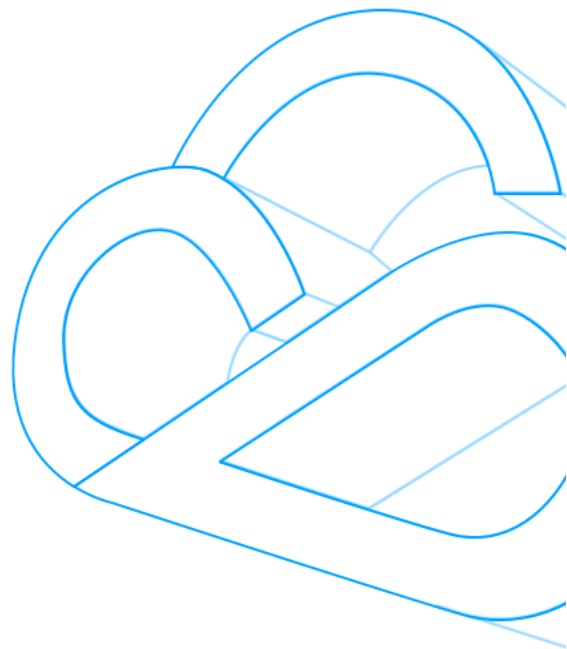




腾讯云数据库 TDSQL MySQL 版 分布式 V10.3.22.8/集中式 V8.0.22.8 常见问题



文档版本:

发布日期:

腾讯云计算（北京）有限责任公司

版权声明

本文档著作权归腾讯云计算（北京）有限责任公司（以下简称“腾讯云”）单独所有，未经腾讯云事先书面许可，任何主体不得以任何方式或理由使用本文档，包括但不限于复制、修改、传播、公开、剽窃全部或部分本文档内容。

本文档及其所含内容均属腾讯云内部资料，并且仅供腾讯云指定的主体查看。如果您非经腾讯云授权而获得本文档的全部或部分内容，敬请予以删除，切勿以复制、披露、传播等任何方式使用本文档或其任何内容，亦请切勿依本文档或其任何内容而采取任何行动。

商标声明



“腾讯”、“腾讯云”及其它腾讯云服务相关的商标、标识等均为腾讯云及其关联公司各自所有。若本文档涉及第三方主体的商标，则应依法由其权利人所有。

免责声明

本文档旨在向客户介绍本文档撰写时，腾讯云相关产品、服务的当时的整体概况，部分产品或服务在后续可能因技术调整或项目设计等任何原因，导致其服务内容、标准等有所调整。因此，本文档仅供参考，腾讯云不对其准确性、适用性或完整性等做任何保证。您所购买、使用的腾讯云产品、服务的种类、内容、服务标准等，应以您和腾讯云之间签署的合同约定为准，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

修订记录

文档版本	发布日期	修订人	修订内容
------	------	-----	------

目录

修订记录..... ii

目录..... iii

前言..... iv

1 产品特性类..... 1

2 问题/方案咨询 8

3 数据迁移类..... 20

4 应用开发类..... 22

5 运维类..... 29

6 备份与恢复..... 45

7 性能优化类..... 47

8 安装部署类..... 49

9 安全合规类..... 50

前言

文档目的

本文档用于帮助用户掌握云产品的操作方法与注意事项。





目标读者

本文档主要适用于如下对象群体：

- 客户
- 交付 PM
- 交付技术架构师
- 交付工程师
- 产品交付架构师
- 研发工程师
- 运维工程师

符号约定

本文档中可能采用的符号约定如下：

符号	说明
 说明：	表示是正文的附加信息，是对正文的强调和补充。
 注意：	表示有低度的潜在风险，主要是用户必读或较关键信息，若用户忽略注意消息，可能会因误操作而带来一定的不良后果或者无法成功操作。
 警告：	表示有中度的潜在风险，例如用户应注意的高危操作，如果忽视这些文本，可能导致设备损坏、数据丢失、设备性能降低或不可预知的结果。
 禁止：	表示有高度潜在危险，例如用户应注意的禁用操作，如果不能避免，会导致系统崩溃、数据丢失且无法修复等严重问题。

1 产品特性类

生产环境主备实例数量

场景描述：生产环境可以部署 1 主 1 备实例吗？

场景说明：存在一定风险，在备宕机时导致只读账号无法连接。

场景建议：生产环境必须至少 1 主 2 备。

Gtid 对 dcn 的影响

场景描述：目标端已执行 gtid，对 dcn 回切有哪些影响？

场景建议：

1、新建 DCN 关系时，要求备实例是空实例，且不能忽略 DB 检查。

2、通常忽略 DB 检查用于以下场景：DCN 关系断掉，但两边数据基本一致，此时重建 DCN 关系。

TDSQL-MySQL 集群是否支持 IP 地址变更？

场景描述：TDSQL-MySQL 集群是否支持 IP 地址变更？如不支持，是否支持域名化部署？集群内部通讯是否只能通过 IP 地址？

场景说明：TDSQL 组件 IP 变更的话需要下架重新初始化再上架，集群内部以 IP 地址通讯。

TDSQL 数据库主备切换耗时是多长时间

场景描述：TDSQL 数据库主备切换耗时是多长时间？在主备切换过程中数据库是否对外提供完整服务？

场景说明：在没有备库延时情况下小于 1 分钟。切换过程中会影响使用。

TDSQL 数据库在分布式模式下在线扩容分片，是如何操作

场景描述：TDSQL 数据库在分布式模式下在线扩容分片，是如何操作？扩容过程中数据库是否对外提供完整服务？是否有短暂的服务中断？

场景说明：支持可视化操作，具体操作见运维手册。扩容过程中在路由切换时会有短暂中断。

参考文档：

公有云：TDSQL MySQL 版官网-弹性扩展-产品简介-文档中心-腾讯云

私有云：实例的弹性扩展能力

TDSQL 数据库在做在线 DDL 操作过程中，数据库是否对外提供完整服务？

场景描述：TDSQL 数据库在做在线 DDL 操作过程中，数据库是否对外提供完整服务？表访问是否有短暂的中断或者只读？

场景说明：如果是更新表结构操作会有短暂中断。建议在低峰期用赤兔 Online DDL 操作，以降低业务影响风险。

TDSQL 是否支持 drop/truncate 分区这种操作

场景描述：TDSQL 是否支持 drop/truncate 分区这种操作？

场景说明：支持 alter table table_name drop partition partition_name

TDSQL 数据库的逻辑备份支持哪些工具

场景描述：TDSQL 数据库的逻辑备份支持哪些工具？是否有图形化操作工具？是否支持备份压缩？是否支持选择数据库对象进行备份？（例如：选择备份数据库个别数据表）

场景说明：都支持。Mydumper，MySQLdumper。

读写分离参数设置

场景描述：如何设置读写分离参数。

场景说明：

rw_split:1,通过 hint 控制是否读从机；（推荐）

rw_split:2,事务内读写主库，其他读从

4 种读写分离方案：

TDSQL 默认支持读写分离能力，架构中的每个从机都能支持只读能力，如果配置有多个从机，将由 SQL Engine 集群（SQL Engine）自动分配到低负载从机上，以支撑大型应用程序的读取流量。

TDSQL 提供以下多种读写分离方案供您选择，且您无需关注若干从机是否完全存活，因为系统将根据策略自动调度。

只读帐号（推荐方案）：

您仅需要在创建帐号时，标记为只读帐号，系统将根据只读策略向将读请求发往从机；只读策略可以根据主从延迟等维度进行灵活配置。

/*slave*/注释（推荐方案）：

您可以在编程过程中，通过注释/*slave*/，系统将把该条语句发往从机，常用于编程阶段将特殊的读逻辑嵌入代码。

全局自动读写分离：

该配置会自动将 SQL 中的所有读请求发向从机，且能识别事务、存储过程中的读语法并灵活处理。

如果从机延迟较大，全局自动读写分离并不具备应对策略，读到的数据可能会有延迟。若使用该功能，性能有大约 50% 的下降。该功能默认未开启。

只读实例：

您也可以自建或申请只读实例，只读实例是专用于读请求的一种实例，不参与高可用切换。读写分离将为您的应用提高总的读取吞吐量。通过多种只读方案的组合，您可以配置出复杂的只读方案，以满足各种业务需求和开发的灵活性。

是否已经支持 flashback

场景描述：是否已经支持 flashback，实现原理机制是怎样的。

场景说明：

库表级 flashback 实现思路：在 TDSQL(MySQL 5.7)版本中实现了 Flashback。

binlog 文件由多个 event 组成：如 ALTER、DROP 等语句在 binlog 中以 Query_event 记录，Write_rows_event 表示 ROW 模式下插入一行数据。

通过"三层反转"实现 Flashback。

- 1、单行数据反转：也就是单个 event 反转(insert 反转为 delete, delete 反转为 insert, update 交换 set 和 where 部分)；
- 2、单个更新 sql 反转：一个 sql 可能更新的多行数据，则需要将单个 sql 更新的行更新顺序反转(第一行反转为最后一行，最后一行反转为第一行)；
- 3、整体 sql 反转：全部 binlog 在整体时间上反转(最开始更新的 sql，反转后，最后执行；最后更新的 sql，反转后，最先执行)。

库表级 flashback 的基础上，当前又实现了记录级过滤和 flashback。在使用 mysqlbinlog 解析 row 格式的 binlog 文件时指定 --filter-rows 过滤条件，输出满足条件的 row_event，达到获取单记录变更流水的目的。配合 --flashback --flashback-tables 选项可以将这条 binlog row 逆转，达到单记录回档的目的。

一套管控纳管最大实例个数

场景描述：一套管控纳管最大实例个数。

场景说明：建议不超过 10 个。

primary(分片键,B) primary(B,分片键)区别

场景描述：primary(分片键,B) primary(B,分片键)这两种有区别吧，如果查询条件 B,不带分片键会怎样。

场景说明：不带分片键的查询，但把查询请求发到所有分片上执行，再到网关做聚合；建议可以用 primary(B,分片键) 这种方式，查询条件 B 有主键索引。

proxy 是不是也有大小写敏感参数设置

场景描述：proxy 是不是也有大小写敏感参数设置。

场景说明：proxy 不区分，proxy 会从存储节点获取这个配置来做相应处理。

分布式实例是否支持变量

场景描述：分布式实例是否支持变量。

场景说明：部分简单的变量使用可以，由于没有做过完全的兼容性开发，建议谨慎使用。

分布式实例是否支持 union

场景描述：分布式实例是否支持 union。

场景说明：支持。

是否支持多语句一次性发到数据库

场景描述：

1.TDSQL 是否支持多语句一次性发到数据库，比如一条 sql 是 set1，一条 sql 是 set2？

2.如何保证不会有问题？

3.比如某个 set 一直不返回，或者比较慢，会出现什么情况？如何解决？

场景说明：

1.支持；

2.跨 set 的相当于分布式事务处理了，单节点的事务由 InnoDB 引擎保障，多节点分布式事务采用标准的两阶段提交协议实现异常处理，有全局死锁检测机制来保障；

3.会根据设置的超时时间处理，超时回滚（默认 50s，默认时间可在赤兔平台修改）。

TDSQL 集中式架构与分布式架构的区别

场景描述：TDSQL 集中式架构与分布式架构的区别？比如集中式架构用到哪些组件，分布式架构用了哪些组件？

场景说明：组件上都是一样的。一样要经过 proxy，同一个集群，只是创建实例的时候，类型不一样；详见“《白皮书》腾讯云数据库 TDSQL 解决方案 > 产品架构”。

TDSQL 错误码相关含义

场景描述：咨询分布式事务未知错误的含义。

场景说明：详见“参考 > 错误信息参考”。

TDSQL 数据库物理备份机制是什么

场景描述：TDSQL 数据库物理备份机制是什么？可以支持哪些备份软硬件？备份恢复是否支持跨群组恢复？（例如：生产数据库备份如何恢复到准生产数据库环境）。

场景说明：支持跨集群恢复(TD 到 TD)，详见“参考 > 系统原理 > 强同步复制”。

proxy 机器，除系统盘外，nvme 盘为什么还需要？

场景描述：proxy 机器，除系统盘外，nvme 盘为什么还需要？机制是什么？是否可以直接用系统盘？

场景说明：详见“部署调测 > 部署规划 > 资源规划”。

极端场景下，此时故障 DB 时，是否可以应急拉起？

场景描述：正常运行情况下，管控全部离线，-极端场景下，此时故障 DB 时，是否可以应急拉起？具体需要描述各组件；同时，在不能自动拉起状态下，应急处理方式和顺序步骤。

