

Micro Frontends mit Web Components

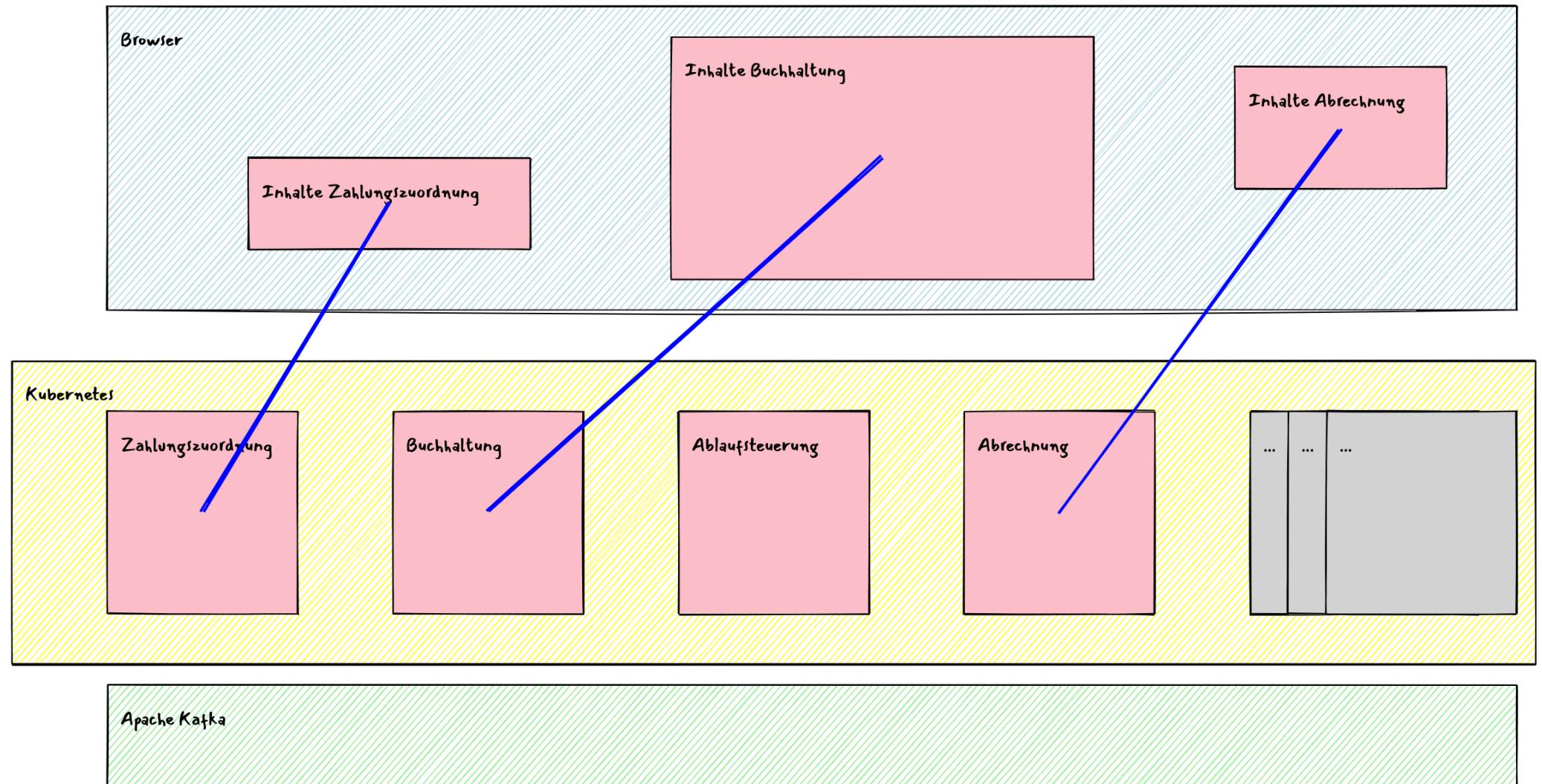
Roger Gilliar
Oliver Widder

Diese Folien
gibt es auch
online und
animiert hier:

bit.ly/webcomp-oop



Webseite mit Inhalten aus mehreren Apps



Microfrontends are the technical representation of a business subdomain, they provide strong boundaries with clear contracts, also they avoid sharing logic with other subdomains

Welche Probleme möchte man mit Micro Frontends lösen?

