



THE
DEVELOPER'S
CONFERENCE

PREPARANDO NOSSAS APLICAÇÕES PARA FALHAREM COM FEATURE TOGGLE E CONFIGURAÇÕES DISTRIBUÍDAS

Trilha Java Enterprise

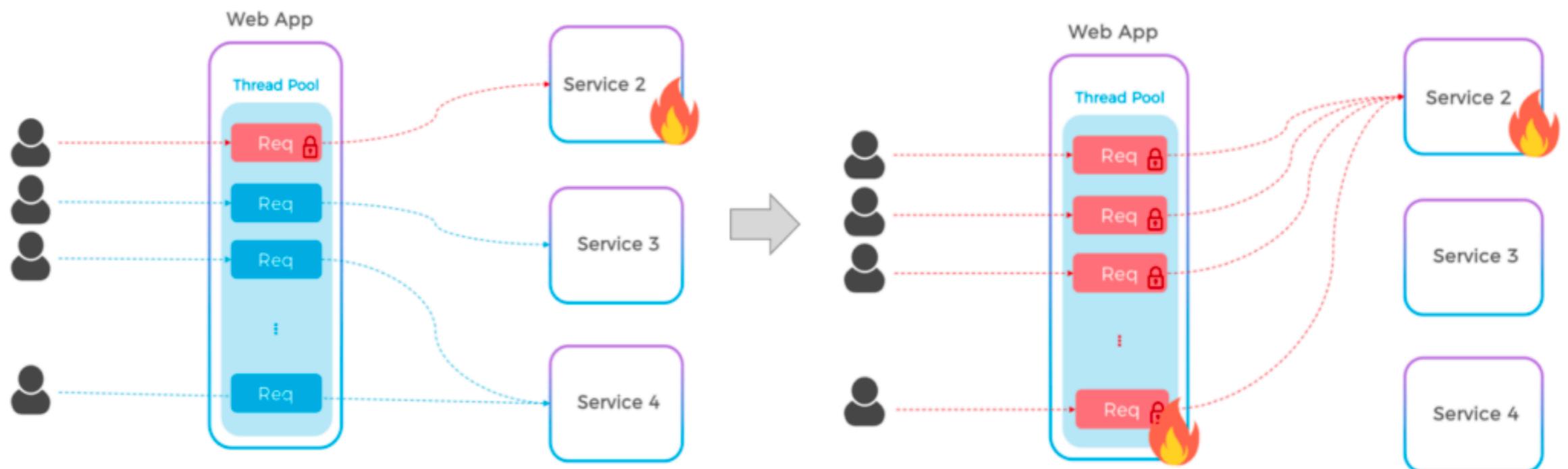
Emmanuel Neri

@emmanuelnerii

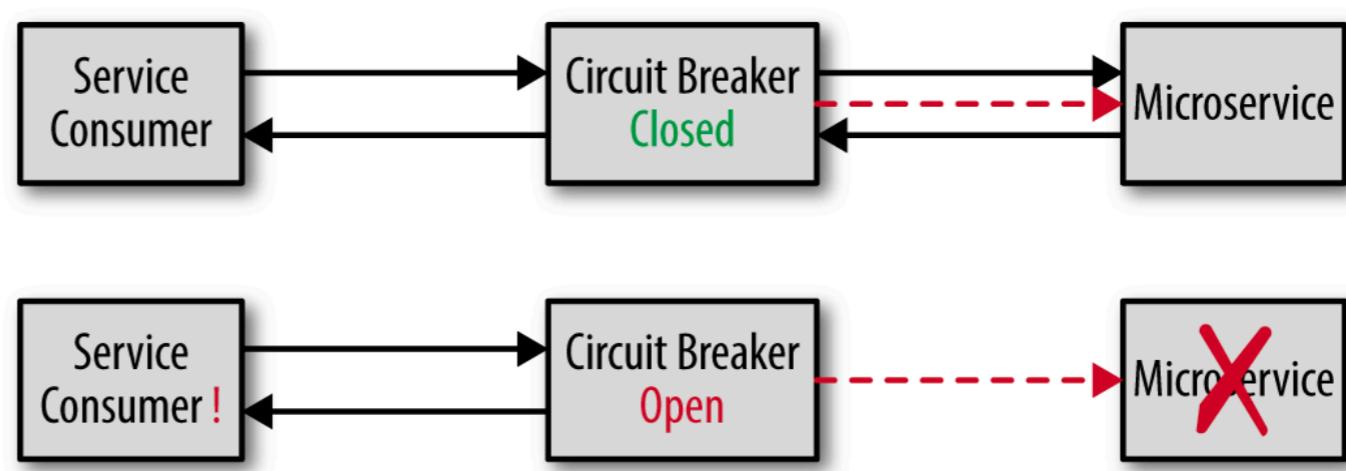
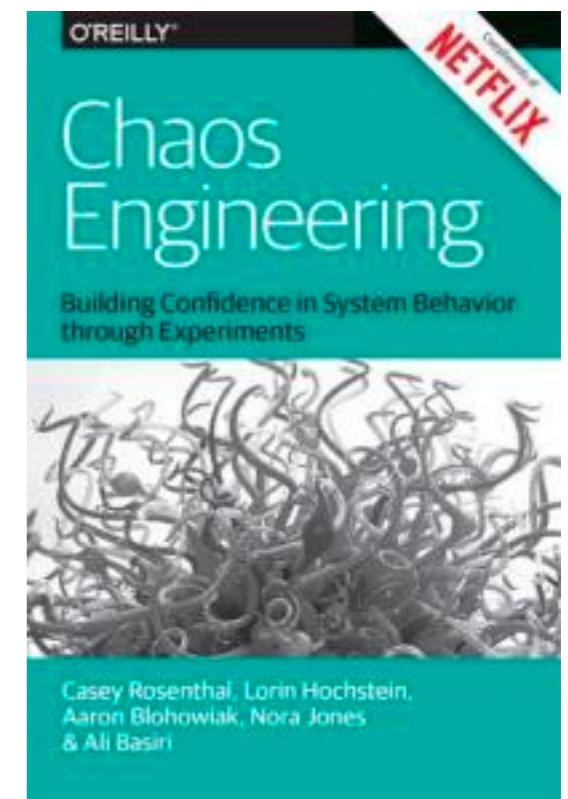
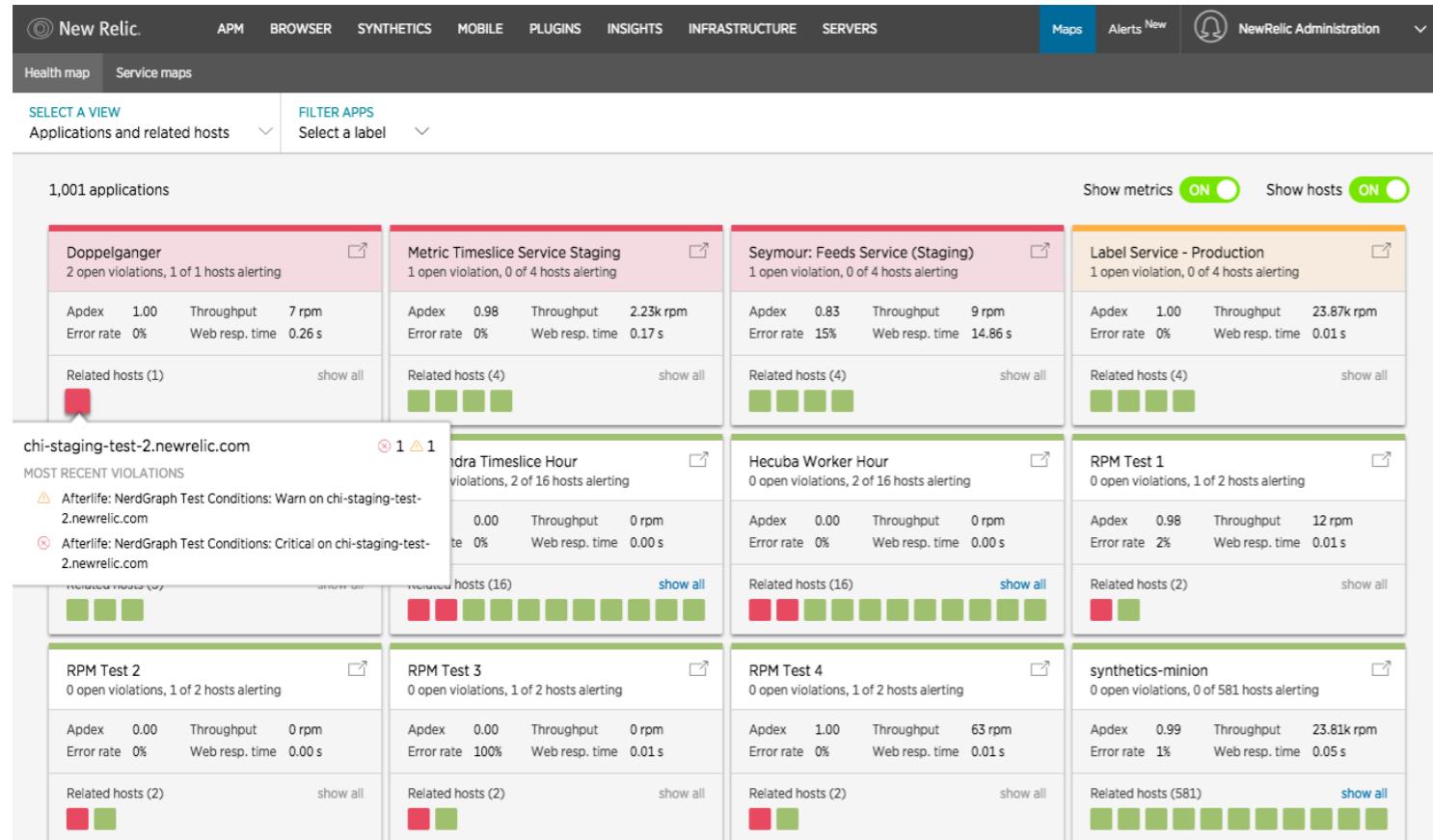


