

数据仓库服务

用户指南

文档版本 01
发布日期 2025-09-24



版权所有 © 华为云计算技术有限公司 2025。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为云计算技术有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为云计算技术有限公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

目录

- 1 产品介绍..... 1
 - 1.1 什么是数据仓库服务..... 1
 - 1.2 产品优势..... 5
 - 1.3 应用场景..... 7
 - 1.4 产品功能..... 9
 - 1.5 基本概念..... 14
 - 1.6 与其他云服务的关系..... 16
 - 1.7 DWS 权限管理..... 17
 - 1.8 如何访问 DWS..... 20
 - 1.9 使用限制..... 21
 - 1.10 DWS 技术指标..... 25
- 2 入门..... 29
 - 2.1 第 1 步：入门前准备..... 29
 - 2.2 第 2 步：创建集群..... 29
 - 2.3 第 3 步：连接集群..... 32
 - 2.4 第 4 步：查看其它资料并清理资源..... 35
- 3 DWS 使用流程..... 37
- 4 准备工作..... 40
 - 4.1 策略语法：细粒度策略..... 40
 - 4.2 策略语法：RBAC..... 69
- 5 创建 DWS 集群..... 72
 - 5.1 创建 DWS 专属集群..... 72
 - 5.1.1 创建 DWS 存算一体集群..... 72
- 6 连接 DWS 集群..... 82
 - 6.1 连接 DWS 集群方式介绍..... 82
 - 6.2 获取 DWS 集群连接地址..... 83
 - 6.3 使用可视化工具连接 DWS 集群..... 85
 - 6.3.1 使用 Data Studio 连接 DWS 集群..... 85
 - 6.4 使用命令行工具连接 DWS 集群..... 88
 - 6.4.1 下载客户端..... 89
 - 6.4.2 使用 Linux gsql 客户端连接集群..... 90

6.4.3 使用 Windows gsql 客户端连接集群.....	94
6.4.4 使用 SSL 进行安全的 TCP/IP 连接.....	96
6.5 使用第三方库连接 DWS 集群.....	102
6.5.1 使用 JDBC 和 ODBC 连接 DWS 集群.....	102
6.5.1.1 开发规范.....	102
6.5.1.2 下载 JDBC 或 ODBC 驱动.....	102
6.5.1.3 使用 JDBC 连接.....	103
6.5.1.4 配置 JDBC 连接（使用负载均衡方式）.....	114
6.5.1.5 配置 JDBC 连接（使用 IAM 认证方式）.....	116
6.5.1.6 JDBC 配置数据库第三方连接池.....	119
6.5.1.7 使用 ODBC 连接.....	132
6.5.2 使用 Python 第三方库 psycopg2 连接 DWS 集群.....	138
6.5.3 使用 Python 第三方库 PyGreSQL 连接 DWS 集群.....	147
7 创建 DWS 数据库和用户.....	163
8 迁移业务数据至 DWS 集群.....	169
8.1 管理数据源.....	169
8.1.1 管理 MRS 数据源.....	169
8.1.1.1 MRS 数据源使用概述.....	169
8.1.1.2 创建 MRS 数据源连接.....	170
8.1.1.3 更新 MRS 数据源配置.....	172
8.1.2 管理 OBS 数据源.....	174
9 DWS 集群数据安全与加密.....	178
9.1 开启 DWS 数据库用户三权分立.....	178
9.2 使用 KMS 对 DWS 集群进行加密.....	180
9.2.1 数据库加密简介.....	180
9.2.2 轮转加密密钥.....	182
9.2.3 转加密集群.....	182
10 DWS 集群管理.....	184
10.1 查看 DWS 集群详情.....	184
10.2 查看 DWS 集群状态.....	188
10.3 查看 DWS 集群拓扑图.....	192
10.4 管理 DWS 集群连接.....	197
10.4.1 管理 DWS 集群访问域名.....	197
10.4.2 为 DWS 集群绑定和解绑弹性负载均衡.....	200
10.4.3 增删 DWS 集群 CN 节点.....	201
10.4.4 管理 DWS 数据库连接.....	203
10.5 DWS 资源负载管理.....	205
10.5.1 概述.....	205
10.5.2 资源池.....	207
10.5.2.1 功能介绍.....	207
10.5.2.2 添加资源池.....	210

10.5.2.3 修改资源池.....	211
10.5.2.4 删除资源池.....	213
10.5.3 资源管理计划.....	214
10.5.3.1 资源管理计划操作.....	214
10.5.3.2 资源管理计划阶段.....	215
10.5.4 空间管理.....	218
10.6 修改 DWS 集群 GUC 参数.....	219
10.7 管理 DWS 标签.....	230
10.7.1 标签简介.....	230
10.7.2 标签管理.....	231
10.8 重置 DWS 数据库系统管理员密码.....	233
10.9 启停和删除 DWS 集群.....	234
10.10 管理企业项目.....	236
11 DWS 集群运维.....	238
11.1 在 Cloud Eye 查看 DWS 集群监控.....	238
11.2 查看并订阅 DWS 集群告警.....	245
11.2.1 告警管理简介.....	245
11.2.2 订阅告警通知.....	249
11.3 查看并订阅 DWS 集群事件.....	251
11.3.1 事件通知概述.....	251
11.3.2 订阅事件通知.....	256
11.3.3 查看事件.....	257
11.4 备份与恢复 DWS 集群.....	258
11.4.1 快照概述.....	258
11.4.2 手动快照.....	259
11.4.2.1 创建集群粒度的手动快照.....	259
11.4.2.2 创建 schema 粒度的手动快照.....	260
11.4.2.3 删除手动快照.....	262
11.4.3 自动快照.....	263
11.4.3.1 自动快照概述.....	263
11.4.3.2 设置自动快照策略.....	264
11.4.3.3 复制自动快照.....	266
11.4.3.4 删除自动快照.....	267
11.4.4 查看快照信息.....	267
11.4.5 恢复快照.....	270
11.4.5.1 恢复快照约束.....	270
11.4.5.2 恢复快照到新集群.....	270
11.4.5.3 恢复快照到当前集群.....	272
11.4.5.4 恢复单表到原集群.....	273
11.4.5.5 恢复单表/多表到新集群.....	275
11.4.6 快照参数配置.....	277
11.4.7 停止快照.....	280

11.5 DWS 集群节点扩缩容.....	281
11.5.1 查看巡检.....	281
11.5.2 节点管理.....	288
11.5.3 节点变更.....	290
11.5.3.1 集群扩容.....	290
11.5.3.2 集群重分布.....	293
11.5.3.2.1 重分布.....	293
11.5.3.2.2 查看重分布详情.....	294
11.6 DWS 集群规格变更.....	296
11.6.1 弹性变更规格.....	296
11.6.2 EVS 集群磁盘扩容.....	297
11.7 DWS 集群容灾管理.....	298
11.7.1 DWS 集群容灾场景介绍.....	298
11.7.2 创建与启动 DWS 集群容灾.....	300
11.7.3 切换 DWS 集群容灾.....	302
11.7.4 停止与删除 DWS 集群容灾.....	304
11.8 DWS 集群升级.....	305
11.9 DWS 集群日志管理.....	306
11.9.1 DWS 集群支持的日志类型说明.....	306
11.9.2 转储 DWS 数据库审计日志.....	307
11.9.3 查看 DWS 数据库审计日志.....	314
11.9.4 查看 DWS 管理控制台操作日志.....	317
11.9.5 查看 DWS 集群其他日志.....	321
11.10 DWS 集群状态异常操作.....	323
11.11 Vacuum 回收 DWS 空间.....	324
11.11.1 智能运维概览.....	324
11.11.2 管理运维计划.....	325
11.11.3 查看运维任务.....	328
12 常见问题.....	330
12.1 产品咨询.....	330
12.1.1 DWS 用户和角色是什么关系？	330
12.1.2 如何查看 DWS 数据库用户的创建时间？	331
12.1.3 如何选择 DWS 区域和可用分区.....	332
12.1.4 数据在 DWS 中是否安全？	333
12.1.5 可以修改 DWS 集群的安全组吗？	334
12.1.6 DWS 的脏页是如何产生的？	334
12.2 数据库连接.....	335
12.2.1 如何与 DWS 进行通信？	335
12.2.2 DWS 是否支持第三方客户端以及 JDBC 和 ODBC 驱动程序？	337
12.2.3 无法连接 DWS 集群时怎么处理？	337
12.2.4 为什么在互联网环境连接 DWS 后，解绑了 EIP 不会立即返回失败消息？	338
12.2.5 使用公网 IP 连接 DWS 集群时如何设置白名单？	339

12.2.6 使用 API 调用和直连数据库两种方式有哪些差异？	339
12.3 数据迁移.....	341
12.3.1 DWS 的 OBS 外表与 GDS 外表支持的数据格式有什么区别？	341
12.3.2 数据如何存储到 DWS？	341
12.3.3 DWS 可以存储多少业务数据？	341
12.3.4 如何使用 DWS 的\copy 导入导出？	341
12.3.5 如何实现 DWS 不同编码库之间数据容错导入.....	342
12.3.6 DWS 导入性能都和哪些因素有关联？	343
12.4 数据库使用.....	343
12.4.1 如何调整 DWS 分布列？	343
12.4.2 如何查看和设置 DWS 数据库的字符集编码格式.....	345
12.4.3 如何处理 DWS 建表时 date 类型字段自动转换为 timestamp 类型的问题？	346
12.4.4 DWS 是否需要定时对常用的表做 VACUUM FULL 和 ANALYZE 操作？	347
12.4.5 如何导出 DWS 某张表结构？	349
12.4.6 DWS 是否有高效的删除表数据的方法？	349
12.4.7 如何查看 DWS 外部表信息？	350
12.4.8 如果 DWS 建表时没有指定分布列，数据会怎么存储？	351
12.4.9 如何将 DWS 联结查询的 null 结果替换成 0？	352
12.4.10 如何查看 DWS 表是行存还是列存？	352
12.4.11 DWS 列存表的常用信息查询.....	353
12.4.12 DWS 查询时索引失效场景解析.....	354
12.4.13 如何使用 DWS 自定义函数改写 CRC32()函数.....	361
12.4.14 DWS 以 pg_toast_temp*或 pg_temp*开头的 Schema 是什么？	362
12.4.15 DWS 查询时结果不一致的常见场景和解决方法.....	362
12.4.16 DWS 哪些系统表不能做 VACUUM FULL.....	367
12.4.17 DWS 语句处于 idle in transaction 状态常见场景.....	368
12.4.18 DWS 如何实现行转列及列转行？	370
12.4.19 DWS 唯一约束和唯一索引有什么区别？	373
12.4.20 DWS 函数和存储过程有什么区别？	374
12.4.21 如何删除 DWS 重复的表数据？	375
12.5 集群管理.....	377
12.5.1 如何清理与回收 DWS 存储空间？	377
12.5.2 为什么 DWS 扩容后已使用存储容量比扩容前减少了很多？	380
12.5.3 DWS 的磁盘空间/容量是如何统计的？	380
12.5.4 DWS 添加云监控服务的告警规则时会话数阈值如何设置？	381
12.5.5 DWS 增加 CN 和扩容集群分别在什么场景下使用？	381
12.5.6 DWS 在 CPU 核数、内存相同的情况下，小规格多节点与大规格三节点集群如何选择？	382
12.5.7 DWS 热数据存储和冷数据存储的区别？	382
12.5.8 DWS 扩容按钮置灰如何处理？	383
12.6 账户与权限.....	383
12.6.1 DWS 如何实现业务隔离.....	383
12.6.2 DWS 数据库账户密码到期了，如何修改？	387

12.6.3 如何给 DWS 指定用户赋予某张表的权限？	387
12.6.4 如何给 DWS 指定用户赋予某个 SCHEMA 的权限？	391
12.6.5 如何创建 DWS 数据库只读用户？	393
12.6.6 如何创建 DWS 数据库私有用户和私有表？	394
12.6.7 DWS 如何 REVOKE 某用户的 connect on database 权限？	395
12.6.8 如何查看 DWS 某个用户有哪些表的权限？	396
12.6.9 DWS 数据库中的 Ruby 是什么用户？	399
12.7 数据库性能.....	399
12.7.1 为什么 DWS 使用一段时间后执行 SQL 很慢？	399
12.7.2 为什么 DWS 的性能在极端场景下并未比单机数据库好.....	400
12.7.3 DWS 业务读写阻塞，如何查看某个时间段的 SQL 执行记录？	400
12.7.4 DWS 的 CPU 资源隔离管控介绍.....	400
12.7.5 为什么 DWS 普通用户比 dbadmin 用户执行的慢？	402
12.7.6 DWS 中单表查询性能与哪些因素有关？	403
12.7.7 如何优化包含多个 CASE WHEN 条件的 SQL 查询？	404
12.8 备份恢复.....	405
12.8.1 为什么 DWS 自动快照创建很慢，很长时间都没有创建好？	405
12.8.2 DWS 快照是否与 EVS 快照功能相同？	405

1 产品介绍

1.1 什么是数据仓库服务

数据仓库服务DWS是一种基于华为云基础架构和平台的在线数据分析处理数据库，提供即开即用、可扩展且完全托管的分析型数据库服务，兼容ANSI/ISO标准的SQL92、SQL99和SQL 2003语法，同时兼容PostgreSQL/Oracle/Teradata/MySQL等数据库生态，为各行业PB级海量大数据分析提供有竞争力的解决方案。

集群逻辑架构

DWS集群逻辑架构如图1-1所示。实例的详细介绍请参见表1-1。

图 1-1 集群逻辑架构图

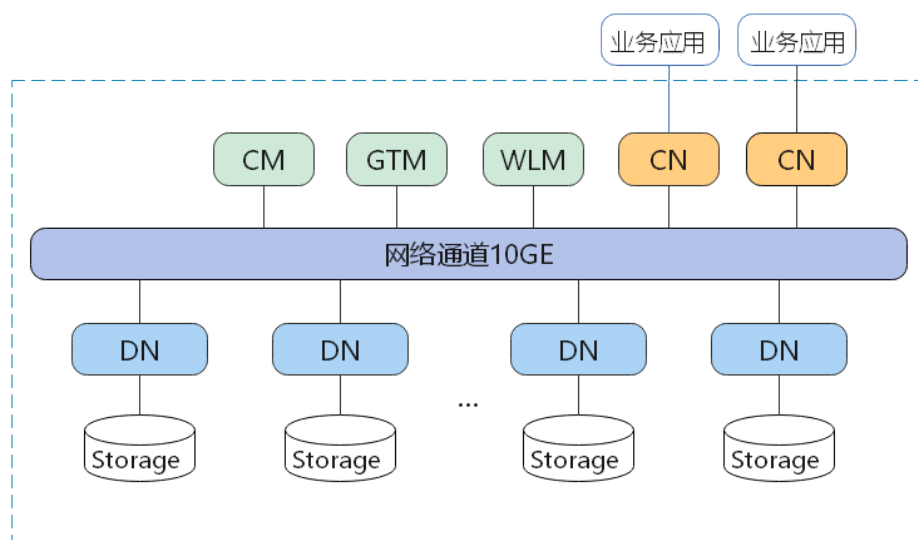


表 1-1 集群架构说明

名称	描述	说明
CM	集群管理模块（Cluster Manager）。管理和监控分布式系统中各个功能单元和物理资源的运行情况，确保整个系统的稳定运行。	CM由CM Agent、OM Monitor和CM Server组成。 - CM Agent：负责监控所在主机上主备 GTM、CN、主备DN的运行状态并将状态上报给CM Server。同时负责执行CM Server下发的仲裁指令。集群的每台主机上均有CM Agent进程。 - OM Monitor：看护CM Agent的定时任务，其唯一的任务是在CM Agent停止的情况下将CM Agent重启。如果CM Agent重启不了，则整个主机不可用，需要人工干预。 说明 CM Agent重启的情况很少发生，如果出现可能是因为系统资源不够用导致无法启动新进程。 - CM Server：根据CM Agent上报的实例状态判定当前状态是否正常，是否需要修复，并下发指令给CM Agent执行。 DWS提供了CM Server的主备实例方案，以保证集群管理系统本身的高可用性。正常情况下，CM Agent连接主CM Server，在主CM Server发生故障的情况下，备CM Server会主动升为主CM Server，避免出现CM Server单点故障。
GTM	全局事务管理器（Global Transaction Manager），负责生成和维护全局事务ID、事务快照、时间戳等全局唯一的信息。	整个集群只有一组GTM：主、备GTM各一个。
WLM	工作负载管理器（Workload Manager）。控制系统资源的分配，防止过量业务负载对系统的冲击而导致业务拥塞和系统崩溃。	不同于集群中的实例（GTM、CM、CN、DN）模块，不需要在安装过程中指定主机名称。安装程序会自动在各主机上安装此模块。

名称	描述	说明
CN	协调节点（Coordinator）。负责接收来自应用的访问请求，并向客户端返回执行结果；负责分解任务，并调度任务分片在各DN上并行执行。	集群中，CN有多个并且CN的角色是对等的（执行DML语句时连接到任何一个CN都可以得到一致的结果）。只需要在CN和应用程序之间增加一个负载均衡器，使得CN对应用是透明的。CN故障时，由负载均衡自动路由连接到另外一个CN，请参见“集群绑定和解绑ELB”章节。 当前分布式事务框架下无法避免CN之间的互连，为了减少GTM上线程过多导致负载过大，建议CN配置数目≤ 10个。 DWS通过CCN（Central Coordinator）负责集群内的资源全局负载控制，以实现自适应的动态负载管理。CM在第一次集群启动时，通过集群部署形式，选择编号最小的CN作为CCN。若CCN故障之后，由CM选择新的CCN进行替换。
DN	数据节点（Datanode）。负责存储业务数据（支持行存、列存、混合存储）、执行数据查询任务以及向CN返回执行结果。	在集群中，DN有多个。每个DN存储了一部分数据。DWS对DN提供了高可用方案：主DN、备DN、从备DN。三者的工作原理如下： 主、备DN同步数据期间，如果主DN突然故障不可用，备DN会升为主DN。 - 在原主DN恢复前，新升为主DN会将数据日志同步到从备DN。 - 原主DN恢复后将成为备DN，并且会使用“从备DN”上的数据日志恢复异常期间的数据。 也就是说从备DN永远只作为从备使用，不会因为主DN或备DN故障而升级为主DN或备DN，从备DN只存放原主DN故障时，新升为主DN同步到从备DN的Xlog数据和数据通道复制产生的数据。因此从备DN不额外占用存储资源，相比传统三副本节约了三分之一的存储空间。
Storage	服务器的本地存储资源，持久化存储数据。	-

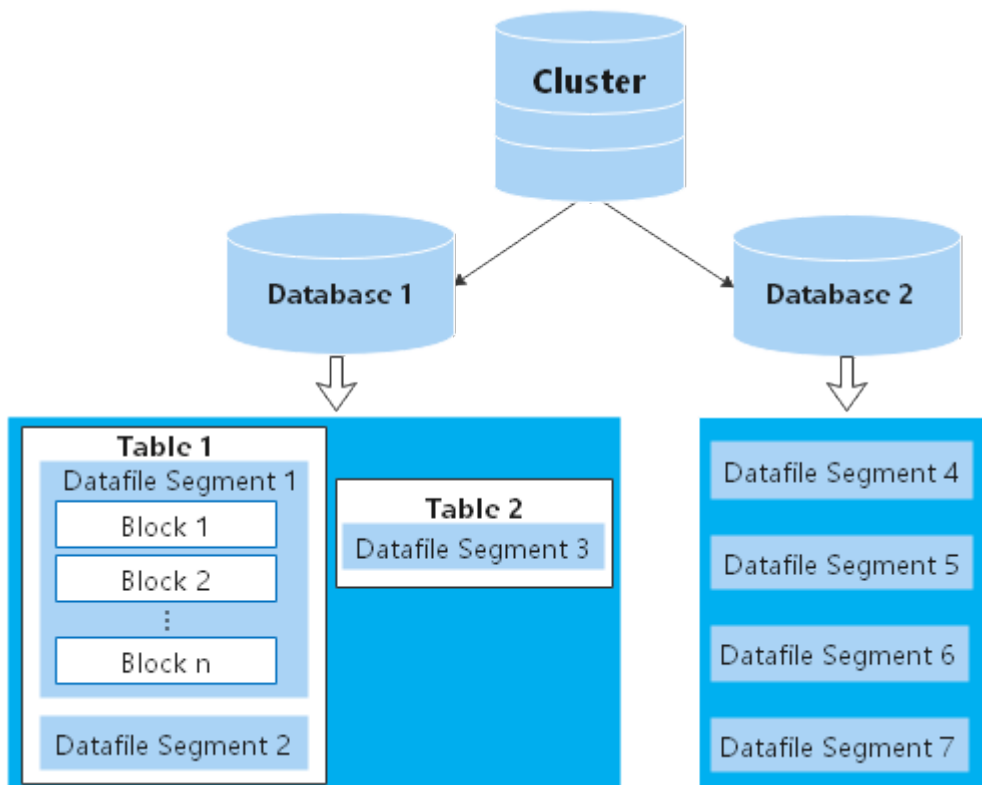
集群的每个DN上负责存储数据，其存储介质也是磁盘。图1-2从逻辑上介绍了每个DN上都有哪些对象，以及这些对象之间的关系，其中：

- Database，即数据库，用于管理各类数据对象，各数据库间相互隔离。
- Datafile Segment，即数据文件，通常每张表只对应一个数据文件。如果某张表的数据大于1GB，则会分为多个数据文件存储。
- Table，即表，每张表只能属于一个数据库。

- Block，即数据块，是数据库管理的基本单位，默认大小为8KB。

数据有三种分布方式，可以在建表的时候指定：REPLICATION、ROUNDROBIN、HASH。

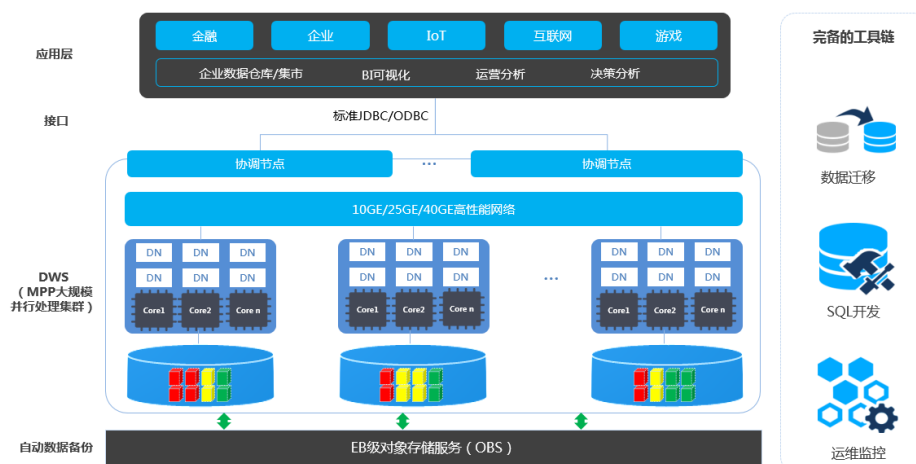
图 1-2 数据库逻辑结构图



存算一体架构

DWS基于Shared-nothing分布式架构，具备MPP（Massively Parallel Processing）大规模并行处理引擎，由众多拥有独立且互不共享的CPU、内存、存储等系统资源的逻辑节点组成。在这样的系统架构中，业务数据被分散存储在多个节点上，数据分析任务被推送到数据所在位置就近执行，并行地完成大规模的数据处理工作，实现对数据处理的快速响应。

图 1-3 产品架构



- **应用层**

数据加载工具、ETL（Extract-Transform-Load）工具、以及商业智能BI工具、数据挖掘和分析工具，均可以通过标准接口与DWS集成。DWS兼容PostgreSQL生态，且SQL语法进行了兼容Oracle和Teradata的处理。应用只需做少量改动即可向DWS平滑迁移。

- **接口**

支持应用程序通过标准JDBC和ODBC连接DWS。

- **DWS**

一个DWS集群由多个在相同子网中的相同规格的节点组成，共同提供服务。集群的每个DN负责存储数据，其存储介质是磁盘。协调节点（Coordinator）负责接收来自应用的访问请求，并向客户端返回执行结果。此外，协调节点还负责分解任务，并调度任务分片在各DN上并行执行。

- **自动数据备份**

支持将集群快照自动备份到EB级对象存储服务OBS（Object Storage Service）中，方便利用业务空闲期对集群做周期备份以保证集群异常后的数据恢复。

快照是DWS集群在某一时间点的完整备份，记录了该时刻指定集群的所有配置数据和业务数据。

- **工具链**

提供了数据并行加载工具GDS（General Data Service）、SQL语法迁移工具DSC（Database Schema Converter）、SQL开发工具Data Studio、迁移工具GDS-Kafka，并支持通过控制台对集群进行运维监控。

1.2 产品优势

数据仓库服务 DWS兼容ANSI/ISO标准的SQL92、SQL99和SQL 2003语法，同时兼容PostgreSQL/Oracle/Teradata/MySQL等数据库生态，为各行业PB级海量大数据分析提供有竞争力的解决方案。

DWS与传统数据仓库相比，主要有以下特点与显著优势，可解决多行业超大规模数据处理与通用平台管理问题：

易使用

- 一站式可视化便捷管理
DWS让您能够轻松完成从项目概念到生产部署的整个过程。通过使用DWS管理控制台，您不需要安装数据仓库软件，也不需要部署数据仓库服务器，就可以在几分钟之内获得高性能、高可靠的企业级数据仓库集群。
您只需单击几下鼠标，就可以轻松完成应用程序与数据仓库的连接、数据备份、数据恢复、数据仓库资源和性能监控等运维管理工作。
- 与大数据无缝集成
您可以使用标准SQL查询HDFS、对象存储服务（Object Storage Service，OBS）上的数据，数据无需搬迁。
- 提供一键式异构数据库迁移工具
DWS提供配套的迁移工具，可支持Oracle和Teradata的SQL脚本迁移到DWS。

高性能

- 云化分布式架构
DWS采用全并行的MPP架构数据库，业务数据被分散存储在多个节点上，数据分析任务被推送到数据所在位置就近执行，并行地完成大规模的数据处理工作，实现对数据处理的快速响应。
- 查询高性能，万亿数据秒级响应
DWS通过算子并行执行、向量化执行引擎实现指令在寄存器并行执行，以及LLVM动态编译减少查询时冗余的条件逻辑判断，助力数据查询性能提升。
DWS支持行列混合存储，可以同时为用户提供更优的数据压缩比（列存）、更好的索引性能（列存）、更好的点更新和点查询（行存）性能。
- 数据加载快
DWS提供了GDS极速并行大规模数据加载工具。
- 列存下的数据压缩
对于非活跃的早期数据可以通过压缩来减少空间占用，降低采购和运维成本。
DWS列存储压缩支持Delta Value Encoding、Dictionary、RLE、LZ4、ZLIB等压缩算法，且能够根据数据特征自适应的选择压缩算法，平均压缩比7:1。压缩数据可直接访问，对业务透明，极大缩短历史数据访问的准备时间。

易扩展

- 按需扩展：Shared-Nothing开放架构，可随时根据业务情况增加节点，扩展系统的数据存储能力和查询分析性能。
- 扩容后性能线性提升：容量和性能随集群规模线性提升，线性比0.8。
- 扩容不中断业务：扩容过程中支持数据增、删、改、查，及DDL操作(Drop/Truncate/Alter table)，表级别扩容技术，扩容期间业务不中断、无感知。
- 支持在线升级：8.1.1及以上源版本支持大版本在线升级，8.1.3及以上源版本支持补丁在线升级，升级期间用户无需停止业务，业务存在闪级秒断。

高可靠

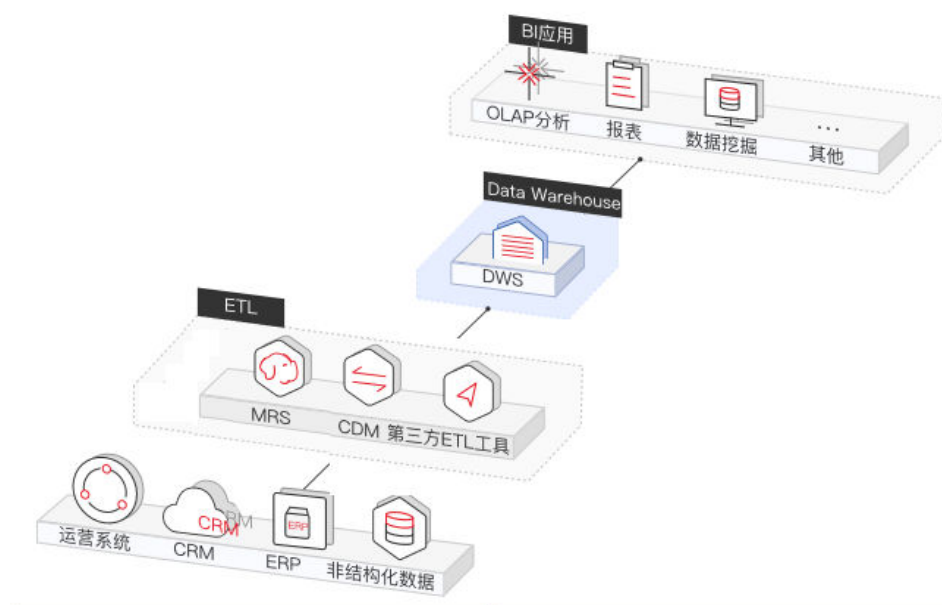
- 事务管理
 - 支持事务块，用户可以通过start transaction命令显式启动一个事务块。
 - 支持单语句事务，用户不显式启动事务，则单条语句就是一个事务。
 - 分布式事务管理。支持全局事务信息管理，包括gxid、snapshot、timestamp的管理，分布式事务状态管理，gxid溢出的处理。

- 分布式事务支持ACID特性（Atomicity, Consistency, Isolation, Durability），数据强一致保证。
- 支持分布式死锁预防，保证在出现死锁时自动解锁或者预防死锁。
- 全方位HA设计
DWS所有的软件进程均有主备保证，集群的协调节点（CN）、数据节点（DN）等逻辑组件全部有主备保证，能够保证在任意单点物理故障的情况下系统依然能够保证数据可靠、一致，同时还能对外提供服务。
- 安全
DWS支持数据透明加密，同时可与数据库安全服务（DBSS）对接，基于网络隔离及安全组规则，保护系统和用户隐私及数据安全。DWS还支持自动数据全量、增量备份，提升数据可靠性。

1.3 应用场景

增强型的 ETL+实时 BI 分析

图 1-4 ETL+BI 分析场景



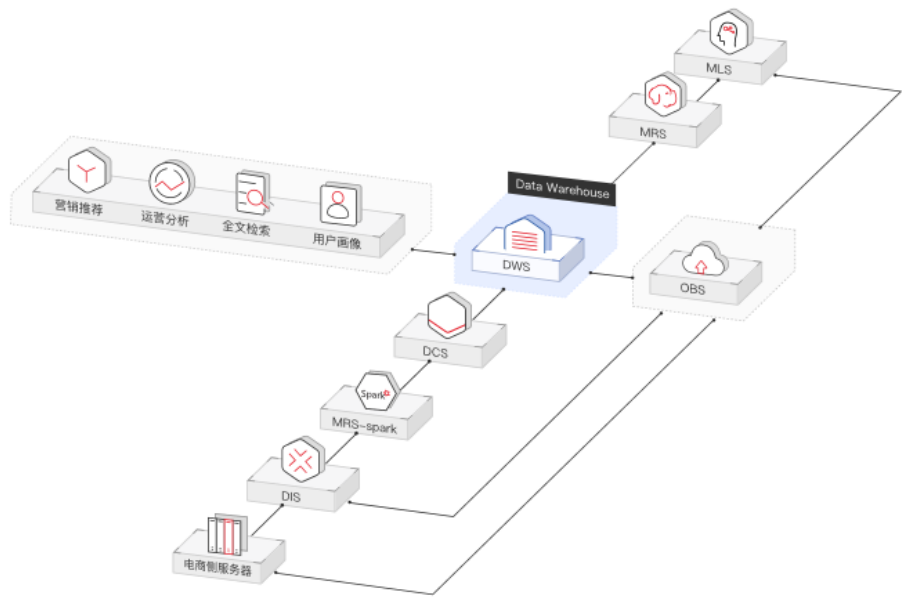
数据仓库在整个BI（business intelligence）系统中起到了支柱的作用，更是海量数据收集、存储、分析的核心。为移动互联网、游戏行业、O2O（Online to Offline）等行业提供强大的商业决策分析支持。

优势如下：

- 数据迁移：多数据源，高效、批量、实时的数据导入。
- 高性能：PB级数据低成本的存储与万亿级数据关联分析秒级响应。
- 实时：业务数据流实时整合，及时对运营决策进行优化与调整。

电商场景

图 1-5 电商场景

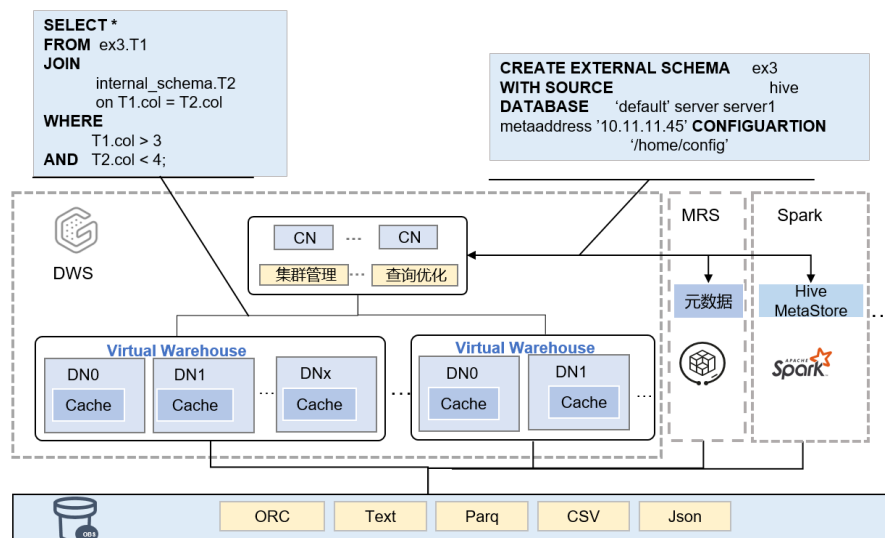


电商的数据分析，主要用于支撑营销推荐、运营分析、全文检索、客户分析等业务。
优势如下：

- 分析维度广：可从商品、用户、运营、地区等多个维度进行复杂的分析。
- 随业务增长扩容：可根据业务增长，随时进行集群规模扩展。
- 可靠性高：DWS的高可靠性支撑电商系统的长时稳定运行。

湖仓一体

- **无缝访问数据湖**
 - 对接Hive Metastore元数据管理，直接访问数据湖的数据表定义，无需用户创建外表，只需创建external schema即可。
 - 支持主要数据格式：ORC，Parquet。
- **融合查询**
 - 混合查询数据湖和仓内的任意数据。
 - 查询一步到位输出到仓内/数据湖，无需额外数据中转拷贝。
- **极致查询性能**
 - 使用数仓高质量的查询计划和高效的执行引擎。
 - 使用数仓的负载管理手段，精准控制。

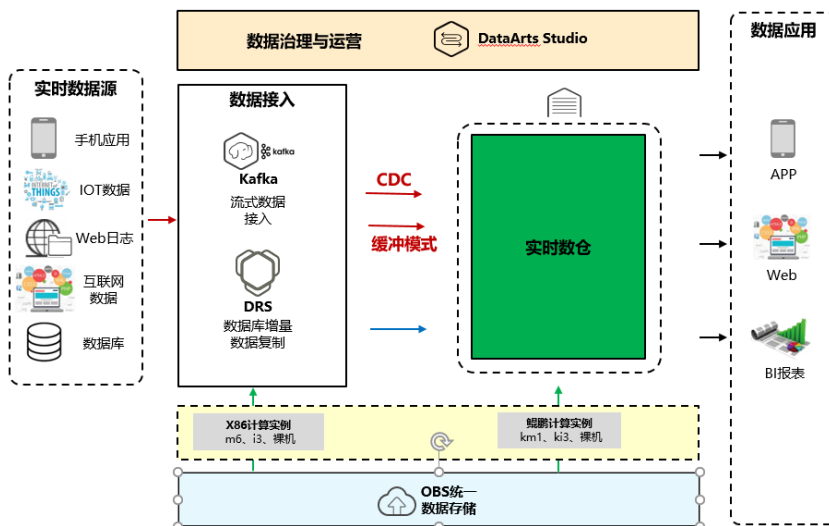


实时写入

提供H-Store存储引擎，微批数据存放在本地，周期合并到OBS存储，支持高吞吐实时写入与更新，同时支持大批量写入场景。

数据实时写入并经实时计算之后，可以用于实时大屏、实时分析、实时监控、实时风控、实时推荐。

实时数据接入，实时分析



1.4 产品功能

DWS为用户提供DWS管理控制台、DWS客户端、REST API等多样化的使用方式。本章节为您介绍DWS的主要功能。

企业级数据仓库和标准 SQL 的支持

当创建好DWS集群后，用户就可以使用SQL客户端工具连接集群，然后执行创建数据库、管理数据库、导入/导出数据以及查询数据的操作。

DWS为用户提供了PB（petabyte）级高性能数据库，主要体现在：

- MPP大规模并行处理框架，支持行列混存、向量化执行，实现万亿数据关联分析秒级响应。
- 内存计算，基于Bloom Filter的Hash Join优化，性能提升2~10倍。
- 支持全对称分布式的Active-Active多节点集群架构，系统无单点故障。
- 基于电信技术优化的大规模集群通信，提升计算节点间的传输效率。
- 基于代价的智能优化器，根据集群规模、数据量、生成最优计划，提高执行效率。

DWS具有完备的SQL能力：

- 兼容ANSI/ISO标准的SQL92、SQL99和SQL 2003语法标准，支持存储过程，支持GBK和UTF-8字符集，支持SQL标准函数与OLAP分析函数。
- 兼容PostgreSQL/Oracle/Teradata/MySQL数据库生态，与主流第三方数据库ETL，BI（business intelligence）工具厂商对接认证。
- 支持位图roaring bitmap数据类型及对应常见函数，在互联网、零售、教育、游戏等行业，提取用户特征、用户画像等场景有广泛应用。
- 在现有范围分区（Range Partitioning）基础上，新增支持list分区方式：PARTITION BY LIST (partition_key,...)，满足用户使用习惯。
- HDFS/OBS外表READ ONLY外表支持JSON文件格式。
- 支持系统表赋权给普通用户，VACUUM能够单独赋权，以及支持可扩展的预定义权限角色功能。
 - 新增表级权限ALTER/DROP/VACUUM。
 - 新增Schema级权限ALTER/DROP。
 - 新增预置角色role_signal_backend、role_read_all_stats。

详细的SQL语法和数据库操作指导，请参见《数据仓库服务数据库开发指南》。

集群管理

一个DWS集群由多个在相同子网中的相同规格的节点组成，共同提供服务。DWS为用户提供了简单易用的Web管理控制台，让用户可以快速申请集群，轻松执行数据仓库管理任务，专注于数据和业务。

集群管理的主要功能如下：

- 创建集群
如果用户需要在云上环境中使用数据仓库服务，首先应创建一个DWS集群。用户可根据业务需求选择相应的产品规格和节点规格快速创建集群。
- 管理快照
快照是DWS集群在某一时间点的完整备份，记录了这一时刻指定集群的所有配置数据和业务数据，快照可用于还原某一时刻的集群。用户可以为集群手动创建快照，也可以开启定时创建自动快照。自动快照有保留天数限制，用户可以对自动快照进行复制，生成手动快照以便长期保留。

当您从快照恢复集群时，系统支持将快照数据恢复到新集群或者原集群中。

对于不再需要的快照，可以选择在控制台删除快照（自动快照不支持手动删除），以释放存储空间。

- 管理节点

用户可查看所创集群的节点列表，在节点管理页面可以清晰的查询各节点的状态、节点规格、是否已使用等信息。若用户需要进行大规模扩容操作时，可通过节点管理功能提前分批次添加准备好用于扩容的节点。例如需要新扩容180个节点，可分3批各添加60个，如果其中有一部分添加失败，可再次添加失败数量的节点，等180个节点添加成功后，再使用这些添加好的节点进行扩容，添加节点过程中不影响集群业务。

- 扩容集群

随着业务的增长，现有集群规模可能无法满足业务需要，此时，用户可以扩容集群，为集群增加计算节点。扩容时业务不中断。进行扩容操作时，用户可根据需求选择是否自动重分布。

- 管理重分布

默认情况下，在扩容之后将自动调起重分布任务，为了增强扩容重分布整个流程的可靠性，可以选择在扩容时关闭自动重分布功能，在扩容成功之后再手动使用重分布功能执行重分布任务，数据重分布后将大大提升业务响应速率。当前重分布支持离线重分布模式，默认情况下，提交重分布任务时将选择离线重分布模式。

- 资源管理

当您有多个数据库用户同时查询作业时，一些复杂查询可能会长时间占用集群资源，从而影响其他查询的性能。例如一组数据库用户不断提交复杂、耗时的查询，而另一组用户经常提交短查询。在这种情况下，短时查询可能不得不在队列中等待耗时查询完成。为了提高效率，DWS提供了资源管理功能，DWS资源管理以资源池为资源承载，对于不同的业务类型可以创建不同的资源池，为这些资源池配置不同的资源占比，并将数据库用户添加至对应的资源池中，以此来限制这些数据库用户的资源使用。

- 重启集群

重启集群将有可能导致正在运行中的业务数据丢失，如果需要执行重启操作，请确定不存在正在运行的业务，所有数据都已经保存。

- 删除集群

当用户不再需要集群时，可选择删除集群。此操作为高危操作，删除集群可能导致数据丢失，请谨慎操作。

DWS为用户提供了以下两种方式管理集群：

- 管理控制台方式

使用管理控制台方式访问DWS集群。用户在注册后，可直接登录管理控制台，选择“数据仓库服务”。

有关集群管理的更多内容，请参见《数据仓库服务用户指南》中的“集群管理”章节。

- REST API方式

支持使用DWS提供的REST API接口以编程的方式管理集群。如果用户需要将DWS集成到第三方系统，用于二次开发，请使用API方式访问。

详细内容，请参见《数据仓库服务API参考》。

多样化的数据导入方式

DWS支持多数据源高效入库，典型的入库方式如下所示。详细指导请参见《数据仓库服务数据库开发指南》中的“迁移数据到DWS”章节。

- 从OBS并行导入数据
- 使用GDS从远端服务器导入数据
- 从MRS导入数据到集群
- 从DWS集群导入数据到新集群
- 使用gsql元命令\COPY导入数据
- 使用COPY FROM STDIN导入数据
- 使用DSC工具迁移SQL脚本
- 使用gs_dump和gs_dumpall命令导出元数据
- 使用gs_restore导入数据

应用程序接口

支持调用JDBC（Java Database Connectivity）、ODBC（Open Database Connectivity）等标准接口开发应用程序访问DWS集群中的数据库。

详细内容请参见《数据仓库服务用户指南》中“使用JDBC和ODBC连接集群”章节。

高可靠性

- 通过实例冗余、数据冗余，实现整个系统无单点故障。
- 数据多副本，且所有数据可手动备份至OBS。
- 自动隔离有故障的节点，利用副本重新恢复数据，并在必要时替换节点。
- 自动快照结合OBS存储，实现Region内容灾，若用户生产集群所处的地理位置发生自然灾害，或者集群内部出现了故障从而导致生产集群无法正常对外提供读写服务，那么灾备集群可以切换为生产集群，从而保障业务连续性。
- 当集群状态为“非均衡”时会出现某些节点主实例增多，从而负载压力较大，用户可在业务低峰期进行集群主备恢复操作。
- 用户使用客户端连接DWS集群时，如果用户仅连接一个CN节点地址，通过该CN节点内网IP或弹性公网IP连接时，只能连接到固定的CN节点上，存在CN单点问题。因此DWS引入了弹性负载均衡服务，解决集群访问的单点问题。弹性负载均衡（ELB）是将访问流量根据转发策略分发到后端多台弹性云服务器器的流量分发控制服务，可以通过流量分发扩展应用系统对外的服务能力，提高应用程序的容错能力。
- 当用户集群创建后，实际需要的CN数量会随着业务需求而发生变化，因此DWS提供了增删CN节点功能，实现用户可以根据实际需求动态调整集群CN数量的要求。

安全管理

- 基于网络隔离、安全组规则以及一系列安全加固项，实现租户隔离和访问权限控制，保护系统和用户的隐私及数据安全。
- 支持SSL安全网络连接、用户权限管理、密码管理等功能，保证数据库在网络层、管理层、应用层和系统层的安全性。

监控与审计

- **监控集群**
DWS与云监控 (Cloud Eye)集成, 使您能够对集群中的计算节点和数据库进行实时监控。详情请参见《数据仓库服务用户指南》中“Cloud Eye监控集群”章节。
- **数据库监控**
数据库监控 (DMS) 是为DWS数据库提供多维度监控服务的系统, 为客户数据库的快速、稳定运行提供保驾护航的能力。该功能对业务数据库使用的磁盘、网络、OS指标数据以及集群运行关键性能指标数据进行收集、监控、分析。通过综合收集到的多种类型指标, 对数据库主机、实例、业务SQL进行诊断, 及时暴露数据库中关键故障及性能问题, 指导客户进行优化解决。详情请参见《数据仓库服务用户指南》中“数据库监控 (DMS)”章节。
- **告警管理**
告警管理包含查看告警规则、告警规则配置与告警信息订阅功能。其中, 告警规则可以提供过去一周的告警信息统计与告警信息明细, 方便用户自行查看租户下的告警。该功能以默认告警规则阈值的方式提供常用的DWS告警监控, 还允许用户根据自己的业务特点, 自定义告警阈值。详情请参见《数据仓库服务用户指南》中“告警管理”章节。
- **事件通知**
DWS与消息通知服务对接, 使您能够订阅事件并查看触发的各类事件。详情请参见《数据仓库服务用户指南》中“事件通知”章节。
- **审计日志**
 - DWS与云审计服务集成, 使您能够对所有的管理控制台操作及API调用进行审计。详情请参见“查看管理控制台关键操作审计日志”章节。
 - DWS数据库还会记录所有的SQL操作, 包括连接尝试、查询和数据库的变动。详情请参见《数据仓库服务用户指南》中“设置数据库审计日志”章节。

支持多种数据库工具

DWS提供了以下几款自研工具, 用户可以在DWS管理控制台下载相关的工具包。有关工具的详细指导请参见《数据仓库服务工具指南》。

- **gsq工具**
它是一款运行在Linux操作系统的命令行SQL客户端工具, 用于连接DWS集群中的数据库, 并对数据库进行操作和维护。
- **Data Studio工具**
它是一款运行在Windows操作系统上的图形界面SQL客户端工具, 用于连接DWS集群中的数据库、管理数据库和数据库对象, 编辑、运行、调试SQL脚本, 查看执行计划等。
- **GDS工具**
它是DWS提供的数据服务工具, 通过和外表机制的配合, 实现数据的高速导入导出。
GDS工具包需要安装在数据源文件所在的服务器上, 数据源文件所在的服务器称为数据服务器, 也称为GDS服务器。
- **DSC SQL语法迁移工具**
DSC (Database Schema Convertor) 是一款运行在Linux或Windows操作系统上的命令行工具, 致力于向客户提供简单、快速、可靠的应用程序SQL脚本迁移服

务，通过内置的语法迁移逻辑解析源数据库应用程序SQL脚本，并迁移为适用于DWS数据库的应用程序SQL脚本。

DSC支持迁移Teradata、Oracle、Netezza、MySQL和DB2数据库的SQL脚本。

- `gs_dump`和`gs_dumpall`

`gs_dump`支持导出单个数据库或其内的对象，而`gs_dumpall`支持导出集群中所有数据库或各库的公共全局对象。

通过导入工具将导出的元数据信息导入至需要的数据库，可以完成数据库信息的迁移。

- `gs_restore`

在数据库迁移场景下，支持使用`gs_restore`工具将事先使用`gs_dump`工具导出的文件格式，重新导入DWS集群，实现表定义、数据库对象定义等元数据的导入。

1.5 基本概念

DWS 的管理概念

- 集群

表示由多个节点组成的服务器群组。DWS以集群为单位进行组织，一个DWS集群由多个在相同子网中的相同规格的节点组成，共同提供服务。

- 节点

每个DWS集群部署节点取值范围为3~256，每个节点都支持存储与分析数据。

- 规格

用户创建DWS集群时需要指定集群节点的规格，不同规格节点使用的CPU、内存和存储资源不同。

- 快照

快照是用户根据需要创建的DWS集群备份，用于还原创建快照时的集群数据。快照会一直保存，直到用户使用管理控制台将此快照删除（自动快照不支持手动删除）。快照会占用用户使用的OBS配额。

- Project

Project用于将OpenStack的资源（计算资源、存储资源和网络资源）进行分组和隔离。Project可以是一个部门或者一个项目组。一个账户中可以创建多个Project。

DWS 的数据库概念

- 数据库

数据库用于管理各类数据对象，与其他数据库隔离。创建数据库时可以指定对应的表空间，如果不指定相应的表空间，相关的对象会默认保存在PG_DEFAULT空间中。数据库管理的对象可分布在多个表空间上。

