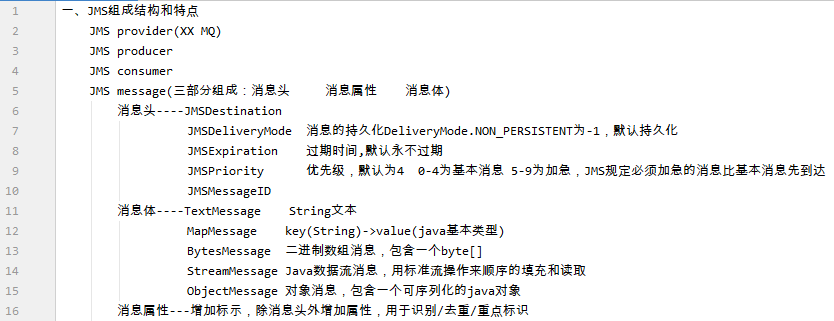
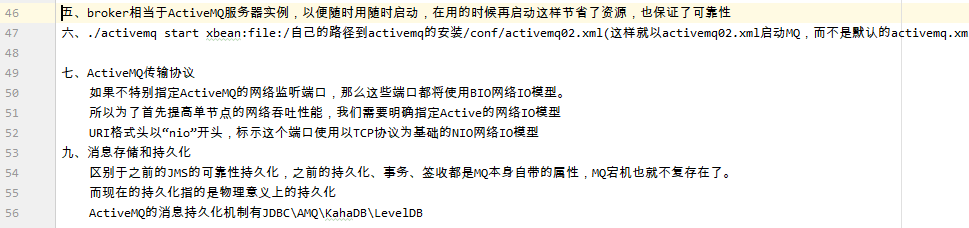
JMS-ActiveMQ参考代码：

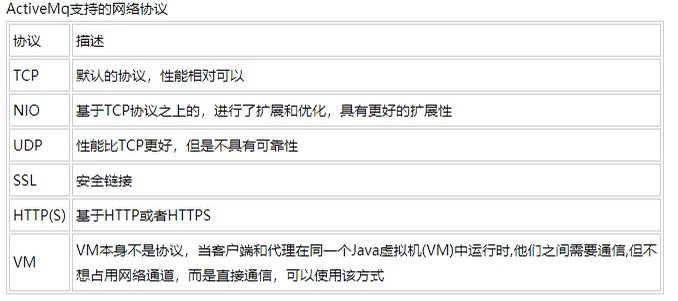
jmx(包含了spring 整合)和（boot-mq-produce/boot-mq-consumer）



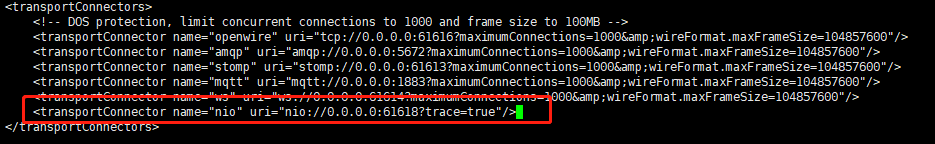




七、ActiveMQ传输协议

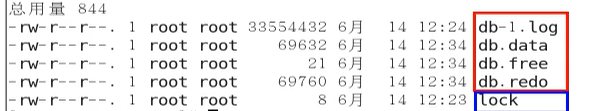


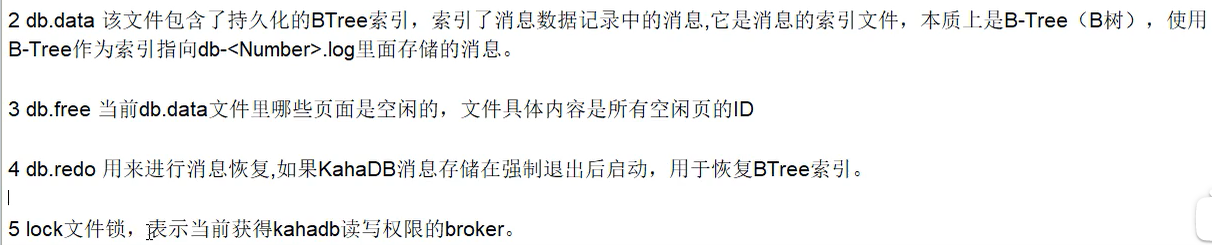
参考https://www.cnblogs.com/ffzzcommsoft/p/14799707.html



