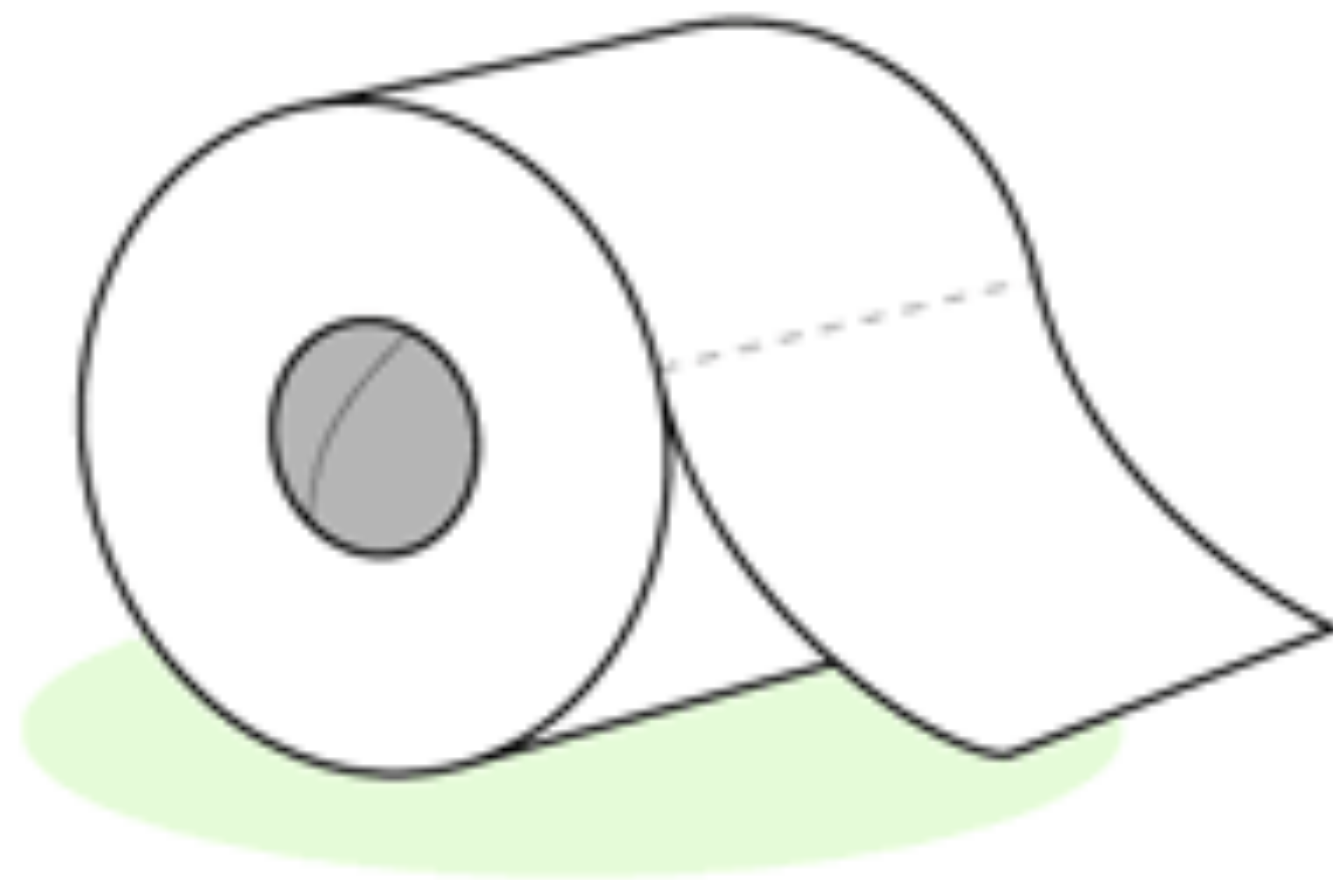
Written with JSX

CommonJS syntax

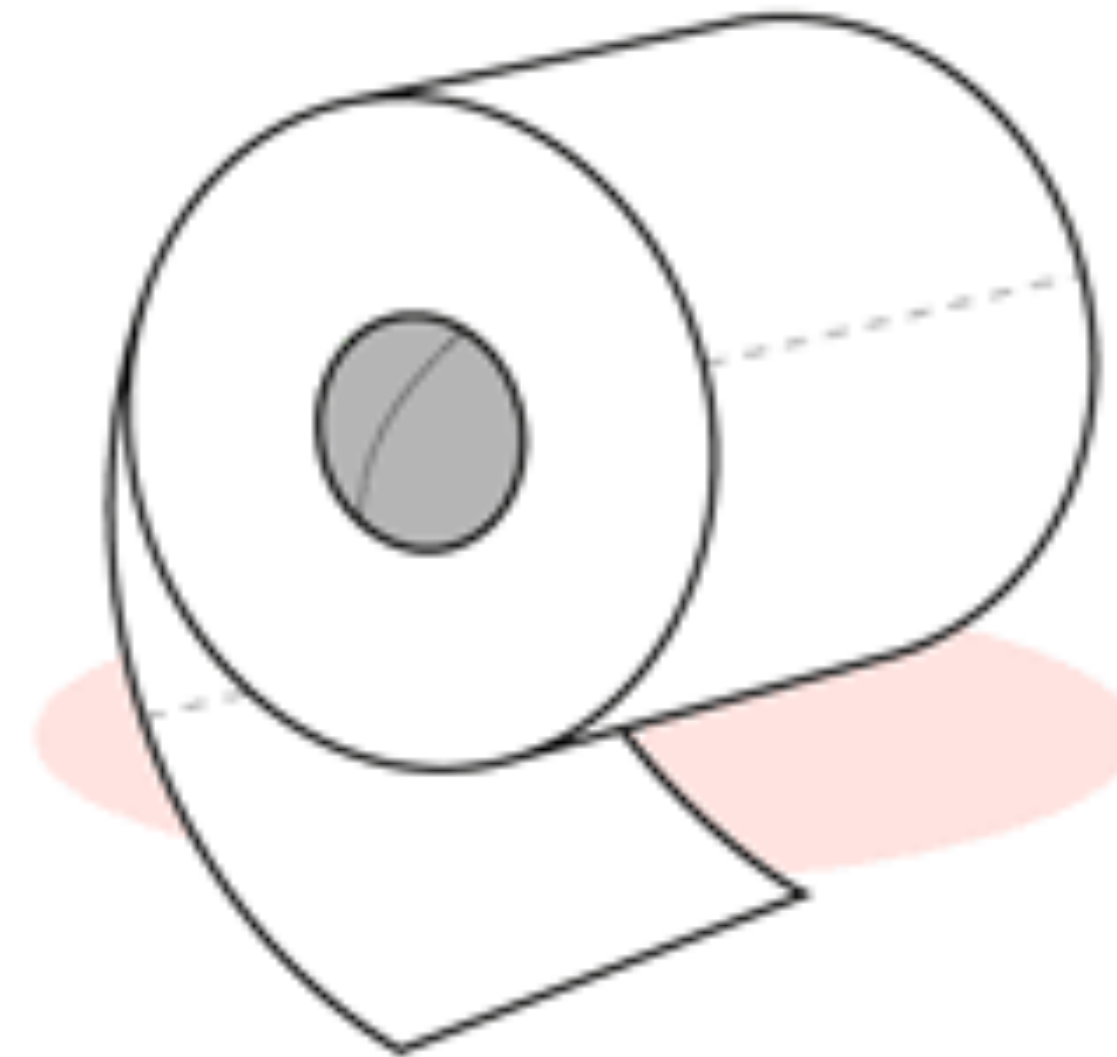
Packed with Browserify



# JSX (Controversy)



This is right.  
(good)



This is wrong.  
(bad)

# HTML IN MY JS?

```
// JSX
React.render(
  <h1>Hello, world!</h1>,
  document.getElementById('example')
);
```

```
// JavaScript
React.render(
  // type, props, children
  React.createElement('h1', null, 'Hello, world!'),
  document.getElementById('example')
);
```



# Live JSX Compiler



DOCS SUPPORT DOWNLOAD BLOG GITHUB

## JSX Compiler

This tool demonstrates how **JSX syntax** is desugared into native JavaScript.

