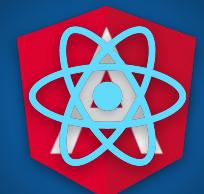


# React tools for better Angular apps



Martin Hochel  
@martin\_hotell



Hello Vikings !

# Martin Hochel

Principal SE, GDE   
Prague / Czech Republic

 @martin\_hotell

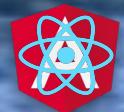
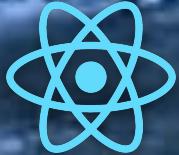
 [github.com/Hotell](https://github.com/Hotell)

- [@ngPartyCz](#) meetup founder
- Author of [ngMetadata](#)
- Member of [@skate\\_js](#), [@ngParty](#)



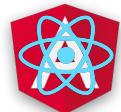


Fight !





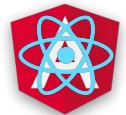
Fight  
leads  
to the ~~dark~~ side  
*Twitter clone flame wars*



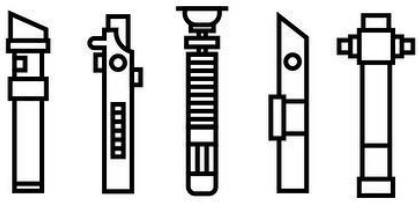
# We can be all friends !



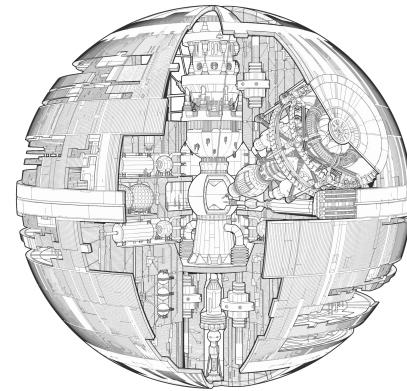
How ?



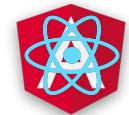
# Tools



# Architecture



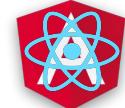
# Libraries



# Baseline



```
$ npx @angular/cli new viking-app
```

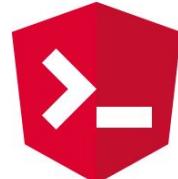


CLI

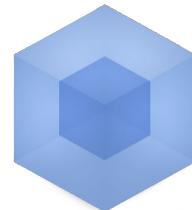
Libraries



Architecture



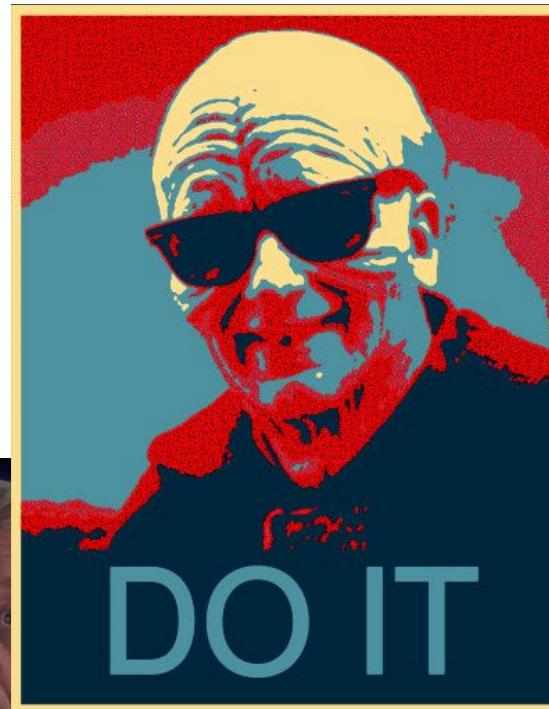
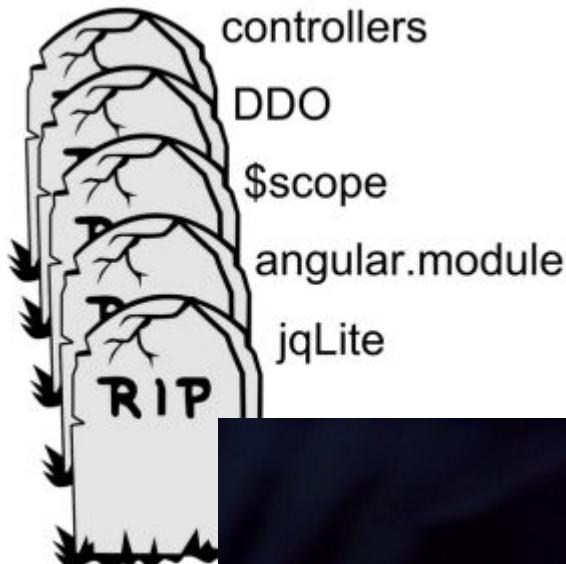
Tools



