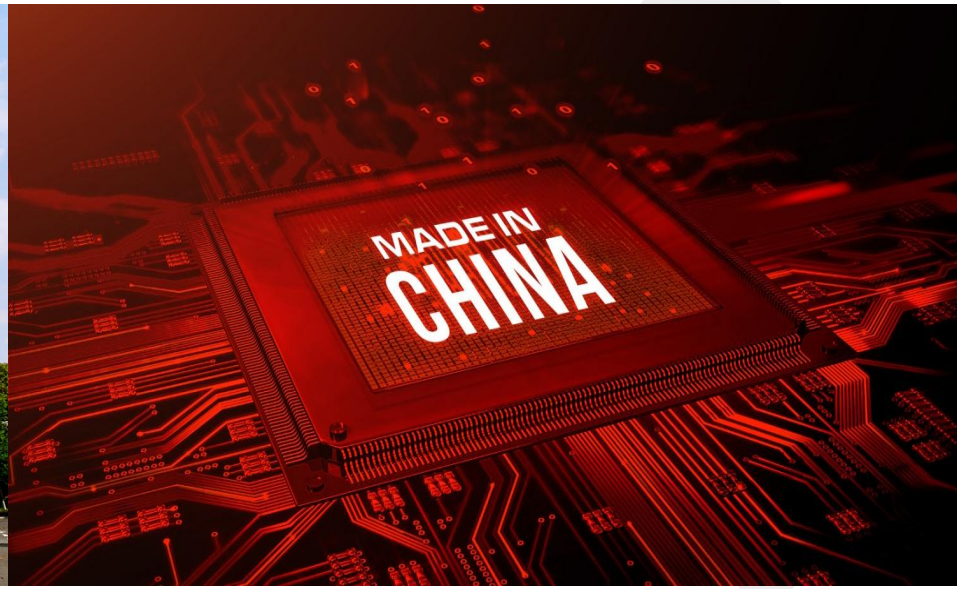
Great DB 万里数据库



MGR Bug修复之路

万里数据库CTO　娄帅

2021 年 3 月 20 日

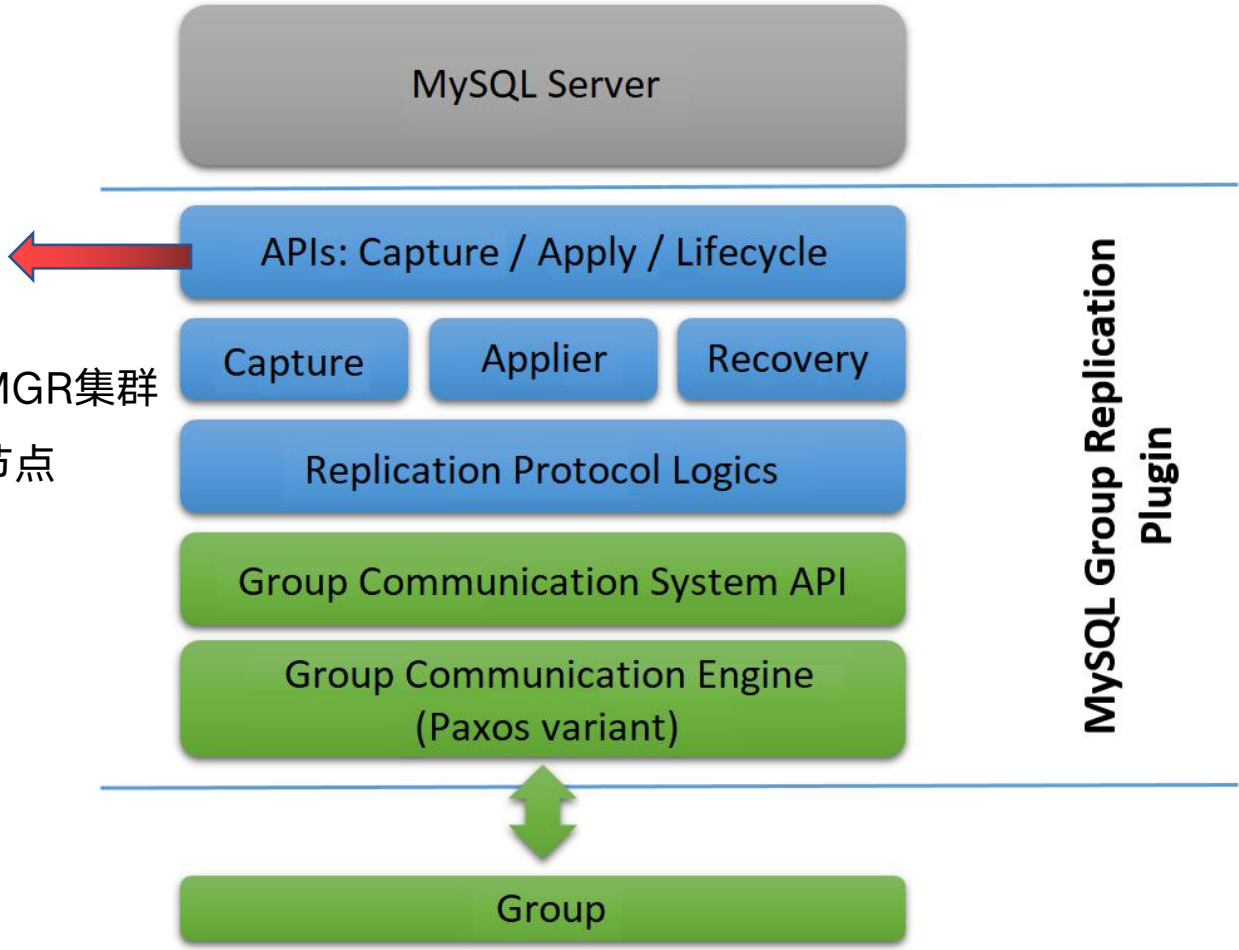
目录

Great DB 万里数据库

分层设计，保持接口和实现相对独立

server层与plugin的接口

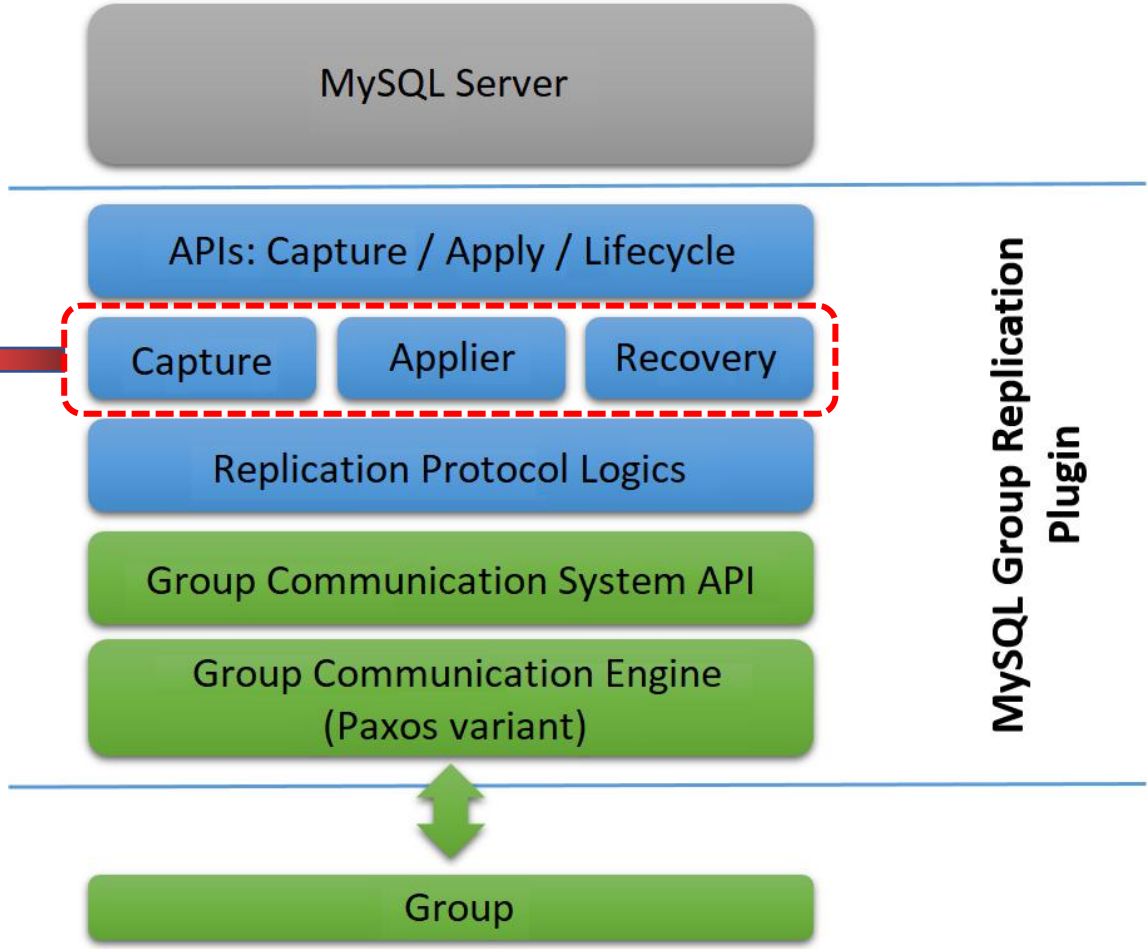
