bye bye frontend monolith

accelerating feature development with micro frontends

#PhillyETE / Emerging Technologies for the Enterprise 2021

Michael Geers / @naltatis

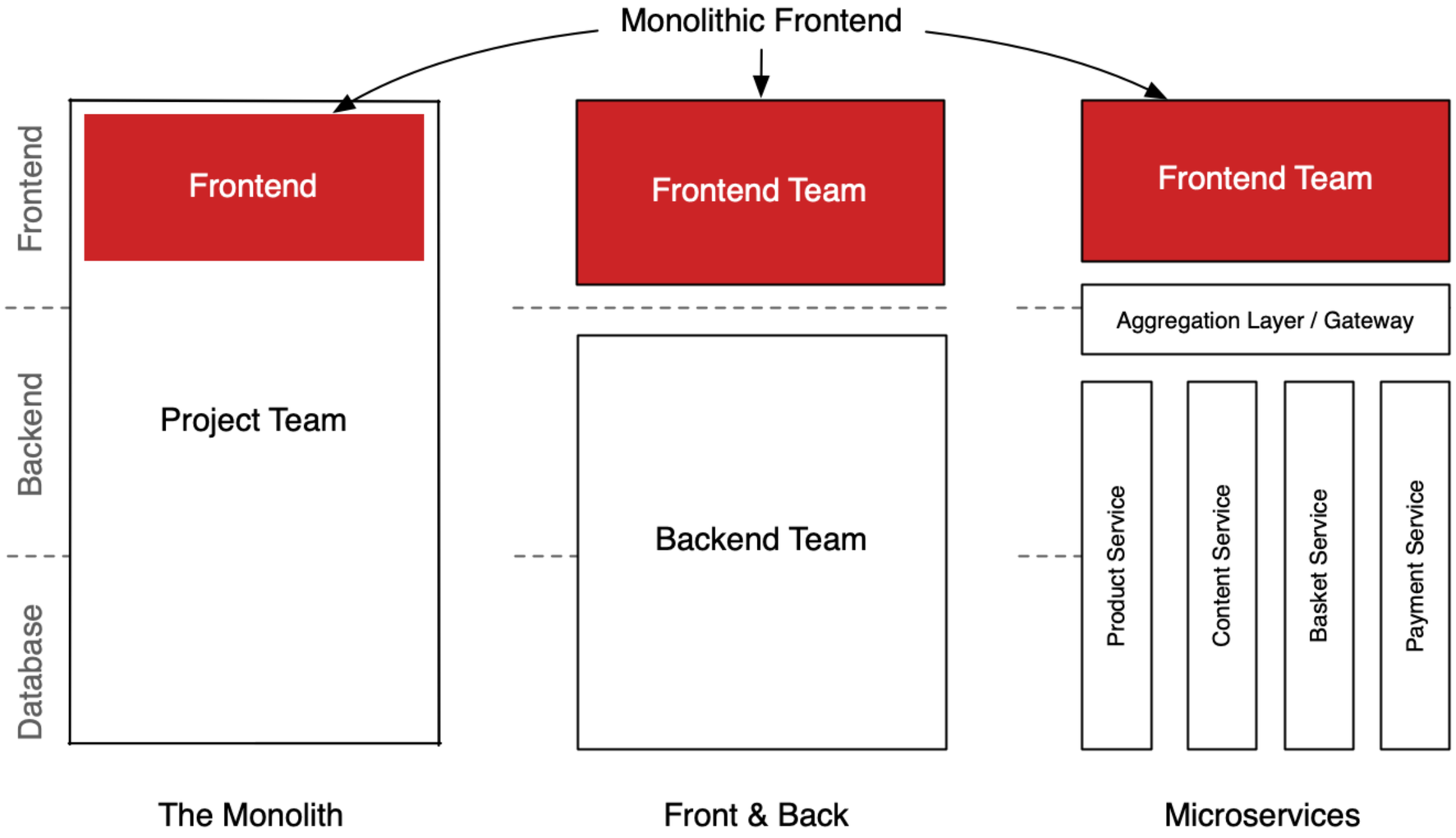
Breaking up the Monolith

New Trend

The Frontend Monolith

presented by

popular architectures nowadays



Micro Frontends

Michael Geers

frontend engineer

naltatis

Twitter & GitHub

author of the book

Micro Frontends in Action

and the website **micro-frontends.org**

building tailored e-commerce systems

neuland:::

Büro für Informatik

The Agenda

What?

How?

Why?

Myths

What are micro frontends?

THE RADAR Nov. 2016

TECHNIQUES

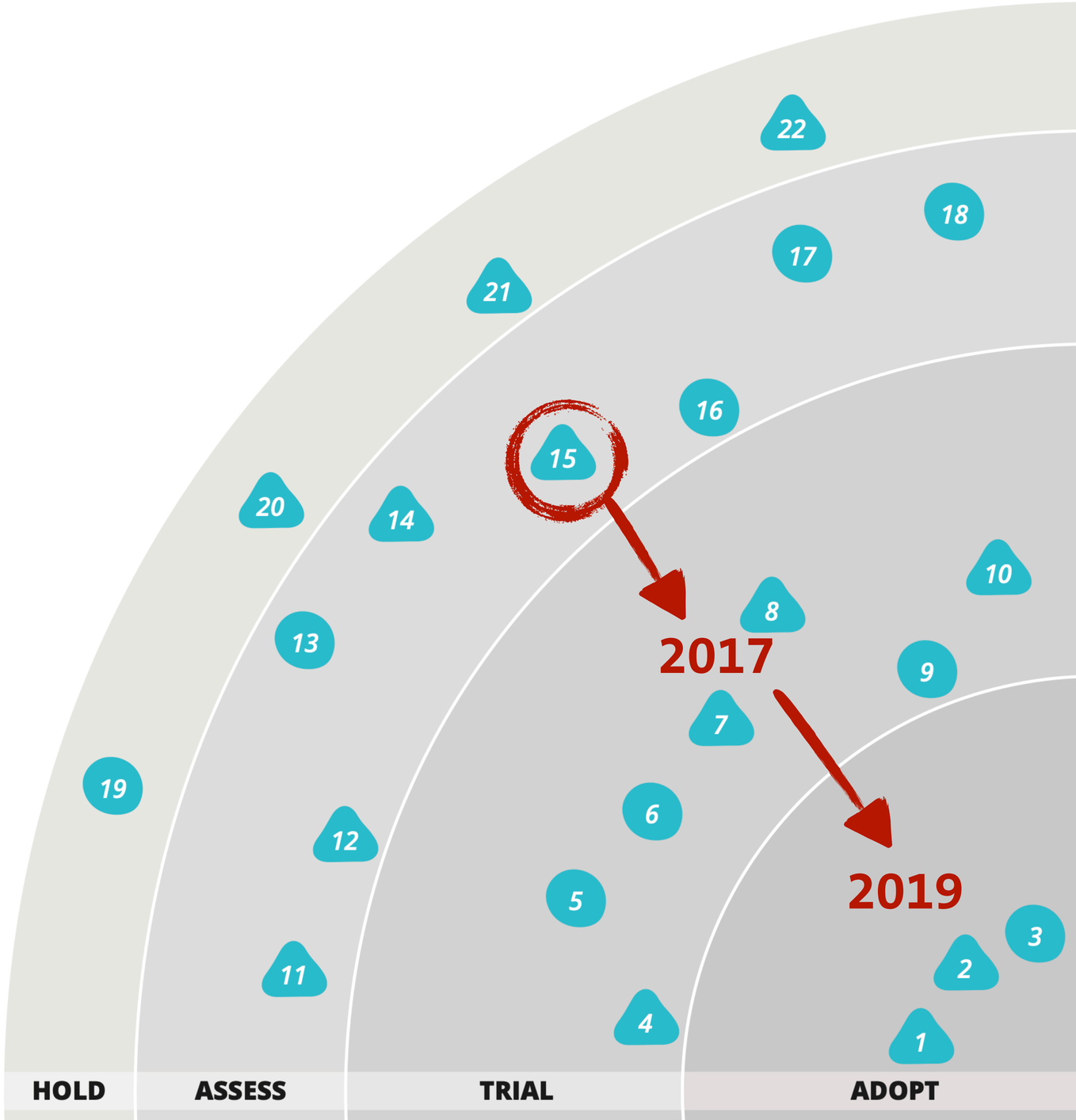
ADOPT
1. Consumer-driven contract testing
2. Pipelines as code new
3. Threat Modeling

TRIAL
4. APIs as a product new
5. Bug bounties
6. Data Lake
7. Hosting PII data in the EU
8. Lightweight Architecture Decision Records new
9. Reactive architectures
10. Serverless architecture

ASSESS
11. Client-directed query new
12. Container security scanning new
13. Content Security Policies
14. Differential privacy new
15. Micro frontends new
16. OWASP ASVS
17. Unikernels
18. VR beyond gaming

HOLD
19. A single CI instance for all teams
20. Anemic REST new
21. Big Data envy
22. Cloud lift and shift

source: ThoughtWorks Technology Radar



A Definition

Our preferred (and proven) approach is to split the browser-based code into micro frontends.

In this approach, the web application is broken down into its features, and each feature is owned, frontend to backend, by a different team.

This ensures that every feature is developed, tested and deployed independently from other features. …

source: ThoughtWorks Technology Radar

A Definition 1/3

Our preferred (and proven) approach is to split the browser-based code into micro frontends.

In this approach, the web application is broken down into its features, and each feature is owned, frontend to backend, by a different team.

This ensures that every feature is developed, tested and deployed independently from other features. …

