

BACK

TO  
THE

Server

JS



PART

IIII





# ABOUT ME

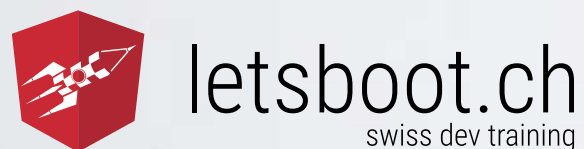
Jonas Bandi

[jonas.bandi@ivorycode.com](mailto:jonas.bandi@ivorycode.com)

Twitter: @jbandi

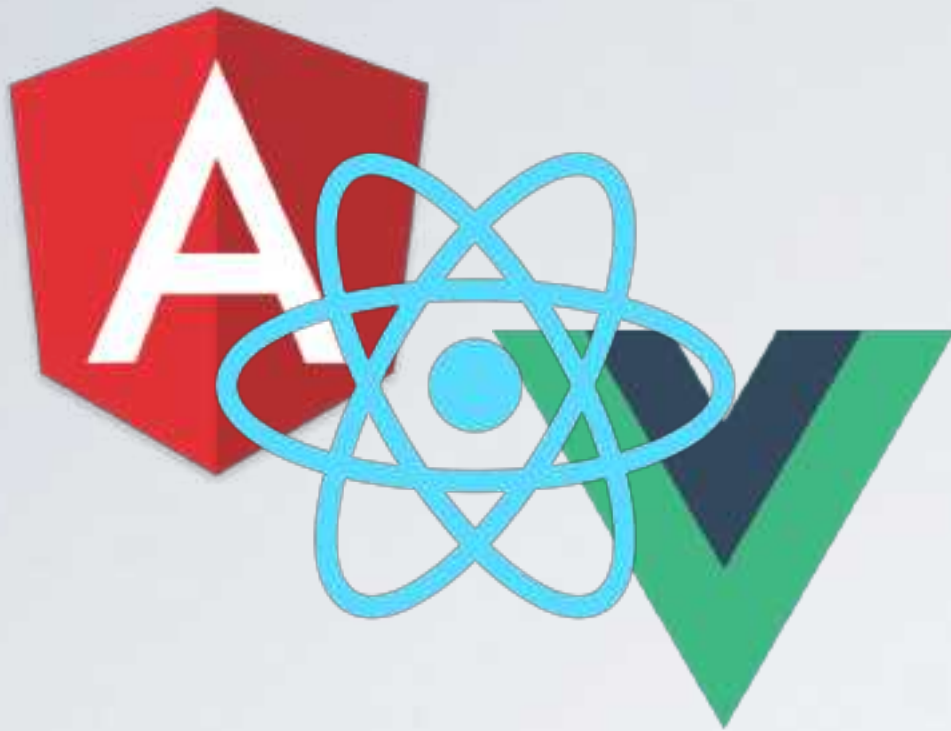


- Freelancer, in den letzten 9 Jahren vor allem in Projekten im Spannungsfeld zwischen modernen Webentwicklung und traditionellen Geschäftsanwendungen.
- Dozent an der Berner Fachhochschule seit 2007
- In-House Kurse & Beratungen zu Web-Technologien im Enterprise: UBS, Postfinance, Mobiliar, AXA, BIT, SBB, Elca, Adnovum, BSI ...



JavaScript / Angular / React / Vue / Vaadin  
Schulung / Beratung / Coaching / Reviews  
[jonas.bandi@ivorycode.com](mailto:jonas.bandi@ivorycode.com)





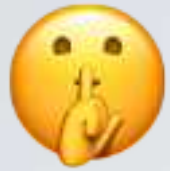
Frontend ...

.... On The Server?

Old Hat?



