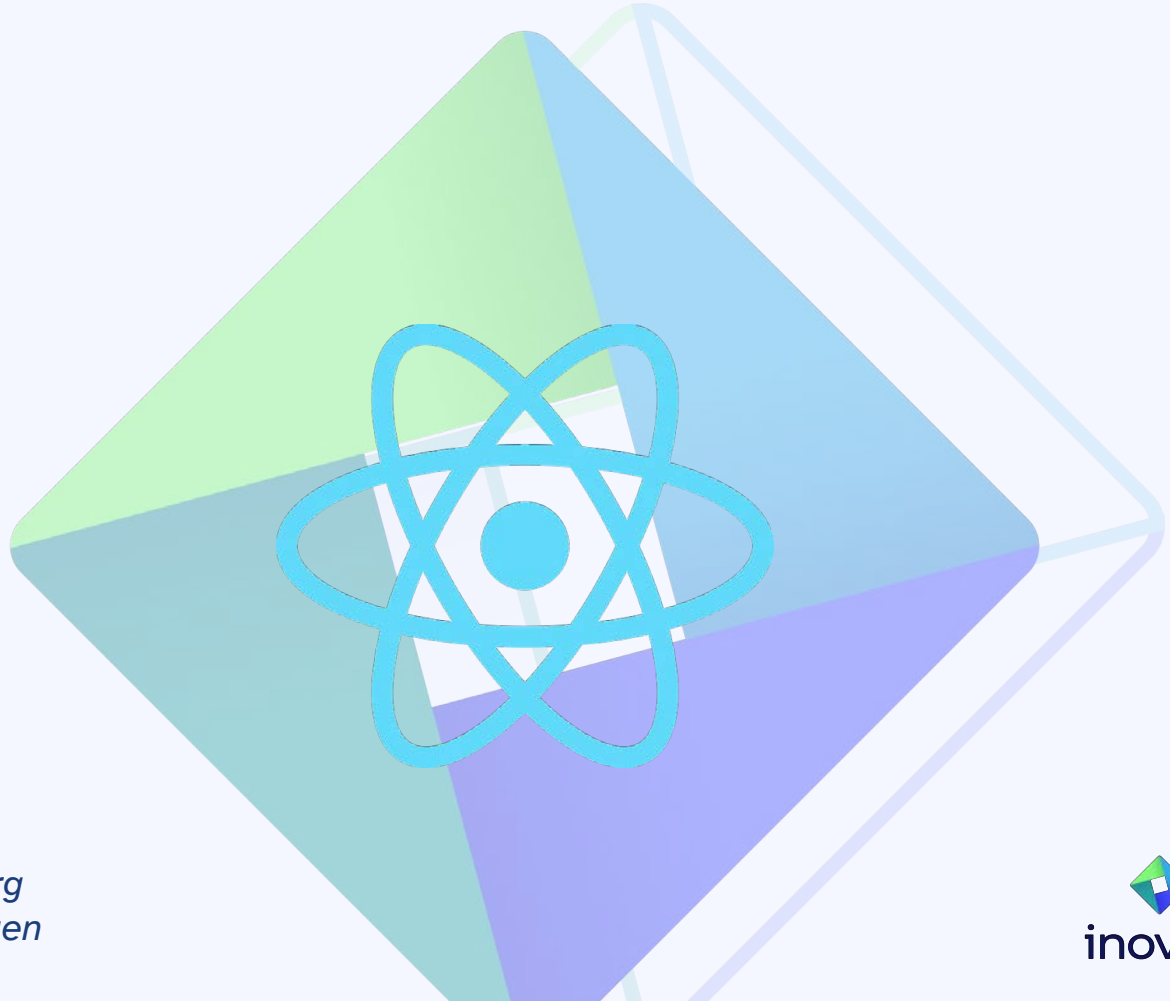
React Roundtrip



Team inovex
Karlsruhe · Köln · München · Hamburg
Berlin · Stuttgart · Pforzheim · Erlangen

inovex

Maximilian Winkelmann - “Max”

Developer at inovex Erlangen

Seneca Camp Orga Team

🚵‍♂️🏃‍♂️🏊‍♂️🛠️

@max-winkelmann

@mxwnk

inovex

A quick look back in time

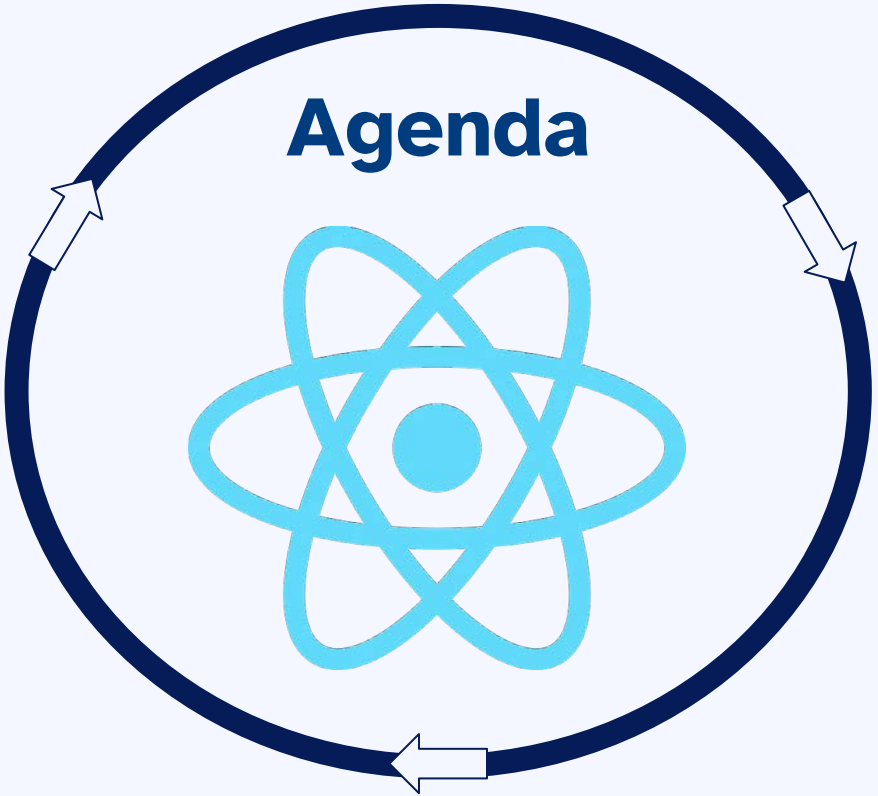
index.php

```php
<?php
$connection = mysql_connect('localhost', 'root', 'password');
mysql_select_db('star_wars_db');

$query = "SELECT title, content FROM posts";
$result = mysql_query($query);

echo "<table border='1'>";
echo "<tr><th>Title</th><th>Content</th></tr>";

while ($row = mysql_fetch_array($result)) {
    echo "<tr><td>" . htmlspecialchars($row['title']) . "</td>";
    echo "<td>" . htmlspecialchars($row['content']) . "</td></tr>";
}

echo "</table>";
mysql_close($connection);
?>
```


