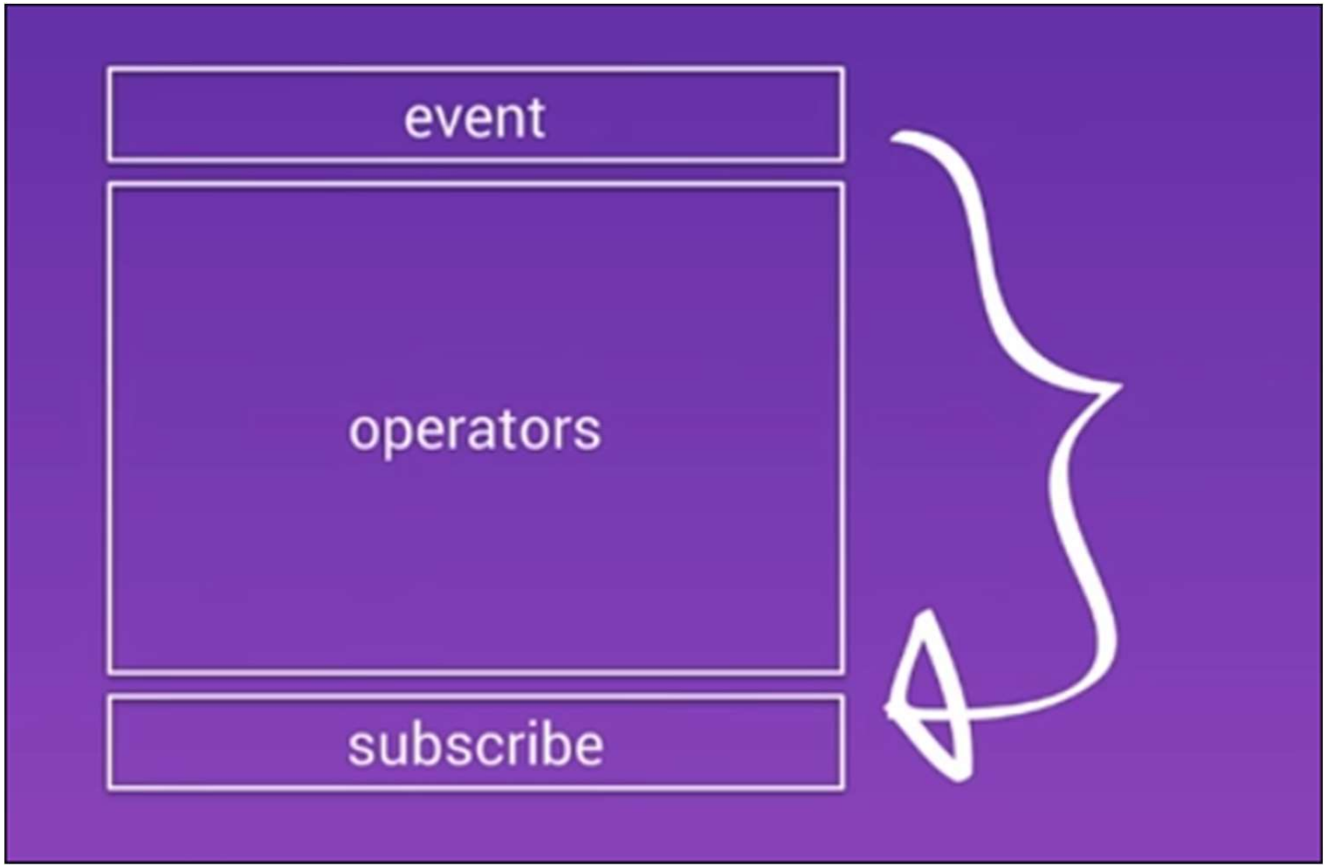
Reactive Angular met RxJS
Short recap

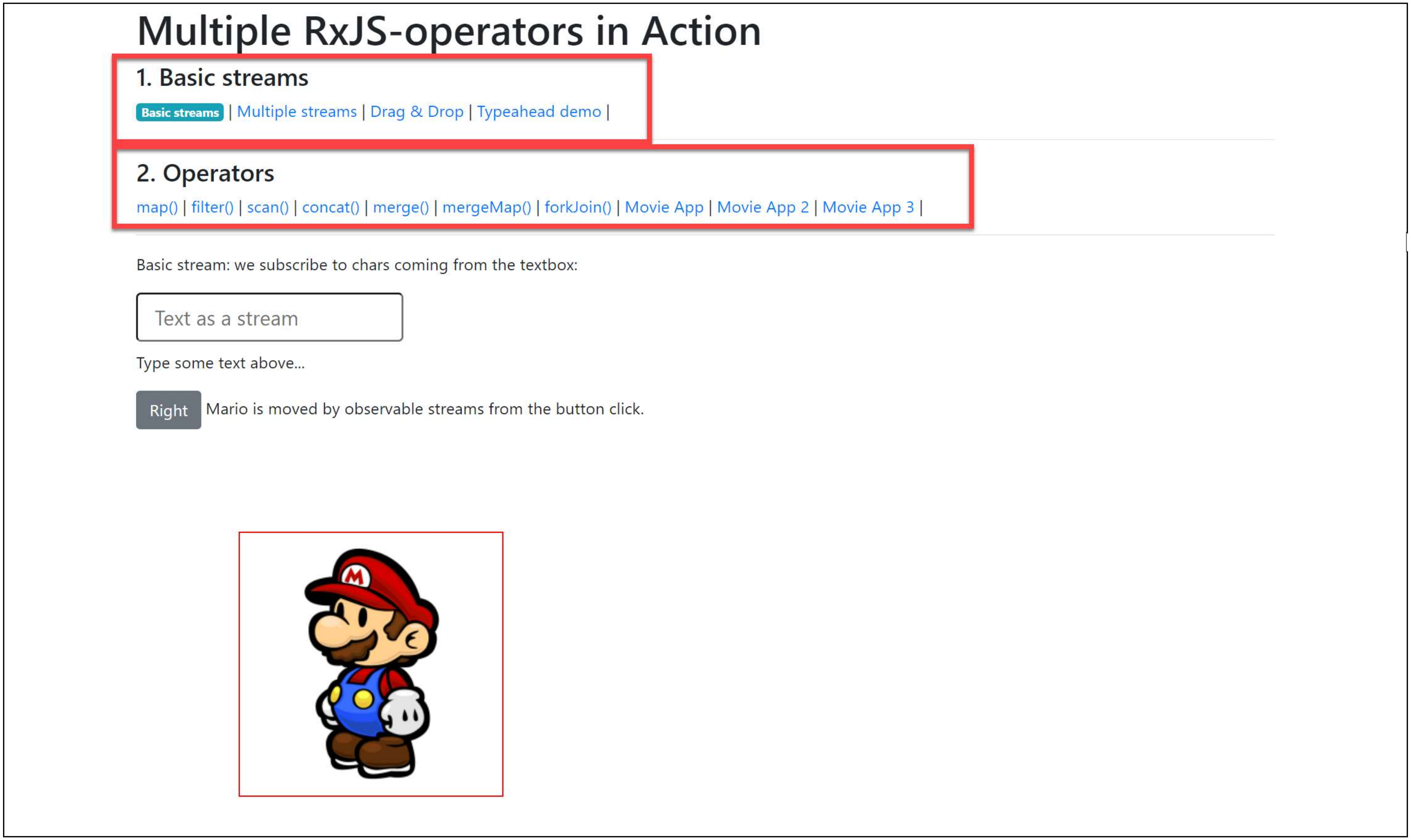
Peter Kassenaar –
info@kassenaar.com

Day 1

- Introduction to the Observable design pattern
- Creating observables from scratch
- Using observables for Http communication, Live API's
- `async` pipe
- Pipeline & operators
  - `map,`
  - `delay`
  - `switchMap`
  - `tap`
  - `debounceTime`
  - `merge, concat, …`

Subscribe to events – or let `async` handle that





Multiple RxJS-operators in Action

1. Basic streams

Basic streams | Multiple streams | Drag & Drop | Typeahead demo |

2. Operators

map() | filter() | scan() | concat() | merge() | mergeMap() | forkJoin() | Movie App | Movie App 2 | Movie App 3 |

Basic stream: we subscribe to chars coming from the textbox:

Type some text above...

Mario is moved by observable streams from the button click.

