

分布式关系型数据库 DRDS

产品简介

产品简介

DRDS 概述

分布式关系型数据库服务 (Distributed Relational Database Service , 简称 DRDS) 专注于解决单机关系型数据库扩展性问题, 具备轻量 (无状态) 、灵活、稳定、高效等特性, 是阿里巴巴集团自主研发的中间件产品。DRDS 兼容 MySQL 协议和语法, 支持分库分表、平滑扩容、服务升降配、透明读写分离和分布式事务等特性, 具备分布式数据库全生命周期的运维管控能力。

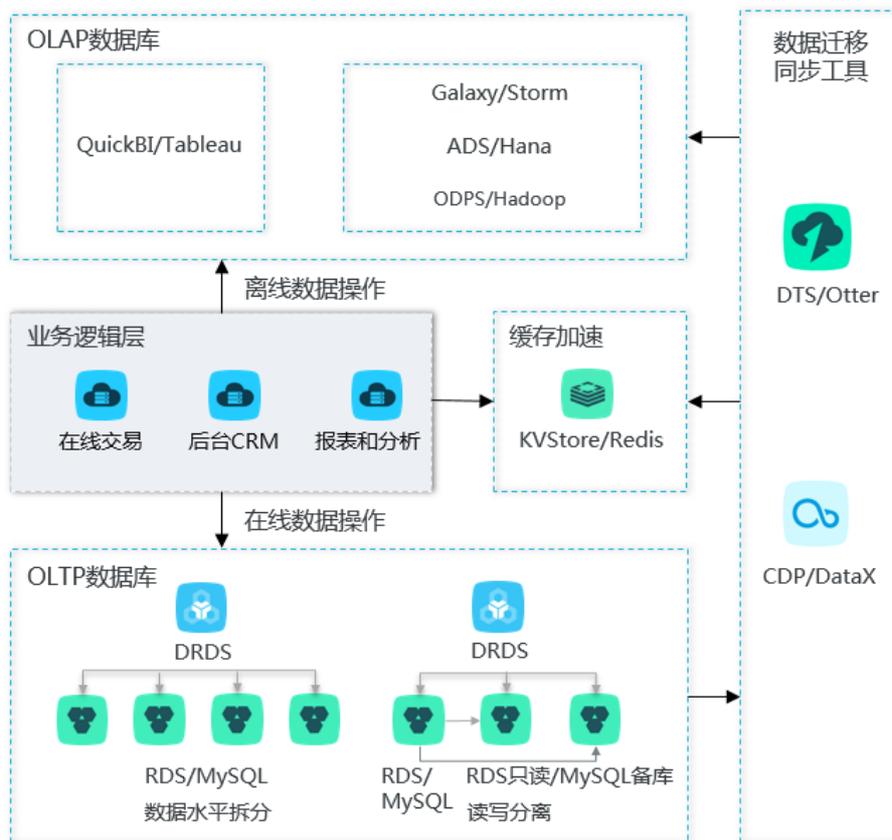
DRDS 主要应用场景在大规模在线数据操作上, 通过贴合业务的拆分方式, 将操作效率提升到极致, 有效满足用户在线业务对关系性数据库要求。

DRDS 主要解决了以下问题：

单机数据库容量瓶颈： 随着数据量和访问量的增长, 单机数据库会遇到很大的挑战, 依赖硬件升级并不能完全解决问题。

单机数据库扩展困难： 传统数据库容量扩展往往意味着服务中断, 很难做到业务无感知或者少感知。

传统数据库使用成本高： 当业务数据和访问量增加到一定量时, 传统数据库需要依赖特定的高端存储和小型机设备, 成本曲线快速上升。



功能特性

DRDS 的功能概要如下。

分库分表

支持 RDS/MySQL 的分库分表，在创建分布式数据库后，只需选择拆分键，DRDS 就可以按照拆分键生成拆分规则，实现数据水平拆分。

透明读写分离

通过使用 RDS 只读实例或者 MySQL 备机实现读写分离，帮助应用解决事务、只读实例或者备机挂掉、指定主备访问等细节问题，对应用无侵入，在 DRDS 控制台即可完成读写分离相关操作。

数据存储平滑扩容

当出现数据存储容量和访问量瓶颈时，DRDS 支持在线存储容量扩展，扩容无需应用改造，扩容进度支持可视

化跟踪。

服务升降配

DRDS 实例可以通过改变资源数量实现服务能力的弹性扩展。

分布式运维指令集

DRDS 提供独有分布式数据库运维指令集，如 SHOW SLOW、TRACE、SHOW NODE 等指令，帮助您快速发现和定位问题。

全局唯一数字序列

DRDS 支持分布式全局唯一且有序递增的数字序列。满足业务在使用分布式数据库下对主键或者唯一键以及特定场景的需求。

数据库账号权限体系

DRDS 支持类单机 MySQL 账号和权限体系，确保不同角色使用的账号操作安全。

分布式事务

DRDS 支持分布式柔性事务，保证分布式数据库数据一致性。

监控报警

DRDS 支持对核心资源指标和数据库实例指标的实时监控和报警，如实例 CPU、网络 IO、活跃线程等，帮助您实时发现资源和性能瓶颈。

产品优势

DRDS 优势概要如下。

分布式

数据读写存储集群化，不受单机限制，业务使用无连接数限制。

弹性

数据服务可升降配，数据存储白屏化 scale-up 和 scale-out，读写分离线性提升读能力。

高性能

分库分表经典方案让操作聚焦少量数据，多种拆分方式适应数据特点，并具备特定 SQL 并行执行能力，进一步提升执行效率。

安全

完整的类单机 MySQL 账号体系，提供具备授权鉴权的 Open API 方便集成能力到业务管控中，产品服务支持体系化。

简单易用

兼容 MySQL 协议和大部分 MySQL SQL 语法，无业务侵入式使用读写分离，全面的运维和监控能力。

成熟度高

参与阿里巴巴全部双十一活动，是阿里巴巴集团接入关系型数据库的标准。

应用场景

本文重点介绍 DRDS 的典型应用场景。

高并发实时交易场景

面向客户端的电商、金融、O2O、零售等行业普遍存在用户基数大、营销活动频繁、核心交易系统数据库响应日益变慢的问题，制约业务发展。DRDS 提供线性水平扩展能力，能够实时提升数据库处理能力，提高访问效率，峰值 TPS 达150万+，轻松应对高并发的实时交易场景。