### Live JSX Editor

```
var HelloMessage = React.createClass({
  render: function() {
    return <div className="bold">Hello {this.props.name}</div>;
  }
});

React.render(<HelloMessage name="John" />, mountNode);
```

```
var HelloMessage = React.createClass({displayName: "HelloMessage",
  render: function() {
    return React.createElement("div", {className: "bold"}, "Hello ",
    this.props.name);
  }
});

React.render(React.createElement(HelloMessage, {name: "John"}),
mountNode);
```

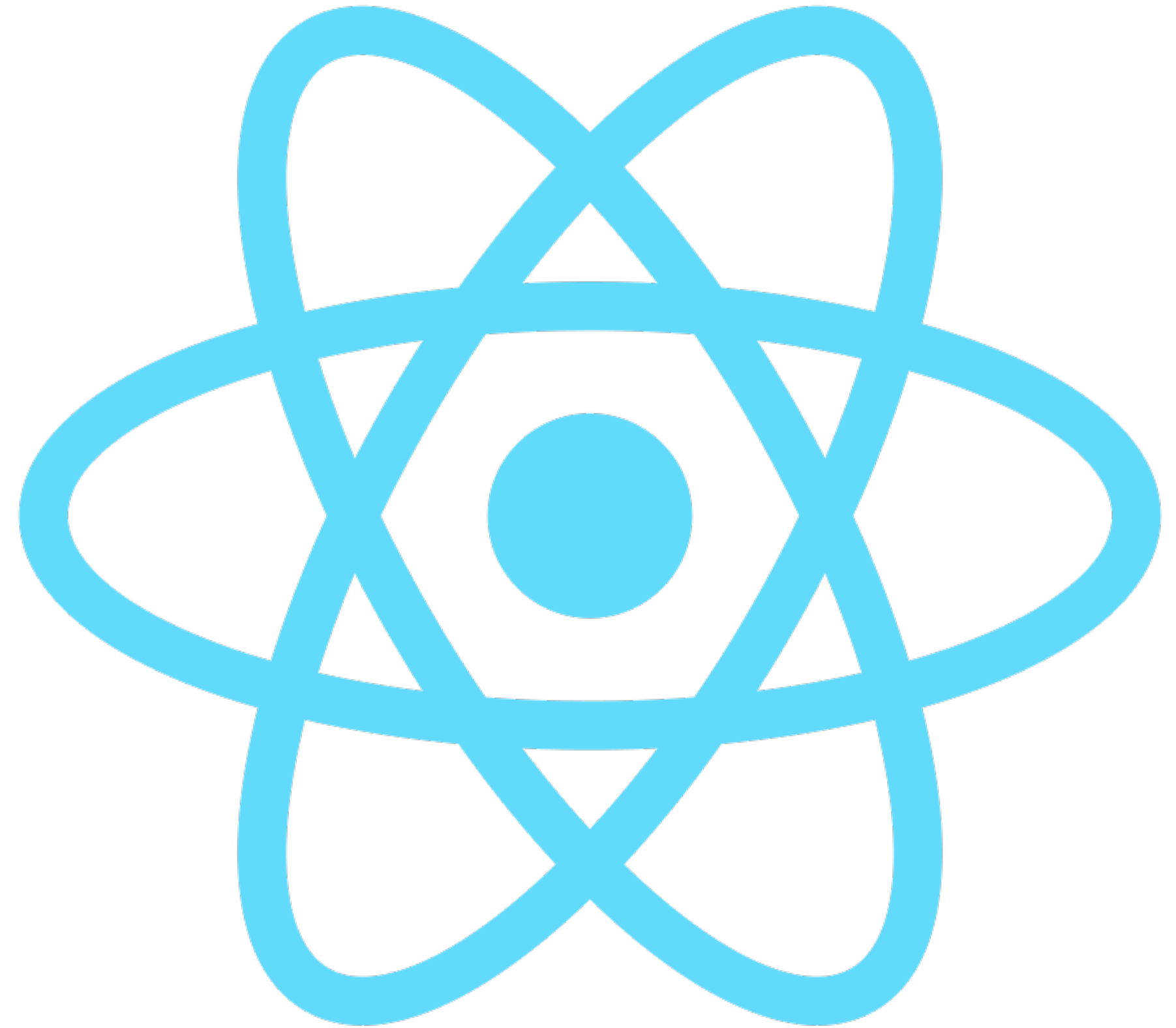
Enable ES6 transforms (--harmony)

A Facebook & Instagram collaboration.  
[Acknowledgements](#)

© 2013–2014 Facebook Inc.  
Documentation licensed under [CC BY 4.0](#).

<http://facebook.github.io/react/jsx-compiler.html>

# Components



# Component Methods

`render()` - required

`getInitialState()`

`getDefaultProps()`

# render()

```
var Div = React.createClass({  
  render: function() {  
    return (  
      <div> Hello World </div>  
    );  
  }  
});  
  
// In other components (JSX)  
<Div />
```

# getDefaultProps()

```
var Div = React.createClass({  
  
  getDefaultProps: function() {  
    return { name: 'Sir' };  
  },  
  
  render: function() {  
    return (  
      <div> Hello {this.props.name} </div>  
    );  
  }  
});
```

*// In other components (JSX)*

```
<Div name="Kevin" />
```

# getInitialState()

```
var Div = React.createClass({  
  
  getInitialState: function() {  
    return getDataFromAPI();  
  },  
  
  render: function() {  
    return (  
      <div> Hello {this.state.name} </div>  
    );  
  }  
});
```

*// In other components (JSX)*

```
<Div />
```

# Additional Component Methods

propTypes (object)

mixins (array)

statics (object)

# Props

```
// Contained in ReportList  
<ReportRow  
  key={report.id}  
  report={report}  
/>
```

```
<DatePicker  
  format="MM/DD/YYYY"  
  initialDate={today}  
/>
```

