Microservices

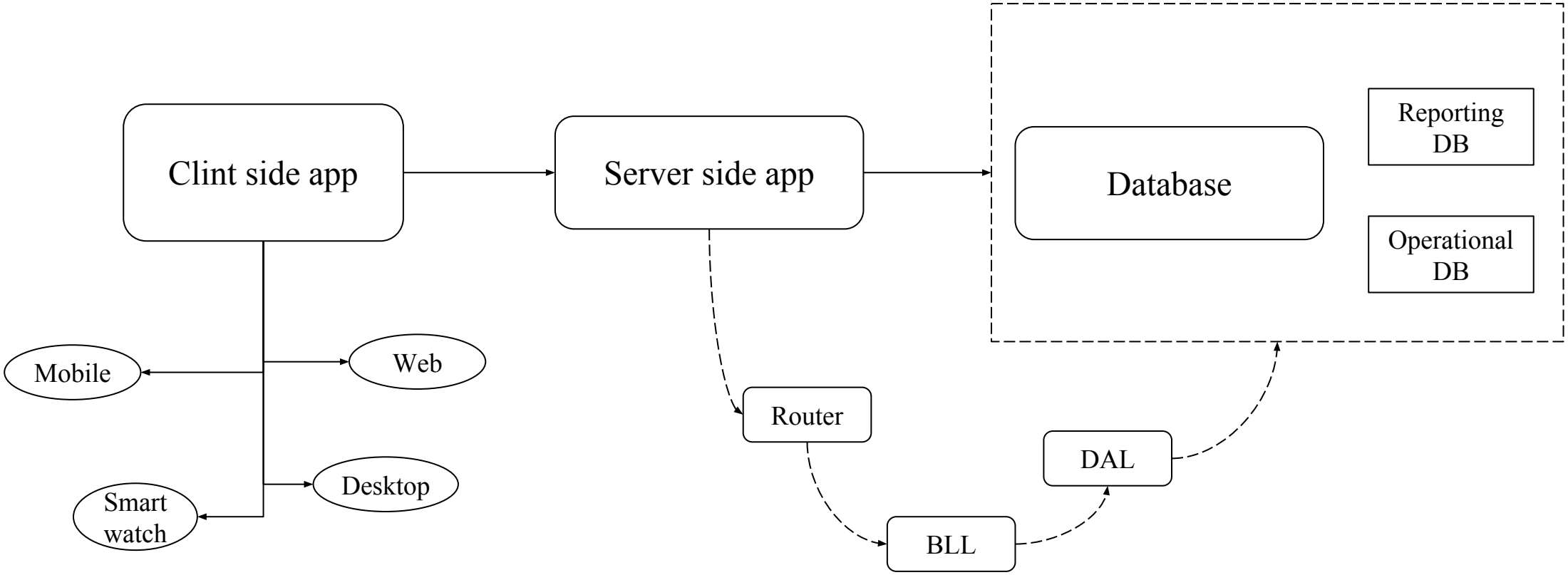
For Software Design and Architecture (SWE 4601)

This presentation is based on article by James Lewis and Martin Fowler:
https://martinfowler.com/articles/microservices.html

James Lewis

Martin Fowler

A typical web application architecture



Monolithic application

Characteristics

- Build as a single unit
- Tested as a single unit
- Deployed as single unit
- Usually the default architecture

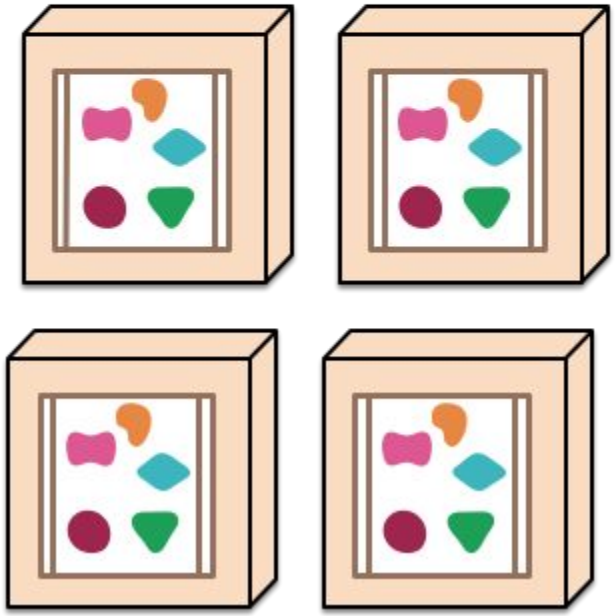
Drawbacks

- Any change require full rebuild and re-deployment
- Scaling of individual parts is not possible
  - What if only reporting need to be scaled?

