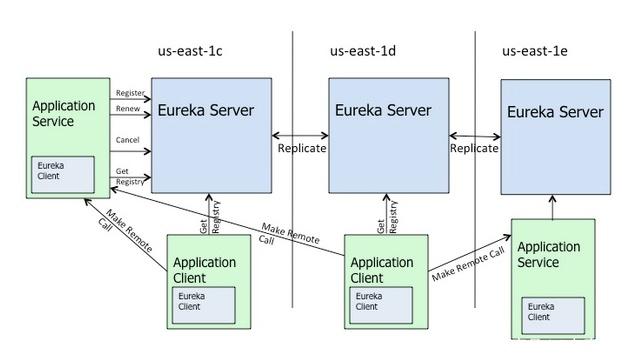
SpringCloud的组件相当繁杂，拥有诸多子项目。重点关注Netflix

下面只简单介绍下经常用的5个

* 服务发现——Netflix Eureka
* 客服端负载均衡——Netflix Ribbon
* 断路器——Netflix Hystrix
* 服务网关——Netflix Zuul
* 分布式配置——Spring Cloud Config

# Eureka



作用：实现服务治理（服务注册与发现）

简介：Spring Cloud Eureka是Spring Cloud Netflix项目下的服务治理模块。

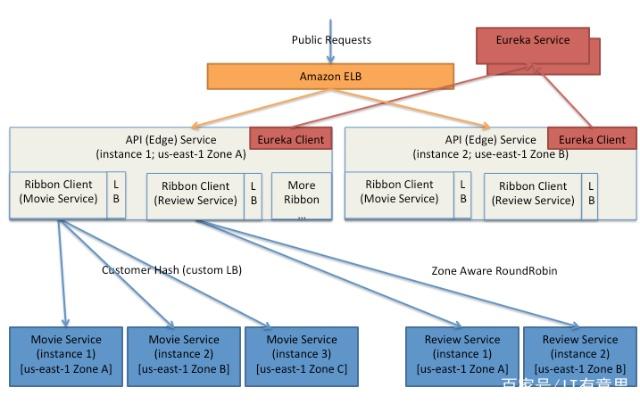
由两个组件组成：Eureka服务端和Eureka客户端。

Eureka服务端用作服务注册中心。支持集群部署。

Eureka客户端是一个java客户端，用来处理服务注册与发现。

在应用启动时，Eureka客户端向服务端注册自己的服务信息，同时将服务端的服务信息缓存到本地。客户端会和服务端周期性的进行心跳交互，以更新服务租约和服务信息。

# Ribbon

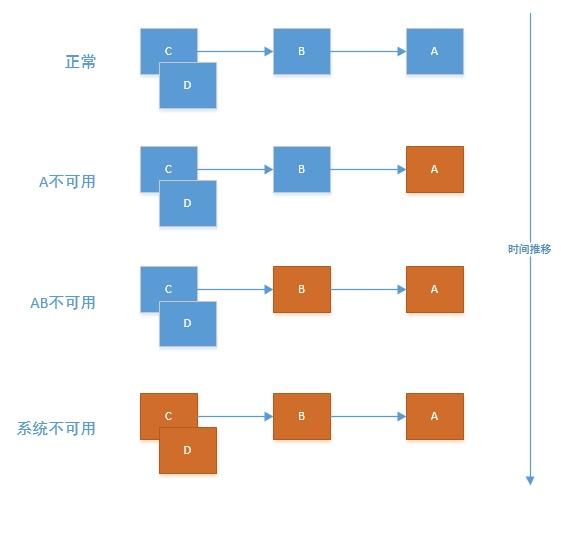


作用：Ribbon，主要提供客户侧的软件负载均衡算法。

简介：Spring Cloud Ribbon是一个基于HTTP和TCP的客户端负载均衡工具，它基于Netflix Ribbon实现。通过Spring Cloud的封装，可以让我们轻松地将面向服务的REST模版请求自动转换成客户端负载均衡的服务调用。

注意看上图，关键点就是将外界的rest调用，根据负载均衡策略转换为微服务调用。Ribbon有比较多的负载均衡策略，以后专门讲解。

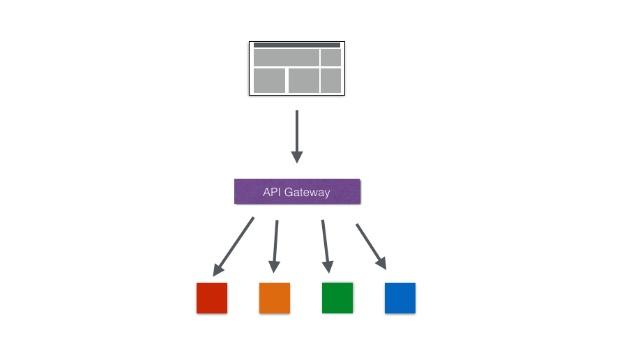
# Hystrix



作用：断路器，保护系统，控制故障范围。

简介：为了保证其高可用，单个服务通常会集群部署。由于网络原因或者自身的原因，服务并不能保证100%可用，如果单个服务出现问题，调用这个服务就会出现线程阻塞，此时若有大量的请求涌入，Servlet容器的线程资源会被消耗完毕，导致服务瘫痪。服务与服务之间的依赖性，故障会传播，会对整个微服务系统造成灾难性的严重后果，这就是服务故障的“雪崩”效应。

# Zuul

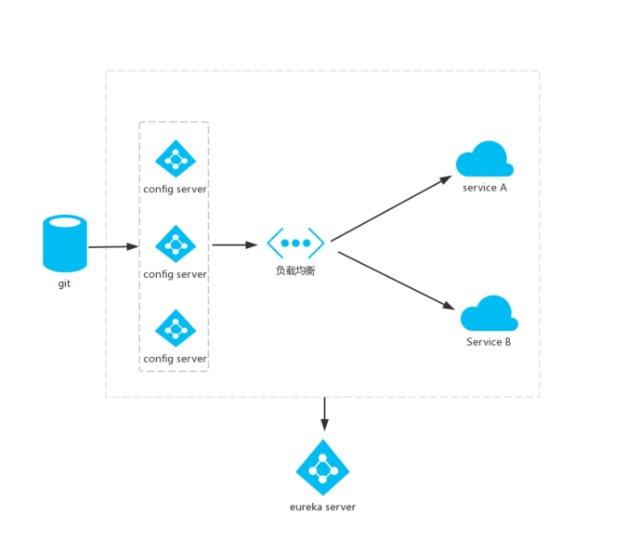


作用：api网关，路由，负载均衡等多种作用

简介：类似nginx，反向代理的功能，不过netflix自己增加了一些配合其他组件的特性。

在微服务架构中，后端服务往往不直接开放给调用端，而是通过一个API网关根据请求的url，路由到相应的服务。当添加API网关后，在第三方调用端和服务提供方之间就创建了一面墙，这面墙直接与调用方通信进行权限控制，后将请求均衡分发给后台服务端。

# Config



作用：配置管理

简介：SpringCloud Config提供服务器端和客户端。服务器存储后端的默认实现使用git，因此它轻松支持标签版本的配置环境，以及可以访问用于管理内容的各种工具。

这个还是静态的，得配合Spring Cloud Bus实现动态的配置更新。