场景说明：可以自动拉起，先恢复 zk 集群。

组件升级，功能缺失

场景描述：升级 17.3 版本组件后，发现【实例详情】-【设置数据同步方式】-【强同步】功能标签【同 IDC/AZ 提升为强同步】缺少。

场景说明：赤兔页面发现页面【强同步】功能标签【同 IDC/AZ 提升为强同步】缺少问题，该功能在可以通过。

（1）设置【可退化】时默认不可提升。

（2）设置【不可退化】时默认可提升。默认值通常是作为常规设置需求，对实际使用影响不大。

部分赤兔界面功能不可用

场景描述：部分赤兔界面功能不可用，做修改甚至可以导致故障，如修改机器 IDC，这些有问题的功能需要给出清单。

场景说明：参考《赤兔用户手册》。

赤兔是否支持 https

场景描述：赤兔是否支持 https。

场景说明：支持。需要有 https 认证。

数据库不支持 LOAD DATA， 是否有替换方案

场景描述：数据库不支持 LOAD DATA， 是否有替换方案。

场景说明：TDSQL 有专门的 load_data 工具，功能跟原生相似，少量不常见用法不支持。

`/*sets*/ hint` 支持一个关键字到第一个 set 上

场景描述：`/*sets*/ hint` 支持一个关键字到第一个 set 上。

场景说明：可以用这个 hint `/*shardkey_hash:0*/ SQL`，将 SQL 发送到第一个 set 上。

分布式事务问题

场景描述：

1. 是不是必须要用 `begin` 才是分布式事务，才能在 proxy 层保证事务？
2. 2. 什么情况下建议使用分布式事务？事务里面更新的表必须才跨 set 才建议使用？

mysql 原事务写法：

```
START TRANSACTION;  
commit;
```

TDSQL 分布式事务写法：

```
beign;  
commit;
```

场景说明：分布式用 `START TRANSACTION` 和 `begin` 都是一样的;开发人员不用关注分布式事务，当普通 mysql 事务执行就行，分布式事务是内部机制(不用管是不是分布式事务，都用 `START TRANSACTION` 开始即可)。

业务到 proxy 连接，proxy 到 db 连接，是同一个连接，还是不同连接

场景描述：业务到 proxy 连接，proxy 到 db 连接，是同一个连接，还是不同连接？

场景说明：连续发请求，或者开启事务，都能保证是同一个连接，最好是开启事务，能 100% 保证。

逐个 set 提交，还是可以通过 proxy 透传提交

场景描述：如果有些特殊的函数，过程，操作只在本 set 内，提交函数、存储过程，如何操作？是要逐个 set 提交，还是可以通过 proxy 透传提交？

场景说明：两种均支持；1.通过 proxy 透传提交，透传了就当普通的 mysql 用了，都可以用，不用逐个提交。`/*sets:allsets*/ xxxx`2.支持透传到所有 set 上执行，但使用时也需透传使用，因为 proxy 无法识别。

建议：透传功能建议业务上尽量少用或不用，函数或存储过程涉及逻辑多且复杂，无法保证操作潜在风险。

升级 oss 这个组件，接口会不会发生改变

场景描述：升级 oss 这个组件，接口会不会发生改变。

场景说明：详情请参考对应版本的《TDSQL-OSS 使用文档》。

LVS 是否需要单独两台物理机器

场景描述：LVS 是否需要单独两台物理机器。

场景说明：详见“部署调测 > 安装部署 > 安装数据库（命令行部署） > 安装 LVS”。

在 dbeaver 上执行 join 报错

场景描述：

- 1、分片表情况，想要修改主键，要如何操作；
- 2、一级分区使用 hash，添加一个 set，数据要不要重新分布，如何做，对业务有没有影响；
- 3、一个 mysql 实例，能否有多个 set。

场景说明：

- 1、分片表不允许修改主键。
- 2、可先参考下产品白皮书内“实例的弹性扩展能力”对应章节介绍内容。
- 3、需明确一下 set 的定义，我们的定义下，set 是指分片，分布式实例就是由多个 set 组成。

分片问题咨询

场景描述：业务到 proxy 连接，proxy 到 db 连接，是同一个连接，还是不同连接？

场景说明：连续发请求，或者开启事务，都能保证是同一个连接，最好是开启事务，能 100% 保证。

导入导出数据工具，以及 dbbridge 的相关文档

场景描述：导入导出数据工具，以及 dbbridge 的相关文档。

场景说明：

mydumper/myloader：详见 [Mydumper/MyLoader 工具使用说明](#)。

load_data：详见 [Load_data 工具使用手册](#)。

数据库应用服务迁移 DTS-DBbridge：详见 [数据库应用服务迁移 DTS-DBbridge](#)。

时间操作

场景描述：oracle 的 2 个时间相关，得到的是以天为单位的浮点数，tdsql 对应什么方案？

场景说明：timestampdiff 函数，参数是 date 或者 datetime 类型。

例如：

```
select timestampdiff(microsecond, '2022-09-08 11:35:00.123456', '2022-09-08 11:40:00.345678')/1000 as time_in_ms;
```

2 问题/方案咨询

如何对 TDSQL 稳定性测试

场景描述：如何对 TDSQL 稳定性测试

场景建议：采用 sysbench 工具对 TDSQL 环境做压力测试，评估系统稳定性、架构合理性。

如何将 TDSQL 告警推送到 Zabbix

场景描述：将告警内容传递给 Zabbix，用于大屏展示，并要支持告警消除。

场景建议：采用直接通过脚本查询数据库，编写脚本程序来控制定时推送。按 Zabbix 提出的需求去设计脚本和 SQL 语句。最终达到将告警去重后推到 zabbix，并提供标识供 zabbix 做告警消除。

如何将 TDSQL 跟原有系统告警系统对接

场景描述：TDSQL 跟厂商告警系统对接，但有部分告警指标不能直接输出。

场景建议：api 和数据库方式方案，详见“部署调测 > 对接调测”。

生产库如何更改字符集和排序规则

场景描述：生产库更改字符集和排序规则。utf8 改为 utf8mb4 排序规则 utf8mb4_bin。

场景建议：导出数据-->更改原数据库名--->新建库（字符集选择 utf8mb4,排序规则：utf8mb4_bin）--->导入数据

DB 退化

场景描述：备 IDC 的两台 DB 同时宕机的时候，主 DB 仅用了 7 秒左右就退化完成（不是原定的退化时间 15 秒）

场景说明：

判断退化或提升有几种：

1. agent 上报 db 挂掉。
2. agent 自己挂掉，10s zk 会话超时会被 scheduler 感知到，3 秒上报心跳，所以即将上报时 agent 挂掉在过 7 秒 scheduler 就感知到了。
3. iodelay 超 15 秒。

告警显示的逻辑和恢复的逻辑

场景描述：确认告警显示的逻辑和恢复的逻辑。

场景说明：

如果配置了备份 idc，备份 idc 满足 30 分钟延迟内的 原备份节点或其他备机成为新备份节点。

如果没配置，则选择为满足 30 分钟延迟的原备份节点，或者选择一个最久没做主库的成为新备份节点。【因为一般切换为切换到同机房，所以可以大致认为备机房是最久没成为主的，也就是大概率选择备机房的备机成为新备份节点。】

等保问题

场景描述：等保问题。

场景说明：

- 1、赤兔的连接完整性和保密性，建议通过开启 https 实现。
- 2、数据库的连接完整性和保密性，通过限制登录 ip 可以风险值降级，ssl 是认可的加密算法，但是要看有没有什么负面影响，比如是否影响业务。

计算节点=分片数量，是否可行？

场景描述：计算节点=分片数量，是否可行？

场景说明：支持计算节点等于分片数量，计算节点跨双中心部署场景数量是 2 的倍数。

单个分片数据量参考值

场景描述：单个分片要求存放 2T 数据，是否可行？

场景说明：原则上 1T 作为分片标准，如果存有大量历史数据或冷数据的场景，可以考虑存放至 2TB。实际业务场景建议结合测试情况评估。

分布式事务控制是由数据库实现还是应用层实现？

场景描述：在分布式场景下，分布式事务控制是由数据库实现还是应用层实现？哪些应用做？哪些数据库做？

场景说明：一个 groupshard 内的分布式事务由数据库实现，跨实例的分布式事务由应用实现，比如使用分布式事务框架。

由于微服务拆分导致的分布式事务由应用侧事务组件解决。

由于数据库拆分导致的分布式事务由 TDSQL 事务组件解决。

TDSQL 分库分表原则，可量化的指标，评估模型

场景描述：TDSQL 库表数据量规范最佳实践或行业通用建议，如何满足未来数据量增长需求？TDSQL 分库分表原则，可量化的指标，评估模型。

场景说明：参考值：对于分布式(Shard)实例，每个分片(set)建议<2TB；对于分布式(Shard)实例，单个分片表(shardtable)，建议 5KW 行-10KW 行（10G~100G）；具体需要结合业务场景评估、测试确认一个合适的值。

序列功能是由应用实现还是数据库实现，需给出建议

场景描述：序列功能是由应用实现还是数据库实现，需给出建议。

场景说明：建议应用侧实现。无法支持全局顺序场景。

表分片是下发到所有分片还是精确落在数据分片执行？

场景描述：表分片后，带范围查询的语句 where id>1000 and id<10000，此类语句执行 TDSQL 是下发到所有分片还是精确落在数据分片执行？是否需要应用干预？

场景说明：hash 模式需要是等值条件（id = 1、id IN (1,2)）才能定位到分片，范围查询会分发到每个分片。

是否支持嵌套事务

场景描述：集中式和分布式事务问题，是否支持嵌套事务。

场景说明：集中式支持 savepoint，分布式实例不支持。

proxy 与 db 之间的连接机制

场景描述：proxy 与 db 之间的连接机制，是长链接还是短连接？是否有绑定关系？应用，proxy，db 之间的连接参数如何配置？

场景说明：tdsql proxy 默认根据客户端链接的生命周期来管理和 db 的连接，即新建客户端链接时按需创建 proxy 与 db 的链接，并在客户端链接断开时断开 proxy 与 db 的链接。当前业务一般会使用连接池来访问 tdsq，此时 proxy 与 db 之间的连接也是长链接。DB 侧有 max_connections=10000，proxy 没有连接参数限制。

DCN 选择延时最小节点同步的选择与切换策略

场景描述：关于 DCN 选择延时最小节点同步的选择与切换策略需确认

场景说明：TDSQL 初始选择延时最小副本，只有当该副本宕机时才再切换其他副本。

TDSQL 数据库在分布式模式下在线扩容分片，应该如何操作

场景描述：TDSQL 数据库主备切换耗时是多长时间？在主备切换过程中数据库是否对外提供完整服务？

场景说明：支持可视化操作，具体操作见运维手册。扩容过程中在路由切换时会有短暂中断。

参考文档：

公有云：[tdsql-mysql 弹性扩展 腾讯云文档](#)

私有云：详见“白皮书> 产品功能 > 弹性的扩展能力”。

proxy 宕机会不会有机制自动拉起来

场景描述：proxy 上层有负载均衡，proxy 宕机会不会有机制自动拉起来。

场景说明：如果进程宕了，会有 Agent 自动拉起。Agent 宕了有定时任务检测拉起 Agent。详见“管理数据库 > 故障处理 > Proxy 机器故障恢复方案”。

proxy 层单中心、双中心

场景描述：proxy 层单中心、双中心对于应用来说，是几个地址配置，不同负载均衡策略下如何使用。

场景说明：

- 1、每个机房暴露一个 VIP，应用需要配主备中心两个 VIP，并优先选择本中心的 VIP 进行访问，本中心 VIP 优先选择本中心 Proxy 访问。
- 2、两个机房暴露一个 VIP，应用只需要配一个 VIP，由 F5 配置优先选择主中心 Proxy 访问。

核心系统数据量评估

场景描述：核心系统数据量的评估，结合发展趋势评估是采用 TDSQL 的集中式架构还是分布式架构。

场景说明：如果核心系统采用了单元化架构，则 TDSQL 使用集中式架构（超大体量）。如果核心系统未采用单元化架构，则 TDSQL 使用分布式架构（中等体量）。另外需要具体结合业务的整体数据量、TPS 等因素做综合考虑。