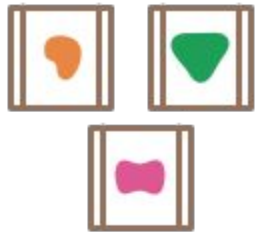
Monoliths and Microservices

A monolithic application puts all its functionality into a single process...

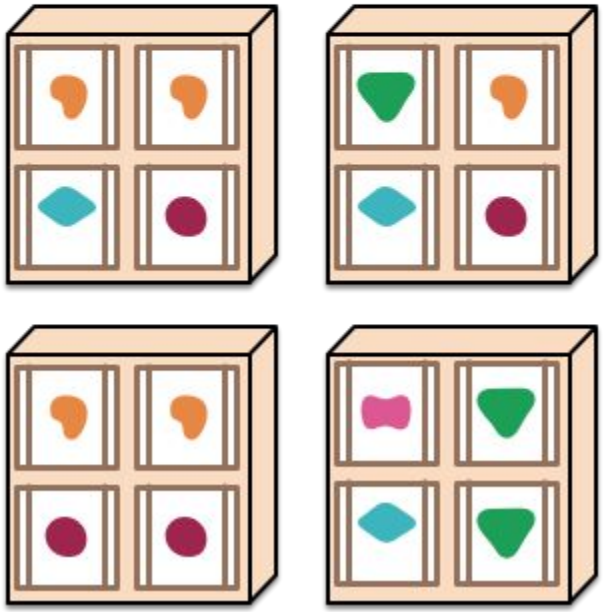
... and scales by replicating the monolith on multiple servers



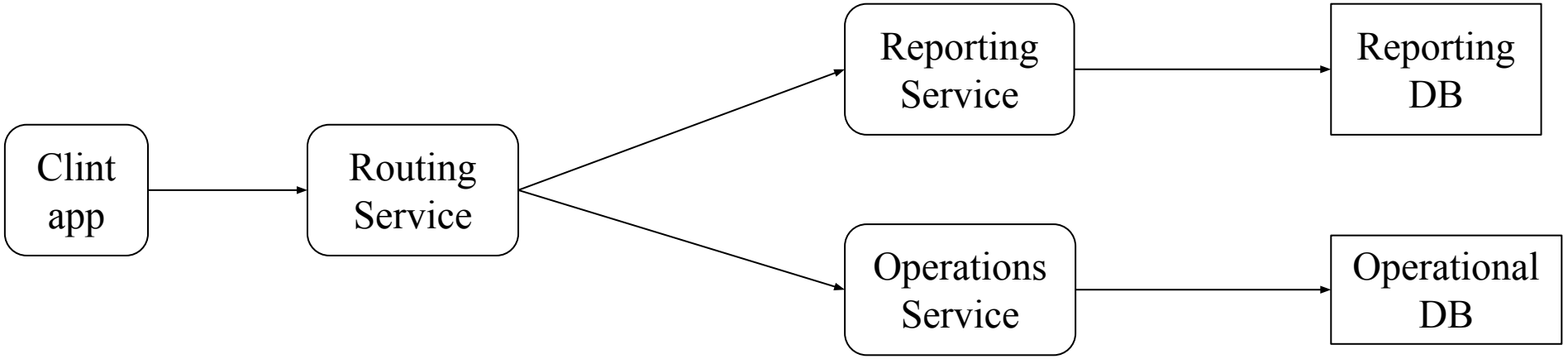
A microservices architecture puts each element of functionality into a separate service...