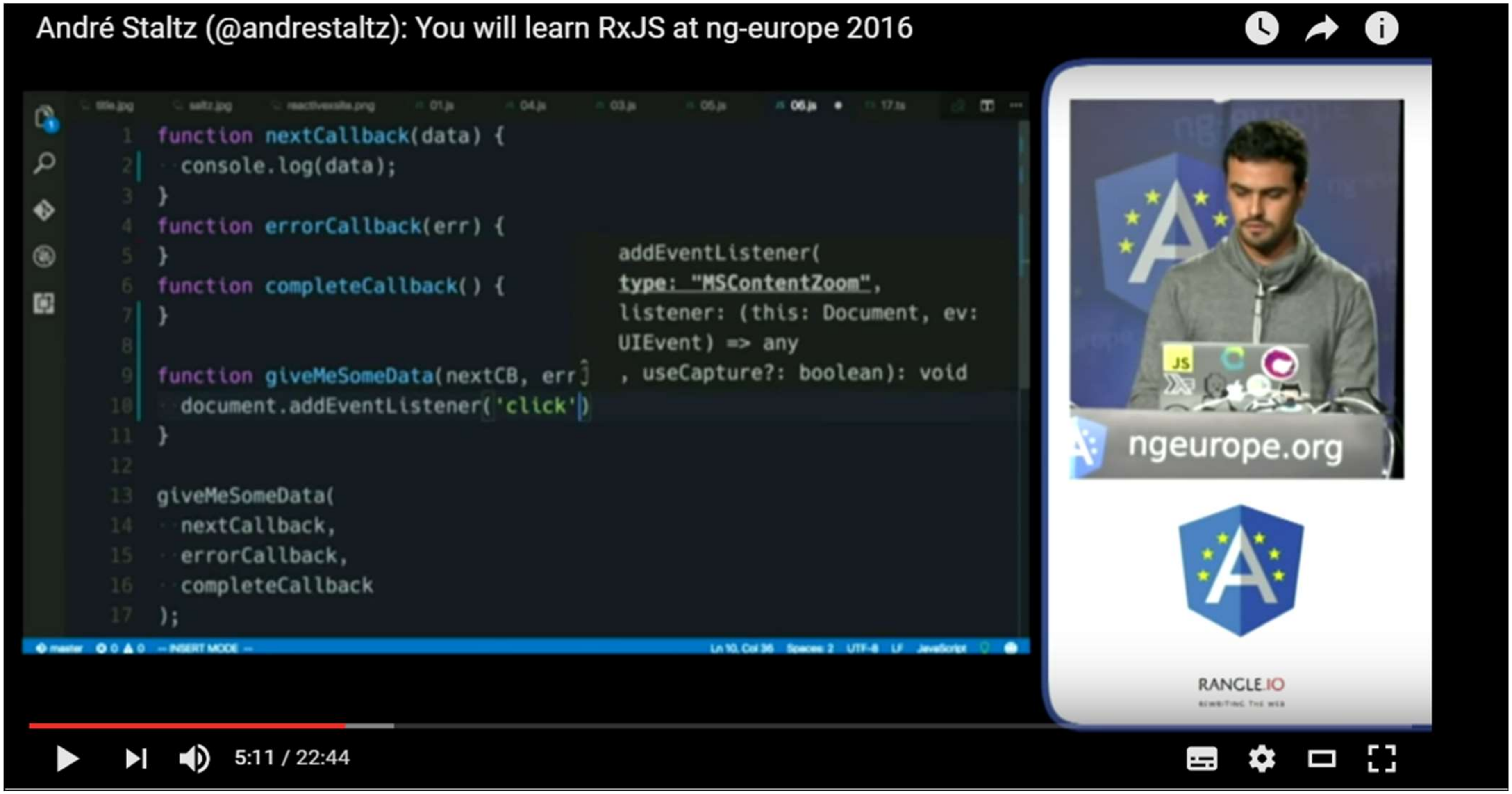
https://github.com/PeterKassenaar/ng-rxjs-operators

Today

- [for starters: Demystifying observables – Andre Staltz
  - Creating your own/custom operators]
- *State Management* Design pattern
- Store concepts & terminology
- `@ngrx/store` – Working with:
  - Actions & dispatchers
  - Reducers
  - `HttpClient` and Stores
  - Store DevTools
  - Store Effects
  - Store Feature Modules