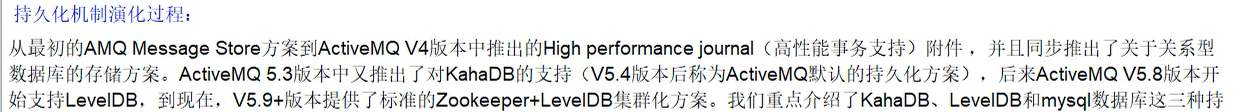


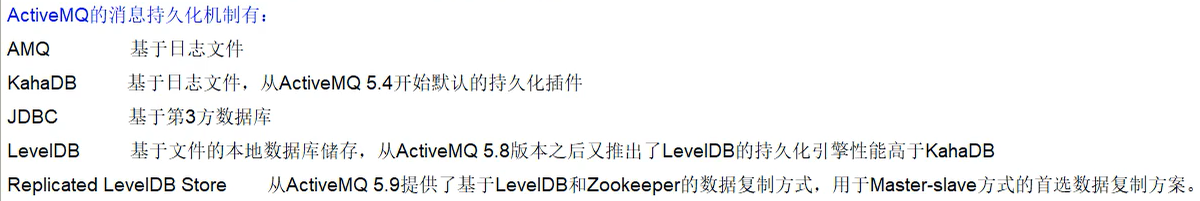
|  |
| --- |
| <transportConnector name="auto+nio" uri="auto+nio://0.0.0.0:61608?maximumConnections=1000&amp;wireFormat.maxFrameSize=104857600&amp;org.apache.activemq.transport.nio.SelectorManager.corePoolSize=20&amp;org.apache.activemq.transport.nio.Se1ectorManager.maximumPoo1Size=50"/> |

