

ABOUT ME

Jonas Bandi

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 ...



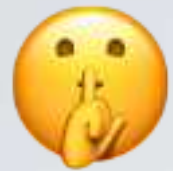
digicomp



JavaScript / Angular / React / Vue / Vaadin
Schulung / Beratung / Coaching / Reviews
jonas.bandi@ivorycode.com



Server Side Rendering (SSR)



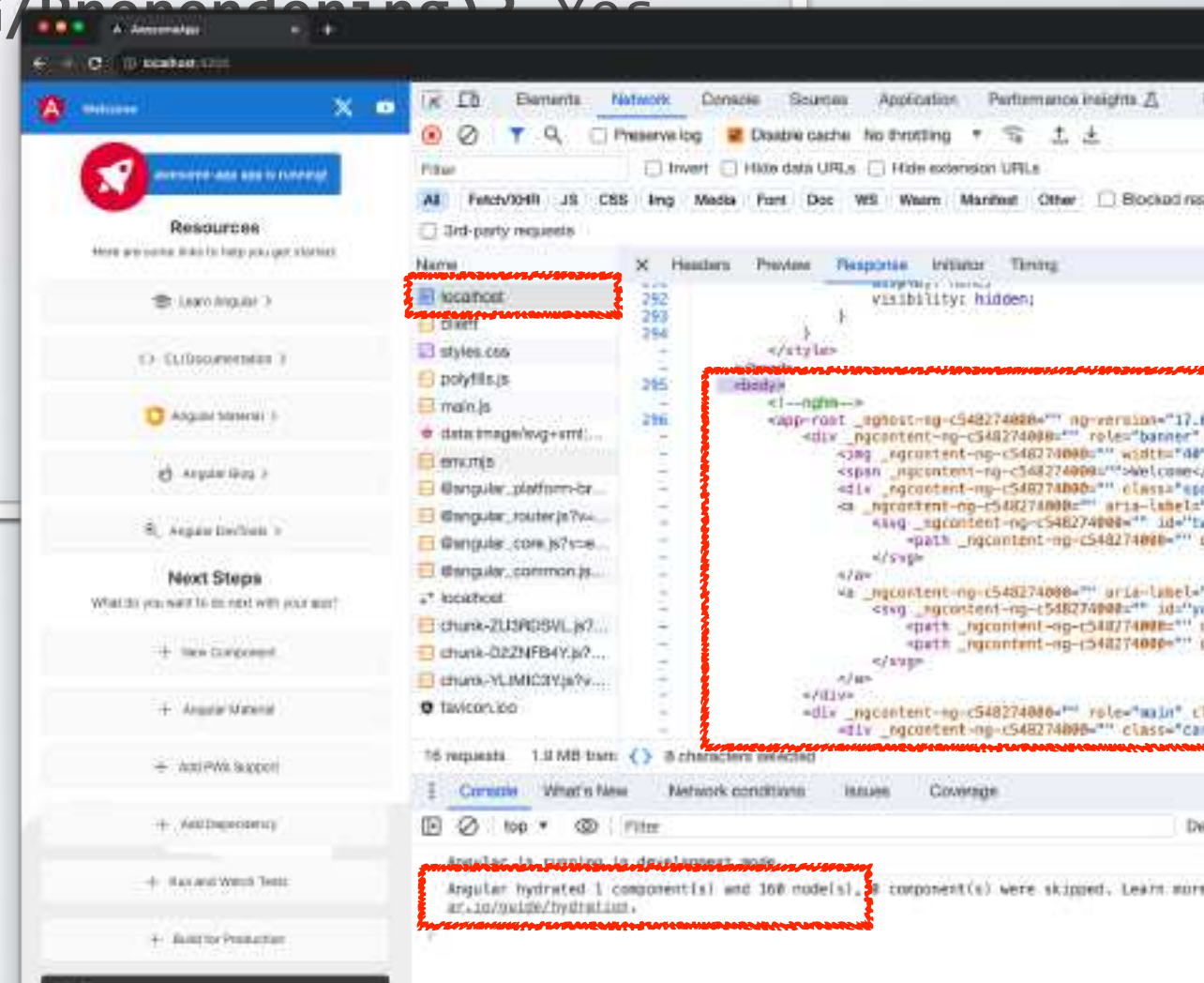
... even Angular can do that ...

Using Angular CLI v17

```
> 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
```

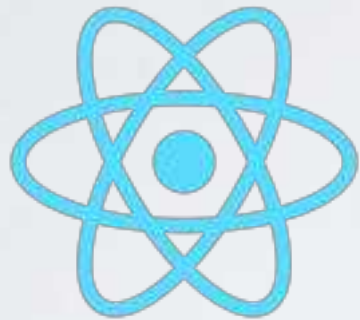


This talk is not about Server Side Rendering ...



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

They say ...



React is Easy!

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


The power of React is a Component Model which enables simple & elegant composition ...



Cory House 

@housecor



I love the simplicity of React's reuse model.

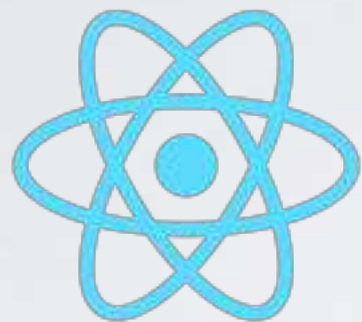
Repeating JSX? Create a component.

Repeating logic? Create a hook.

I can compose these simple building blocks in infinite ways.

