Frontend Tech Trends

August 11th 2021 | Fundamental Conference

Tobias Ouwejan

Principal Frontend Architect

@Spryker

Frontend Tech Trends

August 11th 2021 | Fundamental Conference

The problems I try to solve:

1. Web Application **Performance**
   Core Web Vitals, build time
2. Framework **Agnostic** Components
   Reusability, Adoption

How we got here?

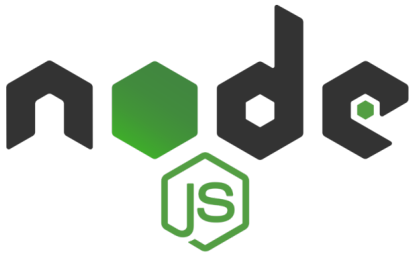
2000 2005 2010 2015 2020

The web platform
gets more mature and
complex at the same time.

How we got here?

The ecosystem *gets ready* for open source…

2000 2005 2010 2015 2020



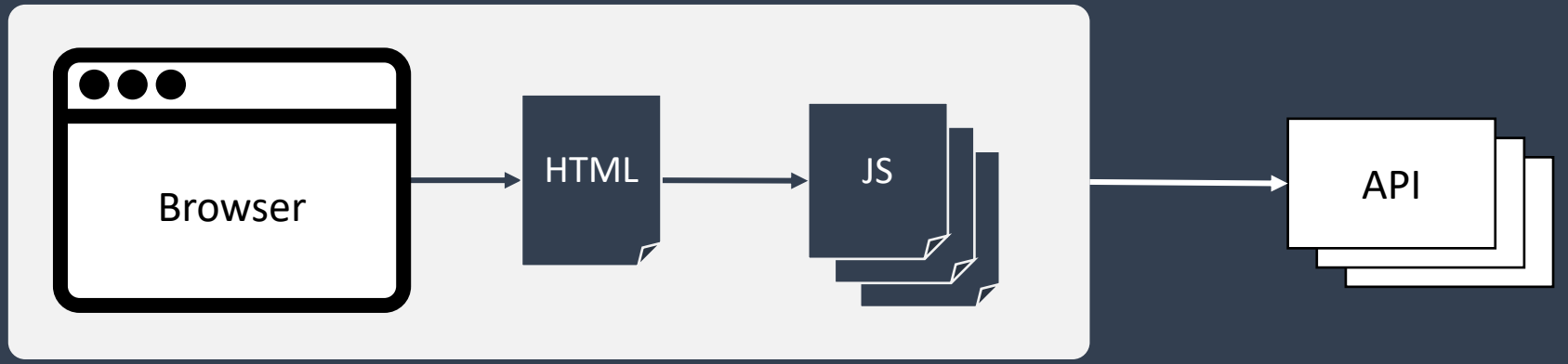
How we got here?

NPM explodes 🔥🔥🔥

The era of building reactive components for Single Page Applications

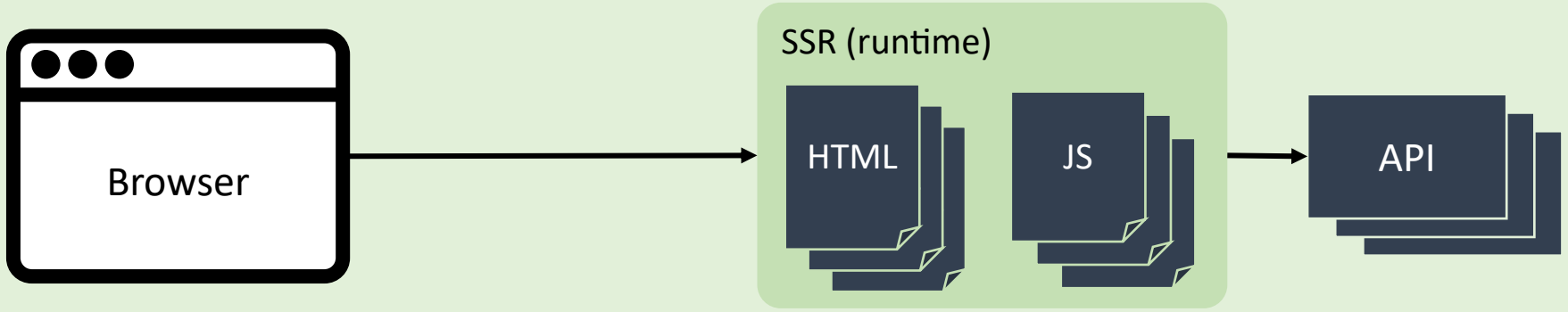
To much JavaScript leaks into the browser