海量数据存储访问场景

企业客户随着业务的快速发展，业务数据增长迅猛，会产生超过单机数据库存储能力极限的数据，造成数据库容量瓶颈，限制业务发展。DRDS 可以线性扩展存储空间，目前可以支持 200+ MySQL 实例的单数据库集群，提供 PB 级存储能力。可广泛应用于工业制造、智能家居、车联网等超大规模数据存储访问场景。

高性价比数据库解决方案

政务机构、大型企业、银行等行业为了支持大规模数据存储和高并发数据库访问，传统方案需要强依赖小型机和高端存储等高成本的商业解决方案，以达到服务能力扩展的目的。DRDS 能够利用普通服务器提供阿里巴巴双十一同等处理能力的高性价比国产化数据库解决方案。

低运维成本数据库

初创型企业初期发展阶段技术积累相对比较薄弱,资金投入有限,业务发展快,数据库的稳定性风险高。DRDS 继承了阿里巴巴多年的分布式数据库技术积累,能够提供简单易用的数据库运维系统,降低企业的技术运维成本,赋予企业强大的数据库支撑能力,为企业发展保驾护航。

数据备份

企业可以利用 DRDS 将自建数据库数据备份到云上,实现云上的数据备份容灾。

名词解释

本文主要对 DRDS 涉及的专有名词及术语进行定义和解析,方便您更好地理解相关概念并使用 DRDS。

DRDS

DRDS (Distributed Relational Database Service) 是阿里巴巴自主研发,高度兼容 MySQL 协议和语法的分布式关系型数据库服务中间件。

DRDS 服务节点 (DRDS Server)

DRDS Server 是 DRDS 核心组件,提供 SQL 的解析、优化、路由和结果归并。

DRDS 实例

DRDS 实例是由一组 DRDS Server 节点组成的分布式数据库服务集群。各服务节点无状态,同时处理 SQL 请求。

DRDS 实例规格

DRDS 实例规格是 DRDS 实例处理能力体现,按照 CPU 和 内存提供不同的规格的实例,规格越高处理能力越强。如 4Core8G 和 8Core16G,在标准的 DRDS 测试场景下,后者的处理能力是前者的两倍。

实例升降配

DRDS 可以通过改变实例规格来改变处理能力,提升实例规格称为升配,降低实例规格称为降配。

水平拆分

水平拆分是将一个单机数据库拆分为多个物理分库,将数据库中的表数据按照拆分规则,拆分为多个物理分表,存储到不同的数据库分库上的过程。

拆分规则

水平拆分过程中将逻辑数据库表拆分为多个物理分表规则称为拆分规则。

拆分键

水平拆分过程中，生成拆分规则的数据库字段称为拆分键。

分库

DRDS 水平拆分后，逻辑数据库数据存储多个物理存储实例上，每个存储实例上的物理库称为分库。

分表

DRDS 水平拆分后，每一个分库上的物理数据表称为分表。

逻辑 SQL

由应用端发送到 DRDS 的 SQL 称为逻辑 SQL。

物理 SQL

由 DRDS 对逻辑 SQL 进行解析之后发送到 RDS 上执行的 SQL 称为物理 SQL。

透明读写分离

DRDS 的单个存储实例节点遇到访问瓶颈时，可通过增加只读实例来分担主实例的压力。DRDS 的读写分离功能不需要修改任何应用代码，称为透明读写分离。

非拆分模式

DRDS 支持不进行数据库水平拆分而仅通过 DRDS 提供的透明读写分离来扩展数据库的服务能力。这种模式称为非拆分模式。

平滑扩容

DRDS 可通过增加存储实例节点完成数据库的扩容。扩容不影响原有数据的正常访问，称为平滑扩容。

小表广播

DRDS 将一些数据量小且更新频度不高的数据表存储为单表模式，这些数据表称为小表。通过数据同步将小表复制到与之 JOIN 的分库上进而提升 JOIN 效率的解决方案称为“小表广播”或者“小表复制”。

全表扫描

数据库拆分模式下，如果 SQL 语句中没有指定拆分键，DRDS 将在所有分表上执行 SQL 并归并结果返回，这个过程称为全表扫描。为避免影响性能，用户应尽量避免全表扫描。

全局唯一数字序列 (DRDS Sequence)

DRDS 全局唯一数字序列 (64位数字，对应 MySQL 中 BIGINT 类型) 的主要目标是为了保证所定义唯一字段中的数据的全局唯一 (比如 PRIMARY KEY , UNIQUE KEY 等) 和有序递增。

DRDS 自定义注释 (DRDS Hint)