- ▶ Mestre em Desenvolvimento de Tecnologia
- ▶ Desenvolvedor (Java) desde 2010
- ▶ Software Developer na Jaya





POSSIBILIDADES



FALHAS LÓGICAS/NEGÓCIO

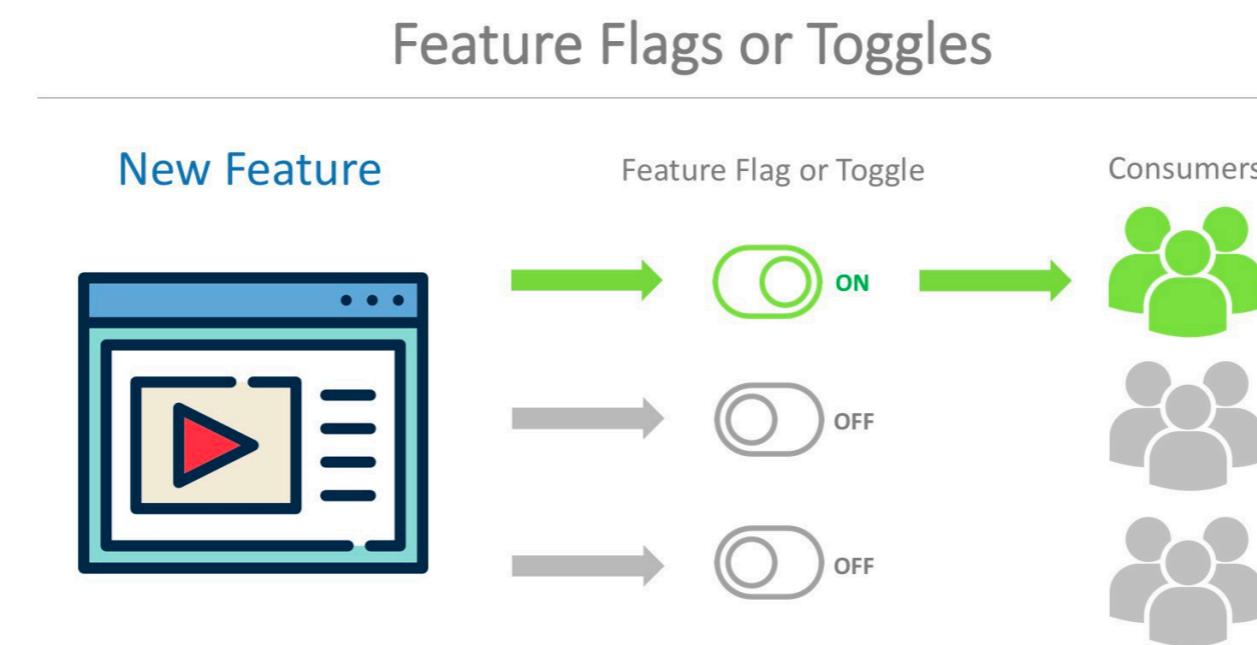


update Order set value = 0 where status in (:status)

valor.divide(quantidade, 2)

Feature Toggles (often also referred to as Feature Flags) are a powerful technique, allowing teams to modify system behavior without changing code. They fall into various usage categories, and it's important to take that categorization into account when implementing and managing toggles.

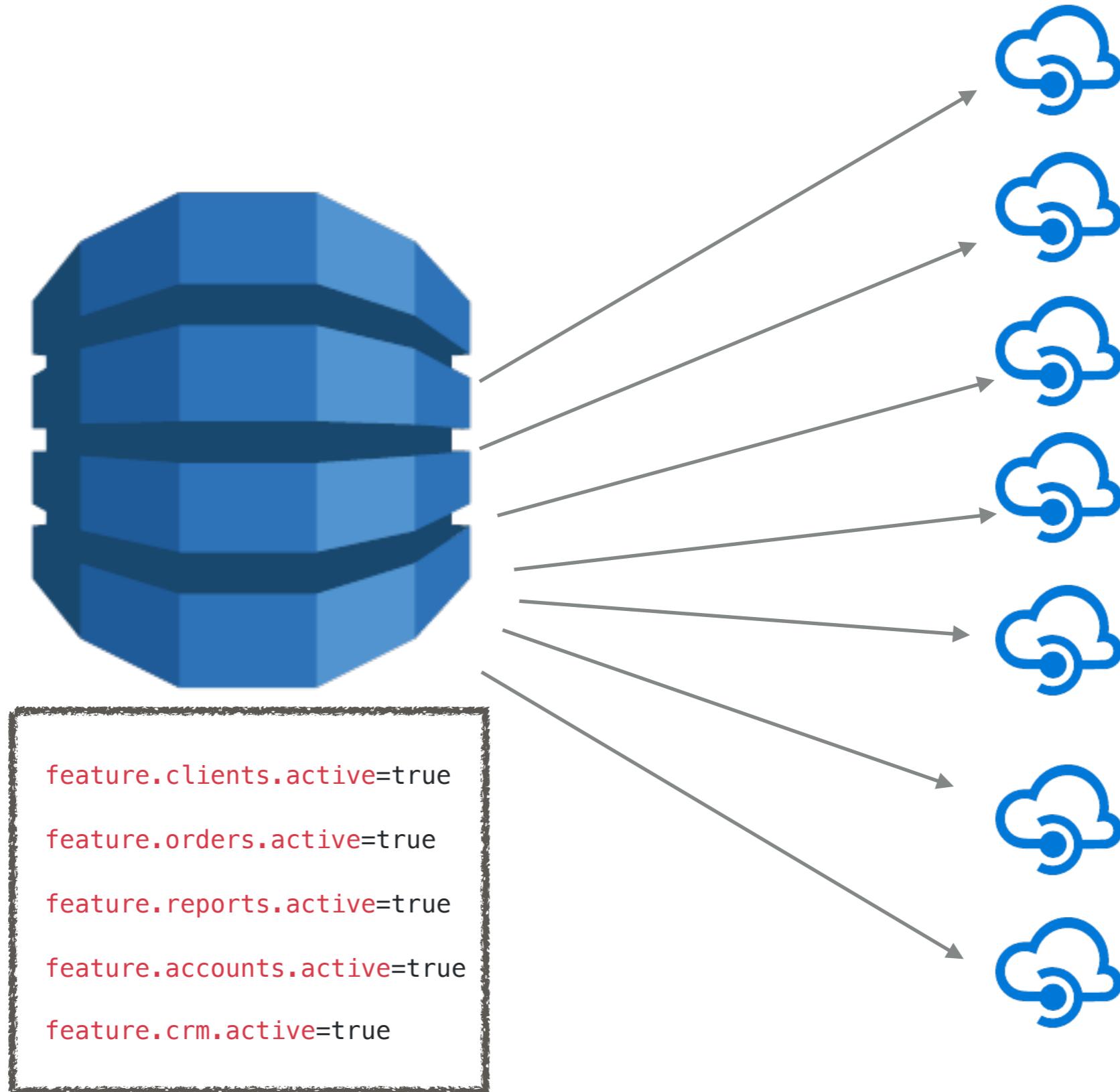
<https://martinfowler.com/articles/feature-toggles.html>



