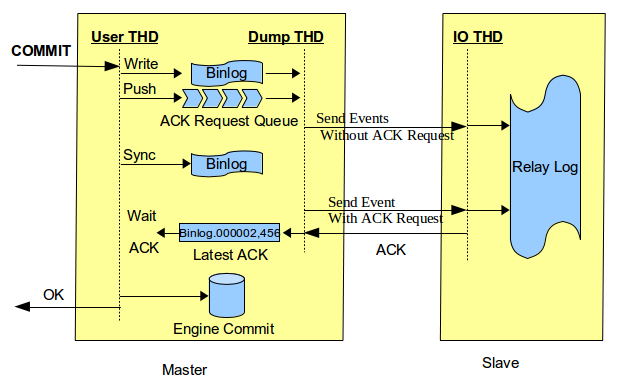
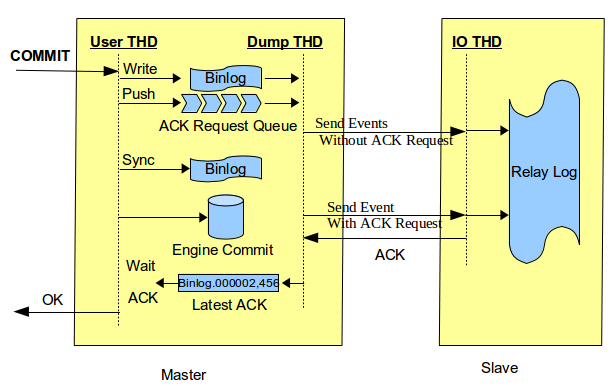
# MySQL57半同步复制简介

## 事务提交和Binlog同步

如下图所示，为MySQL5.7进行commit时，master和slave交互的流程



如下图所示，为MySQL5.6进行commit时，master和slave交互的流程

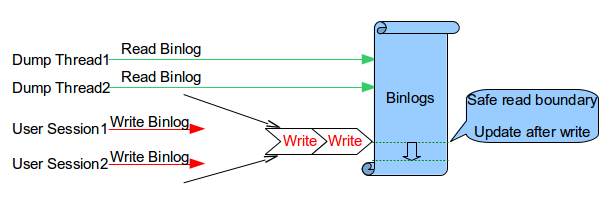


相较于MySQL5.6，5.7将发送binlog及等待slave返回，放在调用存储引擎commit接口之前，这样可以确保在master宕机的时候，commit的数据在slave上有保存，不会出现数据丢失的现象。

## 线程改进

### dump线程改进

下图是改进后的MySQL5.7的复制线程模型图

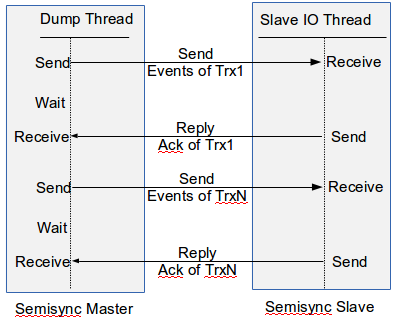


5.7之前对binlog的操作，不论是读还是写线程都通过LOCK\_log进行同步的，这让所有的操作都变成串行操作。但是，读取binlog并发送到slave的dump线程在读的时候，不需要对binlog进行上锁，导致写线程阻塞的。

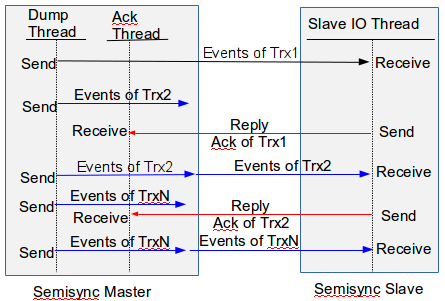
5.7通过设置safe read boundary来保护dump线程不读取无效数据，在读取中不进行lock，只有用户事务在写binlog的时候进行lock保护，提高写入的效率。

### 改进的master-slave ack交互

5.7.3的线程模型

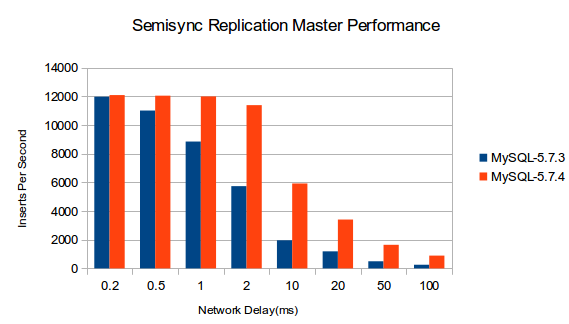


改进后的线程模型



相较于5.7.3的模型，5.7.4使用专门的线程处理slave返回的ack信息，这样dump线程可以不停的向slave发送binlog而不必阻塞在等待slave的ack过程，提高网络使用效率，这尤其在网络延迟较大的情况下，有助于提高binlog的发送效率。

性能对比



参考资料

http://my-replication-life.blogspot.sg/2013/09/loss-less-semi-synchronous-replication.html

http://my-replication-life.blogspot.com/2013/09/dump-thread-enhancement.html

<http://my-replication-life.blogspot.sg/2014/03/faster-semisync-replication.html>