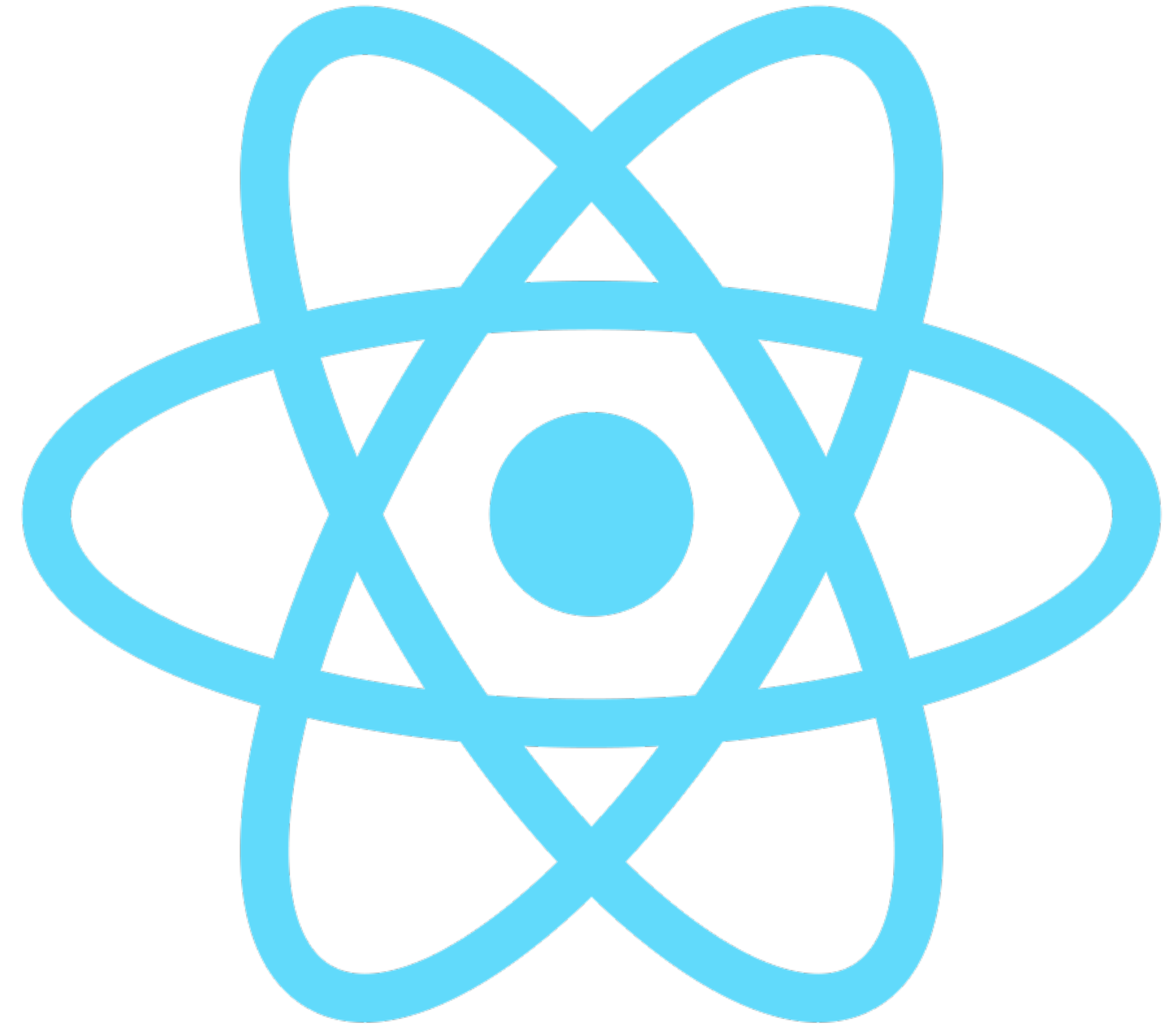


# State

```
var Checkbox = React.createClass({
  getInitialState: function() {
    return {
      isChecked: this.props.initialChecked || false
    };
  },

  _toggleCheckbox: function() {
    this.setState({ isChecked: !this.state.isChecked });
  },
});
```

# Component Lifecycle



# Component Lifecycle

## **Mounting:**

componentWillMount()

componentDidMount()\*

## **Updating:**

componentWillReceiveProps()

shouldComponentUpdate()\*

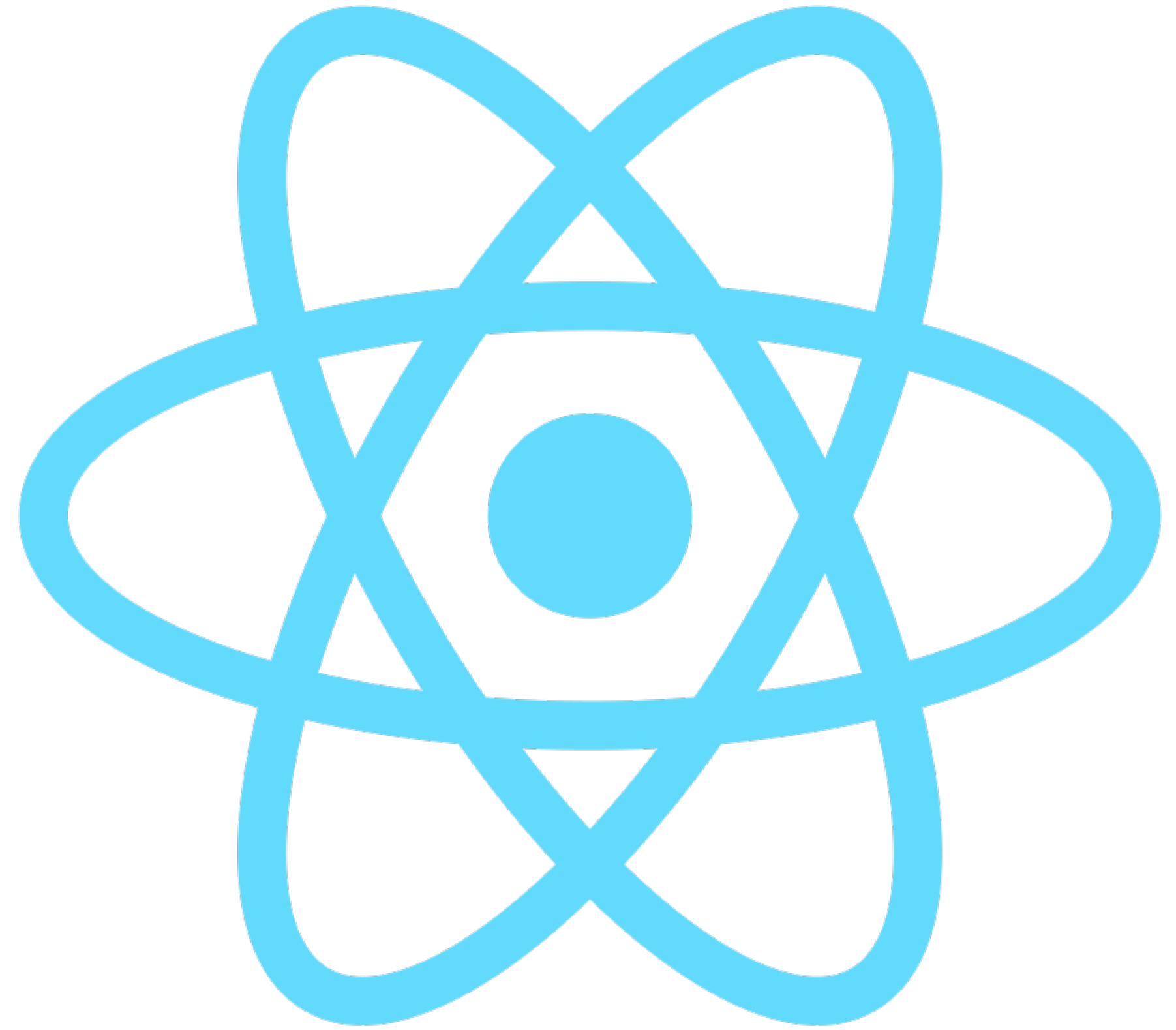
componentWillUpdate()

componentDidUpdate()\*

## **Unmounting:**

componentWillUnmount()

# Reusable Components



# Checkbox

```
var Checkbox = React.createClass({
  getInitialState: function() {
    return {
      isChecked: this.props.initialChecked || false
    };
  },

  _toggleCheckbox: function() {
    this.setState({ isChecked: !this.state.isChecked });
  },
});
```

# Checkbox (cont.)

```
render: function () {  
  return (  
    <div className="checkbox">  
      <label>  
        <input type="checkbox"  
          checked={ this.state.isChecked }  
          ref={ this.props.name }  
          onChange={ this._toggleCheckbox }  
        /> { this.props.label }  
      </label>  
    </div>  
  );  
}  
});
```

# Lists

```
render: function () {  
  
    var items = [];  
  
    _.each(this.state.expenseItems, function(expenseItem) {  
  
        items.push(<ExpenseItemRow  
                    key={expenseItem.id}  
                    expenseItem={expenseItem} />);  
  
    });  
  
    return (  
        <div className="list-group">{items}</div>  
    );  
}
```

