

Spring-Cloud 个人理解

作者: [goker](#)

原文链接: <https://ld246.com/article/1606719915542>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

Spring-Cloud等组件个人理解

微服务的理解

将资源业务进行拆分

拆分的依据

1. 那些业务访问次数多，流量大。
2. 不是越细越好，越细服务之间的调用次数太多，响应时延过多，也会影响系统，用户体验。有些无关紧要的业务可以不拆分。

Spring-Cloud分布式五大组件

Eureka-服务发现

Hystrix-熔断器

Zull-路由网管

Ribbon-负载均衡

Open-Fegin-远程调用

一. Eureka 注册中心

服务发现，服务管理

provider 生产者

consumer 消费者

以服务名注册

二. OpenFegin

自定义接口，远程调用：常用方法

Consumer: (消费者)

```
@FeginClient("PROVIDER")
public interface UcenterClient{
    @GetMapping("/ucenter/getUserVo")
    public WlzxUserVo getWlzxUserVo(@RequestParam("username") String username)
}
```

Controller层

@AutoWired

private UcenterClient ucenterClient;

远程调用（接口名.抽象方法）

ucenterClient.getWlzxUserVo("xxx");

三. 自我保护机制

服务中在运行时，会在15分钟内统计各个心跳失联，如果比例大于15%，就会开启自我保护机制，会这些心跳失联的节点是因为网络问题，当各个节点恢复心跳，达到比例就会退出自我保护机制。

心跳机制：各个节点启动后，每30秒会向Server发送心跳。

四. 熔断器

服务相互调用之间，出现服务雪崩， $A \implies B \implies C$ ，其中一个关联服务挂了，导致后面服务不能使。

机制：当服务宕机，调用失败达到一个阈值，5秒失败20次，链路就会出现open状态。

在此提供一个熔断器，Fallback（实现FallbackFactory<Client>接口，重写create()方法）。

五. 路由网关

统一入口，所有请求服务都从网关调用，做到解耦。

过滤器能识别过滤些信息，权限。