# Server Side Rendering



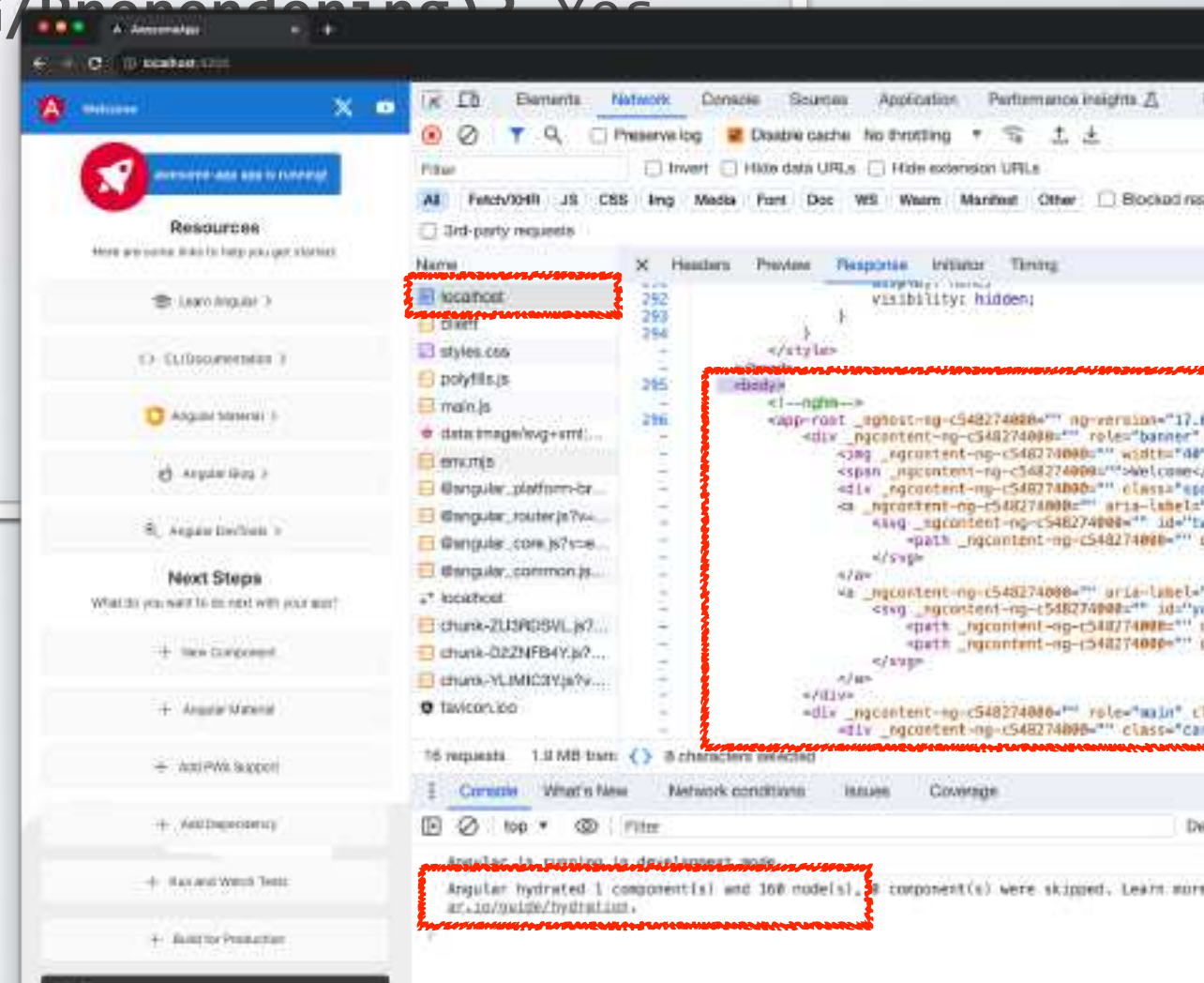
... even Angular can do that ...

Using Angular CLI: 17.0.0-rc.0

```
> npx @angular/cli@next new ng-ssr
? Which stylesheet format would you like to use? CSS
? Do you want to enable Server-Side Rendering (SSR)
  and Static Site Generation (SSG/Pre-rendering)? Yes
```

...

```
✓ Packages installed successfully.
cd ng-ssr
npm start
```



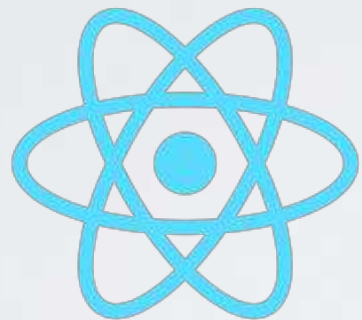


This Talk is not about Server Side Rendering ...



(... but I don't tell you the agenda ...)

They say ...

