Micro-frontends Architecture with Nx and Angular

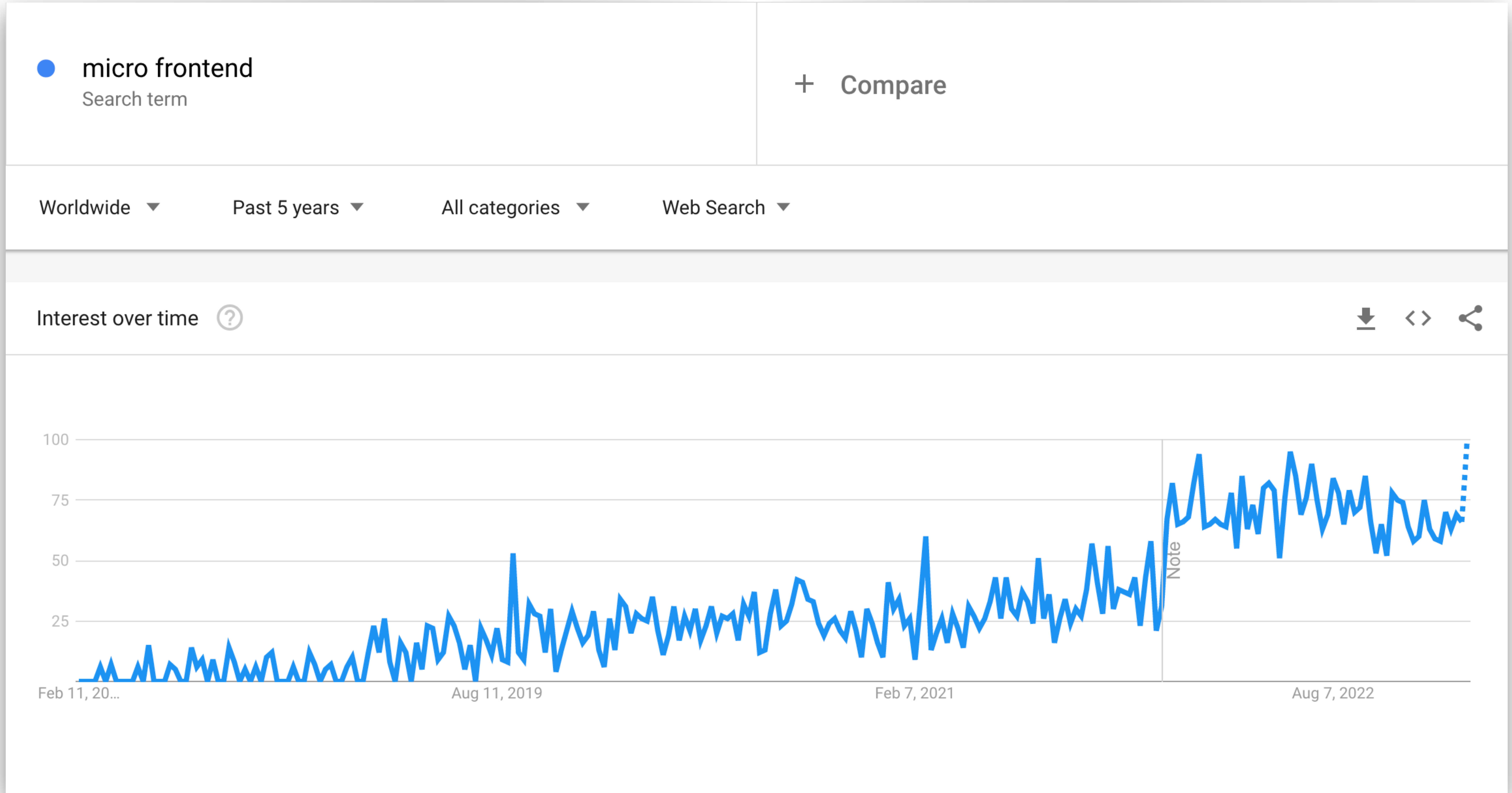
Filip Voska
JavaScript Team Lead

Filip Voska

TEAM LEAD
JAVASCRIPT

01

The journey and the buzz-word

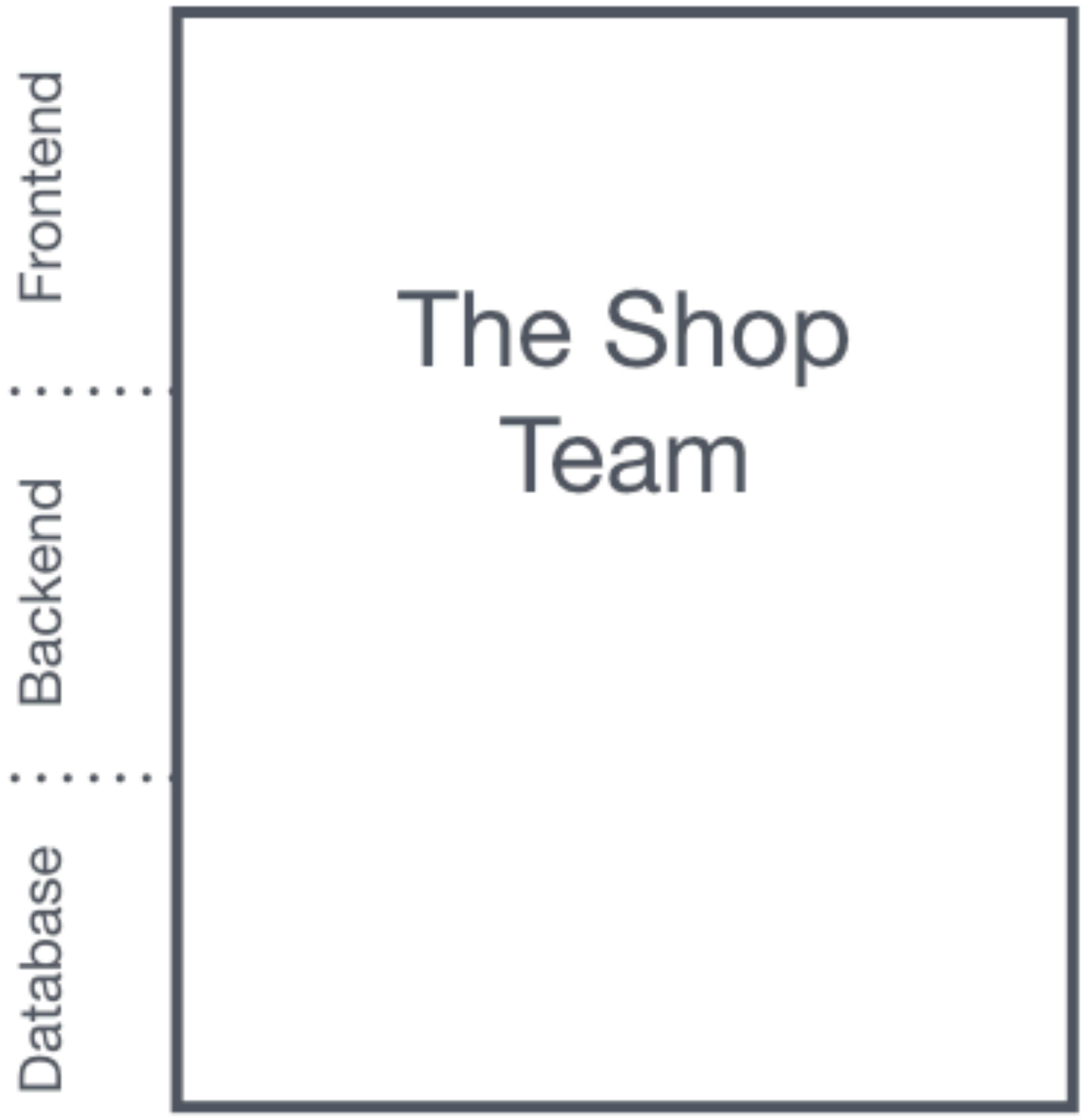


● micro frontend
Search term

Worldwide ▾ Past 5 years ▾ All categories ▾ Web Search ▾

Interest over time

The Monolith



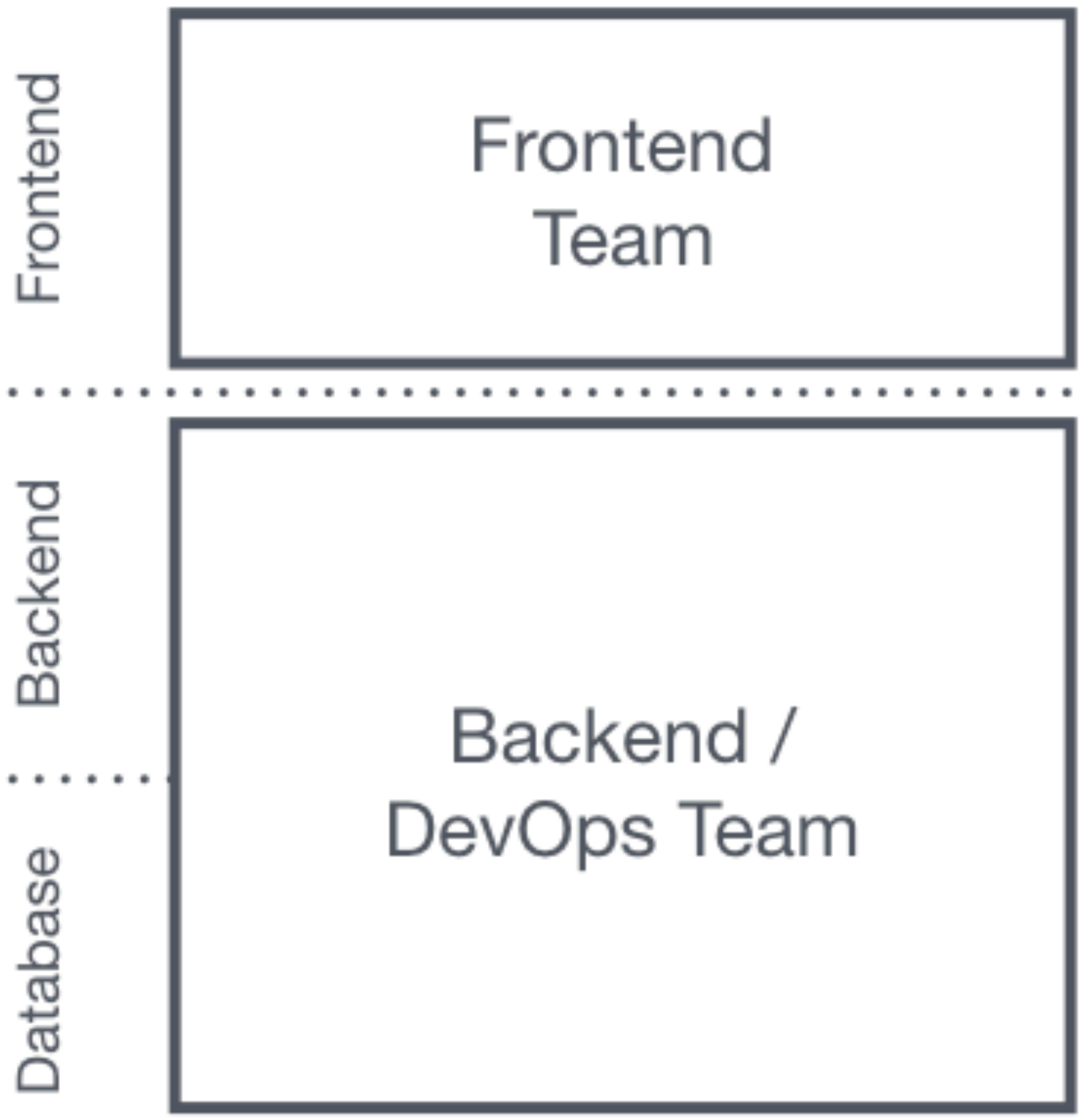
Frontend

Backend

Database

The Shop Team

Front & Back



Frontend

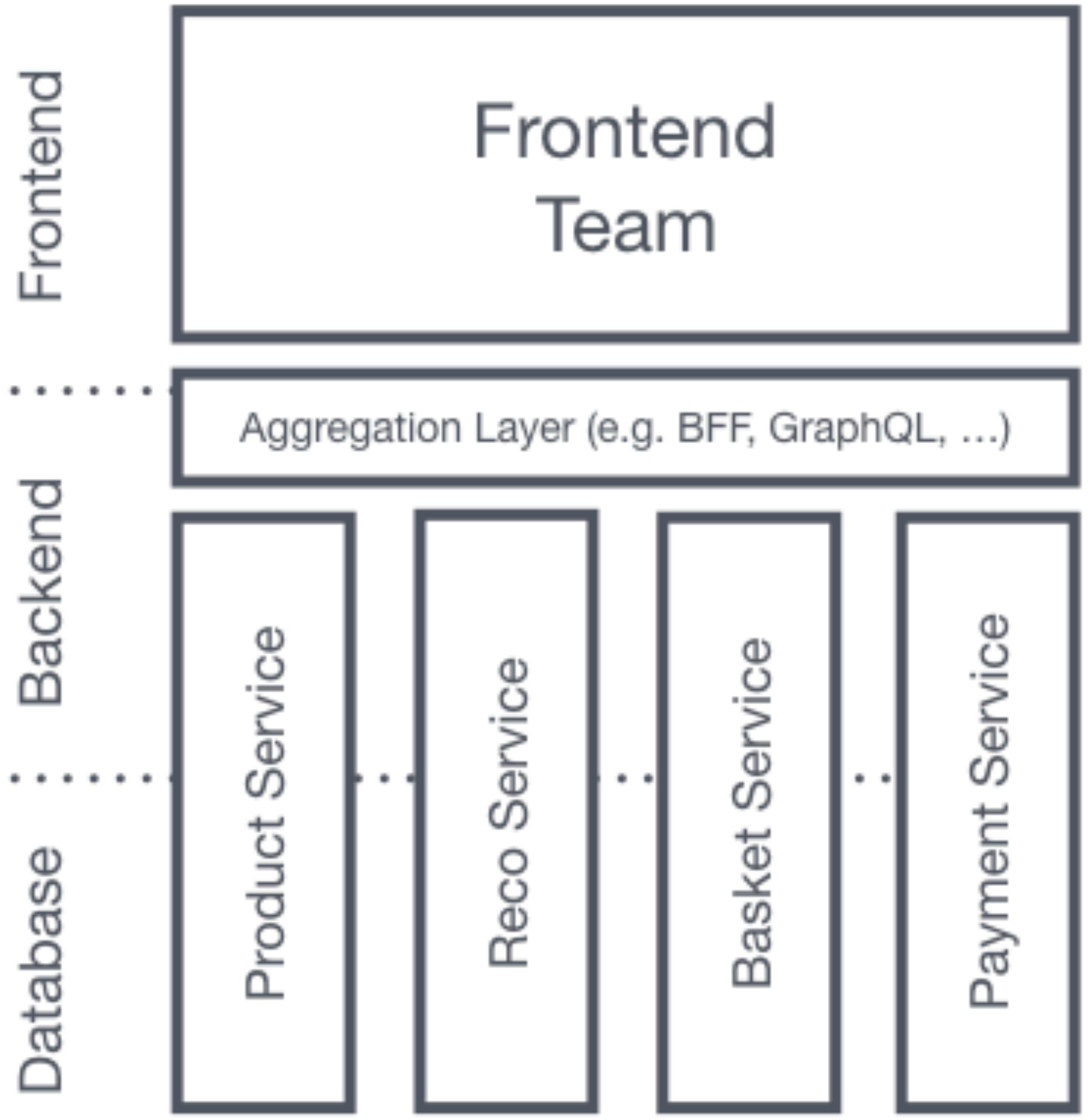
Backend

Database

Frontend Team

Backend / DevOps Team

Microservices



Frontend

Backend

Database

Frontend Team

Aggregation Layer (e.g. BFF, GraphQL, ...)

Product Service

