

Introduction

Tuesday, 18 December 2018

9:41 AM

All services are independent of each other, there is no tight coupling
Ms=microservices

- In microservices we will remove front end, means it will be total SEPARATE unlike
- In monolithic there is instance of all services, whereas in ms there is instance of each service as per need
- In monolithic there is less security, as we have to provide access to all services to do any operation
- Resource distribution is much more efficient in ms, as individually we can provide resource as per need
- eVEN front end can be a ms

Why we need service discovery(sd) in ms?

Since each services will have its own ip address.

In cloud environment the ip address are dynamic in nature

So to communicate between services with dynamic ip address we need **service discovery**

We need url to communicate between services which requires ip address of service

Services run on different port number

Instances of a single services can run on different as well as same port number

service discovery is kind of registry which helps services to communicate with instance of services

Every 30 seconds each instances send request to sd that it is running, if sd doesn't get response it will treat it as fail so **If sd goes down entire ms goes down.**

SD IS NOT PHYSICAL device it is all logical

that of monolithic
ch services depending upon
developer
resource to particular service

Load balancer(lb)

IN MS there are two categories of lb
Elastic lb an client side lb

Circuit breaker in ms

s of different
quest in 90s it

If any instance goes down , circuit breaker breaks the network
between sd and instance and hence sd knows about it. If
instance again comes up the cb again make the network
between instance and sd

CLIENT SIDE lb and cb are not mandatory,it just increases
performance

It is loosely coupled and highly cohesive

Highly cohesive means if any service require any information it is available to

Pros of ms

- Easy to understand
- nO tight coupling
- New components can be easily build and deployed
- Bugs will be limited
- iMProve fault tolerance

Cons

Deal with additional complexit

Required software

1. Oracle JDK 8.0
2. IDE - Spring Tool Suite
3. Database - PostgreSQL
4. DB Client - DBeaver
5. Rest Client - Postman

Microservice element