[**史上最简单的SpringCloud教程 | 第七篇: 高可用的分布式配置中心(Spring Cloud Config)**](http://blog.csdn.net/forezp/article/details/70037513)

标签： [java](http://www.csdn.net/tag/java)[spring-cloud](http://www.csdn.net/tag/spring-cloud)[微服务](http://www.csdn.net/tag/%e5%be%ae%e6%9c%8d%e5%8a%a1)[分布式](http://www.csdn.net/tag/%e5%88%86%e5%b8%83%e5%bc%8f)[spring](http://www.csdn.net/tag/spring)

2017-04-10 21:24 137354人阅读 [评论](http://blog.csdn.net/forezp/article/details/70037513#comments)(34) [收藏](javascript:void(0);) [举报](http://blog.csdn.net/forezp/article/details/70037513#report)

http://static.blog.csdn.net/images/category_icon.jpg 分类：

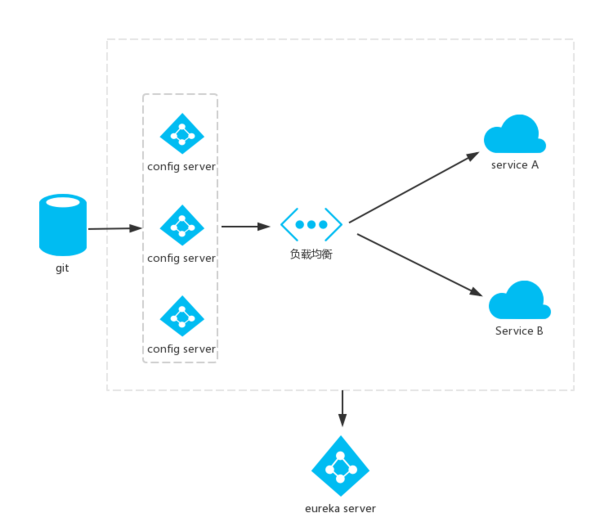
springcloud（26） http://static.blog.csdn.net/images/arrow_triangle%20_down.jpg

版权声明：本文为博主原创文章，欢迎转载，转载请注明作者、原文超链接 ，博主地址：http://blog.csdn.net/forezp。

目录[(?)[+]](http://blog.csdn.net/forezp/article/details/70037513)

转载请标明出处：   
<http://blog.csdn.net/forezp/article/details/70037513>   
本文出自[方志朋的博客](http://blog.csdn.net/forezp)

上一篇文章讲述了一个服务如何从配置中心读取文件，配置中心如何从远程git读取配置文件，当服务实例很多时，都从配置中心读取文件，这时可以考虑将配置中心做成一个微服务，将其集群化，从而达到高可用，架构图如下：



一、准备工作

继续使用上一篇文章的工程，创建一个eureka-server工程，用作服务注册中心。

在其pom.xml文件引入Eureka的起步依赖spring-cloud-starter-eureka-server，代码如下:

<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.forezp</groupId>

<artifactId>eureka-server</artifactId>

<version>0.0.1-SNAPSHOT</version>

<packaging>jar</packaging>

<name>eureka-server</name>

<description>Demo project for Spring Boot</description>

<parent>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-parent</artifactId>

<version>1.5.2.RELEASE</version>

<relativePath/> <!-- lookup parent from repository -->

</parent>

<properties>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

<project.reporting.outputEncoding>UTF-8</project.reporting.outputEncoding>

<java.version>1.8</java.version>

</properties>

<dependencies>

<dependency>

<groupId>org.springframework.cloud</groupId>

<artifactId>spring-cloud-starter-eureka-server</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-web</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-test</artifactId>

<scope>test</scope>

</dependency>

</dependencies>

<dependencyManagement>

<dependencies>

<dependency>

<groupId>org.springframework.cloud</groupId>

<artifactId>spring-cloud-dependencies</artifactId>

<version>Dalston.RC1</version>

<type>pom</type>

<scope>import</scope>

</dependency>

</dependencies>

</dependencyManagement>

<build>

<plugins>

<plugin>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-maven-plugin</artifactId>

