

# Introduzione a React.js + ES6

DevDay Napoli 16 novembre 2015 - Davide Cerbo

# Davide

Davide Cerbo

10+ anni di esperienza

Full stack (o quasi)



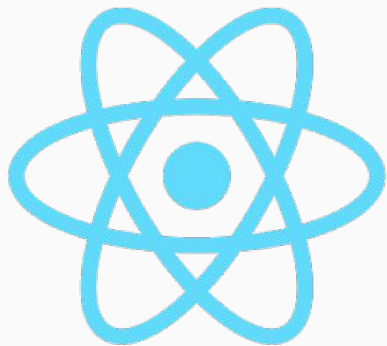
# Cosa è React?

React è una libreria per la UI,  
in MVC è la **V**



+

**facebook**



**React**



**Ryan Florence**

@ryanflorence



Following

My favorite part of React is what I loved about MooTools: to use it effectively you learn JavaScript, not a DSL: useful your whole career.

RETWEETS

66

LIKES

89



1:18 AM - 17 Mar 2015

# Disclaimer

Questa presentazione non contiene tutto lo scibile su REACT e ES6, ma solo quello che ritengo sia utile per iniziare.

Ancora meglio!



*BABEL*

# Introduzione a ES6

- Arrows
- Classes
- Modules



# Arrows

- `var f = (a,b) => a + b; // f(1,2) = 3`
- `["a", "b", "c"].map(v => v.toUpperCase()); // ["A", "B", "C"]`
- Lexical this

# Classes

```
class SkinnedMesh extends THREE.Mesh {
  constructor(geometry, materials) {
    super(geometry, materials);

    this.idMatrix = SkinnedMesh.defaultMatrix();
    this.bones = [];
    this.boneMatrices = [];
    //...
  }
  update(camera) {
    //...
    super.update();
  }
  get boneCount() {
    return this.bones.length;
  }
  set matrixType(matrixType) {
    this.idMatrix = SkinnedMesh[matrixType]();
  }
  static defaultMatrix() {
    return new THREE.Matrix4();
  }
}
```

# Modules

```
// lib/math.js
export function sum(x, y) {
  return x + y;
}
export var pi = 3.141593;
```

```
// app.js
import * as math from "lib/math";
alert("2π = " + math.sum(math.pi, math.pi));
```

```
// otherApp.js
import {sum, pi} from "lib/math";
alert("2π = " + sum(pi, pi));
```

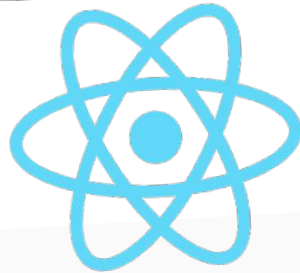
## Ed ancora...

- Promises
- Text template
- Destructuring
- etc...

<https://github.com/lukehoban/es6features>

STEVEN SPINDRICK PRESENTS

**BACK**



React

# Sbam!

```
import React from 'react';

export default class HelloExample extends React.Component {
  constructor(props) {
    super(props);
    this.state = {
      name: '???'
    };
  }

  handleChange(e){
    this.setState({ name:e.target.value });
  }

  render() {
    return (
      <div>
        <input type="text" onChange={e => this.handleChange(e)} />
        <div>Hello {this.state.name}</div>
      </div>
    );
  }
}
```

# Components + JSX

```
export default class HelloExample extends React.Component {  
  constructor(props) {  
    super(props);  
    this.state = {  
      name: '???'  
    };  
  }  
}
```

Componente

Metodo  
obbligatorio

```
  render() {  
    return (  
      <div>  
        <input type="text" onChange={e => this.handleChange(e)} />  
        <div>Hello {this.state.name}!</div>  
      </div>  
    );  
  }  
}
```

JSX

# State & props

Lo **stato** rappresenta la situazione attuale dell'oggetto ed è **mutabile**.

```
handleChange(e){
  this.setState({ name:e.target.value });
}

render() {
  return (
    <div>
      <input type="text" onChange={e => this.handleChange(e)} />
      <div>Hello {this.state.name}!</div>
    </div>
  );
}
```

Le **proprietà** sono opzioni del componente che vengono passate dal padre al figlio e sono **immutabili**

```
render() {
  return (
    <div>
      <input type="text" onChange={e => this.handleChange(e)} />
      <Hello name={this.state.name} />
    </div>
  );
}
```