故障 DB 时，如何应急处理

场景描述：DB 故障时的应急处理步骤。

场景说明：可以自动拉起。如果管控集群故障了，则先恢复管控集群。详见“管理数据库 > 故障处理”。

逻辑表与物理表的对应关系

场景描述：建表的时候，是否会指定对应物理表的数量，这个和 SET 的数量是否一致？例如某个集群有 3 个 SET，某个逻辑表是否可以切成 6 个物理表？

场景说明：建表后，逻辑上对应用来说表只有一张（逻辑表），如果只做一级分区（指定 shardkey），物理表是跟 set 数量对等的，数据跟 set 数量一致。

一级hash，自动按set数拆分，底层为原生hash list，proxy直接路由	一级分区决定数据去那个set 二级分区决定数据去哪个物理分区（表）
一级range，按range分布在set单表,proxy直接路由，底层不再分区分表	
一级list，按list分布在set单表,proxy直接路由，底层不再分区分表	
一级hash二级range，range在set上分布为多个物理表，一个值一张物理表，每个物理表再通过set对应的hash原生分区	一级hash二级list分区，底层按list 分为对应list的多个物理表，每个物理表再通过set对应的hash原生分区
一级list二级range分区:通过list值在不同set上分布为单表，单表上按range进行原生分区，不再分物理表	一级list二级list分区::通过list值在不同set上分布为单表，单表上按list进行原生分区，不再分物理表
一级range二级range分区，通过一级range值在set上分布为单表，单表通过二级range进行原生分区，不再分物理表	一级range二级list分区:通过一级range值在set上分布为单表，单表通过二级list进行原生分区，不再分物理表

广播表数据量

场景描述：广播表数据量及写入量阈值。

场景说明：

单表 建议数据 1 万行内 tps

广播表 建议数据 1 万行内 只允许运维变更，建议用于配置表。

广播表容错机制

场景描述：广播表容错机制是什么。

场景说明：广播表通过分布式事务来保证数据一致性；通过调整 tcp_retries2 系统参数和 proxy 的 keepalive 参数降低影响时长。

如何多源复制

场景描述：在分库结构下，从一个库复制共享表到其它数据库，采用源复制，需要应用容忍延迟和一定的错误。

场景说明：使用 binlog+kafka，当复制一个节点故障，会不断重试，复制操作不会给目标表加锁。强烈建议复制表需要主键，并且源端、目标端结构保持一致。

流式查询数据传输原理

场景描述：流式查询数据传输原理，在集中式和分布式两种实例上使用 jdbc 流式查询时，SQL 执行和数据传输过程是怎样的。

场景说明：

集中式

mysqld 循环写查询结果集，由 tcp 流量控制机制调节发送速率，每批次写入量受 `max(net_buffer_length, 单行大小)` 影响；（结果集未吐完时，连接状态为 `Sending to client`，会 hold 住 readview）

分布式

- 1、可以下推的*简单*查询：proxy 利用 IO 复用从多个后端分片读取结果集，发送给客户端；
- 2、复杂查询：proxy 将结果集写入到临时表（磁盘文件）中再发送给客户端。

业务端感知的现象以及数据一致性如何保证

场景描述：2pc 各阶段异常下，业务端感知的现象以及数据一致性如何保证。

场景说明：

xa prepare 异常，abort 并且返回错误；

xa commit 部分异常，返回错误并保证最终一致，agent 定时补偿，通常。

业务可通过 `select gtid() + select gtid_state()` 确认事务状态；

开启读一致性（gts）的情况下，commit 之后的事务在读取 prepared 状态的数据时会触发阻塞，直到 agent 完成事务补偿为止。

两地三中心的解决方案下 LVS 的部署

场景描述：两地三中心的解决方案，LVS 的两台机器如何部署。

场景说明：

- 1、LVS 机器建议单独部署，不建议和其他组件混合部署。
- 2、生产环境规划中，LVS 设备，不可与 proxy，db 等机器复用。独立的 LVS 硬件资源以最大化确保接入层服务稳定。
- 3、需要额外增加 LVS 设备。
- 4、在资源紧张的场景下，可以考虑采用应用驱动层做负载均衡。

同步延迟

场景描述：如果发生同步延迟，如何追数据，有没有可以落实的解决方案。

场景说明：预防为主

场景一：大事务

【解决方法】TP 业务，应用程序控制、数据库设置超时。

场景二：同步压力过大，达到资源瓶颈。

【解决方法】按目前生产配置下，测试支撑的最大吞吐量，设定监控和告警阈值，应用限流，及时切换流量。

场景三：个别通道同步压力过大。

【解决方法】

（1）拆分通道

(2) 增加同步并发

(3) 扩展机器

场景四：对计划内的流量高峰，比如：活动，“双十一”等。

【解决方法】提前进行扩容。

分区表使用了 global 索引如何适配国产化

场景描述：分区表使用了 global 索引如何适配国产化？

场景说明：不支持 global 索引；建议拆表，拆表后分区数相应减少。

使用了分区交换方式进行 T-1 数据迁移历史流水表，国产化如何适配

场景描述：使用了分区交换方式进行 T-1 数据迁移历史流水表，国产化如何适配。

场景说明：使用 mysqldump 导出数据，使用 load_data 命令导入。

具体分区交换的操作如下：

- 1、检查分区是否存在。
- 2、检查分区是否有数据。
- 3、导出源表（A/B 表）的统计信息至临时表 MID_STATS。
- 4、解锁源表(A/B 表)的统计信息。
- 5、分区交换至 C 表。
- 6、对 C 表分区进行压缩。
- 7、检查交换后 A/B 表的索引状态是否为 UNUSABLE，如果是，马上重建索引 8.检查 C 表中全局索引状态是否为 UNUSABLE，如果是，马上重建索引 9.检查 C 表中分区索引状态是否为 UNUSABLE，如果是，马上重建索引 10.将 MID_STATS 中的统计信息导回 A/B 表 11.锁定 A/B 表统计信息 12.对 C 表进行收集统计信息（6）执行存储过程成功/失败后，更新 bth_extp_status 表状态为 1（成功）或 4（成功）

信息锁定、统计信息导入导出方式保证当日 AB 表清理后统计信息正常，国产化如何适配

场景描述：信息锁定、统计信息导入导出方式保证当日 AB 表清理后统计信息正常，国产化后如何保证。

场景说明：运行 analyze table 命令更新统计信息。

使用 drop、truncate 分区方式清理分区并能保持索引不失效，国产化如何适配

场景描述：使用 drop、truncate 分区方式清理分区并能保持索引不失效，国产化如何适配。

场景说明：支持 drop、truncate；不影响其它分区的索引。

使用在线重建索引方式保证索引可用

场景描述：使用在线重建索引方式保证索引可用，国产化如何适配。

场景说明：一般不重建索引；按需重新组织表。

网联高并发场景，要求 INSERT/UPDATE 响应耗时在 10ms 内，国产化如何保证

场景描述：网联高并发场景，要求 INSERT/UPDATE 响应耗时在 10ms 内，国产化如何保证。

场景说明：保证单事务大小。

大流量、高并发批量跑批场景，国产化如何适配和设计

场景描述：大流量（峰值 IO 600MB/S）高并发（900 并发）批量跑批场景，国产化如何适配和设计。

场景说明：可以支持；单事务 binlog 小于 1G。

高可用怎么做

场景描述：高可用怎么做？比如提供了两个 proxy，对于业务侧，要怎么连。

场景说明：数据库本身提供高可用，例如数据库节点部署的时候会有一主 N 从这样。set 里的节点挂了，会自动做主从切换、故障恢复这些。需要人工干预的场景，文档也有提到。

1、通过前置的负载均衡组件。

2、通过 tdsq1 提供的驱动。

使用原生 mysqldump 对表进行全备，但恢复失败

场景描述：使用原生 mysqldump 对表进行全备，备份成功，恢复失败，是否支持原生 mysqldump 工具，还是需要 TDSQL mysqldump 工具，能否提供备份与恢复命令？

场景说明：如果是 TDSQL 的备份和恢复，可以到赤兔上做操作；原生 mysqldump 出来的，类似上面的 collate/generate 和 lock 问题，有些语句会不兼容，需手工做修改后才能导入。这个要具体结合语句情况看；Generate 只针对建表语句，对于生产环境（批量处理几十 G 大文件）可分开导表结构和数据的方式；对于 lock 和 unlock，加上--compact --single-transaction 可以解决。

同城双活配置

场景描述：同城双活生产环境最小资源配置与硬件、网络设备具体细节与参数。

场景说明：详见“部署调测 > 部署规划 > 资源规划”。

分布式实例 恢复到集中 set，恢复回来，多分片恢复多分片场景

场景描述：分布式实例 恢复到集中 set，恢复回来，多分片恢复多分片场景。

场景说明：详见官网“备份与恢复”章节内容。

三节点和两节点的区别，是否可以处理成 2 节点以及原因

场景描述：三节点和两节点的区别，是否可以处理成 2 节点以及原因。（调整并补充描述）技术上可行？有没有此类生产做法？

场景说明：高可用角度，避免 master 单点故障。生产环境存在 1 主 1 备实例，有一定风险在备宕机时导致只读账号无法连接。生产环境必须 1 主 2 备。

两地三中心架构如何设计

场景描述：市区 A、B 数据中心直线距离 20 公里左右，双中心之间平均时延 10ms 以上，异地中心在 C 与市区直线距离 200 公里以上。两地三中心架构如何设计，明确网络要求，特别是带宽和时延，需要有数据说明。

场景说明：网络配置：同城 2*20Gb，延迟 1~2ms；异地 2*7.5Gb，延迟 30ms 以内。TDSQL 网络要求：同城网络 10Gb，延迟 2ms 以内。异地网络 5Gb，延迟 30ms 以内。

核心系统开发、测试、准生产、生产各环境的资源预估

场景描述：核心系统开发、测试、准生产、生产各环境的资源预估。

场景说明：详见“部署调测 > 部署规划 > 资源规划”。

培训指导文档

场景描述：想要得到一套规整好的标准化文档及系统的培训指导。

场景说明：参考[私有云文档中心](#)。

容量估算规则

场景描述：容量估算规则，需要多少个集群。

场景说明：详见“部署调测 > 部署规划 > 资源规划”。

等保问题设置

场景描述：等保问题设置

场景说明：如下图

安全问题	关联资产	整改建议	风险值	联系人	整改措施
未开启审计功能，无法对审计进程进行保护。	TDSQL数据库	建议开启审计日志功能，并对审计进程进行保护，避免受到未预期的中断。	中		数据库可以开启一般审计日志。低负载开启影响不大，高负载会有影响，同时需要考虑日志会增加磁盘空间占用。
未开启安全审计功能，无相关审计记录及时间。	TDSQL数据库	建议开启审计日志功能，审计记录产生时的时间应由系统范围内唯一确定的时钟产生。	中		数据库可以开启一般审计日志。低负载开启影响不大，高负载会有影响，同时需要考虑日志会增加磁盘空间占用。
未提供漏洞扫描报告和修复记录。	TDSQL数据库、TDSQL-赤兔管理系统	建议定期进行漏洞扫描，并在测试通过的前提下，及时修复风险漏洞。	中		客户定期扫描，我们配合修复。
不可保证重要数据在传输过程中的完整性。	TDSQL数据库	建议采用校验码技术或密码技术保证重要数据在传输过程中的完整性。	高		(1) 数据库已支持ssl访问。 (2) 实例创建用户时，已支持限定ip。 使用ssl的影响： (1) 对性能有影响 (2) 运维方面，在排查问题时，就无法通过抓包排查了 (目前没有其它客户在内网使用ssl)
未采用校验技术或密码技术保证重要数据在传输过程中的完整性。	TDSQL-赤兔管理系统-1、TDSQL-赤兔管理系统-2	建议采用校验码技术或密码技术保证重要数据在传输过程中的完整性。	高		在nginx中配置ssl功能可实现，然后通过https访问。
未采用校验技术或密码技术保证重要数据在传输过程中的完整性。	TDSQL-赤兔管理系统	建议采用校验码技术或密码技术保证重要数据在传输过程中的完整性。	中		在nginx中配置ssl功能可实现，然后通过https访问。
未采用校验技术或密码技术保证重要数据在存储过程中的完整性。	TDSQL-赤兔管理系统	建议在重要数据存储时，采用经国家密码主管部门认可的校验技术或密码技术，保证其在存储过程中数据的完整性。	中		在nginx中配置ssl功能可实现，然后通过https访问。
不可保证重要数据在传输过程中的保密性。	TDSQL数据库	建议对系统管理数据、鉴别信息及重要业务数据采用经国家密码主管部门认可的密码技术，保证其在传输过程中数据的保密性。	高		(1) 数据库已支持ssl访问。 (2) 实例创建用户时，已支持限定ip。 使用ssl的影响： (1) 对性能有影响 (2) 运维方面，在排查问题时，就无法通过抓包排查了 (目前没有其它客户在内网使用ssl)
未采用密码技术保证重要数据在传输过程中的保密性。	TDSQL-赤兔管理系统	建议对系统管理数据、鉴别信息及重要业务数据采用经国家密码主管部门认可的密码技术，保证其在传输过程中数据的保密性。	高		在nginx中配置ssl功能可实现，然后通过https访问。
未采用密码技术保证重要数据在存储过程中的保密性。	TDSQL-赤兔管理系统	建议对系统管理数据、鉴别信息及重要业务数据采用经国家密码主管部门认可的密码技术，保证其在存储过程中数据的保密性。	中		在nginx中配置ssl功能可实现，然后通过https访问。