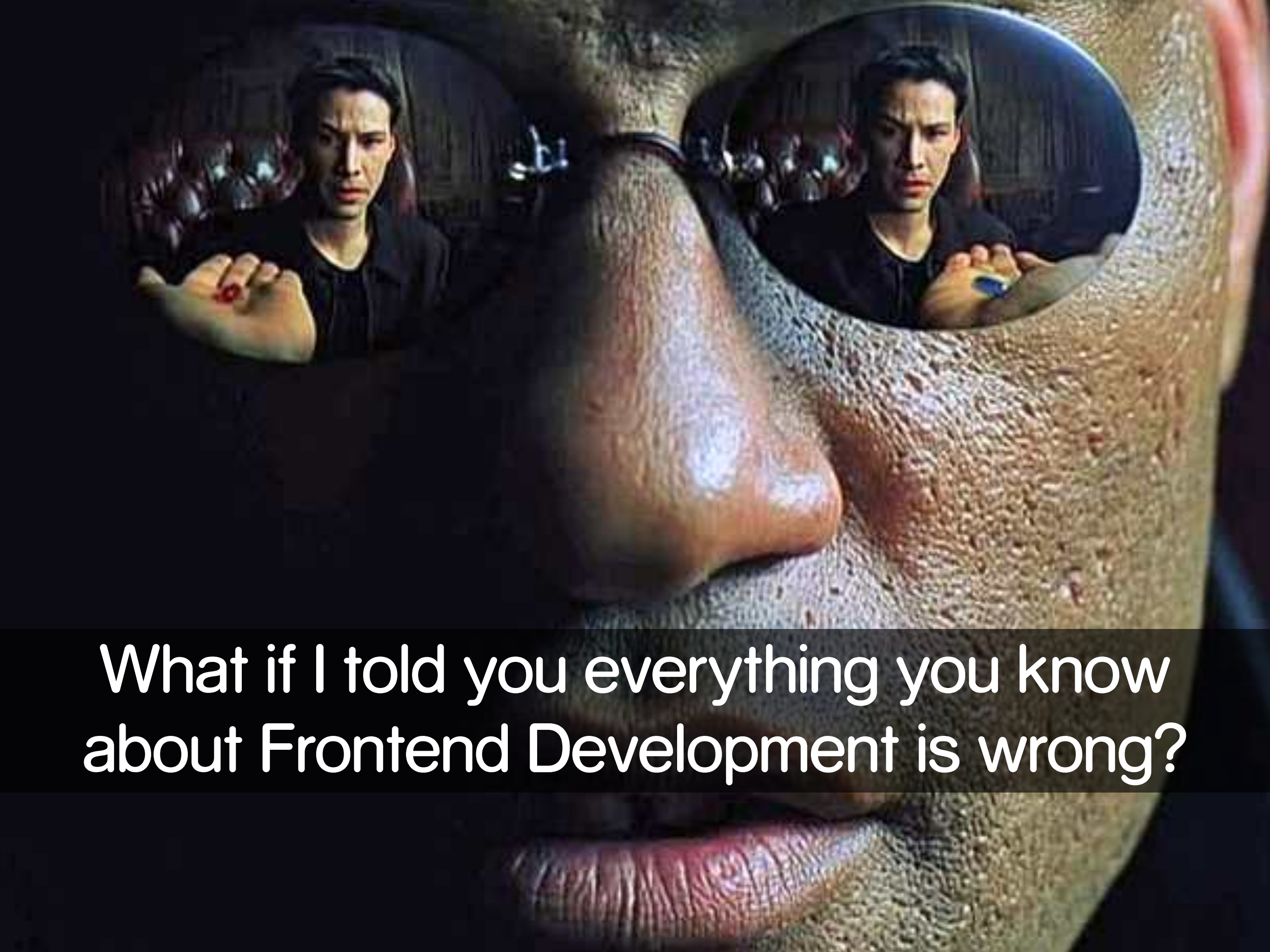


# React is Easy!

```
export function Greeter() {  
  return <h1>Hello World!</h1>;  
}
```

... until you start fetching data!





What if I told you everything you know about Frontend Development is wrong?

They say ...



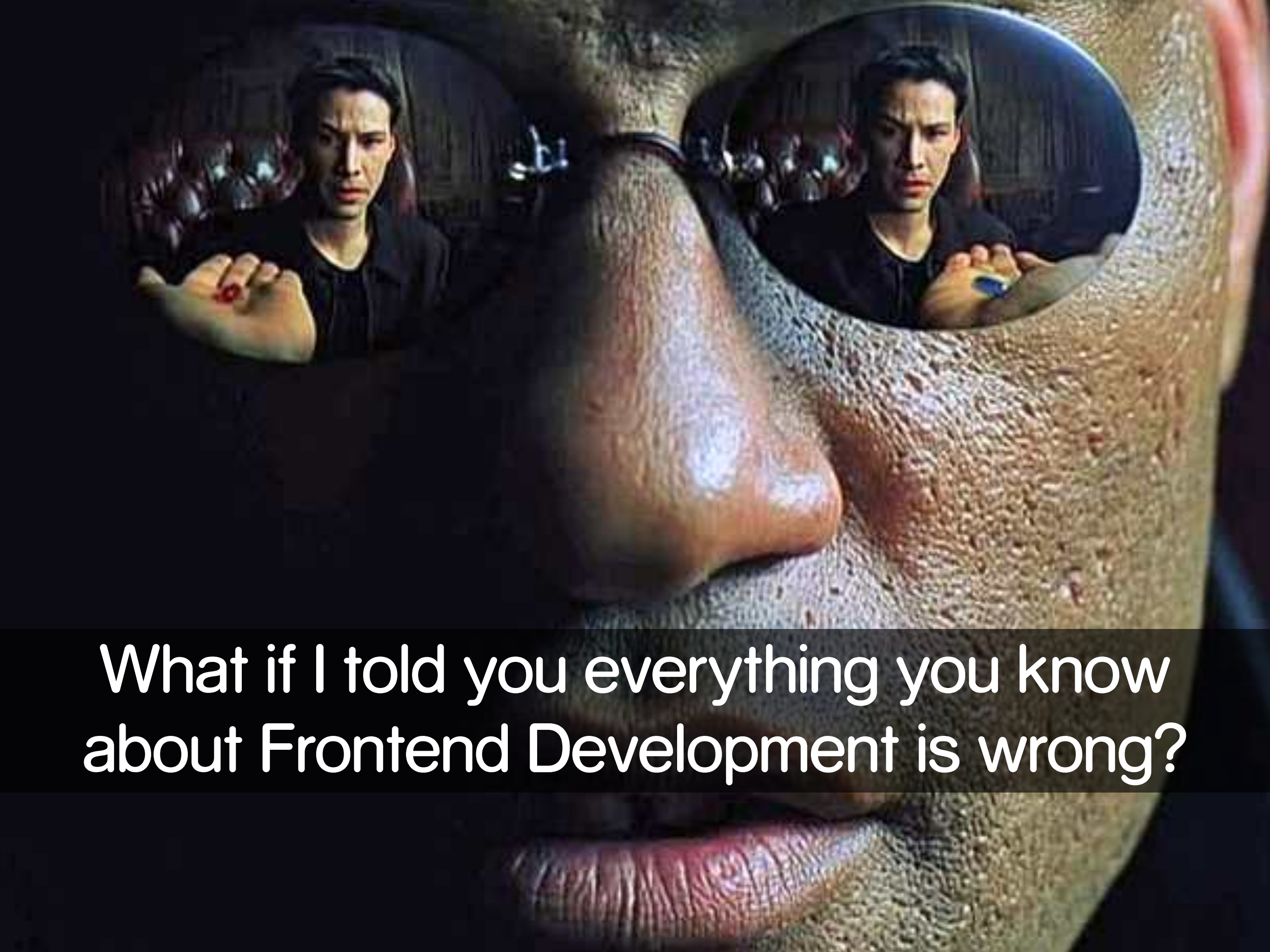
for SPAs thou  
shalt build APIs!

