QINGCLOUD

MySQL数据复制技术演进之路

QINGCLOUD

卢文双

青云QingCloud RDS-MySQL研发负责人

2017年加入青云，主导QingCloud MySQL Plus的研发、部署及运维，专注于数据库高可用架构设计和开发。曾任职于人大金仓，参与分布式数据库 KingbaseDBCloud、KingbaseAnalyticsDB 的内核研发。

大纲

- 复制技术发展史
- 复制模型
- 复制方式
- 并行复制
- QingCloud MySQL Plus 实践

QINGCLOUD

01

复制技术发展史

QINGCLOUD

复制技术发展史

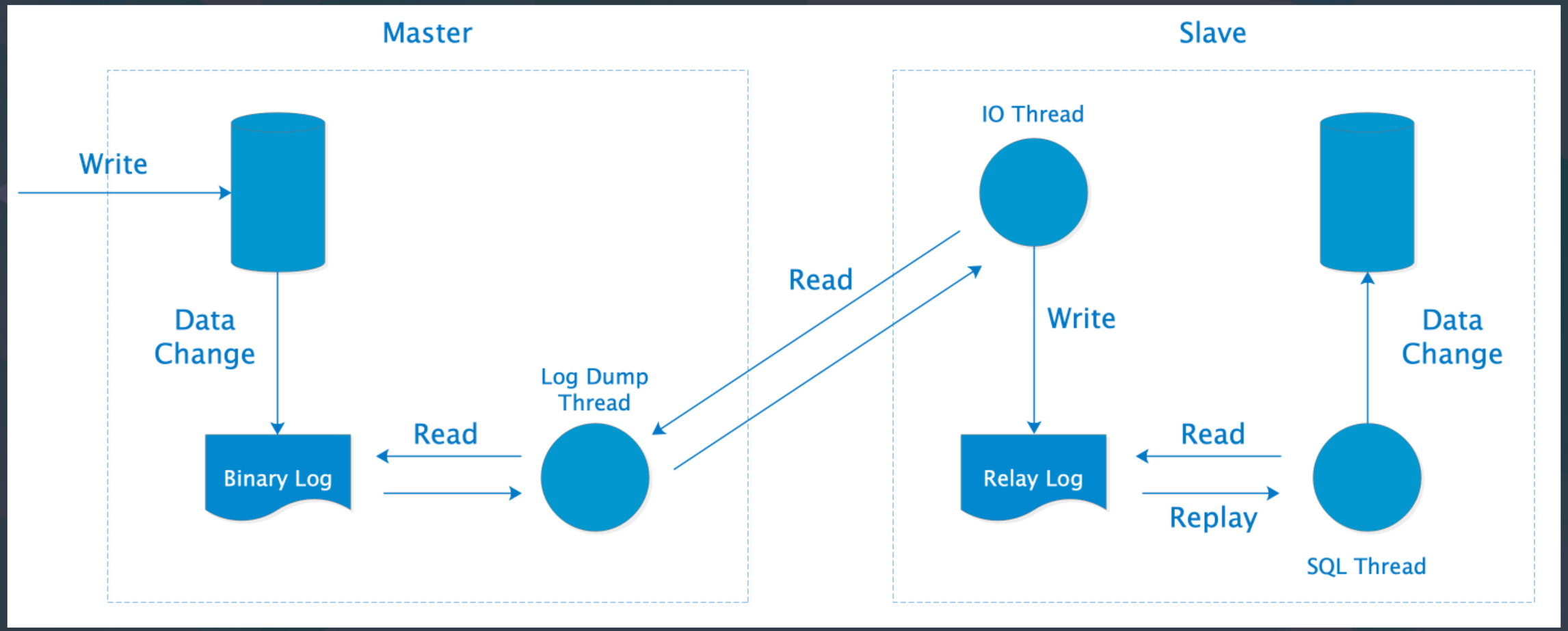
注意：此处为引入该特性的版本，而非GA版本。

02

复制模型

QINGCLOUD

复制模型



binlog-format

binlog-format：事务记录到binlog中的格式。

- STATEMENT（SBR）：语句模式。记录执行的原始SQL语句。
- ROW（RBR）：行模式，从MySQL 5.1开始支持。（强烈推荐）
- MIXED：混合模式。从MySQL 5.1开始出现的一种过渡格式，默认基于语句复制，在某些情况下，依据执行的语句与存储引擎自动切换为 ROW 模式。

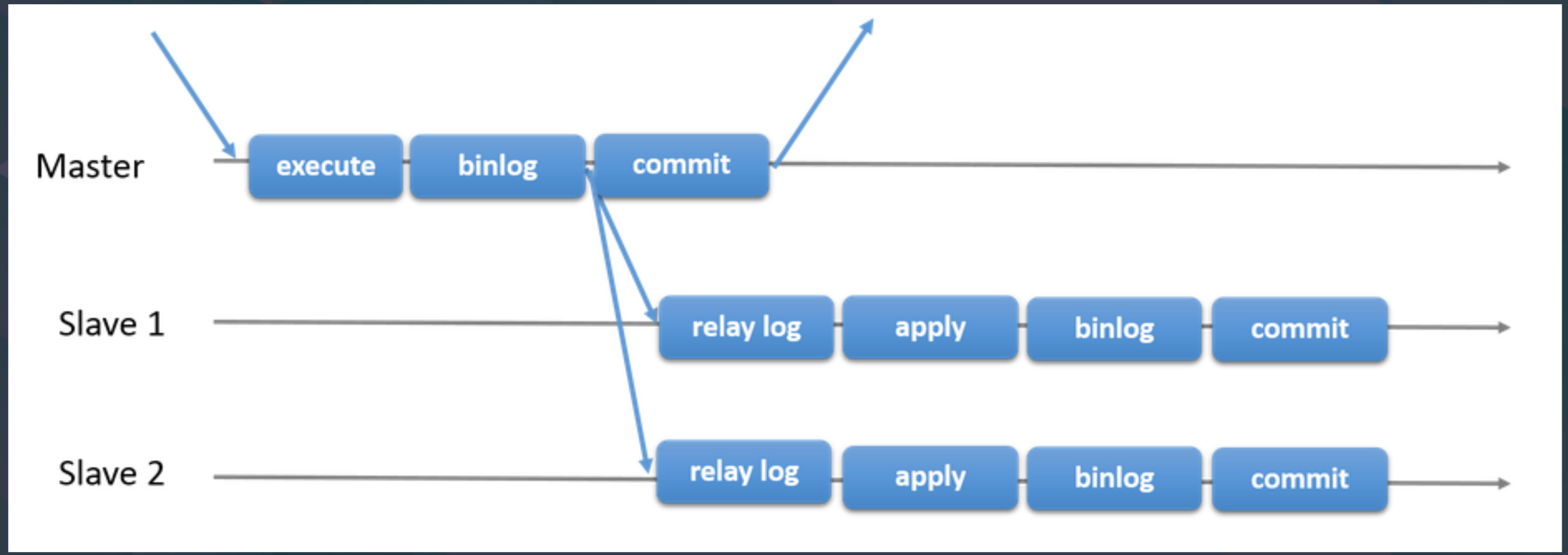
MySQL 5.7.6 及之前的版本，默认值为 STATEMENT， MySQL 5.7.7 及之后的版本，默认值为 ROW 。