- 联机分析处理

联机分析处理（Online Analytical Processing，简称OLAP），是DWS集群的主要工作方式，支持复杂的分析操作，侧重决策支持，并且提供直观易懂的查询结果。

- 大规模并行处理

大规模并行处理（Massive Parallel Processing，简称MPP，也可称为无共享架构），是指在DWS集群的每个节点中，都有独立的内存计算和磁盘存储系统，

DWS根据数据库模型和应用特点，将业务数据划分到不同节点上，通过节点间的网络彼此连接协同计算，使集群作为整体提供数据库服务以满足业务需要。

- 无共享架构

无共享架构（Shared Nothing Architecture）是一种分布式计算架构。这种架构中的每一个节点都是独立、自给的，使整个系统中不出现单点竞争的情况，以提高工作效率。

- 数据库版本

每个DWS集群对应一个数据库版本，用户在创建DWS集群时确认此集群的版本。

- 数据库连接

用户通过客户端连接DWS集群的方式，实现数据库的连接。支持在华为云平台以及互联网环境上使用客户端进行连接。

- 数据库用户和角色

DWS使用用户和角色来控制对数据库的访问。根据角色自身的设置不同，一个角色可以看做是一个数据库用户，或者一组数据库用户。在DWS中角色和用户之间的区别只在于角色默认是没有LOGIN权限的。在DWS中一个用户唯一对应一个角色，不过可以使用角色叠加来更灵活地进行管理。

- 实例

实例在DWS中是运行在内存中的一组数据库进程，一个实例可以管理一个或多个数据库，这些数据库组成一个集簇。集簇是存储磁盘上的一个区域，这个区域在安装时初始化并由一个目录组成，所有数据都存储在这个目录中，这个目录被称为数据目录，使用initdb创建。理论上来说一个服务器上可以在不同的端口启动多个实例，但是DWS一次只能管理一个实例，启动和停止都是依赖于具体的数据目录。以后由于兼容的需要不排除引入实例名这个概念的可能。

- 表空间

在DWS中，表空间是一个目录，可以存在多个，里面存储的是它所包含的数据库的各种物理文件。由于表空间是一个目录，仅是起到了物理隔离的作用，其管理功能依赖于文件系统。

- 模式

DWS的模式是对数据库做一个逻辑分割。所有的数据库对象都建立在模式下面。DWS的模式和用户是弱绑定的，所谓的弱绑定是指虽然创建用户的同时会自动创建一个同名模式，但用户也可以单独创建模式，并且为用户指定其他的模式。

- V2表

V2表特指建表环节时，执行CREATE TABLE语法中定义的colversion取值为2.0的表，表示列存表的每列合并存储在一个文件中，文件名以relfilenode.C1.0命名，数据存储在本地盘。存算一体场景下，不指定colversion取值时，用户创建的列存表默认为V2表。

- V3表

V3表特指建表环节时，执行CREATE TABLE语法中定义的colversion取值为3.0的表，即存算分离表，表示列存表的每列合并存储在一个文件中，文件名以C1_field.0命名，数据存储在OBS文件系统。存算分离场景下，不指定colversion取值时，用户创建的列存表默认为V3表。

- 事务管理

在事务管理上，DWS采取了MVCC（多版本并发控制）结合两阶段锁的方式，其特点是读写之间不阻塞。DWS的MVCC没有将历史版本数据统一存放，而是和当前元组的版本放在了一起。DWS没有回滚段的概念，但是为了定期清除历史版本数据DWS引入了一个VACUUM进程。一般情况下用户不用关注它，除非要做性能调优。此外，DWS是自动提交事务。

1.6 与其他云服务的关系

与弹性云服务器的关系

数据仓库服务使用弹性云服务器（Elastic Cloud Server，简称ECS）作为集群的节点，每个弹性云服务器是集群中的一个节点。

与虚拟私有云的关系

数据仓库服务使用虚拟私有云（Virtual Private Cloud，简称VPC）为集群提供网络拓扑，实现多个不同集群互相隔离并控制访问。

与对象存储服务的关系

数据仓库服务使用对象存储服务（Object Storage Service，简称OBS）作为集群数据与外部数据互相转化的一个方法，实现安全、高可靠和低成本存储需求。

与 MapReduce 服务的关系

数据仓库服务使用MapReduce服务（MapReduce Service，简称MRS）将数据从MRS迁移到DWS集群，实现海量数据通过Hadoop处理后使用DWS进行分析查询。

与云数据迁移的关系

您可使用云数据迁移（Cloud Data Migration，简称CDM）实现多种数据源数据到DWS的迁移。

与云监控的关系

数据仓库服务使用云监控（Cloud Eye Service，简称CES）监控集群中的多项性能指标，从而集中高效地呈现状态信息。云监控支持发送自定义告警，用户可以即时获取异常通知。

与云审计服务的关系

数据仓库服务使用云审计服务（Cloud Trace Service，简称CTS）审计用户在管理控制台页面的非查询操作，可用于检视是否存在非法或越权操作，完善服务安全管理。

与云日志服务的关系

数据仓库服务用户可以在云日志服务（Log Tank Service，简称LTS）查看采集的集群日志或进行日志转储。

与消息通知服务的关系

数据仓库服务使用消息通知服务（Simple Message Notification，简称SMN）依据用户的事件订阅需求主动推送通知消息，使用户可以在事件发生（如集群关键操作）时能立即接收到通知。

与云解析服务的关系

数据仓库服务使用云解析服务（Domain Name Service，DNS）可以提供域名映射到DWS集群的IP地址。

与弹性负载均衡的关系

数据仓库服务利用弹性负载均衡（Elastic Load Balance，简称ELB）健康检查机制可将集群的CN请求快速转发到正常工作的CN节点上，当有CN故障节点时，也能第一时间切换流量到健康节点上，最大化降低集群访问故障。

1.7 DWS 权限管理

如果您需要对华为云上的DWS资源，为企业中的员工设置不同的访问权限，以达到不同员工之间的权限隔离，您可以使用统一身份认证服务（Identity and Access Management，简称IAM）进行精细的权限管理。该服务提供用户身份认证、权限分配、访问控制等功能，可以帮助您安全的控制华为云资源的访问。

通过IAM，您可以在华为云账号中给员工创建IAM用户，并授权控制他们对华为云资源的访问范围。例如您的员工中有负责软件开发的人员，您希望他们拥有DWS的使用权限，但是不希望他们拥有删除集群等高危操作的权限，那么您可以使用IAM为开发人员创建用户，通过授予仅能使用DWS，但是不允许删除集群的权限，控制他们对DWS资源的使用范围。

如果华为云账号已经能满足您的要求，不需要创建独立的IAM用户进行权限管理，您可以跳过本章节，不影响您使用DWS服务的其它功能。

IAM是华为云提供权限管理的基础服务，无需付费即可使用，您只需要为您账号中的资源进行付费。关于IAM的详细介绍，请参见《IAM用户指南》中的“产品介绍”。

DWS 权限

默认情况下，IAM管理员创建的IAM用户没有任何权限，需要将其加入用户组，并给用户组授予策略或角色，才能使得用户组中的用户获得对应的权限，这一过程称为授权。授权后，用户就可以基于被授予的权限对云服务进行操作。

DWS部署时通过物理区域划分，为项目级服务。授权时，“作用范围”需要选择“区域级项目”，然后在指定区域对应的项目中设置相关权限，并且该权限仅对此项目生效；如果在“所有项目”中设置权限，则该权限在所有区域项目中都生效。访问DWS时，需要先切换至授权区域。

- **角色：** IAM最初提供的一种根据用户的工作职能定义权限的粗粒度授权机制。该机制以服务为粒度，提供有限的服务相关角色用于授权。由于华为云各服务之间存在业务依赖关系，因此给用户授予角色时，可能需要一并授予依赖的其他角色，才能正确完成业务。角色并不能满足用户对精细化授权的要求，无法完全达到企业对权限最小化的安全管控要求。
- **策略：** IAM最新提供的一种细粒度授权的能力，可以精确到具体服务的操作、资源以及请求条件等。基于策略的授权是一种更加灵活的授权方式，能够满足企业对权限最小化的安全管控要求。例如：针对DWS服务，IAM管理员能够控制IAM用户仅能对某一类资源进行指定的管理操作。

如表1-2所示，包括了DWS的所有系统权限。

表 1-2 DWS 系统权限

系统角色/策略名称	描述	类别	依赖关系
DWS ReadOnlyAccess	数据仓库服务只读权限，拥有该权限的用户仅能查看数据仓库服务资源。	系统策略	无
DWS FullAccess	数据仓库服务数据库管理员权限，拥有数据仓库服务所有权限。	系统策略	无
DWS Administrator	DWS数据库管理员权限，拥有对数据仓库服务的所有执行权限。 - 拥有VPC Administrator权限的DWS用户可以创建VPC或子网。 - 拥有云监控Administrator权限的DWS用户，可以查看DWS集群的监控指标信息。 - 使用此权限场景下，如果使用到创建委托场景，还需配置对应Security Administrator权限。	系统角色	依赖Tenant Guest和Server Administrator策略，需要在同项目中勾选依赖的策略。
DWS Database Access	DWS数据库访问权限，拥有该权限的用户，可以基于IAM用户生成临时数据库用户凭证以连接DWS集群数据库。	系统角色	依赖DWS Administrator，需要在同项目中勾选依赖的策略。

表1-3列出了DWS常用操作与系统权限的授权关系，您可以参照该表选择合适的系统策略。

说明

- 每个区域的每个项目首次使用弹性IP绑定功能时，系统将提示创建名称为“DWSAccessVPC”委托以授权DWS访问VPC。授权成功后，DWS可以在绑定弹性IP的虚拟机故障时切换至健康虚拟机。
- 在实际业务中，除了具备策略权限外还需要给不同角色的用户授予不同的资源操作权限。例如创建快照、重启集群等操作，详情请参见《数据仓库服务用户指南》中“策略语法：细粒度策略”章节。
- 默认情况下，只有华为云账号或拥有Security Administrator权限的用户才具备查询委托和创建委托的权限。账号中的IAM用户，默认没有查询委托和创建委托的权限，在使用弹性IP绑定功能时页面会屏蔽绑定按钮，此时需联系有“DWS Administrator”权限的用户在当前页面完成对DWS的委托授权。

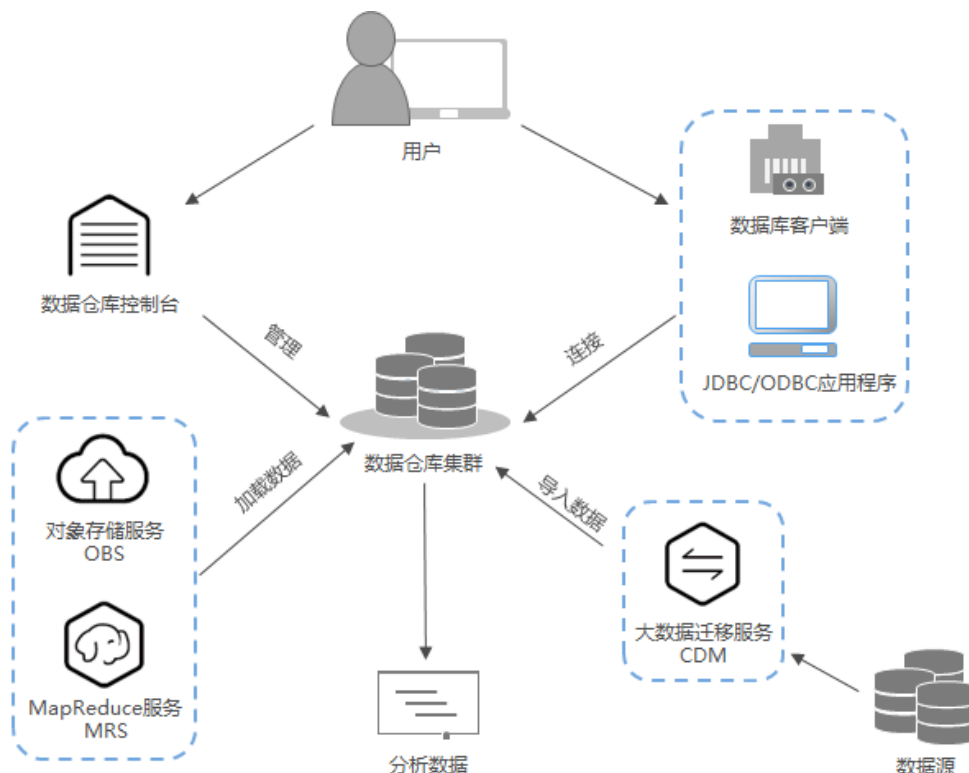
表 1-3 DWS 常用操作与系统权限的关系

操作	DWS FullAccess	DWS ReadOnlyAccess	DWS Administrator	DWS Database Access
创建/恢复集群	√	x	√	x
获取集群列表	√	√	√	x
获取单个集群详情	√	√	√	x
设置自动快照	√	x	√	x
设置安全参数/参数组	√	x	√	x
重启集群	√	x	√	x
扩容集群	√	x	√	x
重置密码	√	x	√	x
删除集群	√	x	√	x
设置可维护时间段	√	x	√	x
绑定EIP	x	x	√	x
解绑EIP	x	x	√	x
创建DNS域名	√	x	√	x
释放DNS域名	√	x	√	x
修改DNS域名	√	x	√	x
创建MRS连接	√	x	√	x
更新MRS连接	√	x	√	x
删除MRS连接	√	x	√	x
添加/删除标签	√	x	√	x
编辑标签	√	x	√	x
创建快照	√	x	√	x
获取快照列表	√	√	√	√
删除快照	√	x	√	x
复制快照	√	x	√	x

1.8 如何访问 DWS

数据仓库服务的基本使用流程如下图所示：

图 1-6 数据仓库服务基本使用流程



访问集群

DWS提供了Web化的服务管理平台，即管理控制台，还提供了基于HTTPS请求的API（Application programming interface）方式管理DWS集群。

说明

集群部署时，如果单个节点故障，访问DWS时会自动屏蔽异常节点，但会影响集群性能。

访问集群数据库

DWS支持如下方式访问集群数据库：

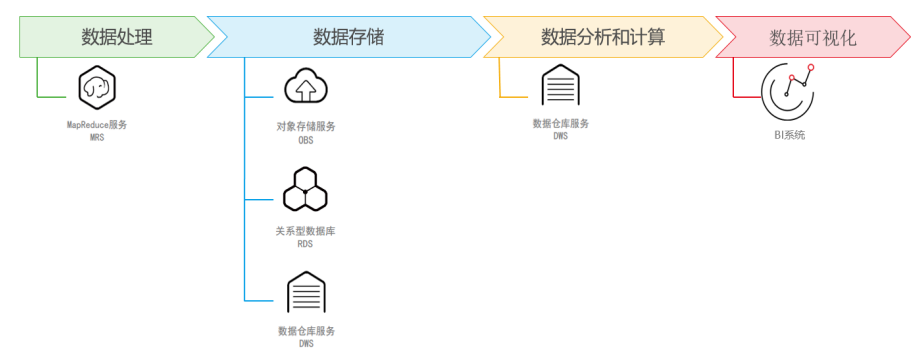
- DWS客户端方式
用户使用DWS客户端访问集群数据库，请参见《数据仓库服务用户指南》中“连接集群”章节。
- JDBC、ODBC等接口调用方式
支持调用JDBC（Java database connectivity）、ODBC（Open Database Connectivity）等标准接口开发应用程序访问DWS集群中的数据库。
详细内容请参见《数据仓库服务用户指南》中“使用JDBC和ODBC连接集群”章节。

端到端数据分析流程

数据仓库服务与华为云平台其他一系列服务已无缝集成，以使用户能够快速部署端到端的数据分析和解决方案。

端到端数据分析流程如下图所示，该流程展示了每个过程可以使用的云服务。

图 1-7 端到端数据分析流程



1.9 使用限制

本文介绍使用DWS产品重点功能限制以及注意事项。

创建DWS集群后，您不需要做数据库的基础运维（例如高可用、打安全补丁等），但需要重点关注如下事项。

规格与性能限制

表 1-4 规格与性能限制

资源类型	限制	说明
多AZ可用分区	3	- 只有当前选择的区域可用分区数量>=3时会展示“多AZ”选项，不满足该条件只能创建单AZ集群。 - 多AZ集群目前只支持同时选择3个AZ，主机节点均匀分配在选择的3个AZ中。 - 多AZ集群节点数量必须满足3的倍数。 - 多AZ集群只支持单节点DN数量<=2的规格。

资源类型	限制	说明
存储空间	- 对于SSD云盘，如果您想提高存储空间扩容上限，请联系客服申请。 - SSD本地盘存储空间范围与所选数据仓库规格有关。	SSD云盘：高性价比的云硬盘，适合中等性能诉求的企业应用。 SSD本地盘：SSD本地盘不支持磁盘扩容。
CN部署量	取值范围为3~集群节点数，最大值为20，默认值为3。	在大规模集群下，建议部署多个CN。
最大连接数	- CN连接数：100 ~ 262,143 - DN连接数：100 ~ 262,143	CN及DN概述详情请参见 集群逻辑架构 。

配额限制

表 1-5 配额限制

资源类型	限制	说明
节点数	256	设置的节点数量不能大于用户可使用的节点配额或256个。如果可使用的节点配额不足，用户可以单击“申请扩大配额”以提工单的形式申请更多节点配额。
标签	每个集群最多可以添加20个标签。	更多信息请参见《数据仓库服务用户指南》中的“标签简介”章节。
自动备份保留天数	可设置为1~31天，默认为7天	系统会在保留期结束时删除到期快照。

命名限制

表 1-6 命名限制

限制项	说明
集群名称	集群名称长度为4到64个字符，且不超过64个字节，1个中文字符为3个字节，其他字符为1个字节，不区分大小写，必须以字母或中文开头，可以包含中文、字母、数字、中划线或者下划线，不能包含其他的特殊字符。

限制项	说明
管理员用户	- 只能由小写字母、数字或下划线组成。 - 必须以小写字母或下划线开头。 - 长度为6~64个字符。 - 用户名不能为DWS数据库的关键字。DWS数据库的关键字，具体请参见《数据仓库服务数据库开发指南》中“SQL参考 > 关键字”章节。
角色名称	以字母开头，可以包含字母、数字、下划线，长度不超过63个字符。
用户名称	以字母开头，可以包含字母、数字、下划线，长度不超过63个字符。
快照名称	快照名称长度为4~64个字符，必须以字母开头，不区分大小写，可以包含字母、数字、中划线或者下划线，不能包含其他的特殊字符。
快照策略名称	策略名称要求在4位到92位之间，必须以字母开头，不区分大小写，可以包含字母、数字、中划线或者下划线，不能包含其他特殊字符，并且名称唯一。
告警规则名称	规则名称长度为6到64个字符，且只能是首位非数字的中文、英文、数字、斜线组合。
容灾名称	容灾名称在4位到64位之间，不区分大小写，必须以字母开头，可以包含字母、数字、中划线或者下划线，不能包含其他的特殊字符。

集群基础操作限制

表 1-7 集群基础操作限制

限制项	说明
绑定负载均衡	- 绑定ELB时，需创建和DWS集群相同的区域、VPC以及企业项目的ELB进行绑定。 - DWS的ELB功能仅支持独享型ELB规格，不支持共享型ELB绑定操作。 - 解绑ELB时，DWS侧会清除ELB中集群相关信息，但并不会删除用户ELB。
增删CN节点	- 资源发放时默认CN数量为3，用户可以根据实际发放节点数量调整CN数量，范围为2~20。 - 删除CN节点过程中，如果只有一个节点异常，只能删除这个异常的节点；如果有两个及以上节点异常，不能删除CN节点，需要先处理恢复异常的节点后，才能删除CN节点。

限制项	说明
资源负载管理	离线扩容过程中不支持资源管理功能，如开启了资源管理计划，请在进行离线扩容操作之前停止。
逻辑集群管理	在实际业务场景中，建议用户尽可能将同一个数据库的表创建到同一个逻辑集群中。
重启集群	重启期间集群将无法提供服务。因此，在重启前，请确定集群中没有正在运行的任务，并且所有数据都已经保存。 如果集群正在处理业务数据，如导入数据、查询数据、创建快照或恢复快照时，一旦重启集群，有可能导致文件损坏或重启失败。因此，建议停止所有集群任务后，再重启集群。
启动/停止集群	集群停止后实例关机，ECS场景下基础资源（vCPU、内存）不再保留，当再次启动云服务时，可能由于资源不足无法正常开机，请耐心等待，稍后再试。

集群运维操作限制

表 1-8 集群运维操作限制

限制项	说明
扩容集群	- 扩容期间集群会有闪断，请谨慎操作。 - 正在扩容的集群将禁用重启集群、停止集群、启动集群、规格变更、增删CN、扩容集群、缩容集群、创建快照、重置数据库管理员密码等功能。 - 只有在扩容之后，集群任务信息为“待重分布”状态时才能手动使用“重分布”功能，其他时段该功能不可使用。
缩容集群	正在缩容的集群禁用重启集群、扩容集群、创建快照、节点管理、智能运维、资源管理、参数修改、安全设置、日志服务、重置数据库管理员密码和删除集群的功能。
弹性规格变更	- 弹性规格变更仅支持ECS+EVS云盘的存算一体集群，ECS本地盘的集群不支持。 - 弹性变更规格需要停止虚拟机，仅支持离线变更，变更时间大约需要5~10分钟。
经典规格变更	- 经典变更规格最大支持调整到240节点，且原节点数量加调整后节点数量不大于480节点。 - 逻辑集群不支持经典变更规格。

限制项	说明
备份恢复集群	- 对于有数据可靠性要求的用户，尤其是针对业务方无法通过上游重新导入进行数据恢复的场景，集群需要备份操作，避免人为或其他因素造成数据丢失后无法恢复。 - 正在创建快照的集群，暂时无法提供完整服务，如无法执行重启、扩容、重置密码、修改集群配置信息等操作。
升级集群	8.1.3以前集群版本下发升级任务后，完成集群升级前不支持回滚或提交操作。 - 集群安装热补丁后无法建立容灾。
容灾管理	- 灾备集群在容灾任务停止或者异常但灾备集群正常的情况下，可以提供读服务，灾备切换成功后可以提供读写服务。 - 容灾创建后，生产集群快照功能正常使用，但是灾备集群禁用快照功能，以及生产、灾备集群均禁用恢复功能。 - 不支持逻辑集群和资源池。
日志管理	- 非OBS场景下，此功能不可用。 - 经典变更规格、增删CN等变更CN场景会导致数据丢失，建议在执行变更期间关闭审计日志转储。

1.10 DWS 技术指标

DWS不同版本的技术指标如下所示。

表 1-9 DWS 技术指标 8.1.3-9.1.0 版本

技术指标	8.1.3最大值	8.2.0最大值	8.2.1最大值	8.3.0最大值	9.1.0最大值
集群节点数	2048	2048	2048	2048	- 存算一体：2048 - 存算分离：采用多VW技术，最大支持256个VW，每个VW 1024个DN，建议VW不超过32个，每个VW不超过128个DN，所有VW的DN节点总数不超过2048。
并发连接数	分钟级复杂查询并发数：80 秒级短查询并发数：500 毫秒级短事务并发数：5000	分钟级复杂查询并发数：80 秒级短查询并发数：500 毫秒级短事务并发数：5000	分钟级复杂查询并发数：80 秒级短查询并发数：500 毫秒级短事务并发数：5000	分钟级复杂查询并发数：80 秒级短查询并发数：500 毫秒级短事务并发数：5000	- 存算一体：分钟级复杂查询并发数：80 秒级短查询并发数：500 毫秒级短事务并发数：5000 - 存算分离：多VW技术，可以实现并发数目的叠加，随着VW个数增加，支撑的并发数也可以相应的扩展，集群总的并发数目受GTM/CCN排队影响，建议不超过8192。

技术指标	8.1.3最大值	8.2.0最大值	8.2.1最大值	8.3.0最大值	9.1.0最大值
集群数据容量	20PB	20PB	20PB	20PB	- 存算一体：20PB - 存算分离：数据存储在 OBS 上，理论上容量可以无限扩展。
单表大小	1PB	1PB	1PB	1PB	1PB
单行数据大小	1GB	1GB	1GB	1GB	1GB
单表列数（除Hudi表外）	1600	1600	1600	1600	- 行存：1600 - 列存：1600 - HStore：1600
Hudi表列数	-	-	5000	5000	5000
分区表的分区个数	32768	32768	32768	32768	最大32768，建议不超过1000
集群内单点故障RTO	60s	60s	60s	60s	60s
集群内单点故障RPO	0	0	0	0	0
集群故障容灾切换RTO	60min	60min	60min	60min	60min
集群故障容灾切换RPO	60min	60min	60min	60min	60min

 说明

VW(Virtual Warehouse)：也称计算组，DWS存算分离支持把一个物理集群拆分成多个VW，不同业务可以绑定不同VW，实现业务负载隔离和并发数目扩展。

表 1-10 DWS 技术指标 8.0.x-8.1.1 版本

技术指标	8.0.x最大值	8.1.0最大值	8.1.1最大值
数据容量	10PB	10PB	20PB
集群节点数	256	256	2048
单表大小	1PB	1PB	1PB
单行数据大小	1GB	1GB	1GB
每条记录单个字段的大小	1GB	1GB	1GB
单表记录数	2 ⁵⁵	2 ⁵⁵	2 ⁵⁵
单表列数	1600	1600	1600
单表中的索引个数	无限制	无限制	无限制
单表索引包含列数	32	32	32
单表约束个数	无限制	无限制	无限制
并发连接数	分钟级复杂查询并发数：60 毫秒级短事务并发数：5000	分钟级复杂查询并发数：60 毫秒级短事务并发数：5000	分钟级复杂查询并发数：80 毫秒级短事务并发数：5000
分区表的分区个数	32768	32768	32768
分区表的单个分区大小	1PB	1PB	1PB
分区表的单个分区记录数	2 ⁵⁵	2 ⁵⁵	2 ⁵⁵

 说明

并发连接数的最大值基于数仓规格48 vCPU和64 vCPU。例如，存算一体云盘规格 dwsk.12xlarge（48 vCPU | 384GB | 24000GB SSD）或dwsx2.16xlarge.m7（64 vCPU | 512GB | 32000GB SSD）。

2 入门

2.1 第 1 步：入门前准备

本指南是一个入门教程，向您演示如何创建示例DWS集群，连接示例DWS集群数据库、导入存储在OBS中的示例数据和分析示例数据的流程。您可以使用该入门教程评估DWS服务。

在开始创建DWS集群之前，请确保您已完成如下前提条件：

- **确定集群端口**

确定集群端口

- 在创建DWS集群时需要指定一个端口供SQL客户端或应用程序通过该端口访问集群。
- 如果您的客户端机器位于防火墙之后，则您需要有一个可用的开放端口，这样才能从SQL客户端工具连接到集群并进行查询分析。
- 如果您不了解可用的开放端口，则请联系网络管理员，在您的防火墙中确定一个开放端口。DWS支持的端口范围为8000 ~ 30000。
- 在集群创建之后无法更改集群的端口号，请务必确保在集群创建过程中指定的端口为可用的开放端口。

2.2 第 2 步：创建集群

在使用DWS执行数据分析任务前，您首先要创建一个集群，一个DWS集群由多个在相同子网中的节点组成，共同提供服务。请参考以下指导创建集群。

创建集群

- 步骤1** 登录DWS管理控制台。
- 步骤2** 单击左侧导航栏的“专属集群 > 集群列表”。
- 步骤3** 在“专属集群”页面，单击右上角“创建数据仓库集群”。
- 步骤4** 选择待创建的集群所属的区域。

- **区域**：选择集群所属的工作区域。
- **可用分区**：默认即可。

步骤5 选择主机规格。

- **产品类型**：根据客户需求选择，例如“**存算一体**”类型。
- **CPU架构**：根据客户需求选择，例如“**X86**”架构。
- **节点规格**：默认即可。
- **节点数量**：默认即可，至少3个。

步骤6 填写集群配置参数。

- **集群名称**：输入“dws-demo”。
- **集群版本**：显示为当前集群版本，暂不支持修改。
- **默认数据库**：显示为“gaussdb”。暂不支持修改。
- **管理员用户**：默认为“dbadmin”，使用默认值即可。集群创建成功后，客户端连接集群数据库时将使用该数据库管理员用户及其密码。
- **管理员密码**：输入密码。
- **确认密码**：重复输入一次数据库管理员密码。
- **数据库端口**：默认即可。客户端或应用程序将通过该端口连接集群中的数据库。

图 2-1 集群配置

集群名称	dws-demo	?
集群版本		
默认数据库	postgres	
管理员用户	dbadmin	?
管理员密码		
确认密码		
数据库端口	8000	?

步骤7 配置网络参数。

- **虚拟私有云**：可以在下拉框中选择已有的虚拟私有云，如果未配置过虚拟私有云，可以单击“查看虚拟私有云”进入虚拟私有云管理控制台，新创建一个虚拟私有云例如“vpc-dws”。然后回到DWS管理控制台的创建集群页面，单击“虚拟私有云”下拉框旁边的进行刷新，再选择新创建虚拟私有云。
- **子网**：创建虚拟私有云时会默认创建一个子网，您可以选择对应的子网名。

- **安全组**：选择“自动创建安全组”。

自动创建的安全组，将被命名为“DWS-**<集群名称>**-**<DWS集群的数据库端口>**”，出方向允许所有访问，入方向只开放“数据库端口”以允许来自客户端或应用程序的访问。

如果您选择的是自定义创建的安全组，则需要在该安全组中添加一条入方向的规则，向访问DWS的客户端主机开放DWS集群的“数据库端口”，如表2-1所示。添加入规则的具体操作请参见《虚拟私有云用户指南》中的“安全性 > 安全组 > 添加安全组规则”章节。

表 2-1 安全组入规则配置样例

参数名	样例值
协议/应用	TCP
端口	8000 说明 输入创建DWS集群时设置的“数据库端口”，这个端口是DWS用于接收客户端连接的端口，默认为8000。
源地址	选择“IP地址”，输入访问DWS的客户端主机的IP地址和子网掩码，例如“192.168.0.10/16”。

- **公网访问**：选择“自动分配”为集群申请一个弹性IP作为集群公网IP。并且，在“带宽”参数中设置弹性IP的带宽。

步骤8 配置集群所属的“企业项目”。已开通企业项目管理服务的用户才可以配置该参数。默认值为default。

企业项目是一种云资源管理方式，企业项目管理服务提供统一的云资源按项目管理，以及项目内的资源管理、成员管理。

您可以选择默认的企业项目“default”或其他已有的企业项目。如果要创建新的企业项目，请登录企业管理控制台进行创建，详细操作请参考《企业管理用户指南》。

步骤9 高级配置，在本示例中，选择“默认配置”即可。

- **默认配置**：表示以下几项高级配置使用系统默认的配置。
 - CN部署量：CN即协调节点，默认部署3个CN节点。
 - 标签：默认未给集群添加标签。
 - 加密数据库：默认为关闭，表示不对数据库进行加密。
- **自定义**：选择该选项时页面上将显示CN部署量、标签、加密数据库这几项高级配置，需要用户进行自定义设置。

步骤10 单击“立即创建”，进入“规格详情”页面。

步骤11 单击“提交”。

提交成功后开始创建。单击“返回集群列表”返回“专属集群”页面，所创集群的初始状态为“创建中”，集群创建需要时间，请等待一段时间。创建成功后状态更新为“可用”，用户可以开始使用集群。

----结束

2.3 第 3 步：连接集群

操作场景

您在创建好数据仓库集群，开始使用数据库服务前，需要使用数据库客户端连接到 DWS 集群中的数据库。本示例将使用 Data Studio 客户端工具通过公网地址连接 DWS 集群中的数据库。您也可以使用其他 SQL 客户端连接集群，更多连接方式请参见[连接 DWS 集群方式介绍](#)。

1. 获取所要连接的数据库名称、用户名和密码。
首次使用客户端连接集群时，您需使用[第2步：创建集群](#)时设置的数据库管理员用户和密码连接到默认数据库“gaussdb”。
2. [获取集群公网访问地址](#)：通过集群公网访问地址连接数据库。
3. [使用 Data Studio 连接到集群数据库](#)：下载配置 Data Studio 客户端并连接集群数据库。

获取集群公网访问地址

步骤1 登录 DWS 管理控制台。

步骤2 在左侧导航栏中，单击“专属集群 > 集群列表”。

步骤3 在集群列表中，选中已创建集群（如 dws-demo），单击“集群名称”前面的向下展开按钮 ，获取并保存公网访问地址。

该公网访问地址将在[使用 Data Studio 连接到集群数据库](#)时使用。

----结束

使用 Data Studio 连接到集群数据库

步骤1 DWS 提供了基于 Windows 平台的 Data Studio 图形界面客户端，该工具依赖 JDK，请先在客户端主机上安装 Java 1.8.0_141 或以上版本的 JDK。

在 Windows 操作系统中，您可以访问 JDK 官方网站，下载符合操作系统版本的 JDK，并根据指导进行安装。

步骤2 登录 DWS 管理控制台。

步骤3 单击“连接客户端”。

步骤4 在“下载客户端和驱动”页面，下载“Data Studio 图形界面客户端”。

- 请根据操作系统类型，选择“Windows x86”或“Windows x64”，再单击“下载”，可以下载与现有集群版本匹配的 Data Studio 工具。

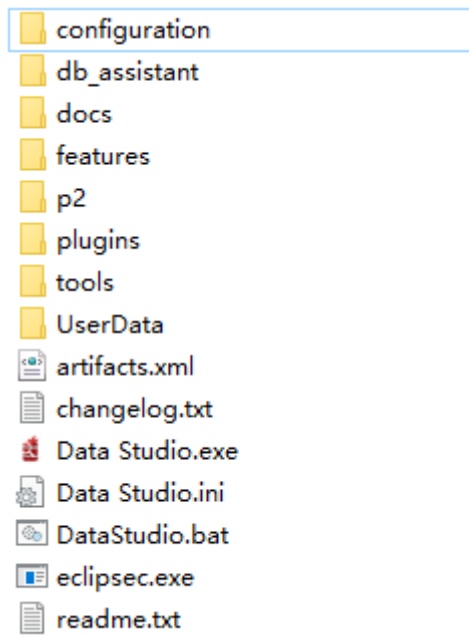
如果同时拥有不同版本的集群，单击“下载”时会下载与集群最低版本相对应的 Data Studio 工具。如果当前没有集群，单击“下载”时将下载到低版本的 Data Studio 工具。DWS 集群可向下兼容低版本的 Data Studio 工具。

- 单击“历史版本”可根据集群版本下载相应版本的 Data Studio 工具，建议按集群版本下载配套的工具。

步骤5 解压下载的客户端软件包（32 位或 64 位）到需要安装的路径。

步骤6 打开安装目录，双击Data Studio.exe，启动Data Studio客户端，如图2-2所示。

图 2-2 启动客户端



步骤7 在主菜单中选择“文件>新建连接”，如图2-3所示。

图 2-3 新建连接



步骤8 在弹出的“新建/选择数据库连接”页面中，如下图所示，输入连接参数。

表 2-2 配置连接参数

字段名称	说明	举例
数据库类型	选择“GaussDB A”	GaussDB A
名称	连接名称。	dws-demo
主机名	所要连接的集群IP地址（IPv4）或域名。	-
端口号	数据库端口。	8000
数据库	数据库名称。	gaussdb
用户名	所要连接数据库的用户名。	-
密码	所要连接数据库的登录密码。	-

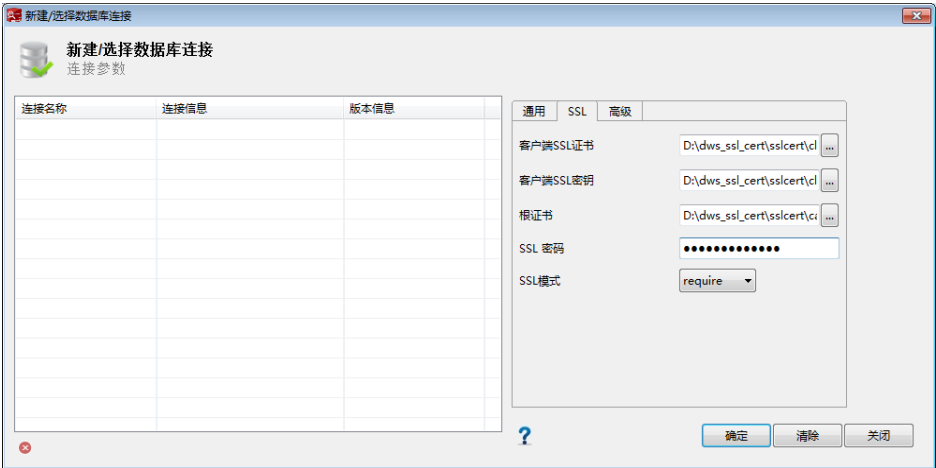
字段名称	说明	举例
保存密码	在下拉列表中选择： “仅当前会话”：仅在当前会话中保存密码。 “不保存”：不保存密码。	-
启用SSL	启用时，客户端将使用SSL加密连接方式。SSL连接方式安全性高于普通模式，建议开启。	-

当“启用SSL”设置为开启时，请先参见[7.3.4-2 下载SSL证书](#)下载SSL证书，并解压证书文件。然后单击“SSL”页签，设置如下参数：

表 2-3 配置 SSL 参数

字段名称	说明
客户端SSL证书	选择SSL证书解压目录下的“sslcert\client.crt”文件。
客户端SSL密钥	客户端SSL密钥只支持PK8格式，请选择SSL证书解压目录下的“sslcert\client.key.pk8”文件。
根证书	当“SSL模式”设为“verify-ca”时，必须设置根证书，请选择SSL证书解压目录下的“sslcert\cacert.pem”文件。
SSL密码	客户端pk8格式SSL密钥密码。
SSL模式	DWS支持的SSL模式有： - require - verify-ca DWS不支持“verify-full”模式。

图 2-4 配置 SSL 参数

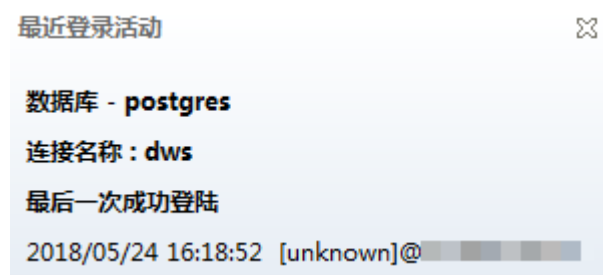


步骤9 单击“确定”建立数据库连接。

如果启用了SSL，在弹出的“连接安全告警”提示对话框中单击“继续”。

登录成功后，将弹出“最近登录活动”提示框，表示Data Studio已经连接到数据库。用户即可在Data Studio界面的“SQL终端”窗口中执行SQL语句。

图 2-5 登录成功



欲详细了解Data Studio其他功能的使用方法，请按“F1”查看Data Studio用户手册。

----结束

2.4 第 4 步：查看其它资料并清理资源

查看其它资料

完成如上操作步骤后，推荐您可以参考如下资料继续对数据仓库服务进行更详细深入的了解：

- 《数据仓库服务用户指南》：本指南在此入门的基础上，对创建、管理、监控以及连接集群的概念和相关操作提供全面详细的信息。
- 《数据仓库服务数据库开发指南》：本指南在此入门的基础上，为数据库开发人员提供全面详细的信息，帮助他们了解如何构建、管理和查询DWS数据库，包括SQL语法、用户管理、数据导入导出等指导。

清理资源

当完成快速入门的样例后，如果您不再需要使用本样例创建的样例数据、集群、ECS以及VPC时，您可以删除这些资源，以免资源浪费或占用您的配额。

步骤1 删除DWS集群。

在DWS管理控制台，选择“专属集群 > 集群列表”，在集群列表中集群“dws-demo”所在行，单击“更多 > 删除”。然后在弹出对话框中勾选“释放与集群绑定的弹性IP”，单击“确定”。

如果待删除集群使用了自动创建的安全组，且该自动创建的安全组没有被别的集群使用，删除集群时，该安全组也会一起被自动删除。

步骤2 删除子网。删除前请先确保该子网未被其他资源绑定。

登录虚拟私有云管理控制台，在左侧导航树单击“虚拟私有云”，在虚拟私有云列表中，单击名称“vpc-dws”，然后在子网列表中“subnet-dws”所在行单击“删除”。

步骤3 删除虚拟私有云。删除前请先确保该虚拟私有云未被其他资源绑定。

登录虚拟私有云管理控制台，在虚拟私有云列表中，找到虚拟私有云“vpc-dws”，单击其所在行的“删除”。

具体步骤，请参见《虚拟私有云用户指南》中“虚拟私有云和子网 > 删除虚拟私有云”章节。

----**结束**

3 DWS 使用流程

数据仓库服务是一种基于华为云基础架构和平台的在线数据处理数据库，提供即开即用、可扩展且完全托管的分析型数据库服务。DWS是基于融合数据仓库GaussDB产品的云原生服务，兼容标准ANSI SQL 99和SQL 2003，同时兼容PostgreSQL/Oracle数据库生态，为各行业PB级海量大数据分析提供有竞争力的解决方案。

DWS提供了简单易用的Web管理控制台，帮助您快速创建数据仓库集群，轻松执行数据仓库管理任务。

DWS 使用流程介绍

图 3-1 DWS 使用流程

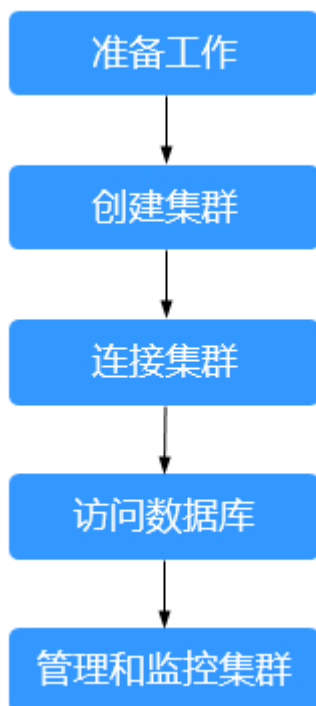


表 3-1 DWS 使用流程介绍

主流程	子任务	说明	操作指导
准备工作	-	在使用DWS服务之前，需要申请云账户。	准备工作
创建集群	-	在使用DWS执行数据分析任务前，您首先要创建一个集群。一个DWS集群由多个在相同子网中的节点组成，共同提供服务。创建集群时，系统会创建一个默认数据库。	● 创建DWS存算一体集群
连接集群	-	DWS集群创建完成后，您就可以使用SQL客户端工具或者JDBC/ODBC等第三方驱动程序连接集群中的数据库。进入DWS控制台的“连接客户端”页面，您可以下载SQL客户端工具和JDBC/ODBC驱动程序。	连接DWS集群
访问数据库	-	连接集群后，您就可以创建和管理数据库，管理用户及权限，导入或导出数据，以及查询和分析数据。	《数据仓库服务数据库开发指南》
管理和监控集群	管理集群	您可以查看集群状态，修改集群配置，添加集群标签，对集群进行扩容、重启、删除等管理操作。	DWS集群管理
	管理快照	通过创建集群快照，可以对集群进行备份，使用快照可以恢复集群。	备份与恢复DWS集群
	运维与监控	您可以通过监控集群、审计日志、事件通知和告警管理了解集群的运行状况和性能。	● 在Cloud Eye查看DWS集群监控 ● 查看并订阅DWS集群事件 ● 查看并订阅DWS集群告警
	扩容与规格变更	● 当需要更多的计算资源或存储资源以满足业务需要时，可以在管理控制台对已有集群进行扩容操作。 ● 集群创建后，支持通过管理控制台变更集群规格。	● 节点变更
	升级集群	8.1.1及以上集群版本支持用户自助通过管理控制台下发集群升级操作。	DWS集群升级

主流程	子任务	说明	操作指导
	资源负载管理	DWS提供了资源管理功能，用户可以根据自身业务将资源（CPU、内存、IO和存储空间）按需划分成不同的资源池，不同资源池之间资源互相隔离。	DWS资源负载管理

4 准备工作

4.1 策略语法：细粒度策略

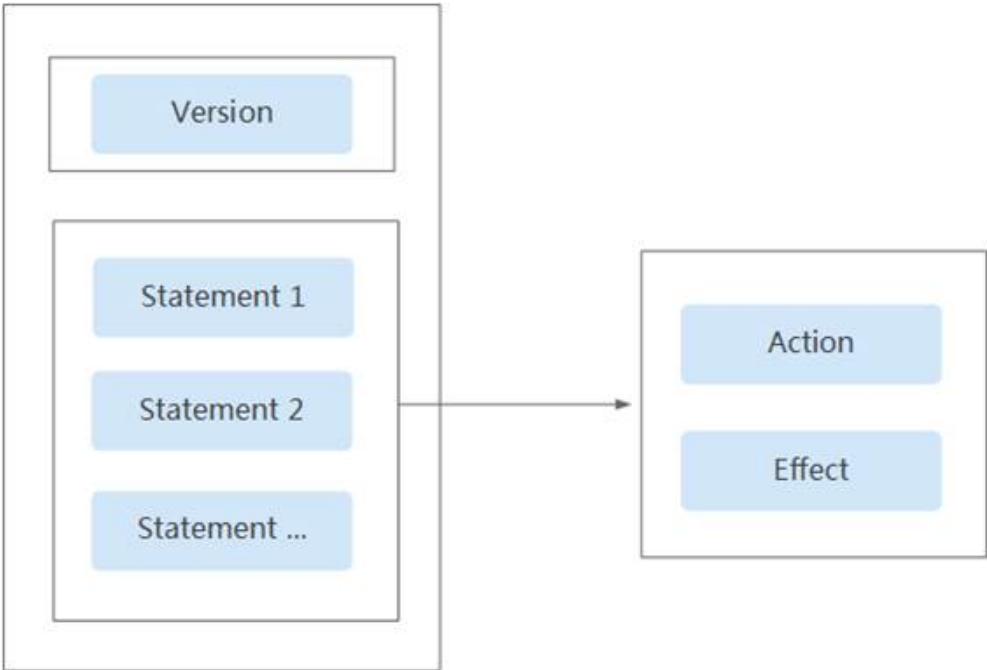
在实际业务中，您可能需要给不同角色的用户授予不同的资源操作权限。IAM服务为用户提供了细粒度访问控制功能。在IAM中，IAM管理员用户（即属于admin用户组的用户）通过创建自定义策略，可以自由搭配需要授予的权限集。通过给用户组授予策略，用户组中的用户就能获得策略中定义的权限。IAM通过策略定义的权限内容实现精细的权限管理。

为了更精细地控制DWS 资源的使用权限，您可以使用IAM的用户管理功能，实现细粒度权限控制，授予不同角色的用户不同的资源操作权限。

策略结构

策略结构包括：Version（策略版本号）和Statement（策略权限语句），其中Statement可以有多个，表示不同的授权项。

图 4-1 策略结构



策略语法

在IAM左侧导航窗格中，单击“策略”，单击策略名称，可以查看策略的详细内容，以“DWS ReadOnlyAccess”为例，说明细粒度策略的语法。

```
{
  "Version": "1.1",
  "Depends": [],
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "dws:*:get*",
        "dws:*:list*",
        "ecs:*:get*",
        "ecs:*:list*",
        "vpc:*:get*",
        "vpc:*:list*",
        "evs:*:get*",
        "evs:*:list*",
        "mrs:*:get*",
        "bss:*:list*",
        "bss:*:get*"
      ]
    }
  ]
}
```

- Version：标识策略的版本号，主要用于区分Role-Based Access Control（RBAC）策略和细粒度策略。
 - 1.0：RBAC策略。RBAC策略是将服务作为一个整体进行授权，授权后，用户可以拥有这个服务的所有权限。
 - 1.1：经典细粒度策略。相比RBAC策略，细粒度策略基于服务的API接口进行权限拆分，授权更加精细。授权后，用户可以对这个服务执行特定的操作。细粒度策略包括系统预置和用户自定义两种。

- Depends：依赖项。
- Statement：策略授权语句，描述策略的详细信息，包含Effect（作用）和Action（授权项）。
 - Effect（作用）

作用包含两种：Allow（允许）和Deny（拒绝），系统预置策略仅包含Allow（允许）的授权语句，自定义策略中可以同时包含Allow（允许）和Deny（拒绝）的授权语句，当策略中既有Allow（允许）又有Deny（拒绝）的授权语句时，遵循Deny（拒绝）优先的原则。
 - Action（授权项）

对资源的具体操作权限，格式为：“**服务名:资源类型:操作**”，支持单个或多个操作权限，支持通配符号“*”，通配符号表示所有。

示例："dws:cluster:create"，其中dws为服务名，cluster为资源类型，create为操作，该授权项表示创建DWS集群的权限。

授权项列表

在IAM中创建自定义策略时，您可以根据需求在策略授权语句的Action列表中添加DWS资源操作或REST API所对应的“授权项”，使得该策略具有相应的操作权限。DWS细粒度策略的授权项列表如下：

- **REST API**

DWS REST API的授权项列表，请参见《数据仓库服务API参考》中的“授权策略和授权项”。

- **管理控制台操作**

DWS资源操作及对应的授权项如[表4-1](#)所示。

说明

- DWS部分授权项依赖的授权项包括了ECS、VPC、EVS、ELB、MRS或OBS等服务的授权项，如果这些服务没有对接相应的服务授权项，则需要添加对应服务的Admin系统权限。
- 由于DWS接口较多，以下列表仅列举了重点高频操作接口，剩余未展示接口仅支持project项目（即IAM鉴权），不支持企业项目鉴权，故如果要使用，请在IAM鉴权界面配置。