- server层状态信息传递
- 用户事务信息传递
  1. 本节点事务信息传递到MGR集群
  2. 其他节点事务应用到本节点



Great DB 万里数据库

分层设计，保持接口和实现相对独立

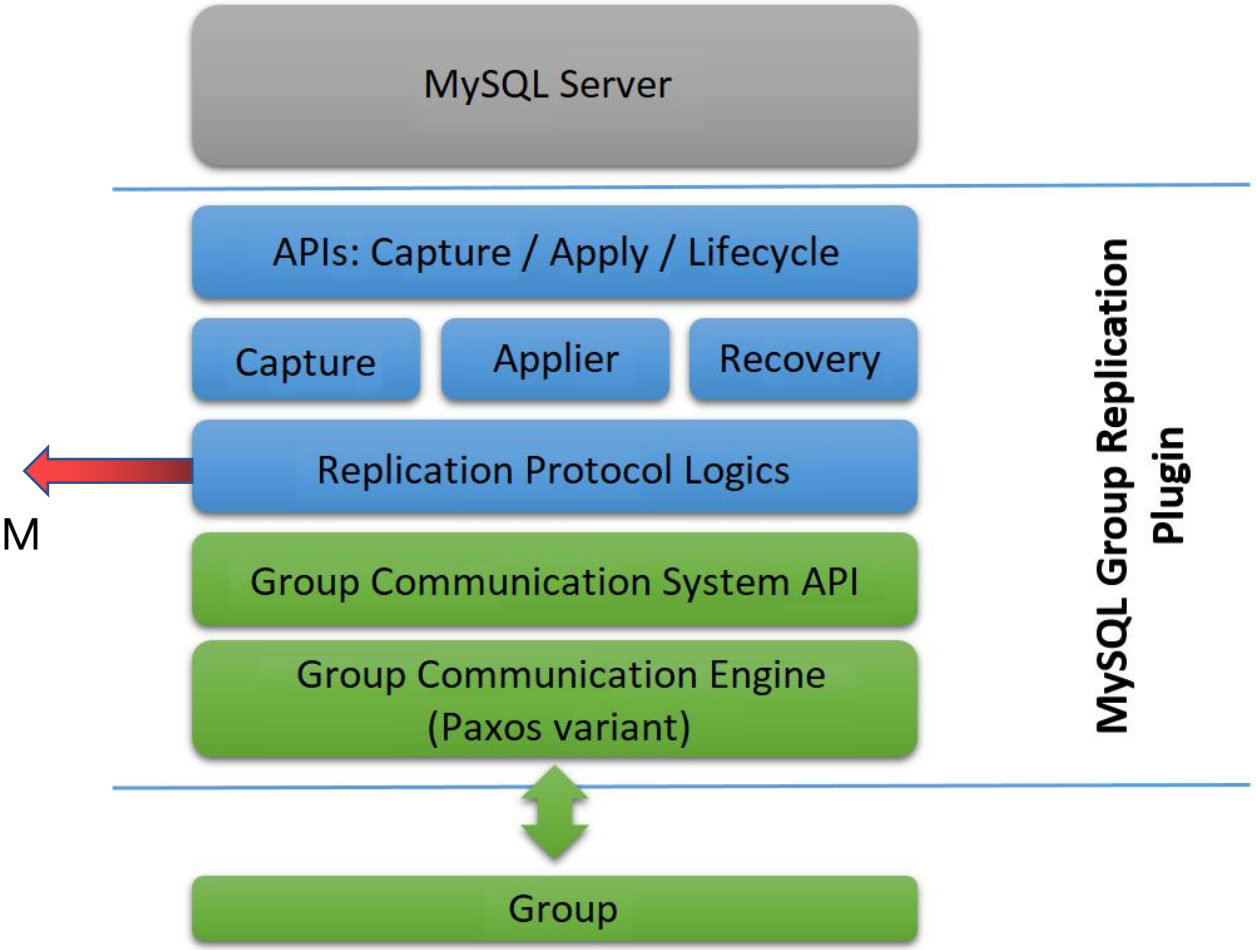
- **Capture:** 跟踪本节点的事务相关信息
- **Applier:** 执行其它节点的远程事务
- **Recovery:** 负责故障恢复时，选择donar节点，catch up binlog等