1:08 PM · Nov 25, 2023 · **16.7K** Views

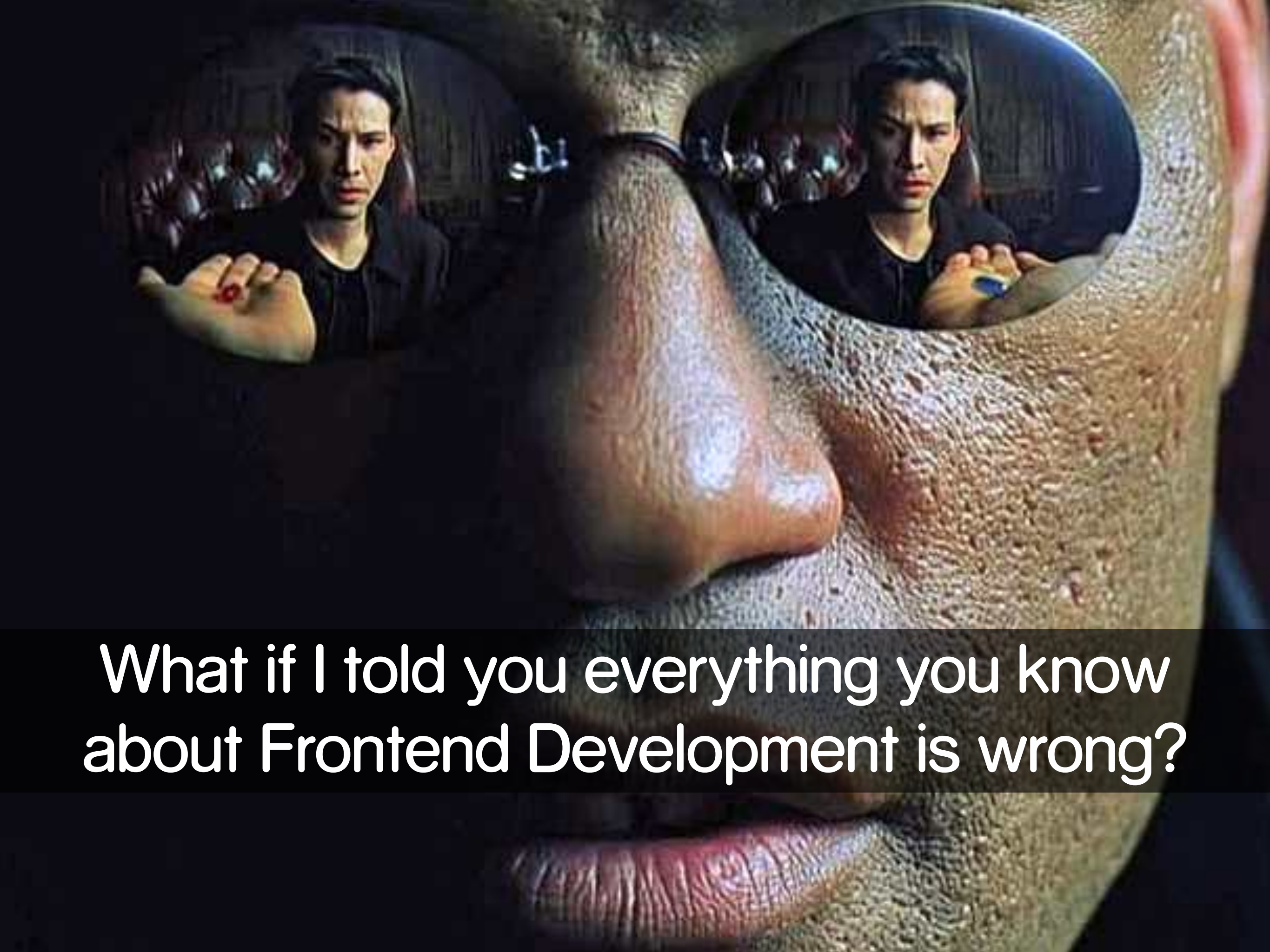
<https://twitter.com/housecor/status/1728385239611789758>



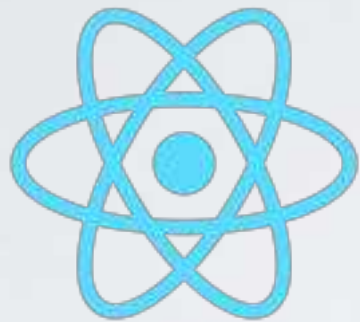
React is Easy!

```
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?



React **is** Easy!

