# 微服务面试题

## Spring Cloud有哪些特点？

Spring cloud 流应用程序启动器是基于 Spring Boot 的 Spring 集成应用程序，提供与外部系统的集成。Spring cloud Task，一个生命周期短暂的微服务框架，用于快速构建执行有限数据处理的应用程序。

特点:

1、约定优于配置。

2、适用于各种环境。开发、部署PC Server或各种云环境如阿里云、AWS等。

3、隐藏了组件的复杂性，并提供声明式、无xml的配置方式。

4、开箱即用快速启动。

5、轻量级组件。

6、组件丰富：Eureka、ribbon、Hystrix、config、bus、zuul。

7、选型中立，例如spring cloud支持Eureka、consul、zookeeper。

## CAP与Base理论

CAP是一个已经经过证实的理论：一个分布式系统最多只能同时满足一致性（Consistency）、可用性（Availability）和分区容错性（Partition tolerance）这三项中的两项。

BASE理论是对CAP理论的延伸，思想是即使无法做到强一致性（CAP的一致性就是强一致性），但可以采用适当的采取弱一致性，即最终一致性。

BASE是指基本可用（Basically Available）、软状态（ Soft State）、最终一致性（ Eventual Consistency）。

## Spring cloud与Dubbo的区别

（1）服务调用方式 dubbo是RPC springcloud Rest Api；

（2）注册中心,dubbo 是zookeeper springcloud是eureka，也可以是zookeeper、consul；

（3）服务网关,dubbo本身没有实现，只能通过其他第三方技术整合，springcloud有Zuul路由网关，作为路由服务器，进行消费者的请求分发,springcloud支持断路器，与git完美集成配置文件支持版本控制，事物总线实现配置文件的更新与服务自动装配等等一系列的微服务架构要素。

## 四、Spring Boot与 Spring cloud之间的关系

1、SpringBoot专注于快速、方便的开发单个微服务个体，SpringCloud关注全局的服务治理框架。

2、SpringCloud是关注全局的微服务协调整理治理框架，它将SpringBoot开发的一个个单体微服务整合并管理起来，为各个微服务之间提供，配置管理、服务发现、断路器、路由、微代理、事件总线、全局锁、决策竞选、分布式会话等等集成服务。

3、SpringBoot可以离开SpringCloud独立使用开发项目， 但是SpringCloud离不开SpringBoot ，属于依赖的关系。

## 五、负载均衡的意义

负载均衡可以改善跨计算机、计算机集群、网络连接、中央处理单元或磁盘驱动等多种计算资源的工作负载分布。负载均衡意在优化资源分配，最大化吞吐量，最小化响应时间并避免任一资源的过载。负载均衡通常会涉及专用硬件或软件，例如F5、nginx、ribbon。

## 六、什么是Hystrix，如何做的容错

**服务隔离：**如果整个系统雪崩是由于一个接口导致的，由于这一个接口响应不及时导致问题，那么我们就有必要对这个接口进行隔离，就是只允许这个接口最多能接受多少的并发，做了这样的限制后，该接口的主机就会空余线程出来接收其他的情况，不会被哪个坏了的接口占用满。Hystrix 就是一个不错的服务隔离框架。

**服务雪崩：**雪崩是系统中的蝴蝶效应导致其发生的原因多种多样，有不合理的容量设计，或者是高并发下某一个方法响应变慢，亦或是某台机器的资源耗尽。从源头上我们无法完全杜绝雪崩源头的发生，但是雪崩的根本原因来源于**服务之间的强依赖**，所以我们可以提前评估。当整个微服务系统中，有一个节点出现异常情况，就有可能在高并发的情况下出现雪崩，导致调用它的上游系统出现响应延迟，响应延迟就会导致 tomcat 连接本耗尽，导致该服务节点不能正常的接收到正常的情况，这就是服务雪崩行为。

