

图文 79 在RocketMQ中，如何解决订单数据库同步的消息乱序问题？

426 人次阅读 2020-01-15 07:00:00

详情 评论



狸猫技术

进店逛

相关频道



从 0 开
件件实站
已更新9

在RocketMQ中，如何解决订单数据库同步的消息乱序问题？



继《从零开始带你成为JVM实战高手》后，救火队长携新作再度出山，重磅推荐：

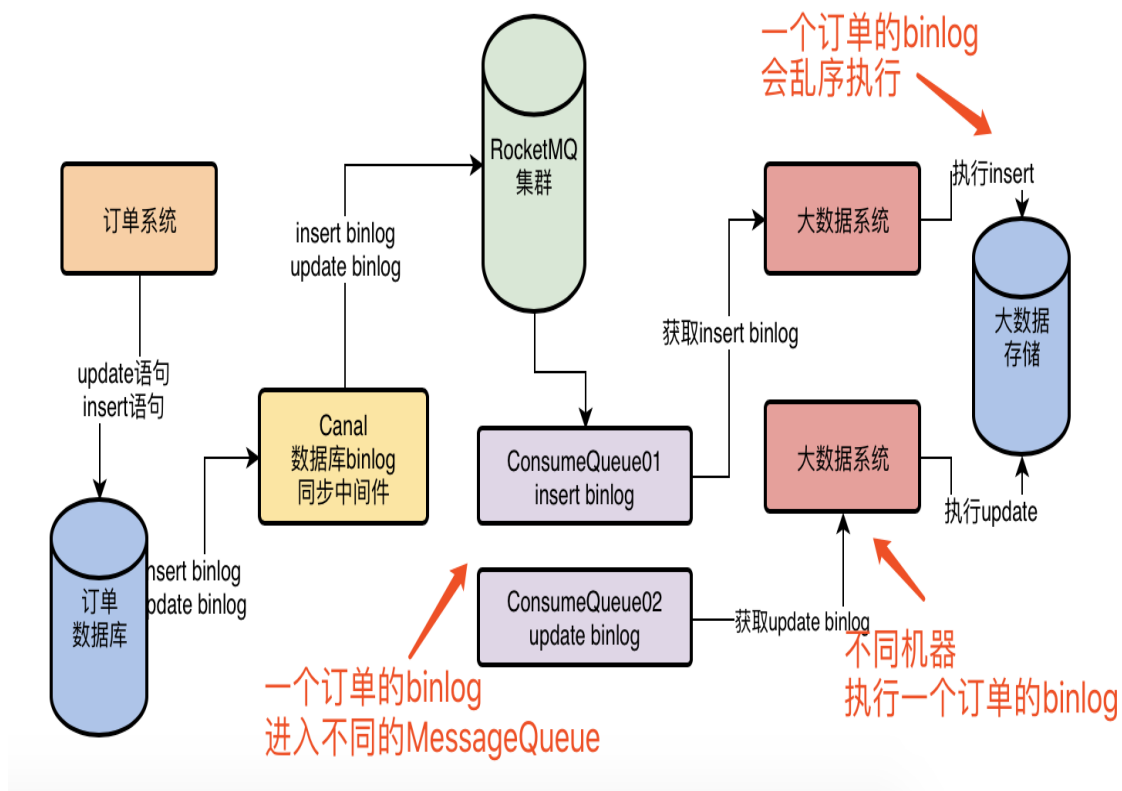
(点击下方蓝字试听)

[《从零开始带你成为MySQL实战优化高手》](#)

1、RocketMQ中消息乱序的原因回顾

上一次我们已经分析过订单数据库同步过程中的消息乱序问题产生的根本原因了，其实最关键的是，属于同一个订单的binlog进入了不同的MessageQueue，进而导致一个订单的binlog被不同机器上的Consumer来获取和处理，那么如果是这样的话，必然会导致这一个订单的binlog会乱序执行。