inovex

React Roundtrip 

Agenda



React Server Components

Server Side Rendering

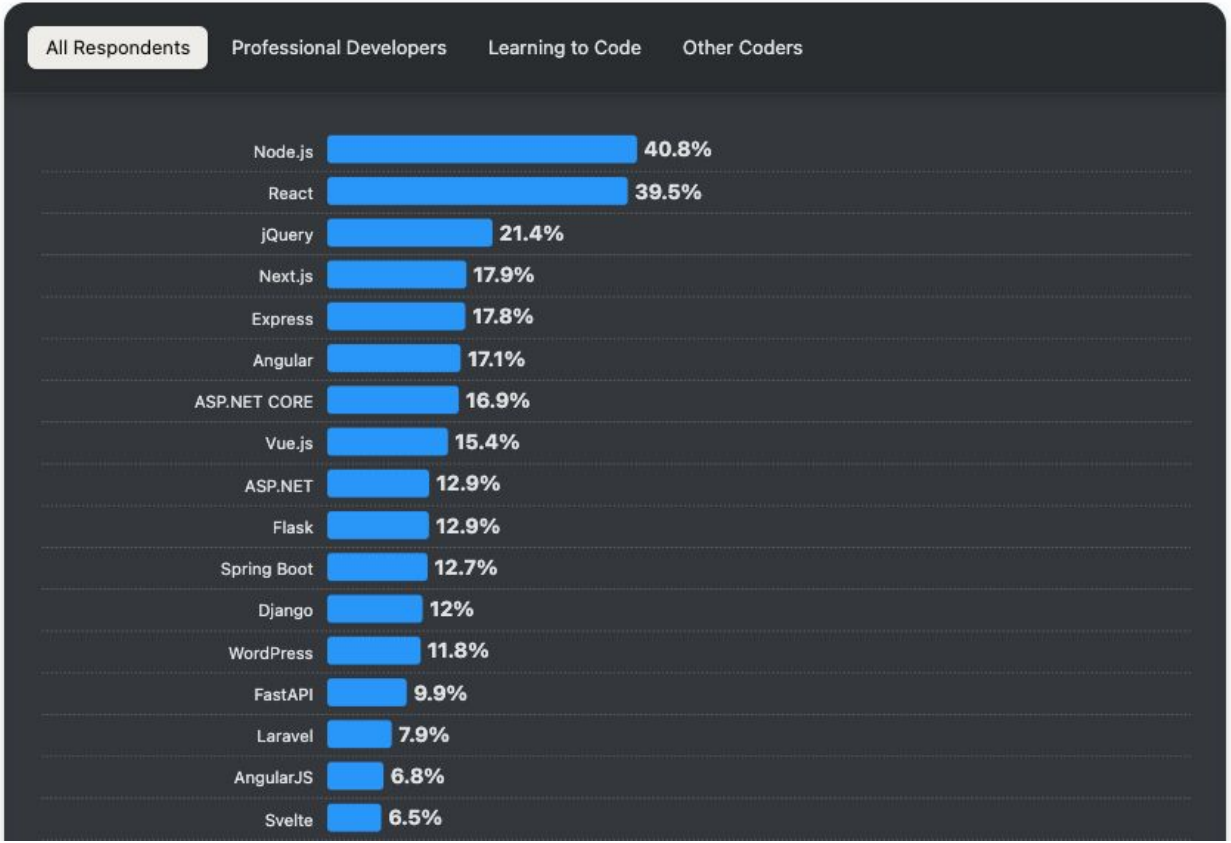
Client Side Rendering

inovex

Web frameworks and technologies

Node.js peaked in 2020 with its highest recorded usage score of 51%. While not as popular, it's still the most used web technology in the survey this year and has increased popularity among those learning to code from last year.