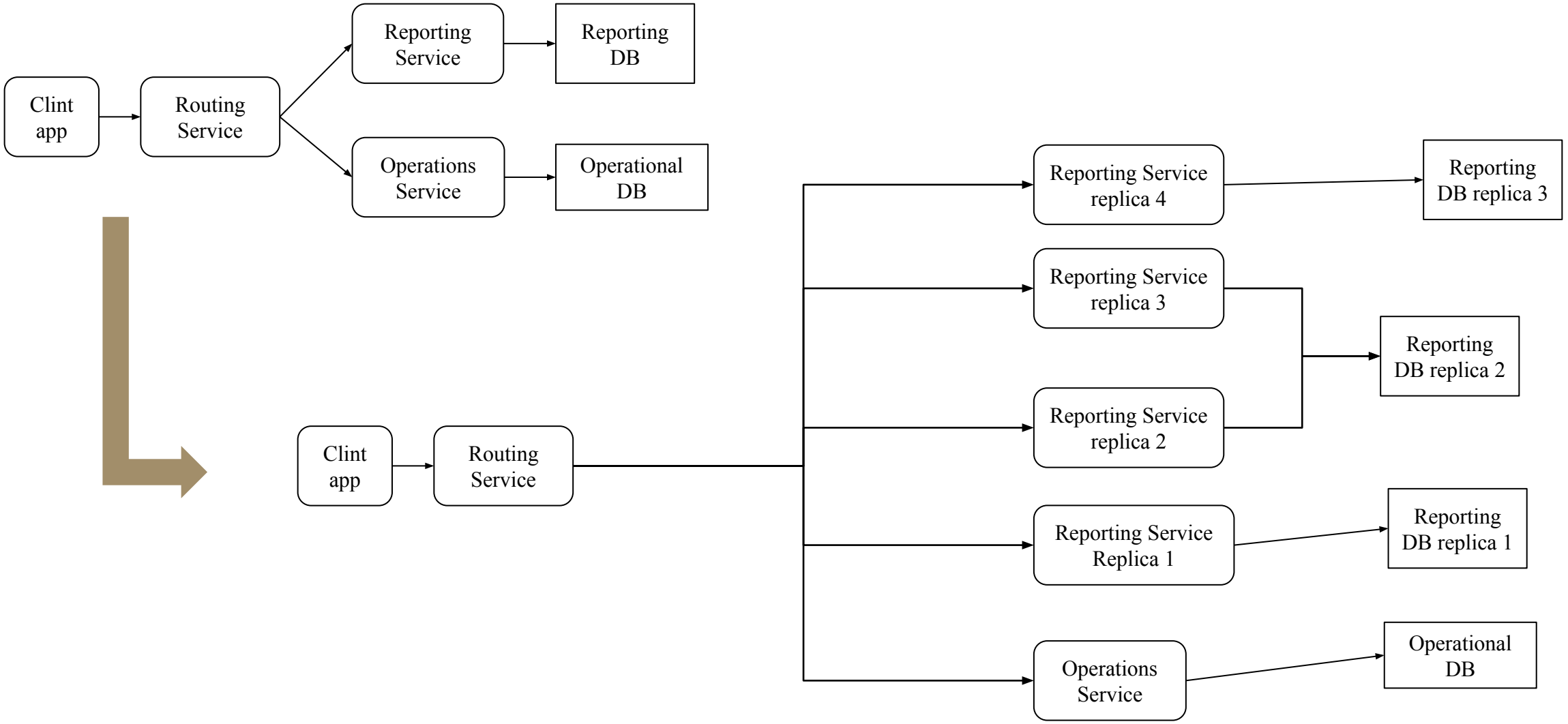
... and scales by distributing these services across servers, replicating as needed.



A naive example of microservice



Scaling the naive example



8

Characteristics of Microservices

Note: The topics in this section are not in the same order as the reference article

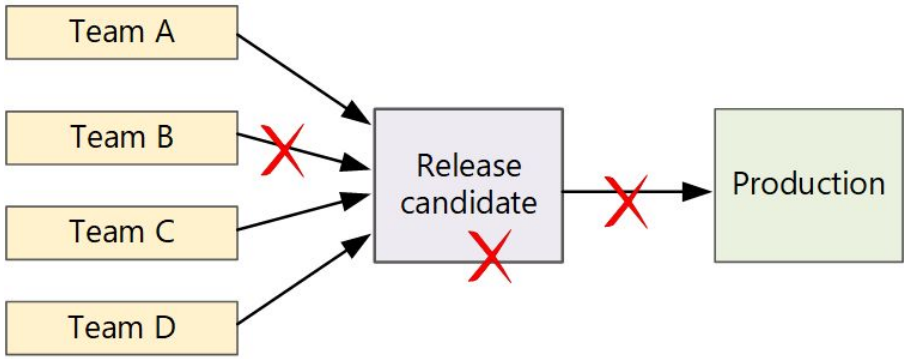
Microservice Characteristics: Componentization via Services

