Spring boot-microservice full course

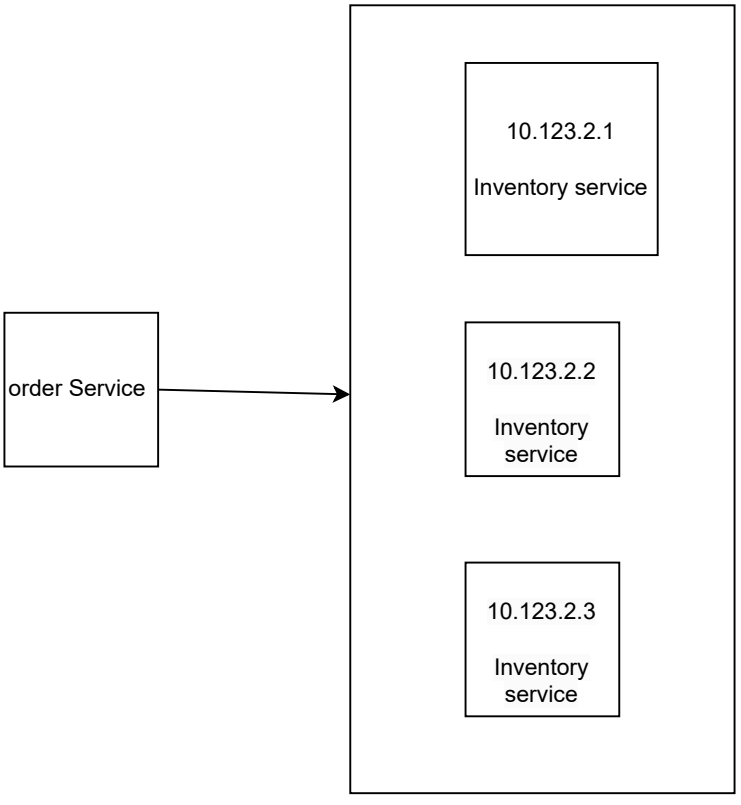
Objective
prerequisite
goals

Objective

- What is microservice?
- why to use them
- Communication between MS
- Deployment on server
- Security aspects

Pre-requisites

- knowledge of spring-boot
- basic knowledge of microservice
- JDK 1.8 and later
- IntelligIDEA or eclipse with maven



springcloud is used to build reliable and robot microservices
Its provide many design patterns

Features

- Distributed/versioned configuration
- Service registration and discovery
- Routing
- Service-to-service calls
- Load balancing
- Circuit Breakers
- Global locks
- Leadership election and cluster state
- Distributed messaging

We will develop simple onlice shopping application.

we will cover below points.

- service discovery
- centralized configuration- config server
- API gateway
- distributed logging
- event driven architecture.
- centralized logging
- circuit breaker
- secure microservice using keyclock

Services

Services that are going to build

Product service - Create and view product as product catalog.

order service - can order product.