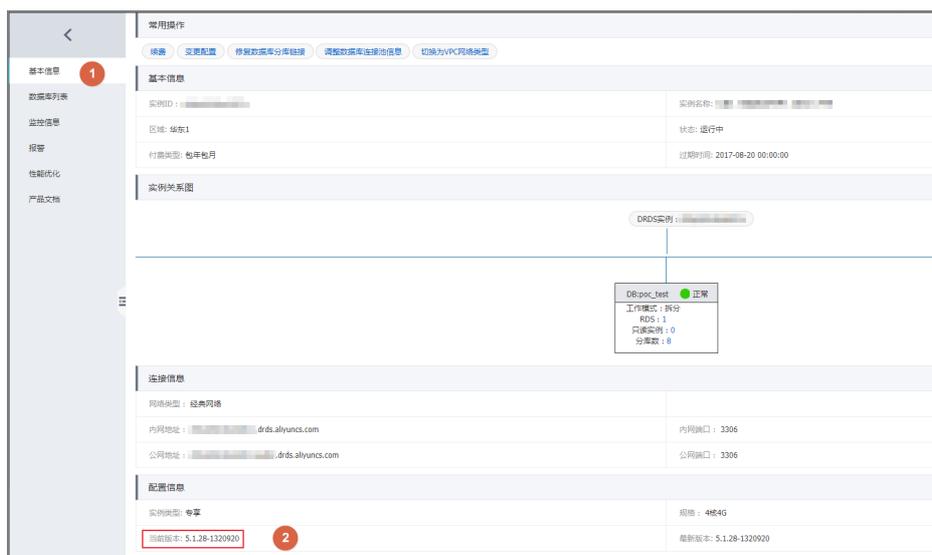
DRDS 提供的自定义注释，用于指定一些特殊行为，通过相关的语法影响 SQL 的执行方式，从而对 SQL 进行特殊的优化。

版本说明

如何查看 DRDS 实例的版本号

通过控制台查看

在实例的基本信息中可以查看当前实例的版本号，如图：



通过 version() 函数查看

通过 MySQL 命令行、DMS 等工具连接 DRDS，执行SELECT version()可以查看 DRDS 版本，例如：

```
mysql> select version();
+-----+
| VERSION() |
+-----+
| 5.6.29-TDDL-5.1.28-1320920 |
+-----+
1 row in set (0.00 sec)
```

其中 5.1.28-1320920 为 DRDS 的版本号。

如何升级 DRDS 实例版本

DRDS 实例版本可自助升级，具体操作请参考 [升级实例版本](#)。

V5.3.7-15460044 日期 2018.12.28

新功能

- 优化白名单/密码导致登陆失败提示信息
- 优化多表 JOIN，物理 SQL 生成多余子查询导致性能问题
- 支持 JOIN 下推顺序调整，优化 JOIN 性能
- 根据表行数调整内存 JOIN 顺序
- 支持逻辑库锁功能
- 优化执行器获取连接时，可能的多次执行 SET NAMES utf8mb4 的问题
- SHOW VARIABLES 性能优化
- 支持显示的 Prepare 和 Execute 语法
- SHOW FULL PROCESSLIST 时 SQL 显示不全问题修复
- 增加 HINT 支持表间并行执行模式

问题修复

- 修复 SQL 如：~1 函数报错问题
- 修复导入（Import数据库）数据库报错，以及导入后建表报错问题
- 多表 JOIN 报错如：（index (16) must be less than size (16)）
- 修复 SCAN HINT 对 LIMIT 处理不符合预期的问题

V5.3.6-15447056 日期 2018.12.13

问题修复

- 修复 用户自建账号查询 information_schema 报错，影响 DMS 及部分 GUI 工具使用的问题
- 修复 Filter-Join 结构执行计划下推不够精确，导致部分查询性能下降的问题
- 修复 特殊场景下生成的物理 SQL 中包含冗余子查询，导致未能使用 MySQL 索引的问题
- 修复 UNION 优化导致部分场景下生成异常物理 SQL 的问题
- 修复 相关联子查询生成物理 SQL 时的一个别名问题
- 修复 特殊场景下 LIMIT 穿透 JOIN 下推后查询结果异常的问题
- 修复 SHOW CREATE TABLE 未显示分表算法的问题
- 修复 建表时无法在 UNI_HASH 中指定字符串列的问题

V5.3.6-15439241 日期 2018.11.29

新功能

- 支持 为单个账户授予多个数据库权限，使用 USE DATABASE 切换数据库并执行 SQL
- 支持 SELECT 语句中，全部的列子查询和 WHERE 中的子查询

- 支持 Apply 子查询算子，允许嵌套执行无法转为 JOIN 的子查询
- 支持 HashBKAJoin 算子
- 支持 下推执行多表 UPDATE/DELETE 语句
- 支持 下推执行地理信息函数
- 支持 异构索引表加列顺序检测，避免 DDL 导致索引表同步异常
- 支持 IPv6

问题修复

- 优化 禁止使用 BINARY 类型作为拆分键，避免数据插入时自动补0导致查询异常
- 优化 全链路压测 PARSER 性能
- 修复 NODE HINT + LIMIT 下发错误 SQL，导致查询异常的问题
- 修复 部分条件未能下推，导致 LEFT JOIN 性能低问题
- 修复 部分场景下 BKA JOIN + LIMIT 执行异常的问题
- 修复 Simple Sequence 设置最大值时，越界的问题
- 修复 CREATE TABLE 对已存在表且带有 IF NOT EXISTS 关键字时报错问题

V5.3.5-15423348 日期 2018.11.20

问题修复

- 修复 并发策略选择导致部分场景下排序结果异常的问题
- 修复 NODE HINT 与 LIMIT 兼容性问题导致部分场景下结果异常的问题
- 修复 事务策略清理机制变化导致部分场景下 DTS 任务异常的问题
- 修复 BATCH INSERT 在部部分场景下执行异常的问题
- 修复 BATCH INSERT 特殊转义字符处理报错的问题