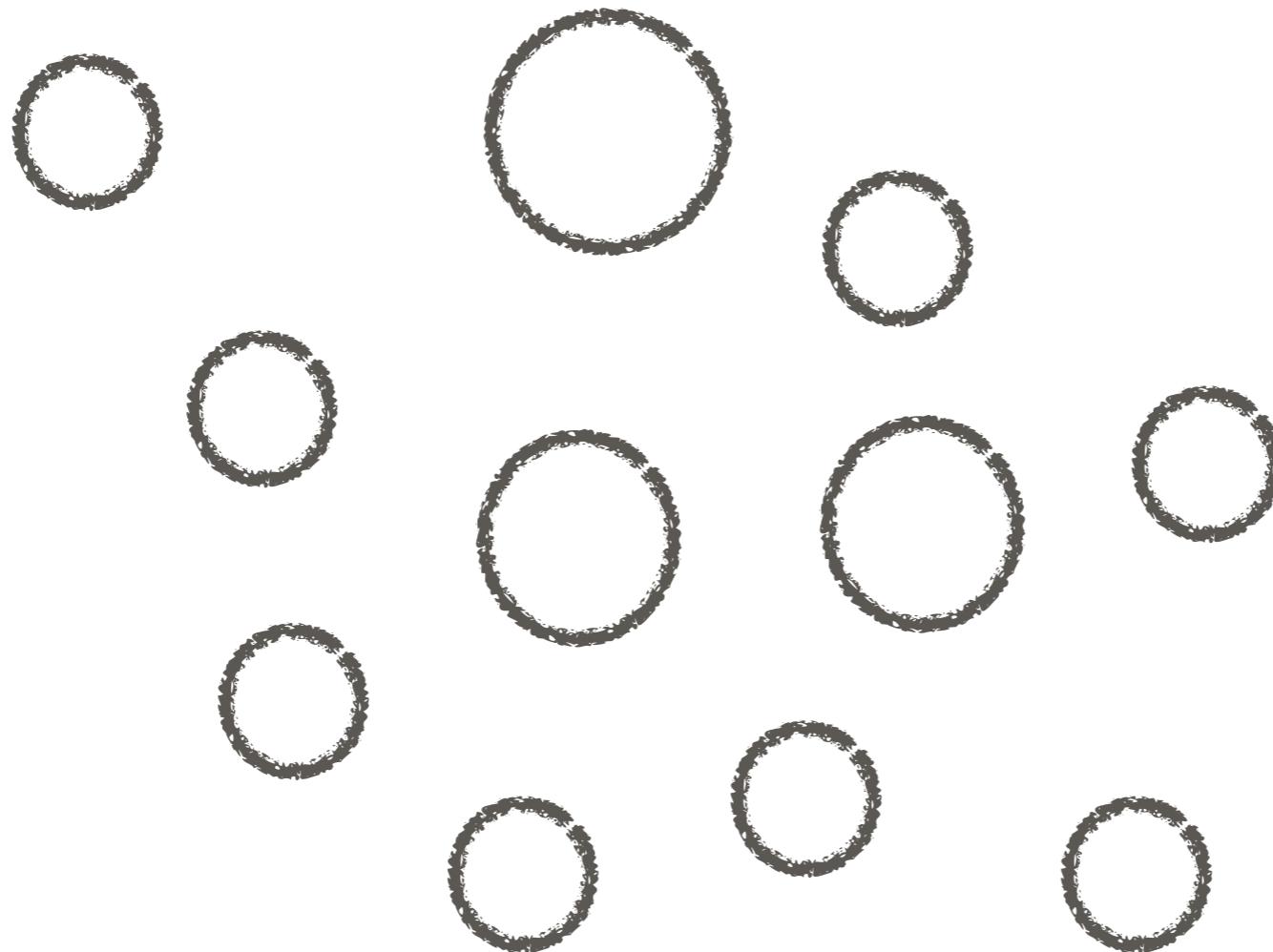
FEATURE TOGGLE

- ▶ Business Toggle
- ▶ Toggle Configuration
- ▶ Canary release
- ▶ A/B Testing
- ▶ Circuit Breaker
- ▶ Trunk based development
- ▶ Continuous Delivery

