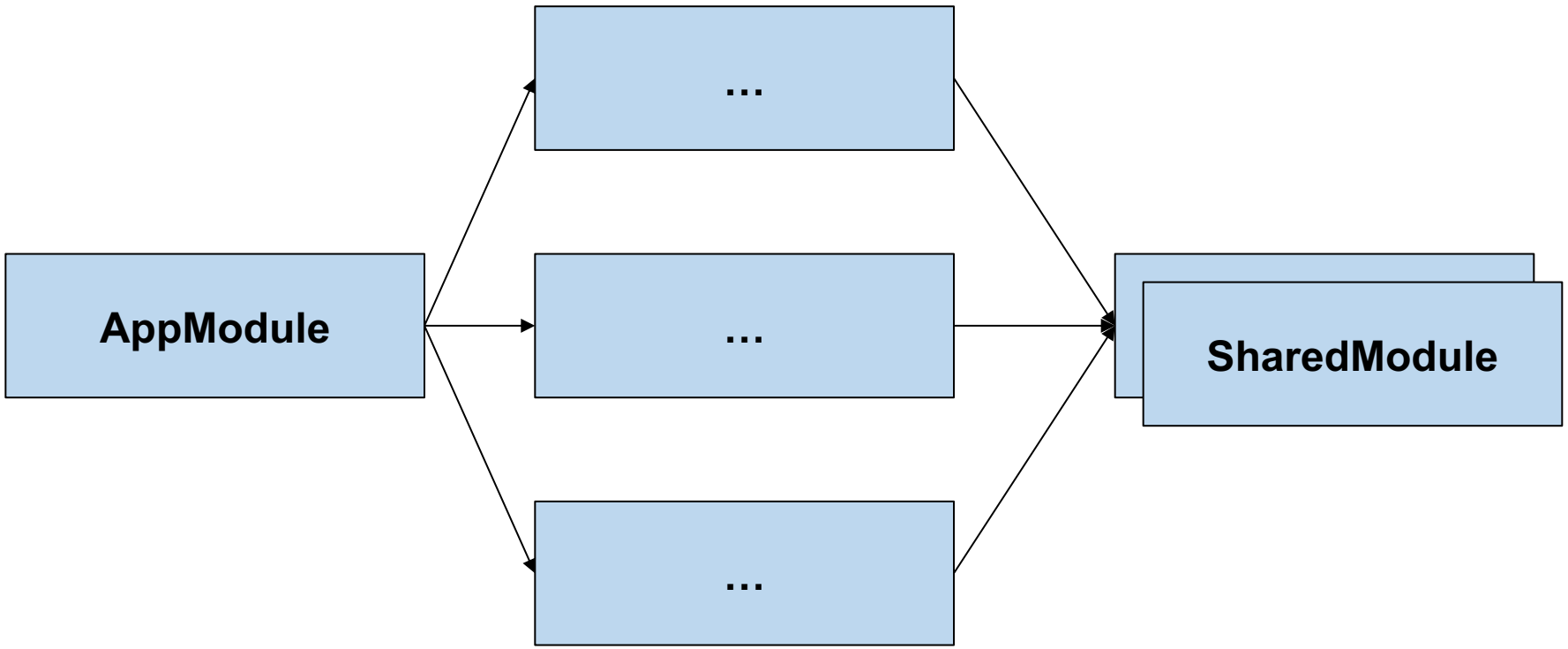
ANGULAR
ARCHITECTS
INSIDE KNOWLEDGE

Angular Architektur-Workshop: Strategic Design and Micro Frontends

ANGULARarchitects.io

Typical Module Structure



Root Module **Feature Modules** **Shared Modules**

Contents

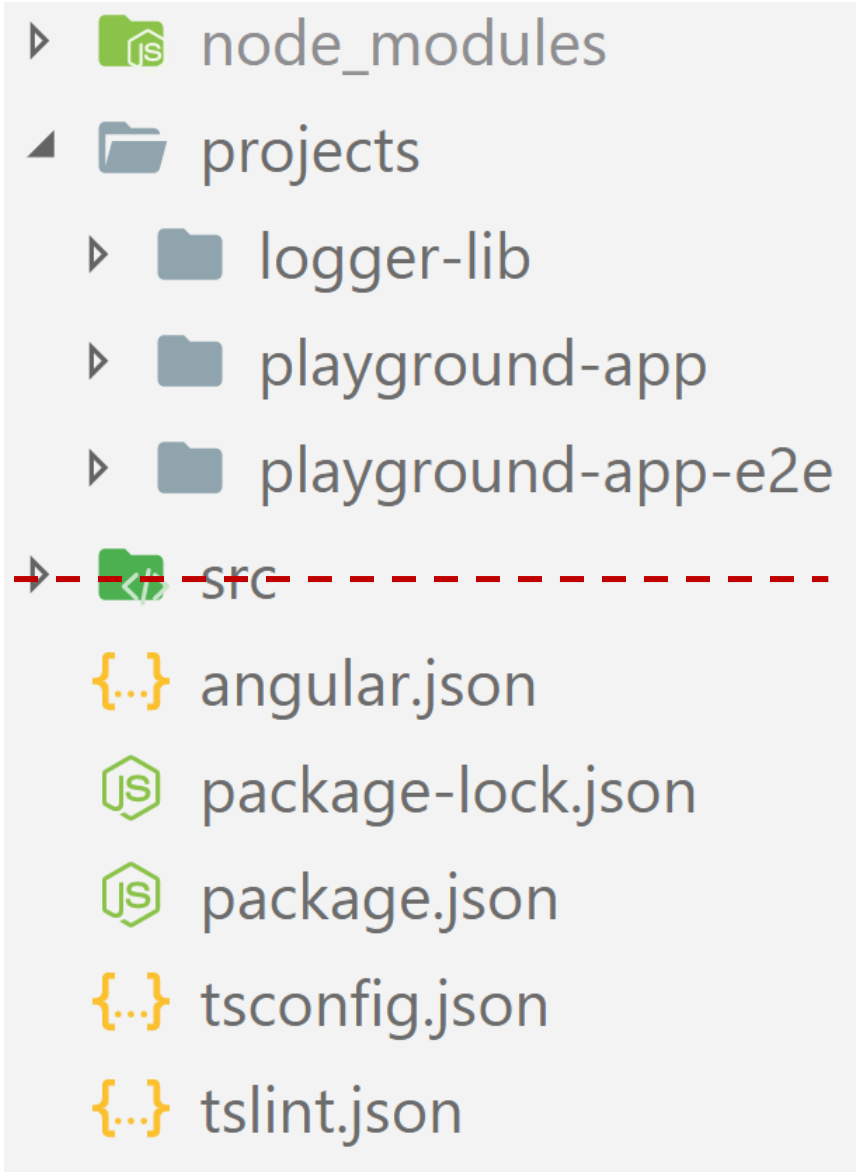
- (npm-)Packages
- Nx Monorepos
- Strategic Design and DDD
- Microfrontends

npm Packages

Create Library with CLI >= 6

```
npm install -g @angular/cli

ng new lib-project

cd lib-project

ng generate library logger-lib
ng generate application playground-app

ng serve --project playground-app
ng build --project logger-lib
```

Folder Structure



Create Library with CLI >= 6

```
npm install -g @angular/cli

ng new lib-project --create-application false

cd lib-project

ng generate library logger-lib
ng generate application playground-app

ng serve --project playground-app
ng build --project logger-lib
```


Leerungszeiten
Deutsche Post
Tageesleerungen
Spätleerung
Nachtleerung
Montag - Freitag
08:00
Samstag
08:00
Sonntag
Sendungen aus allen Tages- und Spätleerungen erreichen die Empfänger bundesweit in der Regel mit der nächsten Zustellung.
Bei Nachtleerungen gilt dies nur für Sendungen, deren Postleitzahl mit den Ziffern 65 beginnt.
Service-Telefon der Briefkastenleerung:
0228 4333112
Briefkästen mit späteren Leerungen:
letzte Tagesleerung
Spätleerung
Nachtleerung
Montag - Freitag
Samstag
Sonntag

Publishing

Publishing to npm Registry

- Increment version in package.json
- ng build logger-lib --prod
- npm publish *dist/logger-lib* --registry http://localhost:4873
- npm install logger-lib --registry http://localhost:4873

Alternatives for setting the Registry

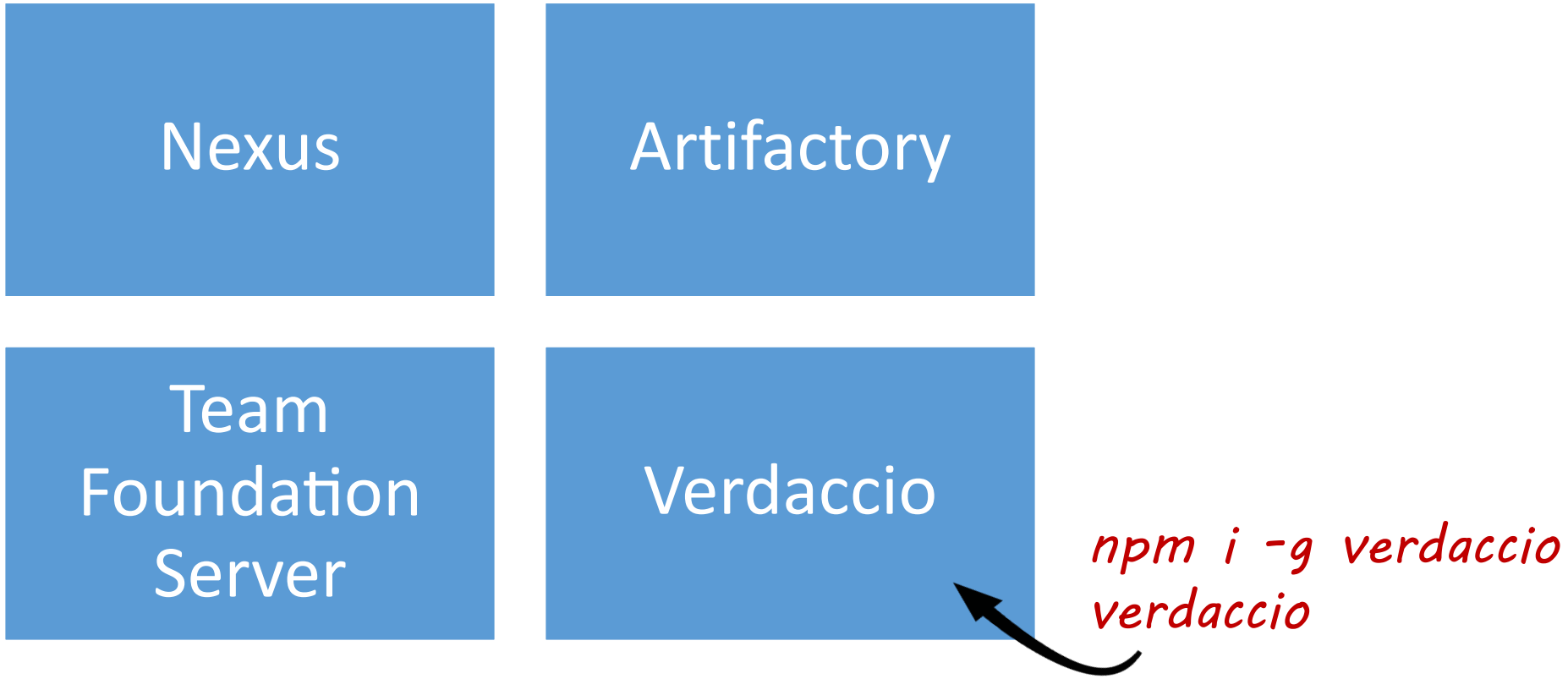
- Global: npm set registry http://localhost:4873
  - Default: registry.npmjs.org
  - npm get registry

- Project: .npmrc in project root

```
registry=http://localhost:4873/
```

```
@cmp:registry=http://my-server:4871/
@cmp-ui:registry=http://my-server:4872/
```

npm Registries



DEMO

Advantages

- Distribution
- Versioning

Disadvantages

- Distribution
- Versioning

;-)

Disadvantages

Distribution

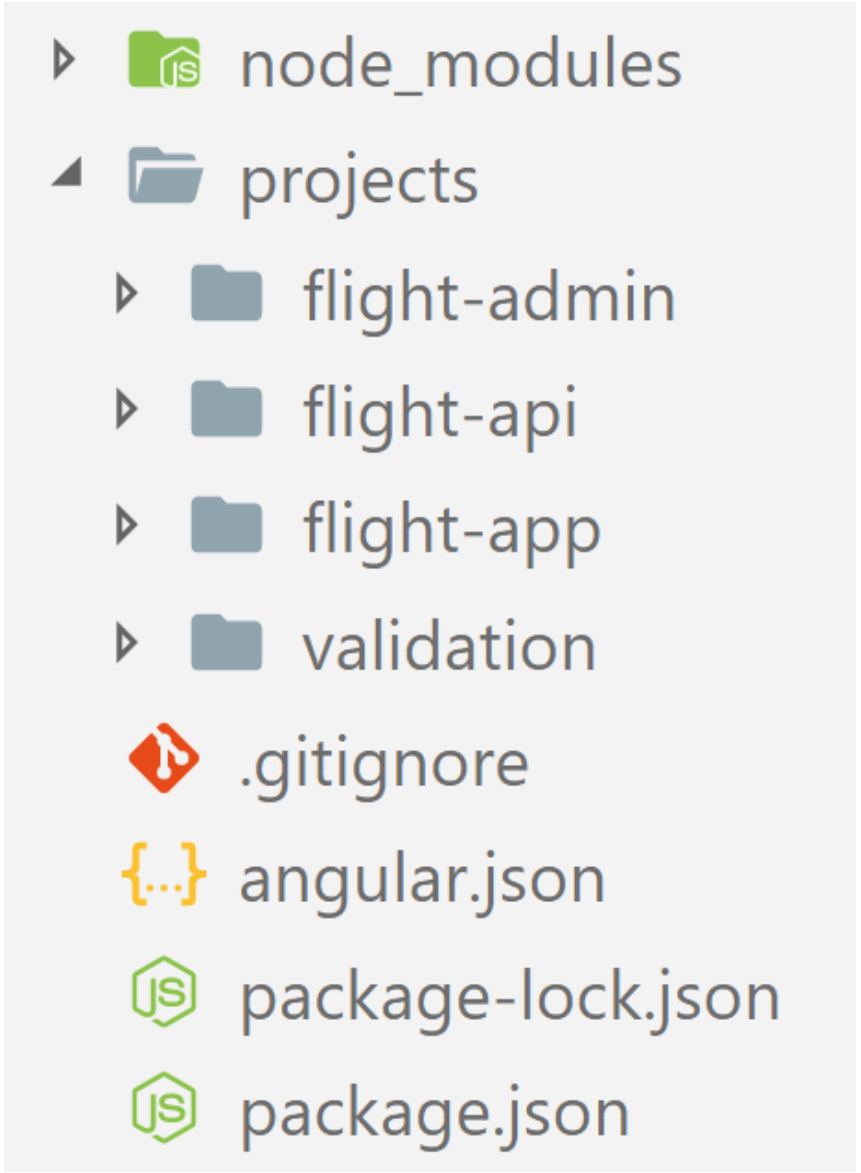
- Annoying within project
- Prevents gritting further libs

Versioning

- Old versions
- Conflicts
- How to force devs to use latest version?