Some extra stuff

Creating Observables from scratch - André Staltz

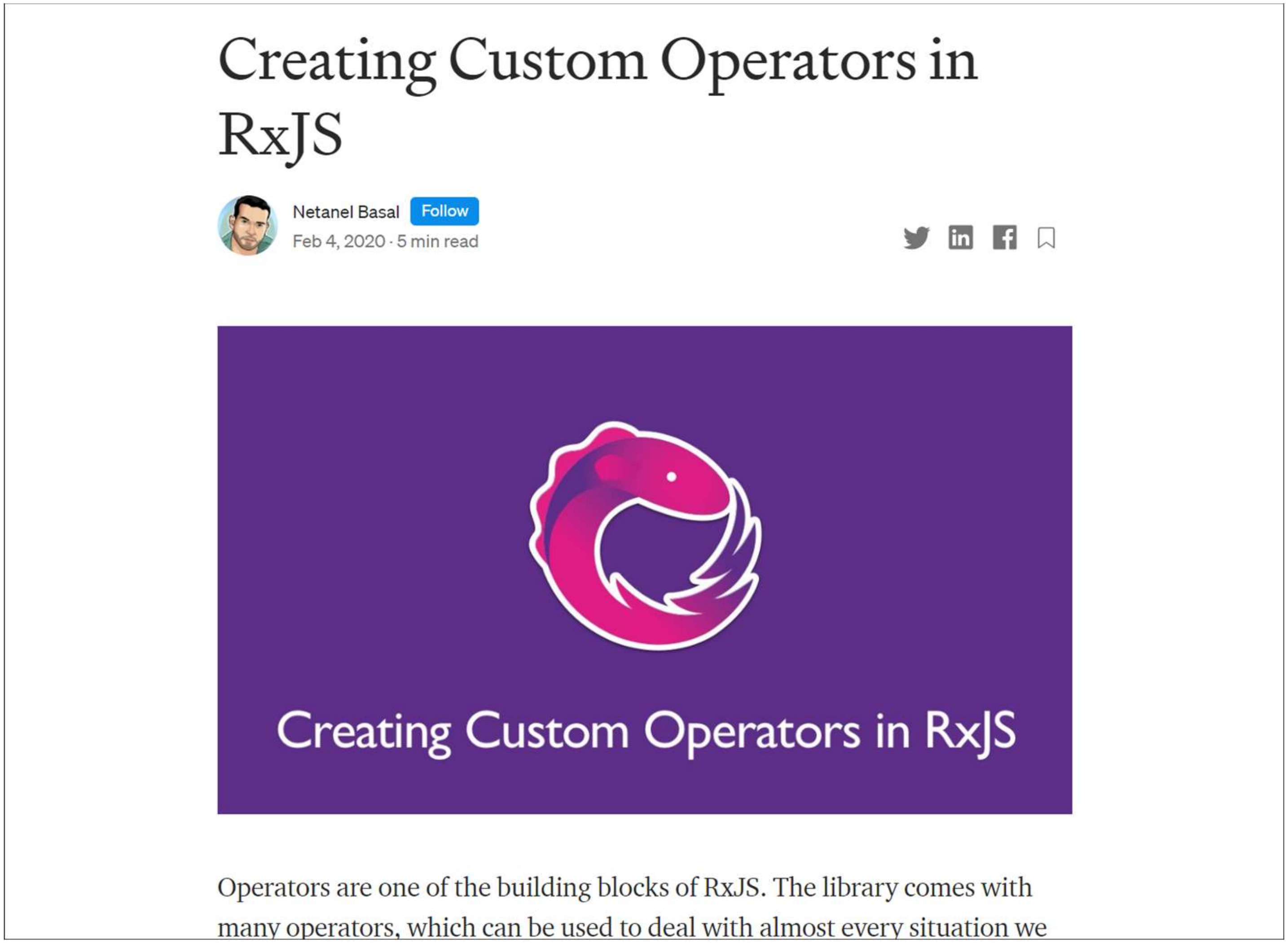


```
function nextCallback(data) {
  console.log(data);
}
function errorCallback(err) {
}
function completeCallback() {
}

function giveMeSomeData(nextCB, err
  document.addEventListener('click')
}

giveMeSomeData(
  nextCallback,
  errorCallback,
  completeCallback
);
```

```
addEventListener(
type: "MSContentZoom",
listener: (this: Document, ev:
UIEvent) => any
, useCapture?: boolean): void
```