03

复制方式

QINGCLOUD

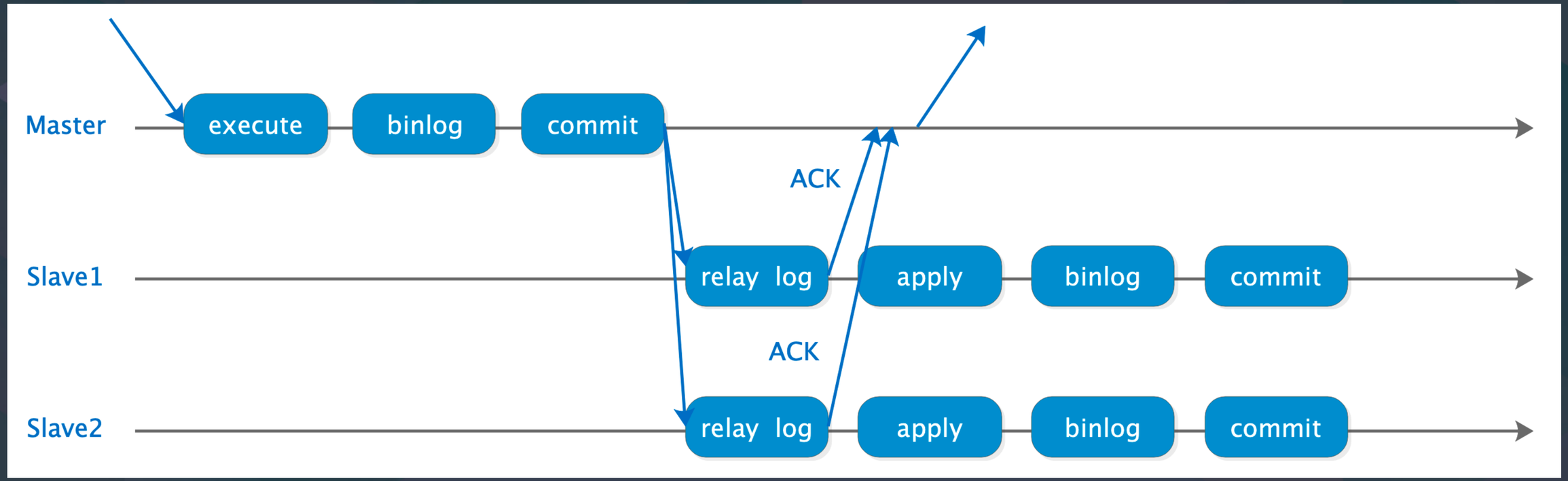
异步复制



QINGCLOUD

传统半同步

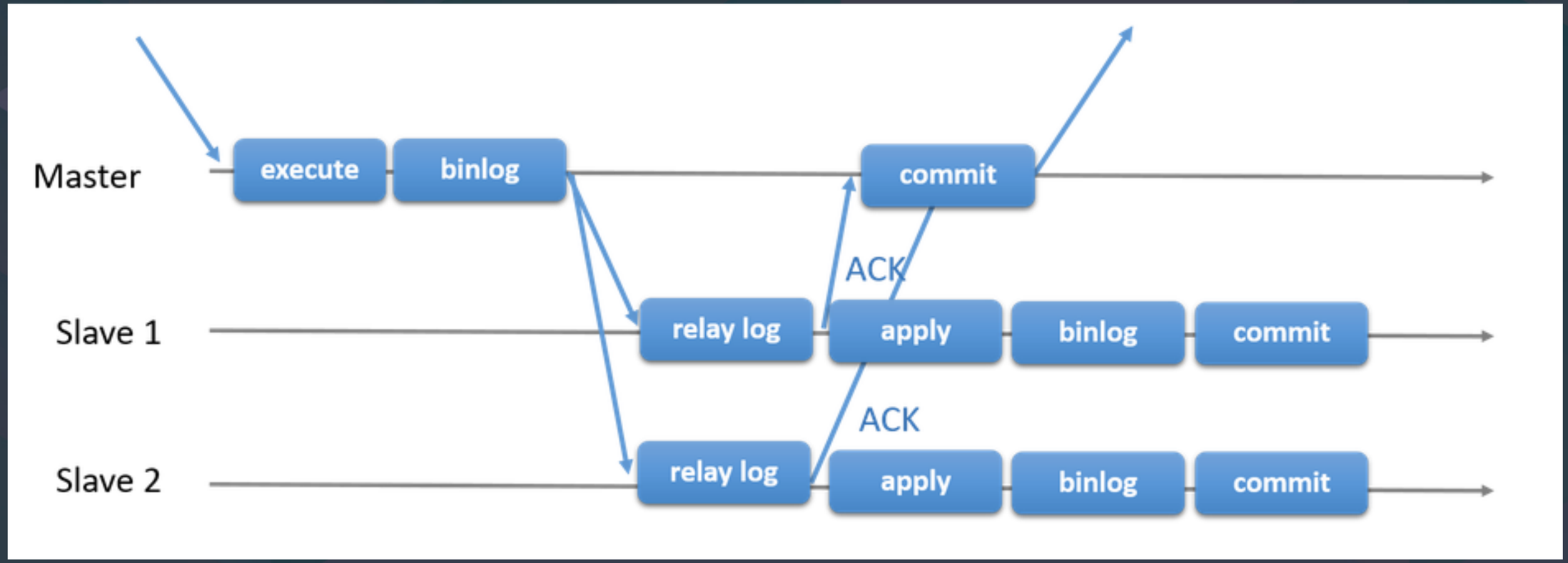
- rpl_semi_sync_master_wait_point = AFTER_COMMIT