V5.3.5-15409153 日期 2018.10.30

新特性

- 优化 大幅降低 BATCH INSERT 内存占用，增强数据导入稳定性
- 优化 INSERT 并发策略，解决后端连接占用过高的问题
- 支持 完整的 GROUP_CONCAT 语法
- 支持 STR_HASH 哈希函数，兼容 DRDS 5.2 所有 HASH 函数
- 支持 UPDATE 语句中使用别名
- 支持 ON DUPLICATE KEY UPDATE 语法中指定表名
- 支持 下推执行 JSON 相关函数
- 支持 CONVERT(X, SIGNED) 语法
- 支持 JOIN USING 语法

问题修复

- 修复 物理SQL 中包含子查询时，JOIN 重复列名报错问题
- 修复 OUTLINE 相关细节问题
- 修复 TO_DAYS 函数时间遇到闰年返回值与 MySQL 不一致的问题
- 修复 建表主键、拆分键列名为关键字导致报错的问题
- 修复 INSERT SELECT 特殊语法报错的问题
- 修复 ORDER BY/GROUP BY 中同时包含函数导致执行计划异常的问题
- 修复 BIGINT 列在结果集中表示为 DECIMAL 的问题
- 修复 SHOW FULL TABLES FROM LIKE 报错的问题
- 修复 SHOW DATABASE 语句中包含换行符导致报错的问题

V5.3.4-15378085 日期 2018.09.20

新特性

- 支持 同步更新广播表，基于XA事务多写，无需异步数据同步
- 支持 全局事务策略等多种动态参数，可通过控制台设置
- 使用 XA 作为默认事务策略

问题修复

- 支持 SET @@SESSION.BINLOG_ROWS_QUERY_LOG_EVENTS
- 修复 BIT 类型问题
- 修复 BKAJoin + UNION 优化导致 JOIN 查询报错的问题
- 修复 Scalar 子查询的类型匹配问题
- 修复 SEMI/ANTI 执行器中 CURRENT() 方法未实现的问题
- 修复 不支持CURRENT_TIMESTAMP 特殊写法的问题

V5.3.3-1670435 日期 2018.8.15

全新特性

- 支持DRDS的XA事务；
- 支持单元化 Group Sequence。

问题修复

- 修复存在 HINT 时，没有正确设置 ThreadLocal 变量导致 NPE 的问题；
- 修复特殊场景下子查询条件上拉失败，导致全表扫描的问题；
- 修复子查询中拆分键条件提取失败导致全表扫描的问题；

- 修复计算物理分片时没有考虑 BETWEEN ... AND 的情况，导致全表扫描的问题；
- 修复可直接下推RDS的 For update SQL可能走了备库的问题；
- 修复 SELECT @@session.tx_read_only 返回值与MySQL不一致的问题；
- 修复算术表达式在某些情况下计算结果不正确的问题；
- 修复处理 0000-00-00 00:00:00 时间戳的取值报错的问题；
- 修复 NESTLOOP JOIN在数据较大的情况下可能导致 OOM 的问题；
- 修复存在DRDS HINT在部分查询场景会导致 NPE 的问题；
- 修复开启事务时，DRDS与RDS之间进行变量传递引发的报错问题；
- 修复SQL前加入多段DRDS的HINT时SQL解析器会报错的问题；
- 修复物理慢SQL输出不正确的物理表表名的问题；
- 修复因算子计算量过大，引发协程过度占用CPU，并影响IO响应的问题；
- 修复UNIX_TIMESTAMP() 函数不支持不带函数的形式，且带参数之后结果均为 NULL的问题；
- 修复UNI_HASH对于bigInt unsigned及int unsigned的路由不正确的问题；
- 修复Insert带HINT指定物理分库物理分表报表会表不存在的问题；
- 修复单库单表建表时错误走GroupSequence的问题；
- 修复Druid连接池在某些特殊情况下的Fail-Fast恢复异常所导致“because null” 报错，并引发连接泄漏问题；
- 修复SQL的limit子句被参数化导致物理SQL查询报错的问题，取消Limit子句参数化；
- 优化MysqlDump 兼容性问题处理，支持MySQL可执行HINT；
- 优化MysqlDump 兼容性问题处理，支持MySQL单行注释 “—” ；
- 优化精确选择 BKAJoin 算法的规则，解决不可下推 JOIN 内存占用过高的问题；
- 修复 INSERT VALUE 中包含转意字符时，下发SQL报错的问题；
- 修复 LEFT JOIN 条件推导不全，导致分片分片计算报错的问题；
- 修复驱动表较大时 BKAJoin的 StackOverFlow 的问题；
- 修复因Union查询优化没及时关闭连接导致连接泄漏问题；
- 修复特殊拆分规则导致拆分键大小写不匹配的问题；
- 修复类型相关的元数据存在的内存泄漏问题；
- 修复参数化缓存在SQL执行后没及时清空占用内存的问题；
- 修复涉及到UNI_HASH/RANGE_HASH/RIGHT_SHIFT三个拆分函数的跨库JOIN查询执行计划不正确下推并导致报错的问题；
- 修复SQL执行后返回JDBC的库名的meta信息不正确导致DMS/DTS操作报错的问题；
- XA事务校验set drds_transaction_policy的语句必须放在事务开始前，否则报错。

V5.3.2-1645122 日期 2018-7.05

- 修复带有Outline的SQL在高并发场景下会因出现无法生效并产生不合预期的执行计划的问题；
- 修复事务中的单个物理库的SELECT因使用并行查询策略导致拿物理连接报错的问题。

V5.3.2-1623342 日期 2018-5.25

新特性

- INSERT / UPDATE / DELETE 语句支持 5.2/5.3 版本 HINT ；

- 完善原生事务对 XA 事务的支持。

问题修复

- 修复结果集元信息中表名不正确的问题；
- 修复部分数据类型精度和范围异常的问题；
- 修复使用 Direct HINT 且仅指定部分逻辑表名时报错的问题；
- SHOW NODE 只统计读请求比例；
- 修复分库不分表情况下，关键字作为表名时查询异常的问题；
- 修复多 UNION 查询异常；
- 修复单表 INSERT , last_insert_id 返回 0 的问题。