```
export async function Greeter() {  
  const data = await fetchData();  
  return <h1>{data}</h1>;  
}
```



They say ...



for SPAs thou
shall build APIs!

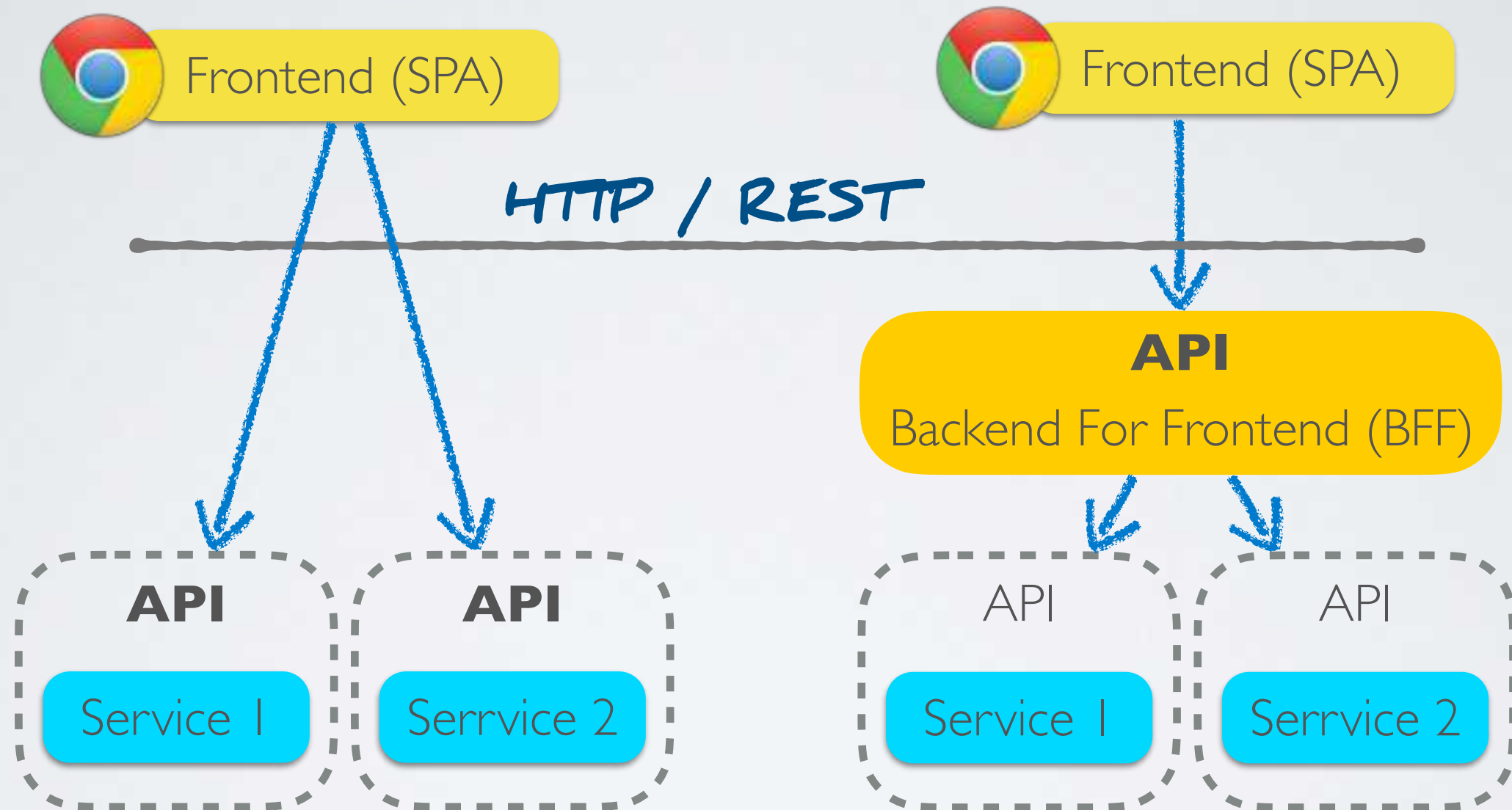
... but maybe you just want access to
server-side data & functionality?

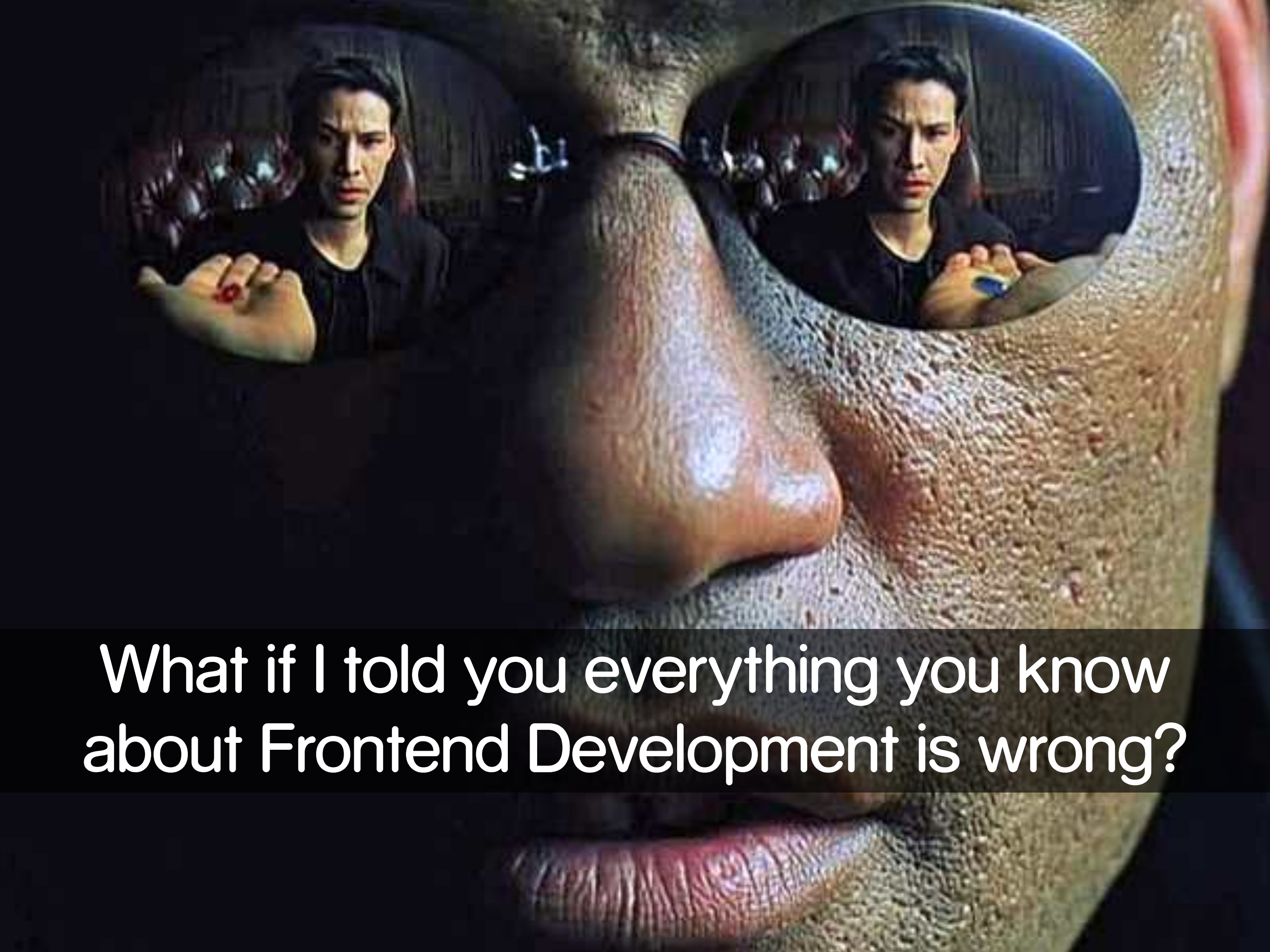
Few organizations are in the business of delivering APIs.

- Stefan Tilkov: Wait, what!? Our microservices have actual human users?

<https://www.youtube.com/watch?v=pUlgXA0rfwc>

Architecture: APIs for SPAs





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



danabra.mov 

@dan_abramov

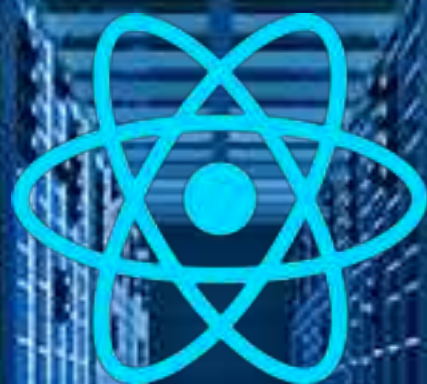


never write another API

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

https://twitter.com/dan_abramov/status/1631887155080429569

Wouldn't that be tempting?



React Server Components (RSC)

NEXT.JS



Back to PHP?

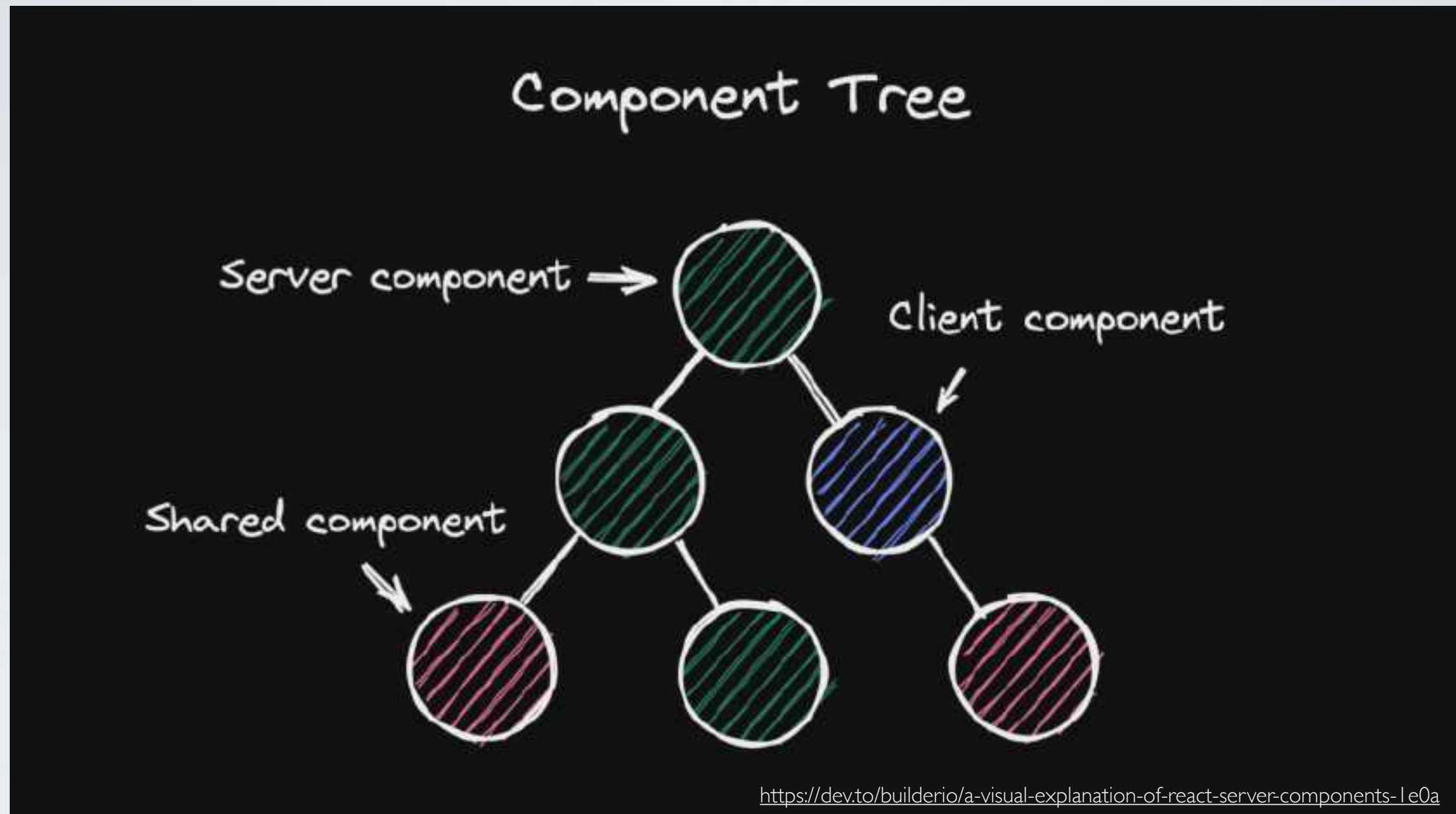


Prepare for ...



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.

