**14-9 直播带货之库存扣减方案**

**背景**

在直播带货场景中，用户可能会在带货过程中出现抢购情况。例如10000个在线观看直播的用户，于同一时刻参与到抢购行为当中。而在我们的商品下单流程中，库存是一个非常重要的因素。

通常来说，我们建议在以下几个步骤进行库存交验：

* 加入购物车的时候（真正有购买行为的产生）
* 生成待订单的时候（库存校验，时间间隔比较大，这批货物是可以购买的）

一般来说，大多数情况都是选择在生成待支付订单的时候才会预先扣减库存。而简单的加入购物车其实用户并没有算做是真实购买行为，所以只需要交验基本库存即可。

* Mysql 2core 4gb的机器，写流量，抗500+tps，读1000+qps（极限）（支撑的并发度不高，成本很高）
* Redis缓存数据库（读1w+ qps，机器配置高，5w+ qps，redis cluster ）
* 顺序写数据库，写日志类型的方案（redo log）

**基于Redis + MySQL + MQ的方案**

抢购活动组成：活动开始前宣传(主播在准备参与抢购活动之前，需要先点击预热按钮) -> 抢购 -> 抢购结束

将库存的数目存在于redis中，依靠decr指令去扣减库存。

* 被动同步：定时同步，定时任务，每隔一段时间，将我们的库存同步到db中
* 主动同步：每次访问redis之后，异步线程刷新同步到db中，对db的压力会比较大

同时需要依靠一个定时任务，定时将redis中的库存数目同步到mysql表中。但是这里需要注意以下几点：

* Redis是内存数据库，内存占用到一定阈值的时候，可能会回收部分数据。
* 不一定要等到ttl清零才回收Redis，有些回收策略，会对ttl进行判断
* 单独拉一台Redis也是可以的
* 库存存入Redis后的有效期要足够长，避免说这份缓存数据在抢购活动还没结束前被回收了。（甚至可以考虑单独拉一台redis，然后不设置过期时间，等带货高峰期过后，全部清空）
* 库存扣减前需要先校验是否充足，然后再记录扣减库存流水，最后才执行库存扣减（这里有多元操作，可以考虑使用Lua脚本）（流水id 临时记录在redis的string中，重复请求）
* 最后库存扣减成功后，需要发送RMQ去告知下游系统，例如执行支付倒计时，系统通知，购物车清空等操作
* 支付成功，利用支付中台（qiyu-live-bank）的RMQ通知下游系统，订单状态修改，商品发货（物流系统打交道），系统通知等操作

**库存回滚问题**

如果购物车下单后，一直处于待支付状态，那么就会导致我们的库存一直被锁住了，所以通常来说，这里会做一层回滚的策略。

**定时任务回滚**

需要将处于待支付状态的信息记录到一张表中，表中需要记录订单的生成时间，然后由一个定时任务每隔15秒扫描一次，进行校验，如果订单超时则修改其状态，回滚对应的库存。

**基于延时消息**

RMQ本身基于时间轮做了一套延时队列，我们可以利用该特效去实现，下单30min后消息反向通知的功能。