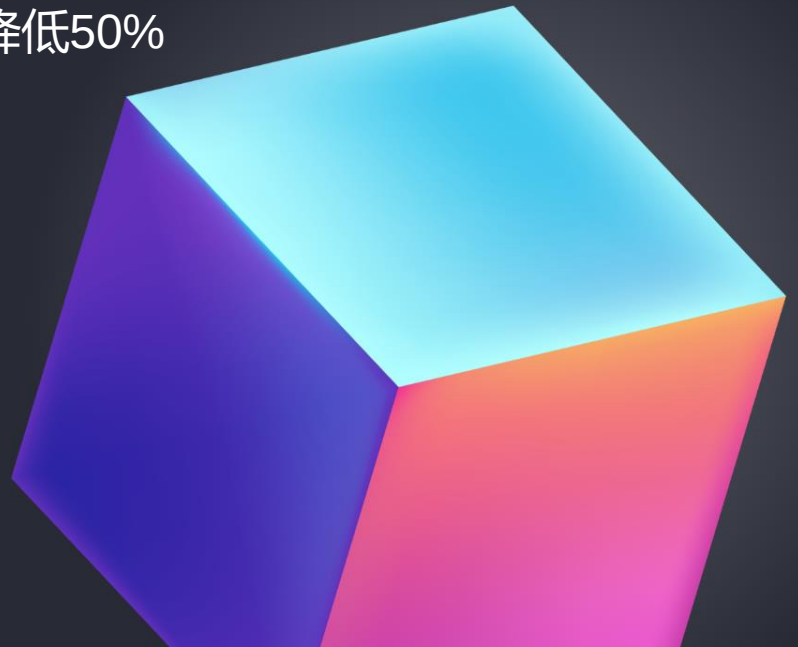
SHOPLINE TiDB4.0到6.5升级之路

得益于TiDB新版本和部署方式的优化，成本降低50%

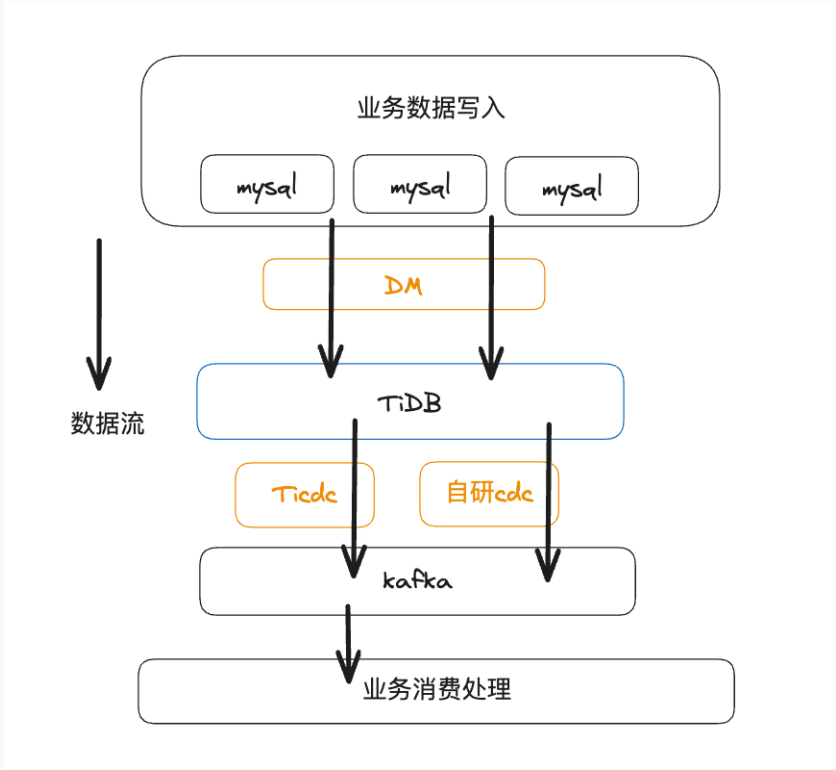


分享人: DBA、TiDB负责人车佳蔚

目录

业务TiDB集群现状

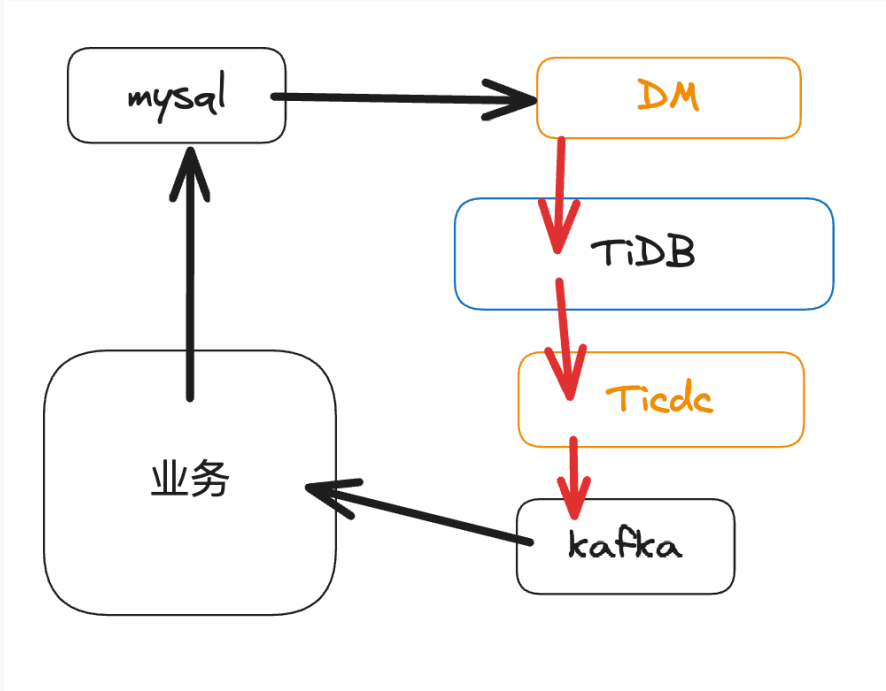
- 10+套TiDB集群，业务范围广泛: 交易、商品、数仓、搜索、广告等
- 大部分业务场景使用数据同步链路 mysql->dm->tidb->ticdc->kfk
