WEBINAR

Unleashing the power of frontend distributed systems with AWS

Harun Hasdal

Sr. Solutions Architect
AWS

Warren Fitzpatrick

Principal Engineer
Dunelm

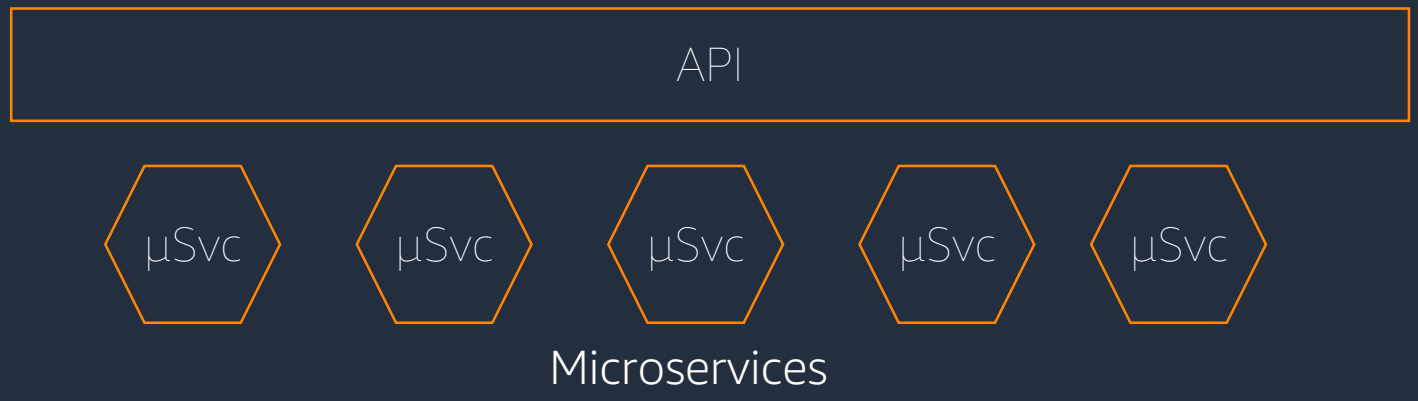
© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Are you experiencing **long lead times** to release changes to your users? even though you have **microservices architecture**?

Have you got too many frontend **team dependencies**?

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Microservices



Team **autonomy** and **ownership**

Organized by **business domains**

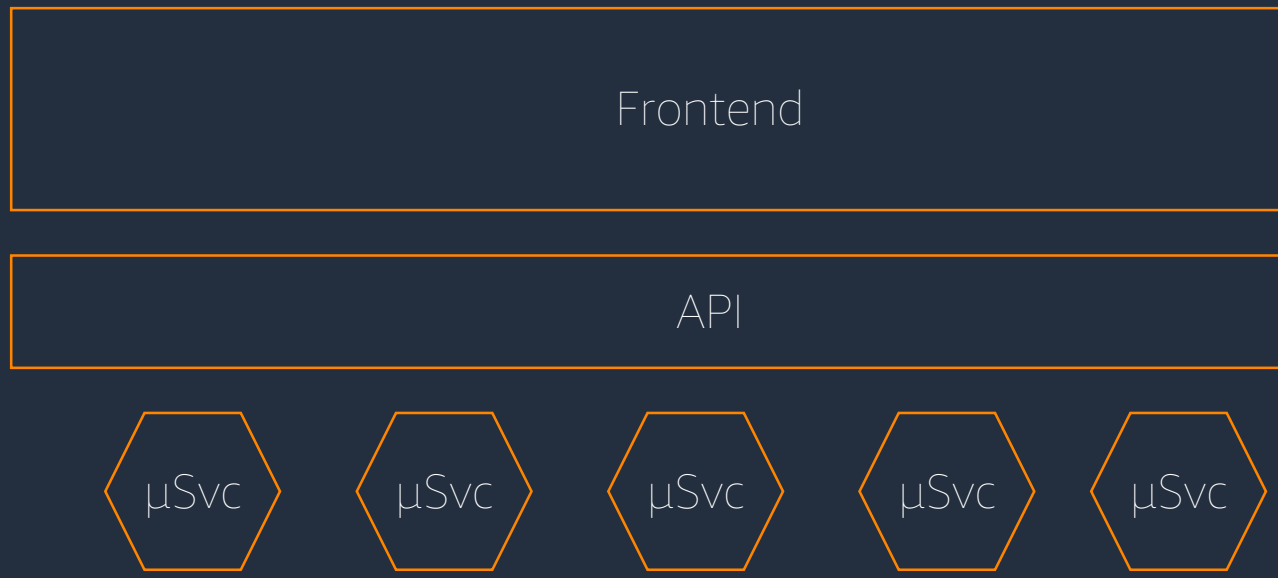
Independent deployment

Loose coupling

Failure Isolation / **Resiliency**

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Microservices with a monolithic frontend



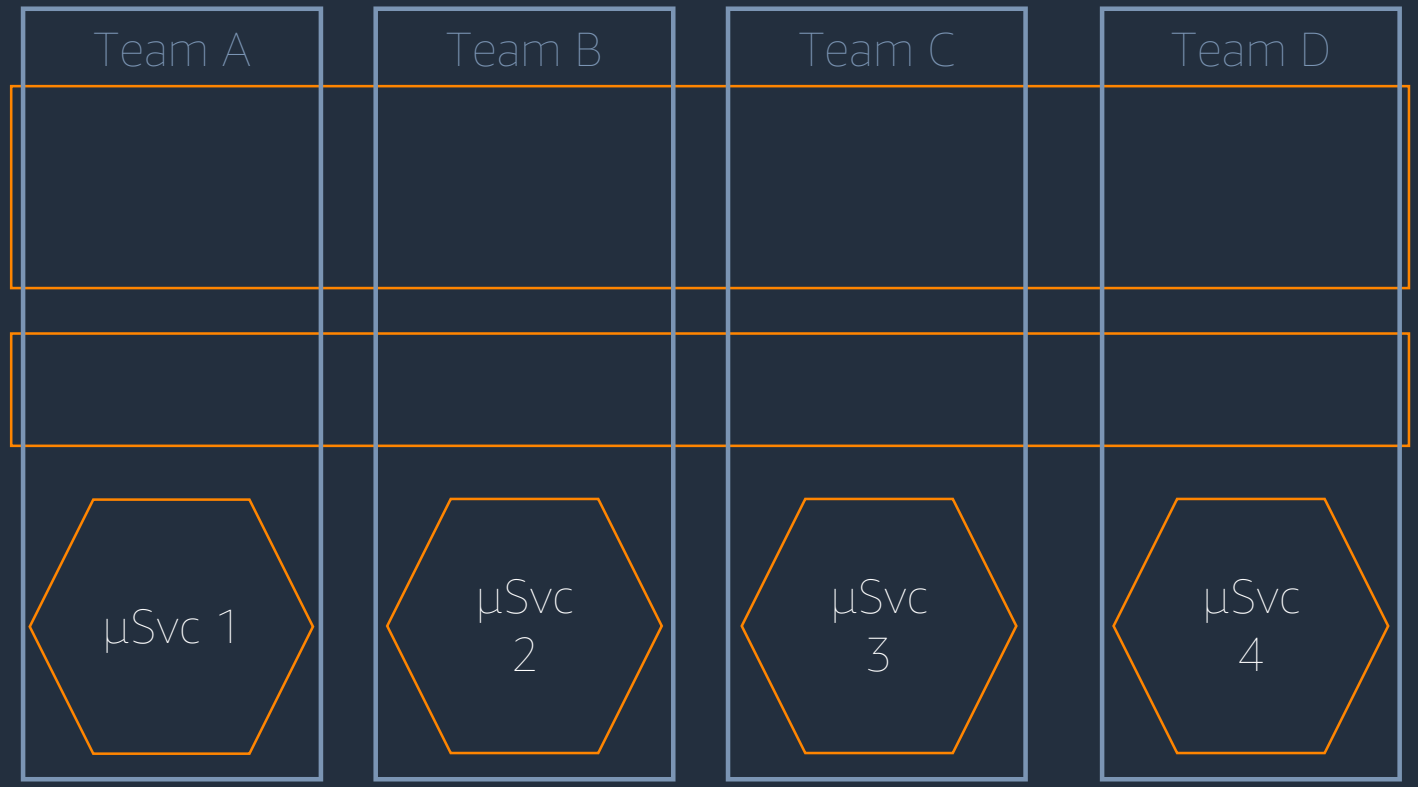
Coordinated changes

High **coupling**

No isolation / **low resiliency**

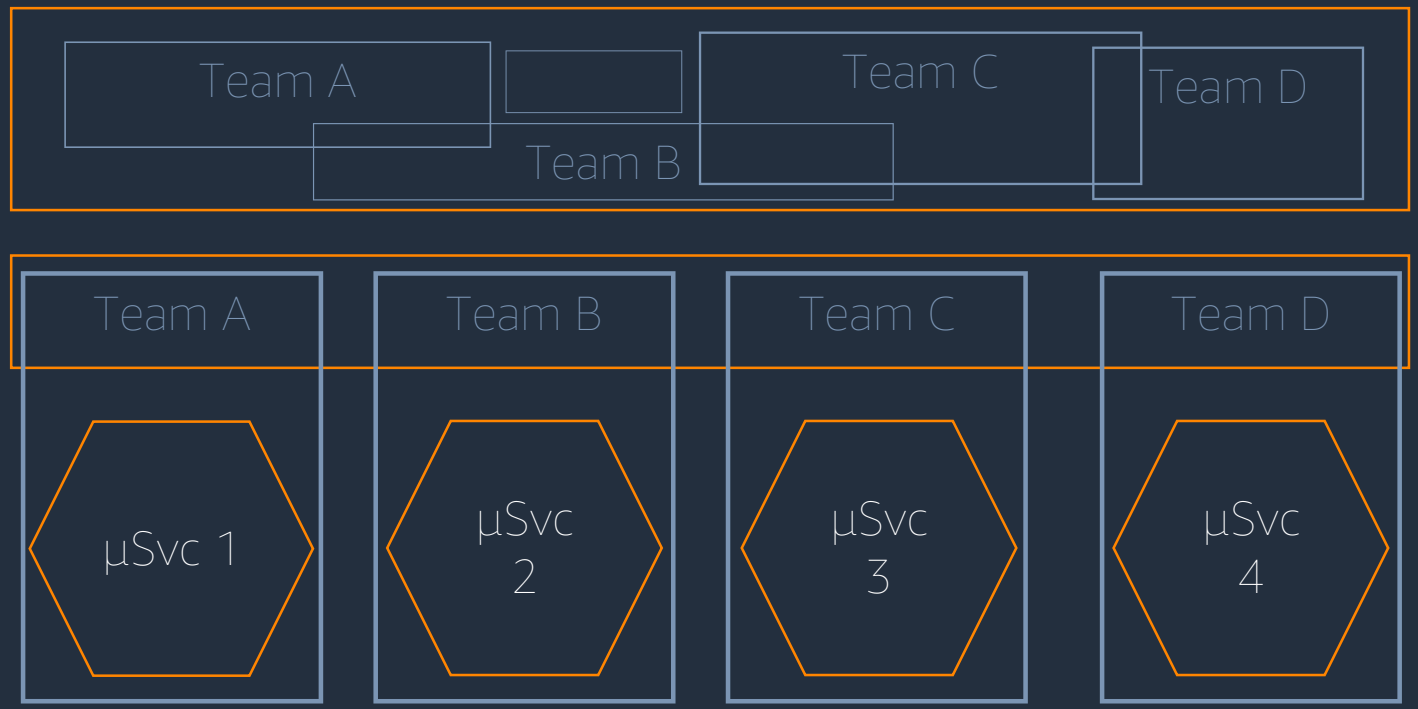
© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Ownership expectations



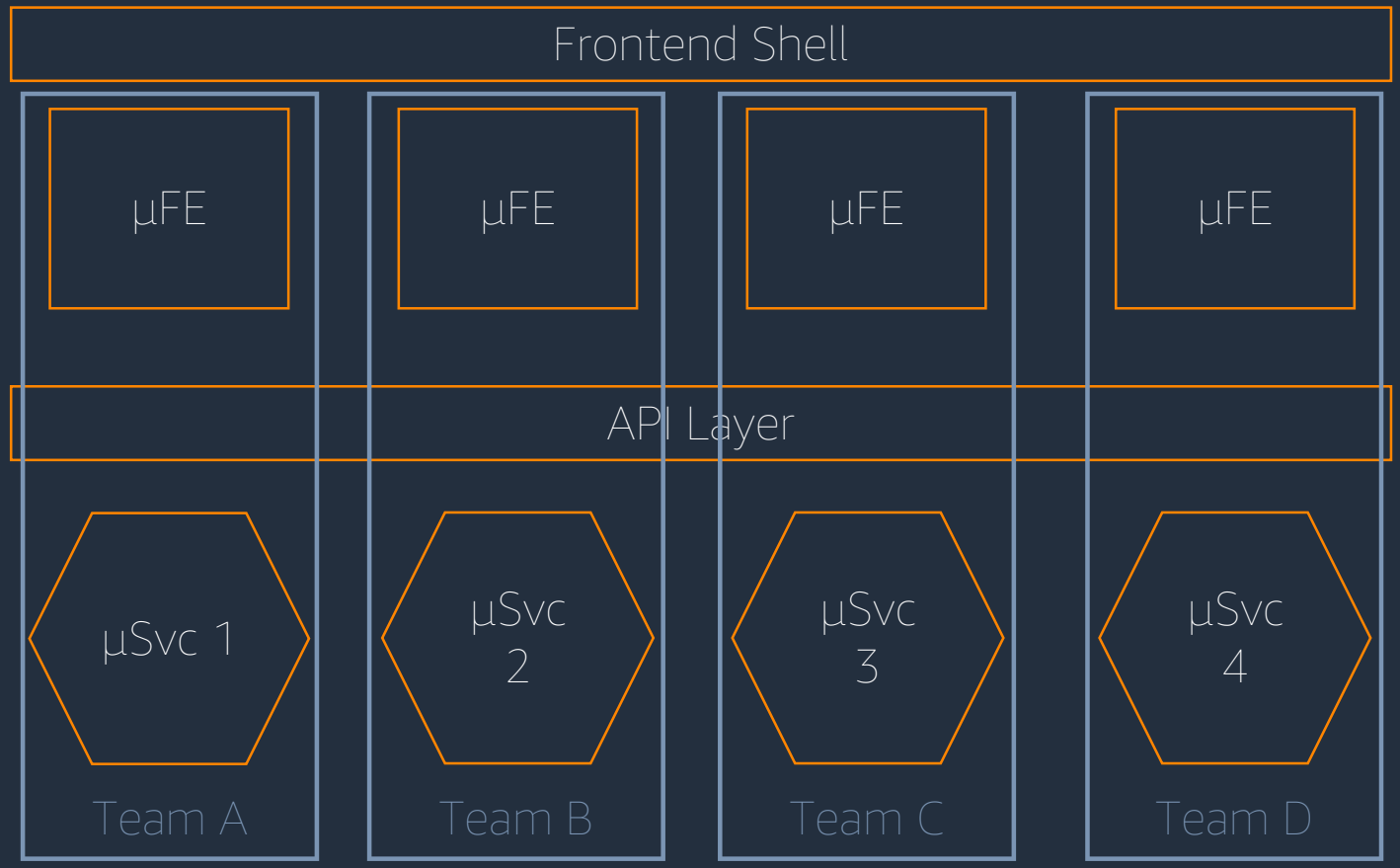
© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

No clear boundaries in the frontend



© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Microfrontends



Microfrontends are:

- **independently deployable**
- providing features of a **business domain**
- **loosely coupled** with the rest of the application
- owned by a ***single*** **team**

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Do you need microfrontends?

Organization context

Application Footprint, # Teams

Optimize key metrics

Web Vitals, Deployment Frequency, Lead Time for Changes

Evaluate trade-offs

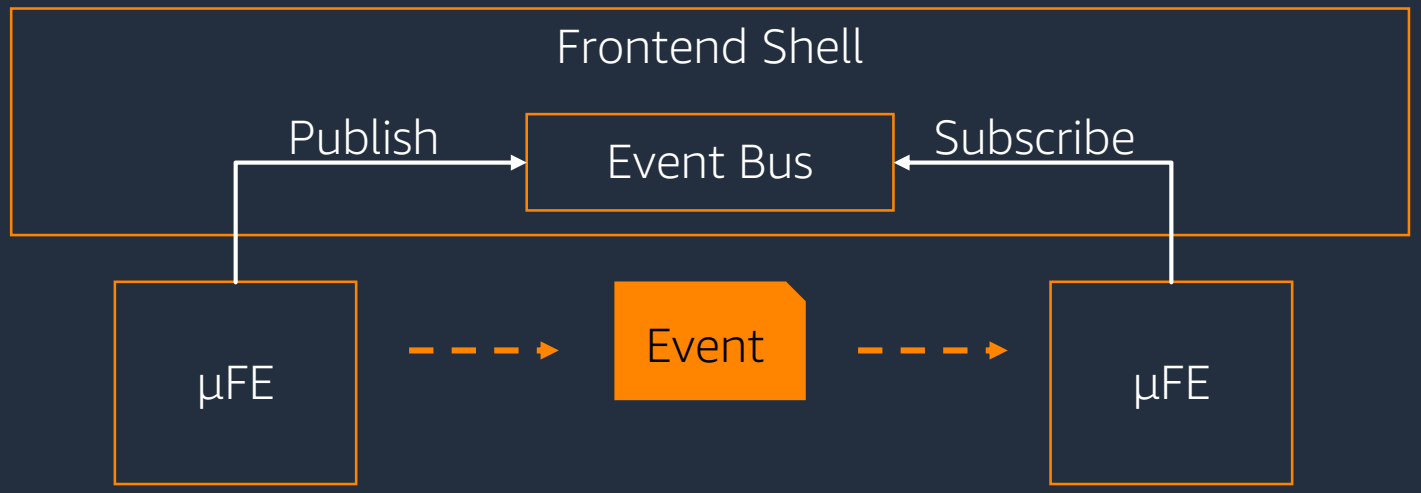
Business Agility vs Complexity

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Distributed System Patterns applied to Microfrontends

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Microfrontends – Communicate

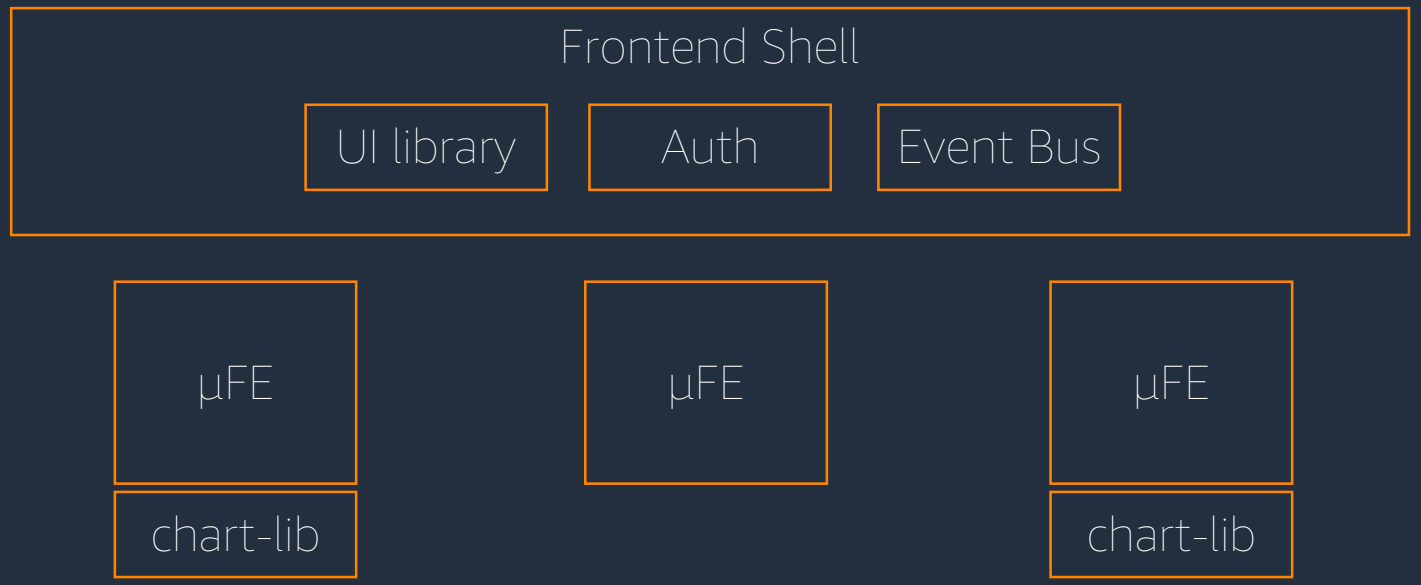


Events provide **well defined contracts** between teams.