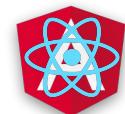
TS



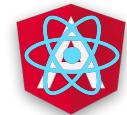
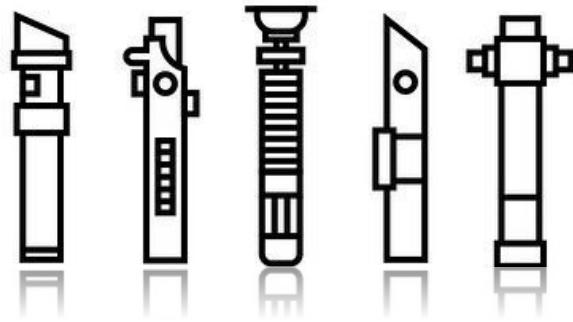
# Disclaimer: Tombstones ahead



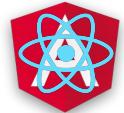
**sorry about that**



# Tools



# Task runner/package manager

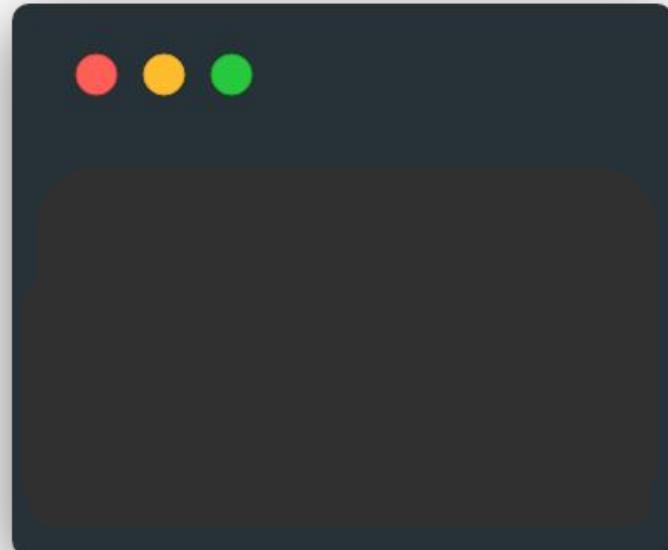
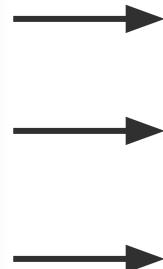
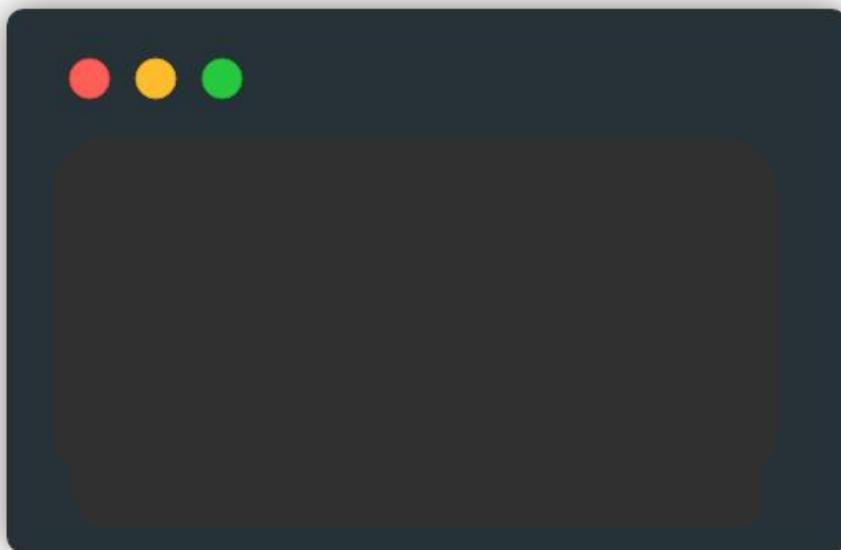