Great DB 万里数据库

分层设计，保持接口和实现相对独立

- **MGR协议逻辑**
  - 消息的封装
  - 接收XCOM返回的消息
  - 发送本节点消息给XCOM
  - 冲突检测等

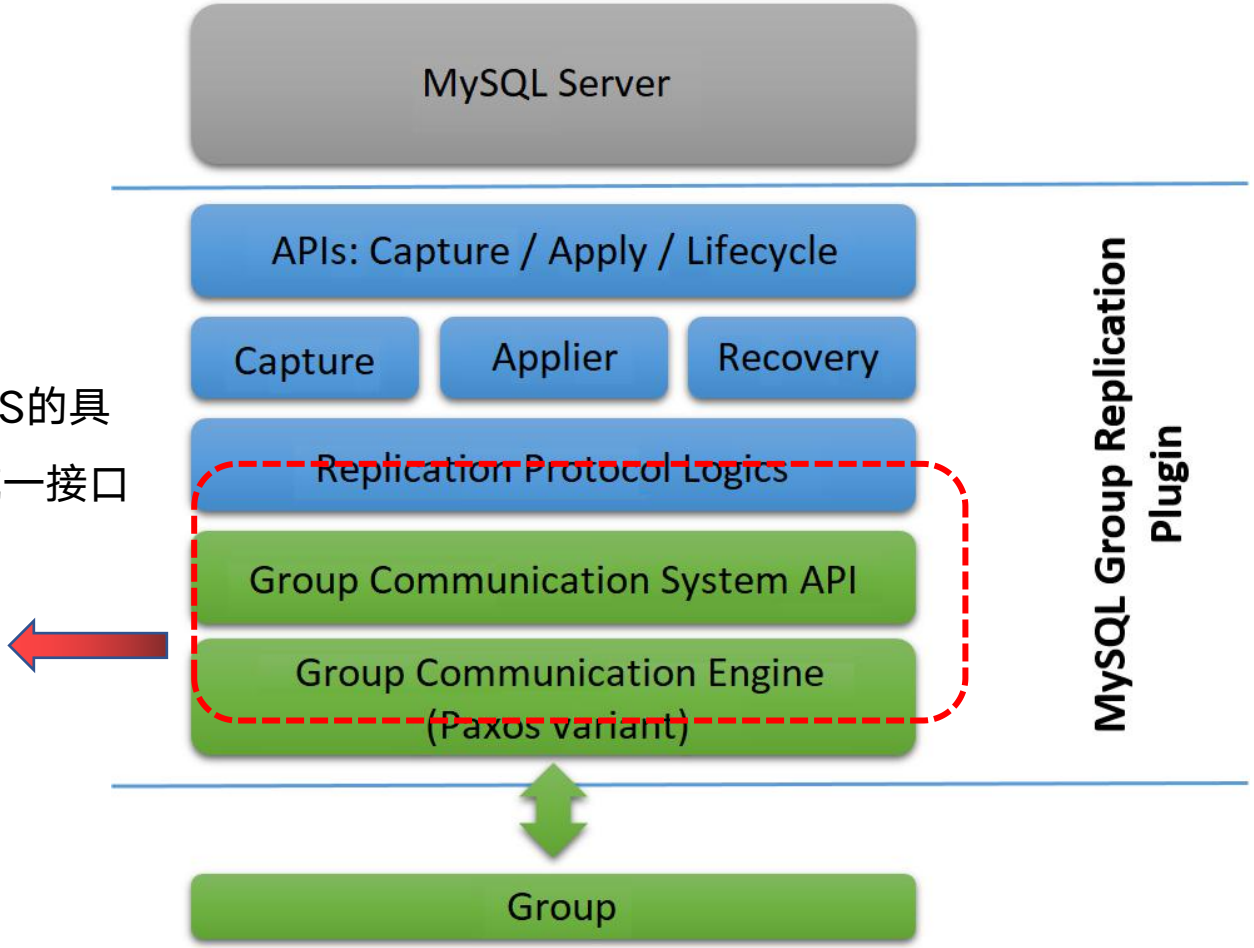


Group Replication Plugin Architecture

Great DB 万里数据库

分层设计，保持接口和实现相对独立

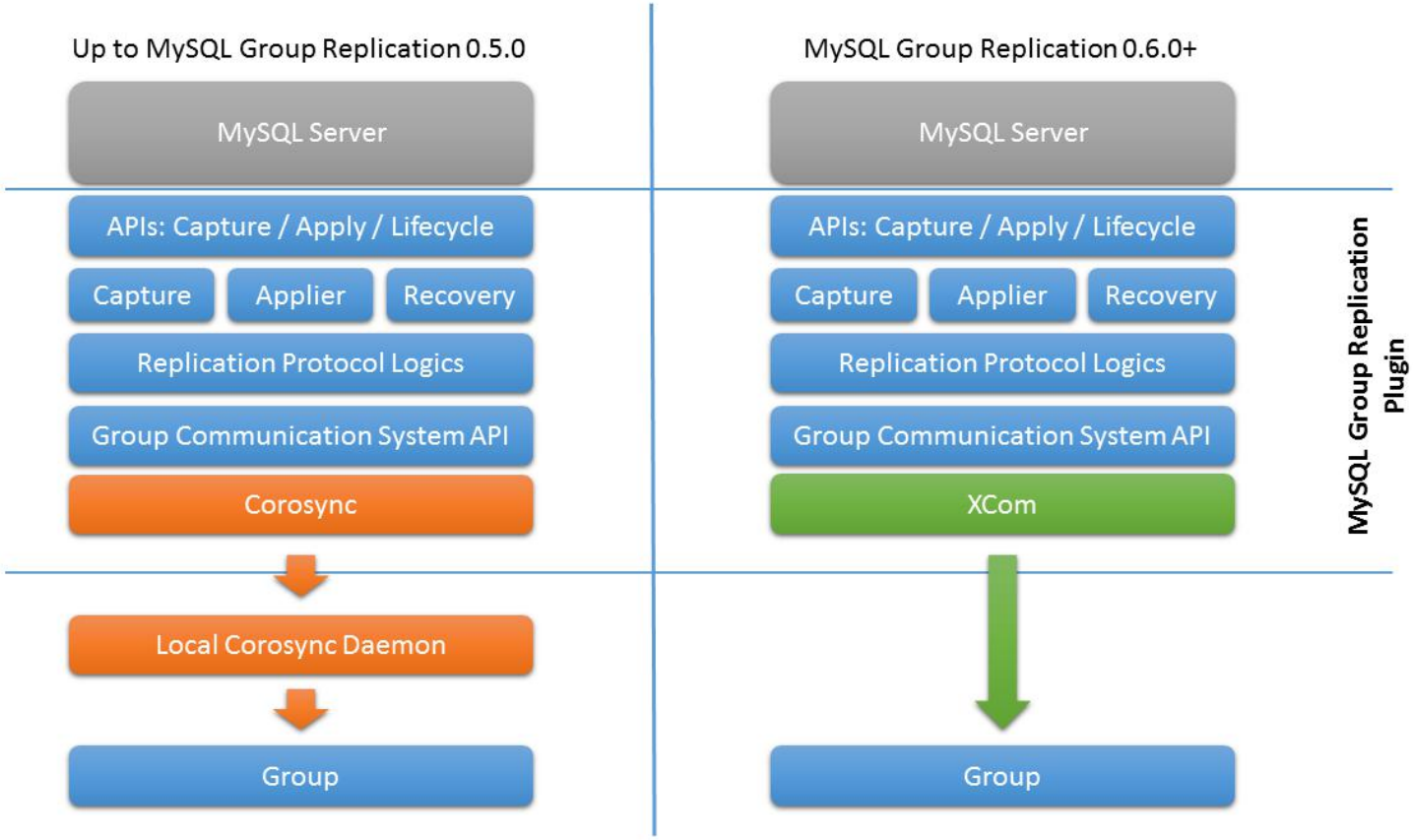
- **GCS Interface:** 定义了GCS的具体实现与处理逻辑之间的统一接口
- **GCE:** GCS的具体实现，Xcom(eXtended COMmunications)



