

Application Modernization – End-to-End Example

From Java EE in 2008 to Quarkus in 2021

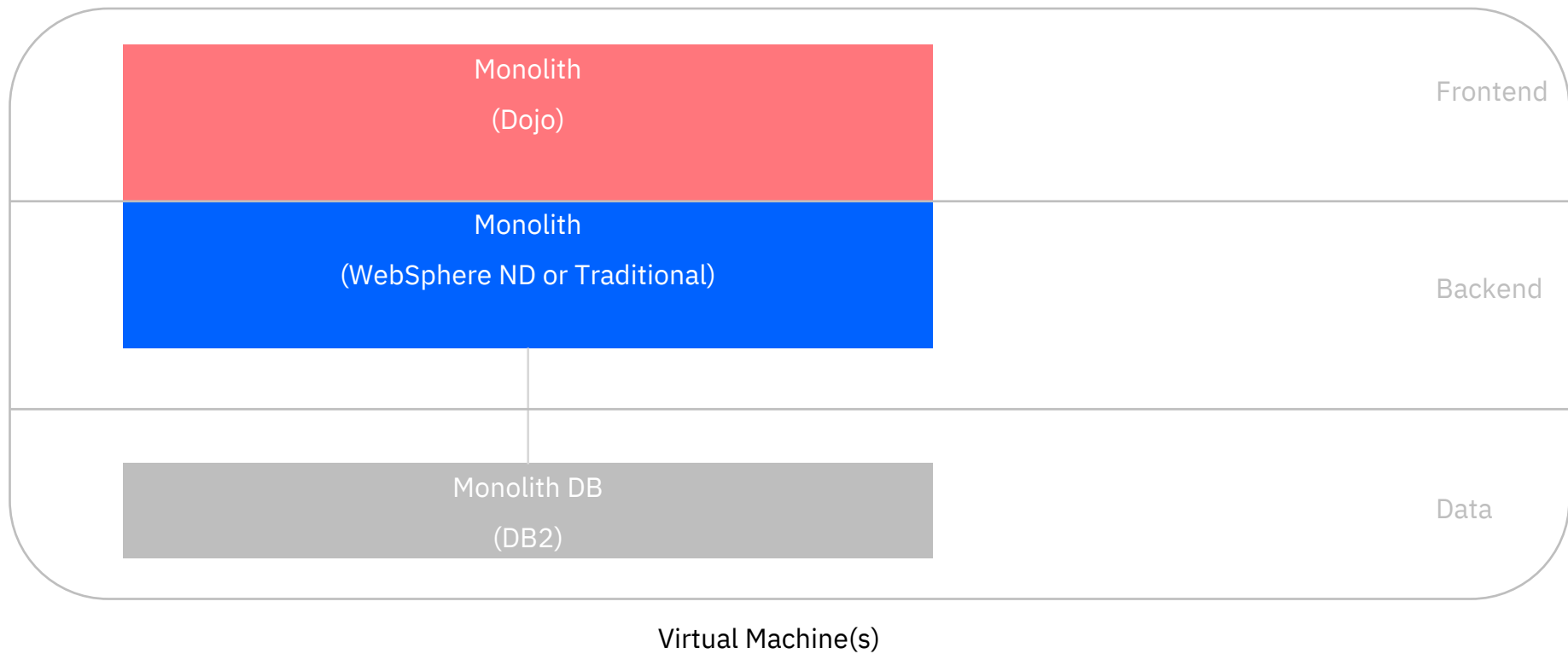
Niklas Heidloff
Developer Advocate, IBM
@nheidloff

Draft only
Work in progress ...

Step 1 / Starting Point: Monolith running in VM(s)

Description	How did we get here?	Used Technologies	Containers
Java EE 6 application Built around 2008 WebSphere 8.5.5 Typical three tier architecture No CI/CD	N/A – starting point	WebSphere ND or Traditional EJBs REST APIs (BFF, not RESTful) Db2 Dojo Toolkit	None Two Virtual Machines

Step 1 / Starting Point: Monolith running in VM(s)



Step 1 / Starting Point: Monolith running in VM(s)

Functionality:
Browse Products

Electronic and Movie Depot

The screenshot displays the 'Electronic and Movie Depot' website. At the top, there is a navigation bar with links for 'Shop', 'Cart', 'Order History', and 'Account'. On the left side, there is a sidebar with 'Entertainment' and 'Electronics' categories. The main content area is divided into two columns. The left column lists products under the heading 'Movies', including 'Star Wars' (Episode VI: Return of the Jedi, Episode V: The Empire Strikes Back, Episode IV: A New Hope) and 'Sony DVD Player'. The right column shows the 'Current Shopping Cart' with a list of items: 'Empire Strikes Back (2)', 'DVD Player (2)', 'Return of the Jedi (4)', 'BlackBerry Curve (1)', 'Sony Ericsson (1)', and 'New Hope (1)'. The total order value is 539.97.

Category	Product Name	Price
Movies	Star Wars Episode VI: Return of the Jedi	29.99
	Star Wars Episode V: The Empire Strikes Back	29.99
	Star Wars Episode IV: A New Hope	29.99
Electronics	Sony DVD Player	100

Current Shopping Cart

- Empire Strikes Back (2)
- DVD Player (2)
- Return of the Jedi (4)
- BlackBerry Curve (1)
- Sony Ericsson (1)
- New Hope (1)

Order Total: 539.97

Step 1 / Starting Point: Monolith running in VM(s)

Functionality:
Add Products to Shopping Cart

Electronic and Movie Depot

The screenshot displays the 'Electronic and Movie Depot' website. At the top, there is a navigation bar with links for 'Shop', 'Cart', 'Order History', and 'Account'. On the left side, there is a sidebar with categories 'Entertainment' and 'Electronics'. The main content area features a 'Movies' section with a 'Star Wars' category. A product card for 'Return of the Jedi' is visible, showing the title, 'Episode VI: Return of the Jedi', and a price of 50. A shopping cart overlay is present in the bottom right corner, titled 'Current Shopping Cart', which lists 'Empire Strikes Back (1)' and 'Return of the Jedi (1)' with an 'Order Total' of 59.98. A modal window titled 'Return of the Jedi' is open, showing the product details and a 'Drag Image To Cart' button. The modal also displays a 'Star Wars' category card and a 'Return of the Jedi' product card.

