分布式配置中心

2018年5月7日

17:04

## 1.构建ConfigServer

Dependencies 
Web 
Template Engines 
SQL 
NOSQL 
Integration 
Cloud core 
Spring Boot 
¯ Config Client 
Server 
- _ Vault Configuration 
Zookeeper Configuration 
Consul Configuration 
2.0_1 
Selected Dependencies 
Cloud Config 
Config Server 
Cloud Discovery 
Cloud Routing 
Cloud Circuit Breaker 
Cloud Tracing 
Cloud Messaging 
Cloud AWS 
Cloud Contract 
Pivotal Cloud Foundry 
Azure 
spring Cloud GCP 
1/0 
ops 
Config Server 
Central management for configuration via a git or 
SVn backend 
Centralized Configuration 
Cancel 
Help 

在程序的入口Application类加上@EnableConfigServer注解开启配置服务器的功能，代码如下

@SpringBootApplication 
@EnableConfigServer 
public class ConfigserverApplication ( 
public static void main (StringC] args) 
SpringAppl ication. run(ConfigserverAppl ication. class, 
args) , 

需要在程序的配置文件application.yml文件配置以下：

spring: 
applicati on: 
name: config-server 
cloud: 
config: 
server : 
git: 
uri: https://github. com/qita0321/springCloudConfig/ 
search-paths: /respo 
username: 
password: 
label: master 
server : 
port: 8888 
eureka : 
client: 
servi ceUrl : 
defau I t.7.one: 
http://localhost : 8761 /eureka/ 

* + spring.cloud.config.server.git.uri：配置git仓库地址
  + spring.cloud.config.server.git.searchPaths：配置仓库路径(可写为/\*\*)
  + spring.cloud.config.label：配置仓库的分支
  + spring.cloud.config.server.git.username：访问git仓库的用户名(public不需要)
  + spring.cloud.config.server.git.password：访问git仓库的用户密码(public不需要)

证明配置服务中心可以从远程程序获取配置信息。

http请求地址和资源文件映射如下:

* + /{application}/{profile}[/{label}]
  + /{application}-{profile}.yml
  + /{label}/{application}-{profile}.yml
  + /{application}-{profile}.properties
  + /{label}/{application}-{profile}.properties

Server端启动后，提供了如下的接口地址，参数说明

* + application：应用名
  + profile：环境
  + label：版本

## 2.构建一个ConfigClient

n New Module 
Dependencies 
Core 
Web 
Template Engines 
SQL 
NoSQL 
Integration 
Cloud Core 
Cloud Discovery 
Cloud Routing 
Cloud Circuit Breaker 
Cloud Tracing 
Cloud Messaging 
Cloud AWS 
Cloud Contract 
Pivotal Cloud Foundry 
Azure 
Spring Cloud GCP 
Ops 
Spring Boot 2.0.1 
Client 
Config Server 
C/ J Vault Configuration 
Zookeeper Configuration 
Consul Configuration 
Config Client 
spring-cloud-config Client 
Previous 
Selected Dependencies 
Next 
Cloud Config 
Config Client 
Cancel 
x 
Help 

新增

**bootstrap.properties配置文件**

spring. cloud. config. label-master 
spri cloud. config. profil 
spring. cloud. config. uri= http://localhost : 8888/ 

**Application.yml配置文件**

server : 
port: 8881 
spr Ing: 
application: 
name: config—client 
eureka : 
client: 
serviceLrl : 
defaul tZone : 
ht tp : // localhost : 8761/eureka/ 

* + spring.cloud.config.label 指明远程仓库的分支
  + spring.cloud.config.profile
    - dev开发环境配置文件
    - test测试环境
    - pro正式环境
  + spring.cloud.config.uri= <http://localhost:8888/> 指明配置服务中心的网址。

程序的入口类，写一个API接口“／hi”，返回从配置中心读取的name1变量的值，代码如下：

j@RestController 
public class Configc1ientApp1ication ( 
public static void main (StringCl args) 
(namel} 
private String namel 
public String hi ( ) { 
return namel; 
( SpringApplication. run(ConfigclientApplication. class, 
args) . 

|  |  |
| --- | --- |
| git远程仓库地址 | <https://github.com/qitao321/springCloudConfig> |

计算机生成了可选文字:
qita0321/springCloudConfig
Projects
0
Wiki
Insights
@Unwatch
Settings
1
Star
0
Fork
0
〈〉Code
O
Issues0
门Pullrequests0
springCloudConfig/respo/
Branch:master
qita0321.ymlchangetoproperties
固config-client-dev.properties
固test-dev.properties
固test-prod.properties
固test.properties
CreatenewfileUploadfi|es
FindfileHistory
.ymlchangetoproperties
change
change
changeymltoproperties
Latestcommit1a8Cd1341minutesago
41minutesago
14hoursago
14hoursago
16hoursago

**该configclient的ApplicationName为config-client，spring.cloud.config.profile为dev，**

**所以会读取config-client-dev.properties文件（默认只会读.properties或.xml文件）**

**config-client-dev.properties文件内容如下**

计算机生成了可选文字:
Branch:master
springCloudConfig/respo/config-client-dev.properties
FindfileCopypath
1a8Cd1344minutesago
qita0321.ymlchangetoproperties
1contributor
1lines（1Sloc)
13Bytes
qitao
1
namel
Raw
Blame
History
歹
画

**为什么默认只会读.properties或.xml文件？**

**查看PropertiesPropertySourceLoader类。**

计算机生成了可选文字:
lient\...\application.yml×bootstrap.properties×
Decompiled.classfile,bytecodeversion:52℃0ava8）
config-server\...\application.yml×
mspringCloud
《、PropertiesPropertySourceLoader.cIass
DownloadSourcesChooseSources..
1
2
3
4
6
8
23
24
26@
27
28
29
//Sourcecoderecreatedfroma．classfi1ebyIntelliJIDEA
//(poweredbyFernflowerdecompiler)
packageorg.springframework.boot.env，
+import
publicclassPropertiesPropertySourceLoaderimplementsPropertySourceLoader{
·xml"
privatestaticfinalStringXMLFILEEXTENSION
publicPropertiesPropertySourceLoader0{
publicStringngetFiIeExtensions0{
returnnewStringn{"properties
publicList<PropertySource<?>>load(Stringname，Resourceresource)throwsIOException{
this.loadProperties(resource)
Map<String，？〉properties
returnproperties.isEmpty()？Collections.emptyList0
Collections.singletonList(newOriginTrackedMapPropertyS0rce(name,
privateMap<String，？〉loadProperties(Resourceresource)throwsIOException{
resource.getFilename0
Stringfilename

|  |  |  |
| --- | --- | --- |
| **会存在中文乱码现象，可参考** | <https://blog.csdn.net/sinat_38843093/article/details/79960777> | **解决（未尝试）。** |

**整体示意图**