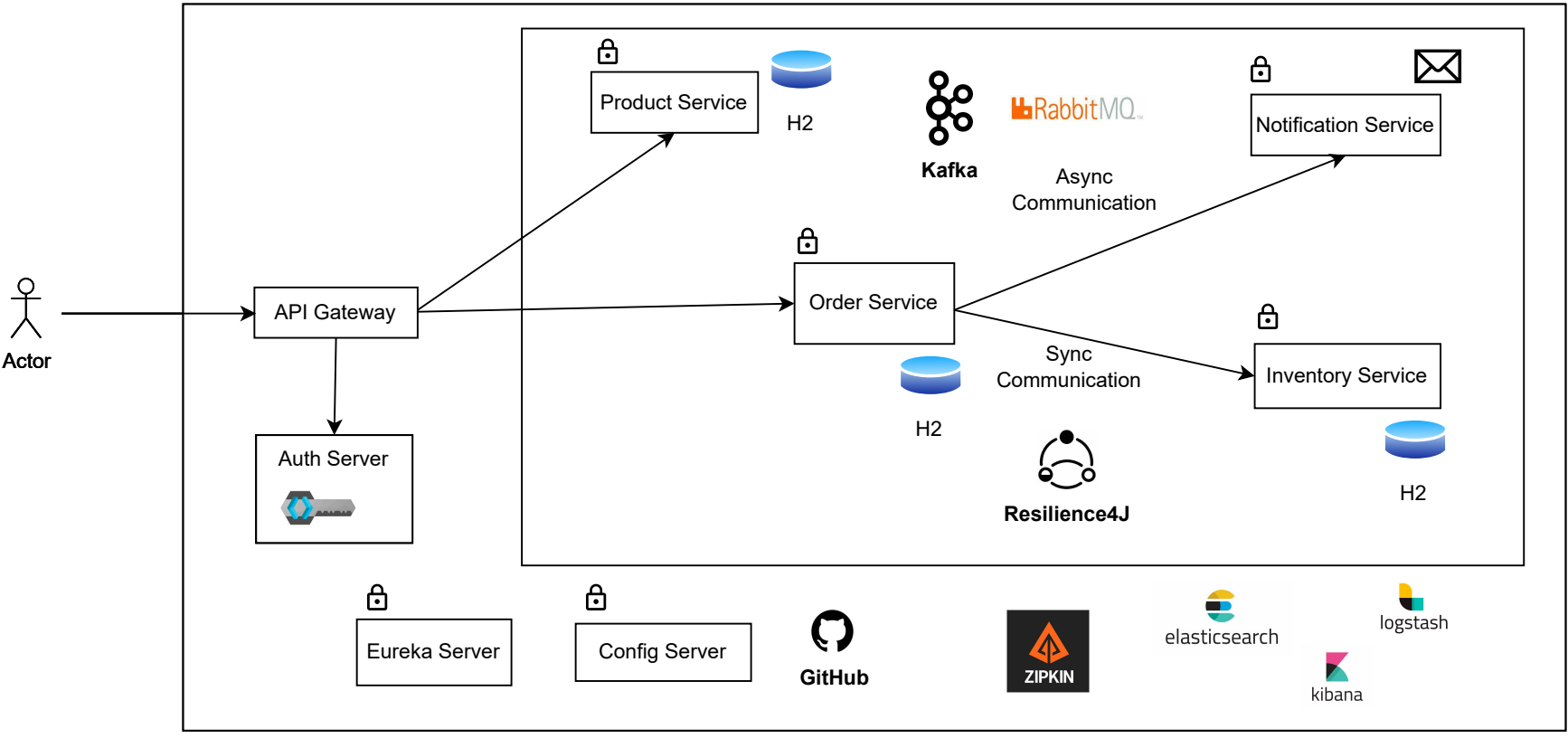
Inventory service - can check if product is in stock or not.

Notification service - Can send notification after order is placed.