我们再看一下下面的图，来回顾一下这个消息乱序的原因。



2、让属于同一个订单的binlog进入一个MessageQueue

所以要解决这个消息的乱序问题，最根本的方法其实非常简单，就是得想办法让一个订单的binlog进入到一个MessageQueue里去。

给大家举个例子，比如对一个订单，我们先后执行了insert、update两条SQL语句，也就对应了2个binlog。

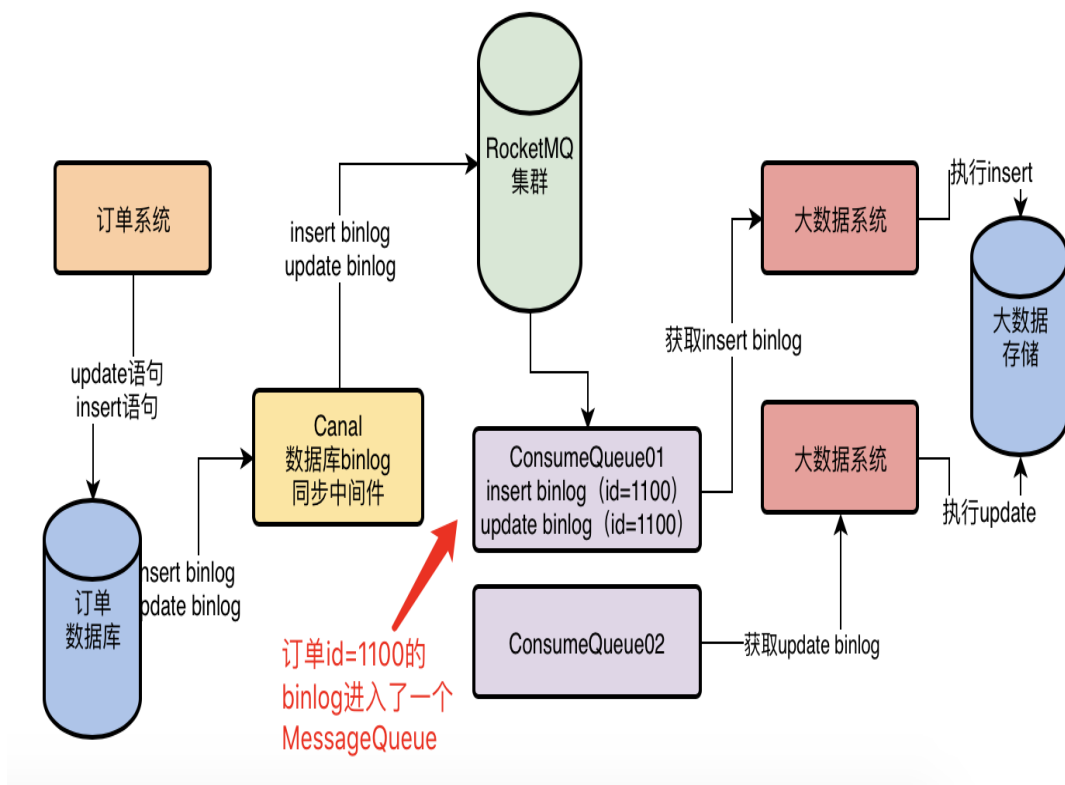
那么我们现在就必须要想办法让这个订单的2个binlog都直接进入Topic下的一个MessageQueue里去。

那么我们这个时候应该怎么做呢？我们完全可以根据订单id来进行判断，我们可以往MQ里发送binlog的时候，根据订单id来判断一下，如果订单id相同，你必须保证他进入同一个MessageQueue。

我们这里可以采用取模的方法，比如有一个订单id是1100，那么他可能有2个binlog，对这两个binlog，我们必须要用订单id=1100对MessageQueue的数量进行取模，比如MessageQueue一共有15个，那么此时订单id=1100对15取模，就是5