# Compose App of Components

```
<ReportApp> ← “Owns State”  
  <Header />  
  <ReportList>  
    <ReportRow />  
    <ReportRow />  
    <ReportRow />  
  </ReportList>  
  <ItemList> ← “Owns State”  
    <ItemRow />  
    <ItemEditRow /> ← “Owns State”  
      <ItemInput />  
      <CategoryDropdown />  
    <ItemRow />  
  </ItemList>  
</ReportApp>
```



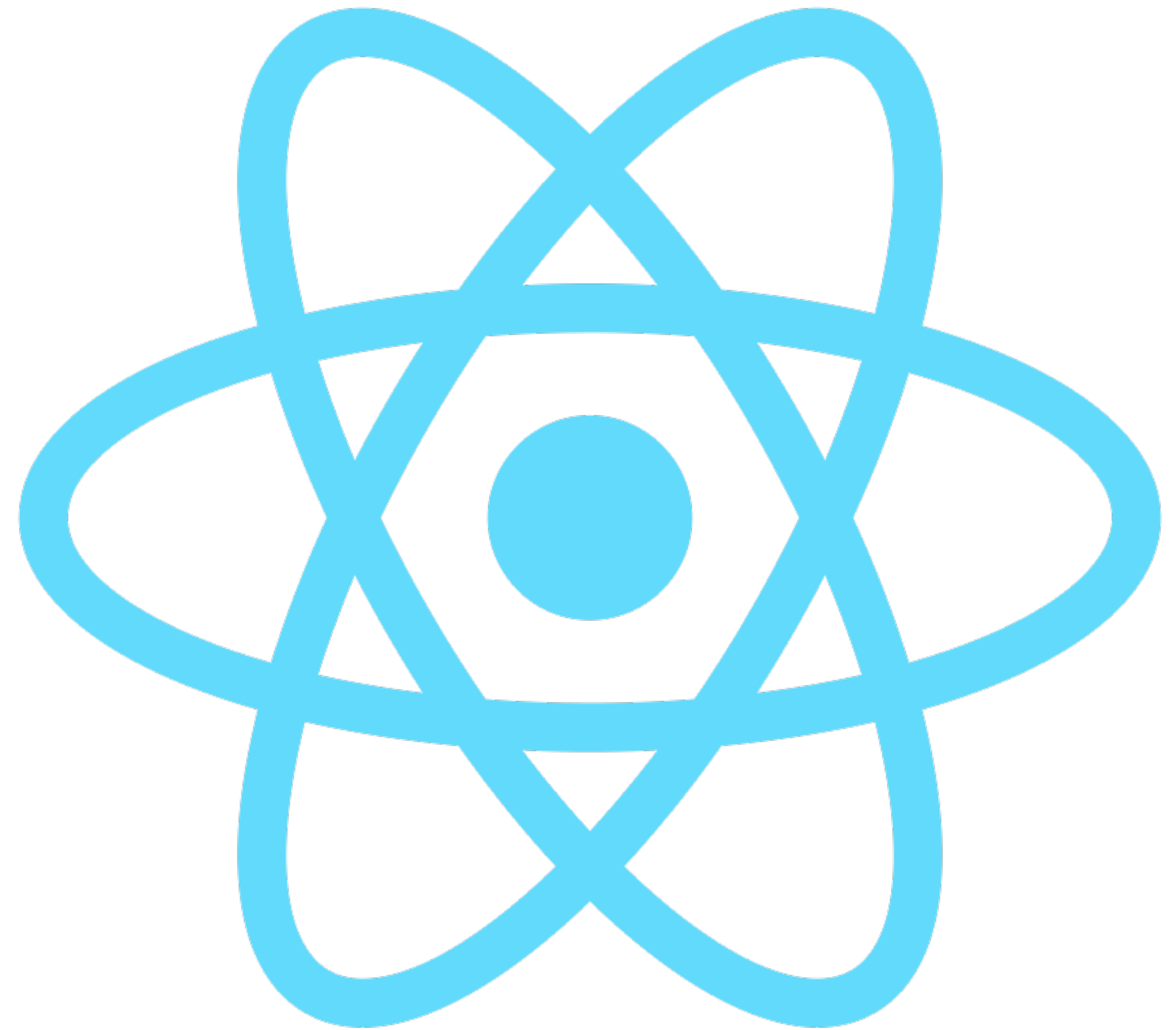
# Pure Component

```
var Div = React.createClass({  
  render: function() {  
    return (  
      <div> Hello World </div>  
    );  
  }  
});
```

Component with State

```
var ExpenseRowForm = React.createClass({
  getInitialState: function() { return { merchant: '' }; },
  handleChange: function(event) {
    this.setState({ merchant: event.target.value });
  },
  render: function() {
    var expense = this.props.expense;
    var formattedTotal = accounting.formatMoney(expense.total);
    return (
      <li href="#" className="list-group-item">
        <form action="">
          <input type="text"
            defaultValue={expense.merchant}
            onChange={ this.handleChange } />
          <button onChange={ this.toggleRow }>Save</button>
        </form>
      </li>
    );
  }
});
```

Tips & Tricks



# Using jQuery inside components

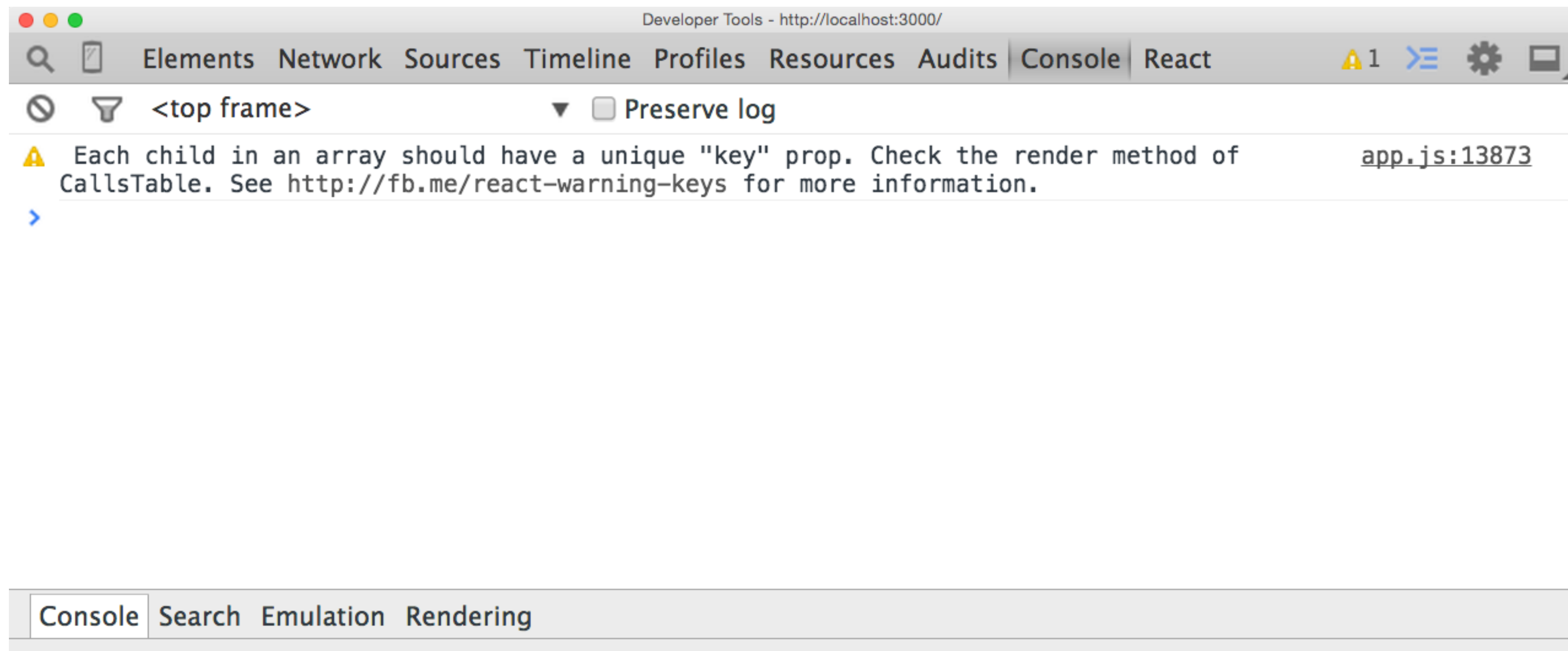
```
var Attachment = React.createClass({
  componentDidMount: function() {
    var upload = jQuery(this.refs.fileInput.getDOMNode());
    upload.uploadify({
      // ...
    });
  },
  render: function() {
    return (
      <input ref="fileInput" type="file" />
    );
  }
});
```