Shop Cart Order History Account

Entertainment ▶
Electronics ▶

Movies

Star Wars

Episode VI:
Return of the Jedi

Return of the Jedi
50

Star Wars

Current Shopping Cart

Empire Strikes Back (1)
Return of the Jedi (1)

Order Total: 59.98

Return of the Jedi

Return of the Jedi
Episode 6, Luke has the final confrontation with his father!
50

Star Wars

Episode VI:
Return of the Jedi

Drag Image To Cart

+ 1

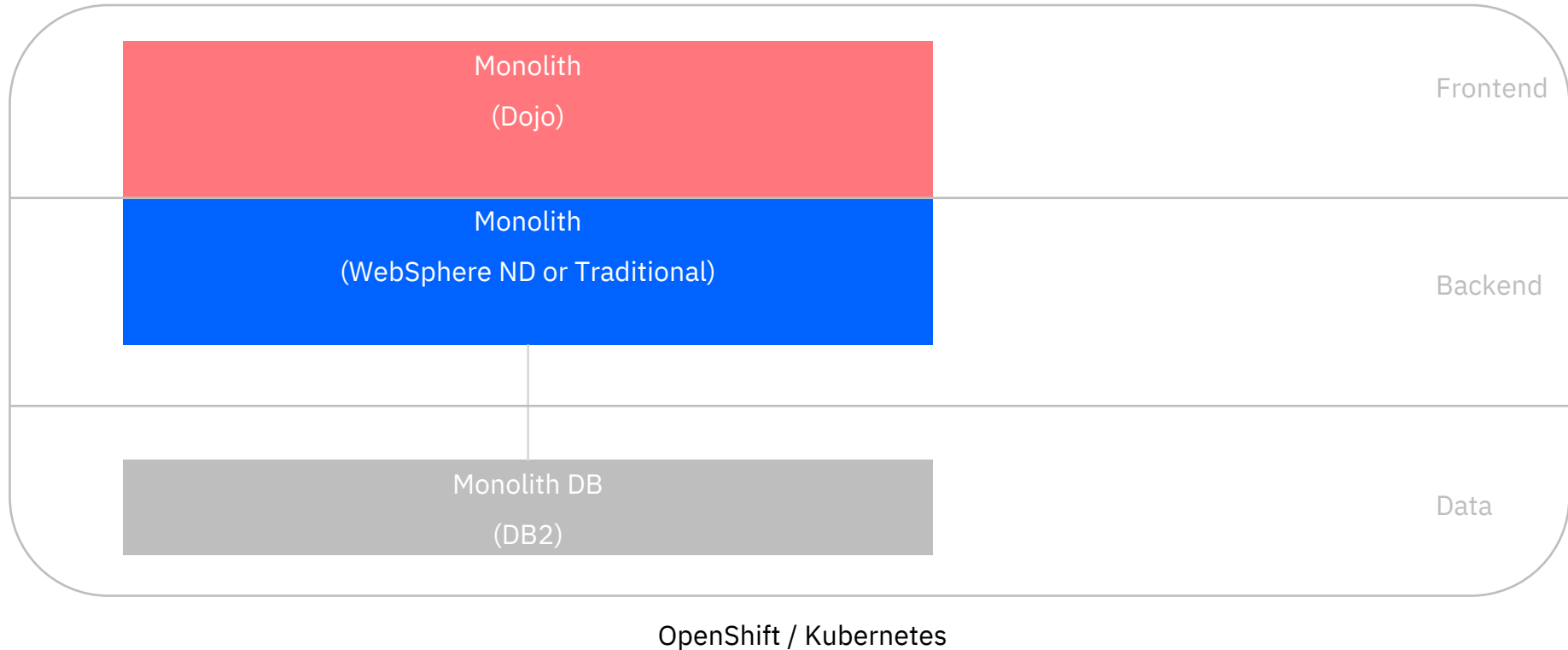
Star Wars

Episode VI:
Return of the Jedi

Step 2: Monolith running in Container on WebSphere Traditional

Description and Value	How did we get here?	Used Technologies	Containers
<p>Monolith running in Container</p> <p>-> Can run on orchestration platforms like OpenShift</p> <p>-> Consistent management of all applications</p> <p>-> CI/CD possible</p>	<p>Transformation Advisor generated Dockerfile</p> <p>Manually created Db2 container</p>	<p>WebSphere Traditional 9</p>	<p>Monolith</p> <p>Db2</p>