The browser became the runtime to compile the DOM of the complete application



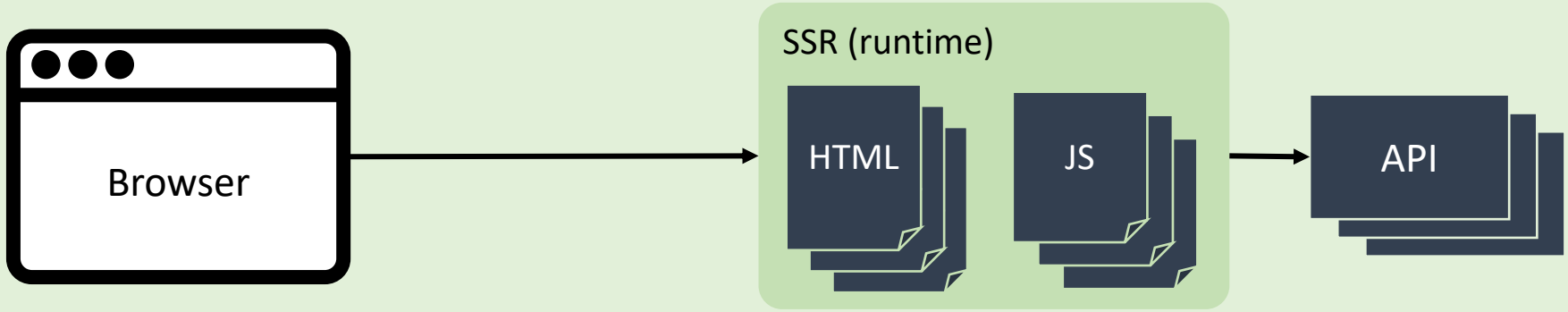
Static pages are back

Server Side
Rendering (SSR)

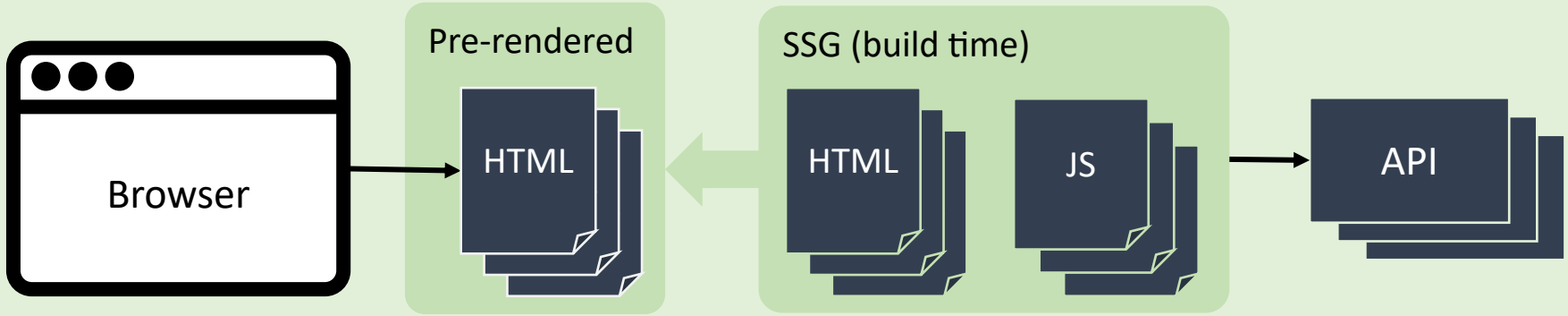


Static pages are back

Server Side Rendering (SSR)



