



链滴


# 微服务学习——Eureka

作者: [strangebob](#)

原文链接: <https://ld246.com/article/1681376421742>

来源网站: [链滴](#)

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

 <https://b3logfile.com/bing/20211115.jpg?imageView2/1/w/960/h/540/interlace/1/q/100>

## 1、基础知识介绍

**提供者与消费者**

服务提供者：一次业务中，被其他微服务调用的服务（提供接口给其他服务）

服务消费者：一次业务中，调用其他微服务的服务（调用其他微服务提供的接口）


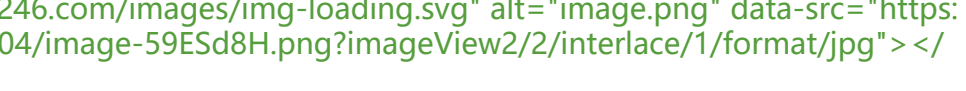
**问题**：服务 A 调用服务 B，服务 B 调用服务 C，那么服务 B 是什么角色？

一个服务既可以是消费者也可以是服务者，具体业务具体分析，抛开业务谈都是扯淡

## 2、Eureka 注册中心

### 2.1 Eureka 原理分析

**原理图**

**分析原理**

● 消费者该如何获取服务提供者具体信息？

■ 服务提供者启动时向 eureka 注册自己的信息

■ eureka 保存这些信息

■ 消费者根据服务名称向 eureka 拉取提供者信息

● 如果有多个服务提供者，消费者该如何选择？

■ 服务消费者利用负载均衡算法，从服务列表中挑选一个

● 消费者如何感知服务提供者健康状态？


■ 服务提供者会每隔 30 秒向 EurekaServer 发送心跳请求，报告健康状态

■ eureka 会更新记录服务列表信息，心跳不正常会被剔除

■ 消费者就可以拉取到最新的信息



**总结**

### 2.2 搭建 Eureka 注册中心

首先注册中心服务端：eureka-server，这必须是一个独立的微服务

#### 2.2.1 创建 eureka-server 服务

然后在新创建的服务里面引入 eureka 依赖

#### 2.2.2 引入 eureka 依赖

```
<dependency>
```

```
<groupId>org.springframework.cloud
```

```
<artifactId>spring-cloud-starter-netflix-eureka-server
```

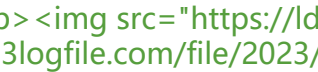
```
</dependency>
```

```
</pre>
```

maven 记得配置阿里云镜像，下载的会更快

### 2.2.3 编写启动类

先创建一个名为 EurekaApplication 的 Java Class 文件

 <https://ld246.com/images/img-loading.svg> alt="image.png" data-bbox="300 86 500 118"/> <https://b3logfile.com/file/2023/04/image-r7q9Yc1.png?imageView2/2/interlace/1/format/jpg>

给 eureka-server 服务编写一个启动类，一定要添加一个 @EnableEurekaServer 注解，开启 eureka 的注册中心功能：

```
package cn.itcast.eureka;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.cloud.netflix.eureka.server.EnableEurekaServer;

@SpringBootApplication
@EnableEurekaServer

public class EurekaApplication {

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



### 2.2.4. 编写配置文件

编写一个 application.yml 文件，内容如下：

```
server:
  port: 10086
```

```
</span></span></span><span class="highlight-line"><span class="highlight-cl"><span cla
s="highlight-w"></span><span class="highlight-nt">spring</span><span class="highlight
p">:</span><span class="highlight-w">
</span></span></span><span class="highlight-line"><span class="highlight-cl"><span cla
s="highlight-w"> </span><span class="highlight-nt">application</span><span class="highl
ight-p">:</span><span class="highlight-w">
</span></span></span><span class="highlight-line"><span class="highlight-cl"><span cla
s="highlight-w"> </span><span class="highlight-nt">name</span><span class="highligh
-p">:</span><span class="highlight-w"> </span><span class="highlight-l">eureka-server<
span><span class="highlight-w">
</span></span></span><span class="highlight-line"><span class="highlight-cl"><span cla
s="highlight-w"></span><span class="highlight-nt">eureka</span><span class="highlight
p">:</span><span class="highlight-w">
</span></span></span><span class="highlight-line"><span class="highlight-cl"><span cla
s="highlight-w"> </span><span class="highlight-nt">client</span><span class="highlight
p">:</span><span class="highlight-w">
</span></span></span><span class="highlight-line"><span class="highlight-cl"><span cla
s="highlight-w"> </span><span class="highlight-nt">service-url</span><span class="highl
ight-p">:</span><span class="highlight-w">
</span></span></span><span class="highlight-line"><span class="highlight-cl"><span cla
s="highlight-w"> </span><span class="highlight-nt">defaultZone</span><span class="
ighlight-p">:</span><span class="highlight-w"> </span><span class="highlight-l">http://
27.0.0.1:10086/eureka</span><span class="highlight-w">
</span></span></span></code></pre>
```

#### 2.2.5.启动服务



 

这边把服务启动以后，点击上图中 10086/或者手动去浏览器访问 <https://ld246.com/forward?goto=http%3A%2F%2F127.0.0.1%3A10086> [http://127.0.0.1](https://ld246.com/forward?goto=http%3A%2F%2F127.0.0.1%3A10086)

显示以下界面就是成功了

### 2.3 服务注册

#### 2.3.1 引入依赖

在 user-service 的 pom 文件中，引入下面的 eureka-client 依赖：