QINGCLOUD

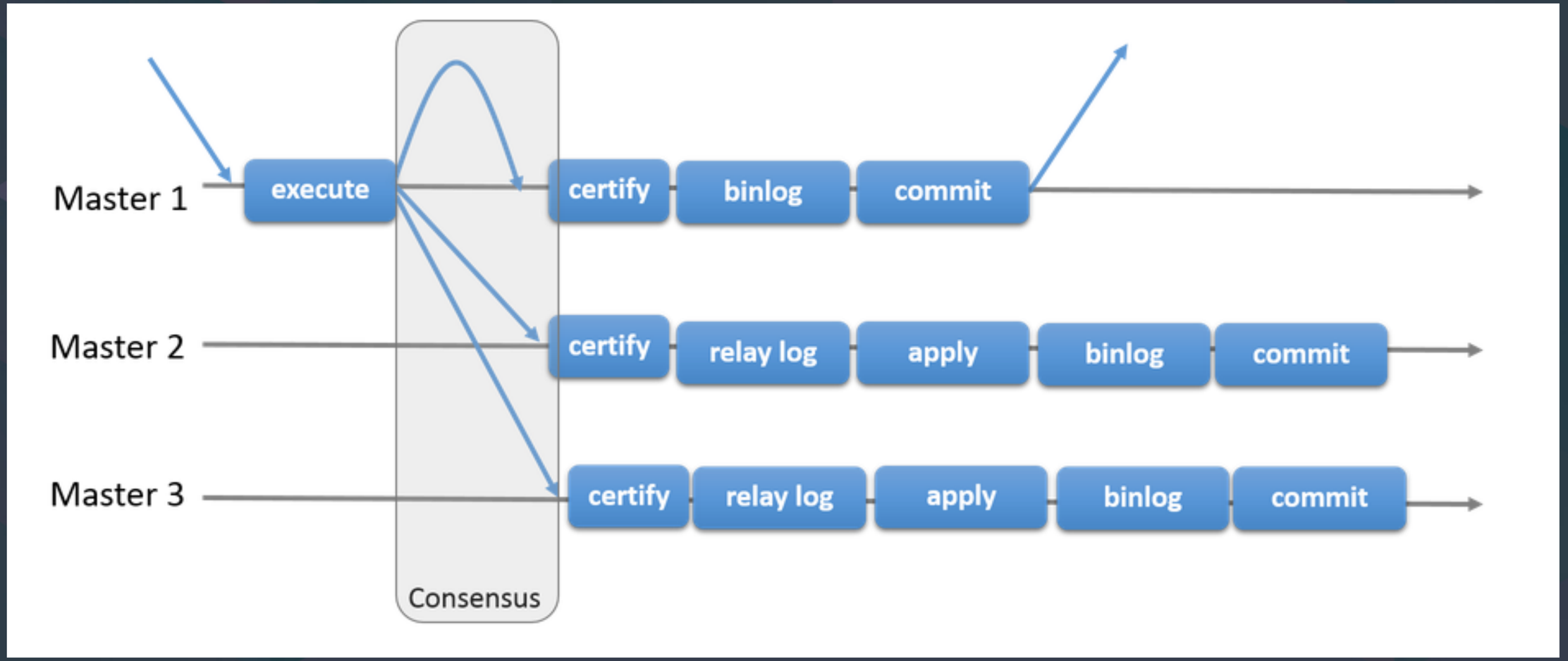
增强半同步复制（lossless replication）

rpl_semi_sync_master_wait_point = AFTER_SYNC



QINGCLOUD

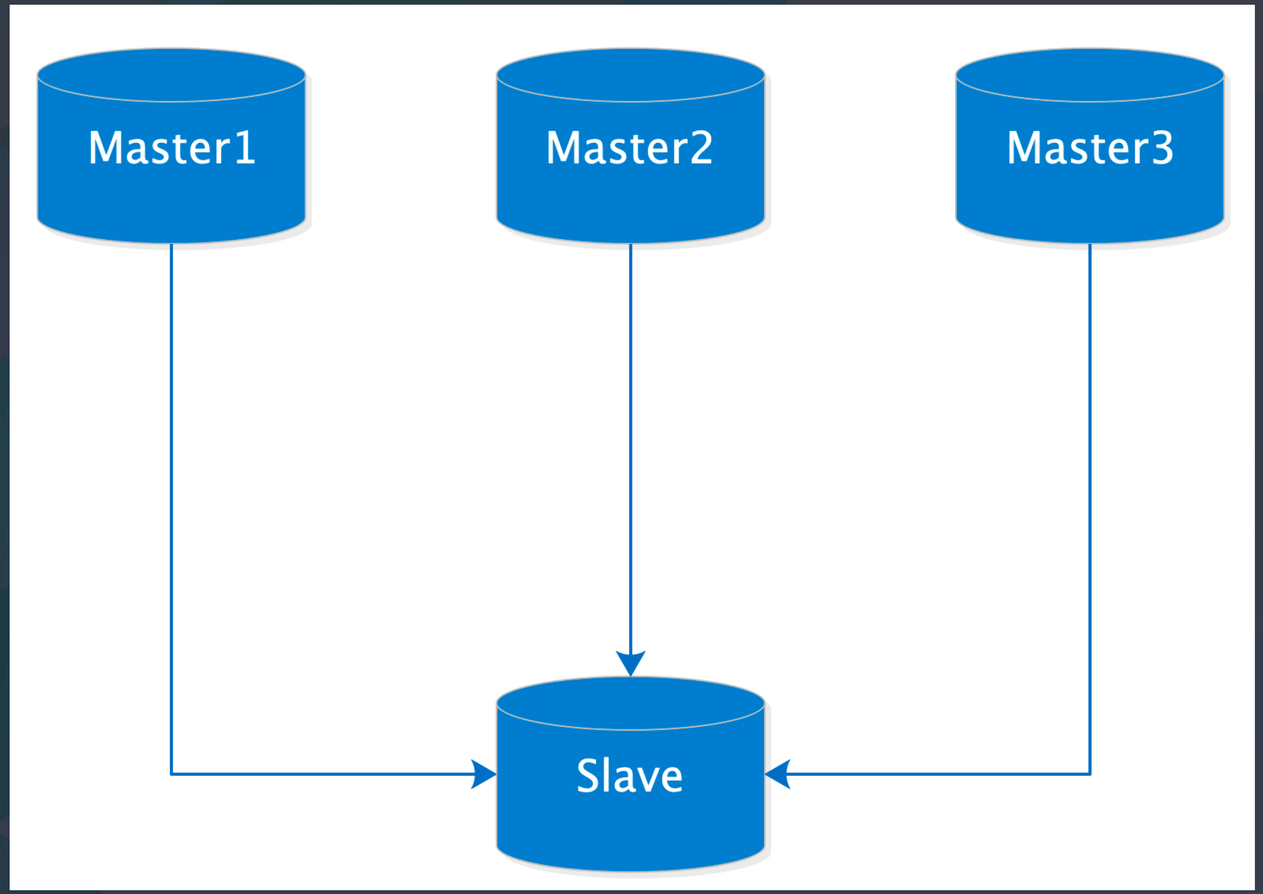
组复制（MGR）



QINGCLOUD

多源复制

将多个主库的数据汇总到一台从库，一般用于分析、灾备。



QINGCLOUD

延迟复制

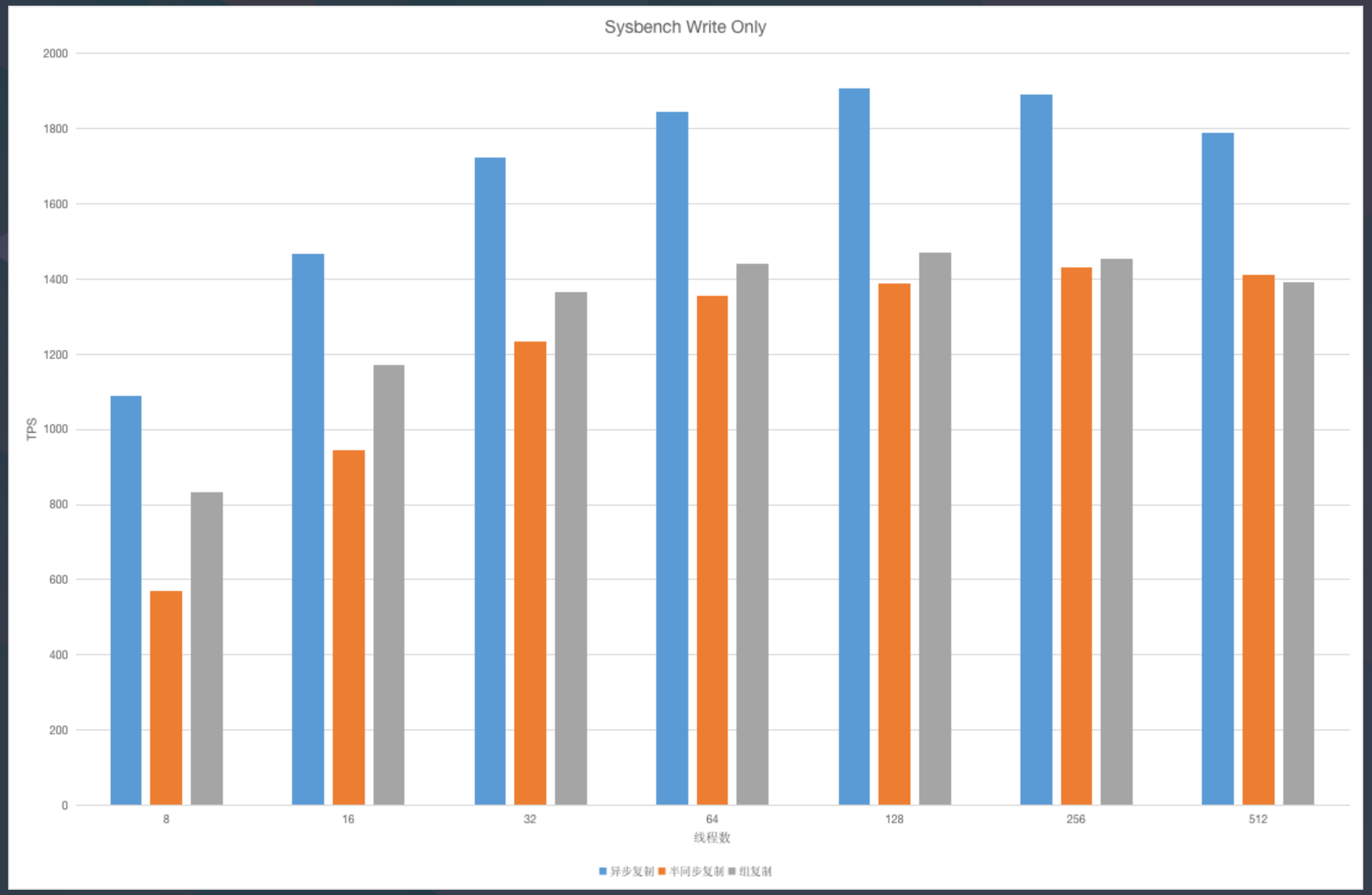
令从库延迟一段时间后再重放事务。

开启方式（延迟3600秒）：
change master to master_delay=3600 ……

适用场景：