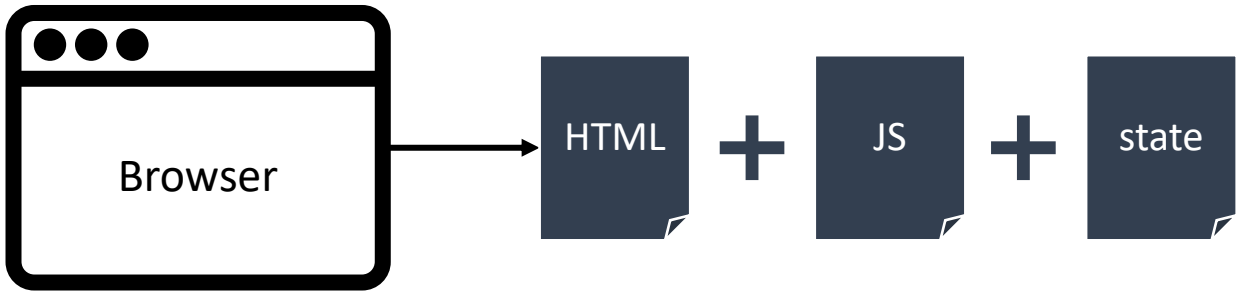
Static Site Generator (SSG)



😮 333 Generators listed at https://jamstack.org/generators/

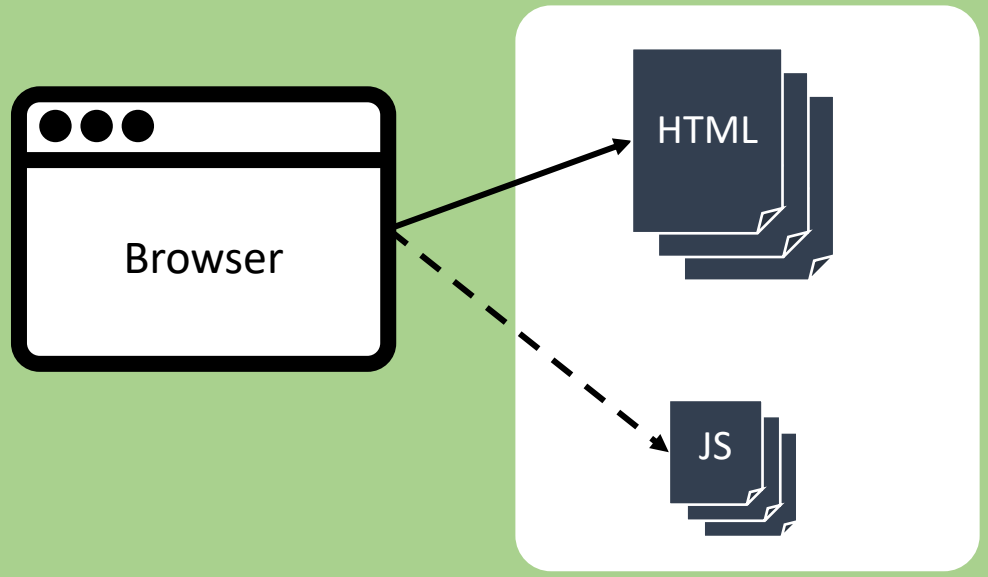
Problems with Static generated pages

The JavaScript framework is bundled and shipped with the static page to “rehydrate” the page so that it can handle events.



Partial Hydration

- Static HTML with sprinkles of JavaScript
- AKA *progressive hydration* or *Islands Architecture*
- Not supported by large JS frameworks
- SPA is unlikely



Partial Hydration

astro

- Static Site Generator
- 0kb client side JS
- opt-in to JS when needed
- Multi-framework support
- Founder: Fred K. Schott
- https://astro.build

- Static Site Generator
- 0kb client side JS
- Svelte only
- https://github.com/Elderjs/elderjs

Qwik

- DOM centric JS framework
- Multi-framework support
- Core team member: Misko Hevery
- https://github.com/BuilderIO/qwik