source: ThoughtWorks Technology Radar

cutting

pages & fragments*

* aka includable Micro Frontends, Podlets, Parcels, Pilets, …

teams can own one or more **pages**

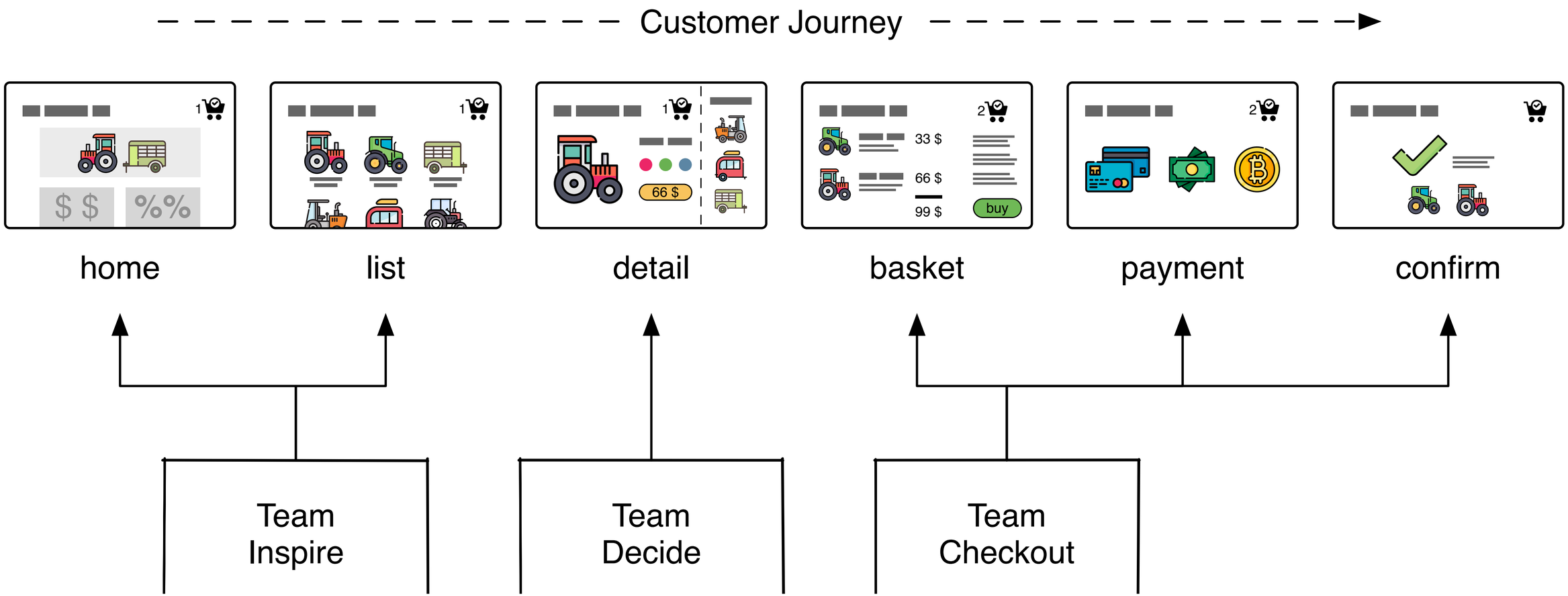


illustration: Micro Frontends in Action

Michael Geers / @naltatis

fragments are embedded mini-applications

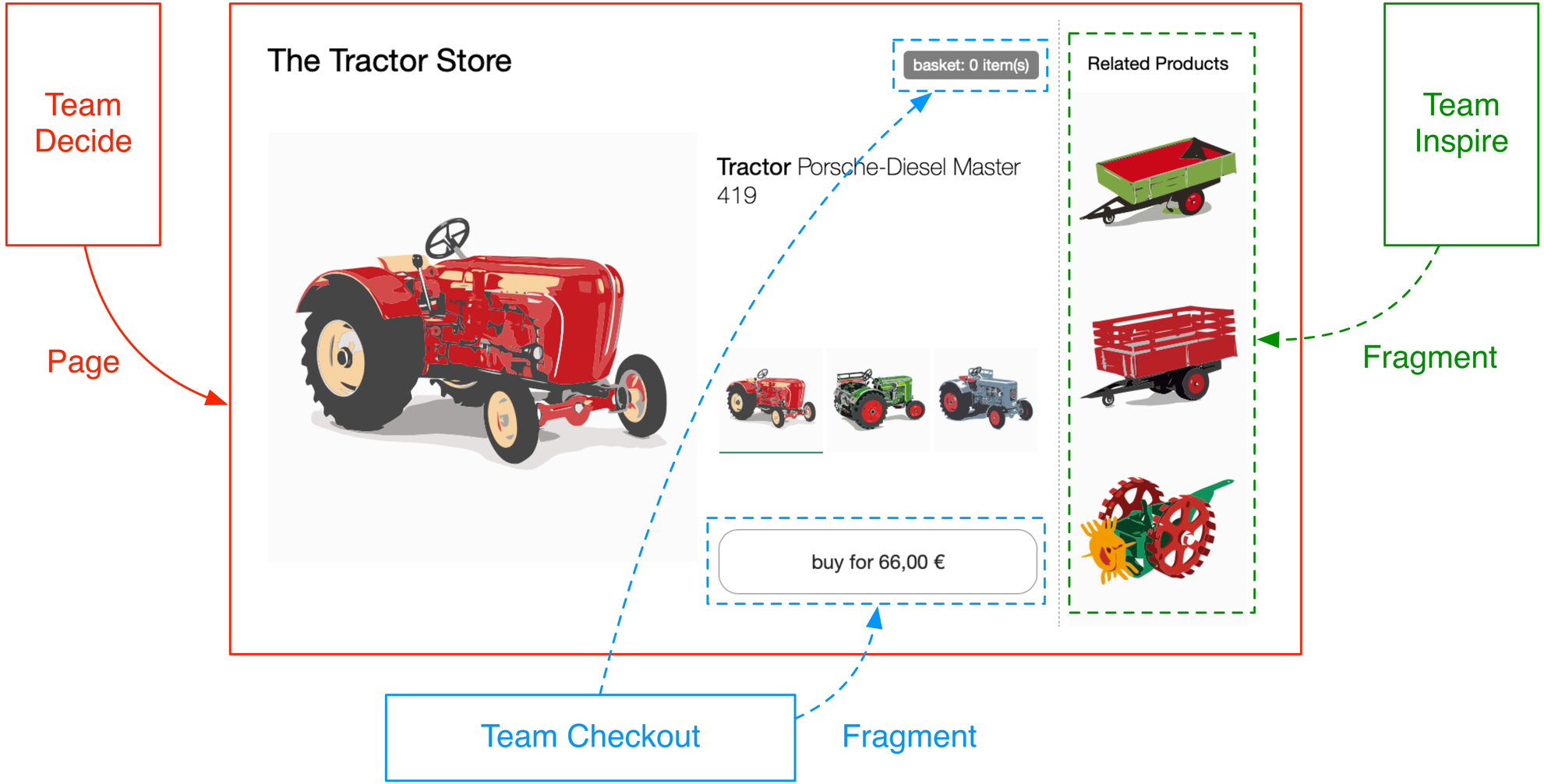


illustration: Micro Frontends in Action

Michael Geers / @naltatis

A Definition 2/3

Our preferred (and proven) approach is to split the browser-based code into micro frontends.

In this approach, the web application is broken down into its features, and each feature is owned, frontend to backend, by a different team.

This ensures that every feature is developed, tested and deployed independently from other features. …

source: ThoughtWorks Technology Radar

Team Missions

Team
Inspire

Mission
helps the customer
to discover products

Team
Decide

Mission
helps making the
buying decision

Team
Checkout

Mission
guides through the
checkout process

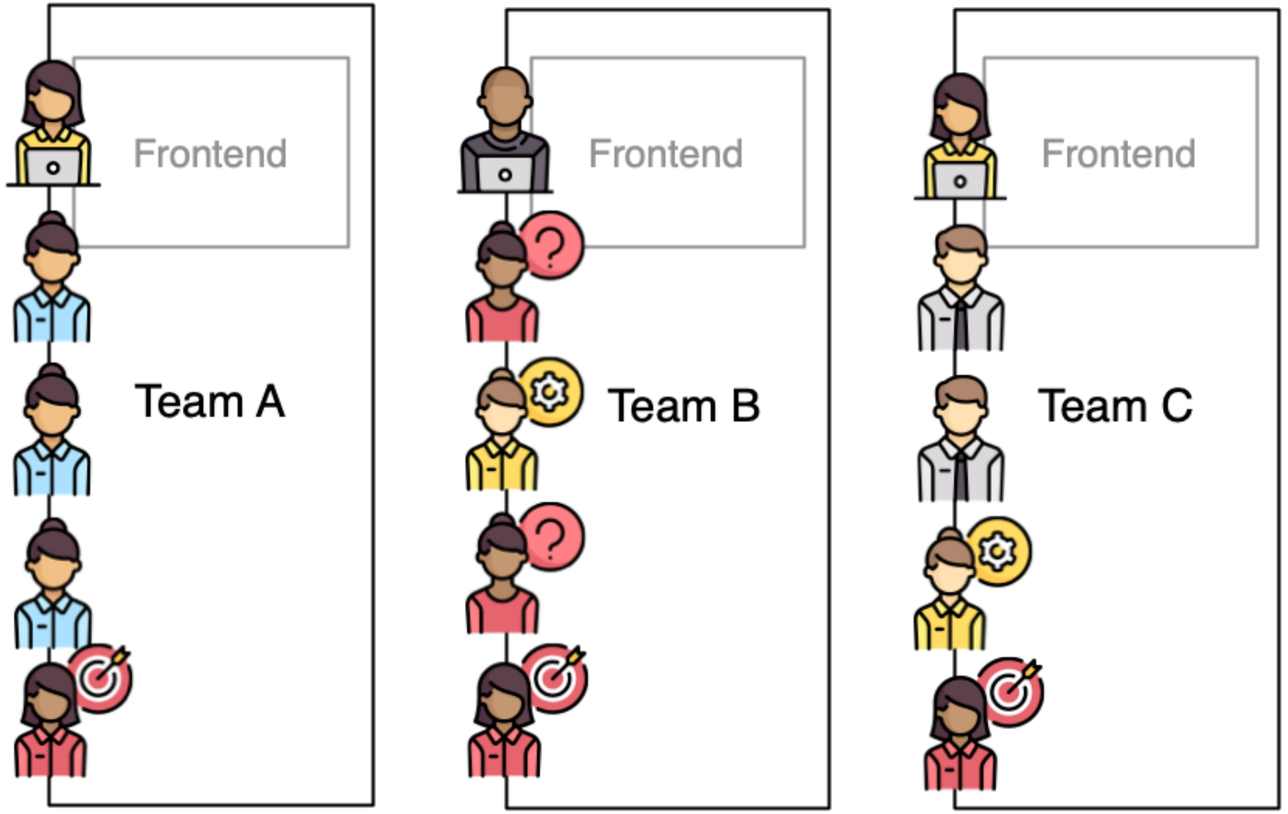
illustration: Micro Frontends in Action

Michael Geers / @naltatis

Independent **Teams** & **Systems**

Cross Functional Teams
ux/design, frontend
backend, data science, ops

End-to-End Responsibility
from user interface
to database

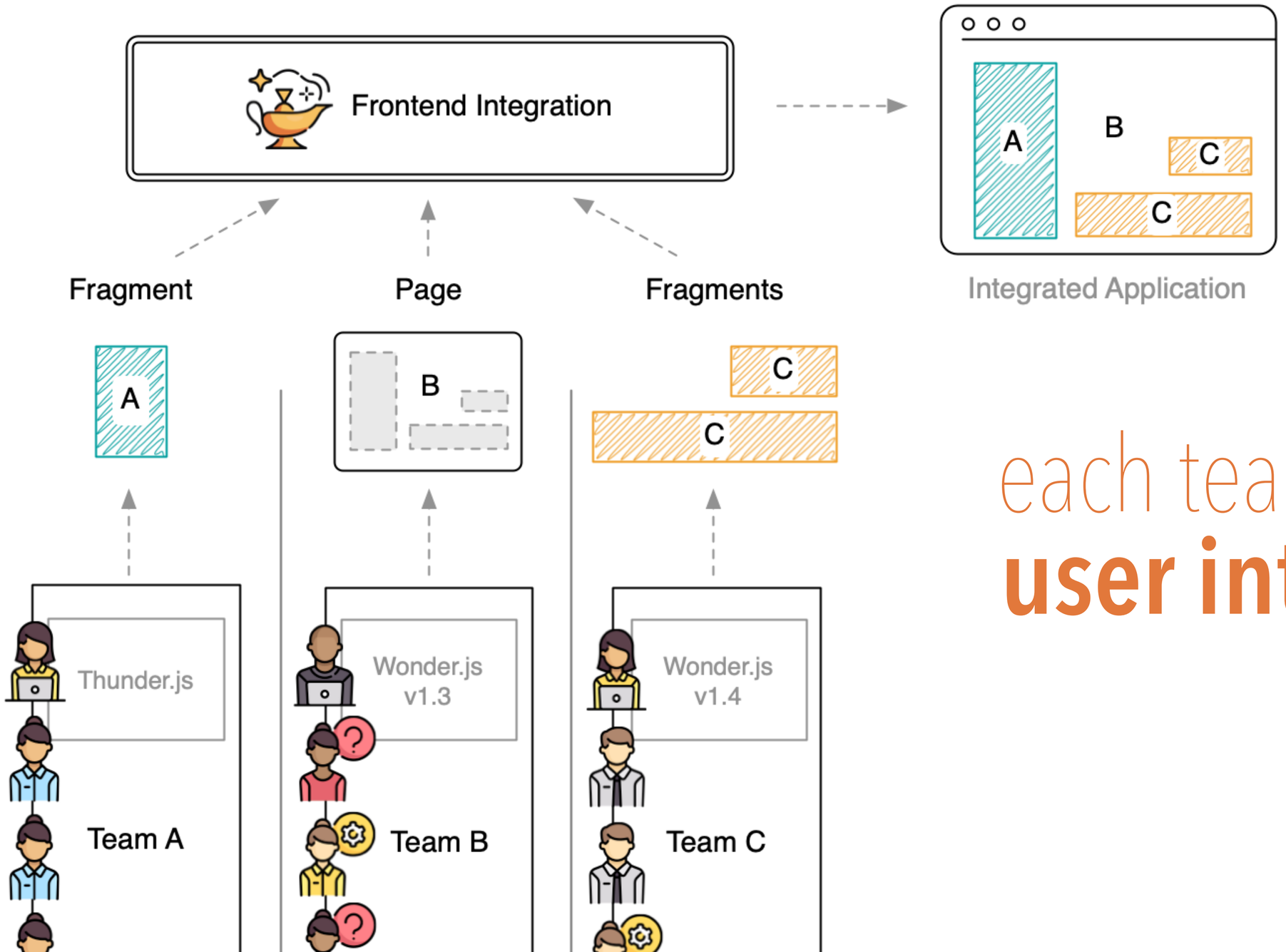


Self-Contained Systems
separate applications that
don't rely on each other

like a chunky microserivce,
but with its own UI

illustration: Micro Frontends in Action

Michael Geers / @naltatis



each team ships
user interface

Michael Geers / @naltatis

A Definition 3/3

Our preferred (and proven) approach is to split the browser-based code into micro frontends.

In this approach, the web application is broken down into its features, and each feature is owned, frontend to backend, by a different team.

This ensures that every feature is developed, tested and deployed independently from other features. …