... but maybe you just want access to  
server-side data & functionality?

Few organizations are in the business of delivering APIs.

- Stefan Tilkov: Wait, what!? Our microservices have actual human users?  
<https://www.youtube.com/watch?v=pUlgXA0rfwc>





What if I told you everything you know about Frontend Development is wrong?



**danabra.mov** 

@dan\_abramov



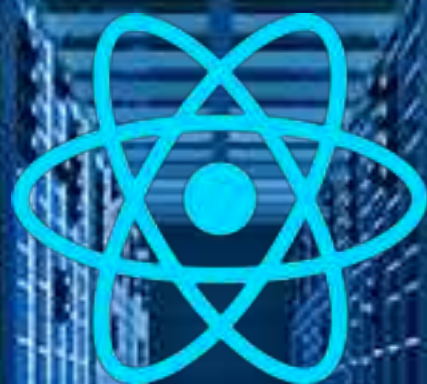
never write another API

6:19 AM · Mar 4, 2023 · **39.5K** Views

[https://twitter.com/dan\\_abramov/status/1631887155080429569](https://twitter.com/dan_abramov/status/1631887155080429569)

Wouldn't that be tempting?





# React Server Components (RSC)





Back to PHP?





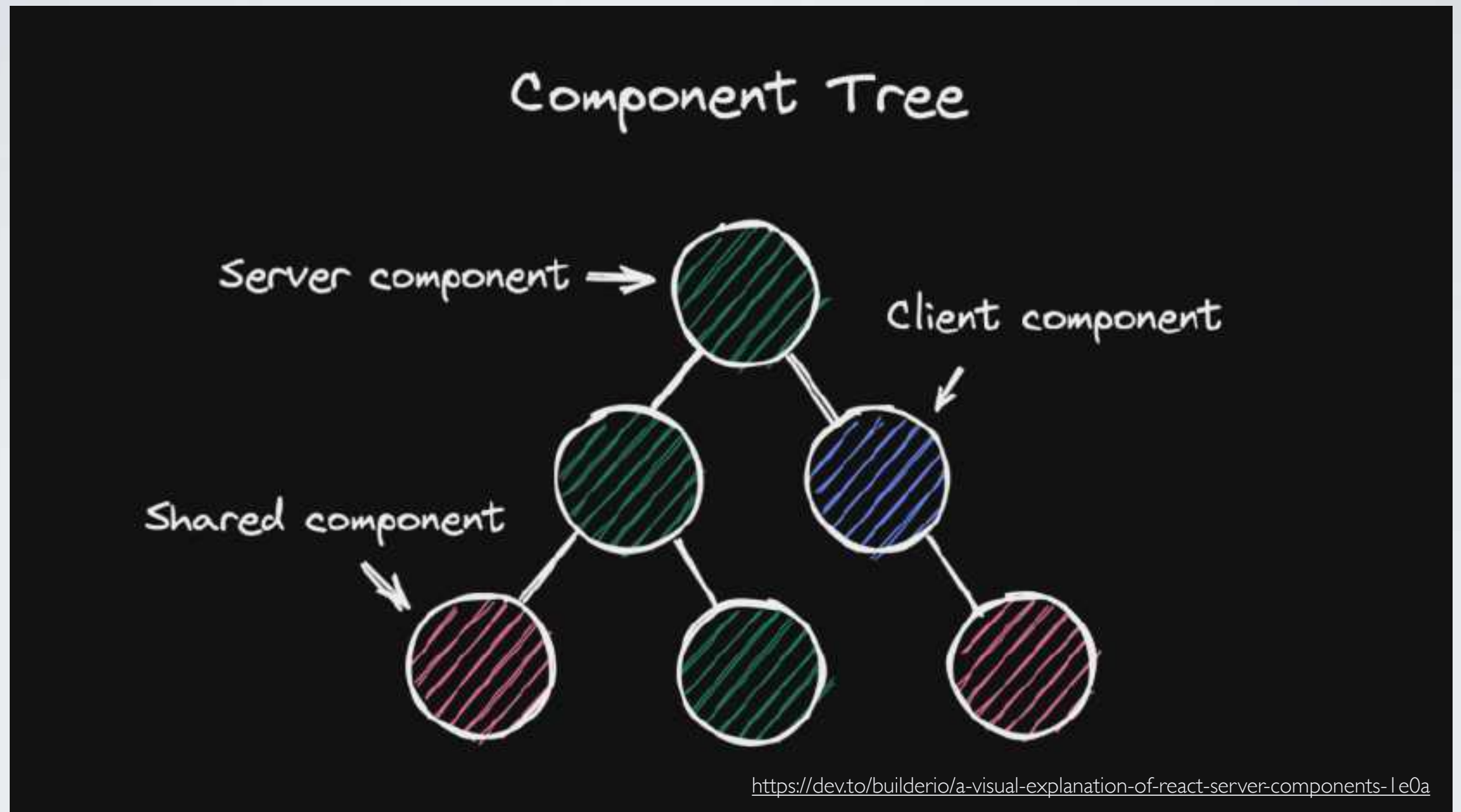
Prepare for ...