大数据平台--TDSQL 对接方案

场景描述：现有 TDSQL 数据库的数据需要同步到某大数据平台，大数据平台自建有 Kafka 系统。即需求：转换 binlog 数据，并灌入客户大数据平台自带的 Kafka 系统。

场景说明：TDSQL 多源同步对接大数据平台的 kafka，有限制--不能带安全认证，且只能对一个 partition。如果该项要求不满足，可以考虑使用生态同步工具来转化数据。

数据同步-binlog 日志格式

场景描述：自研数据同步工具，通过解析开源 MySQL 日志做数据库之间的数据同步。需要确认 TDSQL 集中式的 binlog 日志格式与开源 MySQL 有没有区别？

场景说明：详见“参考 > 日志参考 > Binlog 日志”。

TDSQL 与 F5 对接方案

场景描述：TDSQL 与 F5 对接方案，提供参考配置。

场景说明：详见“对接调测 > 对接商用负载均衡”。

新老系统并行方案，回退方案案例分享，技术细节

场景描述：新老系统并行方案，回退方案案例分享，技术细节。如何实现新老系统 Oracle 与 TDSQL 之间的数据同步及数据一致性保证？

场景说明：数据库并轨切换方文档。

TDSQL 数据库是否支持报表类业务？此类业务系统的数据库设计 TDSQL 有哪些设计建议？

场景描述：TDSQL 数据库是否支持报表类业务？此类业务系统的数据库设计 TDSQL 有哪些设计建议？

场景说明：支持，详见“应用开发”。

根据案例提供技术细节

场景描述：根据案例提供技术细节，包括数据库分片规则，性能指标，事务处理等细节？

场景说明：详见“了解 TDSQL MySQL > 产品功能 > 高度兼容 MySQL 语法 > 物理分片与逻辑分表”。

X86 虚拟机规划了一套两地双中心的 TDSQL 配置

场景描述：由于物理服务器没有就位，规划了一套两地双中心的 TDSQL 配置，采用 X86 虚拟机做临时生产规划。

场景说明：详见“部署调测 > 部署规划 > 资源规划”。

如何采用 sysbench 对 TDSQL 环境做压力测试

场景描述：采用 sysbench 对 TDSQL 环境做压力测试，评估系统稳定性、架构合理性。

场景说明：详见“部署调测 > 验收测试”。

odbc 安装包

场景描述：目前项目组是通过 odbc 安装包链接 tdsq1，tdsq1 是否有对应的安装包，以及 tdsq1 是否有做定制的 so 文件？

场景说明：TDSQL 数据库 [ODBC 使用手册](#) TDSQL [ODBC 驱动包](#)。

如何实现单事务内的读写分离？

场景描述：如何实现单事务内的读写分离？

场景说明：建议应用程序来控制，防止数据不一致。

3 数据迁移类

TDSQL 数据同步工具是否提供 DDL 同步

场景描述：TDSQL 数据同步工具是否提供 DDL 同步？是否提供数据同步到目标库后，自动删除源表数据？（例如：联机库数据同步到历史库后，删除联机库对应数据）

场景说明：DBBridge 支持 DDL 同步，不支持自动删除数据。

DBBridge 性能问题

场景描述：某项目 xxx 业务单中心 2000tps 会遇到数据延迟的问题。

场景说明：某项目 xxx 业务单中心 2000tps 会遇到数据延迟的问题，持续 9 个小时，共 1.5 亿左右的交易量，dbbridge 多个通道出现较大的数据同步延迟(延时 20000s)，抽取量和装载量均为 5000。

解决方法：

- 1、大表拆分通道（一个实例 2 个大表各拆分 1 个通道，其它表共用一个通道）
- 2、配置库相关配置读取增加缓存机制，配置库位点更新写入使用异步方式。
- 3、单表的变更写入时，按照主键并发写入。
- 4、解析 binlog 的逻辑进行优化，减少逻辑判断，精简代码。

含有 generated 生成列的表，导入数据会报错

场景描述：数据迁移到含有 generated 生成列的表，导入数据时遇到含有生成列的表，会报错。dbbridge 的数据全量迁移会报错。

解决方法：

- 1、先迁移到不带这个属性的表里。
- 2、创建一张新表。
- 3、通过 insert select 方式导入。

全量迁移阶段，INSERT 报错

场景描述：Oracle 迁移到 TDSQL, 在全量迁移阶段，INSERT 语句有语法报错问题。

解决方法：调整 proxy 配置 cache_prepared_stmt_count="0"。

INSERT 报错一级 hash，二级 list 分区表创建失败

场景描述：一级 hash 分片，二级 list 分区表创建失败。

场景说明：dbbridge 对一级 hash 分片+二级 list 分区的转换语法存在问题。

解决方法：少量语句报错，先手工修改建表语句。

DBBridge 内存使用限制

场景描述：DBBridge 内存使用限制。

场景说明：目前发现 DDBridge 内存使用较大，需要在不影响性能的情况下进行限制。

解决方法：增加内存限制，更新安装包发布优化。

使用 load data 有问题

场景描述：使用 load data 有问题，报错如下图：

```
e: ,enclosed: ,escaped_by: \, thread_num: 128, chunk_size: 0
it's a file
mysql_embedded: 1146
*** Error in './load_data': malloc(): memory corruption: 0x00000000426ef20 ***
===== Backtrace: =====
/lib64/libc.so.6(+0x82c86)[0x7f3371632c86]
/lib64/libc.so.6(__libc_malloc+0x4c)[0x7f337163584c]
/lib64/libstdc++.so.6(_Znw+0x1d)[0x7f3371ef41bd]
/lib64/libstdc++.so.6(_ZNSt4_Rep9_S_createEmmRKSaIcE+0x59)[0x7f3371f52da9]
/lib64/libstdc++.so.6(_ZNSt9_M_mutateEmmm+0x66)[0x7f3371f52f66]
/lib64/libstdc++.so.6(_ZNSt12_M_leak_hardEv+0x26)[0x7f3371f530f6]
./load_data(_ZN9CNewMySQL11realConnectEv+0x26)[0xb5f0b6]
./load_data(_ZN9CNewMySQL7connectEv+0xdf)[0xb5f36f]
./load_data(_ZN9CNewMySQL4initERKSsjS1_S1_S1_S1_+0x3b)[0xb5f53b]
./load_data(_ZN5tdsql9load_data14TableAttribute4InitEv+0x1bc)[0x73021c]
./load_data(_ZN5tdsql9load_data11ImportTable4InitEiPPKc+0xc2a)[0x7374da]
./load_data(main+0x36)[0x70f276]
/lib64/libc.so.6(__libc_start_main+0xf5)[0x7f33715d2445]
./load_data[0x7250e1]
===== Memory map: =====
00400000-01803000 r-xp 00000000 fd:00 538349 /root/loaddata/load_data
01a03000-01ad9000 r--p 01403000 fd:00 538349 /root/loaddata/load_data
```

场景说明：数据文件格式有问题导致无法导入（数据格式不太规则，数据文件里面分隔符有的长，有的短，导致工具无法识别）。

解决方法：升级新版本 load_data；使用 mysqldump 工具或者 select into outfile 指定间隔符来导出 csv 格式数据，保证分隔符长短一致（一般要指定间隔符，括起符）说明：此数据导出涉及到用户 file 权限，需要去底层机器使用超级权限去赋予权限。

说明：

新版本 load_data 介质获取，请联系交付方案评审人员。

4 应用开发类

属性配置错误

场景描述：含有 limit 1 的 SQL 执行时报错，提示‘FETCH FIRST 1 ROWS ONLY’。

例如：

```
select * from (select * from aunion all select * from a_his) a where a.data = 20220101
```

场景说明：配置文件 schema.xml 中<schema>标签中的 dbType 属性配置为“oracle”时，通过 SQLParseEngine 获取到 druid 的 jar 包中的 OracleStatementParser，该语法解析器自动在含有 limit 1 的 SQL 后追加了 FETCH FIRST 1 ROWS ONLY。

解决方法：修改 dbType 为 mysql。建议将该属性的获取改为自动获取，避免配置错误。

事务提交报错

场景描述：程序运行异常报错 Caused by: java.sql.SQLException: Can't call commit when autocommit=true

场景说明：分布式事务需要由应用来控制提交，TDSQL 数据库默认是自动提交，程序连接数据库后默认自动提交会导致这个报错。

程序在 mysql 数据库开发中是基于 autocommit=off 开发的，这里数据库 TDSQL 默认是 autocommit=on。

解决方法：

方式 1：程序端来控制 autocommit，程序端配置代码中加一个 conn.setAutoCommit(false)。

方式 2：TDSQL 数据库中设置 autocommit=off。

方式 3：Druid 连接池中新增参数 druid.defaultAutoCommit=false。

ResultSet closed

场景描述：程序出现 java.sql.SQLException: Operation not allowed after ResultSet closed。

场景说明：一个 Statement 多个 ResultSet 进行操作引起 ResultSet 已经关闭错误

解决方法：让一个 Statement 对应一个 ResultSet，操作完成后及时释放 ResultSet 和 Statement。

TDSQL 连接超时

场景描述：TDSQL 连接超时，同一个应用不同业务会报超时。

解决方法：

1、建议客户根据以下文章内容在 Druid 连接池中添加 maxEvictableIdleTimeMillis 参数进行测试。

<https://blog.csdn.net/xingor/article/details/106698055>

2、建议在 jdbc 连接串内添加：autoReconnect=true 与 failOverReadOnly=false 参数测试。

The error occurred while handling results 报错

场景描述：程序出现 The error occurred while handling results 报错。

解决方法：

出现该类型错误，常见于数据库表字段类型与应用程序中接收实体类型不一致导致，如用 orm 框架采用 mybatis，则排查 xml 文件中 resultMap 中 jdbcType 类型设置及返回接收实体类中对应属性类型是否一致。

创建删除 SEQUENCE 报 lock wait timeout

场景描述：创建删除 SEQUENCE 报 lock wait timeout。

场景说明：客户使用客户端时未设置事务为自动提交，查询了某个 SEQUENCE，导致事务未提交，进而锁冲突。

解决方法：等未提交的事务自动超时，锁释放。如果没有释放，就把所有连到数据库的连接都 kill 一遍。

主键与唯一索引字段相同，导致性能差

场景描述：主键与唯一索引字段相同时，导致死锁、锁等待、主备同步慢。

解决方法：批量删除与主键字段名相同的唯一索引，字段名相同顺序不一样的，改为普通索引。

二级分区

场景描述：业务系统存在流水表，期望通过一级 hash 二级 range 来实现快速清理。

场景说明：分区表的唯一索引必须包含分区键。

解决方法：尝试推荐在业务层按时间进行分表，在数据库建立多张一级分表，如无法直接二级分区，直接分表设计。

JDBC 的 client 工具

场景描述：某版本 jdbc 驱动不支持 localdatetime。

场景说明：当前 jdbc 版本 1.2.1，在批量插入场景下不支持 localdatetime。原因：原生 jdbc8.0.29 才开始支持，而 client 驱动 1.2 版本是基于 8.0.24 开发，所以需要通过升级 client 的驱动到 1.4 版本才能解决。

解决方法：升级到支持的版本，但需要进行详细测试验证。

ODBC 兼容性

场景描述：TDSQL-MySQL ODBC 驱动库依赖最低版本 gcc-5.2.0 对应的 libstdc++库，因此一些低版本的环境中(比如 gcc-4.8.5)使用不了，而用户的生产环境不允许升级，所以需要适配更低版本的库。

解决方法：建议使用原生的 ODBC 驱动库。待 TDSQL 发布兼容版本的 ODBC 驱动。

多条语句执行报错

场景描述：用 dbeaver，或者 c++程序，同样的语句，mysql 可以，tdsql 报错。测试表结构如下：

```
-- 测试表结构
CREATE TABLE `test2` ( `a` int NOT NULL,
`b` int NOT NULL,
`c` char(20) CHARACTER SET gbk COLLATE gbk_bin NULL DEFAULT NULL,
PRIMARY KEY (`a`, `b`) USING BTREE, UNIQUE INDEX `u_1`(`a`, `c`) USING BTREE
);

-- 执行
sql begin;
insert into TEST2(a,b,c) values(19,2,'r5');
update TEST2 SET b=1 where a=1;
commit;
```