也就是说，凡是订单id=1100的binlog，都应该进入位置为5的MessageQueue中去！

通过这个方法，我们就可以让一个订单的binlog都按照顺序进入到一个MessageQueue中去，看下面的图：



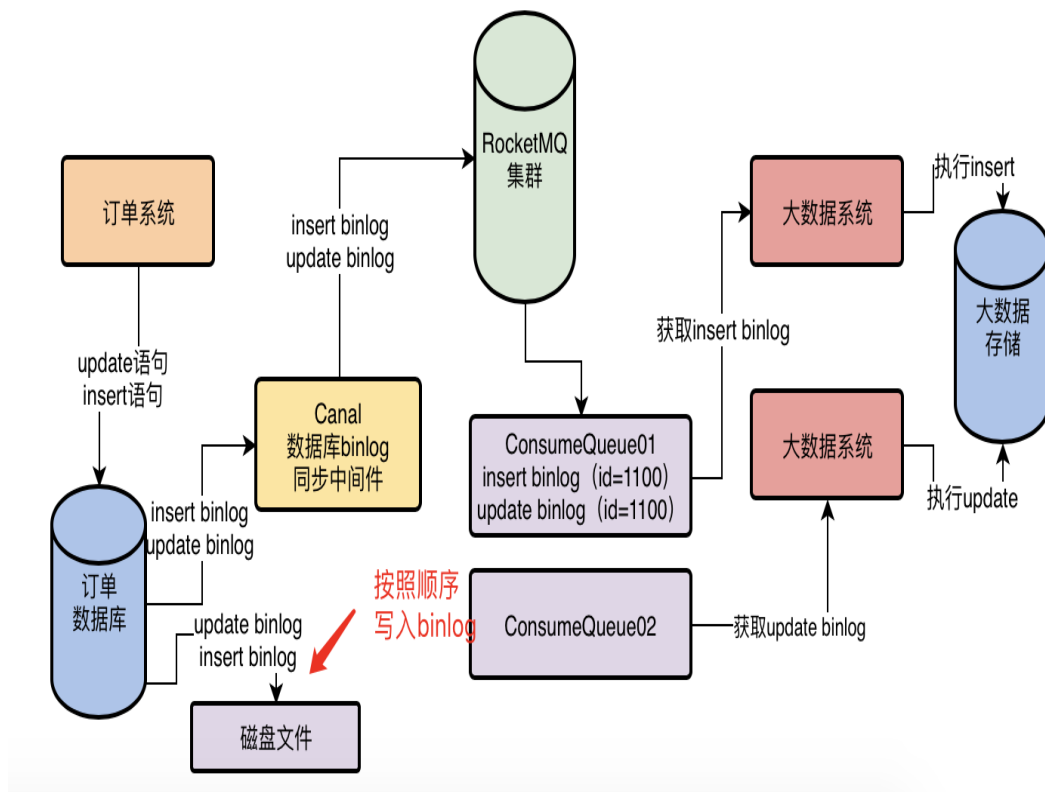
3、真的这么简单吗？获取binlog的时候也得有序！

我们来思考一下，真的就上面说的那么简单，只要一个订单的binlog都进入一个MessageQueue就搞定这个问题了吗？

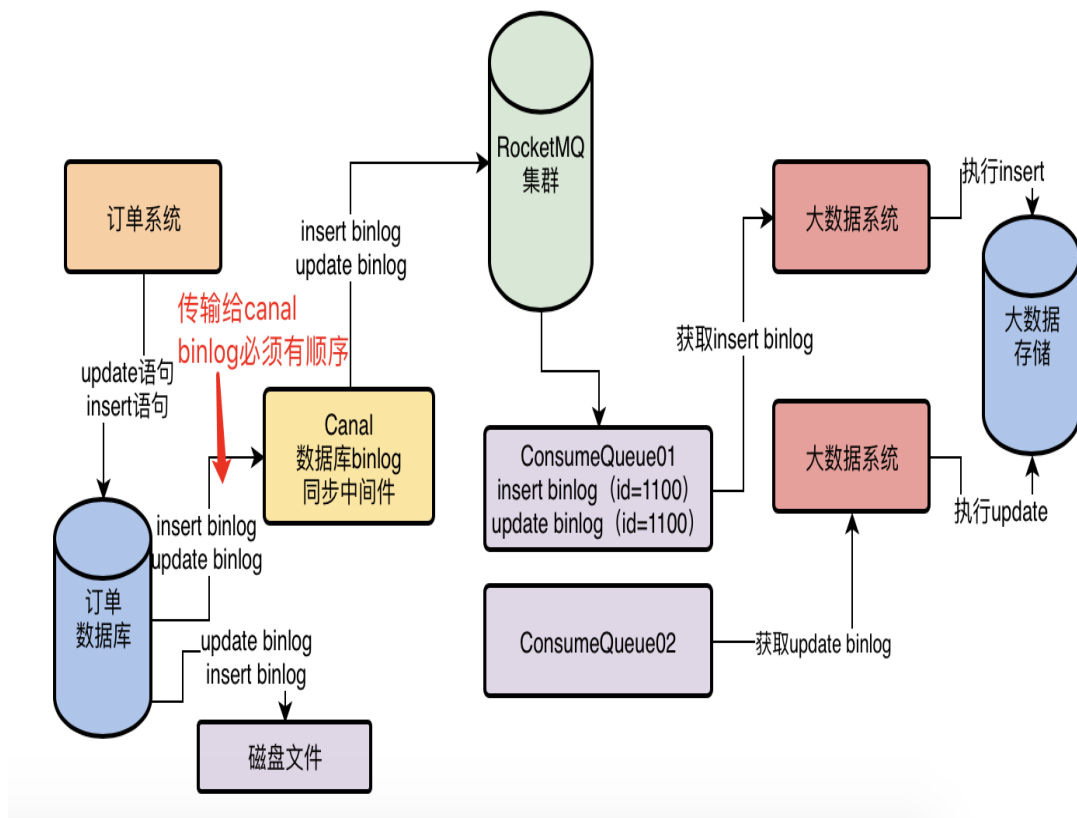
显然不是！

我们要考虑到一个问题，首先，我们的MySQL数据库的binlog一定都是有顺序的。

比如，订单系统对订单数据库执行了两条SQL，先是insert语句，然后是update语句。那么此时MySQL数据库自己必然是在磁盘文件里按照顺序先写入insert语句的binlog，然后写入update语句的binlog，如下图所示：