DEMO: FormDemo



# Your component tree spans across server and client!



Server Components can import Client Components.  
Client Components can't import Server Components.  
Server Components can be **children** of Client Components.

## DEMO: ListDemo - Refreshing Data

# It is still a SPA!

## Your Code

## React Client Runtime

Client Component

Virtual DOM

In-Memory, implemented in JavaScript

Browser DOM 

```
export function Greeter() {  
  return (  
    <h1>Hello World!</h1>  
  );  
}
```

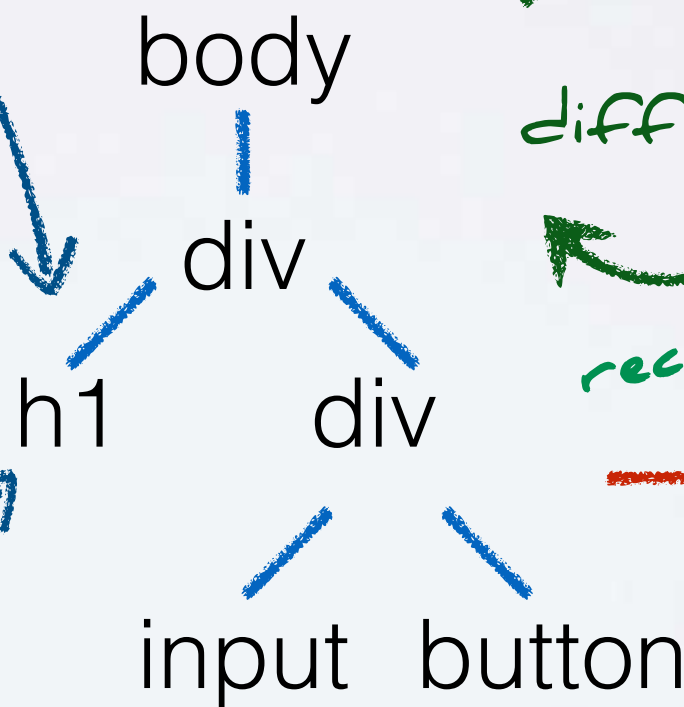
render instructions  
running on the client

render instructions:  
1. created on the server

2. sent to the client

3. executed on the client

Server Component



diff

reconciliation

patch

```
<body>  
  <div>  
    <h1>...</h1>  
    <div>  
      <input/>  
      <button>  
        </button>  
    </div>  
  </div>  
</body>
```

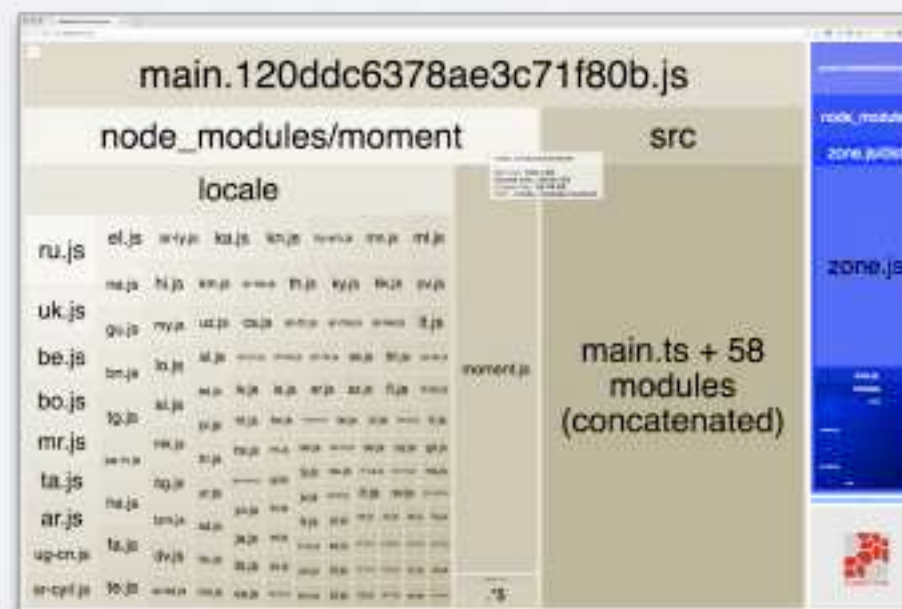
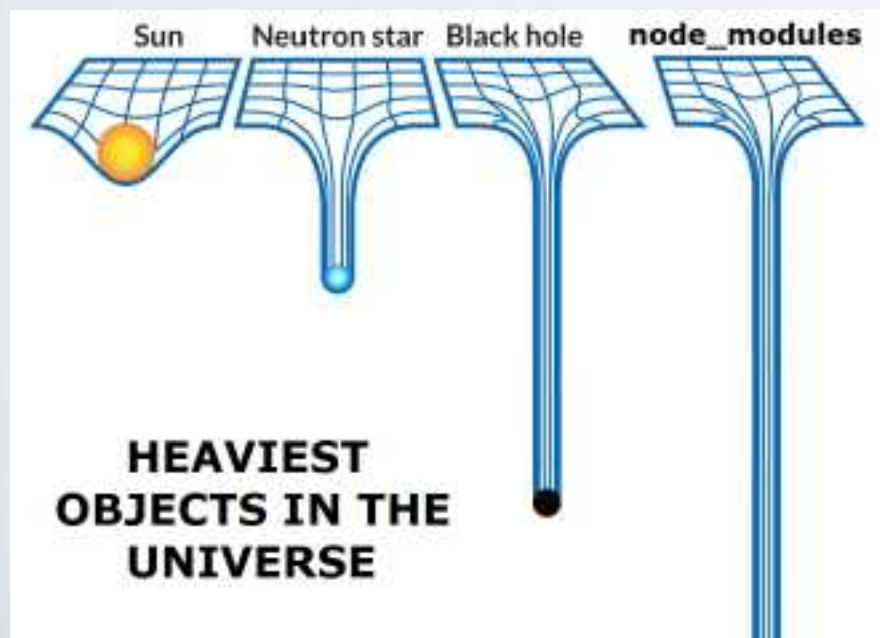




