注:

日期:2017-11-20

技术栈:

Spring boot 1.4.7

Dubbo:2.5.7

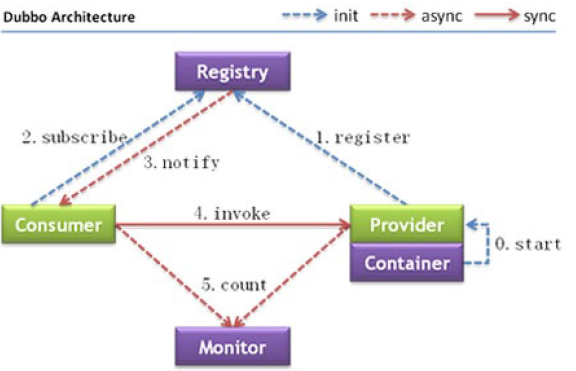
Zoo Keeper集群

参考文档:阿里dubbo-user-book

# Dubbo解决的主要需求

1. 将应用拆分成几个不相干的应用，提高维护效率，加速整体应用响应速度。
2. 提高业务复用。
3. 设立调度中心，基于访问压力实时管理集群容量，提高集群资源利用率。
4. 当独立服务越来越多的时候，基于URL的配置管理变得十分困难，设立服务注册中心，动态的注册和发现服务，使得服务的位置变得透明。
5. 通过在消费方获取服务提供方地址列表，实现软负载和Failover，减低F5硬件负载的依赖，降低成本。

# Dubbo整体架构



1. 角色说明：

Provider:暴露服务的提供方.

Comsumer:调用服务的消费方

Registry:服务注册与发现中心

Monitor:统计服务的调用次数和时间的监控中心

Container:服务运行容器

1. 调用关系说明：

1. 服务容器负责启动，加载，运行服务提供者。

2. 服务提供者在启动时，向注册中心注册自己提供的服务。

3. 服务消费者在启动时，向注册中心订阅自己所需的服务。

4. 注册中心返回服务提供者地址列表给消费者，如果有变更，注册中心将基于长连接推送

变更数据给消费者。

5. 服务消费者，从提供者地址列表中，基于软负载均衡算法，选一台提供者进行调用，如

果调用失败，再选另一台调用。

6. 服务消费者和提供者，在内存中累计调用次数和调用时间，定时每分钟发送一次统计数据到监控中心。

# 三、Dubbo的使用

Dubbo采用全Spring配置方式，透明化接入应用。对应用没有任何API侵入。只需要用Spring加载Dubbo的配置即可。

1. 添加依赖：

<dependency>  
 <groupId>com.alibaba</groupId>  
 <artifactId>dubbo</artifactId>  
 <version>2.5.7</version>  
</dependency>

1. 配置Dubbo Springbeen