V5.2.8-15432885 日期 2018.11.27

问题修复

- 修复当 UNION 可能性出现Error问题
- 修复多语句 UPDATE 返回的影响行数为0的问题
- 修复 Connection because null 问题
- 修复对于特殊查询 sql 鉴权失败问题
- 支持 2PC 分布式事务策略
- 只在满足 MySQL version > 5.6 以后才允许执行 XA 事务
- 兼容分布式事务中的 INSERT/REPLACE ... SELECT 语句
- 减少清理 DRDS 事务日志的周期和数据保留时间，降低 RDS 存储空间占用
- 修复 SHOW FULL TRANS LIMIT x 命令的问题

V5.2.7-1638709 日期 2018.6.25

问题修复

- 修复某些情况下会看到 Cannot find post-values for some PK 这个报错;
- 修复 SHOW TABLE STATUS 的平均行数的精度显示问题.

V5.2.7-1623038 日期 2018.5.24

问题修复

- 修复指定分库HINT不支持以时间函数为拆分键；
- 修复 SHOW TABLE STATUS 字段数值越界问题；
- 修复添加表权限表名大小写问题；
- 修复 YYYYDD 拆分函数在建表时跨年报错问题；

- 修复 YYYYMM 拆分函数对 NULL 值异常问题；
- 修复建表时 TIMESTAMP 类型精度问题。

V5.2.7-1606682 日期 2018.4.27

新特性

- 支持 DRDS 原生分布式事务，无须依赖其它外部组件，详情请参考 DRDS 分布式事务；
- 支持 SHOW TRANS 命令查看事务状态；
- 支持 SSL 加密功能；
- 支持单元化 Group Sequence。

问题修复

- 修复物理连接池初始化失败后导致的连接泄漏的问题；
- 修复当物理连接池初始化失败会导致的频繁反复建连接的现象；
- 修复临时表排序在处理 null 值会报错的问题；
- 解决分库分表键取值如 “_utf8' xxx' ” 前缀类型会报错的问题；
- 修复 MySQL Driver 因默认的 sharedSendBuffer 过大导致物理连接占用内存过多并在数据导入等场景引发 FGC 的问题。

V5.2.6-1556951 日期 2018.2.6

问题修复

- 修复事务及 GTS 事务中的读 SQL 会被路由到备库的问题；
- 修复 DRDS 自定义 HINT 不能同时作用于多张逻辑表（如 JOIN 的 SQL 语句）的问题；
- 修复 DDL 在部分并发场景下执行会报 “Lock wait timeout” 异常，导致 DDL 无法正常完成的问题；
- 修复一个 DB 在被多个 DRDS 实例同时引用时会出现的实例间的建（删）表结果不一致的问题；
- 修复因低版本的 Fastjson 安全缺陷造成的 DRDS 安全风险问题；
- 修复 DRDS 的 LAST_INSERT_ID() 返回值情况与 MySQL 不完全一致的问题。

V5.1.28-1508068 日期 2017.12.15

新特性

- 支持用户查询一张逻辑表下的各个物理分表的数据分布情况，请参考文档统计信息查询类语句；
- 新增拆分函数 UNI_HASH，能让分库数一样的两张表的 JOIN 保持为非跨库 JOIN，不再受到分表数目的影响；
- 增加支持 ISNULL() 函数；

- 优化权限相关的 SQL 错误提示，与 MySQL 保持一致；
- 支持只读库因同步延迟过大自动切断流量并回流主库的功能。

功能优化

- 优化执行器的列查找性能，INSERT/REPLACE 的吞吐提升 14%；
- 优化 JVM 的启动参数，CPU 消耗降低 5%；
- 优化 SQL 类型识别的算法性能；
- 改进跨库查询归并排序的算法性能；
- DDL 建表的组合索引的第一列包含拆分键时将不再重新单独为拆分列自动建索引。

问题修复

- 修复关联子查询中引用的外部字段在直接下推时拼错的问题；
- 修复 GTS 事务超过 100 记录列更新新的事务无法回滚的问题；
- 修复查询结果单个字段值过大会导致连接被意外关闭的问题；
- 修复若干 DRDS 后端连接池的一些连接泄漏问题和部分连接异常场景下不能自动恢复的问题；
- 修复超长 HINT 报数组越界的问题；
- 修复建表时 TIMESTAMP 类型的列指定为 NULL DEFAULT NULL 导致无法建表的问题；
- 修正部分错误码的英文翻译；
- 修复 STR_TO_DATE() 函数返回的时间类型与 MySQL 不一致的问题；
- 修复通过 Prepare 协议调用存储过程时传 NULL 值会报类型转换不正确的异常；
- 修复频繁设置连接编码会导致的内存泄漏问题；
- 修复在使用 GTS 事务之下执行 InsertSelect 出现读可见性不一致的问题；
- 修复带 DISTINCT 的 JOIN 语句在新版本 DRDS 报 Column is ambiguous 的问题；
- 修复 “((x) IS TRUE) IS TRUE” 语句执行报错的问题；
- 解决 DISTINCT 在部分跨库 JOIN 场景下会出现报错的问题；
- 修复分库分表情况下聚合函数查询带 LIMIT 时返回结果不正确的问题；
- 解决 BigDecimal 类型在类型转换过程会消耗大量 CPU 的问题；
- 修复 SHOW FULL STATS 连接数不准确的问题；
- 修复 SHOW FULL STATS CPU，内存不准确的问题；
- 修复当前端连接过多且因流控被堵塞在 Read 时，JVM 的临时内存 Buffer 出现堆积和泄漏的问题；
- 修复个别多语句处理场景中包含 ROLLBACK 语句可能导致的 NPE 错误；
- 修复 SHOW VARIABLES LIKE “autocommit” 显示的值不正确的问题；
- 修复 ResultSet.getObject 方法获取 tinyint(1) 类型的数据，返回结果类型跟 MySQL 不一致的问题。

