



链滴

# springcloud feign 远程调用

作者: [ws](#)

原文链接: <https://ld246.com/article/1536918118155>

来源网站: [链滴](#)

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

## 项目清单

Feign需要注册到eureka服务中,下面分别建立四个项目

- eureka-server (spring-cloud-rureka-server-feign)    name:eureka-server-feign    port:090
- parent -feign[父项目] -(spring-cloud-feign) -- 以下项目采用聚合工程
  - feign-api[契约服务]                                    name:feign-api                                    port:8081
  - feign-consumer[服务消费端]                            name:person-consumer                            port:8082
  - feign-provider[服务提供方]                            name:person-provider                            port:8083

## Eureka Server项目

### 1.Maven导入

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-actuator</artifactId>
</dependency>
<dependency>
  <groupId>org.springframework.cloud</groupId>
  <artifactId>spring-cloud-starter-netflix-eureka-server</artifactId>
</dependency>
```

### 2.Application.java增加注解

```
@SpringBootApplication
@EnableEurekaServer //开启eureka server
public class SpringCloudRurekaServerFeignApplication {

    public static void main(String[] args) {
        SpringApplication.run(SpringCloudRurekaServerFeignApplication.class, args);
    }
}
```

### 3.修改application.properties

```
server.port = 9090
spring.application.name = eureka-server-feign
eureka.client.fetch-registry = false
eureka.client.register-with-eureka = false
# 开启全部actuator管理
management.endpoints.web.exposure.include=*
```

## Parent : Feign 项目(spring-cloud-feign)

### 1.Maven导入

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-actuator</artifactId>
</dependency>
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-web</artifactId>
</dependency>
<dependency>
  <groupId>org.springframework.cloud</groupId>
  <artifactId>spring-cloud-starter-netflix-eureka-client</artifactId>
</dependency>

<dependency>
  <groupId>org.springframework.cloud</groupId>
  <artifactId>spring-cloud-starter-openfeign</artifactId>
</dependency>
```

## 2.修改<packaging> pom</packaging>

### 新建Module : feign-api

#### 1.修改pom

```
<parent>
  <groupId>com.blaaaair</groupId>
  <artifactId>spring-cloud-feign</artifactId>
  <version>0.0.1-SNAPSHOT</version>
</parent>
```

#### 2.Application.java配置

```
@SpringBootApplication
@EnableEurekaClient
public class SpringCloudFeignApiApplication {

    public static void main(String[] args) {
        SpringApplication.run(SpringCloudFeignApiApplication.class, args);
    }
}
```

#### 3.application.properties配置

```
server.port = 8081
spring.application.name = feign-api
eureka.client.service-url.defaultZone = http://localhost:9090/eureka
management.endpoints.web.exposure.include=*
```

#### 4.增加接口

```
@FeignClient("person-provider") //person-provider : 接口提供方应用名称
public interface PersonService {

    @PostMapping("/person/save")
    boolean save(@RequestBody Person person);
}
```

## 新建Module :feign-consumer

### 1.修改pom

```
<parent>
  <groupId>com.blaair</groupId>
  <artifactId>spring-cloud-feign</artifactId>
  <version>0.0.1-SNAPSHOT</version>
</parent>
```

需在依赖中增加feign-api

### 2.Application.java配置

```
@SpringBootApplication
@EnableEurekaClient
@EnableFeignClients(clients = PersonService.class) //接受暴露出的
public class SpringCloudFeignApiApplication {

    public static void main(String[] args) {
        SpringApplication.run(SpringCloudFeignApiApplication.class, args);
    }
}
```

### 3.application.properties配置

```
server.port = 8082
spring.application.name = person-consumer
eureka.client.service-url.defaultZone = http://localhost:9090/eureka
management.endpoints.web.exposure.include=*
```

### 4.增加调用远程service的controller

```
@RestController
public class PersonController implements PersonService {

    private final PersonService personService;

    @Autowired
    public PersonController(PersonService personService) {
        this.personService = personService;
    }

    @Override
```

```
    public boolean save(Person person) {
        return personService.save(person);
    }
}
```

## 新建Module : feign-provider

### 1.修改pom

```
<parent>
  <groupId>com.blaair</groupId>
  <artifactId>spring-cloud-feign</artifactId>
  <version>0.0.1-SNAPSHOT</version>
</parent>
```

需在依赖中增加feign-api

### 2.Application.java配置

```
@SpringBootApplication
@EnableEurekaClient
public class SpringCloudFeignApiApplication {

    public static void main(String[] args) {
        SpringApplication.run(SpringCloudFeignApiApplication.class, args);
    }
}
```

### 3.application.properties配置

```
server.port = 8083
# 这里的name 与feign-api中@FeignClient("person-provider")对应
spring.application.name = person-provider
eureka.client.service-url.defaultZone = http://localhost:9090/eureka
management.endpoints.web.exposure.include=*
```

### 4.提供的调用controller

```
@RestController
public class PersonController {

    @PostMapping("/person/save")
    boolean save(@RequestBody Person person){
        return new HashMap<>().put(person.getId(),person)!=null;
    }
}
```

## 测试

1. 分别启动项目:eureka-server - > feign-provider - > feign-api - > feign-consumer

2. POSTMAN : localhost:8082/person/save ,

- raw:

```
{  
  id:1,  
  name:"ws"  
}
```

3. 返回:true则表示成功!

4. 调用顺序 : postman --> consumer --> provider