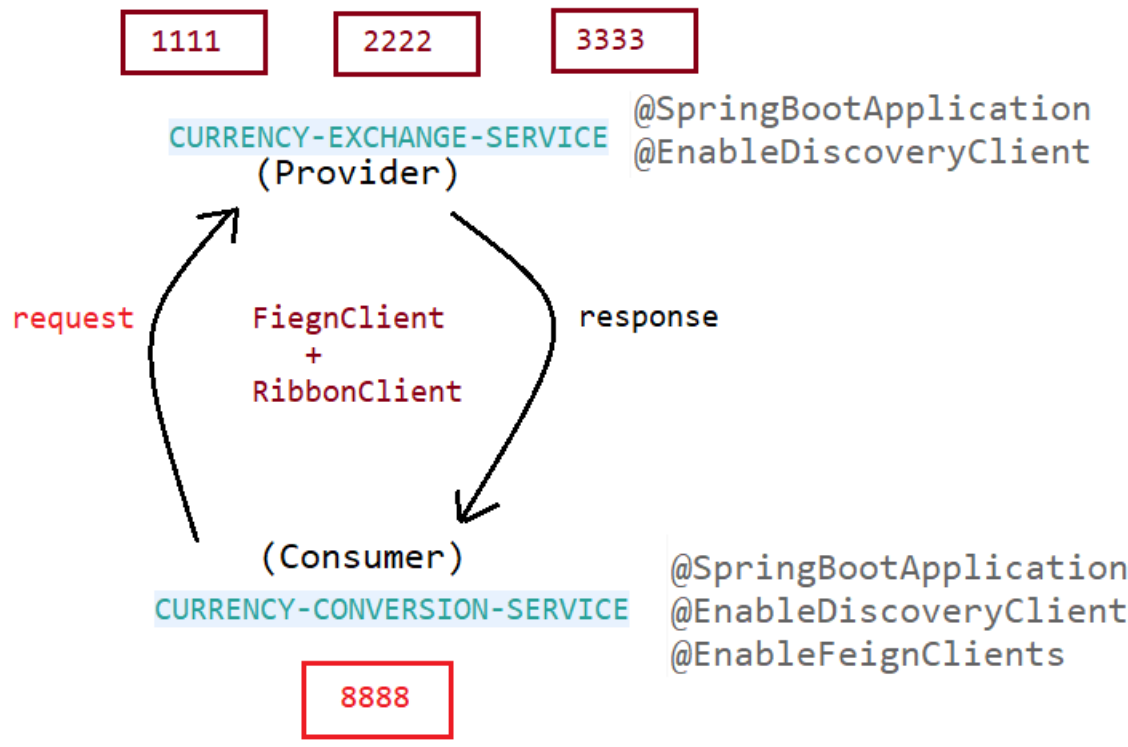
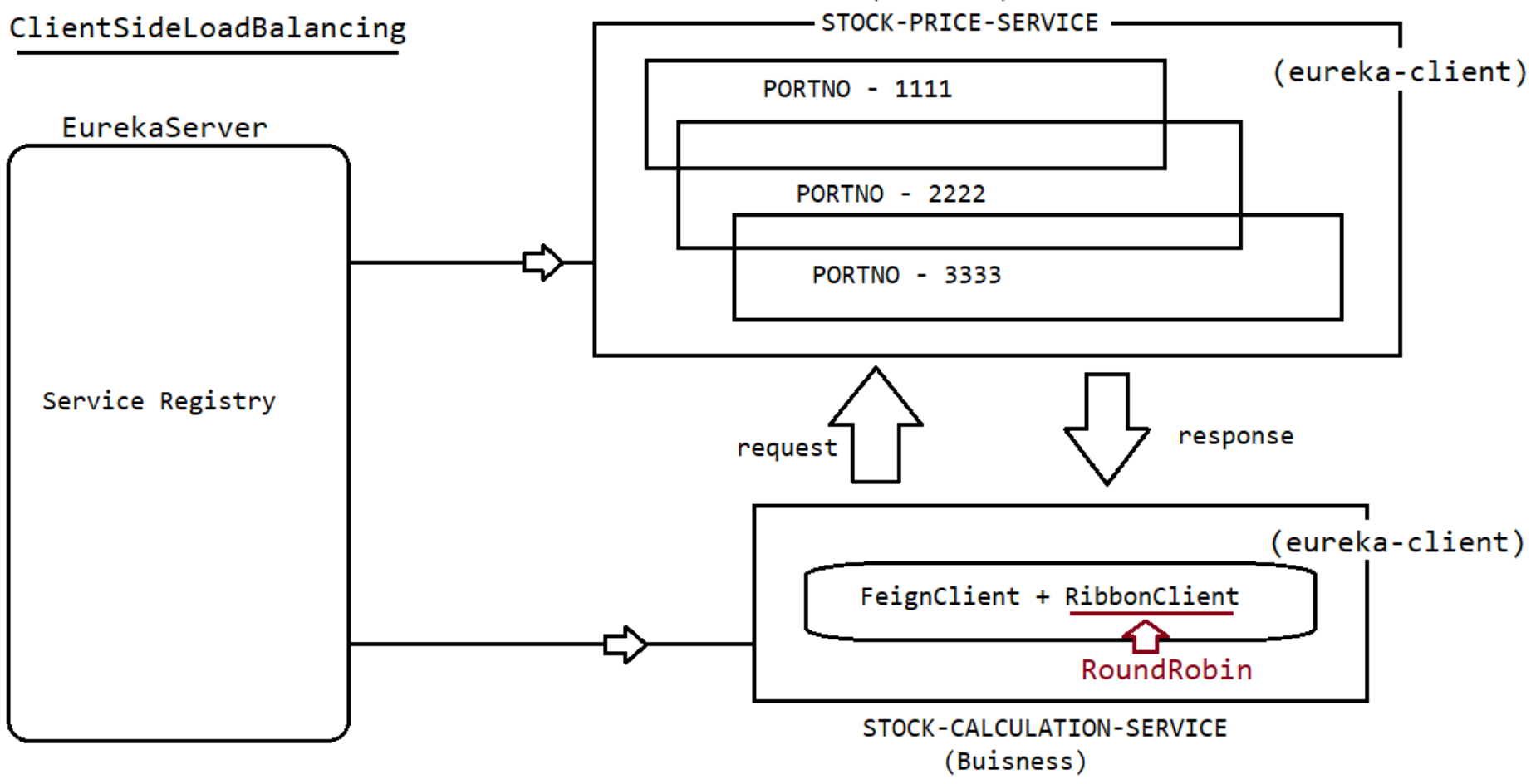
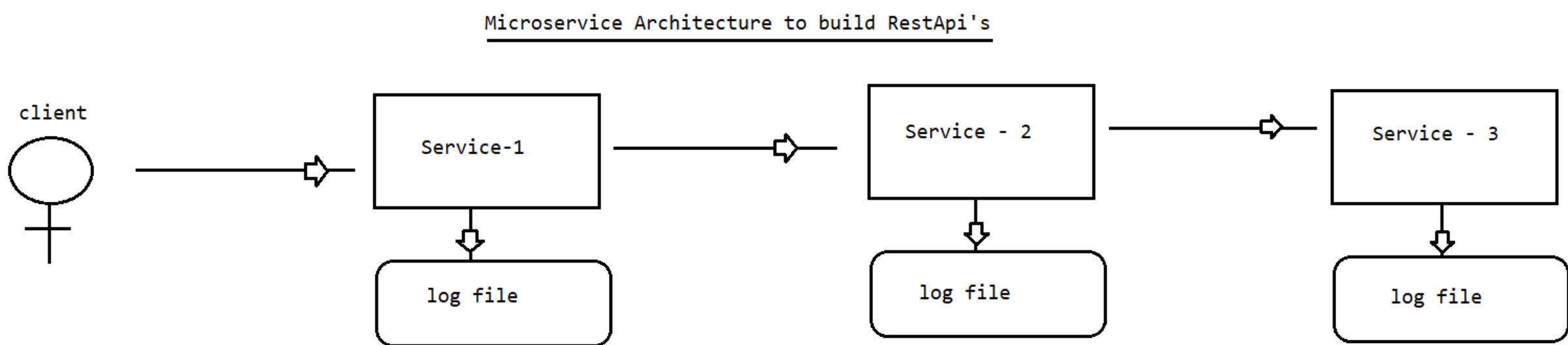
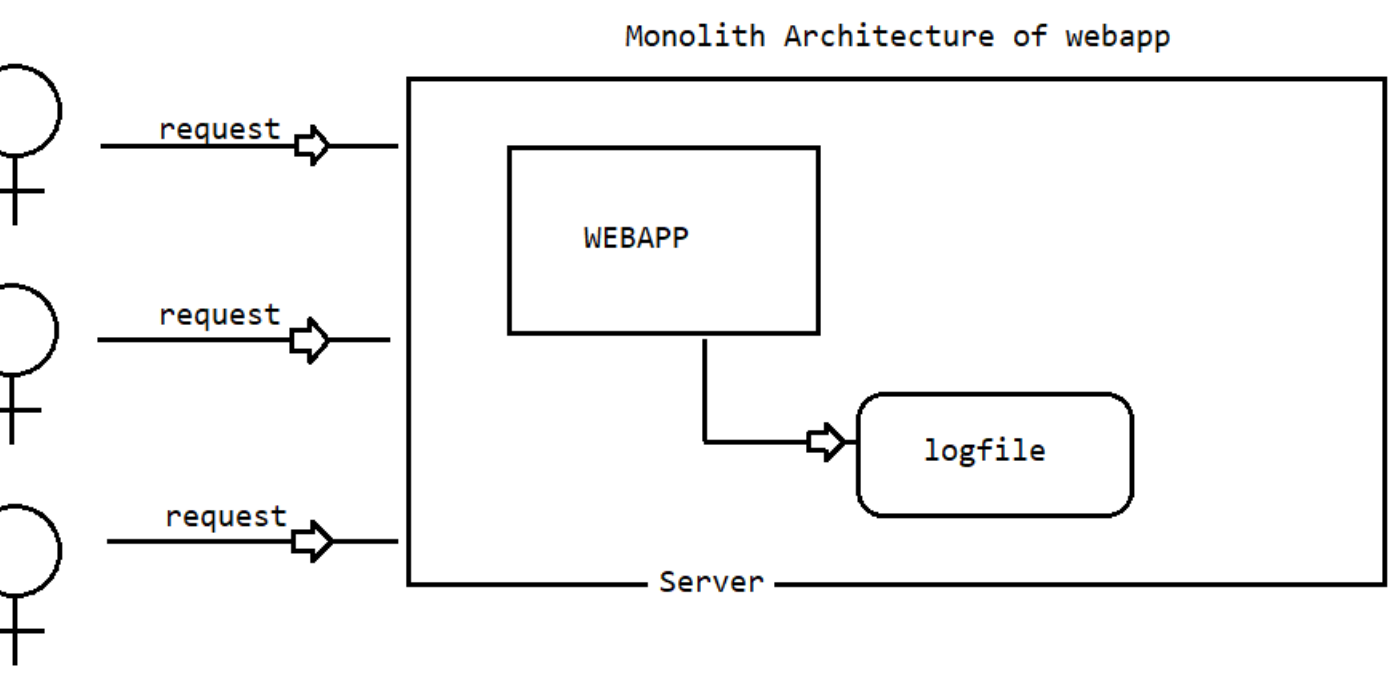


```
SB-DEMO-REST-APP [8080]
  Deployment Descriptor: SB-DEMO-REST-APP
  JAX-WS Web Services
  Java Resources
    src/main/java
      in.neuron
        in.neuron.rest.controller
          WelcomeTestController.java
    src/main/resources
      static
      templates
      application.properties
  src/test/java
    target/generated-sources/annotations
    target/generated-test-sources/test-annotations
  Libraries
  Deployed Resources
    src
    target
    HELP.md
    mvnw
    mvnw.cmd
    pom.xml
```

Instances currently registered with Eureka			
Application	AMIs	Availability Zones	Status
CURRENCY-CONVERSION-SERVICE	n/a (1)	(1)	UP (1) - DESKTOP-1NSU4UJ-CURRENCY-CONVERSION-SERVICE-8888
CURRENCY-EXCHANGE-SERVICE	n/a (3)	(3)	UP (3) - DESKTOP-1NSU4UJ-CURRENCY-EXCHANGE-SERVICE-1111, DESKTOP-1NSU4UJ-CURRENCY-EXCHANGE-SERVICE-2222, DESKTOP-1NSU4UJ-CURRENCY-EXCHANGE-SERVICE-3333



FeignClient will contact Service Registry through which it will get the access to our services, but if our services have many url's then we need to use "RibbonClient" at the client side to resolve the problem.