- TiDB集群为二进制部署，dbms统一管理
- 部分核心业务共用同一套TiDB集群，复用数据同步链路，降低数据存储的冗余



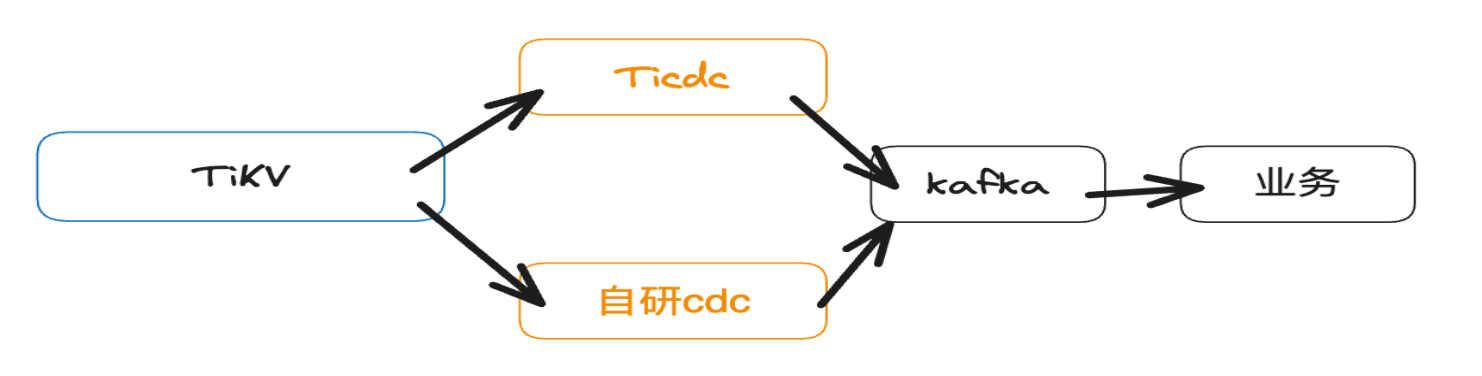
业务TiDB集群的痛点

TiDB集群数据规模增长、业务服务增多

- **核心业务共用同一套TiDB集群，服务间容易互相影响**
- **业务对数据同步链路延迟敏感**
- **mysql ddl变更期间，dm容易产生数据同步的延迟、中断**
- **tikv抖动\kafka抖动 造成 ticdc changfeed任务延迟、中断**