No direct integration -> **Reduced coupling**.

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Microfrontends – To Share or Not to share

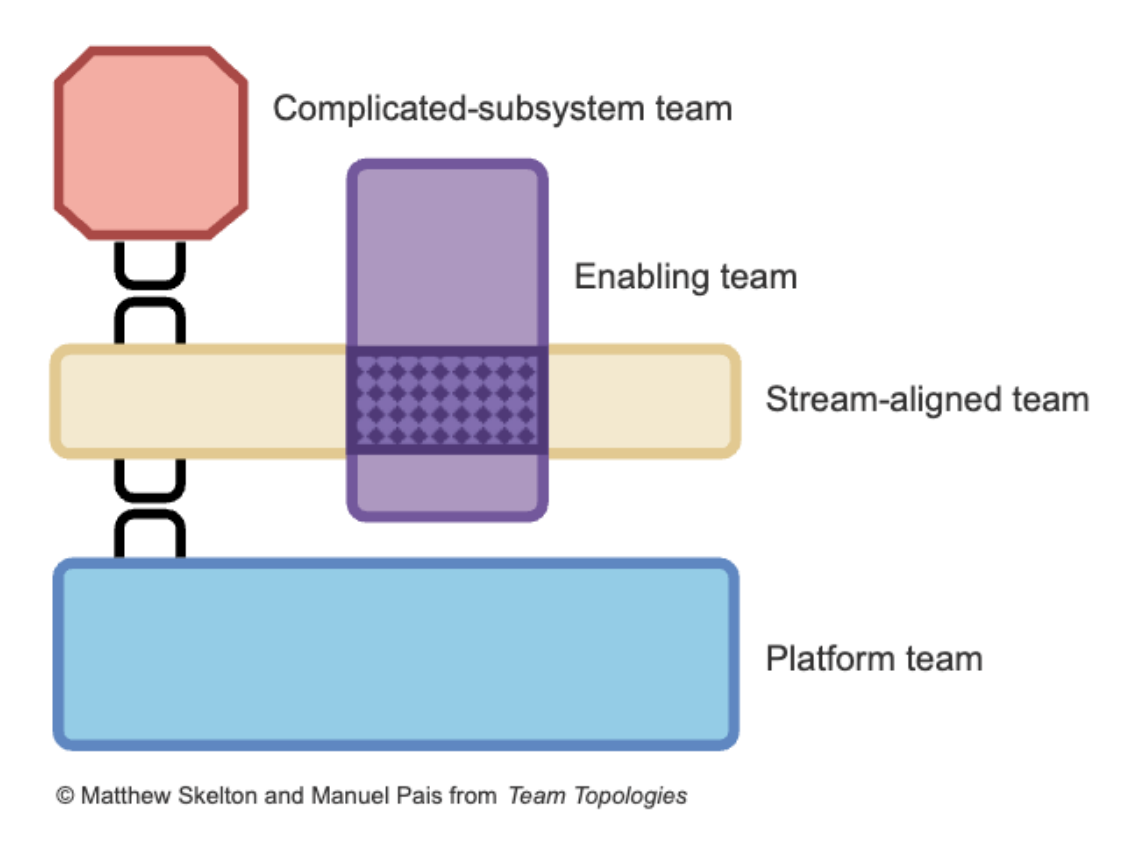


Share nothing if possible to minimize **team dependencies** and runtime coupling.

Trade-offs for **consistency**, **productivity**, and **performance.**

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

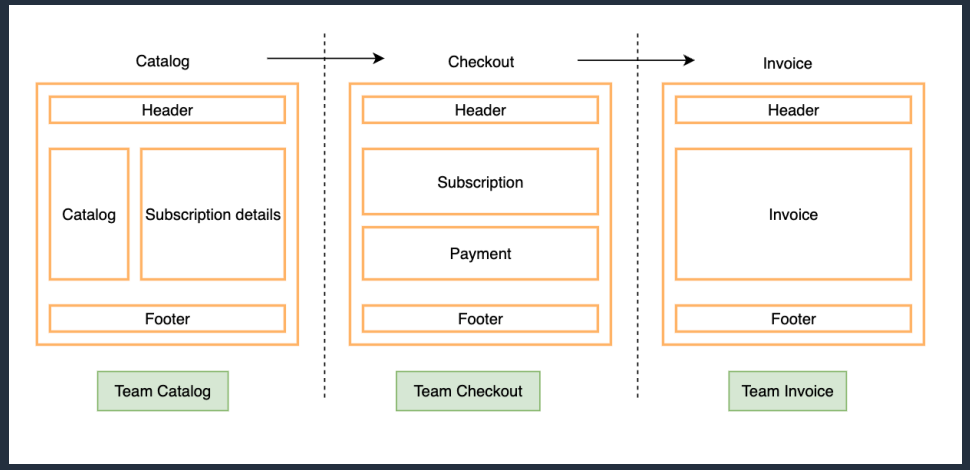
Microfrontends – Team organization patterns

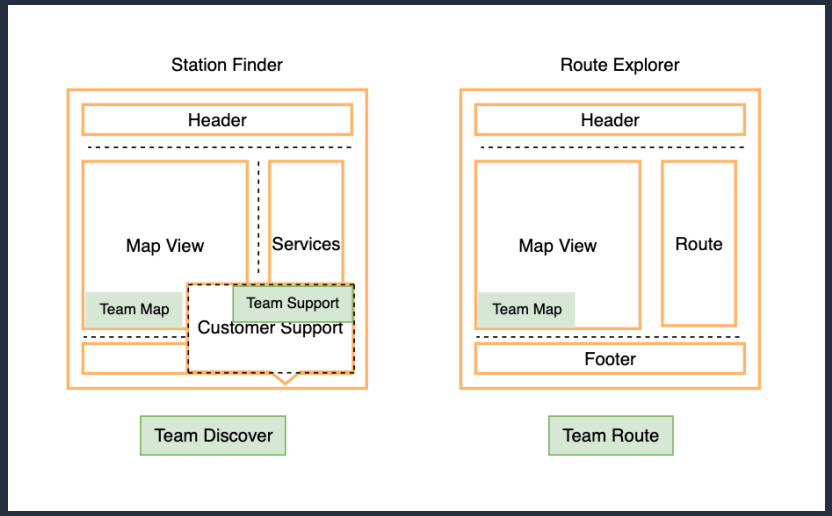


Trade-off some **autonomy** to gain **efficiencies at scale** with platform teams and **enabler teams**.

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Defining boundaries at the user interface





Find the boundaries with **Domain Driven Design** techniques.

Work with **UX**.

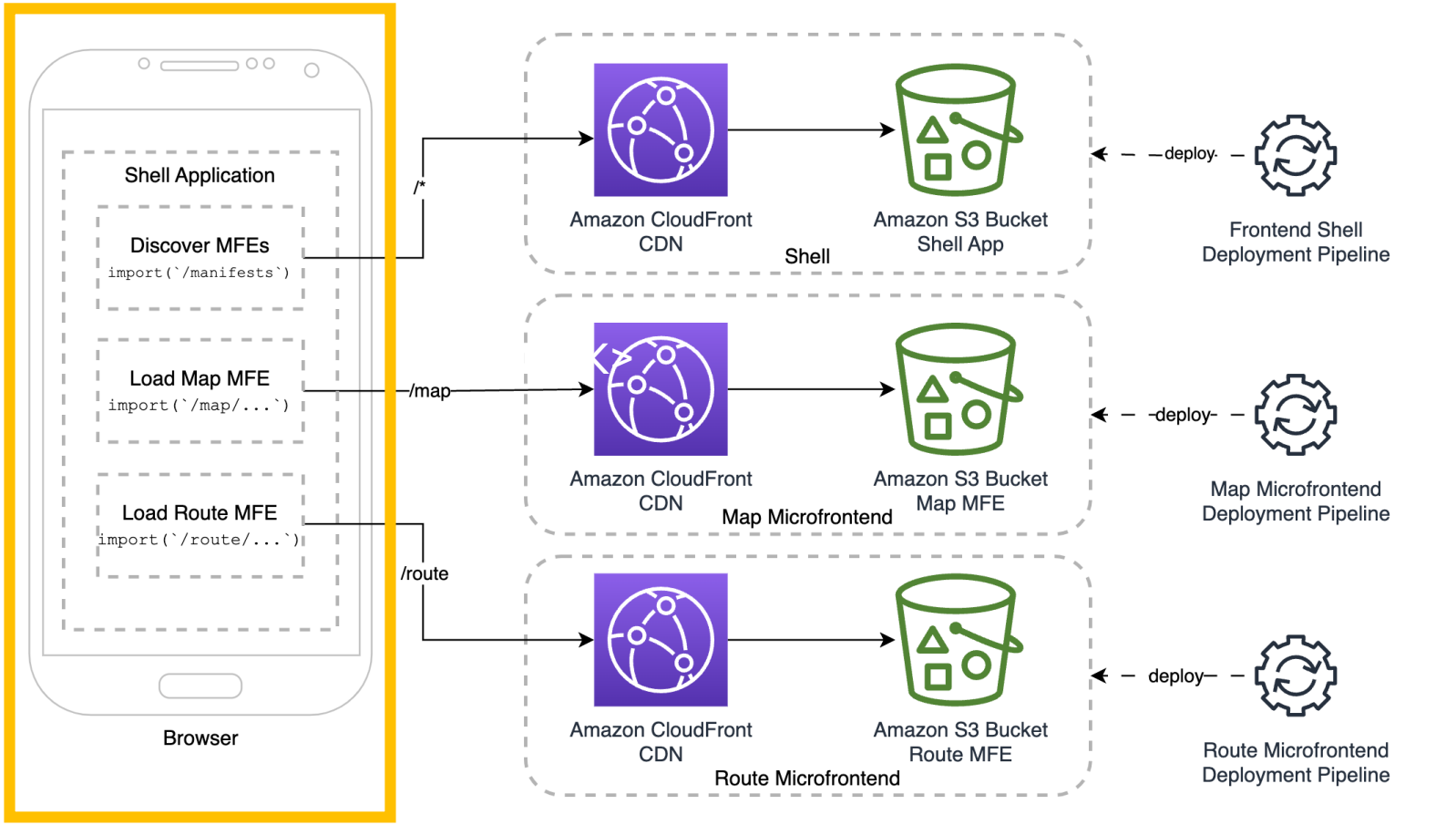
Aim to make it possible to **evolve** your application and **scale your development effort**.

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Microfrontends on AWS

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

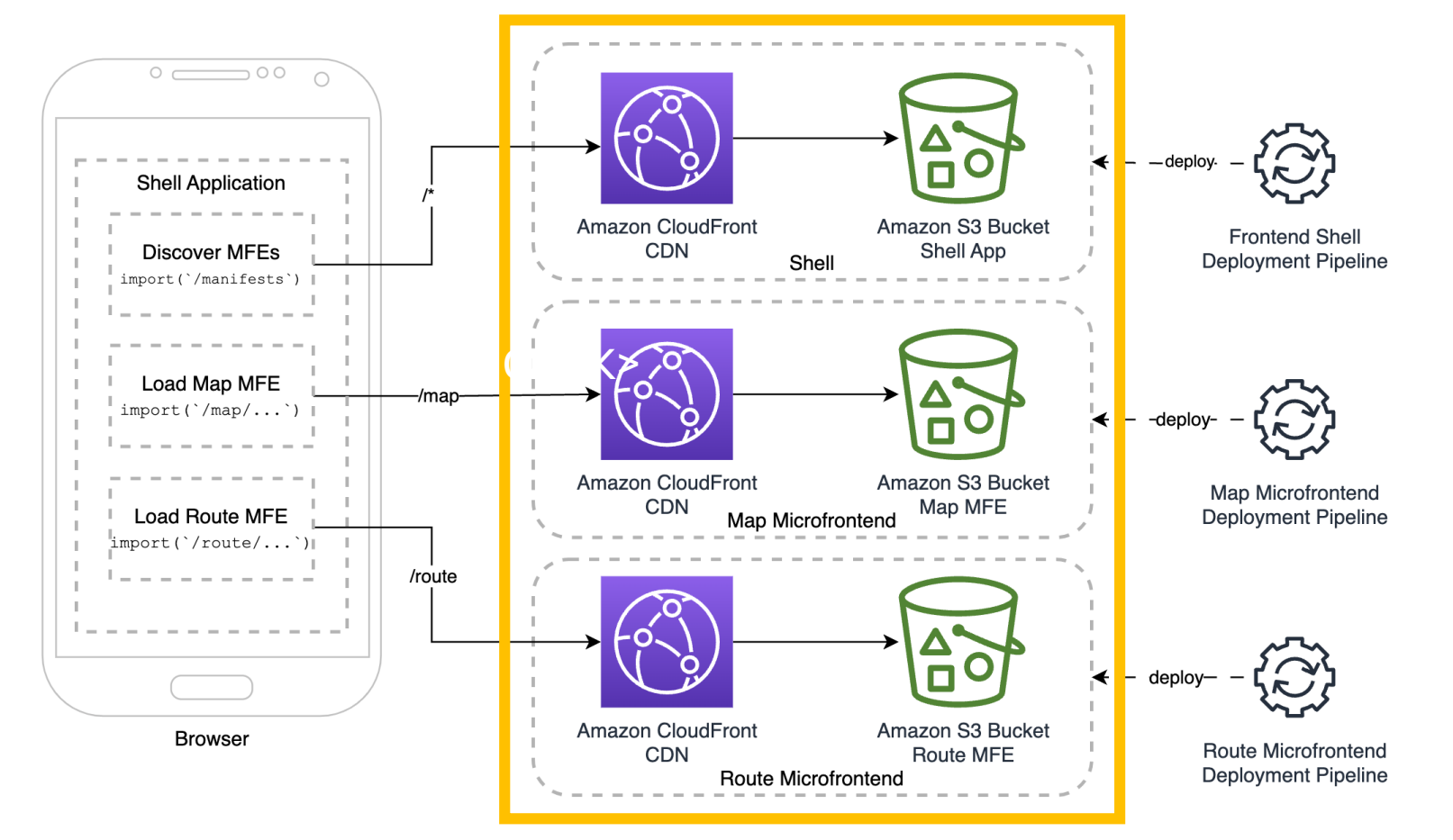
Example - Client side composition



Discovery on the client, via loading manifests / import maps.

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

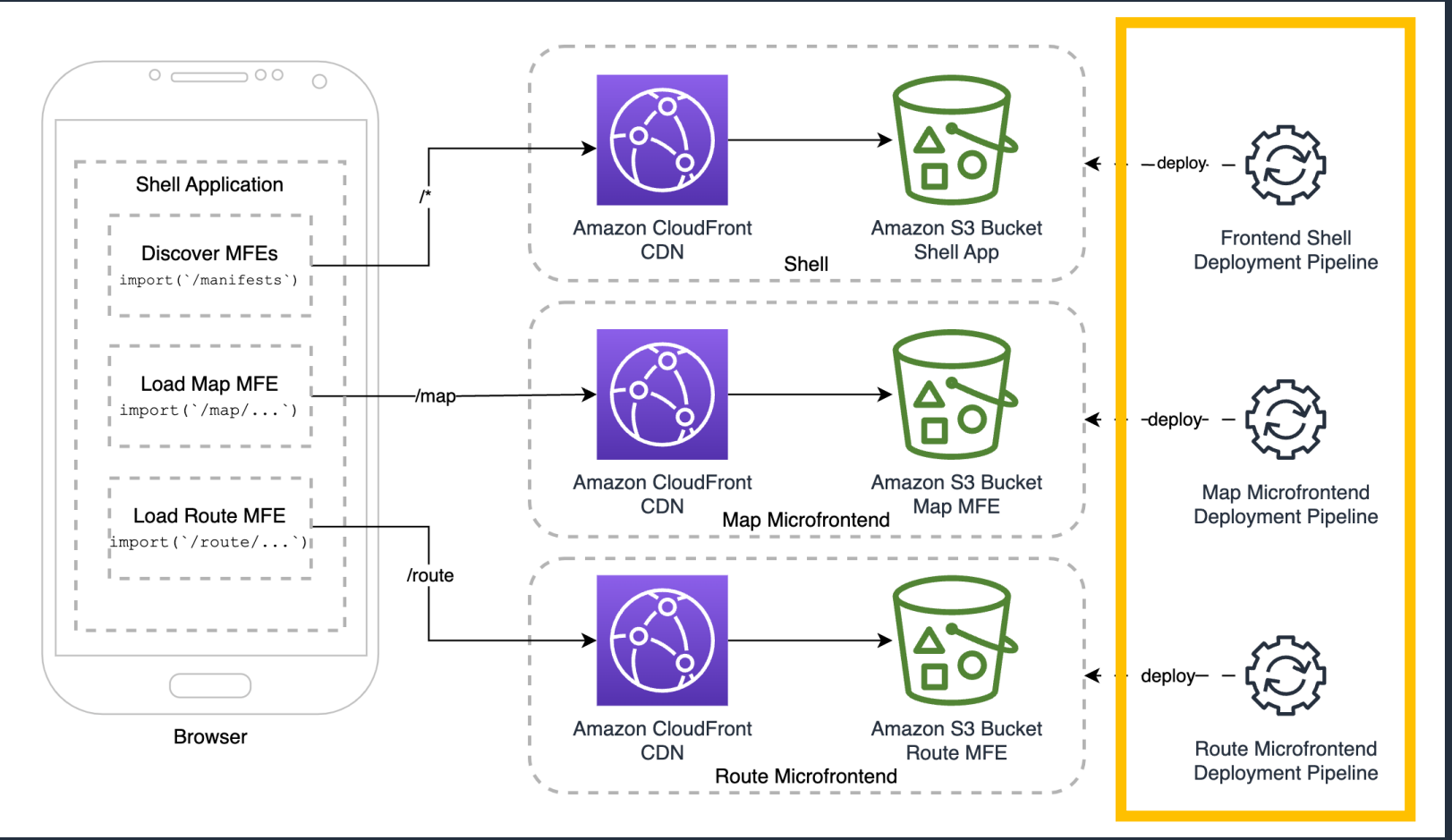
Example - Client side composition



Performant and cost efficient hosting.