解决方法：网关上有一个“multi_query”参数，从 proxy-2.0.21-0-R760D002 版本（10.3.19）开始默认开启。查看了一下当前版本设置是 0，在赤兔上在线修改为 1 保存即可生效（不需要重启，重启后也不会失效）。开启该参数后，多条语句执行正常。

tdsql 透传使用

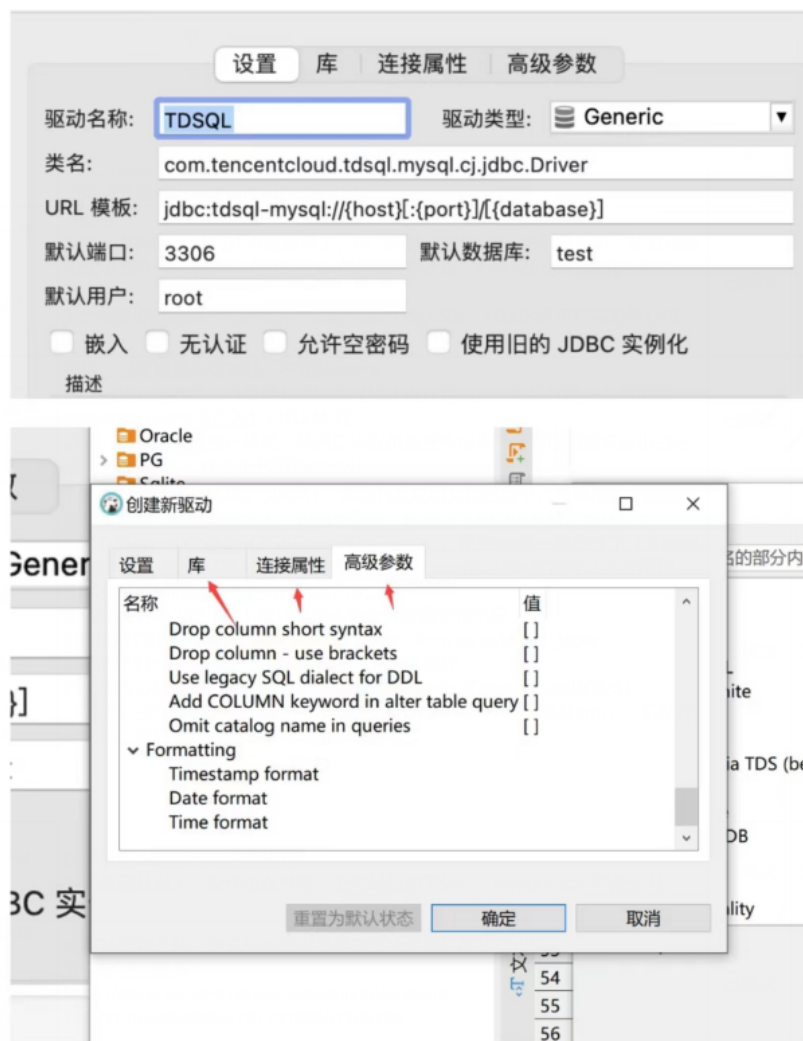
场景描述：mysql.exe 可以使用 -c，来使用注解语法 /*proxy*/ show status，

1.那 dbeaver 这个工具要怎么设置呢？

2.以及 C++代码（顶点侧原使用原生 mysql c api,即 mysqlpp）是否支持；若可以支持，如何操作呢？

解决方法：

1.要用 tdsq1-mysql 驱动（JAVA），驱动包及说明文档已提供，具体配置参数见下方指引图片（将驱动导入库，识别类路径，连接属性和高级参数可默认）。



2.可以支持，需要驱动（TDSQL 的 ODBC 驱动）；原生 api 在一定条件下（非 tdsqldb 特殊语法）可支持；DBA 补充：这种透传的命令，一般是 dba 在管理端才会用到，正常业务应用时是不需要执行这些命令；建具体操作办法：使用 tdsqldb 提供的驱动包即可（odbc 和 mysql c api 均支持）。

执行此语句报错：/*proxy*/show status

场景描述：DBA 验证 SQL 语句可成功执行，client 上执行此语句报错：/*proxy*/show status。

解决方法：操作错误导致，语句最后加了分号，mysql c api 不用分号。

业务改造过程中，需求专人协同优化

场景描述：

- 1、出现的死锁、锁等待问题不好排查。
- 2、对应用侵入性太强，应用改动太大。
- 3、TDSQL 的批量删除易导致锁等待，需要拆分小些。

解决方法：属于分布式数据库带来的特性，本质上还是需要在应用中，对原集中式 SQL 语句做相应的改造。改造过程中，会安排专人协同优化。但在此之前，需要开发技术人员更熟悉 TDSQL，熟悉后排查、修改、优化会更加简单独立，详见“应用开发”。

ODBC 应用开发问题

场景描述：

- 1、ODBC 有一个 SQLPrepare 执行慢的问题，默认在数据库上执行，通过参数修改为在应用端执行，执行的时间没有减少只是分摊到了多台应用服务器上，有没有办法继续优化。
- 2、查询无结果返回固定的错误码：如 ORACLE 返回 100。
- 3、UPDATE 成功后 SQLRowCount 获取影响行数返回的是 0 而不是 1。

解决方法：

- 1、程序基于 C++ 开发的，需要使用 odbc 驱动。
- 2、原有程序会捕获 no_data_found exception->ORA-01403，程序已自定义为错误 100，后续根据业务进行逻辑处理。
- 3、需要使用 mysql 的变量，程序可以按照 mysql 返回 row count 值，获取返回值变量进行处理。

参考文档：《tdsql_mysqlodbc.tar.gz》、《TDSQL 数据库 ODBC 使用说明.pdf》

jdbc 连接参数 useLocalSessionStates 需要配置吗？

场景描述：jdbc 连接参数 useLocalSessionStates 需要配置吗？配置 true 还是 false，有什么影响？

解决方法：详见“应用开发>应用开发（分布式）>连接数据库>使用应用中间件”。

使用 tdsqli 驱动，驱动包应该走什么流程获取

场景描述：TDSQL 程序驱动包应该使用 mysql 驱动还是 tdsqli 驱动，如果使用 tdsqli 驱动，驱动包应该走什么流程获取？

解决方法：参考官网“开发指南”。

如何使用 ODBC 替代 Oracle Pro*C 连接数据库

场景描述：如何使用 ODBC 替代 Oracle Pro*C 连接数据库？

解决方法：使用 MySQL 官方的 ODBC，流式查询分批取数据代替游标。

流水表设计如何适配国产化改造

场景描述：如何使用 ODBC 替代 Oracle Pro*C 连接数据库？

解决方法：结合具体业务，并参考“应用开发”。

使用分区压缩方式保留 500 亿条网联历史流水压缩前估计 180T，国产化如何承接？

场景描述：使用分区压缩方式保留 500 亿条网联历史流水压缩前估计 180T，国产化如何承接？

解决方法：建议使用 tdsql 自带的压缩功能。

国产化如何保证高并发

场景描述：跑批高并发场景，批量库需支持接近 900 个并发连接，25 分钟要并发处理约 5000 万流水，国产化如何保证？

解决方法：确保单事务大小<30M。

sql 客户端

场景描述：有推荐（或者建议）用那个 sql 客户端吗？

解决方法：基本上兼容 mysql 协议的应该都可以的，看个人喜好，选择自己最熟悉的客户端工具即可，例如 navicat、dbeaver、mysql CLI 等。

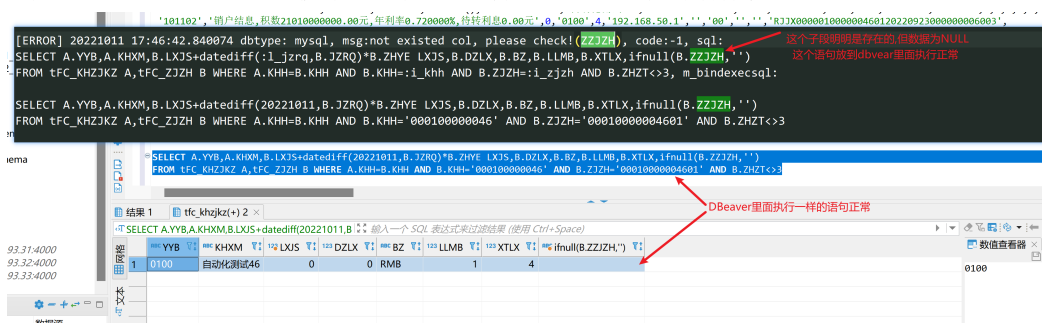
tdsql-mysql-connector-java8-1.4.0.jar 支持 kettle 吗？

场景描述：tdsql-mysql-connector-java8-1.4.0.jar 不支持 kettle 吗？直接用 mysql 的 mysql-connector-java-8.0.15.jar 是可以的。

解决方法：tdsql 对应的类名是：com.tencentcloud.tdsq.mysql.cj.jdbc.Driver（kettle 里的数据源连接类型暂不支持 tdsq）；可以通过[下方办法](#)满足。导入自定义 Driver，把 tdsq 驱动导入后再使用。

报错信息：字段不存在

场景描述：在 C 程序里面调用，报错信息：字段不存在（但此字段实际存在），相同语句放到 DBeaver 里可成功执行；请问是什么原因？如何解决？



解决方法：应用侧 C 程序存在问题，sql 没有那一列，所以无法取出。应用侧修改。

执行过程中权限变成只读

场景描述：执行过程中权限变成只读（连 proxy 代理），具体报错见下图，如何解决？

```
PARTITION pDC2005_06 VALUES LESS THAN (20050701),  
PARTITION pDC2005_07 VALUES LESS THAN (20050801),  
1296 - The MySQL server is running with the --read-only option so it cannot execute this statement  
时间：46.917s
```

解决方法：/*sets:allsets*/ drop if not exists 透传删除，然后再重建。

drop partition 操作

场景描述：创建一级 hash 二级 list 分区后，drop partition 报错。

解决方法：1.19.23 测试是正常的。建议客户升级 proxy。

update returning

场景描述：update ... returning 语法是否支持。

解决方法：1.19.23 测试是正常的。建议客户升级 proxy。

Proxy ERROR:Join internal error: target table must have primary key

场景描述：执行 delete 请求报错：Proxy ERROR:Join internal error: target table must have primary key

解决方法：别名导致的问题。delete 请求中使用别名和大写的表名可能会遇到类似的报错，建议业务在 delete 请求中先不要使用别名和大写的表名。

Proxy ERROR:Join internal error: target table must have primary key

场景描述：update tFC_ZJZH SET ZHYE=999999999999

报错信息：Proxy ERROR:Join internal error: update query has no where clause

解决方法：update 没带 where 条件；对策 1，加上 where 1=1，对策 2 修改参数（把 safe_update 改成 off）。

proxy 返回的时候顺序不对

场景描述：proxy 返回的时候顺序不对（列顺序与建表不一样），集中式实例出现此问题，什么原因，如何解决？

解决方法：对代码层做了兼容，可以先规避。

5 运维类

建表总数超过 5000 报错

场景描述：建表报错：686 - Proxy ERROR: Too much tables,exceed the maximum value。

场景说明：产品特性。

解决方法：

赤兔修改配置。登陆赤兔平台，找到 VIP 端口对应的实例，在【实例详情】【基本信息】【网关权重】，找到 VIP 对应的 proxy 地址，修改实例所有 proxy 机器的配置文件，\$proxyport 对应 VIP 端口；vim
/data/tdsql_run/\$proxyport/gateway/conf/instance_\$proxyport.cnf 找到 table max="5000" 这一行修改为 10000，动态参数 5s 内生效。

单表行数大于 1000 万行报错

场景描述：单表记录大于 1000 万报错：3932 - binlog writethreshold(1610612736) exceeded

场景说明：产品特性。

解决方法：赤兔修改配置。

登陆赤兔平台，找到 VIP 端口对应的实例，在【数据库管理】【数据库参数】。

【binlog_write_threshold】默认值，放大 10 倍。

TDSQL 赤兔系统初始化问题

场景描述：在 TDSQL 安装阶段，出现了跨中心强同步配置下前后端不一致的问题，导致不能初始化。

场景说明：诊断解决，原因是跨中心部分 UDP 网络策略没有放行成功，导致 ack 不能收到。

hdfs 限制 ls 查询权限

场景描述：hdfs 限制 ls 查询权限。

场景说明：建议配置登录连接超时策略，严格限制登录超时时间。

解决方法：如果只是限制 other 用户不能访问目录，只需要开启【普通权限检查】就可以限制了。

```
hdfs-site.xml 设置启动 acl
## 开启普通权限检查
<property>
<name>dfs.permissions</name>
```

```
<value>true</value>
</property>
## 重启 hdfs
## other 用户没有对目录的【执行 x】权限，其子目录权限也无法访问。
hdfs dfs -chmod 750 /<xxxx_root>
```