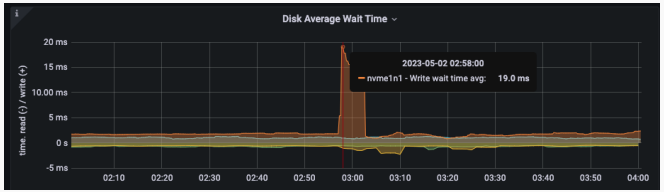
TiKV oom + Ticdc\cdc故障



- 为什么要用自研cdc
  统一mysql\mongo\TiDB 下游binlog格式
  满足业务的binlog定制化需求

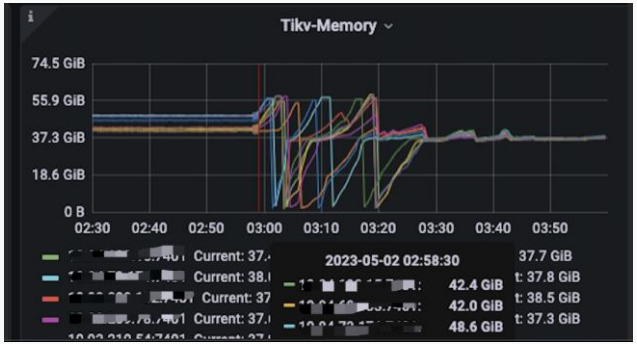
- 2023年4月前，同步链路较稳定；5月后开始出现问题

TiKV oom + Ticdc\cdc故障

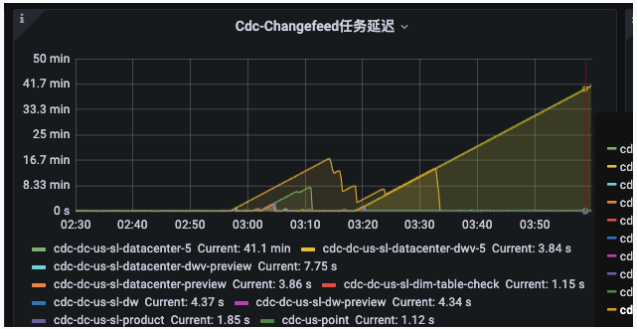


tikv宿主ebs io抖动

tikv leader切换



tikv内存飙升，触发oom



Ticdc\cdc 任务卡死

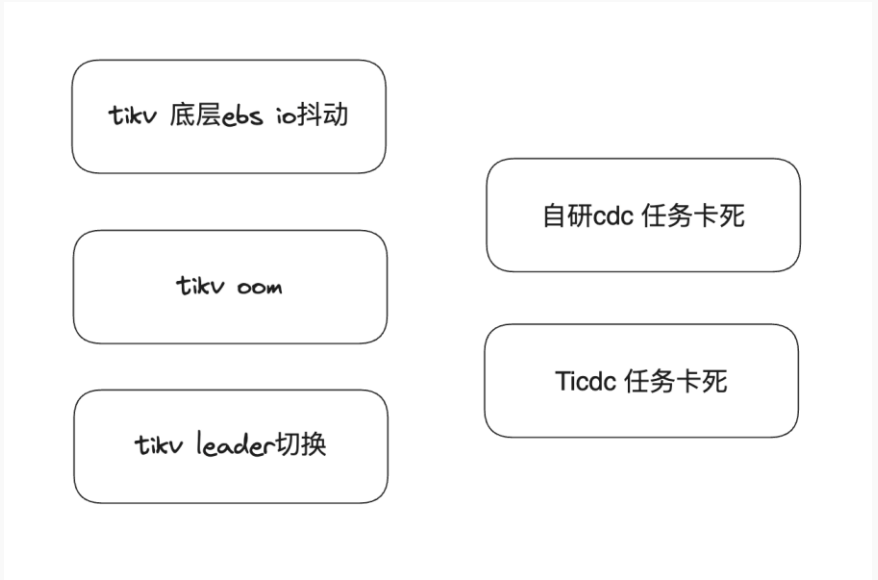
TiKV oom + Ticdc\cdc故障

- **故障的原因**

  直接原因: TiKV底层ebs io抖动

  根本原因: ？

- **面对故障如何快速恢复，SOP方案**
- **长期方案，如何根本解决问题**

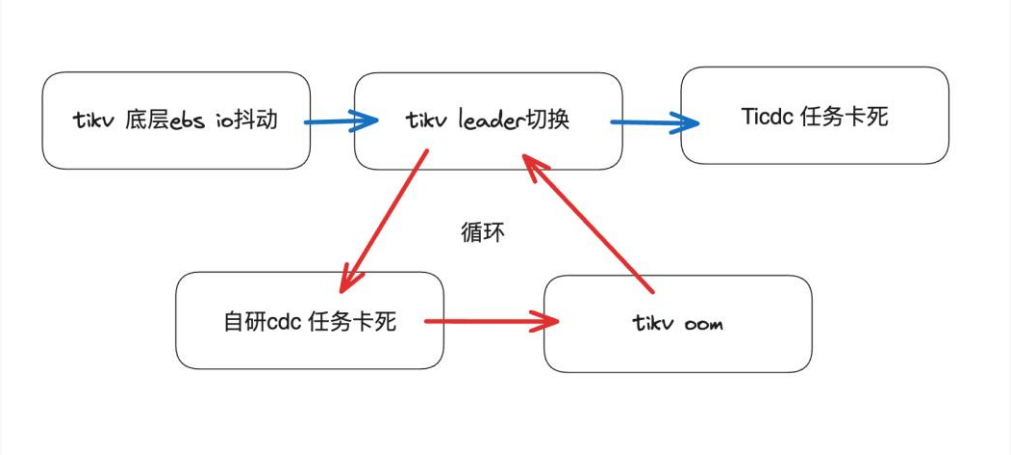


TiKV oom + Ticdc\cdc故障

- **排查**

  故障前TiKV链接数异常升高, 日志相关报错大量出现

  故障前2天 业务大规模接入了cdc订阅服务



- **故障的原因**

  直接原因: TiKV底层ebs io抖动

  根本原因: TiKV悲观锁，大表订阅binlog失败时，cdc重试导致TiKV内存飙升触发oom

TiKV oom + Ticdc\cdc故障

方案A：对以下临时恢复方案，进行验证，并输出结论；

（1）拆分changefeed任务，定位卡主的库表，恢复正常的任务。

（2）count * 故障table。

（3）指定region unlock。

（4）调整kv节点内存大小。

（5）逐一重启kv节点。

备注：尝试恢复预计需20～30分钟。

方案B：select update重发cdc方案；

（1）前置条件：

- 数据域全部切换到cdc (先切重点场景)， 不再依赖ticdc。/
- 识别不出delete 数据。所以不能对delete敏感。
- 表上需要有update_time字段(datetime类型)，或者与update_time等价语义的逻辑，且必须要有索引。
- 表上不能有1分钟以上才commit的大事务。