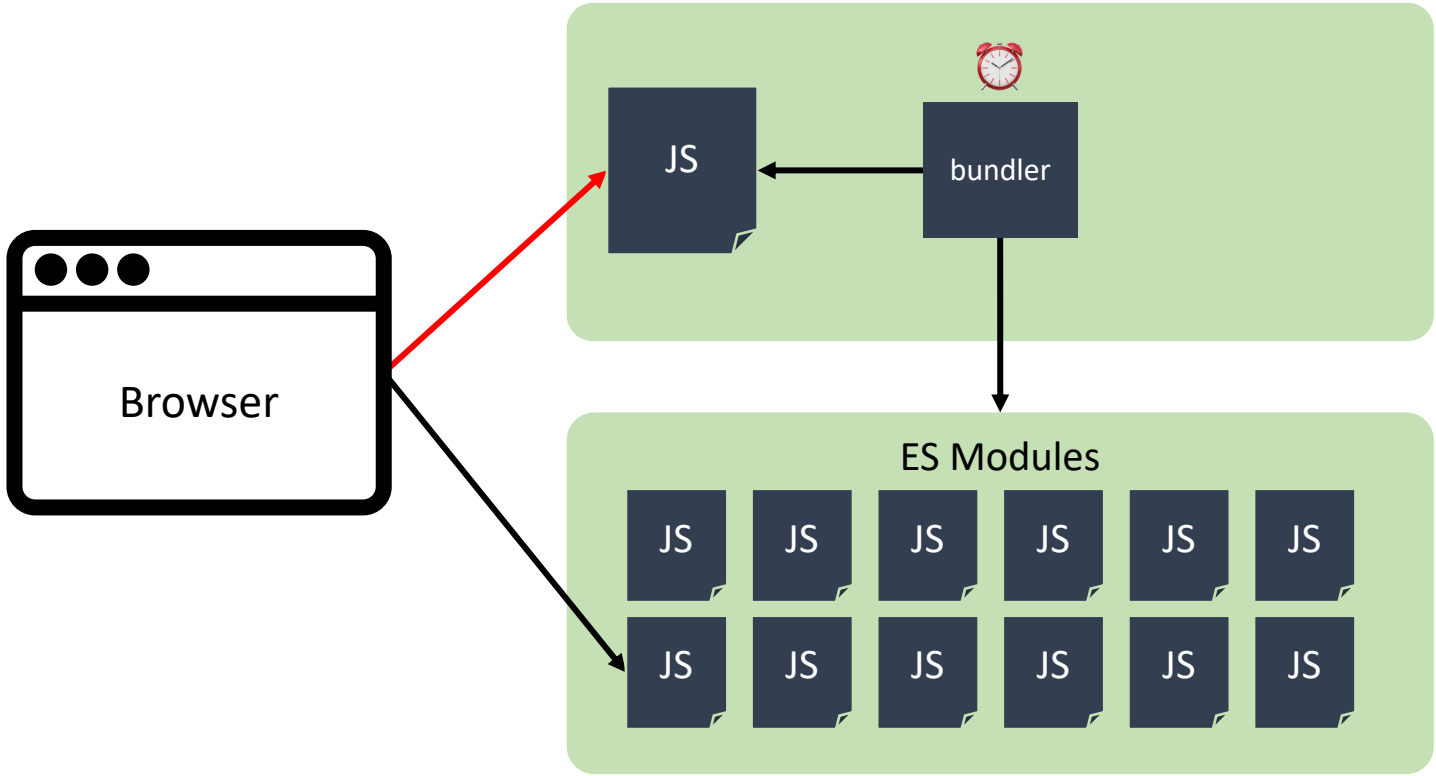
Slow build times and updates of application bundles

Especially when your app grows

New Generation of Build Tools

Using ES Modules directly

- Native ESM is now widely supported
- ES module based development workflow gives 10-100x faster build times
- Production builds still requires bundling
- ImportMaps is on its way to resolve module locations dynamically



New Generation of Build Tools

Using ES Modules directly

esbuild

Snowpack

Vite

wmr

Comparison: https://css-tricks.com/comparing-the-new-generation-of-build-tools/

Modern Web

“Going Buildless”

https://modern-web.dev/

Frameworks don’t really play well together

Can we avoid the vendor lock-in?

Framework specific components

Framework lock-in

- (3rd party) DEVs are forced to use an opiniated framework
- No way to reuse components cross applications
  (i.e. component design system)

Version lock-in

- Forced to use a single version of the framework
- No way to gradually upgrade components

Web components

- Initial only available by JS 🤦
- Declarative Shadow DOM landed in chrome 🎉
- Polyfill is available for browsers not supporting Shadow DOM
- Other frameworks start to embrace web components (finally…)
- Custom Elements Manifest analyzer describes a component metamodel

astro

LIT
(google, Polymer team)
https://lit.dev

Fast
(Microsoft)
https://fast.design

Open Web Components

https://open-wc.org

Meta frameworks support multi-framework

Compile pages from various components

Works like a static site generator

Supports React, Svelte, Vue, Preact, web components, plain HTML + JavaScript or Bring Your Own Framework (BYOF)

https://astro.build

MITOSIS

Build a components meta model and generates framework specific components out of it

Supports React, Svelte, Vue, Angular or Solidjs

https://github.com/BuilderIO/mitosis

Key take aways

- Single Page Application experiences are not for everyone
- Astro ticks all the boxes
- A lot of traction around web components