</plugin>

</plugins>

</build>

<repositories>

<repository>

<id>spring-milestones</id>

<name>Spring Milestones</name>

<url>https://repo.spring.io/milestone</url>

<snapshots>

<enabled>false</enabled>

</snapshots>

</repository>

</repositories>

</project>

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9
* 10
* 11
* 12
* 13
* 14
* 15
* 16
* 17
* 18
* 19
* 20
* 21
* 22
* 23
* 24
* 25
* 26
* 27
* 28
* 29
* 30
* 31
* 32
* 33
* 34
* 35
* 36
* 37
* 38
* 39
* 40
* 41
* 42
* 43
* 44
* 45
* 46
* 47
* 48
* 49
* 50
* 51
* 52
* 53
* 54
* 55
* 56
* 57
* 58
* 59
* 60
* 61
* 62
* 63
* 64
* 65
* 66
* 67
* 68
* 69
* 70
* 71
* 72
* 73
* 74
* 75
* 76
* 77
* 78

在配置文件application.yml上，指定服务端口为8889，加上作为服务注册中心的基本配置，代码如下：

server:

port: 8889

eureka:

instance:

hostname: localhost

client:

registerWithEureka: false

fetchRegistry: false

serviceUrl:

defaultZone: http://${eureka.instance.hostname}:${server.port}/eureka/

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9
* 10
* 11

入口类：

@EnableEurekaServer

@SpringBootApplication

public class EurekaServerApplication {

public static void main(String[] args) {

SpringApplication.run(EurekaServerApplication.class, args);

}

}

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9

二、改造config-server

在其pom.xml文件加上EurekaClient的起步依赖spring-cloud-starter-eureka，代码如下:

<dependencies>

<dependency>

<groupId>org.springframework.cloud</groupId>

<artifactId>spring-cloud-config-server</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-test</artifactId>

<scope>test</scope>

</dependency>

<dependency>

<groupId>org.springframework.cloud</groupId>

<artifactId>spring-cloud-starter-eureka</artifactId>

</dependency>

</dependencies>

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9
* 10
* 11
* 12
* 13
* 14
* 15
* 16
* 17
* 18

配置文件application.yml，指定服务注册地址为<http://localhost:8889/eureka/>，其他配置同上一篇文章，完整的配置如下：

spring.application.name=config-server

server.port=8888

spring.cloud.config.server.git.uri=https://github.com/forezp/SpringcloudConfig/

spring.cloud.config.server.git.searchPaths=respo

spring.cloud.config.label=master

spring.cloud.config.server.git.username= your username

spring.cloud.config.server.git.password= your password

eureka.client.serviceUrl.defaultZone=http://localhost:8889/eureka/

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9

最后需要在程序的启动类Application加上@EnableEureka的注解。

三、改造config-client

将其注册微到服务注册中心，作为Eureka客户端，需要pom文件加上起步依赖spring-cloud-starter-eureka，代码如下：

<dependency>

<groupId>org.springframework.cloud</groupId>

<artifactId>spring-cloud-starter-config</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-eureka</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-test</artifactId>

<scope>test</scope>

</dependency>

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9
* 10
* 11
* 12
* 13
* 14
* 15
* 16
* 17
* 18
* 19

配置文件bootstrap.properties，注意是bootstrap。加上服务注册地址为<http://localhost:8889/eureka/>

spring.application.name=config-client

spring.cloud.config.label=master

spring.cloud.config.profile=dev

#spring.cloud.config.uri= http://localhost:8888/

eureka.client.serviceUrl.defaultZone=http://localhost:8889/eureka/

spring.cloud.config.discovery.enabled=true

spring.cloud.config.discovery.serviceId=config-server

server.port=8881

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9
* 10
* 11
* spring.cloud.config.discovery.enabled 是从配置中心读取文件。
* spring.cloud.config.discovery.serviceId 配置中心的servieId，即服务名。

这时发现，在读取配置文件不再写ip地址，而是服务名，这时如果配置服务部署多份，通过负载均衡，从而高可用。

依次启动eureka-servr,config-server,config-client   
访问网址：<http://localhost:8889/>



访问<http://localhost:8881/hi>，浏览器显示：

foo version 3

本文源码下载：   
<https://github.com/forezp/SpringCloudLearning/tree/master/chapter7>

四、参考资料

[spring\_cloud\_config](http://projects.spring.io/spring-cloud/spring-cloud.html#_spring_cloud_config)