表 4-1 DWS 资源操作授权项列表

DWS 资源操作	授权项	依赖的授权项	授权项作用域
创建集群	"dws:cluster:create"	"dws:*.get*", "dws:*.list*", "ecs:*.get*", "ecs:*.list*", "ecs:*.create*", "vpc:*.get*", "vpc:*.list*", "vpc:*.create*", "vpc:securityGroup Rules:delete", "vpc:ports:update", "evs:*.get*", "evs:*.list*", "evs:*.create*",	● 支持： - 项目 (Project)
获取集群列表	"dws:cluster:list"	--	● 支持： - 项目 (Project)
获取单个集群详情	"dws:cluster:getDetail"	"dws:*.get*", "dws:*.list*", "vpc:vpcs:list", "vpc:securityGroups: get"	● 支持： - 项目 (Project)
设置自动快照	"dws:cluster:setAutomatedSnapshot"	"dws:backupPolicy: list"	● 支持： - 项目 (Project)
设置安全参数/参数组	"dws:cluster:setSecuritySettings"	"dws:*.get*", "dws:*.list*",	● 支持： - 项目 (Project)
重启集群	"dws:cluster:restart"	"dws:*.get*", "dws:*.list*",	● 支持： - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
扩容集群	"dws:cluster:scaleOut"	"dws:*.get*", "dws:*.list*", "dws:cluster:scaleOutOrOpenAPIResize", "ecs:*.get*", "ecs:*.list*", "ecs:*.create*", "vpc:*.get*", "vpc:*.list*", "vpc:*.create*", "vpc:*.update*", "evs:*.get*", "evs:*.list*", "evs:*.create*",	- 支持： - 项目 (Project)
使用API扩容集群或调整大小	"dws:cluster:scaleOutOrOpenAPIResize"	"dws:*.get*", "dws:*.list*", "vpc:vpcs:list", "vpc:ports:create", "vpc:ports:get", "vpc:ports:update", "vpc:subnets:get", "vpc:subnets:update", "vpc:subnets:create", "vpc:routers:get", "vpc:routers:update", "vpc:networks:create", "vpc:networks:get", "vpc:networks:update", "ecs:serverInterfaces:use", "ecs:serverInterfaces:get", "ecs:cloudServerFlavors:get"	- 支持： - 项目 (Project) - 企业项目 (Enterprise Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
重置密码	"dws:cluster:resetPassword"	"dws:*.get*", "dws:*.list"	- 支持： - 项目 (Project)
删除集群	"dws:cluster:delete"	"dws:*.get*", "dws:*.list*", "ecs:*.get*", "ecs:*.list*", "ecs:*.delete*", "vpc:*.get*", "vpc:*.list*", "vpc:*.delete*", "evs:*.get*", "evs:*.list*", "evs:*.delete"	- 支持： - 项目 (Project)
设置可维护时间段	"dws:cluster:setMaintenanceWindow"	"dws:*.get*", "dws:*.list"	- 支持： - 项目 (Project)
绑定EIP	"dws:eip:operate"	"dws:*.get*", "dws:*.list*", "eip:*.get*", "eip:*.list"	- 支持： - 项目 (Project)
解绑EIP	"dws:eip:operate"	"dws:*.get*", "dws:*.list*", "eip:*.get*", "eip:*.list"	- 支持： - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
创建MRS连接	"dws:MRSCONNECTION:create"	"dws:*:get*", "dws:*:list*", "mrs:*:get*", "mrs:*:list*", "mrs:cluster:create", , "ecs:*:get*", "ecs:*:list*", "ecs:*:create*", "vpc:*:get*", "vpc:*:list*", "vpc:*:create*", "evs:*:get*", "evs:*:list*", "evs:*:create"	- 支持： - 项目 (Project)
更新MRS连接	"dws:MRSCONNECTION:update"	"dws:*:get*", "dws:*:list*", "mrs:*:get*", "mrs:*:list*", "mrs:cluster:create", , "ecs:*:get*", "ecs:*:list*", "ecs:*:create*", "vpc:*:get*", "vpc:*:list*", "vpc:*:create*", "evs:*:get*", "evs:*:list*", "evs:*:create"	- 支持： - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
删除MRS连接	"dws:MRSCONNECTION:delete"	"dws:*.get*", "dws:*.list*", "mrs:*.get*", "mrs:*.list*", "mrs:cluster:create" "ecs:*.get*", "ecs:*.list*", "ecs:*.delete*", "vpc:*.get*", "vpc:*.list*", "vpc:*.delete*", "evs:*.get*", "evs:*.list*", "evs:*.delete*",	- 支持： - 项目 (Project)
MRS数据源列表	"dws:MRSSOURCE:list"	"mrs:cluster:list", "mrs:tag:listResource", "mrs:tag:list", "dws:*.get*", "dws:*.list"	- 支持： - 项目 (Project)
添加/删除标签	"dws:tag:addAndDelete"	"dws:*.get*", "dws:*.list*", "dws:openAPITag:update", "dws:openAPITag:getResourceTag",	- 支持： - 项目 (Project)
编辑标签	"dws:tag:edit"	"dws:*.get*", "dws:*.list*", "dws:openAPITag:update", "dws:openAPITag:getResourceTag",	- 支持： - 项目 (Project)
创建快照	"dws:snapshot:create"	"dws:*.get*", "dws:*.list*",	- 支持： - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
获取快照列表	"dws:snapshot:list"	--	- 支持： - 项目 (Project)
查看单个集群快照列表	"dws:clusterSnapshot:list"	"dws:cluster:list", "dws:openAPICluster:getDetail"	- 支持： - 项目 (Project)
删除快照	"dws:snapshot:delete"	"dws:snapshot:list"	- 支持： - 项目 (Project)
复制快照	"dws:snapshot:copy"	"dws:snapshot:list", "dws:snapshot:create"	- 支持： - 项目 (Project)
恢复到新集群	"dws:cluster:restore"	"dws:*.get*", "dws:*.list*", "ecs:*.get*", "ecs:*.list*", "ecs:*.create*", "vpc:*.get*", "vpc:*.list*", "vpc:*.create*", "evs:*.get*", "evs:*.list*", "evs:*.create"	- 支持： - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
集群调整大小	"dws:cluster:resize"	"dws:*.get*", "dws:*.list*", "ecs:*.get*", "ecs:*.list*", "ecs:*.create*", "ecs:*.delete*", "vpc:*.get*", "vpc:*.list*", "vpc:*.create*", "vpc:*.delete*", "evs:*.get*", "evs:*.list*", "evs:*.create*", "evs:*.delete"	- 支持： - 项目 (Project)
主备恢复	"dws:cluster:switchover"	"dws:*.get*", "dws:*.list"	- 支持： - 项目 (Project)
查询弹性负载均衡列表	"dws:elb:list"	"dws:*.get*", "dws:*.list*", "elb:*.get*", "elb:*.list"	- 支持： - 项目 (Project)
绑定弹性负载均衡	"dws:elb:bind"	"dws:*.get*", "dws:*.list*", "ecs:*.get*", "ecs:*.list*", "vpc:*.get*", "vpc:*.list*", "evs:*.get*", "evs:*.list*", "elb:*.get*", "elb:*.list*", "elb:*.delete*", "elb:*.create"	- 支持： - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
解绑弹性负载均衡	"dws:elb:unbind"	"dws:*.get*", "dws:*.list*", "ecs:*.get*", "ecs:*.list*", "vpc:*.get*", "vpc:*.list*", "evs:*.get*", "evs:*.list*", "elb:*.get*", "elb:*.list*", "elb:*.delete"	- 支持： - 项目 (Project)
查询快照配置参数	"dws:snapshotConfig:list"	"dws:*.get*", "dws:*.list"	- 支持： - 项目 (Project)
更新快照策略	"dws:backupPolicyDetail:update"	"dws:*.get*", "dws:*.list"	- 支持： - 项目 (Project)
删除快照策略	"dws:backupPolicy:delete"	"dws:*.get*", "dws:*.list"	- 支持： - 项目 (Project)
查询快照策略	"dws:backupPolicy:list"	"dws:cluster:list"	- 支持： - 项目 (Project)
查询集群加密信息	"dws:clusterEncryptInfo:list"	"dws:*.get*", "dws:*.list*", "KMS Administrator"	- 支持： - 项目 (Project)
创建代理	"dws:createAgency:create"	"dws:*.get*", "dws:*.list*", "security administrator"	- 支持： - 项目 (Project)
查询obs桶信息	"dws:queryBuckets:list"	"dws:*.get*", "dws:*.list"	- 支持： - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
扩容节点	"dws:expandWithExistendNodes:update"	"dws:*.get*", "dws:*.list*", "ecs:*.get*", "ecs:*.list*", "ecs:*.create*", "vpc:*.get*", "vpc:*.list*", "vpc:*.create*", "vpc:*.update*", "evs:*.get*", "evs:*.list*", "evs:*.create*",	- 支持： - 项目 (Project)
删除容灾备份	"dws:disasterRecovery:delete"	"dws:*.get*", "dws:*.list*", "ecs:*.get*", "ecs:*.list*", "ecs:*.delete*", "vpc:*.get*", "vpc:*.list*", "vpc:*.delete*", "evs:*.get*", "evs:*.list*", "evs:*.delete"	- 支持： - 项目 (Project)
创建容灾备份	"dws:disasterRecovery:create"	"dws:*.get*", "dws:*.list*", "ecs:*.get*", "ecs:*.list*", "ecs:*.create*", "vpc:*.get*", "vpc:*.list*", "vpc:*.create*", "evs:*.get*", "evs:*.list*", "evs:*.create*",	- 支持： - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
容灾备份其他操作	"dws:disasterRecovery:otherOperate"	"dws:*.get*", "dws:*.list*", "ecs:*.get*", "ecs:*.list*", "ecs:*.create*", "vpc:*.get*", "vpc:*.list*", "vpc:*.create*", "evs:*.get*", "evs:*.list*", "evs:*.create"	- 支持： - 项目 (Project)
容灾备份查询操作	"dws:disasterRecovery:get"	"dws:*.get*", "dws:*.list*", "ecs:*.get*", "ecs:*.list*", "vpc:*.get*", "vpc:*.list*", "evs:*.get*", "evs:*.list"	- 支持： - 项目 (Project)
增加CN节点	"dws:module:install"	"dws:*.get*", "dws:*.list*",	- 支持： - 项目 (Project)
删除CN节点	"dws:module:uninstall"	"dws:*.get*", "dws:*.list*",	- 支持： - 项目 (Project)
删除节点	"dws:clusterNodes:operate"	"dws:*.get*", "dws:*.list"	- 支持： - 项目 (Project)
更新节点别名	dws:instanceAliasName:update	dws:cluster:list	- 支持： - 项目 (Project)
实施重分布	"dws:redistribution:operate"	"dws:*.get*", "dws:*.list*",	- 支持： - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
查询重分布	"dws:redistributionInfo:list"	"dws:*.get*", "dws:*.list"	- 支持： - 项目 (Project)
停止重分布	"dws:redistribution:suspend"	"dws:*.get*", "dws:*.list"	- 支持： - 项目 (Project)
恢复重分布	"dws:redistribution:recover"	"dws:*.get*", "dws:*.list"	- 支持： - 项目 (Project)
查询规格产品信息	"dws:specProduct:list"	"dws:*.get*", "dws:*.list*", "ecs:*.get*", "ecs:*.list"	- 支持： - 项目 (Project)
创建集群前检查	"dws:checkCluster:create"	"dws:*.get*", "dws:*.list*", "ecs:*.get*", "ecs:*.list*", "ecs:*.create*", "vpc:*.get*", "vpc:*.list*", "vpc:*.create*", "evs:*.get*", "evs:*.list*", "evs:*.create"	- 支持： - 项目 (Project)
绑定管理面IP	"dws:bindManagementIp:operate"	"dws:*.get*", "dws:*.list"	- 支持： - 项目 (Project)
获取用户授权	"dws:checkAuthorize:operate"	"dws:*.get*", "dws:*.list*", "dws:checkSupport:operate"	- 支持： - 项目 (Project)
用户授权	"dws:authorize:operate"	"dws:*.get*", "dws:*.list*", "dws:checkSupport:operate"	- 支持： - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
获取用户数据库	"dws:userDatabase:list"	"dws:*.get*", "dws:*.list*", "dws:checkSupport:operate"	- 支持： - 项目 (Project)
获取用户结构	"dws:schemas:list"	"dws:*.get*", "dws:*.list*", "dws:checkSupport:operate"	- 支持： - 项目 (Project)
获取用户表	"dws:tables:list"	"dws:*.get*", "dws:*.list*",	- 支持： - 项目 (Project)
表恢复	"dws:tableRestore:operate"	"dws:*.get*", "dws:*.list*",	- 支持： - 项目 (Project)
用户恢复表名检测	"dws:tableRestoreCheck:operate"	"dws:*.get*", "dws:*.list*",	- 支持： - 项目 (Project)
检测集群是否支持细粒度备份	"dws:checkSupport:operate"	"dws:*.get*", "dws:*.list*",	- 支持： - 项目 (Project)
查询支持变更的规格列表	"dws:supportFlavors:list"	"dws:*.get*", "dws:*.list*",	- 支持： - 项目 (Project)
执行弹性变更规格	"dws:specResize:operate"	"dws:*.get*", "dws:*.list*", "ecs:*.get*", "ecs:*.list*", "ecs:*.create"	- 支持： - 项目 (Project)
停止快照	"dws:snapshot:stop"	"dws:snapshot:list"	- 支持： - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
终止会话	"dws:dmsSession:terminate"	"dws:dmsGrpcOuter:operation"	- 支持： - 项目 (Project)
负荷诊断报告操作	"dws:dmsWorkloadDiagnosisReport:create"	"dws:dmsGrpcOuter:operation"	- 支持： - 项目 (Project)
修改告警规则	"dws:dmsAlarmRule:update"	"dws:dmsQuery:list"	- 支持： - 项目 (Project)
启用告警规则	"dws:dmsAlarmRule:enable"	"dws:dmsQuery:list"	- 支持： - 项目 (Project)
启用集群告警	"dws:dmsClusterAlarm:enable"	"dws:dmsQuery:list"	- 支持： - 项目 (Project)
禁用集群告警	"dws:dmsClusterAlarm:disable"	"dws:dmsQuery:list"	- 支持： - 项目 (Project)
GRPC对外服务	"dws:dmsGrpcOuter:operation"	"dws:dmsQuery:list", "dws:cluster:setSecuritySettings", "obs:bucket:ListAllMyBuckets"	- 支持： - 项目 (Project)
新增SQL探针	"dws:dmsProbe:add"	"dws:dmsGrpcOuter:operation"	- 支持： - 项目 (Project)
修改SQL探针	"dws:dmsProbe:update"	"dws:dmsGrpcOuter:operation"	- 支持： - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
删除SQL探针	"dws:dmsProbe:delete"	"dws:dmsGrpcOuter:operation"	- 支持： - 项目 (Project)
启用/禁用SQL探针	"dws:dmsProbe:enable"	"dws:dmsGrpcOuter:operation"	- 支持： - 项目 (Project)
创建用户面板	"dws:dmsUserBoard:create"	"dws:dmsQuery:list"	- 支持： - 项目 (Project)
修改用户面板	"dws:dmsUserBoard:update"	"dws:dmsQuery:list"	- 支持： - 项目 (Project)
删除用户面板	"dws:dmsUserBoard:delete"	"dws:dmsQuery:list"	- 支持： - 项目 (Project)
终止查询	"dws:dmsQuery:terminate"	"dws:dmsGrpcOuter:operation"	- 支持： - 项目 (Project)
启停DMS监控服务	"dws:dmsService:enableOrDisable"	"dws:dmsQuery:list"	- 支持： - 项目 (Project)
修改DMS存储配置	"dws:dmsStorageConfig:modify"	"dws:dmsQuery:list"	- 支持： - 项目 (Project)
DDL审核创建获取	"dws:dmsDdlExamine:getOrCreate"	"dws:dmsGrpcOuter:operation"	- 支持： - 项目 (Project)
负荷快照操作	"dws:dmsWorkloadDiagnosisSnapshot:create"	"dws:dmsGrpcOuter:operation"	- 支持： - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
创建告警规则	"dws:dmsAlarmRule:add"	"dws:dmsQuery:list"	- 支持： - 项目 (Project)
删除告警规则	"dws:dmsAlarmRule:delete"	"dws:dmsQuery:list"	- 支持： - 项目 (Project)
执行SQL探针	"dws:dmsProbe:execute"	"dws:dmsGrpcOuter:operation"	- 支持： - 项目 (Project)
删除监控项	"dws:dmsPerformanceMonitor:delete"	"dws:dmsQuery:list"	- 支持： - 项目 (Project)
启停DMS监控采集项	"dws:dmsCollectItem:enableOrDisable"	"dws:dmsGrpcOuter:operation"	- 支持： - 项目 (Project)
修改DMS监控采集配置	"dws:dmsCollectConfig:modify"	"dws:dmsGrpcOuter:operation"	- 支持： - 项目 (Project)
OPENAPI条件查询	"dws:dmsOpenapiQuery:list"	"dws:cluster:list"	- 支持： - 项目 (Project)
禁用告警规则	"dws:dmsAlarmRule:disable"	"dws:dmsQuery:list"	- 支持： - 项目 (Project)
删除告警记录	"dws:dmsAlarmRecord:delete"	"dws:dmsQuery:list"	- 支持： - 项目 (Project)
检查SQL探针	"dws:dmsProbe:check"	"dws:dmsGrpcOuter:operation"	- 支持： - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
新增监控项	"dws:dmsPerformanceMonitor:add"	"dws:dmsQuery:list"	- 支持： - 项目 (Project)
修改监控项	"dws:dmsPerformanceMonitor:update"	"dws:dmsQuery:list"	- 支持： - 项目 (Project)
下载历史监控趋势	"dws:dmsTrendHistory:down"	"dws:dmsQuery:list"	- 支持： - 项目 (Project)
获取集群ring环信息	"dws:ring:list"	"dws:*.get*", "dws:*.list*"	- 支持： - 项目 (Project)
获取群进程拓扑	"dws:processTopo:list"	"dws:*.get*", "dws:*.list*"	- 支持： - 项目 (Project)
查询智能运维信息	"dws:operationalTask:get"	"dws:*.get*", "dws:*.list*"	- 支持： - 项目 (Project)
智能运维执行操作	"dws:operationalTask:operate"	"dws:*.get*", "dws:*.list*"	- 支持： - 项目 (Project)
逻辑集群弹性计划操作	"dws:logicalClusterPlan:operate"	"dws:*.get*", "dws:*.list*", "dws:logicalCluster:*", "dws:cluster:scaleOut", "iam:agencies:*", "iam:permissions:*Agency*"	- 支持： - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
创建终端节点服务	"dws:vpcEndpointService:create"	"dws:*.get*", "dws:*.list"	- 支持： - 项目 (Project)
查询资源管理信息	"dws:workLoadManager:get"	"dws:*.get*", "dws:*.list"	- 支持： - 项目 (Project)
资源管理相关操作	"dws:workLoadManager:operate"	"dws:*.get*", "dws:*.list"	- 支持： - 项目 (Project)
云日志服务相关操作	"dws:ltsAccess:operate"	"dws:*.get*", "dws:*.list"	- 支持： - 项目 (Project)
查询云日志服务信息	"dws:ltsAccess:get"	"dws:*.get*", "dws:*.list"	- 支持： - 项目 (Project)
查询事件信息	"dws:event:list"	"dws:*.get*", "dws:*.list"	- 不支持 - 企业项目 (Enterprise Project) - 支持： - 项目 (Project)
查询事件规格信息	"dws:event:list"	"dws:*.get*", "dws:*.list"	- 不支持 - 企业项目 (Enterprise Project) - 支持： - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
查询事件订阅信息	"dws:eventSub:list"	"dws:*.get*", "dws:*.list"	- 不支持 - 企业项目 (Enterprise Project) - 支持: - 项目 (Project)
创建事件订阅信息	"dws:eventSub:create"	"dws:*.get*", "dws:*.list"	- 不支持 - 企业项目 (Enterprise Project) - 支持: - 项目 (Project)
更新事件订阅信息	"dws:eventSub:update"	"dws:*.get*", "dws:*.list"	- 不支持 - 企业项目 (Enterprise Project) - 支持: - 项目 (Project)
删除事件订阅信息	"dws:eventSub:delete"	"dws:*.get*", "dws:*.list"	- 不支持 - 企业项目 (Enterprise Project) - 支持: - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
查询告警统计信息	"dws:alarmStatistic:list"	"dws:*.get*", "dws:*.list"	- 不支持 - 企业项目 (Enterprise Project) - 支持: - 项目 (Project)
查询告警详情信息	"dws:alarmDetail:list"	"dws:*.get*", "dws:*.list"	- 不支持 - 企业项目 (Enterprise Project) - 支持: - 项目 (Project)
查询告警配置信息	"dws:alarmConfig:list"	"dws:*.get*", "dws:*.list"	- 不支持 - 企业项目 (Enterprise Project) - 支持: - 项目 (Project)
查询告警订阅信息	"dws:alarmSub:list"	"dws:*.get*", "dws:*.list"	- 不支持 - 企业项目 (Enterprise Project) - 支持: - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
创建告警订阅信息	"dws:alarmSub:create"	"dws:*.get*", "dws:*.list"	- 不支持 - 企业项目 (Enterprise Project) - 支持: - 项目 (Project)
更新告警订阅信息	"dws:alarmSub:update"	"dws:*.get*", "dws:*.list"	- 不支持 - 企业项目 (Enterprise Project) - 支持: - 项目 (Project)
删除告警订阅信息	"dws:alarmSub:delete"	"dws:*.get*", "dws:*.list"	- 不支持 - 企业项目 (Enterprise Project) - 支持: - 项目 (Project)
下发集群升级相关操作 (升级、回滚、提交、重试)	"dws:cluster:doUpdate"	"dws:*.get*", "dws:*.list"	- 不支持 - 企业项目 (Enterprise Project) - 支持: - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
查询集群可用的升级路径信息	"dws:cluster:getUpgradePaths"	"dws:*.get*", "dws:*.list"	- 不支持 - 企业项目 (Enterprise Project) - 支持: - 项目 (Project)
查询集群升级记录	"dws:cluster:getUpgradeRecords"	"dws:*.get*", "dws:*.list"	- 不支持 - 企业项目 (Enterprise Project) - 支持: - 项目 (Project)
启动集群	"dws:cluster:startCluster"	"dws:*.get*", "dws:*.list*", "ecs:*.get*", "ecs:*.list*", "ecs:*.start", "ecs:*.stop"	- 不支持 - 企业项目 (Enterprise Project) - 支持: - 项目 (Project)
停止集群	"dws:cluster:stopCluster"	"dws:*.get*", "dws:*.list*", "ecs:*.get*", "ecs:*.list*", "ecs:*.start", "ecs:*.stop"	- 不支持 - 企业项目 (Enterprise Project) - 支持: - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
获取标签	"dws:openAPItag:list"	"dws:*.list*"	● 不支持 - 企业项目 (Enterprise Project) ● 支持: - 项目 (Project)
服务eps列表	"dws:service:listEps"	"dws:*.list*"	● 支持: - 项目 (Project)
容灾信息获取	"dws:disasterRecovery:get"	"dws:*.*)"	● 不支持 - 企业项目 (Enterprise Project) ● 支持: - 项目 (Project)
集群恢复检查	"dws:cluster:checkRestore"	"dws:*.*)"	● 不支持 - 企业项目 (Enterprise Project) ● 支持: - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
告警静态列表	"dws:alarmStatistic:list"	"dws:*.list*"	● 不支持 - 企业项目 (Enterprise Project) ● 支持: - 项目 (Project)
获取资源静态信息	"dws:service:getResourceStatistics"	"dws:*.*)"	● 不支持 - 企业项目 (Enterprise Project) ● 支持: - 项目 (Project)
告警细节列表	"dws:alarmDetail:list"	"dws:*.list*"	● 不支持 - 企业项目 (Enterprise Project) ● 支持: - 项目 (Project)
获取集群细节	"dws:openAPICluster:getDetail"	"dws:*.*)"	● 不支持 - 企业项目 (Enterprise Project) ● 支持: - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
集群事件规格列表	"dws:eventSpec:list"	"dws:*.list**"	● 不支持 - 企业项目 (Enterprise Project) ● 支持: - 项目 (Project)
集群容灾列表	"dws:cluster:listDisasterRecovery"	"dws:*.list**",	● 不支持 - 企业项目 (Enterprise Project) ● 支持: - 项目 (Project)
告警数据总览	"dws:alarm:listStatistics"	"dws:*.list**",	● 不支持 - 企业项目 (Enterprise Project) ● 支持: - 项目 (Project)
查询DWS集群中Schema	"dws:monitor:listClusterOverview"	"dws:*.get**", "dws:*.list**"	● 不支持 - 企业项目 (Enterprise Project) ● 支持: - 项目 (Project)

DWS 资源操作	授权项	依赖的授权项	授权项作用域
查询历史监控数据	"dws:monitor:getHistoryMetrics"	"dws:*.get*", "dws:*.list"	- 不支持 - 企业项目（Enterprise Project） - 支持： - 项目（Project）
查询列表展示列配置	"dws:cluster:listQueryFormDMS"	"dws:*.get*", "dws:*.list"	- 不支持 - 企业项目（Enterprise Project） - 支持： - 项目（Project）
新增或修改列表展示列	"dws:cluster:listQueryFormDMS"	"dws:*.get*", "dws:*.list"	- 不支持 - 企业项目（Enterprise Project） - 支持： - 项目（Project）

细粒度策略授权

步骤1 登录IAM服务管理控制台，创建自定义策略。

具体操作，请参见《统一身份认证服务用户指南》中的“用户指南 > 管理细粒度策略 > 创建自定义策略”。

说明如下：

- 您必须使用IAM管理员用户，即属于admin用户组的用户，因为只有IAM管理员用户具备创建用户组及用户、修改用户组权限等操作权限。

- 由于DWS服务属于项目级服务，“作用范围”必须选择“**项目级服务**”，如果需要该策略对多个项目生效，需要对多个项目分别授权。
- 在IAM中，预置了以下两种DWS策略模板。在创建自定义策略时，您可以选择以下模板，然后基于模板修改策略授权语句。
 - DWS Admin：拥有对数据仓库服务的所有执行权限。
 - DWS Viewer：拥有对数据仓库服务的只读权限。
- 在策略授权语句中，您可以在Action列表中，添加如[授权项列表](#)所述的DWS资源操作或REST API对应的“授权项”，从而使策略获得相应的操作权限。
例如，在策略语句的Action列表中，添加“dws:cluster:create”，那么该策略就拥有了创建集群的权限。
- 如果需要使用其他服务，您同时还需授予其他服务的相关操作权限，具体内容请查阅相关服务的帮助文档。
例如，创建DWS 集群时，需要配置集群所属的虚拟私有云，为了能获取VPC列表，您需在策略语句中添加授权项“vpc:*:get*”。

步骤2 创建用户组。

具体操作，请参见《统一身份认证服务用户指南》中的“用户指南 > 管理用户和用户组 > 创建用户组”。

步骤3 将用户加入用户组，并将新创建的自定义策略授权给用户组，使用户组中的用户具有策略定义的权限。

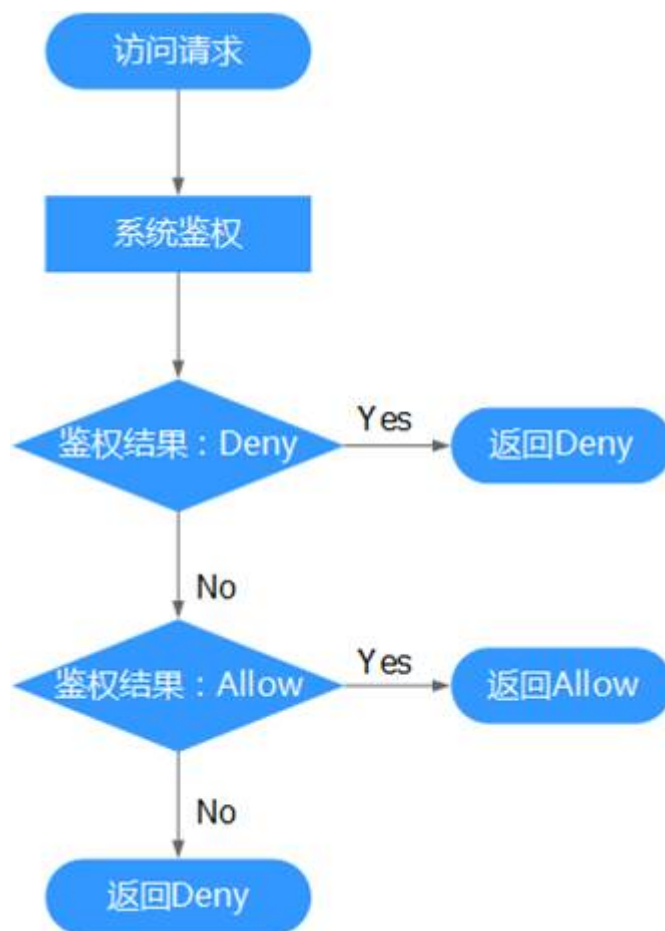
具体操作，请参见《统一身份认证服务用户指南》中的“用户指南 > 管理用户和用户组 > 查看或修改用户组”。

----结束

检查规则

当用户被授予多个策略，或者一个策略中包含多个授权语句，这些策略中既有Allow又有Deny的授权语句时，遵循Deny优先的原则。在用户访问资源时，权限检查逻辑如下所示：

图 4-2 系统鉴权逻辑图



说明

每条策略做评估时，Action之间是或（or）的关系。

1. 用户访问系统，发起操作请求。
2. 系统评估用户被授予的访问策略，鉴权开始。
3. 在用户被授予的访问策略中，系统将优先寻找显式拒绝指令。如找到一个适用的显式拒绝，系统将返回Deny决定。
4. 如果没有找到显式拒绝指令，系统将寻找适用于请求的任何Allow指令。如果找到一个显式允许指令，系统将返回Allow决定。
5. 如果找不到显式允许，最终决定为Deny，鉴权结束。

4.2 策略语法：RBAC

策略结构

策略结构包括：策略版本号（Version）、策略授权语句（Statement）和策略依赖（Depends）。

图 4-3 RBAC 策略结构



策略语法

给用户组选择策略时，单击策略下方的 ，可以查看策略的详细内容，以“DWS Administrator”为例，说明RBAC策略的语法。

图 4-4 RBAC 策略语法

配置权限

若以下列表中没有您需要的策略，可以 [修改已有策略](#)或[新建自定义策略](#)。

查看已选(0)

全部类型 | DWS | | |

|

☐	策略名称	策略描述	项目[作用范围]
☐	DWS Administrator	数据仓库服务（DWS）管理员，拥有读服务...	所有项目（包括未来在所有区域下创建的项...

JSON视图

```
1 {  
2   "Version": "1.0",  
3   "Statement": [  
4     {  
5       "Effect": "Allow",  
6       "Action": [  
7         "DWS:DWS:*"  
8       ]  
9     },  
10  ],  
11  "Depends": [  
12    {  
13      "catalog": "BASE",  
14      "display_name": "Server Administrator"  
15    },  
16    {  
17      "catalog": "BASE",  
18      "display_name": "Tenant Guest"  
19    }  
20  ]  
21 }
```

```
{  
  "Version": "1.0",  
  "Statement": [  
    {  
      "Effect": "Allow",
```

```
        "Action": [
            "dws:dws:*"
        ]
    },
    "Depends": [
        {
            "catalog": "BASE",
            "display_name": "Server Administrator"
        },
        {
            "catalog": "BASE",
            "display_name": "Tenant Guest"
        }
    ]
}
```

表 4-2 RBAC 策略语法

参数		含义	值
Version		策略的版本。	固定为“1.0”。
Statement	Action	定义对DWS 的具体操作。	格式为：服务名:资源类型:操作 "dws:dws:*"，表示对DWS 的所有操作，其中dws为服务名称；“*”为通配符，表示对所有DWS 的资源类型可以执行所有操作。
	Effect	定义Action中所包含的具体操作是否允许执行。	● Allow：允许执行。 ● Deny：不允许执行。
Depends	catalog	依赖的其他策略的所属目录。	服务名称 例如：BASE
	display_name	依赖的其他权限的名称。	权限名称 例如：Server Administrator

 说明

在使用RBAC鉴权时要注意Depends参数，使用时要把依赖的其他权限同时授予。
例如DWS Administrator权限使用时依赖Server Administrator以及Tenant Guest，在给用户授权时要同时把依赖的两个权限授予用户。

5 创建 DWS 集群

5.1 创建 DWS 专属集群

5.1.1 创建 DWS 存算一体集群

如果用户需要在华为云环境中使用DWS，首先要创建一个数据仓库集群。

本章节介绍如何在DWS管理控制台创建一个数据仓库集群。

创建集群前的准备

- 评估集群节点规格
用户可以根据数据量、业务负载以及性能需求，选择能够支撑业务应用的节点数量，数量越多，存储与计算能力越强。
首次使用DWS服务，可以先创建一个规格较小的集群，今后随着数据量和业务负载的变化，再自由调整集群规模和节点规格，自由扩展而不中断业务。详情请参见[集群扩容](#)。
- 确定用户可使用的节点数
用户可使用的节点数需满足以下条件，否则系统会提示无法创建集群：
可使用的节点数取决于所选产品类型，可使用节点数大于或者等于3。

创建集群

- 步骤1** 登录DWS管理控制台。
- 步骤2** 选择左侧导航栏的“专属集群 > 集群列表”。
- 步骤3** 在“专属集群”页面，单击“创建数据仓库集群”。
- 步骤4** 选择“区域”，选择集群节点实际工作区域。
有关区域的详细信息，请参见[地区和终端节点](#)。
- 步骤5** 选择“可用分区”，用户可根据需求选择“单AZ”或“多AZ”。
关于区域和可用分区的描述，请参见[如何选择DWS区域和可用分区](#)。

 说明

- 多AZ集群仅8.2.0.100及以上集群版本支持。
- 只有当前选择的区域可用分区数量>=3时会展示“多AZ”选项，不满足该条件只能创建单AZ集群。
- 多AZ集群目前只支持同时选择3个AZ，主机节点均匀分配在选择的3个AZ中。
- 多AZ集群节点数量必须满足3的倍数。
- 多AZ集群只支持单节点DN数量<=2的规格。

步骤6 配置数据仓库参数，请参见表5-1进行配置。

表 5-1 主机规格参数说明

参数名	参数解释	样例值
产品类型	产品类型有： • 存算一体：面向数据分析场景，为用户提供高性能、高扩展、高可靠、高安全、低时延、易运维的企业级数仓服务，支持2048节点、20PB级超大规模数据分析能力，适用于“库、仓、市、湖”一体化的融合分析业务。	-
节点规格	请根据业务需求合理选择节点规格。在节点规格列表中展示了每一种节点规格单个节点的vCPU、内存和建议使用场景。	dws.m3.xlarge
热数据存储	每节点可用存储容量。 说明 • 申请的存储空间会有必要的文件系统开销，这些开销包括索引节点，以及数据库运行必须的空间。存储空间的数值必须为100的整倍数。 • 界面上显示200GB/节点是用户的业务数据实际的存储容量，例如节点数量选为3个，那么资源总量为600GB。 • 选择冷热数据存储时默认会自动创建表空间，用户不需要手动创建，该特性仅8.1.3及以上集群版本支持。	-
节点数量	选择集群中的节点个数。 支持的节点数范围为3～256。	3
资源总量（GB）	显示集群的总容量。 各个规格对应的存储容量均为数据库存储数据的实际空间，所展示存储容量已扣除副本、RAID所消耗的磁盘空间。	-

步骤7 单击“下一步：网络配置”。

步骤8 配置网络。

表 5-2 网络参数说明

参数名	参数解释	样例值
虚拟私有云	指定集群节点使用的虚拟专用网络，实现不同业务的网络隔离。 首次创建数据仓库集群时，如果未配置过虚拟私有云，可以单击“查看虚拟私有云”进入虚拟私有云管理控制台，新创建一个满足需求的虚拟私有云。 如何创建虚拟私有云，具体请参见《虚拟私有云用户指南》中“虚拟私有云和子网 > 创建虚拟私有云和子网”章节。 在下拉框中选择一个虚拟私有云后，单击“查看虚拟私有云”可以进入虚拟私有云管理控制台查看虚拟私有云的详细信息。 单击可以刷新“虚拟私有云”下拉框中选项值。	vpc-dws
子网	指定虚拟私有云的一个子网。 集群使用子网实现与其他网络的隔离，并独享所有网络资源，以提高网络安全。 说明 集群创建成功后不支持修改子网，如果一定需要修改，可通过集群恢复快照到新集群，新集群的数据和旧集群一样且新集群在创建时子网可以更改。	subnet-dws

参数名	参数解释	样例值
安全组	指定虚拟私有云的安全组。 安全组限制安全访问规则，加强集群与其它服务间的安全访问。 ● 自动创建安全组 系统默认选中“自动创建安全组”，设置为该选项时，系统将会自动创建一个默认安全组。 默认安全组的规则如下：出方向允许所有访问，入方向只开放用户设置的DWS 集群的数据库端口。 默认安全组的名称，其格式如下：dws-<集群名称>-<DWS 集群的数据库端口>，例如，dws-dws-demo-8000。 说明 如果安全组和安全组规则的配额不足，提交创建集群申请后系统会报错，用户可以选择已有的安全组后重试。 ● 手动创建并配置安全组 用户也可以登录VPC管理控制台手动创建一个安全组，创建成功后返回到DWS创建集群页面，单击“安全组”下拉列表旁边的  按钮，刷新后在“安全组”下拉列表中选择新建的安全组。 为了使DWS客户端可以连接集群，用户需要在新建的安全组中添加一条入规则，开放DWS集群的数据库端口的访问权限。入规则示例如下： - 协议：TCP。 - 端口范围：8000。指定为创建DWS集群时设置的数据库端口，这个端口是DWS用于接收客户端连接的端口。 - 源地址：选中“IP地址”，然后指定为客户端主机的IP地址，例如“192.168.0.10/32”。 DWS集群创建成功后，安全组支持更改为其他安全组，也可以在当前的安全组中添加、删除或修改安全组规则。如何修改安全组请参见修改安全组。集群更换安全组过程中有短暂的业务中断，请谨慎操作。为了更好的网络性能，选择安全组时不要多于5个。	自动创建安全组

参数名	参数解释	样例值
公网访问	指定用户是否可以在互联网上使用客户端连接集群数据库。支持如下方式： “暂不使用”：暂不使用弹性IP。 “自动分配”：用户指定弹性IP的带宽，系统将自动为集群分配独享带宽的弹性IP，通过弹性IP可以从互联网对集群进行访问。自动分配的弹性IP的带宽名称以集群名称开头。 “使用已有”：为集群绑定指定的弹性IP。如果下拉框中没有可用的弹性IP，可以单击“查看弹性公网IP”进入弹性公网IP页面创建一个满足需要的弹性IP。带宽可根据用户需要设置。 说明 - 每个区域的每个项目首次使用弹性IP绑定功能时，系统将提示创建名称为“DWSAccessVPC”的委托以授权DWS访问VPC。授权成功后，DWS可以在绑定弹性IP的虚拟机故障时切换至健康虚拟机。 - 默认情况下，只有华为云账号用户或拥有Security Administrator权限的用户才具备查询委托和创建委托的权限。账号中的IAM用户，默认没有查询委托和创建委托的权限，在使用弹性IP绑定功能时页面会屏蔽绑定按钮，此时需联系有“DWS Administrator”权限的用户在当前页面完成对DWS的委托授权。 - 公网访问选择“暂不使用”，即不启用公网访问集群。在集群创建成功后，如果用户想通过公网访问集群，可以先为集群绑定弹性IP再创建公网域名，详情请参见创建公网域名。	自动分配
带宽	当“公网访问”选择“自动分配”时，需设置弹性IP的带宽，设置范围：1～100Mbit/s。	50Mbit/s
弹性负载均衡	指定用户是否绑定弹性负载均衡，利用ELB健康检查机制将集群的CN请求快速转发到正常工作的CN节点上，当有CN故障节点时，也能第一时间切换流量到健康节点上，最大化降低集群访问故障。 “暂不使用”：暂不使用ELB。 “使用已有”：为集群绑定指定的ELB。如果下拉框中没有可用的ELB，可以单击“创建弹性负载均衡”进入弹性负载均衡页面创建一个满足需要的ELB。 警告 为保证集群负载均衡和高可用，避免业务中断风险，不建议业务直连单CN，请配置负载均衡。	使用已有

步骤9 单击“下一步：高级配置”。

步骤10 填写集群配置参数。

表 5-3 集群配置参数说明

参数名	参数解释	样例值
集群名称	设置数据仓库集群的名称。 集群名称长度为4到64个字符，且不超过64个字节，1个中文字符为3个字节，其他字符为1个字节，不区分大小写，必须以字母或中文开头，可以包含中文、字母、数字、中划线或者下划线，不能包含其他的特殊字符。	dws-demo
集群版本	显示集群中安装的数据库实例版本。样例值仅供参考，请以实际显示版本号为准。	-
默认数据库	显示集群的默认数据库名称“gaussdb”。 说明 “gaussdb”为默认数据库名称，暂不支持修改。	gaussdb
管理员用户	设置数据库管理员用户名称。 用户命名要求如下： - 只能由小写字母、数字或下划线组成。 - 必须以小写字母或下划线开头。 - 长度为6~64个字符。 - 用户名不能为DWS数据库的关键字。DWS数据库的关键字，具体请参见《数据仓库服务数据库开发指南》中“SQL参考 > 关键字”章节。	dbadmin
管理员密码	设置数据库管理员用户的密码。 密码复杂度要求如下： - 密码长度为12-32个字符。 - 不能与用户名或倒序的用户名相同。 - 密码至少包含大写字母、小写字母、数字和特殊字符四类中的三类，其中可输入的特殊字符为：(~!?,.,;_(){}[]/<>@#%^&*+ \=-)。 - 弱密码检查。 说明 请妥善保管并定期更新密码，避免安全风险。	-
确认密码	重复输入一次数据库管理员用户的密码。	-
数据库端口	设置客户端或应用程序连接集群数据库时使用的端口。 支持的端口范围为8000~30000。 说明 已创建的集群不支持修改数据库端口，只能在创建时进行指定。	8000
集群时区	支持设置租户集群时区，包含系统OS时区、集群数据仓库时区。	-

步骤11 配置集群所属的企业项目。已开通企业项目管理服务的用户才可以配置该参数。默认值为default。

企业项目是一种云资源管理方式，企业项目管理服务提供统一的云资源按项目管理，以及项目内的资源管理、成员管理。

用户可以选择默认的企业项目“default”或其他已有的企业项目。如果要创建新的企业项目，请登录企业管理控制台进行创建，详细操作请参考《企业管理用户指南》。

步骤12 设置高级配置。选择“自定义”时，可以设置以下高级配置参数，若选择“默认配置”，以下参数将使用它们的默认值。

- **备份设备**

设置当前集群使用的备份设备，各参数配置原则如[表5-4](#)所示。

表 5-4 自动快照参数说明

参数名	参数解释
备份设备	下拉框，支持OBS和NFS。
备份NFS文件系统地址（NFS参数）	NFS的共享地址IP。填写弹性文件服务共享路径IP地址，挂载成功后默认在集群实例的/var/chroot/nfsbackup目录下创建挂载目录。

- **CN部署量**

CN即协调节点（Coordinator），负责接收来自应用的访问请求，并向客户端返回执行结果。此外，协调节点还负责分解任务，并调度任务分片在各DN上并行执行。

取值范围为3~集群节点数，最大值为20，默认值为3。在大规模集群下，建议部署多个CN。

- **标签**

标签（Tag）用于标识集群的键值对。“键”和“值”请参见[表5-5](#)。默认未给集群添加标签。

有关标签的更多信息，请参见[标签简介](#)。

表 5-5 标签配置参数

参数	参数说明	样例值
键	有以下选择： - 在输入框的下拉列表中选择预定义标签键或集群已有的资源标签键。 说明 如果添加预定义标签，用户需要预先在标签管理服务中创建好预定义标签，然后在“键”的下拉框中进行选择。用户可以通过单击“查看预定义标签”进入标签管理服务的“预定义标签”页面，然后单击“创建标签”来创建新的预定义标签。更多信息请参见《标签管理服务用户指南》中的“管理 > 预定义标签 > 创建预定义标签”章节。 - 在输入框中输入标签键名称。输入标签键的最大长度为128个字符，不能为空字符串，且首尾字符不能为空格。不能包含“*”，“<”，“>”，“\”，“，”，“ ”，“/”。 说明 同一集群中的键名不能重复。	key01
值	用户可以选择： - 在输入框的下拉列表中选择预定义标签值或集群的资源标签值。 - 在输入框中输入标签值。输入标签值的最大长度为255个字符，首尾字符不能为空格，可以为空字符串。不能包含“*”，“<”，“>”，“\”，“，”，“ ”，“/”。	value01

● 加密数据库

开启后，DWS将使用KMS（密钥管理服务）为集群及其快照的数据启用加密功能。

每个区域的每个项目首次启用数据库加密时，系统会弹出一个“创建委托”的对话框，单击“是”创建名为“DWSAccessKMS”的委托以授权DWS访问KMS，若单击“否”将不会启用加密功能。然后在“密钥名称”的下拉列表中选择已创建的KMS密钥。

须知

- “加密数据库”开关只有拥有“Tenant Admin”权限的用户才能查看和开关。
- 默认情况下，只有华为云账号用户或拥有Security Administrator权限的用户才具备查询委托和创建委托的权限。账号中的IAM用户，默认没有查询委托和创建委托的权限，此时需联系有权限的用户在当前页面完成对DWS的委托授权。
- 数据库加密功能一旦开启就无法关闭。
- 开启数据库加密功能后，用于加密DWS数据库的KMS密钥在使用过程中不能被禁用、删除或冻结，否则会导致DWS集群异常或数据库不可用。
- 开启数据库加密功能后不支持使用openAPI恢复创建成功的快照。

- 方式一：从密钥中选择密钥名称。
- 方式二：输入密钥ID。输入在KMS侧给当前租户进行授权的KMS密钥ID。

当跳转至KMS侧创建授权时被授权对象**必须是账号，不能是用户**。并且授权操作必须至少包含“**查询密钥信息**”、“**加密数据**”、“**解密数据**”3项权限。

步骤13 设置该集群是否开启IPv6双栈功能。开关开启时，客户端或应用程序可通过IPv6的IP连接数据库。

开启IPv6需同时满足以下3个条件：

- **步骤8**配置网络时的子网是IPv6双栈子网。
- 集群规格支持IPv6且支持最多网卡数大于等于3。
- 仅8.2.1.210及以上集群版本支持。

步骤14 单击“下一步：确认配置”，进入配置详情页面。

步骤15 单击“立即创建”。

提交成功后开始创建。单击“返回集群列表”按钮返回集群管理页面，所创集群的初始状态为“创建中”，集群创建需要时间，请等待一段时间。创建成功后状态更新为“可用”，用户可以开始使用集群。

----结束

创建集群失败处理办法

如果集群创建失败，您可以进入DWS管理控制台的“专属集群 > 集群列表”页面，查看集群的状态及创建失败的原因。

查看创建失败原因

步骤1 登录DWS管理控制台，在左侧导航树，单击“专属集群 > 集群列表”，进入“专属集群”页面。

步骤2 在集群列表中，找到“集群状态”显示为“创建失败”的集群。

步骤3 单击“集群状态”列的 可查看集群创建失败的原因。

如果您的问题仍无法解决，请联系技术支持人员。

----结束

删除创建失败的集群

当不再需要创建失败的集群时，可以将其删除。建议您先查看集群创建失败的原因后，再删除集群。

步骤1 登录DWS管理控制台，在左侧导航树，单击“专属集群 > 集群列表”，进入“专属集群”页面。

步骤2 在集群列表中，找到需要删除的创建失败的集群，然后在该集群所在行选择“更多 > 删除”。

步骤3 在弹出框中，再次进行删除确认，可以选择是否执行以下操作：

- 立即创建集群快照

如果集群状态无异常，可以单击“立即创建集群快照”，跳转至快照列表页面，单击“**创建快照**”，为待删除的集群创建一个最新的快照，详情请参见[手动快照](#)。等快照创建完成后，请再返回“专属集群”页面执行删除集群的操作。

- 关联资源名称
 - 释放与集群绑定的弹性IP
如果集群绑定了弹性IP，建议勾选“弹性IP”，将待删除集群的弹性IP资源释放。
 - 自动快照
 - 手动快照
如果用户创建了手动快照，可勾选“手动快照”进行删除。

步骤4 确认无误后，输入“DELETE”或单击“一键输入”，并单击“确定”，删除此集群。此时集群列表中集群状态会显示为“删除中”，并显示集群删除进度。

如果待删除集群使用了自动创建的安全组，且该自动创建的安全组没有被别的集群使用，删除集群时，该安全组也会被一并删除。

----结束

6 连接 DWS 集群

6.1 连接 DWS 集群方式介绍

如果您已经创建了DWS集群，就可以使用SQL客户端工具或者JDBC（Java database connectivity）/ODBC（Open Database Connectivity）等第三方驱动程序连接集群，访问集群中的数据库。