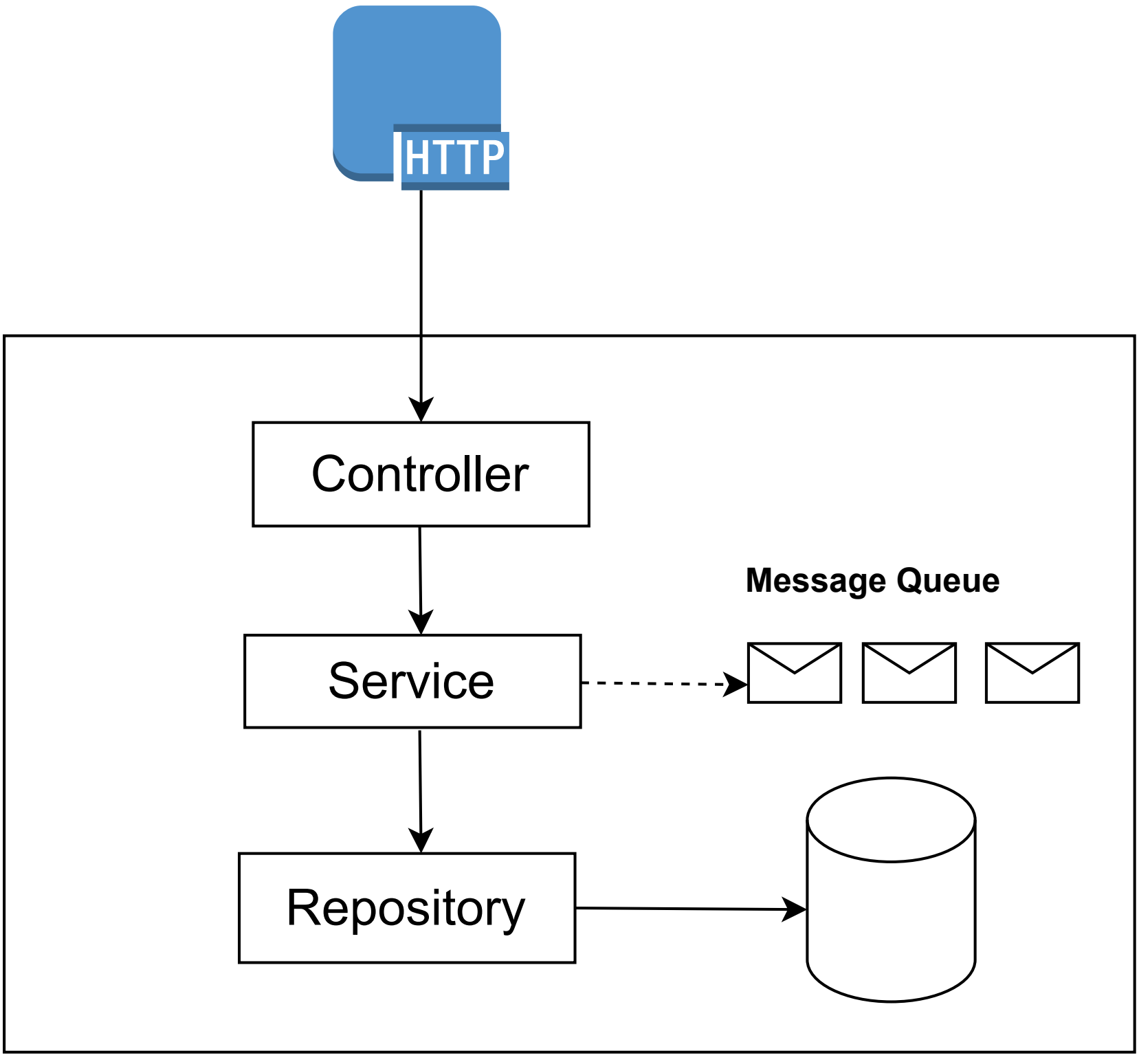
order service, inventory service and notification service are goiing to interact with each other.

Synchronous and asynchronous communication.

Technical Architecture



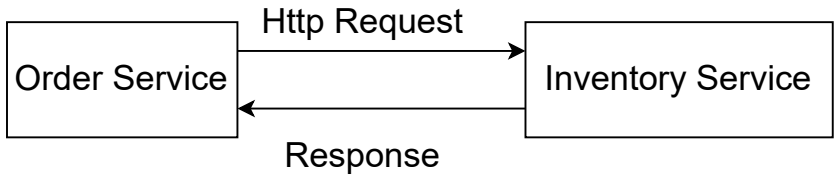
Logical Architecute



Inter Process Communication

RestTemplate

Web Client



Synchronous Communication

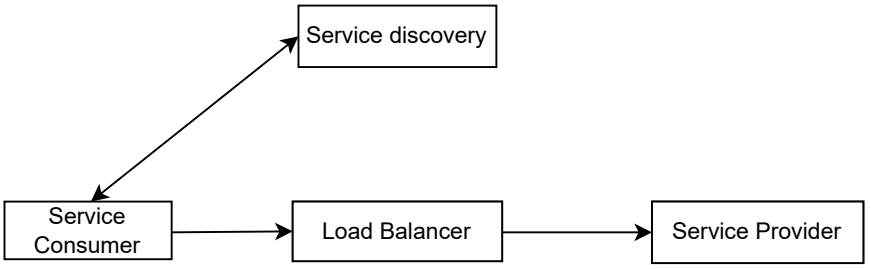
Rest Template

1. RestTemplate are blocking in nature
2. It uses one thread-per-request model of Java Servlet API.
3. RestTemplate only make synchronous requests.
4. RestTemplate creates new Httpconnection every request.
5. Creating and closing the URL connections(HTTP connection pooling) is a costly operation.
6. It puts a load on the server resources in case of hign request received.