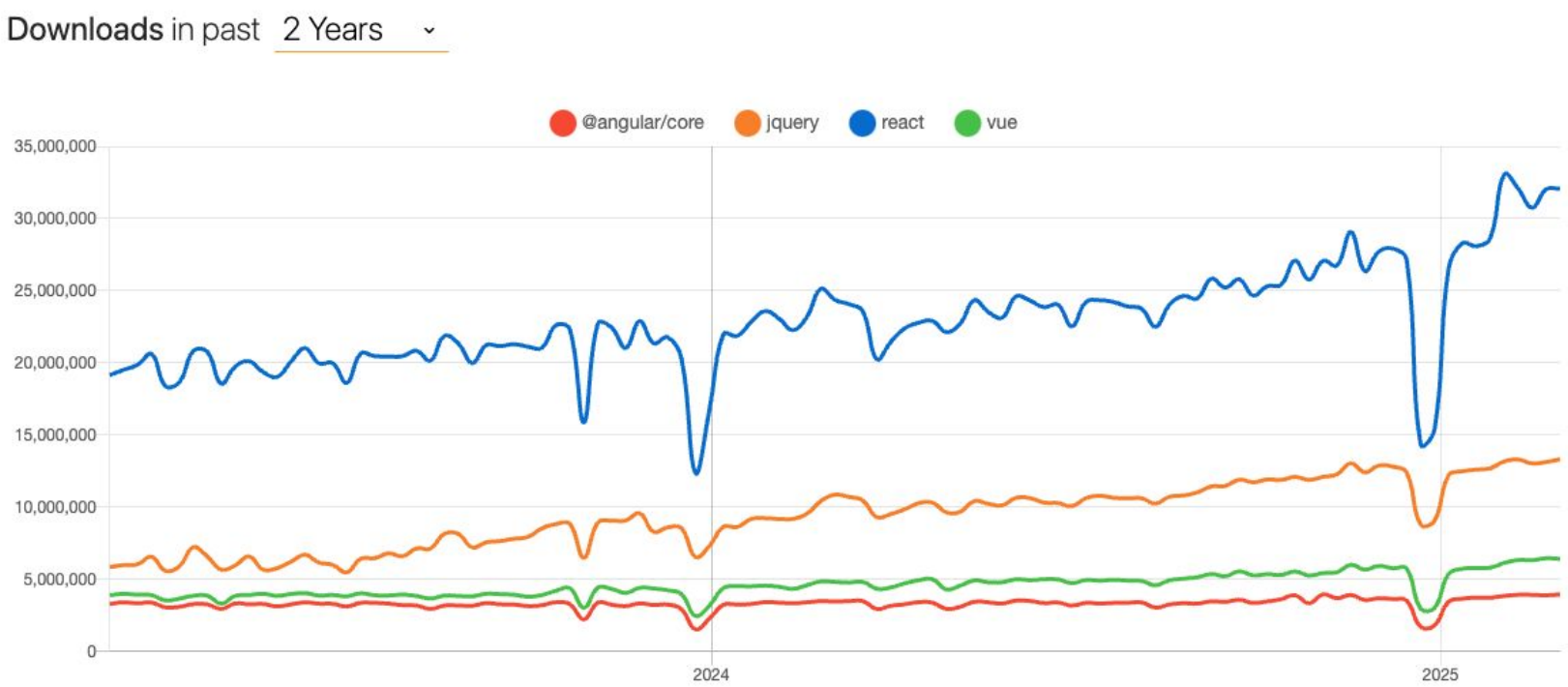
Which **web frameworks and web technologies** have you done extensive development work in over the past year, and which do you want to work in over the next year? (If you both worked with the framework and want to continue to do so, please check both boxes in that row.)



[Source: https://survey.stackoverflow.co/2024/technology#1-web-frameworks-and-technologies]

inovex

NPM Trends

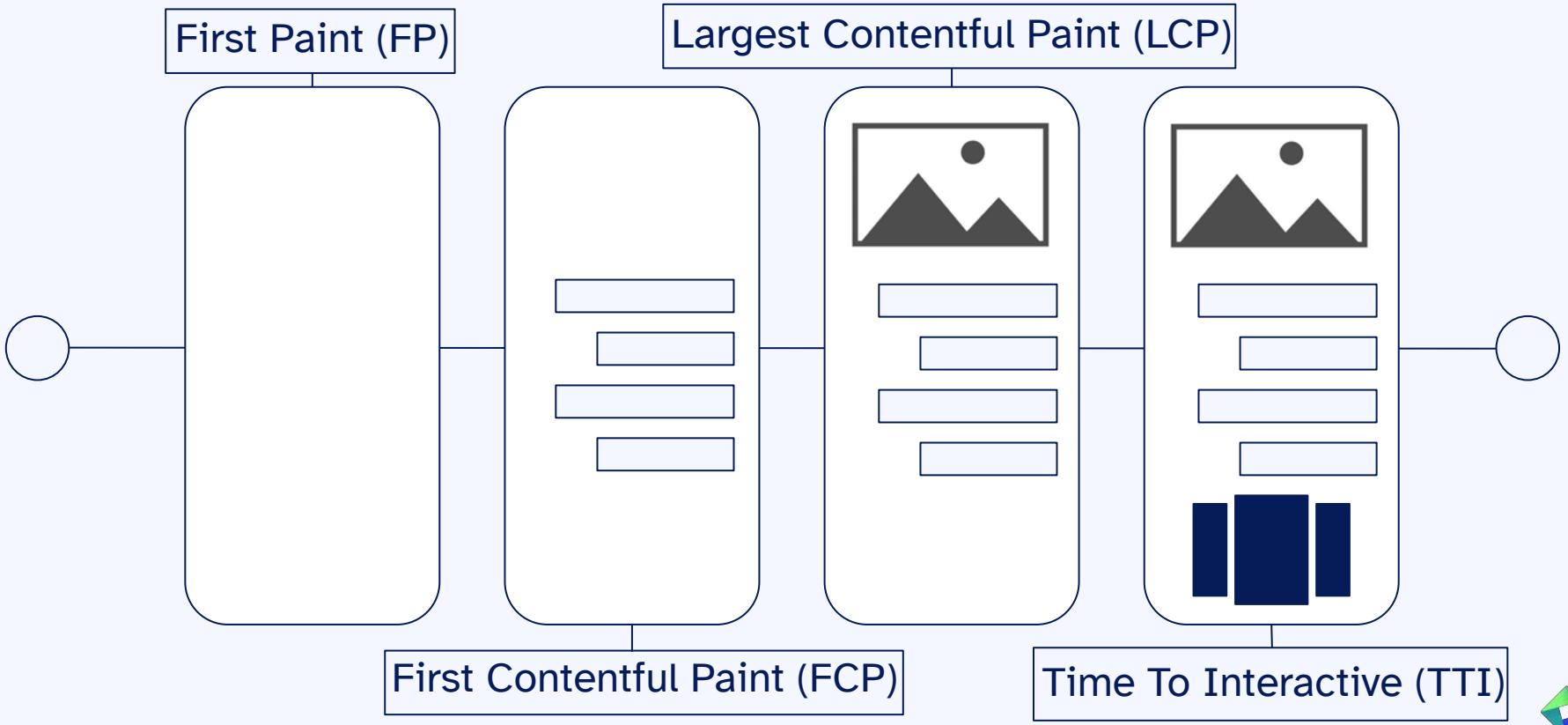


[Source: https://npmtrends.com/@angular/core-vs-jquery-vs-react-vs-vue]