V5.1.28-1408022 日期 2017.09.15

新特性

- 支持逻辑连接的 SHOW PROCESSLIST 与 KILL，详见 SHOW PROCESSLIST Syntax；

- 支持 INSERT INTO ... SELECT < sequence_name > .nextVal [, ...] FROM DUAL 语法；
- 支持 OPTIMIZE TABLE 语法，其作用请参考 Optimize Table；
- 增加对 ODBC 的支持；
- 增加 ANSI_QUOTES 类型 SQL_MODE 的支持，详见 MySQL SQL Mode；
- GROUP BY、ORDER BY、HAVING 子句中出现的函数不再必须出现在 SELECT 中；
- 分布式事务中，SELECT last_txc_xid(timeoutMillisecond) 支持指定超时时间。

功能优化

- 优化复杂 LIMIT 查询的性能；
- 优化 str2date 函数对时分秒的支持；
- 优化 json_set、json_object、json_extract、json_unquote、json_remove、json_merge、json_keys、json_replace、json_insert 函数在复杂 SQL 中的支持；
- 提升对 Navicat 的兼容性。

问题修复

- 修复 SHOW FULL STATS 的元信息类型错误的问题；
- 修复单库单表下 INSERT INTO ... ON DUPLICATE KEY UPDATE ... 语句报错的问题；
- 修复因实例过期续费再次使用时，建表失败的问题；
- 修复 DRDS 分库分表条件中拆分键为整数类型的负数条件查询报错的问题；
- 修复执行时表找不到的问题；
- 修复 DRDS 单库单表时因 LOCK、UNLOCK 语句导致数据无法导出的问题；
- 修复部分场景下中文表名时乱码的问题；
- 修复分布式事务中 FOR UPDATE 查询返回数据错误的问题。

V5.1.28-1349132 日期 2017.08.10

问题修复

- 修复新增 DB 关键字，导致无法使用 DB 作为列名/表名，并且影响 Navicat 从 DRDS 导入/导出数据的问题；
- 修复字符串类型拆分键的哈希值出现负最大值时会导致路由结果不正确的问题；
- 修复 SHOW FULL SLOW 指令在一些情况下会报 SLOW 表不存在的问题；
- 修复 Server Prepare 协议下，有小概率出现权限校验错误的问题；
- 修复使用拆分函数 RIGHT_SHIFT 和 RANGE_HASH 的拆分表在 INSERT 时，某些情况下出现的多语句无法执行的问题。

V5.1.28-1320920 日期 2017.07.18

新特性

- 支持 RENAME TABLE 功能；
- 增加了创建拆分表时新规则函数和多列规则的支持，详见双字段哈希拆分；
- 支持 REPLACE INTO SELECT 的用法；
- 增加了能够展示分库容量的 SHOW DB STATUS 命令，详见统计信息查询类语句；
- 支持 KILL ALL 命令批量中断慢 SQL 执行，详见 SHOW PROCESSLIST 指令与 KILL 指令；
- DRDS 控制台支持 RDS 5.7 时使用非拆分模式建库；
- DRDS 升降配后能够自动调整内部连接池的大小；
- DRDS 专有网络 vswitch 支持跨可用区；
- 通过 DRDS 控制台切换至 VPC 网络后，自动清理 DRDS 白名单；
- 改进了 DRDS 控制台在一些场景下的显示内容和提示信息。

问题修复

- 修复了高并发场景下偶发的客户端与 DRDS 连接意外中断 (MySQL server has gone away) 的问题；
- 解决了 DRDS 频繁执行 DDL 时，偶发出现的 DDL 执行中会挂起而无法恢复的问题，并提升了 DDL 的执行性能；
- 修复了通过 DRDS OPEN API 创建实例时返回 400 错误的问题；
- 解决了子账号通过 DMS 连接/登录 DRDS 的问题；
- 修复了 DRDS 控制台建库过程中，系统表创建失败导致的 DRDS 实例无法释放的问题；
- 修复了无法在 DRDS 控制台中从 VPC 网络切换回经典网络的问题。

V5.1.27-1282781 日期 2017.06.26

功能优化

- 默认禁止全表删除与全表更新的高危操作，可以通过加 HINT 临时跳过此限制，详见高危类 SQL 自动保护；
- 账号和权限功能调整：用户名长度范围改为 2-20，允许出现下划线字符；
- SHOW SLOW 和 SHOW FULL SLOW 指令显示结果中添加用户信息；
- 单表查询支持 LIMIT 0 优化；
- DELETE 语句的 ORDER BY 部分支持子查询；
- 在 SQL 日志中添加标识由哪个账号执行的信息。

问题修复

- 修复不含分号的语句被当成多语句执行的问题；
- 修复 SELECT 语句常量列设置别名导致提示列找不到的问题；
- 不缓存长度超过 2000 字节的 SQL 语句，避免占据内存导致性能下降。

V5.1.27-1217986 日期 2017.05.23

功能优化

- 修复 DRDS 建表时库名带上引号的报错问题；
- 修复 DRDS 建表时 TIMESTAMP 类型带上 on update current_timestamp 子句的报错问题；
- 修复 DRDS 控制台删除只读账号后不生效的问题；
- 修复 SQL 强制指定权重为0的只读实例进行查询的报错问题；
- 修复 DRDS 默认打参数化日志导致性能下降的问题；
- 修复 Group Sequence 更新系统 Sequence 表日期字段无效的问题。