约束和限制

警告

- **尽量避免**所有业务使用同一个数据库用户运行，按业务模块规划不同数据库用户。
- **不建议**使用系统管理员用户跑业务，不同模块业务请通过多用户和权限进行访问控制。
- **不建议**业务直连单CN，需配置为[DWS集群绑定和解绑弹性负载均衡](#)保证各CN连接均衡。
- 连接数据库完成所需操作后，要及时关闭数据库连接，避免空闲连接持续占位，消耗连接和公共资源。
- 使用数据库连接池的场景，在业务中通过SET语句进行数据库GUC参数设置后，归还连接池前必须通过RESET还原参数设置。
- 更多开发设计规范参见《数据仓库服务开发指南》的“DWS总体开发设计规范”章节。

连接方式

连接集群的步骤如下：

1. [获取DWS集群连接地址](#)
2. 如果使用SSL加密连接方式，需要执行：[使用SSL进行安全的TCP/IP连接](#)。
3. 连接集群访问集群数据库。您可以选择以下任意一种方式连接集群。

须知

- 建议使用官方推荐的以下方式连接数据库。
 - 其它客户端不能保证100%兼容性，需要客户自行验证。
 - 使用其它客户端如果出现不兼容而报错，且不能替换客户端的情况，可尝试替换客户端中的libpq驱动。替换方法：参见[下载客户端](#)下载并解压gsq客户端压缩包，获取gsq目录下的libpg.so，替换到客户端指定目录中。
-
- 使用SQL客户端工具连接集群。
 - [使用Linux gsql客户端连接集群](#)
 - [使用Windows gsql客户端连接集群](#)
 - [使用Data Studio连接DWS集群](#)
 - 使用JDBC、ODBC、psycopg2、PyGreSQL第三方驱动程序连接集群。
 - [使用JDBC连接](#)
 - [使用ODBC连接](#)
 - [使用Python第三方库psycopg2连接DWS集群](#)
 - [使用Python第三方库PyGreSQL连接DWS集群](#)

6.2 获取 DWS 集群连接地址

操作场景

DWS支持不同方式连接集群，不同连接方式的连接地址也不同。支持查看并获取华为云平台环境的内网连接地址、互联网环境的公网连接地址和JDBC连接字符串。

获取集群连接地址有以下两种方式：

- [在连接客户端页面获取集群连接地址](#)
- [在“集群详情”页面获取集群连接地址](#)

在“连接客户端”页面获取集群连接地址

步骤1 登录DWS管理控制台。

步骤2 在左侧导航栏中，选择“管理 > 连接客户端”。

步骤3 在“数据仓库连接信息”区域，选择一个可用的集群名称。

只能选择状态为“可用”的集群。

步骤4 查看并获取集群的连接信息。

- “内网IP”
- “公网IP”
- “弹性负载均衡地址”

- “JDBC连接字符串（内网）”
- “JDBC连接字符串（公网）”
- “ODBC连接字符串”

 说明

- 如果创建集群时没有自动绑定弹性IP，“公网IP”显示为空。如果您想使用公网访问地址（由弹性IP和数据库端口组成）从互联网访问集群，可以单击“绑定弹性IP”为集群绑定弹性IP。
- 如果创建集群时绑定了弹性IP，如果您不想使用公网访问地址访问集群，可以单击“解绑弹性IP”为集群解绑弹性IP。弹性IP解绑后，“公网IP”显示为空。
- 如果创建集群时没有绑定弹性负载均衡，“弹性负载均衡地址”显示为空，可以单击“绑定弹性负载均衡”进行绑定，解决CN单点故障。
- 如果集群绑定了弹性负载均衡，为保证集群高可用，请优先使用弹性负载均衡地址连接集群。
- 如果集群开启了IPv6双栈网络功能，则内网IP既有IPv4，也有IPv6，您可根据需要选择IPv4或IPv6进行DWS集群的连接。

----结束

在“集群详情”页面获取集群连接地址

- 步骤1 登录DWS管理控制台。
- 步骤2 在左侧导航栏中，单击“专属集群 > 集群列表”。
- 步骤3 在集群列表中，单击指定集群的名称，打开“集群详情”页面。
- 步骤4 在“连接信息”区域，用户可以查看并获取集群的连接地址信息，包括内网地址、公网地址等。

表 6-1 连接信息

参数名	参数解释
内网域名	通过内部网络访问集群数据库的域名地址，对应所有的内网IP地址，即所有CN的地址。内网访问域名在创建集群时自动生成。 说明 ● 如果集群名称不符合域名规范，默认访问域名前缀会相应做一些调整。 ● 不支持负载均衡能力。 单击“修改”可以修改内网访问域名。访问域名由字母、数字、中划线组成，以大小写字母开头，长度为4~63个字符。
内网IP	通过内部网络访问集群数据库的IP地址。 说明 ● 内网访问IP地址在创建集群时自动生成，生成后的IP地址是固定的。 ● 内网访问IP的数量对应的是CN节点的个数，可以通过登录任一节点连接到集群。 ● 通过内网访问某个固定的IP，资源池会集中在一个CN上。 ● 集群开启IPv6功能后，内网IP会同时展示IPv4和IPv6，您可根据需要进行使用。

参数名	参数解释
公网域名	通过外部网络访问集群数据库的域名地址。 说明 不支持负载均衡能力。
公网IP	通过外部网络访问集群数据库的IP地址。 说明 - 如果创建集群时没有绑定弹性IP，“公网IP”显示为空，可以单击“编辑”为集群绑定弹性IP。 - 如果创建集群时绑定了弹性IP，可以单击“编辑”为集群解绑弹性IP。
初始管理员用户	创建集群时指定的数据库管理员用户。当用户第一次连接集群时，需要使用初始数据库管理员用户及其密码连接到默认数据库。
端口	通过公网或者内网访问集群数据库的端口号，端口号在创建集群时指定。
默认数据库	创建集群时默认自动创建的数据库。当用户第一次连接集群时，需要连接到该默认数据库。
弹性负载均衡地址	为实现集群高可用，解决CN单点问题，集群创建后需手动绑定弹性负载均衡（ELB），连接集群时建议连接ELB地址。 说明 如果集群为IPv4集群，则仅支持手动绑定IPv4的弹性负载均衡（ELB）；如果集群为IPv6双栈集群，则仅支持手动绑定IPv6双栈的弹性负载均衡（ELB），暂不支持交叉场景的ELB绑定。

----结束

6.3 使用可视化工具连接 DWS 集群

6.3.1 使用 Data Studio 连接 DWS 集群

Data Studio是一款运行在Windows操作系统上的SQL客户端工具，有着丰富的GUI界面，能够管理数据库和数据库对象，编辑、运行、调试SQL脚本，查看执行计划等。在DWS管理控制台下载Data Studio软件包，解压后免安装即可使用。

Data Studio可供下载的版本分为“Windows x86”和“Windows x64”两种版本，分别支持32位和64位Windows操作系统。

连接集群前的准备

- 已获取DWS集群的数据库管理员用户名和密码。
- 已获取DWS集群的公网访问地址，含IP地址和端口。具体请参见[获取DWS集群连接地址](#)。
- 已配置DWS集群所属的安全组，添加安全组的入规则允许用户的IP地址使用TCP访问端口。

具体请参见《虚拟私有云用户指南》中“安全性 > 安全组 > 添加安全组规则”章节。

使用 Data Studio 连接到集群数据库

步骤1 DWS提供了基于Windows平台的Data Studio图形界面客户端，该工具依赖JDK，请先在客户端主机上安装JDK。

须知

仅支持Java 1.8版本的JDK。

在Windows操作系统中，您可以访问JDK官方网站，下载符合操作系统版本的JDK，并根据指导进行安装。

步骤2 登录DWS管理控制台。

步骤3 选择“管理 > 连接客户端”。

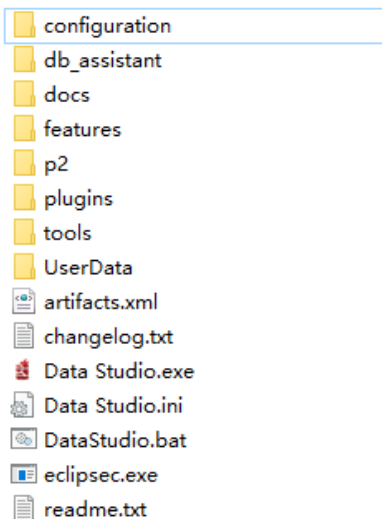
步骤4 在“下载客户端和驱动”页面，下载“Data Studio图形界面客户端”。

- 请根据操作系统类型，选择“Windows x86”或“Windows x64”，再单击“下载”，可以下载与现有集群版本匹配的Data Studio工具。
- 单击“历史版本”可根据集群版本下载相应版本的Data Studio工具，建议按集群版本下载配套的工具。

步骤5 解压下载的客户端软件包（32位或64位）到需要安装的路径。

步骤6 打开安装目录，双击Data Studio.exe，启动Data Studio客户端，如图6-1所示。

图 6-1 启动客户端



说明

若您的电脑阻止应用运行，可对Data Studio.exe文件属性勾选解除锁定即可启动。

步骤7 在主菜单中选择“文件 > 新建连接”，如图6-2所示。

图 6-2 新建连接



步骤8 在弹出的“新建/选择数据库连接”页面中，如下图所示，输入连接参数。

表 6-2 配置连接参数

字段名称	说明	举例
数据库类型	选择“GaussDB A”	GaussDB A
名称	连接名称。	dws-demo
主机	所要连接的集群IP地址（IPv4）或域名。	-
端口号	数据库端口。	8000
数据库	数据库名称。	gaussdb
用户名	所要连接数据库的用户名。	-
密码	所要连接数据库的登录密码。	-
保存密码	在下拉列表中选择： “仅当前会话”：仅在当前会话中保存密码。 “不保存”：不保存密码。	-
启用SSL	启用时，客户端将使用SSL加密连接方式。SSL连接方式安全性高于普通模式，建议开启。	-

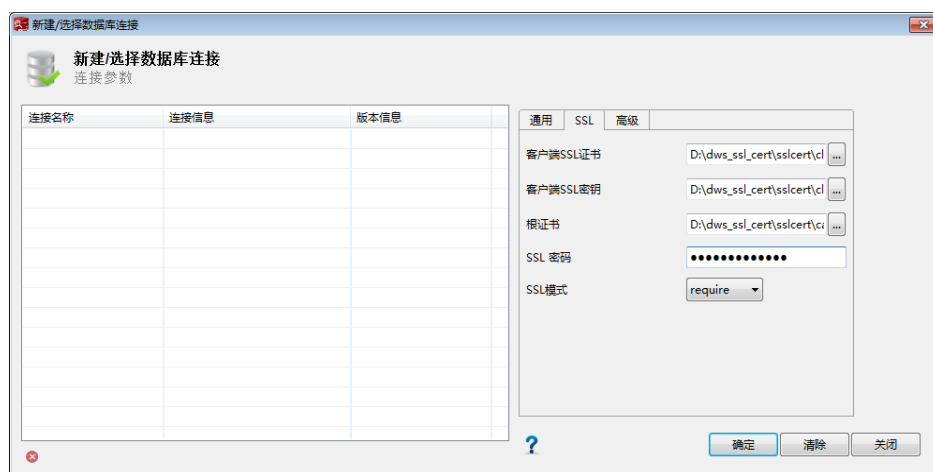
当“启用SSL”设置为开启时，请先[下载SSL证书](#)，并解压证书文件。然后单击“SSL”页签，设置如下参数：

表 6-3 配置 SSL 参数

字段名称	说明
客户端SSL证书	选择SSL证书解压目录下的“sslcert\client.crt”文件。
客户端SSL密钥	客户端SSL密钥只支持PK8格式，请选择SSL证书解压目录下的“sslcert\client.key.pk8”文件。
根证书	当“SSL模式”设为“verify-ca”时，必须设置根证书，请选择SSL证书解压目录下的“sslcert\cacert.pem”文件。
SSL密码	客户端pk8格式SSL密钥密码。

字段名称	说明
SSL模式	DWS支持的SSL模式有： ● require ● verify-ca DWS不支持“verify-full”模式。

图 6-3 配置 SSL 参数



步骤9 单击“确定”建立数据库连接。

如果启用了SSL，在弹出的“连接安全告警”提示对话框中单击“继续”。

登录成功后，将弹出“最近登录活动”提示框，表示Data Studio已经连接到数据库。用户即可在Data Studio界面的“SQL终端”窗口中执行SQL语句。

若需详细了解Data Studio其他功能的使用方法，请按“F1”查看Data Studio用户手册。

📖 说明

- Data Studio中执行增、删、改、查操作后不支持回滚数据。
- Data Studio可以保存连接信息，但是不包含密码。
- 以下对象不支持批量导出DDL/DDL和数据：
 - **导出DDL：**
连接、数据库、外表、序列、列、索引、约束、分区、函数/过程组、普通表组、视图组、模式组及系统表组。
 - **导出DDL和数据：**
连接、数据库、命名空间、外表、序列、列、索引、约束、分区、函数/过程、视图、普通表组、模式组及系统表组。

----结束

6.4 使用命令行工具连接 DWS 集群

6.4.1 下载客户端

DWS提供了与集群版本配套的客户端工具包，用户可以在DWS管理控制台下载客户端工具包。

客户端工具包包含以下内容：

- 数据库连接工具Linux gsql和测试样例数据的脚本**
Linux gsql是一款运行在Linux环境上的命令行客户端，用于连接DWS集群中的数据库。
测试样例数据的脚本用于执行入门示例。
- Windows版本gsql**
Windows gsql是一款运行在Windows环境上的命令行客户端，用于连接DWS集群中的数据库。

 说明

仅8.1.3.101及以上集群版本支持在控制台下载。

下载客户端

- 步骤1** 登录DWS管理控制台。
- 步骤2** 在左侧导航栏中，选择“管理 > 连接客户端”。
- 步骤3** 在“命令行客户端”的下拉列表中，选择对应版本的DWS客户端。

请根据集群版本和安装客户端的操作系统，选择对应版本。

- 步骤4** 单击“下载”可以下载与8.1.x集群版本匹配的gsql。单击“历史版本”可根据集群版本下载相应版本的gsql。
- 推荐下载使用与集群版本匹配的gsql工具，即8.1.0及以上版本集群使用8.1.x版本gsql、8.2.0及以上版本集群使用8.2.x版本gsql。
 - [表6-4](#)列出了下载的Linux gsql工具包中的文件和文件夹。

表 6-4 Linux gsql 工具包目录及文件说明

文件或文件夹	说明
bin	该文件夹中包含了gsql在Linux中的可执行文件。其中包含了gsql客户端工具、GDS并行数据加载工具以及gs_dump、gs_dumpall和gs_restore工具。详情请参见《数据仓库服务工具指南》中的“服务端工具”章节。
gds	该文件夹中包括了GDS数据服务工具的相关文件，GDS工具用于并行数据加载，可将存储在普通文件系统中的数据文件导入到DWS数据库中。
lib	该文件夹中包括执行gsql所需依赖的lib库。

文件或文件夹	说明
sample	该文件夹中包含了以下目录或文件： – setup.sh：在使用gsql导入样例数据前所需执行的配置AK/SK访问密钥的脚本文件。 – tpcds_load_data_from_obs.sql：使用gsql客户端导入TPC-DS样例数据的脚本文件。 – query_sql目录：查询TPC-DS样例数据的脚本文件。
gsql_env.sh	在运行gsql前，配置环境变量的脚本文件。

- **表6-5**列出了下载的Windows gsql工具包中的文件和文件夹。

表 6-5 Windows gsql 工具包目录及文件说明

文件或文件夹	说明
x64	该文件夹中包含了64位Windows gsql执行二进制和动态库。
x86	该文件夹中包含了32位Windows gsql执行二进制和动态库。

📖 说明

在“专属集群”页面的集群列表中，单击指定集群的名称，再选择“集群详情”页签，可查看集群版本。

----结束

6.4.2 使用 Linux gsql 客户端连接集群

用户在创建好数据仓库集群，开始使用集群数据库之前，需要使用数据库SQL客户端连接到数据库。DWS提供了与集群版本配套的Linux gsql命令行客户端工具，您可以使用Linux gsql客户端通过集群的公网地址或者内网地址访问集群。

它的运行环境是Linux操作系统，在使用Linux gsql客户端远程连接DWS集群之前，需要准备一个Linux主机用于安装和运行Linux gsql客户端。如果通过公网地址访问集群，也可以将Linux gsql客户端安装在用户自己的Linux主机上，但是该Linux主机必须具有公网地址。若DWS集群没有配置公网IP，为方便起见，推荐您创建一台Linux弹性云服务器（简称ECS），详情可参见（可选）准备ECS作为gsql客户端主机。

（可选）准备 ECS 作为 gsql 客户端主机

创建弹性云服务器的操作步骤，请参见《弹性云服务器用户指南》中的“快速入门 > 创建弹性云服务器”章节。

创建的弹性云服务器需要满足如下要求：

- 弹性云服务器需要与DWS集群在相同的区域、可用分区。
- 如果使用DWS提供的gsql命令行客户端连接DWS集群，弹性云服务器的镜像必须满足如下要求：
镜像的操作系统必须是gsql客户端所支持的下列Linux操作系统：
 - “Redhat x86_64” 客户端工具支持在以下系统中使用：
 - RHEL 6.4~7.6
 - CentOS 6.4~7.4
 - EulerOS 2.3
 - “SUSE x86_64” 客户端工具支持在以下系统中使用：
 - SLES 11.1~11.4
 - SLES 12.0~12.3
 - “Euler Kunpeng_64” 客户端工具支持在以下系统中使用：
 - EulerOS 2.8
- 如果客户端通过内网地址访问集群，请确保创建的弹性云服务器与DWS集群在同一虚拟私有云里。
虚拟私有云相关操作请参见《虚拟私有云用户指南》中“虚拟私有云和子网”。
- 如果客户端通过公网地址访问集群，请确保创建的弹性云服务器和DWS集群都要有弹性IP。
创建弹性云服务器时，参数“弹性IP”需设置为“自动分配”或“使用已有”。
- 弹性云服务器对应的安全组规则需要确保能与DWS集群提供服务的端口网络互通。
安全组相关操作请参见《虚拟私有云用户指南》中“安全组”章节。
请确认弹性云服务器的安全组中存在符合如下要求的规则，如果不存在，请在弹性云服务器的安全组中添加相应的规则：
 - 方向：出方向。
 - 协议：必须包含TCP。例如TCP、全部。
 - 端口：需要包含DWS集群提供服务的数据库端口，例如，设置为“1-65535”或者具体的DWS数据库端口。
 - 目的地址：设置的IP地址需要包含所要连接的DWS集群的连接地址。其中0.0.0.0/0表示任意地址。
- DWS集群的安全组规则需要确保DWS能接收来自客户端的网络访问。
请确认DWS集群的安全组中存在符合如下要求的规则，如果不存在，请在DWS集群的安全组中添加相应的规则。
 - 方向：入方向。
 - 协议：必须包含TCP。例如，TCP、全部。
 - 端口：设置为DWS集群提供服务的数据库端口。例如，“8000”。
 - 源地址：设置的IP地址需要包含DWS客户端主机的IP地址。例如，“192.168.0.10/32”。

下载 Linux gsql 客户端并连接集群

步骤1 请参见[下载客户端](#)下载Linux gsql客户端，并使用SSH文件传输工具（例如WinSCP工具），将客户端工具上传到一个待安装Linux gsql的Linux主机上。

推荐下载使用与集群版本匹配的gsql工具，即8.1.0及以上版本集群使用8.1.x版本gsql、8.2.0及以上版本集群使用8.2.x版。若下载8.2.x版本gsql工具，需将dws_client_8.1.x_redhat_x64.zip替换为dws_client_8.2.x_redhat_x64.zip。此处仅以dws_client_8.1.x_redhat_x64.zip作为示例。

执行上传Linux gsql操作的用户需要对客户端主机的目标存放目录有完全控制权限。

步骤2 使用SSH会话工具，远程管理客户端主机。

弹性云服务器的登录方法请参见《弹性云服务器用户指南》中的“实例 > 登录Linux弹性云服务器 > SSH密码方式登录”章节。

步骤3 （可选）如果要使用SSL方式连接集群，请参考[使用SSL进行安全的TCP/IP连接](#)章节，在客户端主机配置SSL认证相关的参数。

说明

SSL连接方式的安全性高于非SSL方式，建议在客户端使用SSL连接方式。

步骤4 执行以下命令解压客户端工具。

```
cd <客户端存放路径>
unzip dws_client_8.1.x_redhat_x64.zip
```

其中：

- <客户端存放路径>：请替换为实际的客户端存放路径。
- dws_client_8.1.x_redhat_x64.zip：“RedHat x64”对应的客户端工具包名称，请替换为实际下载的包名。

步骤5 执行以下命令配置客户端。

```
source gsql_env.sh
```

提示以下信息表示客户端已配置成功。

```
All things done.
```

步骤6 执行以下命令，使用gsql客户端连接DWS集群中的数据库。

```
gsql -d <数据库名称> -h <集群地址> -U <数据库用户> -p <数据库端口> -W <集群密码> -r
```

参数说明如下：

- “数据库名称”：输入所要连接的数据库名称。首次使用客户端连接集群时，请指定为集群的默认数据库“gaussdb”。
- “集群地址”：请参见[获取DWS集群连接地址](#)进行获取。如果通过公网地址连接，请指定为集群“公网访问地址”，如果通过内网地址连接，请指定为集群“内网访问地址”。
- “数据库用户”：输入集群数据库的用户名。首次使用客户端连接集群时，请指定为创建集群时设置的默认管理员用户，例如“dbadmin”。
- “数据库端口”：输入创建集群时设置的“数据库端口”。

例如，执行以下命令连接DWS集群的默认数据库gaussdb：

```
gsql -d gaussdb -h 10.168.0.74 -U dbadmin -p 8000 -W password -r
```

显示如下信息表示gsql工具已经连接成功：

```
gaussdb=>
```

----结束

gsql 命令参考

有关gsql的命令参考和更多信息，请参见《数据仓库服务工具指南》。

（可选）使用 gsql 导入 TPC-DS 样例数据

DWS支持用户将数据从集群外导入到集群中。用户可以参考以下指导，快速将样例数据从OBS导入集群，并对样例数据进行查询和分析。导入的样例数据是使用TPC-DS测试基准生成的标准性能测试数据。

TPC-DS是数据库决策支持测试基准。通过使用TPC-DS的测试数据以及测试案例，用户可以模拟真实场景下大数据集的统计、报表生成、联机查询、数据挖掘等复杂场景，从而了解数据库应用的功能和性能。

步骤1 使用SSH远程连接工具登录gsql客户端主机，并进入gsql目录，本例假设gsql客户端放在/opt目录下。

```
cd /opt
```

步骤2 执行以下命令，切换到指定目录并设置用户导入样例数据的用户密钥和OBS访问地址。

系统显示以下信息表示设置成功：

```
setup successfully!
```

说明

<Access_Key_Id>和<Secret_Access_Key>：分别表示访问密钥ID和私有访问密钥。请参见《数据仓库服务数据库开发指南》中的“导入数据 > 从OBS并行导入数据 > 创建访问密钥（AK和SK）”章节进行获取。然后将获取到的值替换到创建外表语句中。

步骤3 返回上一级目录，执行gsql环境变量。

```
cd ..  
source gsql_env.sh  
cd bin
```

步骤4 执行以下命令，将样例数据导入数据仓库。

命令格式：

```
gsql -d <数据库名称> -h <集群公网访问地址> -U <管理员用户> -p <数据仓库端口> -f <样例数据脚本保存路径> -r
```

命令示例：

```
gsql -d gaussdb -h 10.168.0.74 -U dbadmin -p 8000 -f /opt/sample/tpcds_load_data_from_obs.sql -r
```

说明

命令中样例数据脚本“tpcds_load_data_from_obs.sql”存放在DWS客户端的sample目录下，如“/opt/sample/”。

根据界面提示输入管理员密码，成功连接集群数据库后，系统会自动创建样例数据对应的外表用于关联集群外的数据，然后再创建存放样例数据的目标表，最后通过外表将数据导入到目标表中。

由于数据集较大，导入时间取决于当前DWS集群规格，一般为10~20分钟左右，等待系统显示如下执行时间信息表示导入成功，如下时间仅为示例。

Time:1845600.524 ms

步骤5 在Linux命令窗口，执行以下命令，切换到指定目录并查询样例数据。

```
cd /opt/sample/query_sql/  
/bin/bash tpcds100x.sh
```

步骤6 根据命令提示，输入集群公网访问地址的IP地址、数据库端口、数据库名称、数据库访问用户以及用户密码。

- 数据库名称默认为“gaussdb”。
- 数据库访问用户和密码使用创建集群时配置的管理员用户和密码。

查询完成后，在当前查询目录，如“sample/query_sql/”下面会生成一个存放查询结果的目录，命名如“query_output_20170914_072341”。

----结束

6.4.3 使用 Windows gsql 客户端连接集群

用户在创建好数据仓库集群，开始使用集群数据库之前，需要使用数据库SQL客户端连接到数据库。DWS提供了与集群版本配套的Windows gsql命令行客户端工具，您可以使用Windows gsql客户端通过集群的公网地址或者内网地址访问集群。

操作步骤

步骤1 在计算机本地Windows操作系统服务器（Windows cmd）中安装和运行gsq客户端。Windows操作系统支持Windows Server 2008/Windows 7及以上。

步骤2 请参见[下载客户端](#)下载Windows gsql客户端，并将压缩包解压到本地文件夹中。

步骤3 在本地主机单击“开始”并搜索“cmd”用管理员身份运行或单击快捷键“Win+R”打开Windows cmd窗口。

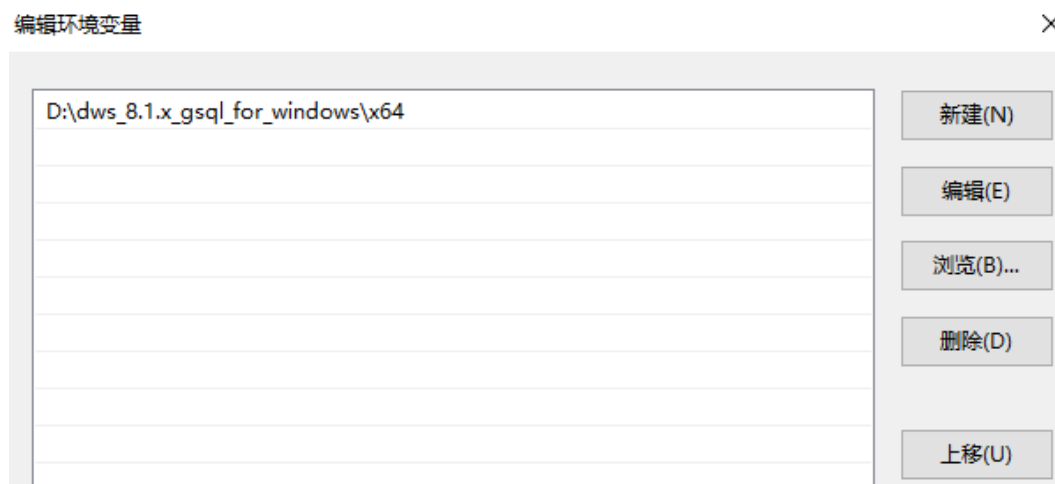
步骤4 设置环境变量，32位选择x86文件夹；64位选择x64文件夹。

方式一：命令行设置环境变量，打开Windows cmd窗口，执行set path=<window gsql>%path%，其中<window gsql>为上一步骤解压Windows gsql客户端的文件夹路径。例如：

```
set path=C:\Users\xx\Desktop\dws_8.1.x_gsql_for_windows\x64;%path%
```

方式二：在控制面板中选择“系统 > 高级系统设置 > 高级 > 环境变量”，在系统环境变量Path中增加gsq路径。例如：

图 6-4 设置 Windows 环境变量



步骤5 （可选）如果要使用SSL方式连接集群，请参考[使用SSL进行安全的TCP/IP连接](#)章节，在客户端主机配置SSL认证相关的参数。

说明

SSL连接方式的安全性高于非SSL方式，建议在客户端使用SSL连接方式。

步骤6 在Windows cmd窗口执行以下命令，使用gsq客户端连接DWS集群中的数据库。

```
gsq -d <数据库名称> -h <集群地址> -U <数据库用户> -p <数据库端口> -W <集群密码> -r
```

参数说明如下：

- “数据库名称”：输入所要连接的数据库名称。首次使用客户端连接集群时，请指定为集群的默认数据库“gaussdb”。
- “集群地址”：请参见[获取DWS集群连接地址](#)进行获取。如果通过公网地址连接，请指定为集群“公网访问域名”，如果通过内网地址连接，请指定为集群“内网访问域名”。
- “数据库用户”：输入集群数据库的用户名。首次使用客户端连接集群时，请指定为创建集群时设置的默认管理员用户，例如“dbadmin”。
- “数据库端口”：输入创建集群时设置的“数据库端口”。

例如，执行以下命令连接DWS集群的默认数据库gaussdb：

```
gsq -d gaussdb -h 10.168.0.74 -U dbadmin -p 8000 -W password -r
```

显示如下信息表示gsq工具已经连接成功：

```
gaussdb=>
```

---结束

注意事项

- Windows cmd默认的字符集是GBK，所以Windows gsql默认的client_encoding为GBK，部分UTF-8编码的字符无法在Windows gsql中显示。
建议：-f执行的文件使用UTF-8编码，并设置默认的编码格式为UTF-8（set client_encoding='utf-8'；）
- Windows gsql中的路径需要使用‘/’作为分隔符，否则会报错。因为在元命令中‘\’是作为元命令开始的标志，在一般的单引号中，‘\’起转义作用。

```
gaussdb=> \i D:\test.sql
D:: Permission denied
postgres=> \i D:\test.sql
id
----
1
(1 row)
```
- Windows gsql使用‘\!’元命令执行系统命令时，需要使用系统命令要求的路径分隔符，一般是‘\’。

```
gaussdb=> \! type D:\test.sql
命令语法不正确。
gaussdb=> \! type D:\test.sql
select 1 as id;
```
- Windows gsql不支持元命令‘\parallel’。

```
gaussdb=> \parallel
ERROR: "\parallel" is not supported in Windows.
```
- Linux shell中可以使用单引号和双引号作为字符串边界，但在Windows必须使用双引号作为字符串边界。

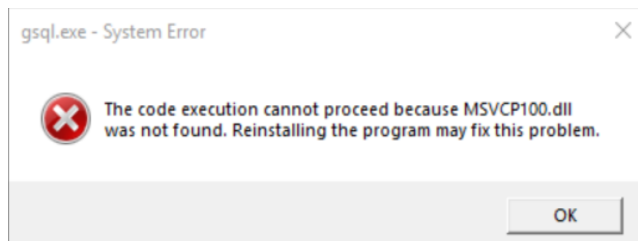
```
gsq -h 192.168.233.189 -p 8109 -d postgres -U odbcuser -W password -c "select 1 as id"
id
```

- ```

1
(1 row)
使用单引号时报错，并忽略输入。
gsql -h 192.168.233.189 -p 8109 -d postgres -U odbcuser -W password -c 'select 1 as id'
gsql: warning: extra command-line argument "1" ignored
gsql: warning: extra command-line argument "as" ignored
gsql: warning: extra command-line argument "id" ignored
ERROR: unterminated quoted string at or near "'select"
LINE 1: 'select

```
- Windows gsql在建立连接之后长时间未使用，连接session超时，会出现SSL报错，需要重新登录。报错如下：  
SSL SYSCALL error: Software caused connection abort (0x00002745/10053), remote datanode <NULL>, error: Result too large
  - Windows下Ctrl+C退出gsql。在当前行输入SQL语句时，若捕获到Ctrl+C信号后，无法将状态调整到重新输入的状态，会按照当前没有输入处理，将直接退出gsql。  
在输入as后执行Ctrl+C，输出\q后退出gsql。  
gaussdb=> select 1  
gaussdb=> as \q
  - Windows gsql不支持连接字符集为LATIN1的数据库，报错信息为：  
gsql: FATAL: conversion between GBK and LATIN1 is not supported
  - gsqlrc.conf文件的位置。  
默认的gsqlrc路径为%APPDATA%/postgresql/gsqlrc.conf，也可通过PSQLRC变量设置。  
set PSQLRC=C:\Users\xx\Desktop\dws\_8.1.x\_gsql\_for\_windows\x64\gsqlrc.conf
  - Windows Server系统中可能缺少MSVCP100.dll，使用gsql报错信息为：

图 6-5 报错信息



解决方法：需要补充MSVCP100.dll动态链接库文件。可下载C++可再发行程序包，安装对应包“**vcredist\_x86.exe/vcredist\_x64.exe**”，即可补充所需的动态链接库文件。

## gsql 命令参考

有关gsql的命令参考和更多信息，请参见《数据仓库服务工具指南》。

### 6.4.4 使用 SSL 进行安全的 TCP/IP 连接

DWS支持SSL标准协议，SSL协议是安全性更高的协议标准，它们加入了数字签名和数字证书来实现客户端和服务器的双向身份验证，保证了通信双方更加安全的数据传输。为支持SSL连接方式，DWS已经从CA认证中心申请到正式的服务器、客户端的证书和私钥（假设服务器的私钥为server.key，证书为server.crt，客户端的私钥为client.key，证书为client.crt，CA根证书名称为cacert.pem）。

SSL连接方式的安全性高于普通模式，集群默认开启SSL功能允许来自客户端的SSL连接或非SSL连接，从安全性考虑，建议用户在客户端使用SSL连接方式。并且DWS服务器

端的证书、私钥以及根证书已经默认配置完成。如果要强制使用SSL连接，需要在集群“安全设置”页面开启“服务器端是否强制使用SSL连接”，操作详情可参见[设置SSL连接](#)，客户端和服务端SSL连接参数组合情况可参见[客户端和服务端SSL连接参数组合情况](#)。

客户端或JDBC/ODBC应用程序使用SSL连接方式，用户必须在客户端或应用程序代码中配置相关的SSL连接参数。DWS管理控制台提供了客户端所需的SSL证书，该SSL证书包含了客户端所需的默认证书、私钥、根证书以及私钥密码加密文件。请将该SSL证书下载到客户端所在的主机上，然后在客户端中指定证书所在的路径，操作详情请参见[在gsq客户端配置SSL认证相关的数字证书参数](#)，SSL认证及客户端参数介绍可参见[SSL认证方式及客户端参数介绍](#)。

#### 说明

使用默认的证书可能存在安全风险，为了提高系统安全性，强烈建议用户定期更换证书以避免被破解的风险。如果需要更换证书，请联系数据库客服。

## 设置 SSL 连接

### 前提条件

- 修改安全配置参数并保存后，生效可能需要重启集群，否则将导致集群暂时不可用。
- 修改集群安全配置必须同时满足以下两个条件：
  - 集群状态为“可用”或“非均衡”。
  - 任务信息不能处于“创建快照中”、“节点扩容”、“配置中”或“重启中”。

### 操作步骤

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航树中，单击“专属集群 > 集群列表”。

**步骤3** 在集群列表中，单击指定集群的名称，然后单击“安全设置”。

默认显示“配置状态”为“已同步”，表示页面显示的是数据库当前最新结果。

**步骤4** 在“SSL连接”区域中，单击“服务器端是否强制使用SSL连接”的设置开关进行设置，建议开启。



：开启，设置参数`require_ssl=1`，表示服务器端强制要求SSL连接。



：关闭，设置参数`require_ssl=0`，表示服务器端对是否通过SSL连接不作强制要求，默认为关闭。设置`require_ssl`参数详情请参见[require\\_ssl（服务器）](#)。

#### 说明

- 如果使用DWS提供的gsq客户端或ODBC驱动，DWS支持的SSL协议为TLSv1.2。
- 如果使用DWS提供的JDBC驱动，支持的SSL协议有SSLv3、TLSv1、TLSv1.1、TLSv1.2。客户端与数据库之间实际使用何种SSL协议，依赖客户端使用的JDK（Java Development Kit）版本，一般JDK支持多个SSL协议。

**步骤5** 单击“应用”。

系统将自动应用保存SSL连接设置，在“安全设置”页面，“配置状态”显示“应用中”。当“配置状态”显示为“已同步”，表示配置已保存生效。

----结束

## 在 gsql 客户端配置 SSL 认证相关的数字证书参数

DWS在集群部署完成后，默认已开启SSL认证模式。服务器端证书，私钥以及根证书已经默认配置完成。用户需要配置客户端的相关参数。

**步骤1** 登录DWS管理控制台，在左侧导航栏中，进入“连接客户端”页面。

**步骤2** 在“下载驱动程序”区域，单击“下载SSL证书”进行下载。

**步骤3** 使用文件传输工具（例如WinSCP工具）将SSL证书上传到客户端主机。

例如，将下载的证书“dws\_ssl\_cert.zip”存放到“/home/dbadmin/dws\_ssl/”目录下。

**步骤4** 使用SSH远程连接工具（例如PuTTY）登录gsql客户端主机，然后执行以下命令进入SSL证书的存放目录，并解压SSL证书：

```
cd /home/dbadmin/dws_ssl/
unzip dws_ssl_cert.zip
```

**步骤5** 在gsql客户端主机上，执行export命令，配置SSL认证相关的数字证书参数。

SSL认证有两种认证方式：双向认证和单向认证。认证方式不同用户所需配置的客户端环境变量也不同，详细介绍请参见[SSL认证方式及客户端参数介绍](#)。

双向认证需配置如下参数：

```
export PGSSLCERT="/home/dbadmin/dws_ssl/sslcert/client.crt"
export PGSSLKEY="/home/dbadmin/dws_ssl/sslcert/client.key"
export PGSSLMODE="verify-ca"
export PGSSLROOTCERT="/home/dbadmin/dws_ssl/sslcert/cacert.pem"
```

单向认证需要配置如下参数：

```
export PGSSLMODE="verify-ca"
export PGSSLROOTCERT="/home/dbadmin/dws_ssl/sslcert/cacert.pem"
```

### 须知

- 从安全性考虑，建议使用双向认证方式。
- 配置客户端环境变量，必须包含文件的绝对路径。

**步骤6** 修改客户端密钥的权限。

客户端根证书、密钥、证书以及密钥密码加密文件需保证权限为600。如果权限不满足要求，则客户端无法以SSL方式连接到集群。

```
chmod 600 client.key
chmod 600 client.crt
chmod 600 client.key.cipher
chmod 600 client.key.rand
chmod 600 cacert.pem
```

----结束

### SSL 认证方式及客户端参数介绍

SSL认证有两种认证方式，如表6-6所示。从安全性考虑，建议使用双向认证方式。

表 6-6 认证方式

| 认证方式     | 含义                                                                        | 配置客户端环境变量                                                                                                                    | 维护建议                                                                       |
|----------|---------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------|
| 双向认证（推荐） | 客户端验证服务器证书的有效性，同时服务器端也要验证客户端证书的有效性，只有认证成功，连接才能建立。                         | 设置如下环境变量： <ul style="list-style-type: none"><li>PGSSLCERT</li><li>PGSSLKEY</li><li>PGSSLROOTCERT</li><li>PGSSLMODE</li></ul> | 该方式应用于安全性要求较高的场景。使用此方式时，建议设置客户端的PGSSLMODE变量为verify-ca。确保了网络数据的安全性。         |
| 单向认证     | 客户端只验证服务器证书的有效性，而服务器端不验证客户端证书的有效性。服务器加载证书信息并发送给客户端，客户端使用根证书来验证服务器端证书的有效性。 | 设置如下环境变量： <ul style="list-style-type: none"><li>PGSSLROOTCERT</li><li>PGSSLMODE</li></ul>                                    | 为防止基于TCP链接的安全攻击，建议使用SSL证书认证功能。除配置客户端根证书外，建议客户端使用PGSSLMODE变量为verify-ca方式连接。 |

在客户端配置SSL认证相关的环境变量，详细信息请参见表6-7。

 说明

客户端环境变量的路径以“/home/dbadmin/dws\_ssl/”为例，在实际操作中请使用实际路径进行替换。

表 6-7 客户端参数

| 环境变量      | 描述                                                       | 取值说明                                                                                       |
|-----------|----------------------------------------------------------|--------------------------------------------------------------------------------------------|
| PGSSLCERT | 指定客户端证书文件，包含客户端的公钥。客户端证书用于表明客户端身份的合法性，公钥将发送给对端用来对数据进行加密。 | 必须包含文件的绝对路径，如：<br>export PGSSLCERT='/home/dbadmin/dws_ssl/sslcert/client.crt'<br><br>默认值：空 |
| PGSSLKEY  | 指定客户端私钥文件，用于数字签名和对公钥加密的数据进行解密。                           | 必须包含文件的绝对路径，如：<br>export PGSSLKEY='/home/dbadmin/dws_ssl/sslcert/client.key'<br><br>默认值：空  |

| 环境变量          | 描述                                                  | 取值说明                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|---------------|-----------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| PGSSLMODE     | 设置是否和服务器进行SSL连接协商，以及指定SSL连接的优先级。                    | <p><b>取值及含义：</b></p> <ul style="list-style-type: none"> <li>disable：只尝试非SSL连接。</li> <li>allow：首先尝试非SSL连接，如果连接失败，再尝试SSL连接。</li> <li>prefer：首先尝试SSL连接，如果连接失败，将尝试非SSL连接。</li> <li>require：只尝试SSL连接。如果存在CA文件，则按设置成verify-ca的方式验证。</li> <li>verify-ca：只尝试SSL连接，并且验证服务器是否具有由可信任的证书机构签发的证书。</li> <li>verify-full：DWS不支持此模式。</li> </ul> <p><b>默认值：</b> prefer</p> <p><b>说明</b><br/>若集群外访问客户端时，部分节点出现报错：ssl SYSCALL error。则可执行export PGSSLMODE="allow"或export PGSSLMODE="prefer"。</p> |
| PGSSLROOTCERT | 指定为客户端颁发证书的根证书文件，根证书用于验证服务器证书的有效性。                  | <p>必须包含文件的绝对路径，如：<br/>export PGSSLROOTCERT='/home/dbadmin/dws_ssl/sslcert/certca.pem'</p> <p><b>默认值：</b> 空</p>                                                                                                                                                                                                                                                                                                                                                               |
| PGSSLCRL      | 指定证书吊销列表文件，用于验证服务器证书是否在废弃证书列表中，如果在，则服务器证书将会被视为无效证书。 | <p>必须包含文件的绝对路径，如：<br/>export PGSSLCRL='/home/dbadmin/dws_ssl/sslcert/sslcrfile.crt'</p> <p><b>默认值：</b> 空</p>                                                                                                                                                                                                                                                                                                                                                                 |

## 客户端和服务端 SSL 连接参数组合情况

客户端最终是否使用SSL加密连接方式、是否验证服务器证书，取决于客户端参数 sslmode与服务端（即DWS集群侧）参数ssl、require\_ssl。参数说明如下：

- ssl（服务器）**  
 ssl参数表示是否开启SSL功能。on表示开启，off表示关闭。
  - 默认认为on，不支持在DWS管理控制台上设置。
- require\_ssl（服务器）**  
 require\_ssl参数是设置服务器端是否强制要求SSL连接，该参数只有当ssl为on时才有效。on表示服务器端强制要求SSL连接。off表示服务器端对是否通过SSL连接不作强制要求。
  - 默认认为off。require\_ssl参数可通过DWS管理控制台上集群的“安全设置”页面中的“服务器端是否强制使用SSL连接”进行设置。

- **sslmode（客户端）**  
可在SQL客户端工具中进行设置。
  - 在gsq命令客户端中，为“PGSSLMODE”参数。
  - 在Data Studio客户端中，为“SSL模式”参数。

客户端参数sslmode与服务器端参数ssl、require\_ssl配置组合结果如下：

表 6-8 客户端与服务器端 SSL 参数组合结果

| ssl<br>（服务器） | sslmode<br>（客户端） | require_ssl<br>（服务器） | 结果                                      |
|--------------|------------------|----------------------|-----------------------------------------|
| on           | disable          | on                   | 由于服务器端要求使用SSL，但客户端针对该连接禁用了SSL，因此无法建立连接。 |
|              | disable          | off                  | 连接未加密。                                  |
|              | allow            | on                   | 连接经过加密。                                 |
|              | allow            | off                  | 连接未加密。                                  |
|              | prefer           | on                   | 连接经过加密。                                 |
|              | prefer           | off                  | 连接经过加密。                                 |
|              | require          | on                   | 连接经过加密。                                 |
|              | require          | off                  | 连接经过加密。                                 |
|              | verify-ca        | on                   | 连接经过加密，且验证了服务器证书。                       |
|              | verify-ca        | off                  | 连接经过加密，且验证了服务器证书。                       |
| off          | disable          | on                   | 连接未加密。                                  |
|              | disable          | off                  | 连接未加密。                                  |
|              | allow            | on                   | 连接未加密。                                  |
|              | allow            | off                  | 连接未加密。                                  |
|              | prefer           | on                   | 连接未加密。                                  |
|              | prefer           | off                  | 连接未加密。                                  |
|              | require          | on                   | 由于客户端要求使用SSL，但服务器端禁用了SSL，因此无法建立连接。      |
|              | require          | off                  | 由于客户端要求使用SSL，但服务器端禁用了SSL，因此无法建立连接。      |
|              | verify-ca        | on                   | 由于客户端要求使用SSL，但服务器端禁用了SSL，因此无法建立连接。      |
|              | verify-ca        | off                  | 由于客户端要求使用SSL，但服务器端禁用了SSL，因此无法建立连接。      |

## 6.5 使用第三方库连接 DWS 集群

### 6.5.1 使用 JDBC 和 ODBC 连接 DWS 集群

#### 6.5.1.1 开发规范

如果用户在APP的开发中，使用了连接池机制，那么需要遵循如下规范。否则，连接池里的连接就是有状态的，会对用户后续使用连接池进行操作的正确性带来影响。

- 如果在连接中设置了GUC参数，那么在将连接归还连接池之前，必须使用“SET SESSION AUTHORIZATION DEFAULT;RESET ALL;”将连接的状态清空。
- 如果使用了临时表，那么在将连接归还连接池之前，必须将临时表删除。

#### 6.5.1.2 下载 JDBC 或 ODBC 驱动

JDBC或ODBC驱动程序用于连接DWS集群，用户可以在管理控制台下载DWS提供的JDBC或ODBC驱动程序，也可以使用开源的JDBC或ODBC驱动程序。

#### 支持的开源 JDBC 或 ODBC 驱动程序

DWS也支持开源的JDBC驱动程序：PostgreSQL JDBC驱动程序9.3-1103或更高版本。

DWS也支持开源的ODBC驱动程序：PostgreSQL ODBC 09.01.0200或更高版本。

#### 下载 JDBC 或 ODBC 驱动程序

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，单击“连接客户端”。

**步骤3** 在“下载驱动程序”区域，选择一个驱动下载。

- **JDBC驱动**

选择“DWS JDBC Driver”，然后单击“下载”可以下载与现有集群版本匹配的JDBC驱动。驱动包名为“dws\_8.1.x\_jdbc\_driver.zip”。

如果同时拥有不同版本的集群，单击“下载”时会下载与集群最低版本相对应的JDBC驱动。如果当前没有集群，单击“下载”时将下载到低版本的JDBC驱动。DWS集群可向下兼容低版本的JDBC驱动。

单击“历史版本”可根据集群版本下载相应版本的JDBC驱动，建议按集群版本进行下载。

JDBC驱动包支持在所有平台所有版本中使用，且依赖JDK 1.6及以上版本。

- **ODBC驱动**

选择相应的版本，然后单击“下载”可以下载与集群版本匹配的ODBC驱动。

单击“历史版本”可根据操作系统和集群版本下载相应版本的ODBC驱动，建议按集群版本进行下载。

 说明

ODBC驱动不兼容Window server 2016系统。

----结束

### 6.5.1.3 使用 JDBC 连接

DWS支持在Linux或Windows环境下使用JDBC应用程序连接数据库。应用程序可以在华为云平台环境的弹性云服务器中，或者互联网环境连接数据库。

用户通过JDBC连接DWS集群时，可以选择是否采用SSL认证方式。SSL认证用于加密客户端和服务端之间的通讯数据，为敏感数据在互联网上的传输提供了一种安全保障手段。DWS管理控制台提供了自签的证书供用户下载。使用该证书，用户需要配置客户端程序，使证书可用，此过程依赖于openssl工具以及java自带的keytool工具。

 说明

SSL模式安全性高于普通模式，建议在使用JDBC连接DWS集群时采用SSL模式。

JDBC接口的使用方法，请自行查阅官方文档。

### 前提条件

- 已安装JDK 1.6或以上版本，并配置环境变量。
- 已下载JDBC驱动，请参见[下载JDBC或ODBC驱动](#)。  
DWS也支持开源的JDBC驱动程序：PostgreSQL JDBC驱动程序9.3-1103或更高版本。
- 已下载SSL证书文件，请参见[下载SSL证书](#)。