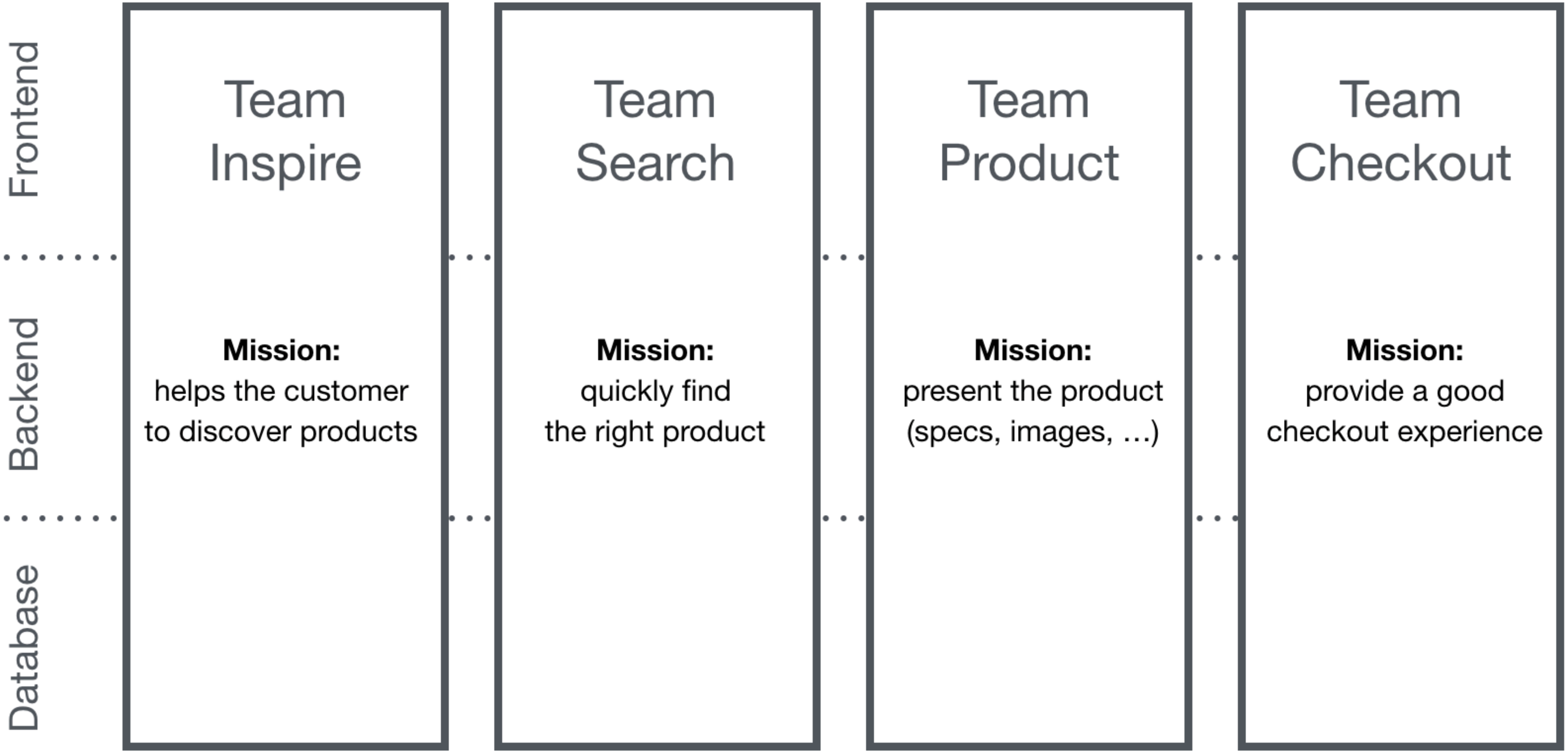
Reco Service

Basket Service

Payment Service

Source: https://micro-frontends.org/

End-to-End Teams with Micro Frontends



Team Inspire

Mission: helps the customer to discover products

Team Search

Mission: quickly find the right product

Team Product

Mission: present the product (specs, images, …)

Team Checkout

Mission: provide a good checkout experience

Source: https://micro-frontends.org/

02 Goals

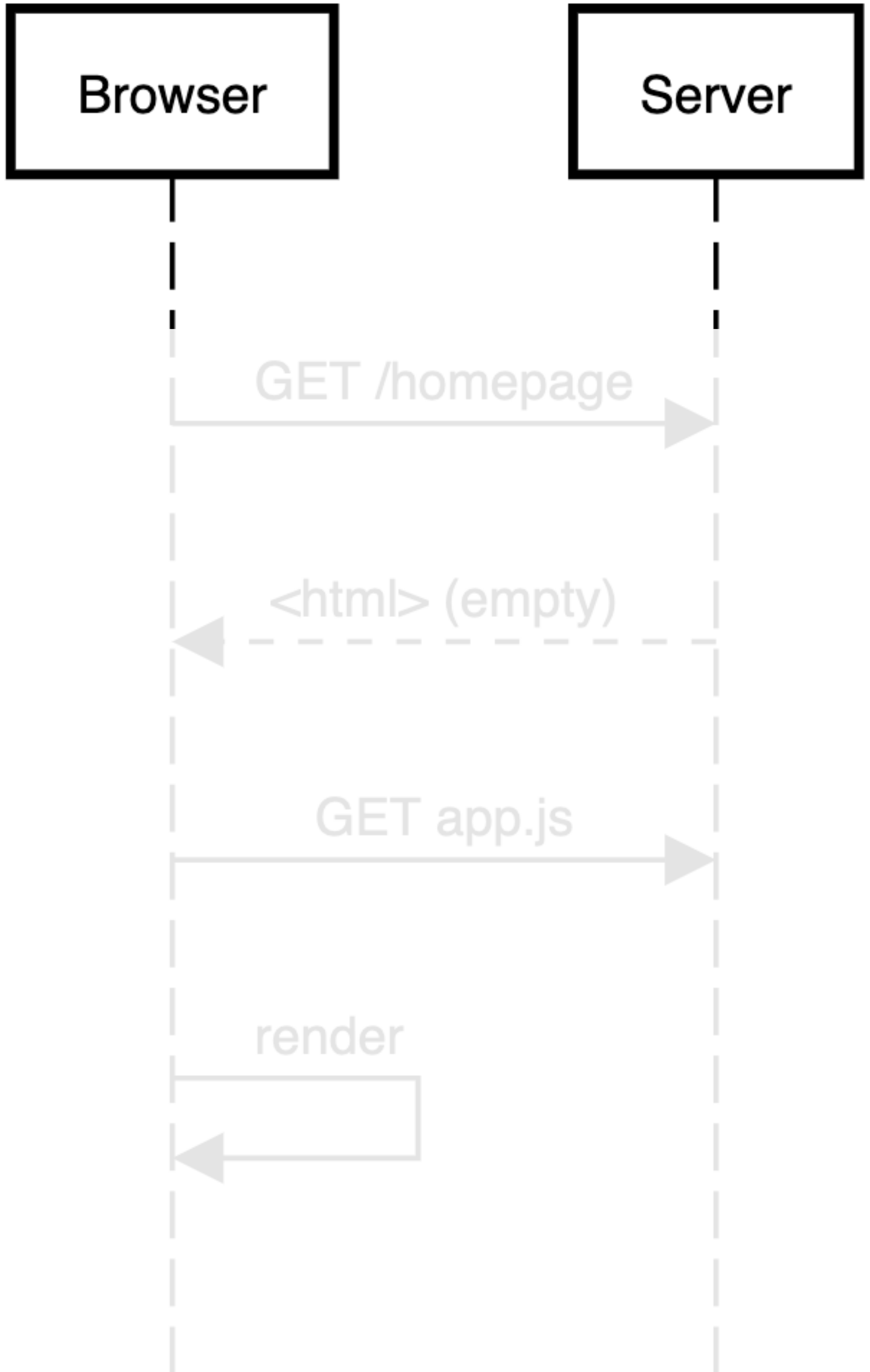
Goals

- Be framework-agnostic
- Independent teams
- Share code and state
- Share coding standards
- Be more resilient
- Follow DDD
- Set clear boundaries
- Faster build times
- Independent deployments