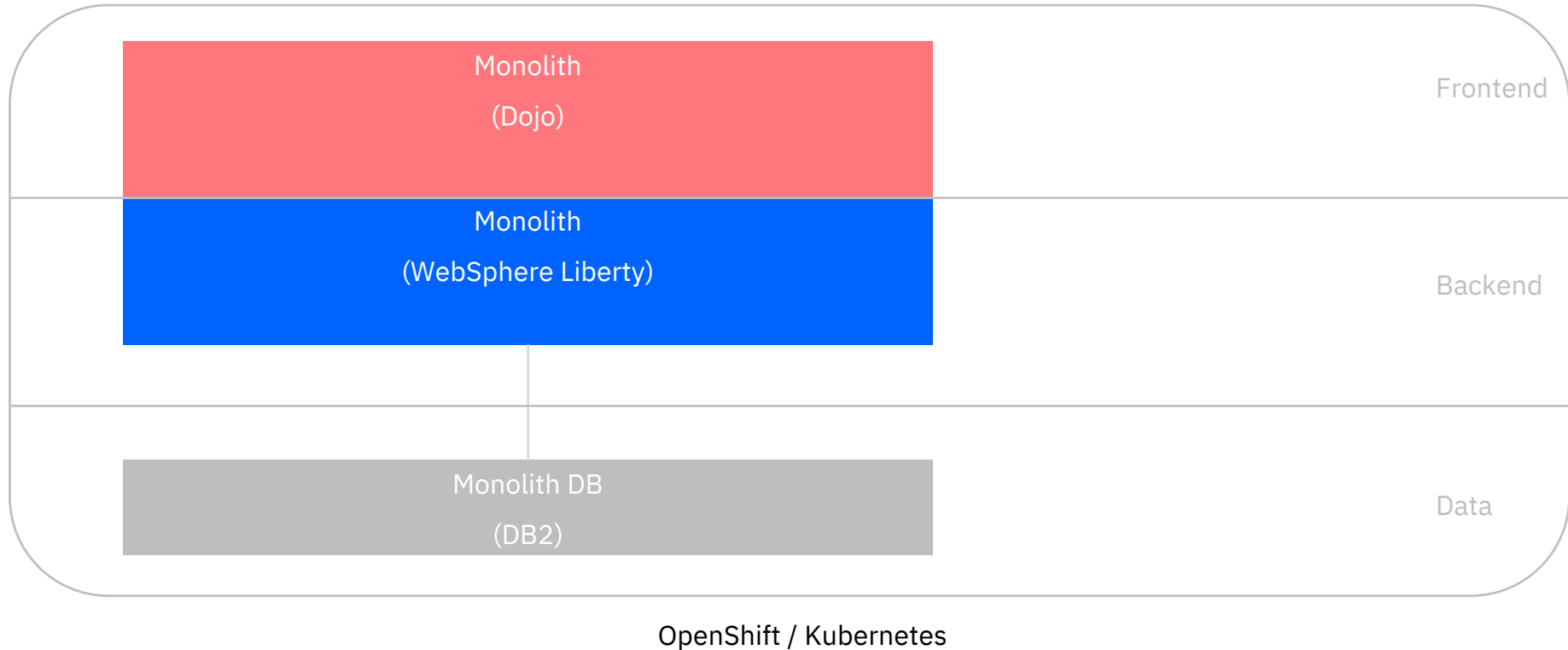
Step 2: Monolith running in Container on WebSphere Traditional



Step 3: Monolith running in Container on WebSphere Liberty

Description and Value	How did we get here?	Used Technologies	Containers
<p data-bbox="19 404 415 442">Monolith running on Liberty</p> <p data-bbox="19 524 415 600">-> More efficient runtime for container workloads</p>	<p data-bbox="492 404 869 524">Transformation Advisor generated Dockerfile and server.xml</p> <p data-bbox="492 611 869 687">Eclipse Migration Tools for code changes (?)</p>	<p data-bbox="975 404 1255 442">WebSphere Liberty</p>	<p data-bbox="1458 404 1593 442">Monolith</p> <p data-bbox="1458 464 1526 502">Db2</p>

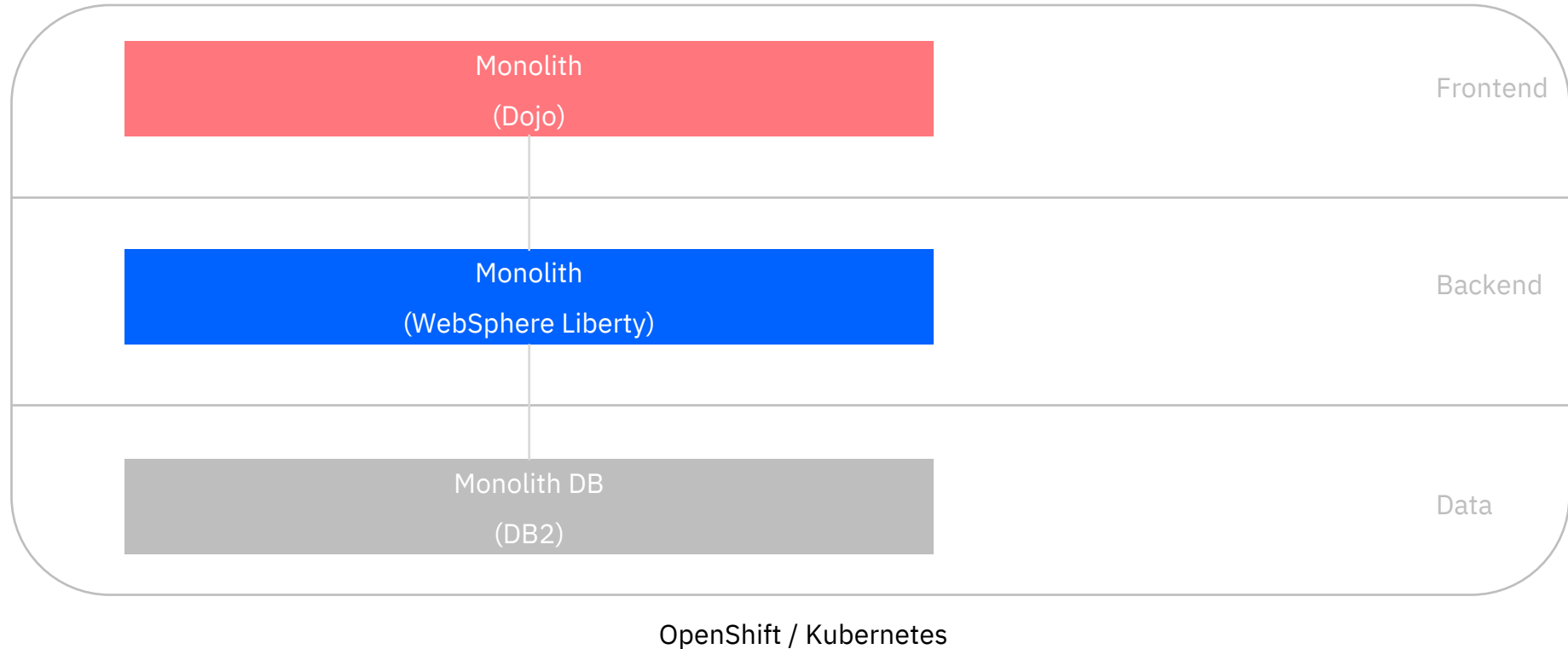
Step 3: Monolith running in Container on WebSphere Liberty



