BUZZWORD MICROSERVICE

**Monolith:**

* Large in app size
* Large in release cycle.
* Little diffcult from agile view
* Large in team size
* Should go with only one language (very sad ah)
* Have to deploy the entire stuff, if u change some code. (OMG)
* Have to hit ur head for the seeing the entire code!! (:

**Microserice:**

* Small automous service that work together
* Small services each running independently

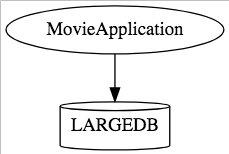
**Advantage:**

* Lightweight
* Try with different language for fun
* Donot need to hit ur head against the entire flow. See less code bro.
* Deploy independently bro! (great)
* Adapt to new techonolgy Or new process faster too
* Go for the faster release cycles
* Scaling. (U just spin up that service alone when u need it!, great ah, Think of the amazon on offer days)

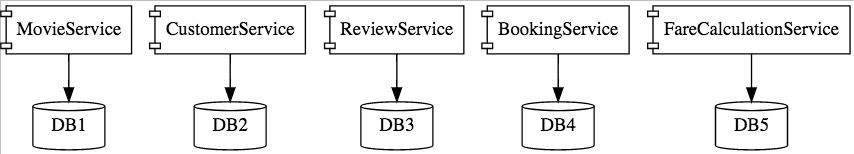
**How it will be?**

* Rest Api’s
* Well chosen deployable units with well defined boundary
* Cloud enabled

MONOLITH

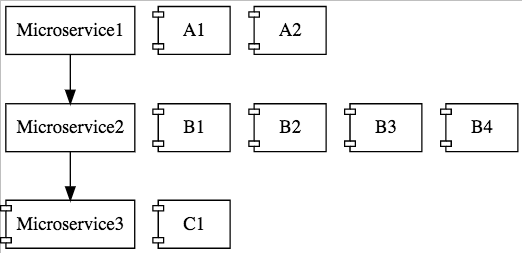


**Opposing to Micro:-**





SCALING



**Ah. Ok. What problems u will see:**

* Visibility and monitering
* Bounded Context. (Think of redefining the Ind – Pak Boundary)
* Configuration Management
* Debugging (Will be the best problem ever)
* Consistency

**List of tools for Solution:-**

Naming Server (Eureka)

Ribbon (Client Side Load Balancing), or Gateway between client and server

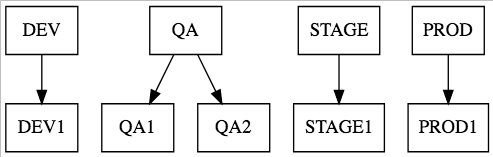
Feign (Declarative easier REST Client, Consider as oppose to WebClient in C#)

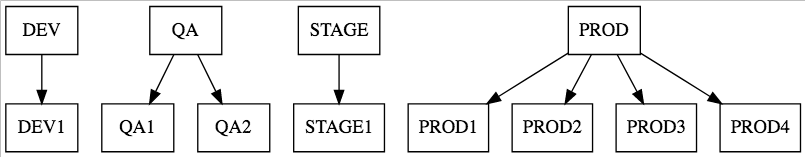
Visibility and Monitering:-

Zipkin Distributed Tracing

Netflix Api Gateway

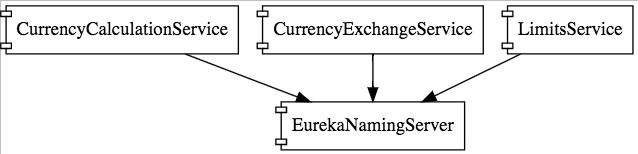
Microservice:-





HOW TO PROVIDE

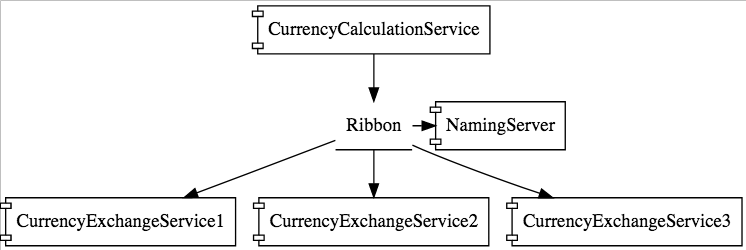
LOCATION TRANSPARENCY?



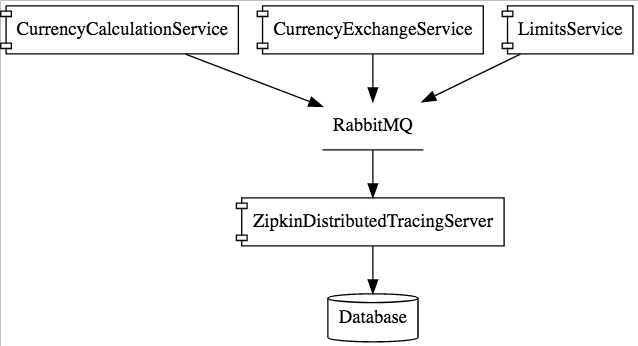
Eureka Naming Server

HOW TO PROVIDE

DYNAMIC LOAD BALANCING IN CLIENT SIDE?



**For messaging driven:-**



**API Gateway**

Authentication, authorization and security

Service Aggregation

**Fault tolerance ex:-**

@SpringBootApplication

@EnableHystrix

public class LimitsServiceApplication {

@GetMapping("/fault-tolerance-example")

@HystrixCommand(fallbackMethod="fallbackRetrieveConfiguration" public LimitConfiguration retrieveConfiguration() {

throw new RuntimeException("Not available");

}

public LimitConfiguration fallbackRetrieveConfiguration() {

return new LimitConfiguration(999, 9);

}

List of tools as alternate:-

**Cloud Discovery:-**

* Eurekha Disc
* Zookeeper Disc
* Clound Foundary Disc
* Consul Disc

**Fault Tolerance:-**

* Hystrix
* Turbine

**Cloud Tracing:-**

* Sleuth
* Zipkin

**Cloud Message – Communication**

* Cloud Bus AMPQ
* Cloud Bus Kafka
* Rabbit MQ, Active MQ etc.
* Azure Bus etc.