Feature	Status
X	
Y	
Z	

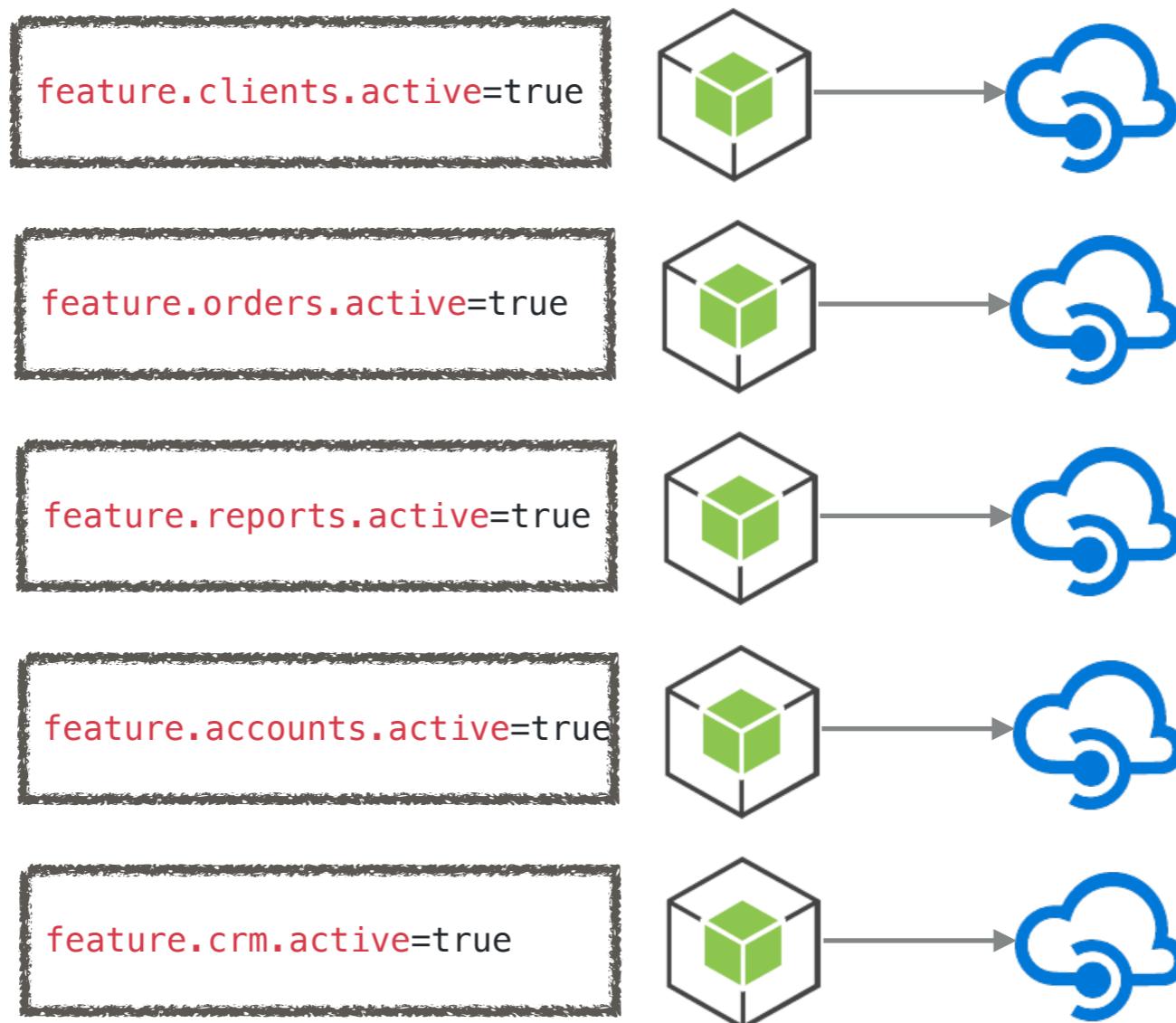


MICROSERVICES

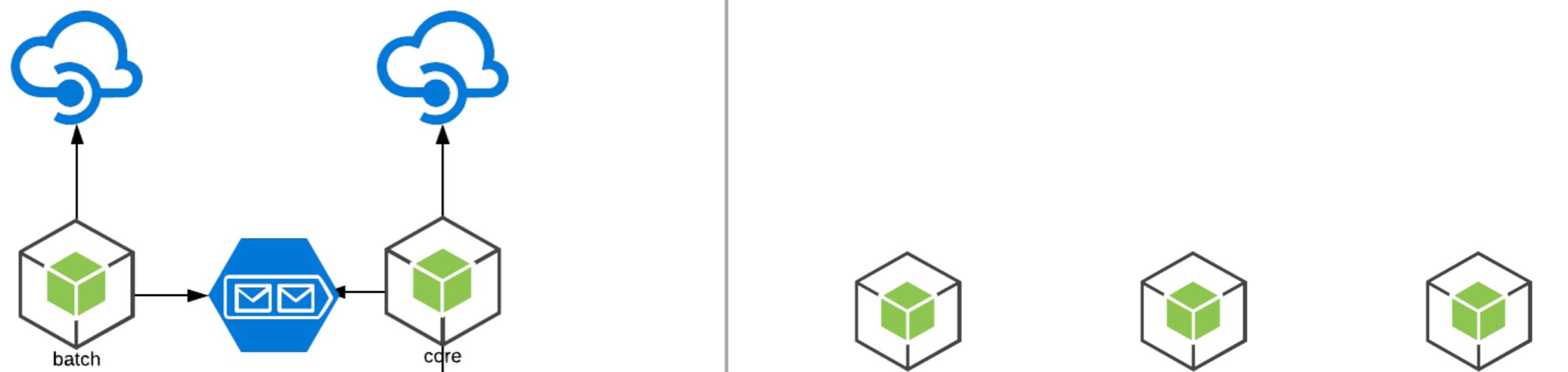


MICROSERVICES

MICROSERVICES



MICROSERVICES



CONFIGURAÇÕES EM MICROSERVICES




`spring.rabbitmq.host=localhost`
`spring.rabbitmq.port=5672`

`spring.data.mongodb.host=localhost`
`spring.data.mongodb.port=27017`




`spring.rabbitmq.host=localhost`
`spring.rabbitmq.port=5672`

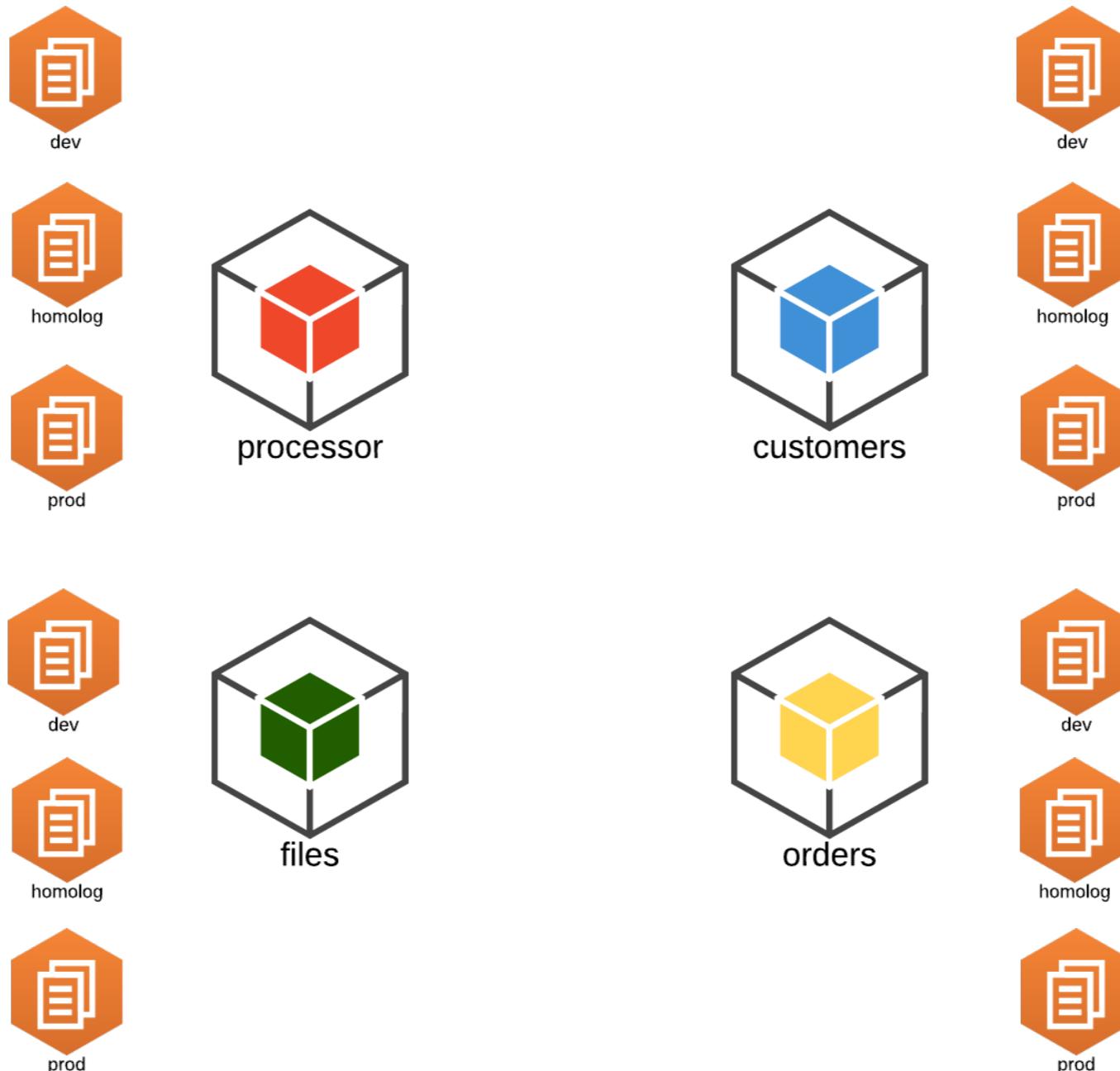
`spring.data.mongodb.host=localhost`
`spring.data.mongodb.port=27017`