inovex

Basics

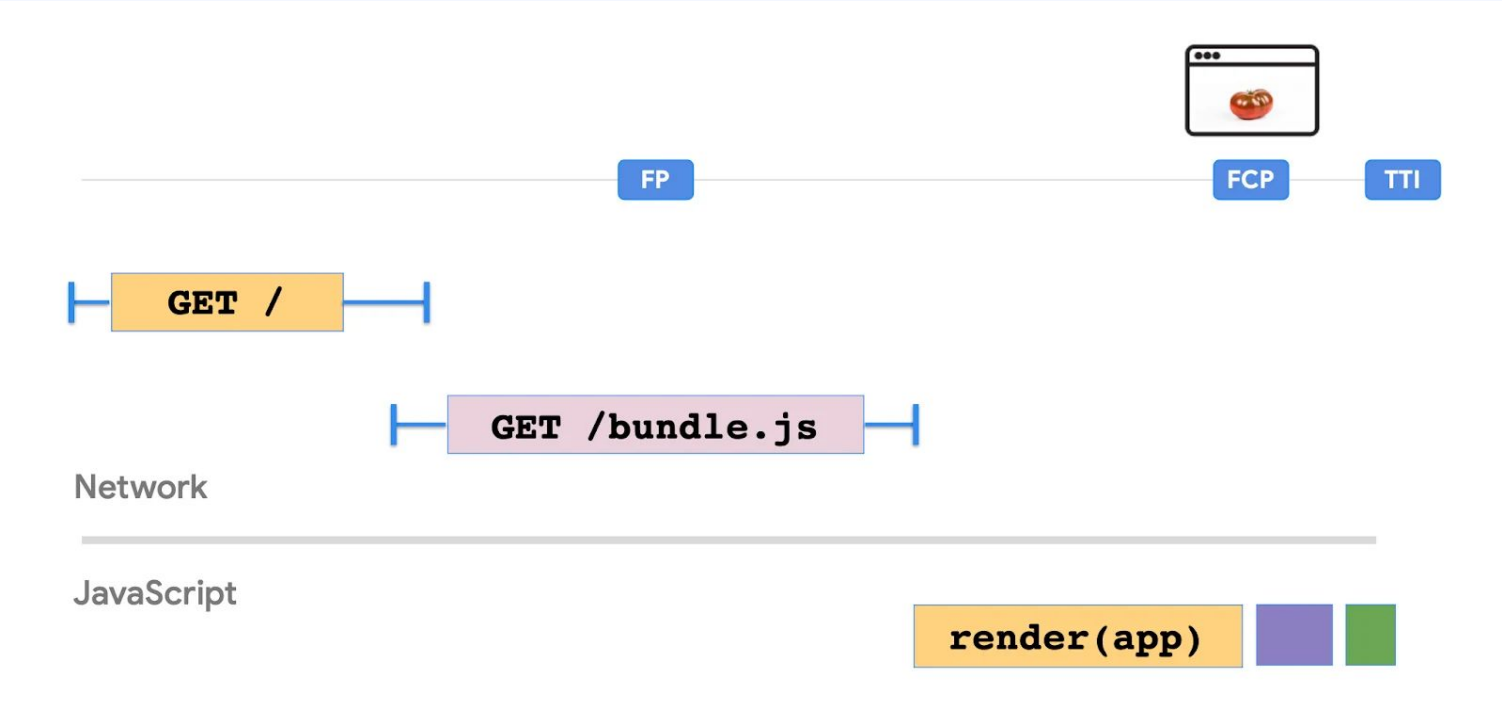
Recap: Page load lifecycle



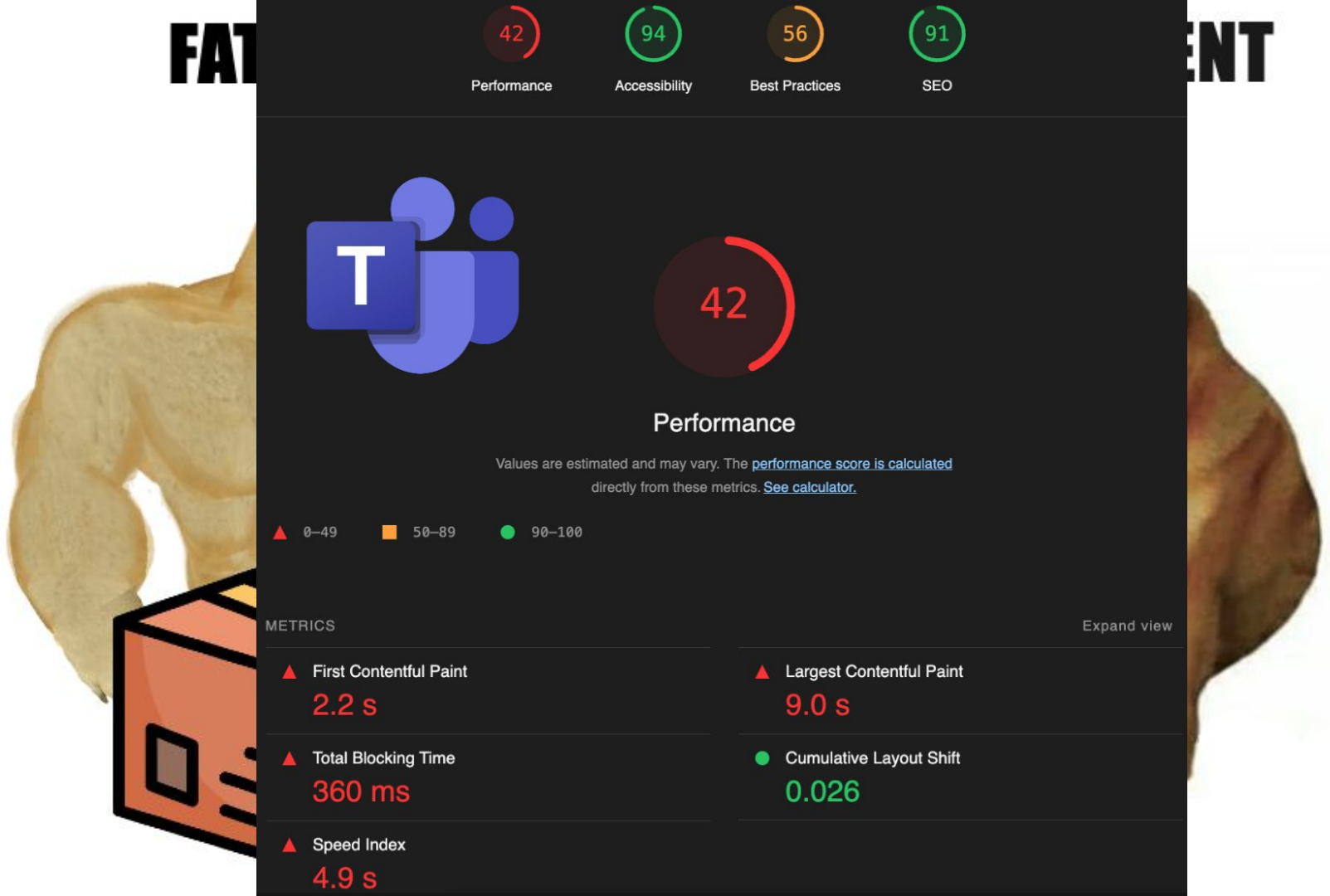
inovex

Client Side Rendering (CSR)

Client Side Rendering (CSR)



inovex


42
Performance
94
Accessibility
56
Best Practices
91
SEO
42
Performance
Values are estimated and may vary. The performance score is calculated directly from these metrics. See calculator.
0–49
50–89
90–100
METRICS
Expand view
First Contentful Paint
2.2 s
Largest Contentful Paint
9.0 s
Total Blocking Time
360 ms
Cumulative Layout Shift
0.026
Speed Index
4.9 s

inovex

Let’s jump in code

inovex

CSR - Pros and Cons

✅ Great interactive experience

🚫 SEO disadvantages

🚫 Slow First Contentful Paint (FCP)

🚫 Slow First Paint (FP)

🛠️ Budgeting JavaScript

🛠️ Pre loading and lazy loading

Server Side Rendering (SSR)

Server Side Rendering (SSR)

0.4.0 (July 17, 2013)

React

- Switch from using `id` attribute to `data-reactid` to track DOM nodes. This allows you to integrate with other JS and CSS libraries more easily.
- Support for more DOM elements and attributes (e.g., `<canvas>`)
- Improved server-side rendering APIs. `React.renderComponentToString(<component>, callback)` allows you to use React on the server and generate markup which can be sent down to the browser.
- `prop` improvements: validation and default values. Read our blog post for details...
- Support for the `key` prop, which allows for finer control over reconciliation. Read the docs for details...
- Removed `React.autoBind` . Read our blog post for details...
- Improvements to forms. We've written wrappers around `<input>` , `<textarea>` , `<option>` , and `<select>` in order to standardize many inconsistencies in browser implementations. This includes support for `defaultValue` , and improved implementation of the `onChange` event, and circuit completion. Read the docs for details...
- We've implemented an improved synthetic event system that conforms to the W3C spec.
- Updates to your component are batched now, which may result in a significantly faster re-render of components. `this.setState` now takes an optional callback as it's second parameter. If you were using `onClick={this.setState.bind(this, state)}` previously, you'll want to make sure you add a third parameter so that the event is not treated as the callback.

[Source: https://github.com/facebook/react/blob/main/CHANGELOG.md#040-july-17-2013]

