1. **分布式系统服务调用的问题总结**

拆分后的order&user系统是一个分布式系统。

**1.对于集群中对不同系统的管理为0**

对集群中某个功能系统的资源空闲判断,需要引入监听的管理

某些系统发起http请求调用其他系统服务,如果出现错误,频发出现错误,会导致压力的积压.

**2.依然存在强耦合,并且导致维护工作量大**

nginx的配置内容,会根据服务器,系统的运行环境发生变化而频繁的修改,不利于维护的方便;

**二、微服务框架(springcloud)**

# 1.springboot与springcloud的关系

## springboot单独存在的意义不大,独立运行,自动配置,完全是为了微服务的框架而创建的.springcloud框架必须基于springboot的工程才可以实现;

# 2.微服务与微服务框架

## 2.1什么是微服务

## 当单体系统进行纵向拆分,拆分的独立系统就叫做微服务;每个微服务都专注于一个功能领域,之间通过服务调用获取,传递数据;

## 2.2什么是微服务框架

## 当微服务特别庞大的时候,就需要从管理.监控角度出发,对整个集群做到架构级别的技术整合--这种技术就成为微服务框架;

## 2.3微服务框架技术

## 常见的有spring cloud，dubbo(12年宣布停更,17年恢复更新);

## spring cloud: 一种轻量级的,组件丰富全面的微服务框架技术,核心注册发现组件Eureka(不能由别人代替),实现的http的协议传输数据;组件特别丰富，所以开发投入成本较低;

## dubbo: 组件没有springcloud丰富,注册发现组件可以使用Eureka(http协议),redis,zookeeper(rpc协议),使用rpc协议的技术,性能高;http性能虽然没有rpc高但是灵活度.

## 技术选型: spring cloud 开发成本低,所有组件是现成的直接使用,更新比较及时,对应市面的各种新技术做最新的进步;

## dubbo 性能高,是dubbo重要的优点,开发成本高