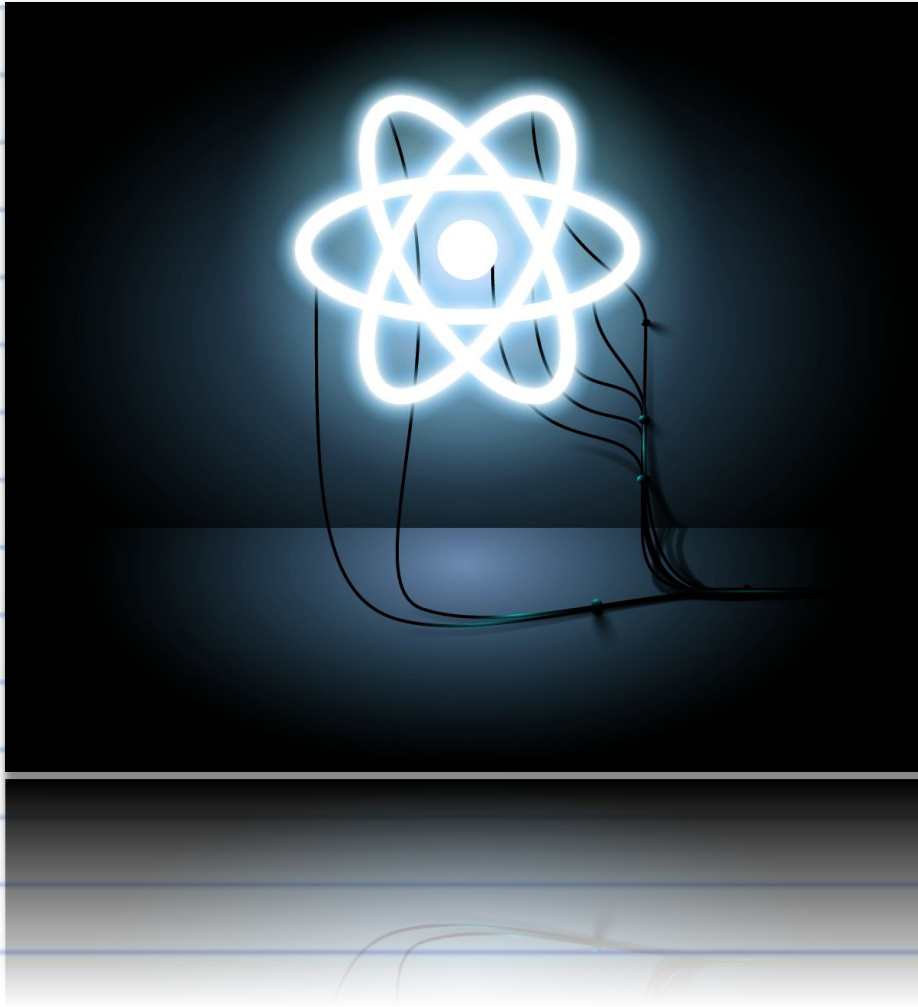


React & Redux

Einführung

Martina Freundorfer
Hochschule für Technik &
Wirtschaft (HTW) Berlin

Was ist React?



- JavaScript Software Bibliothek
- Ursprünglich von Facebook kreiert
- Netflix und Instagram benutzen es heute auch
- Wird benutzt um JavaScript getriebene dynamische Web Applikationen zu erstellen
- Daher kann es auch mit ähnlichen Bibliotheken wie AngularJS & VueJS verglichen werden