对主库误操作的补救措施，由于从库只接收、未重放，因此可基于时间点恢复。

QINGCLOUD

性能比较



工具：Sysbench

场景：Write Only

MySQL版本：5.7.25

蓝色：异步复制

橙色：半同步复制

灰色：组复制

QINGCLOUD

04

并行复制

QINGCLOUD

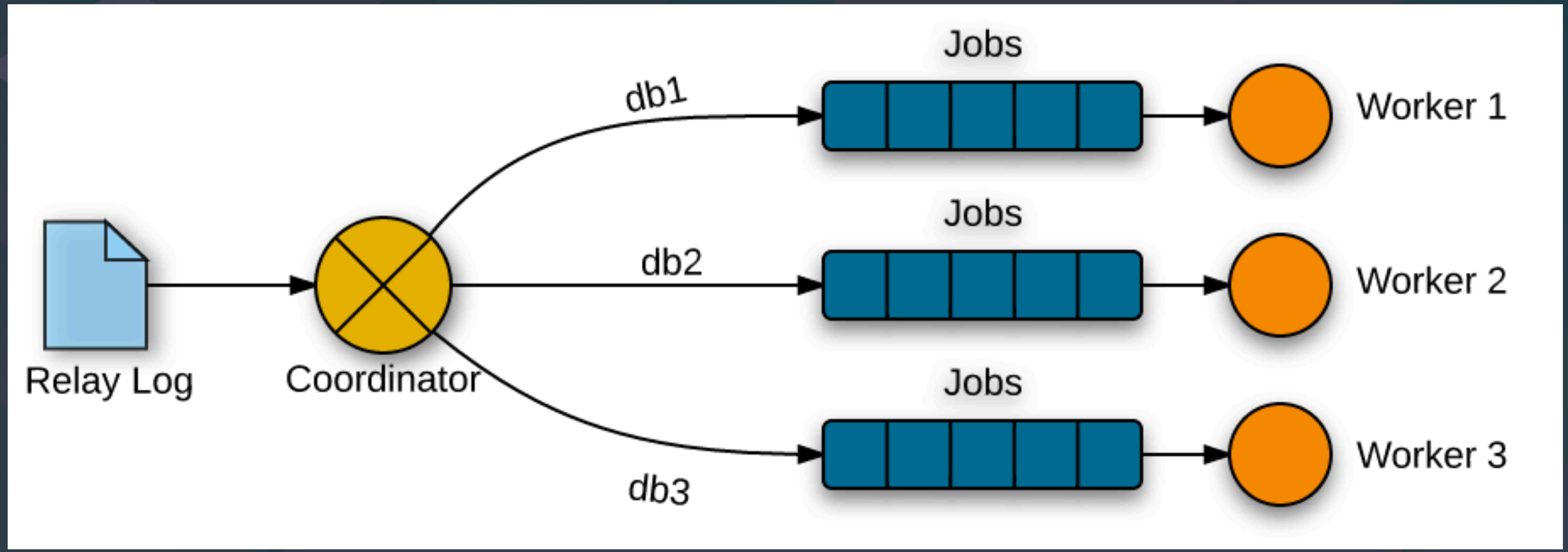
并行复制发展历程

由于事务在主库可以并发执行，而早期版本只能在从库串行重放，导致主从延迟过大，为降低主从延迟，才有了并行复制。

- MySQL 5.6 开始支持基于库级别的并行复制
- MySQL 5.7 开始支持基于组提交的并行复制
  - Commit-Parent-Based
  - Lock-Based
- MySQL 5.7.22、8.0.1 开始支持基于WriteSet的并行复制