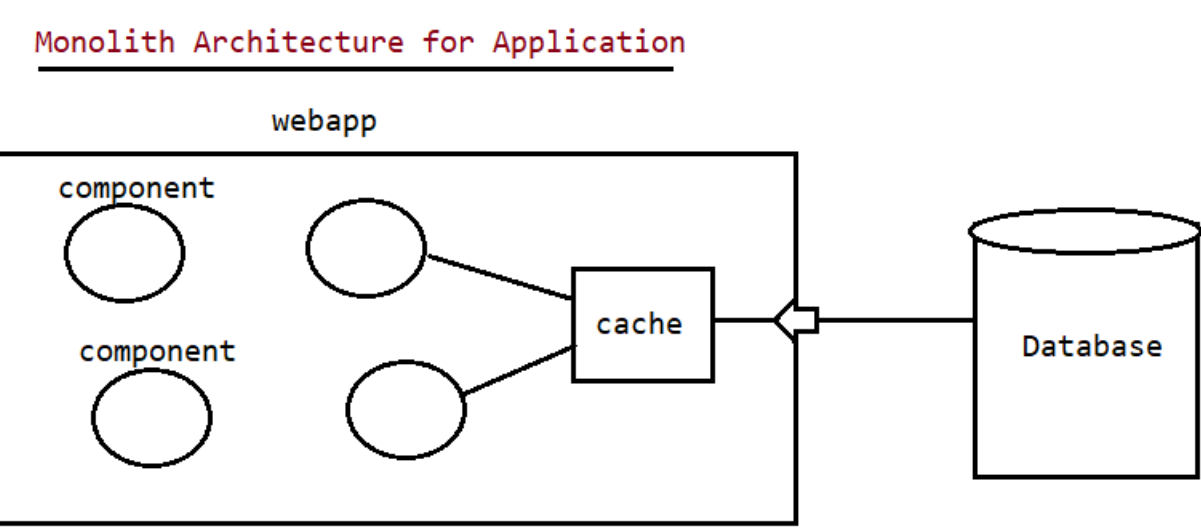
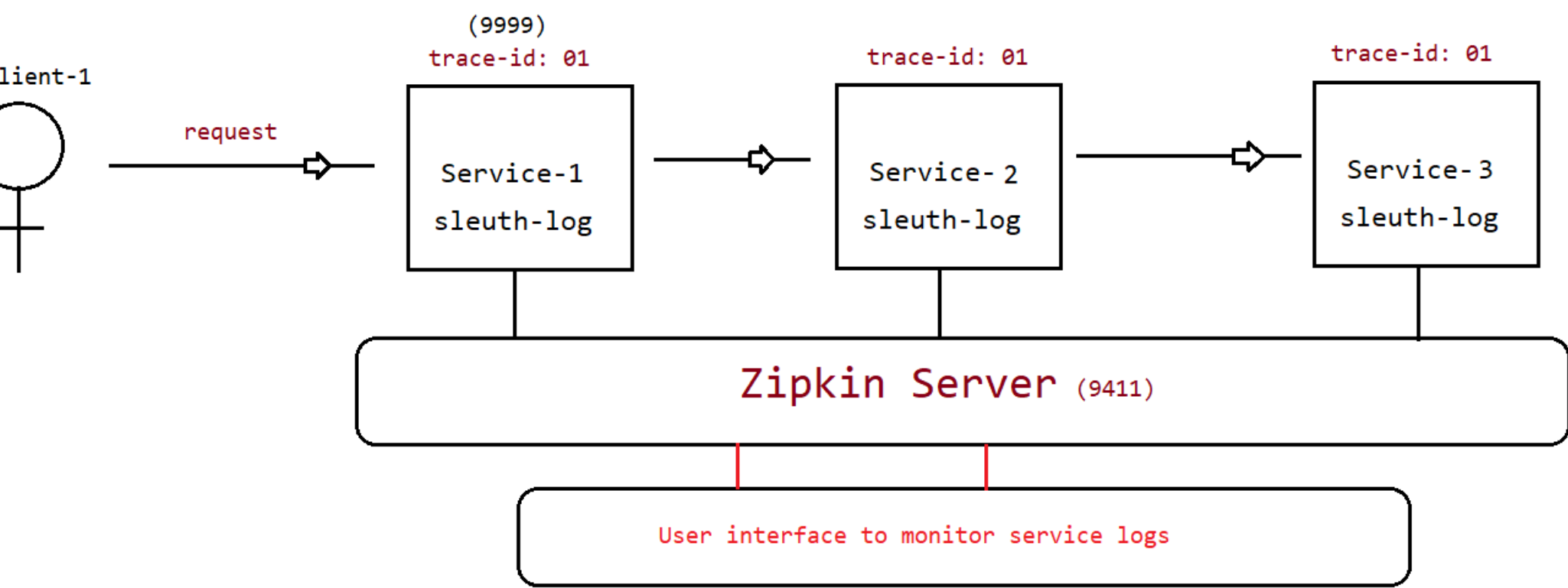
In the above Microservice architecture based project contains multiple microservices.

