

ReactiveX - Web Apps

Roman Roelofsen - w11k GmbH / theCodeCampus

Twitter & GitHub: **romanroe**

Über mich

- Alpha Geek, Entwickler, Trainer
- w11k GmbH - *The Web Engineers*
 - Individualsoftware
- theCodeCampus
 - Schulungsanbieter Angular & TypeScript
 - 30% Rabatt, 26.07. Angular2 Schulung in Siegburg

Reaktive Programmierung

Wir denken in **synchronen** Schritten:

- Wenn der User auf den Button clickt,
- sende den Inhalt eines Textfeldes an den Server,
- nehme das Ergebnis der Anfrage entgegen,
- und stelle es dar.

Aber programmieren wir auch so?

```
button.onclick = function () {  
    var text = $("#text").text();  
    $.post("/server", text, function (result) {  
        $("#ergebnis").text(result);  
    });  
}
```

- Web Apps sind Teil einer asynchronen Umgebung
- Server-Logik wird zunehmend asynchron
- Asynchrone Programmierung wird sehr schwer, wenn die Anzahl der Datenströme wächst

Frameworks, Bibliotheken helfen

- Threads
- Promise
- ReactiveX

ReactiveX

- RxJS, Bibliothek ursprünglich entwickelt von Microsoft
- **<http://reactivex.io>**
- Implementierungen für
 - Java, JavaScript/TypeScript, .NET, Scala, Clojure, Swift, etc.

- Observable
 - Liefert Daten

	Single return value	Multiple return values
Pull/Synchronous/Interactive	Object	Iterables(Array Set Map)
Push/Asynchronous/Reactive	Promise	Observable

- Observer
 - Bekommt Daten
- Subject
 - Observable *und* Observer

Demo

Fazit

- Lernkurve:
 - kurz flach
 - dann lange steil
 - dann wieder flach
- ReactiveX macht komplexe Datenflüsse "einfach"
- Keine komplexe Datenflüsse? Dann Finger weg...
- Kein echtes Projekt zum Lernen nehmen, niemals!
- **<http://rxmarbles.com>**

Roman Roelofsen

@romanroe