Server Side Rendering

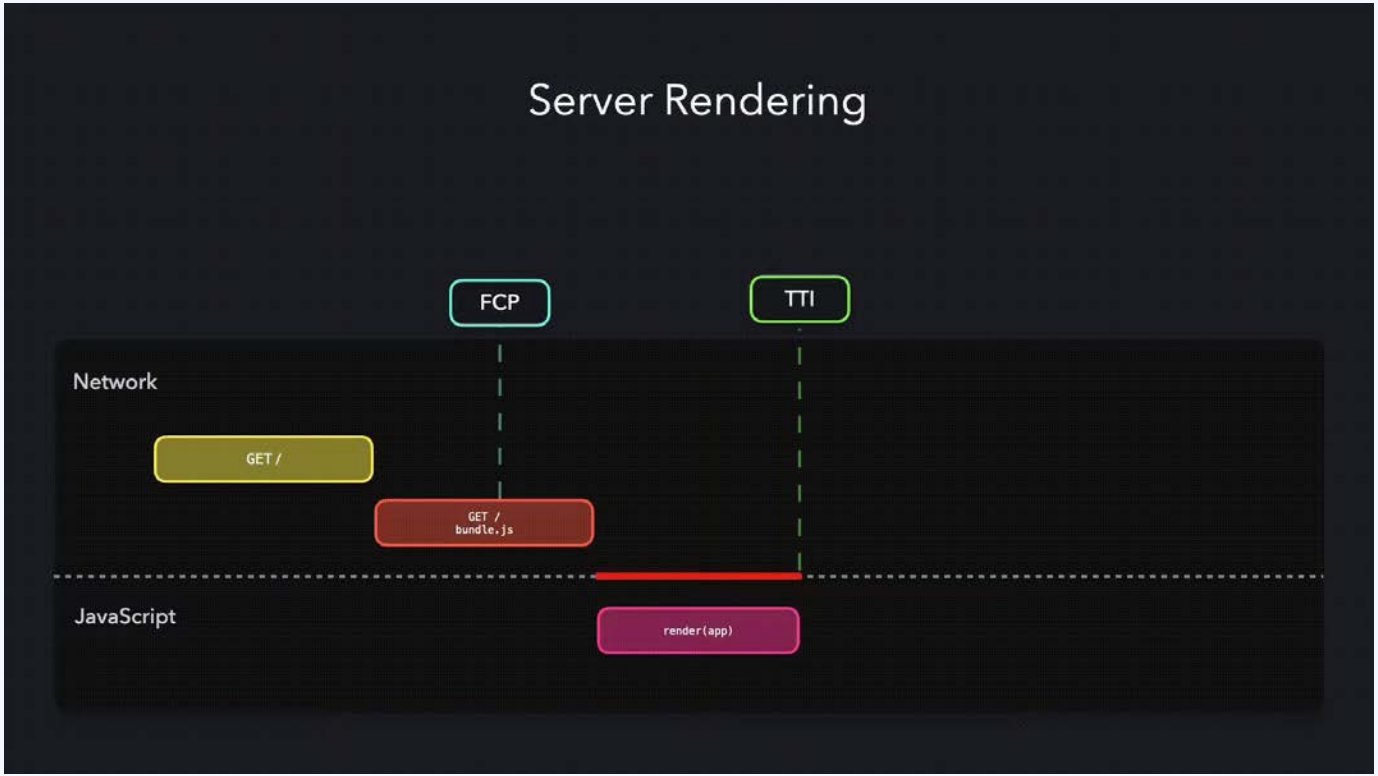
```
// Server Code
import { renderToString } from "react-dom/server";
const html = renderToString(<App />);

// Client Code
import { hydrateRoot } from 'react-dom/client';
const domNode = document.getElementById('root');
const root = hydrateRoot(domNode, reactNode);
```