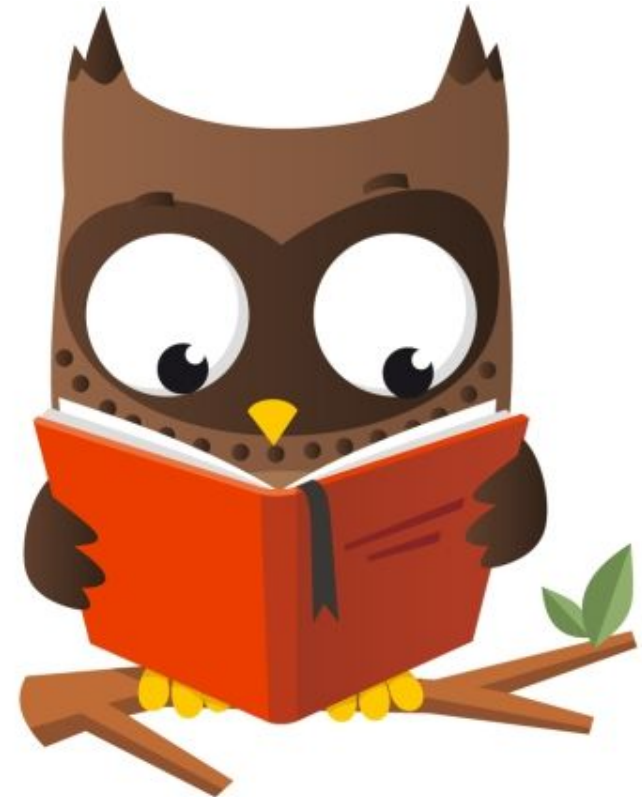
Was ist Redux?

- Ein Layer "on top" von React
- Hilft mit dem State Management unserer App
 - Datenverwaltung innerhalb der App
 - UI-State der App
- Redux lernen wir im 2. Teil der React-Vorlesung

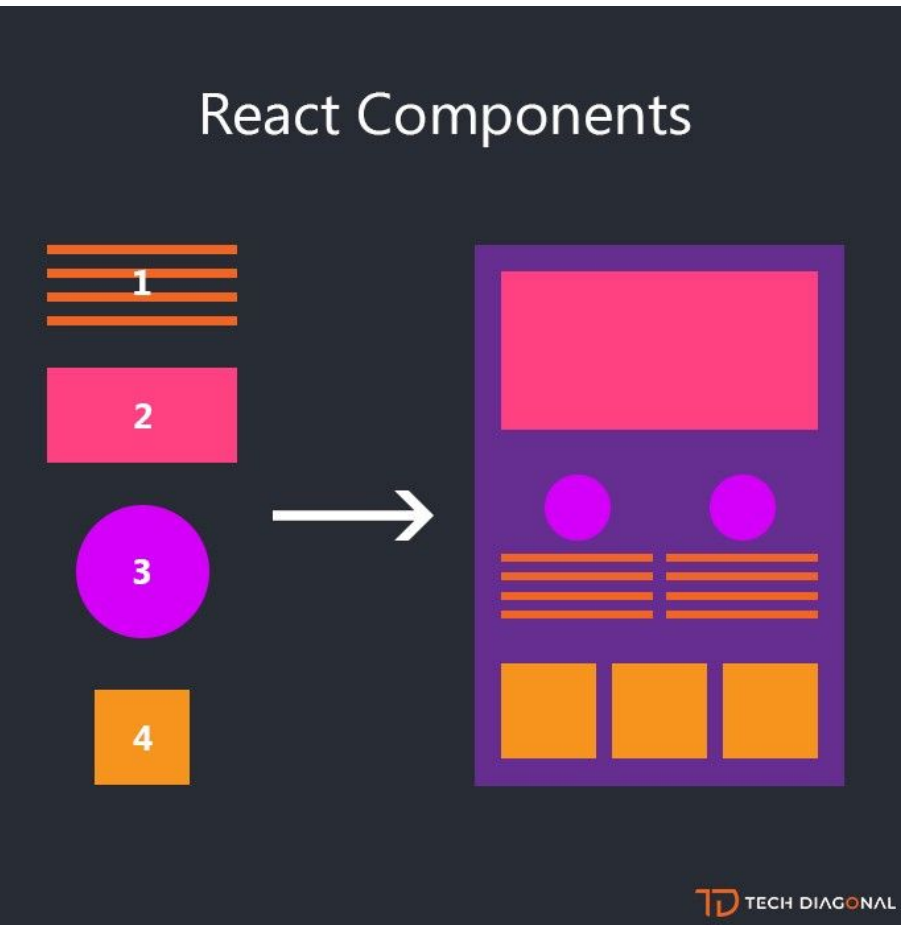


Was wir lernen werden (hoffentlich)

- **React Basics**
 - Komponenten, Events, Templates, Props & Forms
- **React Router**
 - Routen, Route Parameters, Redirects
- **Redux**
 - Stores, Actions, Reducers



React - Komponenten, Komponenten, KOMPONENTEN!!!



