## 监控

监控系统为SpringBoot工程，依赖了MQAdmin、zabbix-api、zabbix-sender。

HAService依赖了Curator选主，使用了LeaderLatch选主，主机会开启监控。

zk选主的原理是使用临时顺序节点，节点序号最小的成为主机，备机注册子节点变更事件。如果主机宕机，对应节点失效，触发所有备机的事件，节点最小的备机成为新的主机。

监控的本质是本地调度任务。

将监控划分为4大类：Broker、Cluster、Topic、ConsumerGroup，每个大类对应一个调度任务。

每个大类包含数量不等的小类别。比如所有Broker的监控项，会划分为3个小项，每个小项都实现固定的接口，工程启动时会获取所有这种接口类型的bean。每个小项要实现固定的方法，例如如何计算监控项，包含哪些监控项和主机等，采用了模板模式。这种设计方法好处是可扩展性好，增加监控只要实现接口，并注册为bean即可，且不影响其他监控。后面多次迭代使用了这个优势。

对于Broker主要监控Broker的运行状态，最核心的监控是：核心线程池活跃线程数、队列长度、队首等待时间；pageCache锁定时长；写消息耗时分布；还包括一些统计数据例如收发消息数、TPS等。

对于CLuster、Topic、ConsumerGroup他们的工作原理跟Broker一样，即模板模式。对于小项来说模板模式表示实现不同的监控逻辑，对于大类来说是创建不同的群组、模板、主机等。

依赖zabbix-api，实现需要的api方法，例如hostGet、hostCreate等。当发现有新增的Cluster、Broker、Topic、ConsumerGroup时，调用hostCreate，在zabbix上创建主机，将其加入模板。因此自动增加监控，无需手动配置。如果新增了监控项，会调用itemCreate，创建监控项。

zabbix-api的实现完全参考zabbix官方文档，拼装参数，使用httpClient发起请求。

每个调度周期，汇总一个大类的监控项列表，使用zabbix-sender发给zabbix，批量发送。zabbix-sender是开源代码，socket通信。

除了监控的调度任务，还有会定期从broker、namesrv上拉去集群拓扑结构、消费订阅关系等。

除了MQ原生的监控项，还经过聚合和二次计算衍生出更多监控项。集群监控就是聚合了集群所有Broker的收发消息数、TPS。衍生出集群读写均衡性、集群可用率。

另外还衍生出Topic路由信息变更监控、消费者群组消费均衡性监控等。

使用了Guava LoadingCache，缓存Broker运行状态、消费者状态。过期策略用的是refreshAfterWrite，过期自动从broker、namesrv获取。