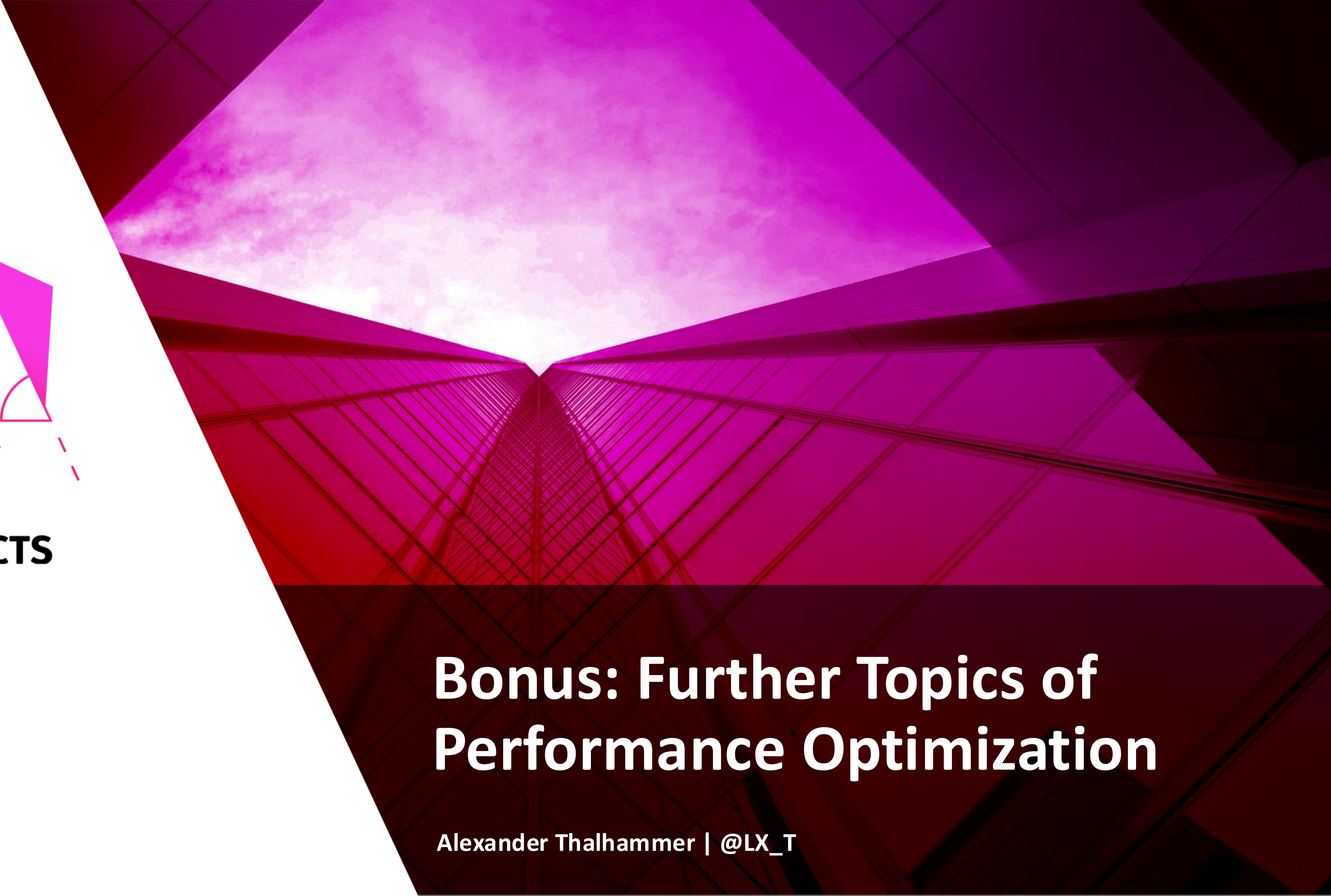
ANGULAR
ARCHITECTS



Bonus: Further Topics of Performance Optimization

Alexander Thalhammer | @LX_T

Further Topics

- Don't use Angular resolvers (if you ask me)
- Smart vs Dumb Components
- API Architecture
- RxJS & NgRx
- Web Worker for heavy calculations
- Service Worker / PWA
- Scheduling
- Building with Nx?
- Building with Vite & esbuild