Monorepos

Monorepo Structure



Advantages

- Everyone uses the latest versions
- No version conflicts
- No burden with distributing libs

Two Flavors

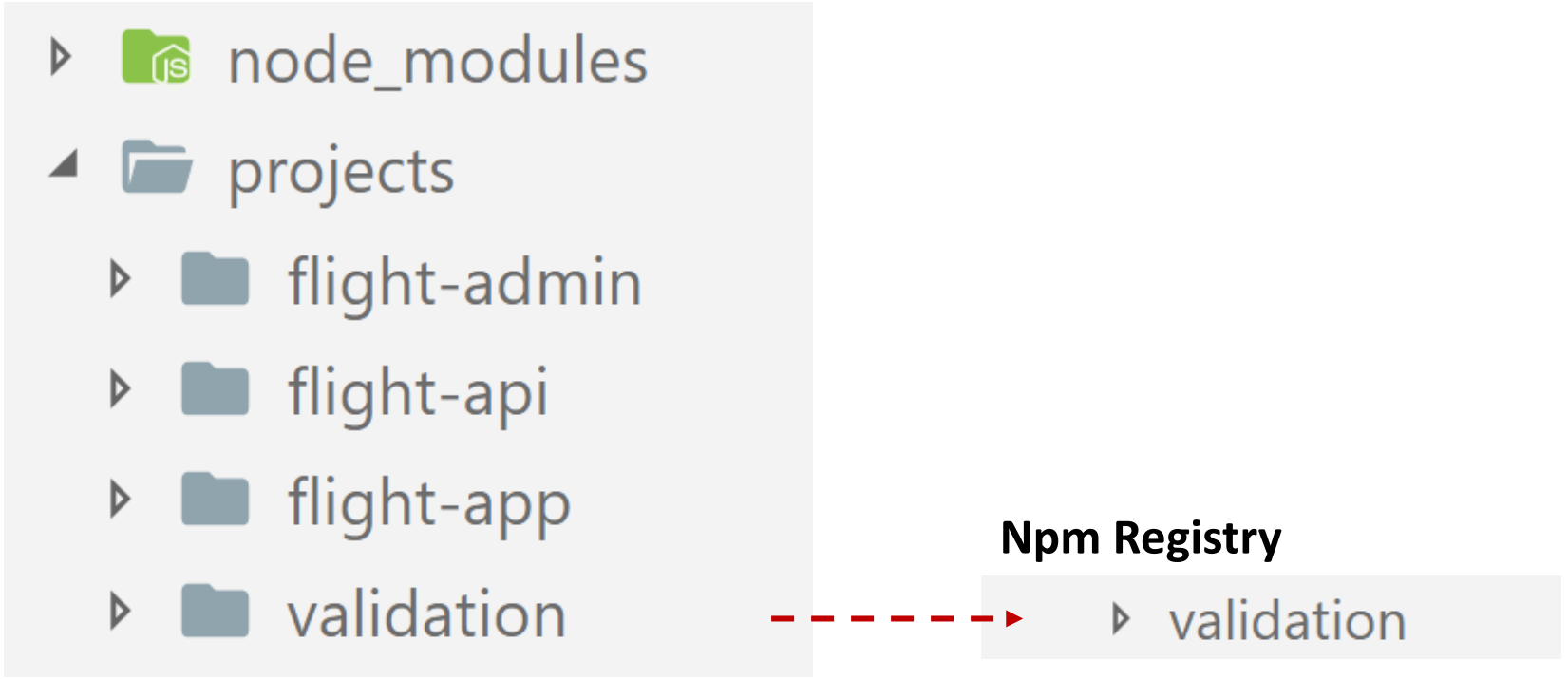
Project Monorepo

- Like Workspaces/Solutions in different IDEs

Company-wide Monorepo

- E. g. used at Google or Facebook

Moving back and forth



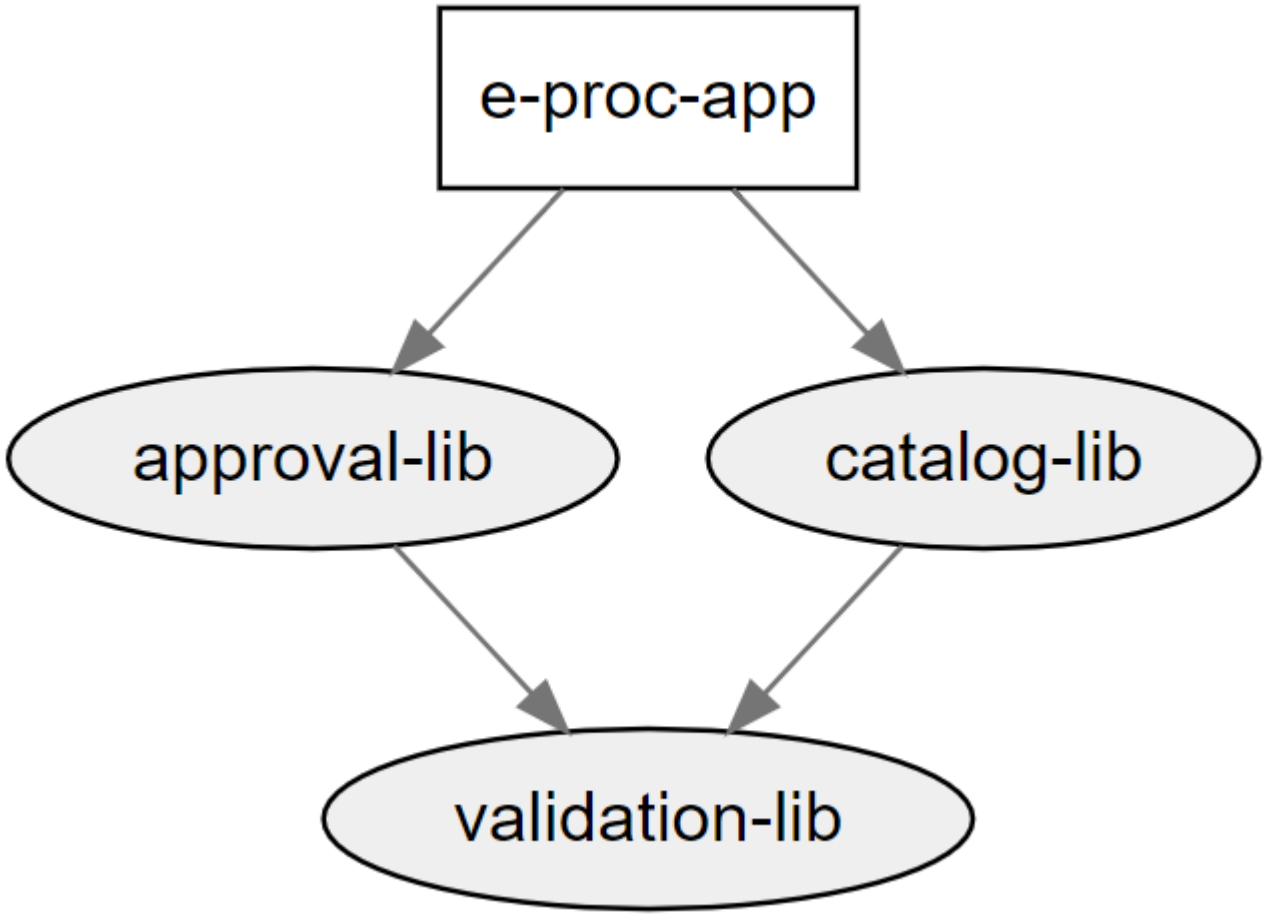
Tooling & Generator

https://nrwl.io/nx

Nrwl Extensions for Angular

An open source toolkit for enterprise Angular applications.

Visualize Module Structure



Further Features of Nx

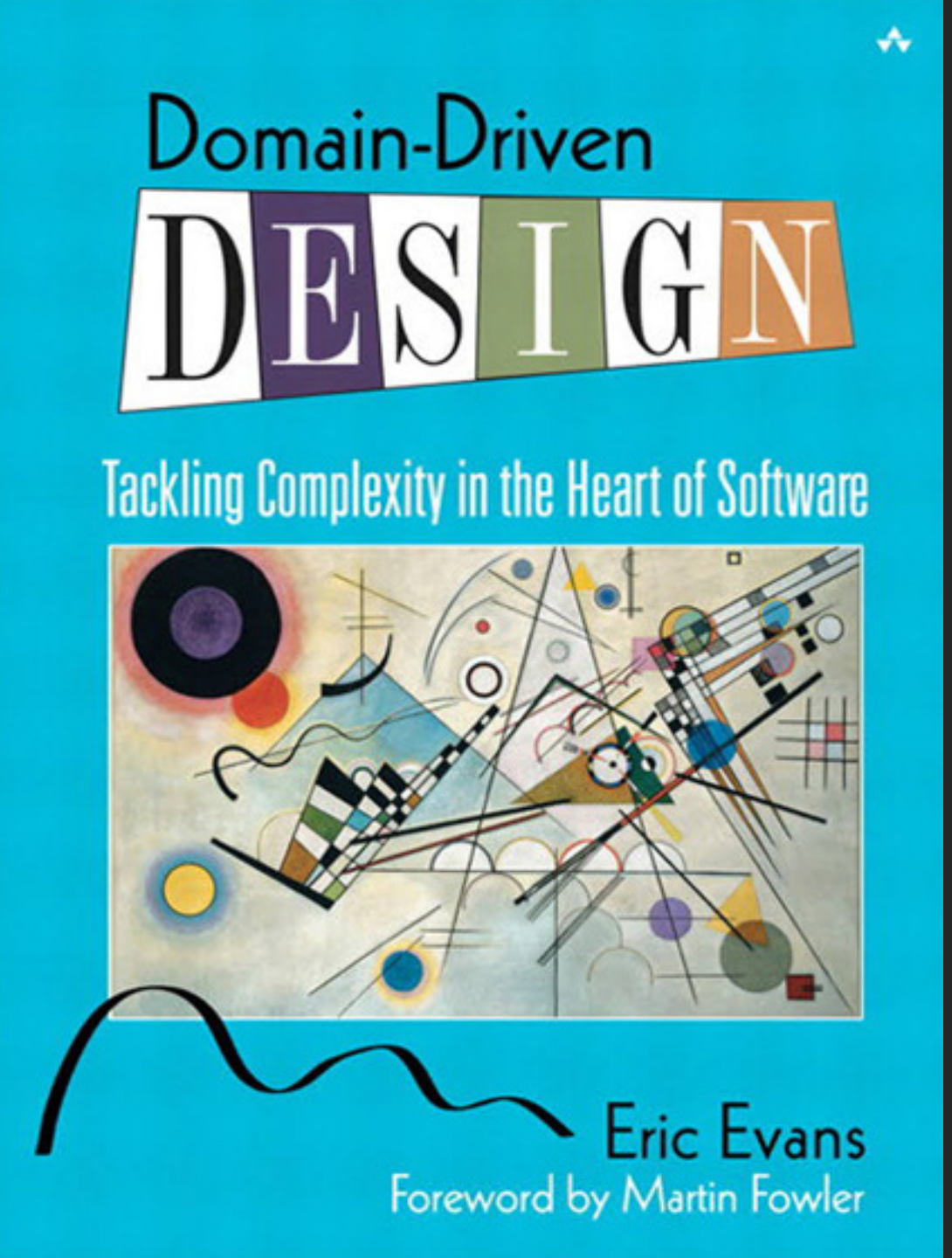
- Define folders for libraries
- Restrict which apps/libs can access which other libs
- Just recompile changed apps
- Visualize module structure and dependencies
- Scaffold Boilerplate