A component is a unit of software that is independently replaceable and upgradeable

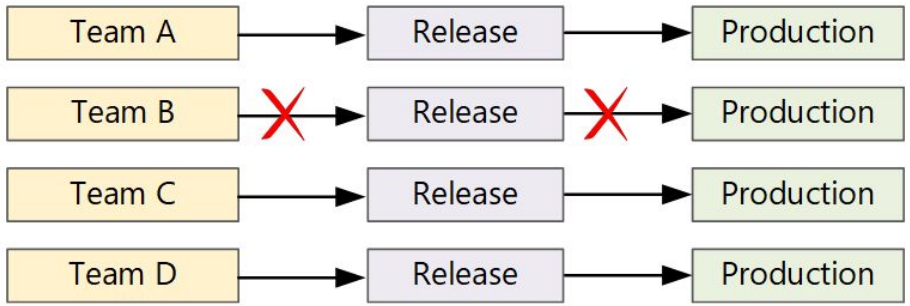
Libraries are components, so are services.

Libraries are linked to the program by in-memory/in-process calls

Services are linked to program by out-of-process calls: more expensive



Monolith



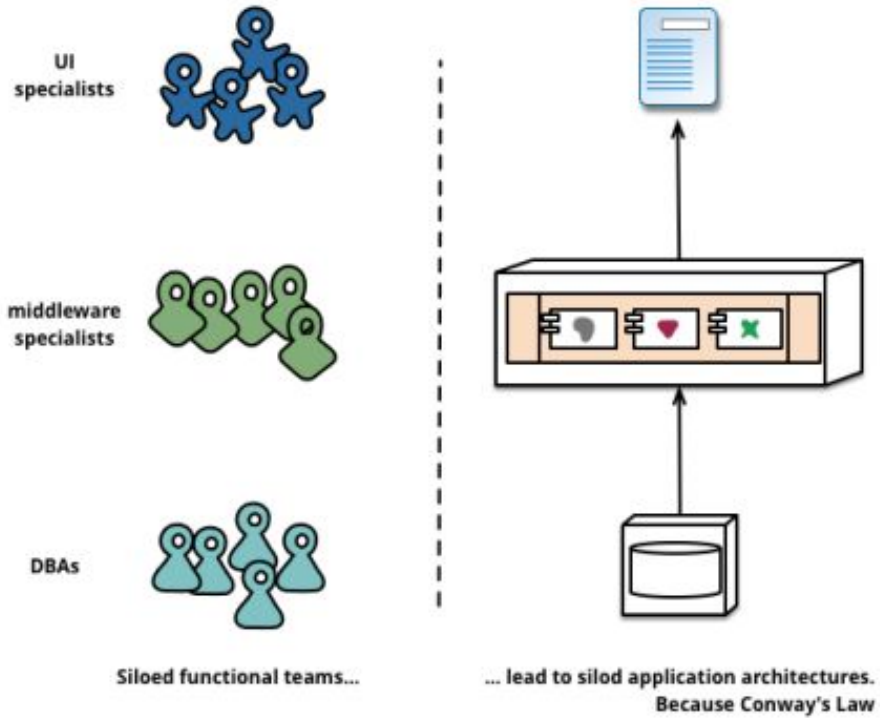
Microservices

Why services?