# Tools: task runner



yarn

# Tools: task runner

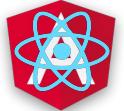


# Tools: package manager

```
$ yarn upgrade-interactive --latest
```

I

# Formatting



# Tools: Formatting

```
You, a few s [tslint] Add space before 'from' (import-spacing)
import {Component} from '@angular/core'
```



# Tools: Formatting



```
$ yarn add -D prettier
```

```
$ yarn prettier {src/e2e}/**/* --write
```

# Tools: Formatting conflicts with linter

```
$ yarn tslint-config-prettier-check ./tslint.json
Error: Conflict rule(s) detected in ./tslint.json:
eofline
import-spacing
indent
max-line-length
no-trailing-whitespace
one-line
quotemark
semicolon
typedef-whitespace
whitespace
```

```
$ yarn add -D tslint-config-prettier
```

```
{
  "rulesDirectory": [
    "node_modules/codelyzer",
    "tslint-config-prettier"
  ]
}
```

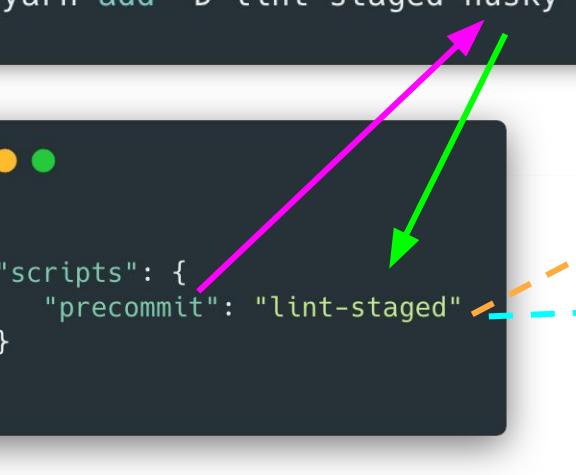
# Tools: Formatting - let robots do the work



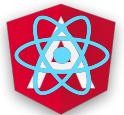
```
$ yarn add -D lint-staged husky
```

```
{
  "scripts": {
    "precommit": "lint-staged"
  }
}
```

```
{
  "lint-staged": {
    "{src,e2e}/**/*.{js,jsx,ts,tsx)": [
      "yarn lint",
      "git add"
    ],
    "{src,e2e}/**/*.{js,jsx,ts,tsx,css,scss,md)": [
      "prettier --write",
      "git add"
    ]
  }
}
```



# Unit Testing



# Unit Testing



# unit testing



## Delightful JavaScript Testing



Easy Setup



Instant Feedback



Snapshot Testing

# Unit testing: add Jest to @angular/cli



```
yarn add -D jest jest-preset-angular
```

# Unit testing: speed

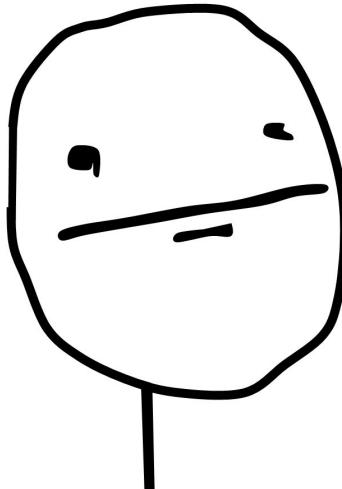
karma-chrome:

**14.911s**



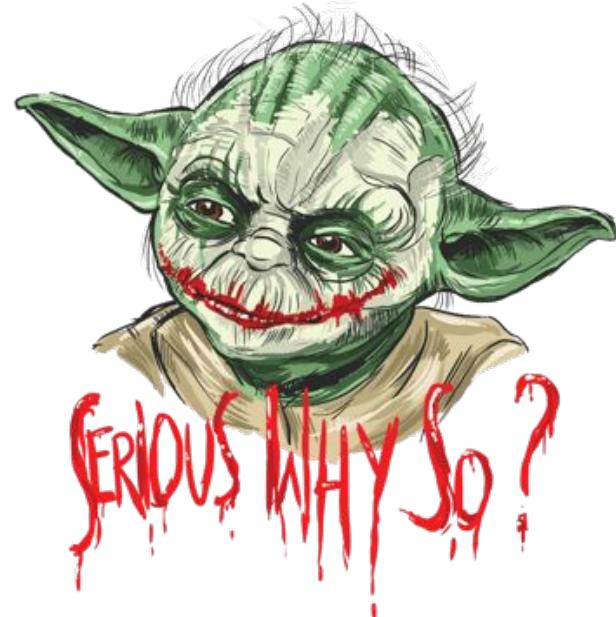
karma-phantomjs:

**13.119s**



jest:

**4.970s**



# Unit testing: Jest - Snapshots

FAIL src/app/products/components/pizza-display/pizza-display.component.spec.ts

PizzaDisplayComponent

✓ should create (12ms)

✗ should display pizza

✓ should display pizza



● PizzaDisplayComponent

expect(value).toMatchSnapshot()

Received value:

- Snapshot  
+ Received

```
it('should display pizza', () => {
  component.pizza = pizzaMock;
  fixture.detectChanges();
```

```
  expect(fixture).toMatchSnapshot();
});
```

@@ -1,6 +1,8 @@

- <pizza-display>

+ <pizza-display>

+ pizza={[Function Object]}

# Unit testing: Jest - Interactive watch mode

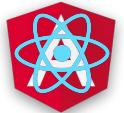
```
$ yarn
```



## Unit testing: Jest - much more awesomeness

- Powerful mocking features ( *ES2015 modules, assets* )
- Code coverage - **OCJS** `jest --coverage`
- Pluggable ( run Puppeteer with Jest )
- Huge ecosystem
  - [jest-axe](#) ( a11y )
  - [jest-images-snapshots](#)

# E2E Testing



# E2E Testing



# E2E Testing



A [node.js](#) tool to automate  
end-to-end web testing

# E2E Testing: add TestCafe to @angular/cli

```
yarn add -D testcafe testcafe-live
```

```
yarn testcafe chrome e2e/**/*.e2e-spec.ts
```

A Redux Tools For Better Angular x

① 10.100.44.181:60344/SykFIYH\_M/http://localhost:4200/products

# ULTIMATE PIZZA

PRODUCTS

NEW PIZZA



BLAZIN' INFERNO

PLAIN OL' PEPPERONI

VIEW PIZZA

VIEW PIZZA

© Ultimate Pizza Inc.

create-pizza.e2e-spec.ts — react-tools-for-better-angular-apps

TS create-pizza.e2e-spec.ts e2e

```
32      jj;  
33  
34      await t.click(pizzaItem.form.actionButtons.delete);  
35  };  
36  
37  test('should add Marvel chili Pizza to the menu!', async (t) => {  
38    await pizzaItem.navigateToNewProduct();  
39  
40    // await t.setTestSpeed(0.5);  
41  
42    await t.typeText(pizzaItem.form.pizzaNameInput, newPizzaName);  
43  
44    await t.click(pizzaItem.form.actionButtons.create);  
45  
46    await pizzaItem.navigateToProducts();  
47    const pizzas = await products.getProducts();  
48  
49    await t.expect(pizzas.count).eql(3);  
50  });
```

PROBLEMS 1 TERMINAL ... 2: node + - ×

- Chrome 64.0.3282 / Mac OS X 10.13.3

Create Pizza

✓ should add Marvel chili Pizza to the menu!

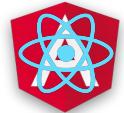
1 passed (3s)

Make changes in the source files or press ctrl+r to restart test run.

# E2E Testing: summary

- TestCafe ( Fast, 100% reliable, just NodeJS ! )
  - Windows, MacOS, Linux
  - Chrome, Firefox, Safari, Opera, Edge, IE, Custom
  - CI integration
  - TypeScript

# Components development/catalogue



# Components development



The UI Development Environment  
You'll ❤️ to use

Filter

- ▽ Welcome
  - to Storybook**
- ▽ Button
  - with text
  - with some emoji
- ▽ App Component
  - Component with separate template

## ADDON

- > Actions
- ▽ Knobs
  - Simple
  - All knobs
- > Links

# Welcome to storybook

This is a UI component dev environment for your app.