基于库级别的并行复制

原理：在slave节点为每一个库分配了一个worker线程。参数 slave_parallel_workers （5.6引入）指定worker线程数量。



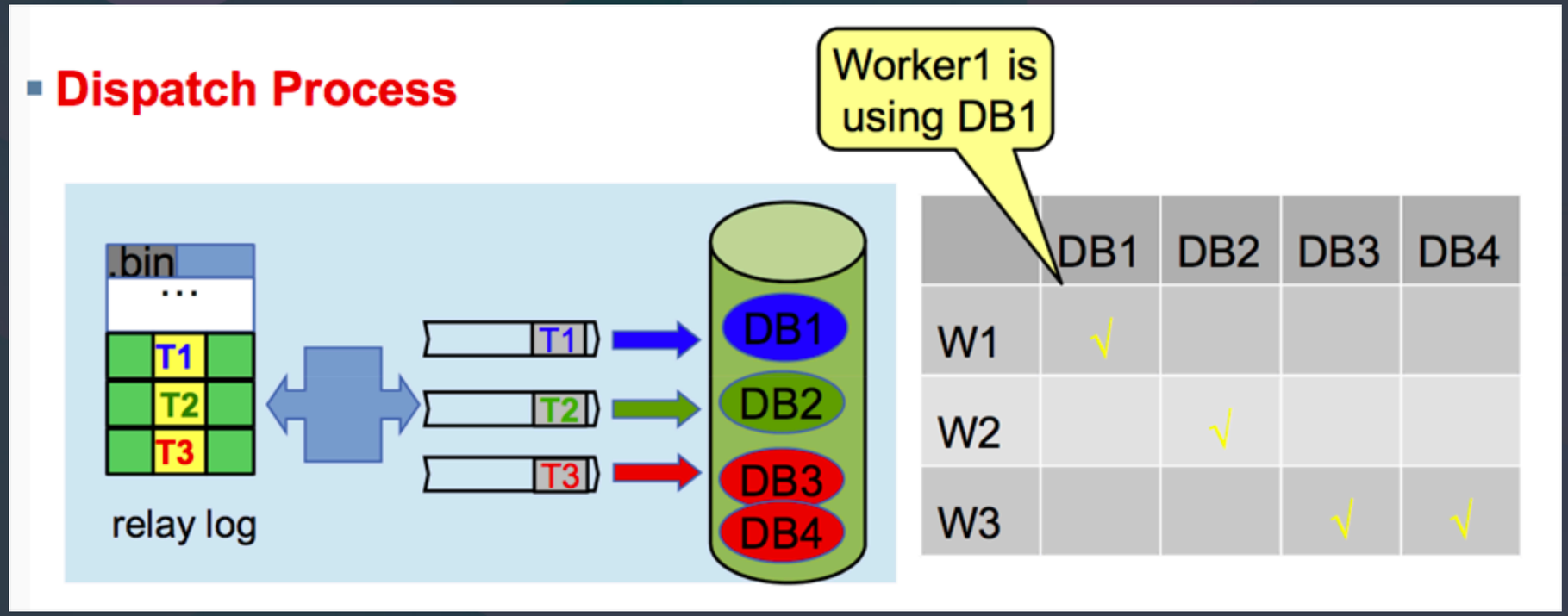
QINGCLOUD

基于库级别的并行复制

存在问题：

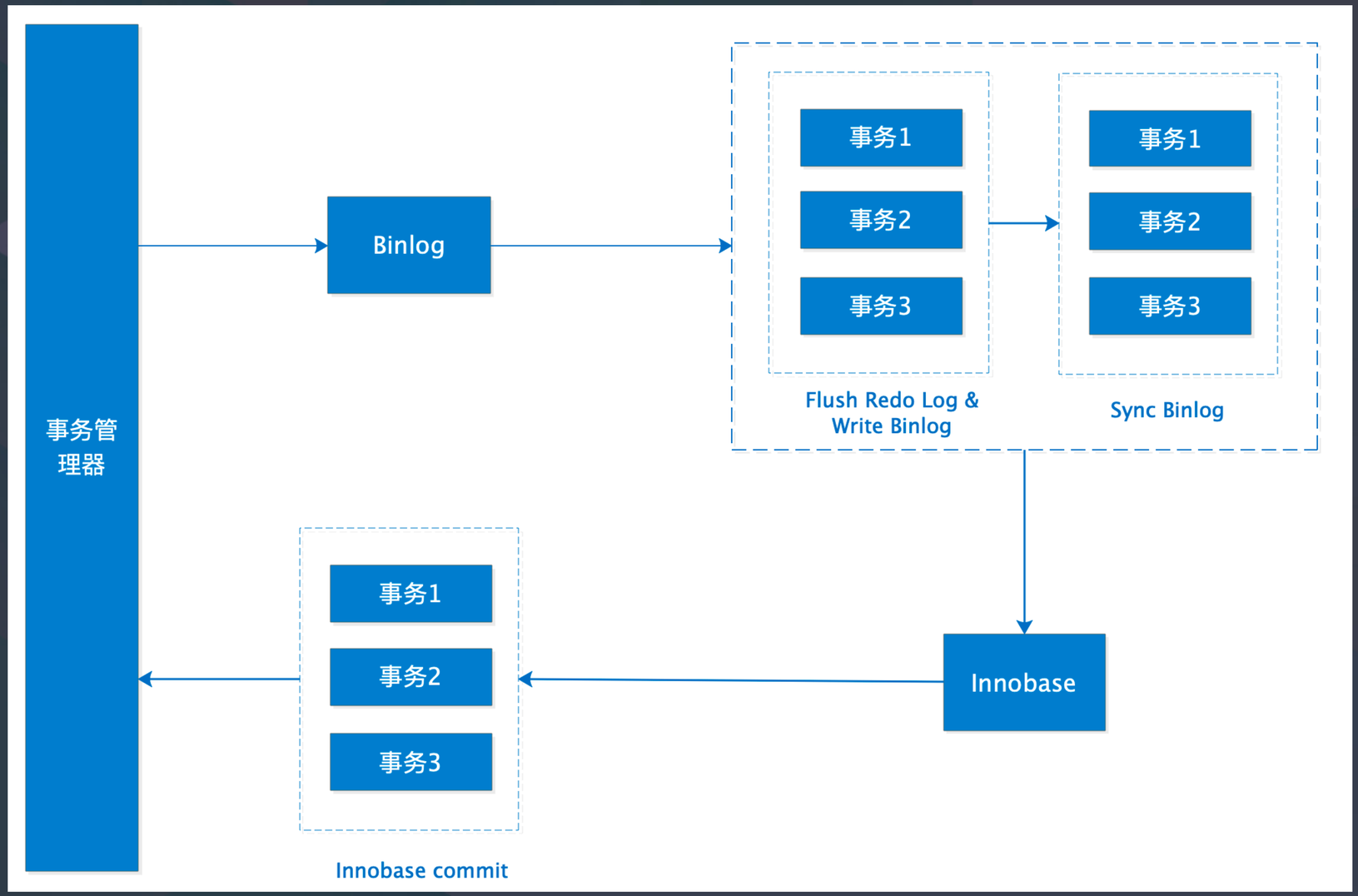
1）当出现跨库的情况时，会变为串行；

2）生产环境中一般是单库多表，不适用。



QINGCLOUD

基于组提交的并行复制



QINGCLOUD

基于组提交的并行复制

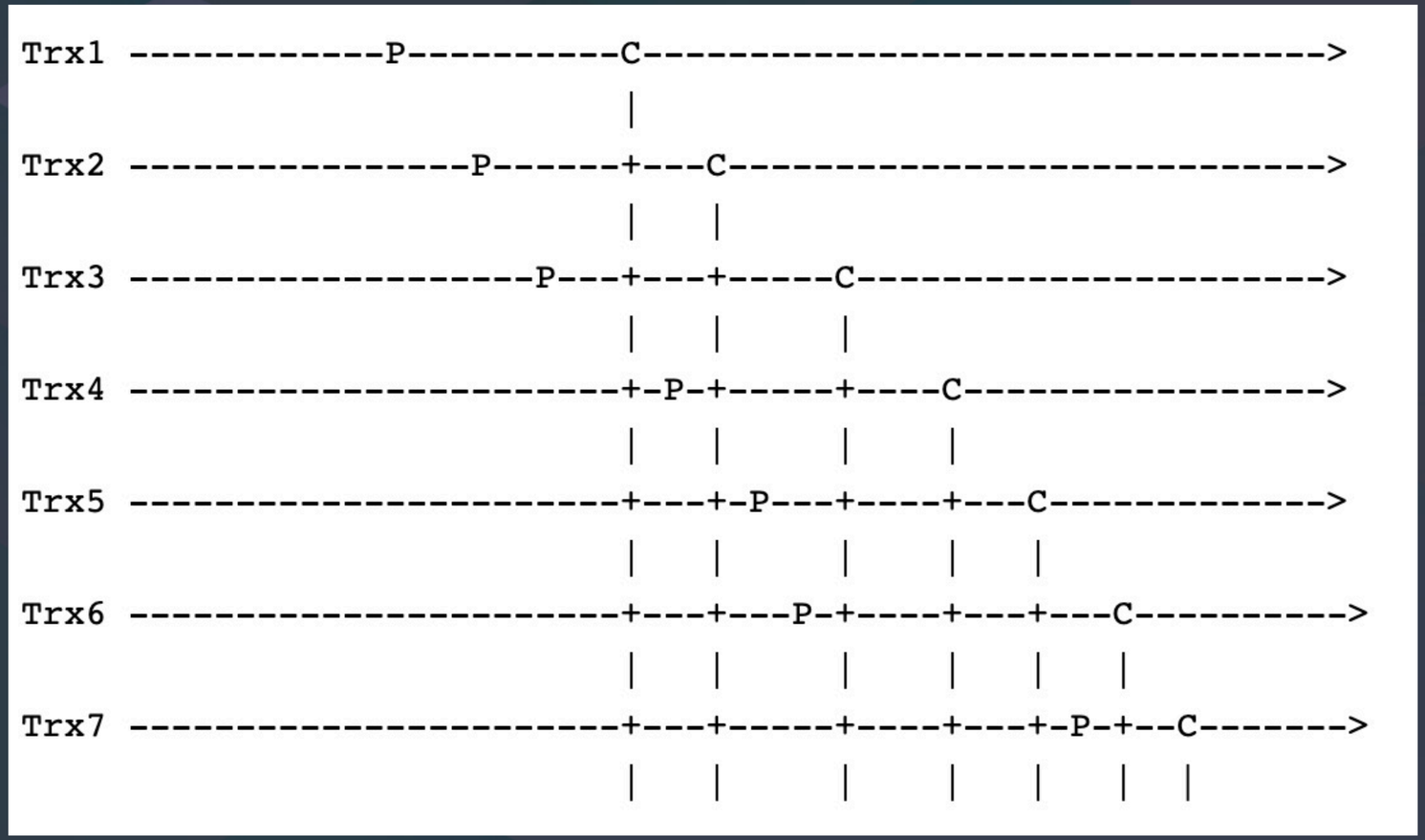
MySQL 5.7引入了新的变量slave-parallel-type，其可以配置的值有：

- DATABASE：默认值，基于库的并行复制方式（兼容5.6）。
- LOGICAL_CLOCK：基于组提交的并行复制方式。

QINGCLOUD

基于组提交的并行复制

Commit-Parent-Based方式的原理：如果两个事务能在master节点同时prepare成功，说明他们之间不存在冲突，那么这两个事务可以在slave节点并行重放。



说明：
横轴为时间线，
P为prepare时间点，
C为commit时间点。

可见，Trx1, Trx2和Trx3可以并行重放，Trx5和Trx6可以并行重放。

缺陷：在该方式下，Trx4不能与Trx5、Trx6并行重放。为了解决这类问题，MySQL实现了Lock-Based的并行复制。

QINGCLOUD

基于组提交的并行复制

Lock-Based方式的原理：如果两个事务同时获得了其所需的所有锁，则表明这两个事务不冲突，可以同时重放。

- 可并行重放：

```
Trx1 -----L----------C------------>
Trx2 ----------L----------C------->
```