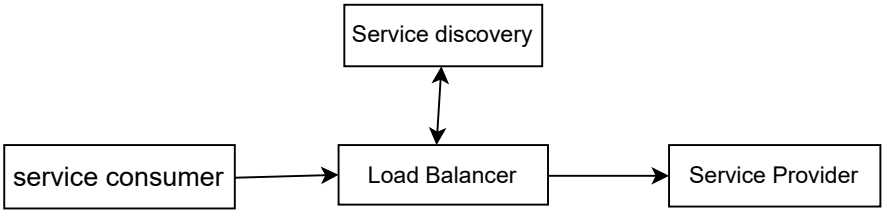
WebClient

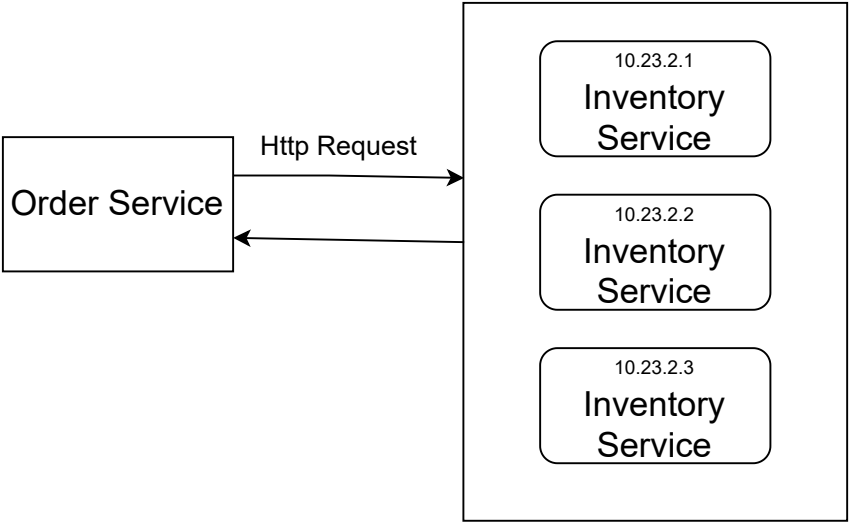
1. WebClient is asynchronous and non-blocking in nature.
2. It uses events-diven architecture from reactive framework of Spring WebFlux
3. We can use WebClient to make asynchronous and synchronous requests