Creating a Workspace

```
npm install -g @angular/cli

npm init nx-workspace workspace

cd workspace

nx generate app my-app
nx generate lib my-lib --buildable

nx serve --project my-app
nx build --project my-app
```

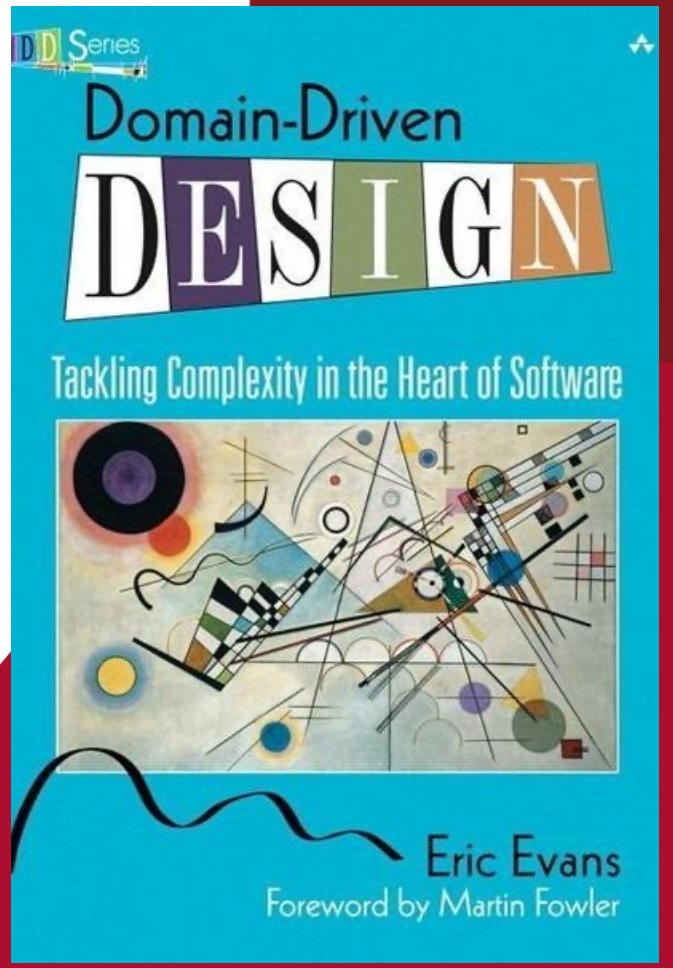
DEMO

LAB



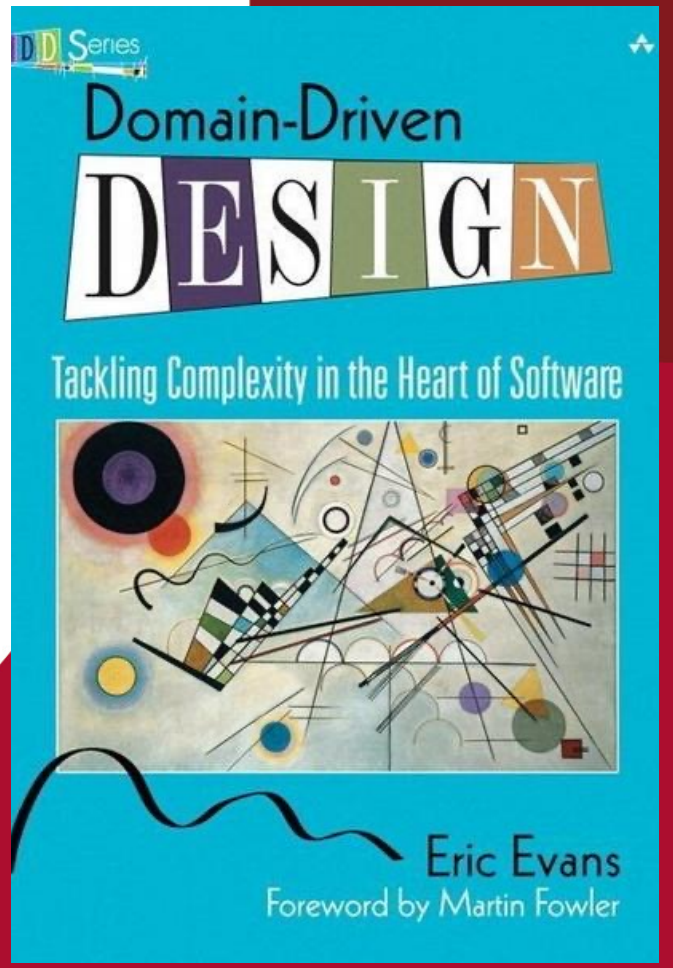
Methodology for bridging the gap b/w requirements and architecture/ design

How to create sustainable frontend architectures with ideas from DDD?



ANGULAR
ARCHITECTS
INSIDE KNOWLEDGE

How to create **sustainable** frontend architectures with **ideas from DDD**?