V5.1.26-1502740 日期 2017.02.24

新特性

- 分布式账号和权限系统发布，支持企业级数据库帐号和资源管理；
- 新增 Time-based Sequence；
- 扩展 Sequence 语法，支持创建不同类型的 Sequence（默认为 Group Sequence），支持修改 Sequence 的类型。

功能优化

优化 Simple Sequence 性能。

V5.1.26-1037257 日期 2017.02.16

功能优化

- 修复激活访问 RDS 无法授权问题；
- 修复 RDS 使用 MySQL5.7 无法建库问题；
- 修复 DRDS 实例平滑扩展出错问题；
- 支持自助优化连接池参数。

V5.1.26-970482 日期 2017.01.10

新特性

- 支持自助优化连接池参数；
- 用户自助切换 DRDS 网络类型为 VPC 新功能；
- 控制台显示分库容量比例，方便用户确定库容方案；
- 控制台支持自建 MySQL 或者 RDS 平滑导入数据到 DRDS。

V5.1.26-9024000 日期 2016.12.13

新特性

- 控制台接入分布式全局事务，用户可以通过 DRDS 开通分布式事务。

V5.1.26-866978 日期 2016.11.24

新特性

- DRDS 控制台增加实例监控，包含 CPU、网络、物理 QPS、逻辑 QPS、连接数、活跃线程数；
- 新增 CPU 报警，第一时间发现 CPU 瓶颈，升配解决资源瓶颈；
- 增加 DRDS 慢 SQL 明细，随时查看实例慢 SQL。

V5.1.25-791778 日期 2016.10.09

新特性

- 支持服务端限流保护实例(默认不打开)；
- 中文表名和列名支持；
- 支持大小写敏感 Collation 类型；
- 支持 LIMIT row_count OFFSET 语法；
- 分布式事务升级到 2.0.17，支持超时时间、隔离级别动态配置。

性能优化：

- 优化第一次访问 DRDS 的性能；
- UPDATE 和部分查询语句局部性能优化。

问题修复：

- 修复子查询中多重 JOIN 列查找问题；
- 修复 SET 指令导致连接污染问题；
- 修复 SQL 语句中含有 Long.MIN_VALUE 导致溢出的问题；
- 修复数字类型的列错传了一个含有非常大的科学计数法数据的子字符串导致内存溢出的问题；
- 修复了唯一序列相关的几个问题；
- 修复了含有 GROUP BY 函数的子查询报错的问题；
- 修复 VARBINARY 类型的列在系统编码是 GBK 的情况下，取数据的结果会和数据库中的不一致的问题；
- 修复 LEFT JOIN 中含有子查询时丢失条件导致查询结果不正确的问题；
- 修复了部分场景下长 SQL 会导致内存溢出的问题；

- 包年包月实例变配前，需要检查用户是否创建了 RAM 角色；
- 修复了部分场景下分布式事务内存溢出问题。

V5.1.24-653395 日期 2016.07.26

新特性

- Sequence 支持全局有序自增，并提供增量、最大值等功能和相应语法的支持；
- 支持 to_days 函数；
- DRDS 支持包年包月购买；
- 接入 RAM 主子账户。

功能优化

- 完善库存不足、数据导入失败的告警信息；
- 数据导入状态查询增加自动重试。

Bug 修复情况

- 修改 USE 语句含有“;” 解析出错 Bug；
- 修复 SELECT、UPDATE 等语句在 WHERE 条件中含有单个条件且是永假式，WHERE 条件会丢掉 Bug；
- 修复升降配检查实例数上限的问题；
- 修复包年包月到期锁定共享实例的问题；
- 修复升降配失败时无法告警的问题；
- 数据导入移除对源表的外键检查；
- 数据导入检测目标表是否包含外键。

V5.1.24-826 日期 2016.05.24

新特性

- 数据拆分列的字段类型增加支持 BigDecimal 类型；
- 支持 Now(3)/Now(2)/Now(1) 函数；
- 增加只读账户 Open API；
- 增加修复 RDS 后端链接功能。

功能优化

- VPC 实例显示 VPC 地址。

Bug 修复情况

- 修复高并发访问 DRDS 后端连接长时间不会自动回收的问题；
- 修复后端连接池在主备切换后会出现概率性无法恢复的问题；
- 修复 Sequence 的缓存清除不干净，导致 Sequence 不可用的问题；
- 修复 INSERT SELECT 语句在广播表存在的情况下不会下推分片执行的问题；
- 修复 BigDecimal 类型的浮点数精度不会显示为指数的问题；
- 修复部分存储过程调用会导致连接进入不可恢复状态；
- 修复 SET @session.sql_log_bin=0 语句会导致连接池污染的问题；
- 修复开启 TXC 功能后 SQL 报错信息会显示异常的问题；
- 修复部分 DDL 执行过慢问题；
- 修复部分 Count(Distinct) 语句出现数据重复的问题。

V5.1.23-783 日期 2016.04.12

新特性

- 控制台新增 DRDS 实例级别监控；
- 控制台新增 DRDS 诊断建议，提前发现实例异常；
- 控制台新增 DRDS 实例 DB 拓扑；
- 控制台新增 DRDS 数据库概览；
- 提供 SHOW FULL SLOW | PHYSICAL_SLOW 指定，支持慢 SQL 明细的查询；
- 支持按用户指定单个 RDS 上的分库数量。

功能优化

- 变配失败后立即修复实例，不需要手动修复；
- 建库等待时间延长至半小时，解决等待 RDS DB 创建超时问题；
- 数据库列表页面删库，自动刷新。

V5.1.23-742 日期 2016.03.22

新特性

- 支持 now(3)/now(6) 这种函数用法；
- 支持非拆分模式下自定义函数执行；
- 支持非拆分表的 SELECT sql_calc_found_rows 与 SELECT found_rows 语法；
- 读写分离，支持多个只读实例之间的 failover 机制；
- 异常连接快速回收机制，比如识别 RDS 锁定状态，访问失败等；
- 支持 CHECK TABLE 语法，检查所有分片表结构一致性；
- 增加应用状态统计信息 SHOW STATS；

