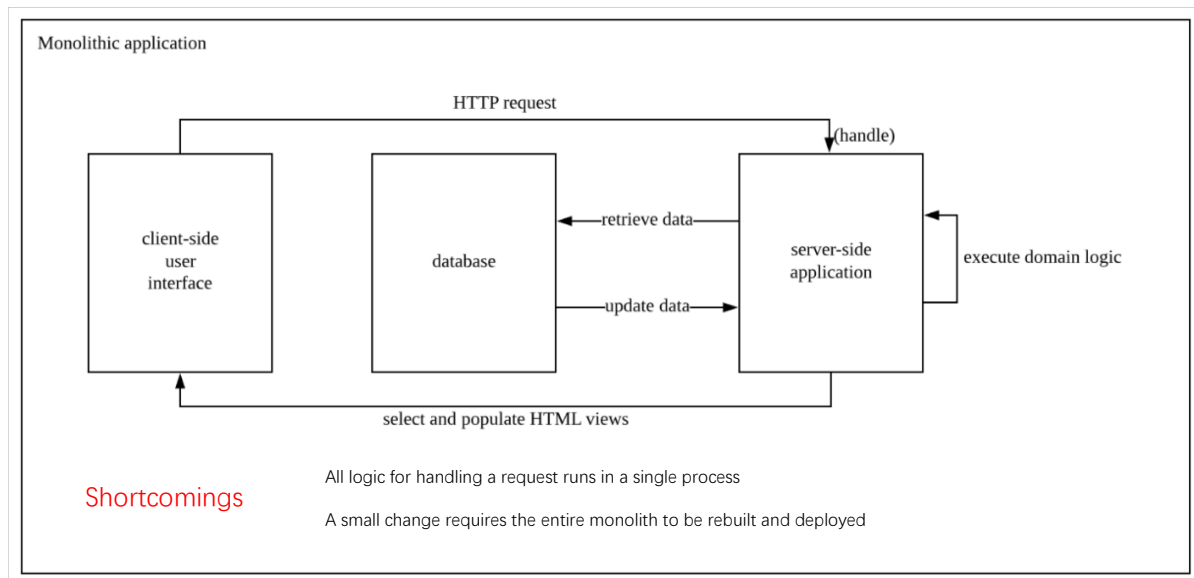


Monolithic



Microservice

Smaller granularity of services can make it easier to create relationships between developers and users



Martin Fowler

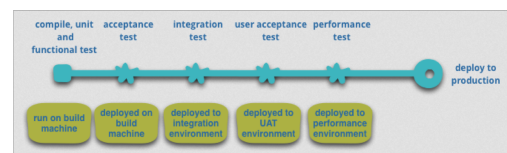
1. Microservice componentizes software by breaking down into **service**
2. Microservice is choreographed using simple RESTish protocols (HTTP request-response with API, lightweight messaging)
3. Decoupled and cohesive: applications own their own domain logic and can be independently deployable
4. Decentralized governance
5. Services can be implemented in various languages, and data management is decentralized

Ethos

Products not projects

You build, you run it

Microservice teams would expect to see sophisticated monitoring and logging setups for each individual service such as dashboards showing up/down status and a variety of operational and business relevant metrics.



Continuous integration & Continuous delivery pipeline

Risks

Not enough time has passed to judge microservices are the future direction for software architectures

It's hard to figure out exactly where the component boundaries should lie

If the components do not compose cleanly, then all you are doing is shifting complexity from inside a component to the connections between components.