We've added some basic stories inside the `src/stories` directory.

A story is a single state of one or more UI components. You can have as many stories as you want.

(Basically a story is like a visual test case.)

See these sample [stories](#) for a component called `Button`.

Just like that, you can add your own components as stories.

You can also edit those components and see changes right away.

(Try editing the `Button` stories located at `src/stories/index.js`.)

Usually we create stories with smaller UI components in the app.

STORY

ACTION LOGGER

NOTES

KNOBS

NO KNOBS

# Components dev: add Storybook to @angular/cli



```
$ npx @storybook/cli@alpha getstorybook
```

# Storybook: write a Story

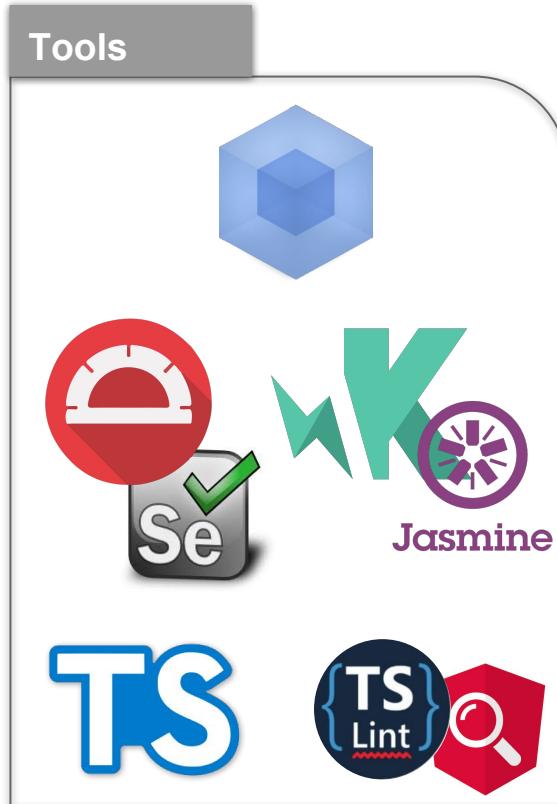
```
storiesOf('Button', module)
  .add('with some emoji', () => ({
    template: `
      <storybook-button-component
        [text]="text"
        (onClick)="onClick($event)"
      ></storybook-button-component>
    `,
    props: {
      text: '😊 😎 🤗 💯',
      onClick: () => {},
    },
  )));

```

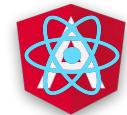
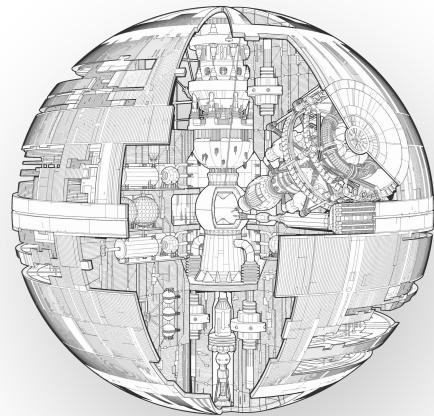
# Storybook: Component stories

```
▲ 📁 pizza-item
  ▶ 📁 _snapshots_
    ↴ 🎨 pizza-item.component.scss
    ↴ 🛡️ pizza-item.component.spec.ts
    ↴ 💾 pizza-item.component.stories.ts
    ↴ 🏙️ pizza-item.component.ts
```