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

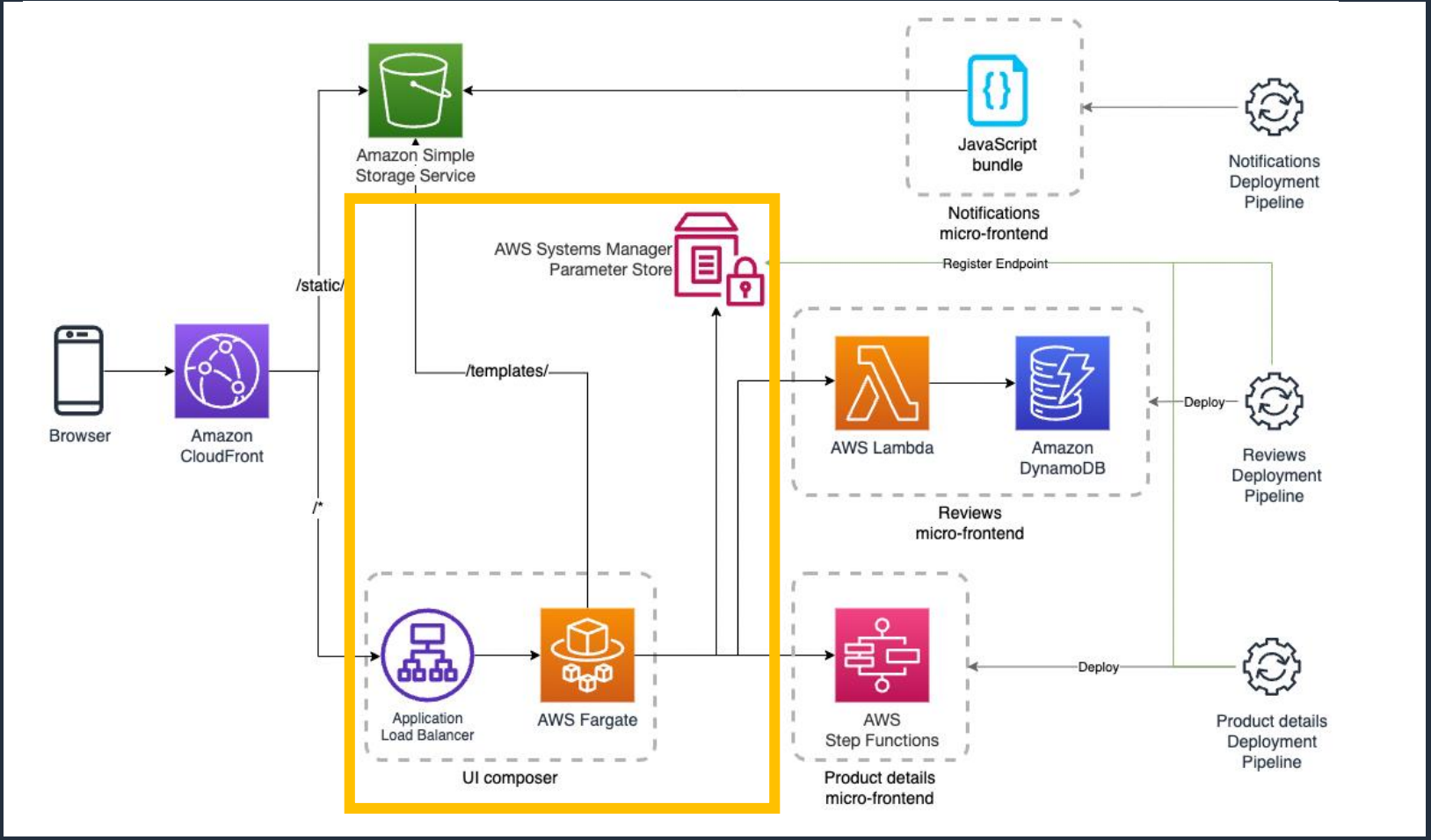
Example - Client side composition



Independent continuous delivery pipelines

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

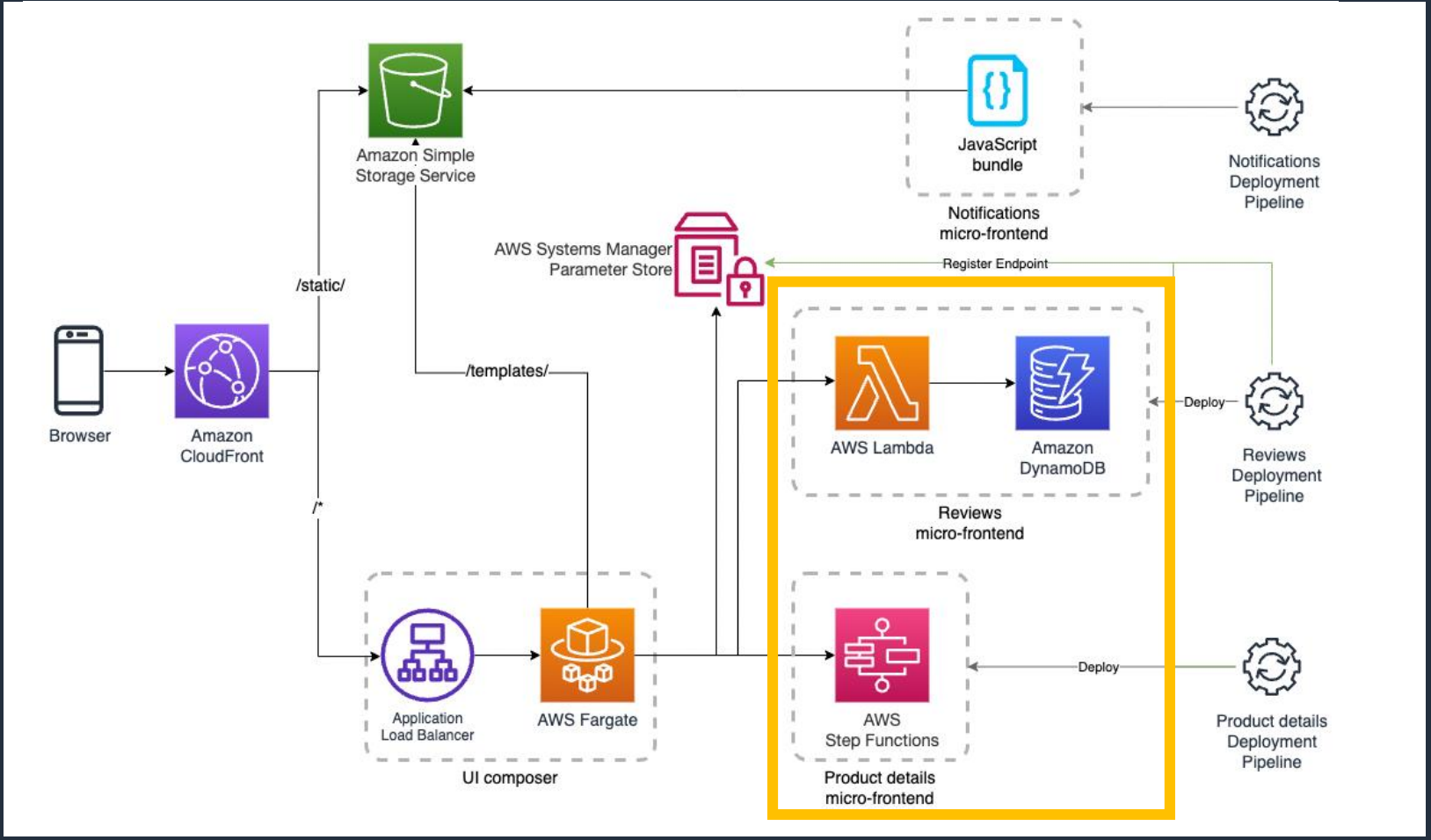
Example – Server-side composition



Discovery on the server via page composer microservice.

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

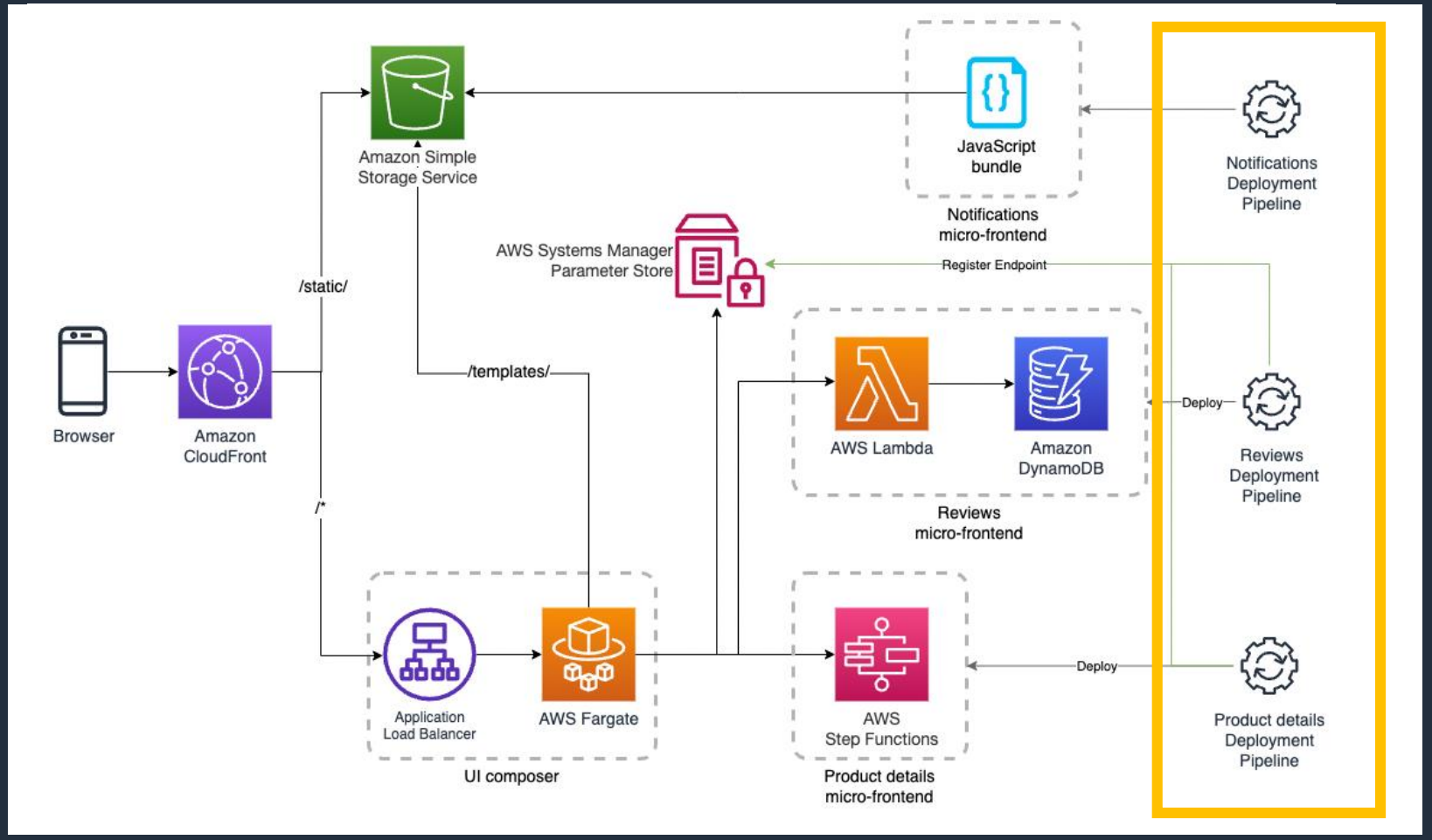
Example – Server-side composition



Choose the best fit serverless compute for your domain.

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Example – Server-side composition



Independent continuous delivery pipelines

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Many patterns are emerging

Let’s have a look at a **real world implementation**.

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Dunelm's Microfrontends Journey