ANGULAR ARCHITECTS
INSIDE KNOWLEDGE

Domain Driven Design

Decomposing a System

Design Patterns
& Practices

Strategic Design

Tactical Design

Domain Driven Design

Decomposing a System

Design Patterns
& Practices

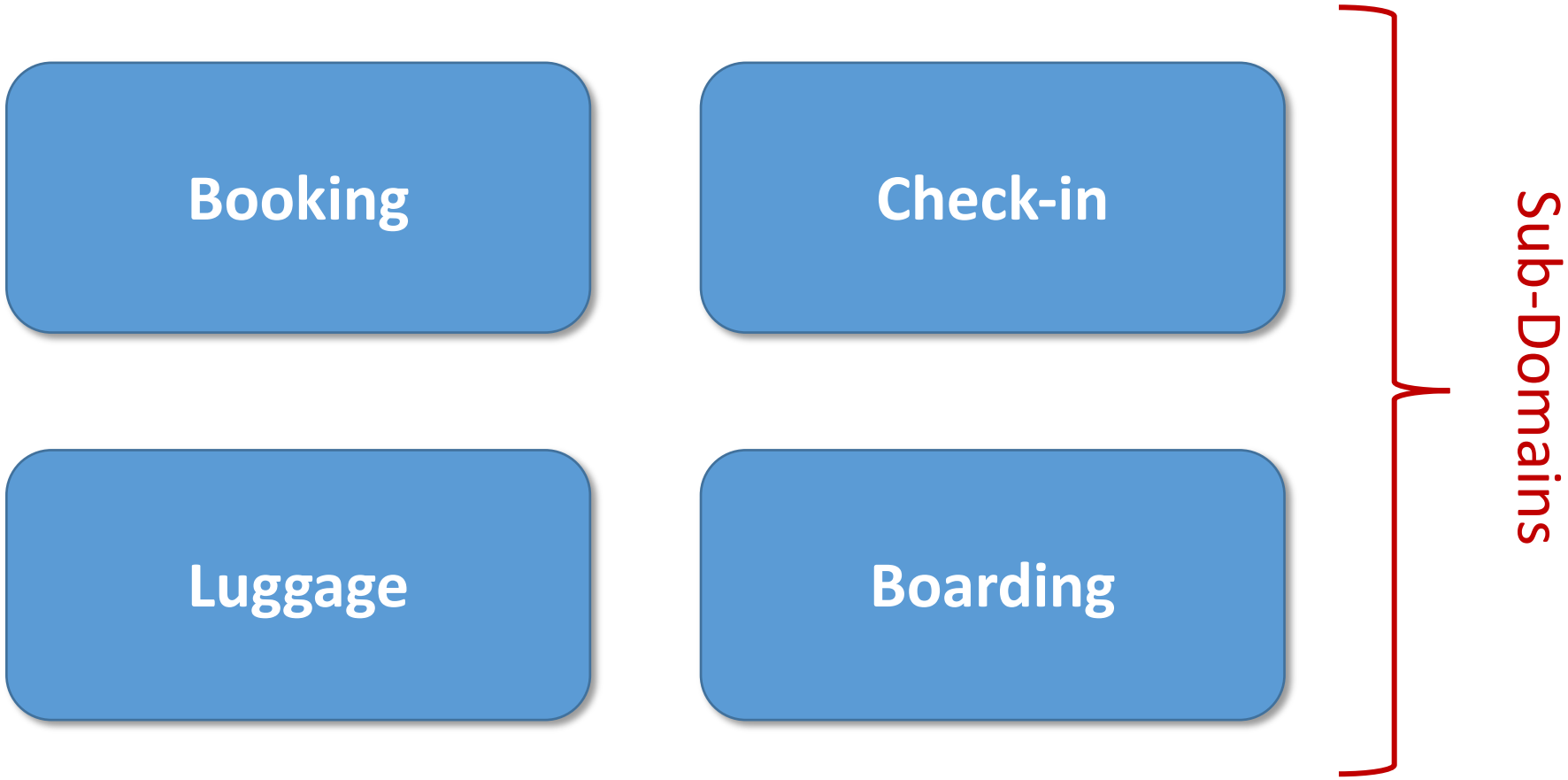
Strategic Design

Tactical Design

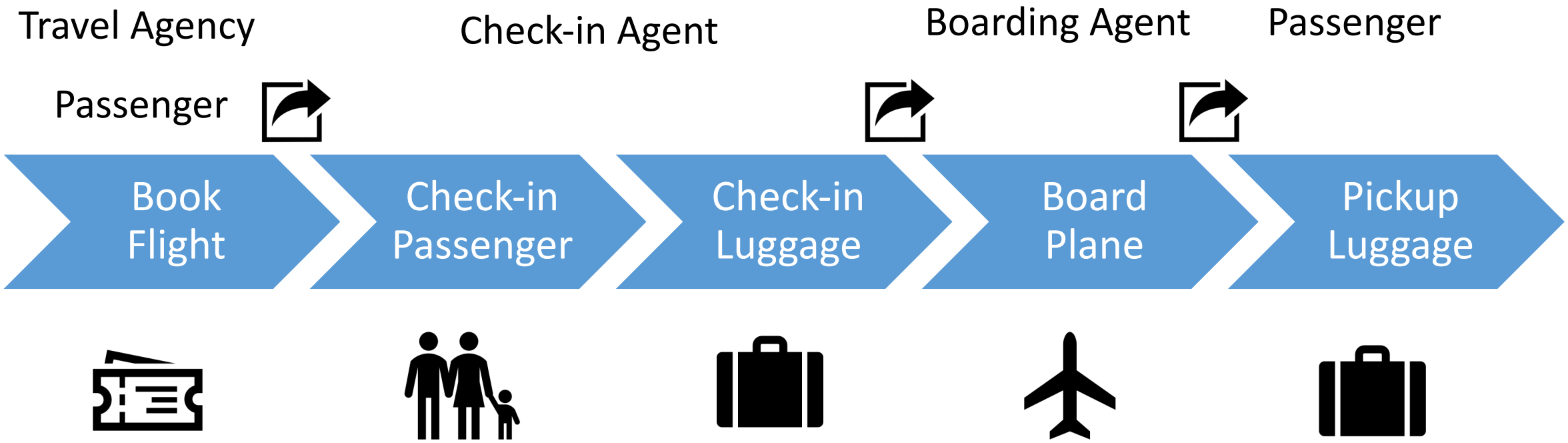
This is what Strategic DDD prevents

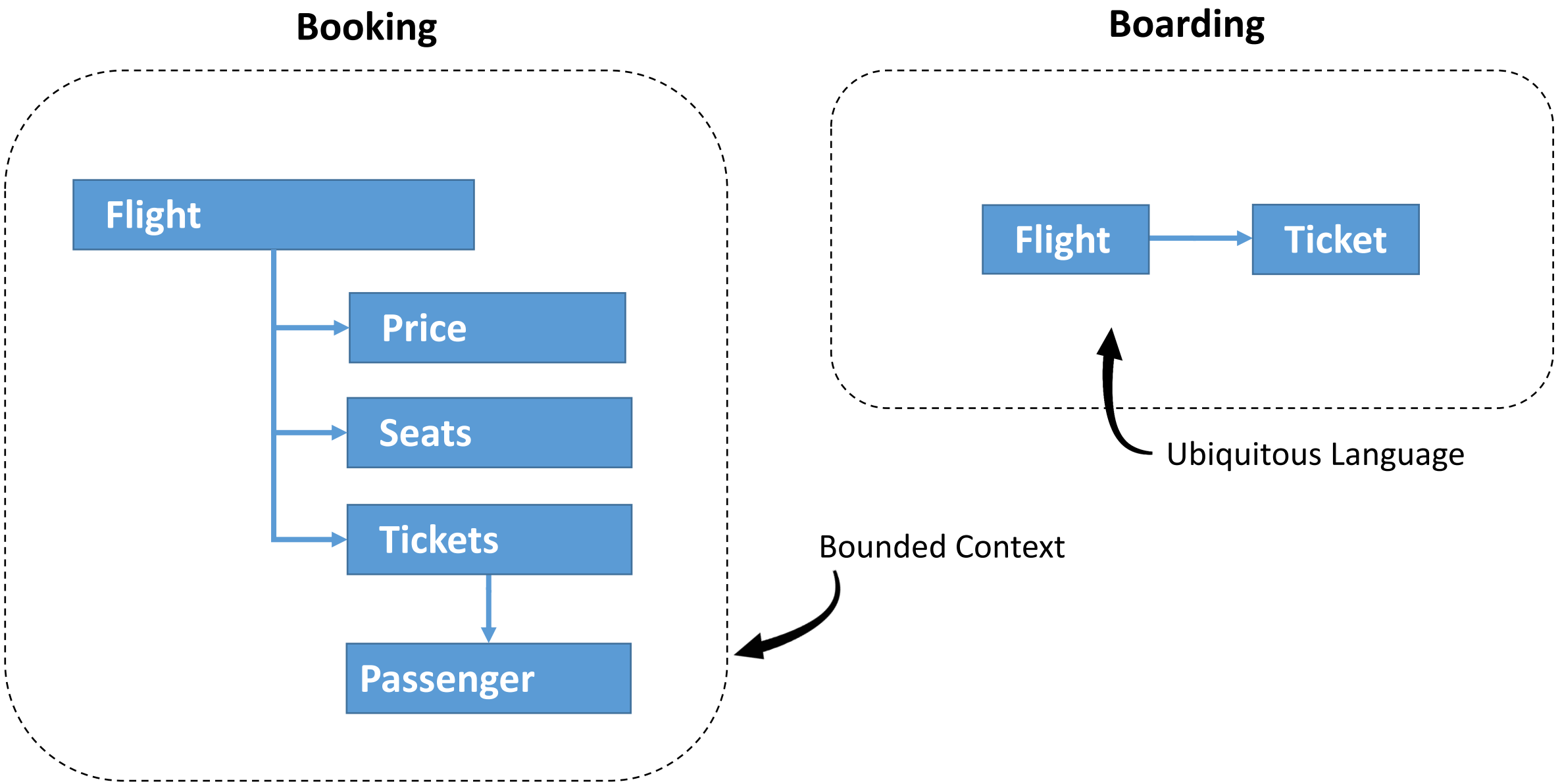
Example

Example

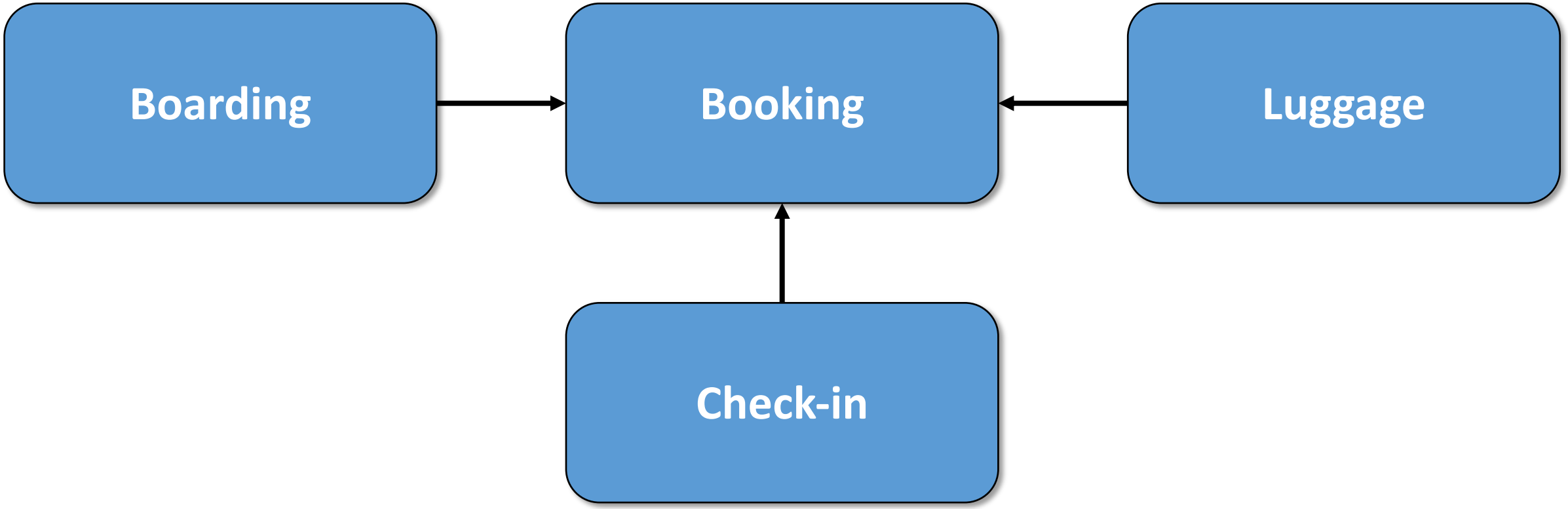


Finding Sub-Domains

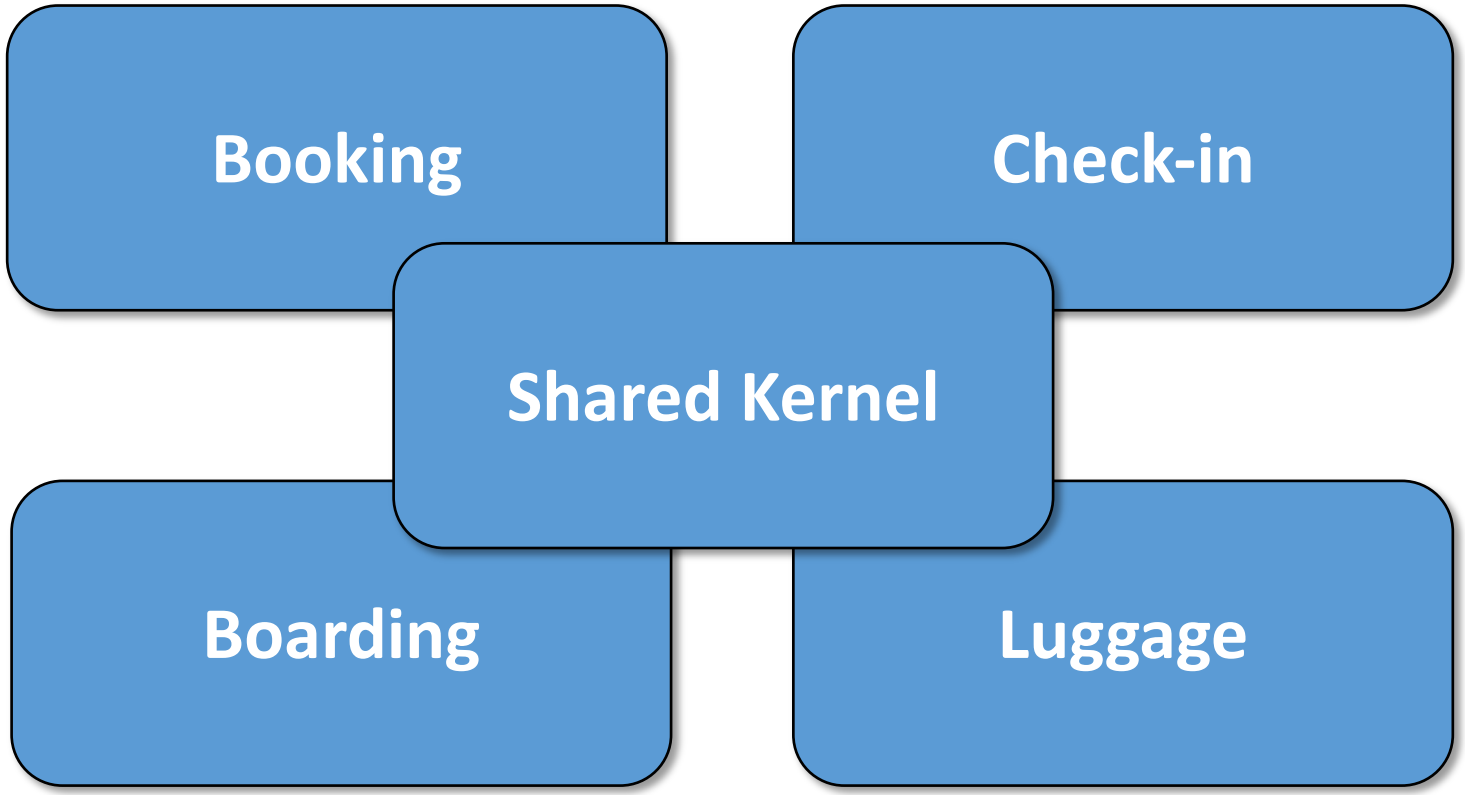


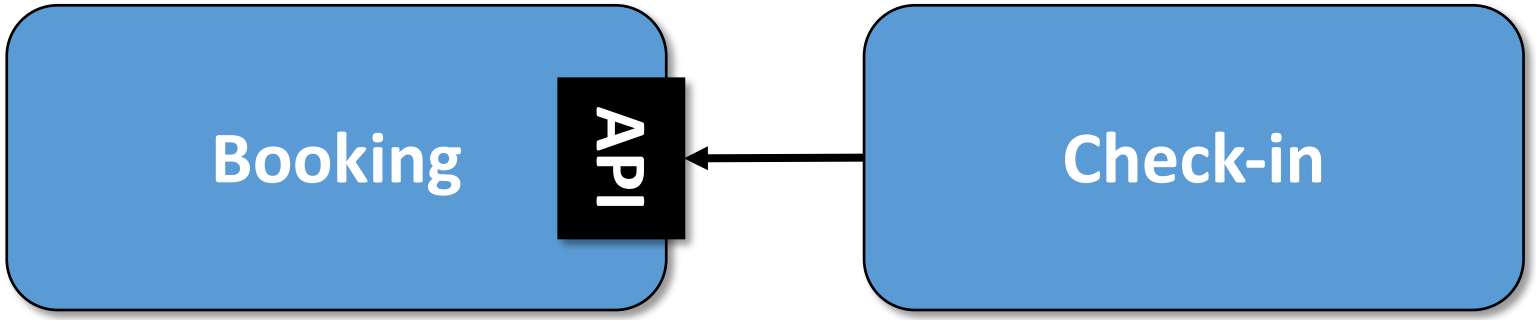

Booking
Boarding
Flight
Price
Seats
Tickets
Passenger
Flight
Ticket
Ubiquitous Language
Bounded Context