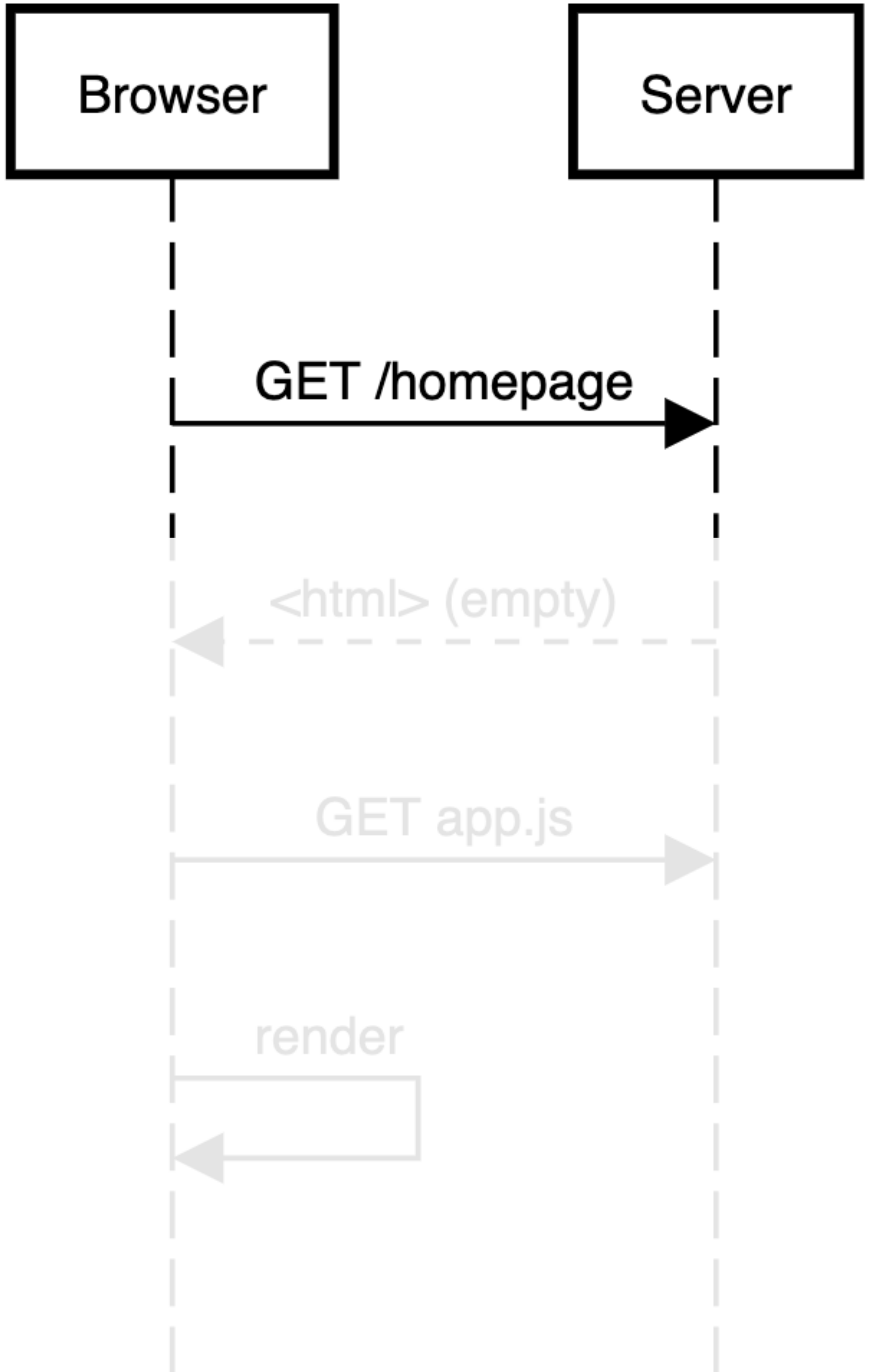
03

How we serve apps

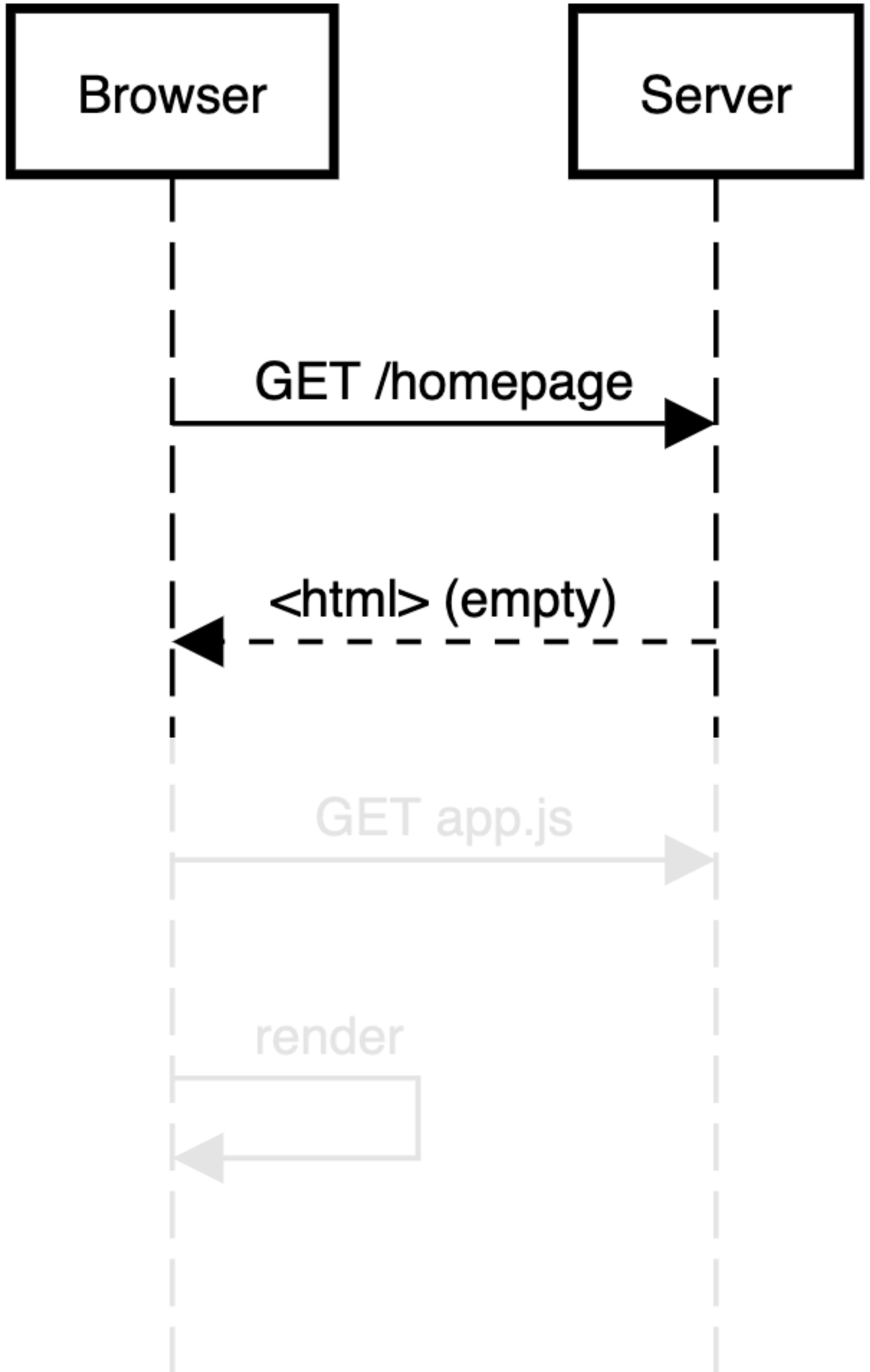
Classic SPA App



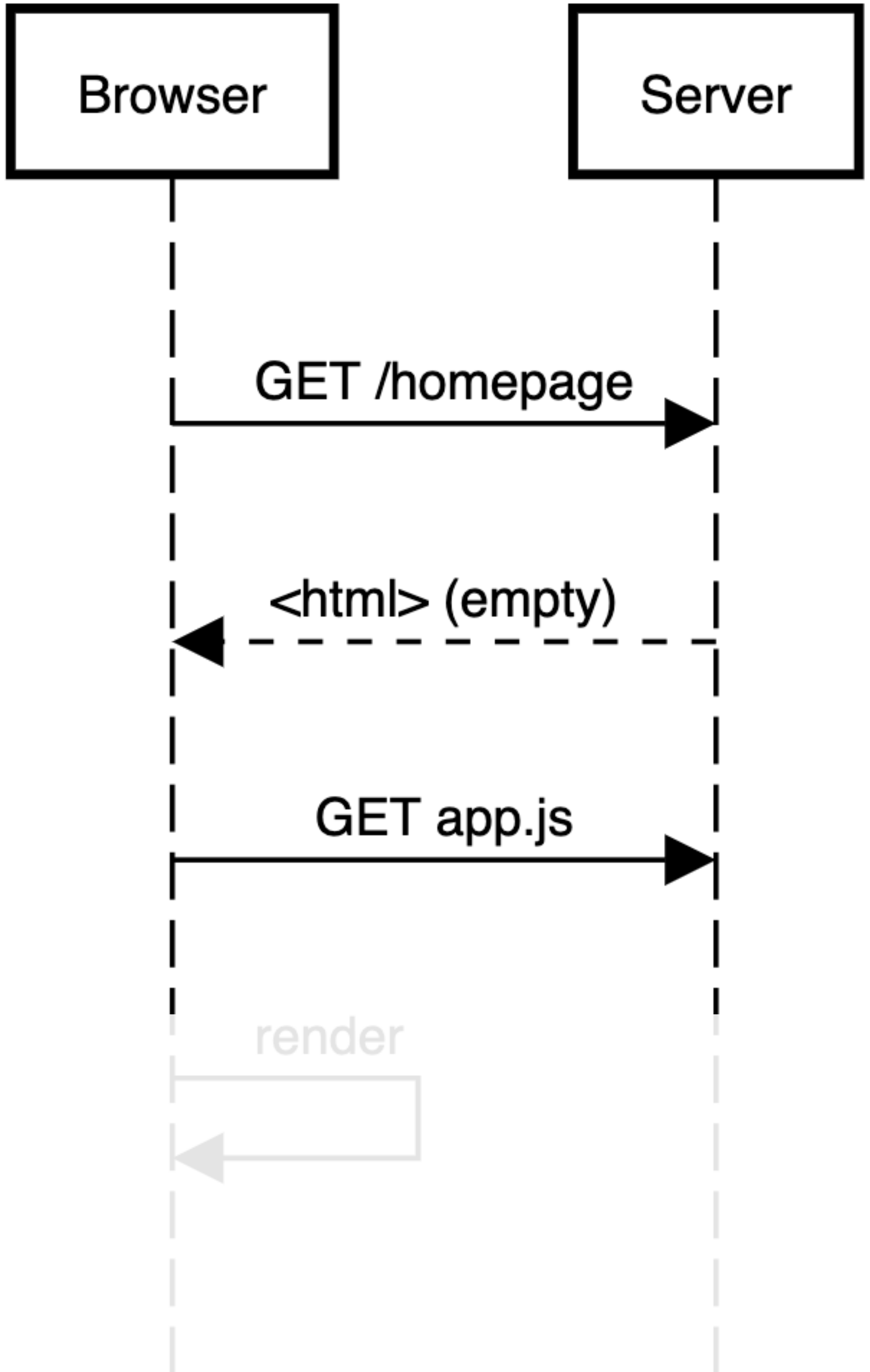
Classic SPA App



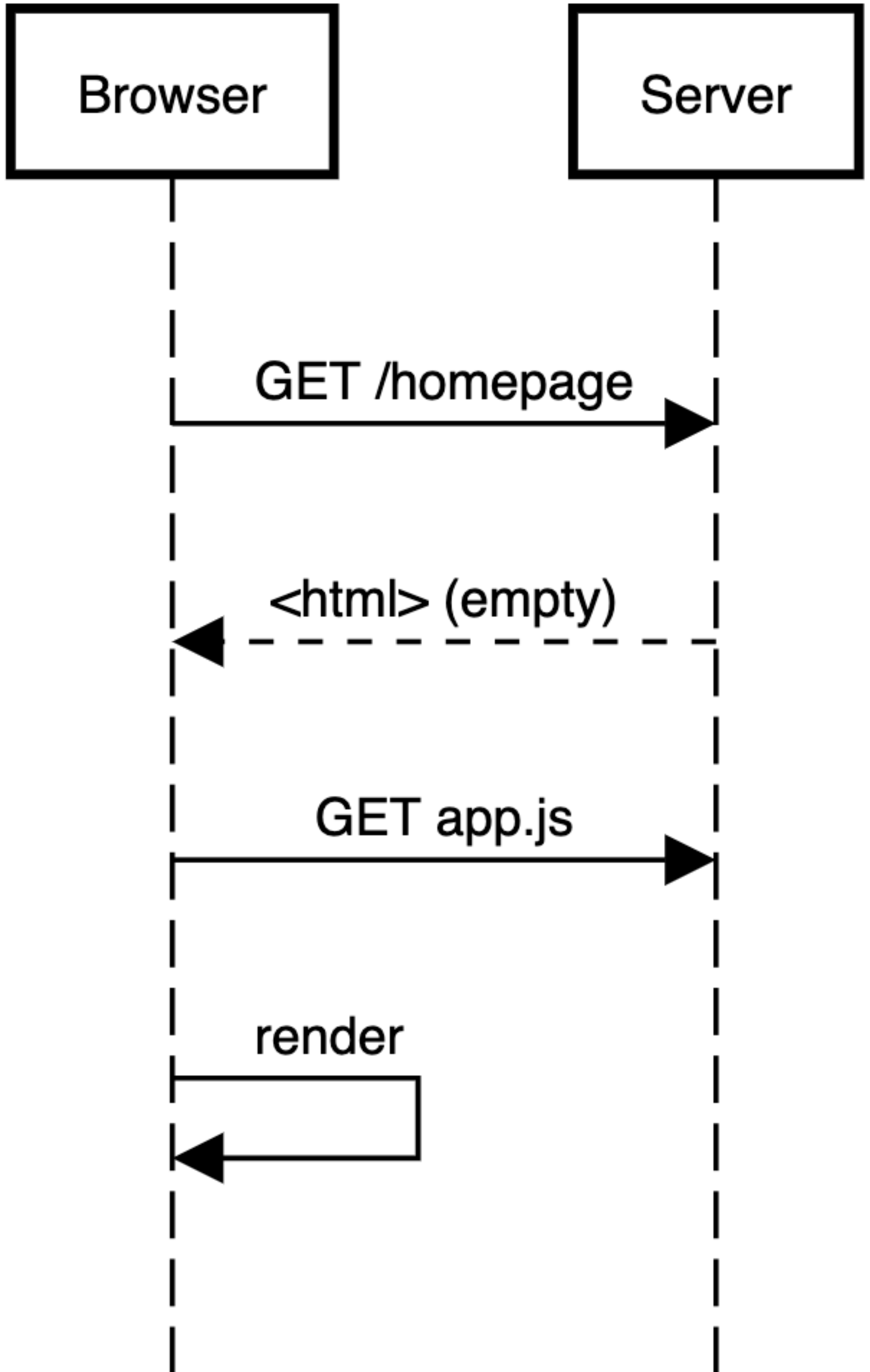
Classic SPA App



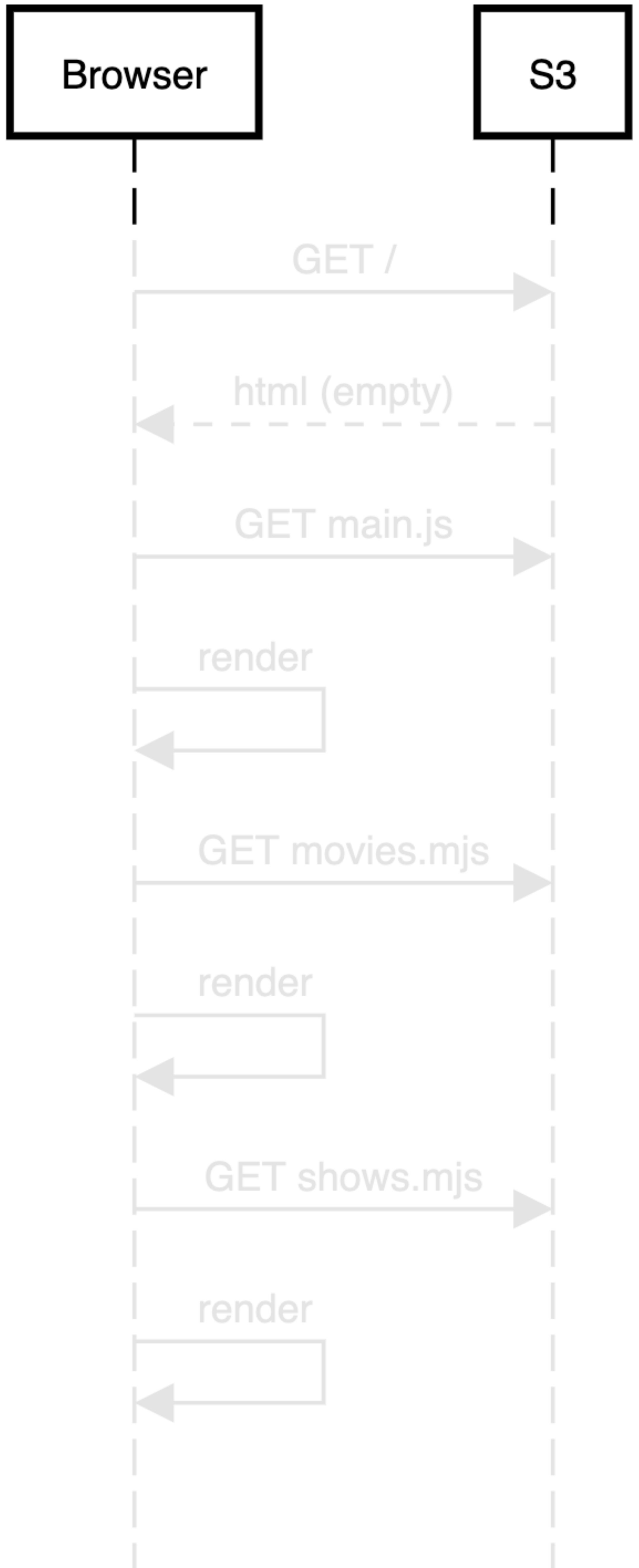
Classic SPA App



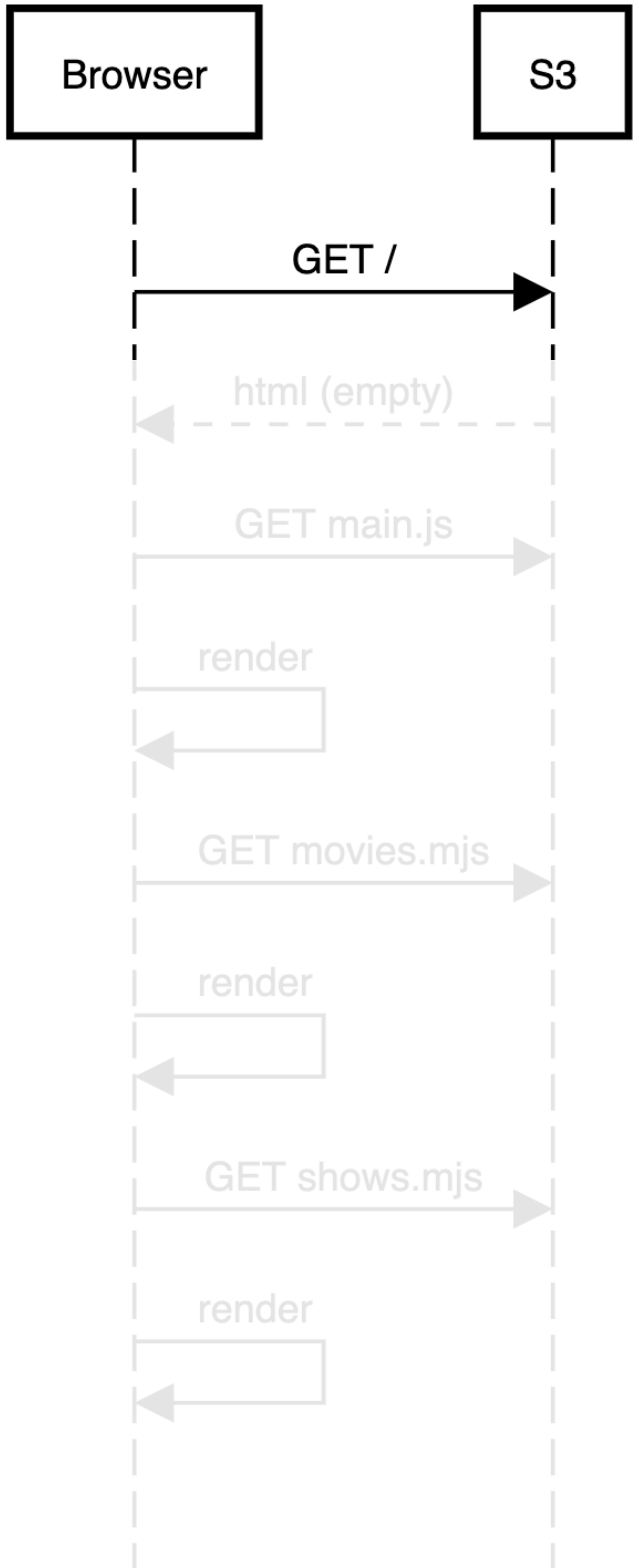