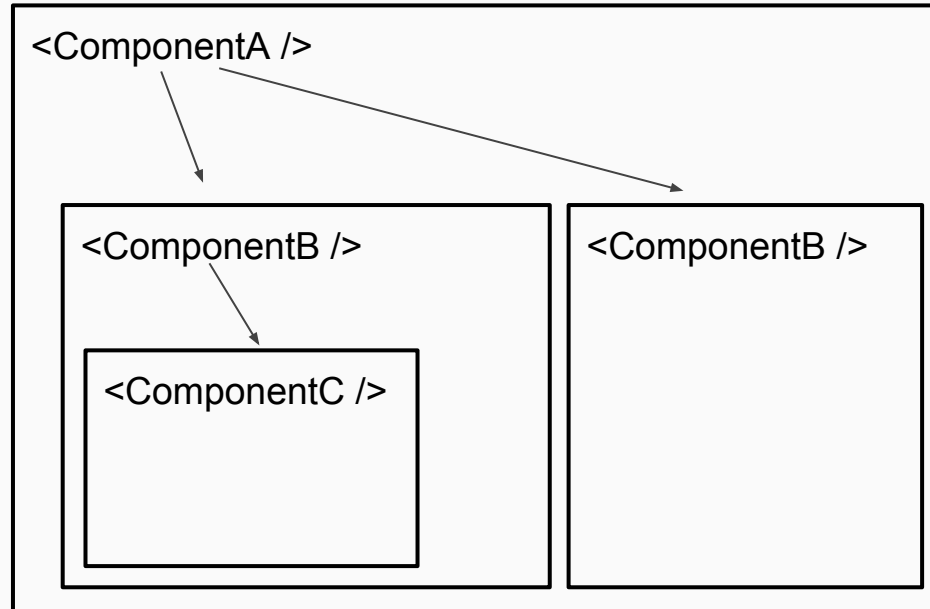
```
render() {
  return (
    <div>Hello <span style={{color:"red"}}>{this.props.name}</span>!</div>
  );
}
```



# Virtual DOM

- React utilizza una rappresentazione virtuale del DOM
- Usa un algoritmo di differenza tra alberi ed una euristica  $O(n^3) \Rightarrow O(n)$
- Questo garantisce elevate prestazioni

# One-way data binding



# Hello Component!

```
import React from 'react';

export default class Hello extends React.Component {
  constructor(props) {
    super(props);
  }

  componentWillMount(){
    console.log("Will mount...");
  }

  componentDidMount(){
    console.log("Did mount!");
  }

  render() {
    return (
      <div>Hello <span style={{color:"red"}}>{this.props.name}</span>!</div>
    );
  }
}
```

# Demo



# Demo

`https://github.com/gesposito/bare-webpack`

`unzip bare-webpack-master.zip`

`cd bare-webpack-master`

`npm install`

`npm run dev`

`http://localhost:3000`

# Hello Component! (+ sorpresa)

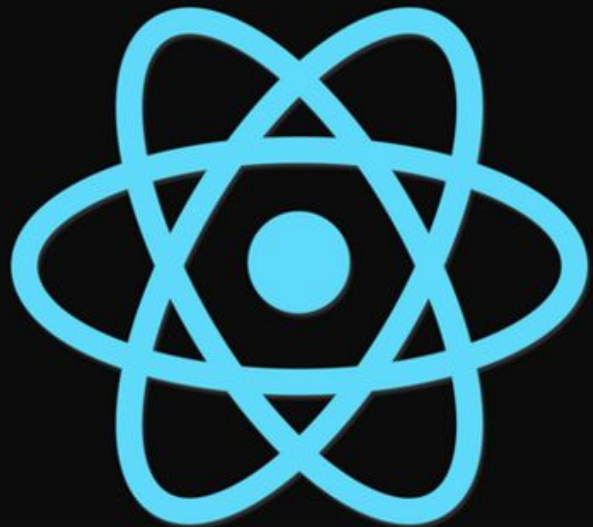
```
import React from 'react';
import HelloSurprise from './HelloSurprise'

export default class HelloComponentExample extends React.Component {
  constructor(props) {
    super(props);
    this.state = {
      name: 'Valentina'
    };
  }

  handleChange(e){
    this.setState({name:e.target.value});
  }

  render() {
    return (
      <div>
        <input type="text" onChange={e => this.handleChange(e)} />
        <HelloSurprise name={this.state.name} />
      </div>
    );
  }
}
```

Non è tutto



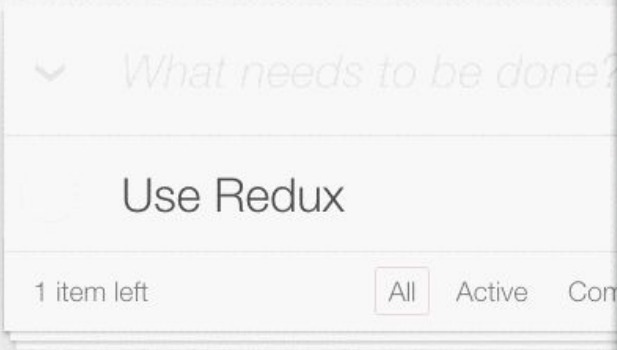
**React**



Flux Architecture

# Time travel

<https://github.com/gaearon/redux-devtools>



Reset

Revert

Sweep

Commit

@@INIT

```
▼ state: {} 1 key
  ► todos: [] 1 item
```



# Link e risorse

- <https://egghead.io/series/react-flux-architecture>
- <https://facebook.github.io/react/>
- [https://www.youtube.com/watch?v=bEI06xF\\_xxE](https://www.youtube.com/watch?v=bEI06xF_xxE) (Video in italiano)
- <https://www.youtube.com/watch?v=-A90OnLeWfA> (Video in italiano)
- <https://www.youtube.com/watch?v=ET0KWjoRMao> (Video in italiano)
- <https://code-cartoons.com/a-cartoon-intro-to-redux-3afb775501a6>
- <https://code-cartoons.com/a-cartoon-guide-to-flux-6157355ab207>

# Grazie a tutti

@davide\_cerbo

davidecerbo@gmail.com