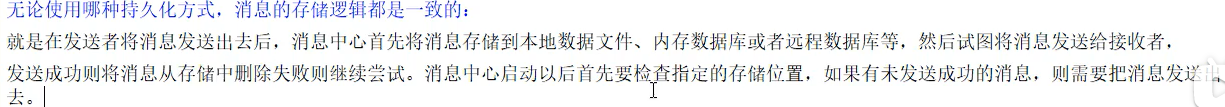






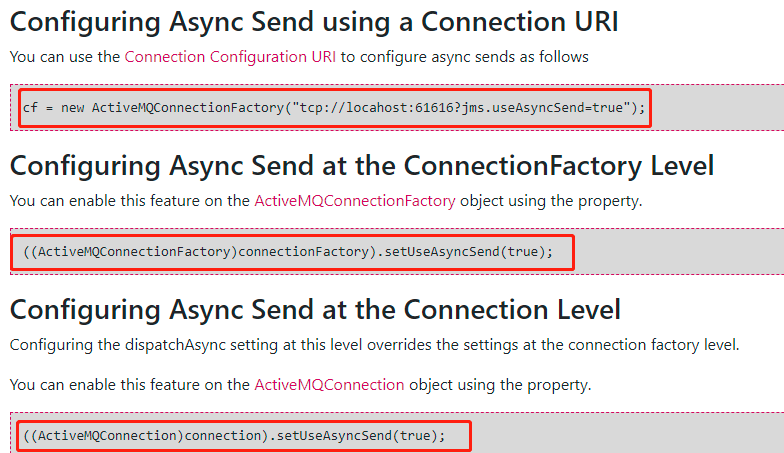


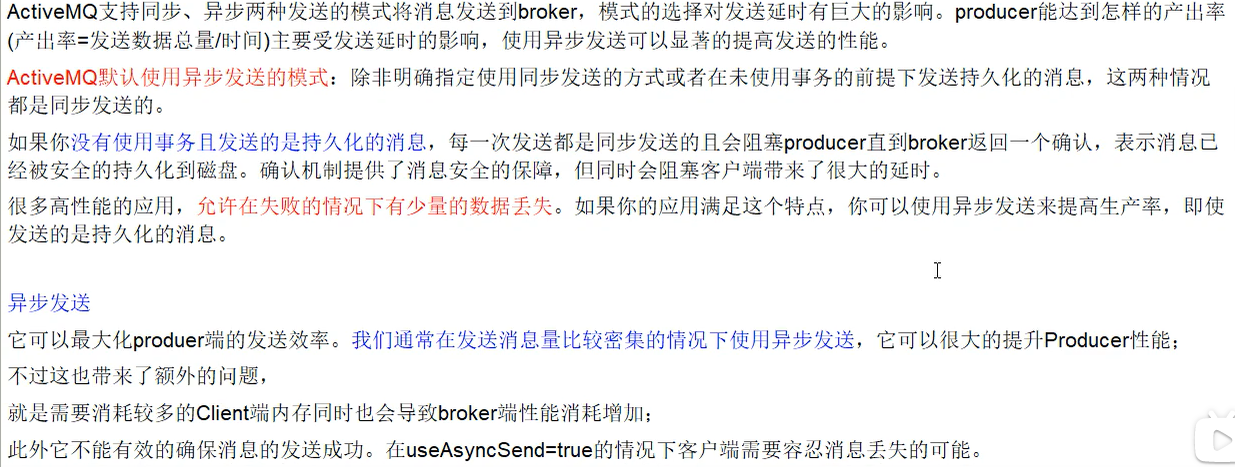




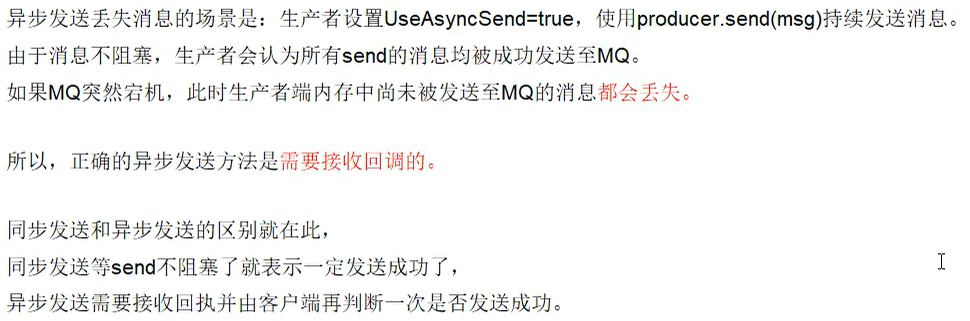
十、异步投递（面试题为什么开启异步投递、异步投递既然不能保证发送成功，那么怎么确认发送成功呢？）





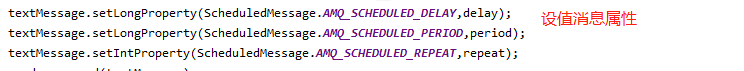


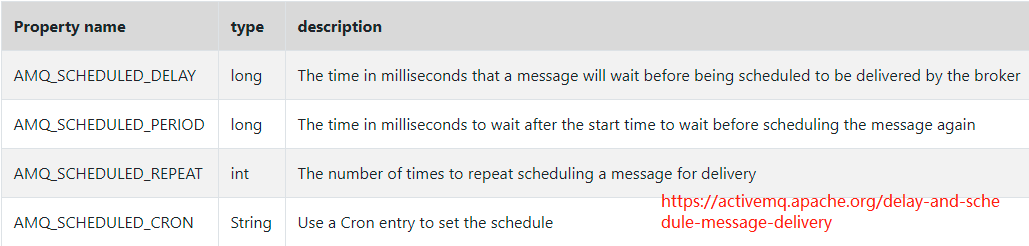
**====================发送消息的一种保证机制====================**



十一、延迟投递和定时投递



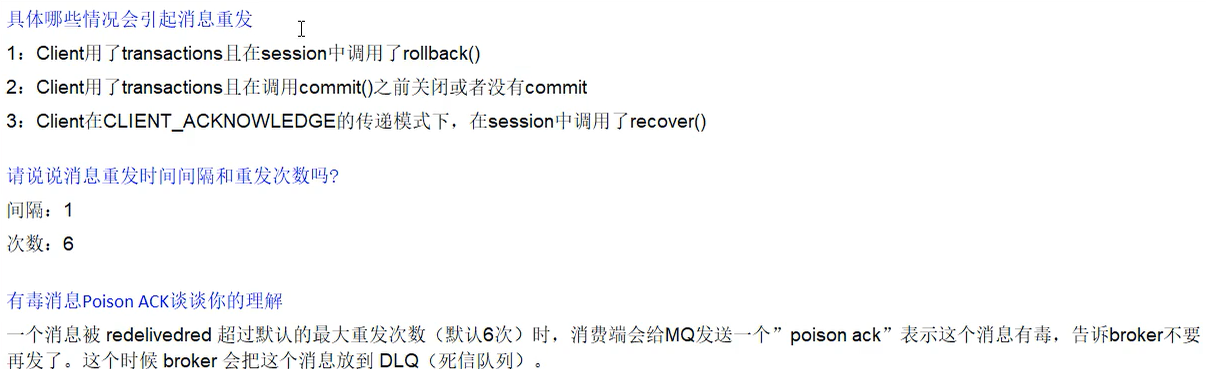
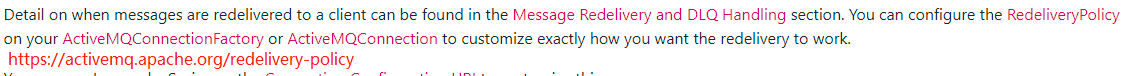