实例扩容内存长时间停留

场景描述：TDSQL 实例扩容内存长时间停留。

解决方法：临时解决方案重做备机。

天巡大盘中心 TDSQL 天巡备份延迟的实例赤兔界面不存在

场景描述：天巡大盘中心 TDSQL 天巡备份延迟的实例赤兔界面不存在。

解决方法：

更新 tdsq1.yml, #重启采集容器 cd /data/tianxun-v2.2.*/components/tdsql/docker-compose down && docker-compose up -d。

server_id 相同

场景描述：tdsql17.5.0 版本，主从切换后发现 server_id 相同。

场景说明：同一个 set 中服务器的 ip 第三位第四位是相同的，导致 subip 变量计算是一样的，后续产生的 server_id 经过 RANDOM%256 之后仍有 1/256 的几率发生重复。

```
更新 tdsq1.yml,
#重启采集容器 cd /data/tianxun-v2.2.*/components/tdsql/
docker-compose down && docker-compose up -d
```

解决方法：对同一个 set 生成 server_id 的时候进行唯一性检查。

修改 install_mysql_innodb.sh 中的 server-id 生成逻辑作为补丁：

需要对一下几个地方进行修改

1. ansible (新机器)
2. home (实例)
3. mysql (重建实例)

oss 批量创建实例，部分实例中缺少 proxy 信息

场景描述：生产中 tdsq1 版本为 17.5，通过 oss 批量创建实例，发现 set 都创建成功，但部分实例中缺少 proxy 信息。

解决方法：

1. 串行执行调用 OSS 接口创建实例。
2. 调用 OSS 接口创建实例时，除指定 proxy 端口外，程序还需保证 proxy 端口的唯一性。

建议：创建完成实例后，建议调用查询接口确认是否完成，或者在页面中手工查询已创建完成的实例信息。

切换 dcn 目标端的 mysqld 会重启且出现短暂告警

场景描述：切换 dcn 目标端的 mysqld 会重启且出现短暂告警。

场景说明：判断逻辑为如果做 dcn 切换，且数据库为空（或者只有 test 数据库），走 xtrabackup 的逻辑，数据库会发生重启。

参考文档：《groupshard DCN 主备切换流程.docx》

通过脚本来实现多个通过 DCN 同步的实例主备集群切换

场景描述：通过脚本实现多个通过 DCN 同步的实例主备集群切换。

解决方法：赤兔目前已支持批量 DCN 切换功能，建议客户优先使用赤兔上的该功能。

告警分析—告警策略存在问题

场景描述：运维人员进行告警策略调试后，若想把备份策略一键恢复到集群默认的样子，点击“恢复出厂策略”，策略变得和初始化的完全不一致了，如何解决？或者如何把初始化的告警策略备份，在后续调整过大的情况下通过导入模板等形式复原？

解决方法：

1. 找到一键部署包中初始化告警规则的文件：

/data/tdsql_10.3.19.1.0/tdsql_install/roles/tdsql_chitu/templates/alarm_config.json

2. 如果客户使用了告警恢复这个功能，则可以考虑用这个文件恢复。

3. 如果需要恢复初始化设置，可以把这个表清空 t_alarm_config_items。

然后再在界面上用刚才那个 json 重新导入一下。

偶现无法拉起实例

场景描述：偶发性出现 mysql 实例的 sock.lock 文件在重启服务器时候无法释放，机器起来后无法自动把实例拉起来，要人工介入处理。

场景说明：官方 mysql 检查 sock 逻辑不够优化，服务器重启时 pid 小概率重用导致 mysqld 启动时误判实例已启动而无法重写 sock.lock。

解决方法：

方法一：作为运维应对条例处理。

方法二：开机清理 sock 相关文件。

sysbench 的 TPS 比 QPS 恢复慢的情况

场景描述：主 DB 宕机，进行切换的时候，经常出现 sysbench 的 TPS 比 QPS 恢复慢的情况（比如迟了 20 秒才恢复）

场景说明：正常情况下，agent 在后台取消新 master 只读状态会有 3 秒左右的轮询时间。在故障 db 为网络访问超时的异常下，agent 检查复制状态的逻辑较长时间才超时，该网络等待超时阻塞了同线程取消新主只读的操作。因而出现在已经切换完毕只读 qps 恢复，而后 20 来秒读写 tps 才恢复的现象。

解决方法：升级新版本 agent 组件（该问题已经在新版本中优化）

新版本 agent 将复制状态检查线程和取消只读的线程拆分，避免了取消新主只读被网络超时阻塞较长时间的情况。

主备切换，TPS 和 QPS 最少都要将近一分钟才能恢复

场景描述：主 DB 宕机，进行主备切换，TPS 和 QPS 最少都要将近一分钟才能恢复，按照之前讨论过的原理（20s 探活+切换时长），超出了原先认为的恢复时长。

场景说明：sysbench 没有 sql 执行超时设定，而默认情况下系统参数 net.ipv4.tcp_retries2=7 时 tcp 异常在启发算法 7 次重传约 51s 后 proxy 感知到连接实际已异常，进而通知 sysbench 连接已经断开，此时 sysbench 才重连恢复 tps。

实际上 30s 左右切换已经完毕，新连接可以正常执行 sql。

解决方法：对于这个问题，一方面可以在应用设置 sockettimeout 主动断开连接避免一直等待较长时间才由 proxy 告知连接异常。另一方面若应用不设 sql 执行超时又不想等 51 秒才感知到要重连，可以设定 proxy 服务器系统参数 net.ipv4.tcp_retries2=5，重传 5 次依旧失败后认为连接异常，可使 proxy 在 30s 内感知到连接异常进而通知应用重连。

HDFS 服务目前无自启动服务

场景描述：HDFS 服务目前无自启动服务，需要添加自启动脚本。

解决方法：

1. crontab 实现自启动
2. 为保证顺序执行，需要先启动 JournalNode，然后 NameNode，ZKFailover，DataNode（可以不分先后）

参考：/data/tools/init_hdfs.sh

function init_three_hdfs()

初始后 format 语句需要去掉！

生产环境中赤兔访问速度过慢

场景描述：赤兔访问速度过慢。

场景说明：

已知问题，访问赤兔页面实例管理页面的时候会通过 oss 接口（GetInstance 和 GetGWGroup 去获取最新的实例列表，然后写入监控库），再从监控库读取实例列表展示给用户。

1. 赤兔-【实例管理】调用 oss 接口 GetGWGroup 获取 group 实例时非常慢，导致页面加载慢。

2. 集群刚部署时，访问都正常，但运行一段时间后开始变慢；重启 oss 后短时间内访问正常。当前 oss 版本中，一个线程有一个任务队列，当这个线程的任务队列里面有这种响应很慢的接口，则会影响这个队列的其他接口，重启 oss 则会将所有线程的任务队列都清空。

解决方法：

修改赤兔机器 /data/website/tdsqlcloud/index.php

搜索 TARGETUSER 这个关键字，`define('TARGETUSER', 'pc');` 修改为 `define('TARGETUSER', 'pcl');`

（所有赤兔机器上都要进行修改。）

影响：

修改此参数后，赤兔不会实时调用 oss 接口，而是直接去监控库查询。所获取查询结果比实时查询可能会略有滞后（赤兔通过定时任务，每分钟拉取 oss 接口的最新数据到监控库）。

dcn 批量切换问题

场景描述：tdsql dcn 批量切换问题，20 个一次，提示成功 19 个任务，实际成功 18。_3 某个实例的 dcn 通道被标记已删除。

场景说明：批量 dcn 切换时可能会触发 zk 相关逻辑超时，导致 create dcns slave 节点失败，dcn 关系不完整，赤兔无法显示 dcn 关系。手动补增节点后 dcn 关系恢复，随后可正常切换。

解决方法：

方法一：调整 scheduler 访问 zk 的超时时间为 30 秒：

```
vi /data/application/scheduler/conf/scheduler.xml
<zookeeper timeout="30000" />
-- 修改后重启 manager 和 scheduler 组件即可。
```

方法二：并行 dcn 切换相关组件优化升级。

批量修改实例用户密码

场景描述：批量修改实例用户密码。

解决方法：【OSS 接口参考】

1. 密码重置（TDSQL.ResetPwd）。
- 2.

批量将无法恢复的退化模式恢复成强同步模式

场景描述：批量将无法恢复的退化模式恢复成强同步模式。

场景说明：批量 dcn 切换时可能会触发 zk 相关逻辑超时，导致 create dcns slave 节点失败，dcn 关系不完整，赤兔无法显示 dcn 关系。手动补增节点后 dcn 关系恢复，随后可正常切换。

解决方法：【OSS 接口参考】

1. 设置实例同步/异步模式（TDSQL.SetSyncMode）
2. 查询设置实例同步/异步模式（TDSQL.QuerySetSyncMode）
3. 可以通过 TDSQL.GetInstance 获取实例同步/异步模式值。

shell 命令查看 manager 和 scheduler 的主从状态

场景描述：如何使用 shell 命令查看 manager 和 scheduler 的主从状态。

解决方法：zk 中查询，id 最小的为主。

1. scheduler: /tdsqlzk/election/elec@ip_0000000002
2. manager: /tdsqlzk/manager/manager_election/elec@ip_0000000002

【赤兔】/【调度与管理系统】/【scheduler | manager 监控】/【当前状态】，看到如何从 zk 中查看主从状态。

OSS 输入参数错误

场景描述：同时发起 32 个任务，一个报【11000】OSS 输入参数错误（the master_zk_path para value illegal）

场景说明：由于赤兔 php 默认参数较小，post 请求接受参数不完整导致实际切换实例任务数少于勾选实例数，并且最后一个实例切换报 oss 输入参数错误。

解决方法：调整/usr/local/php/lib/php.ini 如下参数重启 php 即可

max_input_vars = 50000

post_max_size = 150M

主备切换任务无法重入

场景描述：修复主备切换时，在给 agent 下发 reportrelaylog 任务后，同时 scheduler 的重启，可能主备切换任务无法重入问题。

解决方法：这个主备切换过程卡主，导致需要人为干预才能恢复正常使用。

如何强制取消 dcn 关系

场景描述：修复主备切换时，在给 agent 下发 reportrelaylog 任务后，同时 scheduler 的重启，可能主备切换任务无法重入问题。

解决方法：

```
# cleardcnslave 强制取消 dcn 关系
# su - tdsq
cd /data/tdsql_run/4002/mysqlagent/bin
./dcntool cleardcnslave '127.0.0.1:2118' '/tdsqlzk' 'set _XXXX_XX'
```

回退 ansible 步骤

场景描述：不同机房 proxy 入库不同的 es，之前版本是入库同一个 es，

如果想回退 ansible 到之前的版本应该如何操作？

解决方法：修改 filebeat.yml，两个集群分别推包到对应 proxy 服务器，重新部署即可。

administrator command: Quit

场景描述：环境出现这个慢查询：administrator command: Quit

场景说明：tdsql_compute_query_time_for_slow_logging（tdsql 特有参数，启用此参数会改变慢日志的时间计算逻辑——计算锁等待时间和建立连接的时间，目前版本有 bug）。建议关闭此参数，则采用 mysql 的原生计算逻辑。

解决方法：tdsql_compute_query_time_for_slow_logging 关闭该参数。

修改监控数据的保留时长

场景描述：客户想修改监控数据的保留时长，应该调整哪些参数。

解决方法：修改参数

```
/data/application/tdsql_collector/conf/conf.properties  
sink.tdsql.keepday=60（默认 60 天）
```