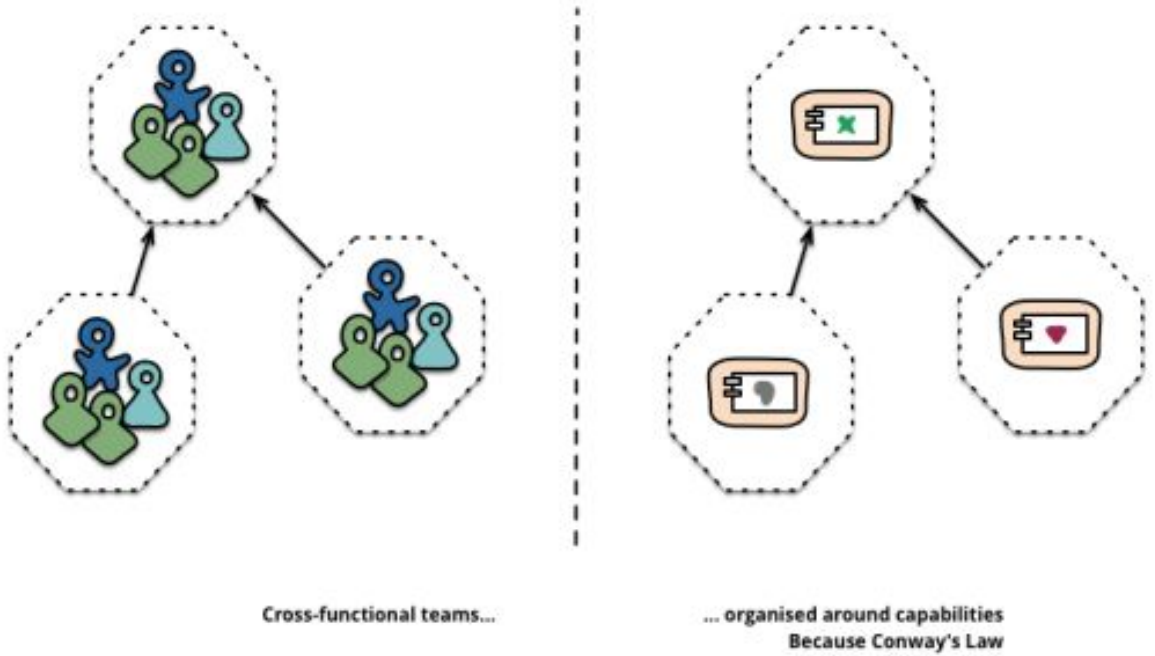
Services are independentl
deployable

Microservice Characteristics: Organized around Business Capabilities

1. If management focuses on the technology layer, leading to UI teams, server-side logic teams, and database teams.
2. simple changes can lead to a cross-team project taking time and budgetary approval



1. The microservice approach to division is different, splitting up into services organized around business capability.
2. the teams are cross-functional, including the full range of skills required for the development: user-experience, database, and project management



Microservice Characteristics: Products not Projects

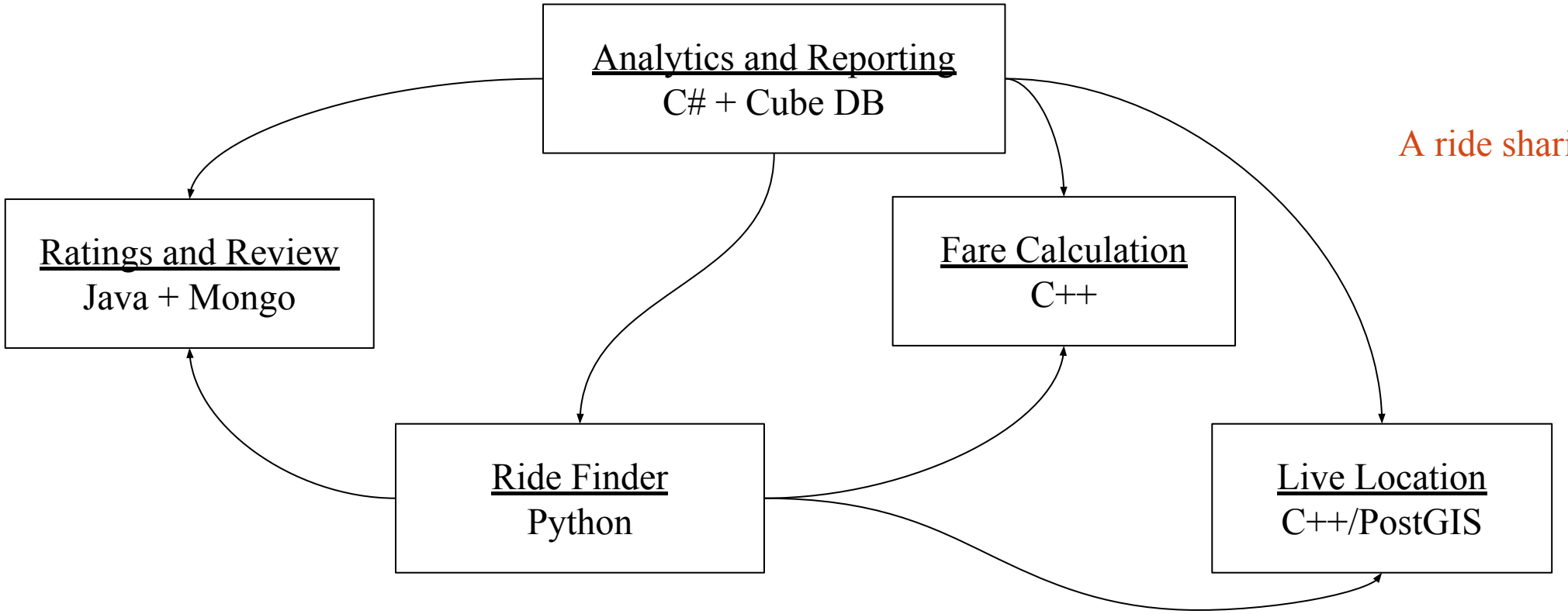
1. a **project model**: where the aim is to deliver some piece of software which is then considered to be completed.
2. Software is handover to a maintenance organization and the project team that built it is disbanded.