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

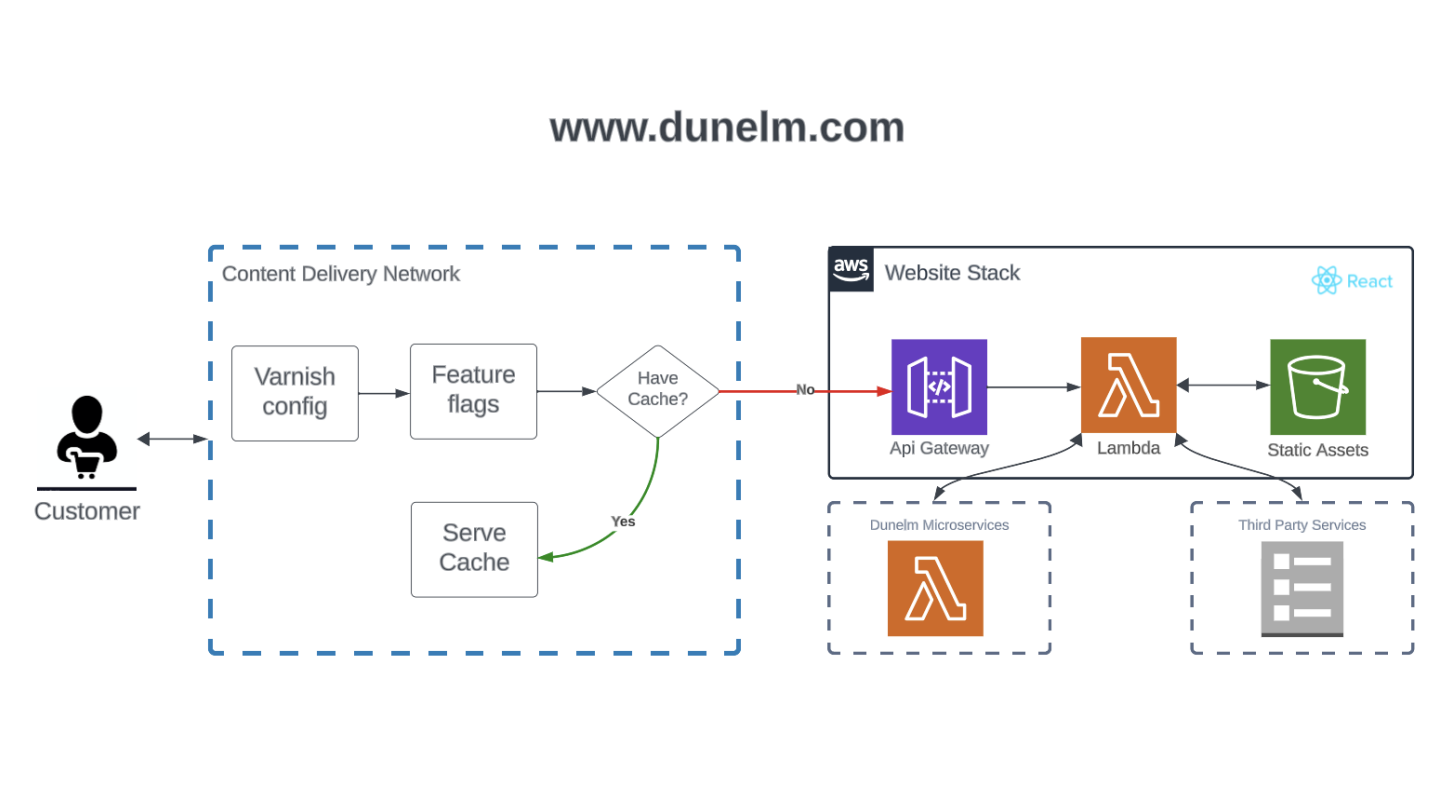
Introduction

Warren Fitzpatrick
Principal Engineer, Dunelm
Digital Optimisation

Dunelm
Homewares Retailer, UK
Founded in 1979
Over 170 stores nationwide

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Where our story begins



Microservices architecture

Monolithic frontend

Isomorphic React application

Monorepo codebase

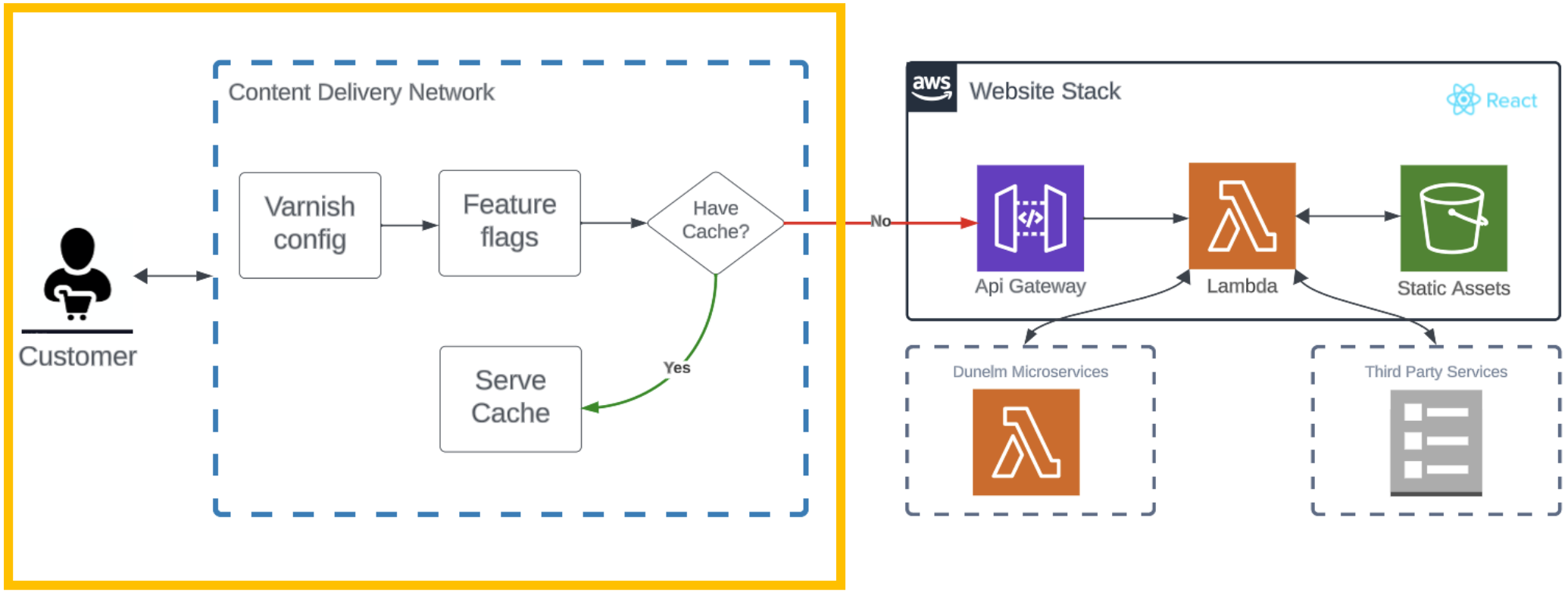
High risk deployment

Slows down over time

Difficulty in scaling **teams**

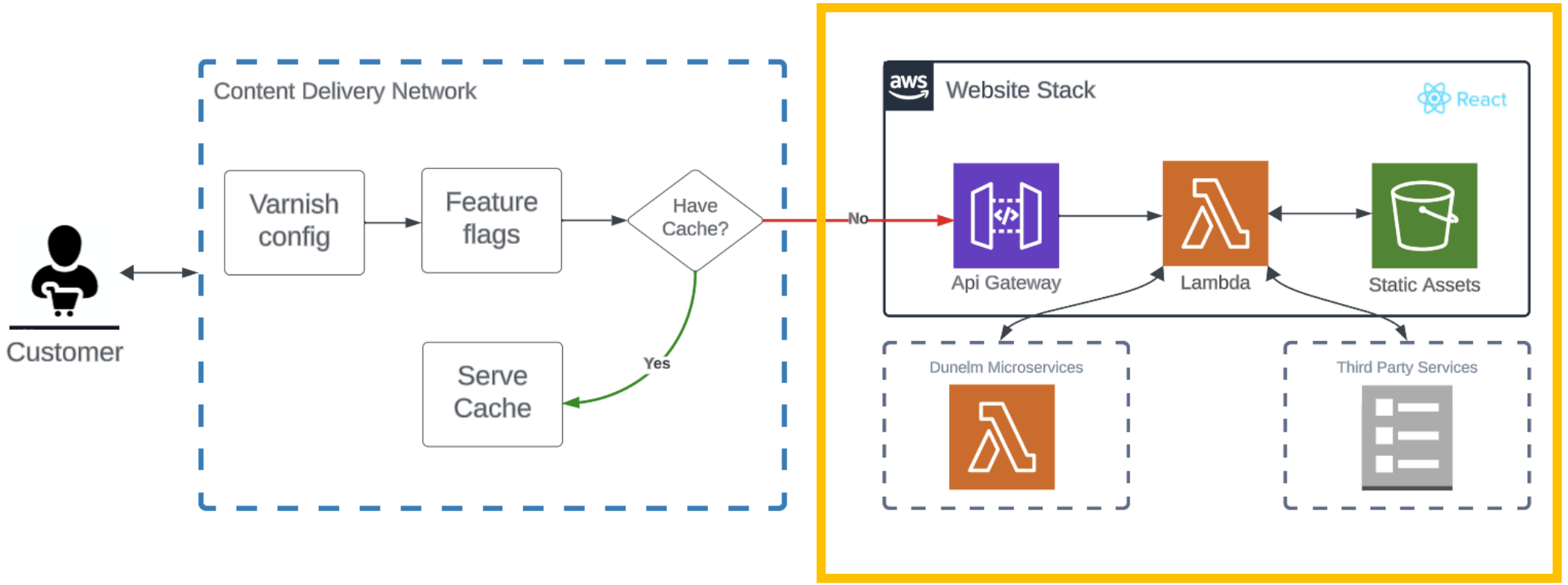
© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

www.dunelm.com



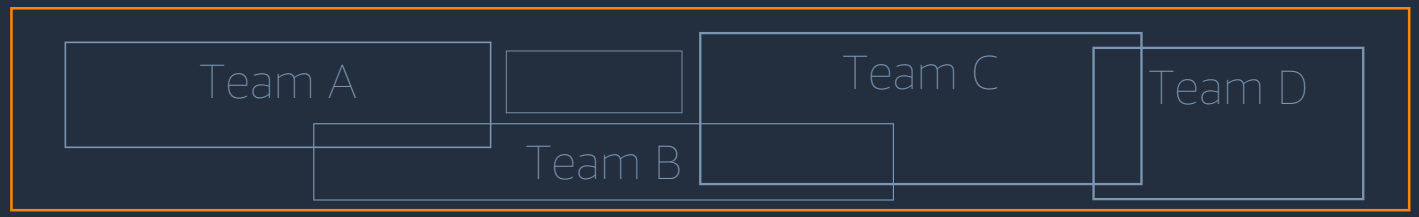
© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

www.dunelm.com



© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Issues at scale



11 teams working in same repo

Communication becomes difficult

Challenges with ownership

Performance declines over time

Complexity increases with changes

Lead time incrementally increases

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Why is lead time increasing?

Data report carried out

Lead time doubled

Time spent reverse engineering

Unintentional bugs introduced

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

"Our website has a lot of tests..."

8200
Unit tests

Pipeline becoming slower

400
Integration tests

Tests becoming flaky

150
End-to-end tests

Blocking teams releasing

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

What were our options?

Can we run our tests more reliably?

- In house solution created
- Improved, but not fixed

Can we refactor the codebase in place?

- Complex dependency structure
- Cannot halt work for teams

High priority work must continue uninterrupted

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

What we want to achieve

Make it easy for teams to take ownership

Decrease lead time on frontend changes

Improve website performance

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

What we want to achieve

Make it easy for teams to take ownership

Decrease lead time on frontend changes

Improve website performance

Improve online accessibility

Create consistency for all GUIs

Improve developer experience

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Are Microfrontends suitable for our goals?

Make it easy for teams to take ownership

Decrease lead time on frontend changes

Improve website performance

Improve online accessibility

Create consistency for all GUIs

Improve developer experience

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Are Microfrontends suitable for our goals?

Make it easy for teams to take ownership

Decrease lead time on frontend changes

Improve website performance

Improve online accessibility

Create consistency for all GUIs

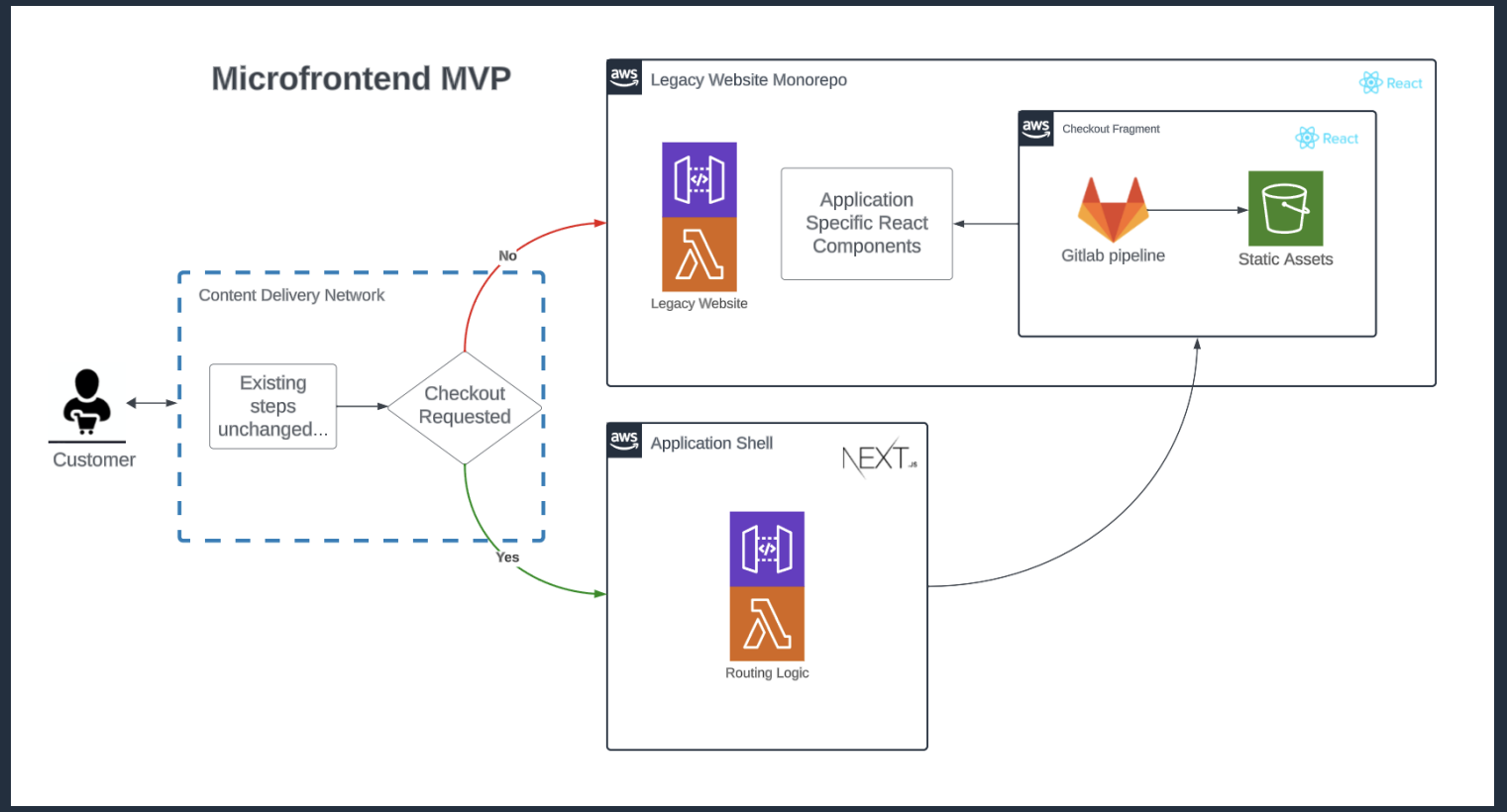
Improve developer experience

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

"Microfrontend architecture is not a silver bullet"

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Microfrontend MVP – Checkout

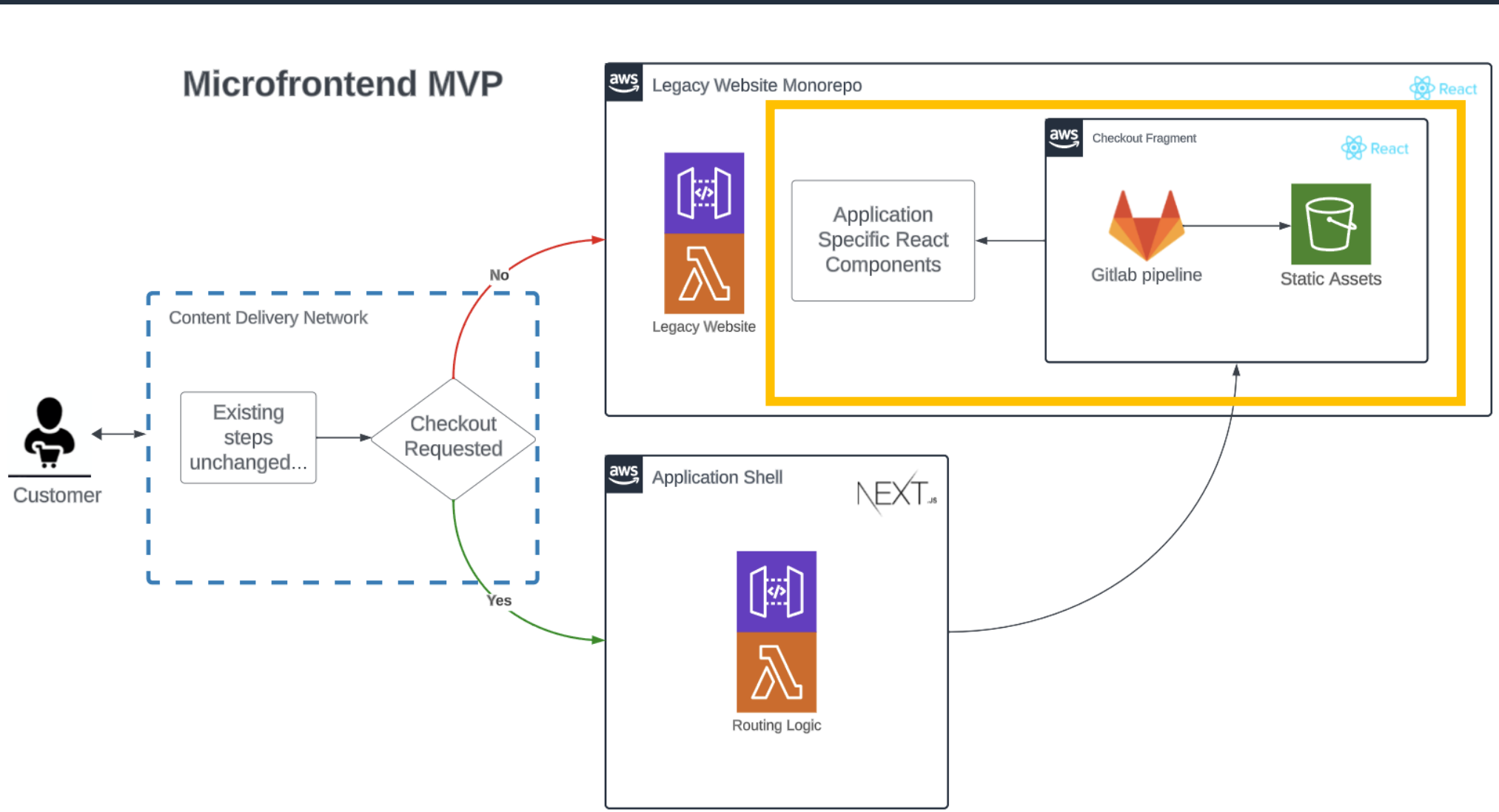


MVP approach to start small

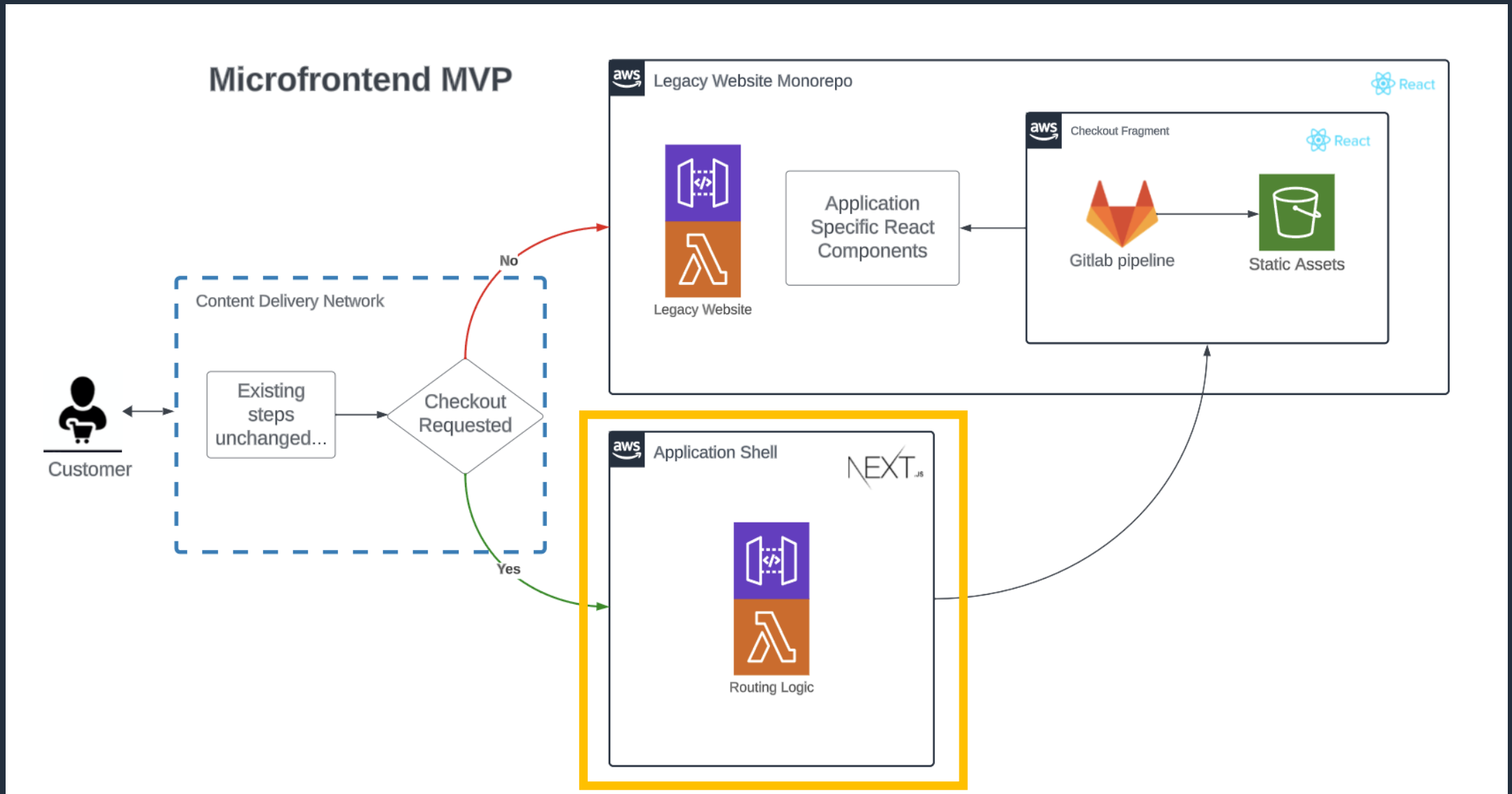
Goal is to prove hypothesis

Ownership assigned to a team

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

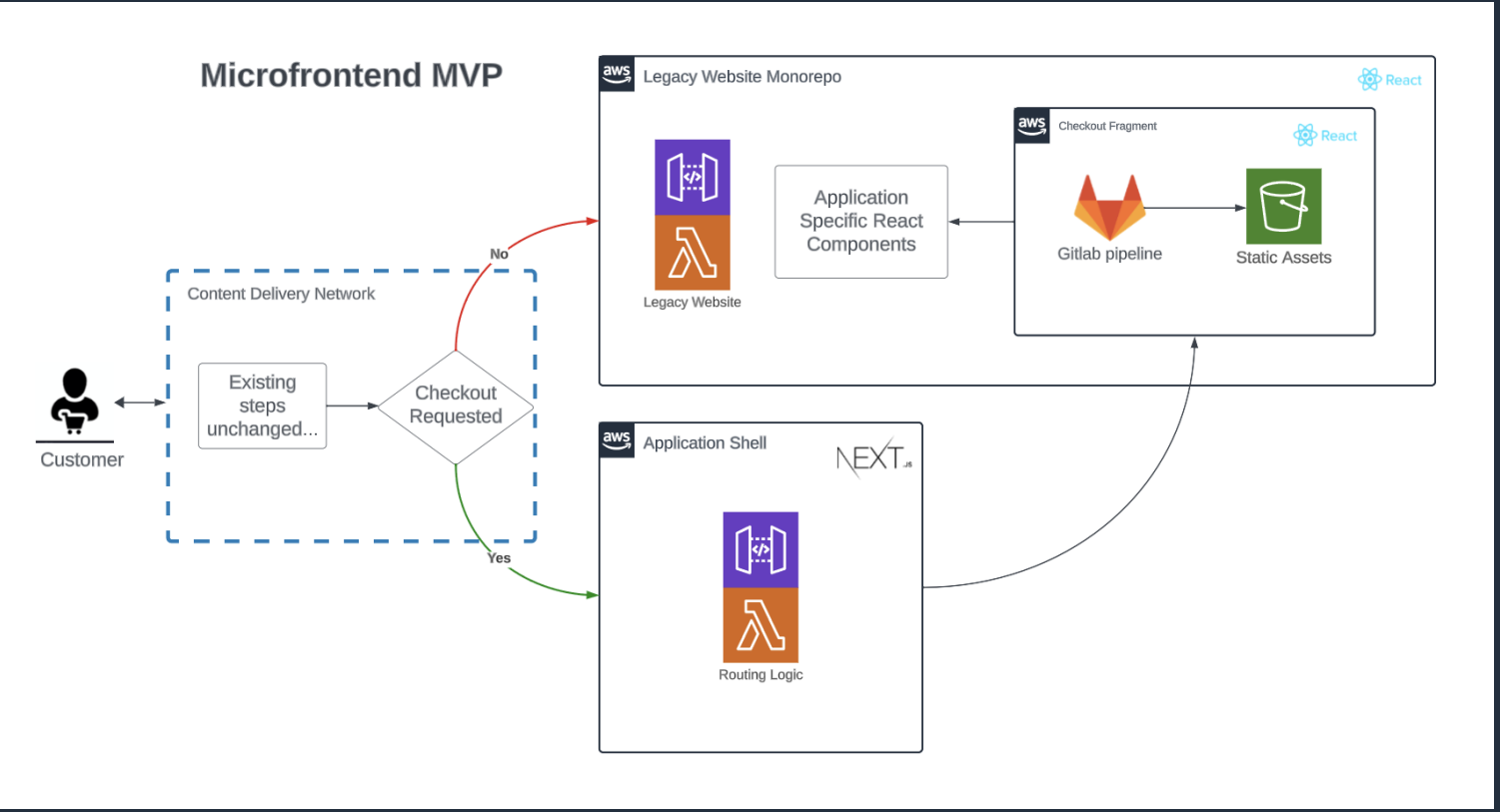


© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.



© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Checkout Results



Revenue uplift generated

Shorter lead time on changes

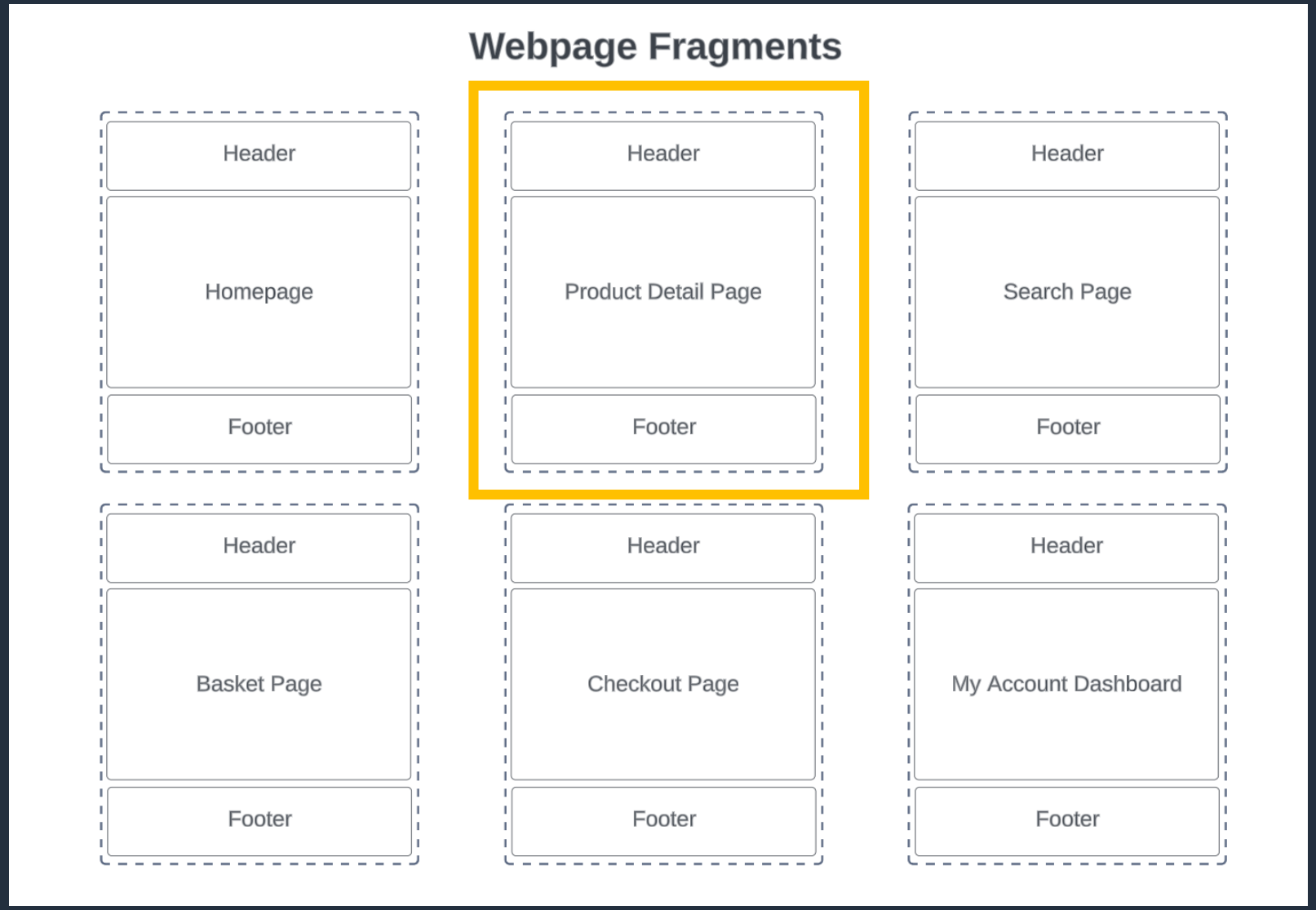
Tight coupling created

Tests began to fail

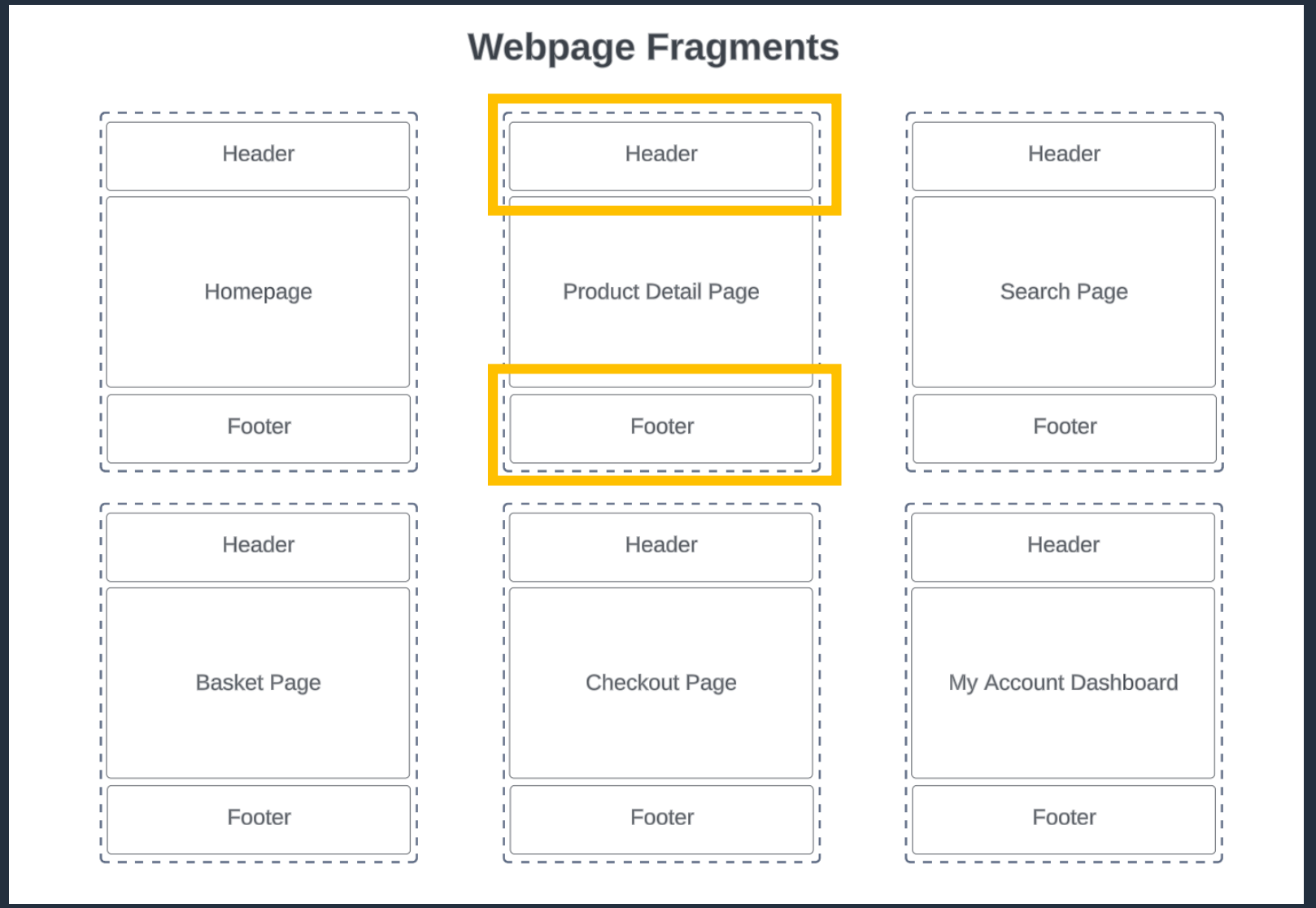
Coupled checkout back to website

Convinced by results

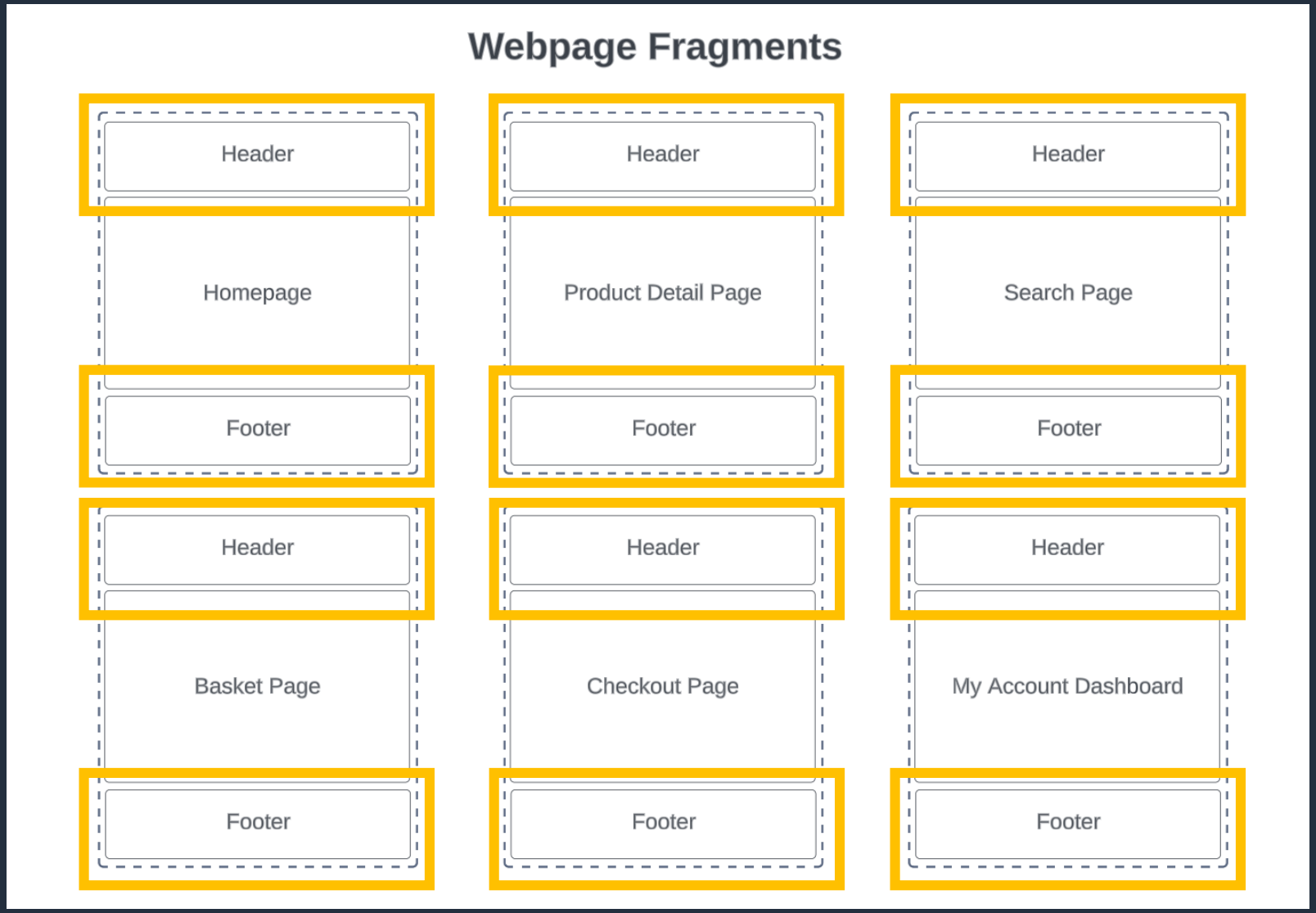
© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.



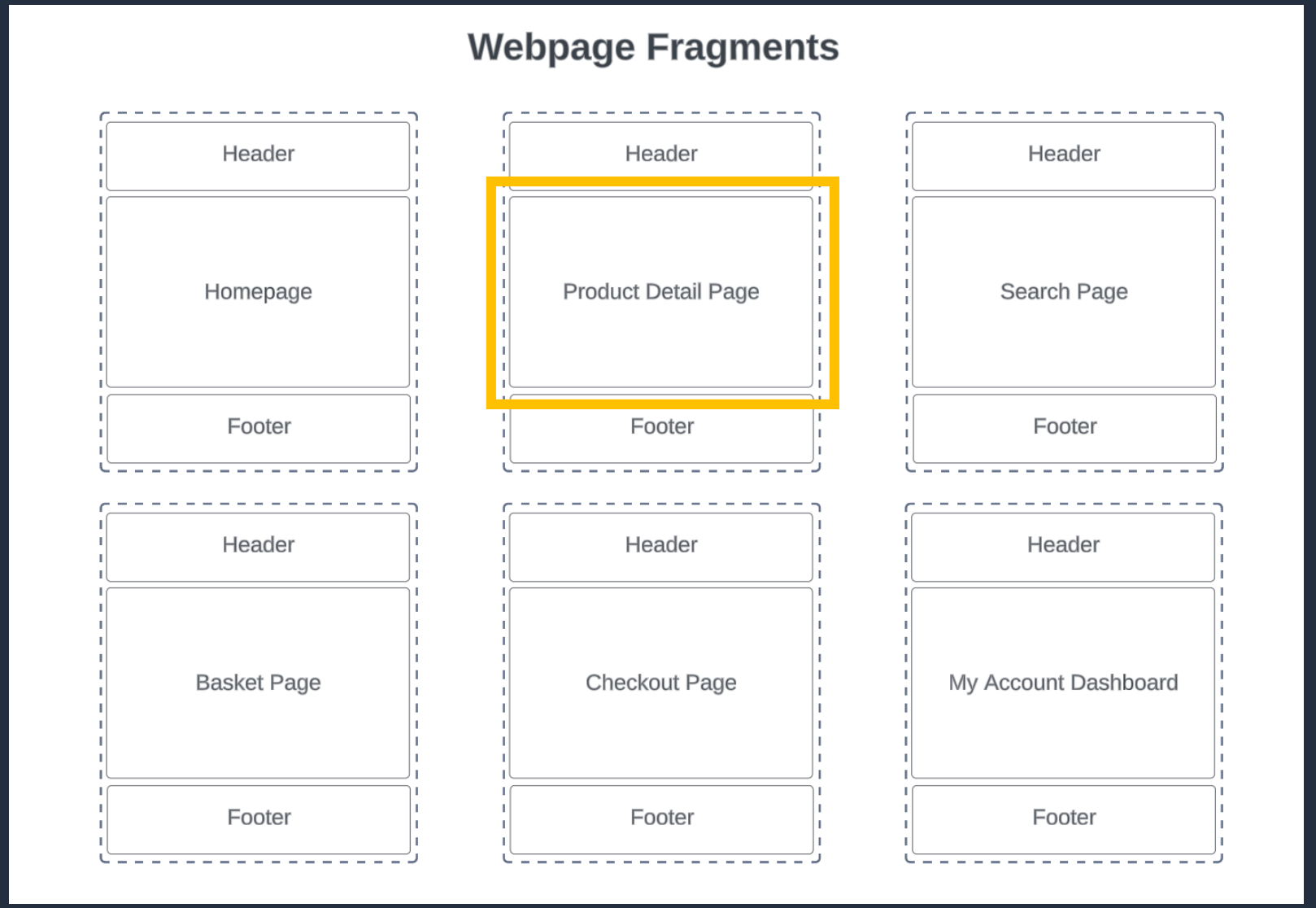
© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.



© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

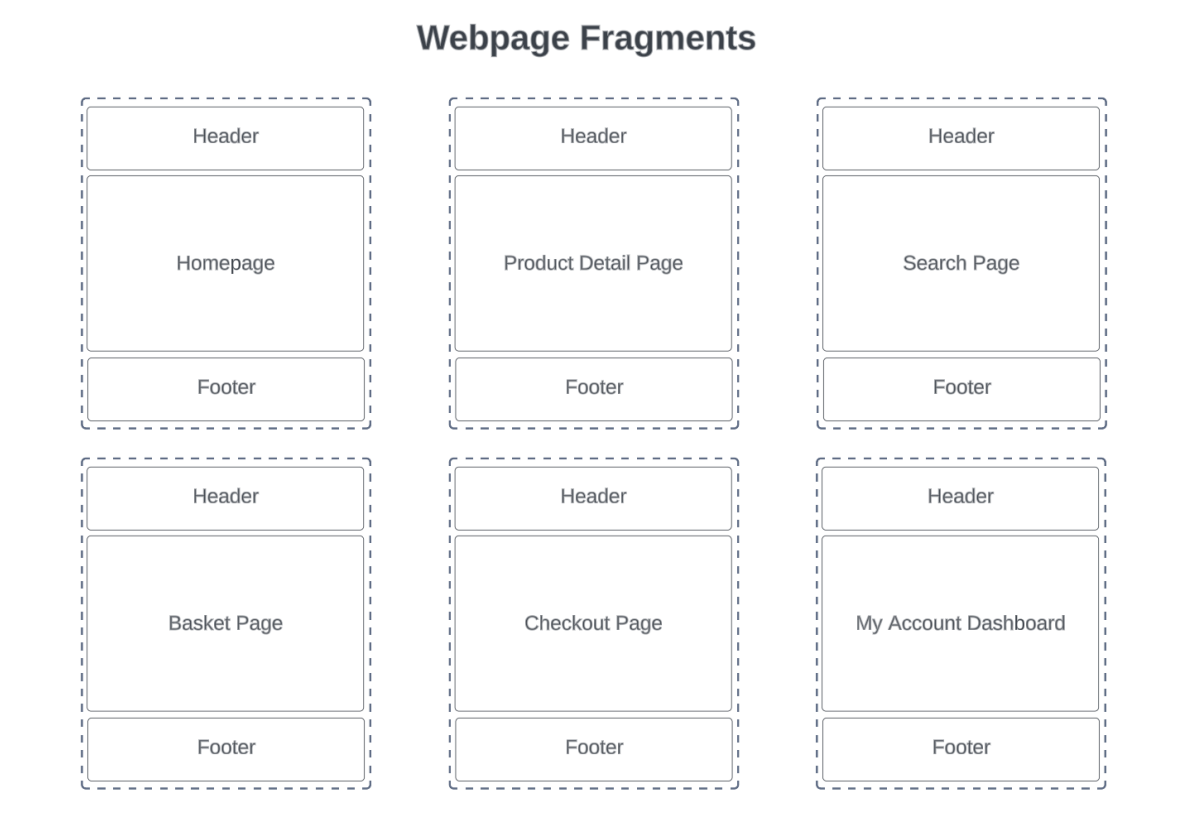


© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.



© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Designing our fragments and approach



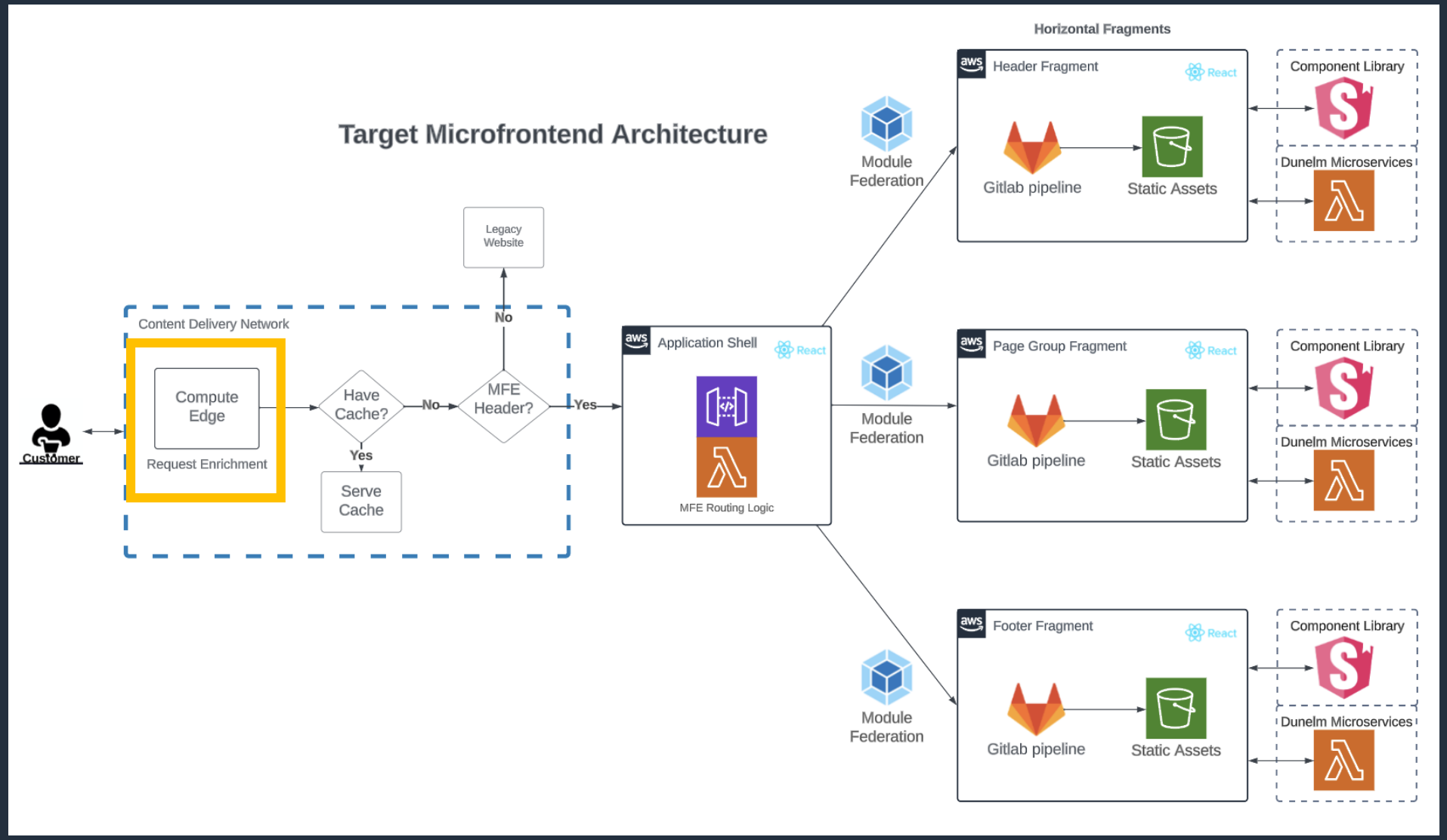
Poly repo approach decided

Consistency must be orchestrated

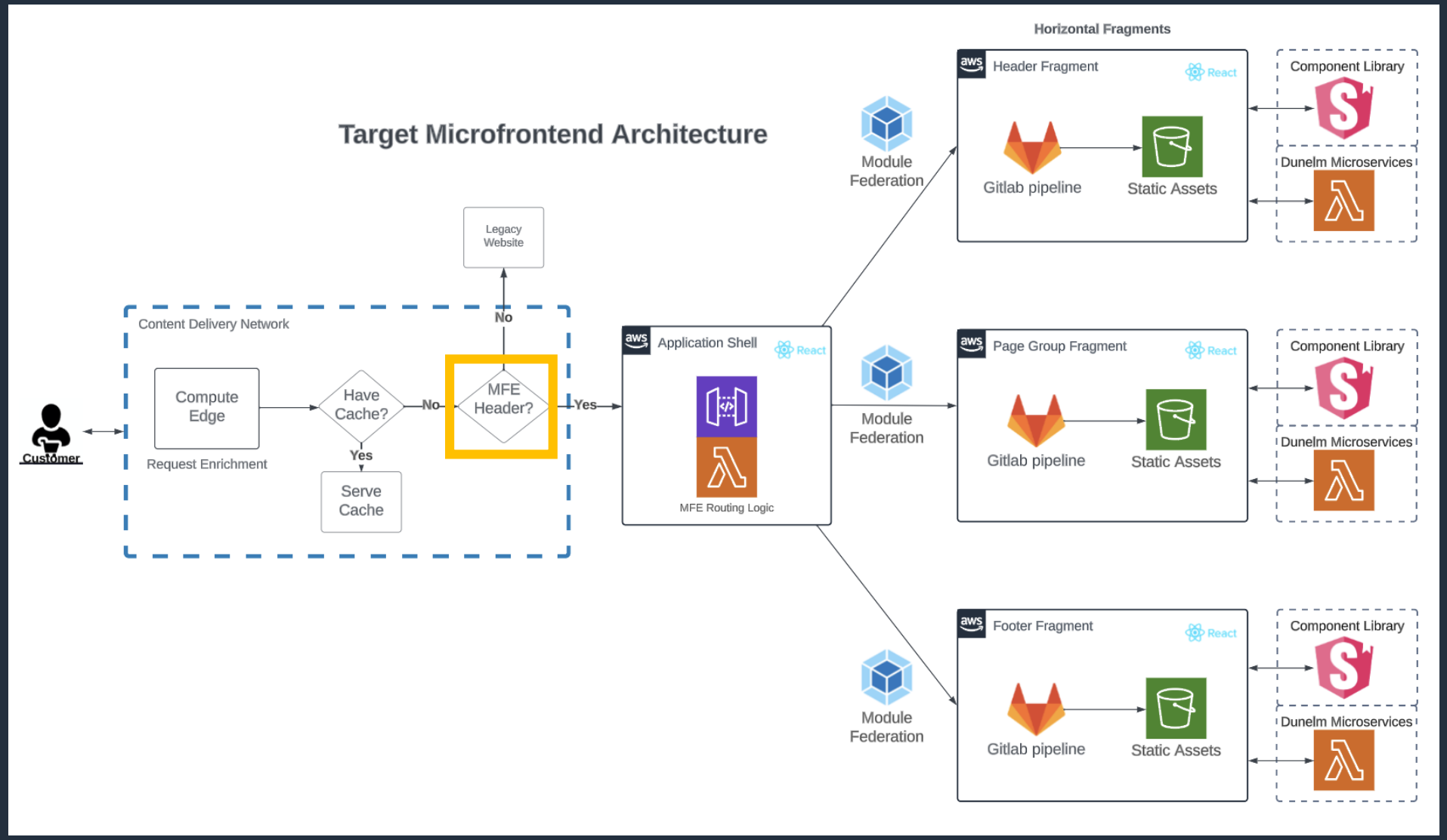
Component Library necessary

Share Nothing by default

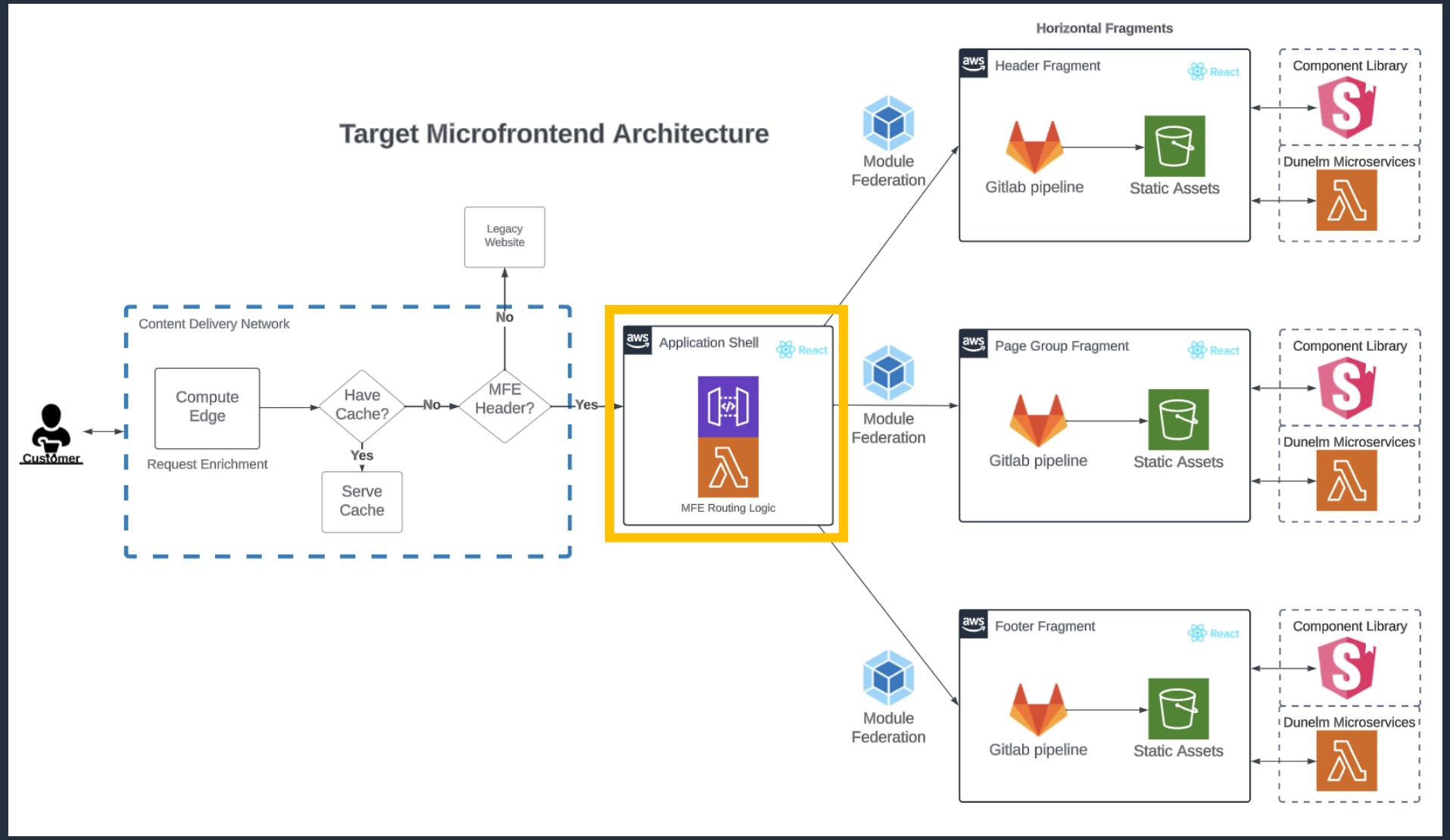
© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.



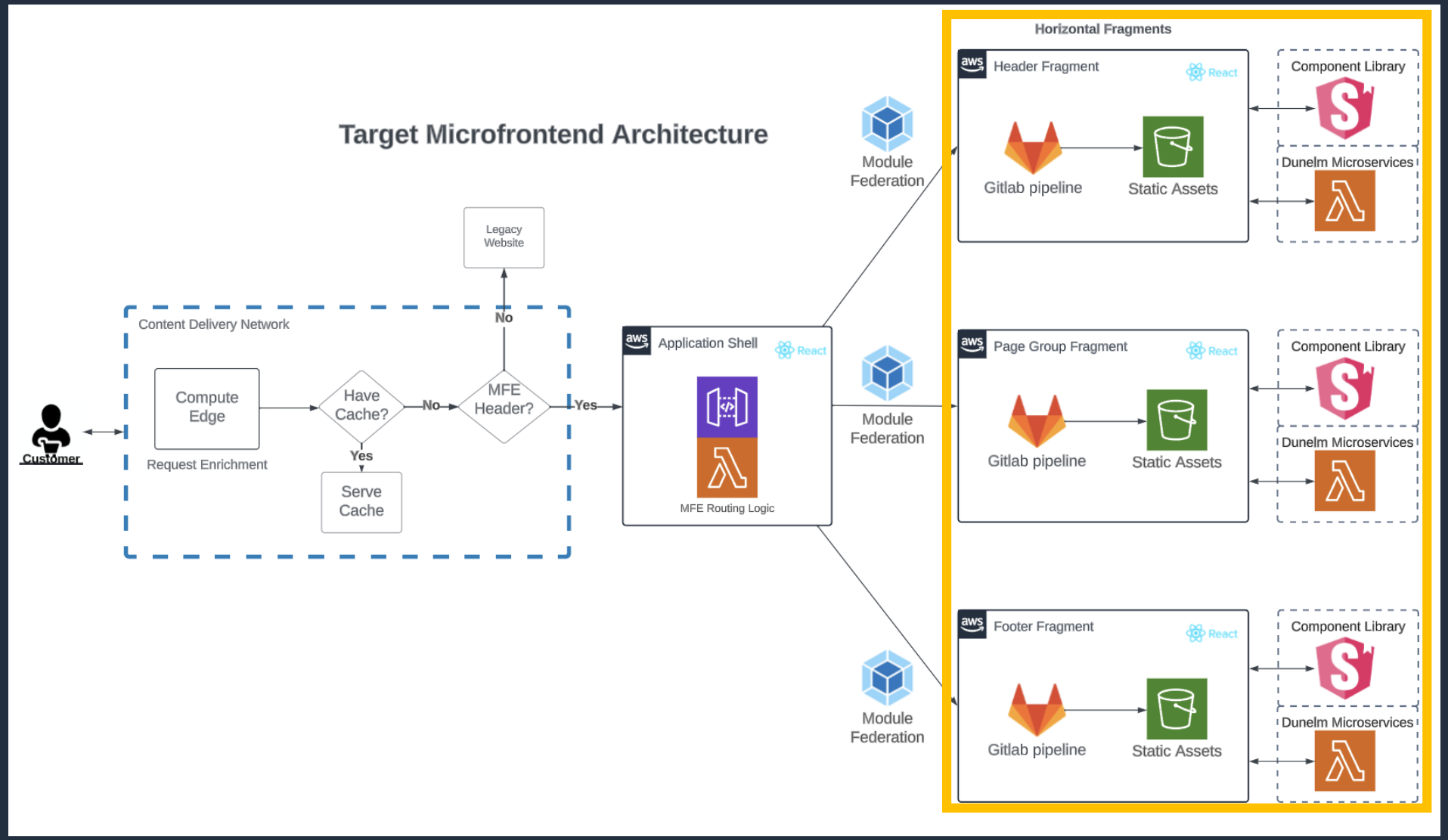
© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.