inovex

Server Side Rendering

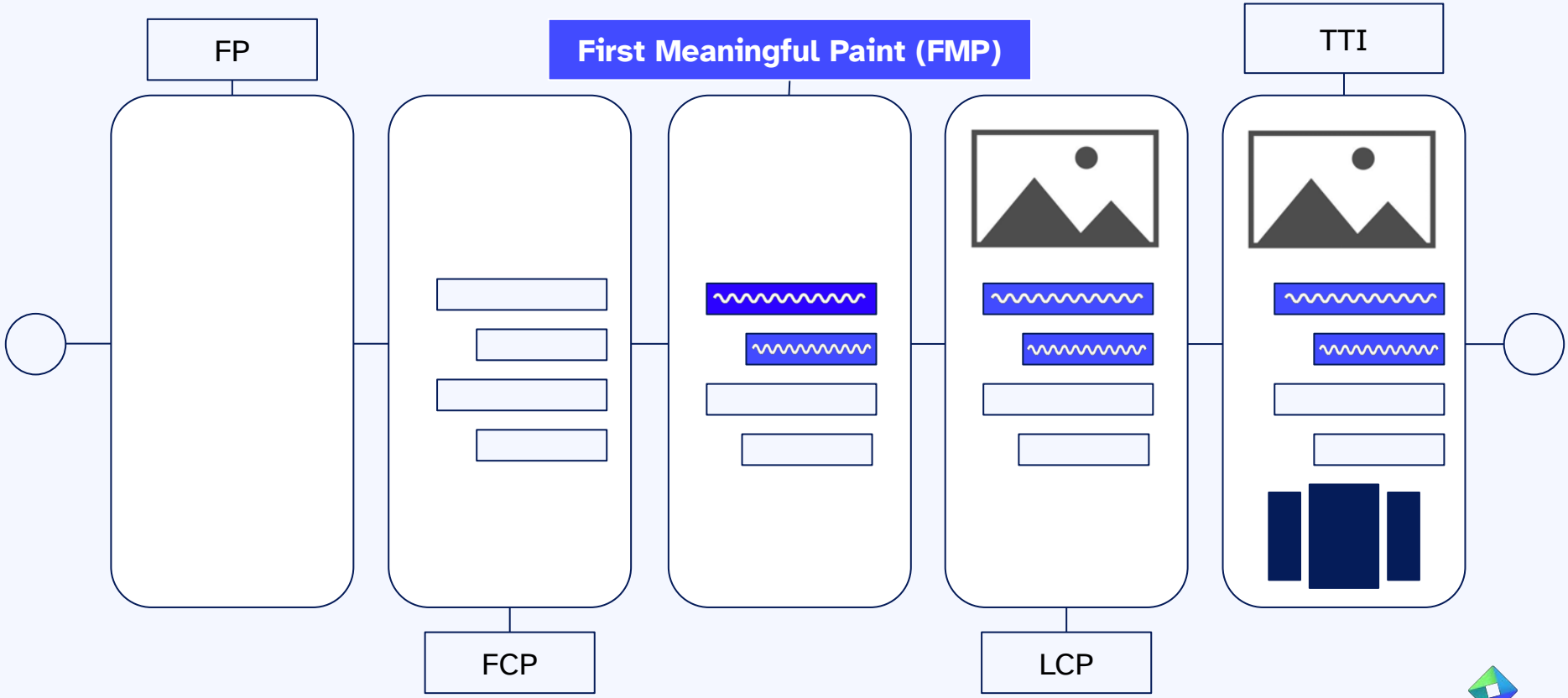
inovex

Improvement: Streaming



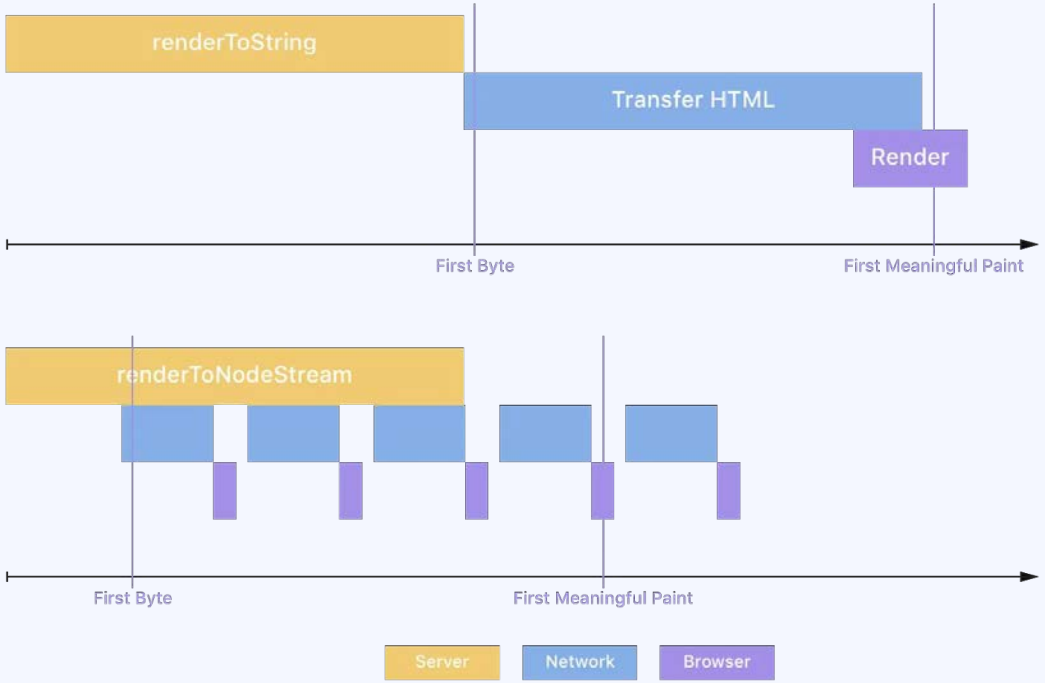
inovex

Page load lifecycle: First Meaningful Paint (FMP)



inovex

Comparison: Streaming vs Static



[Source: https://mxstbr.com/thoughts/streaming-ssr/]

inovex

Let’s jump in code