TDSQL 的集群安装好之后，因不能再更改 IP，如何快速扩展集群

场景描述：在研发模式下（非云化模式），TDSQL 的集群安装好之后，不能再更改 IP，不方便快速扩展环境。

解决方法：基于单机安装，采用 NAT 或者 host-only 固定 IP 的方式复制镜像。

还原数据库实例时，当前实例为什么不能被覆盖

场景描述：赤兔上还原数据库时，要建一个新实例（需更多资源），不能覆盖某个实例。

场景说明：安全考虑，不能对当前实例进行覆盖。

大批量数据做归档时异常

场景描述：某业务需要大批量数据处理做归档时报

解决方法：批量更新 SQL 不能一次大批提交，要改为多次小批提交。

数据库备份还原慢

场景描述：数据库备份还原慢。

解决方法：官方提供的 mysqldump、mysql 改为第三方多线程备份还原工具 mydump、myloader。

数据库备份进行时，某查询业务报锁等待超时

场景描述：在数据库备份过程中，某查询业务报锁等待超时。

解决方法：mysqldump 加--single-transaction；mydump 加 --nolocks。

SQL 字符超过 1412 字节时报错

场景描述：ODBC 上执行的 SQL 字符超过 1412 字节时报错。

解决方法：TDSQL 部署时 lvs、应用、db 的 mtu 不一致，要改成一样的。

information_schema.COLUMNS 查询表结构时，有时没有返回

场景描述：使用 information_schema.COLUMNS 查询表结构时，有时没有返回。

解决方法：调整 TDSQL 参数

```
vim /data/tdsql_run/15002/gateway/conf/instance_15002.cnf
<information_schema randomness="0"/>
-- randomness="0" 改成 0
```

修改分片进程数量

场景描述：TDSQL 分片默认进程数量为 1，导致性能上不去。

解决方法：赤兔平台【实例管理】【set 管理】可以看到分配的进程数量，检查并修改这个值。

赤兔的慢查询界面优化

场景描述：赤兔的慢查询界面优化（增加查询条件，过虑重复数据）。

解决方法：直接从监控库查询，过滤重复数据 tdsqplcloud_monitor.classesanalysis。

一键部署

场景描述：TDSQL 安装的组件较多（某券商初步测算后台运行的组件、控件有十几个），希望 TDSQL 可按照证券客户需求实现轻量化部署，比如对组件进行封装，不要暴露过多组件；

解决方法：基于单机安装，采用 NAT 或者 host-only 固定 IP 的方式复制镜像。

TDSQL 被操作系统 KILL 掉

场景描述：TDSQL 被操作系统 KILL 掉。

场景说明：某测试环境虚拟机只有 32G 内存，TDSQL 默认使用 80%内存，导致操作系统内存不够用，TDSQL 内存下调为 20G。

解决方法：操作系统过度分配策略改为按实际情况分配。

无法创建和删除某个 sequence

场景描述：创建或删除 SEQUENCE 时报 lock wait timeout，客户使用客户端未设置事务自动提交，查询某个 SEQUENCE 事务后未提交，导致锁等待超时。

解决方法：等未提交的事务自动超时，锁进行释放。如果没有释放，则把所有连到 score 库的连接都 kill 一遍。客户端设置事务自动提交后再执行创建或删除操作。

免切删除脚本不能用，红帽官方 yum 源不提供 jq 命令

场景描述：红帽官方 yum 源不提供 jq 命令，导致使用 jq 命令解析 json 的脚本无法使用问题。

场景说明：脚本调用 OSS 接口后，返回的结果都是 json 格式，其数据结构形式不同，存在嵌套情况。采用 jq 命令进行解析，便于开发，可读性和维护性好，不受格式变化影响。

解决方法：

方法一：直接采用 jq 二进制版，放入脚本包中，无需编译和安装且没有依赖，简单配置即可运行。同时，目前脚本逻辑无需改动。

操作步骤：

```
chmod +x jq-linux64
ln -s jq-linux64 jq
export PATH=$PATH:./
./xxxxxx.sh
```

方法二：服务器已安装 python/python3，可以使用 python 对 json 数据进行解析，与 jq 相同，便于开发，可读性和维护性好，不受格式变化影响。目前采用 jq 命令的解析逻辑需要替换成 python 语句实现。

修正 tdsq1 的资源限制

场景描述：操作系统配置设置屏蔽了 TDSQL 优化参数配置，需要进行 OS 参数调整。

解决方法：

```
ps -ef | grep -v grep | grep tdsq1 | awk '{print $2}' | xargs -II prlimit --pid I --nofile=400000:400000 --nproc=60000:80000 --core=unlimited:unlimited --stack=8192:unlimited
```

强制取消 dcn 关系

场景描述：强制取消 dcn 关系。

解决方法：

```
# cleardcnslave 强制取消 dcn 关系
# su - tdsq1
cd /data/tdsq1_run/4002/mysqlagent/bin
./dcntool cleardcnslave '127.0.0.1:2118' '/tdsq1zk' 'set_XXX_XX'
```

Communications link failure

场景描述：应用服务在正常运行过程中，偶发异常 Communications link failure, The last packet sent successfully to the server was 0 milliseconds ago. The driver has not received any packets from the server.

场景说明：测试环境中可能遇到该问题，受限于资源，一般测试环境的配置比较低，其中有几台机器的磁盘都占满，proxy 有台 server 正好在列。负载均衡到 proxy 服务器之间存在探活机制，由于该 proxy 进程还在，故通过 telnet proxy 端口来判断存活不会把该 proxy 服务器剔除，进而连接到该 proxy 机器的连接报错。在连接报错期间，应用可以正常连接其他两个 proxy 服务器。

解决方法：首先，要明确数据库服务端的部署架构，有完整的监控预警系统（如无，建议增加），其次，优先排查 proxy 服务端进程情况，含网络、io、磁盘使用等情况。

TDSQL-MySQL_proxy_调整参数紧急修复

场景描述："1.19.10-M-V2.0R750D004"版本发生 coredump 问题。

场景说明：已知 noshard 的版本在使用 19.10 的版本，会出现 coredump 问题。

解决方法：

将变量 print_binary_protocal_sql(gateway.mod.prepare.print_binary_protocol_sql)设置为 0，代表含义：将 noshard 的 execute, colse, reset 协议转成 sql 记录在日志中。

变量取值范围：

1：打开，0：关闭。

密码安全配置

场景描述：密码重复使用。

场景说明：建议配置口令重复使用次数限制策略，防止口令被轻易破解（建议口令重复使用次数限制不小于 4 次：修改密码 4 次内不允许重复）

解决方法：MySQL 8.0 已支持密码重复使用策略，配置参数 password_history。

password_history 参数定义了一个阈值，来控制之前使用的密码，在用户修改多少次之后，才能重复使用，该参数设置为 0，表示禁用该策略。

TDSQL 要怎么 kill 某个连接呢

场景描述：Lock wait timeout exceeded; try restarting transaction 之前 mysql 可以直接 kill 某个连接，现在连接代理不让 kill, TDSQL 要怎么 kill 某个连接呢。

解决方法：可以在赤兔上面 kill 或者透传到具体的 set 上面执行 kill。

mysqldump 导出来的数据，没有带列名，导致插入分片表失败

场景描述：原来 mysqldump 导出来的数据，没有带列名，导致插入分片表失败，这种怎么做。

解决方法：数据导出时，加上 -c 参数；导出来的数据中，有 lock tables，需要加 --compact 参数进行导入，或者将 lock tables 去掉，否则可能会有报错：XAER_OUTSIDE: Some work is done outside global transaction。

lvs 扩容操作

场景描述：资源不够需要用到 lvs 的服务器，怎么操作。

解决方法：重新安装配置 lvs，详见“部署调测>安装部署>安装数据库（命令行部署）>部署其他（可选）组件>安装 LVS”。

实例被意外下架

场景描述：用户在创建实例的过程中，发现已有实例被意外下架，可能的原因是什么？

解决方法：实例被删之前，有创建实例的任务被卡住的情况，用户手工进行了终止，并重新进行创建实例动作。初步诊断就是这个环境导致，创建实例的时候就选择了已经被占用的端口。

lvs 扩容操作

场景描述：资源不够需要用到 lvs 的服务器，怎么操作。

解决方法：重新安装配置 lvs，详见“部署调测>安装部署>安装数据库（命令行部署）>部署其他（可选）组件>安装 LVS”。

验证 cache 生效失败，是什么原因？如何解决

场景描述：验证 cache 生效失败，是什么原因？如何解决？（当前数据库版本为 10.3.17.3.0）TDSQL 升级至 19.5 版本后，序列号性能还是比较差，cache 仍未生效，如何解决？具体报错：Proxy ERROR: You have an error in your SQL syntax; check the manual that corresponds to your Mysql server version for the right syntax to use near 'GENERATED ALWAYS AS （proxy 版本："1.19.15-M-V2.0R759D002"）

解决方法：当前数据库版本低，不支持 cache。

将数据库升级至 19.5 版本，然后在网关参数里面 sequence cache 设置为 1（默认关闭）；备注：数据库版本升级对服务影响最小的方法（比如从 10.3.17.3.0 升级到最新版）到 proxy 修改配置文件（文件名：/data/tdsql_run/端口号/gateway/conf/instance_端口号.cnf）。

mysql 组件升级

场景描述：mysql 组件如何升级。

解决方法：详见“管理数据库>升级”。

HDFS 服务目前无自启动服务

场景描述：HDFS 服务目前无自启动服务，需要协助编制自启动脚本。

解决方法：

1. crontab 实现自启动
2. 为保证顺序执行，需要先启动 JournalNode，然后 NameNode，ZKFailover，DataNode（可以不分先后）参考：`/data/tools/init_hdfs.sh``function init_three_hdfs()`初始后 `format` 语句需要去掉！

HDFS 服务目前无自启动服务

场景描述：装了 tdsq1 集群，而后创建实例 set3，操作日志显示成功，之后进行了该实例的扩容。直到大概中午的时候使用 oss 批量创建实例后，出现了 set3 的 oss 任务 `initinstance` 失败告警。其原因是 zk 中 `proxy_jobs` 和 `manager_jobs` 两个不同目录可能取用相同的 zk 顺序节点序列号，导致两个不同目录下的不同类型任务的 `taskid` 相同为 `/noshard@000000xxxx`，该任务写到监控库 `t_ossjob` 时表时，`fjob_id` 字段都是相同的 `answer_/noshard@000000xxxx`。而 `monior` 组件查询时没有区别相同的 `fjob_id` 是不是同属于一个 `fjob_type` 字段，导致两个任务的日志用 `fjob_id` 查询时出现合并，`monitor` 组件在这种情况下判断逻辑异常而触发了 `initinstance` 告警。

解决方法：

1. 告警消除方式：该告警在连续多次创建实例的过程中可能会出现，发生告警不影响正常使用。消除告警的方式是在 oss 任务流程界面将异常的 oss 任务设置运营状态为已下架，设置该任务为已下架不影响任何其他功能或任务。
2. 临时规避：鉴于实例创建是否成功有明确的控制台操作日志或 oss 接口可查询到成功，oss 任务告警策略一般而言没有什么意义。建议在告警策略-oss 任务告警策略中关闭是否成功-初始化 `is_success` 告警项即可。

使用脚本进行批量部署 DB 节点，oss 报错

场景描述：mysqlagent 升级包的 `cnf` 模板文件，缺少 `bindport` 参数，需要对 DB 机器公共包和 `ansible` 公共包进行修改操作。

解决方法：agent 升级完成后，需要增加补充步骤，将原公共安装包中的模版配置文件进行还原。

dcn 切换过程卡住

场景描述：dcn 切换过程卡住。

场景说明：可能原因

1. zkhost 在创建 dcn 同步前没有配置，导致备实例在创建与主实例同步时访问异常。
2. 非产品问题，老集群升级了，新搭建的没升，批量 dcn 切换的版本需要两个集群的版本一样的。

解决方法：

1. 取消 dcn 关系重新创建。
2. 升级组件到相同版本，升级版本记得执行那个 load_dcn_to_db。

调整 zk 节点选举优先顺序变更控制

场景描述：原来的 zk 选主的时候优先会选到上海去，这样导致 zk 之间通讯的时间会比较长，现在就是把 zk 的 ID，主机房改大，把上海的改小，这样来保证选主的时候优先选到主机房来解决这个问题。

解决方法：调整 zk 配置文件的 zkid 以及 data 目录 myid，修改为大小分别为 100，80，60，40，20

1. conf/zoo.cfg
2. data/myid

nginx 开启 ssl 的变更

场景描述：nginx 开启 ssl 的变更。

解决方法：参考《[Nginx 安装 SSL 配置 HTTPS 超详细完整全过程](#)》

建议配置口令重复使用次数限制策略，防止口令被轻易破解（建议口令重复使用次数限制不小于 4 次）。

OSS 文档相关设置

场景描述：

1. 脚本需求：实例批量修改参数。
2. 查询已经产生免切的节点并删除该节点的免切场景：当 DB 主机宕机时候，通过脚本判断是主机宕机还是数据库阻塞；若是前者，自动对实例已经生成的免切配置进行删除，避免短时间内出现再次宕机无法切换的情况。
3. 批量修改实例用户密码。
4. 批量将无法恢复的退化模式恢复成强同步模式

