# 浅析binlog中的xid

1、xid是什么

XA是由X/Open组织提出的分布式事务的规范(X代表transaction; A代表accordant？)。XA规范主要定义了(全局)事务管理器(TM: Transaction Manager)和(局部)资源管理器(RM: Resource Manager)之间的接口。XA为了实现分布式事务，将事务的提交分成了两个阶段：也就是2PC (tow phase commit)，XA协议就是通过将事务的提交分为两个阶段来实现分布式事务，binlog中xid就是mysql内部XA事务的id。

2、xid的形成

pxc中xid形成方式与mysql原生的xid形成方式不一样，下面分别介绍。

1）首先介绍PXC，PXC中事务提交与参数repl.commit\_order有关，它的默认值是3，也就是NO\_OOOC(NO out of order commit),意思是串行提交，此时的condition是当前的GTID必须要比最新处理的GTID大1，在binlog文件中，看到的xid都是依次增长，步长是1。这样的好处是，所有节点的binlog内容都是一样的，如果集群下面再挂slave的话，它的主库down掉了，就可以根据xid来找到位置，继续复制，非常方便。即使进行了reset master操作，pxc中binlog中的xid也不会从头开始。

2）mysql binlog中的xid是一个内部自增序列，ulonglong类型，占用8字节。当执行reset master时，xid就会从头开始。mysql5.6引入了基于gtid的主从复制，主要是为了解决主从切换找不到position的问题，基于gtid的复制，主从库每一个事务的gtid值是相同的，所以可以直接切换。

3、xid在崩溃恢复时的作用：

MySQL两阶段提交(内部XA的两阶段提交)：

第一阶段：InnoDB prepare，持有prepare\_commit\_mutex，并且write/sync redo log； 将回滚段设置为Prepared状态，binlog不作任何操作；

第二阶段：包含两步，1> write/sync Binlog； 2> InnoDB commit (写入COMMIT标记后释放prepare\_commit\_mutex)；

以 binlog 的写入与否作为事务提交成功与否的标志，innodb commit标志并不是事务成功与否的标志。因为此时的事务崩溃恢复过程如下：

1> 崩溃恢复时，扫描最后一个Binlog文件，提取其中的xid；

2> InnoDB维持了状态为Prepare的事务链表，将这些事务的xid和Binlog中记录的xid做比较，如果在Binlog中存在，则提交，否则回滚事务。