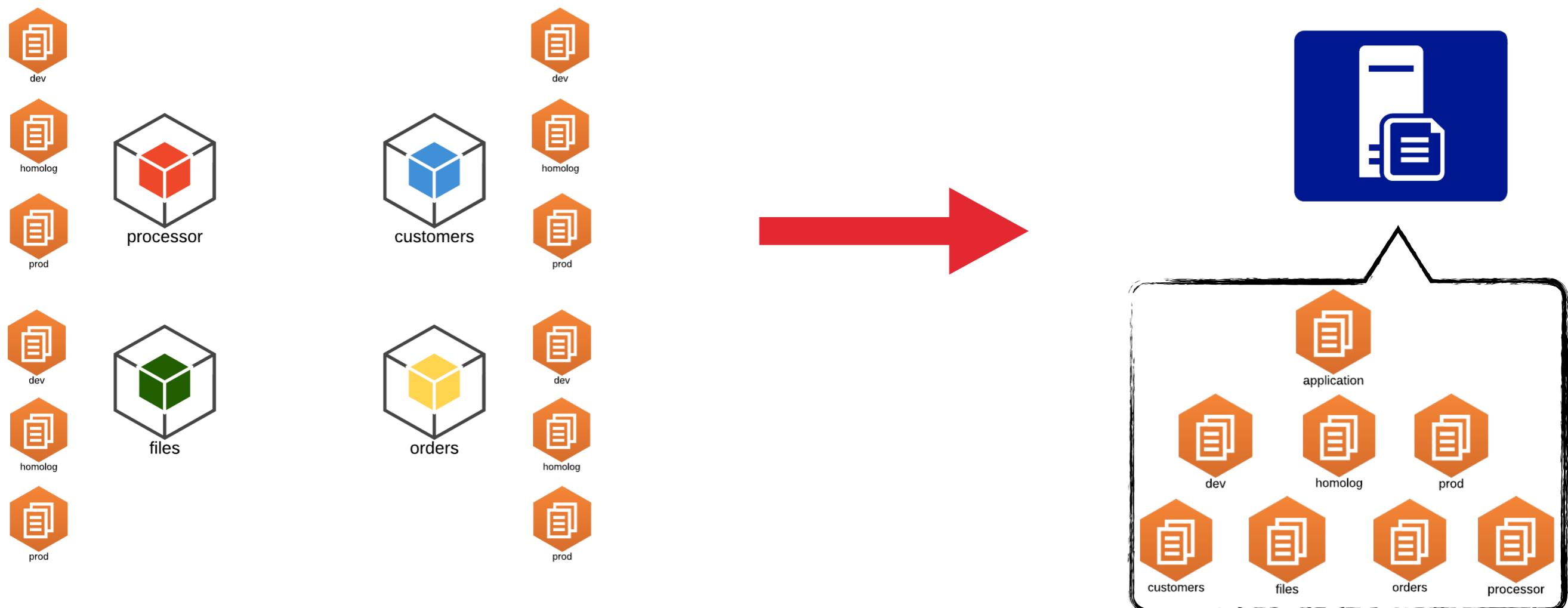

`spring.rabbitmq.host=localhost`
`spring.rabbitmq.port=5672`

`spring.data.mongodb.host=localhost`
`spring.data.mongodb.port=27017`

CONFIGURAÇÕES EM MICROSERVICES



CONFIGURAÇÕES DISTRIBUÍDAS



TECNOLOGIAS

- ▶ Netflix Archaius
- ▶ Consul
- ▶ Spring Cloud Config
- ▶ Configmap

SPRING CLOUD CONFIG

SPRING-CLOUD-CONFIG-SERVER



REPOSITÓRIO
CÓDIGO FONTE

FILE STORE

BANCO DE
DADOS

SPRING-CLOUD-CONFIG-CLIENT

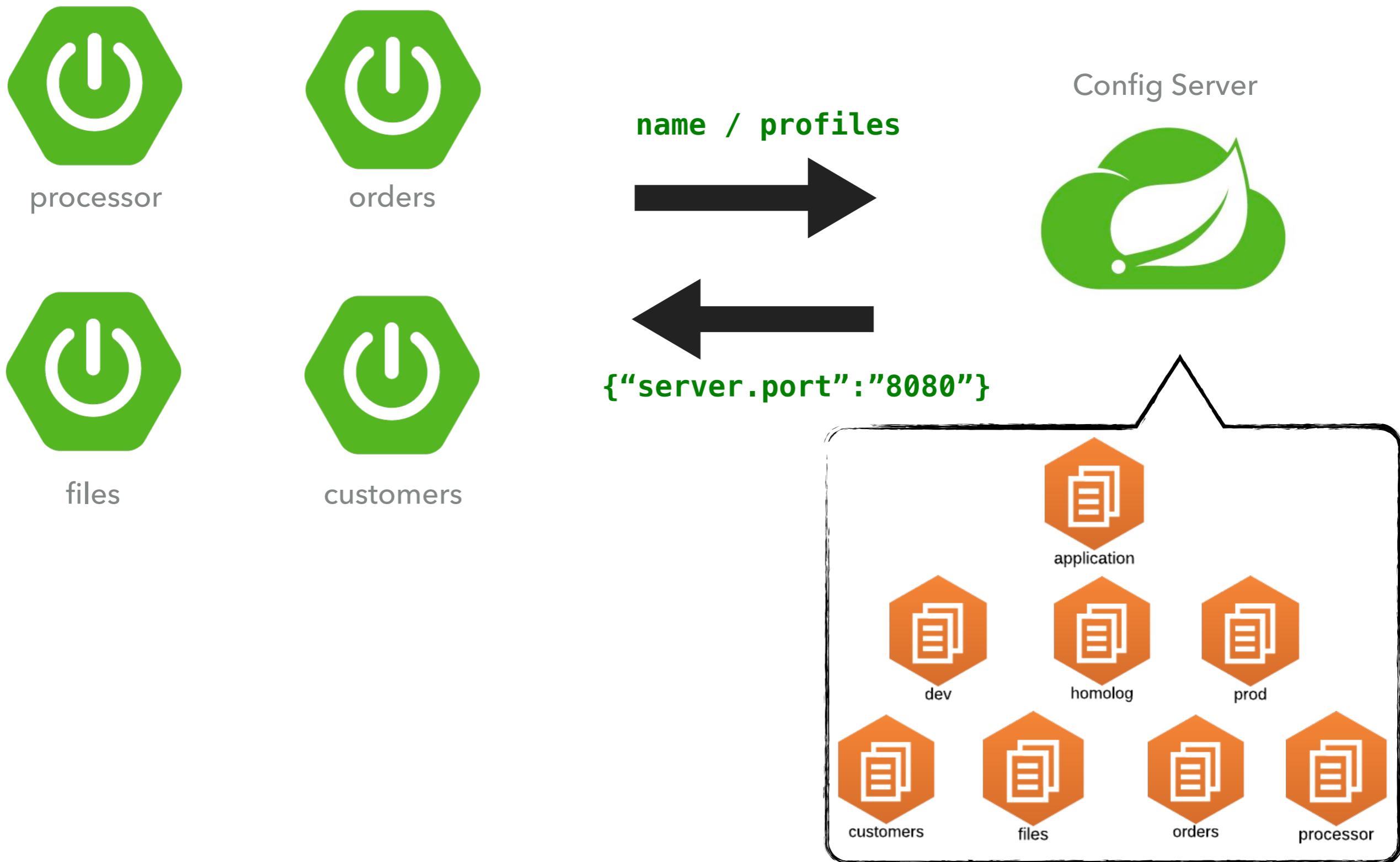
.JAR



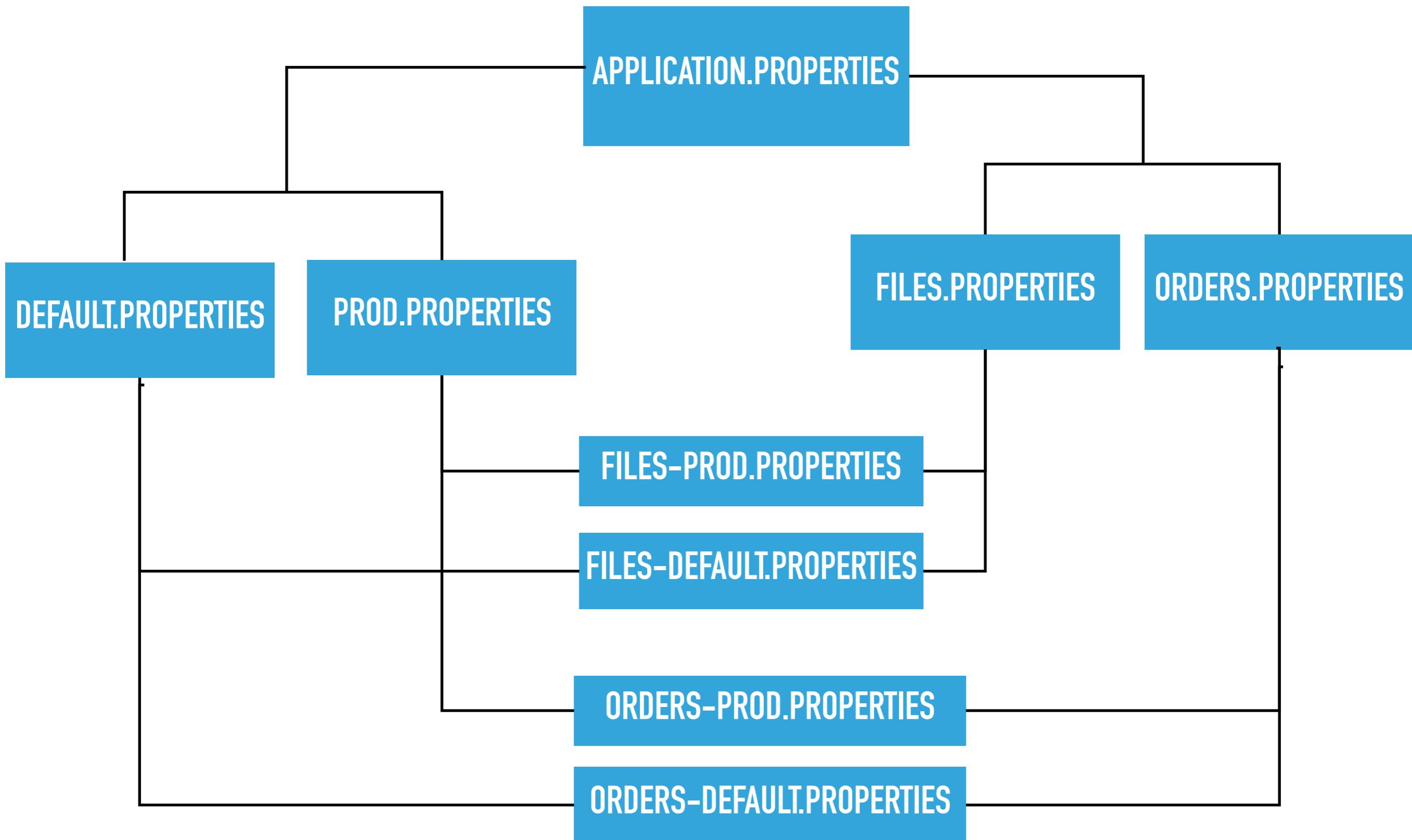
JDBC



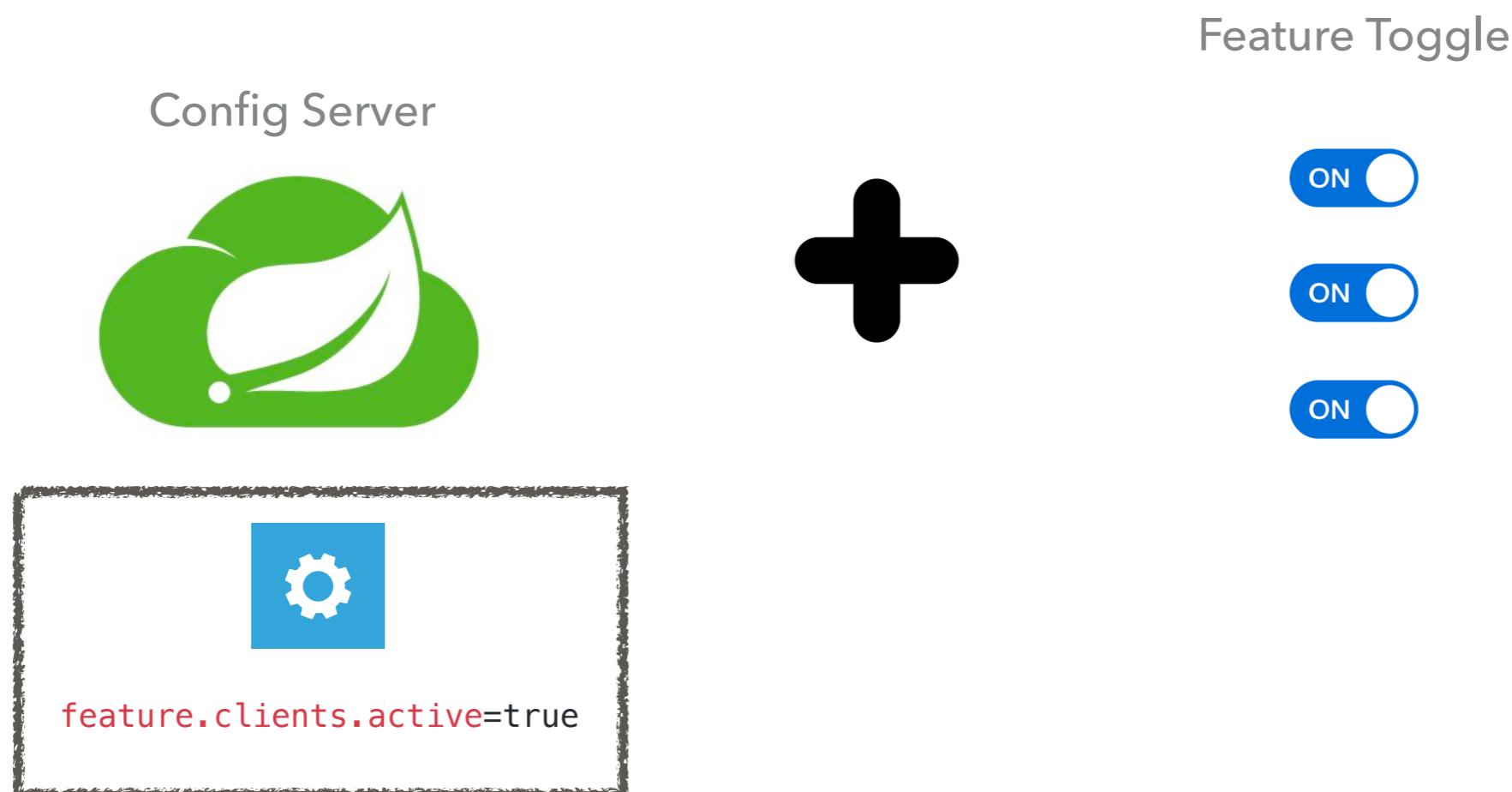
SPRING CLOUD CONFIG



HIERARQUIA DE CONFIGURAÇÕES



SPRING CLOUD CONFIG + FEATURE TOGGLE