Don't use Angular resolvers

– Better to show title & everyting possible, even just the frame

– Instead use local spinners where data is being loaded

Thought experiment

- What if <app-flight-card> would handle use case logic?
  - e.g. communicate with API (thru a service)
- Number of requests → Performance?
- Traceability?
- Reusability?

Smart vs. Dumb Components

Smart vs. Dumb Components

Smart / Controller	Dumb / Presentational
• Use Case controller • Container	• Independent of Use Case • Reusable • Leaf

ANGULAR ARCHITECTS

API Architecture

- Try to minimize API calls
  - E.g. fetch data in list not list item
  - If possible aggregate data in backend, not frontend
- Think about caching API calls
  - If possible, maybe valid for limited time only
- Maybe use GraphQL?

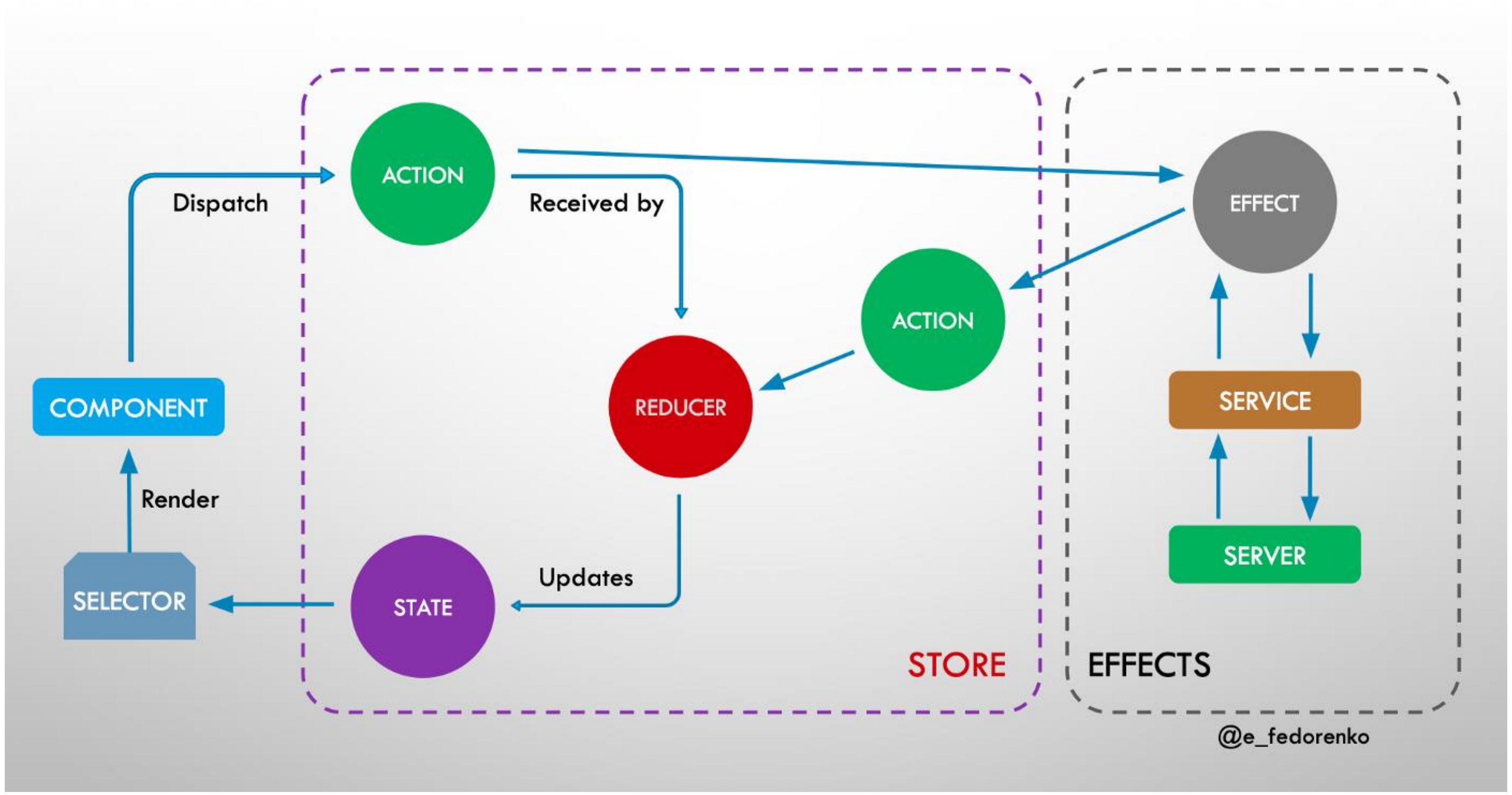
Use RxJS & NgRx

- Use RxJS properly
  - Share hot observables where possible