Group Replication Plugin Architecture

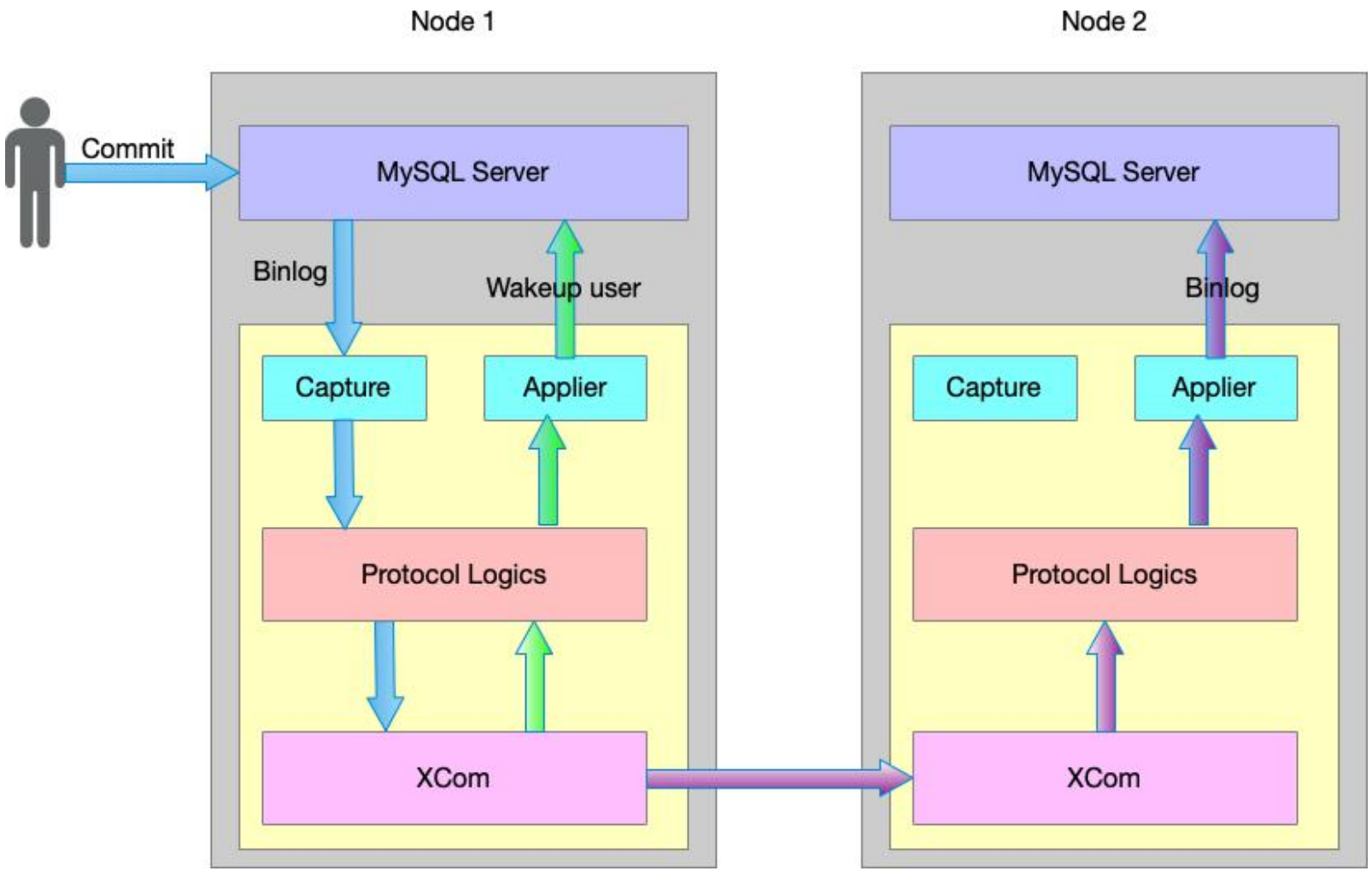
Great DB 万里数据库

分层设计，保持接口和实现相对独立

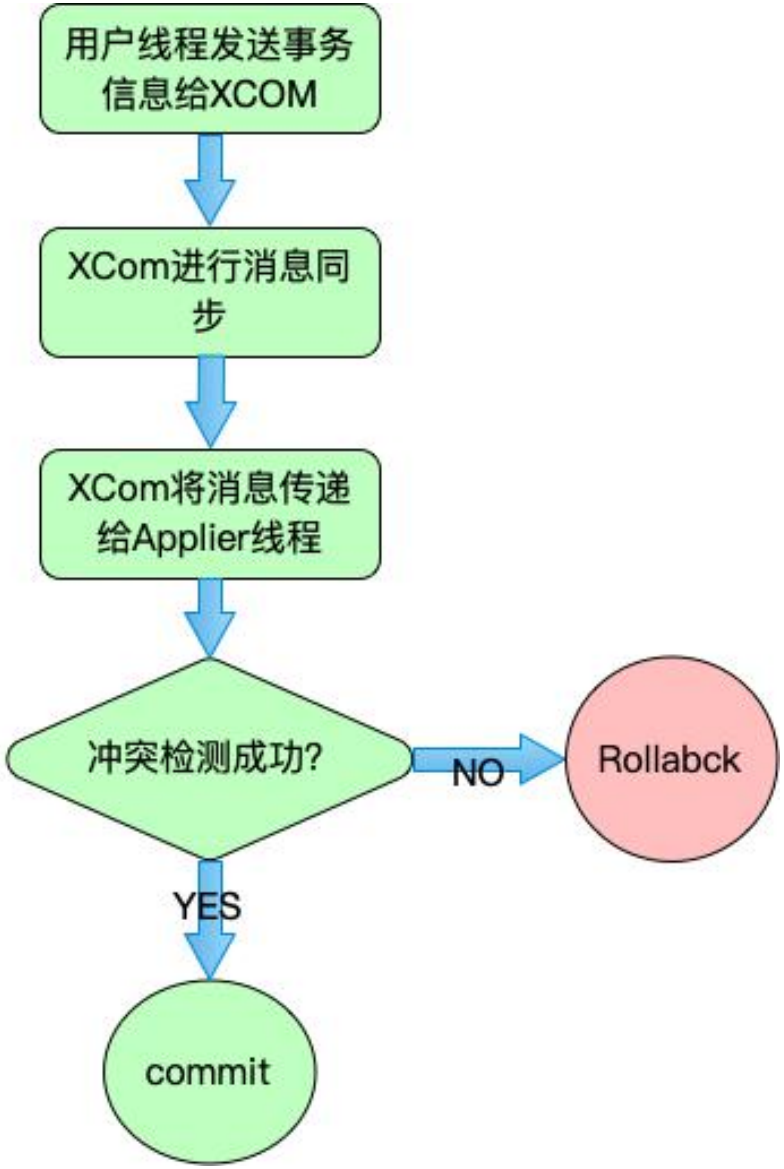


Towards a generic group communication service

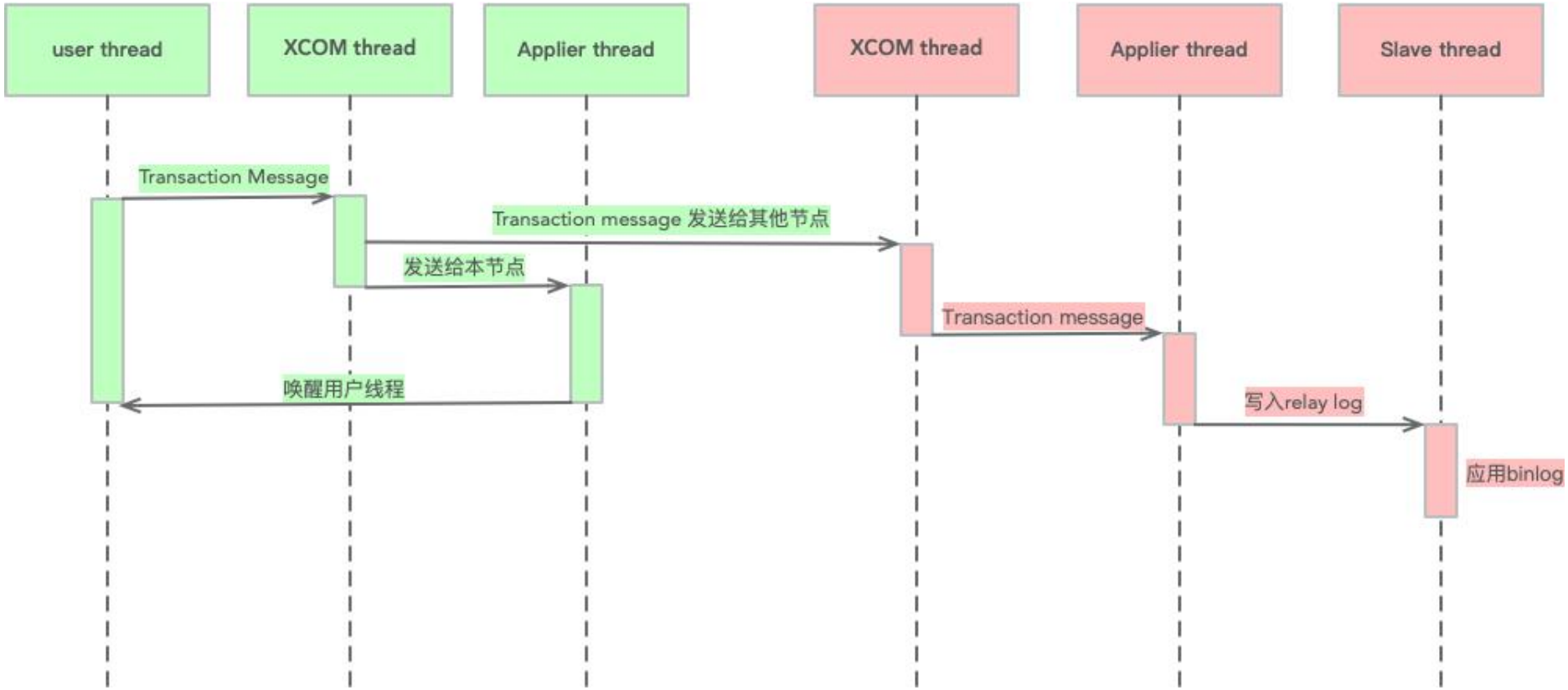
Life of a Transaction Commit in MGR



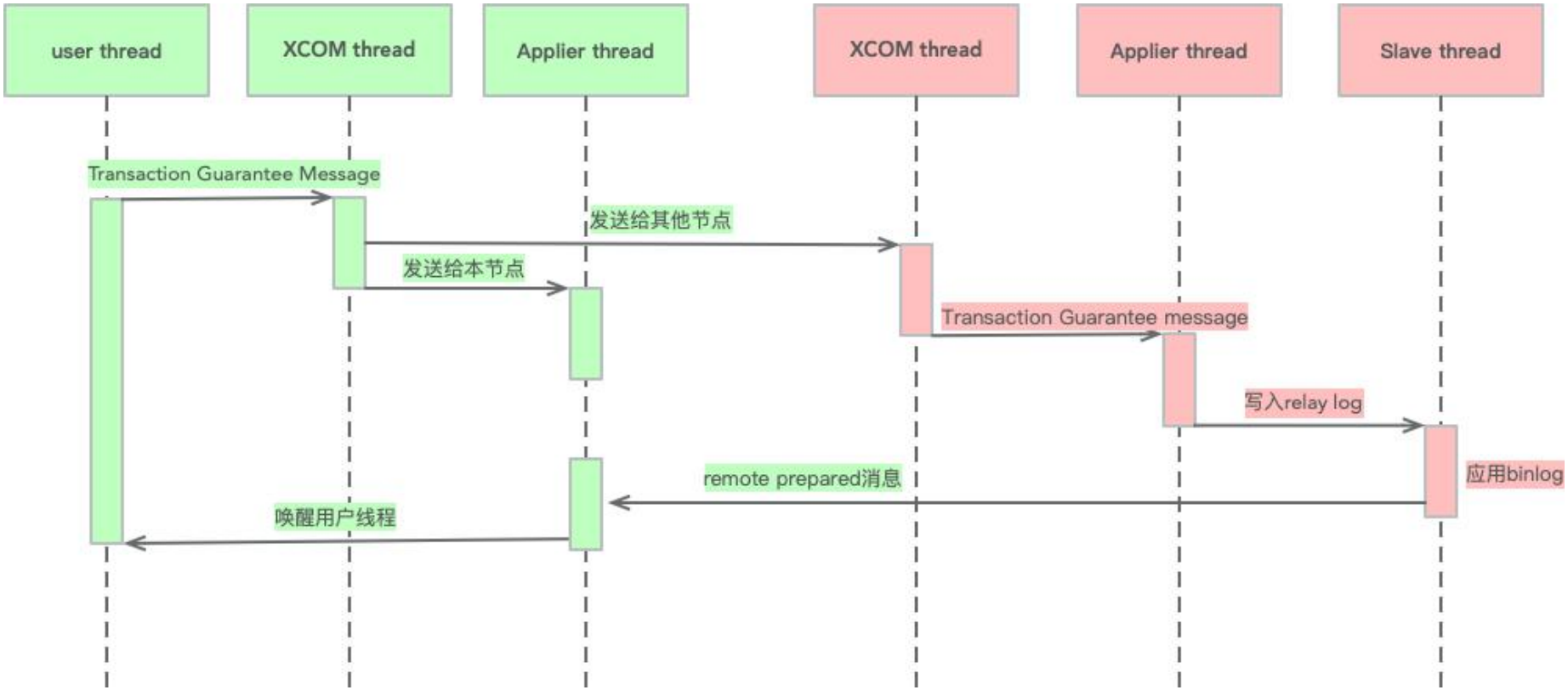
Life of a Transaction Commit in MGR



Life of a Transaction Commit in MGR



Life of Transaction Commit under AFTER Consistency Level



目录

1.server1, server2, server3组成的三节点MGR集群

2.server1, server2处于ONLINE状态，server3手动关闭

3.server1上在不间断并发执行事务

4.当重启server3实例，start group_replication后，server1报错异常退出。

server1执行请求的客户端报错信息：

ERROR 3798 (HY000) at line 1: Error while waiting for transactions with group_replication_consistency= 'AFTER' to commit.

server1节点的error log日志:

2020-09-29T06:40:09.508840Z 17 [ERROR] [MY-013309] [Repl] Plugin group_replication reported: 'Transaction '1:247' does not exist on Group Replication consistency manager while receiving remote transaction prepare.'