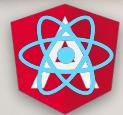
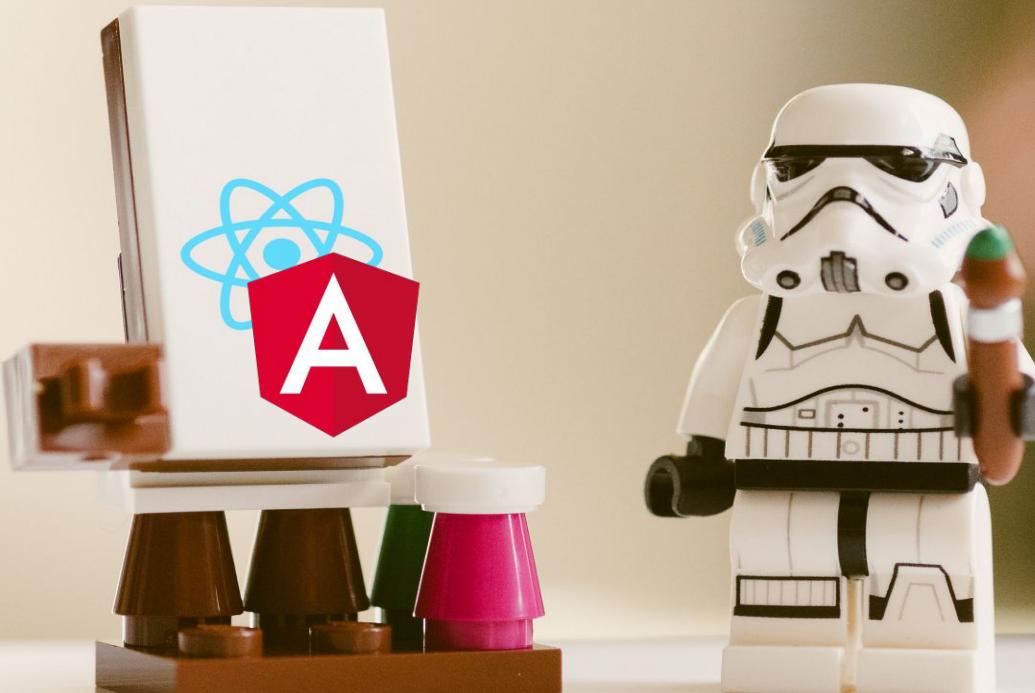
# Tooling: Summary



# Architecture



# Components



# Components: architecture

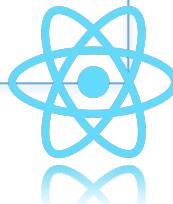
React.Component ----->

HOC f(C)=> C.f ----->

render ( JSX ) ----->

CSS / CSS in JS ----->

Immutable props ----->



@Component

@Directive

Inline template

Inline CSS

ChangeDetectionStrategy.OnPush

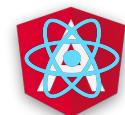
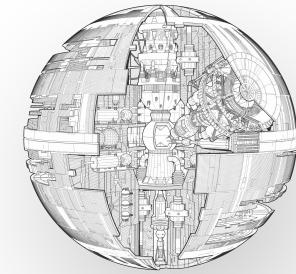
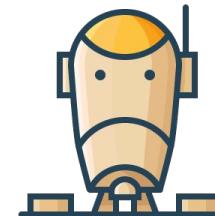