Context Map

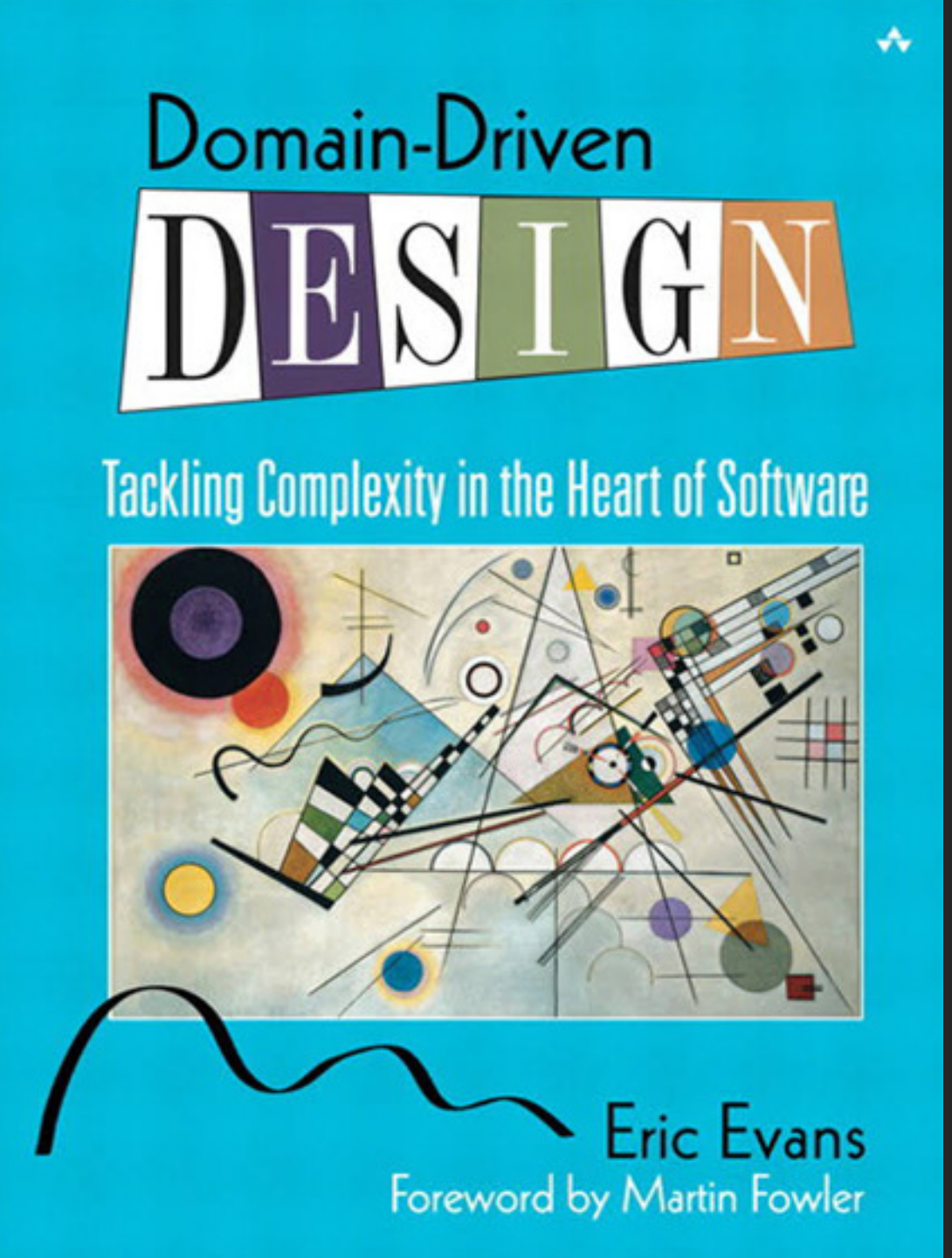


Context Map

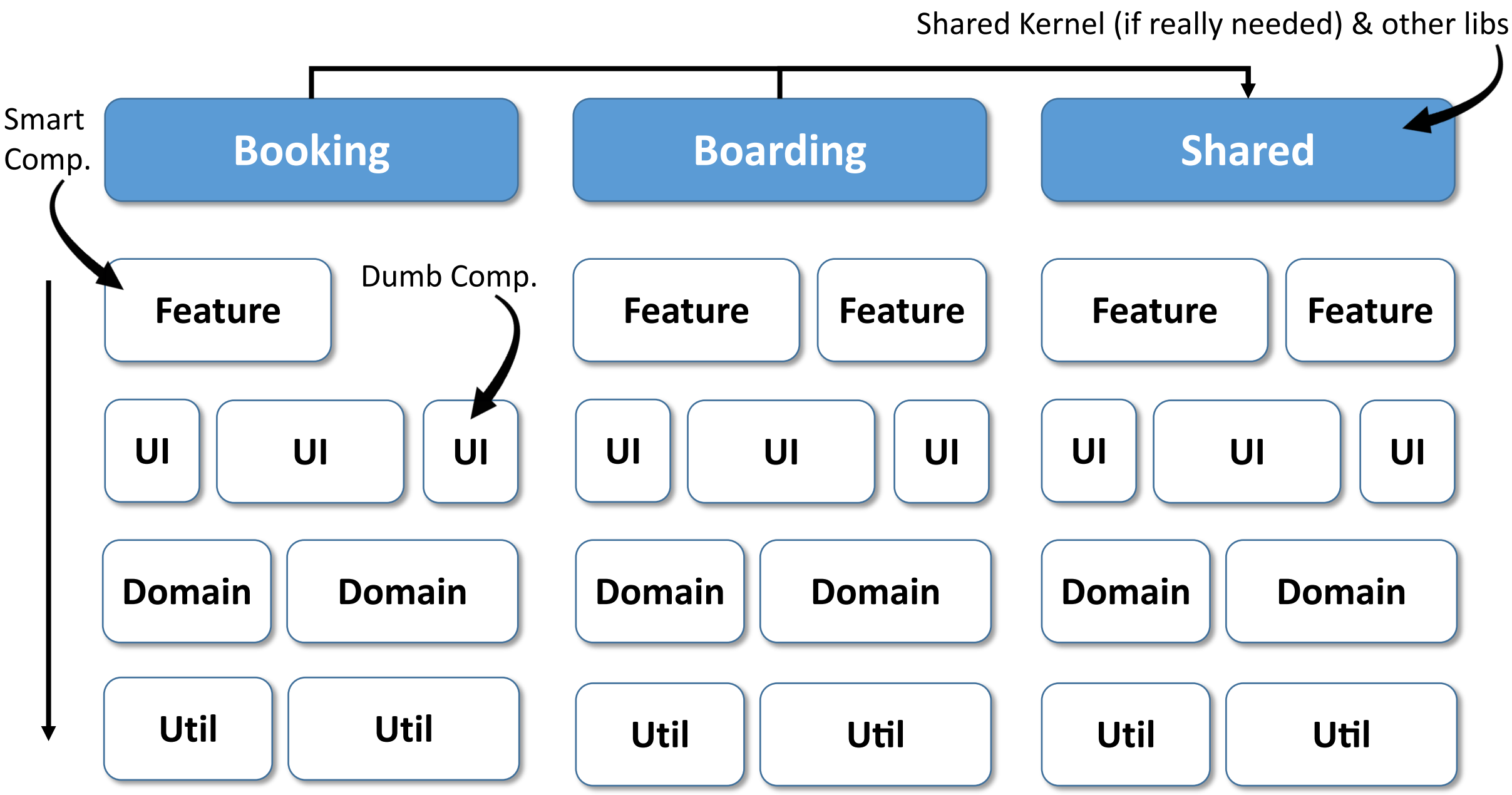




Open-/Host-Service

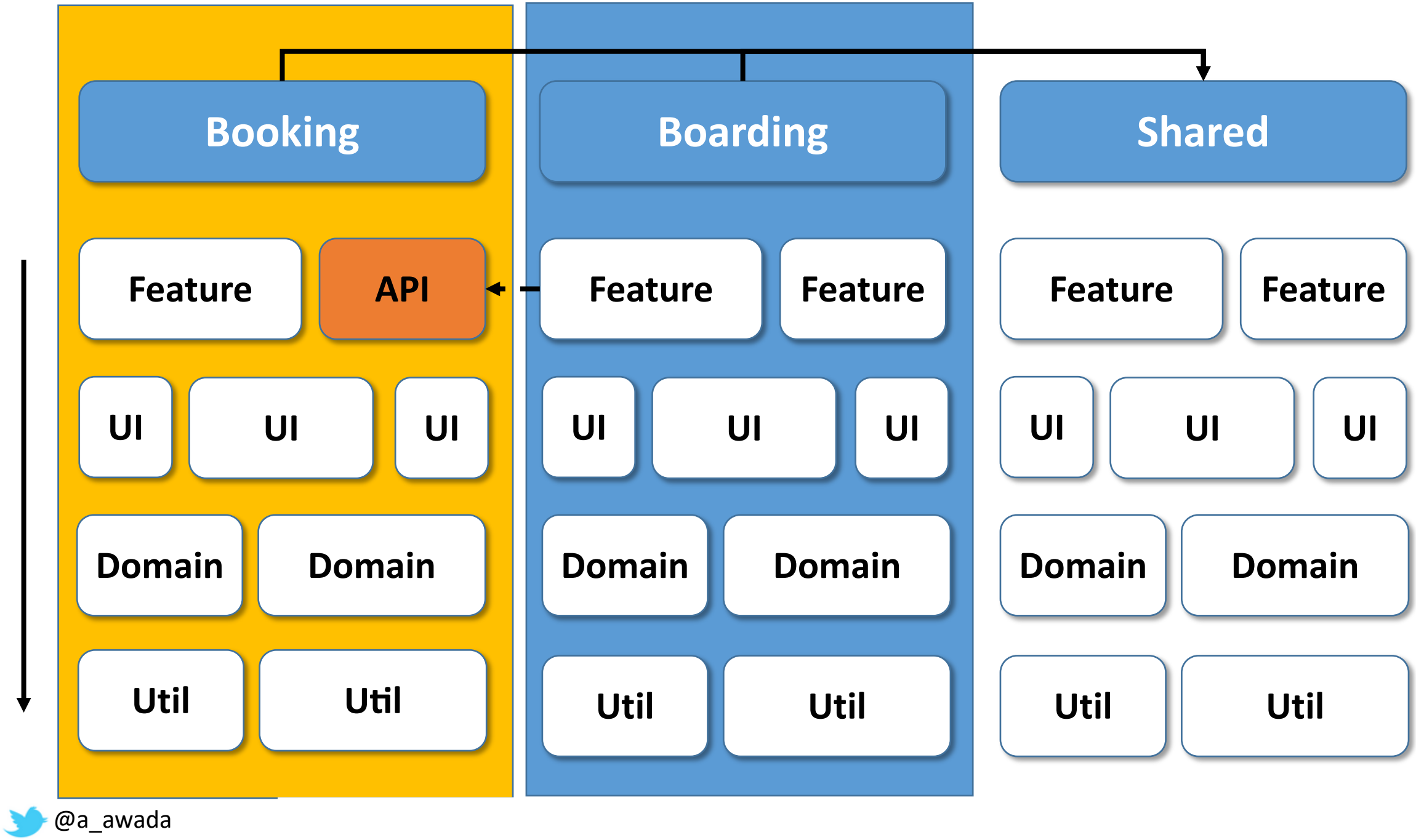


Lots of approaches for cross-domain communication and more …



@a_awada

Enterprise Monorepo Patterns, Nrwl 2018: https://tinyurl.com/y2jjxld7


Booking
Boarding
Shared
Feature
API
Feature
Feature
Feature
Feature
UI
UI
UI
UI
UI
UI
UI
UI
UI
Domain
Domain
Domain
Domain
Domain
Domain
Util
Util
Util
Util
Util
Util

@a_awada

Alternatives to layering

- e. g. Hexagonal Architecture, Clean Architecture
- Anyway: We need to **restrict access** b/w libraries

DEMO

LAB

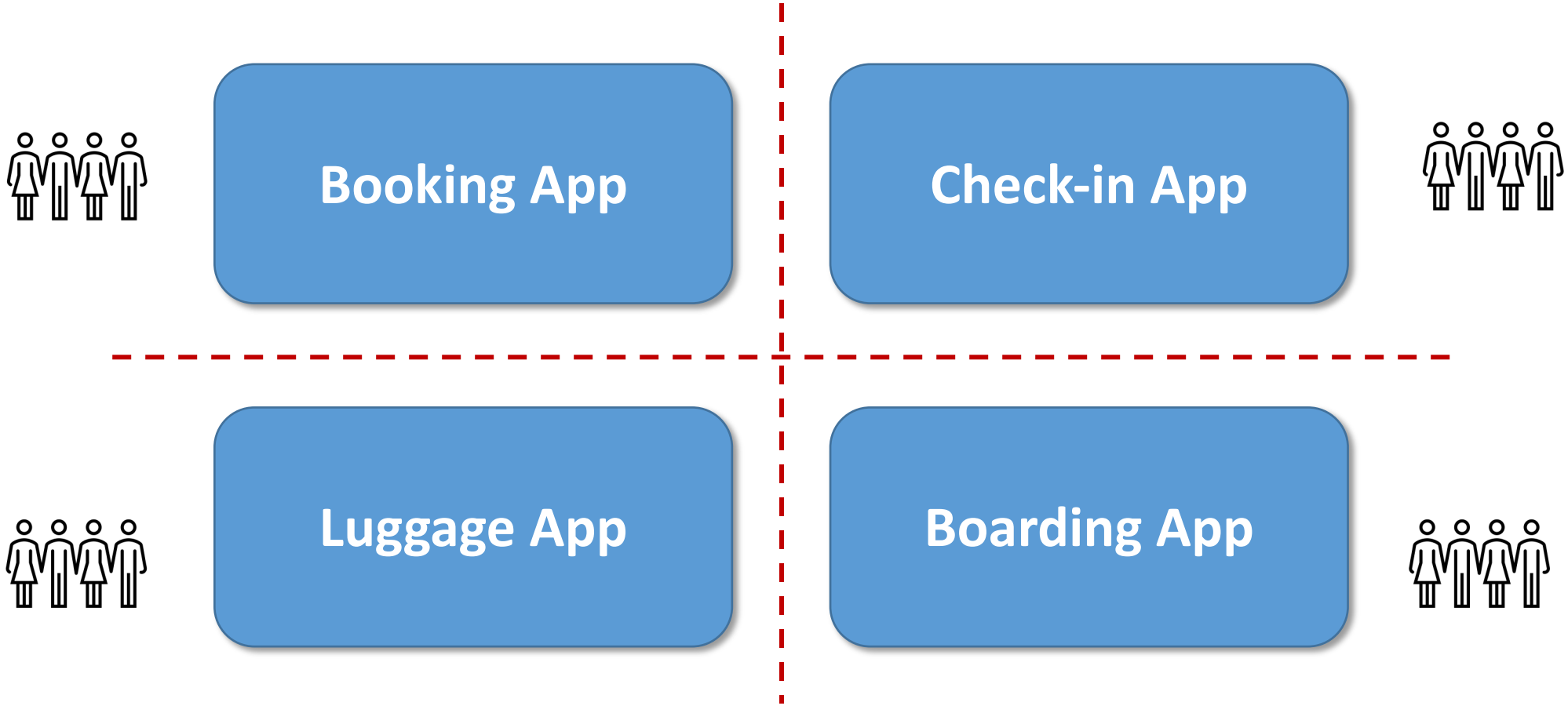
Finegrained Libraries

- Unit of recompilation
- Unit of retesting
- Access restrictions
- Information Hiding
- Easy: Just *ng g lib …*
- Future replacement for NgModules?

Micro Frontends?