Welche besondere Herausforderungen ergeben sich durch Micro Frontends?

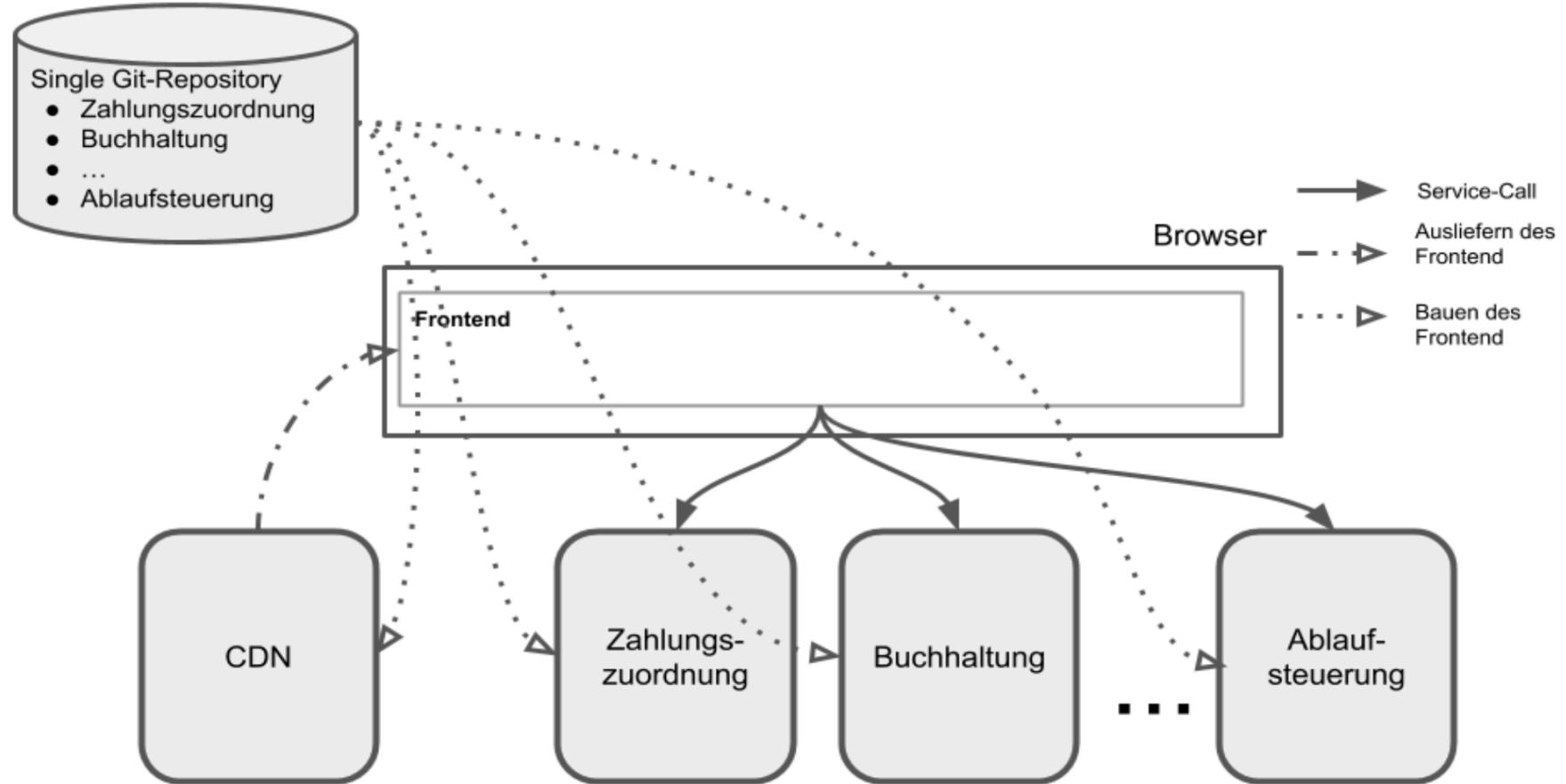
Micro Frontend Patterns

- 1 Build Time Integration
- 2 Server Side Integration
- 3 Client Side Integration