source: ThoughtWorks Technology Radar

Independent **Deployments**

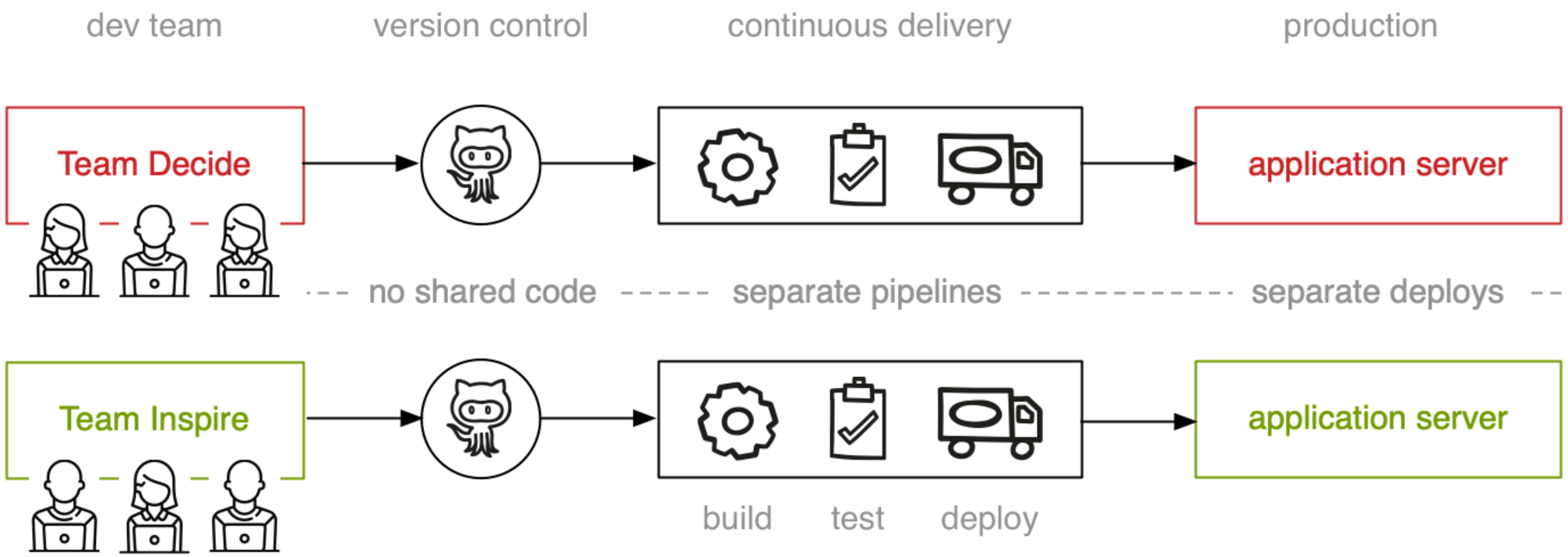


illustration: Micro Frontends in Action

Michael Geers / @naltatis

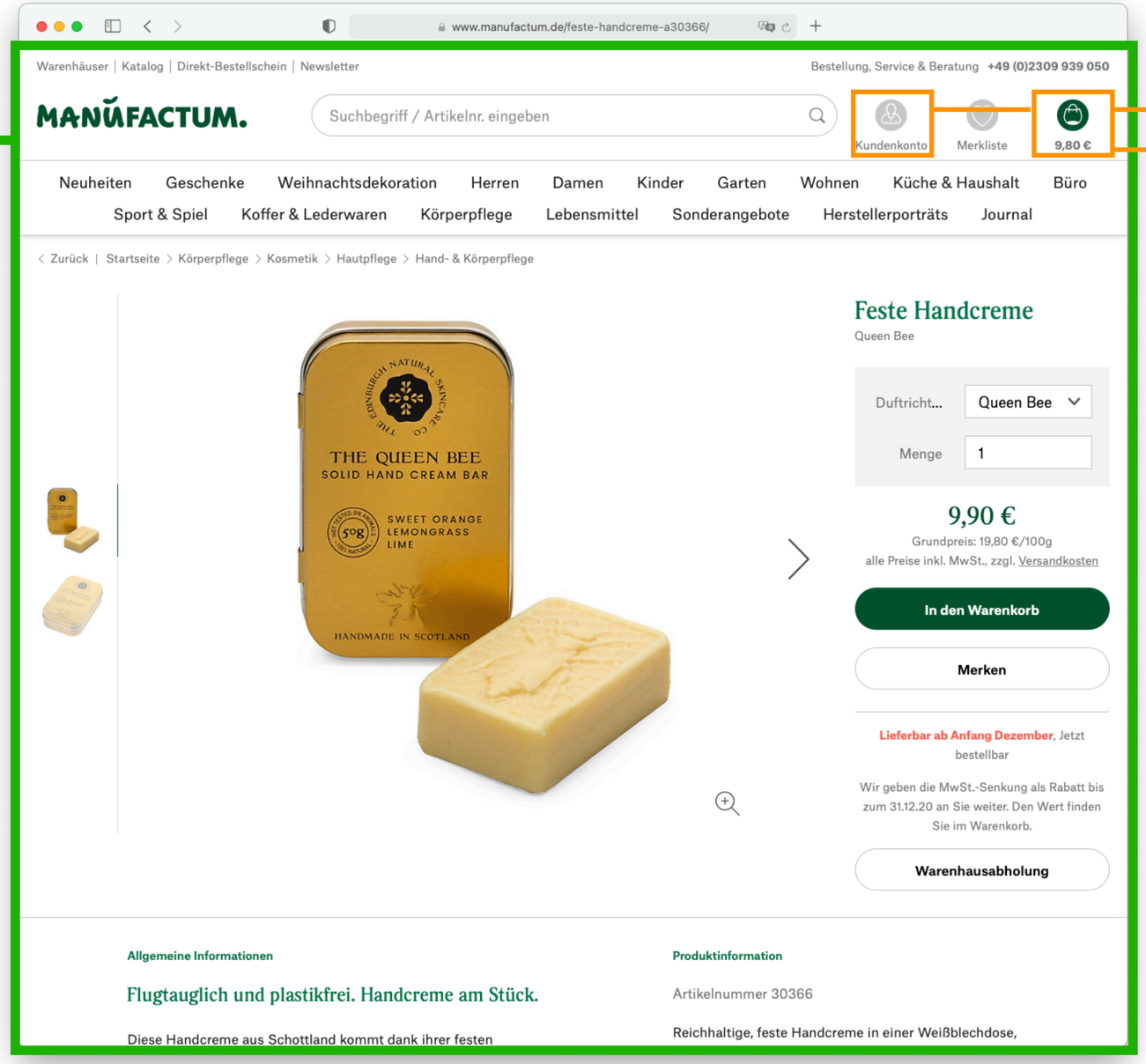
Real World Projects

manufactum.com

2 teams

Team Sell

product page



Team After-Sell

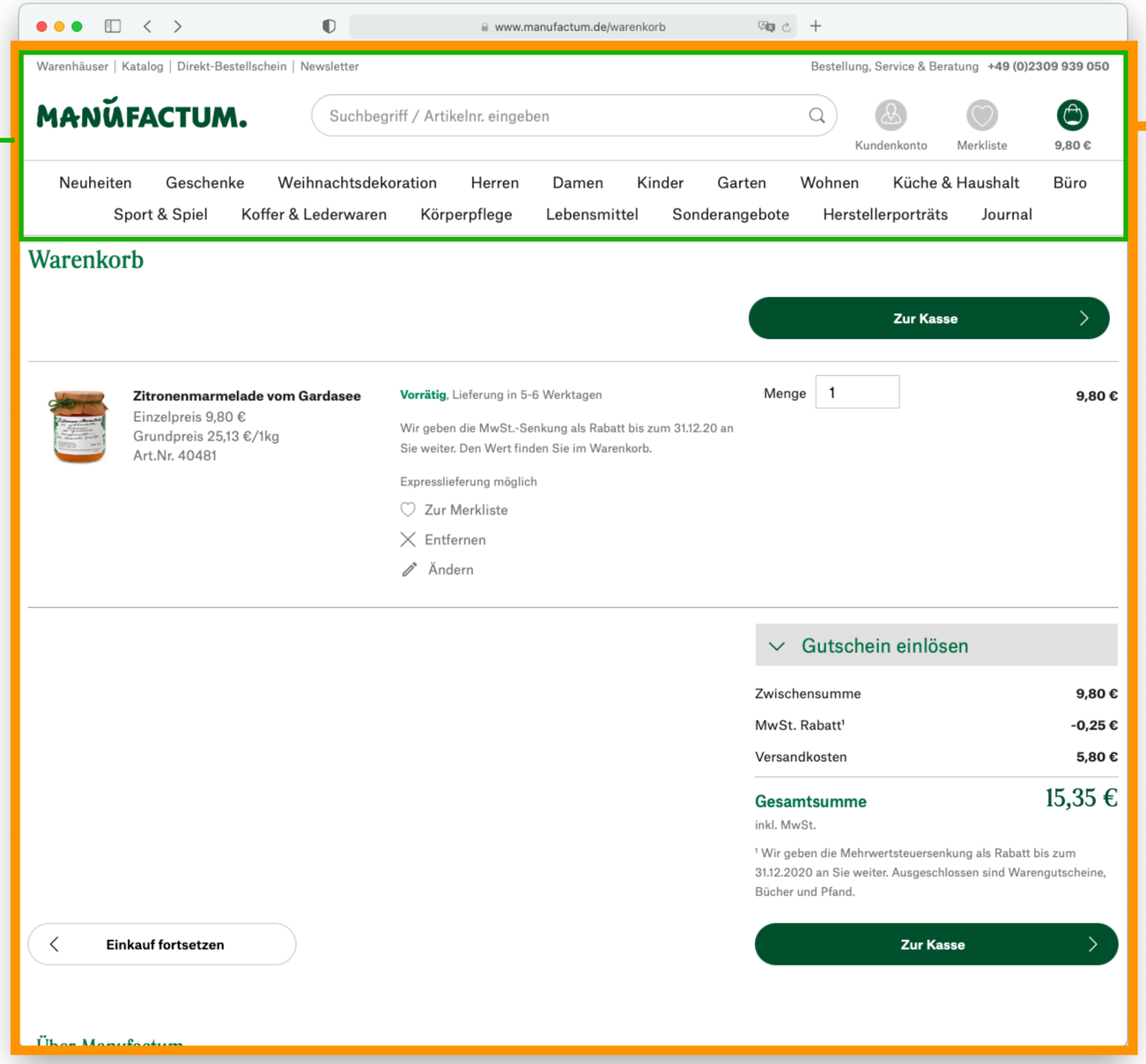
login status fragment

mini basket fragment

Michael Geers / @naltatis

Team Sell
navigation fragment

Team After-Sell
basket page



Michael Geers / @naltatis

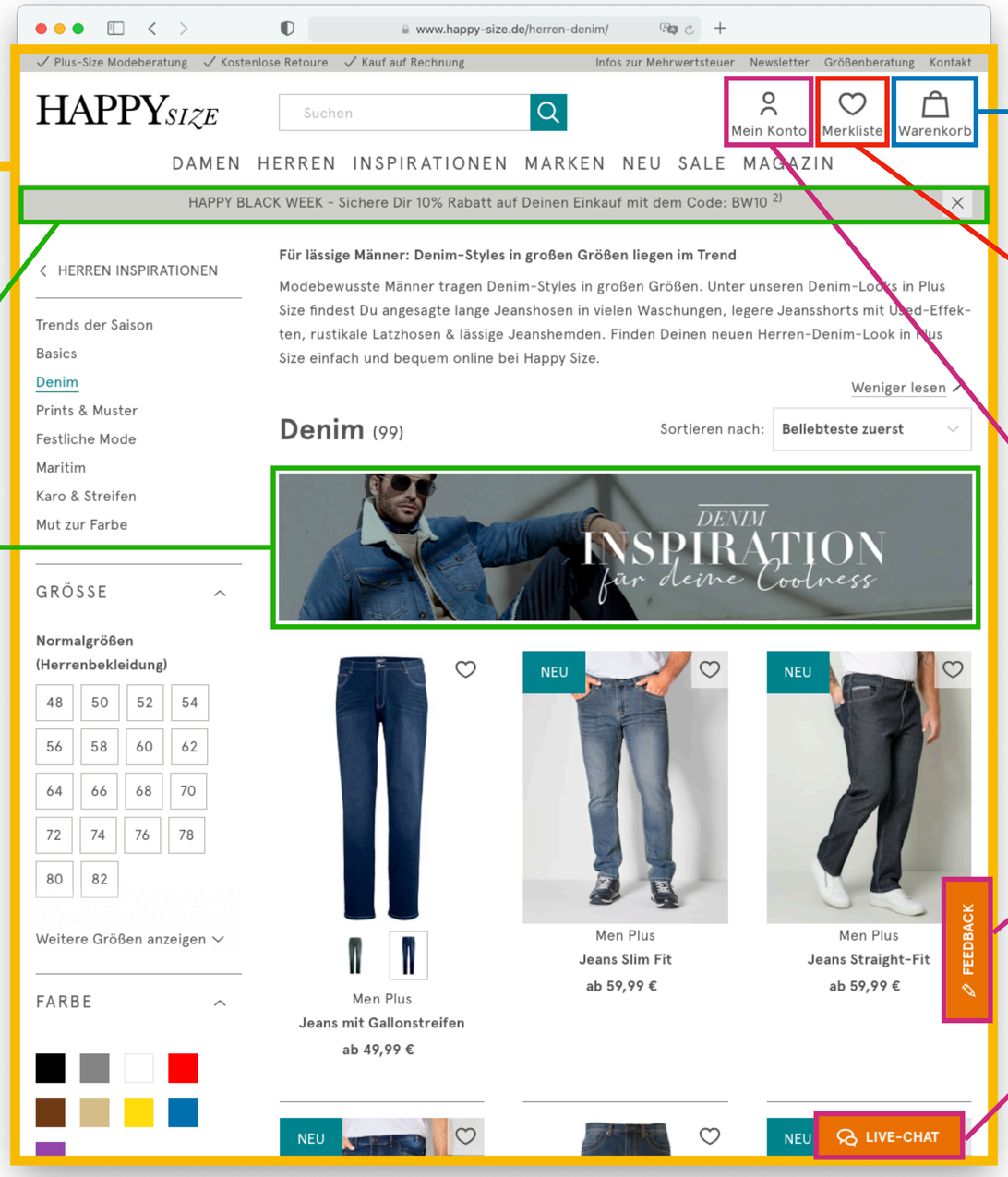
Klingel Group

5 teams from 4 software companies

15 brands, 12 countries, ~65 shops

Team Search
category page

Team Inspire
promotion
inspiration