Step 4: Separated Frontend

Description and Value	How did we get here?	Used Technologies	Containers
<p>Separate container for Dojo frontend</p> <p>-> Easier extensions and potential replacement of frontend code</p>	<p>Manually</p>	<p>WebSphere Liberty</p> <p>Dojo</p>	<p>Backend monolith</p> <p>Frontend monolith</p> <p>Db2</p>

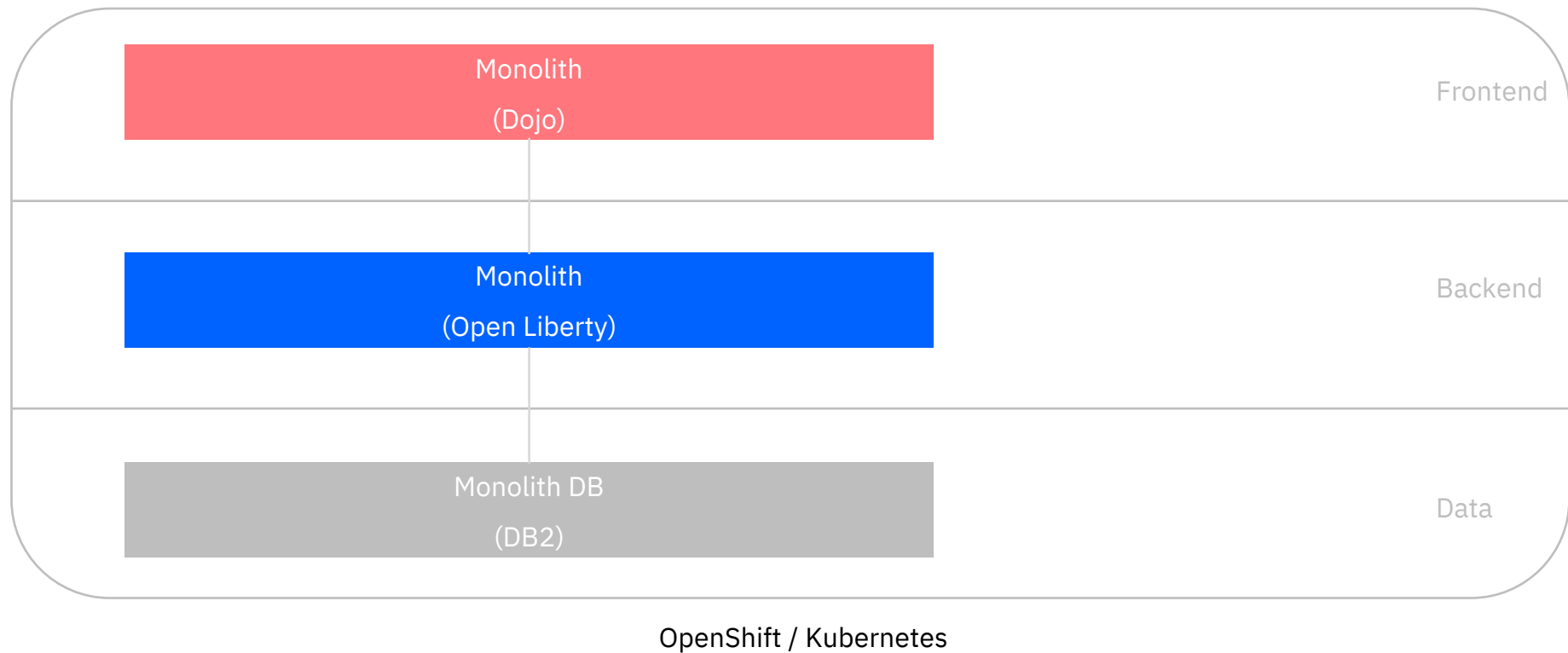
Step 4: Separated Frontend



Step 5: Open Liberty Monolith

Description and Value	How did we get here?	Used Technologies	Containers
<p>Usage of Open Liberty</p> <p>-> License cost savings</p> <p>-> Leverage latest MicroProfile and Open Liberty versions early and leverage open source communities</p>	<p>Manually (primarily changing image only)</p>	<p>Open Liberty</p> <p>MicroProfile</p>	<p>Backend monolith</p> <p>Frontend monolith</p> <p>Db2</p>

