

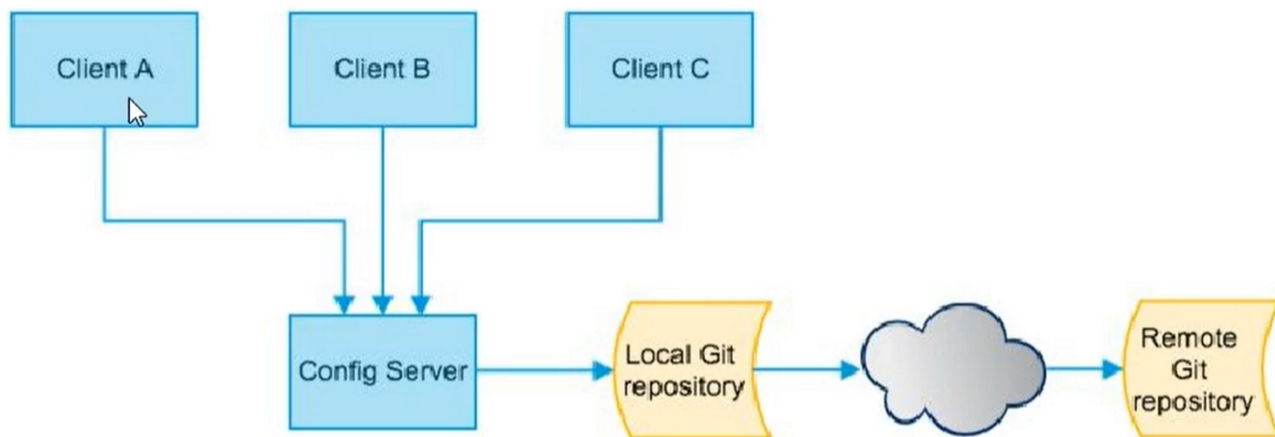
服务配置

2020年8月6日 11:34

分布式微服务面临的配置问题

一个model需要一个配置文件，不便于管理，springcloud提供了configserver来解决这个问题

微服务意味着要将单体应用中的业务拆分成一个个子服务，每个服务的粒度相对较小，因此系统中会出现大量的服务。由于每个服务都需要必要的配置信息才能运行，所以一套集中式的、动态的配置管理设施是必不可少的。



是什么

SpringCloud Config为微服务架构中的微服务提供集中化的外部配置支持，配置服务器为各个不同微服务应用的所有环境提供了一个中心化的外部配置。

怎么用

SpringCloud Config分为服务端和客户端两部分。

服务端也称为分布式配置中心，它是一个独立的微服务应用，用来连接配置服务器并为客户端提供获取配置信息，加密/解密信息等访问接口

客户端则是通过指定的配置中心来管理应用资源，以及与业务相关的配置内容，并在启动的时候从配置中心获取和加载配置信息配置服务器默认采用git来存储配置信息，这样就有助于对环境配置进行版本管理，并且可以通过git客户端工具来方便的管理和访问配置内容。

能干些什么

集中管理配置文件

不同环境不同配置，动态化的配置更新，分环境部署比如dev/test/prod/beta/release

运行期间动态调整配置，不再需要在每个服务部署的机器上编写配置文件，服务会向配置中心统一拉取配置自己的信息

当配置发生变动时，服务不需要重启即可感知到配置的变化并应用新的配置

将配置信息以REST接口的形式暴露

config服务端配置

用你自己的账号在GitHub上新建一个名为springcloud-config的新Repository

由上一步获得刚新建的git地址 `git@github.com:zzyybs/springcloud-config.git`

本地硬盘目录上新建git仓库并clone

3

此时在本地D盘符下D:\44\SpringCloud2020\springcloud-config

6

新建Module模块cloud-config-center-3344

它即为Cloud的配置中心模块cloudConfig Center

POM

POM 

YML 

主启动类

2

windows下修改hosts文件，增加映射

1

测试通过Config微服务是否可以从GitHub上获取配置内容

2

配置读取规则

20

成功实现了用SpringCloud Config通过GitHub获取配置信息

`/{{label}}-{{name}}-{{profiles}}.yml`

label: 分支(branch)

name: 服务名

profiles: 环境(dev/test/prod)

applicaiton.yml是用户级的资源配置项

bootstrap.yml是系统级的，优先级更加高