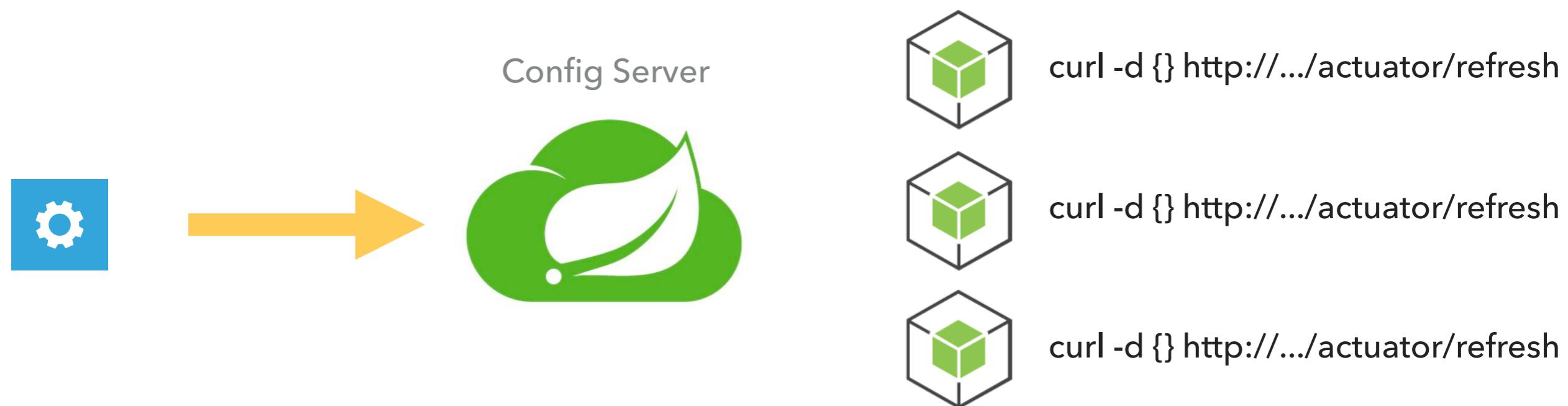
ATUALIZAÇÃO

```
@Component  
@ConfigurationProperties  
@RefreshScope  
public class RefreshProperties {  
  
    @Value("${feature.toggle.feature.hello.active}")  
    private boolean helloApiActive;  
  
}
```

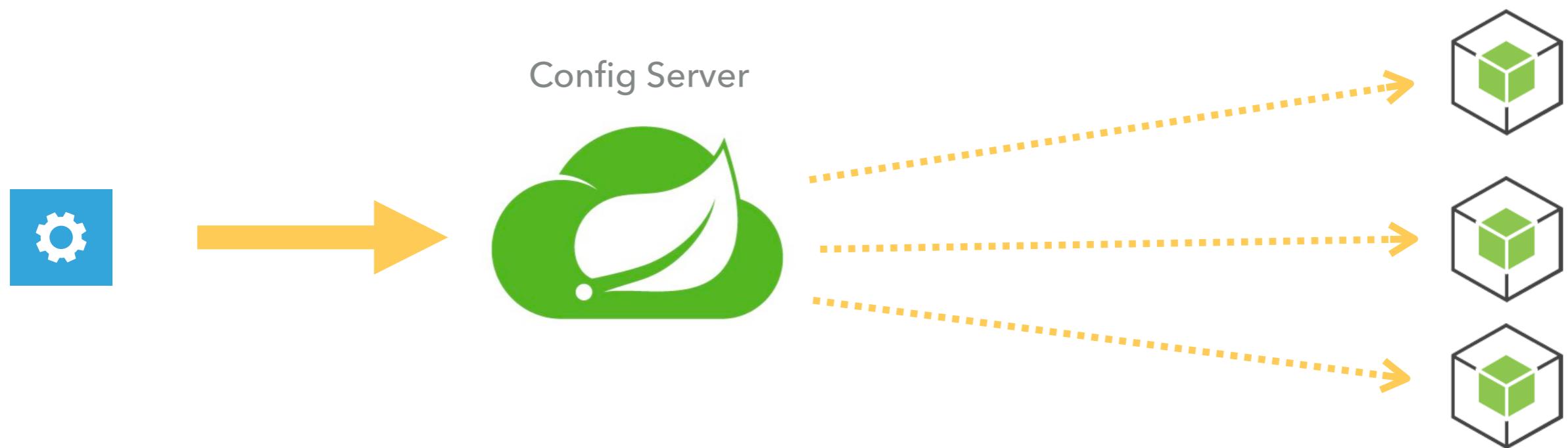
```
curl -d {} http://localhost:8090/actuator/refresh
```

```
@RestController  
@RequestMapping("/web")  
public class WebController {  
  
    @GetMapping(value = "/hello")  
    public String getHello() {  
        if(!refreshProperties.isHelloFeatureActive()) {  
            throw new InactiveFeatureException("hello inactive");  
        }  
  
        return "Web API";  
    }  
}
```

