

复盘今天被问到的几个问题

来自： 小6互联网求职面试

马听

2023年11月29日 23:02



扫码加入
查看更多优质内容

今天，有几个课程学员问了几个MySQL相关的问题

晚上复盘了一下，就来跟大家一起分享交流下。

问题一

线上有张表数据100多万，700多M，但是磁盘碎片有4个T，这种能直接drop掉吗？还是有什么方式可以删掉？

不能直接删除，要当成4T的表处理，给数据文件.ibd创建一个硬链接，这样drop就很快了。

过程为：

```
ln table_name.ibd table_name.ibd.bak
```

在drop表

```
drop table table_name;
```

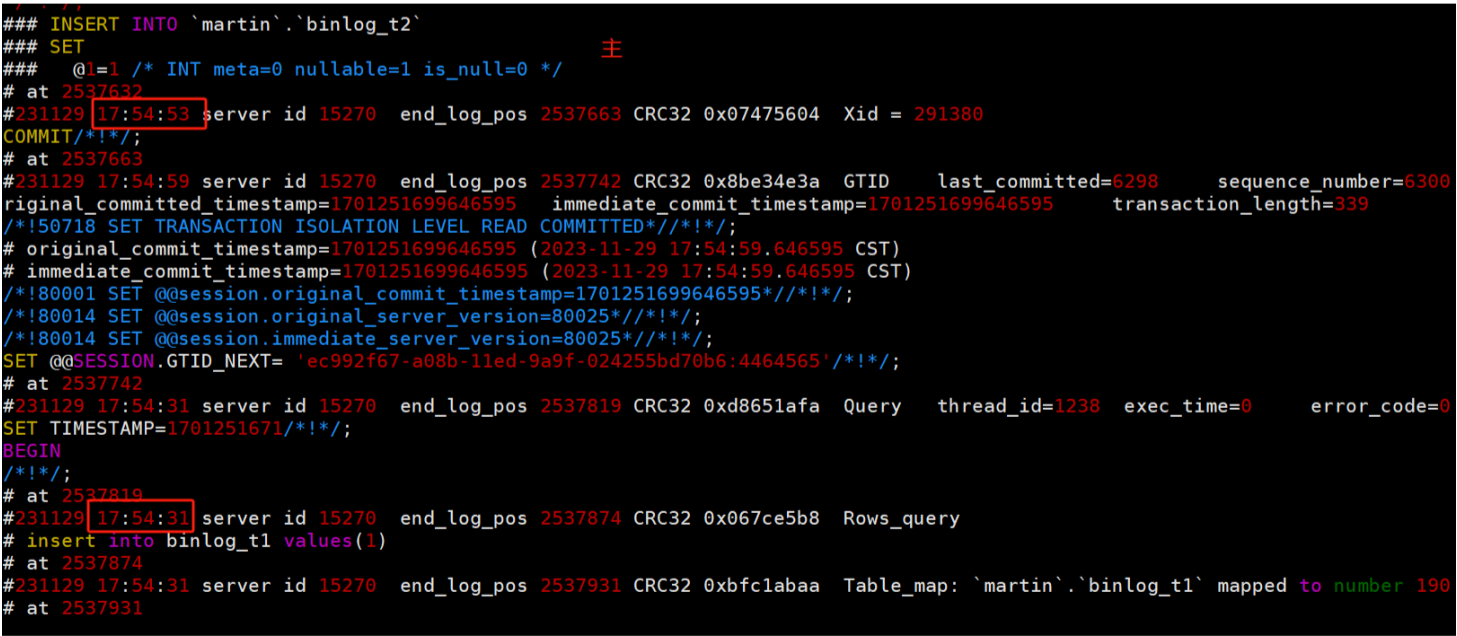
最后再删除物理文件

```
for i in `seq 4096 -1 1 ` ; do sleep 1; truncate -s ${i}G table_name.ibd.bak; done
rm -f table_name.ibd.bak
```

当然，如果是云上RDS，就需要找云厂商，研究怎么安全的删除，毕竟我们不能操作机器。

问题二

MySQL binlog里面的时间，有些不是顺序的。一般在什么情况下发生？是因为 主库 并发执行的事务原因吗？



（这张图不是学员的原图，是我自己做实验之后的截图，重点观察红框里面的时间点）

是的，应该就是主库并发执行的事务。

我做了一个这样的实验：

session1	session2
begin;	begin;
insert into binlog_t1 values(1);/* SQL1 */	
	insert into binlog_t2 values(1);/* SQL2 */
	commit;
commit;	

最终解析之后，就出现了上图这种情况，Binlog记录位置靠后的，时间点还靠前。

从库其实也一样的。

```
/*!*/;
### INSERT INTO `martin`.`binlog_t2`
### SET
### @1=1 /* INT meta=0 nullable=1 is_null=0 */
# at 450156292
#231129 17:54:50 server id 15270 end_log_pos 450156323 CRC32 0x46a5299b Xid = 11812021
COMMIT/*!*/;
# at 450156323
#231129 17:54:31 server id 15270 end_log_pos 450156409 CRC32 0xdefebaaa GTID last_committed=863434 sequence_number=1
yes original_committed_timestamp=1701251699646595 immediate_commit_timestamp=1701251699645563 transaction_length=341
/*!50718 SET TRANSACTION ISOLATION LEVEL READ COMMITTED*//*!*/;
# original_commit_timestamp=1701251699646595 (2023-11-29 17:54:59.646595 CST)
# immediate_commit_timestamp=1701251699645563 (2023-11-29 17:54:59.645563 CST)
/*!80001 SET @@session.original_commit_timestamp=1701251699646595*//*!*/;
/*!80014 SET @@session.original_server_version=80025*//*!*/;
/*!80014 SET @@session.immediate_server_version=80025*//*!*/;
SET @@SESSION.GTID_NEXT= 'ec992f67-a08b-11ed-9a9f-024255bd70b6:4464565'/*!*/;
# at 450156409
#231129 17:54:31 server id 15270 end_log_pos 450156481 CRC32 0xd5ec583b Query thread_id=1238 exec_time=28 error=0
SET TIMESTAMP=1701251671/*!*/;
BEGIN
/*!*/;
# at 450156481
#231129 17:54:31 server id 15270 end_log_pos 450156536 CRC32 0x19559671 Rows_query
# insert into binlog_t1 values(1)
# at 450156536
#231129 17:54:31 server id 15270 end_log_pos 450156593 CRC32 0x09fa63a7 Table_map: `martin`.`binlog_t1` mapped to number 1
# at 450156593
```

也就是并发事务，先执行SQL的，后面再提交（什么时候提交，什么时候记录Binlog），可能会出现时间在前面事务，Binlog位置在后面，看起来像乱序了。

问题三

keepalived+主从，如果发生了vip飘逸，那老的主启动，是不是不能同步数据了，这样会不会导致数据不一致？

一般建议是keepalived+双主，也就是两套数据库互为主备，这样及时切换了，反向还是同步的，那原来的主还是一直同步新主数据的。

但是一般只建议用一个VIP，也就是只在一个主上写，另外，如果要避免主键冲突，建议是主从跳主键的形式，比如主的主键 1 3 5，从的主键 2 4 6。