



producer如何保证消息不丢失

1. 默认情况下，可以通过同步的方式阻塞式的发送，check SendStatus，状态是OK，表示消息一定成功的投递到了Broker，状态超时或者失败，则会触发默认的2次重试。此方法的发送结果，可能Broker存储成功了，也可能没成功
2. 采取事务消息的投递方式，并不能保证消息100%投递成功到了Broker，但是如果消息发送Ack失败的话，此消息会存储在CommitLog当中，但是对ConsumerQueue是不可见的。可以在日志中查看到这条异常的消息，严格意义上来讲，也并没有完全丢失
3. RocketMQ支持 日志的索引，如果一条消息发送之后超时，也可以通过查询日志的API，来check是否在Broker存储成功

broker如何保证消息不丢失

1. 消息支持持久化到Commitlog里面，即使宕机后重启，未消费的消息也是可以加载出来的
2. Broker自身支持同步刷盘、异步刷盘的策略，可以保证接收到的消息一定存储在本地的内存中
3. Broker集群支持 1主N从的策略，支持同步复制和异步复制的方式，同步复制可以保证即使Master 磁盘崩溃，消息仍然不会丢失

consumer如何保证消息不丢失

1. Consumer自身维护一个持久化的offset（对应MessageQueue里面的min offset），标记已经成功消费或者已经成功发回到broker的消息下标
2. 如果Consumer消费失败，那么它会把这个消息发回给Broker，发回成功后，再更新自己的offset
3. 如果Consumer消费失败，发回给broker时，broker挂掉了，那么Consumer会定时重试这个操作
4. 如果Consumer和broker一起挂了，消息也不会丢失，因为consumer 里面的offset是定时持久化的，重启之后，继续拉取offset之前的消息到本地