inovex

SSR - Pros and Cons

✅ Faster initial Load (FP + FCP)

✅ Improved SEO

✅ Better performance on low-power devices

🚫 Interactivity can be delayed

🚫 Time to First Byte (TTFB) can be slow
-> page needs to be generated on-demand
-> Static Site Generation (SSG)

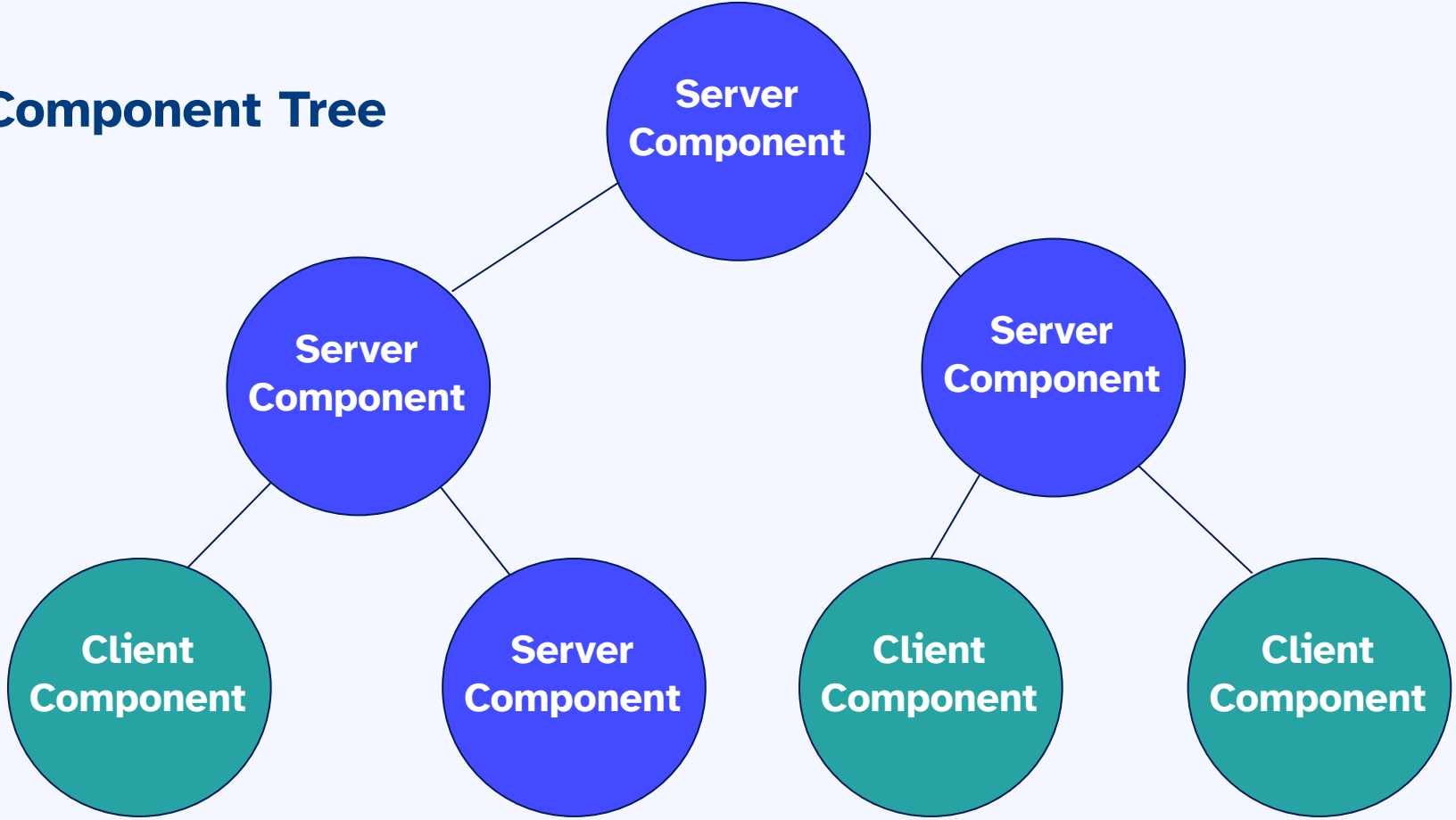
React Server Components (RSC)