  - Pipe operaters
  - Use async pipe
  - Manage subscriptions
- Use State Management (NgRx, else SignalStore)
  - By using Redux libraries properly, you can improve its performance, by reducing the number of events that occur during data communication

State management options

- Global
  - NgRx Store (Redux, better imho), should be divided into features
- Local
  - Services / Facades w BehaviourSubjects or Signals (very lightweighted)
  - NgRx SignalStore (rather lightweighted)

Global state management (NgRx)



https://medium.com/angular-in-depth/how-i-wrote-ngrx-store-in-63-lines-of-code-dfe925fe979b

Web Workers for heavy calculations

- Problem: JS is single threaded, how to do heavy calculations?

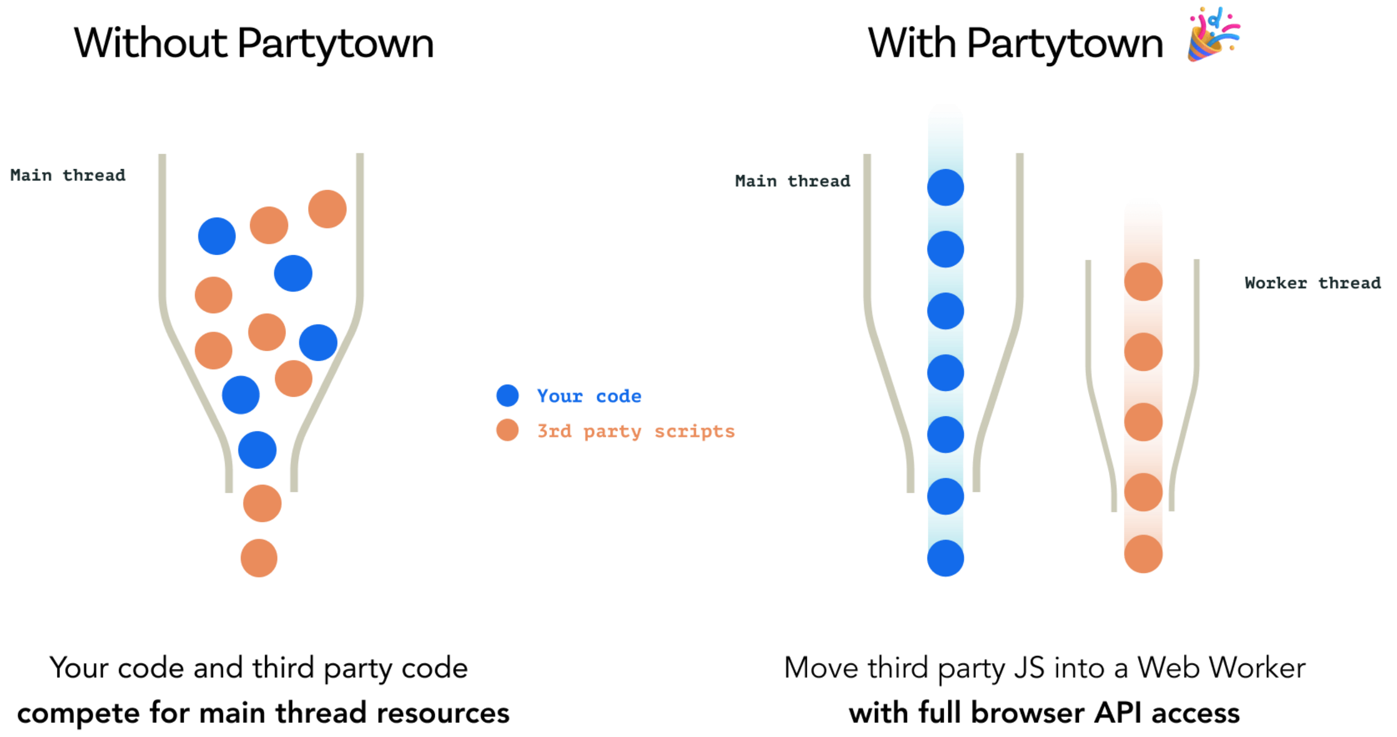
- Solution: Delegate to web worker, it will create a new thread called the Worker Thread that will run a JS script parallel to the main thread