### 使用 JDBC 连接数据库

在Linux和Windows环境下操作方法相同，以下步骤以Windows环境为例。

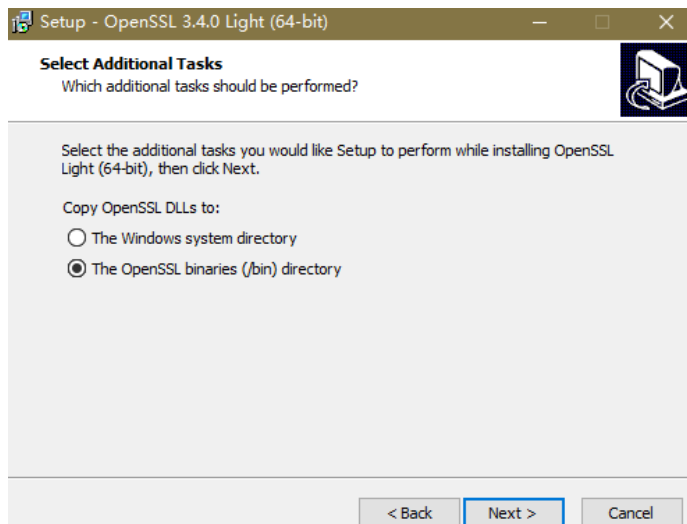
**步骤1** 是否采用SSL方式连接DWS集群。

- 是，参见[设置SSL连接](#)开启SSL连接，默认为开启。执行[步骤2](#)。
- 否，参见[设置SSL连接](#)关闭SSL连接，执行[步骤4](#)。

**步骤2** （可选）如果使用Linux环境，使用WinScp工具将SSL证书上传到Linux环境上。

**步骤3** 配置证书以使用SSL加密连接。

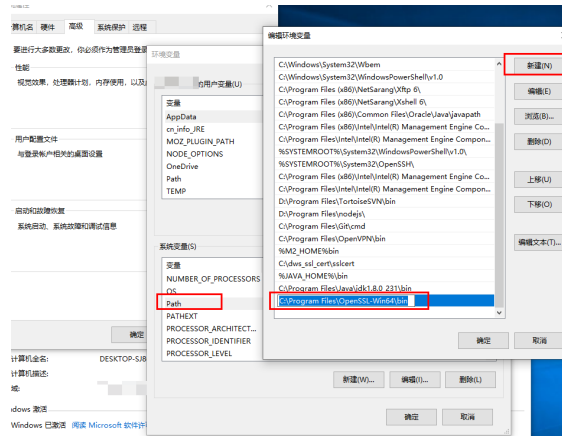
1. 下载windows版本的OpenSSL工具。下载地址：<https://slproweb.com/products/Win32OpenSSL.html>，最新稳定版本是3.4系列，所有旧版本（包括1.1.1、1.1.0、1.0.2、1.0.0和0.9.8）现在不受支持，不应使用。请选择下载“Win64 OpenSSL v3.4.0 Light”版本。
2. 双击安装包“Win64OpenSSL\_Light-3.4.0.exe”安装到C盘默认路径即可，选择复制DLLs文件到OpenSSL目录下，如下图，剩余步骤默认单击下一步直到安装成功。



3. 安装环境变量。单击本地PC左下角“开始”，右击“此电脑”，选择“更多 > 属性 > 高级系统设置”，切换到“高级”，单击“环境变量”。



4. 在下方“系统变量”，双击“Path”变量，单击“新建”，在最后一行增加 OpenSSL的bin路径，例如“C:\Program Files\OpenSSL-Win64\bin”，单击“确定”，再次单击“确定”，配置变量成功。



5. 解压压缩包得到证书文件。假设解压路径为“C:\”。

证书文件建议保存在英文路径，配置证书时需要根据实际情况指定路径，否则可能提示文件不存在。

6. 打开“命令提示符”程序，切换到“C:\dws\_ssl\_cert\sslcert”路径下，执行以下两条命令，转化根证书并导入到trustStore中。

```
openssl x509 -in cacert.pem -out cacert.crt.der -outform der
keytool -keystore mytruststore -alias cacert -import -file cacert.crt.der
```

- “cacert.pem”为解压后获取的根证书。
- “cacert.crt.der”为生成的中间文件。根据实际情况，可以填写保存到其他路径，也可以自定义文件名称。
- “mytruststore”为生成的可信库名称，“cacert”为别名用户，二者可以根据需要进行修改。

请用户根据提示信息输入自定义的可信库密码并确认密码，然后输入“y”确认信任证书。

```
C:\dws_ssl_cert\sslcert>keytool -keystore mytruststore -alias cacert -import -file cacert.crt.der
输入密钥库口令:
再次输入新口令:
所有者: CN=Huawei Equipment CA, O=Huawei, C=CN
发布者: CN=Huawei Equipment CA, O=Huawei, C=CN
序列号: 762560022a475eeca62b7c68ac9b3f2986
有效期为 Tue Dec 06 15:34:23 CST 2011 至 Thu Nov 28 15:34:23 CST 2041
证书指纹:
MD5: 2F:3F:BB:BC:93:47:0C:5F:92:D9:75:7E:3A:8A:2E:96
SHA1: 60:B2:B7:BE:BF:2F:83:2C:59:6E:FF:75:CD:4C:F8:2B:19:B0:79:04
SHA256: DA:2A:BE:92:25:D7:95:A3:80:E2:89:9D:53:0E:F0:31:8A:5B:32:BC:81:AA:FD:44:88:BA:96:1D:DF:75:02:D4
签名算法名称: SHA256withRSA
主体公共密钥算法: 4096 位 RSA 密钥
版本: 3
扩展:
#1: ObjectId: 2.5.29.19 Criticality=false
BasicConstraints:[
CA:true
PathLen:2147483647
]
#2: ObjectId: 2.5.29.15 Criticality=false
KeyUsage [
Key_CertSign
Crl_Sign
]
#3: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: 2A F8 10 59 27 80 35 1F A7 7C BA 3B 9F 2A E4 4A *.Y'.5.....*.J
0010: AA 9B 92 EA
]
]
是否信任此证书? [否]: y
证书已添加到密钥库中
```

7. 执行以下命令转化客户端私钥。

```
openssl pkcs12 -export -out client.pkcs12 -in client.crt -inkey client.key
```

需要输入客户端私钥的密码“Gauss@MppDB”，然后输入并确认用户自定义的私钥密码。

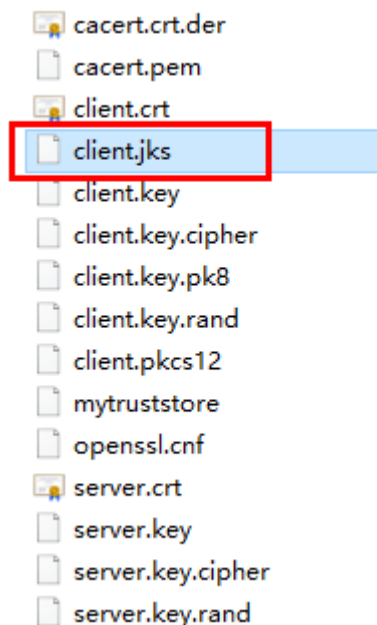
8. 执行以下命令，将私钥导入到keyStore中。

```
keytool -importkeystore -deststorepass Gauss@MppDB -destkeystore client.jks -srckeystore client.pkcs12 -srcstorepass password -srcstoretype PKCS12 -alias 1
```

## 说明

- 命令中 “password” 为示例自定义密码，具体请根据用户实际输入密码为准。
- 回显如下类似信息且没有报错，则表示导入成功。此时 “C:\dws\_ssl\_cert\sslcert” 下会生成目标密钥文件：client.jks。

```
C:\dws_ssl_cert\sslcert>keytool -importkeystore -deststorepass ***** -destkeystore client.jks -srcstore client.pkcs12 -srcstorepass ***** -srcstoretype PKCS12 -alias 1
正在将密钥库 client.pkcs12 导入到 client.jks...
```

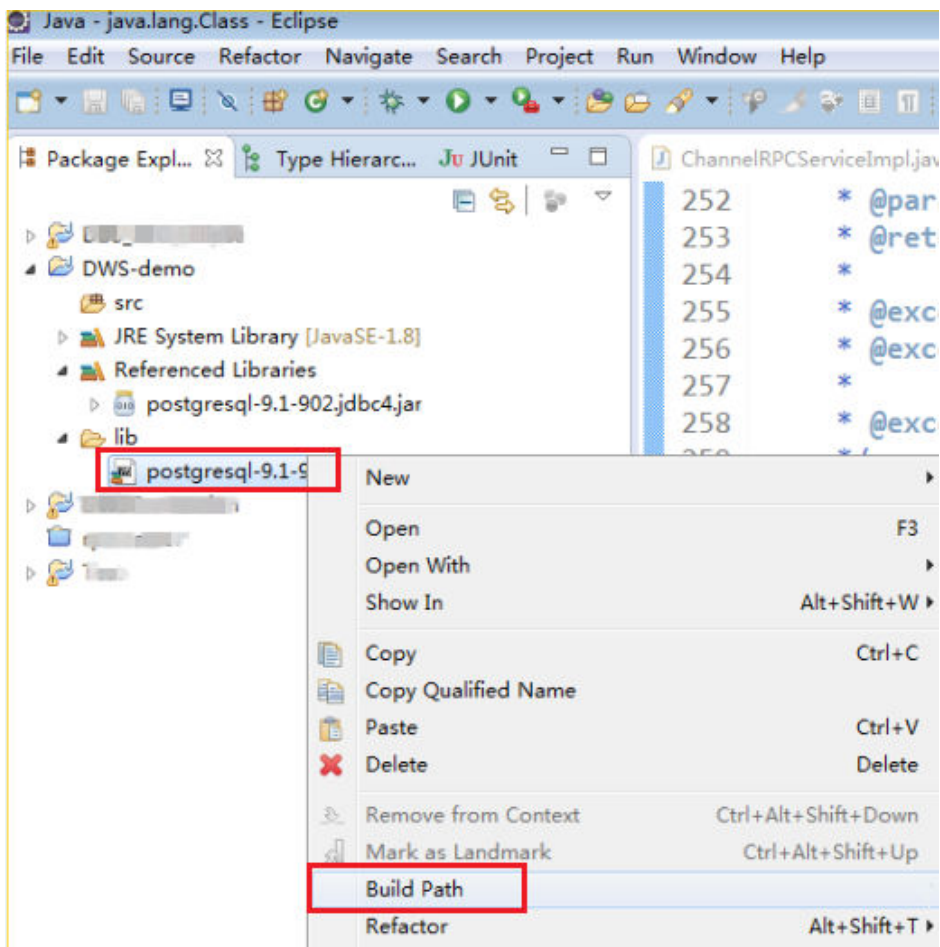


**步骤4** 下载包名为dws\_8.1.x\_jdbc\_driver.zip的驱动包，解压后有两个JDBC的驱动jar包“gsjdbc4.jar”和“gsjdbc200.jar”，用户可根据需求选择。

**步骤5** 在应用程序的工程中，设置引用Jar包。

以Eclipse工程为例，先将jar包存放在工程目录下，例如将jar包放在工程目录的lib目录下，然后在Eclipse工程中，右键单击lib目录下的该jar包，再选择菜单“Build Path”，即可引用此jar包。

图 6-6 引用 jar 包



#### 步骤6 加载驱动。

支持以下两种方法加载：

- 在代码中隐含装载：Class.forName("org.postgresql.Driver");
- 在JVM启动时参数传递：java -Djdbc.drivers=org.postgresql.Driver jdbcctest

#### 说明

DWS下载的JDBC驱动包中提供gsjdbc.jar。

- gsjdbc4.jar：与PostgreSQL保持兼容，其中类名、类结构与PostgreSQL驱动完全一致，曾经运行于PostgreSQL的应用程序可以直接移植到当前系统中使用。

#### 步骤7 调用JDBC的数据库连接方法DriverManager.getConnection()连接DWS数据库。

JDBC接口不提供重试连接的能力，您需要在业务代码中实现重试处理。

DriverManager.getConnection()方法支持以下重载：

- DriverManager.getConnection(String url);
- DriverManager.getConnection(String url, Properties info);
- DriverManager.getConnection(String url, String user, String password);

表 6-9 数据库连接参数

| 参数  | 描述                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|-----|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| url | <p>数据库连接描述符，可以在管理控制台查看，具体步骤请参见<a href="#">获取DWS集群连接地址</a>。</p> <p>url的格式如下：</p> <ul style="list-style-type: none"><li>• jdbc:postgresql:database</li><li>• jdbc:postgresql://host/database</li><li>• jdbc:postgresql://host:port/database</li><li>• jdbc:postgresql://host:port[,host:port][...]/database</li></ul> <p><b>说明</b></p> <ul style="list-style-type: none"><li>• 使用gsjdbc200.jar时，将“jdbc:postgresql”修改为“jdbc:gaussdb”。<ul style="list-style-type: none"><li>- database为要连接的数据库名称。</li><li>- host为数据库服务器名称或IP地址，当集群绑定弹性负载均衡(ELB)时，应设置为ELB的IP地址。</li><li>- port为数据库服务器端口。缺省情况下，会尝试连接到localhost的8000端口的database。</li><li>- 支持多ip端口配置形式，jdbc自动实现了负载均衡，多ip端口配置形式是采取随机访问+failover的方式，这个过程系统会自动忽略不可达IP。以","隔开，例如jdbc:postgresql://10.10.0.13:8000,10.10.0.14:8000/database</li></ul></li><li>• 使用JDBC连接集群时集群链接地址只支持指定jdbc连接参数，不支持增加变量参数。</li></ul> |

| 参数   | 描述                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| info | <p>数据库连接属性。常用的属性如下：</p> <ul style="list-style-type: none"> <li>• user: String类型，表示创建连接的数据库用户。</li> <li>• password: String类型，表示数据库用户的密码。</li> <li>• ssl: Boolean类型，表示是否使用SSL连接。</li> <li>• loggerLevel: string类型，为LogStream或LogWriter设置记录进DriverManager当前值的日志信息量。目前支持"OFF"、"DEBUG"和"TRACE"。值为"DEBUG"时，表示只打印DEBUG级别以上的日志，将记录非常少的信息。值等于TRACE时，表示打印DEBUG和TRACE级别的日志，将产生详细的日志信息。默认值为OFF，表示不打印日志。</li> <li>• prepareThreshold: integer类型，用于确定在转换为服务器端的预备语句之前，要求执行方法PreparedStatement的次数。缺省值是5。</li> <li>• batchSize: boolean类型，用于确定是否使用batch模式连接。</li> <li>• fetchsize: integer类型，用于设置数据库链接所创建statement的默认fetchsize。</li> <li>• ApplicationName: string类型，表示应用名称。未设置时，缺省值为PostgreSQL JDBC Driver。</li> <li>• allowReadOnly: boolean类型，用于设置connection是否允许设置readonly模式。默认为false，若该参数不被设置为true，则执行connection.setReadOnly不生效。</li> <li>• blobMode: string类型，用于设置setBinaryStream方法为不同的数据类型赋值。设置为on时表示为blob数据类型赋值，设置为off时表示为bytea数据类型赋值，默认为on。</li> <li>• currentSchema: string类型，用于指定连接数据库的schema。</li> <li>• defaultQueryMetaData: Boolean类型。默认查询sql元信息开关，默认false。开启后可以支持raw类型操作。但不兼容PreparedStatement中create table as和select into操作。</li> <li>• connectionExtraInfo: Boolean类型，表示驱动是否上报当前驱动的部署路径、进程属主用户到数据库。</li> </ul> <p><b>说明</b></p> <p>取值范围：true或false，默认值为true。设置connectionExtraInfo为true，JDBC驱动会将当前驱动的部署路径、进程属主用户上报到数据库中，记录在connection_info参数里；同时可以在PG_STAT_ACTIVITY和PGXC_STAT_ACTIVITY中查询到。</p> <ul style="list-style-type: none"> <li>• TCP_KEEPIIDLE=30: 连接空闲30s后开始检测。当“tcpKeepAlive=true”时生效。</li> <li>• TCP_KEEPCOUNT=9: 总共检测9次。当“tcpKeepAlive=true”时生效。</li> <li>• TCP_KEEPINTERVAL=30: 每次检测间隔30s。当“tcpKeepAlive=true”时生效。</li> <li>• cnListRefreshSwitch: string类型，用于设置JDBC是否开启自动感知CN存活列表。设置为on时表示开启自动感知CN存活列表，设置为off时表示不开启，默认为off。</li> </ul> |

| 参数       | 描述                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|          | <ul style="list-style-type: none"><li>cnListRefreshDelay: integer类型, 当“cnListRefreshSwitch=on”时生效, 用于设置扫描感知CN存活列表任务的启动时间。默认为1800000, 单位毫秒。</li><li>cnListRefreshPeriod: integer类型: 当“cnListRefreshSwitch=on”时生效, 用于设置扫描感知CN存活列表的时间周期。默认为1800000, 单位毫秒。</li><li>autoReconnect: Boolean类型, 用于配置是否开启数据库连接自动重连。设置为true时表示开启自动重连, 设置为false时, 表示不开启, 默认为false。</li><li>reConnectCount: Integer类型, 当“autoReconnect=true”时生效, 用于设置数据库连接自动重连的次数。默认为10, 如果重连次数超过设置的数值, 重连失败。</li><li>sslCrl: string类型, 用于设置JDBC吊销证书的路径, 默认为空。</li></ul> |
| user     | 数据库用户。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| password | 数据库用户的密码。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |

以SSL证书方式加密连接样例代码如下:

//以下代码将获取数据库SSL连接操作封装为一个接口。

```
public static Connection GetConnection(String username, String passwd) {
 //定义驱动类。
 String driver = "org.postgresql.Driver";
 //设置keystore。
 System.setProperty("javax.net.ssl.trustStore", "mytruststore");
 System.setProperty("javax.net.ssl.keyStore", "client.jks");
 System.setProperty("javax.net.ssl.trustStorePassword", "password");
 System.setProperty("javax.net.ssl.keyStorePassword", "password");

 Properties props = new Properties();
 props.setProperty("user", username);
 props.setProperty("password", passwd);
 props.setProperty("ssl", "true");

 String url = "jdbc:postgresql://" + "10.10.0.13" + ':' + "8000" + '/' + "gaussdb";
 Connection conn = null;

 try {
 //加载驱动。
 Class.forName(driver);
 } catch (Exception e) {
 e.printStackTrace();
 return null;
 }
 try {
 //创建连接。
 conn = DriverManager.getConnection(url, props);
 System.out.println("Connection succeed!");
 } catch (SQLException throwables) {
 throwables.printStackTrace();
 return null;
 }
 return conn;
}
```

**步骤8 执行SQL语句。**

1. 创建语句对象。  
Statement stmt = con.createStatement();
2. 执行语句对象。  
int rc = stmt.executeUpdate("CREATE TABLE tab1(id INTEGER, name VARCHAR(32));");
3. 释放语句对象。  
stmt.close();

**步骤9 调用方法close()关闭连接。**

----结束

## 示例代码

此示例将演示如何基于DWS提供的JDBC接口开发应用程序。

### 说明

在完成以下示例前，需要先创建存储过程。具体请参见《数据仓库服务开发指南》中的“教程：使用JDBC或ODBC开发”章节。

```
create or replace procedure testproc
(
 psv_in1 in integer,
 psv_in2 in integer,
 psv_inout in out integer
)
as
begin
 psv_inout := psv_in1 + psv_in2 + psv_inout;
end;
/
```

//DBtest.java  
//以下用例以gsjdbc4.jar为例。  
//演示基于JDBC开发的主要步骤，会涉及创建数据库、创建表、插入数据等。

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import java.sql.Statement;
import java.sql.CallableStatement;
import java.sql.Types;

public class DBTest {
 //创建数据库连接，以下IP地址和database替换为对应的数据库连接地址和数据库名称。
 public static Connection GetConnection(String username, String passwd) {
 String driver = "org.postgresql.Driver";
 String sourceURL = "jdbc:postgresql://10.10.0.13:8000/database";
 Connection conn = null;
 try {
 //加载数据库驱动。
 Class.forName(driver).newInstance();
 } catch (Exception e) {
 e.printStackTrace();
 return null;
 }

 try {
 //创建数据库连接。
 conn = DriverManager.getConnection(sourceURL, username, passwd);
 System.out.println("Connection succeed!");
 } catch (Exception e) {
 e.printStackTrace();
 return null;
 }
 }
}
```

```
}

return conn;
};

//执行普通SQL语句，创建customer_t1表。
public static void CreateTable(Connection conn) {
 Statement stmt = null;
 try {
 stmt = conn.createStatement();

 //执行普通SQL语句。
 int rc = stmt
 .executeUpdate("CREATE TABLE customer_t1 (c_customer_sk INTEGER, c_customer_name
VARCHAR(32));");

 stmt.close();
 } catch (SQLException e) {
 if (stmt != null) {
 try {
 stmt.close();
 } catch (SQLException e1) {
 e1.printStackTrace();
 }
 }
 e.printStackTrace();
 }
}

//执行预处理语句，批量插入数据。
public static void BatchInsertData(Connection conn) {
 PreparedStatement pst = null;

 try {
 //生成预处理语句。
 pst = conn.prepareStatement("INSERT INTO customer_t1 VALUES (?,?)");
 for (int i = 0; i < 3; i++) {
 //添加参数。
 pst.setInt(1, i);
 pst.setString(2, "data " + i);
 pst.addBatch();
 }
 //执行批处理。
 pst.executeBatch();
 pst.close();
 } catch (SQLException e) {
 if (pst != null) {
 try {
 pst.close();
 } catch (SQLException e1) {
 e1.printStackTrace();
 }
 }
 e.printStackTrace();
 }
}

//执行预编译语句，更新数据。
public static void ExecPreparedSQL(Connection conn) {
 PreparedStatement pstmt = null;
 try {
 pstmt = conn
 .prepareStatement("UPDATE customer_t1 SET c_customer_name = ? WHERE c_customer_sk = 1");
 pstmt.setString(1, "new Data");
 int rowcount = pstmt.executeUpdate();
 pstmt.close();
 } catch (SQLException e) {
 if (pstmt != null) {
 try {
```

```
 pstmt.close();
 } catch (SQLException e1) {
 e1.printStackTrace();
 }
 }
 e.printStackTrace();
}
}

//执行存储过程。
public static void ExecCallableSQL(Connection conn) {
 CallableStatement cstmt = null;
 try {

 cstmt=conn.prepareCall("{? = CALL TESTPROC(?,?,?)}");
 cstmt.setInt(2, 50);
 cstmt.setInt(1, 20);
 cstmt.setInt(3, 90);
 cstmt.registerOutParameter(4, Types.INTEGER); //注册out类型的参数，类型为整型。
 cstmt.execute();
 int out = cstmt.getInt(4); //获取out参数
 System.out.println("The CallableStatment TESTPROC returns:"+out);
 cstmt.close();
 } catch (SQLException e) {
 if (cstmt != null) {
 try {
 cstmt.close();
 } catch (SQLException e1) {
 e1.printStackTrace();
 }
 }
 e.printStackTrace();
 }
}

/**
 * 主程序，逐步调用各静态方法。
 * @param args
 */
public static void main(String[] args) {
 //创建数据库连接，以下User、Password替换为实际连接的数据库用户和密码。
 Connection conn = GetConnection("User", "Password");

 //创建表。
 CreateTable(conn);

 //批插数据。
 BatchInsertData(conn);

 //执行预编译语句，更新数据。
 ExecPreparedSQL(conn);

 //执行存储过程。
 ExecCallableSQL(conn);

 //关闭数据库连接。
 try {
 conn.close();
 } catch (SQLException e) {
 e.printStackTrace();
 }
}
}
```

### 6.5.1.4 配置 JDBC 连接（使用负载均衡方式）

#### 背景信息

很多用户在使用JDBC连接集群时只连接集群的一个CN，这就导致单个CN压力较大并且别的CN资源浪费，并且这种方式还有单点故障导致连接不可用的风险。

用户可使用JDBC连接多个CN避免以上问题。主要有以下三种方式：

- 使用ELB连接集群：弹性负载均衡（ELB）是将访问流量根据转发策略分发到后端多台弹性云服务器器的流量分发控制服务，可以通过流量分发扩展应用系统对外的服务能力，提高应用程序的容错能力。
- 使用JDBC负载均衡连接集群：开启JDBC负载均衡，URL需要配置至少一个CN节点内网IP，之后会自动扫描所有的CN节点IP，JDBC负载均衡可以使连接随机到其中一个CN，从而实现类似ELB的效果。
- 使用multi-host方式连接集群：使用JDBC配置多个节点的方式，也能实现类似ELB的效果。

#### 方式一：使用 ELB 连接集群（推荐）

**步骤1** 获取弹性负载均衡地址。在控制台打开指定集群详情页面，获取弹性负载均衡IP。

**步骤2** 配置驱动。请参见[下载JDBC或ODBC驱动](#)。

**步骤3** 获取数据库连接。

```
private static final String USER_NAME = "dbadmin";
private static final String PASSWORD = "password";
// jdbc:postgresql://ELB_IP:PORT/dbName"
private static final String URL = "jdbc:postgresql://100.95.153.169:8000/gaussdb";
private static Properties properties = new Properties();
static {
 properties.setProperty("user", USER_NAME);
 properties.setProperty("password", PASSWORD);
}
/**
 * 获取数据库连接
 */
public static Connection getConnection() {
 Connection connection = null;
 try {
 connection = DriverManager.getConnection(URL, properties);
 } catch (SQLException e) {
 e.printStackTrace();
 }
 return connection;
}
```

----结束

#### 方式二：使用 JDBC 负载均衡方式连接集群（推荐）

**步骤1** 获取内网IP。打开控制台指定集群拓扑页面，获取CN的内网IP，详情请参见[获取DWS集群连接地址](#)。

**步骤2** 配置驱动。请参见[下载JDBC或ODBC驱动](#)。

#### 说明

使用JDBC负载均衡方式连接集群方式驱动从8.3.1.200版本开始支持，如果使用此方式连接集群，需确认JDBC驱动版本是否升级到8.3.1.200及以上版本。

**步骤3** 获取数据库连接。URL参数配置请参见[使用JDBC连接](#)。

```
private static final String USER_NAME = "dbadmin";
private static final String PASSWORD = "password";
// jdbc:postgresql://host1:port1,host2:port2/dbName"
private static final String URL = "jdbc:postgresql://
100.95.146.194:8000,100.95.148.220:8000,100.93.0.221:8000/gaussdb?
loadBalanceHosts=true&cnListRefreshSwitch=on&cnListRefreshDelay=100000&cnListRefreshPeriod=5000
";
private static Properties properties = new Properties();
static {
 properties.setProperty("user", USER_NAME);
 properties.setProperty("password", PASSWORD);
}
/**
 * 获取数据库连接
 */
public static Connection getConnection() {
 Connection connection = null;
 try {
 connection = DriverManager.getConnection(URL, properties);
 } catch (SQLException e) {
 e.printStackTrace();
 }
 return connection;
}
```

----结束

**方式三：使用 multi-host 方式连接集群****步骤1** 获取公网IP。打开控制台指定集群详情页面，获取弹性公网IP。**步骤2** 配置驱动。请参见[下载JDBC或ODBC驱动](#)。**步骤3** 获取数据库连接。

```
private static final String USER_NAME = "dbadmin";
private static final String PASSWORD = "password";
// jdbc:postgresql://host1:port1,host2:port2/dbName"
private static final String URL = "jdbc:postgresql://
100.95.146.194:8000,100.95.148.220:8000,100.93.0.221:8000/gaussdb?loadBalanceHosts=true";
private static Properties properties = new Properties();
static {
 properties.setProperty("user", USER_NAME);
 properties.setProperty("password", PASSWORD);
}
/**
 * 获取数据库连接
 */
public static Connection getConnection() {
 Connection connection = null;
 try {
 connection = DriverManager.getConnection(URL, properties);
 } catch (SQLException e) {
 e.printStackTrace();
 }
 return connection;
}
```

----结束

### 6.5.1.5 配置 JDBC 连接（使用 IAM 认证方式）

#### 概述

DWS提供了使用IAM认证方式访问数据库的功能。当使用JDBC应用程序连接集群时，您可以在JDBC连接中配置IAM用户名及其用户凭证等信息，在连接数据库时系统就会自动生成临时数据库凭证，从而成功连接到数据库。

#### 说明

- 当前仅支持1.3.1及以上版本的集群及其配套的JDBC驱动程序使用IAM认证方式访问数据库。请先参考[下载JDBC或ODBC驱动](#)下载JDBC驱动程序。

IAM用户凭证有密码和访问密钥（Access Key ID和Secret Access Key，简称AK和SK）两种类型，您要为JDBC连接提供IAM访问密钥。

如需使用IAM用户凭证访问数据库，必须先给您的IAM用户授予DWS Database Access权限，同时拥有DWS Administrator和DWS Database Access权限的用户，才能基于IAM用户生成临时数据库用户凭证以连接DWS数据库。

需要注意的是，DWS Database Access是用户组级别的权限，您可以通过为用户组授权并将用户加入到用户组的方式，使用户具有用户组中的权限。

在IAM中，只有admin用户组的用户可以管理用户。如需给IAM用户授权，您的IAM账号必须属于IAM的admin用户组，否则，请联系IAM账号管理员帮您授权。

使用IAM用户凭证访问数据库的流程如下：

1. [授予IAM用户DWS Database Access权限](#)
2. [创建IAM用户凭证](#)
3. [配置JDBC连接使用IAM认证方式连接集群](#)

#### 授予 IAM 用户 DWS Database Access 权限

**步骤1** 登录华为云管理控制台，单击“服务列表 > 管理与监管 > 统一身份认证服务”，打开IAM管理控制台。

**步骤2** 修改您的IAM用户所属的用户组，给用户组设置策略，授予用户组DWS Database Access权限，并将您的IAM用户添加到该IAM用户组中。

只有IAM的admin用户组的用户才能执行此步骤。在IAM中，只有admin用户组的用户可以管理用户，包括创建用户组及用户、设置用户组权限等。

具体操作请参见《统一身份认证服务用户指南》中的“用户指南 > 管理用户和用户组 > 查看或修改用户组”。

您也可以新建一个IAM用户组，并给用户组设置策略，授予用户组DWS Administrator和DWS Database Access权限，然后将您的IAM用户添加到该IAM用户组中。具体操作请参见《统一身份认证服务用户指南》中的“用户指南 > 管理用户和用户组 > 创建用户组”。

----结束

#### 创建 IAM 用户凭证

用户可以登录管理控制台创建访问密钥，如果您已经创建过了，也可以使用已有的访问密钥。

**步骤1** 登录管理控制台。

**步骤2** 将鼠标移到右上角的用户名，单击“我的凭证”。

**步骤3** 再单击“管理访问密钥”页签，可以查看已有的访问密钥，也可以单击“新增访问密钥”进行创建。

访问密钥是IAM身份认证的重要凭证，只有在新增访问密钥时，用户才可以下载到含有Access Key ID（AK）和Secret Access Key（SK）的密钥文件，在管理控制台只能查看到Access Key ID，如果您未曾下载过该密钥文件，请联系您的管理员进行获取，或者重新创建。

#### 说明

每个用户最多可创建2个访问密钥，有效期为永久。为了账号安全性，建议您定期更换并妥善保存访问密钥。

----结束

## 配置 JDBC 连接使用 IAM 认证方式连接集群

### 配置JDBC连接参数

表 6-10 数据库连接参数

| 参数  | 描述                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|-----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| url | <p>gsjdbc4.jar/gsjdbc200.jar数据库连接描述符。JDBC接口不提供重试连接的能力，您需要在业务代码中实现重试连接的处理。url示例如下：</p> <pre>jdbc:dws:iam://dws-IAM-demo:eu-west-0/gaussdb?<br/>AccessKeyID=XXXXXXXXXXXXXXXXXXXX&amp;SecretAccessKey=XXXXXXXXXXXXXXXXXXXX<br/>XXXXXXXXXXXX&amp;DbUser=user_test&amp;AutoCreate=true</pre> <p><b>JDBC URL参数说明：</b></p> <ul style="list-style-type: none"><li>• jdbc:dws:iam是URL格式的前缀。</li><li>• dws-IAM-demo为数据库集群名称。</li><li>• eu-west-0是集群所在的区域。JDBC会根据所填写区域，访问对应DWS服务，向所在集群下发IAM证书，用于IAM用户认证。DWS服务地址已内置在JDBC配置文件中。<br/>有关DWS的区域信息，请参考<a href="#">地区和终端节点</a>。</li><li>• gaussdb是要连接的数据库名。</li><li>• AccessKeyID/SecretAccessKey为参数DbUser指定的IAM用户所对应的访问密钥ID和秘密访问密钥。</li><li>• DbUser请设置为IAM用户名，注意，当前版本暂不支持IAM用户名中含有中划线的情况。<ul style="list-style-type: none"><li>- 如果数据库中已存在DbUser指定的用户，则临时用户凭证具有与现有用户相同的权限。</li><li>- 如果数据库中不存在DbUser指定的用户，且AutoCreate参数值为true，则自动创建一个以DbUser参数值作为用户名的新用户，默认创建的用户为数据库普通用户。</li></ul></li><li>• AutoCreate可以不设置，默认为false。该参数表示是否在数据库中自动创建一个以DbUser参数值作为用户名的数据库用户。<ul style="list-style-type: none"><li>- true表示自动创建。如果用户已存在则不会再创建。</li><li>- false表示不会自动创建。如果数据库中不存在DbUser指定的用户名将返回失败。</li></ul></li><li>• addressType连接使用的地址类型，默认为auto。<ul style="list-style-type: none"><li>- auto：自动选择IP连接。表示自动选取IP进行连接，优先级EIP &gt; ELB &gt; 私有IP。</li><li>- eip：使用弹性公网IP连接。</li><li>- elb：使用弹性负载均衡连接，优先级elb_public &gt; elb_private。</li><li>- elb_public：使用弹性负载均衡公网IP连接。</li><li>- elb_private：使用弹性负载均衡私网连接。</li></ul></li></ul> |

| 参数   | 描述                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| info | <p>数据库连接属性。常用的属性如下：</p> <ul style="list-style-type: none"><li>• ssl: Boolean类型。表示是否使用SSL连接。</li><li>• loglevel: Integer类型。为LogStream或LogWriter设置记录进DriverManager当前值的日志信息量。目前支持org.postgresql.Driver.DEBUG和org.postgresql.Driver.INFO。值为1时，表示只打印org.postgresql.Driver.INFO，将记录非常少的信息。值大于等于2时，表示打印org.postgresql.Driver.DEBUG和org.postgresql.Driver.INFO，将产生详细的日志信息。默认值为0，表示不打印日志。</li><li>• charSet: String类型。表示在向数据库发送数据或从数据库接收数据时使用到的字符集。</li><li>• prepareThreshold: Integer类型。用于确定在转换为服务器端的预备语句之前，要求执行方法PreparedStatement的次数。缺省值是5。</li></ul> |

## 示例

```
//以下用例以gsjdbc4.jar为例。
//以下代码将获取数据库连接操作封装为一个接口，可通过给定集群所在的区域、集群名称、AccessKeyID、
SecretAccessKey及对应的IAM用户名来连接数据库。
public static Connection GetConnection(String clustername, String regionname, String AK, String SK,
String username) {
 //驱动类。
 String driver = "org.postgresql.Driver";
 //数据库连接描述符。
 String sourceURL = "jdbc:dws:iam://" + clustername + ":" + regionname + "/postgresgaussdb?" +
"AccessKeyID=" +
 AK + "&SecretAccessKey=" + SK + "&DbUser=" + username + "&autoCreate=true";

 Connection conn = null;

 try {
 //加载驱动。
 Class.forName(driver);
 } catch (ClassNotFoundException e) {
 return null;
 }
 try {
 //创建连接。
 conn = DriverManager.getConnection(sourceURL);
 System.out.println("Connection succeed!");
 } catch (SQLException e) {
 return null;
 }
 return conn;
}
```

### 6.5.1.6 JDBC 配置数据库第三方连接池

#### 背景信息

由于DWS当前没有自己的JDBC连接池，且继承于PG的连接池已下线，因此DWS建议使用第三方的druid、Hikari CP、dbcp2等连接池。

 说明

- JDBC继承于PG的连接池已下线，不推荐使用。
- 下述JDBC和驱动下载哪个版本，连接池参数值如何配置，请根据业务实际情况决定。

## dbcp2 连接池配置

**步骤1** 下载JDBC驱动包，请参见[下载JDBC或ODBC驱动](#)。

- 下载commons-dbc2驱动包，下载地址：[https://commons.apache.org/dbcp/download\\_dbcp.cgi](https://commons.apache.org/dbcp/download_dbcp.cgi)。
- 下载commons-logging驱动包，下载地址：[https://commons.apache.org/proper/commons-logging/download\\_logging.cgi](https://commons.apache.org/proper/commons-logging/download_logging.cgi)。
- 下载commons-pool2驱动包，下载地址：[https://commons.apache.org/proper/commons-pool/download\\_pool](https://commons.apache.org/proper/commons-pool/download_pool)。

**步骤2** 将JDBC驱动包和commons-dbc2、commons-logging、commons-pool2驱动包添加到工程中，配置数据库连接池相关参数。

 说明

- 如果启用了**removeAbandoned**，则一个已遗弃的连接将会被连接池回收再利用，在“(getNumIdle() < 2) and (getNumActive() > getMaxTotal() - 3)”成立时，这个机制将被触发。
  - 例如maxTotal=20，这里有18个活跃连接，1个限制连接，则会触发“**removeAbandoned**”。
  - 只有在活动连接超过“**removeAbandonedTimeout**”所指定的秒数内未使用才会被删除（默认为300秒）。
  - 遍历一个结果集并不被统计为被使用，创建一个语句，预处理语句，可调用语句或使用它们其中的一个执行查询（使用执行方法中的某一个）会重新设置其父连接的“**lastUsed**”属性。
- 如果在高负载的系统中将“**maxIdle**”的值设置的很低，可能会导致一个新的连接刚刚被创建的时候就立即被关闭了。这是因为活跃的线程及时关闭连接要比那些打开连接的线程要快，导致空闲的连接数大于“**maxIdle**”。高负载系统中“**maxIdle**”的最合适的配置值取决于应用场景，但是缺省值是一个好的开始点。

**表 6-11** dbcp2 连接池参数配置

| 参数名             | 默认值                     | 说明         |
|-----------------|-------------------------|------------|
| driverClassName | 填写org.postgresql.Driver | 数据库驱动名称。   |
| url             | -                       | 连接数据库的URL。 |
| username        | -                       | 用户名。       |
| password        | -                       | 密码。        |

| 参数名                         | 默认值  | 说明                                                                                                                                                                                                                                       |
|-----------------------------|------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| connectionProperties        | -    | 连接参数是在建立一个新连接时发送给JDBC驱动的，字符串的格式必须是[参数名=参数值;]。<br><b>说明</b><br>用户名和密码属性是需要明确指出的，所以这两个参数不需要包含在这里。                                                                                                                                          |
| defaultAutoCommit           | -    | 自动提交。通过当前连接池创建连接的默认为自动提交状态，如果不设置，则“setAutoCommit”方法将不被调用。                                                                                                                                                                                |
| defaultReadOnly             | -    | 只读设置。通过当前连接池创建连接的默认为只读状态，如果不设置，则“setReadOnly”方法将不被调用。                                                                                                                                                                                    |
| defaultTransactionIsolation | -    | 事务隔离级别。<br>通过这个池创建连接的默认为事务隔离策略，设置值为下列中的某一个： <ul style="list-style-type: none"> <li>• NONE：无。</li> <li>• READ_COMMITTED：读已提交。</li> <li>• READ_UNCOMMITTED：读未提交。</li> <li>• REPEATABLE_READ：可重复读。</li> <li>• SERIALIZABLE：可串行化。</li> </ul> |
| defaultCatalog              | -    | 通过这个池创建连接的默认为缺省的catalog。                                                                                                                                                                                                                 |
| cacheState                  | true | 连接池缓存状态。<br>设置为true，连接资源池后将在第一次读或写，以及随后写的时候缓存当前的只读状态和自动提交设置。这样就省去了对getter的任何进一步调用时对数据库的额外查询。<br>如果直接访问底层连接，只读状态或自动提交设置改变缓存值将不会被反映到当前的状态，在这种情况下，应该将该属性设置为false以禁用缓存。                                                                      |
| defaultQueryTimeout         | null | 查询超时时间。 <ul style="list-style-type: none"> <li>• 设置为一个整数值，用于在创建Statement时，指定为查询超时时间。</li> <li>• 设置为null，则使用驱动程序默认设置。</li> </ul>                                                                                                          |
| enableAutoCommitOnReturn    | true | 连接归还到池时，设置为自动提交。<br>如果设置为true，则连接被归还到连接池时，会指定设置为“ <b>autoCommit = true</b> ”。                                                                                                                                                            |
| rollbackOnReturn            | true | 连接归还到池时，回滚所有操作。<br>如果设置为true，则连接被归还到连接池时，会自动执行一次“ <b>rollback()</b> ”；前提是自动提交为true。                                                                                                                                                      |

| 参数名                    | 默认值      | 说明                                                                                                                                                                           |
|------------------------|----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| initialSize            | 0        | 初始连接数。当前连接池被启动时初始化的创建的连接个数，起始生效版本为1.2版本。                                                                                                                                     |
| maxTotal               | 8        | 最大活动连接数。可以在当前连接池中同一时刻被分配的有效连接数的最大值，如果设置为负数，则不限制。                                                                                                                             |
| maxIdle                | 8        | 最大空闲连接数。在连接池中，可以保持空闲状态的最大连接数，超出设置值之外的空闲连接在归还到连接池时将被释放，如设置为负数，则不限制。                                                                                                           |
| minIdle                | 0        | <p>最小空闲连接数。可以在池中保持空闲的最小连接数，低于设置值时，空闲连接将被创建，以确保最小空闲连接数大于等于minIdle值；如果设置为0，则不创建。</p> <p><b>说明</b><br/>设置的数值生效的前提是：timeBetweenEvictionRunsMillis（空闲对象驱逐线程运行时的休眠毫秒数）被设置为正数。</p>  |
| maxWaitMillis          | -        | <p>从连接池获取一个连接时，最大的等待时间。</p> <ul style="list-style-type: none"> <li>• 设置为-1时，如果没有可用连接，连接池会一直无限期等待，直到获取到连接为止。</li> <li>• 设置为N（毫秒）时，则连接池会等待N毫秒，等待不到，抛出异常。</li> </ul>            |
| validationQuery        | SELECT 1 | <p>确认查询SQL。在连接池返回连接给调用者前用来进行连接校验的查询SQL。</p> <ul style="list-style-type: none"> <li>• 如果指定，则这个查询必须是至少返回一行数据的SELECT语句。</li> <li>• 如果没有指定，则连接将通过调用“isValid()”方法进行校验。</li> </ul> |
| validationQueryTimeout | -        | <p>确认连接有效SQL的执行查询超时时间（秒）。</p> <p>如果设置为正数，则传递给JDBC驱动的“setQueryTimeout()”方法，这个设置对执行确认有效查询SQL起作用。</p>                                                                           |
| testOnCreate           | false    | 连接创建后，马上验证有效性。指明对象在创建后是否需要验证有效性，如果对象验证失败，则触发对象创建的租借尝试将失败。                                                                                                                    |
| testOnBorrow           | true     | 从连接池获取一个连接时，验证有效性。指明在从池中租借对象时是否要进行验证有效，如果对象验证失败，则对象将从连接池释放，可尝试租借另一个。                                                                                                         |
| testOnReturn           | false    | 连接被归还到连接池时，验证有效性。指明在将对象归还给连接池前是否需要校验。                                                                                                                                        |
| testWhileIdle          | false    | 连接空闲时，验证有效性。指明对象是否需要通过对对象驱逐者进行校验（如果有的话），假如一个对象验证失败，该对象将从连接池中释放。                                                                                                              |

| 参数名                            | 默认值            | 说明                                                                                                                                                                                                                                                                                    |
|--------------------------------|----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| timeBetweenEvictionRunsMillis  | -1             | 空闲对象驱逐线程运行时的休眠毫秒数，如果设置为非正数，不运行空闲对象驱逐线程。                                                                                                                                                                                                                                               |
| numTestsPerEvictionRun         | 3              | 在每个空闲对象驱逐线程运行过程中进行检查的对象个数。                                                                                                                                                                                                                                                            |
| minEvictableIdleTimeMillis     | 1000 * 60 * 30 | 符合对象驱逐条件的对象在池中最小空闲毫秒总数。空闲的连接最低需要N毫秒后被释放，单位：ms。                                                                                                                                                                                                                                        |
| softMinEvictableIdleTimeMillis | -1             | 符合对象驱逐条件的对象在池中最小空闲毫秒总数。空闲的连接最低需要N毫秒后被释放，但有额外条件，条件是池中至少保留有minIdle所指定的个数的连接。<br>当“miniEvictableIdleTimeMillis”被设置为正数时，空闲连接驱逐者首先检测miniEvictableIdleTimeMillis，当空闲连接被驱逐者访问时，首先与miniEvictableIdleTimeMillis所指定的值进行比较（不考虑当前池中的空闲连接数），然后比较softMinEvictableIdleTimeMillis所指定的连接数，包括minIdle条件。 |
| maxConnLifetimeMillis          | -1             | 一个连接的最大存活毫秒数。如果超过这个时间，则连接在下次激活、钝化、校验时都将会失败。如果设置为0或小于0的值，则连接的存活时间是无限的。                                                                                                                                                                                                                 |
| logExpiredConnections          | true           | 一个过期的连接被连接池关闭时，写日志标识。如果连接存活时间超过maxConnLifetimeMillis，连接将被连接池回收，此时默认输出日志。如果设置为false，这不会写日志。                                                                                                                                                                                            |
| connectionInitSqls             | -              | 连接被第一次创建时，执行的初始化SQL。在第一次创建时用来初始化物理连接的SQL语句集合。这些语句只在配置的连接工厂创建连接时被执行一次。                                                                                                                                                                                                                 |
| lifo                           | true           | 后进先出。 <ul style="list-style-type: none"><li>• 设置为true表明连接池（如果池中有可用的空闲连接时）将返回最后一次使用的租借对象（最后进入）。</li><li>• 设置为false则表明池将表现为FIFO队列（先进先出），将会按照它们被归还的顺序从空闲连接实例池中获取连接。</li></ul>                                                                                                            |
| poolPreparedStatements         | false          | 设置该连接池的预处理语句池是否生效。                                                                                                                                                                                                                                                                    |

| 参数名                                                     | 默认值   | 说明                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|---------------------------------------------------------|-------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| maxOpenPreparedStatements                               | -     | <p>可以在语句池中同时分配的最大语句数。设置为负数则不限制。</p> <p>这个设置同时作用于预处理语句池。当一个可用的语句池被创建给每一个连接时，通过以下方法创建的预处理语句将被池化。</p> <pre>public PreparedStatement prepareStatement(String sql) public PreparedStatement prepareStatement(String sql, int resultSetType, int resultSetConcurrency)</pre> <p><b>说明</b></p> <p>确保连接会留下一些资源给其他语句。池化预处理语句可能会在数据库中保持他们的游标，可能会引起连接的游标越界，尤其是maxOpenPreparedStatements的值被设置为默认值（无限的），且一个应用程序可能会为每个连接打开大量不同的预处理语句。为了避免这个问题，maxOpenPreparedStatements应该被设置为一个小于连接可以打开的最大游标数的值。</p> |
| accessToUnderlyingConnectionAllowed                     | false | 控制PoolGuard是否可以访问底层连接。                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| removeAbandonedOnMaintenance<br>removeAbandonedOnBorrow | false | <p>标记是否删除超过removeAbandonedTimeout所指定时间的被遗弃的连接。</p> <p>如果设置为true，则一个连接在超过removeAbandonedTimeout所设定的时间未使用即被认为是应该被抛弃并应该被移除的。</p> <p>创建一个语句，预处理语句，可调用语句或使用它们的一个执行查询（使用执行方法中的某一个）会重新设置其父连接的lastUsed属性。</p> <p>在写操作较少的应用程序中将该参数设置为true，可以将数据库连接从连接关闭失败中恢复。</p>                                                                                                                                                                                                                               |
| removeAbandonedTimeout                                  | 300   | 一个被抛弃连接可以被移除的超时时间，单位：s。                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| logAbandoned                                            | false | 标志是否为应用程序中遗弃语句或连接的代码开启日志堆栈追踪。因为一个堆栈跟踪已被创建，被抛弃的语句和连接相关的日志将被覆盖到打开每个连接或者创建一个Statement。                                                                                                                                                                                                                                                                                                                                                                                                    |
| abandonedUsageTracking                                  | false | <p>废弃使用跟踪。如果为true，则连接池将在每次在池化连接上调用方法时记录堆栈跟踪，并保留最新的堆栈跟踪以帮助调试已放弃的连接。</p> <p><b>说明</b></p> <p>将其设置为true会增加大量开销，请谨慎操作。</p>                                                                                                                                                                                                                                                                                                                                                                 |

| 参数名                     | 默认值   | 说明                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|-------------------------|-------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| fastFailValidation      | false | 验证快速失败。如果发生致命的异常，验证语句会快速失败。而不会再执行“isValid()”，也不再去执行验证查询语句。致命的异常（SQL_STATE）指以下： <ul style="list-style-type: none"><li>• 57P01 (ADMIN SHUTDOWN)</li><li>• 57P02 (CRASH SHUTDOWN)</li><li>• 57P03 (CANNOT CONNECT NOW)</li><li>• 01002 (SQL92 disconnect error)</li><li>• JZ0C0 (Sybase disconnect error)</li><li>• JZ0C1 (Sybase disconnect error)</li><li>• Any SQL_STATE code that starts with “08”</li></ul> 需要覆盖异常码，请参考disconnectionSqlCodes。 |
| disconnectionSqlCodes   | -     | 异常码。以逗号分割的SQL_STATE码，“fastFailValidation”为“true”时生效。                                                                                                                                                                                                                                                                                                                                                                                            |
| jmxName                 | -     | 以指定的名称将DataSource注册为JMX MBean。该名称必须符合JMX对象名称语法。                                                                                                                                                                                                                                                                                                                                                                                                 |
| registerConnectionMBean | true  | 注册连接JMX Mbean。                                                                                                                                                                                                                                                                                                                                                                                                                                  |

----结束

## Hikari CP 连接池配置

**步骤1** 下载JDBC驱动包，请参见[下载JDBC或ODBC驱动](#)。

- 下载HikariCP驱动包，下载地址：<https://mvnrepository.com/artifact/com.zaxxer/HikariCP/4.0.3>。
- 下载slf4j驱动包，下载地址：<https://www.slf4j.org/download.html>。

**步骤2** 将JDBC驱动包和HikariCP、slf4j驱动包添加到工程中，配置数据库连接池相关参数。

表 6-12 Hikari CP 连接池参数配置

| 参数名             | 默认值                     | 说明                 |
|-----------------|-------------------------|--------------------|
| driverClassName | 填写org.postgresql.Driver | 数据库驱动名称。           |
| jdbcUrl         | -                       | 连接数据库的URL。         |
| username        | -                       | 用户名。               |
| password        | -                       | 密码。                |
| autoCommit      | true                    | 连接返回连接池时，是否自动提交事务。 |

| 参数名                       | 默认值     | 说明                                                                                                                                                                                                                                                                    |
|---------------------------|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| connectionTimeout         | 30000   | 从连接池获取连接的最大超时时间。                                                                                                                                                                                                                                                      |
| idleTimeout               | 60000   | 空闲连接存活的最大时间。只有当 minimumIdle 小于 maximumPoolSize 时，该设置才生效。 <ul style="list-style-type: none"> <li>空闲连接数大于 minimumIdle 且连接的空闲状态时间大于 idleTimeout 值时，将把该连接从连接池中删除。</li> <li>0 表示永不超时。</li> </ul>                                                                           |
| keepaliveTime             | 0       | 保持空闲连接可用的检测频率，单位：ms。0 表示不检测。                                                                                                                                                                                                                                          |
| maxLifetime               | 1800000 | 连接存活的最大时间，单位：ms。0 表示没有限制。                                                                                                                                                                                                                                             |
| connectionTestQuery       | -       | 连接检测的查询语句。                                                                                                                                                                                                                                                            |
| minimumIdle               | 10      | 最小空闲连接数。为了提高性能，建议不要设置此参数，使连接池为固定大小。                                                                                                                                                                                                                                   |
| maximumPoolSize           | 10      | 最大连接数。                                                                                                                                                                                                                                                                |
| metricRegistry            | -       | 该参数仅通过编程配置或 IoC 容器可用。该参数用于指定池使用的 “Codahale/ Dropwizard MetricRegistry” 实例来记录各种指标。                                                                                                                                                                                     |
| healthCheckRegistry       | -       | 该参数仅通过编程配置或 IoC 容器可用。该参数用于指定池使用的 “Codahale/ Dropwizard HealthCheckRegistry” 实例来记录健康信息。                                                                                                                                                                                |
| poolName                  | -       | 连接池名称。                                                                                                                                                                                                                                                                |
| initializationFailTimeout | 1       | 启动连接池时不能成功初始化连接，是否快速失败。 <ul style="list-style-type: none"> <li>大于 0 时，会尝试获取连接。如果获取时间超过指定时长（ connectionTimeout + initializationFailTimeout ），不会开启连接池，并抛出异常。</li> <li>等于 0 时，会尝试获取并验证连接。如果获取成功但验证失败则不开启池，但是如果获取失败还是会开启池。</li> <li>小于 0 时，直接启动连接池，不进行初始化连接尝试。</li> </ul> |

| 参数名                    | 默认值   | 说明                                                                                    |
|------------------------|-------|---------------------------------------------------------------------------------------|
| isolateInternalQueries | false | 是否在事务中隔离HikariCP自己的查询。<br>“autoCommit”为“false”时，该设置生效。                                |
| allowPoolSuspension    | false | 是否允许通过JMX挂起和恢复连接池。挂起时获取连接不会超时，直到连接池恢复。                                                |
| readOnly               | false | 连接是否只读。                                                                               |
| registerMbeans         | false | 是否开启JMX。                                                                              |
| catalog                | -     | 默认的数据库catalog。                                                                        |
| connectionInitSql      | -     | 连接池初始化后执行的SQL。                                                                        |
| transactionIsolation   | -     | 默认的事务隔离级别。                                                                            |
| validationTimeout      | 5000  | 连接检测的超时时间。必须大于<br>connectionTimeout，最小允许的为250。                                        |
| leakDetectionThreshold | 0     | 连接可以被借出多久。<br>超过该时间将打印连接可能泄露的日志，最小允许的为2000，单位：ms。                                     |
| schema                 | -     | 默认的数据库schema。                                                                         |
| threadFactory          | -     | 指定连接池用于创建线程的<br>java.util.concurrent.ThreadFactory实例。此参数仅通过编程配置或IoC容器可用。              |
| scheduledExecutor      | -     | 指定连接池用于执行定时任务的<br>java.util.concurrent.ScheduledExecutorService实例。此参数仅通过编程配置或IoC容器可用。 |

----结束

## Druid 连接池配置

**步骤1** 下载JDBC驱动包，请参见[下载JDBC或ODBC驱动](#)。

下载Druid驱动包，下载地址：<https://druid.apache.org/downloads/>。

**步骤2** 将JDBC驱动包和Druid驱动包添加到工程中，配置数据库连接池相关参数。

**表 6-13** Druid 连接池参数配置

| 参数名 | 默认值 | 说明         |
|-----|-----|------------|
| url | -   | 连接数据库的URL。 |

| 参数名                       | 默认值                         | 说明                                                                                                                                           |
|---------------------------|-----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------|
| username                  | -                           | 用户名。                                                                                                                                         |
| password                  | -                           | 密码。                                                                                                                                          |
| driverClassName           | 填写<br>org.postgresql.Driver | 数据库驱动名称。                                                                                                                                     |
| initialSize               | 0                           | 初始化时建立物理连接的个数。初始化发生在显示调用init方法，或者第一次getConnection时。                                                                                          |
| maxActive                 | 8                           | 线程池中最大的连接数。                                                                                                                                  |
| minIdle                   | 0                           | 线程池最小空闲数。Druid会定期扫描连接数情况，如果扫描的值大于该值就关闭多余的连接数，小于就创建符合要求的连接数；这个参数主要应用于突然有大量请求的时候，创建新的连接数，该操作比较耗时。                                              |
| connectTimeout            | -                           | 连接数据库超时时间，单位：ms。                                                                                                                             |
| socketTimeout             | -                           | socket连接数据库连接超时时间，单位：ms。                                                                                                                     |
| maxWait                   | -1                          | 连接池中连接用完时，新的请求等待时间，单位：ms。<br>-1表示无限等待，直到超时为止。                                                                                                |
| poolPreparedStatements    | false                       | 是否缓存preparedStatement，也就是PSCache。PSCache对支持游标的数据库性能提升较大。                                                                                     |
| maxOpenPreparedStatements | -                           | 启用PSCache，必须配置大于0。<br>当大于0时，poolPreparedStatements自动触发修改为true。                                                                               |
| validationQuery           | SELECT 1                    | 用来检测连接是否有效的SQL。<br>如果validationQuery为空，则testOnBorrow、testOnReturn、testWhileIdle这三个参数都不会起作用，因为这三个参数都是通过执行参数validationQuery指定的SQL来验证数据库连接的有效性。 |
| testOnBorrow              | -                           | 申请连接时执行validationQuery检测连接是否有效，这个配置可能会降低性能，请谨慎操作。                                                                                            |
| testOnReturn              | -                           | 归还连接时执行validationQuery检测连接是否有效，这个配置可能会降低性能，请谨慎操作。                                                                                            |

| 参数名                           | 默认值   | 说明                                                                                                                                                                                                                      |
|-------------------------------|-------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| testWhileIdle                 | true  | 申请连接的时候检测，建议配置为true，不影响性能，并且保证安全性。如果空闲时间大于timeBetweenEvictionRunsMillis，执行validationQuery检测连接是否有效都不会起作用。                                                                                                                |
| timeBetweenEvictionRunsMillis | 60s   | 执行validationQuery检测连接是否有效。判断连接池的连接空闲数是否大于minIdle，如果是则关闭多余的连接数，少则会补上（如果当前连接池中某个连接在空闲了timeBetweenEvictionRunsMillis时间后仍然没有使用，则会被物理性关闭掉）。有两个含义：<br>1. Destroy线程会检测连接的间隔时间。<br>2. testWhileIdle的判断依据，详细看testWhileIdle属性的说明。 |
| minEvictableIdleTimeMillis    | 30min | 连接保持空闲而不被驱逐的最长存活时间。Destroy线程中如果检测到当前连接的最后活跃时间和当前时间的差值大于minEvictableIdleTimeMillis，则关闭当前连接。<br><b>说明</b><br>这个参数和timeBetweenEvictionRunsMillis参数有点冲突，可默认不配置该参数。                                                          |
| connectionInitSqls            | -     | 物理连接初始化的时候执行SQL。                                                                                                                                                                                                        |
| exceptionSorter               | -     | 当数据库抛出一些不可恢复的异常时，抛弃连接。                                                                                                                                                                                                  |
| filters                       | -     | 通过别名的方式配置扩展插件，属性类型是字符串。常用的插件有监控统计用的filter：<br><ul style="list-style-type: none"> <li>stat：监控统计。</li> <li>log4j：日志记录。</li> <li>wall：防御sql注入。</li> </ul>                                                                  |
| proxyFilters                  | -     | 类型是List<com.alibaba.druid.filter.Filter>，可同时配置filter和proxyFilters，这两个是组合关系。                                                                                                                                             |

| 参数名                                       | 默认值    | 说明                                                                                                                                                            |
|-------------------------------------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| removeAbandoned                           | false  | 如果连接泄露，是否需要回收泄露的连接。在getNumActive()快要到getMaxActive()的时候，系统会进行无效的连接回收，回收的连接为removeAbandonedTimeout（默认300秒）中设置的秒数后没有使用的连接；对于建立时间超过removeAbandonedTimeout的连接强制关闭。 |
| removeAbandonedTimeout                    | 300s   | 设置Druid强制回收连接的时限，单位：s。当程序从连接池中得到连接开始算起，指定连接建立多长时间druid将强制回收该连接。                                                                                               |
| logAbandoned                              | false  | 如果回收泄露的连接，是否要打印一条log。<br>指定发生removeabandoned的时候，是否记录当前线程的堆栈信息到日志中。                                                                                            |
| removeAbandonedTimeoutMillis              | 5min   | 连接回收的超时时间。设置“removeAbandoned”为“true”，Druid会定期检查线程池溢出的情况，如果不是运行状态，且超过设置的时间就会被回收。                                                                               |
| maxEvictableIdleTimeMillis                | 7hours | 最大空闲时间，默认为7小时。                                                                                                                                                |
| maxPoolPreparedStatementPerConnectionSize | 20     | 每个连接最多缓存的SQL数。                                                                                                                                                |
| keepAlive                                 | false  | 初始化连接池时会填充到minIdle的数量。连接池中的minIdle数量以内的连接，空闲时间超过minEvictableIdleTimeMillis时，则会执行keepAlive操作，打开会一直保持minIdle的数量值。                                               |
| notFullTimeoutRetryCount                  | 0      | 连接池内借出的连接加上可用连接小于最大连接数时，则进行重试操作次数，默认为0。                                                                                                                       |
| logSlowSql                                | false  | 是否打印慢SQL。                                                                                                                                                     |

### 步骤3 Druid连接demo示例。

resource目录下新建db.properties文件：