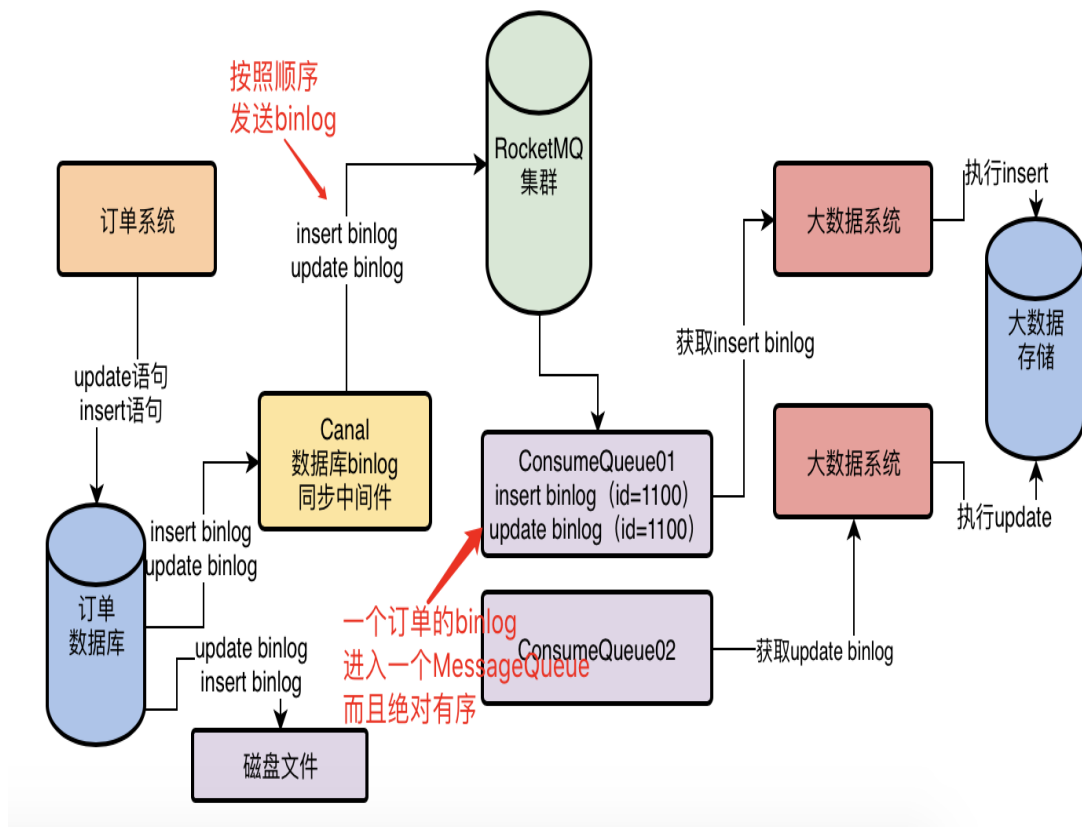


当我们从MySQL数据库中获取他的binlog的时候，此时也必须是按照binlog的顺序来获取的，也就是说比如Canal作为一个中间件从MySQL那里监听和获取binlog，那么当binlog传输到Canal的时候，也必然是有先后顺序的，先是insert binlog，然后是update binlog，如下图所示。



接着我们将binlog发送给MQ的时候，必须将一个订单的binlog都发送到一个MessageQueue里去，而且发送过去的时候，也必须是严格按照顺序来发送的