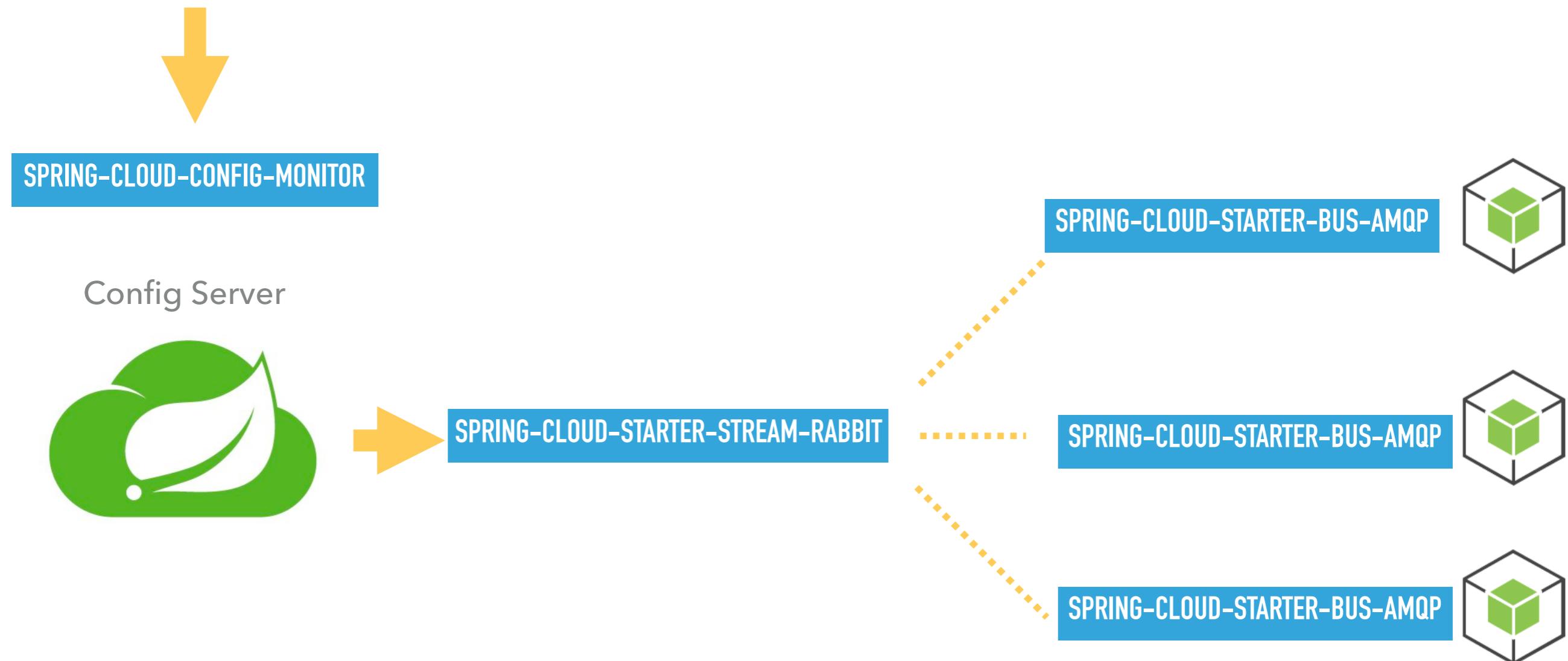
ATUALIZAÇÃO



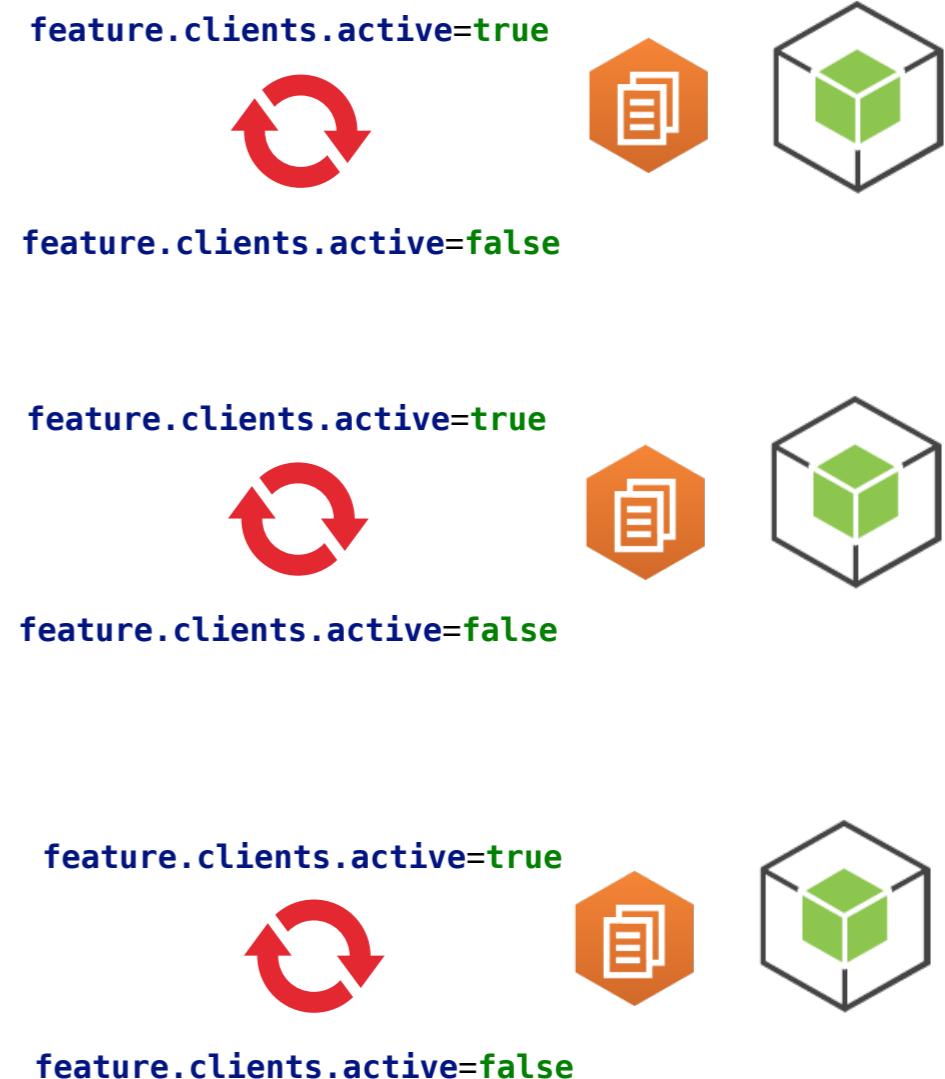
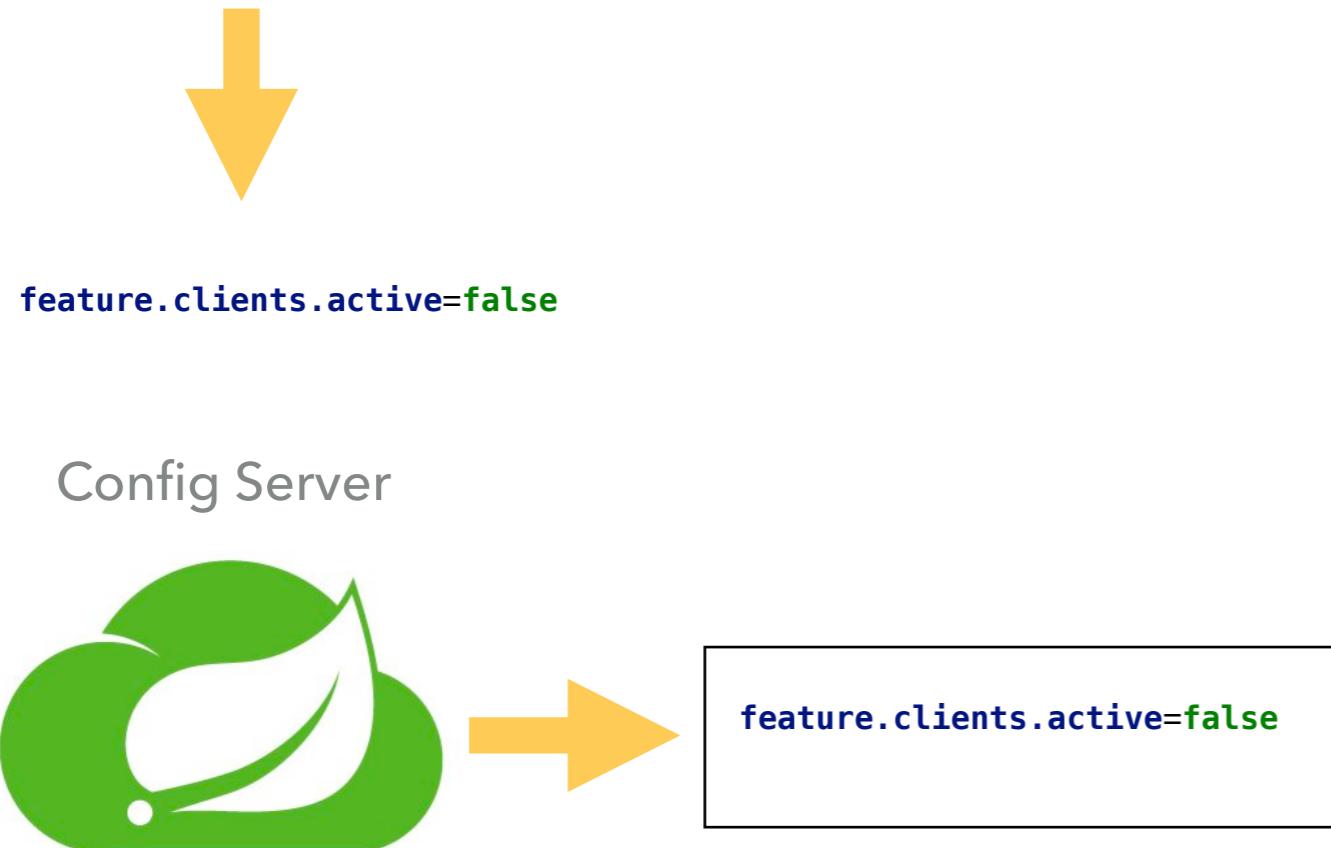
ATUALIZAÇÕES AUTOMÁTICAS



