



Kenzan Learning Series



MICROSERVICES

ARE THEY JUST REALLY SMALL SERVICES?

Bill Schwantz
Technical Architect

Monolith vs. Microservices



What is a monolith?

Monolith vs. Microservices



What is a monolith?

A single-tiered application where interface, business logic and data access are combined in a single program

Monolith



Problems:

Monolith



Problems:

- ▣ Scaling

Monolith



Problems:

- ▣ Scaling
- ▣ Handling DDOS Attacks

Monolith



Problems:

- ▣ Scaling
- ▣ Handling DDOS Attacks
- ▣ Handling Security Attacks

Monolith



Problems:

- ▣ Scaling
- ▣ Handling DDOS Attacks
- ▣ Handling Security Attacks
- ▣ Changing / Testing Code

Monolith



Problems:

- ▣ Scaling
- ▣ Handling DDOS Attacks
- ▣ Handling Security Attacks
- ▣ Changing / Testing Code
- ▣ Releasing New Versions

Microservices to the Rescue!



What are microservices?

Microservices to the Rescue!



What are microservices?

Yes, they really are just small services

Microservices to the Rescue!



What are microservices?

Yes, they really are just small services

Each service obeys the Single Responsibility Principle

Microservices to the Rescue!



What are microservices?

Yes, they really are just small services

Each service obeys the Single Responsibility Principle

Microservices talk to other microservices

Microservices Better Than Monolith?



Scaling

Scale just the service under load

- ▢ Saves resources
- ▢ Scales faster

Microservices Better Than Monolith?



Handle DDOS Attack

Only need to scale the service being attacked

- ▣ All other services unaffected

Microservices Better Than Monolith?



Handle Security Attack

Use a gatekeeper

Apply Principal of Least Privilege

Microservices Better Than Monolith?



Changing / Testing Code

The API for each microservice is its contract

So to test a change, we only need to test the API

Microservices Better Than Monolith?



Releasing New Versions

Just need to release the service(s) that changed

What Could Possibly Go Wrong?



What Could Possibly Go Wrong?



Dependency on monitoring to keep services running

What Could Possibly Go Wrong?



Dependency on monitoring to keep services running

Completely dependent on the network

- ▢ Network latency leads to poor service performance
- ▢ Broken interprocess communication

Netflix OSS Saves the Day



Service Discovery

Eureka

- Service registry
- Services register as they are starting up
- Periodic health checks and heartbeats
- Mid-tier load balancing

Netflix OSS Saves the Day



Interprocess Communication

Hystrix & Ribbon

- ▣ Graceful fallback
- ▣ Circuit breaking

Netflix OSS Saves the Day



Gatekeeper

Zuul

- ▣ Authentication (auth-n) & security
- ▣ Load shedding

Netflix OSS Saves the Day



Environment Monitoring

Asgard

- ▣ A/B testing
- ▣ Canary testing
- ▣ Changing number of instances of a service

Netflix OSS Saves the Day



Configuration Service

Archaius

- ▣ Dynamic changes of config values
- ▣ Can source config params from file or service

Netflix OSS is Not the Only Game in Town



Spring Cloud

Kubernetes

More than a handful of others

Q & A