React Server Components

React implementation: ***react-server-dom-****

1. Server: render component tree
2. Server: execute code for all server components
3. Server: generates node stream
4. Send node stream with containing JSON

Component Tree



inovex

JSX Tree

```
{
  $$typeof: Symbol.for("react.element"),
  type: 'html',
  props: {
    children: [
      {
        $$typeof: Symbol.for("react.element"),
        type: 'head',
        props: {
          children: {
            $$typeof: Symbol.for("react.element"),
            type: 'title',
            props: { children: 'My blog' }
          }
        }
      }
    ]
  }
}
```


inovex

Let’s jump in code

inovex

RSC - Pros and Cons

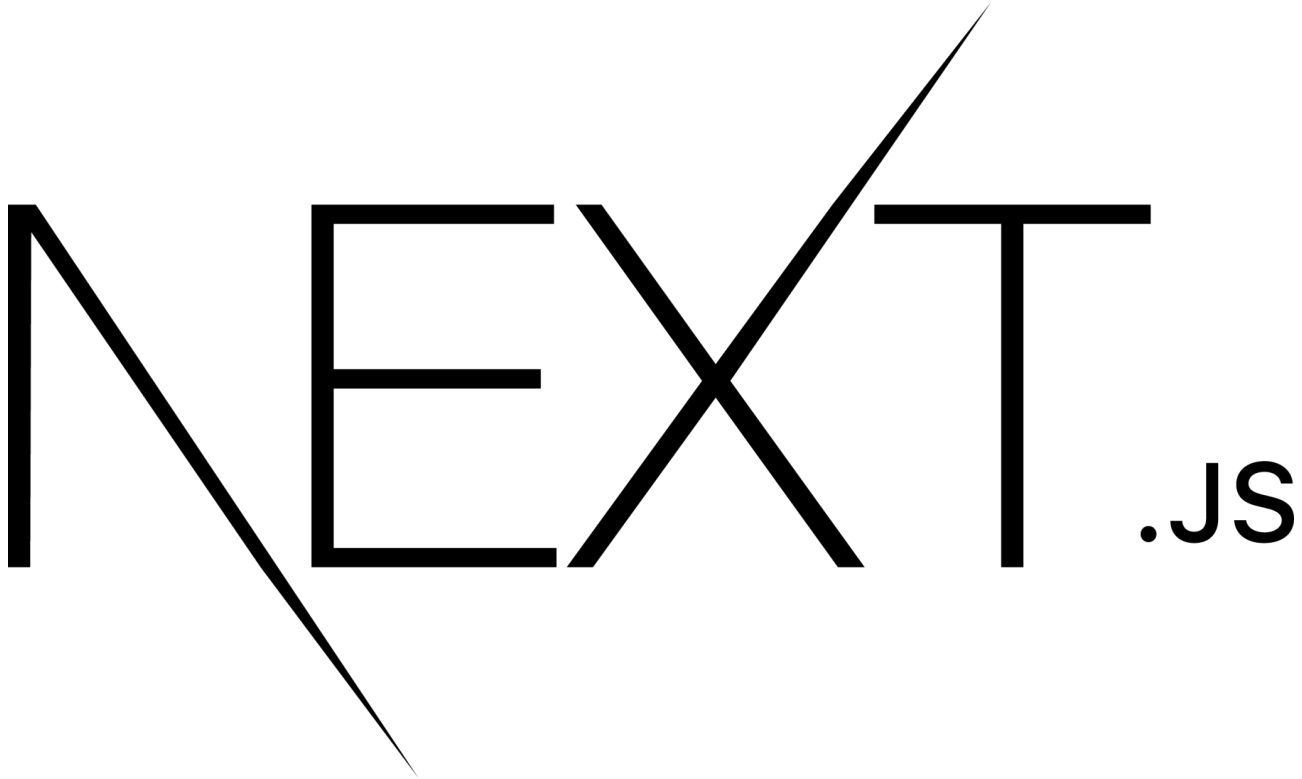
✅ Great interactive experience, combined benefits of SSR & CSR

✅ One single code base - full stack application

⚠️ Tradeoff: adds more complexity - Server vs. Client

⚠️ Separation of concerns - layered architecture

Frameworks



inovex

Let’s jump in code

inovex

Vielen Dank!

Maximilian Winkelmann
Software Developer

maximilian.winkelmann@inovex.de

Allee am Röthelheimpark 11
91052 Erlangen

@max-winkelmann

@mxwnk

inovex

Schön, dass du hier bist – bleib in Kontakt!

We’re hiring!

Podcast

Überall, wo es Podcasts gibt: Spotify, Overcast, Pocket Casts, ...

Soziale Medien

Insta: @inovexlife

LinkedIn: @inovex GmbH

inovex