1. a **product model**: where the development team takes responsibility of the product development and maintenance following the notion “you build, you run it”.
2. The product mentality, ties in with the linkage to business capabilities. Microservice follows product model.

Microservice Characteristics: Decentralized Governance

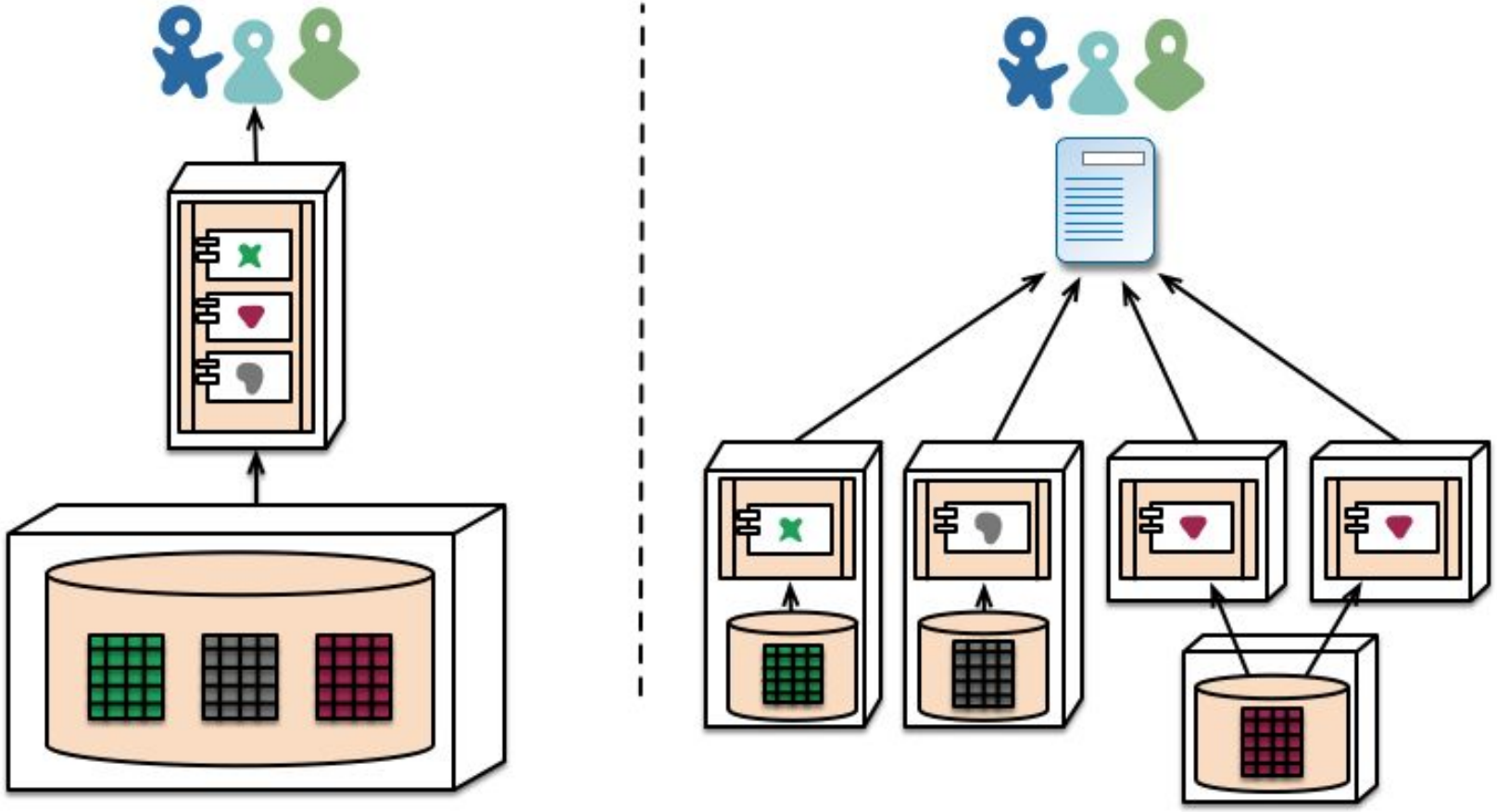
not every problem is a nail and not every solution a hammer