解决方法：参考《OSS 文档》

1. 获取实例参数（TDSQL.ListConfig）；修改实例参数（TDSQL.ModifyConfig）。
2. 获得当前集群免切的参数设置（TDSQL.GetInstanceSwitchStatus）；删除当前集群免切的参数设置（TDSQL.DeleteStopInstanceSwitch）。
3. 密码重置（TDSQL.ResetPwd）。

4. 设置实例同步/异步模式（TDSQL.SetSyncMode）；查询设置实例同步/异步模式（TDSQL.QuerySetSyncMode）；可以通过 TDSQL.GetInstance 获取实例同步/异步模式值。

前端服务管理系统 reboot 高可用测试 tps 出现三个低点

场景描述：前端服务管理系统 reboot 高可用测试 tps 出现三个低点。在 DB 主节点重启后，事务监控曲线图会有三个低谷期。主备切换期间，事务处理应该降为 0TPS，切换完成后恢复正常，应该是一个低谷期，而不是三个。

解决方法：操作系统在 reboot 时，结束所有进程后需要等待 10 秒钟，无新进程再重启，否则会杀掉新进程再重启。而 TDSQL 的 agent 探活频度过高，导致不断尝试重启，出现多次进程被杀。需要 TDSQL 研发给出探活的频度说明和调整方案。

提供应急预案，类似故障处理集

场景描述：提供应急预案，类似故障处理集。

解决方法：详见“管理数据库>故障处理”。

数据导入报错

场景描述：数据导入报错。

解决方法：上一次导入数据中断后，表中已经存在部分数据，导致重复导入时报错。清空后重新导入即正常。

性能分析-实时诊断提示有冗余索引

场景描述：赤兔管理后台上，性能分析-实时诊断提示有冗余索引。

解决方法：删除冗余的索引即可。

监控库主 DB 发生内存泄露

场景描述：监控库主 DB 发生内存泄露。

解决方法：监控库缓冲池设置比较小，内存使用率会超过 100% 比较多的。建议先建一个告警策略给监控库，调高其阈值继续观察。

升级 TDSQL（19.5 版本）过程中，创建监控库实例时，出现报错

场景描述：升级 TDSQL（19.5 版本）过程中，创建监控库实例时，出现报错（如下图），如何解决？

说明：tdsqlcloud-监控库密码修改后（创建监控库实例时发现密码不对，修改后报错，监控库在数据库端可连接，web 端连不上）

操作对象:

中止Manager任务

任务进度:

调用接口: AddGWInstance

任务状态: 进行中(2)

执行反馈:

```
{"cur_step":"require_set","dealttime":1662623729,"description":"require to create set","domain":"","err_code":0,"err_msg":"require to create set","extra":"","group_id":"","port":0,"process":10,"proxy_fail":0,"proxy_fail_ips":"","set":"","status":2,"subtask_detail":[{"require_set":{"arch":"","clientName":"","coldback_idc":"","cpu":"32542","cur_step":"start","data_disk":"32542","dbversion":"","dealttime":1662623729,"degrade":0,"degrade_time":0,"description":"start to begin require","err_code":0,"err_msg":"start to begin
```

require to create set

解决方法:

1. 创建的时候要在服务区列表那里手动填一下 ip;
2. 根据 select * from t_cluster_expand;确认 mdb_user 的值;把监控库密码改回去;
3. DBA 建议监控库密码与库里一致(改库的需要改其他地方)。

Proxy ERROR:Join internal error: error:Can't create table 'T3042' (errno: 28)

场景描述: 报错: 913 - Proxy ERROR:Join internal error: error:Can't create table 'T3042' (errno: 28), sql:xxx

解决方法: 偶发性问题, 没有磁盘空间导致。清理磁盘空间。

无法连接某个数据库实例

场景描述: 使用客户端/telnet/赤兔工具均无法连接某个数据库实例。

解决方法: 自动触发了一次主备切换, tdsq1 自动打开了“自动免切”设置, 后续再次执行高可用测试时, 实例不会再进行自动切换, 因此无法连接。解决方案高可用切换等测试时, 需先检查免切设置情况。

主备切换失败

场景描述: 查看赤兔的主备切换日志发现, 实例每隔半分钟左右不断执行主备切换, 且切换结果为失败。

场景说明: 排查发现该实例所在主机 db02 于 23 日持续发生 IO 使用率 100%情况, 持续的 IO 繁忙导致心跳检测失败自动发起主备切换。实际切换时间为 23 日, 又由于 IO 繁忙影响到监控数据存入监控库, 导致数据写入延迟, 因此在赤兔上看到的 24 日主备切换记录实为 23 日切换时的数据。

解决方法:

1. 添加 DB 节点个数, 将 IO 消耗高的实例单独分布到新节点上, 避免影响同 DB 节点上的其他实例。

2. 将 DB 节点虚拟机迁移至性能更好的物理机上。

通过 lvs 的 vip 连接 proxy 的端口连接失败

场景描述：查看赤兔的主备切换日志发现，实例每隔半分钟左右不断执行主备切换，且切换结果为失败。

场景说明：检查发现 mysqlagent 上报 zookeeper 失败，zookeeper 显示 db 不可用。由于 proxy 需依赖 zookeeper 数据进行路由更新，导致 proxy 连接 db 失败。

解决方法：重启 proxy 后恢复正常。

无法正常启动监控库及赤兔

场景描述：监控库所在主机文件系统/data1 使用率 100%，查看 mysqld.err 日志发现监控库数据文件损坏报错，无法正常启动监控库及赤兔。

解决方法：部署日志清理脚本。

查询响应时间长

场景描述：查看赤兔的主备切换日志发现，实例每隔半分钟左右不断执行主备切换，且切换结果为失败。

场景说明：未充分使用 TDSQL-MySQL 分布式特性。

解决方法：扫描子分区过多，避免全表扫描。查询、join 关联条尽量带上分片键。

DCN 页面不显示 DCN 同步任务

场景描述：DCN 页面不显示 DCN 同步任务。

场景说明：OSS 重启导致/etc/hosts 中 zklist 信息丢失。

解决方法：重写/etc/hosts zilist。

6 备份与恢复

备份失败告警

告警指标	告警指标 (中文)	告警描述	处理建议
backup_monitor_binlog	Binlog 备份失败	DB Binlog 备份状态异常出现 5 次/5 分钟以上产生告警，后续 60 分钟内屏蔽告警。	检查 Binlog 备份日志、HDFS 状态等，恢复 Binlog 正常备份。
xtrabackuptmp_not_empty	xtrabackuptmp 目录不为空	xtrabackuptmp 目录不为空指标值等于 1 产生告警，判断频率(1%/1)，后续 10 分钟内屏蔽告警	如当前 DB 未进行 alldump，清空该目录即可。
backup_monitor_xtrabckup	镜像备份状态	DB 镜像备份状态异常出现 10 次/10 分钟以上产生告警，后续 240 分钟内屏蔽告警。	检查备份日志、HDFS 运行状态，恢复正常物理备份。

slowlog/errlog 等日志文件的备份，备份完成后是否会继续持有文件句柄

- slowlog/Binlog 等日志文件一般没有备份完这么一说。这几个日志是持续增长的，备份也是实时备份，所以会一直占有文件句柄。
- errlog 是隔一段时间备份完一次，故不会持续占有句柄。

对于 slowlog/errlog 是否需要先完成备份，才能删除；如果关闭备份，是否也意味着之后不会清理历史的文件

- slowlog 的本地文件删除：
本地删除和备份是通过 2 个线程执行的，没有任何关联。slowlog 备份是持续备份，删除则是直接删除几天前的本地文件。
如果关闭备份，也对本地删除没有影响。
- errlog 的本地文件删除：
errlog 的本地删除和备份是通过一个线程执行的。必须开启 errlog 备份，才会去执行删除。
线程执行内容：备份到 HDFS+按天分割日志+按天清理本地日志。
如果关闭了备份，则就不会按天分割日志，因此也不会清理本地日志。

是否完成一次 xtrabackup 备份后，也会关闭对相关备份文件的访问

会的，这个是在 cold_xtrabackup.sh 脚本里执行的，脚本进程结束就关闭访问了。

7 性能优化类

内层查询无条件

场景描述：带子查询的语句，内层查询无条件导致的性能问题。

例如：

```
select * from (select * from a union all select * from a_his) a where a.data = 20220101
```

场景说明：TDSQL 优化器无法处理内层查询条件，需要在应用层手动在子查询注入查询条件。

解决方法：索引字段 where 条件放到内层查询，尽量让最内层查询出来的结果最少。

使用示例：

```
select * from (select * from a where a.data = 20220101 union all select * from a_his where a.data = 20220101 ) a where a.data = 20220101
```

使用非分片键 join 性能差

场景描述：使用非分片键 join 性能差。

解决方法：结合业务场景，把小表改成广播表，join 可下推到具体的 set 中，有效提升性能。

如果是高频场景，结合业务评估，重新选择分片键。

ODBC 参数设置优化

场景描述：ODBC 参数设置优化。

解决方法：ODBC 参数设置 no_ssps=1，性能可提升 50%-80%。

inner join 优化

场景描述：inner join 优化。

解决方法：

- 1、关联更新使用 inner join 比 left join 快，部分场景可以使用。
- 2、部分场景 left join 比 exist 快。

SQL 转代码优化

场景描述：SQL 转代码优化。

解决方法：

- 1、针对大事务改成多次小批量提交：批量更新 SQL 不能一次大批提交，要改为多次小批提交。例如每提交 200 条（实践过程结合业务测试调优，再得出一个合理的值）
- 2、SQL 关联多张表汇总查询：改为通过应用层来组装业务逻辑，然后统计封装返回。

索引的使用

场景描述：需要正确使用索引，有些场景走不到索引影响性能。

使用建议：

- 1、MYSQL 索引的使用需要遵守最左原则，如果不按联合索引的最左原则使用，则该索引无效。
- 2、索引字段上加函数会导致索引失效。
- 3、索引字段如果做了隐式类型转换或者字符编码转换也会导致索引失效。

事务中有很多 SQL 语句会很慢

场景描述：一个事务中有很多 SQL 语句会很慢，或者 TDSQL 的批量写\修改性能慢。如清算 2000 客户提交一次改成了 200 提交一次。

解决方法：尽量把一些典型外部调用移出事务，事务里批量更新数据需要控制数量，进行必要的 sleep，做到少量多次。详见“应用开发>应用开发（分布式）>开发设计规范”。

8 安装部署类

TDSQL 支持哪些国产操作系统，赤兔管理台支持哪些国产桌面操作系统？

TDSQL 支持国产操作系统如下：

- openEuler 20.3 LTS SP3 算力平台。
- 统信服务器操作系统（a 版）V20。
- 统信服务器操作系统 V20。
- 银河麒麟高级服务器操作系统（飞腾版、鲲鹏版） V10。
- 银河麒麟高级服务器操作系统（兆芯版、海光版、AMD64 版） V10。
- 中标麒麟操作系统。
- 红旗操作系统。
- TencentOS Server 2。
- TencentOS Server 3。

赤兔管理台是基于浏览器访问，因此赤兔管理体系可以兼容全部国产桌面操作系统。

TDSQL 支持哪些国产服务器芯片？

TDSQL 支持国产服务器芯片如下：

- 飞腾 FT-2000_4 和 FT-2000+64 处理器等。
- 海光 3000、5000、7000 系列。
- 海光三号 3000、5000、7000 系列。
- 龙芯 3C5000L 等。
- 兆芯系列。
- 华为 Kunpeng920 等。

TDSQL 是否支持不同芯片服务器混合部署？

TDSQL 支持不同芯片服务器混合部署，但一个实例内主从节点的设备需要是相同架构芯片。

TDSQL 软件在国产环境、非国产环境的功能特性有什么区别？

没有区别，TDSQL 软件功能特性在国产环境和非国产环境功能特性完全一致。

9 安全合规类

TDSQL 是否能够《信息安全等级保护》3 级测评和备案

能够通过，TDSQL 已在多个客户侧通过等保三级或更高级别测评和备案。

TDSQL 是否能够《商用密码应用安全性评估》相关的密码安全（含密评 2.0）评估

能够通过，TDSQL 已在多个客户侧通过密评。需要说明的时，密评的密钥管理安全部分需要配合合规的密钥管理服务（KMS），这意味着您可能需要购买腾讯云 KMS 相关产品。

TDSQL 密码默认加密方式是什么，密码忘记了是否可以通过密文找回

TDSQL 密码默认采用 password（）函数加密，该函数默认使用 MySQLSHA1（一种安全哈希算法）算法；目前没有逆向解密或碰撞的方案，因此密码忘记后无法通过密文找回。