Server Components can't:

- can't use **useState** / **useReducer**
- can't use **useContext**

DEMO: ListDemo - Refreshing Data

It is still a SPA!

Your Code

 React Client Runtime

 generate a react tree on the client

Client Component

Virtual DOM

In-Memory, implemented in JavaScript

Browser DOM 

render instructions
running on the client

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

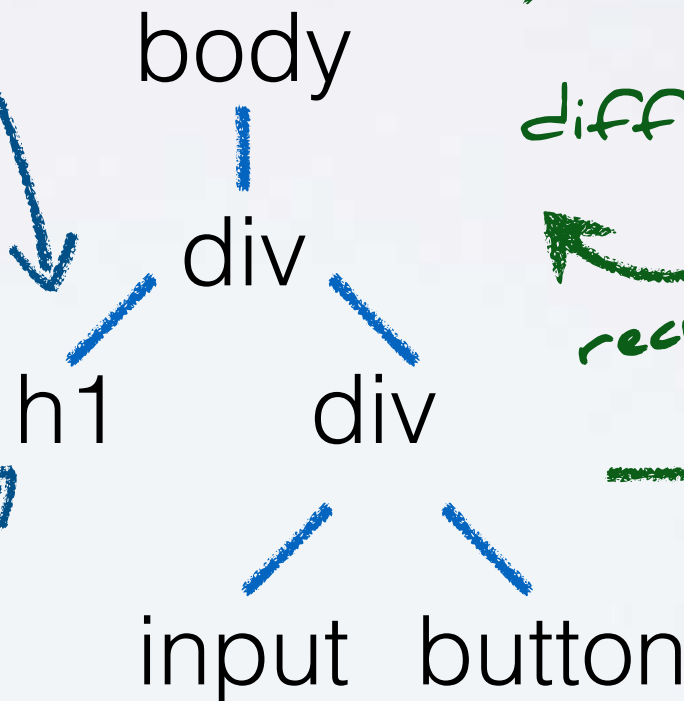
render instructions:
1. created on the server

2. sent to the client

Server Component

3. executed on the client

 generate a react tree on the server and send it to the client for rendering



diff

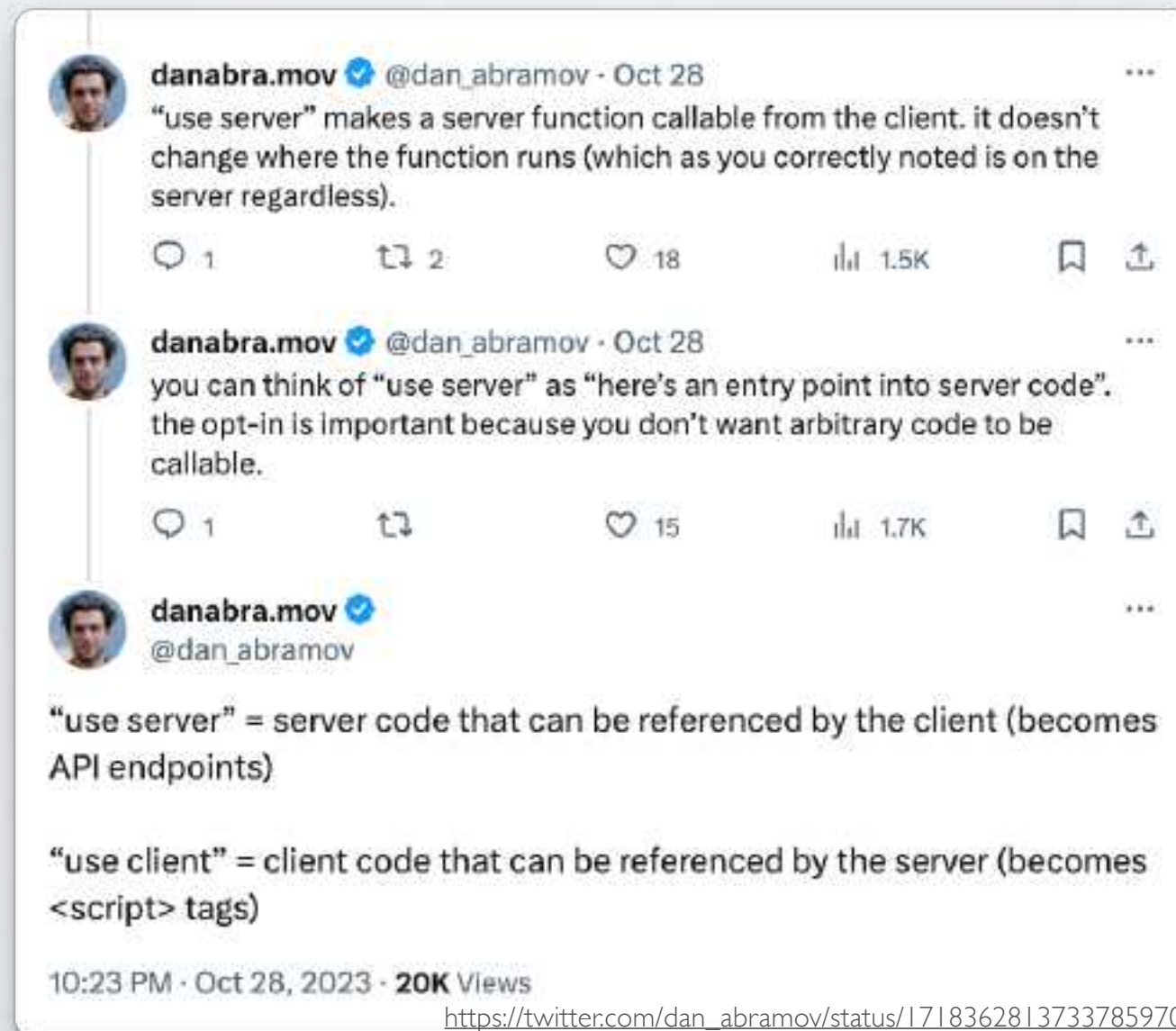
reconciliation

patch

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

How does it work ...

'use client'
'use server'