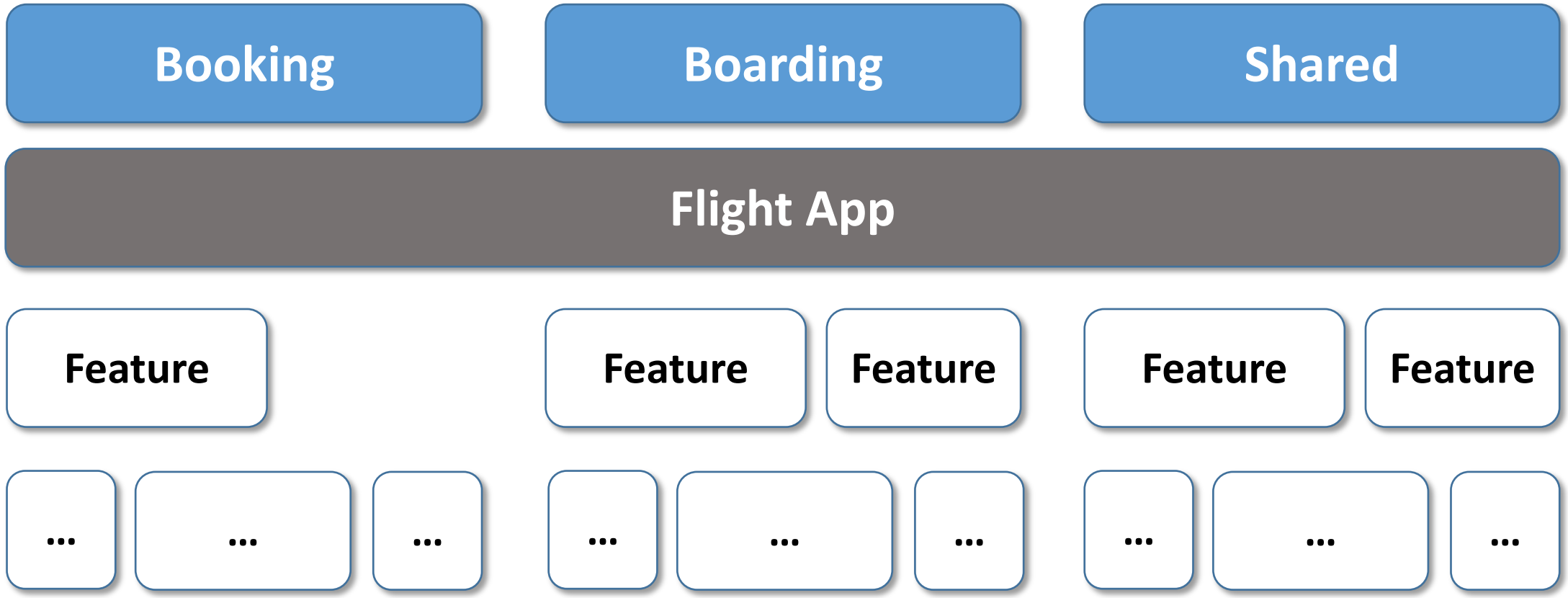
Short outlook

Microfrontends

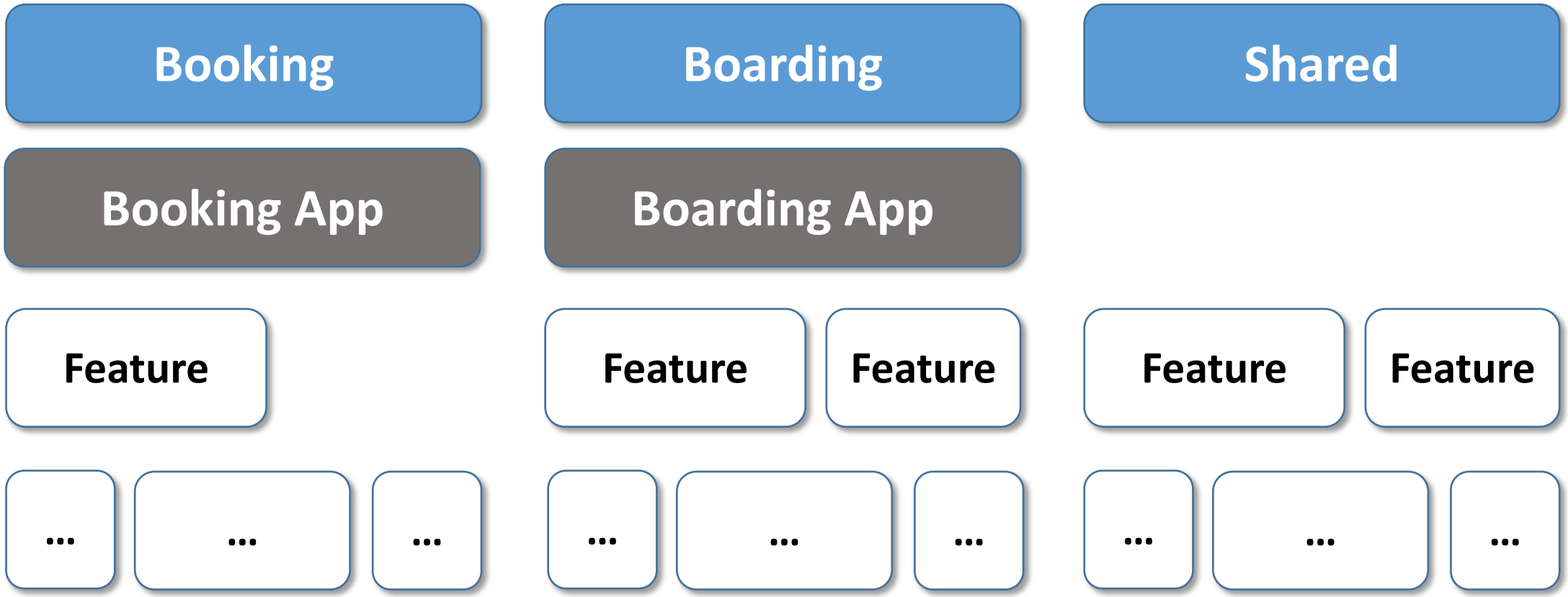


Microfrontends
are first and foremost
about **scaling teams!**

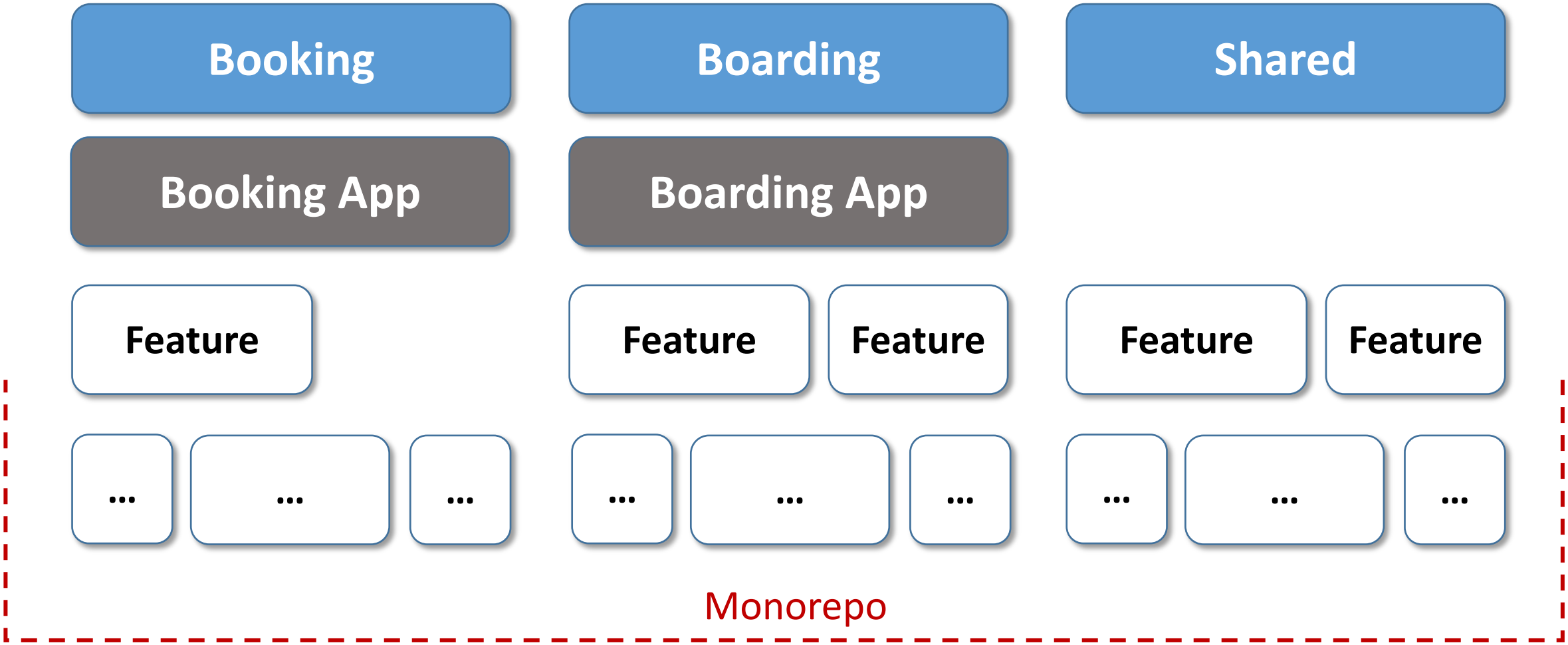
Deployment Monolith



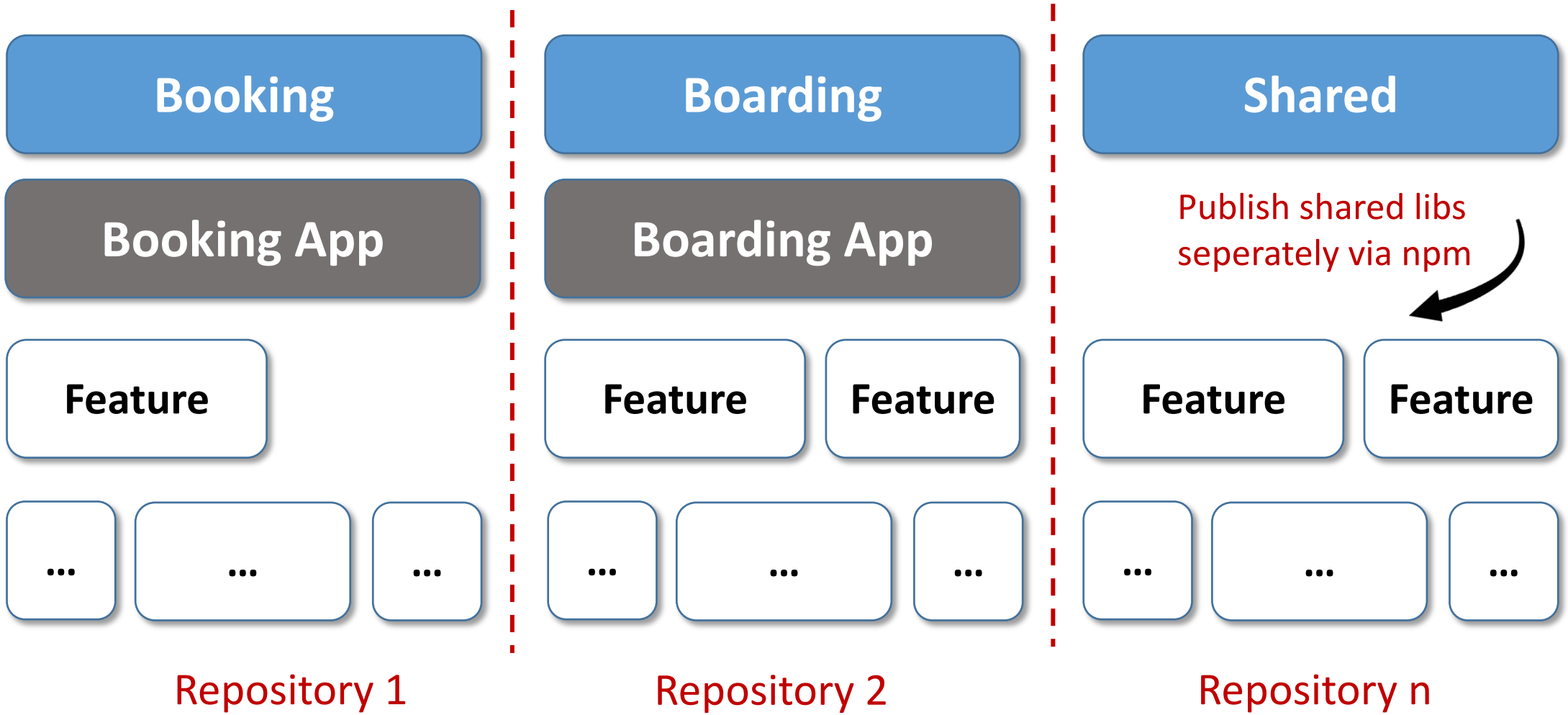
Microfrontends



Option 1: One App per Domain



Option 2: One Monorepo per Domain

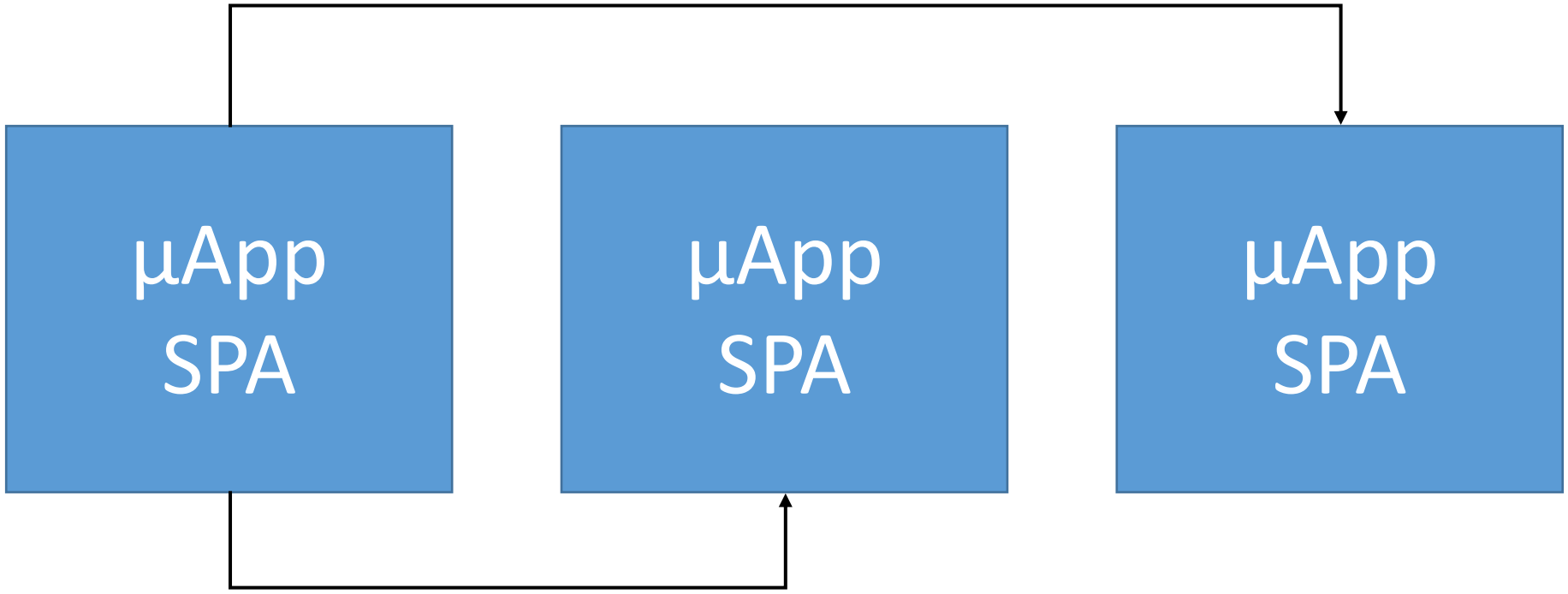


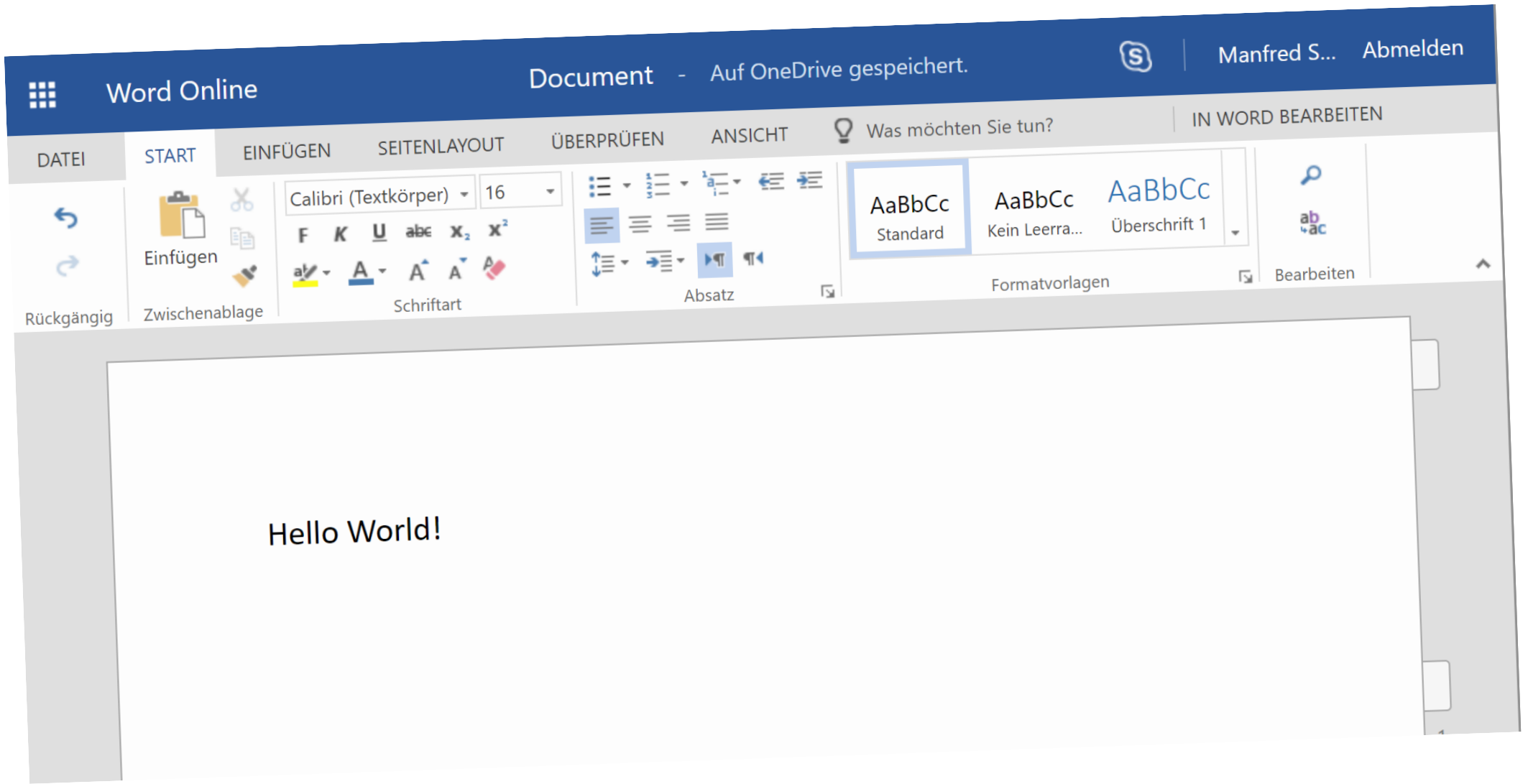
Benefits

- **Autonomous Teams**
- Separate Development
- Separate Deployment
- Own architecture decisions
- Own technology decisions

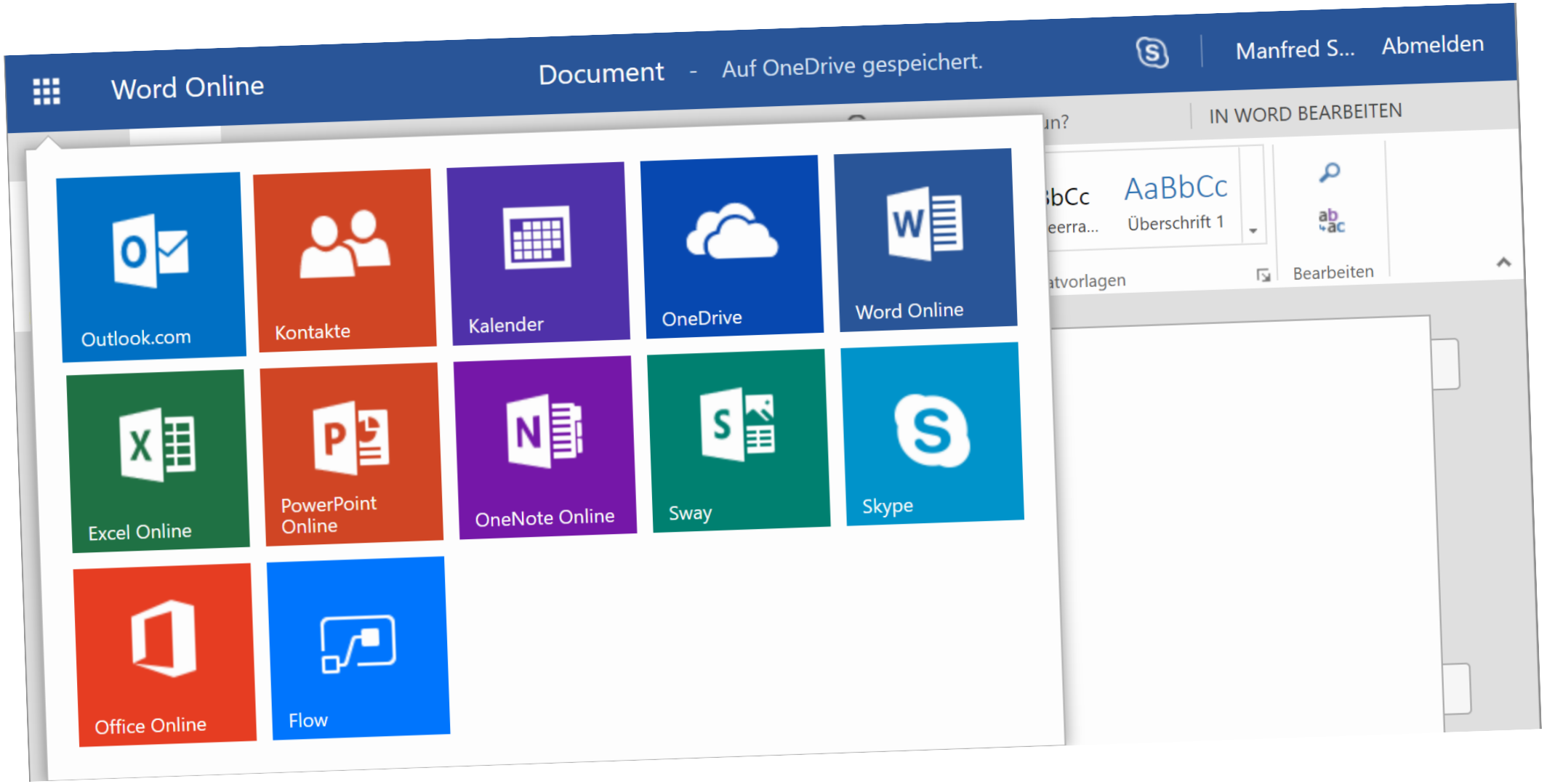
Integration via Hyperlinks

UI Composition