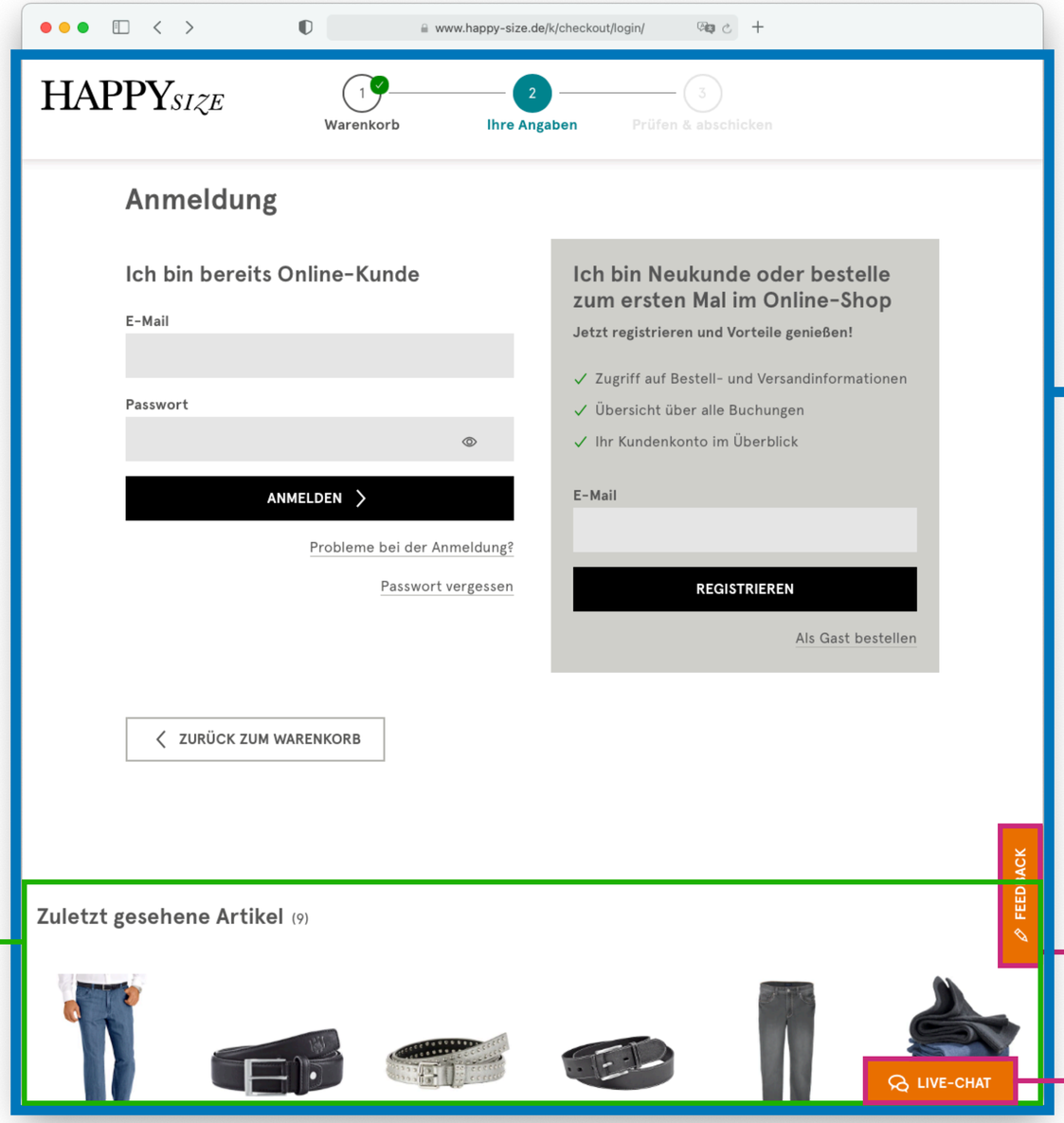


Team Checkout
mini basket

Team Decide
wishlist

Team Account
login status
feedback
support

Michael Geers / @naltatis



Team Checkout
login page

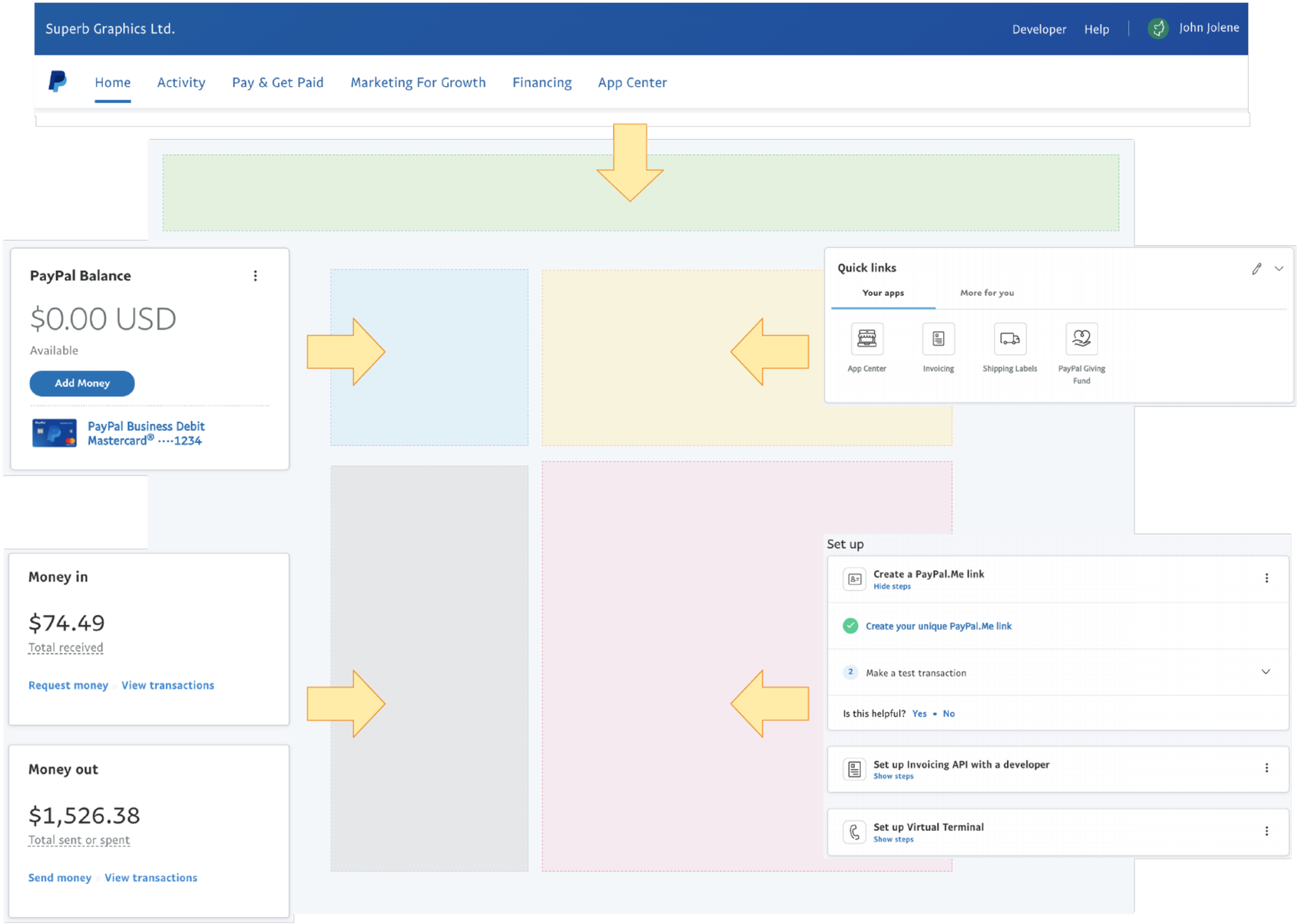
Team Inspire
recommendations

Team Account
feedback
support

Michael Geers / @naltatis

PayPal

PayPal



PayPal Egnineering Blog

Michael Geers / @naltatis

Who uses micro frontends?

logos link to the sources

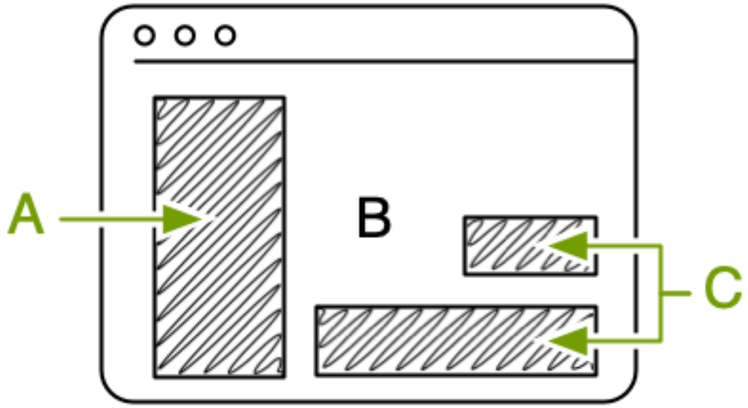
Michael Geers / @naltatis

How can they be implemented?

putting it back together

What needs to be integrated?

Composition



Routing & Page Transitions

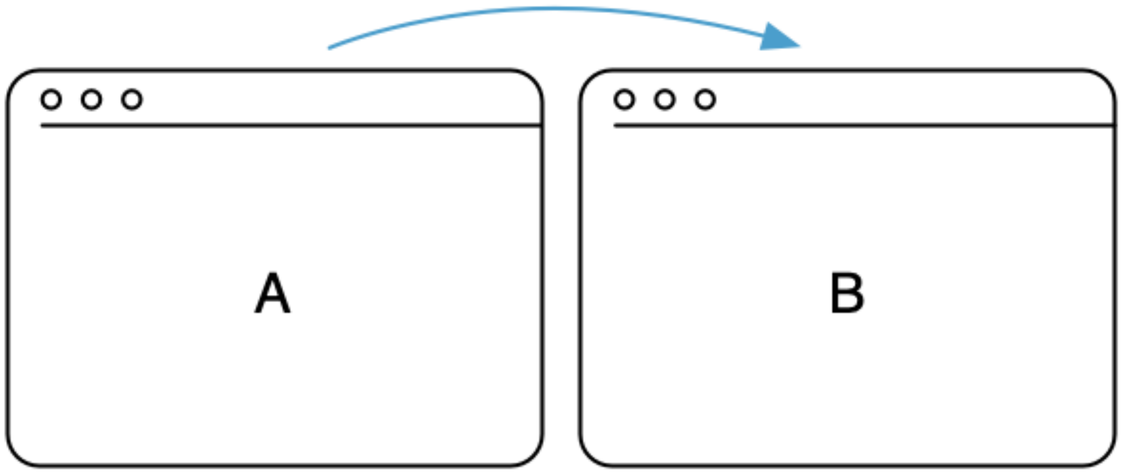
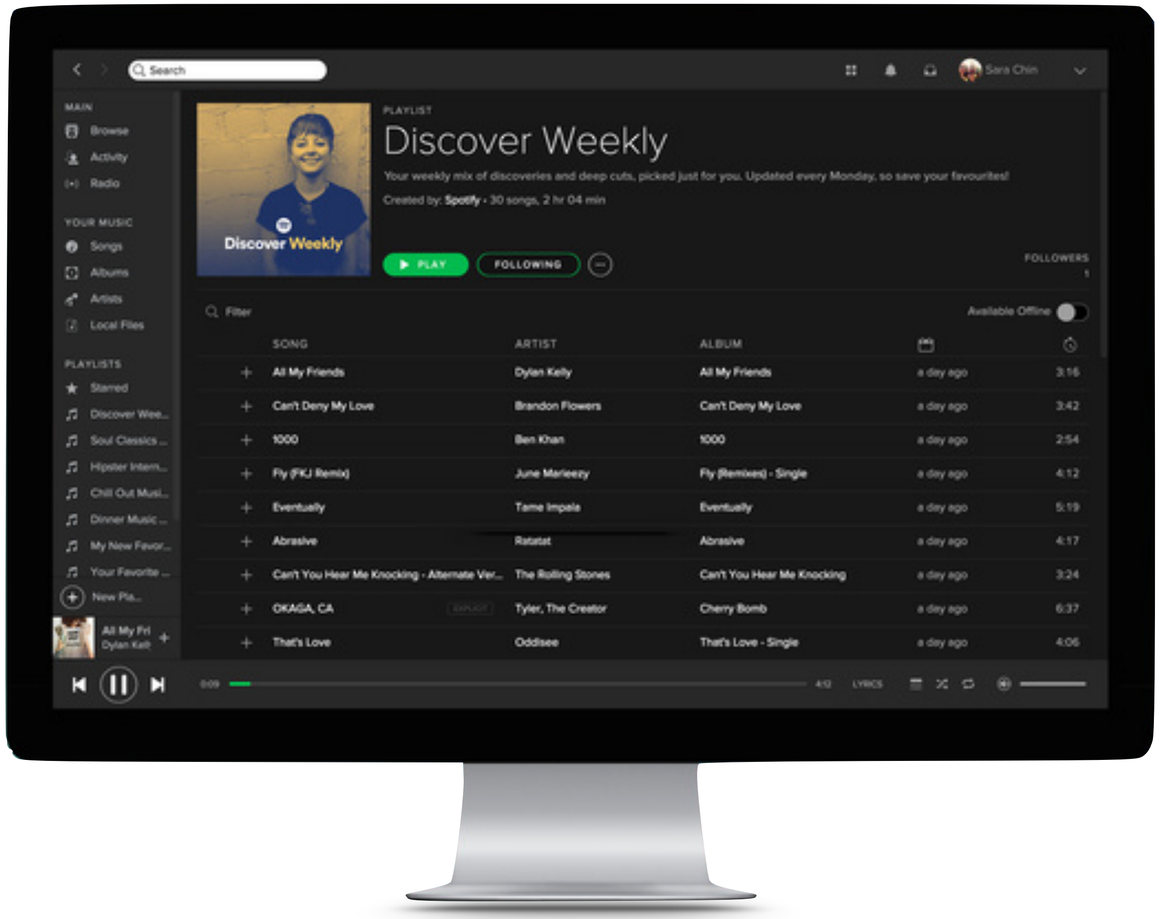