Classic SPA App



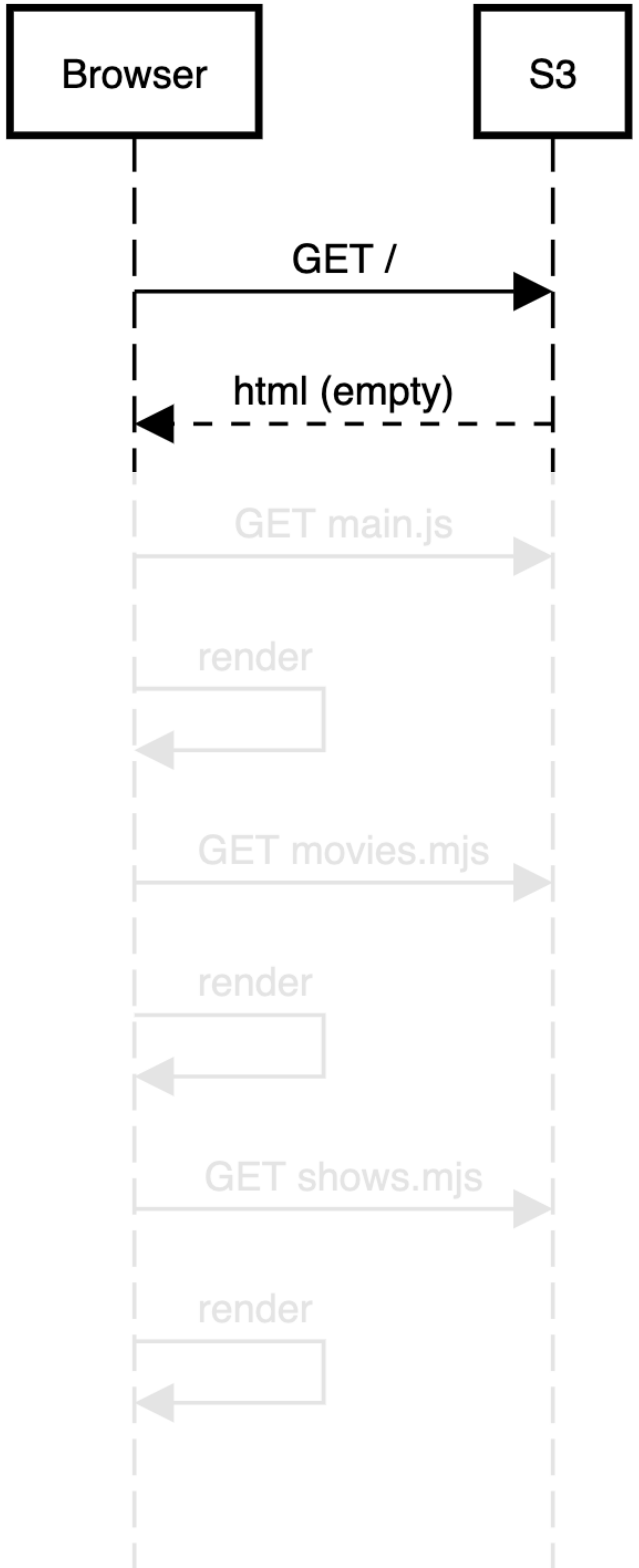
SPA Monoliths



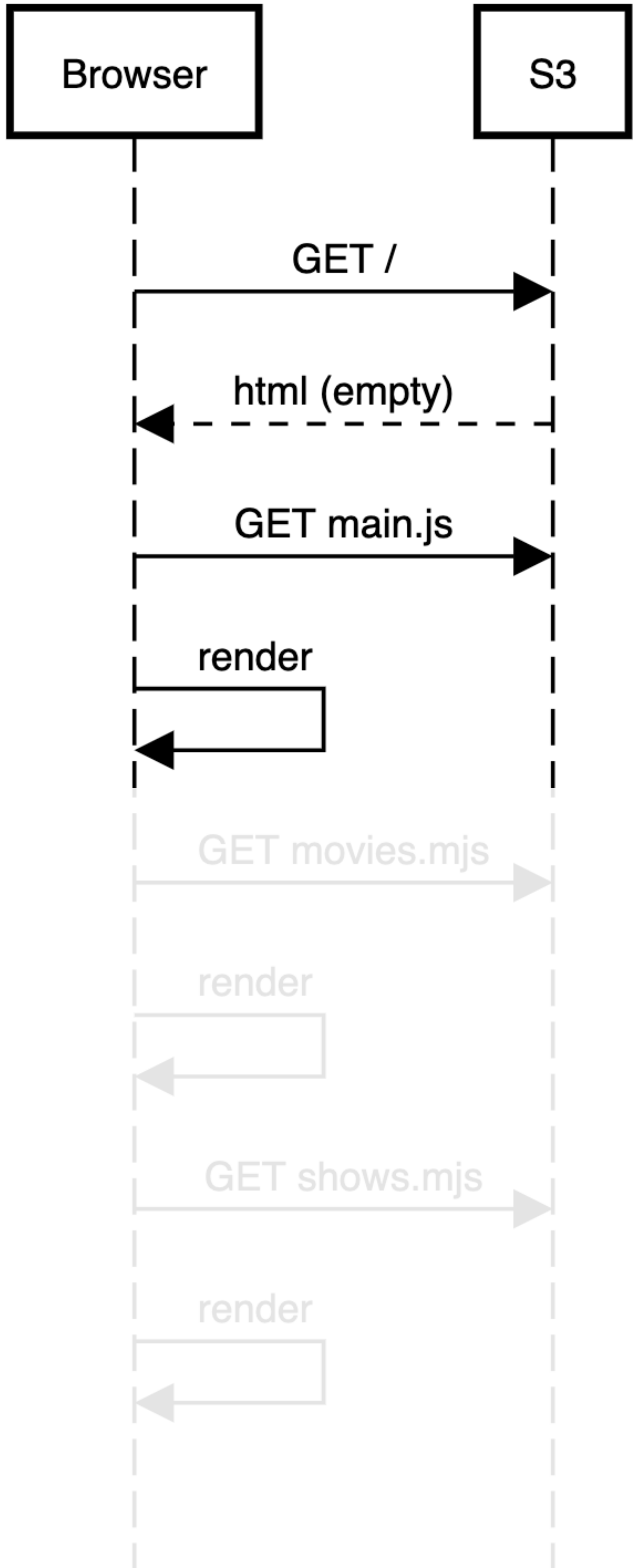
SPA Monoliths



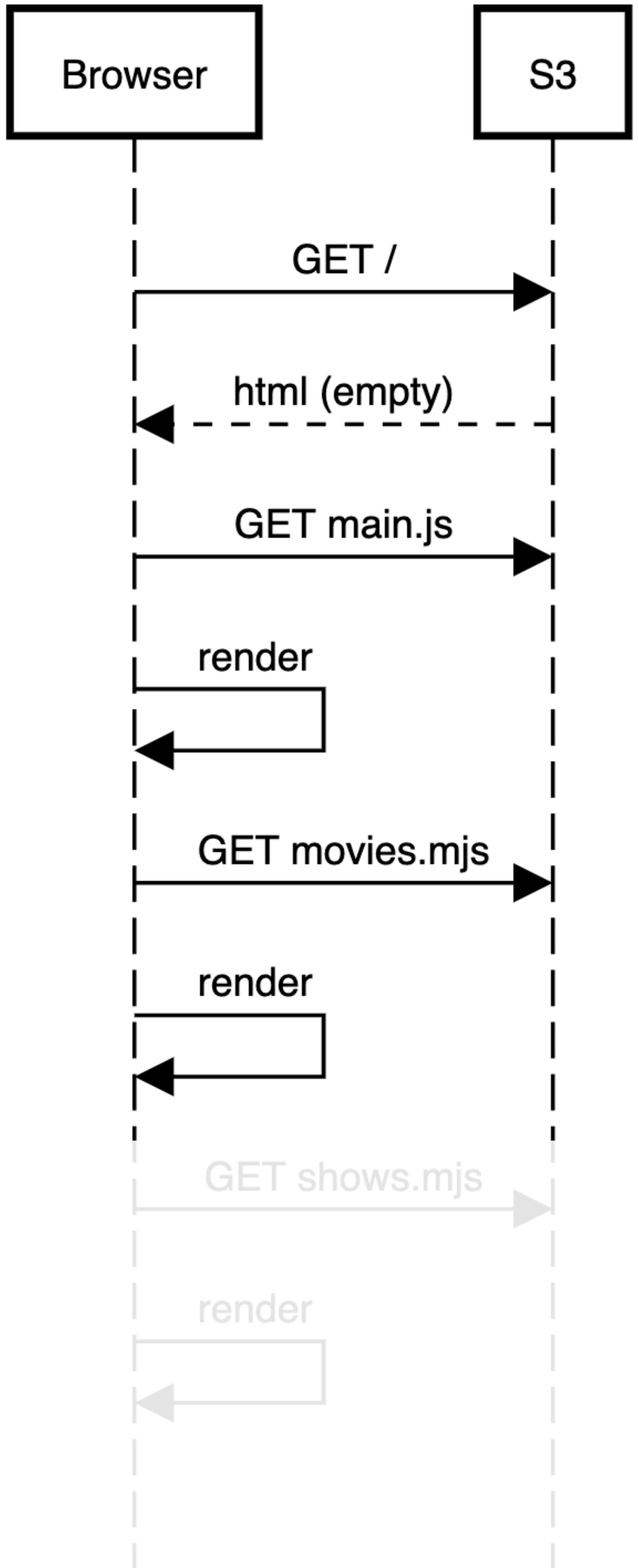
SPA Monoliths



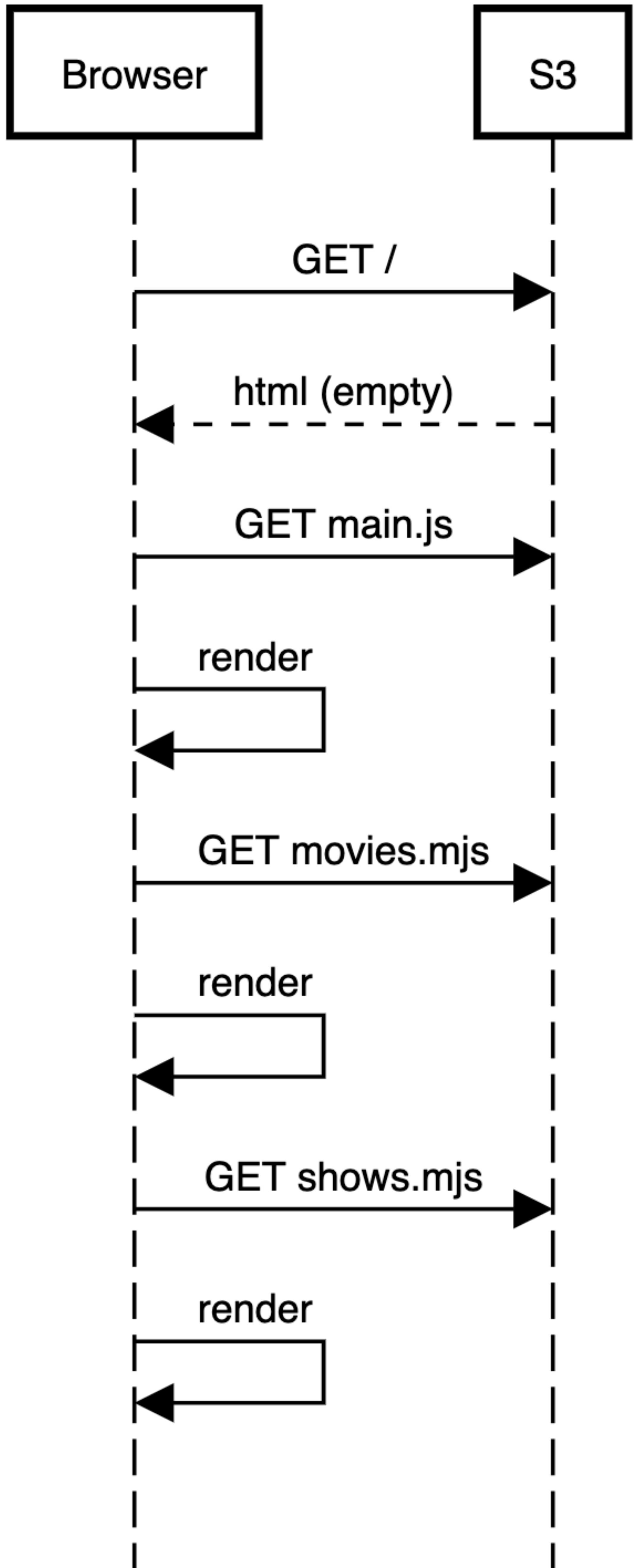
SPA Monoliths



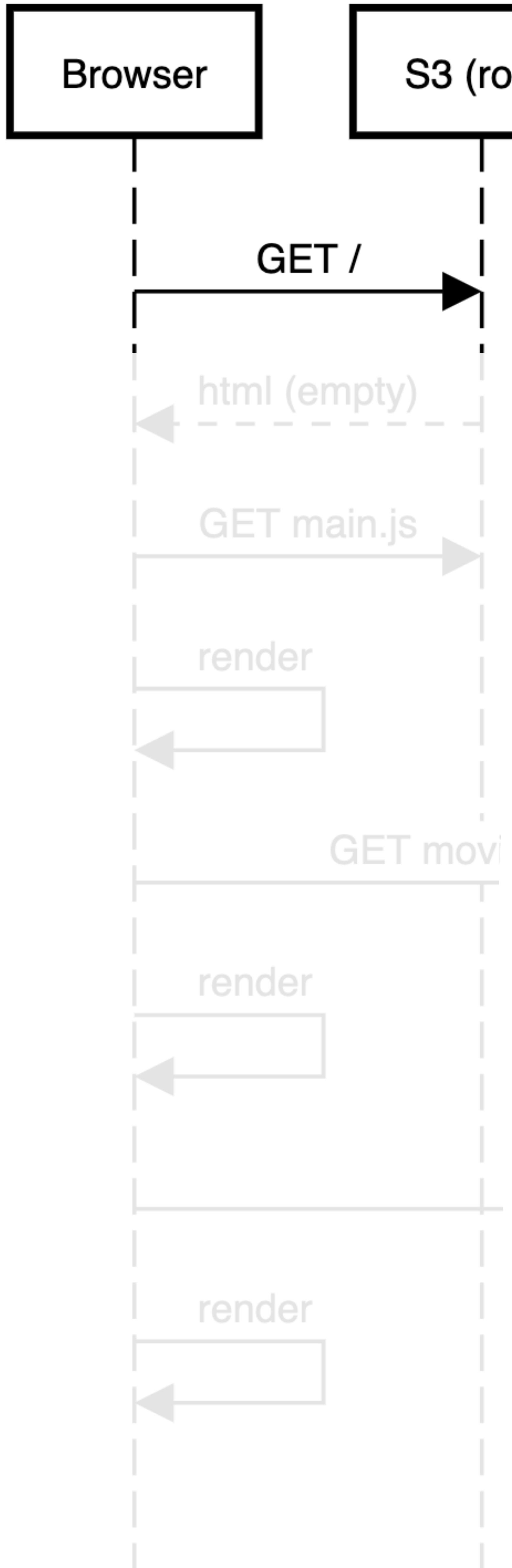
SPA Monoliths



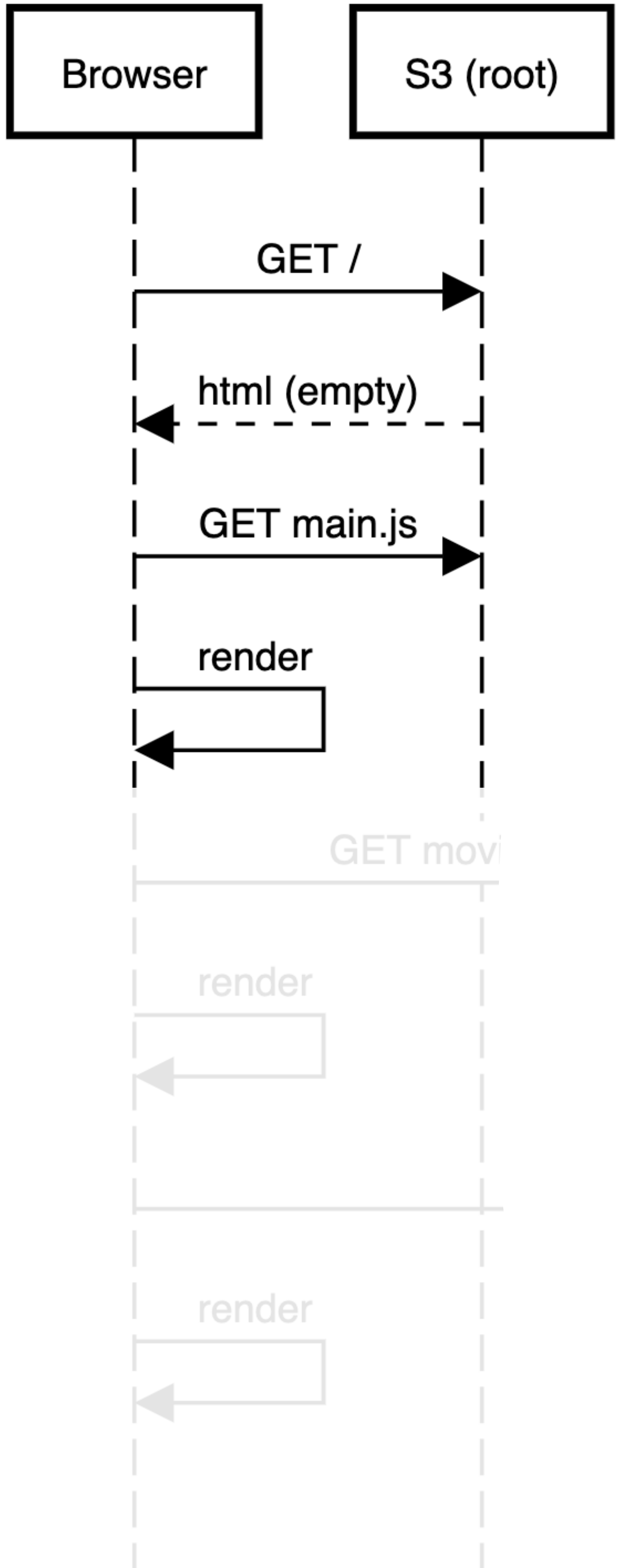