```
<pre><code class="language-xml highlight-chroma"><span class="highlight-line"><span cl
ss="highlight-cl"><span class="highlight-nt">&lt;dependency&gt;</span>
</span></span><span class="highlight-line"><span class="highlight-cl"> <span class="h
ghlight-nt">&lt;groupId&gt;</span>org.springframework.cloud<span class="highlight-nt">
&lt;/groupId&gt;</span>
</span></span><span class="highlight-line"><span class="highlight-cl"> <span class="h
ghlight-nt">&lt;artifactId&gt;</span>spring-cloud-starter-netflix-eureka-client<span class="
ighlight-nt">&lt;/artifactId&gt;</span>
</span></span><span class="highlight-line"><span class="highlight-cl"><span class="highl
ight-nt">&lt;/dependency&gt;</span>
</span></span></code></pre>
```

#### 2.3.2 配置文件

在 user-service 中，修改 application.yml 文件，添加服务名称、eureka 地址：

```
<pre><code class="language-yaml highlight-chroma"><span class="highlight-line"><span c
```

```
ass="highlight-cl"><span class="highlight-nt">spring</span><span class="highlight-p">:</span></span><span class="highlight-w"></span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-w"></span></span><span class="highlight-nt">application</span><span class="highlight-p">:</span></span><span class="highlight-w"></span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-w"></span></span><span class="highlight-nt">name</span><span class="highlight-p">:</span></span><span class="highlight-w"></span></span><span class="highlight-l">userservice</span></span><span class="highlight-w"></span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-w"></span></span><span class="highlight-nt">eureka</span><span class="highlight-p">:</span></span><span class="highlight-w"></span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-w"></span></span><span class="highlight-nt">client</span><span class="highlight-p">:</span></span><span class="highlight-w"></span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-w"></span></span><span class="highlight-nt">service-url</span><span class="highlight-p">:</span></span><span class="highlight-w"></span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-w"></span></span><span class="highlight-nt">defaultZone</span><span class="highlight-p">:</span></span><span class="highlight-w"></span></span><span class="highlight-l">http://27.0.0.1:10086/eureka</span></span><span class="highlight-w"></span></span></span></code></pre>
```

<p>user-service 配置文件<br>

<br>order-service 配置文件<br>

</p>

<h4 id="2-3-3重启服务">2.3.3 重启服务</h4>

<p>重启服务后查看 eureka-server 管理页面<br>

<br>

如图所示，注册成功</p>

<h4 id="2-3-4模拟多实例部署">2.3.4 模拟多实例部署</h4>

<p>为了演示一个服务有多个实例的场景，我们添加一个 SpringBoot 的启动配置，再启动一个 user-service</p>

<p></p>

<p><br></p>

在环境变量 VM options 中配置-Dserver.port=8082</p>

<p>重启后发现 eureka-server 管理页面中 userService 服务中注册了两个实例<br>

</p>

<h3 id="2-4服务发现">2.4 服务发现</h3>

<h4 id="2-4-1大致步骤">2.4.1 大致步骤</h4>

<p></p>

<p></p>

<h4 id="2-4-2服务拉取和负载均衡">2.4.2 服务拉取和负载均衡</h4>

<p>给 RestTemplate 这个 Bean 添加一个 @LoadBalanced 注解<br>

修改访问的 url 路径，用服务名代替 ip、端口</p>

<p> <b>

 <br>

 <br>

 <br>

由上图可以看到，我在浏览器请求了两次，分别在 userService 的两个实例中各执行了一次，说明拉成功。</p>

<p>由此得知，spring 会自动帮助我们 from eureka-server 端，根据 userservice 这个服务名称，获实例列表，而后完成负载均衡。</p>

<h2 id="3-总结">3、总结</h2>

<p> </p>