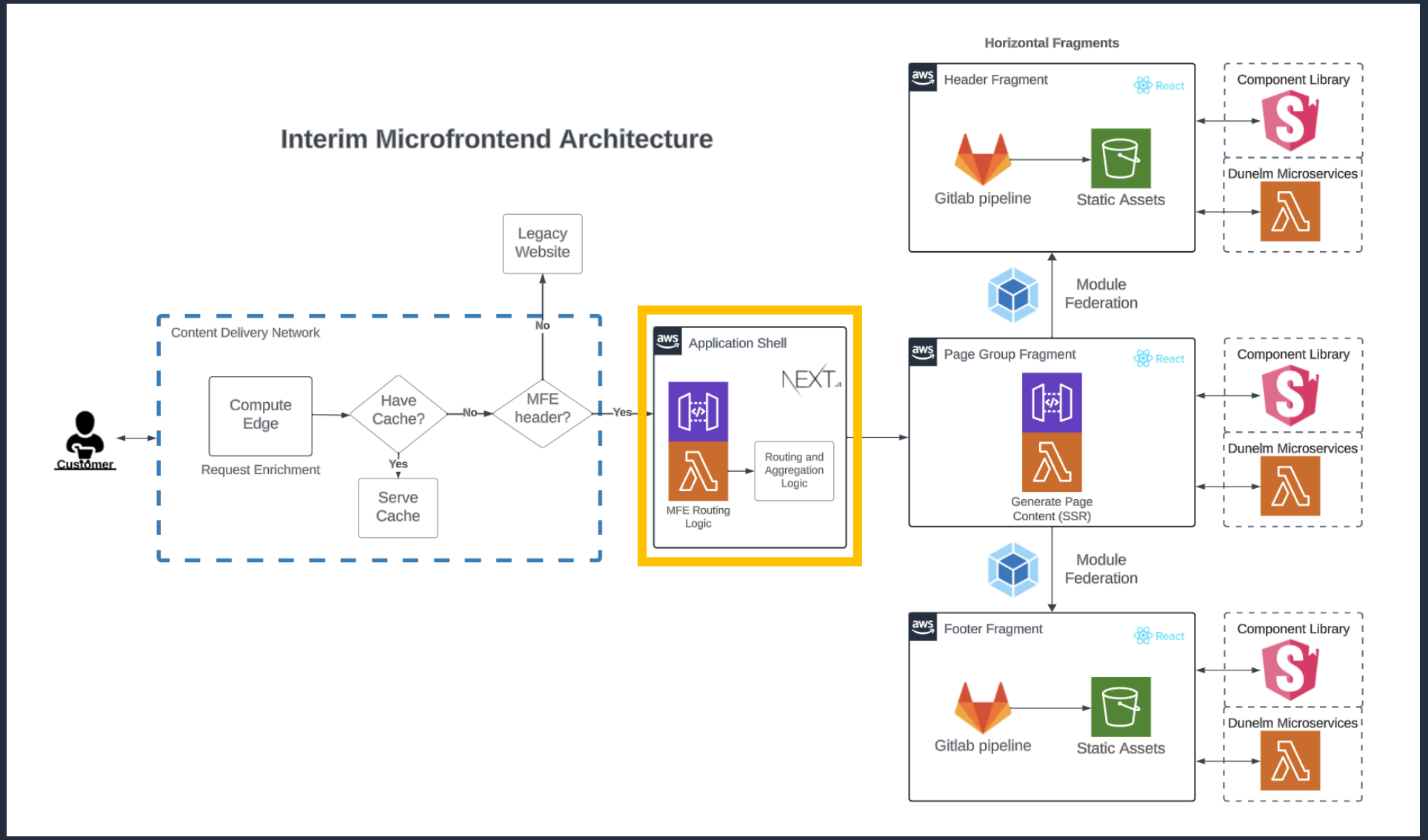
© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.



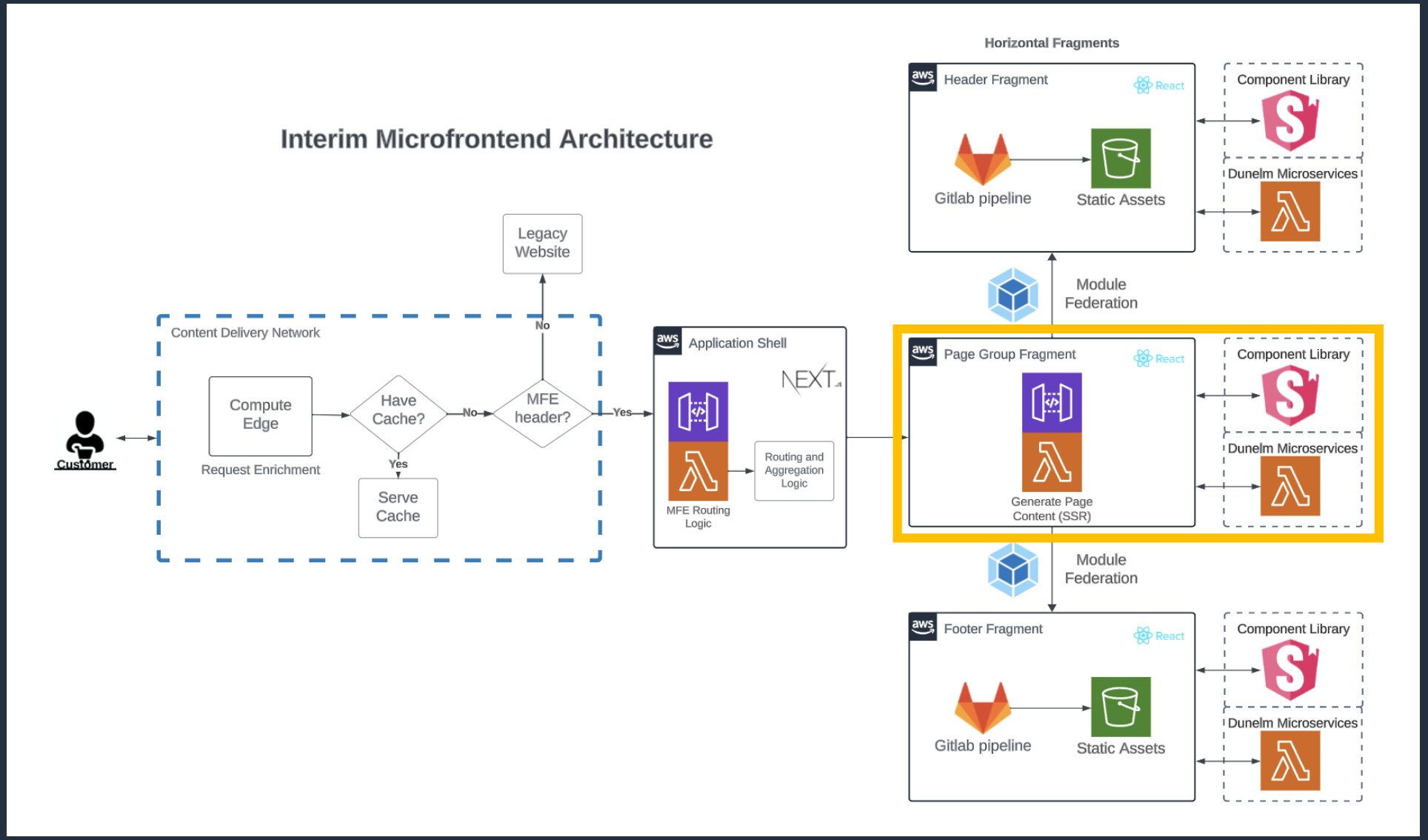
© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.



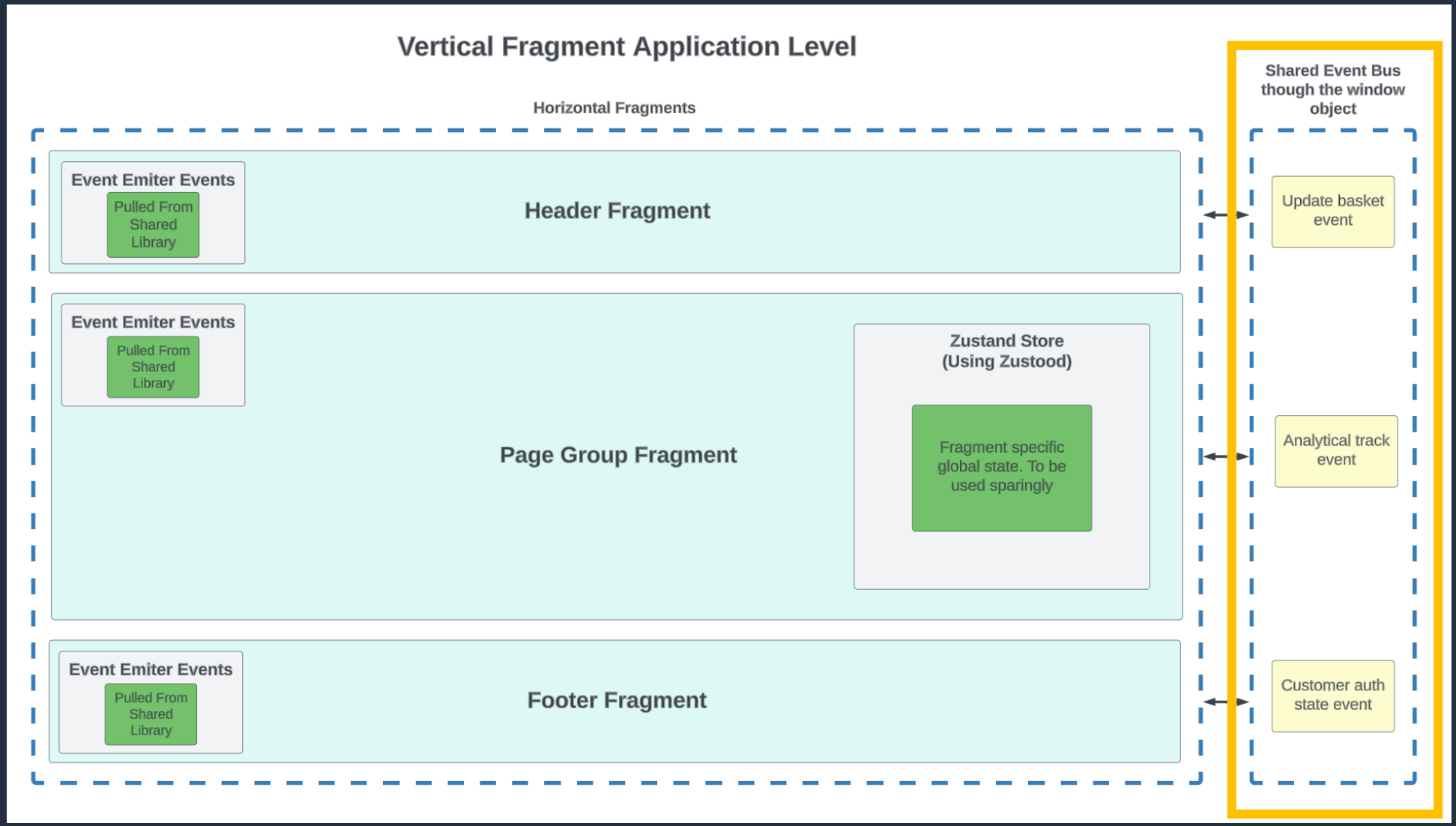
© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.



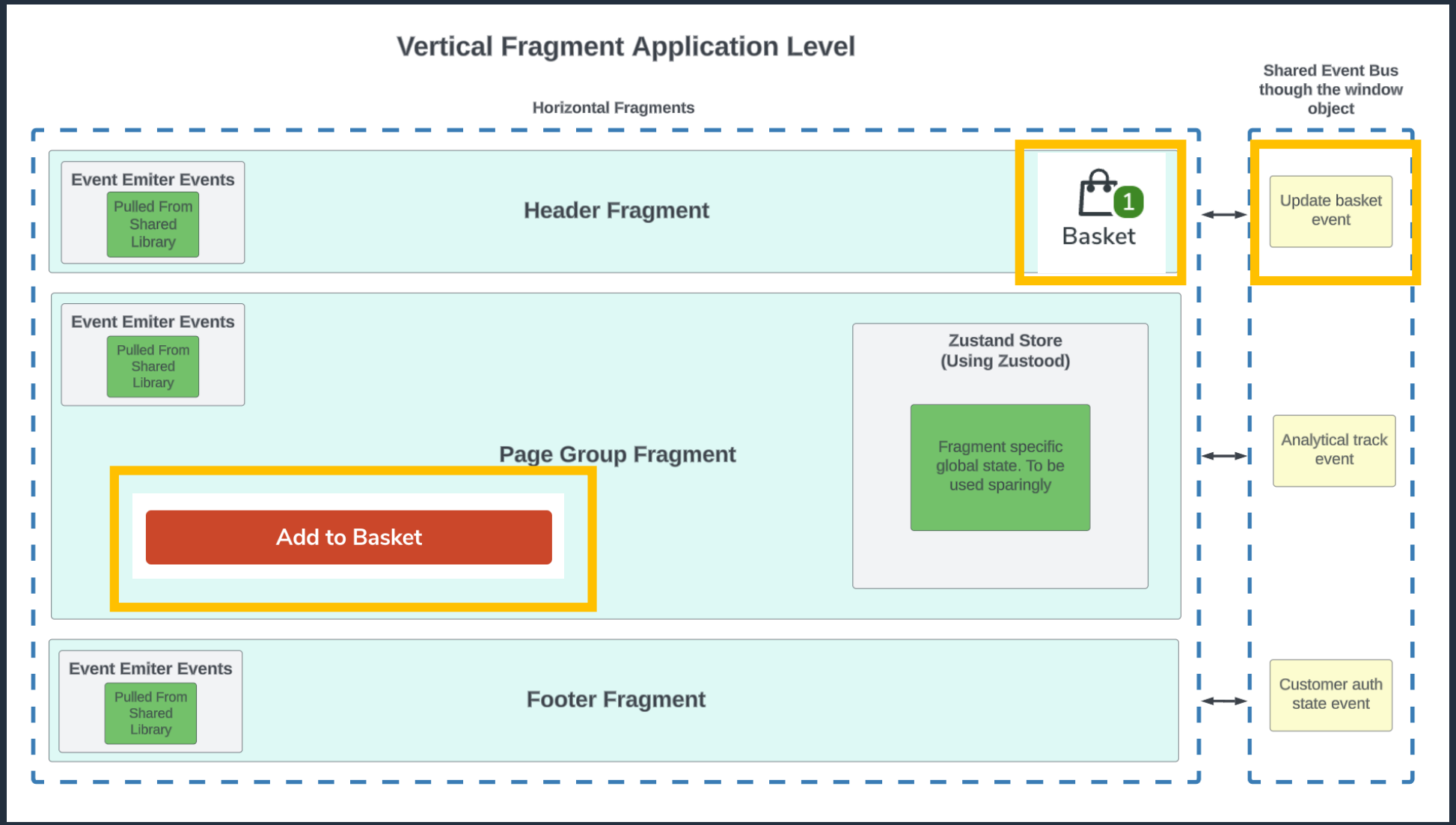
© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.



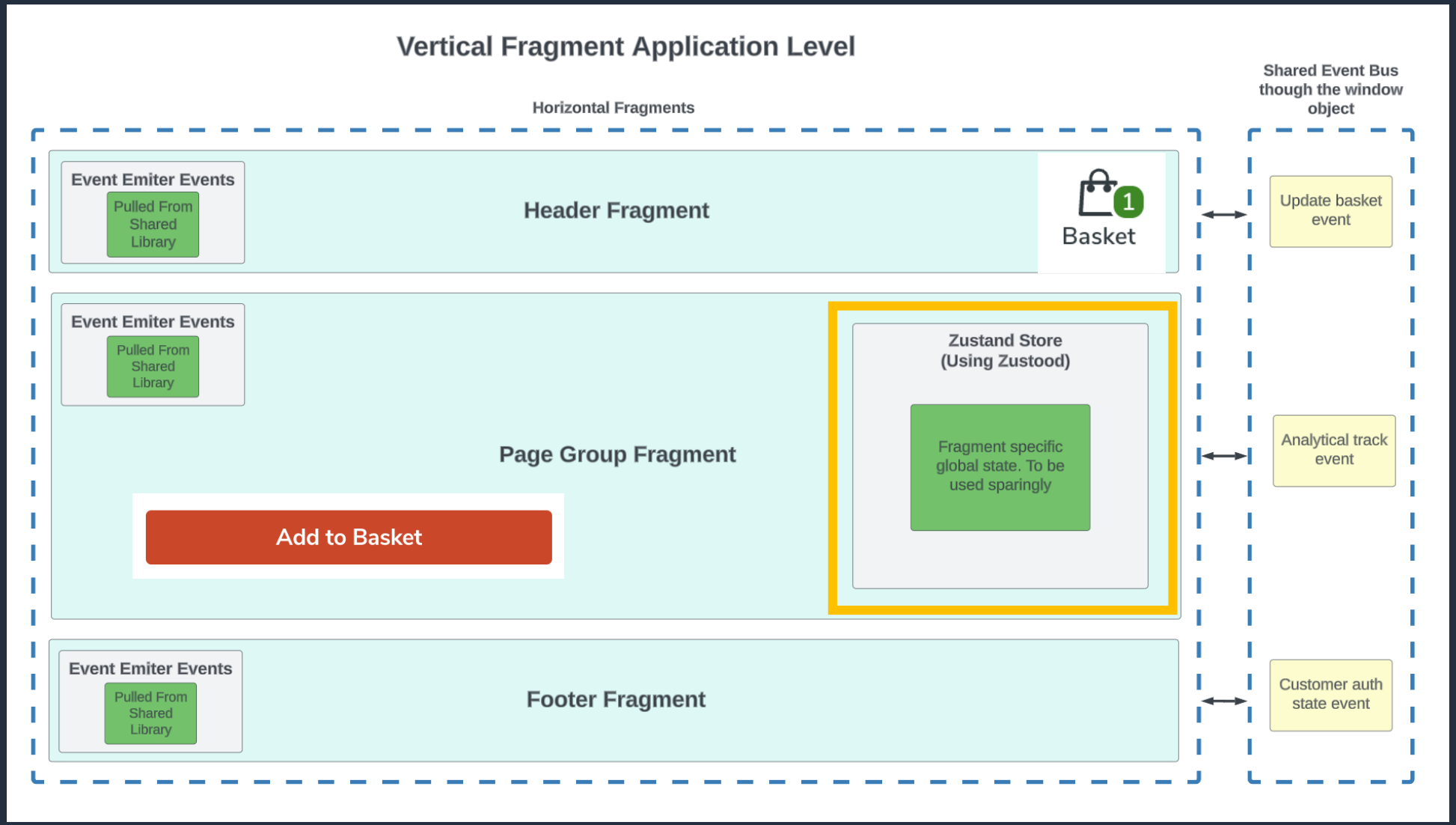
© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.



© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

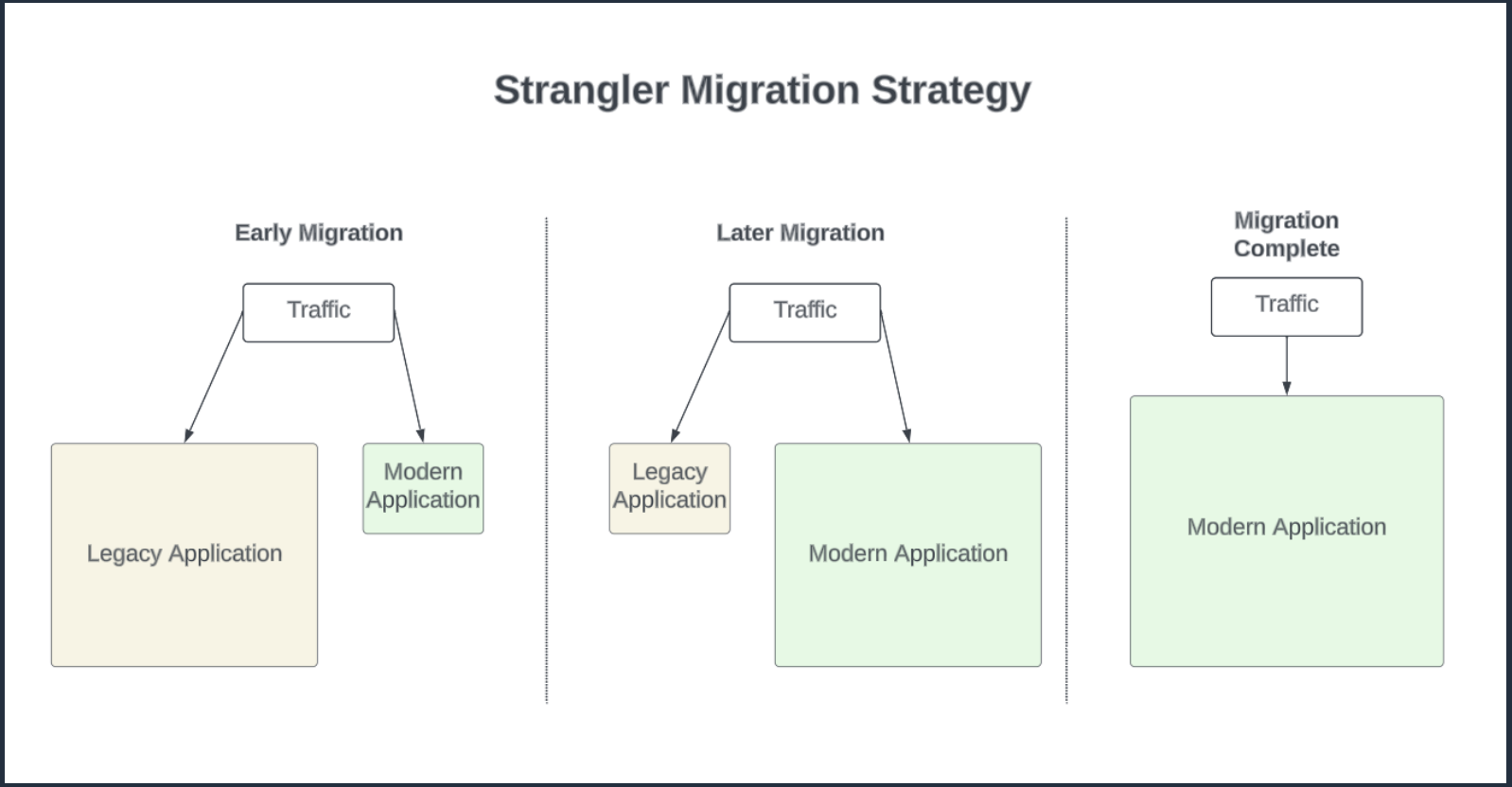


© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.



© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Migration Strategy



Iterative migration strategy

Microservices pattern

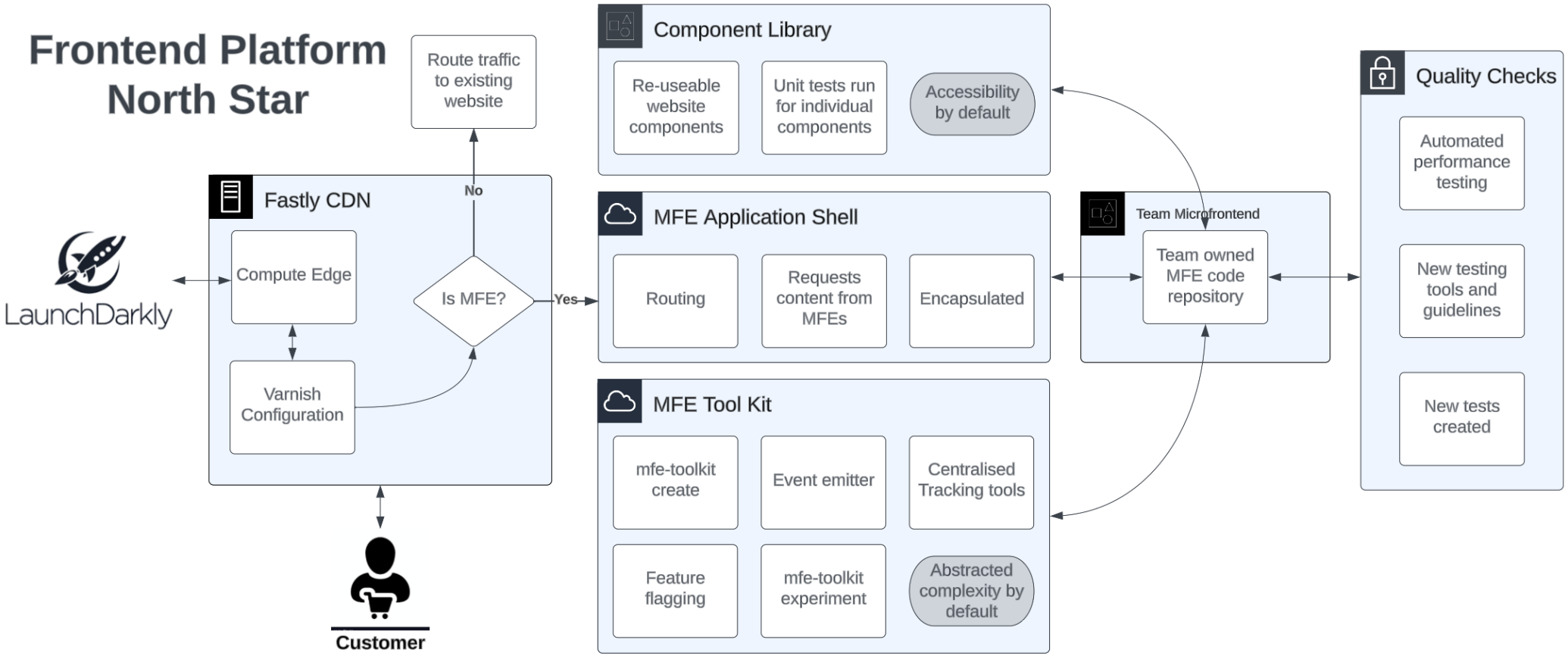
Works for **Microfrontends**

One **page group** at a time

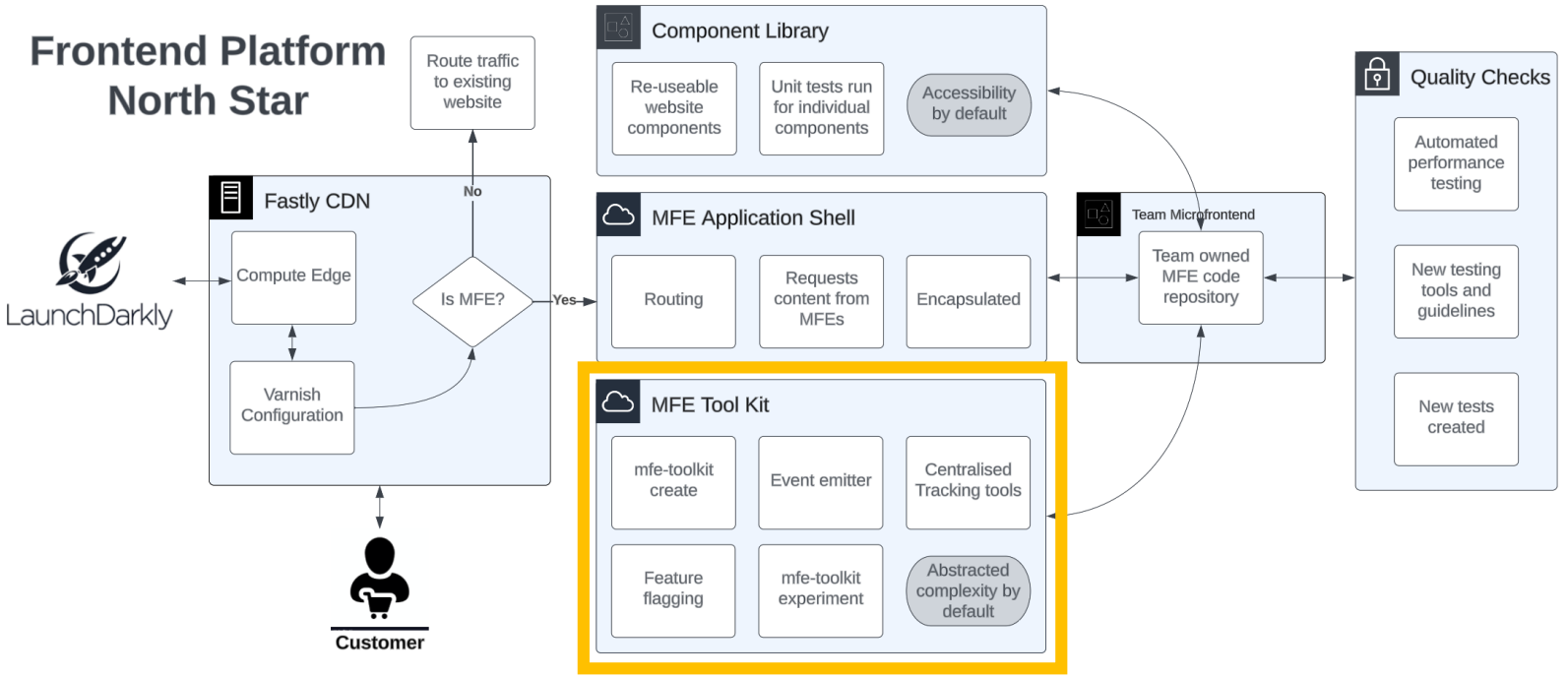
Low impact to existing work

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Frontend Platform North Star



© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.



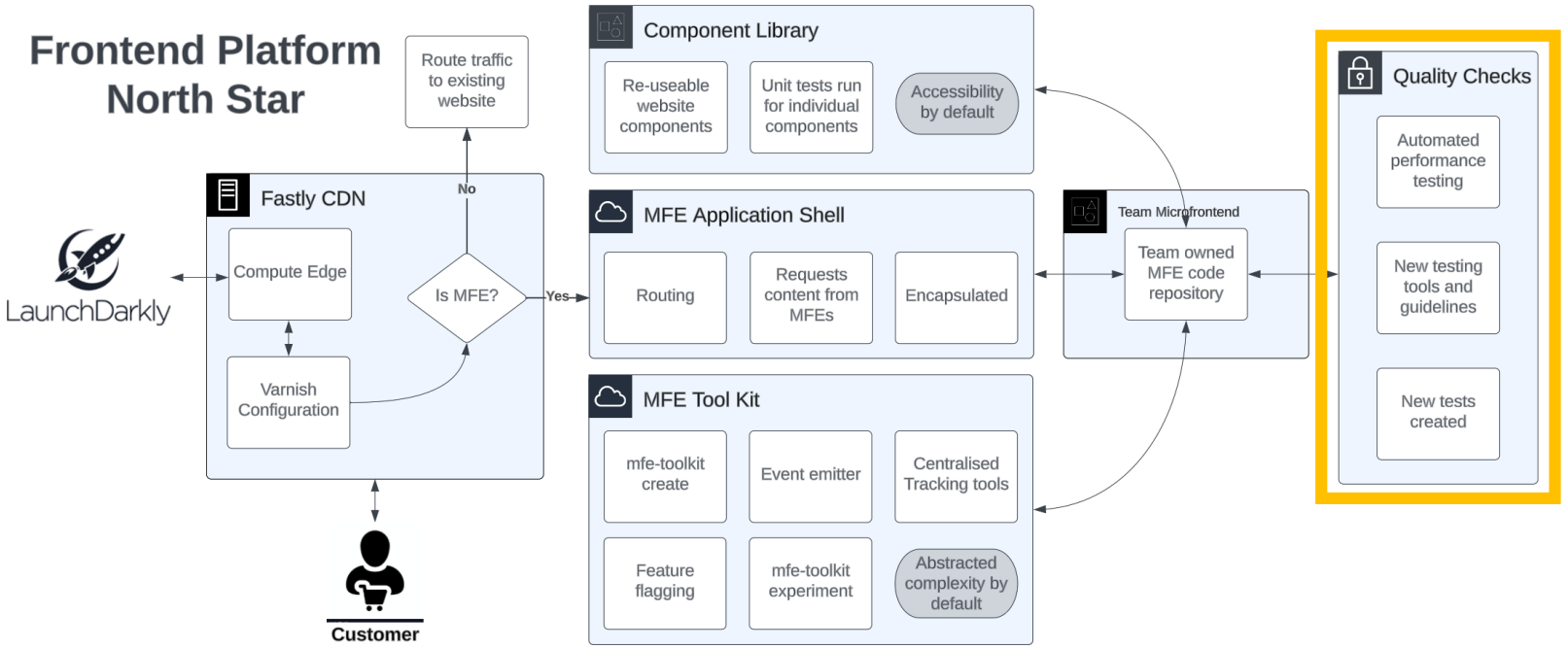
Abstract complexity with tools

Ability to scaffold a project

Examples of common patterns

Saves time for teams

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.



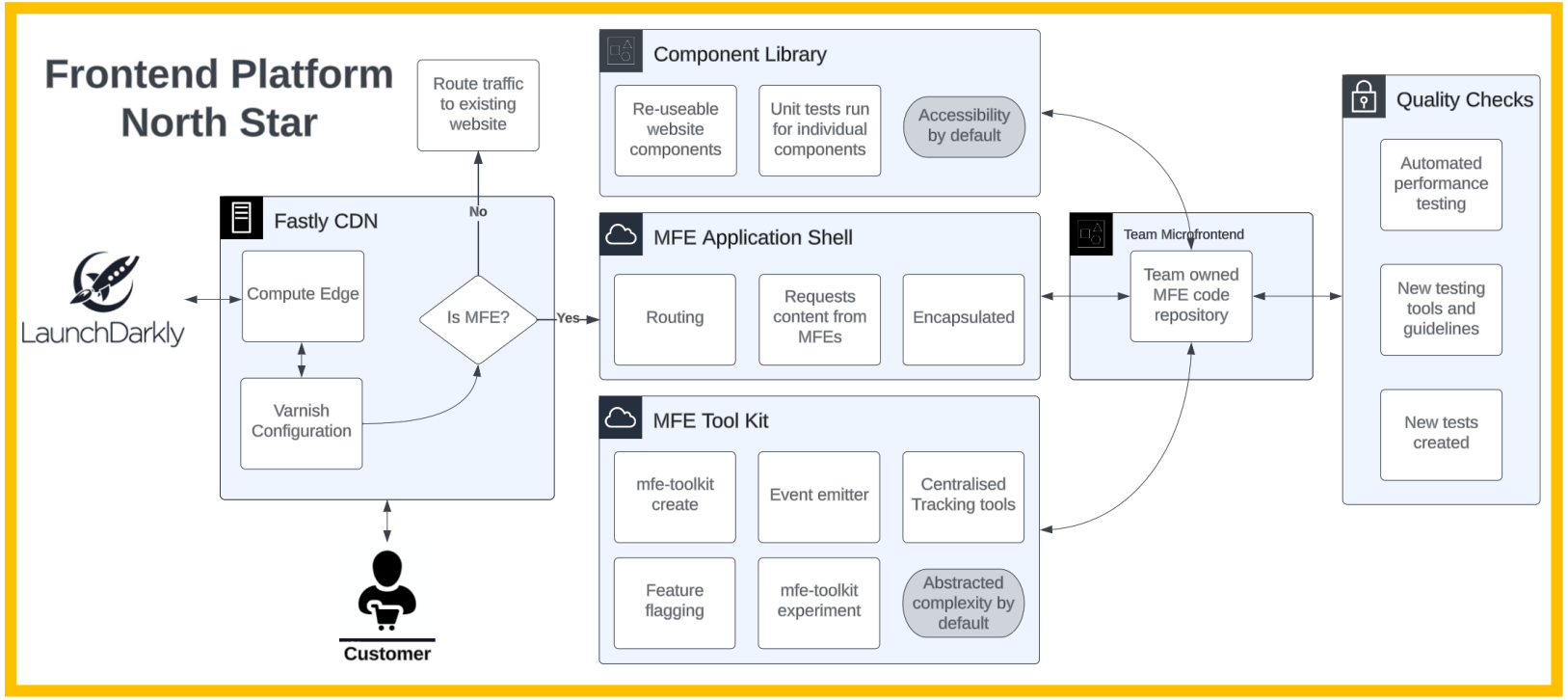
Prevent issues ahead of release

Checks during development cycle

Address issues before production

New tools and guidelines

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.



Dedicated Ownership required

Enablement topology missing

Consistency and efficiency

New enablement crew created

Enable and empower

Decentralise knowledge

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

What progress have we made?

- ~90% of the Frontend Platform is built
- 60,000 Product Detail Pages delivered recently
- First migration was successful, despite challenges

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

"Issues will always arise"

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

What progress have we made?

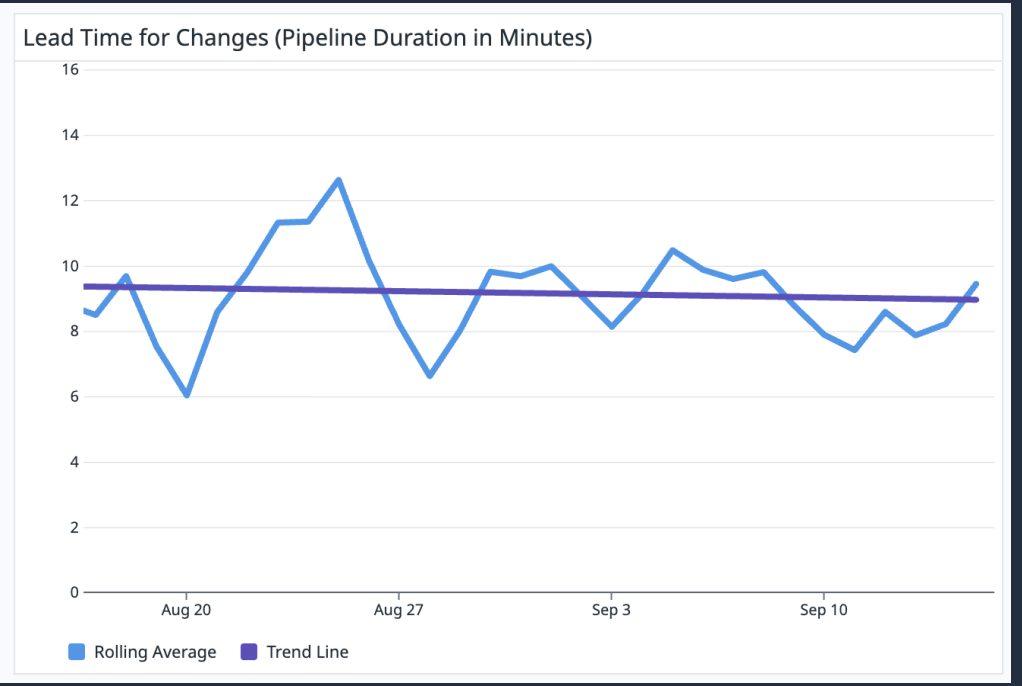
Teams being onboarded with new process

Projects outside of the website benefitting from the component library

Developer Experience improved

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

What about performance and lead time?



© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Final thoughts

We are still on our Micro-Frontend journey!

Every organisation is different

Tailor teams to business domains

Ability to scale infinitely

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

"Microfrontends open avenues for your teams"

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Thank you!

Warren Fitzpatrick

warren.fitzpatrick@dunelm.com

https://www.linkedin.com/in/warren-fitzpatrick-b1885263/

Dunelm Technology

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved. Amazon Confidential and Trademark.

Thank you!

Harun Hasdal
hhasdal@amazon.com

Warren Fitzpatrick
warren.fitzpatrick@dunelm.com

AWS Architecture Blog

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.