# 分布式微服务配置中心

## 一．百度开源项目——disconf

***官方文档*** [*https://disconf.readthedocs.io/zh\_CN/latest/*](https://disconf.readthedocs.io/zh_CN/latest/)

### 简述.

Disconf可以为各种业务平台提供统一的配置管理服务，实现配置统一化，其使用方式简捷（注解式或xml模式），代码的侵入性低，具备强兼容性（disconf-web不能正常work，则使用本地配置启动应用），使用相对广泛；但是其已经停止更新，社区活跃程度不高，使用过程仍然有些BUG以及设计不完全等问题，如springboot项目使用disconf托管配置文件第一次启动并未使用disconf上的配置，而是使用jar内部配置；disconf对于不同APP或不同环境不能实现共享配置文件的效果，所以仅仅依靠disconf实现”打散”配置文件提取不同应用中相同配置是不够的，这需要在应用中将共同配置抽取出来一个properties放到disconf托管，从而才能实现同套配置文件一处更改多处生效的效果.

### springboot项目与disconf使用xml模式结合(client).

1. 引入**POM**

<dependency>

<groupId>com.baidu.disconf</groupId>

<artifactId>disconf-client</artifactId>

<version>2.6.36</version>

</dependency>

1. 在resources下添加一个dsconf.properties (参考官方文档)

disconf.enable.remote.conf=true

disconf.conf\_server\_host=192.168.9.9:8015

disconf.version=1\_0\_0\_0

disconf.app=wch\_disconf

disconf.env=rd

disconf.ignore=

disconf.conf\_server\_url\_retry\_times=1

disconf.conf\_server\_url\_retry\_sleep\_seconds=1

disconf.user\_define\_download\_dir=./config

#disconf.enable\_local\_download\_dir\_in\_class\_path=true

1. 在resources下添加一个disconf.xml

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

<beans xmlns="http://www.springframework.org/schema/beans"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:aop="http://www.springframework.org/schema/aop"

xsi:schemaLocation="http://www.springframework.org/schema/beans

http://www.springframework.org/schema/beans/spring-beans-3.1.xsd

http://www.springframework.org/schema/aop

http://www.springframework.org/schema/aop/spring-aop-3.0.xsd">

<aop:aspectj-autoproxy proxy-target-class="true"/>

<!-- 使用disconf必须添加以下配置 -->

<bean id="disconfMgrBean" class="com.baidu.disconf.client.DisconfMgrBean"

destroy-method="destroy">

<property name="scanPackage" value="com.wch.sessionTest"/>

</bean>

<bean id="disconfMgrBean2" class="com.baidu.disconf.client.DisconfMgrBeanSecond"

init-method="init" destroy-method="destroy">

</bean>

<!-- 使用托管方式的disconf配置(无代码侵入, 配置更改会自动reload)-->

<bean id="configproperties\_disconf"

class="com.baidu.disconf.client.addons.properties.ReloadablePropertiesFactoryBean">

<property name="locations">

<list>

<value>classpath\*:application.properties</value>

<value>classpath\*:dubbo.properties</value>

<value>classpath\*:dsp.properties</value>

</list>

</property>

</bean>

<bean id="propertyConfigurer"

class="com.baidu.disconf.client.addons.properties.ReloadingPropertyPlaceholderConfigurer">

<property name="ignoreResourceNotFound" value="true"/>

<property name="ignoreUnresolvablePlaceholders" value="true"/>

<property name="propertiesArray">

<list>

<ref bean="configproperties\_disconf"/>

</list>

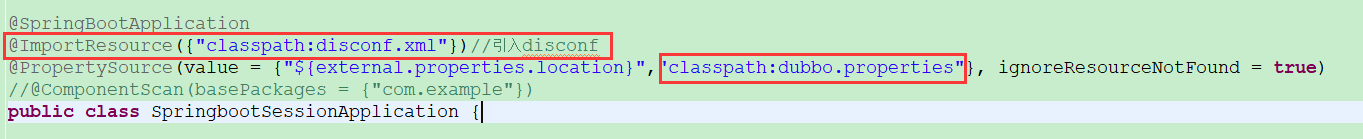
</property>

</bean>

</beans>

需要disconf托管的配置文件都放在红色标记部分.

1. 在springboot启动类上引入disconf.xml ，以及加载提取出的properties文件



Springboot项目每次启动disconf-client都会从disconf-web上下载最新配置，但并未使用最新的配置，重启一次才会使用上次下载下来的配置，这是disconf对springboot项目支持的一个bug.（client存活会自动load最新配置）

## 二．Ctrip开源项目——Apollo

***Github地址：***[*https://github.com/ctripcorp/apollo*](https://github.com/ctripcorp/apollo)