illustration: Micro Frontends in Action

Michael Geers / @naltatis

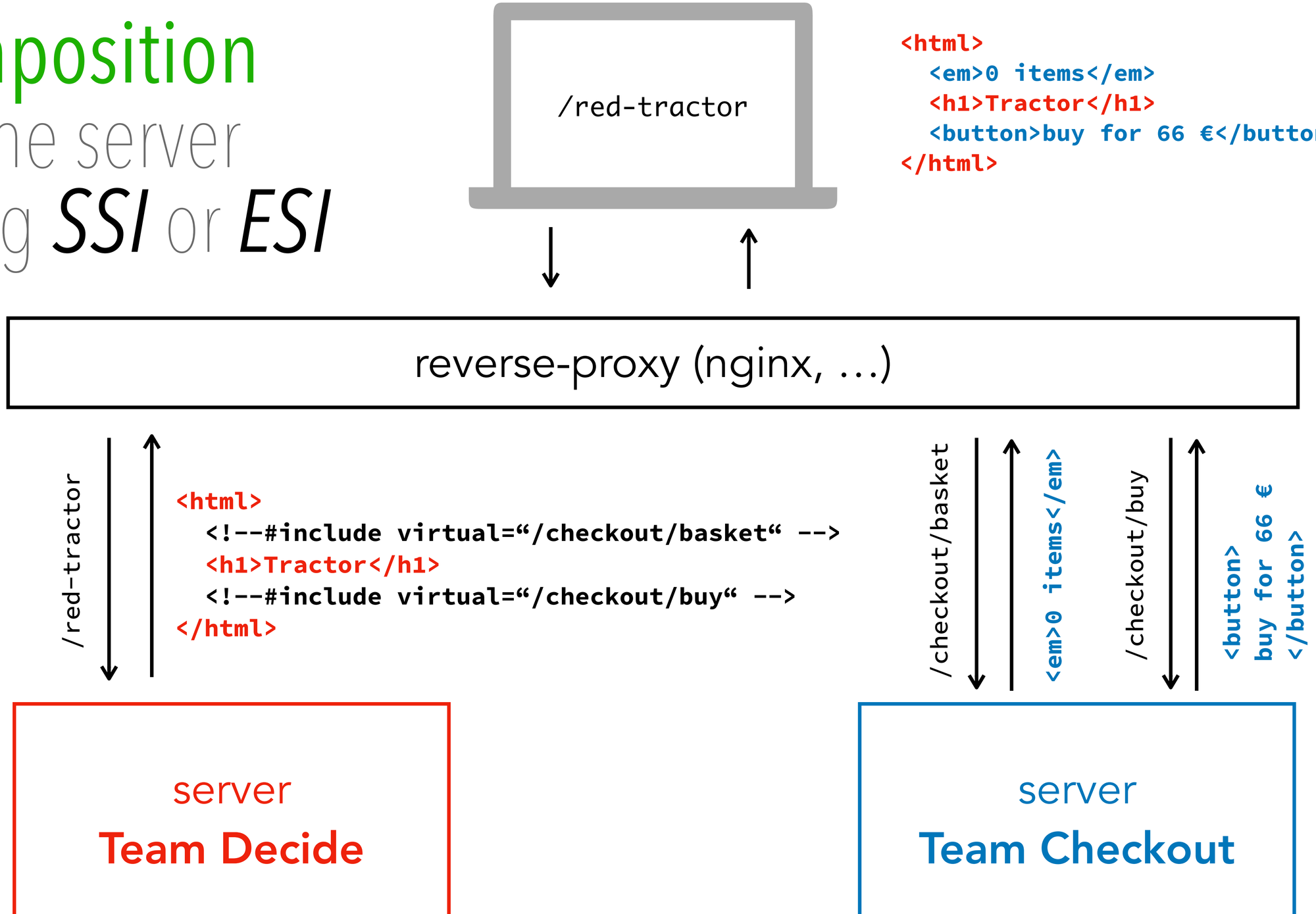
composition with *iframes*



Spotify web player (until early 2019) & Desktop App

https://labs.spotify.com/2019/03/25/building-spotifys-new-web-player/

composition
on the server using *SSI* or *ESI*



composition
in the browser
with *Custom Elements*

```
<checkout-basket>
</checkout-basket>
```

basket: 0 item(s)

Tractor Porsche-Diesel Master 419

buy for 66,00 €

Related Products

```
<inspire-reco sku="t_red">
</inspire-reco>
```

```
<checkout-buy sku="t_red">
</checkout-buy>
```

Michael Geers / @naltatis

page transitions using links

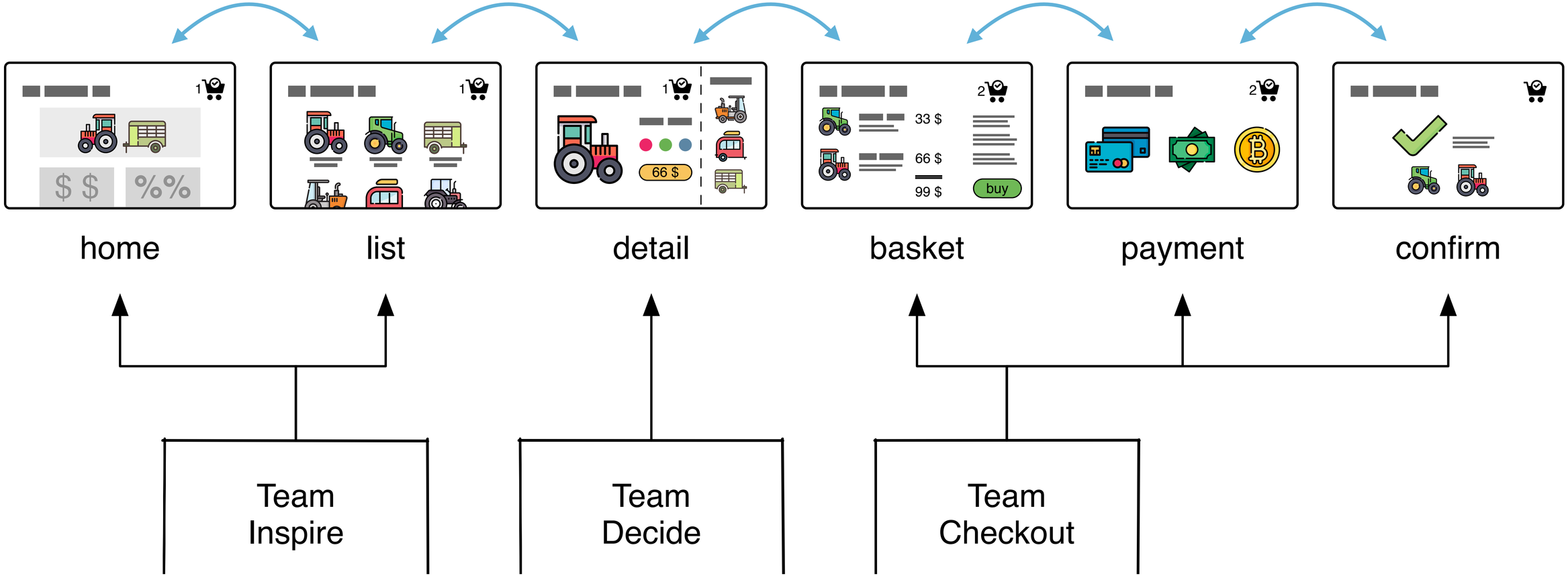
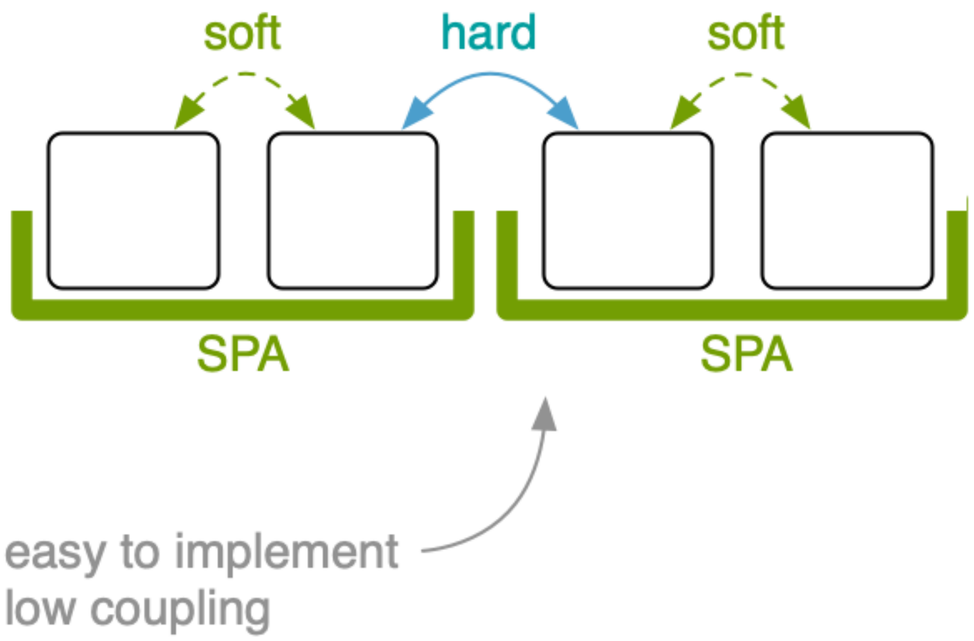


illustration: Micro Frontends in Action

Michael Geers / @naltatis

page transitions between SPAs

Linked Single Page Apps



Unified Single Page App

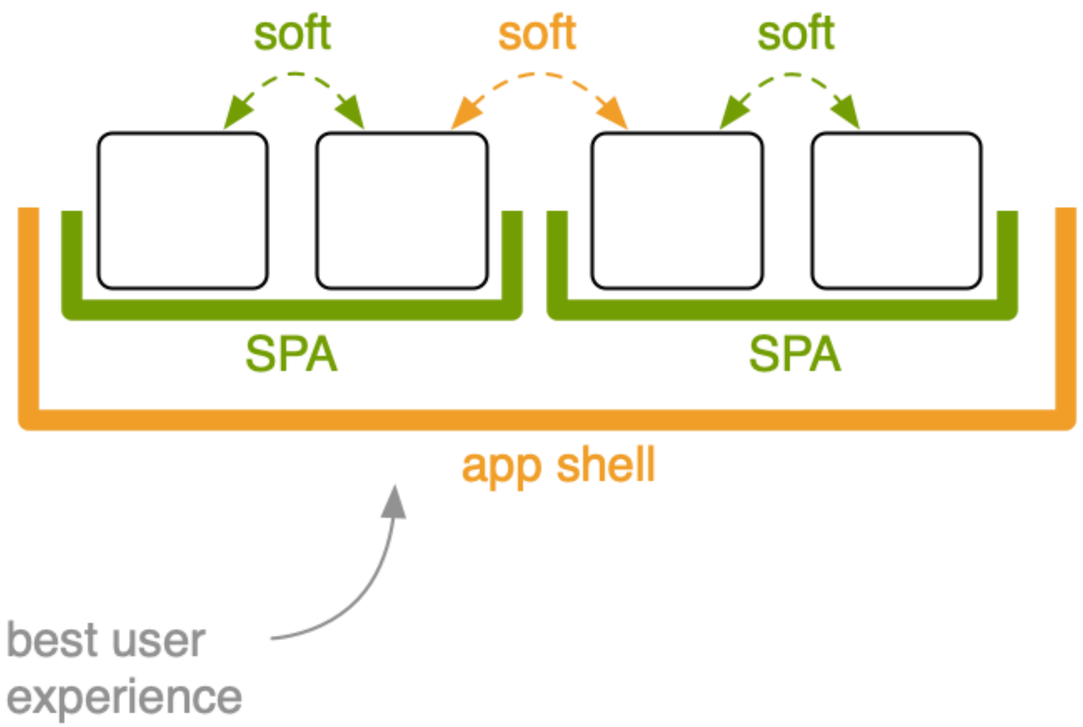


illustration: Micro Frontends in Action

Michael Geers / @naltatis

jobs of the application shell?

- central entry point for the user
- can load, start and stop all SPAs
- switches between SPAs if necessary

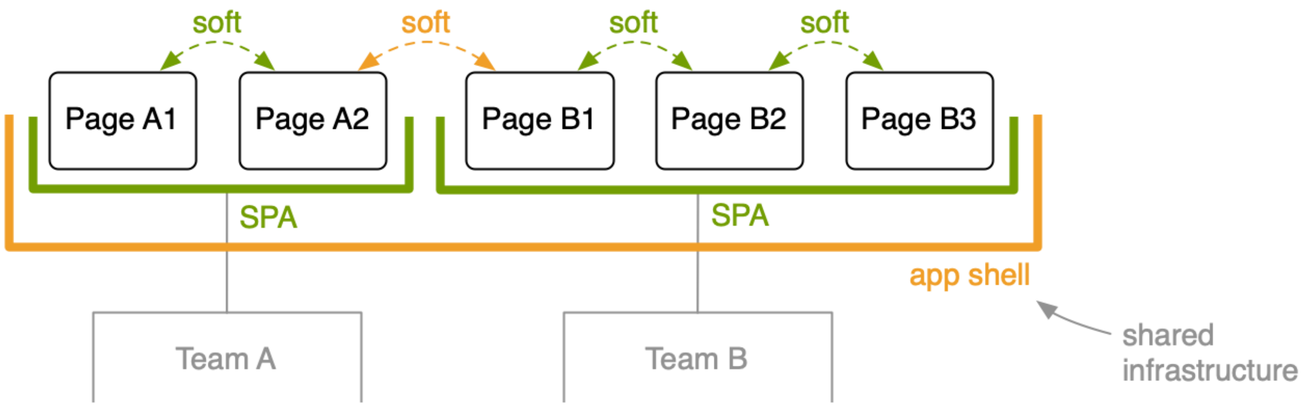


illustration: Micro Frontends in Action

single-spa

- ready-to-use application shell
- most popular solution
- plugins for nearly all frameworks
- also supports composition

libraries & meta frameworks

qiankun
https://qiankun.umijs.org

Podium
https://podium-lib.io

icestark
https://ice-lab.github.io/icestark/

Piral
https://piral.io

OpenComponents
https://github.com/opencomponents/oc

microfe
https://github.com/onerzafer/microfe-client

Ara
https://github.com/ara-framework

meta-spa-router
https://github.com/manfredsteyer/meta-router

Luigi
https://github.com/SAP/luigi

Tailor
https://github.com/zalando/tailor

puzzle.js
https://github.com/puzzle-js

ILC
https://github.com/namecheap/ilc

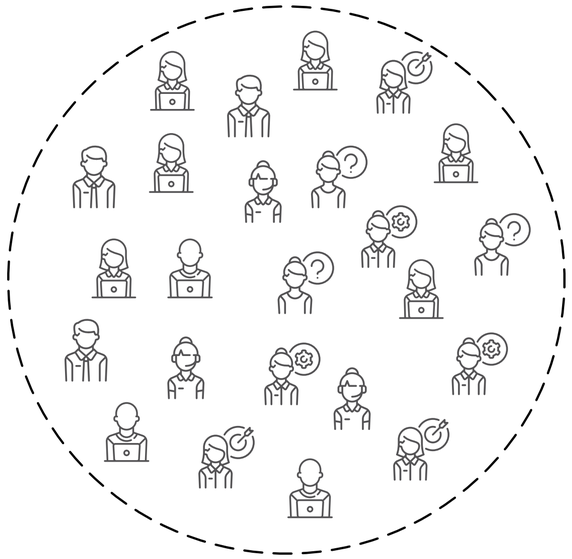
Michael Geers / @naltatis

Why would I use micro frontends?