- 不能并行重放：

```
Trx1 -----L----C------------------>
Trx2 ---------------L----C------->
```

QINGCLOUD

基于组提交的并行复制

```
#191023 22:49:34 ... end_log_pos 6122799 CRC32 0xfc389b2f   GTID last_committed=24490 sequence_number=24491 ...
#191023 22:49:34 ... end_log_pos 6123049 CRC32 0x0e8219f9   GTID last_committed=24490 sequence_number=24492 ...
#191023 22:49:34 ... end_log_pos 6123299 CRC32 0x43473b15   GTID last_committed=24490 sequence_number=24493 ...
#191023 22:49:34 ... end_log_pos 6123549 CRC32 0xec6b0bcf   GTID last_committed=24490 sequence_number=24494 ...
#191023 22:49:34 ... end_log_pos 6123799 CRC32 0xa1ae2923   GTID last_committed=24490 sequence_number=24495 ...
#191023 22:49:34 ... end_log_pos 6124049 CRC32 0xd04112dc   GTID last_committed=24490 sequence_number=24496 ...
#191023 22:49:34 ... end_log_pos 6124299 CRC32 0xf078f172   GTID last_committed=24490 sequence_number=24497 ...
#191023 22:49:34 ... end_log_pos 6124549 CRC32 0x89a269f2   GTID last_committed=24497 sequence_number=24498 ...
#191023 22:49:34 ... end_log_pos 6124799 CRC32 0x8755b294   GTID last_committed=24498 sequence_number=24499 ...
#191023 22:49:34 ... end_log_pos 6125049 CRC32 0x92f090db   GTID last_committed=24498 sequence_number=24500 ...
#191023 22:49:34 ... end_log_pos 6125299 CRC32 0x0dc7a4fd   GTID last_committed=24498 sequence_number=24501 ...
#191023 22:49:34 ... end_log_pos 6125549 CRC32 0x097ad503   GTID last_committed=24498 sequence_number=24502 ...
#191023 22:49:34 ... end_log_pos 6125799 CRC32 0x15f29d5a   GTID last_committed=24498 sequence_number=24503 ...
#191023 22:49:34 ... end_log_pos 6126049 CRC32 0xddf7cd6f   GTID last_committed=24503 sequence_number=24504 ...
#191023 22:49:34 ... end_log_pos 6126299 CRC32 0x7a2fddb5   GTID last_committed=24504 sequence_number=24505 ...
#191023 22:49:34 ... end_log_pos 6126549 CRC32 0xbd06fc98   GTID last_committed=24505 sequence_number=24506 ...
```

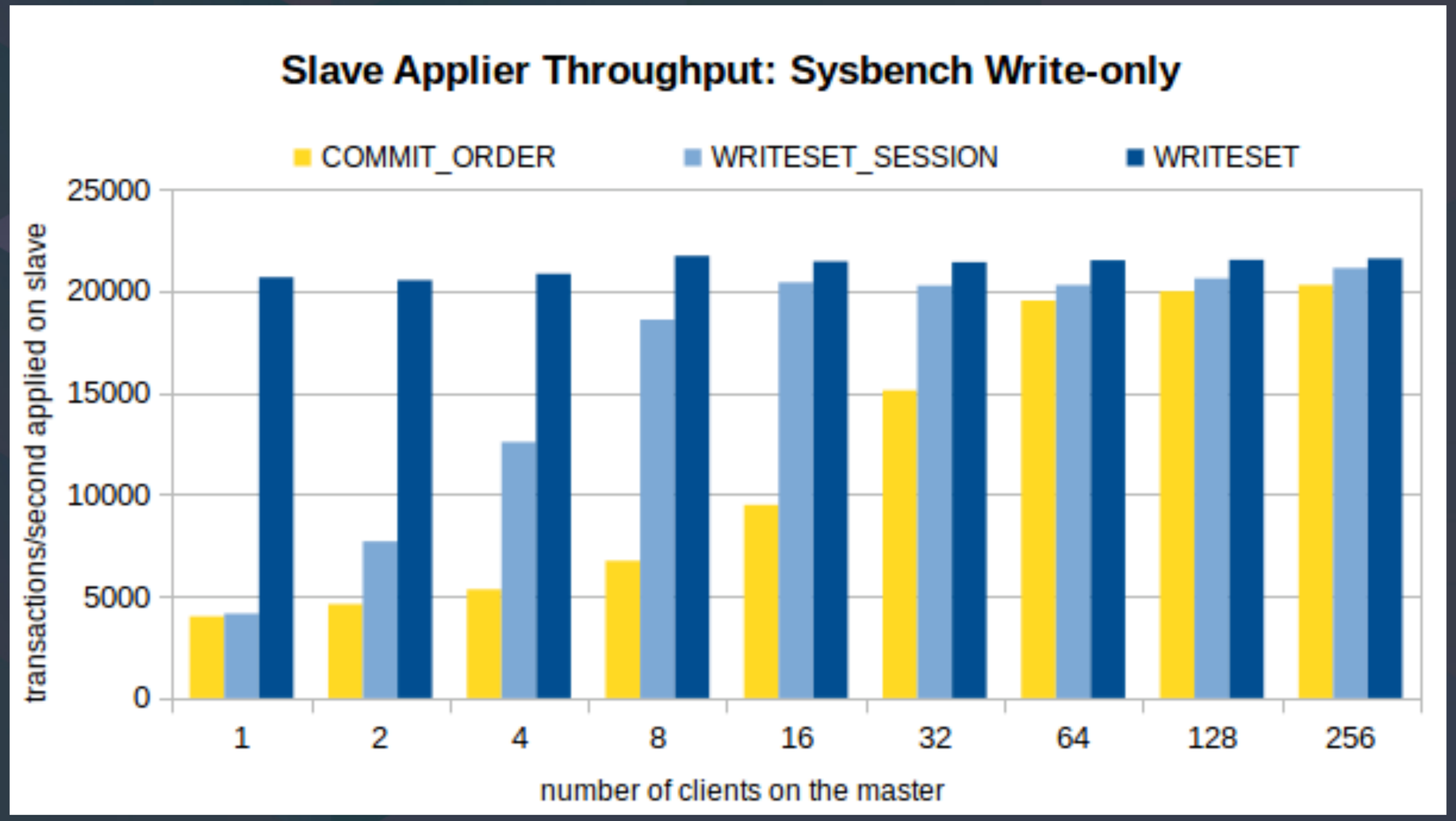
QINGCLOUD

基于WriteSet的并行复制

MySQL 会对这个提交的事务中的一行记录做一个 HASH 值，也就是 writeset，并将这些 writeset 存入一张 HASH 表。其他事务提交时会检查这张 HASH 表中是否有相同的记录，如果有，表明有其他事务修改同一行记录，不能并行；反之，可以并行复制。也就是说，并行的粒度从事务级细化为记录级。需要在主库添加如下参数：

- transaction_write_set_extraction=XXHASH64　# 指定生成HASH值的算法，可选值：OFF（默认）、MURMUR32、XXHASH64
- binlog_transaction_dependency_tracking=WRITESET　# 主库使用何种方法生成并行重放所需的依赖信息，可取值为COMMIT_ORDER（默认值，表示Lock-Based）、WRITESET、WRITESET_SESSION（同一个session内的事务不可并发）。