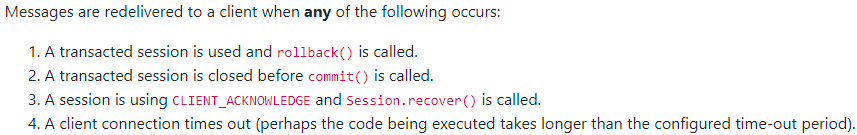


**启动activemq问题**



十二、Activemq的重试机制（触发条件）





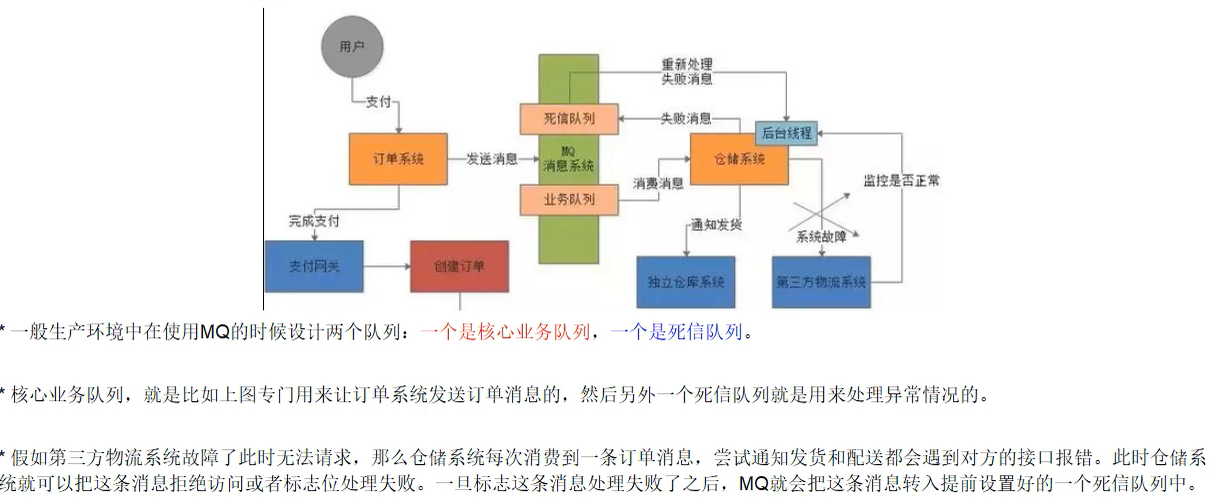
12.1、属性说明



**12.2Spring 整合重发机制**



12.3、死信队列



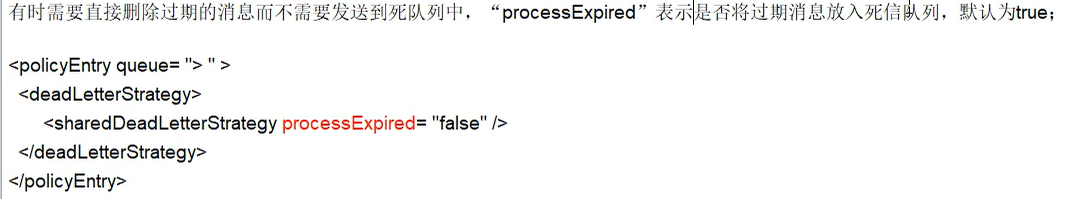
12.3.1、SharedDeadLetterStrategy（默认的）



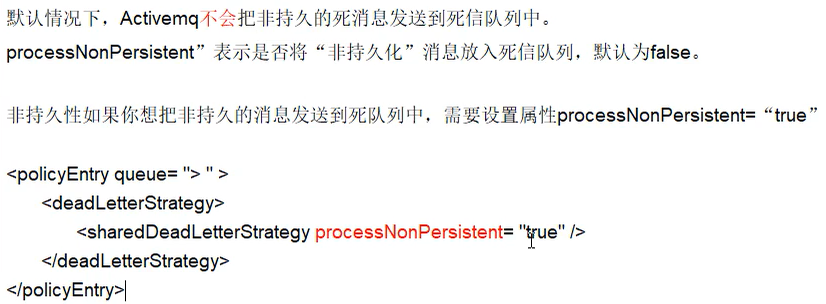
12.3.2、IndividualDeadLetterStrategy(独立的/可定制的)



12.3.3、自动删除过期案例



12.3.4、存放非持久的消息到死信队列



12.3.5、配置

12.4、如何避免消息重复消费，解决幂等性问题