# renderToString()

```
var React = require('react/addons');
require('node-jsx').install();
var App = React.createFactory(require('./js/App.react'));
// ...
server.route({
  method: 'GET',
  path: '/',
  handler: function (request, reply) {
    var reactHtml = React.renderToString(App({}));
    reply.view('index', { reactHtml: reactHtml });
  }
});

// Template
<div id="app">{{ reactHtml }}</div>
```

# React has great logs



# Recommendations

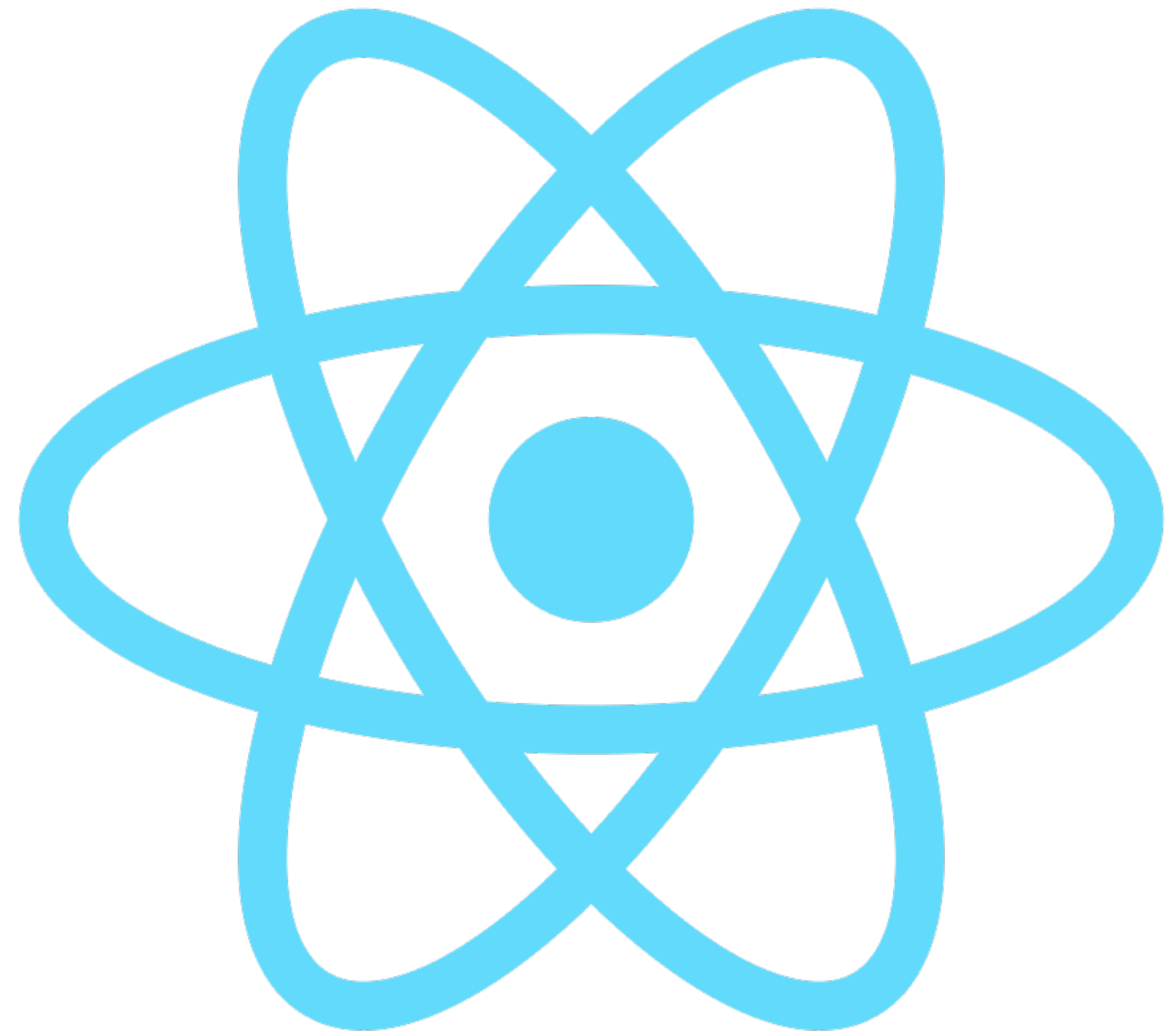
CommonJS

Browserify

JSX

`npm install react`

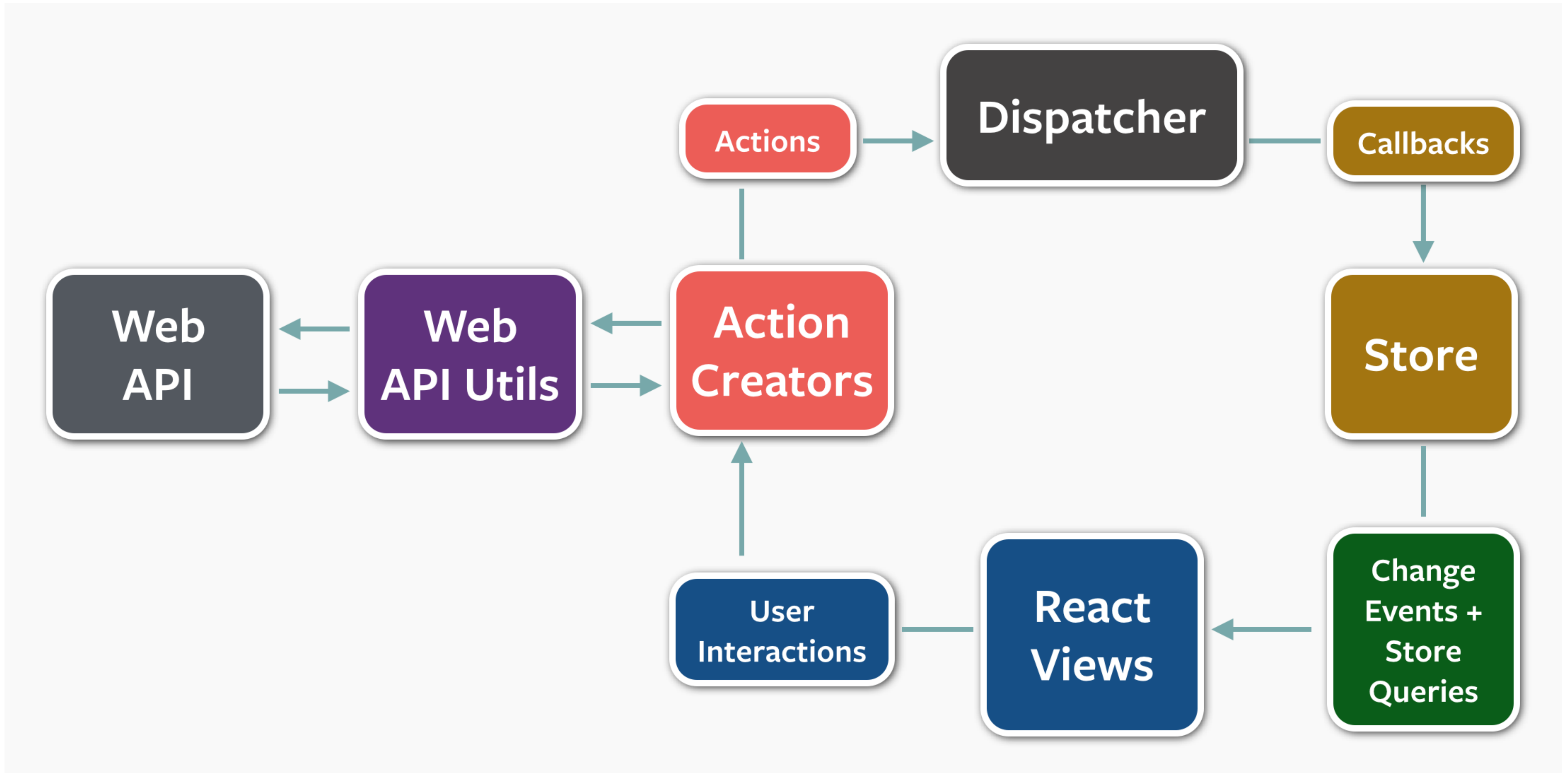
`npm install react-tools`

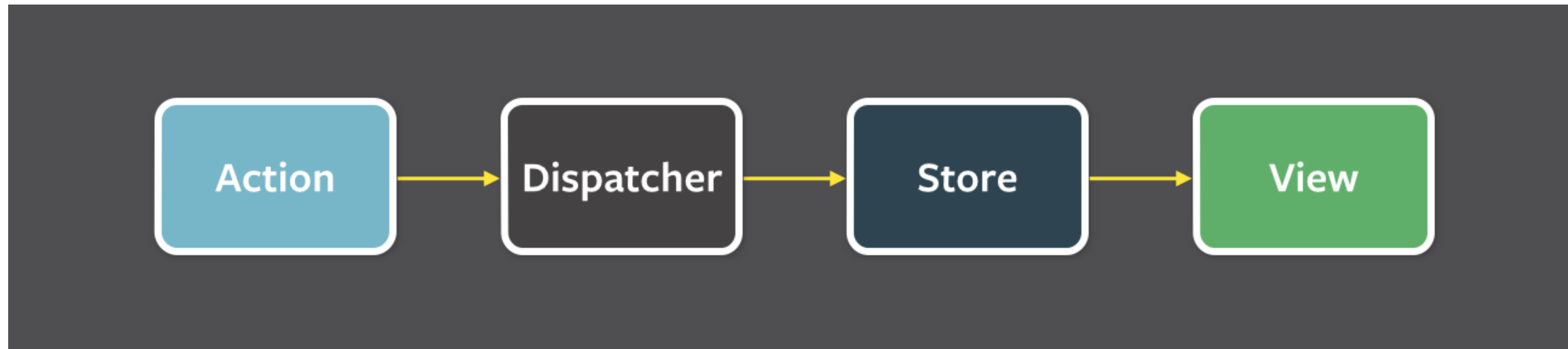




React Questions?  
(Flux is up next)

# Flux





# Simple Structure and Data Flow

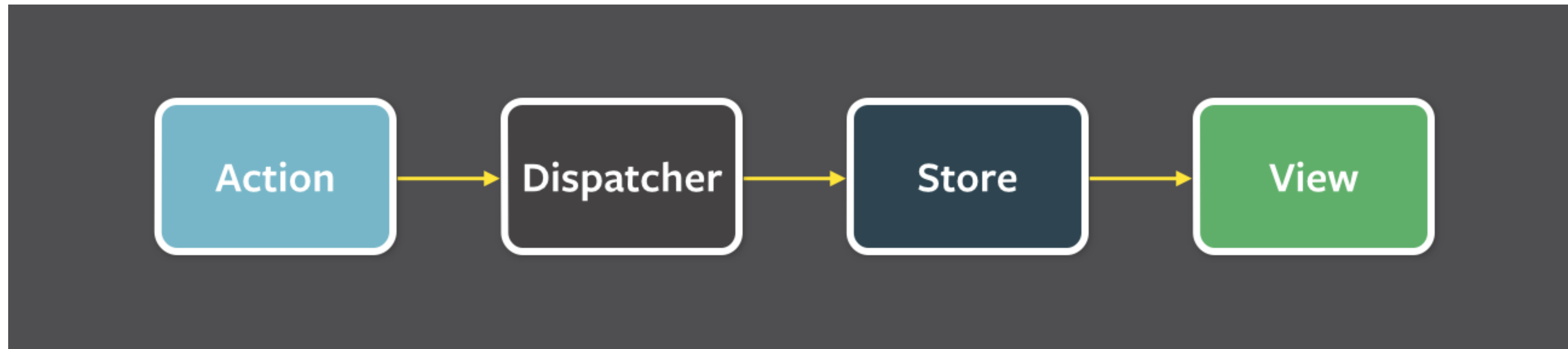
# Expense Reports

<b>San Mateo Trip</b> >
ID: 1 Start: 8/25 End: 8/29 123
<b>FL Trip</b> >
ID: 2 Start: 4/2 End: 5/15 123
<b>Atlanta Trip</b> >
ID: 3 Start: 1/12 End: 1/19 123

## Expenses for San Mateo Trip

10