```
数据库连接参数
url=jdbc:postgresql://10.10.0.13:8000/gaussdb
username=user
password=pass
validationQuery=select 1
validationQueryTimeout=300
```

```
#driverClassName=JDBC驱动名
driverClassName=org.postgresql.Driver
初始化连接的数量
initialSize=1
最大连接的数量
maxActive=20
#核心线程数，大于该数会释放
minIdle=10
```

#### 示例代码：

```
import com.alibaba.druid.pool.DruidDataSource;
import com.alibaba.druid.pool.DruidDataSourceFactory;
import com.alibaba.druid.pool.DruidPooledConnection;
import java.io.IOException;
import java.io.InputStream;
import java.sql.ResultSet;
import java.sql.Statement;
import java.util.Properties;

public class TestDataSource {
 private static DruidDataSource dataSource;

 public static void main(String[] args) throws Exception {
 Properties properties = loadProperties();
 dataSource = (DruidDataSource) DruidDataSourceFactory.createDataSource(properties);
 dataSource.setTimeBetweenEvictionRunsMillis(50 * 1000);
 dataSource.setRemoveAbandoned(true);
 dataSource.setRemoveAbandonedTimeout(120);
 dataSource.setSocketTimeout(5000);
 dataSource.setConnectTimeout(5000);
 dataSource.setQueryTimeout(5);
 final DruidPooledConnection connection = dataSource.getConnection();
 final Statement statement = connection.createStatement();
 execute(statement);
 }

 public static void execute(Statement statement) {
 ResultSet resultSet = null;
 try {
 resultSet = statement.executeQuery("select 1");
 while (resultSet.next()) {
 String str = resultSet.getString(1);
 System.out.println("n1 :" + str);
 }
 return;
 } catch (Exception e) {
 e.printStackTrace();
 }
 }

 /**
 * 加载配置文件，从配置文件获取参数
 */
 public static Properties loadProperties() {
 InputStream inputStream =
TestDataSource.class.getClassLoader().getResourceAsStream("db.properties");
 Properties ps = new Properties();
 try {
 ps.load(inputStream);
 } catch (IOException e) {
 e.printStackTrace();
 }
 return ps;
 }
}
```

----结束

### 6.5.1.7 使用 ODBC 连接

DWS支持使用ODBC应用程序连接数据库。应用程序可以在华为云平台环境的弹性云服务器中，或者互联网环境连接数据库。

ODBC接口的使用方法，请自行查阅官方文档。

#### 前提条件

- 已下载Linux版本的ODBC驱动包“dws\_x.x.x\_odbc\_driver\_for\_xxx.zip”和Windows版本的ODBC驱动包“dws\_odbc\_driver\_for\_windows.zip”，请参见[下载JDBC或ODBC驱动](#)。  
DWS也支持开源的ODBC驱动程序：PostgreSQL ODBC 09.01.0200或更高版本。
- 已下载开源unixODBC代码文件，支持版本为2.3.0，下载地址：<https://sourceforge.net/projects/unixodbc/files/unixODBC/2.3.0/unixODBC-2.3.0.tar.gz/download>
- 已下载SSL证书文件，请参见[下载SSL证书](#)。

#### 在 Linux 环境使用 ODBC 连接

**步骤1** 将ODBC驱动包和代码文件上传到Linux环境，并解压到指定目录。

**步骤2** 登录Linux环境。

**步骤3** 准备unixODBC。

- 解压unixODBC代码文件。  

```
tar -xvf unixODBC-2.3.0.tar.gz
```
- 编译并安装。  

```
cd unixODBC-2.3.0
./configure --enable-gui=no --prefix=[your_path]
make
make install
```

##### 说明

- 其中 “[your\_path]” 表示安装路径，可自由指定，需要绝对路径。
- 此时unixODBC编译安装完成后，安装目录下会有\*.so.2的库文件，如果需要编译出\*.so.1的库文件，需要将configure文件中的LIB\_VERSION修改为：  

```
LIB_VERSION="1:0:0"
LIB_VERSION="1:0:0"
```
- 在本驱动中，会动态的加载库文件libodbcinst.so.\*，加载成功其中的一个则完成对该库文件的加载。**其中加载的优先级为libodbcinst.so>libodbcinst.so.1>libodbcinst.so.1.0.0>libodbcinst.so.2>libodbcinst.so.2.0.0。**  
例如在某目录下可以动态的链接到libodbcinst.so.1、libodbcinst.so.1.0.0、libodbcinst.so.2。驱动文件会先加载libodbcinst.so，如果当前环境中无法找到libodbcinst.so，则会继续寻找优先级低的libodbcinst.so.1，当成功的加载到libodbcinst.so.1后即完成了对该动态链接库的加载。

**步骤4** 替换驱动文件（本文以redhat操作系统的包dws\_8.1.x\_odbc\_driver\_for\_x86\_redhat.zip为例）。

- 解压“dws\_8.1.x\_odbc\_driver\_for\_x86\_redhat.zip”。  

```
unzip dws_8.1.x_odbc_driver_for_x86_redhat.zip
```
- 将“dws\_8.1.x\_odbc\_driver\_for\_x86\_redhat.zip”解压后“lib”目录下所有文件，替换到 “[your\_path]/lib”。

3. 将“dws\_8.1.x\_odbc\_driver\_for\_x86\_redhat.zip”解压后“odbc/lib”目录下的“psqlodbcw.la”和“psqlodbcw.so”，保存到 “[your\_path]/lib/odbc”。

**步骤5** 执行以下命令，修改驱动文件配置。

```
vi [your_path]/etc/odbcinst.ini
```

将以下内容保存到配置中：

```
[DWS]
Driver64=[your_path]/lib/odbc/psqlodbcw.so
```

参数说明如下：

- “[DWS]”：表示驱动器名称，支持自定义。
- “Driver64”或“Driver”：表示驱动动态库的路径。64位系统优先查找“Driver64”配置项，如果未配置则会继续查找“Driver”。

**步骤6** 执行以下命令，修改数据源文件。

```
vi [your_path]/etc/odbc.ini
```

将以下内容保存到配置文件中，并退出修改。

```
[DWSODBC]
Driver=DWS
Servername=10.10.0.13
Database=gaussdb
Username=dbadmin
Password=password
Port=8000
Sslmode=allow
```

| 参数名        | 说明                                       | 参数值样例                 |
|------------|------------------------------------------|-----------------------|
| [DSN]      | 数据源的名称。                                  | [DWSODBC]             |
| Driver     | 驱动名称，对应“odbcinst.ini”中的DriverName。       | Driver=DWS            |
| Servername | 服务器的IP地址，当集群绑定弹性负载均衡(ELB)时，应设置为ELB的IP地址。 | Servername=10.10.0.13 |
| Database   | 要连接的数据库的名称。                              | Database=gaussdb      |
| Username   | 数据库用户名称。                                 | Username=dbadmin      |
| Password   | 数据库用户密码。                                 | Password=password     |
| Port       | 服务器的端口号。                                 | Port=8000             |

| 参数名     | 说明                                                                                                                                                                                                                                                                                                                                                                                                                                                      | 参数值样例         |
|---------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------|
| Sslmode | <p>SSL认证工作模式。集群默认开启。</p> <p>取值及含义：</p> <ul style="list-style-type: none"><li>• disable：只尝试非SSL连接。</li><li>• allow：首先尝试非SSL连接，如果连接失败，再尝试SSL连接。</li><li>• prefer：首先尝试SSL连接，如果连接失败，将尝试非SSL连接。</li><li>• require：只尝试SSL连接。如果存在CA文件，则按设置成verify-ca的方式验证。</li><li>• verify-ca：只尝试SSL连接，并且验证服务器是否具有由可信任的证书机构签发的证书。</li><li>• verify-full：DWS不支持此模式。</li></ul> <p><b>说明</b></p> <p>SSL模式安全性高于普通模式，集群默认开启SSL功能允许来自客户端的SSL连接或非SSL连接，建议在使用ODBC连接DWS集群时采用SSL模式。</p> | Sslmode=allow |

 说明

- 其中，参数“Servername”和“Port”的值，可以在DWS的管理控制台查看。请登录DWS管理控制台，单击“连接客户端”，在“数据仓库连接字符串”区域，选择指定的集群，获取该集群的“内网访问地址”或“公网访问地址”。具体步骤请参见[获取DWS集群连接地址](#)。

**步骤7** 配置环境变量。

```
vi ~/.bashrc
```

在配置文件中追加以下内容。

```
export LD_LIBRARY_PATH=[your_path]/lib/:$LD_LIBRARY_PATH
export ODBCYSINI=[your_path]/etc
export ODBCINI=[your_path]/etc/odbc.ini
```

 说明

麒麟OS环境中，此处不建议追加LD\_LIBRARY\_PATH，可能造成libssl.so库冲突。集群8.2.1和9.1.0最新版本中已增加rpath，不需要LD\_LIBRARY\_PATH即可找到依赖。

**步骤8** 导入环境变量。

```
source ~/.bashrc
```

**步骤9** 执行以下命令，开始连接。

```
[your_path]/bin/isql -v DWSODBC
```

界面显示以下信息表示连接成功：

```
+-----+
| Connected! |
| |
| sql-statement |
| help [tablename] |
| quit |
| |
+-----+
SQL>
```

----结束

## 在 Windows 环境使用 ODBC 连接

**步骤1** 解压Windows版本的ODBC驱动包“dws\_odbc\_driver\_for\_windows.zip”，并安装“psqlodbc.msi”。

**步骤2** 解压SSL证书压缩包，并准备证书文件。

用户可以根据实际情况选择自动或手动部署方法：

- 自动部署：

双击“sslcert\_env.bat”文件，即可完成证书的默认位置的部署。

 **说明**

该sslcert\_env.bat为了保证证书环境的纯净，在%APPDATA%\postgresql目录存在时，会提示是否需要移除相关目录。如果有需要，请备份该目录中的文件。

- 手动部署：

- 在“%APPDATA%\”目录创建一个新文件夹，并命名为“postgresql”。
- 将证书文件包中的“client.crt”、“client.key”、“client.key.cipher”、“client.key.rand”文件保存至“%APPDATA%\postgresql”目录，并且将文件名中的client改为**postgres**，例如“client.key”修改为“postgres.key”。
- 将“cacert.pem”文件保存至“%APPDATA%\postgresql”目录，并更名为“root.crt”。

**步骤3** 打开驱动管理器。

因为目前DWS提供了32位和64位的ODBC驱动程序，用户可根据自身需求选择；在配置数据源时，请使用对应的驱动管理器（假设操作系统安装盘符为C:盘，如果是其他盘符，请对路径做相应修改）：

- 64位操作系统上进行**32**位程序开发，安装**32**位程序驱动后。使用**32**位的驱动管理器：C:\Windows\SysWOW64\odbcad32.exe

请勿直接使用“控制面板> 管理工具 > 数据源（ODBC）”。

 **说明**

WoW64的全称是“Windows 32-bit on Windows 64-bit”，C:\Windows\SysWOW64\存放的是64位系统上的32位运行环境。

- 64操作系统上进行**64**位程序开发，安装**64**位驱动程序后，使用**64**位的驱动管理器：C:\Windows\System32\odbcad32.exe

请勿直接使用“控制面板 > 管理工具> 数据源（ODBC）”。

说明

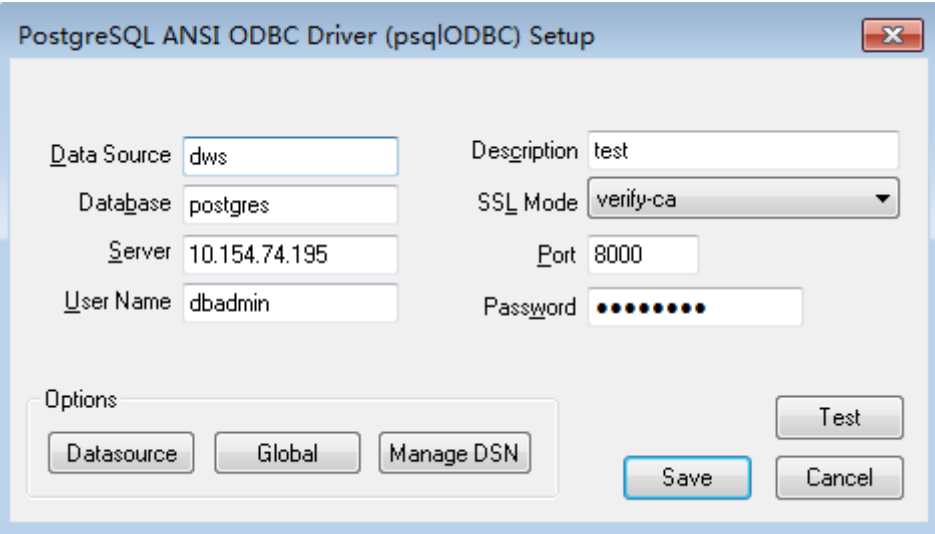
C:\Windows\System32\存放的是与操作系统一致的运行环境，具体的技术信息请查阅 Windows 的相关技术文档。

- 32位操作系统请使用：C:\Windows\System32\odbcad32.exe  
或者单击“计算机 > 控制面板 > 管理工具 > 数据源（ODBC）”打开驱动管理器。

步骤4 配置连接数据源。

- 在打开的驱动管理器上，选择“用户DSN > 添加 > PostgreSQL Unicode”，然后进行配置。

图 6-7 配置连接数据源



其中，配置项“Server”和“Port”的值，可以在DWS的管理控制台查看。请登录DWS管理控制台，单击“连接客户端”，在“数据仓库连接字符串”区域，选择指定的集群，获取该集群的“内网访问地址”或“公网访问地址”。具体步骤请参见[获取DWS集群连接地址](#)。

- 单击“Test”验证连接正确，界面提示“Connection successful”。

步骤5 编写ODBC样例程序连接数据源。

ODBC接口不提供重试连接数据库的能力，您需要在业务代码中实现重试连接的处理。

样例代码如下：

```
// 此示例演示如何通过ODBC方式获取DWS中的数据。
// DBtest.c (compile with: libodbc.so)
#include <stdlib.h>
#include <stdio.h>
#include <sqlext.h>
#ifdef WIN32
#include <windows.h>
#endif
SQLHENV V_OD_Env; // Handle ODBC environment
SQLHSTMT V_OD_hstmt; // Handle statement
SQLHDBC V_OD_hdbc; // Handle connection
char typename[100];
SQLINTEGER value = 100;
SQLINTEGER V_OD_erg,V_OD_buffer,V_OD_err,V_OD_id;
```

```
int main(int argc, char *argv[])
{
 // 1. 申请环境句柄
 V_OD_erg = SQLAllocHandle(SQL_HANDLE_ENV, SQL_NULL_HANDLE, &V_OD_Env);
 if ((V_OD_erg != SQL_SUCCESS) && (V_OD_erg != SQL_SUCCESS_WITH_INFO))
 {
 printf("Error AllocHandle\n");
 exit(0);
 }
 // 2. 设置环境属性 (版本信息)
 SQLSetEnvAttr(V_OD_Env, SQL_ATTR_ODBC_VERSION, (void*)SQL_OV_ODBC3, 0);
 // 3. 申请连接句柄
 V_OD_erg = SQLAllocHandle(SQL_HANDLE_DBC, V_OD_Env, &V_OD_hdbc);
 if ((V_OD_erg != SQL_SUCCESS) && (V_OD_erg != SQL_SUCCESS_WITH_INFO))
 {
 SQLFreeHandle(SQL_HANDLE_ENV, V_OD_Env);
 exit(0);
 }
 // 4. 设置连接属性
 SQLSetConnectAttr(V_OD_hdbc, SQL_ATTR_AUTOCOMMIT, SQL_AUTOCOMMIT_ON, 0);
 // 5. 连接数据源, 其中的用户名和用户密码等信息之所以能省略, 是因为在odbc.ini文件中进行了配置, 若没
 // 配置需要在SQLConnect函数参数中具体写明要连接数据库的用户名和用户密码等信息。
 V_OD_erg = SQLConnect(V_OD_hdbc, (SQLCHAR*) "gaussdb", SQL_NTS,
 (SQLCHAR*) "", SQL_NTS, (SQLCHAR*) "", SQL_NTS);
 if ((V_OD_erg != SQL_SUCCESS) && (V_OD_erg != SQL_SUCCESS_WITH_INFO))
 {
 printf("Error SQLConnect %d\n", V_OD_erg);
 SQLFreeHandle(SQL_HANDLE_ENV, V_OD_Env);
 exit(0);
 }
 printf("Connected !\n");
 // 6. 设置语句属性
 SQLSetStmtAttr(V_OD_hstmt, SQL_ATTR_QUERY_TIMEOUT, (SQLPOINTER *)3, 0);
 // 7. 申请语句句柄
 SQLAllocHandle(SQL_HANDLE_STMT, V_OD_hdbc, &V_OD_hstmt);
 // 8. 直接执行SQL语句。
 SQLExecDirect(V_OD_hstmt, "drop table IF EXISTS testtable", SQL_NTS);
 SQLExecDirect(V_OD_hstmt, "create table testtable(id int)", SQL_NTS);
 SQLExecDirect(V_OD_hstmt, "insert into testtable values(25)", SQL_NTS);
 // 9. 准备执行
 SQLPrepare(V_OD_hstmt, "insert into testtable values(?)", SQL_NTS);
 // 10. 绑定参数
 SQLBindParameter(V_OD_hstmt, 1, SQL_PARAM_INPUT, SQL_C_SLONG, SQL_INTEGER, 0, 0,
 &value, 0, NULL);
 // 11. 执行准备好的语句
 SQLExecute(V_OD_hstmt);
 SQLExecDirect(V_OD_hstmt, "select id from testtable", SQL_NTS);
 // 12. 获取结果集某一列的属性
 SQLColAttribute(V_OD_hstmt, 1, SQL_DESC_TYPE, typename, 100, NULL, NULL);
 printf("SQLColAttribute %s\n", typename);
 // 13. 绑定结果集
 SQLBindCol(V_OD_hstmt, 1, SQL_C_SLONG, (SQLPOINTER)&V_OD_buffer, 150,
 (SQLLEN *)&V_OD_err);
 // 14. 通过SQLFetch取结果集中数据
 V_OD_erg = SQLFetch(V_OD_hstmt);
 // 15. 通过SQLGetData获取并返回数据。
 while(V_OD_erg != SQL_NO_DATA)
 {
 SQLGetData(V_OD_hstmt, 1, SQL_C_SLONG, (SQLPOINTER)&V_OD_id, 0, NULL);
 printf("SQLGetData ----ID = %d\n", V_OD_id);
 V_OD_erg = SQLFetch(V_OD_hstmt);
 }
 printf("Done !\n");
 // 16. 断开数据源连接并释放句柄资源
 SQLFreeHandle(SQL_HANDLE_STMT, V_OD_hstmt);
 SQLDisconnect(V_OD_hdbc);
 SQLFreeHandle(SQL_HANDLE_DBC, V_OD_hdbc);
 SQLFreeHandle(SQL_HANDLE_ENV, V_OD_Env);
}
```

```
 return(0);
}
```

----结束

## 6.5.2 使用 Python 第三方库 psycopg2 连接 DWS 集群

用户在创建好数据仓库集群后使用psycopg2第三方库连接到集群，则可以使用Python访问DWS，并进行数据表的各类操作。

### 连接集群前的准备

- DWS集群已绑定弹性IP。
- 已获取DWS集群的数据库管理员用户名和密码。

请注意，由于MD5算法已经被证实存在碰撞可能，已严禁将之用于密码校验算法。当前DWS采用默认安全设计，默认禁止MD5算法的密码校验，可能导致开源客户端无法正常连接的问题。建议先联系技术支持人员检查数据库参数password\_encryption\_type参数是否为1，如果取值不为1，需要修改；然后修改一次准备使用的数据库用户的密码。

#### 📖 说明

- 当前DWS出于安全考虑，已经默认不再使用MD5存储密码摘要了，这将导致使用开源驱动或者客户端无法正常连接数据库。需要您调整密码策略后再创建一个新用户或者对老用户做一次密码修改，方可使用开源协议中的MD5认证算法。
- 数据库中是不会存储用户的密码原文，而是存储密码的HASH摘要，在密码校验时与客户端发来的密码摘要进行比对（中间会有加盐操作）。故当您改变了密码算法策略时，数据库也是无法还原您的密码，再生成新的HASH算法的摘要值的。必须您手动修改一次密码或者创建一个新用户，这时新的密码将会采用您设置的HASH算法进行摘要存储，用于下次连接认证。
- 已获取DWS集群的公网访问地址，含IP地址和端口。具体请参见[获取DWS集群连接地址](#)。
- 已安装psycopg2第三方库。下载地址：<https://pypi.org/project/psycopg2/>，安装部署操作请参见：<https://www.psycopg.org/install/>。

#### 📖 说明

- CentOS、Redhat等操作系统中使用yum命令安装，命令为：  
yum install python-psycopg2
- psycopg2的使用依赖于PostgreSQL的libpq动态库（32位的psycopg2需要对应32位的libpq；64位的psycopg2对应64位的libpq），Linux中可以依赖yum命令解决。在Windows系统使用psycopg2需要先安装libpq，主要方式有两种：
  - 安装PostgreSQL，并配置libpq、ssl、crypto动态库位置到环境变量PATH中。
  - 安装psqlodbc，使用PostgreSQL ODBC驱动携带的libpq、ssl、crypto动态库。

### 版本说明

由于DWS集群、Python、psycopg2的版本较多，下方表格仅列举出当前主流版本的支持情况。

表 6-14

| psycopg2版本 | Python版本 | DWS集群版本  |
|------------|----------|----------|
| 2.7.x      | 3.8.x    | 8.1.3及以上 |
|            | 3.9.x    | 8.1.3及以上 |
| 2.8.x      | 3.8.x    | 8.1.3及以上 |
|            | 3.9.x    | 8.1.3及以上 |
| 2.9.x      | 3.8.x    | 8.1.3及以上 |
|            | 3.9.x    | 8.1.3及以上 |

使用约束

由于psycopg2是基于PostgreSQL的客户端接口，它的功能DWS并不能完全支持。具体支持情况请见下[表6-15](#)。

说明

以下接口支持情况是基于Python 3.8.5及psycopg 2.9.1版本。

表 6-15 DWS 对 psycopg2 主要接口支持情况

| 类名          | 功能描述                             | 函数/成员变量                                                                              | 支持 | 备注                          |
|-------------|----------------------------------|--------------------------------------------------------------------------------------|----|-----------------------------|
| connections | basic                            | <code>cursor(name=None, cursor_factory=None, scrollable=None, withhold=False)</code> | Y  | -                           |
|             |                                  | <code>commit()</code>                                                                | Y  | -                           |
|             |                                  | <code>rollback()</code>                                                              | Y  | -                           |
|             |                                  | <code>close()</code>                                                                 | Y  | -                           |
|             | Two-phase commit support methods | <code>xid(format_id, gtrid, bqual)</code>                                            | Y  | -                           |
|             |                                  | <code>tpc_begin(xid)</code>                                                          | Y  | -                           |
|             |                                  | <code>tpc_prepare()</code>                                                           | N  | 内核不支持显式prepare transaction。 |
|             |                                  | <code>tpc_commit([xid])</code>                                                       | Y  | -                           |
|             |                                  | <code>tpc_rollback([xid])</code>                                                     | Y  | -                           |
|             |                                  | <code>tpc_recover()</code>                                                           | Y  | -                           |
|             |                                  | <code>closed</code>                                                                  | Y  | -                           |
|             |                                  | <code>cancel()</code>                                                                | Y  | -                           |

| 类名 | 功能描述                                        | 函数/成员变量                                                                                     | 支持 | 备注                                                |
|----|---------------------------------------------|---------------------------------------------------------------------------------------------|----|---------------------------------------------------|
|    |                                             | reset()                                                                                     | N  | 不支持 DISCARD ALL。                                  |
|    |                                             | dsn                                                                                         | Y  | -                                                 |
|    | Transaction control methods and attributes. | set_session( <i>isolation_level=None, readonly=None, deferrable=None, autocommit=None</i> ) | Y  | 数据库不支持 session 中设置 default_transaction_read_only。 |
|    |                                             | autocommit                                                                                  | Y  | -                                                 |
|    |                                             | isolation_level                                                                             | Y  | -                                                 |
|    |                                             | readonly                                                                                    | N  | 数据库不支持 session 中设置 default_transaction_read_only。 |
|    |                                             | deferrable                                                                                  | Y  | -                                                 |
|    |                                             | set_isolation_level( <i>level</i> )                                                         | Y  | -                                                 |
|    |                                             | encoding                                                                                    | Y  | -                                                 |
|    |                                             | set_client_encoding(enc)                                                                    | Y  | -                                                 |
|    |                                             | notices                                                                                     | N  | 数据库不支持 listen/notify。                             |
|    |                                             | notifies                                                                                    | Y  | -                                                 |
|    |                                             | cursor_factory                                                                              | Y  | -                                                 |
|    |                                             | info                                                                                        | Y  | -                                                 |
|    |                                             | status                                                                                      | Y  | -                                                 |
|    |                                             | object                                                                                      | N  | 数据库不支持大对象相关操作。                                    |
|    | Methods related to asynchronous support     | poll()                                                                                      | Y  | -                                                 |
|    |                                             | fileno()                                                                                    | Y  | -                                                 |
|    |                                             | isexecuting()                                                                               | Y  | -                                                 |
|    | Interoperation with other C API modules     | pgconn_ptr                                                                                  | Y  | -                                                 |
|    |                                             | get_native_connection()                                                                     | Y  | -                                                 |

| 类名     | 功能描述                                         | 函数/成员变量                            | 支持 | 备注                            |
|--------|----------------------------------------------|------------------------------------|----|-------------------------------|
|        | informative methods of the native connection | get_transaction_status()           | Y  | -                             |
|        |                                              | protocol_version                   | Y  | -                             |
|        |                                              | server_version                     | Y  | -                             |
|        |                                              | get_backend_pid()                  | Y  | 获取到的不是后台的pid, 是逻辑连接的id号。      |
|        |                                              | get_parameter_status(parameter)    | Y  | -                             |
|        |                                              | get_dsn_parameters()               | Y  | -                             |
| cursor | basic                                        | description                        | Y  | -                             |
|        |                                              | close()                            | Y  | -                             |
|        |                                              | closed                             | Y  | -                             |
|        |                                              | connection                         | Y  | -                             |
|        |                                              | name                               | Y  | -                             |
|        |                                              | scrollable                         | N  | 数据库不支持 SCROLL CURSOR。         |
|        |                                              | withhold                           | N  | withhold cursor在 commit前需要关闭。 |
|        | Commands execution methods                   | execute(query, vars=None)          | Y  | -                             |
|        |                                              | executemany(query, vars_list)      | Y  | -                             |
|        |                                              | callproc(procname[, parameters])   | Y  | -                             |
|        |                                              | mogrify(operation[, parameters])   | Y  | -                             |
|        |                                              | setinputsizes(sizes)               | Y  | -                             |
|        |                                              | fetchone()                         | Y  | -                             |
|        |                                              | fetchmany([size=cursor.arraysize]) | Y  | -                             |
|        |                                              | fetchall()                         | Y  | -                             |
|        |                                              | scroll(value[, mode='relative'])   | N  | 数据库不支持 SCROLL CURSOR。         |

| 类名 | 功能描述                                                 | 函数/成员变量                                                                                                                         | 支持 | 备注 |
|----|------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------|----|----|
|    |                                                      | arraysize                                                                                                                       | Y  | -  |
|    |                                                      | itersize                                                                                                                        | Y  | -  |
|    |                                                      | rowcount                                                                                                                        | Y  | -  |
|    |                                                      | rownumber                                                                                                                       | Y  | -  |
|    |                                                      | lastrowid                                                                                                                       | Y  | -  |
|    |                                                      | query                                                                                                                           | Y  | -  |
|    |                                                      | statusmessage                                                                                                                   | Y  | -  |
|    |                                                      | cast( <i>oid</i> , <i>s</i> )                                                                                                   | Y  | -  |
|    |                                                      | tzinfo_factory                                                                                                                  | Y  | -  |
|    |                                                      | nextset()                                                                                                                       | Y  | -  |
|    |                                                      | setoutputsize( <i>size</i> [, <i>column</i> ])                                                                                  | Y  | -  |
|    | COPY-related methods                                 | copy_from( <i>file</i> , <i>table</i> , <i>sep</i> =' <br> t', <i>null</i> ='   N', <i>size</i> =8192,<br><i>columns</i> =None) | Y  | -  |
|    |                                                      | copy_to( <i>file</i> , <i>table</i> , <i>sep</i> =' t',<br><i>null</i> ='   N', <i>columns</i> =None)                           | Y  | -  |
|    |                                                      | copy_expert( <i>sql</i> , <i>file</i> ,<br><i>size</i> =8192)                                                                   | Y  | -  |
|    | Interopera<br>tion with<br>other C<br>API<br>modules | pgresult_ptr                                                                                                                    | Y  | -  |

在 Linux 环境使用 psycopg2 第三方库连接集群

- 步骤1 以root用户登录Linux环境。
- 步骤2 执行以下命令创建python\_dws.py文件。

```
vi python_dws.py

请复制粘贴以下内容放入python_dws.py文件中:

#!/usr/bin/python
-*- coding: UTF-8 -*-

from __future__ import print_function

import psycopg2

def create_table(connection):
 print("Begin to create table")
```

```
try:
 cursor = connection.cursor()
 cursor.execute("drop table if exists test;"
 "create table test(id int, name text);")
 connection.commit()
except psycopg2.ProgrammingError as e:
 print(e)
else:
 print("Table created successfully")
 cursor.close()

def insert_data(connection):
 print("Begin to insert data")
 try:
 cursor = connection.cursor()
 cursor.execute("insert into test values(1,'number1');")
 cursor.execute("insert into test values(2,'number2');")
 cursor.execute("insert into test values(3,'number3');")
 connection.commit()
 except psycopg2.ProgrammingError as e:
 print(e)
 else:
 print("Insert data successfully")
 cursor.close()

def update_data(connection):
 print("Begin to update data")
 try:
 cursor = connection.cursor()
 cursor.execute("update test set name = 'numberupdated' where id=1;")
 connection.commit()
 print("Total number of rows updated :", cursor.rowcount)
 cursor.execute("select * from test order by 1;")
 rows = cursor.fetchall()
 for row in rows:
 print("id = ", row[0])
 print("name = ", row[1], "\n")
 except psycopg2.ProgrammingError as e:
 print(e)
 else:
 print("After Update, Operation done successfully")

def delete_data(connection):
 print("Begin to delete data")
 try:
 cursor = connection.cursor()
 cursor.execute("delete from test where id=3;")
 connection.commit()
 print("Total number of rows deleted :", cursor.rowcount)
 cursor.execute("select * from test order by 1;")
 rows = cursor.fetchall()
 for row in rows:
 print("id = ", row[0])
 print("name = ", row[1], "\n")
 except psycopg2.ProgrammingError as e:
 print(e)
 else:
 print("After Delete, Operation done successfully")

def select_data(connection):
 print("Begin to select data")
 try:
 cursor = connection.cursor()
 cursor.execute("select * from test order by 1;")
 rows = cursor.fetchall()
```

```
for row in rows:
 print("id = ", row[0])
 print("name = ", row[1], "\n")
except psycopg2.ProgrammingError as e:
 print(e)
 print("select failed")
else:
 print("Operation done successfully")
 cursor.close()

if __name__ == '__main__':
 try:
 conn = psycopg2.connect(host='10.154.70.231',
 port='8000',
 database='gaussdb', # 需要连接的database
 user='dbadmin',
 password='password') # 数据库用户密码
 except psycopg2.DatabaseError as ex:
 print(ex)
 print("Connect database failed")
 else:
 print("Opened database successfully")
 create_table(conn)
 insert_data(conn)
 select_data(conn)
 update_data(conn)
 delete_data(conn)
 conn.close()
```

**步骤3** 按照实际集群信息，修改python\_dws.py文件中的集群公网访问地址、集群端口号、数据库名称、数据库用户名、数据库密码。

psycopg2接口不提供重试连接的能力，您需要在业务代码中实现重试处理。

```
conn = psycopg2.connect(host='10.154.70.231',
 port='8000',
 database='gaussdb', # 需要连接的database
 user='dbadmin',
 password='password') # 数据库用户密码
```

**步骤4** 执行以下命令，使用psycopg第三方库连接集群。

```
python python_dws.py
```

----结束

## 在 Windows 环境使用 psycopg2 第三方库连接集群

**步骤1** 在Windows系统中，单击“开始”按钮，在搜索框中，键入cmd，然后在结果列表中单击“cmd.exe”打开命令提示符窗口。

**步骤2** 在命令提示符窗口中，执行以下命令创建python\_dws.py文件。

```
type nul> python_dws.py
```

请复制粘贴以下内容放入python\_dws.py文件中：

```
#!/usr/bin/python
-*- coding:UTF-8 -*-

from __future__ import print_function

import psycopg2

def create_table(connection):
 print("Begin to create table")
 try:
```

```
 cursor = connection.cursor()
 cursor.execute("drop table if exists test;"
 "create table test(id int, name text);")
 connection.commit()
 except psycopg2.ProgrammingError as e:
 print(e)
 else:
 print("Table created successfully")
 cursor.close()

def insert_data(connection):
 print("Begin to insert data")
 try:
 cursor = connection.cursor()
 cursor.execute("insert into test values(1,'number1');")
 cursor.execute("insert into test values(2,'number2');")
 cursor.execute("insert into test values(3,'number3');")
 connection.commit()
 except psycopg2.ProgrammingError as e:
 print(e)
 else:
 print("Insert data successfully")
 cursor.close()

def update_data(connection):
 print("Begin to update data")
 try:
 cursor = connection.cursor()
 cursor.execute("update test set name = 'numberupdated' where id=1;")
 connection.commit()
 print("Total number of rows updated :", cursor.rowcount)
 cursor.execute("select * from test order by 1;")
 rows = cursor.fetchall()
 for row in rows:
 print("id = ", row[0])
 print("name = ", row[1], "\n")
 except psycopg2.ProgrammingError as e:
 print(e)
 else:
 print("After Update, Operation done successfully")

def delete_data(connection):
 print("Begin to delete data")
 try:
 cursor = connection.cursor()
 cursor.execute("delete from test where id=3;")
 connection.commit()
 print("Total number of rows deleted :", cursor.rowcount)
 cursor.execute("select * from test order by 1;")
 rows = cursor.fetchall()
 for row in rows:
 print("id = ", row[0])
 print("name = ", row[1], "\n")
 except psycopg2.ProgrammingError as e:
 print(e)
 else:
 print("After Delete, Operation done successfully")

def select_data(connection):
 print("Begin to select data")
 try:
 cursor = connection.cursor()
 cursor.execute("select * from test order by 1;")
 rows = cursor.fetchall()
 for row in rows:
```

```
 print("id = ", row[0])
 print("name = ", row[1], "\n")
 except psycopg2.ProgrammingError as e:
 print(e)
 print("select failed")
 else:
 print("Operation done successfully")
 cursor.close()

if __name__ == '__main__':
 try:
 conn = psycopg2.connect(host='10.154.70.231',
 port='8000',
 database='postgresgaussdb', # 需要连接的database
 user='dbadmin',
 password='password') # 数据库用户密码
 except psycopg2.DatabaseError as ex:
 print(ex)
 print("Connect database failed")
 else:
 print("Opened database successfully")
 create_table(conn)
 insert_data(conn)
 select_data(conn)
 update_data(conn)
 delete_data(conn)
 conn.close()
```

**步骤3** 按照实际集群信息，修改python\_dws.py文件中的集群公网访问地址、集群端口号、数据库名称、数据库用户名、数据库密码。

```
conn = psycopg2.connect(host='10.154.70.231',
 port='8000',
 database='gaussdb', # 需要连接的database
 user='dbadmin',
 password='password') # 数据库用户密码
```

**步骤4** 在命令提示符窗口中，执行以下命令，使用psycopg2第三方库连接集群。

```
python python_dws.py
```

----结束

## psycopg2 连接集群不支持 CN Retry 特性的问题说明

DWS支持在SQL语句执行出错时的自动重试功能（简称CN Retry）。CN Retry对于客户端和驱动发送的SQL语句在执行失败时可以自动识别错误类型，并进行重试。但使用psycopg2默认连接方式创建的连接在语句执行失败时没有自动重试，会直接报错退出。如常见的主备切换场景下，未自动重试会报如下错误，但在自动重试期间完成主备切换，则会返回正确结果。

```
psycopg2.errors.ConnectionFailure: pooler: failed to create 1 connections, Error Message: remote node
dn_6003_6004, detail: could not connect to server: Operation now in progress
```

### 报错原因：

1. psycopg2在发送SQL语句前先发送了BEGIN语句开启事务。
2. CN Retry不支持事务块中的语句是特性约束。

### 解决方案：

- 在同步方式连接时，可以通过主动结束驱动开启的事务。

```
cursor = conn.cursor()
增加end语句主动结束驱动开启的事务
cursor.execute("end; select * from test order by 1;")
rows = cursor.fetchall()
```

- 使用异步连接方式主动开启事务，异步连接介绍具体请参见psycopg官网：  
<https://www.psycopg.org/docs/advanced.html?highlight=async>。