The screenshot shows a Twitter thread from the user **danabra.mov** (@dan_abramov) dated Oct 28. The thread consists of three tweets. The first tweet explains that "use server" makes a server function callable from the client without changing where the function runs. The second tweet explains that "use server" can be thought of as an entry point into server code, where the opt-in is important to prevent arbitrary code from being callable. The third tweet defines "use server" as server code referenced by the client (becoming API endpoints) and "use client" as client code referenced by the server (becoming <script> tags). The thread has 10:23 PM · Oct 28, 2023 · 20K Views and a link to the tweet.

danabra.mov ✓ @dan_abramov · Oct 28
"use server" makes a server function callable from the client. it doesn't change where the function runs (which as you correctly noted is on the server regardless).
1 2 18 1.5K

danabra.mov ✓ @dan_abramov · Oct 28
you can think of "use server" as "here's an entry point into server code". the opt-in is important because you don't want arbitrary code to be callable.
1 15 1.7K

danabra.mov ✓ @dan_abramov
"use server" = server code that can be referenced by the client (becomes API endpoints)
"use client" = client code that can be referenced by the server (becomes <script> tags)
10:23 PM · Oct 28, 2023 · 20K Views
https://twitter.com/dan_abramov/status/1718362813733785970



Back to PHP?

... yes but in a good way!

The simplicity of PHP combined with the power of React .



Adam Rackis
@AdamRackis

Follow



React team: "Hey we cooked up a way to directly invoke server code from the front end. Basically our own RPC. Now you have less need to cobble together http endpoints for every little thing"

The internet: "hurr durr I guess we're back to asp"

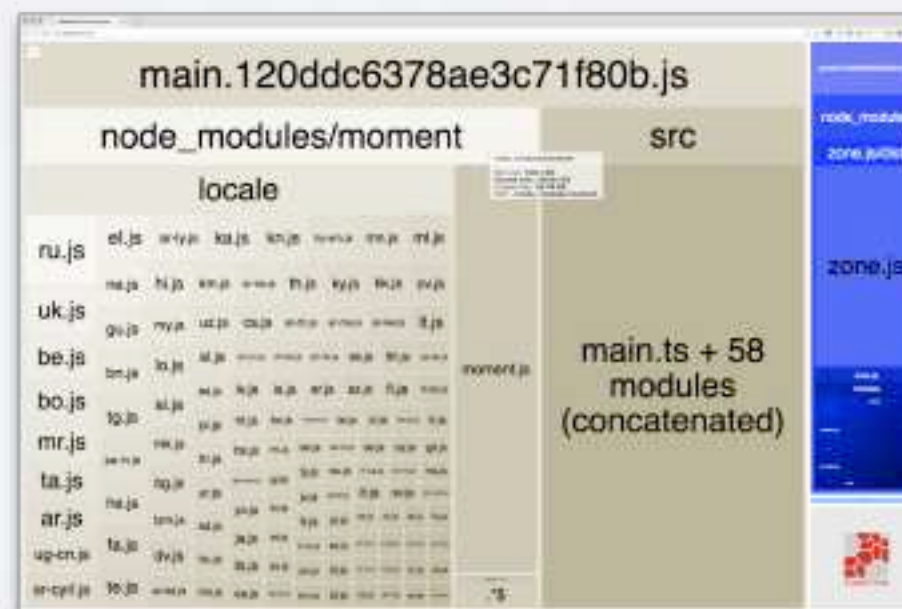
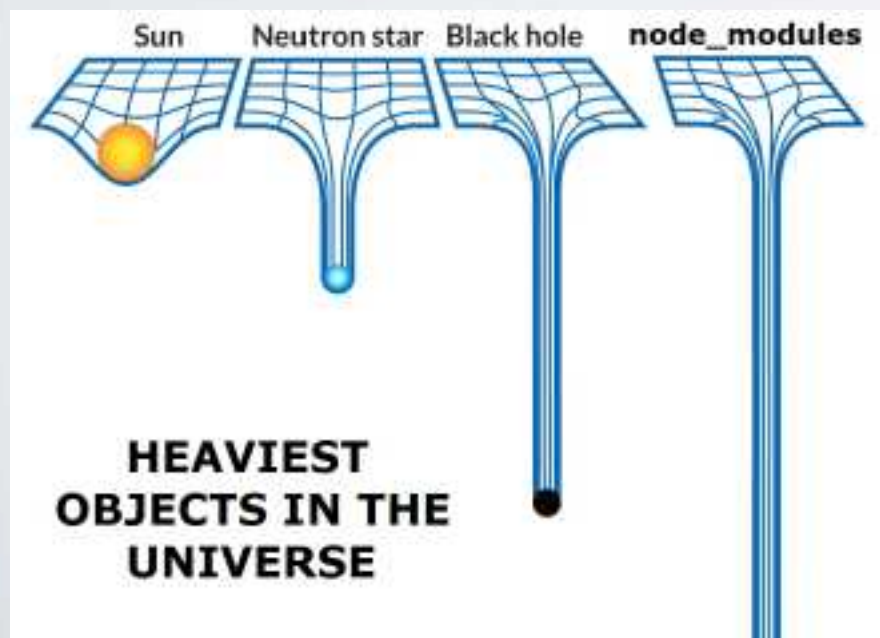
9:41 PM · Oct 27, 2023 · 84.6K Views

<https://twitter.com/AdamRackis/status/1717989813784138225>

They say ...



Thou shalt watch your dependencies!



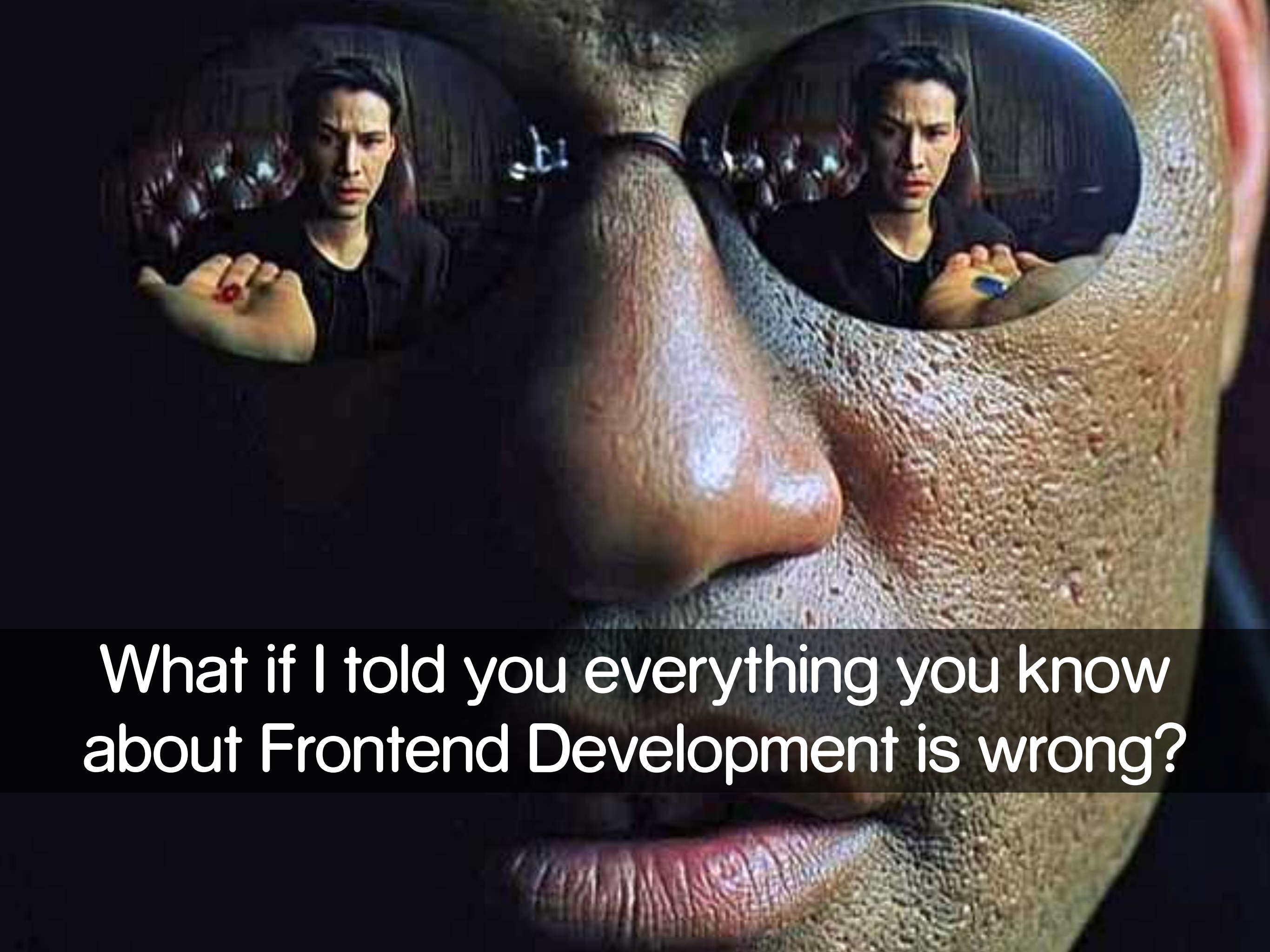
Webpack Bundle Analyzer