<b>Lemke-Fadel</b> ID: 500 Description: voluptas dolores vel	\$86.00
<b>Bruen-Toy</b> ID: 315 Description: maxime minus rerum ipsum	\$43.00
<b>Brown, Mosciski and McGlynn</b> ID: 879 Description: corrupti ut	\$50.00
<b>White, Barrows and Bartell</b> ID: 513 Description: aut eveniet ex amet dicta	\$76.00
<b>Bartoletti and Sons</b> ID: 369 Description: vel numquam sunt dicta ullam	\$83.00
<b>Pagac Group</b> ID: 184 Description: dolore expedita quis hic esse blanditiis	\$26.00
<b>Berge, Schowalter and Herman</b> ID: 340 Description: corrupti exercitationem	\$120.00
<b>Kuvalis-Vandervort</b> ID: 222	\$100.00



# Actions

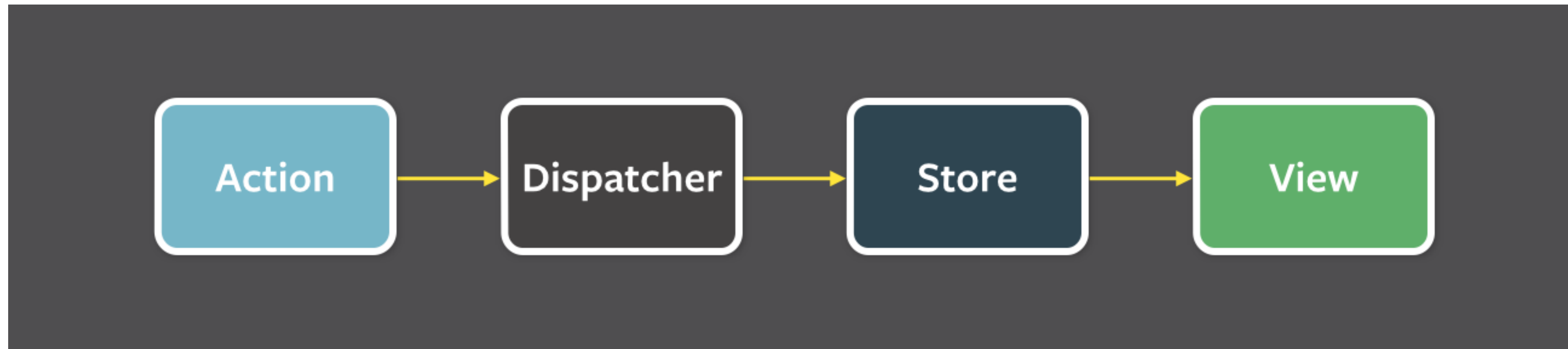
```
var AppDispatcher = require('../dispatcher/AppDispatcher');
var AppConstants = require('../constants/AppConstants');

var ActionTypes = AppConstants.ActionTypes;

module.exports = {

  clickReport: function(reportID) {
    AppDispatcher.handleViewAction({
      type: ActionTypes.CLICK_REPORT,
      reportID: reportID
    });
  }

};
```