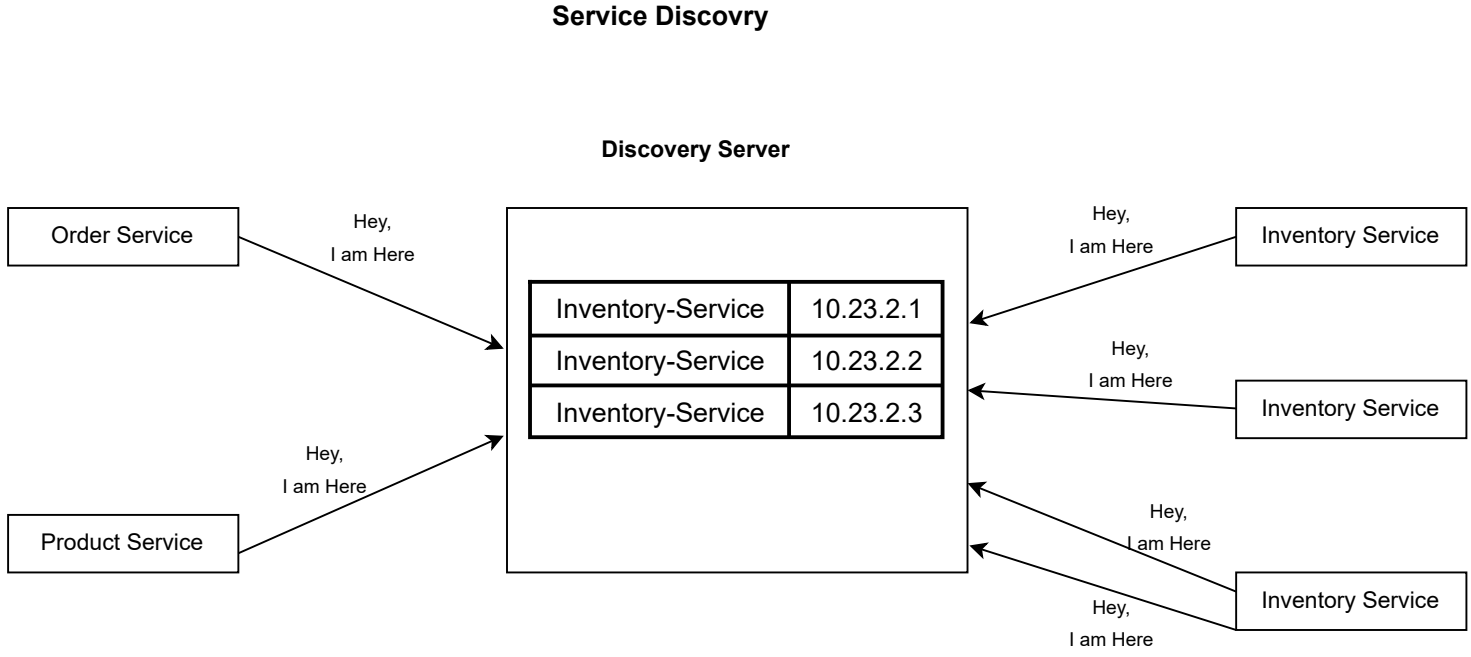
Client-side discovery

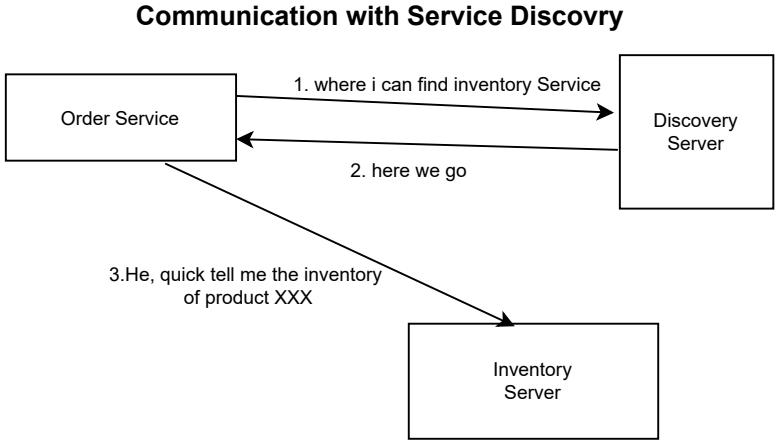


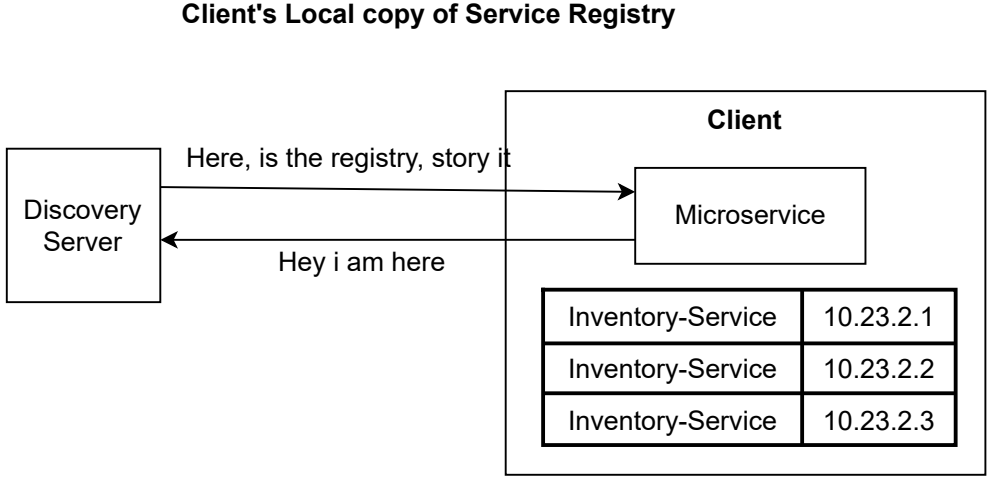
Server-side dicovery