Can choose the right tool for a specific problem



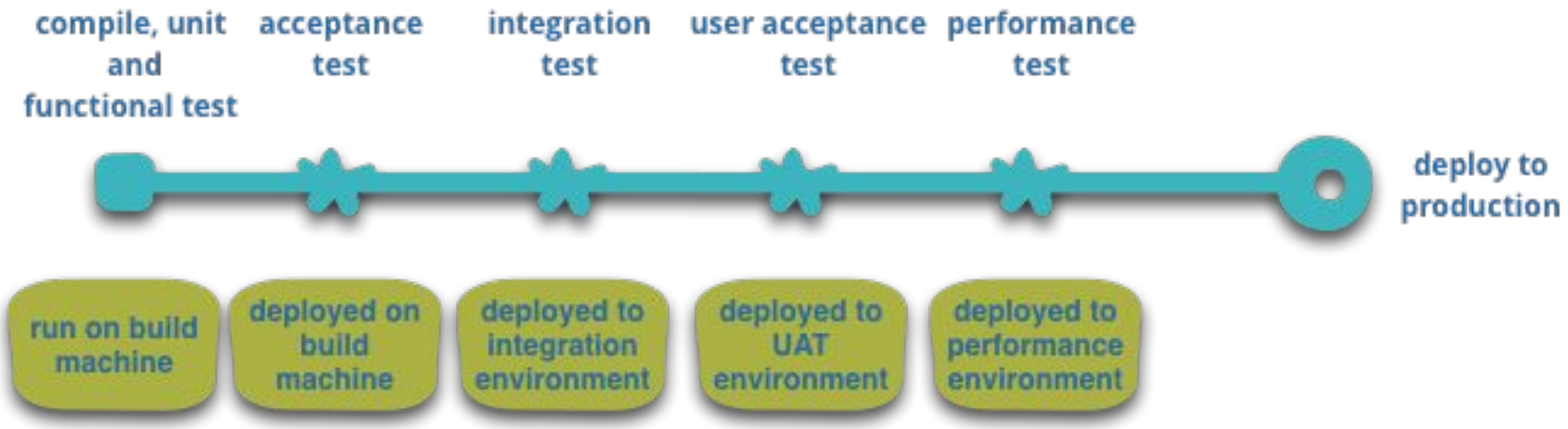
A ride sharing example

Microservice Characteristics: Decentralized Data Management



Concern: How to manage consistency across datastores?

Microservice Characteristics: Infrastructure Automation



Microservice Characteristics: Evolutionary Design

- When a team plans to move from monolith to microservice, usually the change will be “evolutionary”.
- Implement new functionality in services referring to the original monolith
- Gradually split services from the monolith

Not to be confused with evolutionary database design