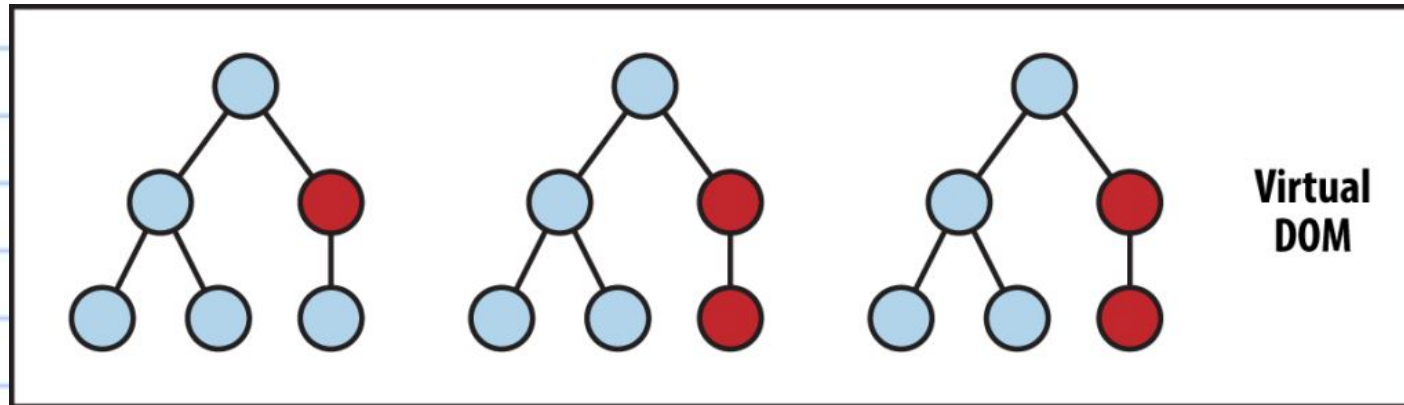
React besteht ausschließlich aus Komponenten.

Wir kreieren Komponenten in React für verschiedene Teile unserer Website, z.B. Navbar, Suchbox, Footer

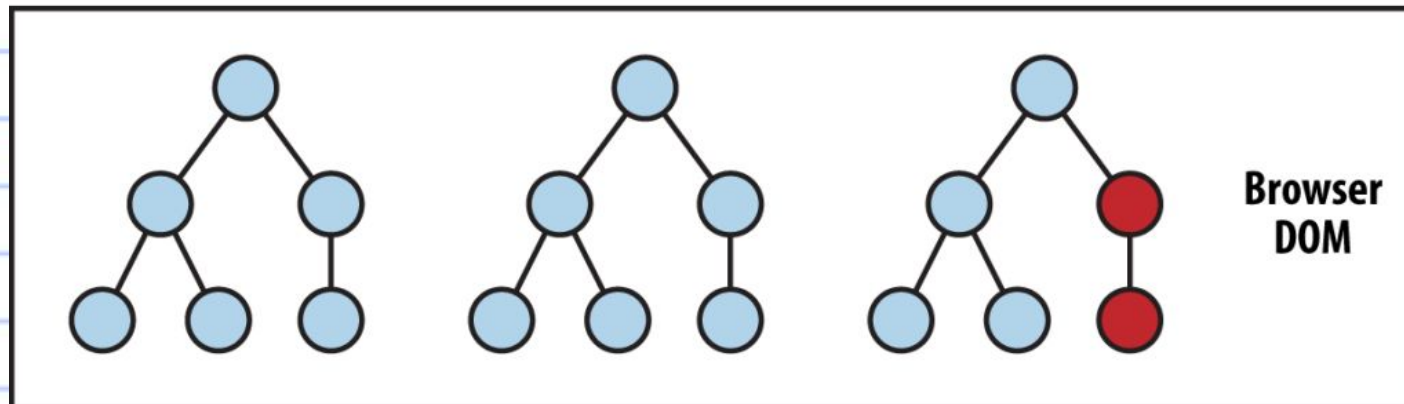
Und dann übernimmt React die Arbeit, diese Komponenten per JavaScript in den DOM einzupflegen, so dass wir diese auf der Website sehen.

Der Virtuelle DOM

Der Virtuelle DOM macht React schnell...