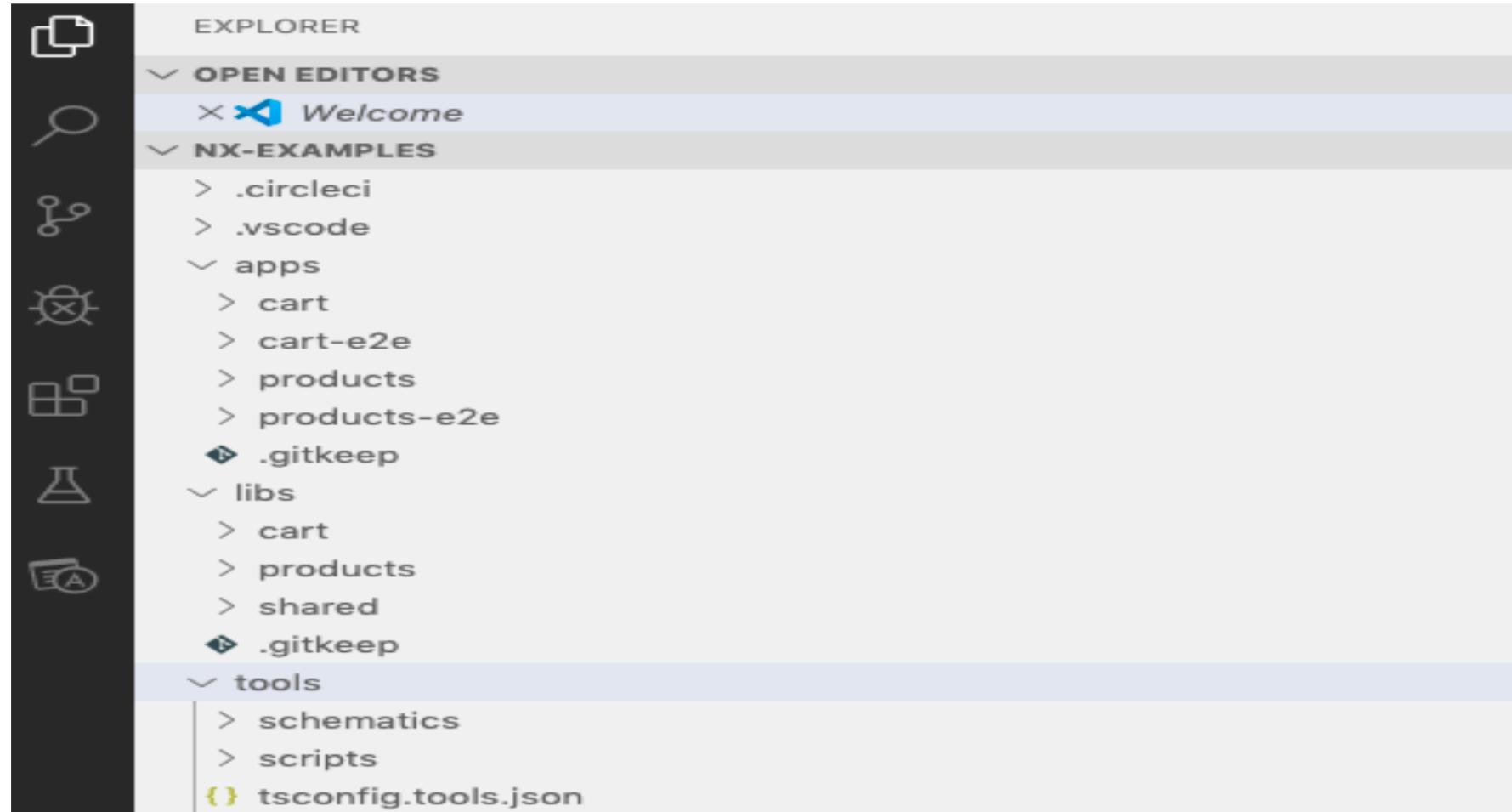
Pattern 1: Build Time Integration

- Polyrepo
- Monorepo ([nx.dev/web](#))

Build Time Integration (Monorepo)



