

Spring Cloud Consul项目是针对Consul的服务治理实现。

与eureka不同的是，consul提供了专门的服务端，不需要我们自己搭建服务端了，所以直接开始介绍搭建服务提供者。

先上pom。

```
<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>

  <parent>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-parent</artifactId>
    <version>1.5.10.RELEASE</version>
  </parent>

  <groupId>com.xyz</groupId>
  <artifactId>consul-client</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>jar</packaging>

  <name>consul-client</name>
  <url>http://maven.apache.org</url>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <jdk.version>1.8</jdk.version>
  </properties>

  <dependencies>
    <dependency>
      <groupId>org.springframework.cloud</groupId>
      <artifactId>spring-cloud-starter-consul-discovery</artifactId>
    </dependency>
    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-web</artifactId>
    </dependency>
  </dependencies>

  <dependencyManagement>
    <dependencies>
      <dependency>
        <groupId>org.springframework.cloud</groupId>
        <artifactId>spring-cloud-dependencies</artifactId>
        <version>Dalston.SR1</version>
        <type>pom</type>
      </dependency>
    </dependencies>
  </dependencyManagement>
</project>
```

```

        <scope>import</scope>
      </dependency>
    </dependencies>
  </dependencyManagement>

  <build>
    <plugins>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>3.3</version>
        <configuration>
          <!-- 指定source和target的版本 -->
          <source>${jdk.version}</source>
          <target>${jdk.version}</target>
        </configuration>
      </plugin>
    </plugins>
  </build>
</project>

```

然后再上个配置文件。

```

spring.application.name=consul-client
server.port=2002

spring.cloud.consul.host=localhost
spring.cloud.consul.port=8500

```

其它的和上一篇的eureka-client里面一样，为什么呢？

DiscoveryClient是Spring Cloud对服务治理做的一层抽象，所以可以屏蔽Eureka和Consul服务治理的实现细节，我们的程序不需要做任何改变，只需要引入不同的服务治理依赖，并配置相关的配置属性就能轻松的将微服务纳入Spring Cloud的各个服务治理框架中。

最后的效果。

Service	Node Health
consul	✓ 1
consul-client	✓ 1

← → ↻ 🏠 ⓘ localhost:2002/test

应用 📁 VP+企业级管理平台 📁 github开源项目 H H

Services: [consul, consul-client]