Web Workers – Use cases

- Import external scripts
- Make XMLHttpRequest / API requests
- Use setTimeout() and setInterval()
- Spawn other workers
- Use IndexedDB, Notifications API, Web Crypto API, WebAssembly, WebSockets, WebGL, OffscreenCanvas, ImageData…
- Terminate themselves when you deem they are no longer needed

Web Workers – Implementations

- Worklet API
- partytown
- Comlink?
- …



Your code and third party code **compete for main thread resources**

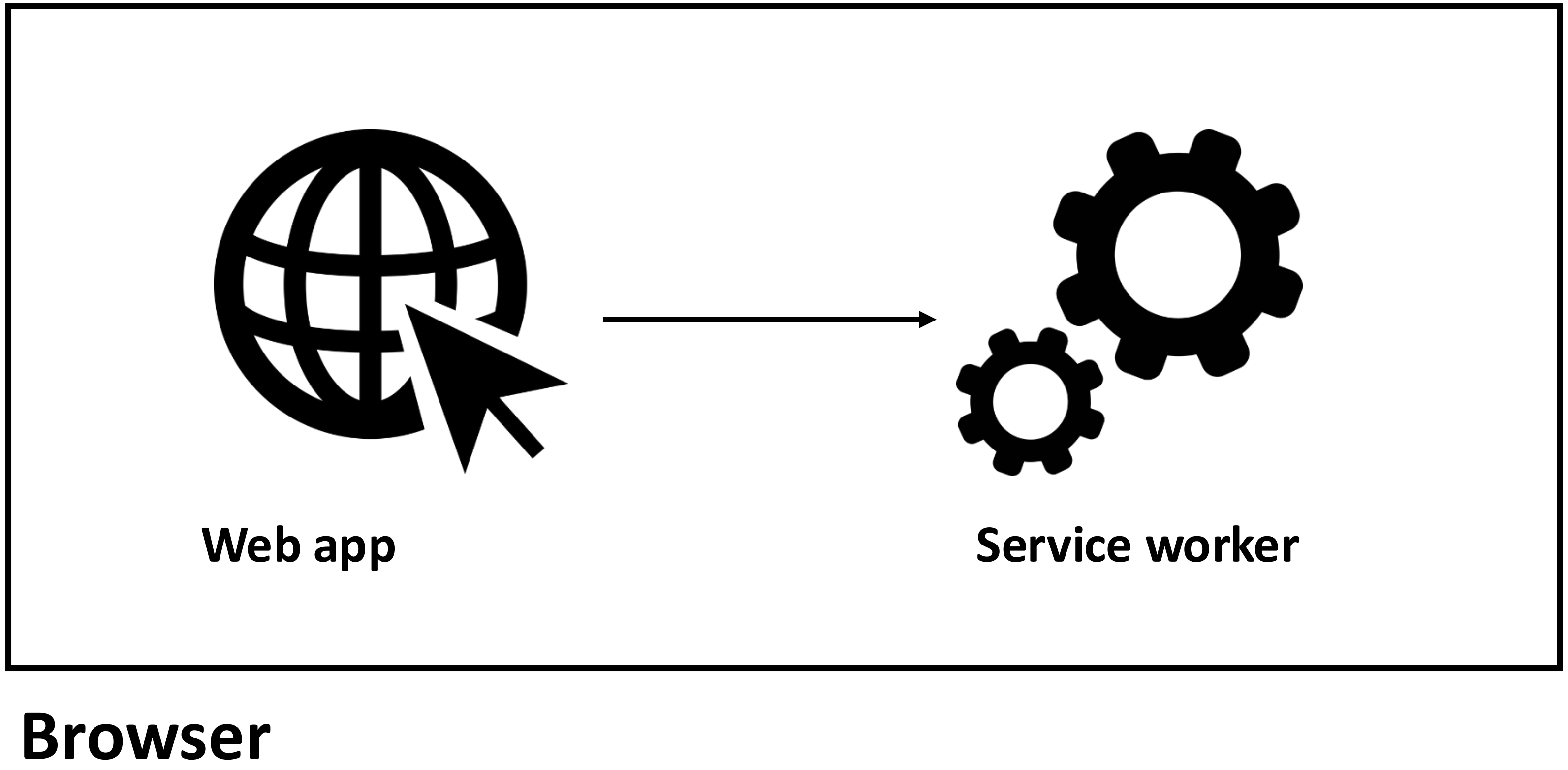
Move third party JS into a Web Worker **with full browser API access**