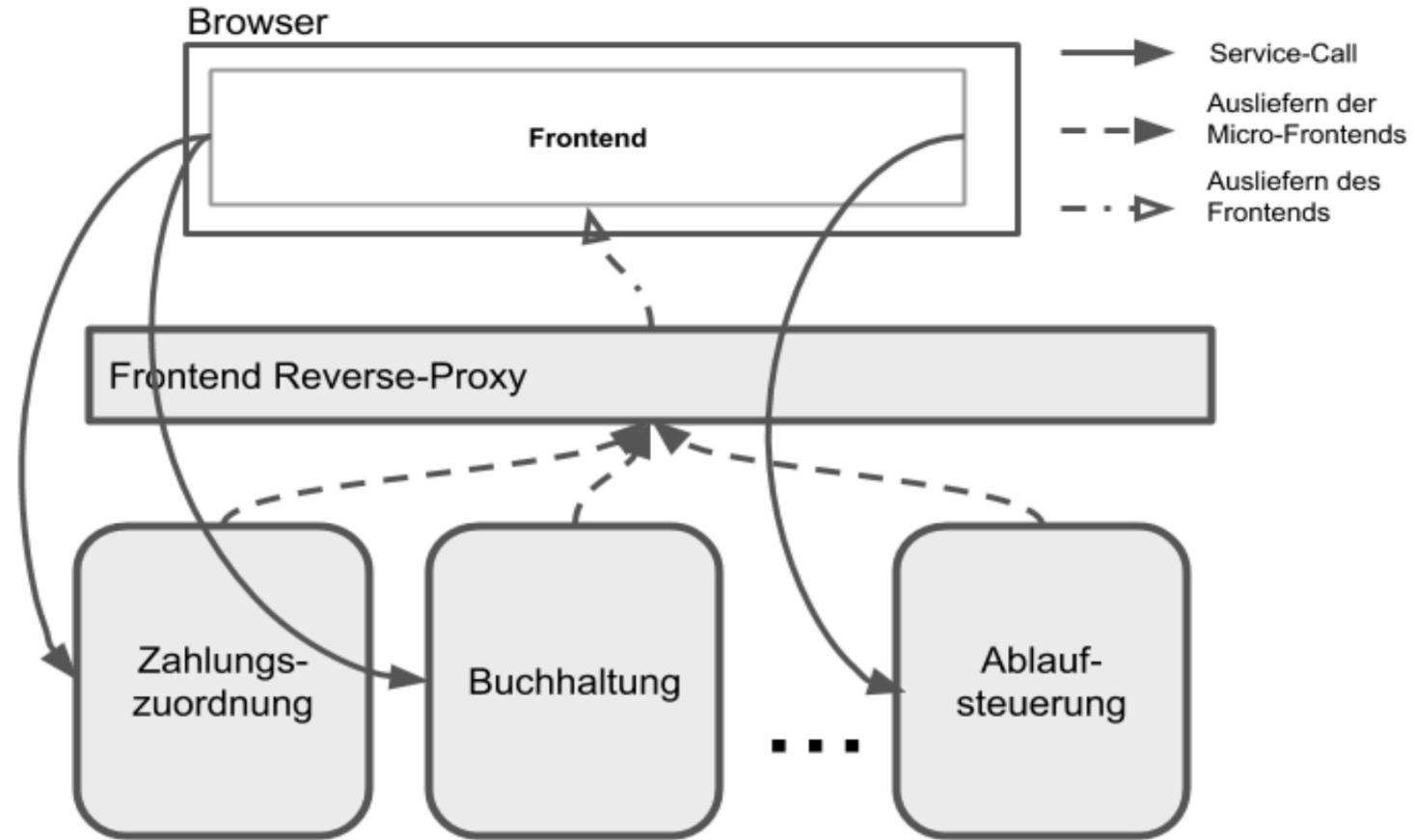
Monorepo mit NX



Pattern 2: Server Side Integration

- Server Side Includes
- Edge Side Includes
- Project Mosaic (www.mosaic9.org/)

Server Side Integration



Edge Side Include

```
1 <esi:include  
2   src="http://example.com/1.html"  
3   alt="http://bak.example.com/2.html"  
4   onerror="continue"  
5 />
```

