Why are microservices needed?

Why spring cloud?

<https://github.com/in28minutes/spring-microservices/03.microservices>

Introduction to microservices

Small autonomous microservices that work together

REST

Deployable units – well thought about boundaries

Cloud enabled

Diagram

Description automatically generated

Cloud enabled – multiple instances for each microservice

If there is more load on microservice3, I should be able to bring up an instance of microservice3 (this is what he means by cloud enabled)

How to cloud enable them ?

What are the challenges associated with cloud enabling your restful web services?

1. bounded context

2. configuration management

3. dynamic scale up and scale down

4. visibility (where is the bug?)

5. pack of cards if not well designed – if 1 goes down the rest goes down

Introduction to spring cloud

1. Spring cloud – provides tools to quickly build some of the common patterns in distributed systems
2. There are many projects under the umbrella of spring cloud
3. Spring Cloud Netflix – this is important – Netflix started playing around with microservices very early
4. Spring Cloud Config (centralized configuration management)
5. Configuration management – spring cloud config server

Diagram

Description automatically generated

1. Dynamic scale up and scale down

Diagram

Description automatically generated

Naming Server (Eureka) - all instances must register with naming server (service registration) – service discovery

Ribbon (Client Side Load Balancing) -

Feign – (rest client)

Visibility

* Zipkin distributed tracing (trace a request across multiple components)
* Netflix API Gateway – apply some common stuff across multiple microservices

Fault tolerance – hystrix

Advantages of microservices

1. new technology & process adaptation adoption – basically loose coupling

2. dynamic scaling – due to cloud enabled

3. faster release cycles – easier to release compared to monolithic applications

4.

5.

6.

7.

7 different projects

Limits service 8080 , 8081

Spring cloud config server 8888

Currency exchange service 8000 8001 8002

Currency conversion server 8100 8101 8102

Netflix eureka naming server 8761

Netflix zuul API gateway 8765

Zipkin distributed tracing 9411

There is a URLs section in the github