State Change → Compute Diff → Re-render



Komponenten & Templates

```
App.js dictionary.js x
1 import React, { Component } from 'react';
2 import './App.css';
3
4 class App extends Component {
5
6   state = {
7     value: "Some text here",
8     isInEditMode: false
9   }
10
11   changeEditMode = () => {
12
13     console.log("should go to edit mode now")
14   }
15
16   render() {
17     return this.state.isInEditMode ?
18     <div>
19       <input
20         type="text"
21         defaultValue={this.state.value}
22       />
23     </div> :
24     <div onClick={this.changeEditMode}>
25       {this.state.value}
26     </div>
27   }
28 }
29
30 export default App;
31
```

- Komponenten sehen aus wie HTML Templates (eigentlich JSX = JavaScript XML)
- Sie können 'state' beinhalten (Data oder UI State)
- Sie können auch JavaScript Funktionalität beinhalten
- Mehr Infos unter <https://reactjs.org/docs/add-react-to-a-website.html>

Setup React

Wir haben zwei Möglichkeiten, React für unser Projekt zu benutzen

1. "create-react-app"

Ein Webpack Setup Paket, das hilft ein komplettes React-Projekt zu kreieren und den Flow zu verwalten

2. Erste zwei CDN*-Links von folgender Website in unseren HTML HEAD (unter dem CSS) einbinden, wenn wir nur Teile von React ausprobieren wollen (schneller & besser zum Lernen):

<https://reactjs.org/docs/cdn-links.html>

*Content-Delivery-Network

Visual Studio Code Example

```
<head>
```

```
  <script crossorigin  
src="https://unpkg.com/react@16/umd/react.develo  
pment.js"></script>
```

```
  <script crossorigin  
src="https://unpkg.com/react-dom@16/umd/react-do  
m.development.js"></script>
```

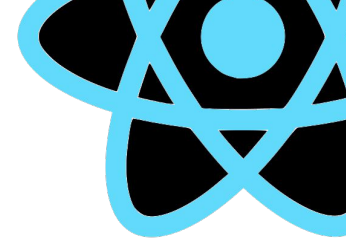
```
</head>
```

```
<body>
```

```
  <div id="app"></div>
```

```
</body>
```

Visual Studio Code Extensions



1. ES7 React/Redux/GraphQL/React-Native snippets

<https://marketplace.visualstudio.com/items?itemName=dsznajder.es7-react-js-snippets>

2. Babel JavaScript

<https://marketplace.visualstudio.com/items?itemName=mgmcdermott.vscode-language-babel>

3. Monokai++ Theme (optional)

<https://marketplace.visualstudio.com/items?itemName=dcasella.monokai-plusplus>

4. Material Icon Theme (optional)

<https://marketplace.visualstudio.com/items?itemName=PKief.material-icon-theme>



Erste React Komponente (noch nicht funktional)



```
<body>
  <div id="app"></div>
  <script>
    class App extends React.Component {
      render()
        return (
          <div className="app-content">
            <h1>Hey, HTW Studenten!</h1>
          </div>
        )
      }
    }
    ReactDOM.render(<App>, document.getElementById('app') );
  </script>
</body>
```

Browser Demo



Setup Babel



JSX wird nicht direkt vom Browser unterstützt. Deswegen brauchen wir einen "Übersetzer" namens Babel:

Babel ist ein sogenannter "Transpiler" von JSX in eine HTML Sprache, die der Browser versteht. Daher müssen wir noch ein Script hinzufügen:

<https://babeljs.io/setup#installation>

Klick auf "In the Browser"

Kopieren Sie die Script Zeile in ihren HEAD:

```
<script  
src="https://unpkg.com/@babel/standalone/babel.min.js"<  
/script>
```

und ändern sie den bestehenden Script-Tag im Body zu

```
<script type="text/babel">
```

Erste React Komponente (jetzt funktional wg Babel)



```
<body>
  <div id="app"></div>
  <script type="text/babel">
    class App extends React.Component {
      render()
        return (
          <div className="app-content">
            <h1>Hey, HTW Studenten!</h1>
          </div>
        )
      }
    }
    ReactDOM.render(<App>, document.getElementById('app'));
  </script>
</body>
```