2020-09-29T06:40:09.508882Z 17 [ERROR] [MY-011452] [Repl] Plugin group_replication reported: 'Fatal error during execution on the Applier process of Group Replication. The server will now leave the group.'

通过比较server1和server2的binlog，发现server2比server1多了246和247两条日志。

245是drop table t3

246是view change日志。

247是server1错误日志中的报错事务, drop table t1.

```
SET @@SESSION.GTID_NEXT= 'aaa8c463-39cc-11eb-8dab-e454e8995a0e:245'

use test; DROP TABLE IF EXISTS t3 /* generated by server / / xid=293 */

SET @@SESSION.GTID_NEXT= 'aaa8c463-39cc-11eb-8dab-e454e8995a0e:246'

BEGIN

view_id=16098443053852267:5
COMMIT

SET @@SESSION.GTID_NEXT= 'aaa8c463-39cc-11eb-8dab-e454e8995a0e:247'

use test; DROP TABLE IF EXISTS t1 /* generated by server / / xid=296 */
```

SERVER 2比SERVER1多的日志

通过server1的报错信息，定位到具体的报错函数:
handler_remote_prepare

具体的报错逻辑:

1. 收到remote_prepare消息
2. 根据remote_prepare的gtid
   1. 查找本地处于MGR提交状态的事务列表，是否有对应的gtid的事务
   2. 或者已经提交的gtid
3. 如果都找不到，则报错

为什么server2上产生的247号事务的remote prepare，在server1上没有对应的事务呢？

```
int Transaction_consistency_manager::handle_remote_prepare(
    const rpl_sid *sid, rpl_gno gno,
    const Gcs_member_identifier &gcs_member_id) {
  DBUG_TRACE;
  rpl_sidno sidno = 0;

  Transaction_consistency_manager_key key(sidno, gno);

  m_map_lock->rdlock();
  typename Transaction_consistency_manager_map::iterator it = m_map.find(key);
  if (it == m_map.end()) {
    /*
      If this member is or just was in RECOVERING state, it may have applied
      consistent transactions through recovery channel, so before throw a
      error on a unknown prepare acknowledge message, first we check if the
      transaction is already committed on this member.
      This happens because the consistent transaction was executed while this
      member was in RECOVERING state, so the transaction was not being tracked.
    */
    Gtid gtid = {sidno, gno};
    if (is_gtid_committed(gtid)) {
      m_map_lock->unlock();
      return 0;
    }

    /* purecov: begin inspected */
    LogPluginErr(ERROR_LEVEL,
                 ER_GRP_RPL_TRX_DOES_NOT_EXIST_ON_TCM_ON_HANDLE_REMOTE_PREPARE,
                 sidno, gno);
    m_map_lock->unlock();
    return 1;
    /* purecov: end */
  }
```