SPA Monoliths



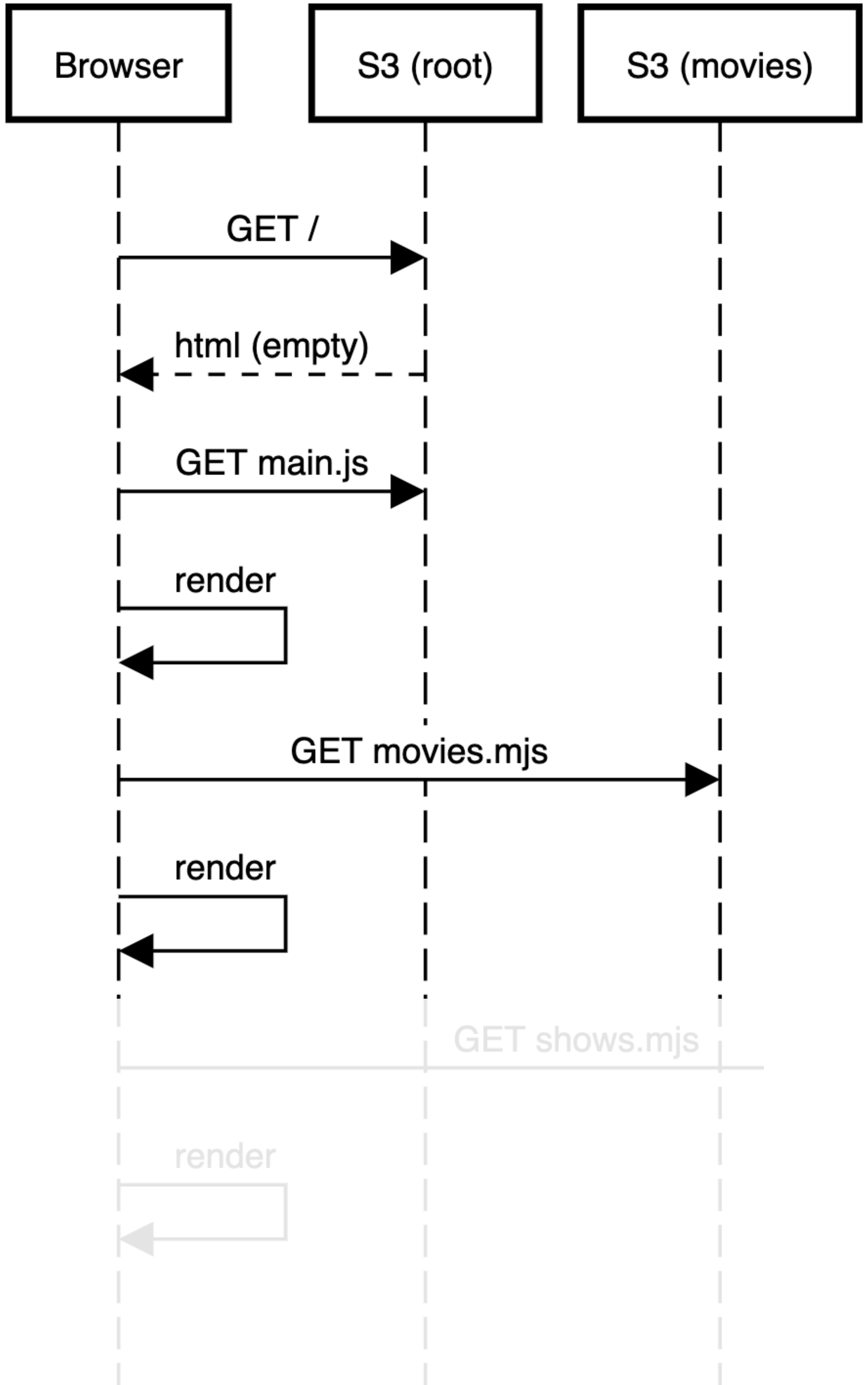
MFE Apps


Browser
S3 (root)
GET /
html (empty)
GET main.js
render
GET movi
render
render

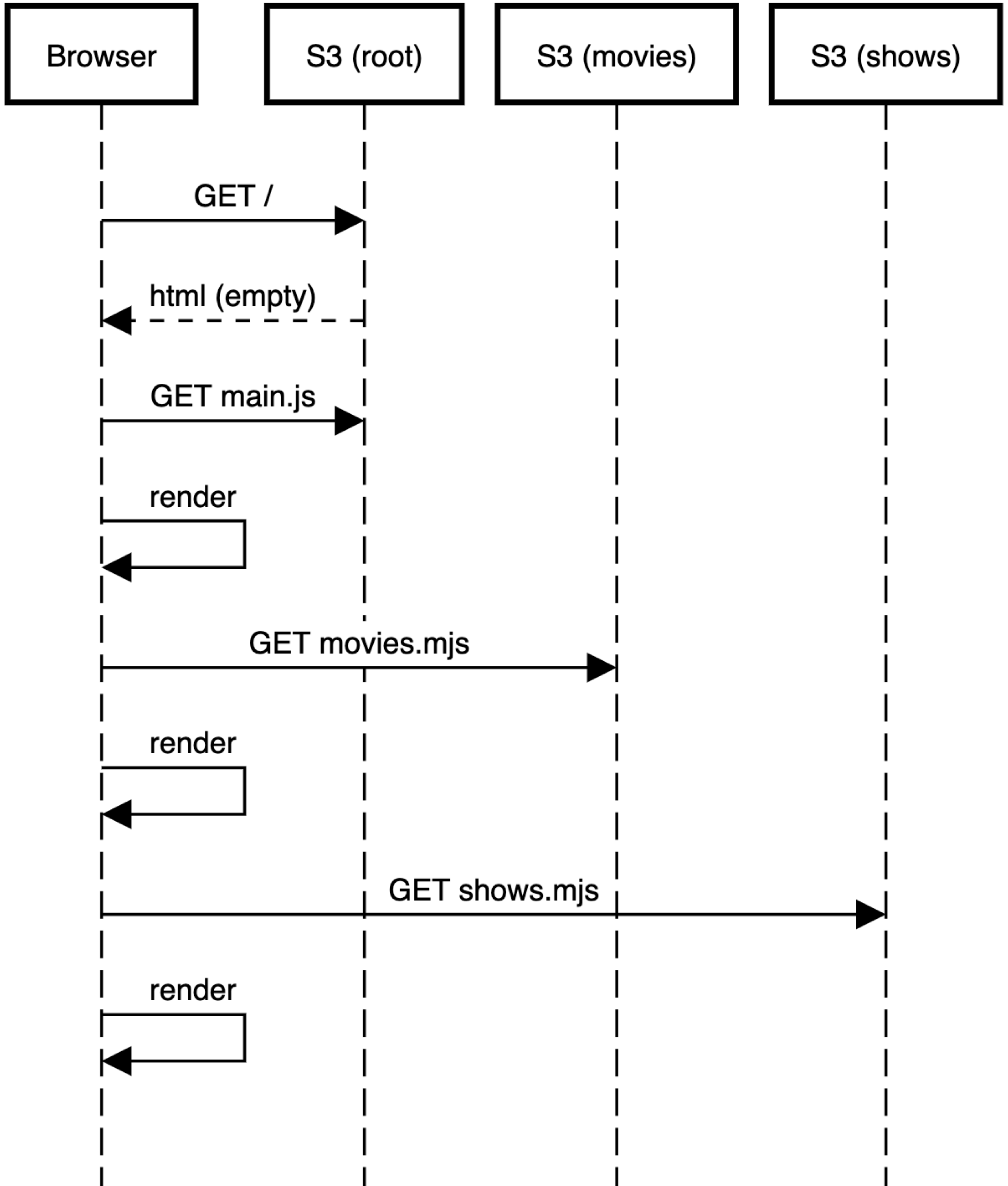
MFE Apps


Browser
S3 (root)
GET /
html (empty)
GET main.js
render
GET movi
render
render

MFE Apps

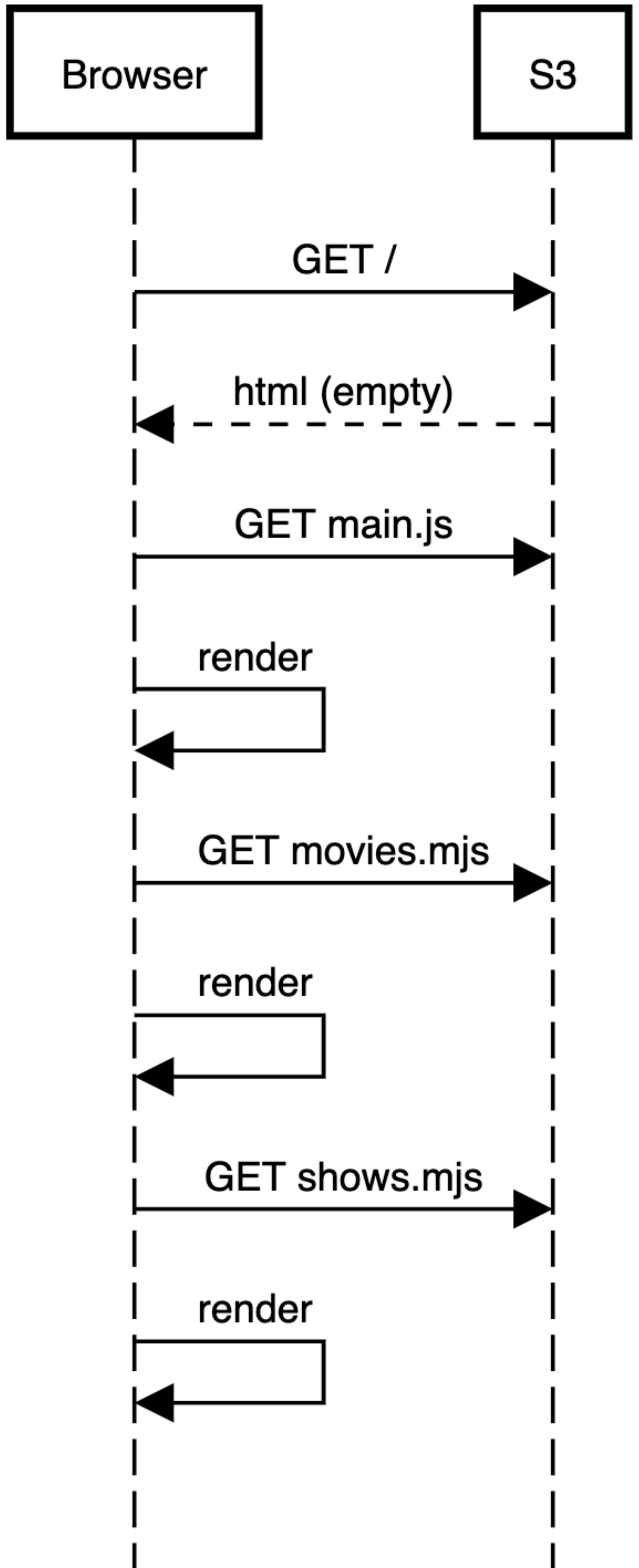