ANGULAR ARCHITECTS

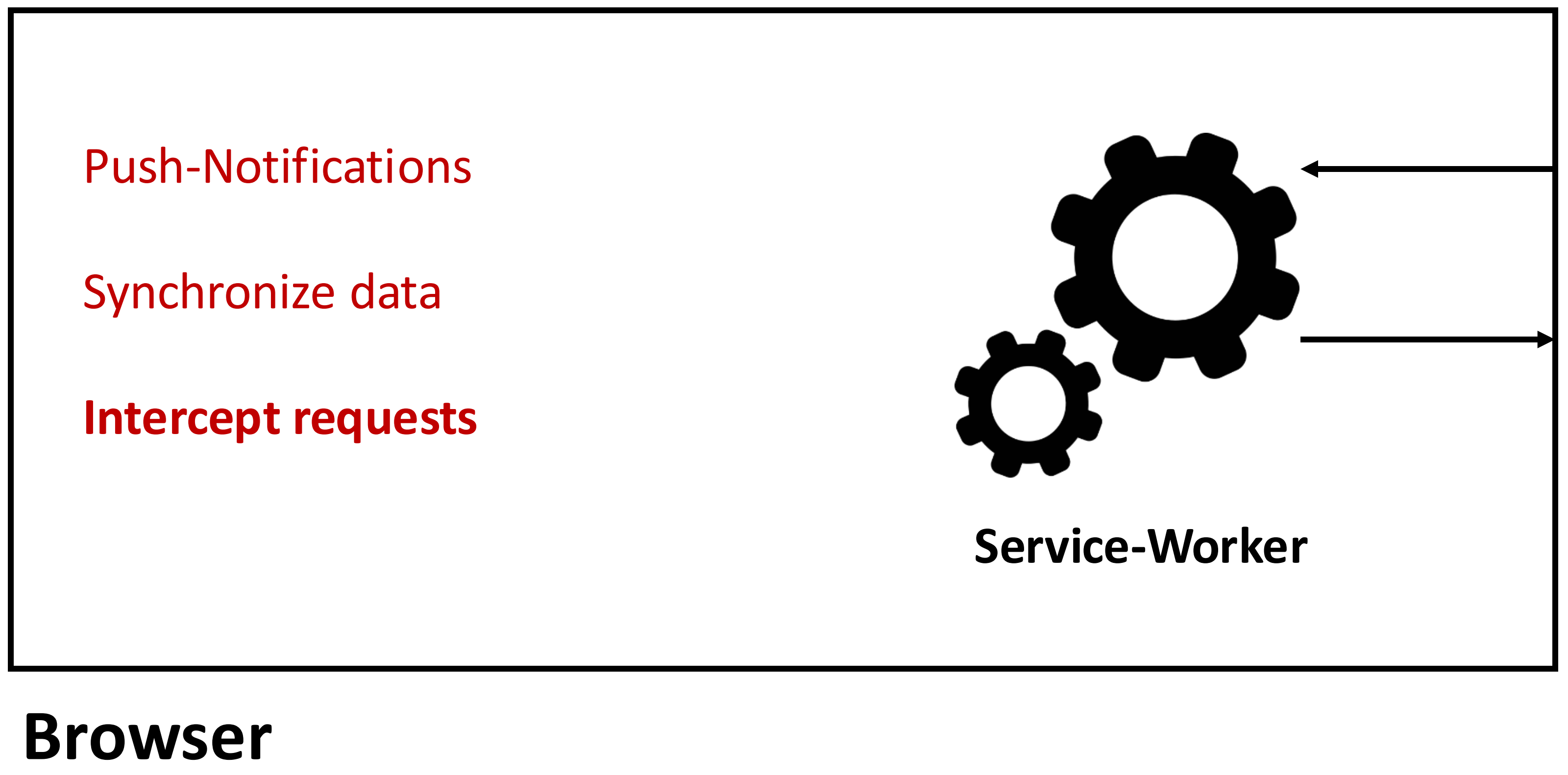
Service Workers (PWA)