（2）需要验证吞吐量、10分钟流程验证；

备注：预计10分钟内可重发数据。

方案C：

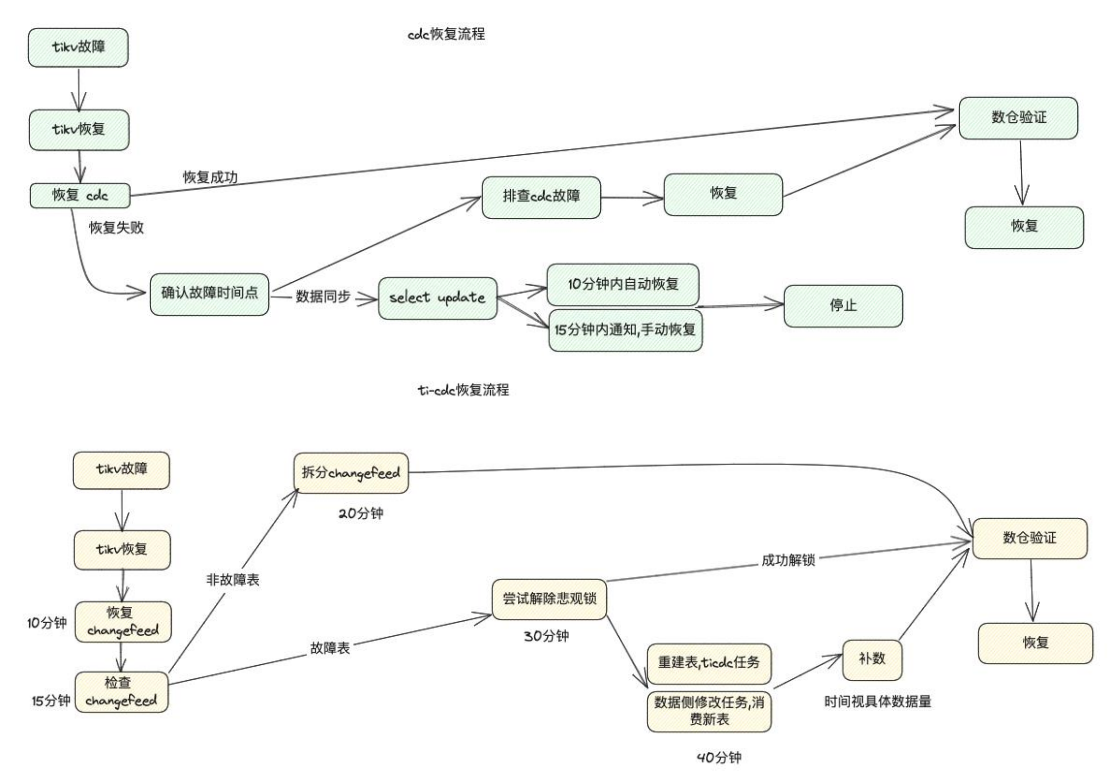
（1）dba创建新表，并同步新表的cdc到旧topic。

（2）数据应用侧修改相关flink代码读写新表的topic，并重启服务，恢复故障期间的历史数据及增量数据。

备注：预计1～2小时内恢复。

- **TiKV轮流 oom期间业务有重试逻辑**
- **但业务对 binlog订阅延迟、中断非常敏感**
- **第一次故障时: 采用 方案C**
- **后续再次出现故障: 方案A + 方案B**