Bug 分析

Great DB 万里数据库

由于本地有prepared事务245，故延迟应用view change

```
T@10: 18:59:48.659912 info: View change GTID information: output_set: aaa8c463-39cc-11eb-8dab-e454e8995a0e:1-244
T@10: 18:59:48.660157 info: Delaying the log of the view '16098443053852267:5' to after local prepared transactions

T@10: 18:59:48.661603 | | | info: Group replication Certifier: certification result: 246
T@10: 18:59:48.661672 | | | info: thread_id: 266; local_transaction: 1; gtid: 2:246; sid_specified: 0; consistency_level: 4; transaction_prepared_locally: 1; transaction_pr
: 0
T@10: 18:59:48.661692 | | | info: insert gtid to m_map: 2:246; consistency_level: 4;

T@10: 18:59:48.661778 | <Transaction_consistency_info::handle_remote_prepare
T@10: 18:59:48.661791 | info: remove gtid from map: 2:245; consistency_level: 4;
T@265: 18:59:48.678833 | | | | | | | | | | | | | >Transaction_consistency_manager::remove_prepared_transaction
T@265: 18:59:48.678862 | | | | | | | | | | | | | >int Certification_handler::handle_event
T@265: 18:59:48.678873 | | | | | | | | | | | | | | >Certification_handler::log_view_change_event_in_order
T@269: 18:59:48.681160 | | | | | | | | | | | | | | | | | | | | | | query: SELECT WAIT_FOR_EXECUTED_GTID_SET('aaa8c463-39cc-11eb-8dab-e454e8995a0e:246', 10)

T@10: 18:59:48.752255 >Transaction_consistency_manager::handle_remote_prepare
T@10: 18:59:48.752361 | | | enter: buffer: 2021-01-05T10:59:48.752349Z 10 [ERROR] [MY-013309] [Repl] Plugin group_replication reported: 'Transaction '2:247' does not exit
```

drop table t1抢占了246号gtid

接收到了245的remote prepare消息，
激活了view change的执行

接收到了247的remote prepare消息，异常退出

1. drop table t3处于local prepared状态，事务号245
2. view change到来，发现有本地为提交的事务，故delay
3. drop table t1应用，分配了246事务号
4. 245 remote prepare消息到来，激活245号事务提交
5. 247 remote prepare消息到来，找不到对应的gtid，异常退出

主要原因是local prepared状态的事务导致了view change delay，view change delay后续的事务占用本该属于view change的gtid，导致了gtid在节点间不一致的情况。

修复方案

根本原因是因为delay view change后续的事务占用了view change的GTID。

故我们需要保证delay view change之后的事务需要等待view change执行完成之后，才能真正应用。

完善测例，保证问题可回归。

MGR目前版本里，关于并发时序问题的Bug不仅这一个。

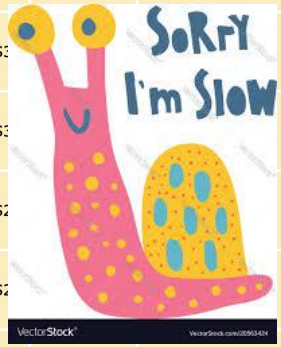
```
int Applier_module::applier_thread_handle() {
  DBUG_TRACE;

@@ -478,8 +566,15 @@ int Applier_module::applier_thread_handle() {
  while (!applier_error && !packet_application_error && !loop_termination) {
    if (is_applier_thread_aborted()) break;

+   /* Delayed packets are activated by later packets */
    this->incoming->front(&packet);  // blocking

+   if (has_delayed_view_change_event) {          延迟后续的事务应用
+     if (check_and_delay_packet_after_delayed_view_change(packet)) {
+       continue;
+     }
+   }
+
    switch (packet->get_packet_type()) {
      case ACTION_PACKET_TYPE:
        this->incoming->pop();
@@ -517,6 +612,13 @@ int Applier_module::applier_thread_handle() {
            static_cast<Leaving_members_action_packet *>(packet));
        this->incoming->pop();
        break;
+     case SYNC_PREPARED_COMPLETE_TYPE:          开启后续事务应用
+       has_delayed_view_change_event = false;
+       this->incoming->pop();
+       if (delayed_packets_queue->size() > 0) {
+         add_delayed_packets();
+       }
+       break;
      default:
        DBUG_ASSERT(0); /* purecov: inspected */
```

目录

All Verified Bugs

Showing 1-10 of 40 (Edit, Save, CSV, Feed) Show Next 10 Entries »

ID#	Date	Updated	Type	Status	Sev	Version	OS	CPU	Summary
102556	2021-02-10 15:48	2021-03-04 13:25	MySQL Server: Group Replication	Verified (12 days)	S2	8.0.21	CentOS (Release: 7.7.1908)	x86 (Intel(R) Xeon(R) CPU E5-2620 v4 @ 2.10GHz)	Apparent deadlock involving group replication applier threads
102515	2021-02-07 11:03	2021-03-09 7:36	MySQL Server: Group Replication	Verified (34 days)	S2	8.0.23	Any	Any	Group replication member remains ONLINE even when it gets (far) behind
102433	2021-02-01 8:49	2021-02-01 14:04	MySQL Server: Group Replication	Verified (42 days)	S4	8.0	Any	Any	Lightweigth group replication consensus member without the full database
102249	2021-01-14 15:56	2021-01-27 17:40	MySQL Server: Group Replication	Verified (56 days)	S3	8.0.22	Any	Any	Slow query log filling up with performance_schema queries
101635	2020-11-17 6:29	2020-11-24 9:29	MySQL Server: Group Replication	Verified (111 days)	S3			Any	group_replication_local_address port overflow
101237	2020-10-20 8:38	2020-10-27 7:37	MySQL Server: Group Replication	Verified (139 days)	S3			Any	stop group_replicaiton may block long time when restart server
100299	2020-07-23 2:57	2020-07-23 10:58	MySQL Server: Group Replication	Verified (235 days)	S2			Any	secondly role cannot join to group_replication after fail-over
100163	2020-07-09 0:56	2020-07-09 12:21	MySQL Server: Group Replication	Verified (249 days)	S2			Any	xa commit failed when stop group_replication will lead node error
99735	2020-05-29 1:46	2020-06-30 12:07	MySQL Server: Group Replication	Verified (258 days)	S4	8.0.20	Any	Any	auto rejoin group_replication when server restart
99689	2020-05-26 4:02	2020-06-09 19:31	MySQL Server: Group Replication	Verified (285 days)	S3	8.0.18	Any	Any	member cannot add to group_replication cluster after failover