Browser Demo



Erste React Komponente Inklusive JavaScript



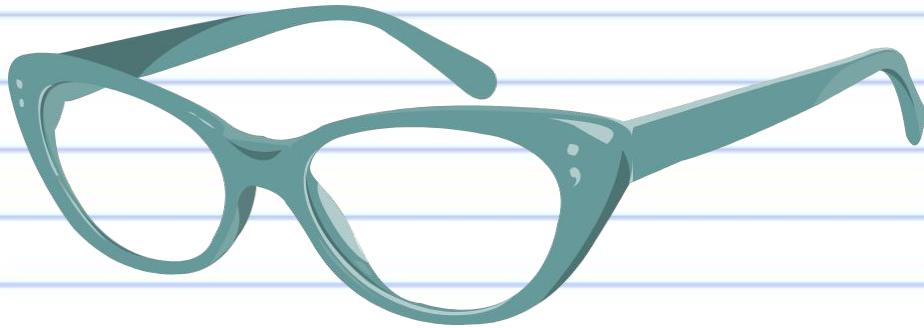
```
<body>
  <div id="app"></div>
  <script type="text/babel">
    class App extends React.Component {
      render()
        return (
          <div className="app-content">
            <h1>Hey, HTW Studenten!</h1>
            <p>{ Math.random() * 10 }</p>
          </div>
        )
      }
    }

    ReactDOM.render(<App>, document.getElementById('app') );
  </script>
```


Browser Demo



Komponenten State



- JavaScript Object
- Beschreibt den aktuellen Zustand der Komponente
 - - Data oder UI-state
- Der Status einer Komponente kann sich mit der Zeit ändern
 - Ein Modal kann geschlossen werden
 - Die Daten, die wir anzeigen könnten sich im Browser ändern

Einkaufswagen Komponente (Data)

```
{  
  items: [  
    {name: 'navy jumper', price: 9.99},  
    {name: 'ninja mask', price: 20.00}  
  ]  
}
```



Einkaufswagen Komponente (Data)

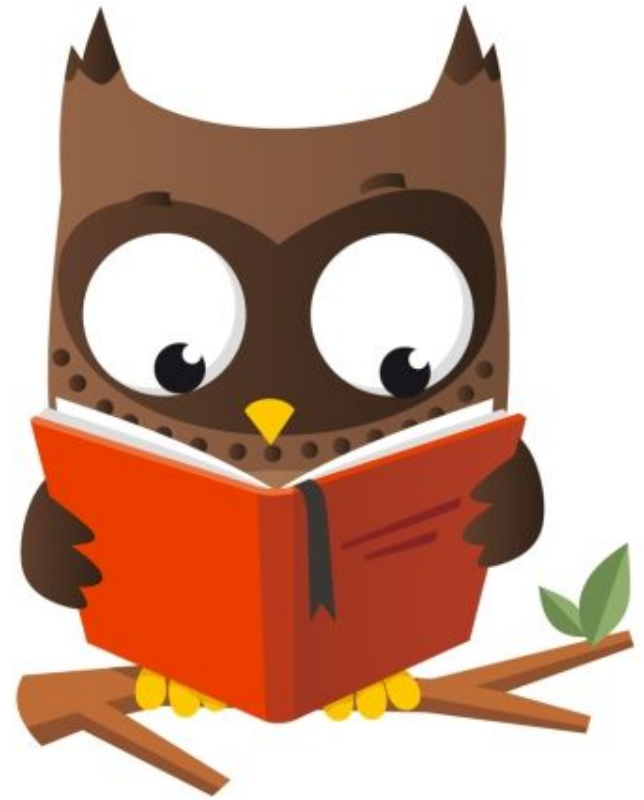
```
{  
  items: [  
    {name: 'navy jumper', price: 9.99},  
    {name: 'ninja mask', price: 20.00},  
    {name: 'black cloak', price: 15.00}  
  ]  
}
```



Popup Komponente (UI-State)

```
{  
  showPopup: true  
}
```

```
{  
  showPopup: false  
}
```



Erste React Komponente Inklusive State



```
<body>
  <div id="app"></div>
  <script type="text/babel">
    class App extends React.Component {
      state = {
        name: 'Martina Freundorfer',
        age: 35
      }
      render()
        return (
          <div className="app-content">
            <h1>Hey, HTW Studenten!</h1>
            <p>{ Math.random() * 10 }</p>
          </div>
        )
    }
  </script>
</body>
```