性能比较



工具：Sysbench

场景：Write Only

MySQL版本：8.0

黄色：COMMIT_ORDER

浅蓝色：WRITESET_SESSION

深蓝色：WRITESET

QINGCLOUD

05

QingCloud MySQL Plus 实践

QINGCLOUD

QingCloud MySQL Plus 实践

数据强一致性

支持金融级强数据一致性，确保业务层面的连续性。

双存储引擎

同时支持InnoDB和TokuDB两种存储引擎。

极致性能

性能优异，按需弹性扩展，大并发负载轻松应对。

读请求负载均衡

业务连接高可用读IP，读请求自动负载到集群多个节点。

服务高可用

支持一主多从架构，多可用区主从部署，
灵活满足各类可用性需求。

主从秒级切换

无中心化选主，业务仅需连接高可用写IP。

自动化运维

自动修复网络、主机、mysql、agent等各类故障。

安全性

运行于私有网络内，多重安全保障，确保数据安全。

QINGCLOUD

QingCloud MySQL Plus 实践

旧形态：

QINGCLOUD

QingCloud MySQL Plus 实践

旧形态：

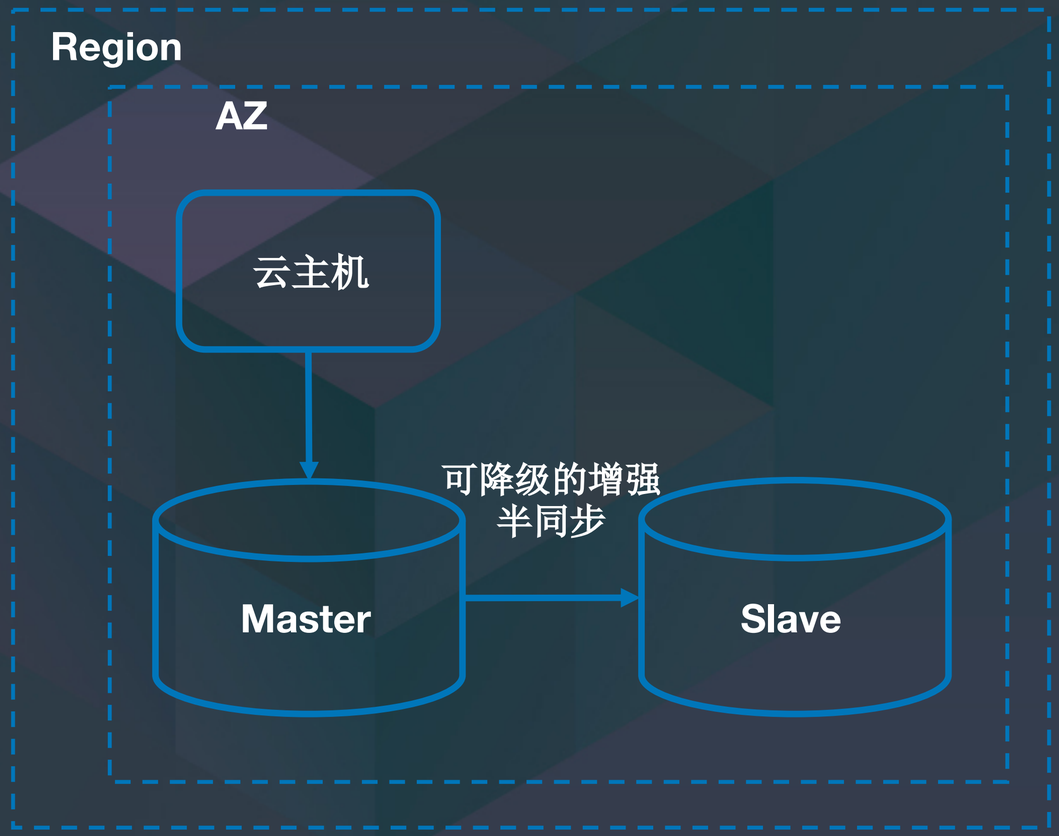
节点	名称	区域	角色	主机类型	节点状态	服务状态	配置	IP	防火墙	告警状态	监控
cln-6s0eq8dq	无	广东2区-B	无	超高性能型	活跃	正常	2核4G 30G	192.168.1.2		无	
cln-rnuowmjd	无	广东2区-B	无	超高性能型	活跃	正常	2核4G 30G	192.168.1.3		无	
cln-okiga2m5	无	广东2区-B	无	超高性能型	活跃	正常	2核4G 30G	192.168.1.4		无	

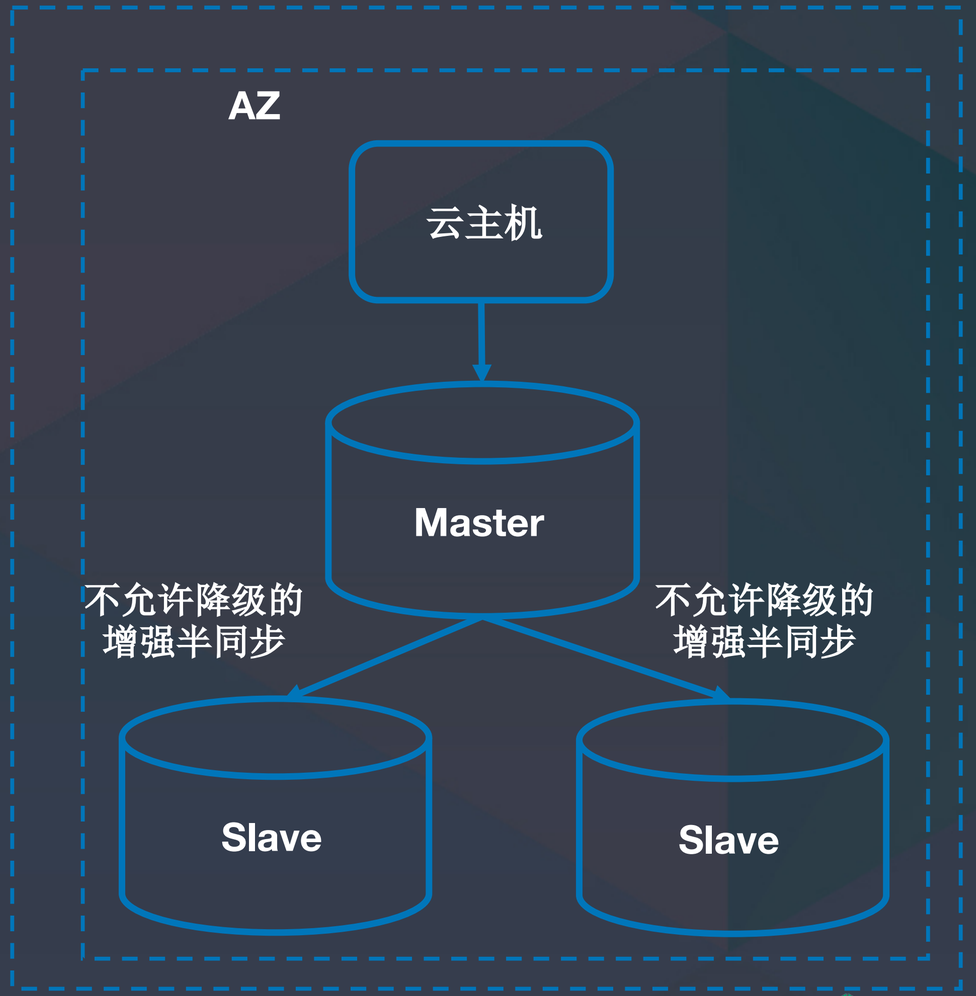
* 提示：可通过在各个资源上点击「右键」来进行常用操作，以及「双击」来修改基本属性。