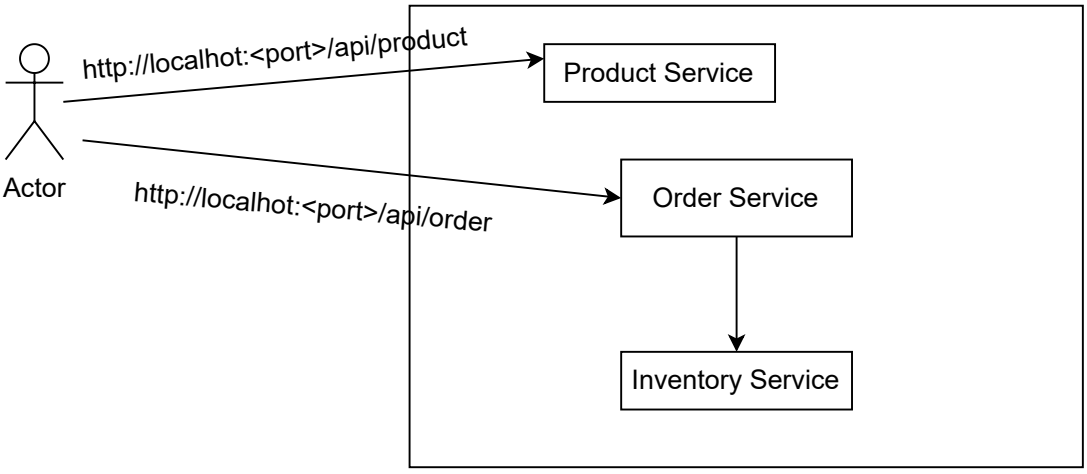

Order Service
Http Request
10.23.2.1
Inventory Service
10.23.2.2
Inventory Service
10.23.2.3
Inventory Service