🥇 faster feature development ...

three 8-person **teams**

are **more effective** than

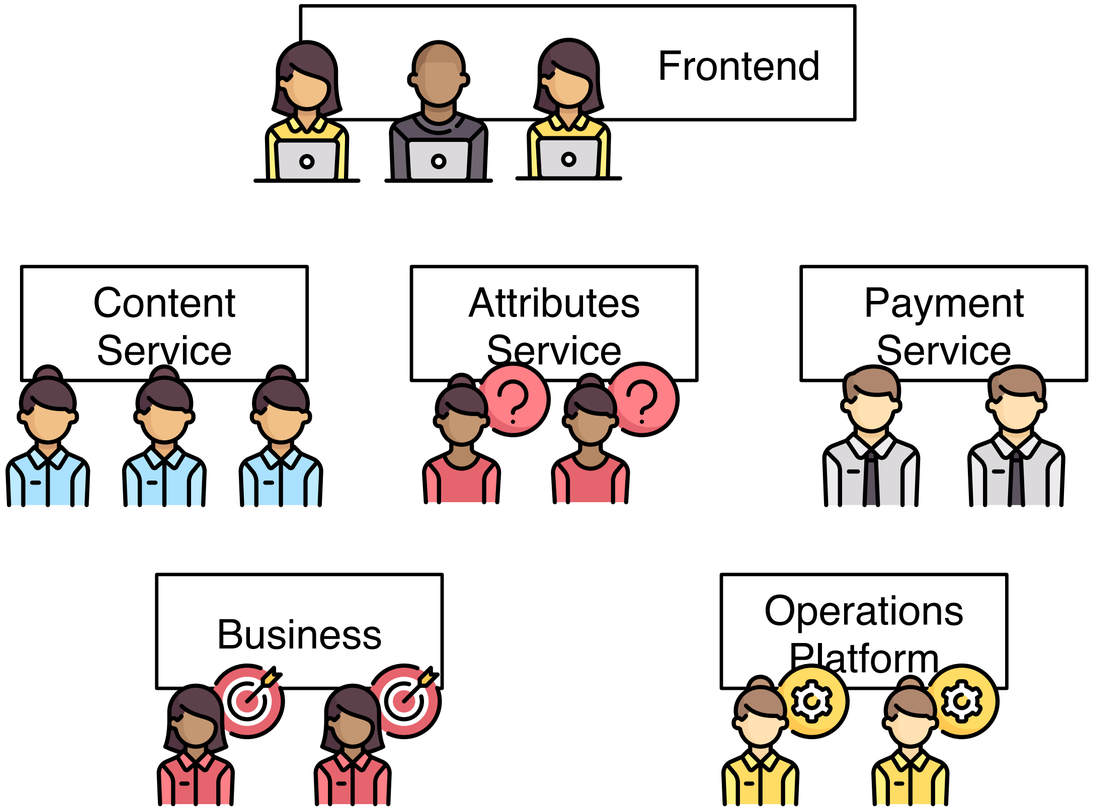


one 24-person **team**

Michael Geers / @naltatis

new balance
new balance
KEEN
teams should be **decoupled**

Specialist Teams



Cross Functional Teams

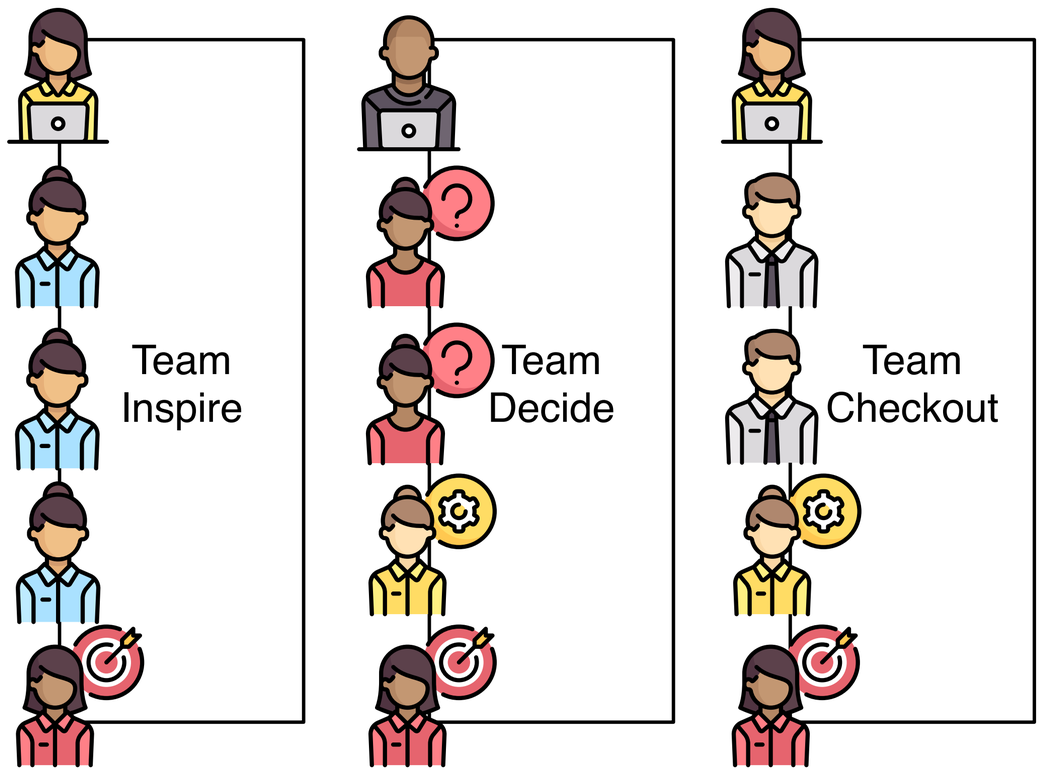
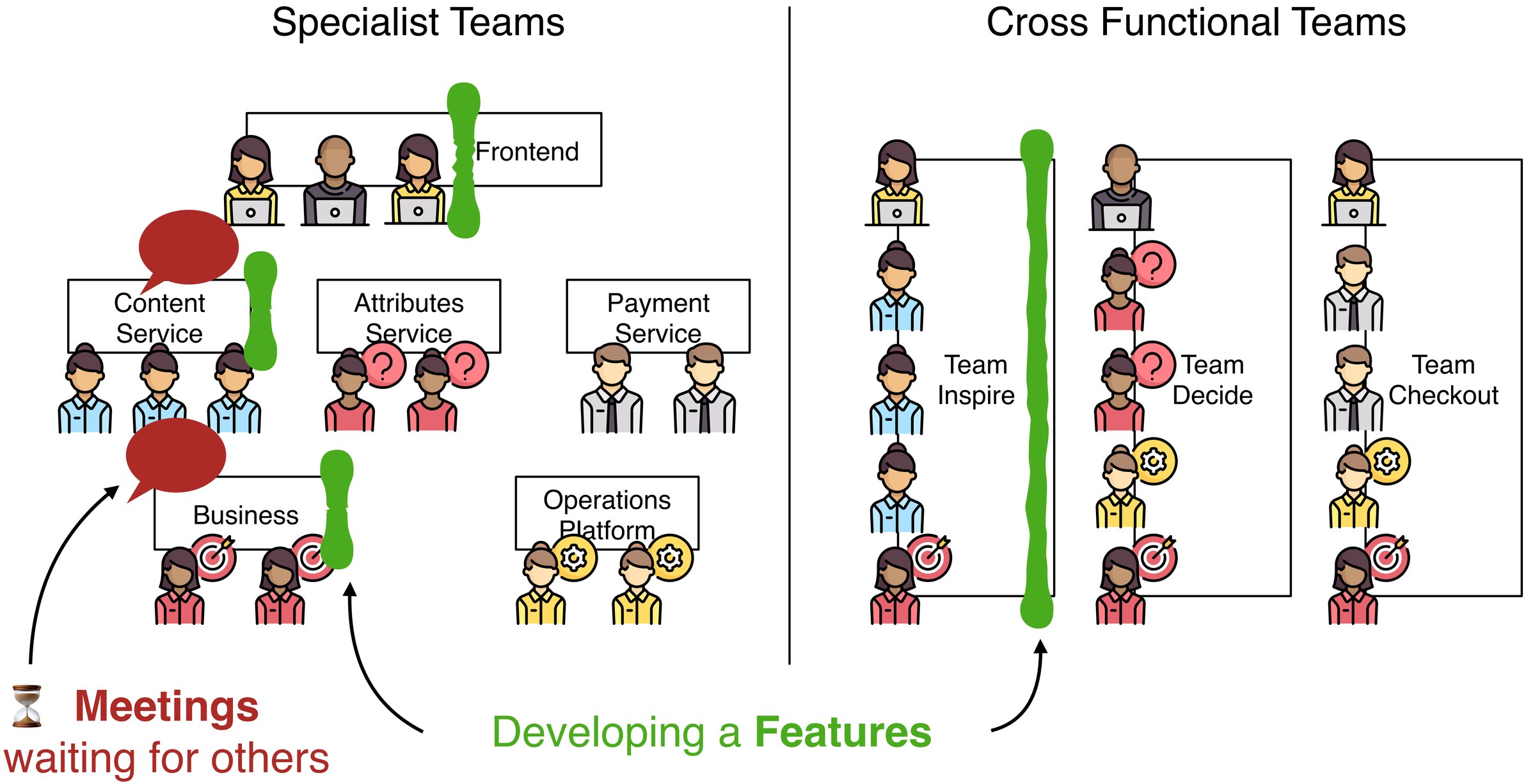


illustration: Micro Frontends in Action

Michael Geers / @naltatis



Michael Geers / @naltatis

customer focus

direct feedback

no pure API teams

frontend renovation

without throwing everything away

technology change becomes a team decision

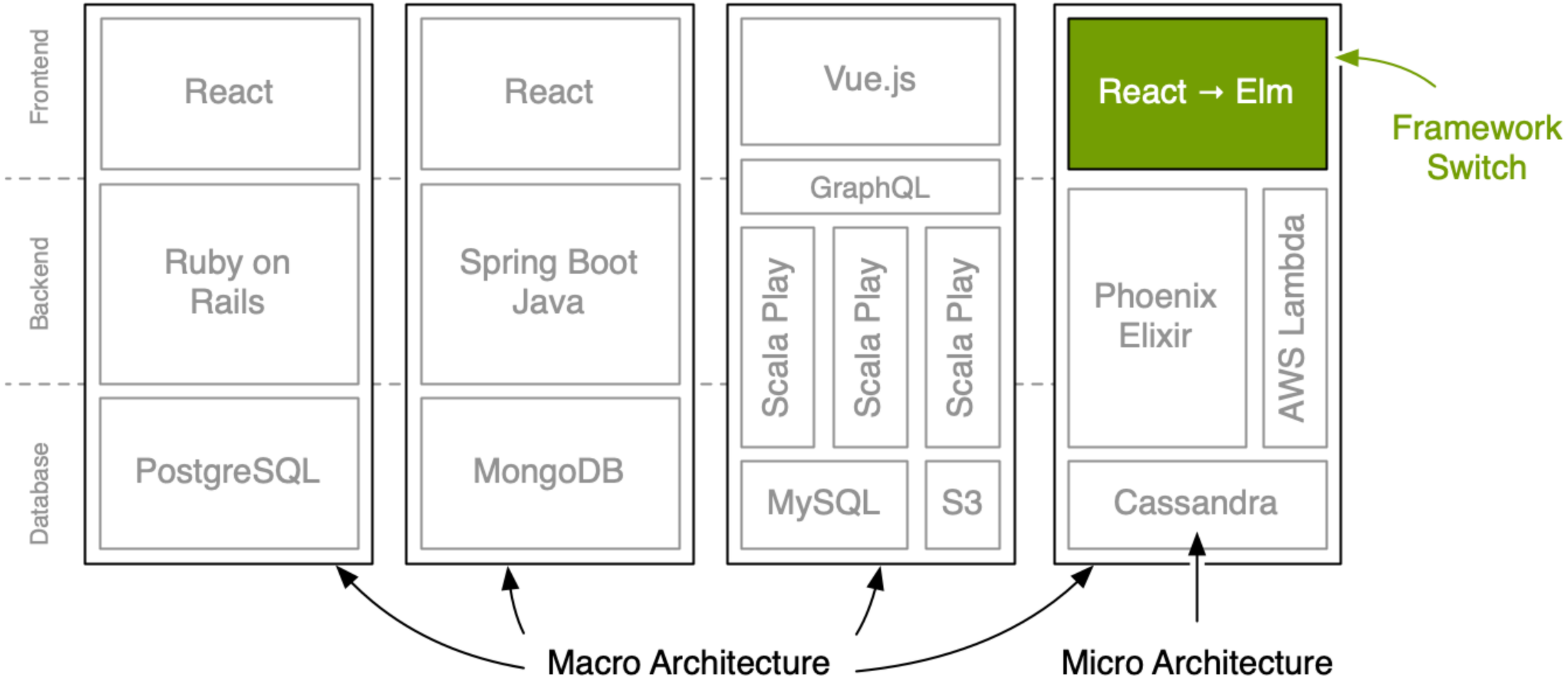
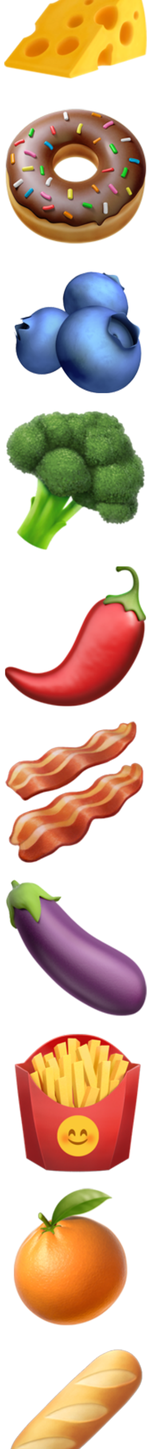