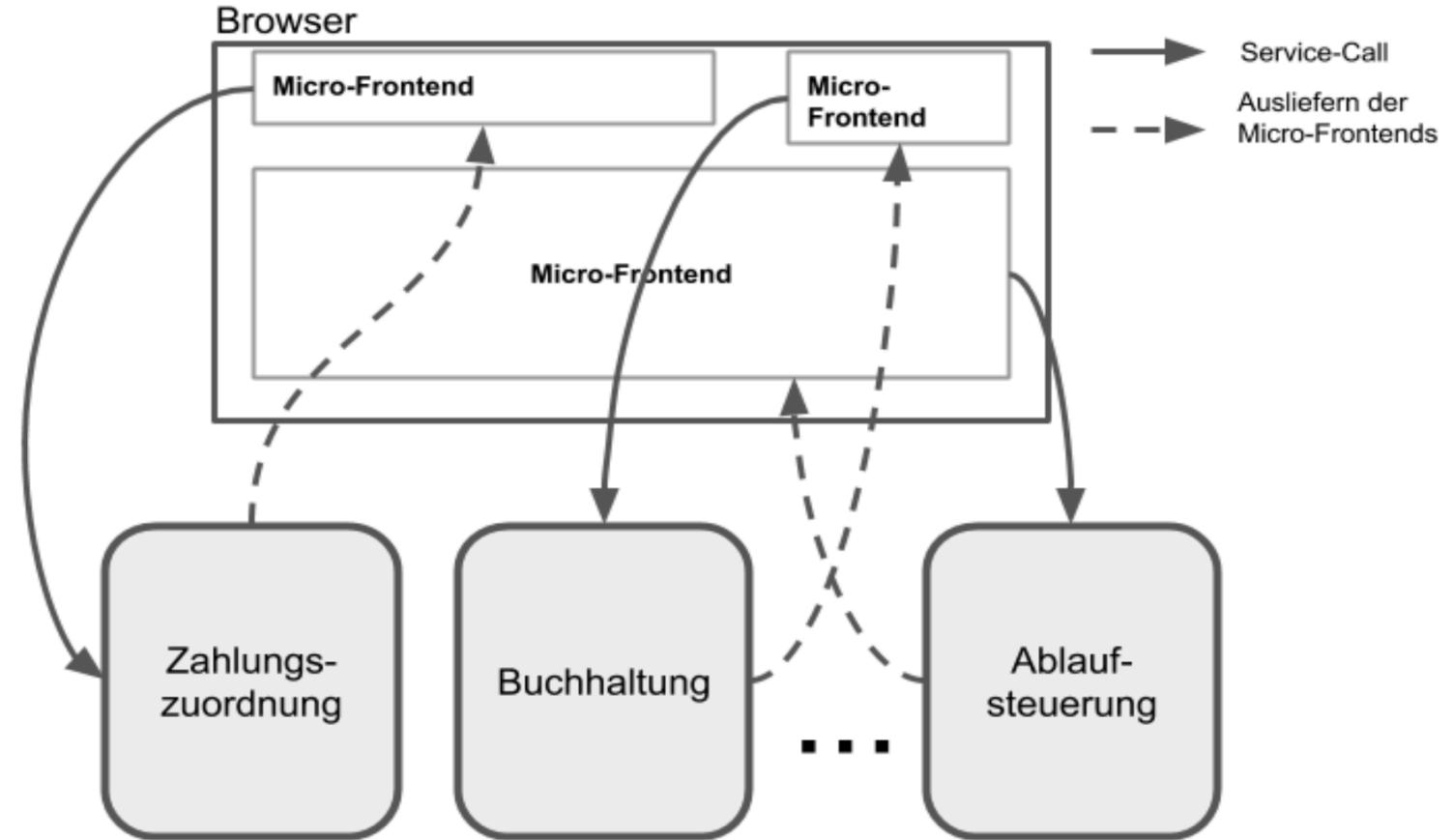
Project Mosaic

```
1 <html>
2 <head>
3 <script type="fragment" src="http://assets.domain.com"></script>
4 </head>
5 <body>
6   <fragment src="http://header.domain.com"></fragment>
7   <fragment src="http://content.domain.com" primary></fragment>
8   <fragment src="http://footer.domain.com" async></fragment>
9 </body>
10 </html>
```