- Handle offline state (no connection)
- Web push notifications (new in iOS)
- Proxy or caching HTTP requests
- Background code execution
- Process payments
- …

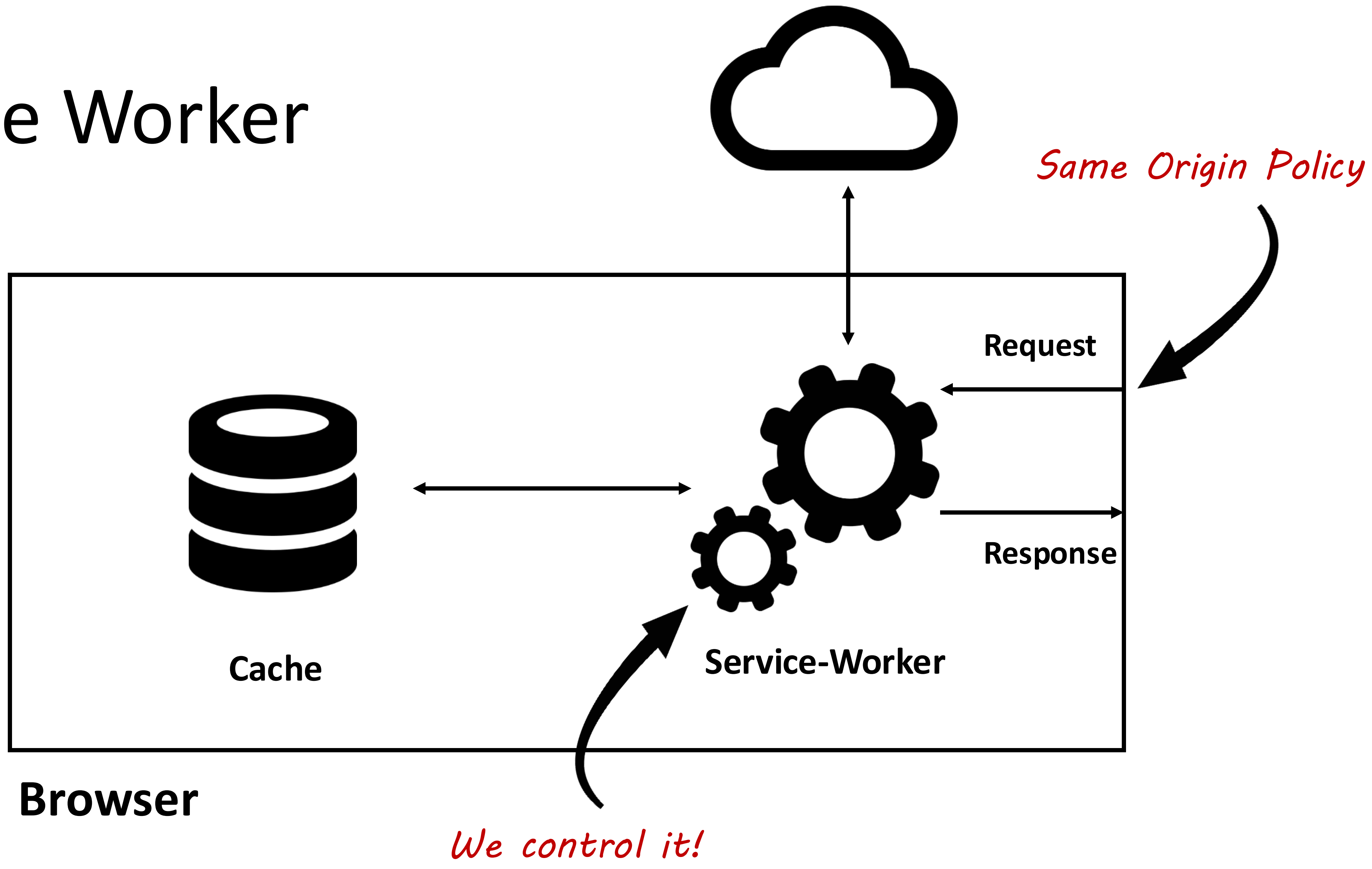
Service Worker



Service Worker



Service Worker



ANGULAR ARCHITECTS

Cache Strategies

Cache only

Network only

Try Cache, fallback to Network

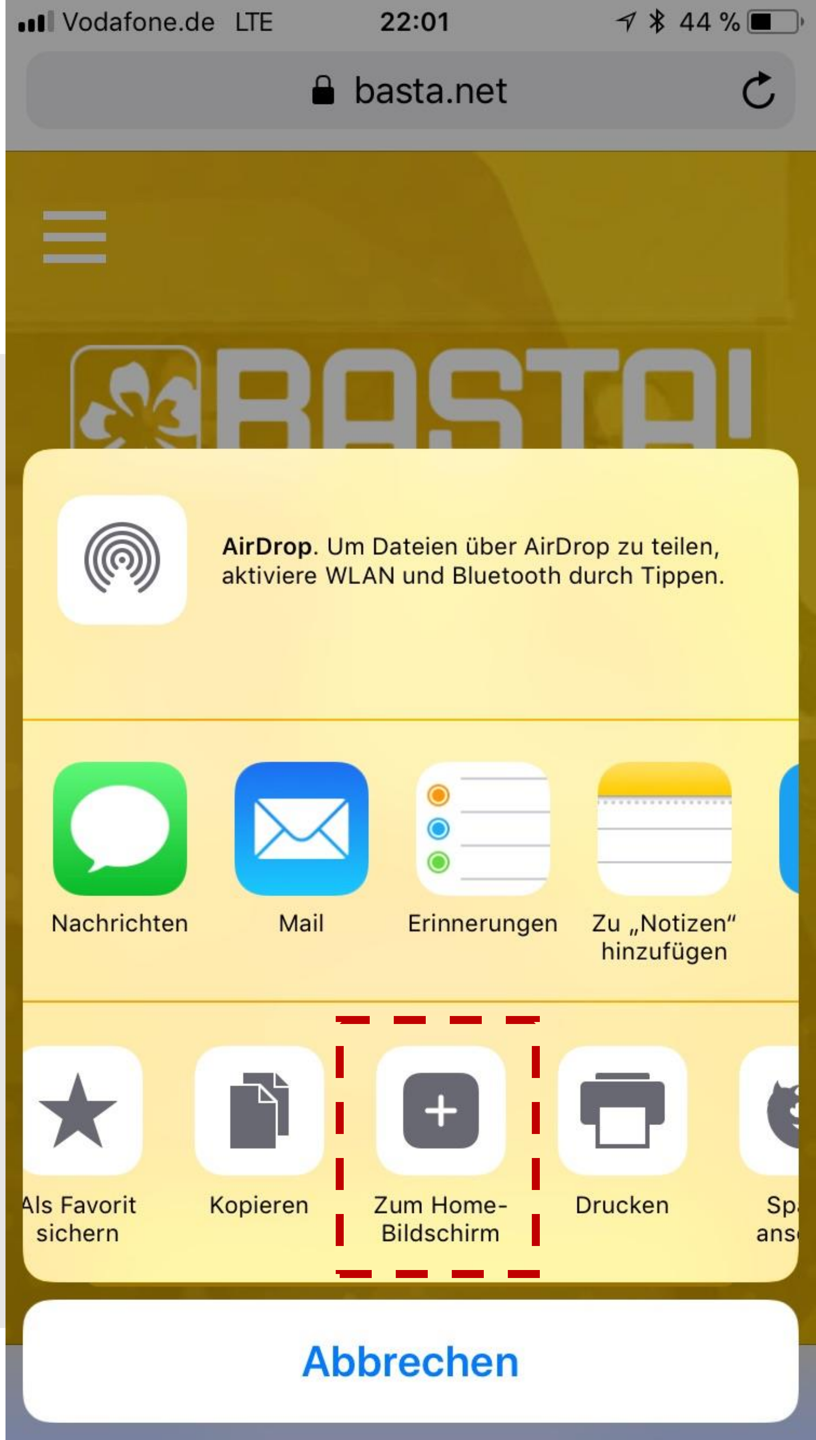
Try Network, fallback to Cache

…

Add to home screen

Web App Manifest

```
{
  "name": "Hotel PWA-Demo",
  "short_name": "Hotel",
  "icons": [{
        "src": "images/touch/icon-128x128.png",
        "sizes": "128x128",
        "type": "image/png"
      }, […] ],