One request is processing by all services in our project and every service will have logging and it will generate the log file.

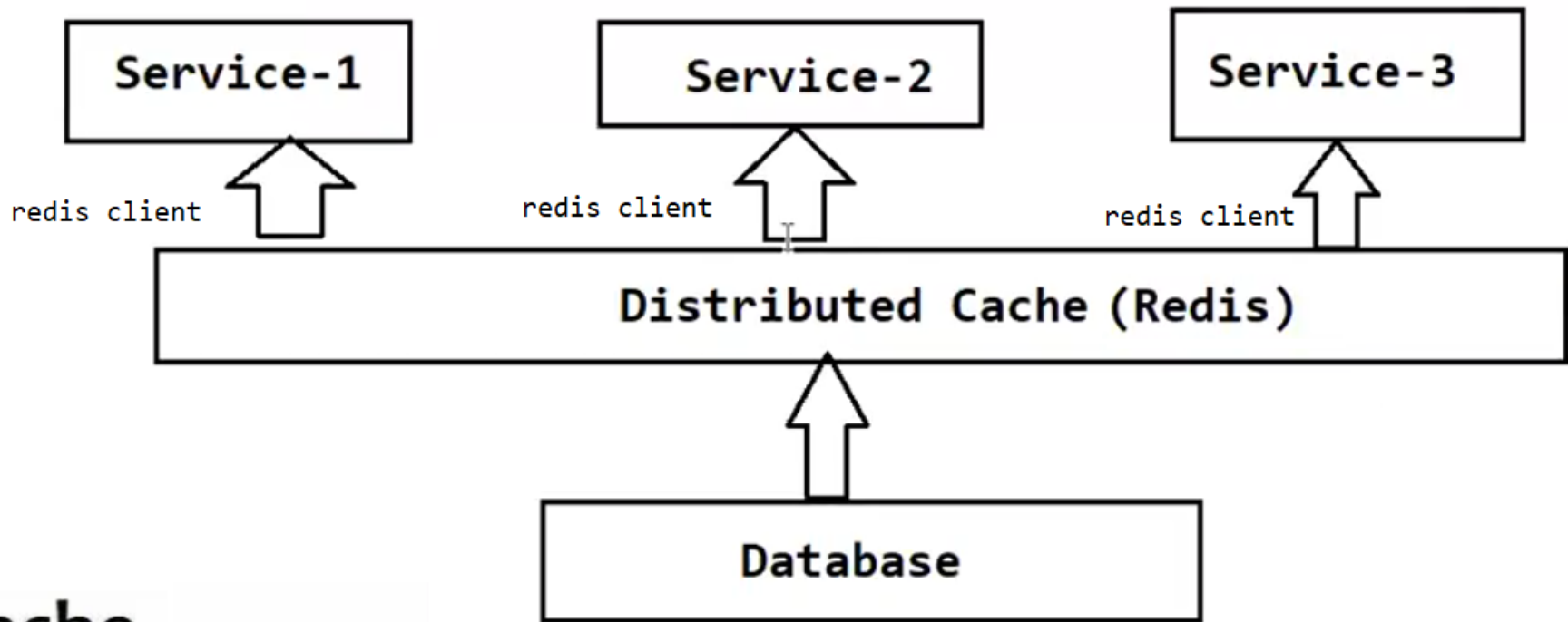
If request is failed or request is taking more processing time then which service is taking more time it is difficult to identify.

To simplify this process we use "Distributed logging".