Pattern 3: Client Side Integration

- IFrame
- single-spa.js.org
- Web Components

Client Side Integration



IFrame

```
1 <iframe  
2   width="560"  
3   height="315"  
4   src="https://www.youtube.com/embed/owsfdh4gxyc"  
5   allowfullscreen>  
6 </iframe>
```

single-spa

```
1 import {registerApplication, start} from 'single-spa'  
2  
3 registerApplication(  
4   // Name of our single-spa application  
5   'home',  
6   // Our loading function  
7   () => import('./src/home/home.app.js'),  
8   // Our activity function  
9   () => location.pathname === "" ||  
10    location.pathname === "/" ||  
11    location.pathname.startsWith('/home')  
12 );  
13  
14 start()
```

Der Web Components Standard

- 1 Custom Elements: Definition eigener **DOM**-Elemente
- 2 Shadow **DOM**: Teile des DOMs kapseln
- 3 ES Modules: **JS** Code dynamisch laden
- 4 **HTML** Templates: **HTML**-Fragmente laden und mehrfach verwenden

Web Components im Projekt FX

Micro Frontends in der UI von FX

Verpflichtung AAAA009901A

Übersicht

Aktivitäten ausgesetzt. Arbeitsauftrag am 28.01.2020

Verpflichteter
Du-YZojXAcyvfEeVJHE, Egon (m.)
* 01.01.1988 (32)
Steindamm 12a, 20099 Hamburg, DE

Umsätze
Saldo: 226,45 € davon Zinsen: 0,25 €
Unverrechnet: 0,00 €
Avisiert: 0,00 € Hauptkonto