w/ Hyperlinks




Word Online
Document - Auf OneDrive gespeichert.
Manfred S...
Abmelden
DATEI
START
EINFÜGEN
SEITENLAYOUT
ÜBERPRÜFEN
ANSICHT
Was möchten Sie tun?
IN WORD BEARBEITEN
Einfügen
Calibri (Textkörper)
16
AaBbCc
Standard
AaBbCc
Kein Leerra...
AaBbCc
Überschrift 1
Rückgängig
Zwischenablage
Schriftart
Absatz
Formatvorlagen
Bearbeiten
Hello World!

@a_awada

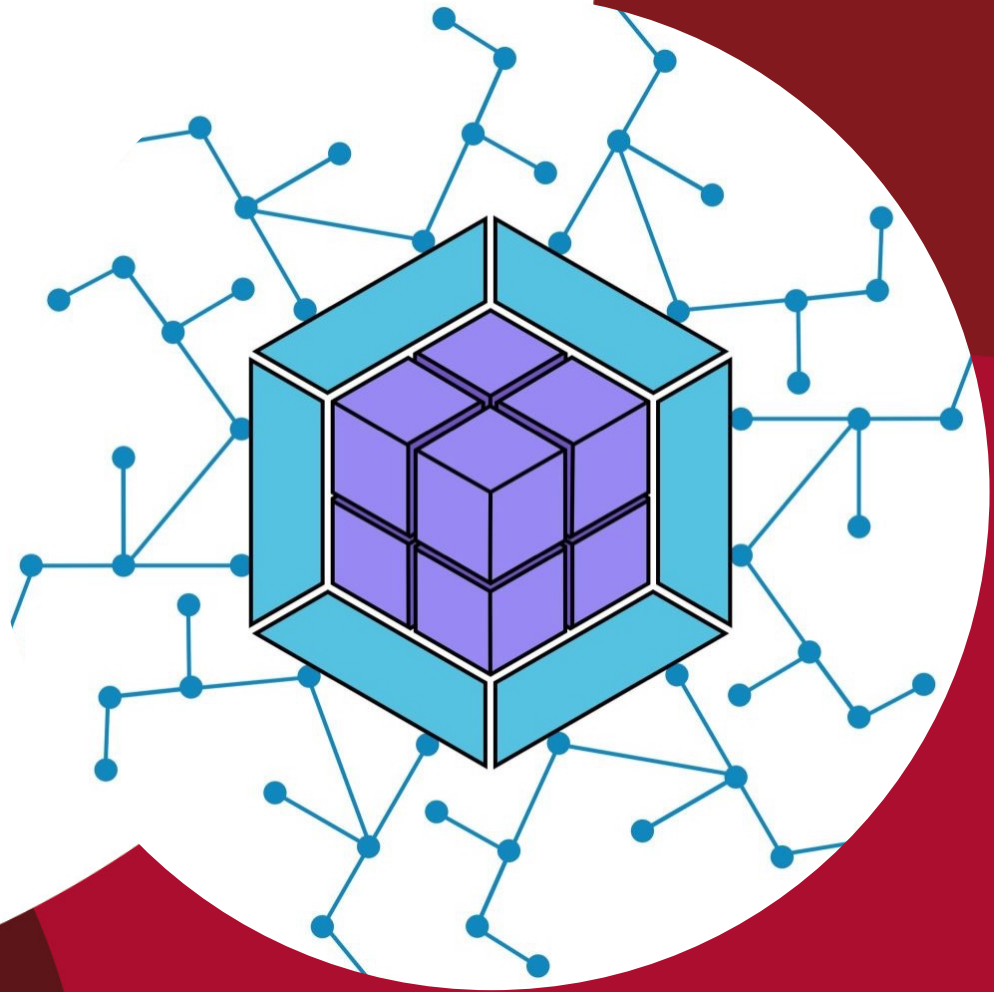

Word Online
Document - Auf OneDrive gespeichert.
Manfred S...
Abmelden
IN WORD BEARBEITEN
AaBbCc
Überschrift 1
Bearbeiten
Outlook.com
Kontakte
Kalender
OneDrive
Word Online
Excel Online
PowerPoint Online
OneNote Online
Sway
Skype
Office Online
Flow