# Components: architecture

Stateful component

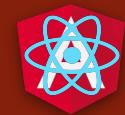


Stateless component





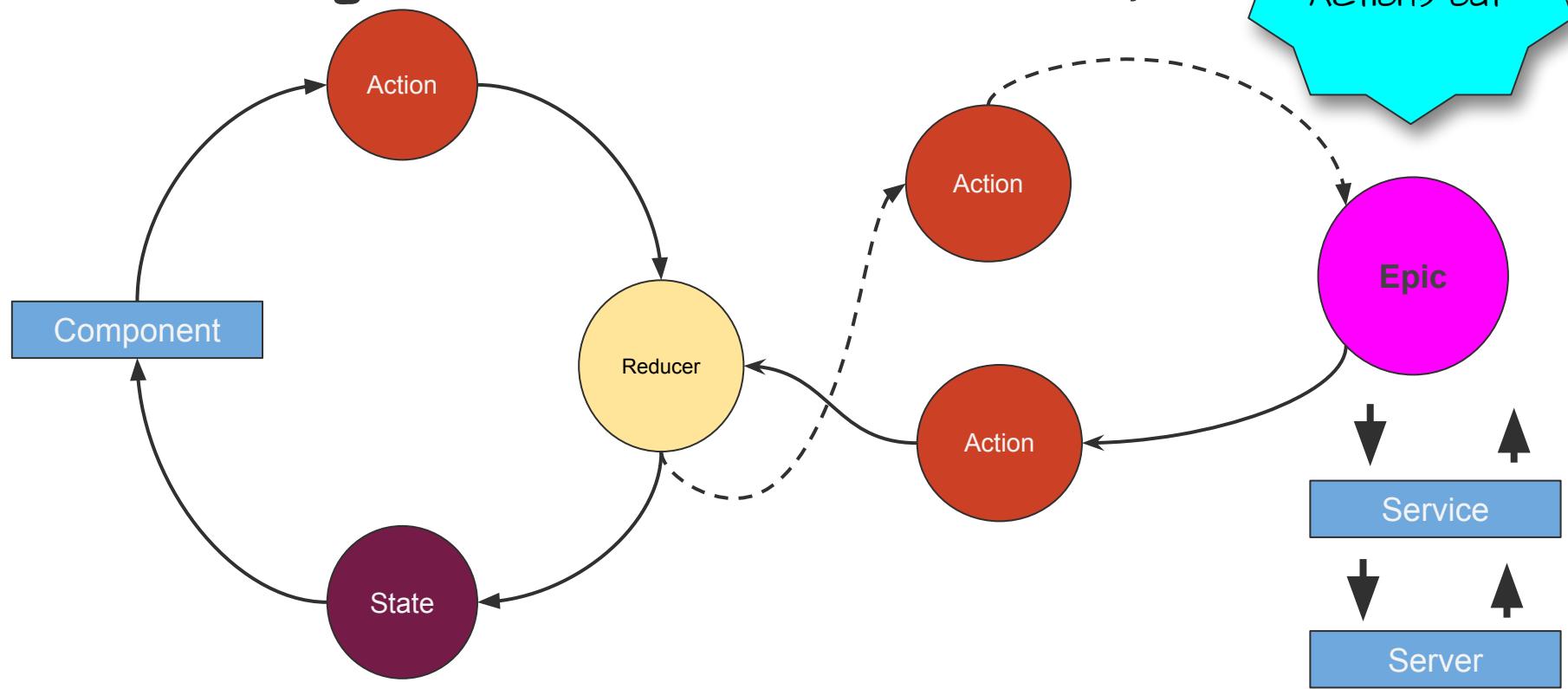
Vikings STATE



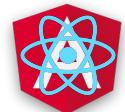
# State Management

- Redux
- Epics for handling side effects ( Rx )

# State Management: side effects via Epic



# Libraries



# Libraries: Redux



```
$ yarn add redux @angular-redux/store reselect
```

# Libraries: Epics



```
$ yarn add redux-observable rxjs
```

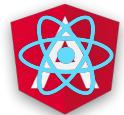
# Libraries: Axios

```
switchMap(() => {
  return this.toppingsService
    .getToppings()
    .pipe(
      map((toppings) => new toppingsActions.LoadToppingsSuccess(toppings)),
      catchError((error) => of(new toppingsActions.LoadToppingsFail(error)))
    );
  +   .then((toppings) => new toppingsActions.LoadToppingsSuccess(toppings))
  +   .catch((error) => of(new toppingsActions.LoadToppingsFail(error)));
})
```



```
$ yarn add axios
```

