《28\_C2C电商社会化治理平台的蓝绿部署、灰度发布以及滚动发布》

对大多数中小公司而言，可能用jenkins之类的工具做一个自动化的滚动部署就差不多了，此时教大家一些经验，就是说，灰度发布，其实我们可以玩儿成伪灰度发布，什么意思呢？比如你一个服务部署了3台机器

那么当你有一个新版本的时候，你先部署1台机器，这不就变相的相当于是灰度了？默认会有30%+的流量进入这台机器，你先观察一下这30%的流量表现如何，如果感觉不错，再把另外2台机器部署了

2台机器，你有新版本，先部署1台机器，50%流量过来，先观察一下再说

说白了，就是人为的去控制这个部署的节奏和过程，流量分发这块不要动，还是均匀分发，搞出一个伪灰度的效果

蓝绿部署也是同样的，一般注册中心都是支持主动调用它的API进行服务下线的，此时你可以让你每个服务都实现一个下线的接口，里面调用eureka/nacos之类的API可以主动下线你的服务实例

此时你可以部署一个新版本的集群，部署好之后，他们都会注册过去，然后你依次手动调用老版本集群各个服务实例的下线接口，一般请求nacos/eureka的http接口，就可以把服务下线了，此时一旦老版本服务实例下线，那么流量就会全部给新版本的集群了

但是你老版本集群别停机，他仅仅是服务下线了而已，没请求了

如果一旦发现新版本集群有问题，再调用你的服务的注册接口，直接让各个服务主动发送http之类的请求到服务注册中心，把自己这个服务实例注册上去，把新版本集群全部手动下线一下

这么搞也是可以的，就是有开发成本

还有一种，就是说你新版本的集群直接部署，然后从存储到缓存到所有中间件，到数据库，到机器，全部是独立的一套，全新，全独立，如果需要以前的数据，可以半夜凌晨做一个停机维护，比如凌晨1点到5点，停机4个小时，把老系统所有数据都复制一份到新版本系统的数据存储里去

然后5点过后，简单控制一下流量分发入口，让每一个请求到老版本集群一份，也到新版本集群一份，这个其实好做，网关那块可以控制一下，然后新老集群一起跑，观察新集群是否稳定，如果稳定，再把老集群下线了，全部采用新系统对外提供服务