https://www.youtube.com/watch?v=uQ1zhJHclvs

Creating your own operators



Creating Custom Operators in RxJS

Netanel Basal Follow

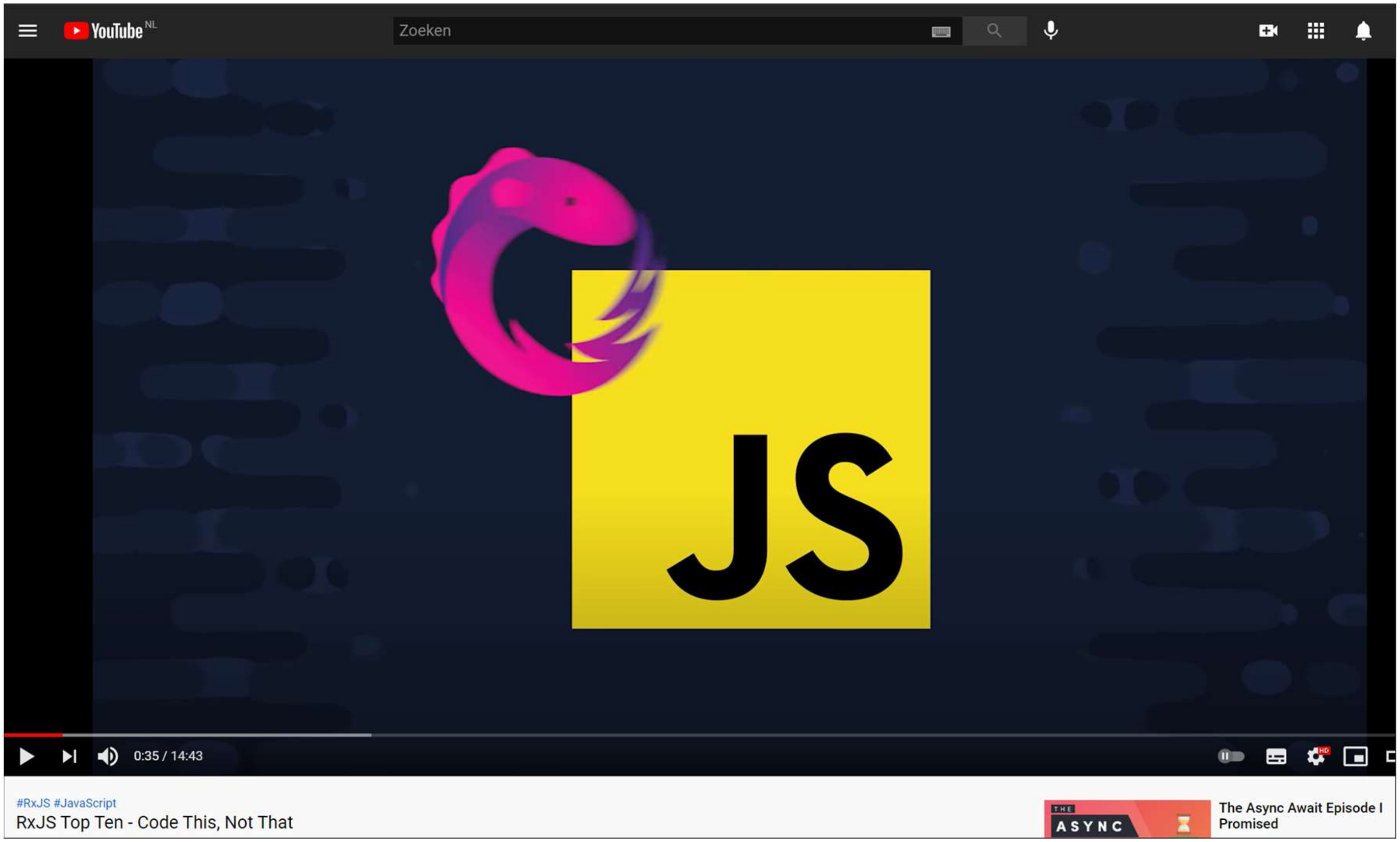
Feb 4, 2020 · 5 min read

Operators are one of the building blocks of RxJS. The library comes with many operators, which can be used to deal with almost every situation we

https://netbasal.com/creating-custom-operators-in-rxjs-32f052d69457

RxJS Top 10 concepts

- *A lot is now familiar* – but just as a reminder/refresher and from a different point of view:



RxJS Top Ten - Code This, Not That

https://www.youtube.com/watch?v=ewcoEYS85Co

Today

- [for starters: Demystifying observables – Andre Staltz
  - Creating your own/custom operators]
- *State Management* Design pattern
- Store concepts & terminology
- `@ngrx/store` – Working with:
  - Actions & dispatchers
  - Reducers
  - `HttpClient` and Stores
  - Store DevTools
  - Store Effects
  - Store Feature Modules

Tomorrow

- Angular – Routing w/ multiple modules
    - Lazy loading
- Building a Custom store
- Change Detection
- Store - Case
- …
- If possible – finish a little bit early