They say ...



Thou shalt watch your dependencies!



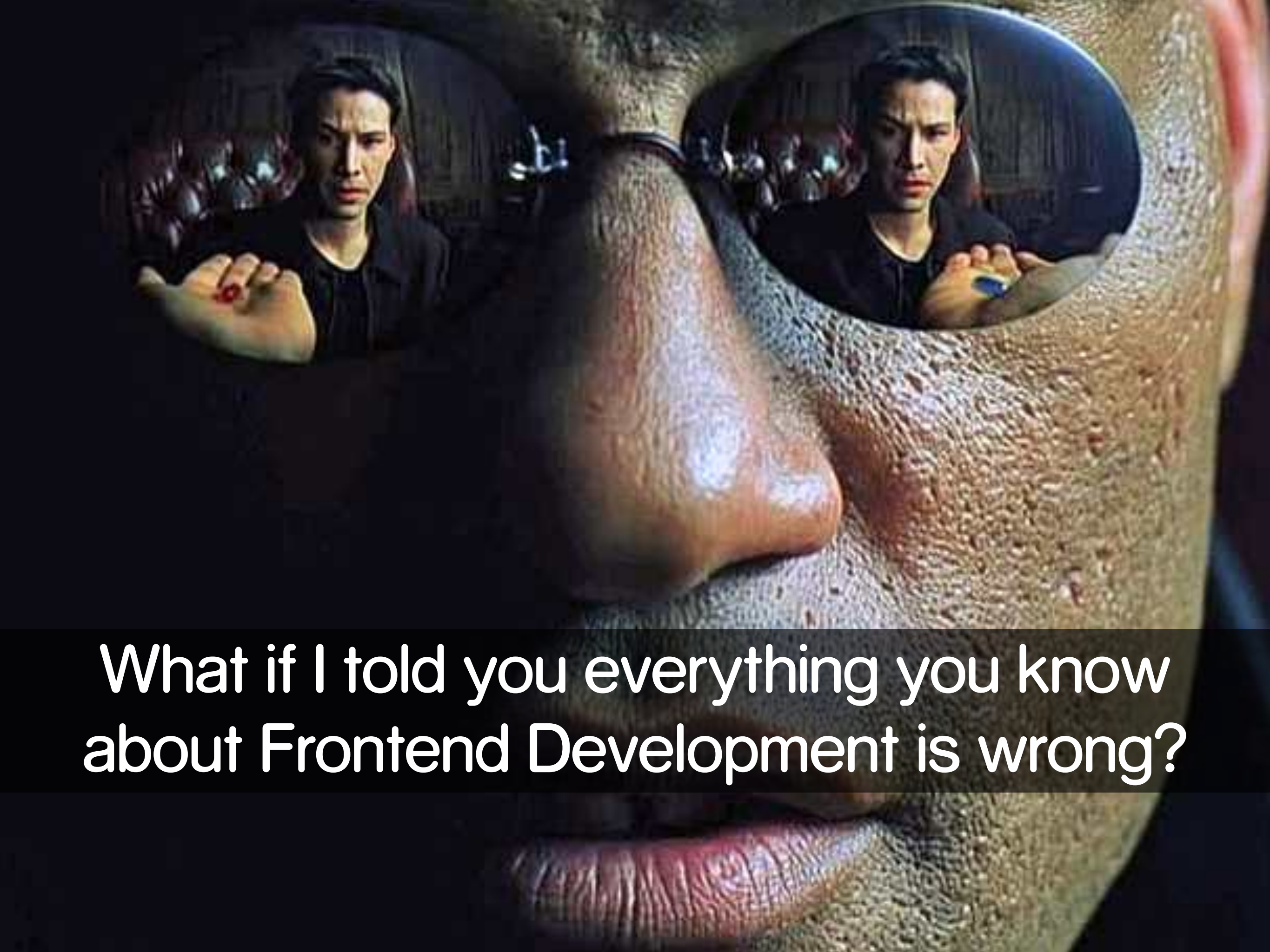
Webpack Bundle Analyzer

```
import React from 'react'; 44K
import ReactDOM from 'react-dom'; 218.9K
import uniqueId from 'lodash/uniqueId'; 2.2K
```

VS Code Plugin: Import-Cost



<https://bundlephobia.com/>



What if I told you everything you know about Frontend Development is wrong?