illustration: Micro Frontends in Action

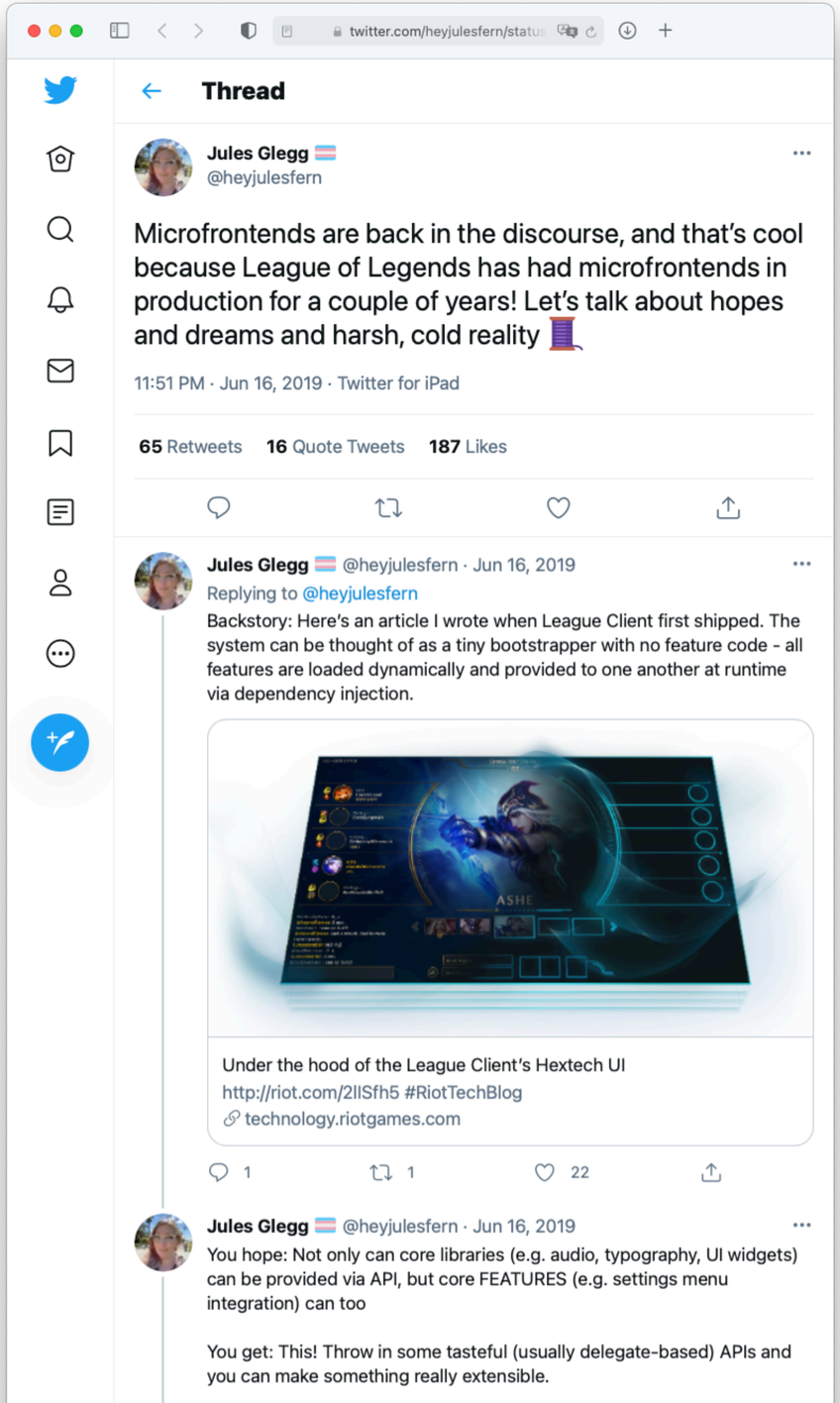
Michael Geers / @naltatis

micro frontend **myths**



micro frontends **produce ugly tech mixes**

how micro frontends **did not work out** for League of Legends



Twitter thread by Jules Glegg / @heyjulesfern (June 2019)

micro frontend **anarchy**

just because you can, doesn't mean you should

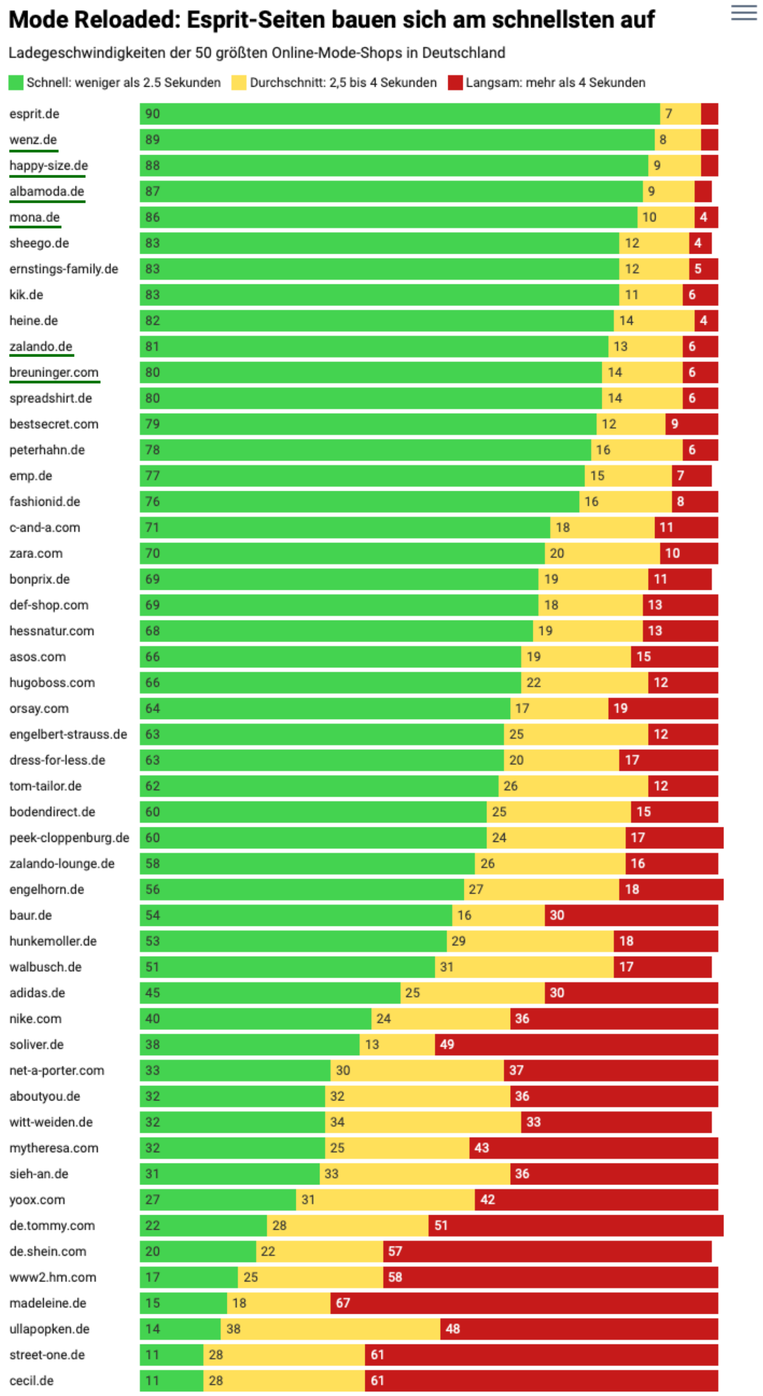
- agree upon a tech stack
- make it easy to get started (starter kit, blueprint)
- exchange learnings between teams
- try not to over-regulate

Technology Radar #23

micro frontends **are bad for performance**

Analyses of loading performance (LCP) in German fashion e-commerce shops

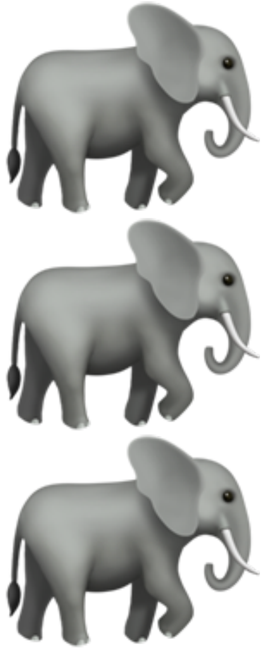
all using a
micro frontends
architecture



micro frontend sites
can be very fast

or slow - it's mostly about good engineering

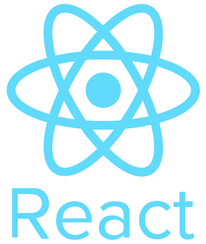
source: Textil Wirtschaft



but **what about** the **redundancy**?