Browser
S3 (root)
S3 (movies)
GET /
html (empty)
GET main.js
render
GET movies.mjs
render
GET shows.mjs
render

MFE Apps



Browser
S3 (root)
S3 (movies)
S3 (shows)
GET /
html (empty)
GET main.js
render
GET movies.mjs
render
GET shows.mjs
render

SPA Monoliths



Browser

S3

GET /

html (empty)

GET main.js

render

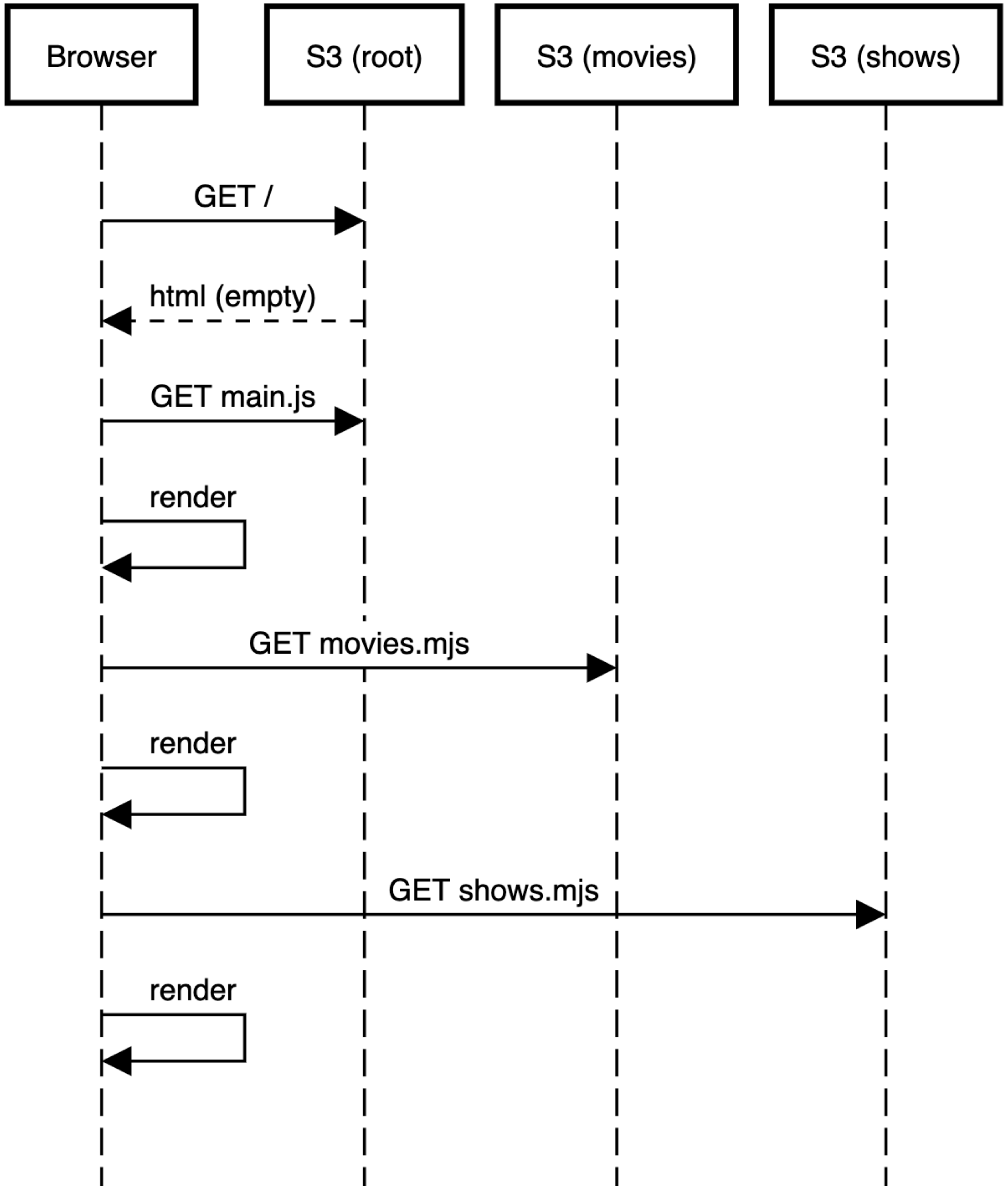
GET movies.mjs

render

GET shows.mjs

render

MFE Apps



Browser

S3 (root)

S3 (movies)

S3 (shows)

GET /

html (empty)

GET main.js

render

GET movies.mjs

render

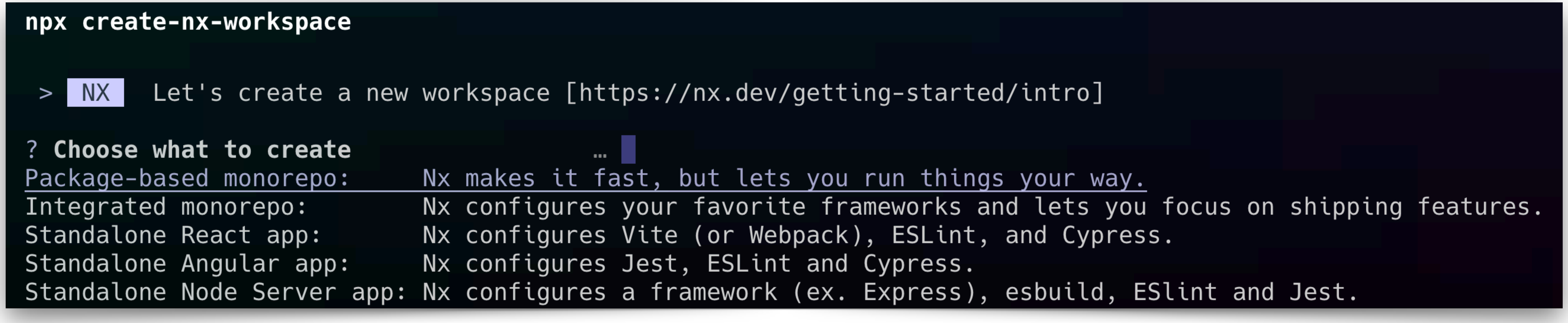
GET shows.mjs

render

04

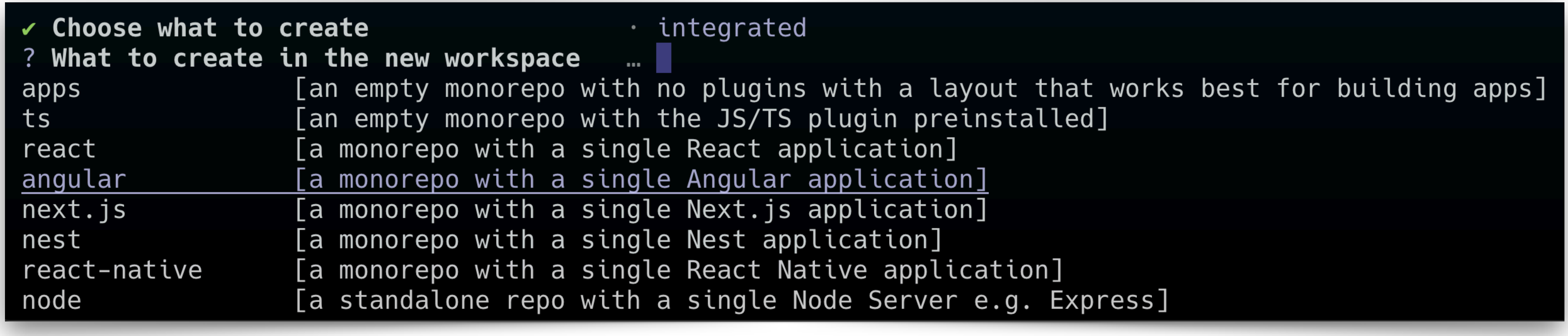
What can we do with Nx?