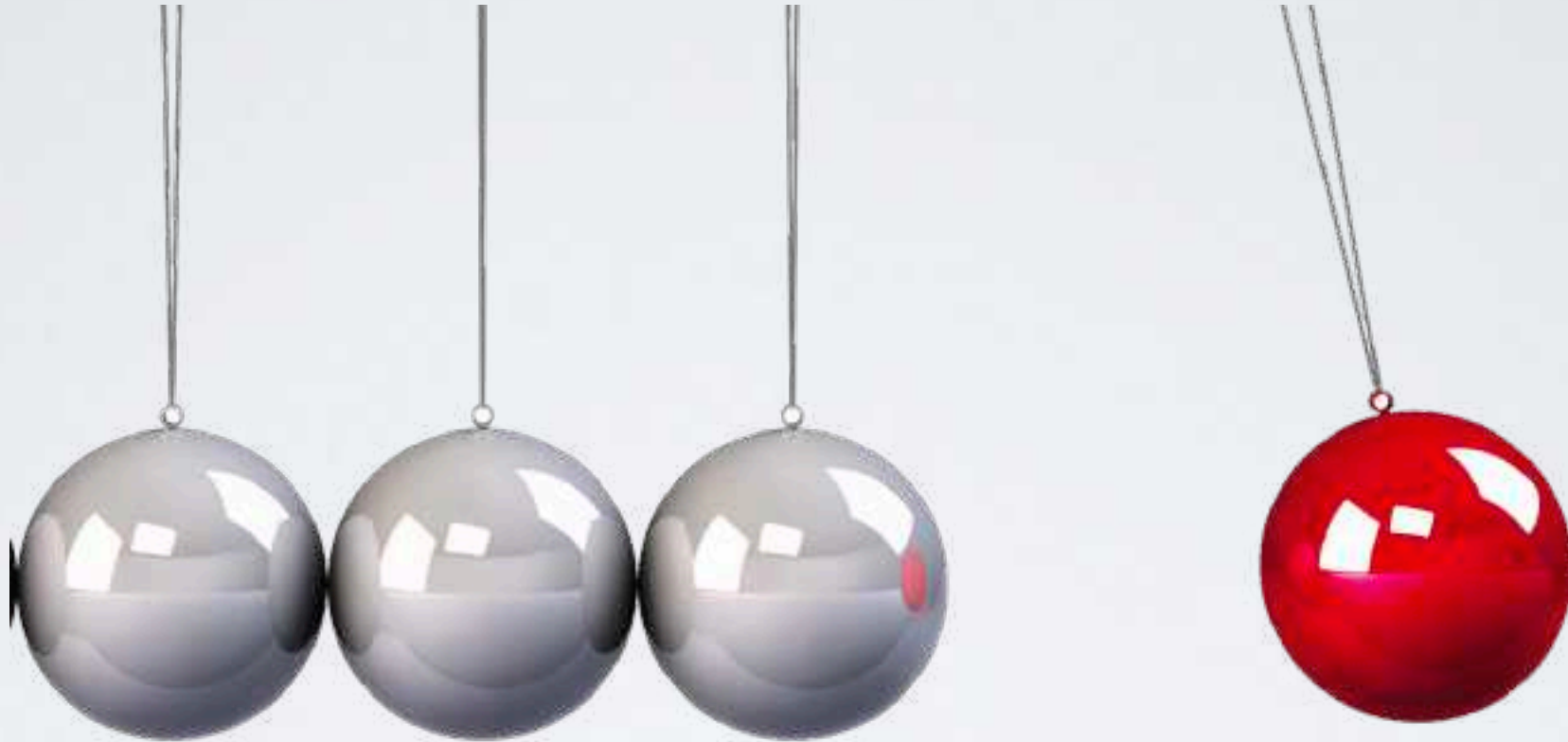
## DEMO: Chart with D3

```
npm install date-fns  
npm install d3 @types/d3
```





# The Pendulum is Swinging ...



Server ←————→ Client

... will it ever stop?

# My personal Takeaway

React Server Components try to:

- be an answer for data-fetching and mutations in SPAs
- simplify the full-stack programming model
- improve performance by enabling smaller JS bundles



# Prior Art

*Prior Art - Part I:*  
Data Fetching & Mutations

**STAR WARS**  
**RETURN<sup>OF</sup> THE RPC**

... in other motheren frontend frameworks.



<https://remix.run/docs/en/main/route/loader>

<https://remix.run/docs/en/main/guides/data-writes#remix-mutation-start-to-finish>



SOLIDJS

<https://start.solidjs.com/core-concepts/data-loading>

<https://start.solidjs.com/core-concepts/actions>



SVELTEKIT

<https://kit.svelte.dev/docs/load>

<https://kit.svelte.dev/docs/form-actions>



qwik Qwik City

<https://qwik.builder.io/docs/route-loader/>



# *Prior Art - Part 2:*

## Server Driven SPAs

aka "Live View"



Vaadin



Blazor Server



Phoenix Framework

Enabling SPAs with a server-side programming model and  
no need for a REST API.