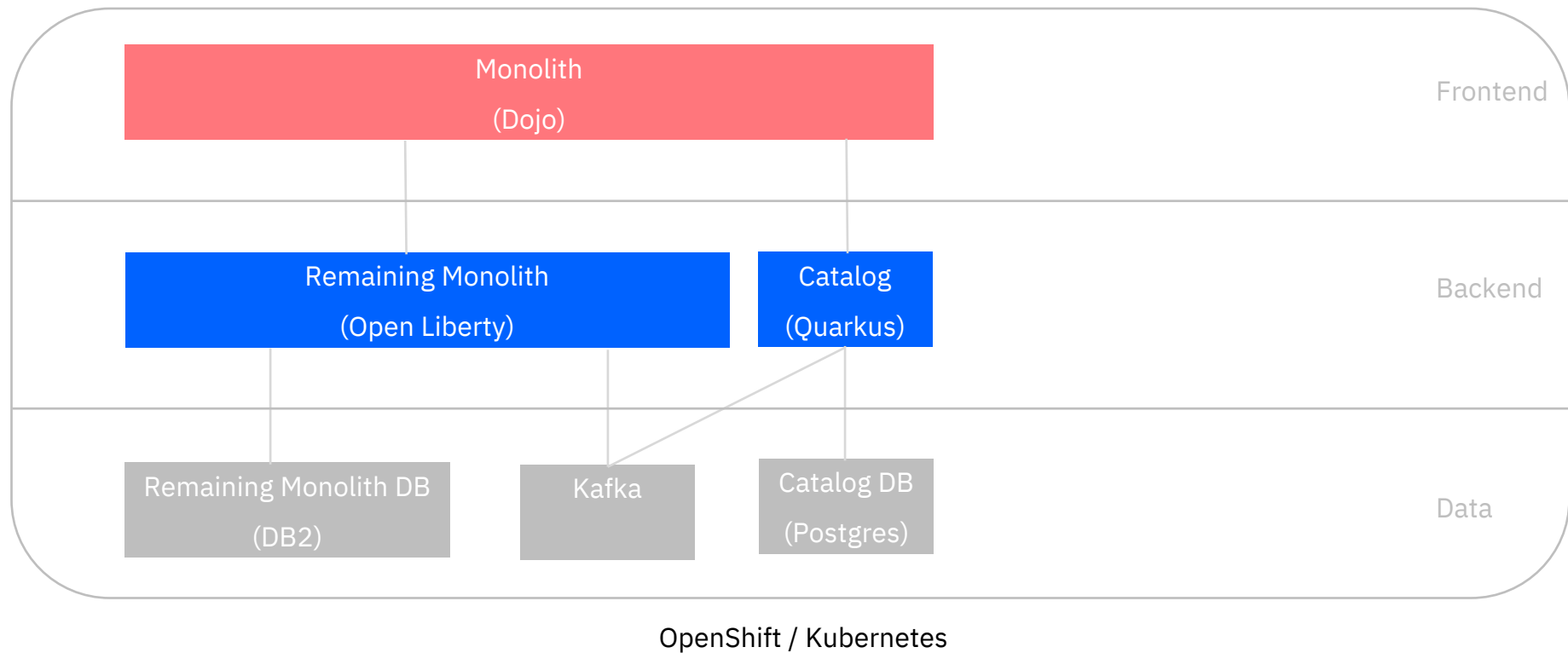
Step 5: Open Liberty Monolith



Step 6: Strangled Catalog Service and Open Liberty Monolith

Description and Value	How did we get here?	Used Technologies	Containers
<p>Strangled catalog service</p> <ul style="list-style-type: none">- Quarkus- Kafka for CQRS- Panache <p>Open Liberty monolith</p> <ul style="list-style-type: none">- Modern project structure- No EJBs, but CDI- Kafka for CQRS <p>-> Separate scalability</p>	<p>Inspired by Mono2Micro</p> <p>Manually</p>	<p>Quarkus</p> <p>OpenJ9</p> <p>Kafka</p> <p>Open Liberty</p> <p>MicroProfile</p>	<p>Backend monolith (Liberty)</p> <p>Catalog service</p> <p>Frontend monolith</p> <p>Db2</p> <p>Kafka</p> <p>Postgres</p>