```
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/>



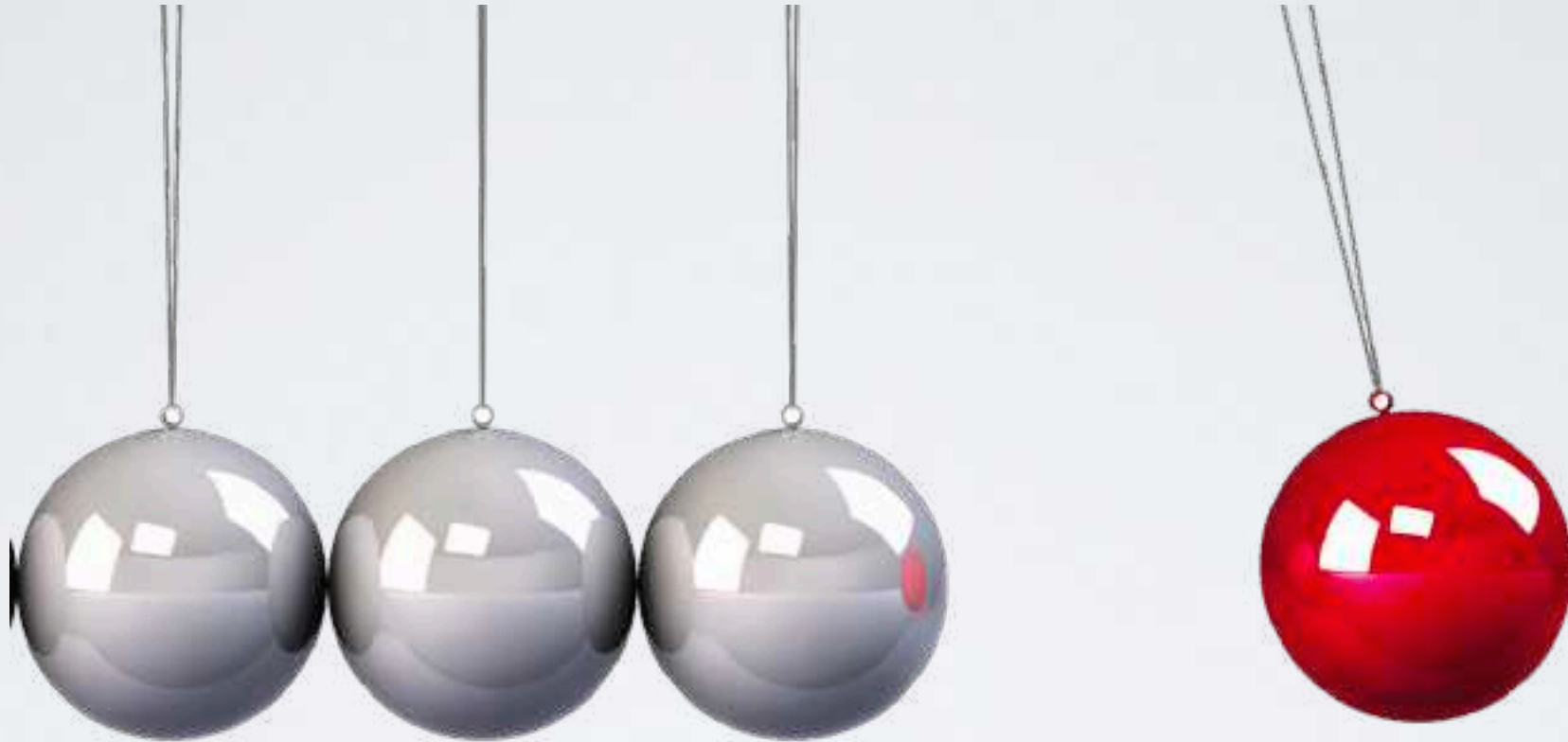
What if I told you everything you know about Frontend Development is wrong?

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

RSC attempts to improve React in many ways:

- Provide an official API for loading data or code
- Ensure that API is composable across the ecosystem
- Remove the need for framework-specific data-fetching APIs
- Allow non-interactive components to skip client-side rendering (and thereby not adding any JS to the client bundle)



Sam Selikoff ✓

@samselikoff



So the benefits of Server Components have nothing to do with HTML.
Rather, they let React developers

- Use server-only libraries directly in components
- Compose server and client features without stepping outside React's programming model
- Build non-interactive pages with OCJS

9:06 PM · Oct 17, 2023 · **28.8K** Views

<https://twitter.com/samselikoff/status/1714357316789719076>

The power of React is simple & elegant composition ...