SPRING CLOUD STACK



ATUALIZAÇÕES AUTOMÁTICAS



ATUALIZAÇÕES AUTOMÁTICAS

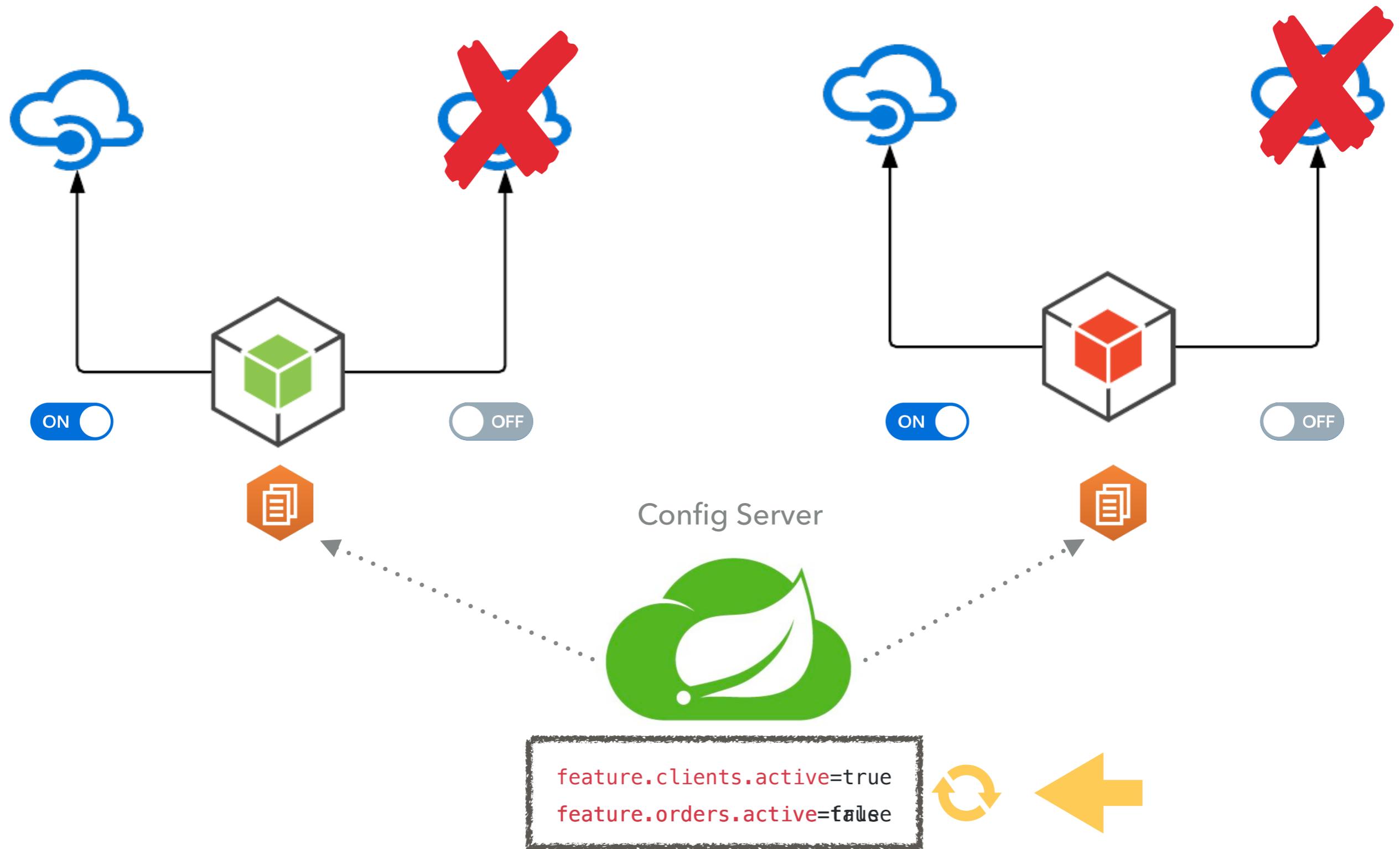


```
@GetMapping(value = "/hello")
@FeatureToggle(
    enableKey = "feature.toggle.feature.hello.active",
    nameKey = "feature.toggle.feature.hello.name")
public String getHello() {
    return "Hello Feature Toggle";
}
```

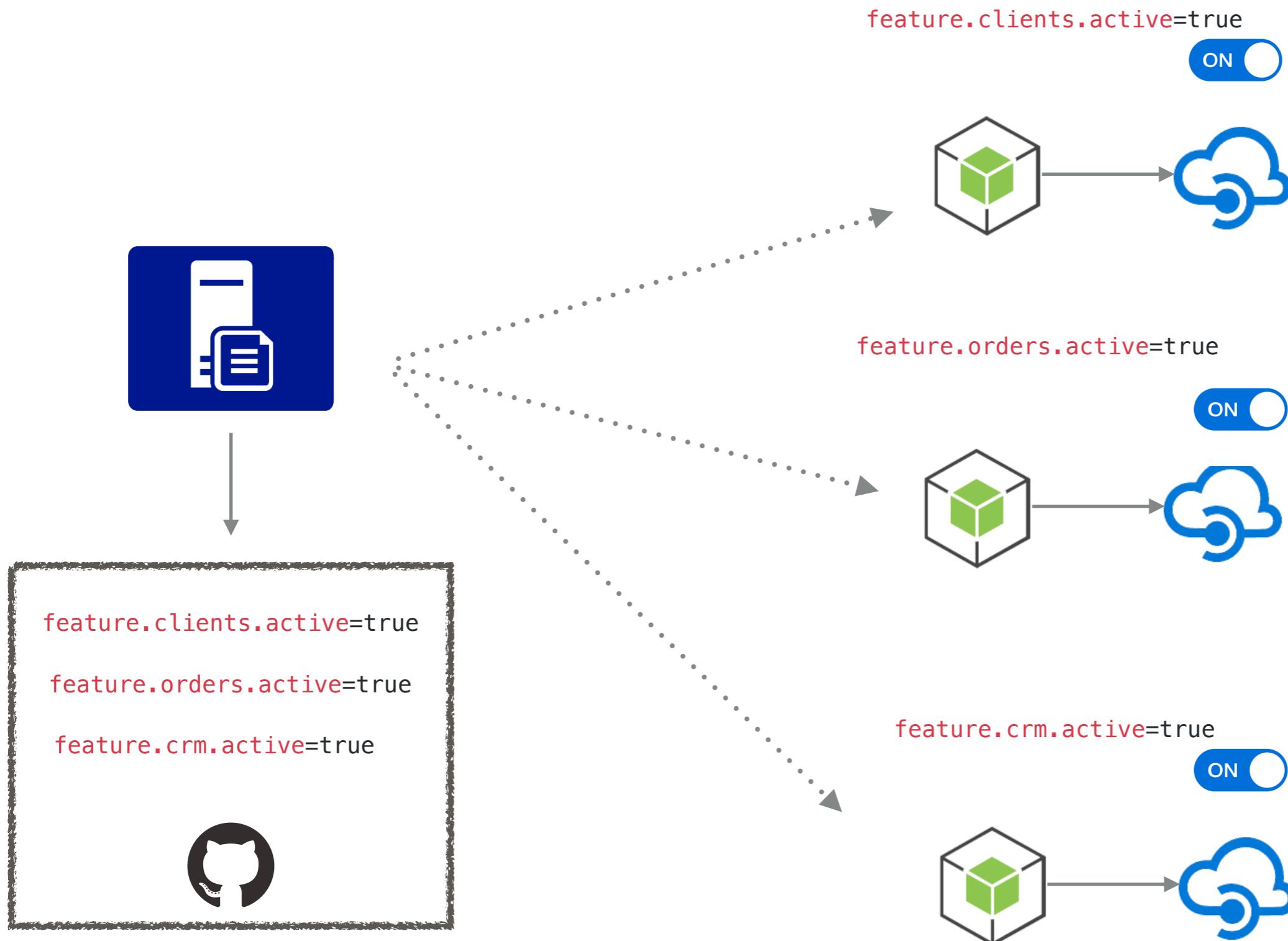
ConfigServer/application.properties

```
feature.toggle.feature.hello.name=Hello
feature.toggle.feature.hello.active=true
```

PREPARANDO



FEATURE TOGGLE + CONFIGURAÇÕES DISTRIBUÍDAS



FRAMEWORKS FEATURE TOGGLES



- ▶ Strategy-based Toggling
- ▶ Role-based Toggling
- ▶ Auditing
- ▶ Database Support
- ▶ Web console

<https://github.com/emmanuelneri/fault-tolerant-features>

OBRIGADO!



@emmanuelnerii



www.linkedin.com/in/emmanuelnerisouza



emmanuelnerisouza@gmail.com



github.com/emmanuelneri



www.emmanuelneri.com.br