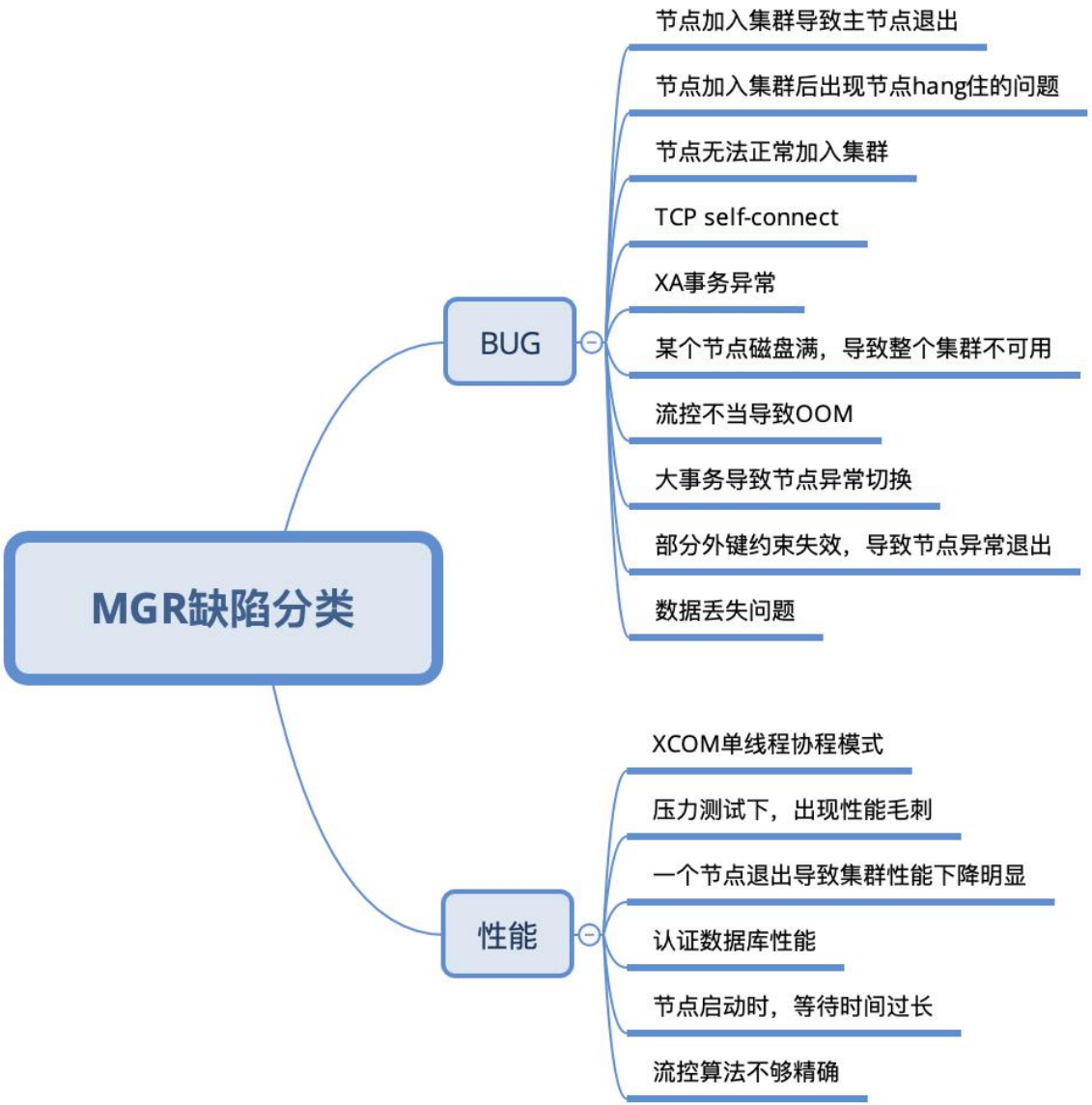
重现MGR Bug比一般Bug更复杂，导致用户无法准确描述和复现Bug。

Need Feedback, Can't repeat，一些Bug石沉大海。

Bugs reported by 万里数据库

100906	2020-09-22 8:38	2020-10-30 11:12	MySQL Server: Group Replication	Not a Bug (136 days)	S2	8.0.18, 8.0.21	Any	Any	cannot execute read-only transaction when group_replication in ERROR state
100299	2020-07-23 2:57	2020-07-23 10:58	MySQL Server: Group Replication	Verified (235 days)	S2	8.0.18, 8.0.21	Any	Any	secondly role cannot join to group_replication after fail-over
100163	2020-07-09 0:56	2020-07-09 12:21	MySQL Server: Group Replication	Verified (249 days)	S2	8.0.18	Any	Any	xa commit failed when stop group_replication will lead node error
100052	2020-07-01 0:45	2020-07-20 8:59	MySQL Server: Group Replication	Not a Bug (238 days)	S3	8.0.18, 8.0.20	Any	Any	install group_replication will write error log
99735	2020-05-29 1:46	2020-06-30 12:07	MySQL Server: Group Replication	Verified (258 days)	S4	8.0.20	Any	Any	auto rejoin group_replication when server restart
99689	2020-05-26 4:02	2020-06-09 19:31	MySQL Server: Group Replication	Verified (285 days)	S3	8.0.18	Any	Any	member cannot add to group_replication cluster after failover
101237	2020-10-20 8:38	2020-10-27 7:37	MySQL Server: Group Replication	Verified (139 days)	S3	8.0.21	Any	Any	stop group_replicaiton may block long time when restart server
98643	2020-02-18 8:11	2020-05-12 15:07	MySQL Server: Group Replication	Closed (307 days)	S3	8.0.18	Any	Any	group replication will be block primary node shutdown
98473	2020-02-04 4:18	2020-03-02 10:18	MySQL Server: Group Replication	Not a Bug (378 days)	S3	8.0.18	Any	Any	group replication will be block after lock table
98151	2020-01-08 1:03	2020-08-20 6:35	MySQL Server: Group Replication	Verified (363 days)	S3	8.0.18	Any	Any	group replication with wrong member_state after server shutdown
101635	2020-11-17 6:29	2020-11-24 9:29	MySQL Server: Group Replication	Verified (111 days)	S3	8.0.21, 8.0.22	Any	Any	group_replication_local_address port overflow

MGR缺陷分类


MGR缺陷分类
BUG
节点加入集群导致主节点退出
节点加入集群后出现节点hang住的问题
节点无法正常加入集群
TCP self-connect
XA事务异常
某个节点磁盘满，导致整个集群不可用
流控不当导致OOM
大事务导致节点异常切换
部分外键约束失效，导致节点异常退出
数据丢失问题
性能
XCOM单线程协程模式
压力测试下，出现性能毛刺
一个节点退出导致集群性能下降明显
认证数据库性能
节点启动时，等待时间过长
流控算法不够精确

Great DB 万里数据库

4月1日

发布二进制包

扫码入群，领取软件试用版&演讲ppt

互动答疑环节

不忘DB初心，牢记万里使命

谢谢！

万里数据库官微