

React 101

Finally!



Remember when I said jQuery is awesome?

Surprise, it actually sucks!



Other
frameworks
made single
page apps
easier

- Utilized MVC architectures
- Enabled mapping data to HTML
- Allowed visual components to sync with DB data
- Let you break your application into “modules”



Other
frameworks
made single
page apps
easier

```
// main.js
var layout = Backbone.View.extend({
  template: "#layout-template",
  initialize: function() {
    this.model = new Backbone.model({
      foo: "bar"
    });
  }
});
```

```
app.main.showView(layout);
```

```
<!-- layout.tmpl -->
<script id="layout-template">
  <h1> <%= foo %> </h1>
</script>
```



The downsides...

- The code base gets super bloated
- calls to `render()` everywhere
- Too many frameworks and templating engines to learn
- Difficult to determine which functions get called when



Facebook invented React to rescue us from framework hell

Praise be to Mark Zuckerberg



The Virtual DOM

- React manages a virtual copy of the DOM
- Your code changes the virtual DOM, not the actual DOM
- React is optimized to make the minimum number of changes to the real DOM to match the virtual DOM



Components

- React ditches MVC
- Instead, they opt for a Component/OOP architecture
- No more jQuery, just plain JS!
- Components define both the *Visual* information and the *State* information



The simplest way to create a component

```
1
2
3
4 var foo = React.createElement('div', {}, 'bar');
5
6 ReactDOM.render(foo, document.getElementById('root'));
7
8
9
```

Creates the virtual component

Tag

Data

Content

Attaches to virtual DOM,
updates real DOM to match



Hmm, this doesn't feel any better than jQuery...

```
1
2
3
4 var foo = React.createElement('div', {}, 'bar');
5
6 ReactDOM.render(foo, document.getElementById('root'));
7
8
9
```

Creates the virtual component

Tag

Data

Content

Attaches to virtual DOM, updates real DOM to match



JSX to the rescue!

```
1  
2  
3  
4 var foo = <div>bar</div>;  
5  
6 ReactDOM.render(foo, document.getElementById('root'));  
7  
8  
9
```



JSX

- JSX merges the world of HTML and Javascript
- *JSX is not HTML*
- It gets converted into basic Javascript via Babel

```
Card.js
1 import React, { Component } from 'react';
2 import './Card.css';
3
4 class Card extends Component {
5
6   render() {
7     return (
8       <div className="card">
9         <h4 className="suit"> Clubs </h4>
10        <h4 className="number"> 5 </h4>
11      </div>
12    );
13  }
14
15 }
16
17 export default Card;
18
```



JSX Cheat Sheet

https://docs.google.com/document/d/1GX9YSg5ZX_Ei_Ng-Tom7bjhVltRF8gk2JcJgyt-CvyAQ/edit