Cory House 
@housecor



I love the simplicity of React's reuse model.

Repeating JSX? Create a component.
Repeating logic? Create a hook.

I can compose these simple building blocks in infinite ways.

1:08 PM · Nov 25, 2023 · **16.7K** Views

<https://twitter.com/housecor/status/1728385239611789758>

... RSC aim to extended this to the server.



Sam Selikoff at NextJs Conf 2023

<https://www.youtube.com/watch?v=9CN9RCzznZc&t>



Sam Selikoff 
@samselikoff



The onus is definitely on React to figure out how to best communicate the RSC architecture and clear the misconception. It's a new paradigm so that will take some time.

In short, RSC lets your server generate request-time React trees using things like db connections directly.

8:54 PM · Oct 17, 2023 · **5,456** Views

<https://twitter.com/samselikoff/status/1714354334060654673>



Jonas Bandi
@jbandi



React Server Components is rather about RPC than about Server Side Rendering.

That is my new intro, when I try to explain RSC to those grumpy backend devs that prematurely conclude that React is going back to where PHP/ASP/JSP has been 20 years ago.

11:57 AM · Oct 28, 2023 · **474** Views

<https://twitter.com/jbandi/status/1718205427857362965>

Prior Art

Prior Art - Part I:
Data Fetching & Mutations

STAR WARS
RETURN^{OF} THE RPC

... in other modern 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/>

<https://qwik.builder.io/docs/action/>

[https://qwik.builder.io/docs/server\\$/](https://qwik.builder.io/docs/server$/)



Also:

<https://trpc.io/>

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/>

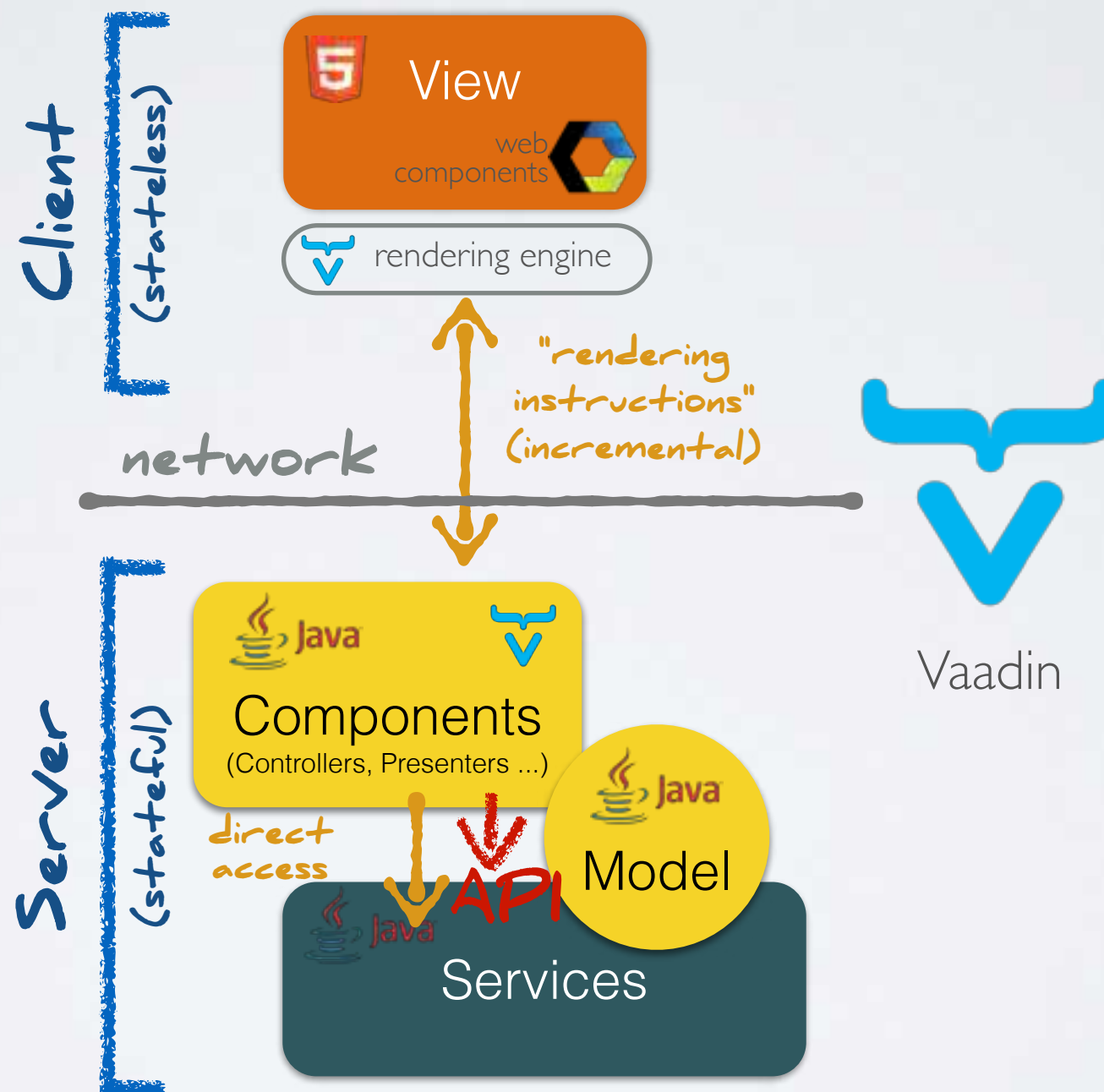
<https://blazor.syncfusion.com/demos/datagrid/overview?theme=bootstrap4>

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

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

Vaadin Architecture

using the browser just as a "rendering engine"



Same concepts are used in .NET with Blazor Server

<https://vaadin.com/>

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/dev-day-rsc>

Questions? Discussions ...



Jonas Bandi

JavaScript / Angular / React / Vue / Vaadin

Schulung / Beratung / Coaching / Reviews

jonas.bandi@ivorycode.com