只有这样，最终才能让一个订单的binlog进入同一个MessageQueue，而且还是有序的，如下图所示：

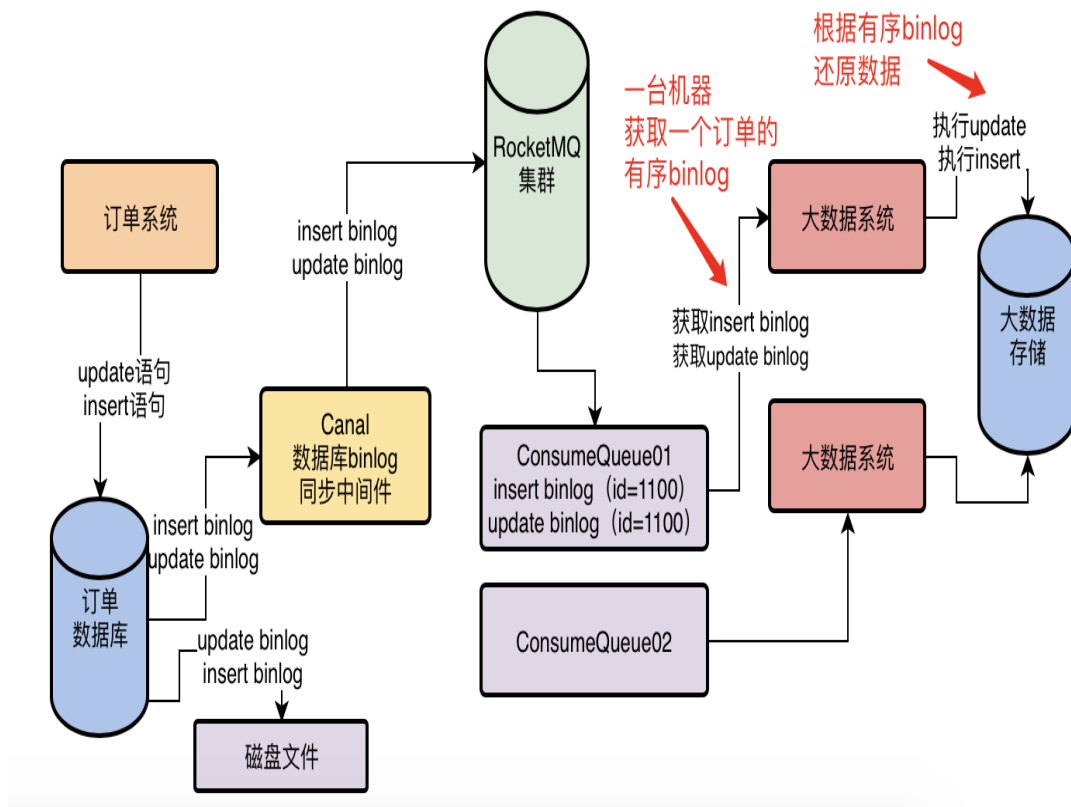


所以我们必须要严格做到以上几点，才能保证一个订单的binlog绝对有序的进入一个MessageQueue中。

4、Consumer有序处理一个订单的binlog

接着我们可以想一下，一个Consumer可以处理多个MessageQueue的消息，但是一个MessageQueue只能交给一个Consumer来进行处理，所以一个订单的binlog只会有序的交给一个Consumer来进行处理！

