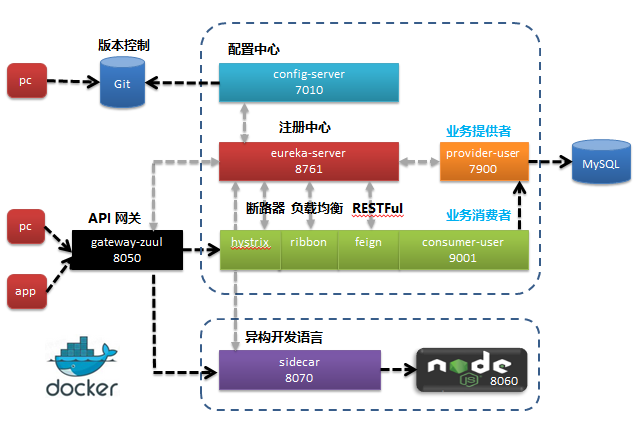
知识回顾：



1. SpringCloud微服务框架

基于分布式

1. 断路器解决异常，单点微服务故障（Hystrix断路器方式可以解决，类似雪崩；Ribbon负载均衡，对一个业务启动多个微服务实例）
2. 每个微服务都是部署在集群环境中，可能不在一台机器。Eureka和Zuul首先配置为集群，保证单点故障后，其他节点接替工作。根据性能的搭配，来分布在整个集群环境中。但相同的业务微服务，多例时，尽量分散在不同的机器。Hystrix+ribbon+feign+consumer是在一起部署。提供者也是分散部署在各个机器中。
3. 数据库传统大集中部署，不会和业务系统部署在一起。但是在微服务情况下，会有一个数据库集群，数据库分散在各个机器。
4. SpringCloud微服务
5. Why？故障隔离，好升级（动态升级，不停机，停很小范围，影响范围更小，影响时间更短。）是不是必由之路，是。是不是万能的？不是。
6. 定义、用途、内容。SpringCloud实现微服务一套解决方案。最重要组件：Eureka注册中心（Eureka类似ZooKeeper，但是比zk好）、Ribbon前端负载均衡（类似Nginx，它比Nginx更加强大），Feign实现声明式web服务（接口无需学写api，RestTemplate.getForObject），Hystrix实现断路器降级，Zuul API网关，Sidecar异构开发语言支持，SpringCloudConfig配置中心。
7. 同类产品比较，主流微服务框架有：Dubbo和SpringCloud。Dubbo阿里，在实践中检验。阿里内部用了6年。而且很多公司使用：当当Dubbox（RESTFul+升级spring版本）Dubbo重新维护，修改了内核stream，RPC TCP/IP请求-响应 nio非阻塞IO，线程复用，不释放，没有创建和销毁过程。长链接。Ping 1/s。流，响应式，事件。比之前结构性能更高，耗费资源更少。学习kafka MQ。Redis5.0 基于stream。它没有API网关概念。Dubbo+Shiro。Zuul实现一次请求，内部含有多个微服务。SpringCloud外延是远远超过Dubbo的。SpringCloud在RPC解决方案中不如Dubbo，它考虑是整个架构通用性，根上基于REST+json，性能就干不过RPC。将来怎么发展？SpringCloud解决整体解决方案，涉及到编程各个方便，简化编程；Dubbo就是SpringCloud它的一个RPC的专门解决方案。Dubbo是SpringCloud子集。
8. 和项目来结合：我们项目中采用SpingBoot+SpringCloud微服务技术，实现整体架构，实现单点登录、实现购物车、实现订单等等微服务。我负责京淘项目中购物车的业务模块。我们先构建API网关，用了Zuul。实现它到消费者微服务映射。给消费者映射一个目录，在浏览器请求时，多加映射路径，这样方便API对url的权限控制。接着就创建消费者，使用Hystrix断路器（实现一个ZuulFallbakProvider接口，一个获取路由，配置一个\*，代表所有的方法都可以fallback；写一个响应头，设置格式，json+UTF8；写一个返回内容body，字符串json），使用Ribbon负载均衡（加入hystrix，它依赖ribbon，内部集成，利用Eureka配合会自动选择可以执行通路），使用Feign（接口，坑对对象直接支持不好，日期），调用消费者。最后写提供者，提供和传统业务系统差异不大，主要差异在构建pom不同，必须springBoot构建，而且它是一个Eureka客户端。

2）重构购物车

业务垂直拆分，购物车单独子系统，购物车Dubbo微服务，购物车SpringBoot+SpringCloud

1. Zuul API网关（复用，修改配置文件，映射）
2. SpringBoot构建业务提供者，基本功能实现，基于SSM+MP，对外controller REST支持
3. 消费者，Hystrix（接口），Ribbon（内置），Feign（接口，接收对象参数@RequestBody），完成自身业务
4. Eureka服务端（复用）

提供者：

1. Pojo Cart.java
2. Mybatis接口 CartMapepr继承BaseMapper
3. 业务层，CartService，CartServiceImpl
4. 控制层，CartController
5. 返回值SysResult（json）

大数据要学吗？

1. 大数据解决方案，之前的传统技术解决不了的！
2. 大数据解决什么问题？海量数据，入门亿条记录。Mysql单表能支持多少数据，1千万。500w最佳，如果不优化，10分钟。创建索引，秒级返回。如果字段不含有字符串，800w最佳。Oracle单表上1亿。

海量数据，亿级以上，秒级返回。

1. solr/es全文检索（索引），解决查询问题，海量数据分布式存储和计算
2. 大数据数据存储和分析（计算）
3. 两个生态链：离线分析Hadoop HDFS分布式海量数据存储，MapReduce(MR)分布式计算，Hive计算，数据仓库（冗余）数据挖掘，HBase列式存储，实时分析 Spark（1s~5s）批量、Storm（0.1s~0.5s）一条一条
4. 网络流量日志系统PV/UV/VV
5. 大数据集群动态伸缩