Demos:

<https://labs.vaadin.com/business/>

<https://blazor.syncfusion.com/demos/datagrid/overview?theme=bootstrap4>

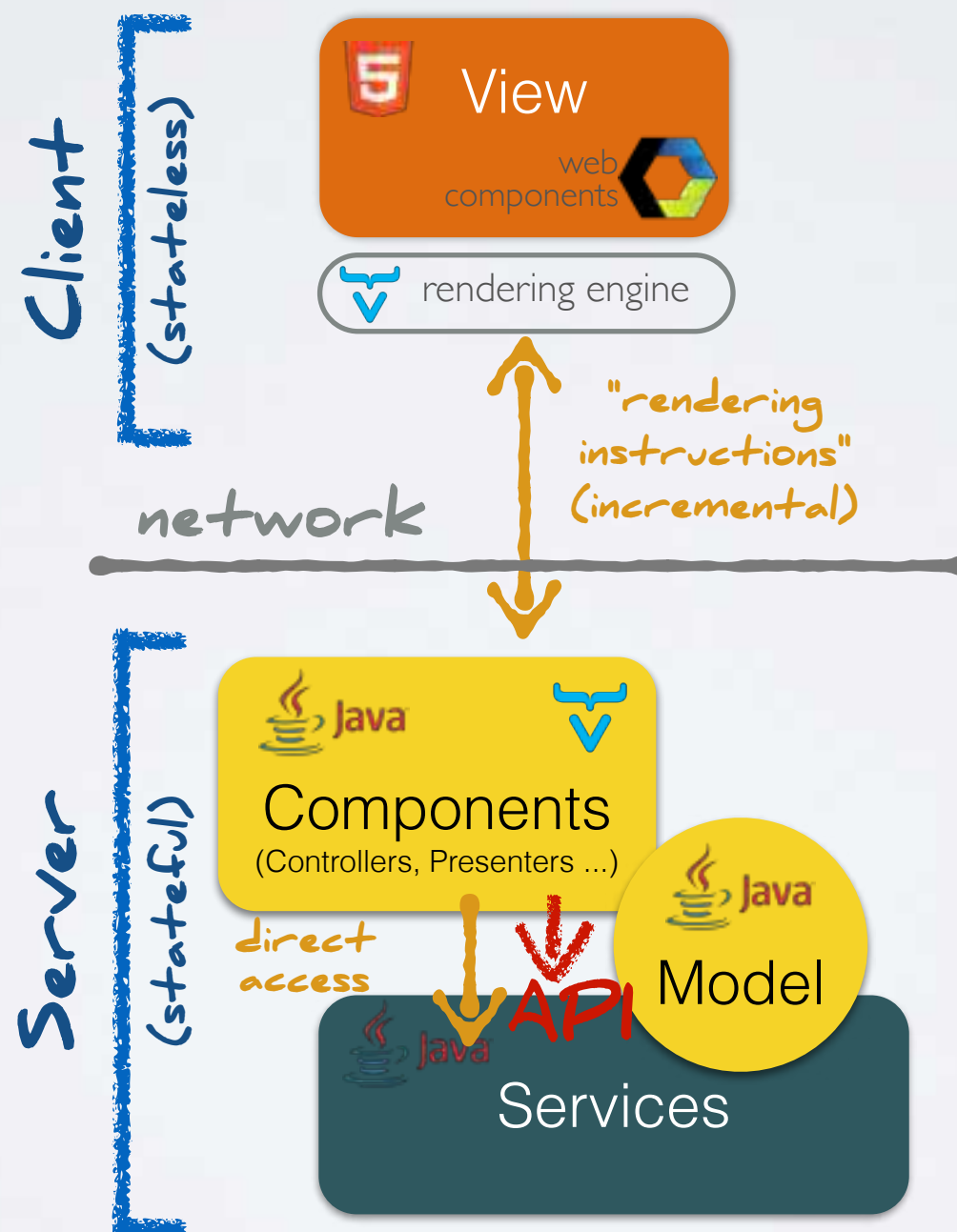
<https://liveview.zorbash.com/>

Term definition: <https://github.com/dbohdan/liveviews>

# Vaadin Architecture

## Flow

using the browser just as a "rendering engine"



# *Prior Art - Part 3:*

## Island Architecture

The islands architecture encourages small, focused chunks of interactivity within server-rendered web pages



<https://www.patterns.dev/posts/islands-architecture>



# Follow-Up Topics:

React is getting more complicated:

```
'use client'  
'use server'  
import 'server-only'
```

```
use  
useEffectEvent  
useTransition  
Suspense  
useFormState  
useFormStatus  
taintObjectReference
```

The boundary between React and Next.js becomes blurry ...

Compatibility and implications on 3rd-Party Libraries.

New patterns will emerge ...

Component Composition

DataFlow & DataFetching / Mutations

# Thank you!

Slides & Code: <https://github.com/jbandi/baernerjs-rsc>

## Questions? Discussions ...



Jonas Bandi

JavaScript / Angular / React / Vue / Vaadin

Schulung / Beratung / Coaching / Reviews

[jonas.bandi@ivorycode.com](mailto:jonas.bandi@ivorycode.com)