一般项目里面都会有很多的配置文件,如数据库连接信息、日志打印格式方式、属性文件、服务配置等等，在非分布式环境中常常一个项目部署一台机器即可，修改配置文件只需要在一台机器上面改一下；如果到了分布式项目一个服务可能就需要成千上万台机器，工作量巨大，而且系统的很多参数运行时还需要进行动态更新。

* 配置需要治理

SOA微服务需要治理，配置也需要治理。同一份应用程序在不同的环境（开发，测试，预发、生产）、不同的集群（如不同的数据中心）经常需要有不同的配置，所以需要有完善的环境管理

* 框架配置

虽然框架类组件是由架构组团队开发、维护，但是运行时是在业务实际应用内的，所以本质上可以认为框架类组件也是应用的一部分。

* 配置统一管理托管以及动态开关

黑白名单、灰度发布(**支持配置的灰度发布，比如点了发布后，只对部分应用实例生效，等观察一段时间没问题后再推给所有应用实例**)、AB测试、特殊情况留些内部的后门

国税、

财务中心、

安全配置:csrf配置

* 提供Java接入客户端
* 提供注解释编程或XML无代码侵入模式

极简的使用方式（注解式编程 或 XML无代码侵入模式）：我们追求的是极简的、用户编程体验良好的编程方式。目前支持两种开发模式：基于XML配置或者基于注解，即可完成复杂的配置分布式化

* 客户端配置信息监控

可以方便的看到配置在被哪些实例使用