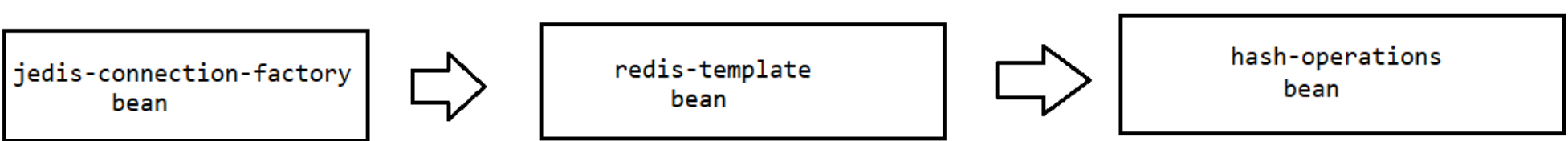
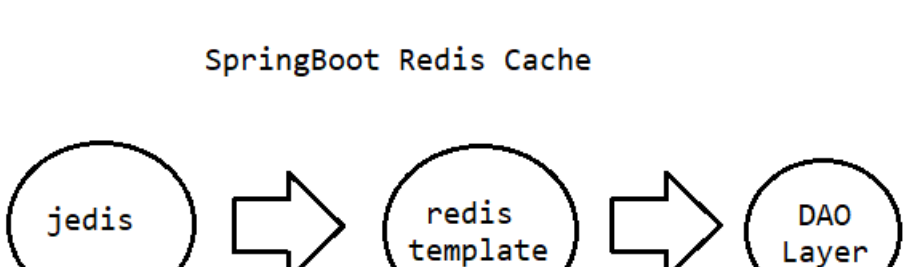
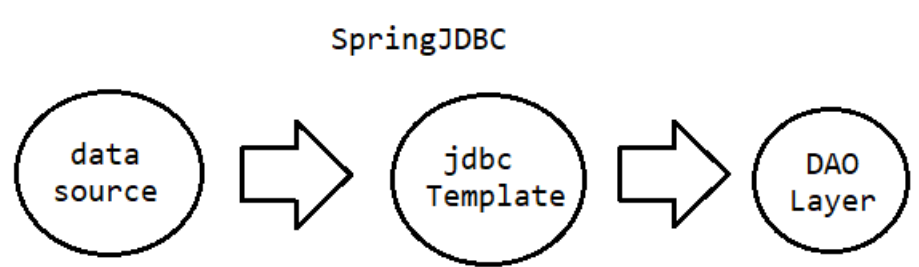
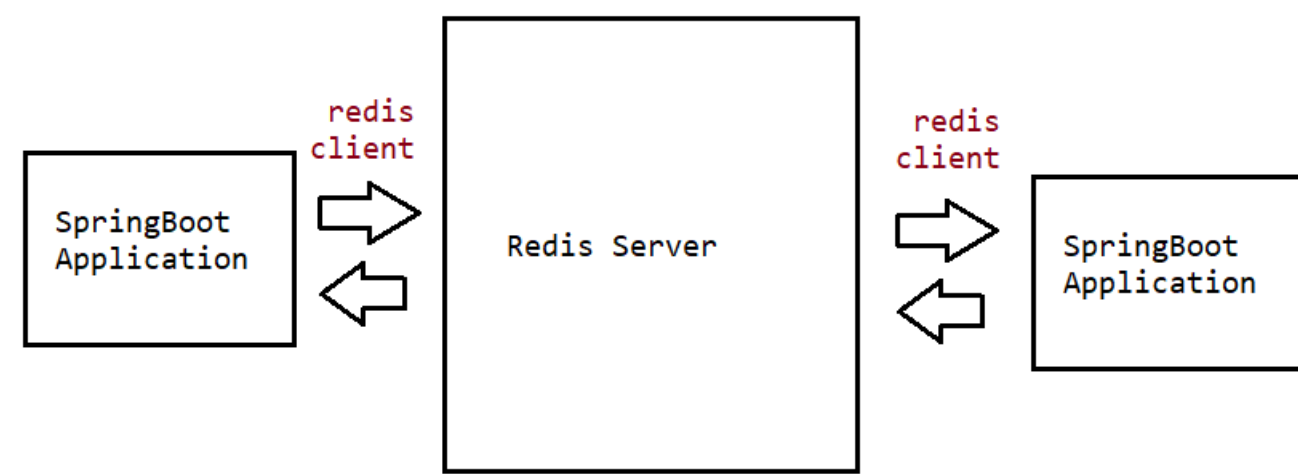
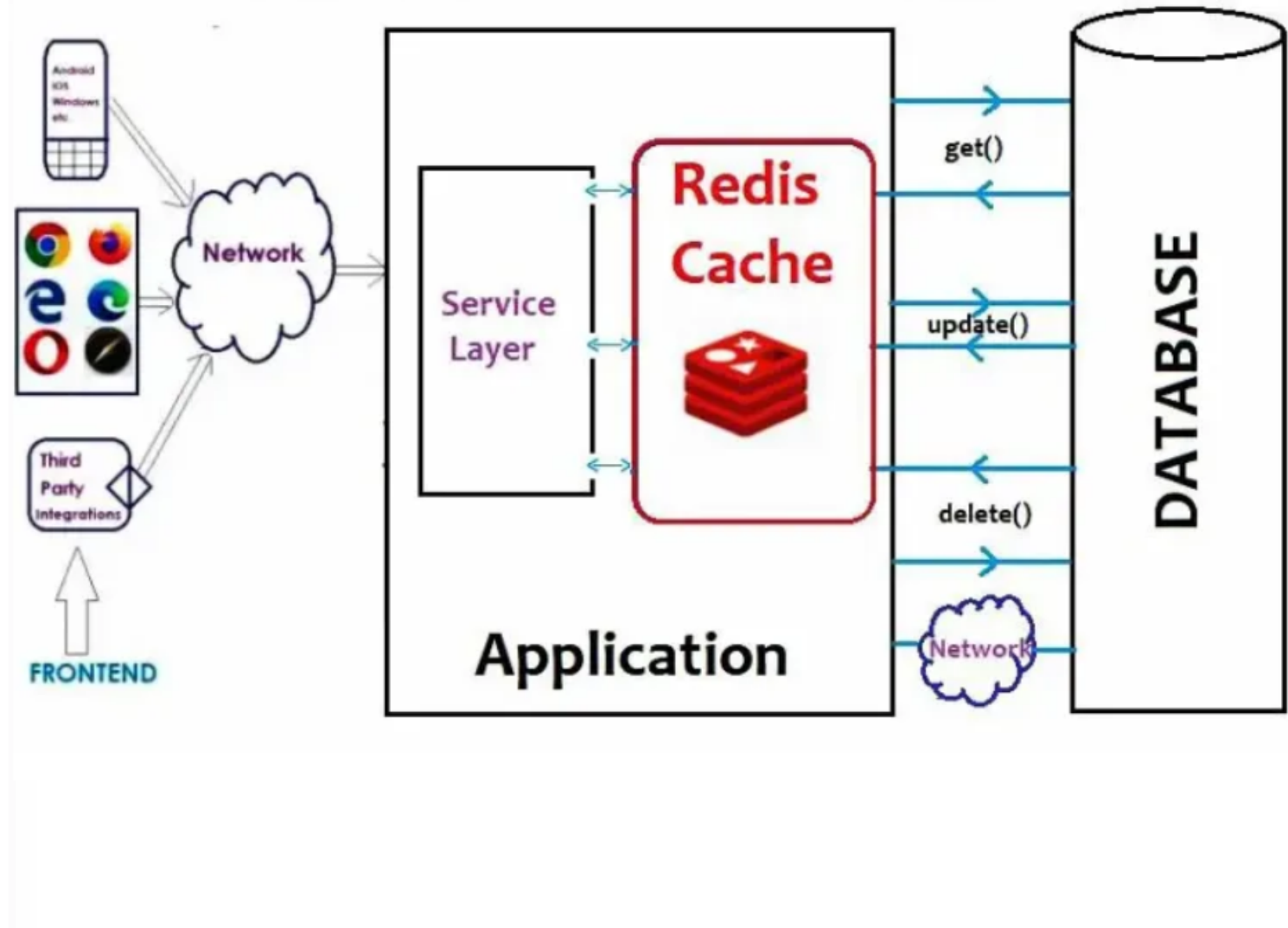
To perform Distributed logging we use "Sleuth and Zipkin".



Microservices Architecture Based Project



Caching with Redis Cache



```
put(String hk, key, value)
entries(String hk)
get(String hk, key)
delete(String hk, key)
```

methods to perform redis operation

Entities			
hashKey	USER	EMPS	Books
101-ABC	101-Nitin	101-J2SE	
102-XYZ	102-Haider	102-JEE	
103-DEF	103-Navin	103-Spring	