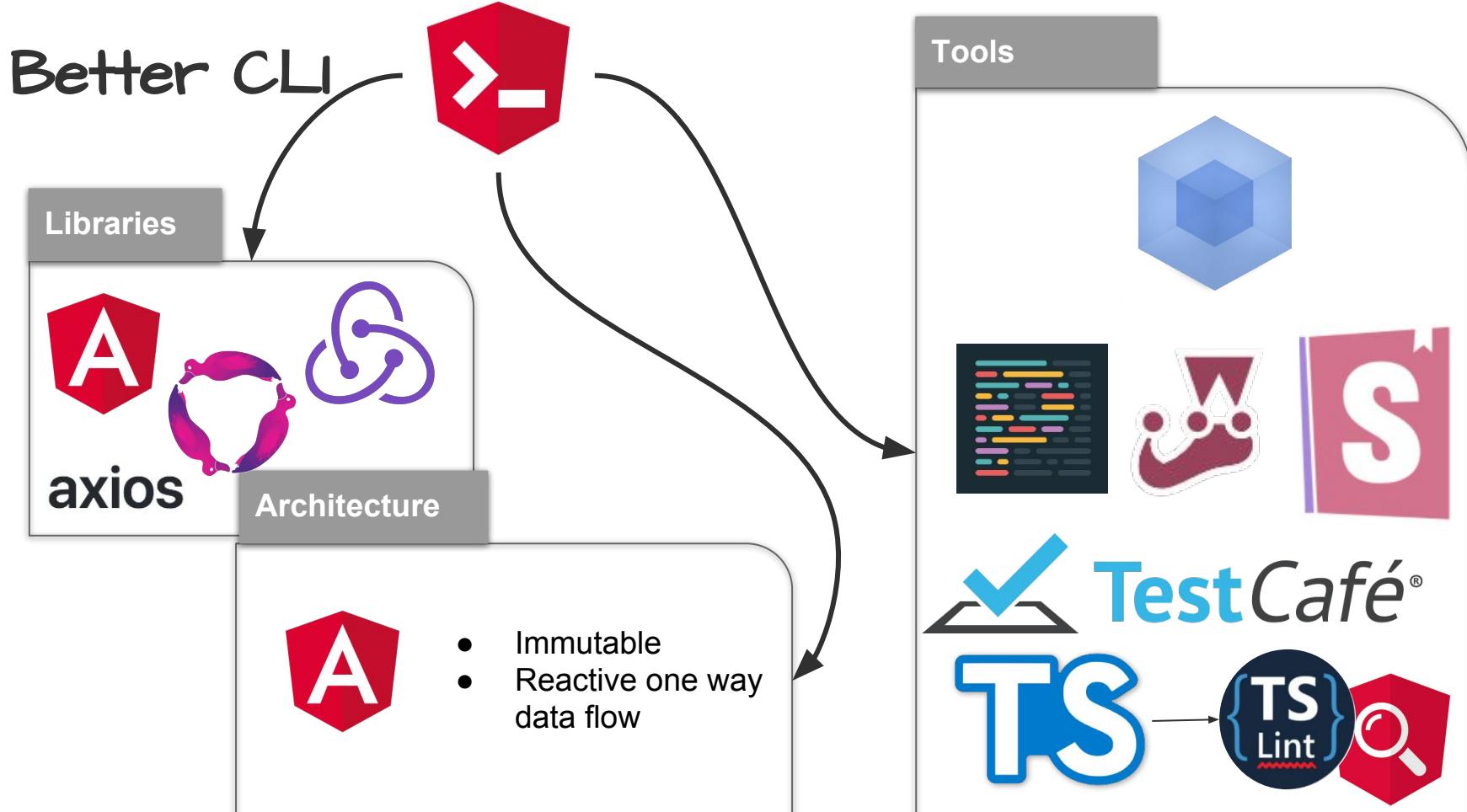
# Summary



# Summary

- Common
  - Tools
  - Architecture
  - Libraries





# Summary: OSS

- Starter
- Example app

[Hotell / react-tools-for-better-angular-apps](#)

Code Issues 0 Pull requests 1 Projects 0 Wiki Insights Settings

Use React ecosystem for unified and top notch DX for angular development

Add topics

26 commits 2 branches 2 releases 2 contributors

Branch: master ▾ New pull request Create new file Upload files Find file Clone or download ▾

 Hotell fix(npm-scripts): remove html from prettier as it is not supported Latest commit 42e7164 3 days ago

 .storybook feat(book): add storybook (#8) 5 days ago

 .vscode chore(vscode): add IDE settings 24 days ago

 api feat(api): add json-server support 4 days ago

 e2e feat(book): add storybook (#8) 5 days ago

 example-apps chore(example-apps): remove backup files from ngrx 3 days ago

 scripts test(e2e): switch to testcafe instead of selenium and protractor (#7) 6 days ago

 src feat(ng): add hmr 4 days ago

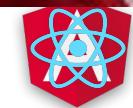
 ... footer/footer.html: add footer.html (#8) 5 days ago

# Summary: TODO

- Custom Angular CLI Schematics !



# Summary: BFF



# Office Hours: Meet The ~~Heroes~~ New Friends !



March 2, 11:55 - 12:25

Room: Midgaard

Office Hours: Meet The Heroes

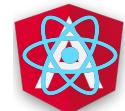
## Speakers



Martin Hochel  
ngPartyCz / EmbedIT / Czech Republic



Majid Hajian  
Futurehome.no / Norway



# Thank you !



---

Martin Hochel  
@martin\_hotell