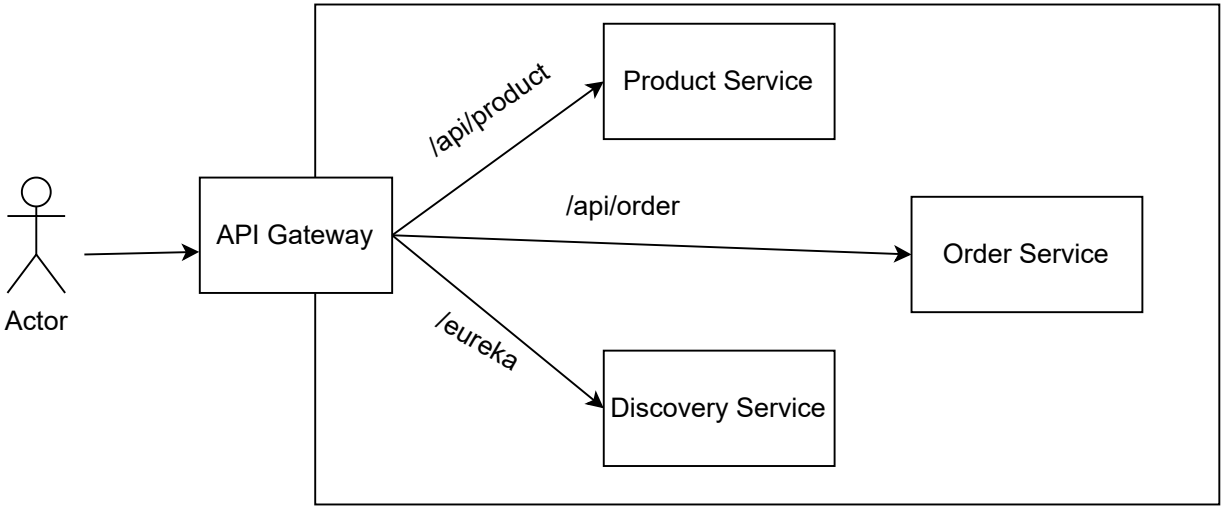

Service Discovry
Discovery Server
Order Service
Hey, I am Here
Product Service
Hey, I am Here
Inventory-Service 10.23.2.1
Inventory-Service 10.23.2.2
Inventory-Service 10.23.2.3
Hey, I am Here
Inventory Service
Hey, I am Here
Inventory Service
Hey, I am Here
Hey, I am Here
Inventory Service


Communication with Service Discovry
Order Service
1. where i can find inventory Service
2. here we go
Discovery Server
3.He, quick tell me the inventory of product XXX
Inventory Server


Client's Local copy of Service Registry
Client
Discovery Server
Here, is the registry, story it
Hey i am here
Microservice
Inventory-Service 10.23.2.1
Inventory-Service 10.23.2.2
Inventory-Service 10.23.2.3