- 支持 yyyyymm 分区。

功能优化

- 优化 DDL 执行过慢问题；
- 完善 INSERT SELECT 语法的下推优化；
- 全局 ID 优化为大小写不敏感；
- 慢 SQL 统计优化，增加连接获取时间和 SQL 执行时间，支持过滤条件分析。

Bug 修复情况

- 修复 PREPARE 协议对于大文本对象解析的问题；
- 修复多表 JOIN 连接条件推导无效的问题。

V5.1.22-675 日期 2016.02.23

功能优化

- 建库输入库名后实时进行校验；
- 共享实例迁移 DB 到专享实例，自动刷新 RDS 白名单；
- Open API 建库刷新 RDS 白名单；
- 使用 RDS 高权限账号建库，扩容；
- 升降配过滤掉状态不正常的 RDS 实例；
- 数据导入支持 DECIMAL 类型和 DATE 类型的表主键；
- 数据导入切换时自动修改 Sequence；
- 设置实例读写比例时刷新只读实例的白名单；
- 自动恢复失败的数据导入任务。

V5.1.22-614 日期 2015.12.08

新特性

- 支持查看一个表非精确数据量；
- HINT 支持过滤条件；
- SCAN HINT 支持 EXPLAIN 语句，支持使用 EXPLAIN 查看所有分片上的执行计划；
- 支持 MySQLDump；
- 支持 MySQL 地理字段类型 geometry；
- 支持 INSERT SELECT 语句；
- 支持 GROUP_CONCAT 函数；
- 支持 GROUP BY 语法。

Bug 修复

- 修复只读账号不支持 SET 语句的问题；
- 修复 DELETE 语句中存在嵌套子查询，子查询中列无法找到的问题；
- 修复在 SELECT * FROM (UNION) WHERE 语句中，WHERE 条件丢失的问题；
- 修复建分库分表时，同时存在分库分表字段时，如果分库分表字段一个带反引号，一个不带，SQL 解析列名错误的问题；
- 修复 SQL 中存在大量 OR 条件时内存占用过多的问题；
- 修复当 SQL 中 GROUP BY/DISTINCT/AGGREGATE 和 LIMIT 混用时结果错误的问题；
- 修复 LEFT OUTER JOIN 中如果广播表在左边结果集错误的问题；
- 修复 SET sql_mode="" 会报错的问题。

V5.1.22-580 日期 2015.10.27

新特性

- 控制台新增 DRDS 数据库只读账号管理；
- 公有云支持域名作连接串。

功能优化

- 只读账户密码修改支持短信确认；
- 开启读写分离时，支持根据 Slave 延迟控制是否走只读实例；
- 建表时自动为分库键创建索引；
- 按时间和 ID 做双字段分区时，优化针对分库键的 GROUP BY 语句；
- 针对 Top N 的 ORDER BY LIMIT，使用内存堆排；
- 针对 DML 类型优化为单库下推模式；
- BEGIN/ROLLBACK/COMMIT 语句分号精确识别；
- DDL 语句，支持实例间并行执行；
- 支持 SET AUTOCOMMIT=on/off；
- CREATE TABLE IF NOT EXIST 语句优化容错；
- 针对广播表优化 TRUNCATE 语句，自动下发到所有分库上执行。

Bug 修复

- 修复 DMS 截断 DBPARTATION 语句的问题；
- 修复批量 INSERT 出现 "fatal exception" 时的空指针异常；
- 提前判断 MySQL 变量名称，防止传入错误的系统变量名，导致链接失效；
- 修复系统编码为 GBK 时，VARBINARY 类型 getBytes 结果与数据库不一致的问题；
- 修复 DDL 锁记录冲突 问题；
- 修复虚表 LEFT JOIN 问题；
- 修复三表做分库键上的 JOIN 时，分库键的值只被传递一次的问题；
- 修复 DATE(gmt_modified)<=? 语句 当右侧值为 DATE 类型时的空指针异常。

V5.1.21-513 日期 2015.09.07

新特性

- 支持 VPC 和商业化实例的 Open API ；
- 增加实例列表中显示网络类型 ；
- 跳转到 DMS 页面时自动填写可用 VPC 实例连接信息 ；
- 支持查看 DRDS 实例机器 IP 列表。

V5.1.21-478 日期 2015.08.06

新特性

- 如果 RDS 有白名单，提示用户将相关 IP 加入 RDS 白名单 ；
- DRDS 数据库初始化增加分布式事务日志表 ；
- DRDS 数据表拆分键自动加索引 ；
- 支持金融云用户购买实例。

功能优化

- DRDS 建库失败时自动清理垃圾数据 ；
- DRDS 导入数据库失败时，自动清理配置，支持重复导入 ；
- DDL 时过滤带 schema 的语句 ；
- 优化数据导入功能，增加 DRDS InstanceId 参数。

Bug 修复

- 修复聚石塔购买页面错误。

V5.1.21-464 日期 2015.07.21

新特性

- DRDS 商业化，增加按量实例购买入口，完成选配、询价、批量下单等功能 ；
- 实例生命周期管理，包括创建、欠费、续费、欠费释放 ；
- 支持主动释放实例 ；
- 数据导入候选表列表中过滤视图 ；
- 支持 MySQL Compress 压缩协议的读和写 ；
- 支持数据平滑导入 ；
- 支持读写分离。

功能优化

- 完善 MySQL Prepare 协议支持。