# Dispatcher

```
var AppConstants = require('../constants/AppConstants');
var Dispatcher = require('flux').Dispatcher;
var assign = require('object-assign');

var PayloadSources = AppConstants.PayloadSources;

var AppDispatcher = assign(new Dispatcher(), {

  handleViewAction: function(action) {
    var payload = {
      source: PayloadSources.VIEW_ACTION,
      action: action
    };
    this.dispatch(payload);
  }

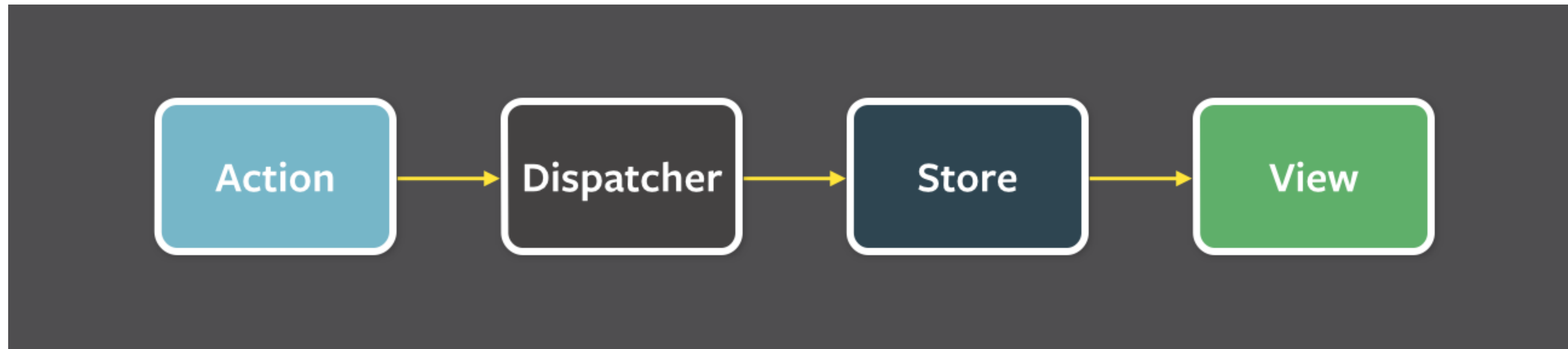
});
```



```
/*  
 * Copyright (c) 2014, Facebook, Inc.  
 * All rights reserved.  
 *  
 * This source code is licensed under the BSD-style license found in the  
 * LICENSE file in the root directory of this source tree. An additional grant  
 * of patent rights can be found in the PATENTS file in the same directory.  
 *  
 * AppDispatcher  
 *  
 * A singleton that operates as the central hub for application updates.  
 */
```

```
var Dispatcher = require('flux').Dispatcher;
```

```
module.exports = new Dispatcher();
```



Store(s)

```
var AppDispatcher = require('../dispatcher/AppDispatcher');
var AppConstants = require('../constants/AppConstants');
var EventEmitter = require('events').EventEmitter;
var assign = require('object-assign');

var ActionTypes = AppConstants.ActionTypes;
var CHANGE_EVENT = 'change';

var _currentID = 1;
var _reports = [
  { id: 1, title: "San Mateo Trip", start: "8/25", end:"8/29" },
  { id: 2, title: "FL Trip", start: "4/2", end:"5/15" },
  { id: 3, title: "Atlanta Trip", start: "1/12", end:"1/19" }
];
```

```
var ExpenseReportStore = assign({}, EventEmitter.prototype, {  
  
  emitChange: function() {  
    this.emit(CHANGE_EVENT);  
  },  
  
  addChangeListener: function(callback) {  
    this.on(CHANGE_EVENT, callback);  
  },  
  
  removeChangeListener: function(callback) {  
    this.removeListener(CHANGE_EVENT, callback);  
  },  
}
```

```
get: function(id) {  
    return _reports[id];  
},  
  
getCurrentReport: function() {  
    return _reports[this.getCurrentID()];  
},  
  
getAll: function() {  
    return _reports;  
},  
  
getCurrentID: function() {  
    return _currentID;  
},  
  
});
```

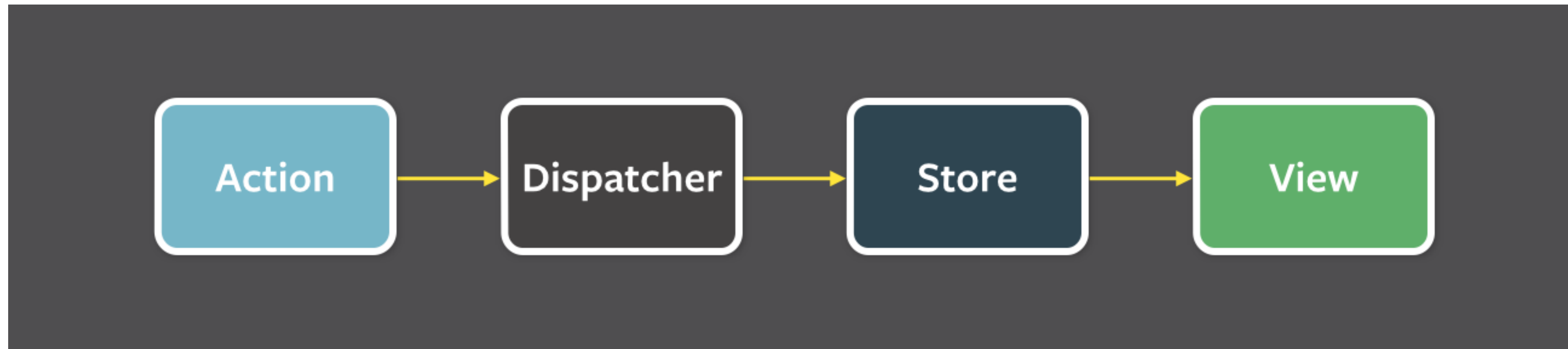
```
ExpenseReportStore.dispatchToken =
AppDispatcher.register(function(payload) {
  var action = payload.action;

  switch(action.type) {

    case ActionTypes.CLICK_REPORT:
      _currentID = action.reportID;
      ExpenseReportStore.emitChange();
      break;

    default:
      // do nothing
  }

});
```



View (React)

# Expense Reports

<b>San Mateo Trip</b> >
ID: 1 Start: 8/25 End: 8/29 123
<b>FL Trip</b> >
ID: 2 Start: 4/2 End: 5/15 123
<b>Atlanta Trip</b> >
ID: 3 Start: 1/12 End: 1/19 123