```
#!/usr/bin/env python3
-*- encoding=utf-8 -*-

import psycopg2
import select

psycopg2官方提供的异步连接方式时的wait函数
详见https://www.psycopg.org/docs/advanced.html?highlight=async
def wait(conn):
 while True:
 state = conn.poll()
 if state == psycopg2.extensions.POLL_OK:
 break
 elif state == psycopg2.extensions.POLL_WRITE:
 select.select([], [conn.fileno()], [])
 elif state == psycopg2.extensions.POLL_READ:
 select.select([conn.fileno()], [], [])
 else:
 raise psycopg2.OperationalError("poll() returned %s" % state)

def psycopg2_cnretry_sync():
 # 创建连接
 conn = psycopg2.connect(host='10.154.70.231',
 port='8000',
 database='gaussdb', # 需要连接的database
 user='dbadmin',
 password='password', # 数据库用户密码
 async=1) # 使用异步方式连接

 wait(conn)

 # 执行查询
 cursor = conn.cursor()
 cursor.execute("select * from test order by 1;")
 wait(conn)
 rows = cursor.fetchall()
 for row in rows:
 print(row[0], row[1])

 # 关闭连接
 conn.close()

if __name__ == '__main__':
 psycopg2_cnretry_async()
```

### 6.5.3 使用 Python 第三方库 PyGreSQL 连接 DWS 集群

用户在创建好数据仓库集群后使用PyGreSQL第三方库连接到集群，则可以使用Python访问DWS，并进行数据表的各类操作。

#### 连接集群前的准备

- DWS集群已绑定弹性IP。
- 已获取DWS集群的数据库管理员用户名和密码。

请注意，由于MD5算法已经被证实存在碰撞可能，已严禁将之用于密码校验算法。当前DWS采用默认安全设计，默认禁止MD5算法的密码校验，可能导致开源客户端无法正常连接的问题。建议先联系技术支持人员检查数据库参数password\_encryption\_type参数是否为1，如果取值不为1，需要修改；然后修改一次准备使用的数据库用户的密码。

说明

- 当前DWS出于安全考虑，已经默认不再使用MD5存储密码摘要了，这将导致使用开源驱动或者客户端无法正常连接数据库。需要您调整密码策略后再创建一个新用户或者对老用户做一次密码修改，方可使用开源协议中使用的MD5认证算法。
- 数据库中是不会存储您的密码原文的，而是存储的密码的HASH摘要，在密码校验时与客户端发来的密码摘要进行比对（中间会有加盐操作）。故当您改变了密码算法策略时，数据库也是无法还原您的密码，再生成新的HASH算法的摘要值的。必须您手动修改一次密码或者创建一个新用户，这时新的密码将会采用您设置的HASH算法进行摘要存储，用于下次连接认证。
- 已获取DWS集群的公网访问地址，含IP地址和端口。具体请参见[获取DWS集群连接地址](#)。
- 已安装PyGreSQL第三方库。  
下载地址：<http://www.pygresql.org/download/index.html>。
- 安装部署操作请参见：<http://www.pygresql.org/contents/install.html>。

说明

- CentOS、Redhat等操作系统中使用yum命令安装，命令为：  
yum install PyGreSQL
- PyGreSQL的使用依赖于PostgreSQL的libpq动态库（32位的PyGreSQL对应32位的libpq，64位的PyGreSQL对应64位的libpq），Linux中可以依赖yum命令解决。在Windows系统使用PyGreSQL需要先安装libpq，主要方式有两种：
  - 安装PostgreSQL，并配置libpq、ssl、crypto动态库位置到环境变量PATH中。
  - 安装psqlodbc，使用PostgreSQL ODBC驱动携带的libpq、ssl、crypto动态库。

使用约束

由于PyGreSQL是基于PostgreSQL的客户端接口，它的功能DWS并不能完全支持。具体支持情况请见下表。

说明

以下接口支持情况是基于Python 3.8.5及PyGreSQL 5.2.4版本。

表 6-16 DWS 对 PyGreSQL 主要接口支持情况

| PyGreSQL                       |                                                    | 支持 | 备注 |
|--------------------------------|----------------------------------------------------|----|----|
| Module functions and constants | connect – Open a PostgreSQL connection             | Y  | -  |
|                                | get_pqlib_version – get the version of libpq       | Y  | -  |
|                                | get/set_defhost – default server host [DV]         | Y  | -  |
|                                | get/set_defport – default server port [DV]         | Y  | -  |
|                                | get/set_defopt – default connection options [DV]   | Y  | -  |
|                                | get/set_defbase – default database name [DV]       | Y  | -  |
|                                | get/set_defuser – default database user [DV]       | Y  | -  |
|                                | get/set_defpasswd – default database password [DV] | Y  | -  |

| PyGreSQL                           |                                                                    | 支持 | 备注 |
|------------------------------------|--------------------------------------------------------------------|----|----|
|                                    | escape_string – escape a string for use within SQL                 | Y  | -  |
|                                    | escape_bytea – escape binary data for use within SQL               | Y  | -  |
|                                    | unescape_bytea – unescape data that has been retrieved as text     | Y  | -  |
|                                    | get/set_namedresult – conversion to named tuples                   | Y  | -  |
|                                    | get/set_decimal – decimal type to be used for numeric values       | Y  | -  |
|                                    | get/set_decimal_point – decimal mark used for monetary values      | Y  | -  |
|                                    | get/set_bool – whether boolean values are returned as bool objects | Y  | -  |
|                                    | get/set_array – whether arrays are returned as list objects        | Y  | -  |
|                                    | get/set_bytea_escaped – whether bytea data is returned escaped     | Y  | -  |
|                                    | get/set_jsondecode – decoding JSON format                          | Y  | -  |
|                                    | get/set_cast_hook – fallback typecast function                     | Y  | -  |
|                                    | get/set_datestyle – assume a fixed date style                      | Y  | -  |
|                                    | get/set_typecast – custom typecasting                              | Y  | -  |
|                                    | cast_array/record – fast parsers for arrays and records            | Y  | -  |
|                                    | Type helpers                                                       | Y  | -  |
|                                    | Module constants                                                   | Y  | -  |
| Connection – The connection object | query – execute a SQL command string                               | Y  | -  |
|                                    | send_query – executes a SQL command string asynchronously          | Y  | -  |
|                                    | query_prepared – execute a prepared statement                      | Y  | -  |
|                                    | prepare – create a prepared statement                              | Y  | -  |
|                                    | describe_prepared – describe a prepared statement                  | Y  | -  |
|                                    | reset – reset the connection                                       | Y  | -  |
|                                    | poll – completes an asynchronous connection                        | Y  | -  |
|                                    | cancel – abandon processing of current SQL command                 | Y  | -  |

| PyGreSQL |                                                                  | 支持 | 备注                           |
|----------|------------------------------------------------------------------|----|------------------------------|
|          | close – close the database connection                            | Y  | -                            |
|          | transaction – get the current transaction state                  | Y  | -                            |
|          | parameter – get a current server parameter setting               | Y  | -                            |
|          | date_format – get the currently used date format                 | Y  | -                            |
|          | fileno – get the socket used to connect to the database          | Y  | -                            |
|          | set_non_blocking - set the non-blocking status of the connection | Y  | -                            |
|          | is_non_blocking - report the blocking status of the connection   | Y  | -                            |
|          | getnotify – get the last notify from the server                  | N  | 数据库不支持 listen / notify。      |
|          | inserttable – insert a list into a table                         | Y  | copy 命令中如果有 \n, 请使用双引号引用此字段。 |
|          | get/set_notice_receiver – custom notice receiver                 | Y  | -                            |
|          | putline – write a line to the server socket [DA]                 | Y  | -                            |
|          | getline – get a line from server socket [DA]                     | Y  | -                            |
|          | endcopy – synchronize client and server [DA]                     | Y  | -                            |
|          | locreate – create a large object in the database [LO]            | N  | 大对象相关操作。                     |
|          | getlo – build a large object from given oid [LO]                 | N  | 大对象相关操作。                     |

| PyGreSQL             |                                                                | 支持 | 备注       |
|----------------------|----------------------------------------------------------------|----|----------|
|                      | loimport – import a file to a large object [LO]                | N  | 大对象相关操作。 |
|                      | Object attributes                                              | Y  | -        |
| The DB wrapper class | Initialization                                                 | Y  | -        |
|                      | pkey – return the primary key of a table                       | Y  | -        |
|                      | get_databases – get list of databases in the system            | Y  | -        |
|                      | get_relations – get list of relations in connected database    | Y  | -        |
|                      | get_tables – get list of tables in connected database          | Y  | -        |
|                      | get_attnames – get the attribute names of a table              | Y  | -        |
|                      | has_table_privilege – check table privilege                    | Y  | -        |
|                      | get/set_parameter – get or set run-time parameters             | Y  | -        |
|                      | begin/commit/rollback/savepoint/release – transaction handling | Y  | -        |
|                      | get – get a row from a database table or view                  | Y  | -        |
|                      | insert – insert a row into a database table                    | Y  | -        |
|                      | update – update a row in a database table                      | Y  | -        |
|                      | upsert – insert a row with conflict resolution                 | Y  | -        |
|                      | query – execute a SQL command string                           | Y  | -        |
|                      | query_formatted – execute a formatted SQL command string       | Y  | -        |
|                      | query_prepared – execute a prepared statement                  | Y  | -        |
|                      | prepare – create a prepared statement                          | Y  | -        |
|                      | describe_prepared – describe a prepared statement              | Y  | -        |
|                      | delete_prepared – delete a prepared statement                  | Y  | -        |
|                      | clear – clear row values in memory                             | Y  | -        |

| PyGreSQL      |                                                                           | 支持 | 备注                      |
|---------------|---------------------------------------------------------------------------|----|-------------------------|
|               | delete – delete a row from a database table                               | Y  | 元组必须有唯一键或者主键。           |
|               | truncate – quickly empty database tables                                  | Y  | -                       |
|               | get_as_list/dict – read a table as a list or dictionary                   | Y  | -                       |
|               | escape_literal/identifier/string/bytea – escape for SQL                   | Y  | -                       |
|               | unescape_bytea – unescape data retrieved from the database                | Y  | -                       |
|               | encode/decode_json – encode and decode JSON data                          | Y  | -                       |
|               | use_regtypes – determine use of regular type names                        | Y  | -                       |
|               | notification_handler – create a notification handler                      | N  | 数据库不支持 listen / notify。 |
|               | Attributes of the DB wrapper class                                        | Y  | -                       |
| Query methods | getresult – get query values as list of tuples                            | Y  | -                       |
|               | dictresult/dictiter – get query values as dictionaries                    | Y  | -                       |
|               | namedresult/namediter – get query values as named tuples                  | Y  | -                       |
|               | scalarresult/scalariter – get query values as scalars                     | Y  | -                       |
|               | one/onedict/onenamed/onescalar – get one result of a query                | Y  | -                       |
|               | single/singledict/singlenamed/singlescalar – get single result of a query | Y  | -                       |
|               | listfields – list fields names of previous query result                   | Y  | -                       |
|               | fieldname, fieldnum – field name/number conversion                        | Y  | -                       |
|               | fieldinfo – detailed info about query result fields                       | Y  | -                       |

| PyGreSQL                    |                                                                   | 支持 | 备注                      |
|-----------------------------|-------------------------------------------------------------------|----|-------------------------|
|                             | ntuples – return number of tuples in query object                 | Y  | -                       |
|                             | memsize – return number of bytes allocated by query result        | Y  | -                       |
| LargeObject – Large Objects | open – open a large object                                        | N  | 大对象相关操作。                |
|                             | close – close a large object                                      | N  | 大对象相关操作。                |
|                             | read, write, tell, seek, unlink – file-like large object handling | N  | 大对象相关操作。                |
|                             | size – get the large object size                                  | N  | 大对象相关操作。                |
|                             | export – save a large object to a file                            | N  | 大对象相关操作。                |
|                             | Object attributes                                                 | N  | 大对象相关操作。                |
| The Notification Handler    | Instantiating the notification handler                            | N  | 数据库不支持 listen / notify。 |
|                             | Invoking the notification handler                                 | N  | 数据库不支持 listen / notify。 |

| PyGreSQL                           |                                                                | 支持 | 备注                      |
|------------------------------------|----------------------------------------------------------------|----|-------------------------|
|                                    | Sending notifications                                          | N  | 数据库不支持 listen / notify。 |
|                                    | Auxiliary methods                                              | N  | 数据库不支持 listen / notify。 |
| <b>pgdb</b>                        |                                                                |    |                         |
| Module functions and constants     | connect – Open a PostgreSQL connection                         | Y  | -                       |
|                                    | get/set/reset_typecast – Control the global typecast functions | Y  | -                       |
|                                    | Module constants                                               | Y  | -                       |
|                                    | Errors raised by this module                                   | Y  | -                       |
| Connection – The connection object | close – close the connection                                   | Y  | -                       |
|                                    | commit – commit the connection                                 | Y  | -                       |
|                                    | rollback – roll back the connection                            | Y  | -                       |
|                                    | cursor – return a new cursor object                            | Y  | -                       |
|                                    | Attributes that are not part of the standard                   | Y  | -                       |
| Cursor – The cursor object         | description – details regarding the result columns             | Y  | -                       |
|                                    | rowcount – number of rows of the result                        | Y  | -                       |
|                                    | close – close the cursor                                       | Y  | -                       |
|                                    | execute – execute a database operation                         | Y  | -                       |
|                                    | executemany – execute many similar database operations         | Y  | -                       |
|                                    | callproc – Call a stored procedure                             | Y  | -                       |
|                                    | fetchone – fetch next row of the query result                  | Y  | -                       |
|                                    | fetchmany – fetch next set of rows of the query result         | Y  | -                       |
|                                    | fetchall – fetch all rows of the query result                  | Y  | -                       |

| PyGreSQL                             |                                                          | 支持 | 备注 |
|--------------------------------------|----------------------------------------------------------|----|----|
|                                      | arraysize - the number of rows to fetch at a time        | Y  | -  |
|                                      | Methods and attributes that are not part of the standard | Y  | -  |
| Type – Type objects and constructors | Type constructors                                        | Y  | -  |
|                                      | Type objects                                             | Y  | -  |

## 在 Linux 环境使用 PyGreSQL 第三方库连接集群

**步骤1** 以root用户登录Linux环境。

**步骤2** 执行以下命令创建python\_dws.py文件。

```
vi python_dws.py
```

请复制粘贴以下内容放入python\_dws.py文件中：

```
#!/usr/bin/env python3
* encoding:utf-8 _*_

from __future__ import print_function

import pg

def create_table(connection):
 print("Begin to create table")
 try:
 connection.query("drop table if exists test;"
 "create table test(id int, name text);")
 except pg.InternalError as e:
 print(e)
 else:
 print("Table created successfully")

def insert_data(connection):
 print("Begin to insert data")
 try:
 connection.query("insert into test values(1,'number1');")
 connection.query("insert into test values(2,'number2');")
 connection.query("insert into test values(3,'number3');")
 except pg.InternalError as e:
 print(e)
 else:
 print("Insert data successfully")

def update_data(connection):
 print("Begin to update data")
 try:
 result = connection.query("update test set name = 'numberupdated' where id=1;")
 print("Total number of rows updated :", result)
 result = connection.query("select * from test order by 1;")
 rows = result.getresult()
 for row in rows:
```

```

 print("id = ", row[0])
 print("name = ", row[1], "\n")
 except pg.InternalError as e:
 print(e)
 else:
 print("After Update, Operation done successfully")

def delete_data(connection):
 print("Begin to delete data")
 try:
 result = connection.query("delete from test where id=3;")
 print("Total number of rows deleted :", result)
 result = connection.query("select * from test order by 1;")
 rows = result.getresult()
 for row in rows:
 print("id = ", row[0])
 print("name = ", row[1], "\n")
 except pg.InternalError as e:
 print(e)
 else:
 print("After Delete, Operation done successfully")

def select_data(connection):
 print("Begin to select data")
 try:
 result = connection.query("select * from test order by 1;")
 rows = result.getresult()
 for row in rows:
 print("id = ", row[0])
 print("name = ", row[1])
 except pg.InternalError as e:
 print(e)
 print("select failed")
 else:
 print("Operation done successfully")

if __name__ == '__main__':
 try:
 conn = pg.DB(host='10.154.70.231',
 port=8000,
 dbname='gaussdb', # 需要连接的database
 user='dbadmin',
 passwd='password') # 数据库用户密码
 except pg.InternalError as ex:
 print(ex)
 print("Connect database failed")
 else:
 print("Opened database successfully")
 create_table(conn)
 insert_data(conn)
 select_data(conn)
 update_data(conn)
 delete_data(conn)
 conn.close()

```

或使用dbapi接口实现：

```

#!/usr/bin/python
-*- coding: UTF-8 -*-

from __future__ import print_function

import pg
import pgdb

def create_table(connection):

```

```
print("Begin to create table")
try:
 cursor = connection.cursor()
 cursor.execute("drop table if exists test;"
 "create table test(id int, name text);")
 connection.commit()
except pg.InternalError as e:
 print(e)
else:
 print("Table created successfully")
 cursor.close()

def insert_data(connection):
 print("Begin to insert data")
 try:
 cursor = connection.cursor()
 cursor.execute("insert into test values(1,'number1');")
 cursor.execute("insert into test values(2,'number2');")
 cursor.execute("insert into test values(3,'number3');")
 connection.commit()
 except pg.InternalError as e:
 print(e)
 else:
 print("Insert data successfully")
 cursor.close()

def update_data(connection):
 print("Begin to update data")
 try:
 cursor = connection.cursor()
 cursor.execute("update test set name = 'numberupdated' where id=1;")
 connection.commit()
 print("Total number of rows updated :", cursor.rowcount)
 cursor.execute("select * from test;")
 rows = cursor.fetchall()
 for row in rows:
 print("id = ", row[0])
 print("name = ", row[1], "\n")
 except pg.InternalError as e:
 print(e)
 else:
 print("After Update, Operation done successfully")

def delete_data(connection):
 print("Begin to delete data")
 try:
 cursor = connection.cursor()
 cursor.execute("delete from test where id=3;")
 connection.commit()
 print("Total number of rows deleted :", cursor.rowcount)
 cursor.execute("select * from test;")
 rows = cursor.fetchall()
 for row in rows:
 print("id = ", row[0])
 print("name = ", row[1], "\n")
 except pg.InternalError as e:
 print(e)
 else:
 print("After Delete, Operation done successfully")

def select_data(connection):
 print("Begin to select data")
 try:
 cursor = connection.cursor()
 cursor.execute("select * from test;")
```

```
rows = cursor.fetchall()
for row in rows:
 print("id = ", row[0])
 print("name = ", row[1], "\n")
except pg.InternalError as e:
 print(e)
 print("select failed")
else:
 print("Operation done successfully")
 cursor.close()

if __name__ == '__main__':
 try:
 conn = pgdb.connect(host='10.154.70.231',
 port='8000',
 database='gaussdb', # 需要连接的database
 user='dbadmin',
 password='password') # 数据库用户密码
 except pg.InternalError as ex:
 print(ex)
 print("Connect database failed")
 else:
 print("Opened database successfully")
 create_table(conn)
 insert_data(conn)
 select_data(conn)
 update_data(conn)
 delete_data(conn)
 conn.close()
```

**步骤3** 按照实际集群信息，修改python\_dws.py文件中的集群公网访问地址、集群端口号、数据库名称、数据库用户名、数据库密码。

#### 说明

PyGreSQL接口不提供重试连接的能力，您需要在业务代码中实现重试处理。

```
conn = pgdb.connect(host='10.154.70.231',
 port='8000',
 database='gaussdb', # 需要连接的database
 user='dbadmin',
 password='password') # 数据库用户密码
```

**步骤4** 执行以下命令，使用PyGreSQL第三方库连接集群。

```
python python_dws.py
```

----结束

## 在 Windows 环境使用 PyGreSQL 第三方库连接集群

**步骤1** 在Windows系统中，单击“开始”按钮，在搜索框中，键入cmd，然后在结果列表中单击“cmd.exe”打开命令提示符窗口。

**步骤2** 在命令提示符窗口中，执行以下命令创建python\_dws.py文件。

```
type nul> python_dws.py
```

请复制粘贴以下内容放入python\_dws.py文件中：

```
#!/usr/bin/env python3
*_ encoding:utf-8 *_

from __future__ import print_function

import pg
```

```
def create_table(connection):
 print("Begin to create table")
 try:
 connection.query("drop table if exists test;"
 "create table test(id int, name text);")
 except pg.InternalError as e:
 print(e)
 else:
 print("Table created successfully")

def insert_data(connection):
 print("Begin to insert data")
 try:
 connection.query("insert into test values(1,'number1');")
 connection.query("insert into test values(2,'number2');")
 connection.query("insert into test values(3,'number3');")
 except pg.InternalError as e:
 print(e)
 else:
 print("Insert data successfully")

def update_data(connection):
 print("Begin to update data")
 try:
 result = connection.query("update test set name = 'numberupdated' where id=1;")
 print("Total number of rows updated :", result)
 result = connection.query("select * from test order by 1;")
 rows = result.getresult()
 for row in rows:
 print("id = ", row[0])
 print("name = ", row[1], "\n")
 except pg.InternalError as e:
 print(e)
 else:
 print("After Update, Operation done successfully")

def delete_data(connection):
 print("Begin to delete data")
 try:
 result = connection.query("delete from test where id=3;")
 print("Total number of rows deleted :", result)
 result = connection.query("select * from test order by 1;")
 rows = result.getresult()
 for row in rows:
 print("id = ", row[0])
 print("name = ", row[1], "\n")
 except pg.InternalError as e:
 print(e)
 else:
 print("After Delete, Operation done successfully")

def select_data(connection):
 print("Begin to select data")
 try:
 result = connection.query("select * from test order by 1;")
 rows = result.getresult()
 for row in rows:
 print("id = ", row[0])
 print("name = ", row[1])
 except pg.InternalError as e:
 print(e)
 print("select failed")
 else:
 print("Operation done successfully")
```

```
if __name__ == '__main__':
 try:
 conn = pg.DB(host='10.154.70.231',
 port=8000,
 dbname='gaussdb', # 需要连接的database
 user='dbadmin',
 passwd='password') # 数据库用户密码
 except pg.InternalError as ex:
 print(ex)
 print("Connect database failed")
 else:
 print("Opened database successfully")
 create_table(conn)
 insert_data(conn)
 select_data(conn)
 update_data(conn)
 delete_data(conn)
 conn.close()
```

或使用dbapi接口实现：

```
#!/usr/bin/python
-*- coding: UTF-8 -*-

from __future__ import print_function

import pg
import pgdb

def create_table(connection):
 print("Begin to create table")
 try:
 cursor = connection.cursor()
 cursor.execute("drop table if exists test;"
 "create table test(id int, name text);")
 connection.commit()
 except pg.InternalError as e:
 print(e)
 else:
 print("Table created successfully")
 cursor.close()

def insert_data(connection):
 print("Begin to insert data")
 try:
 cursor = connection.cursor()
 cursor.execute("insert into test values(1,'number1');")
 cursor.execute("insert into test values(2,'number2');")
 cursor.execute("insert into test values(3,'number3');")
 connection.commit()
 except pg.InternalError as e:
 print(e)
 else:
 print("Insert data successfully")
 cursor.close()

def update_data(connection):
 print("Begin to update data")
 try:
 cursor = connection.cursor()
 cursor.execute("update test set name = 'numberupdated' where id=1;")
 connection.commit()
 print("Total number of rows updated :", cursor.rowcount)
 cursor.execute("select * from test;")
 rows = cursor.fetchall()
 for row in rows:
```

```
 print("id = ", row[0])
 print("name = ", row[1], "\n")
 except pg.InternalError as e:
 print(e)
 else:
 print("After Update, Operation done successfully")

def delete_data(connection):
 print("Begin to delete data")
 try:
 cursor = connection.cursor()
 cursor.execute("delete from test where id=3;")
 connection.commit()
 print("Total number of rows deleted :", cursor.rowcount)
 cursor.execute("select * from test;")
 rows = cursor.fetchall()
 for row in rows:
 print("id = ", row[0])
 print("name = ", row[1], "\n")
 except pg.InternalError as e:
 print(e)
 else:
 print("After Delete, Operation done successfully")

def select_data(connection):
 print("Begin to select data")
 try:
 cursor = connection.cursor()
 cursor.execute("select * from test;")
 rows = cursor.fetchall()
 for row in rows:
 print("id = ", row[0])
 print("name = ", row[1], "\n")
 except pg.InternalError as e:
 print(e)
 print("select failed")
 else:
 print("Operation done successfully")
 cursor.close()

if __name__ == '__main__':
 try:
 conn = pgdb.connect(host='10.154.70.231',
 port='8000',
 database='gaussdb', # 需要连接的database
 user='dbadmin',
 password='password') # 数据库用户密码
 except pg.InternalError as ex:
 print(ex)
 print("Connect database failed")
 else:
 print("Opened database successfully")
 create_table(conn)
 insert_data(conn)
 select_data(conn)
 update_data(conn)
 delete_data(conn)
 conn.close()
```

**步骤3** 按照实际集群信息，修改python\_dws.py文件中的集群公网访问地址、集群端口号、数据库名称、数据库用户名、数据库密码。

PyGreSQL接口不提供重试连接的能力，您需要在业务代码中实现重试处理。

```
conn = pgdb.connect(host='10.154.70.231',
 port='8000',
 database='gaussdb', # 需要连接的database
```

```
user='dbadmin',
password='password') # 数据库用户密码
```

**步骤4** 执行以下命令，使用PyGreSQL第三方库连接集群。

```
python python_dws.py
```

----结束

# 7 创建 DWS 数据库和用户

DWS默认的数据库gaussdb一般不作为客户的业务数据库使用，且考虑业务隔离，一般会规划多个数据库。使用系统管理员dbadmin首次连接gaussdb数据库后，需根据业务实际需求规划相应的业务数据库、用户和角色，包括新建业务场景和已有上游业务数据需迁移到DWS的场景。

角色是一组权限的集合，用户和角色的关系，请参见《开发指南》中“权限管理”章节。在创建用户前，可以先创建角色，即先预置一些常用角色，例如拥有创建数据库的角色，再创建用户，在创建用户的过程将预先创建好的角色授权予该用户。

用户、角色和相应的权限列表支持导出，详情参见[导出用户](#)、[导出用户权限](#)、[导出角色](#)、[导出角色权限](#)。

## 约束和限制

- **尽量避免**所有业务使用同一个数据库用户运行，请按业务模块规划不同数据库用户。
- **不建议**使用系统管理员用户跑业务，不同模块业务请通过多用户和权限进行访问控制。
- 更多开发设计规范参见《数据仓库服务开发指南》的“DWS总体开发设计规范”章节。

## 新建数据库

支持通过DDL语法和通过SQL编辑器创建。

- DDL语法方式创建：创建语法参见语法章节“CREATE DATABASE”。

## 创建角色

- 步骤1** 登录DWS管理控制台，在左侧导航栏中，选择“专属集群 > 集群列表”。
- 步骤2** 在集群列表中，单击指定集群的名称，进入“集群详情”页面。
- 步骤3** 在左侧导航栏中，单击“用户管理”，进入用户管理页面。
- 步骤4** 切换至“角色列表”页签，单击“创建角色”按钮，跳转至创建角色详情页面。
- 步骤5** 完善常规配置角色信息，参数描述如下所示：

表 7-1 角色信息常规配置参数信息

| 参数名称  | 描述                                        | 样例值      |
|-------|-------------------------------------------|----------|
| 角色名称  | 以字母开头，可以包含字母、数字、下划线，长度不超过63个字符。           | dws-demo |
| 到期时间  | 设置该角色权限的到期时间。                             | -        |
| 系统管理员 | 表示该角色是否拥有系统管理员权限。                         | -        |
| 创建数据库 | 表示该角色是否有创建数据库权限。                          | -        |
| 创建角色  | 表示该角色是否有创建新用户/角色权限。                       | -        |
| 继承权限  | 表示该角色是否“继承”它所在组的角色的权限。 <b>默认打开，不推荐修改。</b> | -        |

**步骤6** 确认无误后，单击“下一步”。

**步骤7** 权限配置，配置该角色拥有的权限。

单击“添加”，添加一条权限配置，选择数据库对象类型以及对应的数据库对象。然后选择权限进行授权。具体权限释义，请参见《数据仓库服务SQL语法参考》中的“DCL语法 > GRANT”章节。

**步骤8** 授权完毕，单击“立即创建”。完成角色创建。

----结束

## 创建数据库用户

支持通过DDL语法和通过DWS管理控制台创建。DDL语法请参见语法章节CREATE USER。

### 说明

- 如果当前控制台界面不支持该功能，请联系技术支持人员。
- 集群创建完成即存在的用户/角色不允许修改。
- 使用该功能需确认集群状态为“可用”。

**步骤1** 登录DWS管理控制台，在左侧导航栏中，选择“专属集群 > 集群列表”。

**步骤2** 在集群列表中，单击指定集群的名称，进入“集群详情”页面。

**步骤3** 在左侧导航栏中，单击“用户管理”，进入用户管理页面。

**步骤4** 在“用户列表”页签，单击“创建用户”，跳转至创建用户详情页面。

**步骤5** 进入“常规配置”页面，参数说明如下所示：

表 7-2 用户信息常规配置参数说明

| 参数名称  | 描述                                                                                                                   | 样例值      |
|-------|----------------------------------------------------------------------------------------------------------------------|----------|
| 用户名称  | 以字母开头，可以包含字母、数字、下划线，长度不超过63个字符。                                                                                      | Dws-demo |
| 密码    | 可以包含字母、数字、下划线及特殊字符，长度12~32个字符。<br><b>说明</b><br>密码至少包含大写字母、小写字母、数字和特殊字符四类中的三类，其中可输入的特殊字符为：(~!?,;:_){}/<>@#%^&*+ \=-)。 | -        |
| 连接数限制 | 该用户连接数据库的连接数限制，-1表示不限制。                                                                                              | -1       |
| 到期时间  | 设置该用户权限的到期时间。                                                                                                        | -        |
| 系统管理员 | 表示该用户是否拥有系统管理员权限。                                                                                                    | -        |
| 创建数据库 | 表示该用户是否拥有创建数据库权限。                                                                                                    | -        |
| 创建角色  | 表示该用户是否拥有创建新用户、角色权限。                                                                                                 | -        |
| 继承权限  | 表示该用户是否“继承”所在组的角色权限。 <b>默认打开，不推荐修改。</b>                                                                              | -        |

- 步骤6 确认无误后，单击“下一步”。
- 步骤7 进入“角色配置”页面，选择需要对该用户授权的角色，确定后单击“下一步”。
- 步骤8 权限配置，对该用户授权角色后，若还需要补充其他权限，可以单独配置。  
  
单击“添加”按钮，添加一条权限配置，选择数据库对象类型以及对应的数据库对象，然后选择权限进行授权。具体权限释义，请参见《数据仓库服务SQL语法参考》中的“DCL语法 > GRANT”章节。
- 步骤9 授权完毕，单击“立即创建”完成用户创建。
- 结束

修改用户

- 步骤1 登录DWS管理控制台，在左侧导航栏中，选择“专属集群 > 集群列表”。
- 步骤2 在集群列表中，单击指定集群的名称，进入“集群详情”页面。
- 步骤3 在左侧导航栏中，单击“用户管理”，进入用户管理页面。
- 步骤4 在用户列表中选择其中一个用户，单击“修改”按钮，跳转至修改用户详情页面。
- 步骤5 修改用户信息，具体信息请参见表7-2，确认无误后单击“下一步”。
- 步骤6 角色配置，选择需要对该用户授权的角色，单击“下一步”。

**步骤7** 权限配置，选择权限类型后，可单击所在行操作列的“**编辑**”按钮，再在权限列单击“**修改**”按钮对权限进行添加或删除。

**步骤8** 授权完毕，单击“保存”完成修改。

----结束

## 删除用户

### 前提条件

先确认是否存在未删除的表等数据库对象依赖关系，如果有的话请先删除关系，否则用户将删除失败。

### 操作步骤

**步骤1** 登录DWS管理控制台，在左侧导航栏中，选择“专属集群 > 集群列表”。

**步骤2** 在集群列表中，单击指定集群的名称，进入“集群详情”页面。

**步骤3** 在左侧导航栏中，单击“用户管理”，进入用户管理页面。

**步骤4** 在用户列表选择其中一个用户，单击“**删除**”按钮，弹出确认页面。如果勾选“强制删除，自动解除依赖关系”则当前用户名下的表及其他数据库对象，将转移至管理员账号名下。

**步骤5** 确认无误后，单击“确定”删除用户。

----结束

## 导出用户

**步骤1** 登录DWS管理控制台，在左侧导航栏中，选择“专属集群 > 集群列表”。

**步骤2** 在集群列表中，单击指定集群的名称，进入“集群详情”页面。

**步骤3** 在左侧导航栏中，单击“**用户管理**”，进入用户管理页面。

**步骤4** 在用户列表上方单击“**导出**”按钮，选择导出条数，即可导出用户列表。

**步骤5** 确认无误后，单击“**导出**”即可。

----结束

## 导出用户权限

**步骤1** 登录DWS管理控制台，在左侧导航栏中，选择“专属集群 > 集群列表”。

**步骤2** 在集群列表中，单击指定集群的名称，进入“集群详情”页面。

**步骤3** 在左侧导航栏中，单击“用户管理”，进入用户管理页面。

**步骤4** 在用户列表选择指定用户，单击“**导出权限**”，即可导出用户拥有权限列表。

----结束

## 修改角色

- 步骤1** 登录DWS管理控制台，在左侧导航栏中，选择“专属集群 > 集群列表”。
- 步骤2** 在集群列表中，单击指定集群的名称，进入“集群详情”页面。
- 步骤3** 在左侧导航栏中，单击“用户管理”，进入用户管理页面。
- 步骤4** 在角色列表中选择其中一个角色，单击“**修改**”按钮，跳转至修改角色详情页面。
- 步骤5** 修改角色信息，参数描述请参见[表7-1](#)。
- 步骤6** 确认无误后，单击“下一步”。
- 步骤7** 权限配置，根据需要选择权限类型，单击所在行操作列的“**编辑**”按钮，再在权限列单击“**修改**”按钮对权限进行添加或移除。
- 步骤8** 授权完毕，单击“保存”完成修改。

----结束

## 删除角色

### 前提条件

先确认是否存在未删除的数据库对象等依赖关系，如果有的话请先删除关系，否则角色将删除失败。

### 操作步骤

- 步骤1** 登录DWS管理控制台，在左侧导航栏中，选择“专属集群 > 集群列表”。
- 步骤2** 在集群列表中，单击指定集群的名称，进入“集群详情”页面。
- 步骤3** 在左侧导航栏中，单击“用户管理”，进入用户管理页面。
- 步骤4** 在角色列表中选择其中一个角色，单击“**删除**”按钮，弹出确认页面。
- 步骤5** 单击“确定”删除角色。

----结束

## 导出角色

- 步骤1** 登录DWS管理控制台，在左侧导航栏中，选择“专属集群 > 集群列表”。
- 步骤2** 在集群列表中，单击指定集群的名称，进入“集群详情”页面。
- 步骤3** 在左侧导航栏中单击“用户管理”，切换至“**角色列表**”页面。
- 步骤4** 在角色列表上方单击“**导出**”按钮，选择导出条数，即可导出角色列表。
- 步骤5** 确认无误后，单击“**导出**”按钮。

----结束

## 导出角色权限

- 步骤1** 登录DWS管理控制台，在左侧导航栏中，选择“专属集群 > 集群列表”。

**步骤2** 在集群列表中，单击指定集群的名称，进入“集群详情”页面。

**步骤3** 在左侧导航栏中单击“用户管理”，切换至“角色列表”页面。

**步骤4** 在角色列表选择指定用户，单击“导出权限”按钮，选择导出条数，即可导出角色权限。

----结束

# 8 迁移业务数据至 DWS 集群

## 8.1 管理数据源

### 8.1.1 管理 MRS 数据源

#### 8.1.1.1 MRS 数据源使用概述

##### MRS 集群简介

MapReduce服务（MapReduce Service，简称MRS）是一个基于开源Hadoop生态环境而运行的大数据集群，对外提供大容量数据的存储和分析能力，可解决用户的数据存储和处理需求。有关MRS服务的详细信息，请参考《MapReduce服务用户指南》。

用户可以将海量业务数据，存储在MRS的分析集群，即使用Hive/Spark组件保存。Hive/Spark的数据文件则保存在HDFS中。DWS 支持在相同网络中，配置一个DWS 集群连接到MRS集群，然后将数据从HDFS中的文件读取到DWS 。

##### 使用流程

从MRS导入数据到集群流程如下：

1. 前提条件
  - a. 创建一个MRS集群，具体操作步骤请参见《MapReduce服务用户指南》中“购买自定义集群”章节。
  - b. 创建一个HDFS外表，外表通过外部服务器的接口，从MRS集群查询数据。具体操作步骤请参见《数据仓库服务数据库开发指南》中“导入数据 > 从MRS导入数据到集群”章节。

##### 说明

- 同一个网络下可以有多个MRS数据源，但是DWS集群每次只能和一个MRS集群建立连接。
2. 在DWS 集群创建一个MRS数据源连接，具体操作步骤请参见[创建MRS数据源连接](#)。

3. 使用MRS数据源导入数据到集群，具体操作步骤请参见《数据仓库服务数据库开发指南》中的“导入数据 > 从MRS导入数据到集群”章节。
4. （可选）当MRS集群的HDFS配置发生变更时，在DWS服务中，需要执行MRS数据源配置的更新操作，详情请参见[更新MRS数据源配置](#)。

### 8.1.1.2 创建 MRS 数据源连接

#### 操作场景

DWS从MRS的HDFS读取数据前，需要先创建一个MRS数据源连接，作为DWS集群与MRS集群的数据通道。

#### 对系统的影响

- 一个DWS集群在创建MRS数据源连接时，不能同时创建两个连接。
- 创建MRS数据源连接时，系统默认自动为DWS集群和MRS集群的安全组增加出规则和入规则，允许相同子网中节点的访问。
- 启用Kerberos认证的MRS集群，系统会自动增加一个类型为“机机”的用户，属于“supergroup”用户组。

#### 前提条件

- DWS集群已创建好，并记录集群所在的虚拟私有云和子网。
- 创建MRS数据源连接需要创建MRS集群类型为分析集群。

#### 操作步骤

**步骤1** 登录华为云管理控制台。

**步骤2** 打开MRS管理控制台，创建MRS集群。

创建集群时，请按要求配置以下参数，其他配置无特别要求，具体操作请参见《MapReduce服务用户指南》中的“自定义创建集群”章节：

- MRS集群的虚拟私有云需要和DWS集群相同。
- MRS集群版本，支持版本如下所示：
  - 8.1.1.300及以上版本集群，MRS集群支持连接1.6.\*、1.7.\*、1.8.\*、1.9.\*、2.0.\*、3.0.\*、3.1.\*、3.2.\*、3.3.\*及以上版本（“\*”代表的是数字）。
  - 8.1.1.300以下版本集群，MRS集群支持连接1.6.\*、1.7.\*、1.8.\*、1.9.\*、2.0.\*版本（“\*”代表的是数字）。
- 组件选择，需要选择Hadoop组件。

如果已有符合如上条件的MRS集群，则可跳过此步骤。

**步骤3** 打开DWS管理控制台，进入DWS管理控制台页面。

**步骤4** 在DWS管理控制台，选择“专属集群 > 集群列表”。

**步骤5** 在集群列表中单击指定集群的名称，进入“集群详情”页面。

**步骤6** 在左侧导航栏，选择“数据源>MRS数据源”页签。

**步骤7** 单击“创建MRS数据源连接”，填写配置参数。

表 8-1 MRS 连接公共参数说明

| 参数名   | 说明                                                                                                                                                                                                                                                                                                                                |
|-------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 数据源名称 | 对应DWS数据库server名称，包含小写字母、数字或者下划线，且必须以小写字母开头，长度为3到63个字符。                                                                                                                                                                                                                                                                            |
| 配置方式  | <p>表示配置时系统获取文件使用的方式：</p> <ul style="list-style-type: none"><li>• MRS用户：配置MRS manager用户/密码，系统登录MRS manager自动下载配置文件及认证文件。参数说明详情请参见表8-2。</li><li>• 文件上传：自行在MRS manager下载配置文件，通过上传配置文件的方式配置，该方式适用于开启Kerberos认证。参数说明详情请参见表8-3。</li></ul> <p><b>说明</b></p> <ul style="list-style-type: none"><li>- 选择文件上传方式时，确保MRS和DWS集群网络互通。</li></ul> |
| 数据库   | 数据源所在数据库。                                                                                                                                                                                                                                                                                                                         |
| 描述    | 表示此连接的说明信息。                                                                                                                                                                                                                                                                                                                       |

表 8-2 MRS 用户方式参数配置说明

| 参数名    | 说明                                                                                                                                                                                                           |
|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| MRS数据源 | <p>下拉框选择DWS可连接的MRS集群。默认显示当前用户可连接的，即与当前DWS集群在相同虚拟私有云和子网下且为可用状态的自定义型、混合型以及分析型MRS集群。</p> <p>选择一个MRS集群后，将自动显示已选择的MRS是否启用了Kerberos认证。单击“查看MRS集群”可进入MRS查看该MRS集群信息。</p> <p>如果“MRS数据源”下拉框为空，用户可以单击“创建MRS集群”进行创建。</p> |
| MRS用户  | DWS集群连接MRS集群时使用的用户名。                                                                                                                                                                                         |
| 用户密码   | <p>填写连接用户的密码。如果用户密码被修改，则需要重新创建连接。</p> <p><b>须知</b></p> <p>用户密码必须成功登录过MRS Manager，新用户使用初始密码第一次登录MRS Manager时会提示修改密码，这种情况会导致配置MRS数据源失败。</p>                                                                    |
| 使用机账号  | 开启后，会自动在MRS创建一个名称为dws的机账号用于后续dws数据库和MRS的交互，该机账号固定为supergroup组，拥有所有权限；若关闭，则直接将配置的人机用户用于dws数据库和MRS交互，需要保证用户拥有数据权限，否则在使用数据源时会提示文件不存在。                                                                           |

表 8-3 文件上传方式参数配置说明

| 参数名     | 说明                                                                                                                                                                                                                                      |
|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 认证凭据    | 用户keytab文件。在MRS manager下载的用户认证凭据文件，文件名称格式为：用户名称_时间戳_keytab.tar： <ul style="list-style-type: none"><li>● <b>MRS 2.x及之前版本下载路径：</b>系统设置&gt;用户管理&gt;更多&gt;下载认证凭据</li><li>● <b>MRS 3.x及之后版本下载路径：</b>系统&gt;权限&gt;用户&gt;更多&gt;下载认证凭据</li></ul> |
| 客户端配置文件 | 包含HDFS、Hive、hosts的客户端配置文件，下载客户端时“客户端类型”需选择“仅配置文件”： <ul style="list-style-type: none"><li>● <b>MRS 2.x及之前版本下载路径：</b>服务管理&gt;下载客户端</li><li>● <b>MRS 3.x及之后版本下载路径：</b>主页&gt;更多&gt;下载客户端</li></ul>                                          |

**步骤8** 单击“提交”保存连接。

创建连接需要一段时间，此时“配置状态”显示为“创建中”，成功后在MRS数据源列表中可看到已创建的连接，且状态为“可用”。

#### 说明

- 在“操作”列，可以单击“更新配置”，更新当前连接的“MRS集群状态”和“配置状态”。在更新配置时，无法创建新的连接，且会检查安全组规则是否正常并自助修复。具体请参见[更新MRS数据源配置](#)。
- 在“操作”列，可以单击“删除”将不再使用的连接删除释放。删除连接时，不会自动删除安全组规则，请根据需要手工删除。
- 安全组规则若不删除，DWS集群中的节点与MRS集群中的节点网络仍是互通的。如果用户对网络安全要求较严格，建议手动删除安全组规则。

----结束

### 8.1.1.3 更新 MRS 数据源配置

#### 操作场景

MRS的HDFS集群参数配置变更时，可能造成DWS集群无法从HDFS集群导入数据。使用HDFS集群导入数据前，需要执行MRS数据源配置的更新操作。

#### 前提条件

DWS集群已创建MRS数据源连接。

#### 对系统的影响

更新MRS数据源连接时，DWS集群会自动重启并无法提供服务。

#### 操作步骤

**步骤1** 在DWS管理控制台，选择“专属集群 > 集群列表”。

**步骤2** 在集群列表，单击指定集群的名称，然后单击“MRS数据源”。

**步骤3** 在MRS数据源列表中，选中需要更新的MRS数据源，在“操作”列中，单击“更新配置”。

更新当前连接的“MRS集群状态”和“配置状态”。在更新配置时，无法创建新的连接，且会检查安全组规则是否正常并自助修复。参数说明如下所示：

表 8-4 参数说明

| 参数名                                                       | 参数解释                                                                                                                                                                                                                                                                                            |
|-----------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| dfs.client.read.shortcircuit                              | 是否开启本地读。                                                                                                                                                                                                                                                                                        |
| dfs.client.read.shortcircuit.skip.checksum                | 本地读时是否跳过数据校验。                                                                                                                                                                                                                                                                                   |
| dfs.client.block.write.replace-datanode-on-failure.enable | 向HDFS写数据块发生失败时，是否替换新的节点作为副本存储位置。                                                                                                                                                                                                                                                                |
| dfs.encrypt.data.transfer                                 | 是否开启数据加密。设置为“true”表示加密，默认不加密。<br><b>说明</b> <ul style="list-style-type: none"><li>此参数仅对启用Kerberos认证的集群有效。</li><li>仅当hadoop.rpc.protection设置为privacy时使用。</li></ul>                                                                                                                                |
| dfs.encrypt.data.transfer.algorithm                       | 指定密钥传输的加密解密算法。<br>只有在dfs.encrypt.data.transfer配置项设置为“true”，此参数才会生效。<br>默认值为“3des”，表示采用3DES算法进行加密。                                                                                                                                                                                               |
| dfs.encrypt.data.transfer.cipher.suites                   | 指定实际存储数据传输的加密解密算法。<br>如果不指定此参数，则使用“dfs.encrypt.data.transfer.algorithm”参数指定的加密算法进行数据加密。默认值为“AES/CTR/NoPadding”。                                                                                                                                                                                 |
| dfs.replication                                           | 默认数据副本个数。                                                                                                                                                                                                                                                                                       |
| dfs.blocksize                                             | 默认数据块大小。                                                                                                                                                                                                                                                                                        |
| hadoop.security.authentication                            | 安全认证模式。                                                                                                                                                                                                                                                                                         |
| hadoop.rpc.protection                                     | RPC通信保护模式。<br>默认值： <ul style="list-style-type: none"><li>安全模式（启用Kerberos认证）：privacy</li><li>普通模式（未启用Kerberos认证）：authentication</li></ul> <b>说明</b> <ul style="list-style-type: none"><li>“authentication”：只进行认证，不加密。</li><li>“integrity”：进行认证和一致性校验。</li><li>“privacy”：进行认证、一致性校验、加密。</li></ul> |

| 参数名                    | 参数解释                  |
|------------------------|-----------------------|
| dfs.domain.socket.path | 本地使用的Domain socket路径。 |

----结束

## 8.1.2 管理 OBS 数据源

DWS支持以委托方式访问客户OBS上的数据，客户通过创建DWS云服务委托并授予OBS OperateAccess或OBS Administrator权限，然后在创建OBS数据源绑定该委托，便可通过OBS外表的方式访问OBS上的数据。

### 说明

- 该特性仅8.2.0及以上集群版本支持。
- 同一个集群的OBS数据源在创建、修改、删除时是互斥的，不支持同时执行多个操作。

## 创建 OBS 委托

### 操作场景

创建OBS数据源前需要用户提前创建好授权给DWS具有OBS OperateAccess或OBS Administrator权限的委托。

### 操作步骤

- 步骤1** 鼠标移动至页面右上角账号，单击“统一身份认证”，进入统一身份认证服务页面。
- 步骤2** 在左侧导航栏单击“委托”，在委托页面右上角单击“创建委托”。
- 步骤3** 创建委托时委托类型选择“云服务”，云服务选择“DWS”。
- 步骤4** 单击“下一步”，对委托授予OBS服务的“OBS OperateAccess”或“OBS Administrator”权限。
- 步骤5** 单击“下一步”，选择授权资源范围为“所有资源”或需要访问的资源，然后确认无误后提交。

----结束

## 创建 OBS 数据源

### 前提条件

已创建好授权给DWS具有OBS OperateAccess权限的委托。

### 操作步骤

- 步骤1** 在DWS管理控制台，选择“专属集群 > 集群列表”。
- 步骤2** 在集群列表，单击指定集群的名称，选择“数据源>OBS数据源”。
- 步骤3** 在OBS数据源页面，单击“创建OBS数据源连接”，填写配置参数。

表 8-5 OBS 数据源连接参数说明

| 参数名称  | 说明                                                         |
|-------|------------------------------------------------------------|
| 数据源名称 | 要创建的OBS数据源连接名称，由用户自定义。<br>该数据源名称作为后续创建OBS外表语句中指定的server名称。 |
| OBS委托 | 当前用户授权给DWS服务的具有OBS OperateAccess权限的委托。                     |
| 数据库   | 要创建的OBS数据源连接所在的数据库。                                        |
| 描述    | 要创建的OBS数据源连接的描述信息。                                         |

**步骤4** 确认无误后，单击“确认”按钮，提交创建操作，创建过程大约需要10秒钟左右。

----结束

## 更新 OBS 数据源配置

### 操作场景

当OBS数据源连接创建之后，DWS会定期更新数据源使用的临时委托授权信息，如果24小时内自动更新均失败的话，数据源连接将不可用，此时可以手动在控制台临时进行一次手动更新操作。

### 操作步骤

**步骤1** 在DWS管理管制台，选择“专属集群 > 集群列表”。

**步骤2** 在集群列表，单击指定集群的名称，选择“数据源>OBS数据源”。

**步骤3** 在“OBS数据源”列表中选择要更新的OBS数据源，单击所在行“操作”列的“更新配置”按钮。

**步骤4** 确认无误后，单击“确认”按钮，提交更新操作，更新过程大概需要10秒钟左右。

----结束

## 修改 OBS 数据源委托

### 操作场景

如果创建OBS数据源时绑定的委托不能满足使用需要，可以使用管理委托功能更换其他委托。

### 操作步骤

**步骤1** 在DWS管理管制台，选择“专属集群 > 集群列表”。

**步骤2** 在集群列表，单击指定集群的名称，选择“数据源>OBS数据源”。

**步骤3** 在“OBS数据源”列表中选择要修改的OBS数据源，单击所在行“操作”列的“管理委托”按钮，在弹窗中选择新的委托。

**步骤4** 确认无误后，单击“确认”按钮，提交修改委托操作，修改过程大概需要10秒钟左右。

----结束

## 删除 OBS 数据源

- 步骤1** 在DWS管理控制台，选择“专属集群 > 集群列表”。
  - 步骤2** 在集群列表，单击指定集群的名称，选择“数据源>OBS数据源”。
  - 步骤3** 在“OBS数据源”列表中选择要删除的OBS数据源，单击所在行“操作”列的“删除”按钮。
  - 步骤4** 确认无误后，单击“确认”按钮，提交删除操作，删除操作过程大概需要10秒钟左右。
- 结束

## 使用 OBS 数据源

DWS使用外表方式访问OBS上的数据。委托方式与非委托方式，在外表上体现出来的差异仅在于指定了不同的SERVER。

对于非委托方式，控制台提供的SERVER包含access\_key和secret\_access\_key参数，分别对应OBS访问协议的AK和SK值。

对于委托方式，控制台提供的SERVER包含access\_key、secret\_access\_key和security\_token参数，分别对应OBS访问协议的临时AK、临时SK和统一身份认证服务IAM中临时安全凭证的SecurityToken值。

在创建好OBS委托和OBS数据源之后，用户从控制台获得相应的包含委托信息的SERVER，假设OBS数据源名称为obs\_server。用户创建和使用外表与非委托方式无差异。关于如何使用OBS数据源，具体请参见《数据仓库服务开发指南》中“数据迁移 > 从OBS导入数据”章节。

如下示例为某普通用户jim通过外表读取OBS上的数据。

1. 参见以上步骤创建OBS数据源，名称为obs\_server。
2. 使用系统管理员dbadmin连接数据库，创建普通用户，并授权该普通用户使用OBS server和OBS外表权限。其中{Password}替换为用户实际密码，obs\_server替换成实际的OBS数据源名称。