API Gateway

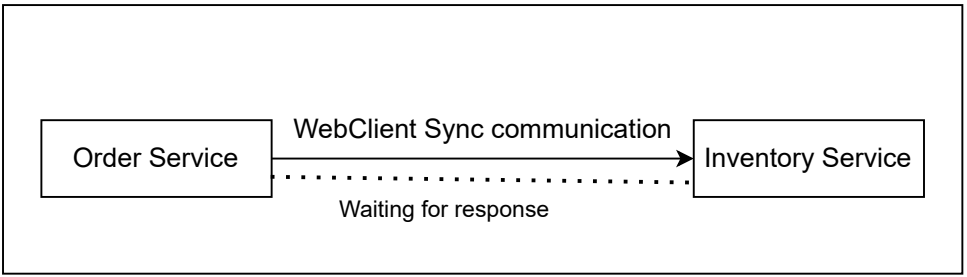




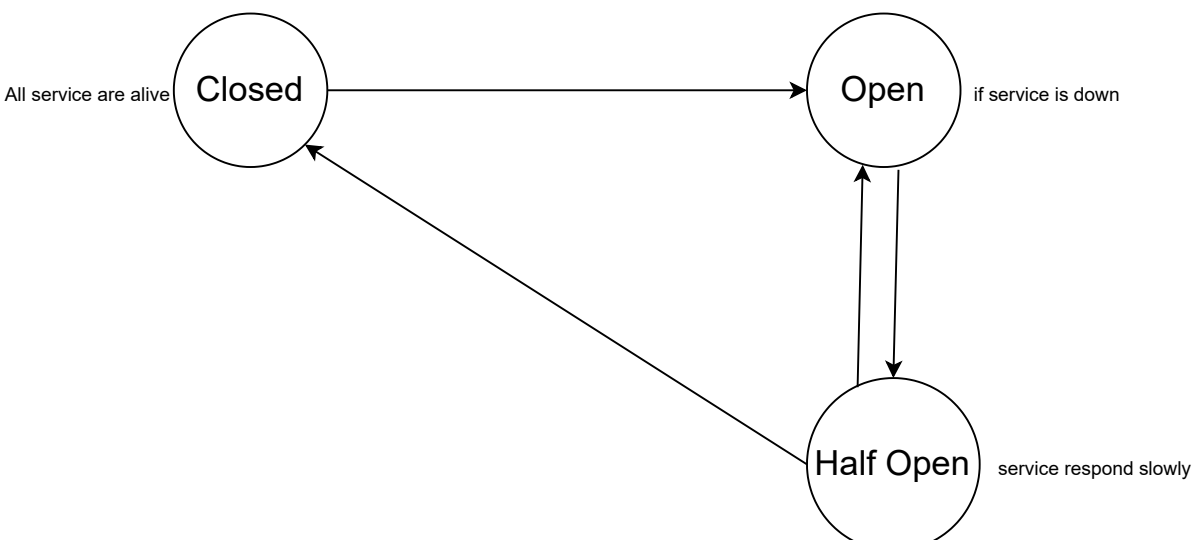
API gateway acts as

- Routing based Request Headers
- Authentication
- Security
- Load balancing
- SSL termination

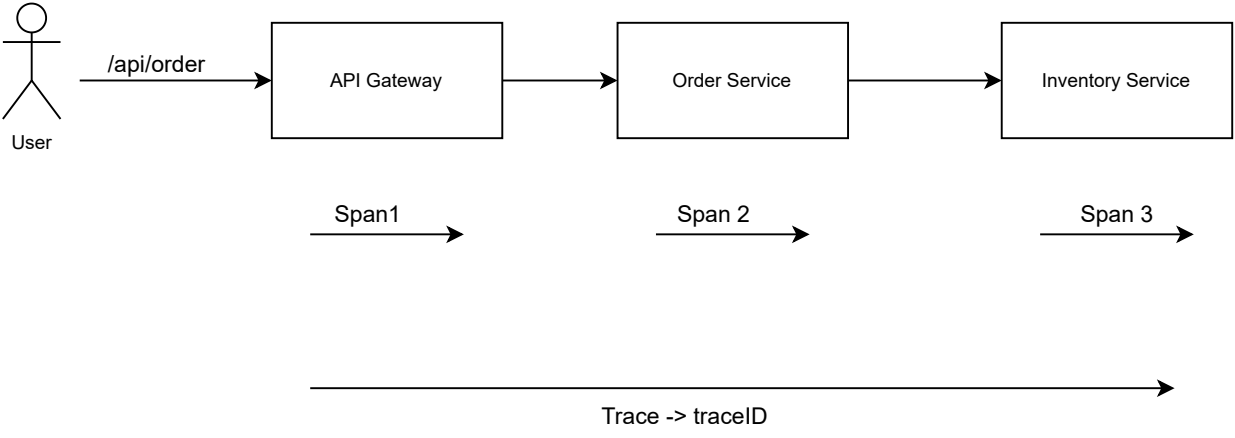
Circuit Breaker



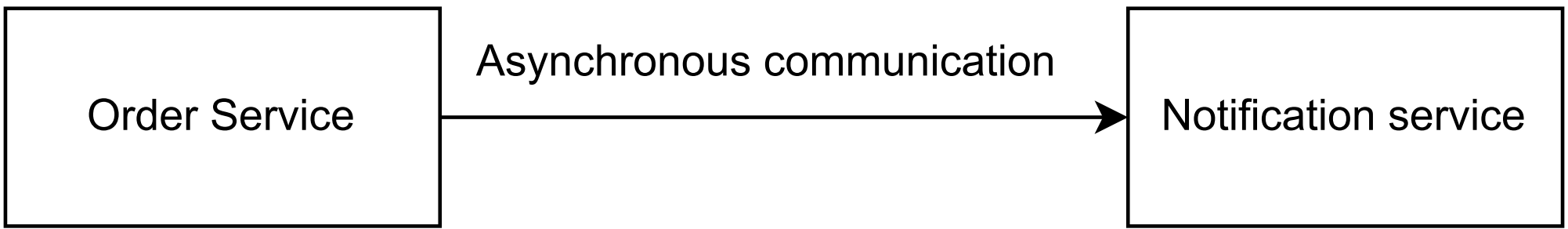
States of circuit breaker



Distributed Tracing



Event Driven Architecture



Monitoring of microservice