Browser Demo



Erste React Komponente

Inkl. Benutzung des States

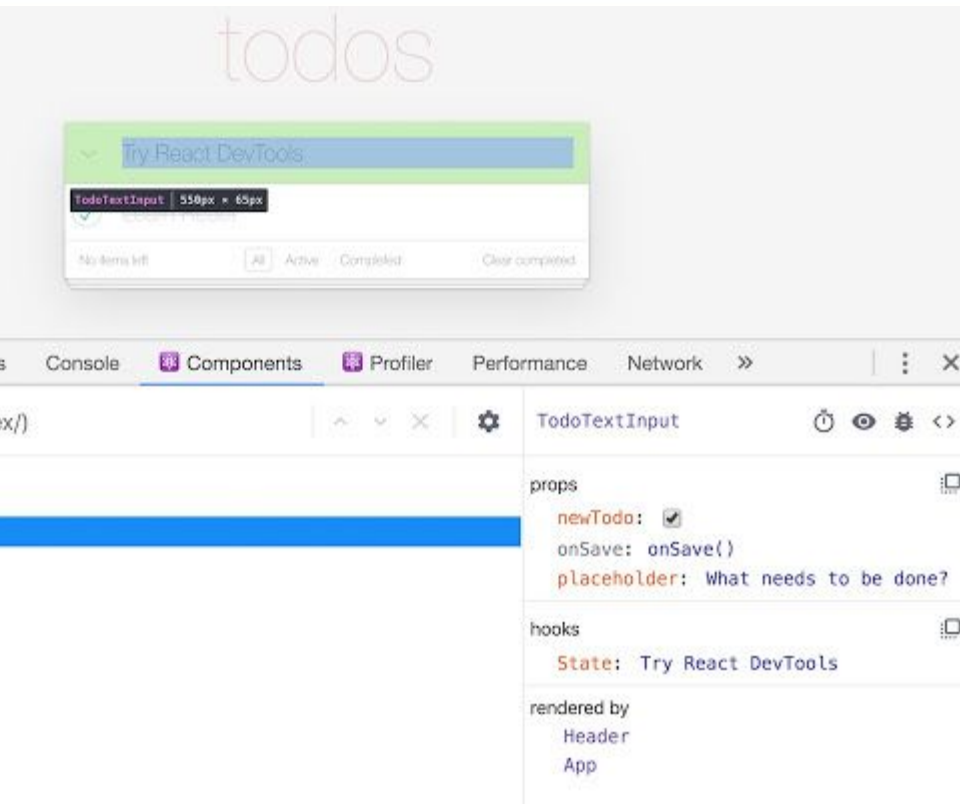


```
<body>
  <div id="app"></div>
  <script type="text/babel">
    class App extends React.Component {
      state = {
        name: 'Martina Freundorfer',
        age: 35
      }
      render()
        return (
          <div className="app-content">
            <h1>Hey, HTW Studenten!</h1>
            <p>Mein Name ist: { this.state.name } und
            ich bin {this.state.age} Jahre alt</p>
          </div>
```


Browser Demo



Extra: React Developer Tools



Praktische Extension für
Google Chrome:

<https://chrome.google.com/webstore/detail/react-developer-tools/fmkadmapgofadopljbjfkapdkoienihi>

Ausprobieren unter

<https://reactjs.org/> oder
<https://codepen.io/gaearon/pen/gWWZgR>

Oder mit der eigenen
Website in den Developer
Tools zwei neue Tabs:
Components & Profiler

Browser Demo



Credits & Lesestoff



Die Kursdateien für das React & Redux Tutorial

<https://github.com/htw-web/react-redux-complete-playlist>

Shaun "The Net Ninja" Pelling (Github: iamshaunjp)

<https://www.youtube.com/channel/UCW5YeuERMmlnqo4oq8vwUpq>

<https://github.com/iamshaunjp>

Tutorial: Intro to React (Tic Tac Toe)

<https://reactjs.org/tutorial/tutorial.html>

What is Babel?

<https://babeljs.io/docs/en/>

Vielen Dank für Ihre Aufmerksamkeit!