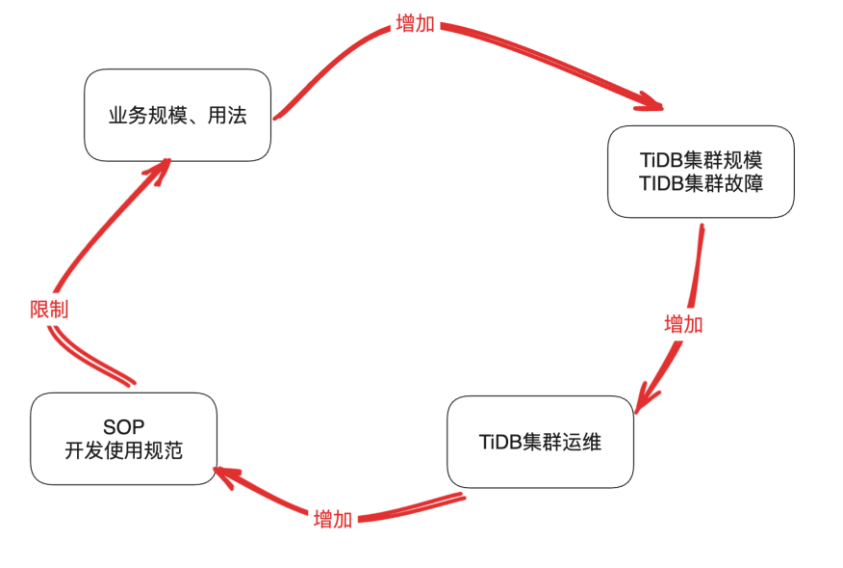
TiKV oom + Ticdc\cdc故障

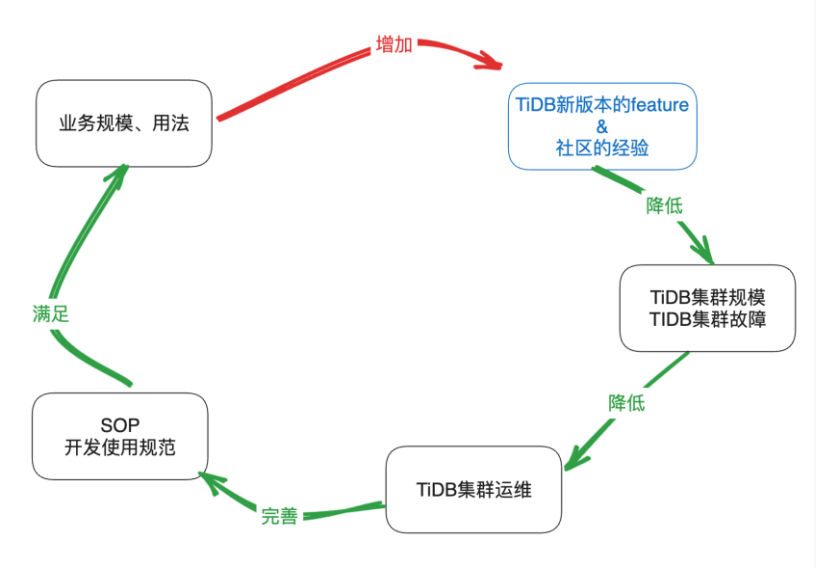


- sop方案流程
- 后续此类故障保证30分钟内恢复
- 如何根本解决问题?
- 测试后: TiDBv6.5.2版本已解决

而且新版本TiDB性能、稳定性也有提升

如何解决业务TiDB集群的痛点

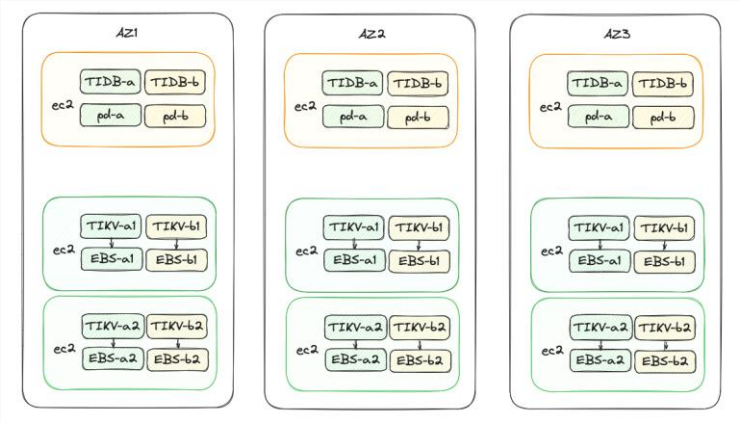


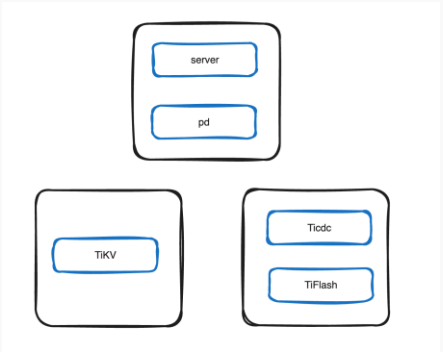


升级准备工作