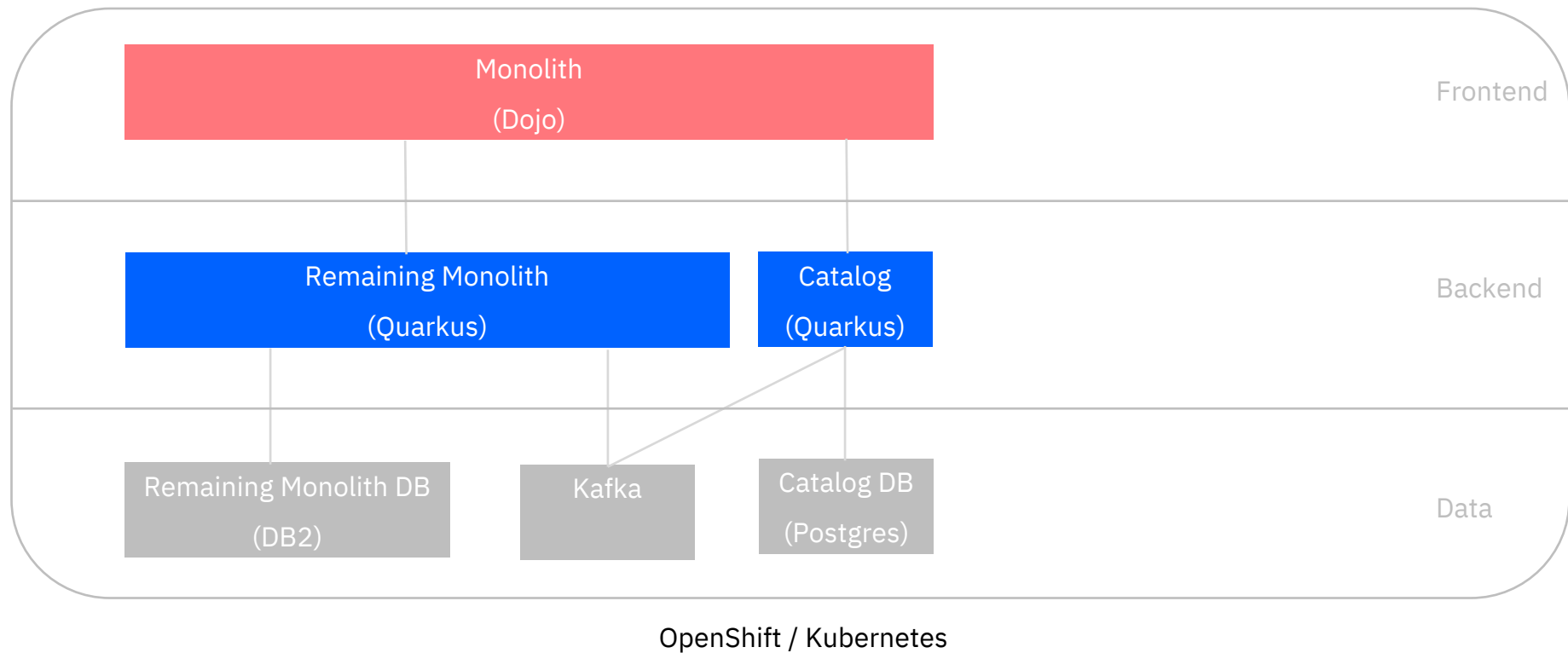
Step 6: Strangled Catalog Service and Open Liberty Monolith



Step 7: Strangled Catalog Service and Quarkus Monolith

Description and Value	How did we get here?	Used Technologies	Containers
<p>Strangled catalog service</p> <ul style="list-style-type: none">- Reactive endpoints- Reactive Postgres access <p>-> More efficient than synchronous code</p> <p>Quarkus monolith</p> <ul style="list-style-type: none">- Compiled to native binary <p>-> Faster startup time and less memory usage</p>	<p>Manually</p>	<p>Quarkus</p> <p>OpenJ9</p> <p>Kafka</p> <p>MicroProfile</p>	<p>Backend monolith (Quarkus)</p> <p>Catalog service</p> <p>Frontend monolith</p> <p>Db2</p> <p>Kafka</p> <p>Postgres</p>

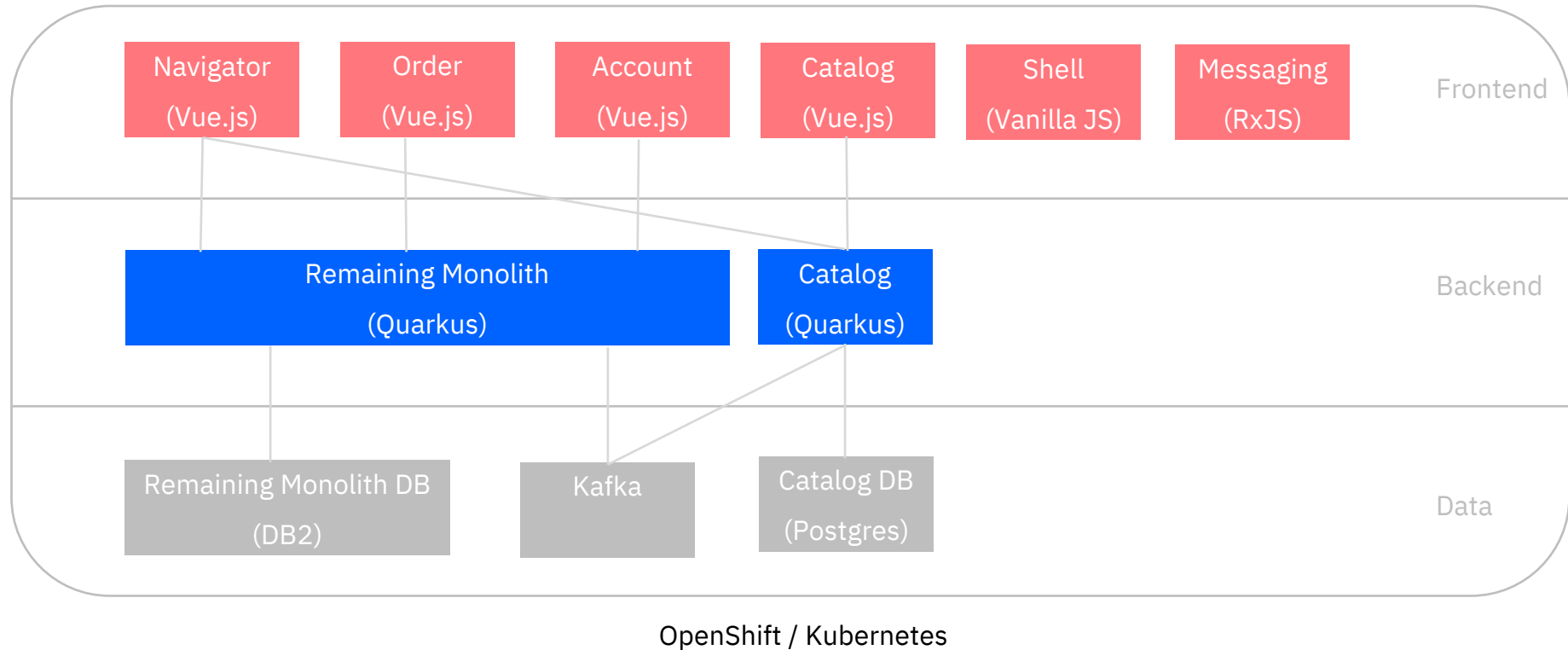
Step 7: Strangled Catalog Service and Quarkus Monolith