  "start_url": "/index.html?homescreen=1",
  "display": "standalone",
  […]
}
```



AirDrop. Um Dateien über AirDrop zu teilen, aktiviere WLAN und Bluetooth durch Tippen.

@angular/pwa

- installs @angular/service-worker
  - npm install
  - imports Angular-Modul
  - generates ngsw-config.json
  - generates Web App Manifest
- ng add @angular/pwa

Scheduling

- Use setTimeout() to delay work
- Use setInterval() to invoke tasks continously
- Don't forget to clearTimeout() & clearInterval() onDestroy
- Can lead to unwanted Change Detection

Building with Nx?

- For bigger / enterprise Apps use @nx

- Nx is a 3rd party extension for Angular CLI supporting
  - Monorepo workspace
    - Split App(s) into buildable parts / libs
    - Only recompile changed parts (both during serve & build)
    - Possible to have a cloud build cache
  - Other features like
    - Schematics / generators
    - Access restrictions
    - Dependy graph
  - Out-of-the-box support for JEST, (Cypress | Playwright) & Storybook

Building with vite & esbuild

Further Topics

- – Don't use Angular resolvers (if you ask me)
- – Smart vs Dumb Components
- – API Architecture
- – RxJS & NgRx
- – Web Worker for heavy calculations
- – Service Worker / PWA
- – Scheduling
- – Building with Nx?
- – Building with Vite & esbuild

Questions?

ANGULAR
ARCHITECTS