## Expenses for San Mateo Trip

10

<b>Lemke-Fadel</b> ID: 500 Description: voluptas dolores vel	\$86.00
<b>Bruen-Toy</b> ID: 315 Description: maxime minus rerum ipsum	\$43.00
<b>Brown, Mosciski and McGlynn</b> ID: 879 Description: corrupti ut	\$50.00
<b>White, Barrows and Bartell</b> ID: 513 Description: aut eveniet ex amet dicta	\$76.00
<b>Bartoletti and Sons</b> ID: 369 Description: vel numquam sunt dicta ullam	\$83.00
<b>Pagac Group</b> ID: 184 Description: dolore expedita quis hic esse blanditiis	\$26.00
<b>Berge, Schowalter and Herman</b> ID: 340 Description: corrupti exercitationem	\$120.00
<b>Kuvalis-Vandervort</b> ID: 222	\$100.00



```
var React = require('react');
var cx = require('react/lib/cx');
var ExpenseReportStore = require('../stores/ExpenseReportStore');
var ExpenseReportActions = require('../actions/ExpenseReportActionCreators');

function getCurrentID() {
  return { currentID: ExpenseReportStore.getCurrentID() };
}

var ExpenseReportRow = React.createClass({
  getInitialState: function() {
    return getCurrentID();
  },
```

```
componentWillMount: function() {  
  ExpenseReportStore.addChangeListener(this._onChange);  
},  
  
componentUnWillMount: function() {  
  ExpenseReportStore.removeChangeListener(this._onChange);  
},  
  
_onChange: function() {  
  this.setState(getCurrentID());  
},  
  
handleClick: function() {  
  ExpenseReportActions.clickReport(this.props.expenseReport.id);  
},
```

```
render: function () {
  var grandTotal = generateGrandTotal(er);
  var classes = {
    'list-group-item': true,
    'active': this.state.currentID === this.props.expenseReport.id
  };
  return (
    <li className={cx(classes)} onClick={this.handleClick}>

      <h4 className="list-group-item-heading">
        {this.props.expenseReport.title}
      </h4>

      <span className="badge">{grandTotal}</span>
    </li>
  )
}
});
```

# Expense Reports

<b>San Mateo Trip</b> >
ID: 1 Start: 8/25 End: 8/29 123
<b>FL Trip</b> >
ID: 2 Start: 4/2 End: 5/15 123
<b>Atlanta Trip</b> >
ID: 3 Start: 1/12 End: 1/19 123

## Expenses for San Mateo Trip

10

<b>Lemke-Fadel</b> ID: 500 Description: voluptas dolores vel	\$86.00
<b>Bruen-Toy</b> ID: 315 Description: maxime minus rerum ipsum	\$43.00
<b>Brown, Mosciski and McGlynn</b> ID: 879 Description: corrupti ut	\$50.00
<b>White, Barrows and Bartell</b> ID: 513 Description: aut eveniet ex amet dicta	\$76.00
<b>Bartoletti and Sons</b> ID: 369 Description: vel numquam sunt dicta ullam	\$83.00
<b>Pagac Group</b> ID: 184 Description: dolore expedita quis hic esse blanditiis	\$26.00
<b>Berge, Schowalter and Herman</b> ID: 340 Description: corrupti exercitationem	\$120.00
<b>Kuvalis-Vandervort</b> ID: 222	\$100.00

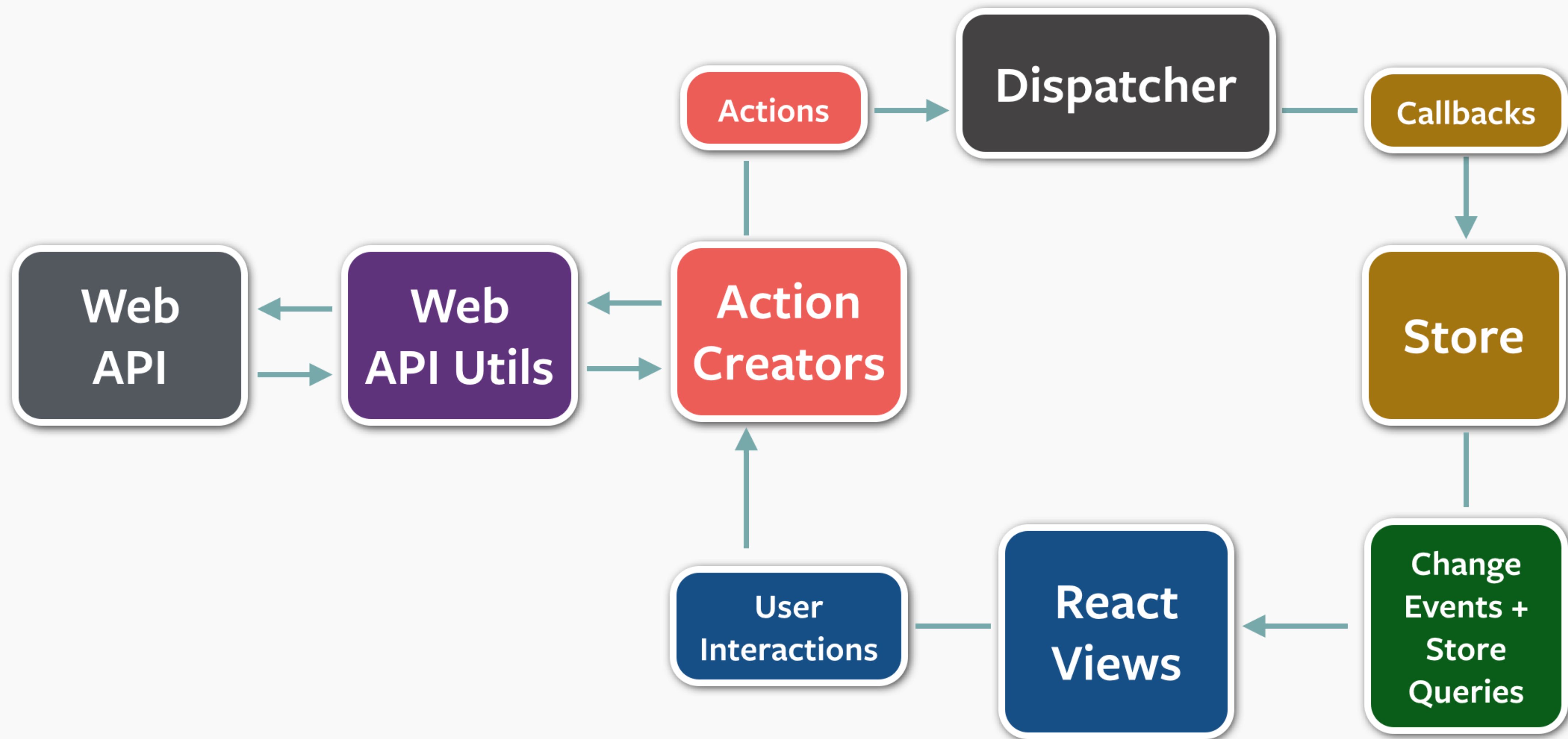
```
var ExpenseItemList = React.createClass({
  getInitialState: function() {
    return getStateFromStores();
  },
  componentWillMount: function() {
    ExpenseReportStore.addChangeListener(this._onChange);
    ExpenseItemStore.addChangeListener(this._onChange);
  },
  componentWillUnmount: function() {
    ExpenseReportStore.removeChangeListener(this._onChange);
    ExpenseItemStore.removeChangeListener(this._onChange);
  },
  _onChange: function() {
    this.setState(getStateFromStores());
  },
});
```

```
render: function () {
  var state = this.state;
  var items = this.state.expenseItems.map(function (i) {
    if (state.currentExpenseItemID === i.id) {
      items.push(<ExpenseItemEditRow
        key={i.id} expenseItem={i} />);
    } else {
      items.push(<ExpenseItemRow
        key={i.id} expenseItem={i} />);
    }
  });

  return ( <div className="list-group"> {items} </div> );
}
});
```

# Recommended App Flow







# Resources

React site - <http://facebook.github.io/react/>

Flux - <http://facebook.github.io/flux/>

Thinking in React - <http://facebook.github.io/react/docs/thinking-in-react.html>

Egghead.io React Fundamentals - <https://egghead.io/series/react-fundamentals>

Egghead.io Flux Architecture - <https://egghead.io/series/react-flux-architecture>

“New” Dispatcher Example - <https://github.com/facebook/flux/commit/ec8bba6893da01dc0f4e7a136d47acfd2cea3ac4>

Thanks!  
Questions?