1. Pick your monorepo style



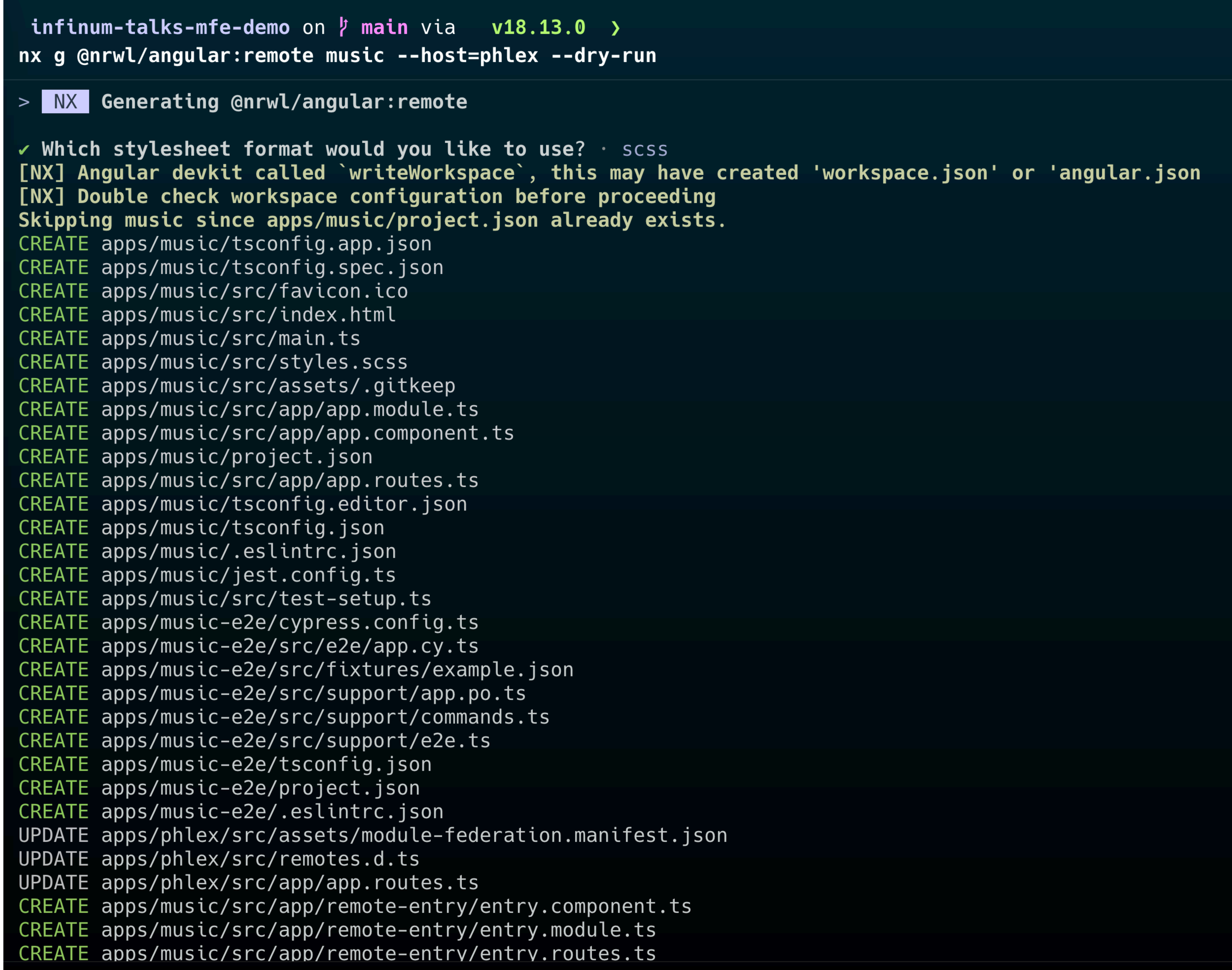
```
npx create-nx-workspace

 >  NX   Let's create a new workspace [https://nx.dev/getting-started/intro]

? Choose what to create                   …
Package-based monorepo:     Nx makes it fast, but lets you run things your way.
Integrated monorepo:        Nx configures your favorite frameworks and lets you focus on shipping features.
Standalone React app:       Nx configures Vite (or Webpack), ESLint, and Cypress.
Standalone Angular app:     Nx configures Jest, ESLint and Cypress.
Standalone Node Server app: Nx configures a framework (ex. Express), esbuild, ESlint and Jest.
```

2. Pick your framework



```
✔ Choose what to create                 · integrated
? What to create in the new workspace   …
apps                  [an empty monorepo with no plugins with a layout that works best for building apps]
ts                    [an empty monorepo with the JS/TS plugin preinstalled]
react                 [a monorepo with a single React application]
angular               [a monorepo with a single Angular application]
next.js               [a monorepo with a single Next.js application]
nest                  [a monorepo with a single Nest application]
react-native          [a monorepo with a single React Native application]
node                  [a standalone repo with a single Node Server e.g. Express]
```

3. Use generators



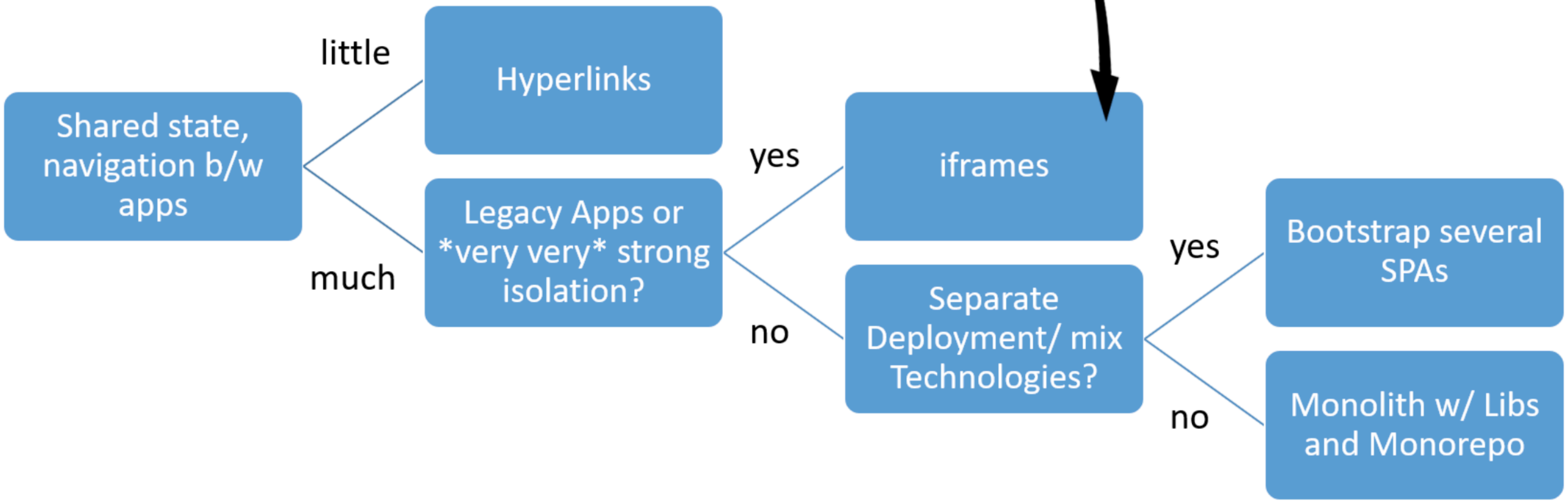
```
 infinum-talks-mfe-demo on  main via   v18.13.0 ❯
nx g @nrwl/angular:remote music --host=phlex --dry-run

>  NX   Generating @nrwl/angular:remote

✔ Which stylesheet format would you like to use? · scss
[NX] Angular devkit called `writeWorkspace`, this may have created 'workspace.json' or 'angular.json
[NX] Double check workspace configuration before proceeding
Skipping music since apps/music/project.json already exists.
CREATE apps/music/tsconfig.app.json
CREATE apps/music/tsconfig.spec.json
CREATE apps/music/src/favicon.ico
CREATE apps/music/src/index.html
CREATE apps/music/src/main.ts
CREATE apps/music/src/styles.scss
CREATE apps/music/src/assets/.gitkeep
CREATE apps/music/src/app/app.module.ts
CREATE apps/music/src/app/app.component.ts
CREATE apps/music/project.json
CREATE apps/music/src/app/app.routes.ts
CREATE apps/music/tsconfig.editor.json
CREATE apps/music/tsconfig.json
CREATE apps/music/.eslintrc.json
CREATE apps/music/jest.config.ts
CREATE apps/music/src/test-setup.ts
CREATE apps/music-e2e/cypress.config.ts
CREATE apps/music-e2e/src/e2e/app.cy.ts
CREATE apps/music-e2e/src/fixtures/example.json
CREATE apps/music-e2e/src/support/app.po.ts
CREATE apps/music-e2e/src/support/commands.ts
CREATE apps/music-e2e/src/support/e2e.ts
CREATE apps/music-e2e/tsconfig.json
CREATE apps/music-e2e/project.json
CREATE apps/music-e2e/.eslintrc.json
UPDATE apps/phlex/src/assets/module-federation.manifest.json
UPDATE apps/phlex/src/remotes.d.ts
UPDATE apps/phlex/src/app/app.routes.ts
CREATE apps/music/src/app/remote-entry/entry.component.ts
CREATE apps/music/src/app/remote-entry/entry.module.ts
CREATE apps/music/src/app/remote-entry/entry.routes.ts
```

05 Demo time!

06

When to use MFEs

Not a good fit for public web sites


Shared state, navigation b/w apps
little
much
Hyperlinks
Legacy Apps or *very very* strong isolation?
yes
no
iframes
Separate Deployment/ mix Technologies?
yes
no
Bootstrap several SPAs
Monolith w/ Libs and Monorepo

Source: Manfred Steyer - Enterprise Angular - Micro Frontends and Moduliths with Angular

“Don’t do it if you don’t have to”

~~“Don’t do it if you don’t have to”~~

“Don’t do it if you can’t reap the benefits”

07

Things to look out for

The devil is in the details

- **Individual MFE app initialisation logic**
  - Solution: “MODULE_INITIALIZER”
- **Serving and fetching of assets**
  - Solution: Provide APP_URL to each MFE
- **Environment-specific configuration at runtime**
  - Solution: Fetch env.json for each MFE
- **Global styles and providers**
  - Solution: Do them all in root MFE
- **Deployment synchronisation**
  - Solution: Nx affected
- **Framework Pick n Mix**
  - Solution: Don’t do it if you don’t have to (you lose some of the benefits)

The devil is in the details

- **Dependency updates & management**
  - Solution: Do it regularly, on a schedule
- **Inter-team dependencies**
  - Solution: Plan & communicate
- **Branching process**
  - Solution: Less branches / Trunk-based development / Good tagging
- **Making people stick to coding standards**
  - Solution: Enforce as much as possible with tooling
- **Apps that stop being maintained**
  - Solution: Fork and do classic stand-alone non-MFE SPA build
- **Noise in the monorepo**
  - Solution: Proper CODEOWNERS setup & clear definition of responsibilities

08

Alternate approaches

Alternate approaches

- **Custom Elements (a.k.a. Web Components)**
  - https://web.dev/custom-elements-v1/

```
<head>
  <script src=“https://components.mycompany.com/my-widget.js”>
</head>
<body>
  <my-widget></my-widget>
</body>
```

- **Server-side includes**
  - https://en.wikipedia.org/wiki/Server_Side_Includes

```
<!--#include virtual="menu.cgi" -->
<!--#include file="footer.html" -->
```

- **Edge-side includes**
  - https://en.wikipedia.org/wiki/Edge_Side_Includes

```
<esi:include
  src="http://example.com/1.html"
  alt="http://bak.example.com/2.html"
  onerror="continue"
/>
```

Thank you!

Clutch
Facebook
Instagram
Twitter
Linkedin
Dribbble

Resources

- https://micro-frontends.org
- https://www.youtube.com/watch?v=tslZjUAtF1U
- https://www.angulararchitects.io/en/book
- https://webpack.js.org/concepts/module-federation
- https://nx.dev/more-concepts/micro-frontend-architecture
- https://nx.dev/recipes/module-federation/dynamic-module-federation-with-angular
- https://auth0.com/blog/micro-frontends-with-angular-module-federation-and-auth0
- https://nx.dev/more-concepts/library-types
- https://nx.dev/concepts/mental-model
- https://nx.dev/concepts/integrated-vs-package-based

Questions?

Clutch
Facebook
Instagram
Twitter
Linkedin
Dribbble