€

Geplante nächste Aufgaben

Hinweise Antraggegner prüfen
28.01.2020 11:12:23

Prüfe Aussetzen beenden
28.01.2020 11:10:56

F2 Personensuche
08.01.2020 02:15:22

Ratenvereinbarung

Notizen: 0/11

Geplante nächste Aufgaben

Hinweise Antraggegner prüfen
28.01.2020 11:12:23

Prüfe Aussetzen beenden
28.01.2020 11:10:56

F2 Personensuche
08.01.2020 02:15:22

28.01.2020 MB erlassen (am 11.12.18)

28.01.2020 MB-Antrag an Gericht versendet

28.01.2020 MB gestartet

28.01.2020 GM vermutlich zugestellt

Systemzeit: 28.01.2020 16:49

Bearbeitung ▲ Kommunikation ▲ Zahlungsvereinbarungen ▲

Einbinden einer Web Component (Custom Element)

```
1 <script type="text/javascript"
2   src="https://fx/ratenvereinbarung/webComponents/rave-mini.js">
3 </script>
4 ...
5 <rave-mini verpflichtung="AAAA009901A"/>
```

```
1 import * as React from "react";
2 import * as ReactDOM from "react-dom";
3 import {RaveMini} from "./RaveMini" gb  
javascript:(function()%7Bvar  
a%3Dwindow,b%3...
4
5 class RaveMiniElement extends HTMLElement {
6     static get observedAttributes() {return ["verpflichtung"];}
7
8     drawReactComponent() {
9         ReactDOM.render(<RaveMini verpflichtung={this.getAttribute("verpflichtung")}>, this);
10    }
11
12    attributeChangedCallback() {this.drawReactComponent();}
13    connectedCallback() {this.drawReactComponent();}
14 }
15
16 customElements.define("rave-mini", RaveMiniElement);
```

Bauen einer Web Component mit Webpack

```
1 module.exports = {
2   entry: {
3     mbQuickViewElement: "./src/MbQuickViewElement.tsx",
4     testPage: "./src/testPage.ts",
5   },
6   output: {
7     filename: "build/[name].js",
8   },
9   plugins: [
10   new HtmlWebpackPlugin({
11     filename: "mbQuickViewTestPage.html",
12     inject: "body",
13     chunks: ["testPage", "mbQuickViewElement"],
14     html: <div id="container"></div>
15   }),
16 ]
17 }
```



Test-Seite für die Web Component

```
1 const {verpflichtung} = require("query-string").parse(window.location.search);
2
3 const vbQuickViewElement = document.createElement("vb-quick-view");
4 vbQuickViewElement.setAttribute("verpflichtung", verpflichtung);
5
6 document.getElementById("container").appendChild(vbQuickViewElement);
```