QINGCLOUD

QingCloud MySQL Plus 实践

旧形态：

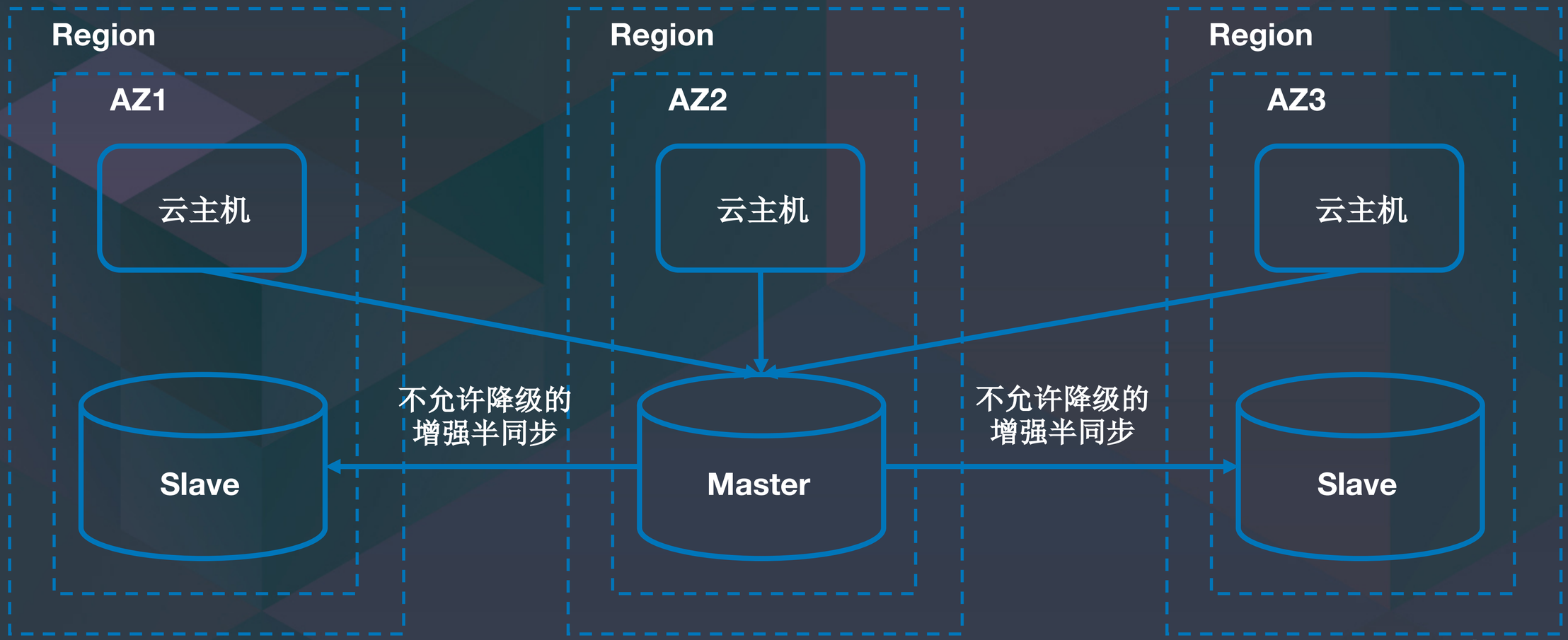




QINGCLOUD

QingCloud MySQL Plus 实践

旧形态：



QINGCLOUD

QingCloud MySQL Plus 实践

新形态：

QINGCLOUD

QingCloud MySQL Plus 实践

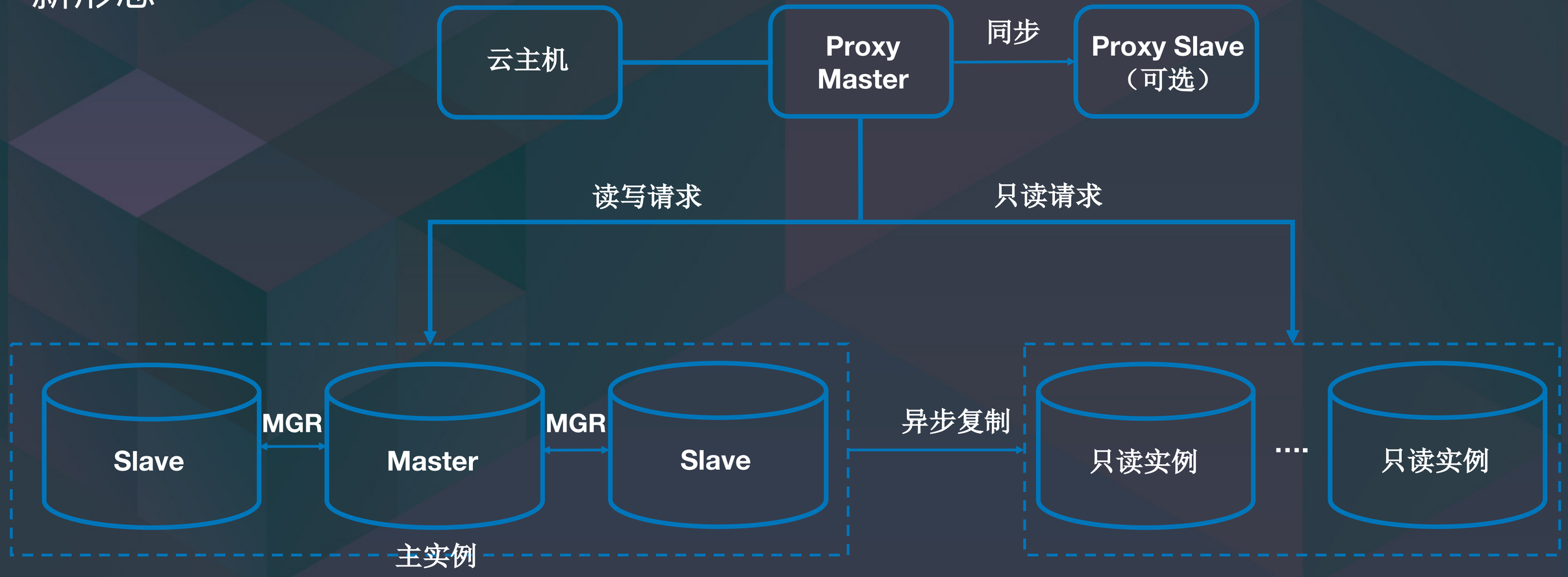
新形态：

节点	名称	区域	角色	节点状态	服务状态	配置	防火墙	告警状态	监控
cln-5fsionti	无	北京3区-D	主实例	活跃	正常	2核8G 10G		无	
cln-ip0nv016	无	北京3区-B	主实例	活跃	正常	2核8G 10G		无	
cln-thhqoogx	无	北京3区-C	主实例	活跃	正常	2核8G 10G		无	
cln-a69fkvih	无	北京3区-C	Proxy 实例	活跃	正常	2核8G 10G		无	
cln-2uvw53cc	无	北京3区-D	Proxy 实例	活跃	正常	2核8G 10G		无	
cln-tmdhxzek	无	北京3区-C	只读实例	活跃	正常	2核8G 10G		无	
cln-0mdj0w2p	无	北京3区-B	只读实例	活跃	正常	2核8G 10G		无	
cln-0glim3dj	无	北京3区-D	只读实例	活跃	正常	2核8G 10G		无	

QINGCLOUD

QingCloud MySQL Plus 实践

新形态：



1. 该图以新形态金融版为例说明，只读请求也可选择负载到Slave节点；
2. Proxy节点、只读实例都是可选的。

QINGCLOUD

QingCloud MySQL Plus 实践

类别	系列	节点数量	复制方式	并行复制技术	优化
旧形态	1.x.x	支持2、3、5个节点。	两节点采用增强半同步复制，超时降级为异步复制； 三节点、五节点采用增强半同步复制，不允许降级为异步复制。	采用组提交中 Lock-Based 的并行复制方式。	1）三节点及以上基于Raft算法实现高可用、选主。 2）读请求负载均衡。 3）自动化运维。
新形态	基础版	1个节点。	单节点，不涉及复制。	单节点，不涉及复制。	
	高可用版	默认2个主实例（与旧形态2节点对应），可添加Proxy、只读实例。	采用增强半同步复制，超时可降级为异步复制。	采用组提交中 Lock-Based 的并行复制方式。	1）Proxy实例做读写分离； 2）读请求负载均衡； 3）只读实例采用异步复制，在主实例切主后自动change master到新主； 4）高可用； 5）自动化运维。
	金融版	默认3个主实例，可添加Proxy、只读实例。	采用组复制（MGR）。	采用组提交中 WRITESET 的并行复制方式。	

Thank you

QINGCLOUD