@a_awada


Integration via
Shell

Providing a (SPA based) Shell

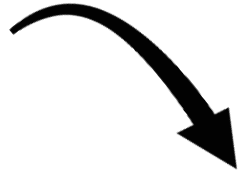
Webpack 5
Module Federation



ANGULAR
ARCHITECTS
INSIDE KNOWLEDGE

Idea

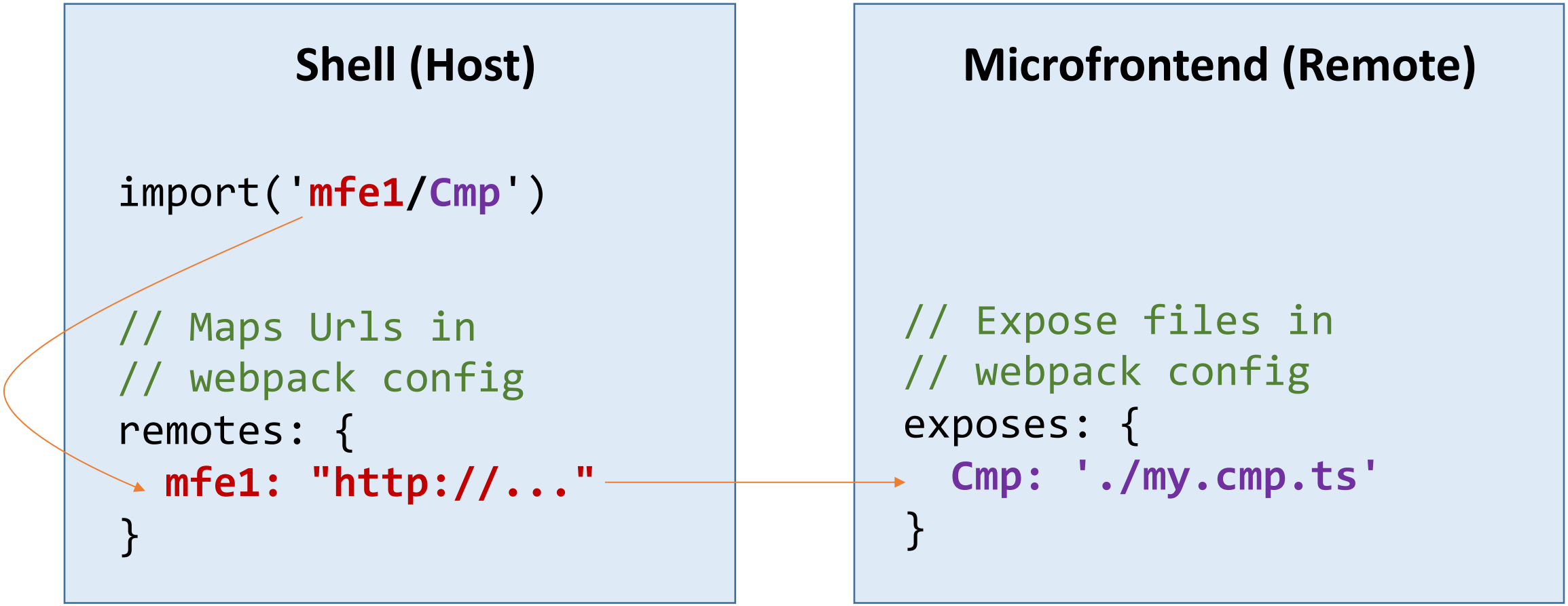
Does not work with
webpack/ Angular CLI



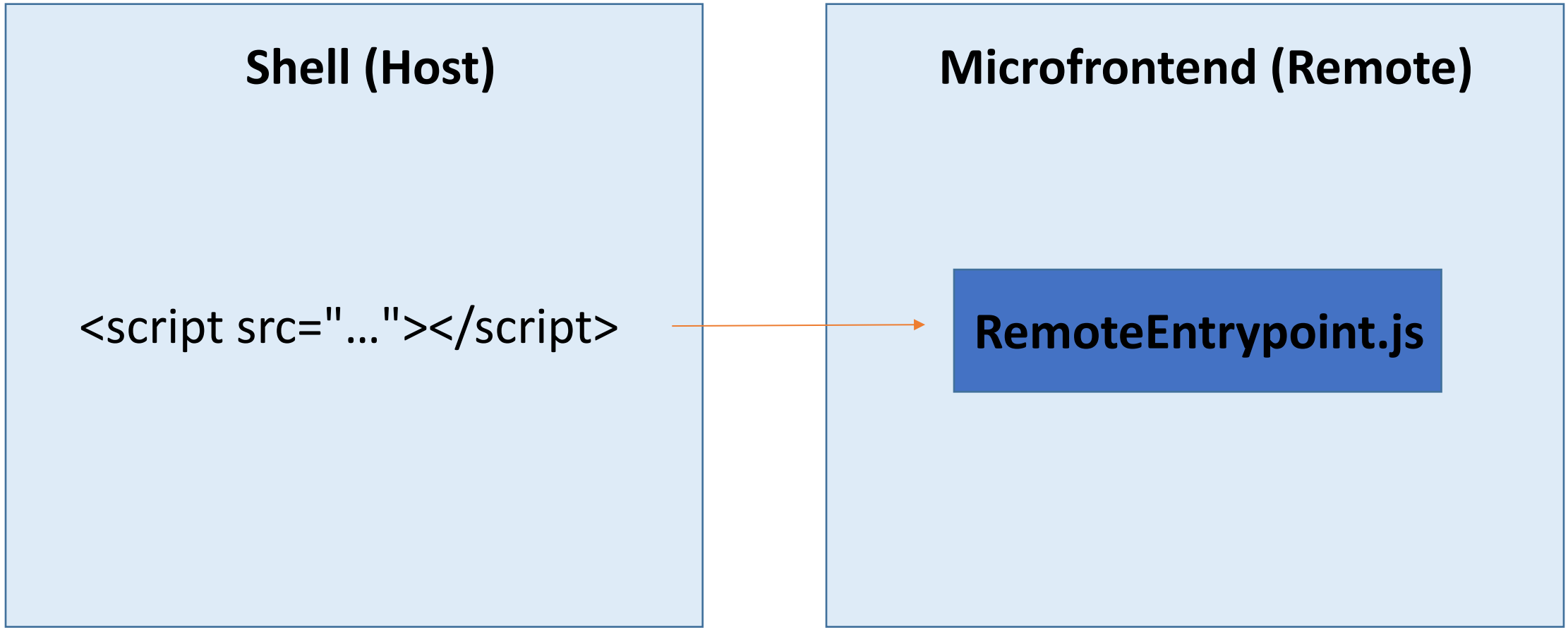
```
const Component = import('http://other-app/xyz')
```

Even lazy parts must be
known at compile time!

Webpack 5 Module Federation



How to Get the Microfrontend's URL?



How to Share Libs?

Shell (Host)

```
shared: [
  "@angular/core", "..."
]
```

Microfrontend (Remote)

```
shared: [
  "@angular/core", "..."
]
```

Dealing with Version Mismatches

ANGULAR ARCHITECTS
INSIDE KNOWLEDGE

Default Behavior

Selecting the highest compatible version

Default Behavior

Conflict: No highest compatible version

Example

- Shell: my-lib: ^10.0
- MFE1: my-lib: ^10.1
- MFE2: my-lib: ^9.0
- MFE3: my-lib: ^9.1

Result:

- Shell and MFE1 share ^10.1
- MFE2 and MFE3 share ^9.1

Configuring Singletons

```
shared: {
  "my-lib": {
    singleton: true
  }
}
```

11.0  10.1 

Configuring Singletons

```
shared: {
  "my-lib": {
    singleton: true,
    strictVersion: true // Error instead of warning!
  }
}
```

11.0 ✓ 10.1 ✗

Relaxing Version Requirements

```
shared: {
  "my-lib": {
     requiredVersion: ">=1.0.1 <11.1.1"
  }
}
```

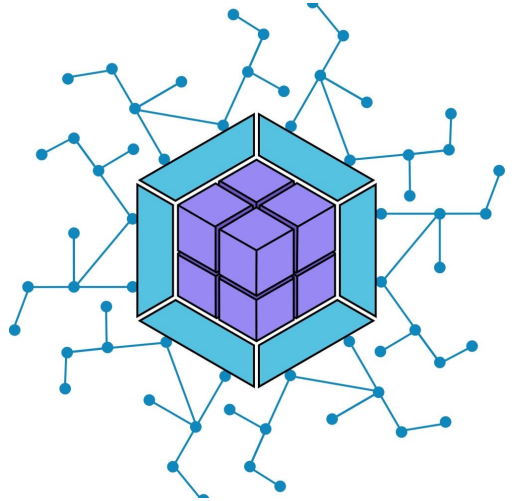
Federated Angular:
Angular, CLI, & Module Federation

ANGULAR ARCHITECTS
INSIDE KNOWLEDGE

webpack

Angular CLI

?

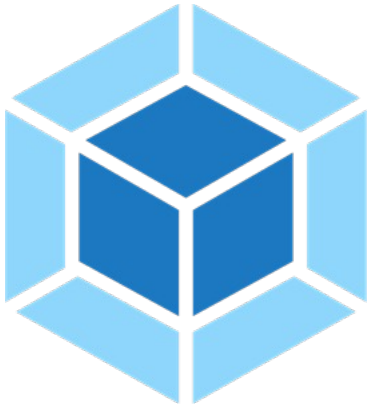


Module Federation
Configuration

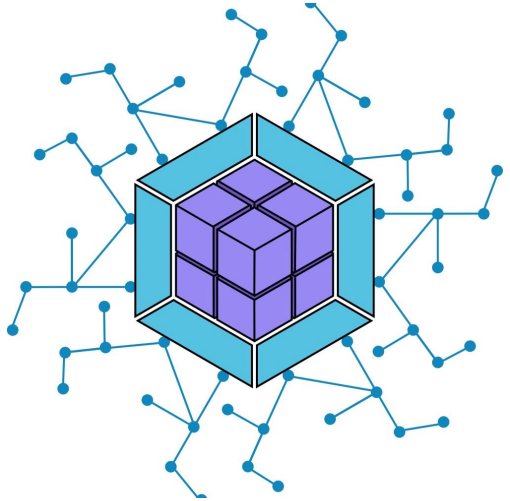
@a_awada

Custom Builder

Angular CLI



webpack



Module Federation
Configuration

@a_awada

@angular-architects/module-federation

1.0.2 • Public • Published 18 hours ago

Readme | Explore BETA | 3 Dependencies

Features

- ✅ Generates the skeleton for a Module Federation config.
- ✅ Installs a custom builder to enable Module Federation.
- ✅ Assigning a new port to serve (`ng serve`) several projects at once.

Usage

1) CLI
    1) ng add @angular-architects/module-federation
2) Nx
    1) npm install @angular-architects/module-federation –D
    2) nx generate @angular-architects/module-federation:init --project <project-name>
3) Adjust generated configuration
4) ng serve

DEMO

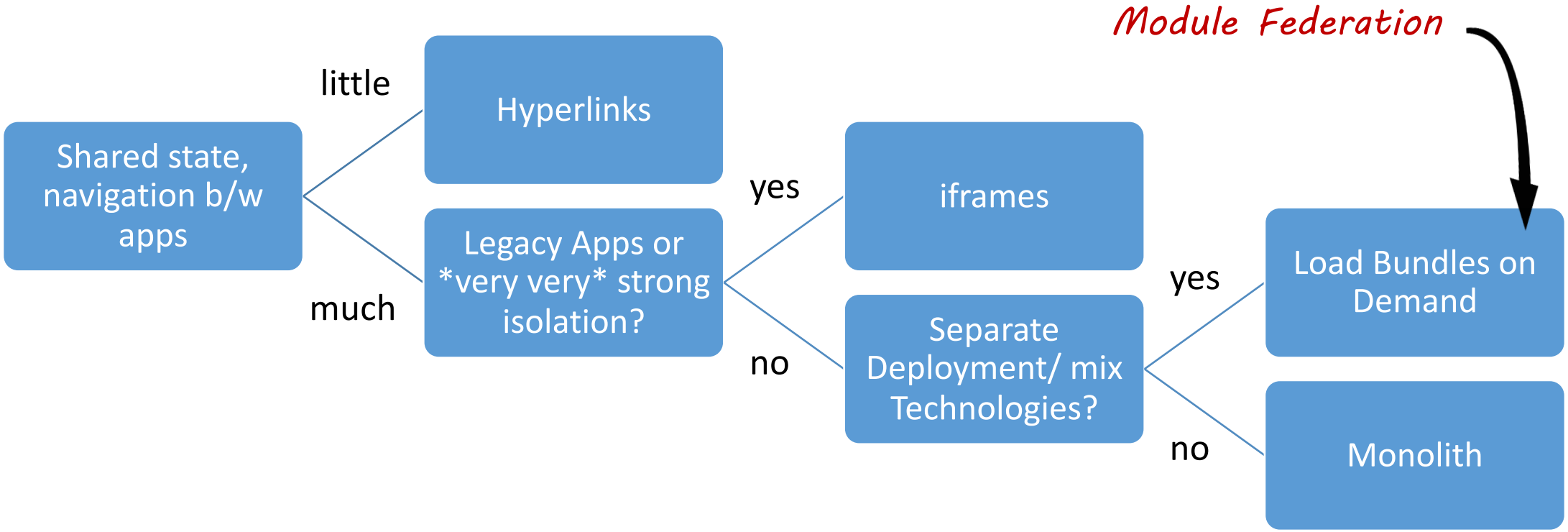
ANGULAR
ARCHITECTS
INSIDE KNOWLEDGE

LAB

ANGULAR
ARCHITECTS
INSIDE KNOWLEDGE


Choosing a Solution

Some General Advice



Summary

- Libs: Subdividing big solution into tiny parts
- Monorepo: No burden with distributing libs
- Strategic Design: Cut system into loosely coupled domains
- Nx: Access restrictions, visualization, incremental compilation/ testing, etc.
- Architecture Matrix: Orientation and access restrictions
- Microfrontends: Autarkic Teams, separate deployment, challenges