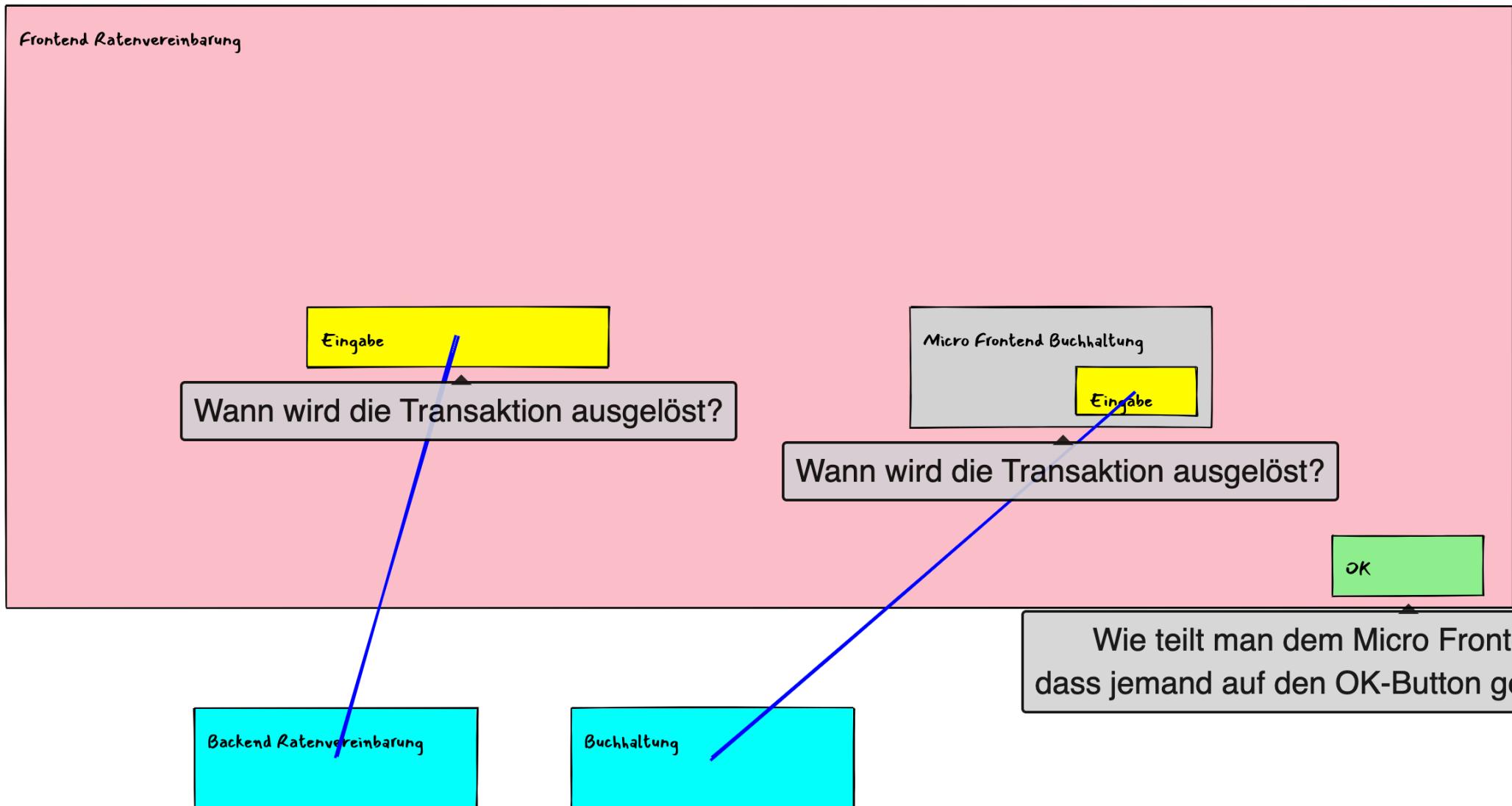
Auswirkungen & Herausforderungen

Welche Auswirkungen haben Micro Frontends auf den Entwicklungsprozess?

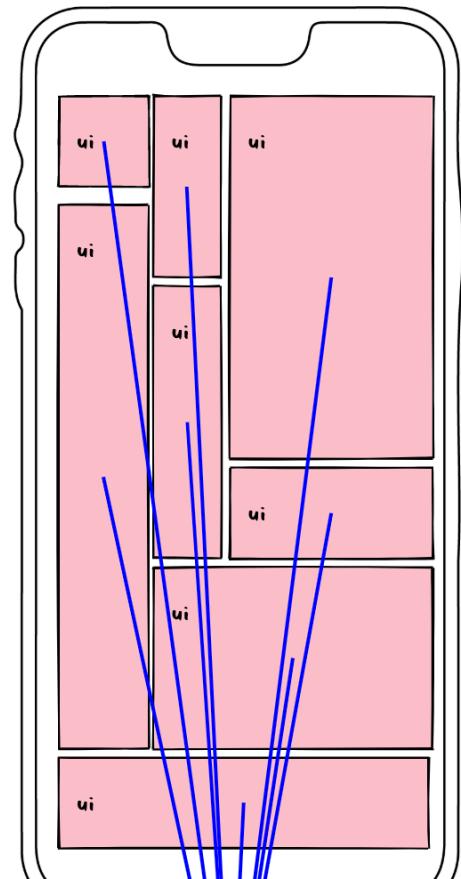
- 1 Unabhängiges Deployment
- 2 Schnittstellen
- 3 Layout ist Bestandteil der Schnittstelle
- 4 Testen
- 5 Bundle Size
- 6 Migrationspfad für Technologiewechsel
- 7 Minimierung der öffentlichen Schnittstellen
- 8 Workflow

- 1 Client Transaktionen
- 2 Anzahl der Requests

Client Transactions



Anzahl der Requests



Jedes Custom Element muss geladen werden und macht seinen eigenen XHR-Call



Fragen?



Vielen Dank

- r.gilliar@eos-ts.com
- oliver.widder@iteratec.com



Diese Folien: bit.ly/webcomp-oop