我们看下图，这样的话一个大数据系统就可以获取到一个订单的有序的binlog，然后有序的根据binlog把数据还原到自己的存储中去。



5、这就完了吗？没有，万一消息处理失败了可以走重试队列吗？

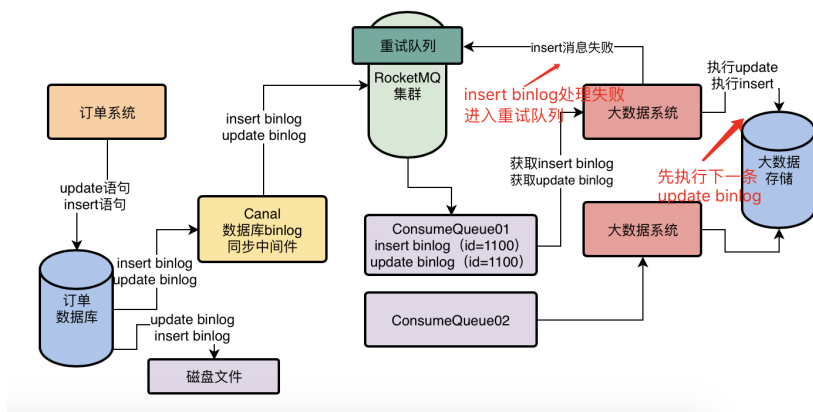
那么大家觉得这样就完了吗？

绝对不是，我之前给大家讲过，在Consumer处理消息的时候，可能会因为底层存储挂了导致消息处理失败，之前我们说过，此时可以返回RECONSUME_LATER状态，然后broker会过一会儿自动给我们重试。

但是这个方案用在我们的有序消息中可以吗？

那绝对是不行的，因为如果你的consumer获取到订单的一个insert binlog，结果处理失败了，此时返回了RECONSUME_LATER，那么这条消息会进入重试队列，过一会儿才会交给你重试。

但是此时broker会直接把下一条消息，也就是这个订单的update binlog交给你来处理，此时万一你执行成功了，就根本没有数据可以更新！又会出现消息乱序的问题，我们看下图



所以对于有序消息的方案中，如果你遇到消息处理失败的场景，就必须返回SUSPEND_CURRENT_QUEUE_A_MOMENT这个状态，意思是先等一会儿，一会儿再继续处理这批消息，而不能把这批消息放入重试队列去，然后直接处理下一批消息。

6、有序消息方案与其他消息方案的结合

如果你一定要求消息是有序的，那么必须得用上述的有序消息方案，同时对这个方案，如果你要确保消息不丢失，那么可以和消息零丢失方案结合起来，如果你要避免消息重复处理，还需要在消费者那里处理消息的时候，去看一下，消息如果已经存在就不能重复插入，等等。

同时还需要设计自己的消息处理失败的方案，也就是不能进入重试队列，而是暂停等待一会儿，继续处理这批消息。

7、作业：为你的系统设计一个多种方案结合的顺序消息方案

假设你的系统里有需要保证消息顺序性的场景，那么此时如果你要保证消息的顺序性，就必然要上消息顺序方案，此时可能还需要结合其他的消息方案

所以你可以思考一下，在你的业务场景中，应该如何针对你的业务设计多种方案结合的消息方案出来。

希望大家踊跃分享在评论区跟其他朋友一起交流。

End

专栏版权归公众号**狸猫技术窝**所有

未经许可不得传播，如有侵权将追究法律责任

狸猫技术窝精品专栏及课程推荐：

[《从零开始带你成为JVM实战高手》](#)
[《21天互联网Java进阶面试训练营》（分布式篇）](#)
[《互联网Java工程师面试突击》（第1季）](#)
[《互联网Java工程师面试突击》（第3季）](#)

重要说明：

如何提问：每篇文章都有评论区，大家可以尽情留言提问，我会逐一答疑

如何加群：购买狸猫技术窝专栏的小伙伴都可以加入狸猫技术交流群，一个非常纯粹的技术交流的地方

具体加群方式，请参见目录菜单下的文档：《付费用户如何加群》（**购买后可见**）