Forex Service (FS)

Currency Conversion Service (CCS)

FS:

Ie: Request: GET to http://localhost:8000/currency-exchange/from/EUR/to/INR

Response:

{

id: 10002,

from: "EUR",

to: "INR",

conversionMultiple: 75,

port: 8000,

}

CCS:

Request: GET to http://localhost:8100/currency-converter/from/EUR/to/INR/quantity/10000

Response:

{

id: 10002,

from: "EUR",

to: "INR",

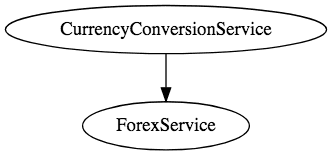
conversionMultiple: 75,

quantity: 10000,

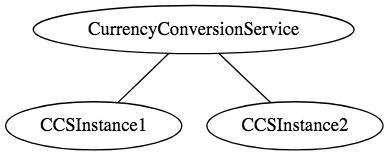
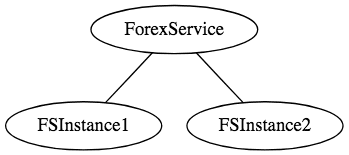
totalCalculatedAmount: 750000,

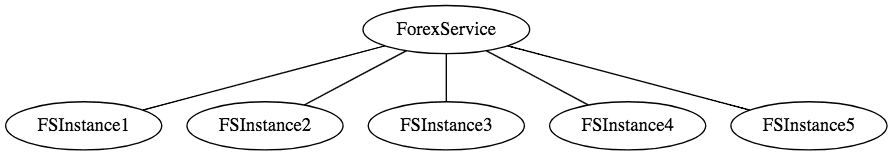
port: 8000,

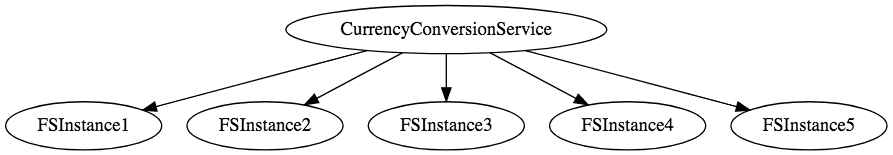
}

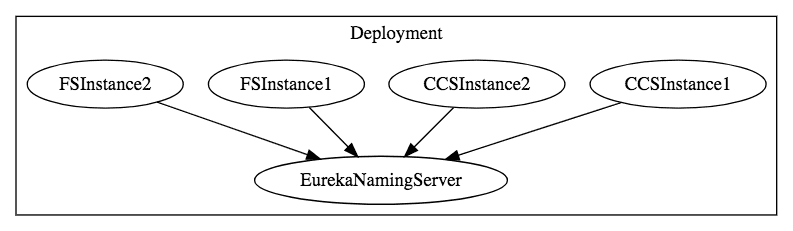


Spring Cloud – Eureka Naming Server and Ribbon - SCALING:

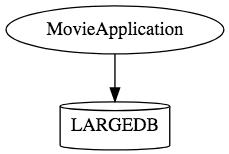
 



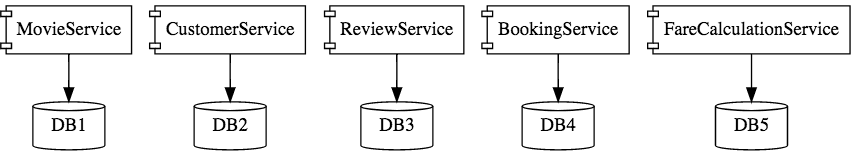


Deployments:  
  


Monolith Architecture:



Microservices Architecture:





**Spring Boot**

Enable building production ready applications quickly

Provide non-functional features

* embedded servers (easy deployment with containers): Tomcat
* metrics (monitoring)
* health checks (monitoring)
* externalized configuration

Where do you deploy? Either on Spring Cloud OR Kubernetes:

### Spring Cloud

Spring Cloud provides solutions to cloud enable your microservices. It leverages and builds on top of some of the Cloud solutions opensourced by Netflix (Netflix OSS).

#### Important Spring Cloud Modules

Dynamic Scale Up and Down. Using a combination of

* Naming Server (Eureka)
* Ribbon (Client Side Load Balancing)
* Feign (Easier REST Clients)

Visibility and Monitoring with

* Zipkin Distributed Tracing
* Netflix API Gateway

Configuration Management with

* Spring Cloud Config Server

Fault Tolerance with

* Hystrix