Step 8: Quarkus Backend and Micro Frontends

Description and Value	How did we get here?	Used Technologies	Containers
<p>Break down frontend monolith in micro frontends</p> <p>-> Independent updates to separate pages</p>	<p>Manually</p>	<p>single-spa</p> <p>Vue.js</p> <p>RxJS</p>	<p>Backend monolith (Quarkus)</p> <p>Catalog service</p> <p>Frontend monolith</p> <p>Db2</p> <p>Kafka</p> <p>Postgres</p> <p>Navigator, catalog, account, order, shell, messaging</p>

Step 8: Quarkus Backend and Micro Frontends



Micro Frontends via single-spa


Shell
(Vanilla JS)


Messaging
(RxJS)

Electronic and Movie Depot

 Catalog

Entertainment

 Movies

 Music

 Games

Electronics

 TV

 Cellphones

Navigator
(Vue.js)



Shopping



Account

Catalog - Movies

Star Wars

Episode VI:
Return of the Jedi

Return of the Jedi

Episode 6, Luke has the final confrontation with his fa...

29.99

Star Wars

Episode V:
The Empire Strikes Back

Empire Strikes Back

Episode 5, Luke finds out a secret that will change hi...

29.99

Star Wars

Episode IV:
A New Hope

New Hope

Episode 4, after years of oppression, a band of rebels...

29.99

Catalog
(Vue.js)

Order
(Vue.js)

Account
(Vue.js)

Micro Frontends via single-spa

Navigator
(Vue.js)

Catalog
(Vue.js)

Order
(Vue.js)

Account
(Vue.js)

Shell
(Vanilla JS)

```
import { registerApplication, start } from "single-spa";

registerApplication({
  name: "@vue-app-mod/navigator",
  app: () => System.import("@vue-app-mod/navigator"),
  activeWhen: ["/"],
});

registerApplication({
  name: "@vue-app-mod/order",
  app: () => System.import("@vue-app-mod/order"),
  activeWhen: ["/", "/catalog", "/order"],
});

registerApplication({
  name: "@vue-app-mod/account",
  app: () => System.import("@vue-app-mod/account"),
  activeWhen: ["/account"],
});

registerApplication({
  name: "@vue-app-mod/catalog",
  app: () => System.import("@vue-app-mod/catalog"),
  activeWhen: ["/", "/catalog"],
});

|
start();
```

Messaging
(RxJS)

```
export default {
  ...
  getObservable(microFrontendName) {
    switch (microFrontendName) {
      case MICRO_FRONTEND_NAVIGATOR:
        if (!observableForNavigator) {
          observableForNavigator = new Subject()
        }
        return observableForNavigator
      ...
    }
  },
  send(message) {
    let topic = message.topic
    let commandId = message.commandId

    switch (topic) {
      case TOPIC_COMMAND_ADD_ITEM:
        if (observableForOrder) {
          observableForOrder.next(message)
        }
        break
      ...
    }
  }
}
```

Micro Frontends via single-spa

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta name="importmap-type" content="systemjs-importmap" />
  <script type="systemjs-importmap">
    {
      "imports": {
        "single-spa": "https://cdn.jsdelivr.net/npm/single-spa@5.1.1/lib/system/single-spa.min.js",
        "vue": "https://cdn.jsdelivr.net/npm/vue@2.6.11/dist/vue.min.js",
        "vue-router": "https://cdn.jsdelivr.net/npm/vue-router@3.1.6/dist/vue-router.min.js",
        "rxjs": "http://cdnjs.cloudflare.com/ajax/libs/rxjs/6.6.3/rxjs.umd.min.js",