Hystrix两种隔离策略：线程池隔离THREAD 、信号量SEMAPHORE隔离。

**熔断发生的三个必要条件：**

1、有一个统计的**时间周期**，滚动窗口

相应的配置属性metrics.rollingStats.timeInMilliseconds 默认 10000 毫秒

2、**请求次数**必须达到一定数量

相应的配置属性 circuitBreaker.requestVolumeThreshold默认 20 次

3、**失败率达到默认失败率**

相应的配置属性 circuitBreaker.errorThresholdPercentage 默认 50%

**用到的注解：**

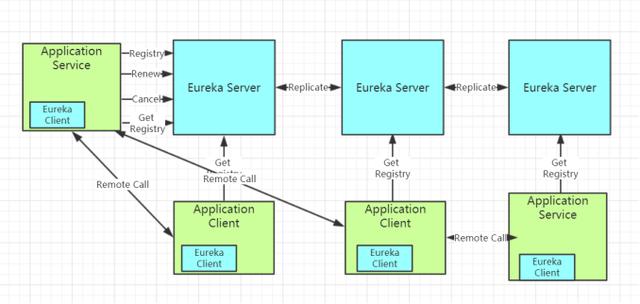
@EnableCircuitBreaker

@HystrixCommand(fallbackMethod="queryOrderByIdFallback",commandKey = "queryOrderById",groupKey = "querygroup-one",commandProperties = { @HystrixProperty(name ="execution.isolation.strategy", value="THREAD")},threadPoolKey = "queryOrderByIdhystrixJackpool")

Hystrix有哪几种状态：开启、关闭、办开启。

参考文档：<https://www.toutiao.com/i6805112656714793475/>

## 七、介绍一下Eureka



用到的注解：

@EnableEurekaServer // 注册服务中心

@EnableEurekaClient // 服务提供端

**服务注册与续约：**首先服务注册到eureka server上，注册之后服务提供者会维护一个心跳，用来告诉eureka server “我还OK”，也就是续约Renew，心跳默认三十秒检测一次。

**服务剔除：**当服务实例正常下线时，服务实例会告诉eureka server “我要下线了”，将服务状态改为 down。有时服务会出现非正常下线，eureka server 在启动时会去创建一个定时任务检测服务提供者，剔除服务，检测时间默认90秒。

**自我保护：**eureka server 会去将心跳在15分钟内是否低于85%，如果出现eureka会将他们保护起来，让这些实例不会过期，这样客户端就很容易拿到不存在的实例，这就需要容错机制，客户端重试、断路器等机制。

**zone与region：**

1、机房隔离

2、Ribbon默认优先访问一个zone中的服务实例

## 八、介绍一下Ribbon

spring Cloud Ribbon 是一个客户端的负载均衡器，它提供对大量的HTTP和TCP客户端的访问控制。

客户端负载均衡即是当浏览器向后台发出请求的时候，客户端会向 Eureka Server 读取注册到服务器的可用服务信息列表，然后根据设定的负载均衡策略（没有设置即用默认的），抉择出向哪台服务器发送请求。

重要注解：@Loadbalanced 注解修饰RestTemplate

负载均衡策略：线性轮询、随机、重试机制。

Ribbon内部有一个chooseServer()方法用于选择服务。

## 聊聊Feign

Feign使代码变得整洁，内部封装了Ribbon的负载以及Hystrix。但是性能会比较差，建议复杂业务不要使用。

## 为什么使用Spring Cloud Bus

Spring Cloud Bus 提供了跨多个实例刷新配置的功能

## 十一、什么是SpringCloudConfig?

在分布式系统中，由于服务数量巨多，为了方便服务配置文件统一管理，实时更新，所以需要分布式配置中心组件。在Spring Cloud中，有分布式配置中心组件spring cloud config ，它支持配置服务放在配置服务的内存中（即本地），也支持放在远程Git仓库中。在spring cloud config 组件中，分两个角色，一是config server，二是config client。

使用：

（1）添加pom依赖

（2）配置文件添加相关配置

（3）启动类添加注解@EnableConfigServer

## 十二、Spring Cloud Gateway?

Spring Cloud Gateway是Spring Cloud官方推出的第二代网关框架，取代Zuul网关。网关作为流量的，在微服务系统中有着非常作用，网关常见的功能有**路由转发、权限校验、限流控制**等作用。

使用了一个RouteLocatorBuilder的bean去创建路由，除了创建路由RouteLocatorBuilder可以让你添加各种predicates和filters，predicates断言的意思，顾名思义就是根据具体的请求的规则，由具体的route去处理，filters是各种过滤器，用来对请求做各种判断和修改。