- 标准化tiup togology.yaml 配置文件
  大部分沿用4.0的配置参数
- 各组件部署规划
- 功能测试-业务测试
- 性能压测-DBA内部压测、业务压测
- 故障演练-TiDB配置变更，各组件节点启停
- 业务会感知到的TiDB版本差异
  sql_mode: 4.0无限制，6.5默认严格模式
  新的字符集框架: new_collations_enabled_on_first_bootstrap

影响:字段结尾包含空格等特殊字符





升级方式

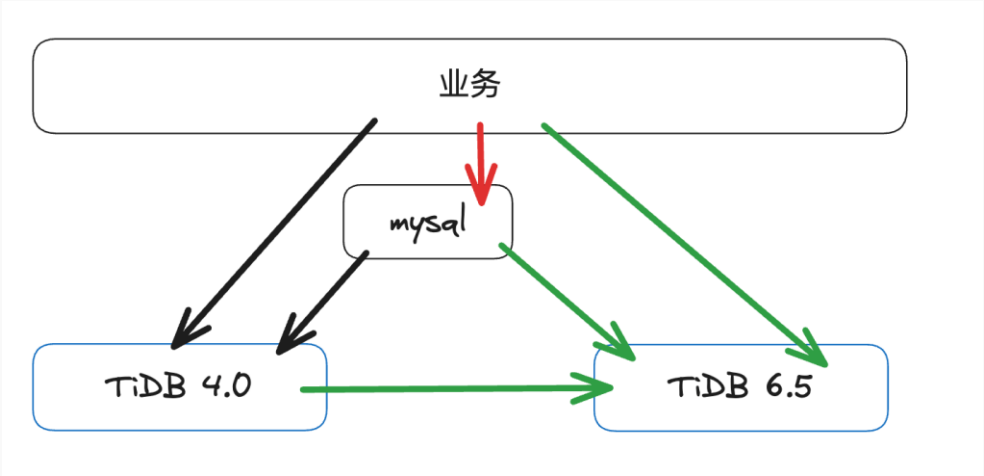
- 根据业务服务场景确定升级迁移方案
  基本每个业务一套迁移文档

通用化的流程规范:

- 集群间数据同步
  TiDB4.0 ->TiDB6.5
  MySQL -> TiDB6.5

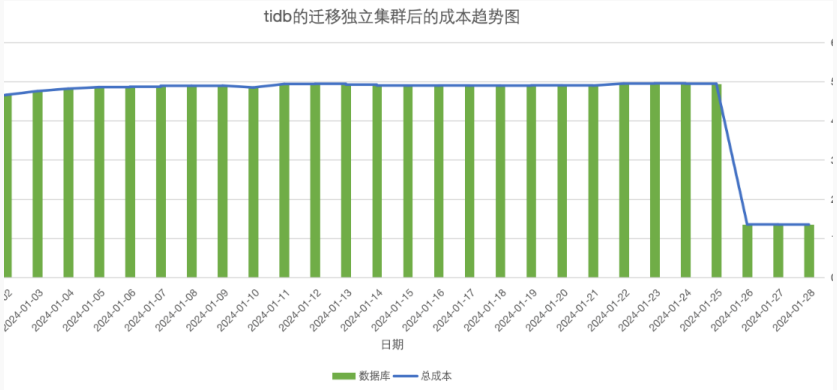
- 表粒度数据同步
  排查定位主键冲突等问题

- 业务切换-灰度放量
  完善回滚策略; 有条件尽量实现双写



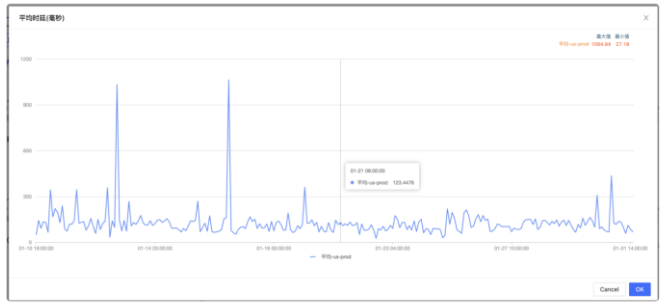
集群升级迁移总结

- 3个月内 0故障升级迁移完成全部4.0集群
- 核心服务按集群独立
- Tiup管理，根据业务特性定制化TiDB参数



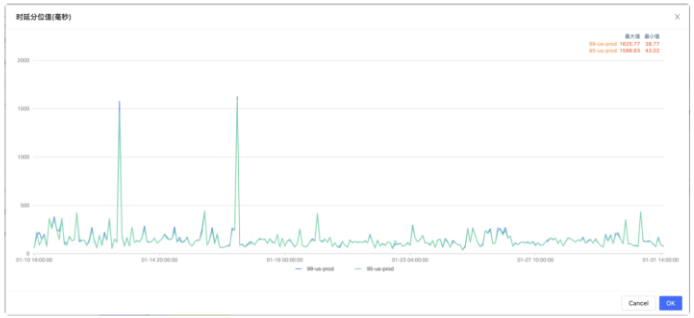
一、平均时延

在集群资源减少过半的情况下，处理请求的平均时间基本维持优化前的水平，平均时延在100ms~130ms。如图：



二、稳定性

迁移到独立集群后，服务稳定性更好，时延尖刺明显减少，服务更平衡。P99 和 P95 都在150ms以内，而优化前 P99 和 P95 的尖刺时延 1s+。如图：



升级TiDB的收益

- 质量: 2023年3月-8月，6起TiDB集群相关故障; 2023年底至今0故障.
- 成本: TiDB集群数量增多，但TiDB机器总量下降50%、单个TiDB集群内部，节点数砍半; server-tikv、tikv间、ticdc-kafka 开启数据压缩，跨可用区流量下降30%.
- 效率: Tiup部署维护便捷, 新集群交付由原来的2小时级别提升至分钟级别， 业务变更、升降配快速且平滑.

TiDB

THANK YOU.