        "@vue-app-mod/shell": "http://localhost:8080/vue-app-mod-shell.js",
        "@vue-app-mod/messaging": "http://localhost:9001/vue-app-mod-messaging.js",
        "@vue-app-mod/navigator": "http://localhost:8501/js/app.js",
        "@vue-app-mod/order": "http://localhost:8504/js/app.js",
        "@vue-app-mod/catalog": "http://localhost:8503/js/app.js",
        "@vue-app-mod/account": "http://localhost:8502/js/app.js"
      }
    }
  </script>
  <script src="https://cdn.jsdelivr.net/npm/import-map-overrides/dist/import-map-overrides.js"></script>
  <script src="https://cdn.jsdelivr.net/npm/systemjs/dist/system.js"></script>
  <script src="https://cdn.jsdelivr.net/npm/systemjs/dist/extras/amd.js"></script>
  <script src="https://cdn.jsdelivr.net/npm/systemjs/dist/extras/named-exports.js"></script>
  <link href="https://unpkg.com/material-components-web@latest/dist/material-components-web.min.css" rel="stylesheet">
  <script src="https://unpkg.com/material-components-web@latest/dist/material-components-web.min.js"></script>
  <link rel="stylesheet" href="https://fonts.googleapis.com/icon?family=Material+Icons">
</head>
<body>
  <div style="display: flex; flex-direction: row;">
    <div id="navigator" style="order: 1; flex-grow: 2; max-width: 300px; width: 300px"></div>
    <div id="catalog" style="order: 2; flex-grow: 10; left: 320px; position: fixed;"></div>
    <div id="order" style="order: 2; flex-grow: 10; left: 320px; position: fixed;"></div>
    <div id="account" style="order: 2; flex-grow: 10; left: 320px; position: fixed;"></div>
  </div>
  <script>
    System.import('@vue-app-mod/shell');
  </script>
  <import-map-overrides-full show-when-local-storage="devtools" dev-libs></import-map-overrides-full>
</body>
</html>
```

Catalog - Movies

Entertainment

Movies

Music

Games

Electronics

TV

Cellphones

Star Wars

Return of the Jedi

Episode 6, Luke has the final confrontation with his fa...

29.99

Star Wars

Empire Strikes Back

Episode 5, Luke finds out a secret that will change hi...

29.99

Apprentice, a band of rebels...

29.99

override. See documentation for more info.

@vue-app-mod/navigator

Default URL: <http://localhost:8501/js/app.js>Override URL:

Cancel

Reset to default

Search modules

Add new module

Add import map

Reset all overrides

Module Status	Module Name	Domain	Filename
Inline Override	single-spa	cdn.jsdelivr.net	single-spa.dev.js
Inline Override	vue	cdn.jsdelivr.net	vue.js
Inline Override	vue-router	cdn.jsdelivr.net	vue-router.js
Default	rxjs	cdnjs.cloudflare.com	rxjs.umd.min.js
Default	@vue-app-mod/shell	localhost:8080	vue-app-mod-shell.js
Default	@vue-app-mod/messaging	localhost:9001	vue-app-mod-messaging.js
Default	@vue-app-mod/navigator	localhost:8501	app.js

Elements Console Sources Network Performance Memory >> X

Preserve log Disable cache Online

Filter Hide data URLs All XHR JS CSS Img Media Font Doc WS Manifest Other

Has blocked cookies Blocked Requests

50 ms 100 ms 150 ms 200 ms 250 ms

Name	S...	Domain	Type	Initiator	Size	T...	Waterfall
catalog	304	localhost	docu...	Other	431 B	6...	
import-map-overrides.js	200	cdn.jsdelivr...	script	catalog	(mem...	0...	
system.js	200	cdn.jsdelivr...	script	catalog	(mem...	0...	
amd.js	200	cdn.jsdelivr...	script	catalog	(mem...	0...	
named-exports.js	200	cdn.jsdelivr...	script	catalog	(mem...	0...	
material-components-web.mi...	302	unpkg.com	styles...	catalog	70 B	3...	
material-components-web.mi...	302	unpkg.com	script...	catalog	94 B	2...	
icon?family=Material+Icons	200	fonts.go...	styles...	catalog	(disk ...	2...	
material-components-web.mi...	200	unpkg.com	script	material-co...	(disk ...	2...	
material-components-web.mi...	200	unpkg.com	styles...	material-co...	(disk ...	3...	
vue-app-mod-shell.js	200	localhost	script	system.js:654	(disk ...	1...	
fIUhRq6tzZciQEJ-Vdg-luiaDs...	200	fonts.gst...	font	icon?family...	(mem...	0...	
single-spa.min.js	200	cdn.jsdelivr...	script	system.js:654	(disk ...	1...	
app.js	200	localhost	script	system.js:654	(disk ...	9...	
app.js	200	localhost	script	system.js:654	(disk ...	2...	
app.js	200	localhost	script	system.js:654	(disk ...	3...	
vue-router.min.js	200	cdn.jsdelivr...	script	system.js:654	(disk ...	1...	
vue.min.js	200	cdn.jsdelivr...	script	system.js:654	(disk ...	3...	
vue-app-mod-messaging.js	200	localhost	script	system.js:654	(disk ...	3...	
?categoryId=2	200	localhost	fetch	Home.vue:1...	640 B	1...	
Category	200	localhost	fetch	App.vue:113	499 B	1...	
Orders	200	localhost	fetch	App.vue:82	550 B	1...	
Return.jpg	200	localhost	jpeg	vue_min.js:6	(mem...	0...	
Empire.jpg	200	localhost	jpeg	vue_min.js:6	(mem...	0...	
NewHope.jpg	200	localhost	jpeg	vue_min.js:6	(mem...	0...	

25 requests 2.3 kB transferred 2.0 MB resources Finish: 235 ms DOMContentLoaded: 80 ms Load: 230 ms

Micro Frontends – Messaging via RxJS for loosely Coupling

Command Pattern – Example: Add Item to Cart

Catalog -> Order

messaging - messaging.js - send invoked - topic: **TOPIC_COMMAND_ADD_ITEM** commandId: messaging.js:49
1611049822817

▶ {__ob__: we} messaging.js:50

order - App.vue - created - message: App.vue:20

Topic: TOPIC_COMMAND_ADD_ITEM App.vue:21

CommandId: 1611049822817 App.vue:22

ProductId: 1 App.vue:23

Product has been added to order App.vue:47

Products in shopping cart: 1 App.vue:54

Line items in shopping cart: 2 App.vue:55

Order -> Navigator

messaging - messaging.js - send invoked - topic: **TOPIC_EVENT_AMOUNT_LINE_ITEMS_CHANGED** commandId: undefined messaging.js:49

▶ {topic: "TOPIC_EVENT_AMOUNT_LINE_ITEMS_CHANGED", payload: {...}} messaging.js:50

Order -> Catalog

navigator - App.vue - amountLineItems: 2 App.vue:74

messaging - messaging.js - send invoked - topic: **TOPIC_COMMAND_RESPONSE_ADD_ITEM** commandId: undefined messaging.js:49

▶ {topic: "TOPIC_COMMAND_RESPONSE_ADD_ITEM", payload: {...}} messaging.js:50

catalog - Home.vue - message: Home.vue:69

Appendix

lorem

lorem

“lorem”

lorem

@nheidloff

lorem

lorem

lorem

lorem

lorem

IBM