smaller tools mean less redundancy

larger runtime



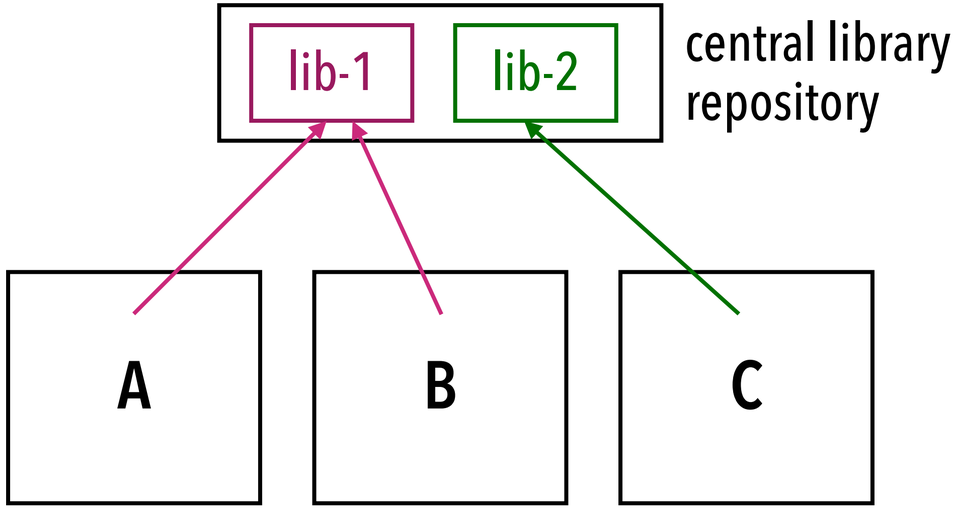
svelte

lit-html

small runtime

reusing **vendor libraries** between systems

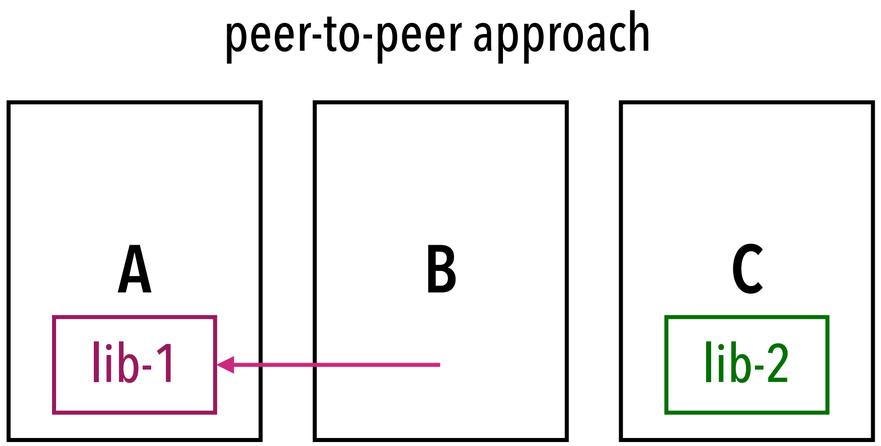
with **import maps**



applications can **reference central libraries** to reduce their bundle size

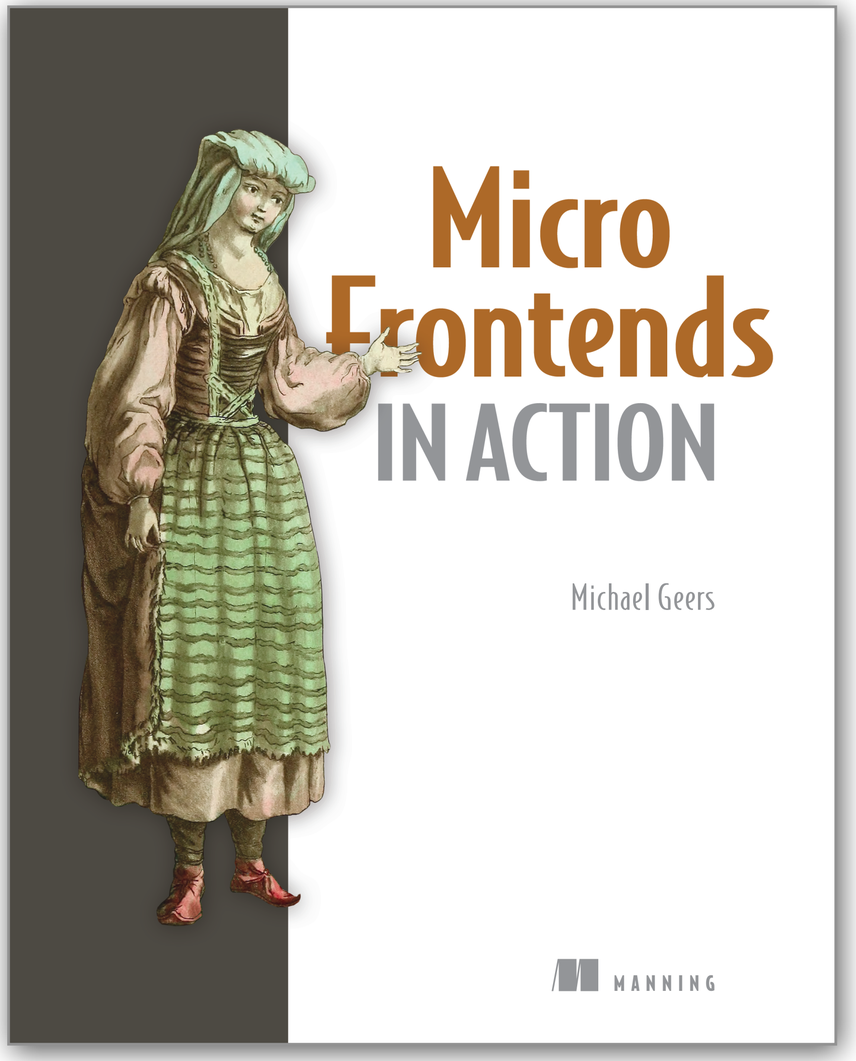
https://github.com/systemjs/systemjs/blob/master/docs/import-maps.md

with **module federation**



applications can **reuse already loaded libraries** from inside other applications

https://module-federation.github.io



I've written a book

270 pages

143 illustrations

21 examples

64 listings

35% off with code **ctwemerge21**

and all other books at manning.com

book examples **the-tractor.store**

the-tractor.store

The Tractor Store

micro frontends sample code

view on GitHub | by Michael Geers

The Book

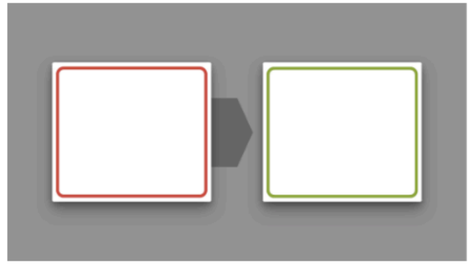
Curious about micro frontends? Grab a copy of my book **Micro Frontends in Action** to learn about the concepts, its benefits, integration techniques and common pitfalls.

Buy the Book

Amazon | micro-frontends.org

Chapter 2: My First Micro Frontends Project

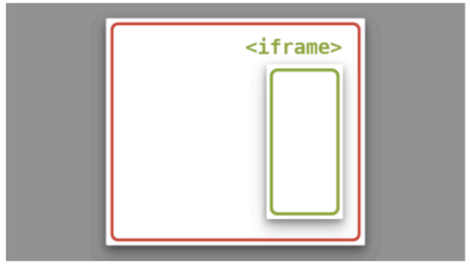
Pages & Links



try out | view code

Topics: page transition, links

iFrames



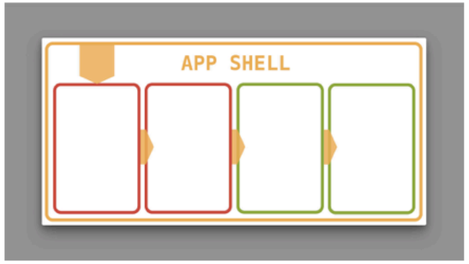
try out | view code

Topics: composition, iframe

the-tractor.store

Chapter 7: Client-side Routing

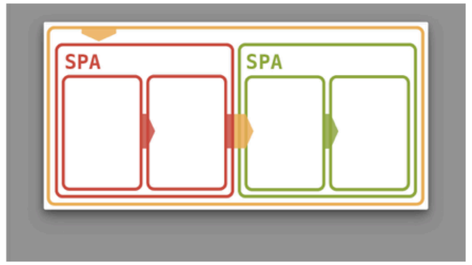
Client-side Routing (Flat)



try out | view code

Topics: app-shell, routing, client-side, Web Components

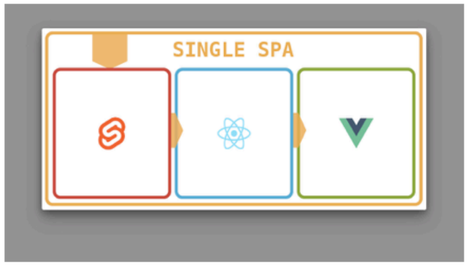
Client-side Routing (Two-Tiered)



try out | view code

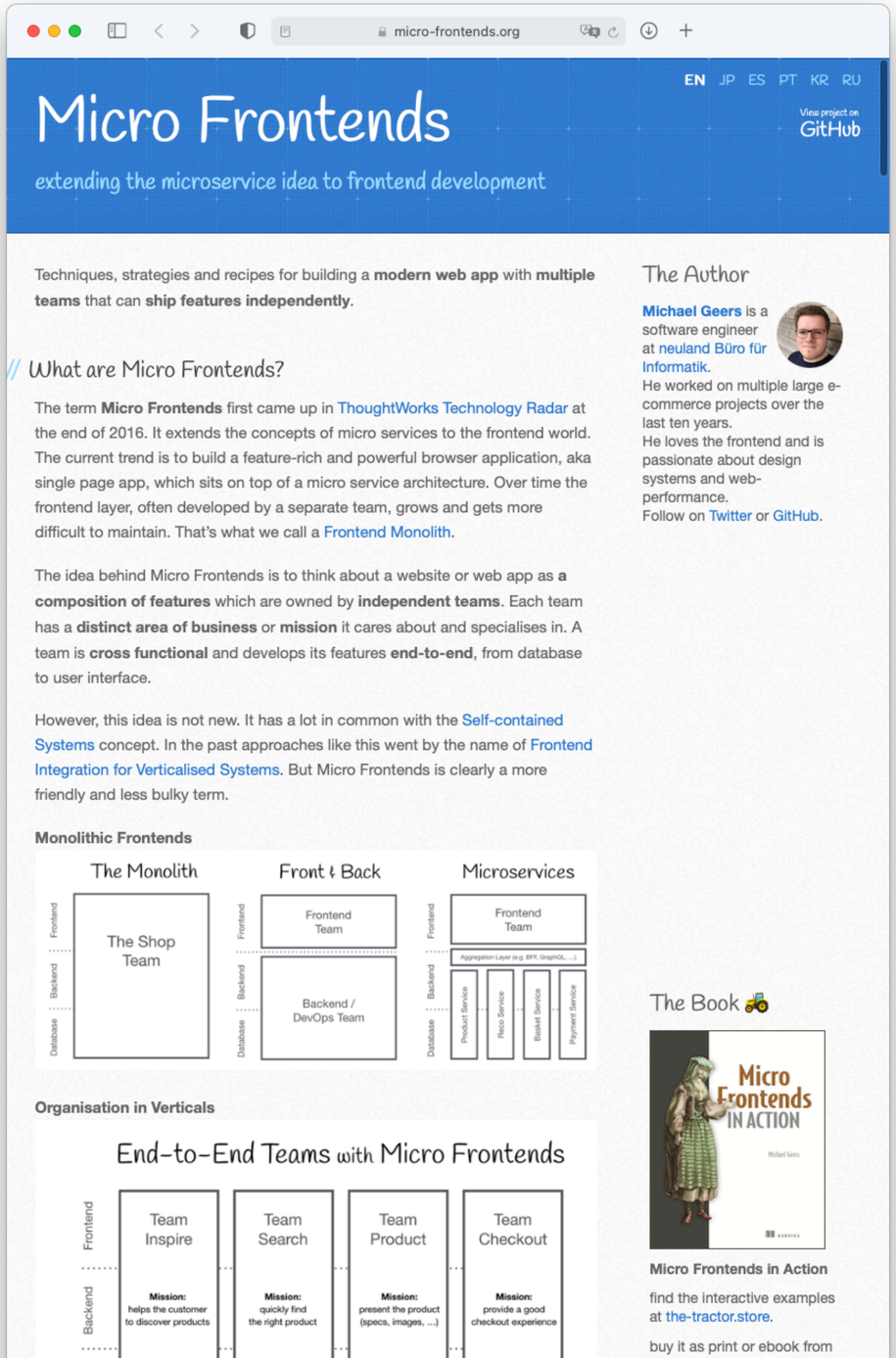
Topics: app-shell, routing, client-side, Web Components

Single SPA



try out | view code

Topics: app-shell, routing, client-side, single-spa,



need more?

micro-frontends.org

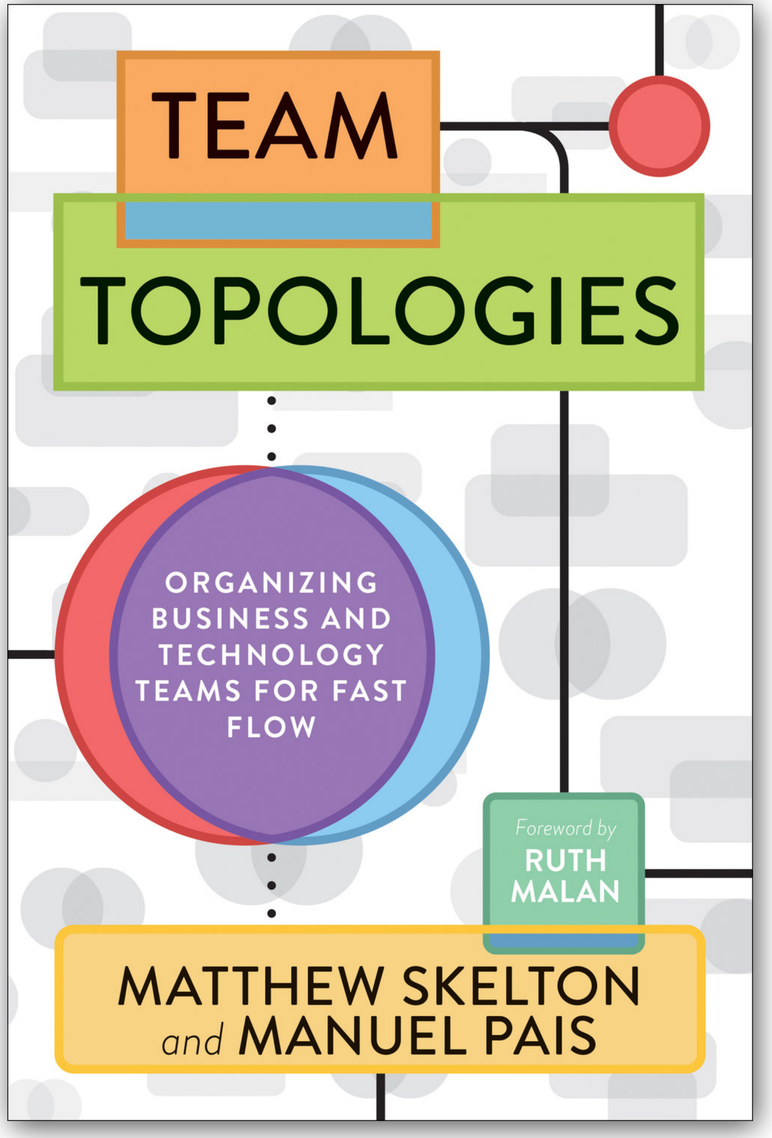
article by Cam Jackson

article by Gustaf Nilsson Kotte

blogposts by Florian Rappl

blogposts by Luca Mezzalira

👓 awesomelist by Rajasegar Chandran



book recommendation

about structuring your organisation's

teams & systems

Michael Geers / @naltatis

thank you for listening!

stay healthy

Michael Geers / @naltatis

neuland:::
Büro für Informatik

image credits

Mirror Facade Jonas Ferlin
https://www.pexels.com/photo/mirror-facade-of-tall-building-1963557/

Monolith 1 Jeff Hendricks
https://unsplash.com/photos/yIKdc86jNBs

Monolith 2 Rana Osman
https://unsplash.com/photos/JcSsu-NF3qo

Buildings Juhasz Imre
https://www.pexels.com/photo/apartment-architecture-buildings-business-425047/

Metal Blade Russ Ward
https://unsplash.com/photos/aMlbxs8SYig

Man Welding Metal Kateryna Babaieva
https://www.pexels.com/photo/man-welding-metal-2880871/

Action Bicycling Roman Pohorecki
https://www.pexels.com/photo/action-bicycling-bike-biking-287398/

Running Kids (me)

The Tool Guy Tirachard Kumtanom
https://unsplash.com/photos/UuW4psOb388

Woodshop Igor Ovsyannykov
https://unsplash.com/photos/iXV0i4Wo4yc

Tractors Manufactum
https://www.manufactum.com/tin-toys-c193667/

built with micro frontends