```
CREATE USER jim PASSWORD '{Password}';
ALTER USER jim USEFT;
GRANT USAGE ON FOREIGN SERVER obs_server TO jim;
```

3. 使用普通用户jim连接数据库，建立OBS外表customer\_address，不包含分区列。

以下命令中，obs\_server替换为上面实际创建的OBS数据源名称。目录/user/obs/region\_orc11\_64stripe1/替换为实际数据文件存放的OBS目录，user表示用户的OBS桶名称。

```
CREATE FOREIGN TABLE customer_address
(
 ca_address_sk integer not null,
 ca_address_id char(16) not null,
 ca_street_number char(10) ,
 ca_street_name varchar(60) ,
 ca_street_type char(15) ,
 ca_suite_number char(10) ,
 ca_city varchar(60) ,
 ca_county varchar(30) ,
 ca_state char(2) ,
 ca_zip char(10) ,
 ca_country varchar(20) ,
 ca_gmt_offset decimal(36,33) ,
 ca_location_type char(20)
)
```

```
SERVER obs_server OPTIONS (
 FOLDERNAME '/user/obs/region_orc11_64stripe1/',
 FORMAT 'ORC',
 ENCODING 'utf8',
 TOTALROWS '20'
)
DISTRIBUTE BY roundrobin;
```

4. 通过外表查询OBS上的数据。

```
SELECT COUNT(*) FROM customer_address;
count

20
(1row)
```

# 9 DWS 集群数据安全与加密

## 9.1 开启 DWS 数据库用户三权分立

### 操作场景

默认情况下，创建DWS集群时指定的管理员用户属于数据库的系统管理员，能够创建其他用户和查看数据库的审计日志，即权限不分立，三权分立模式为关闭。

为了保护集群数据的安全，DWS支持对集群设置三权分立，使用不同类型的用户分别控制不同权限的模式。

默认的权限模型和开启三权分立后的权限模型，请参见《数据仓库服务数据库开发指南》的“管理数据库安全 > 管理用户及权限 > 三权分立”章节。

### 对系统的影响

- 修改安全配置参数并保存生效可能需要重启集群，将导致集群暂时不可用。

### 前提条件

修改集群安全配置必须同时满足以下两个条件：

- 集群状态为“可用”或“非均衡”。
- 集群任务信息不能处于“节点扩容”、“经典变更规格”、“配置中”、“升级中”、“重分布中”和“重启中”等状态。

### 操作步骤

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航树中，单击“专属集群 > 集群列表”。

**步骤3** 在集群列表中，单击指定集群的名称，然后左导航栏单击“安全设置”。

默认显示“配置状态”为“已同步”，表示页面显示的是数据库当前最新结果。

**步骤4** 在“安全设置”页面，设置三权分立。

开启“三权分立”后，分别设置“安全管理员”和“审计管理员”的用户名和密码，系统将会创建这两个用户。使用这两个用户可以连接数据库并执行数据库的相关操作。**默认为关闭状态。**

表 9-1 安全配置参数说明

| 参数名   | 参数解释                                                                                                                                                                                                                     | 样例值            |
|-------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------|
| 安全管理员 | 用户命名要求如下： <ul style="list-style-type: none"><li>只能由小写字母、数字或下划线组成。</li><li>必须以小写字母或下划线开头。</li><li>长度为6~64个字符。</li><li>用户名不能为DWS数据库的关键字。DWS数据库的关键字，具体请参见《数据仓库服务数据库开发指南》中“SQL参考 &gt; 关键字”章节。</li></ul>                      | security_admin |
| 密码    | 密码复杂度要求如下： <ul style="list-style-type: none"><li>密码长度为12~32个字符。</li><li>不能与用户名或倒序的用户名相同。</li><li>密码至少包含大写字母、小写字母、数字和特殊字符四类中的三类，其中可输入的特殊字符为：(~!?,.,:;_){}[]/&lt;&gt;@#%^&amp;*+ \\=-)。</li><li>对用户设置的密码进行弱密码检查。</li></ul> | -              |
| 确认密码  | 再次输入安全管理员密码。                                                                                                                                                                                                             | -              |
| 审计管理员 | 用户命名要求如下： <ul style="list-style-type: none"><li>只能由小写字母、数字或下划线组成。</li><li>必须以小写字母或下划线开头。</li><li>长度为6~64个字符。</li><li>用户名不能为DWS数据库的关键字。DWS数据库的关键字，具体请参见《数据仓库服务数据库开发指南》中“SQL参考 &gt; 关键字”章节。</li></ul>                      | audit_admin    |
| 密码    | 密码复杂度要求如下： <ul style="list-style-type: none"><li>密码长度为12~32个字符。</li><li>不能与用户名或倒序的用户名相同。</li><li>密码至少包含大写字母、小写字母、数字和特殊字符四类中的三类，其中可输入的特殊字符为：~!@#%^&amp;*()-_+=+ [{]};,.&lt;.&gt;/?。</li><li>弱密码检查。</li></ul>              | -              |
| 确认密码  | 再次输入审计管理员密码。                                                                                                                                                                                                             | -              |

**步骤5** 单击“应用”。

**步骤6** 在弹出的“保存配置”窗口中，选择是否勾选“立即重启集群”，然后单击“是”。

- 如果勾选“立即重启集群”，系统将保存“安全设置”页面的配置并立即重启集群，集群重启成功后安全设置将立即生效。
- 如果不勾选“立即重启集群”，系统将只保存“安全设置”页面的配置。稍后，用户需要手动重启集群才能使安全设置生效。

安全设置完成后，在“安全设置”页面，“配置状态”有如下3种状态：

- “应用中”：表示系统正在保存配置。
- “已同步”：表示配置已保存生效。
- “需重启生效”：表示配置已保存但还未生效。如需生效，需重启集群。

----结束

## 9.2 使用 KMS 对 DWS 集群进行加密

### 9.2.1 数据库加密简介

#### DWS 数据库加密

在DWS中，您可以为集群启用数据库加密，以保护静态数据。当您为集群启用加密时，该集群及其快照的数据都会得到加密处理。您可以在创建集群时启用加密。加密是集群的一项可选且不可变的设置。要从未加密的集群更改为加密集群，必须从现有集群导出数据，然后在已启用数据库加密的新集群中重新导入这些数据。数据库加密是在数据落盘时进行加密，即在往DWS写入数据时DWS会对数据进行加密，而在用户查询数据时DWS会将数据自动进行解密后再将结果返回给用户。

如果希望加密，可以在集群创建时启用加密。虽然加密是DWS集群中的一项可选设置，但建议您为集群启用该设置以保护数据。

#### 须知

- 数据库加密功能创建集群时选择是否开启，一旦开启就无法关闭。详情请参见[加密数据库](#)。普通集群创建完后，可以通过转加密集群功能，将集群转成加密集群。
- 开启数据库加密功能后，用于加密DWS数据库的KMS密钥在使用过程中不能被禁用、删除或冻结，否则会导致DWS集群异常或数据库不可用。
- 开启数据库加密功能后创建的快照不支持使用openAPI做恢复。

#### 查看数据库加密信息

**步骤1** 登录DWS管理控制台。

**步骤2** 单击左侧导航栏的“专属集群 > 集群列表”。

**步骤3** 在“专属集群”页面，单击指定集群名称，进入“集群详情”页面。

**步骤4** 在“集群详情”页面的“数据加密信息”区域查看数据库加密信息，如[表9-2](#)所示。

表 9-2 数据加密信息

| 参数       | 参数说明                        |
|----------|-----------------------------|
| 密钥名称     | 当“加密数据库”开启时，显示集群的数据库加密密钥。   |
| 上次密钥轮转时间 | 当“加密数据库”开启时，显示上一次加密密钥轮转的时间。 |
| 加密算法     | 当“加密数据库”开启时，显示集群的加密算法。      |

 说明

如果创建集群时默认没有开启数据库加密功能，则在集群详情页面不显示加密信息模块。

----结束

使用 KMS 服务加密 DWS 数据库

当选择KMS（密钥管理服务）对DWS进行密钥管理时，加密密钥层次结构有三层。按层次结构顺序排列，这些密钥为主密钥（CMK）、集群加密密钥（CEK）、数据库加密密钥（DEK）。

- 主密钥用于给CEK加密，保存在KMS中。
- CEK用于加密DEK，CEK明文保存在DWS集群内存中，密文保存在DWS服务中。
- DEK用于加密数据库中的数据，DEK明文保存在DWS集群内存中，密文保存在DWS服务中。

密钥使用流程如下：

1. 用户选择主密钥。
2. DWS随机生成CEK和DEK明文。
3. KMS使用用户所选的主密钥加密CEK明文并将加密后的CEK密文导入到DWS服务中。
4. DWS使用CEK明文加密DEK明文并将加密后的DEK密文保存到DWS服务中。
5. DWS将DEK明文传递到集群中并加载到集群内存中。

当该集群重启时，集群会自动通过API向DWS请求DEK明文，DWS将CEK、DEK密文加载到集群内存中，再调用KMS使用主密钥CMK来解密CEK，并加载到集群内存中，最后用CEK明文解密DEK，并加载到集群内存中，返回给集群。

加密密钥轮转

加密密钥轮转是指更新保存在DWS服务的密文。在DWS中，您可以轮转已加密集群的加密密钥CEK。

密钥轮转流程如下：

1. DWS集群启动密钥轮转。
2. DWS根据集群的主密钥来解密保存在DWS服务中的CEK密文，获取CEK明文。
3. 用获取到的CEK明文解密保存在DWS服务中的DEK密文，获取DEK明文。

4. DWS重新生成新的CEK明文。
5. DWS用新的CEK明文加密DEK并将DEK密文保存在DWS 服务中。
6. 用主密钥加密新的CEK明文并将CEK密文保存在DWS服务中。

您可以根据业务需求和数据类型计划轮转加密密钥的时间。为了提高数据的安全性，建议用户定期执行轮转密钥以避免密钥被破解的风险。一旦您发现密钥可能已泄露，请及时轮转密钥。

#### 说明

- DWS轮转集群的CEK时，集群相关的快照不需要轮转CEK，因为CEK不会在集群快照中进行保存，CEK明文是保存在DWS集群内存中的，密文是保存在DWS服务中的。
- 由于密钥轮转不更新DEK，所以对于数据的加解密不会有影响。

## 9.2.2 轮转加密密钥

在创建集群时，如果您在“高级配置”中为集群启用了“加密数据库”的功能，在集群创建成功后，您可以为集群轮转加密密钥。当普通集群转为加密集群时，也可以为集群轮转加密密钥。每执行一次密钥轮转，就更新一次集群加密密钥（CEK）。执行密钥轮转时，集群仍处于“可用”状态。

### 为 DWS 集群轮转加密密钥

- 步骤1 登录DWS管理控制台。
- 步骤2 在左侧导航树，单击“专属集群 > 集群列表”。
- 步骤3 在集群列表中找到所需要的集群，然后单击集群名称，进入“集群详情”页面。
- 步骤4 在“集群详情”页面的“数据加密信息”区域，单击“轮转密钥”。
- 步骤5 在弹出的对话框中，单击“是”为集群轮转一次加密密钥。

----结束

## 9.2.3 转加密集群

DWS支持在控制台对未加密的集群，在集群状态为“可用”的情况下转为加密集群。为了保障数据的安全，由于转加密集群是**不可逆的高危操作**，会重启集群，可能会导致有业务短暂不可用，用户请谨慎操作。

#### 说明

如果当前控制台界面不支持该功能，请联系技术支持人员。

### 约束与限制

- 存算分离集群仅9.1.0及以上集群版本支持数据库转加密功能。
- 如果该集群存在容灾关系，即使集群在“可用”状态也无法转加密，用户需先解除容灾关系，转加密成功后再重新建立容灾关系。
- 数据库加密功能一旦开启就无法关闭。
- 开启数据库加密功能后，用于加密DWS数据库的KMS密钥在使用过程中不能被禁用、删除或冻结，否则会导致DWS集群异常或数据库不可用。
- 开启数据库加密功能后创建的快照不支持使用openAPI做恢复。

- 默认情况下，只有华为云账号或拥有Security Administrator权限的用户才具备查询委托和创建委托的权限。账号中的IAM用户，默认没有查询委托和创建委托的权限，此时需联系有权限的用户在当前页面完成对DWS的委托授权。

## 创建 KMS 委托

### 操作场景

转加密集群前需要用户提前创建好授权给DWS具有KMS Administrator权限的委托。

### 操作步骤

- 步骤1** 鼠标移动至页面右上角账号，单击“统一身份认证”，进入统一身份认证服务页面。
- 步骤2** 在左侧导航栏单击“委托”，在委托页面右上角单击“创建委托”。
- 步骤3** 创建委托时委托类型选择“云服务”，云服务选择“DWS”。
- 步骤4** 单击“完成”后，在弹窗中单击“立即授权”对委托授予KMS服务的“KMS Administrator”权限。
- 步骤5** 单击“下一步”，选择授权资源范围为“所有资源”或需要访问的资源，然后确认无误后提交。

----结束

## 操作步骤

- 步骤1** 登录DWS管理控制台，在左侧导航栏中，选择“专属集群 > 集群列表”。
- 步骤2** 集群列表中，在指定集群所在行操作列选择“更多 > 转加密集群”。

### 说明

集群列表操作栏功能键位置不固定，如果该局部点部分功能不支持，“更多”里的功能键可能会前移和“更多”按钮并列。

- 步骤3** 在弹出选择框中，选择密钥源、密钥名称和加密算法，将集群转成加密集群。
- 方式一：从密钥中选择密钥名称。
  - 方式二：输入密钥ID。输入在KMS侧给当前租户进行授权的KMS密钥ID。  
当跳转至KMS侧创建授权时被授权对象**必须是账号，不能是用户**。并且授权操作必须至少包含“**查询密钥信息**”、“**加密数据**”、“**解密数据**”3项权限。
- 步骤4** 转换后用户可单击集群名称，进入“集群详情”页面查看，详情请参见[查看数据库加密信息](#)。

----结束

# 10 DWS 集群管理

## 10.1 查看 DWS 集群详情

登录DWS管理控制台，在左侧导航树，选择“专属集群 > 集群列表”，在集群列表中找到所需要的集群，然后单击集群名称，进入“集群详情”页面。

在“集群详情”页面，可以查看如下相关信息：

- 基本信息：相关参数如[表10-1](#)所示。
- 连接信息：相关参数如[表10-2](#)所示。
- 网络：相关参数如[表10-3](#)所示。
- 存储/备份空间：相关参数如[表10-4](#)所示。
- 数据加密信息：相关参数如[表10-5](#)所示。

 说明

创建集群时选择开启数据加密功能可查看此模块。

表 10-1 基本信息

| 参数     | 参数说明                                     |
|--------|------------------------------------------|
| 集群名称   | 集群的名称，创建集群时设置。                           |
| 集群状态   | 集群的状态信息，详情请参见 <a href="#">集群状态</a> 。     |
| 参数配置状态 | 显示集群关联的参数配置状态。                           |
| 任务信息   | 集群任务的状态信息，详情请参见 <a href="#">集群任务信息</a> 。 |
| 规格详情   | 显示集群的节点规格详情。                             |
| 节点数量   | 集群的节点数量。                                 |
| 集群ID   | 集群的ID。                                   |
| 集群版本   | 集群的版本信息。                                 |

| 参数   | 参数说明                                                                                                                                    |
|------|-----------------------------------------------------------------------------------------------------------------------------------------|
| 节点规格 | 显示集群规格。                                                                                                                                 |
| 企业项目 | 集群所属的企业项目。单击企业项目名称，可进入企业项目服务的 Console 页面对企业项目进行查看和编辑。                                                                                   |
| 集群时区 | 集群的时区，包含集群节点 OS 时区、集群日志文件和集群数据仓库时区，此处仅支持修改集群节点 OS 和日志文件时区，集群数据库时区需要评估业务影响通过 GUC 参数 timezone 修改，详情请参见 <a href="#">修改 DWS 集群 GUC 参数</a> 。 |

表 10-2 连接信息

| 参数名     | 参数解释                                                                                                                                                                                                                                                                                  |
|---------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 内网域名    | <p>通过内部网络访问集群数据库的域名地址，对应所有的内网 IP 地址，即所有 CN 的地址。内网访问域名在创建集群时自动生成。</p> <p><b>说明</b></p> <ul style="list-style-type: none"><li>如果集群名称不符合域名规范，默认访问域名前缀会相应做一些调整。</li><li>不支持负载均衡能力。</li></ul> <p>单击“修改”可以修改内网访问域名。访问域名由字母、数字、中划线组成，以大小写字母开头，长度为 4~63 个字符。</p>                              |
| 内网 IP   | <p>通过内部网络访问集群数据库的 IP 地址。</p> <p><b>说明</b></p> <ul style="list-style-type: none"><li>内网访问 IP 地址在创建集群时自动生成，生成后的 IP 地址是固定的。</li><li>内网访问 IP 的数量对应的是 CN 节点的个数，可以通过登录任一节点连接到集群。</li><li>通过内网访问某个固定的 IP，资源池会集中在一个 CN 上。</li><li>集群开启 IPv6 功能后，内网 IP 会同时展示 IPv4 和 IPv6，您可根据需要进行使用。</li></ul> |
| 公网域名    | <p>通过外部网络访问集群数据库的域名地址。</p> <p><b>说明</b></p> <p>不支持负载均衡能力。</p>                                                                                                                                                                                                                         |
| 公网 IP   | <p>通过外部网络访问集群数据库的 IP 地址。</p> <p><b>说明</b></p> <ul style="list-style-type: none"><li>如果创建集群时没有绑定弹性 IP，“公网 IP”显示为空，可以单击“编辑”为集群绑定弹性 IP。</li><li>如果创建集群时绑定了弹性 IP，可以单击“编辑”为集群解绑弹性 IP。</li></ul>                                                                                            |
| 初始管理员用户 | 创建集群时指定的数据库管理员用户。当用户第一次连接集群时，需要使用初始数据库管理员用户及其密码连接到默认数据库。                                                                                                                                                                                                                              |
| 端口      | 通过公网或者内网访问集群数据库的端口号，端口号在创建集群时指定。                                                                                                                                                                                                                                                      |

| 参数名      | 参数解释                                                                                                                                                                                 |
|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 默认数据库    | 创建集群时默认自动创建的数据库。当用户第一次连接集群时，需要连接到该默认数据库。                                                                                                                                             |
| 弹性负载均衡地址 | <p>为实现集群高可用，解决CN单点问题，集群创建后需手动绑定弹性负载均衡（ELB），连接集群时建议连接ELB地址。</p> <p><b>说明</b></p> <p>如果集群为IPv4集群，则仅支持手动绑定IPv4的弹性负载均衡（ELB）；如果集群为IPv6双栈集群，则仅支持手动绑定IPv6双栈的弹性负载均衡（ELB），暂不支持交叉场景的ELB绑定。</p> |

表 10-3 网络

| 参数    | 参数说明                                                                                                                                                                                                                                                                                                                                         |
|-------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 区域    | 集群的工作区域。                                                                                                                                                                                                                                                                                                                                     |
| 可用分区  | 显示创建集群时所选择的可用分区信息。                                                                                                                                                                                                                                                                                                                           |
| 虚拟私有云 | <p>VPC信息，创建集群时所选。</p> <p>VPC即虚拟私有云，是通过逻辑方式进行网络隔离，提供安全、隔离的网络环境。</p> <p>DWS集群创建成功后，其虚拟私有云将不能更改为其他的虚拟私有云，但是您可以编辑和修改当前的虚拟私有云。单击此处的虚拟私有云名称，进入虚拟私有云详情页面，您可以对虚拟私有云进行设置。有关虚拟私有云的详细操作，请参见《虚拟私有云用户指南》中的“虚拟私有云和子网”章节。</p>                                                                                                                               |
| 子网    | <p>子网信息，创建集群时所选。</p> <p>通过子网提供与其他网络隔离的、可以独享的网络资源，以提高网络安全。</p> <p>DWS集群创建成功后，其子网将不能更改为其他子网，但是您可以编辑和修改当前的子网。单击此处的子网名称，进入子网详情页面，您可以对子网进行设置。有关子网的详细操作，请参见《虚拟私有云用户指南》中的“虚拟私有云和子网 &gt; 修改子网网络信息”章节。</p>                                                                                                                                            |
| 安全组   | <p>显示创建集群时所选择的安全组信息。</p> <p>DWS集群创建成功后，安全组支持更改为其他安全组，也可以在当前的安全组中添加、删除或修改安全组规则。集群更换安全组过程中有短暂的业务中断，请谨慎操作。为了更好的网络性能，选择安全组时不要多于5个。</p> <ul style="list-style-type: none"><li>• 修改为其他安全组：单击此处安全组名称右边的修改按钮，选择需要变更的安全组名称，单击确认，即修改安全组完成。</li><li>• 修改已有安全组规则：单击此处的安全组名称，进入安全组详情页面，您可以对安全组进行设置。有关安全组的详细操作，请参见《虚拟私有云用户指南》中的“安全性 &gt; 安全组”章节。</li></ul> |

表 10-4 存储/备份空间

| 参数       | 参数说明                                                                                                                                                                           |
|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 存储空间     | 显示存储类型为超高I/O，以及集群存储空间的使用状况。<br><b>说明</b> <ul style="list-style-type: none"><li>已使用存储容量不包含外表OBS上的数据，只包含单独的DWS本身的数据，其中包括文件，日志，快照，索引等数据。</li><li>显示可存储空间容量为真实磁盘容量的二分之一。</li></ul> |
| 备份空间     | 显示集群备份已使用的容量空间、备份的赠送空间以及备份已使用的超出赠送空间外的收费空间。                                                                                                                                    |
| 冷分区数据已用量 | 冷数据已使用的OBS容量。<br><b>说明</b><br>该参数显示OBS已使用容量每4小时同步一次。                                                                                                                           |

表 10-5 数据加密信息

| 参数       | 参数说明                        |
|----------|-----------------------------|
| 密钥名称     | 当“加密数据库”开启时，显示集群的数据库加密密钥。   |
| 上次密钥轮转时间 | 当“加密数据库”开启时，显示上一次加密密钥轮转的时间。 |

## 修改集群名称

DWS集群创建成功后，您可以根据需求，修改集群的名称，支持修改为包含中文字符的集群名称。

集群名称修改成功后，当前集群的所有节点名称，也会相应进行修改。

### 说明

- 如果当前控制台界面不支持修改集群名称，请联系技术支持人员升级使用。
- 如果修改集群名称任务失败，不影响集群正常功能，请联系技术支持人员进行恢复。

### 约束与限制

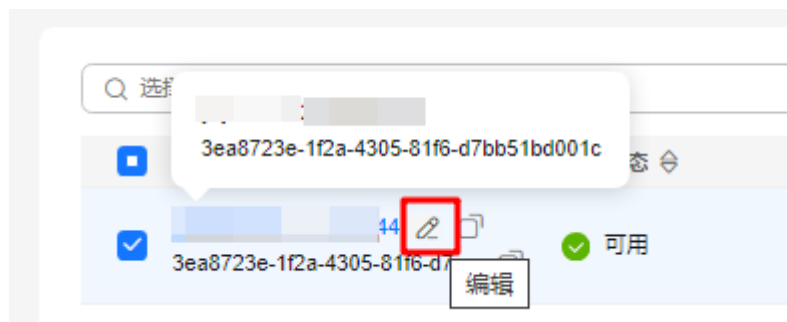
当集群为“不可用”状态或正在进行其他任务，则不支持修改集群的名称，待集群状态恢复为“可用”或其他任务完成后再进行名称修改操作。

### 修改方式一：

**步骤1** 登录DWS管理控制台。

**步骤2** 在集群列表中单击指定集群名称旁边的修改符号进行修改。

图 10-1 集群列表修改集群名称



**步骤3** 在弹窗中输入新集群名称。

**步骤4** 确认无误后，单击“确定”进行集群名称修改。

----结束

**修改方式二：**

**步骤1** 登录DWS管理控制台。

**步骤2** 在集群列表中单击指定集群名称。

**步骤3** 进入“集群详情”页面，在“基本信息”模块单击集群名称旁边的修改符号进行修改。

**步骤4** 确认无误后，单击“确定”下发集群修改任务，待任务完成后，集群名称修改成功。

----结束

## 10.2 查看 DWS 集群状态

在DWS管理控制台的“专属集群 > 集群列表”页面，用户可以在集群列表中查看集群的概要信息，例如集群状态、任务信息、节点规格和近期事件等信息。

### 查看集群概览信息

登录DWS管理控制台，在左侧导航树，选择“专属集群 > 集群列表”，在集群列表中会显示所有DWS 集群，集群数量较多时，可采用翻页显示，您可以查看任何状态下的集群。

在集群列表上方，可单击搜索框，在下拉列表中根据筛选条件（集群名称、集群状态、集群版本、任务信息、节点规格、近期事件和企业项目）查找所需要的集群。单击

，可以刷新集群状态。您也可以下划到“资源标签”筛选处，然后根据标签过滤条件搜索集群，详情请参考[根据标签搜索集群](#)。

集群列表默认按时间顺序排列，时间最近的集群显示在最前端。集群列表参数说明如[表10-6](#)所示。

表 10-6 集群列表参数

| 参数   | 参数说明                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 集群名称 | 集群的名称，创建集群时设置。<br><b>说明</b><br>如果当前控制台界面不支持修改集群名称，请联系技术支持人员。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| 集群状态 | 集群的状态信息，详情请参见 <a href="#">集群状态</a> 。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| 集群版本 | 当前集群的集群版本，详情请参见 <a href="#">表10-8</a> 。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| 任务信息 | 集群任务的状态信息，详情请参见 <a href="#">集群任务信息</a> 。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| 节点规格 | 集群的节点规格。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| 近期事件 | 显示集群近期的事件个数，单击可查看事件详情。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| 企业项目 | 集群所属的企业项目。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| 操作   | <ul style="list-style-type: none"><li>● 更多<ul style="list-style-type: none"><li>- 转加密集群：请参见<a href="#">转加密集群</a>。</li><li>- 查看监控指标：请参见<a href="#">在Cloud Eye查看DWS集群监控</a>。</li><li>- 重启：单击“重启”，重启集群。详情请参见<a href="#">重启集群</a>。</li><li>- 启动：已停止的集群单击“启动”，启动集群。详情请参见<a href="#">启动集群</a>。</li><li>- 停止：单击“停止”，停止运行的集群。详情请参见<a href="#">停止集群</a>。</li><li>- 节点变更：请参见<a href="#">DWS集群节点扩缩容</a>。</li><li>- 规格变更：请参见<a href="#">DWS集群规格变更</a>。</li><li>- 重置密码：请参见<a href="#">重置DWS数据库系统管理员密码</a>。</li><li>- 创建快照：请参见<a href="#">备份与恢复DWS集群</a>。</li><li>- 删除：单击“删除”，删除集群。</li><li>- 管理CN节点：请参见<a href="#">增删DWS集群CN节点</a>。</li></ul></li></ul> <b>说明</b><br>集群列表操作栏功能键位置不固定，如果该节点部分功能不支持，“更多”里的功能键可能会前移和“更多”按钮并列。 |

集群状态

表 10-7 集群状态说明

| 状态 | 说明        |
|----|-----------|
| 可用 | 表示集群工作正常。 |

| 状态      | 说明                                                                                                                                                                                                                                                                                                                                                                                                                            |
|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 只读      | <p>集群存储容量使用率或单节点磁盘使用率大于90%时会进入此状态，该状态下集群仍可工作但只能支持查询操作，不支持任何写操作。当集群进入只读状态时，可通过<a href="#">解除只读</a>功能进行解除，如解除失败请联系技术支持人员解除只读状态。</p> <p>解除集群只读状态后，建议您参考以下措施进行处理：</p> <ul style="list-style-type: none"><li>● 使用SQL客户端工具，以管理员用户连接数据库，执行以下命令定期清理和回收存储空间。<br/>VACUUM FULL;<br/>DWS数据仓库中保存的数据在删除后，可能没有释放占用的磁盘空间形成脏数据，导致磁盘浪费，建议定期清理存储空间。</li><li>● 建议用户检查磁盘容量，分析现有集群规格是否满足业务需求，若不满足，建议您对集群进行扩容，具体操作请参见<a href="#">集群扩容</a>。</li></ul> |
| 非均衡     | <p>如果集群中存在gtm或dn的角色和初始角色不一致，就认为处于非均衡状态。非均衡状态下会出现某些节点主实例增多，从而负载压力较大。这种情况下集群状态是正常的，但整体性能要不如均衡状态。建议业务低峰期，切换为可用状态，详情操作请参见<a href="#">集群主备恢复</a>。</p>                                                                                                                                                                                                                                                                              |
| 重分布中    | <p>集群扩容时新节点添加完成后，原节点存储的业务数据明显多于新节点，此时系统自动在所有节点重新分布保存数据。该状态下集群仍可工作。</p>                                                                                                                                                                                                                                                                                                                                                        |
| 重分布失败   | <p>调整数据分布情况失败，但没有数据丢失。该状态下集群仍可工作。建议用户联系技术支持人员进行处理。</p>                                                                                                                                                                                                                                                                                                                                                                        |
| 节点故障    | <p>表示集群中个别节点出现问题无法正常工作，但整个集群正常。建议用户联系技术支持人员进行处理。</p>                                                                                                                                                                                                                                                                                                                                                                          |
| 不可用     | <p>表示集群无法提供数据库服务。建议用户联系技术支持人员进行处理。</p>                                                                                                                                                                                                                                                                                                                                                                                        |
| 创建中     | <p>表示集群正在创建中。</p>                                                                                                                                                                                                                                                                                                                                                                                                             |
| 创建失败    | <p>表示集群创建失败。</p>                                                                                                                                                                                                                                                                                                                                                                                                              |
| 创建中，恢复中 | <p>表示集群正在创建中，且该集群是快照恢复而创建的集群，此时集群正在恢复中。当对快照执行了恢复操作时，会恢复快照到新集群，此时该新集群会进入此状态。</p>                                                                                                                                                                                                                                                                                                                                               |
| 删除中     | <p>表示集群正在删除中。</p>                                                                                                                                                                                                                                                                                                                                                                                                             |
| 待重启     | <p>表示集群修改过GUC参数，需重启才能生效。若并未重启集群，此时部分运维操作会被限制。在用户手动重启集群后，GUC参数生效，集群状态将恢复为“可用”。</p>                                                                                                                                                                                                                                                                                                                                             |
| 已停止     | <p>表示集群已停止。</p>                                                                                                                                                                                                                                                                                                                                                                                                               |

## 集群版本

表 10-8 集群版本说明

| 状态            | 说明                                                                                                       |
|---------------|----------------------------------------------------------------------------------------------------------|
| x.x.x（集群当前版本） | 当前集群版本。为了提高系统运行的稳定性，如果存在可升级到新版本的集群，在可升级集群版本下显示“可升级”按钮，单击跳转至集群详情的升级管理界面，操作详情请参见 <a href="#">DWS集群升级</a> 。 |

## 集群任务信息

表 10-9 任务信息说明

| 状态         | 说明                   |
|------------|----------------------|
| 创建快照中      | 表示集群正在执行创建快照的操作。     |
| 创建快照失败     | 表示集群执行快照创建操作失败。      |
| 观察中        | 表示集群在用户自助升级操作后待提交阶段。 |
| 配置中        | 表示系统正在保存集群参数的修改。     |
| 重启中        | 表示集群正在重启过程中。         |
| 重启失败       | 表示集群重启操作失败。          |
| 转加密集群中     | 表示集群正在转加密集群过程中。      |
| 转加密集群失败    | 表示集群转加密失败。           |
| 节点扩容       | 表示集群正在扩容中。           |
| 扩容失败       | 表示集群扩容失败。            |
| 磁盘扩容中      | 表示集群正在磁盘扩容中。         |
| 磁盘扩容失败     | 表示集群磁盘扩容失败。          |
| 弹性负载均衡绑定中  | 表示集群正在绑定弹性负载均衡。      |
| 弹性负载均衡绑定失败 | 表示集群绑定弹性负载均衡失败。      |
| 弹性负载均衡解绑中  | 表示集群正在解绑弹性负载均衡。      |

| 状态         | 说明                                                                                                                                                                                                                                                                                                                  |
|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 弹性负载均衡解绑失败 | 表示集群解绑弹性负载均衡失败。                                                                                                                                                                                                                                                                                                     |
| 主备恢复中      | 表示集群正在主备恢复中。                                                                                                                                                                                                                                                                                                        |
| 主备恢复失败     | 表示集群主备恢复失败，失败原因有以下几种情况： <ul style="list-style-type: none"><li>正在执行DN REDO，请稍后重试，等待redo做完之后重试主备恢复。</li><li>查询DN REDO失败，请查看租户侧日志分析失败原因。</li><li>存在主备追赶，请稍后重试，等待主备追赶做完之后重试主备恢复。</li><li>查询主备追赶失败，请查看租户侧日志分析失败原因。</li><li>主备恢复失败，请联系技术支持人员或稍后重试。请查看租户侧日志分析失败原因。</li><li>集群状态异常，无法进行主备恢复操作，请检查集群状态是否是normal状态。</li></ul> |
| 弹性变更规格中    | 表示集群正在弹性变更集群规格中。                                                                                                                                                                                                                                                                                                    |
| 弹性变更规格失败   | 表示集群弹性规格变更失败。                                                                                                                                                                                                                                                                                                       |
| 维护中        | 表示集群正在进行维护变更操作，如集群升级、插件升级等。                                                                                                                                                                                                                                                                                         |
| 维护失败       | 表示集群升级操作失败。                                                                                                                                                                                                                                                                                                         |
| 停止中        | 表示集群正在停止中。                                                                                                                                                                                                                                                                                                          |
| 启动中        | 表示集群正在启动中。                                                                                                                                                                                                                                                                                                          |
| 巡检中        | 表示集群正在做变更前的巡检。                                                                                                                                                                                                                                                                                                      |
| 巡检失败       | 表示集群巡检操作失败。                                                                                                                                                                                                                                                                                                         |

## 10.3 查看 DWS 集群拓扑图

### 概述

集群拓扑图向用户展示了集群内所有节点的拓扑结构，方便用户查看节点状态、节点进程、节点IP地址等信息。

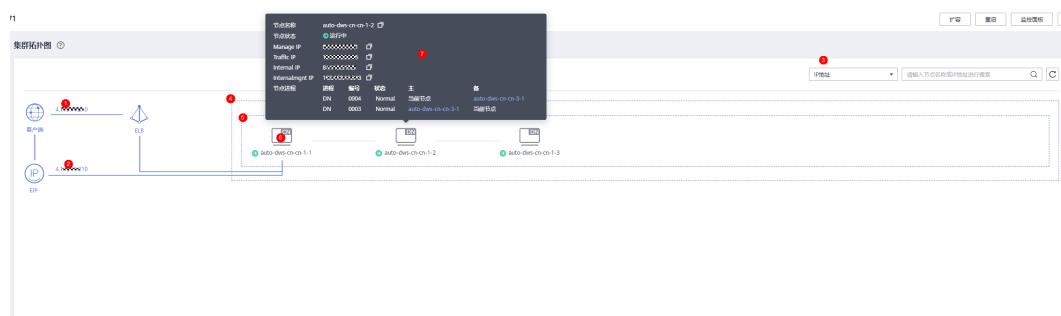
#### 说明

- 集群拓扑图分为拓扑结构展示与节点进程展示两个功能。
- 拓扑结构节点展示仅8.0.0及以上集群版本支持，节点进程展示仅8.2.0及以上集群版本支持。

查看集群拓扑图

- 步骤1 登录DWS管理控制台。
- 步骤2 在集群列表中单击指定集群名称。
- 步骤3 进入“集群详情”页面，在左导航栏切换至“集群拓扑”页签。
- 步骤4 页面右上方可选择“IP地址”或“节点名称”，在搜索框完成输入后可查看该“IP地址”或“节点名称”在集群拓扑图中的位置。
- 结束

拓扑展示逻辑



节点拓扑整体示意图如上所示，其中按照图中标注的顺序解释如下：

1. 集群所绑定的ELB的公网IP地址（若绑定的ELB没有绑定弹性IP则显示为服务地址）。
2. 集群所绑定的EIP的IP地址。
3. 搜索分类，可以按照“IP地址”或“节点名称”进行精确搜索。
4. 该框表示集群内所有的Ring环。
5. 该框表示Ring环，一行为一个Ring环，该Ring环内的每个图标表示其内的一个节点，如果创建的集群Ring环大于等于3，可查看CN分散部署情况。
6. 节点以图标的形式进行表示，图标右上角为该节点的类型，暂只显示“CN”和“DN”两种类型（如果节点中有CN进程，则图标为CN；如果节点内没有CN进程，则显示DN图标）。
7. 当鼠标悬停在节点图标上时，展示节点详细信息，包括节点名称、节点状态、节点的不同IP地址、节点进程。

## 拓扑图专业名词解释

表 10-10 集群结构描述

| 名称   | 描述                                                                 | 说明                                                                                                                                                                                                                                             |
|------|--------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ELB  | 弹性负载均衡（Elastic Load Balance，简称ELB）是将访问流量根据分配策略分发到后端多台服务器的流量分发控制服务。 | 用户使用客户端连接DWS集群时，如果用户仅连接一个CN节点地址，通过该CN节点内网IP或弹性公网IP连接时，只能连接到固定的CN节点上，存在CN单点问题。如果通过内网域名连接，域名解析服务会对每个客户端随机选择一个内网/公网IP地址，其解析机制并不能保证负载均衡，同样也存在CN单点问题。因此引入了弹性负载均衡服务（Elastic Load Balance，下称ELB）解决集群访问的单点问题。详情请参见 <a href="#">为DWS集群绑定和解绑弹性负载均衡</a> 。 |
| EIP  | 弹性公网IP（Elastic IP，简称EIP）提供独立的公网IP资源，包括公网IP地址与公网出口带宽服务。             | 可以与弹性云服务器、虚拟IP、弹性负载均衡、NAT网关等资源灵活地绑定及解绑。                                                                                                                                                                                                        |
| Ring | 安全环，集群内部的故障隔离域，主要作用是故障隔离，环内主机出现故障，故障不会扩散到环外。                       | <p>如果环内某一单节点故障，所有DN节点在环内都有副本，数据不会丢失。</p> <p>例如Server1发生故障后，DN1的备节点在Server2上，DN2的备节点在Server3上，DN3的备节点在Server3上，每个Server运行4个主DN，环内的各主机性能仍然保证均衡。</p> <p>整个集群可以承受的主机故障数量范围为1~安全环的数量。</p> <p><b>说明</b><br/>集群按照安全环节点数量的倍数进行扩容，也就是以环为最小单位进行扩容。</p>   |

表 10-11 节点 IP 描述

| 名称          | 描述                 | 说明                            |
|-------------|--------------------|-------------------------------|
| Manage IP   | 数据仓库节点与管理面连接通信的IP。 | 主要用于管理面命令下发，以及节点状态、监控信息上报。    |
| Traffic IP  | 数据仓库节点对外开放的IP。     | 该IP用于绑定EIP、ELB或直接通过VPC内部进行连接。 |
| Internal IP | 数据仓库集群内部数据通信IP。    | -                             |

| 名称              | 描述                      | 说明 |
|-----------------|-------------------------|----|
| Internalmgnt IP | 数据仓库集群内部管理命令在节点之间通信 IP。 | -  |

表 10-12 节点进程描述

| 名称  | 描述                                                                                      | 说明                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|-----|-----------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CMS | 集群管理模块（Cluster Manager）。管理和监控分布式系统中各个功能单元和物理资源的运行情况，确保整个系统的稳定运行。<br>CMS为该模块中的CM Server。 | CM由CM Agent、OM Monitor和CM Server组成。 <ul style="list-style-type: none"><li>CM Agent：负责监控所在主机上主备GTM、CN、主备DN的运行状态并将状态上报给CM Server。同时负责执行CM Server下发的仲裁指令。集群的每台主机上均有CM Agent进程。</li><li>OM Monitor：看护CM Agent的定时任务，其唯一的任务是在CM Agent停止的情况下将CM Agent重启。如果CM Agent重启不了，则整个主机不可用，需要人工干预。</li></ul> <b>说明</b><br>CM Agent重启的情况很少发生，如果出现可能是因为系统资源不够用导致无法启动新进程。 <ul style="list-style-type: none"><li>CM Server：根据CM Agent上报的实例状态判定当前状态是否正常，是否需要修复，并下发指令给CM Agent执行。</li></ul> DWS提供了CM Server的主备实例方案，以保证集群管理系统本身的高可用性。正常情况下，CM Agent连接主CM Server，在主CM Server发生故障的情况下，备CM Server会主动升为主CM Server，避免出现CM Server单点故障。 |
| GTM | 全局事务管理器（Global Transaction Manager），负责生成和维护全局事务ID、事务快照、时间戳等全局唯一的信息。                     | 整个集群只有一组GTM：主、备GTM各一个。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |

| 名称  | 描述                                                                   | 说明                                                                                                                                                                           |
|-----|----------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CN  | 协调节点（Coordinator）。负责接收来自应用的访问请求，并向客户端返回执行结果；负责分解任务，并调度任务分片在各DN上并行执行。 | 集群中，CN有多个并且CN的角色是对等的（执行DML语句时连接到任何一个CN都可以得到一致的结果）。只需要在CN和应用程序之间增加一个负载均衡器，使得CN对应用是透明的。CN故障时，由负载均衡自动路由连接到另外一个CN。<br><br>当前分布式事务框架下无法避免CN之间的互连，为了减少GTM上线程过多导致负载过大，建议CN配置数目≤10个。 |
| CCN | 中心协调节点（Central Coordinator）。                                         | DWS通过CCN（Central Coordinator）负责集群内的资源全局负载控制，以实现自适应的动态负载管理。CM在第一次集群启动时，通过集群部署形式，选择编号最小的CN作为CCN。若CCN故障之后，由CM选择新的CCN进行替换。                                                       |
| DN  | 数据节点（Datanode）。负责存储业务数据（支持行存、列存、混合存储）、执行数据查询任务以及向CN返回执行结果。           | 在集群中，DN有多个。每个DN存储了一部分数据。如果DN无高可用方案，则故障时会导致该实例上的数据无法访问。                                                                                                                       |

## 10.4 管理 DWS 集群连接

### 10.4.1 管理 DWS 集群访问域名

#### 概述

域名（Domain Name）是由一串用“点”分隔的字符组成的Internet上某一台计算机或计算机组的名称，如www.example.com。用户在Web浏览器的地址栏输入域名来访问某个网站或者Web应用程序。

DWS支持通过内网域名和公网域名访问集群。

内网域名是指通过内部网络访问集群数据库的域名地址。内网域名在创建集群时自动生成。

公网域名是指通过外部网络访问集群数据库的域名地址。如果集群没有绑定弹性IP，则不支持以“公网域名”的方式访问集群。如果集群在创建时绑定了弹性IP，公网域名在创建集群时自动生成。

#### 说明

公网域名和内网域名均不支持负载均衡能力，如果需要使用负载均衡能力，请参见配置JDBC连接（使用负载均衡方式）。

创建了集群后，用户可以根据实际需求设置访问集群的内网域名和公网域名，具体包含如下操作：

- [修改内网域名](#)
- [创建公网域名](#)
- [修改公网域名](#)
- [释放公网域名](#)

## 修改内网域名

内网域名在创建集群时自动生成，集群创建成功后用户可以根据实际情况修改默认生成的域名。

修改内网域名的操作步骤如下：

- 步骤1** 登录DWS管理控制台。
- 步骤2** 在左侧导航栏中，单击“专属集群 > 集群列表”。
- 步骤3** 在集群列表中找到所需要的集群，然后单击集群名称，进入“集群详情”页面。
- 步骤4** 在“连接信息”区域下，显示自动生成的内网域名，单击“内网域名”旁边的“修改”按钮。
- 步骤5** 在“修改内网域名”对话框中，输入目标域名，然后单击“确定”完成域名的修改。

内网域名由字母，数字和中划线组成，以大小写字母开头，长度为4~63个字符。

域名修改完成后，单击内网域名旁边的复制按钮，可以复制内网域名。

----结束

## 创建公网域名

创建集群时默认不绑定弹性IP，即不启用公网访问集群。集群创建成功后，如果用户想通过公网访问集群，可以先为集群绑定弹性IP再创建公网域名。

#### 说明

默认情况下，只有华为云账号或拥有Security Administrator权限的用户才具备查询委托和创建委托的权限。账号中的IAM用户，默认没有查询委托和创建委托的权限，在使用弹性IP绑定功能时页面会屏蔽绑定按钮，此时需联系有“DWS Administrator”权限的用户在当前页面完成对DWS的委托授权。

创建公网域名的操作步骤如下：

- 步骤1** 登录DWS管理控制台。
- 步骤2** 在左侧导航栏中，单击“专属集群 > 集群列表”。

**步骤3** 在集群列表中找到所需要的集群，然后单击集群名称，进入“集群详情”页面。

**步骤4** 在“连接信息”区域下，“公网域名”和“公网IP”显示为空，单击“编辑”为集群绑定弹性IP。

**步骤5** 在“绑定/解绑弹性IP”弹出窗的下拉框中，选择一个弹性IP，可以为指定的CN节点绑定弹性IP。

如果下拉框中没有可用的弹性IP，可以单击“查看弹性IP”进入弹性公网IP页面创建一个满足需要的弹性IP。新的弹性IP创建成功后，单击绑定按钮，“弹性IP”下拉框中就会出现新创建的弹性IP。

弹性IP绑定成功后，在“连接信息”区域下显示具体的公网IP。

**步骤6** 在“连接信息”区域下，单击“公网域名”旁边的“创建”为集群创建公网域名。

**步骤7** 在“申请公网域名”弹出窗口中，输入目标域名，单击“确定”。

公网域名由字母、数字和中划线组成，以大小写字母开头，长度为4~63个字符。

公网域名创建成功后，在“连接信息”区域下显示具体的公网域名。单击复制按钮



，可以复制公网域名。

----结束

## 修改公网域名

如果集群在创建时绑定了弹性IP，公网域名在创建集群时自动生成。集群创建成功后，用户可以根据实际需求修改公网域名。

修改公网域名的操作步骤如下：

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，单击“专属集群 > 集群列表”。

**步骤3** 在集群列表中找到所需要的集群，然后单击集群名称，进入“集群详情”页面。

**步骤4** 在“连接信息”区域下，显示“公网域名”，单击“修改”。

**步骤5** 在“修改公网域名”弹出窗口中，输入目标域名，单击“确定”。

----结束

## 释放公网域名

集群创建成功后，对于不需要通过公网访问的集群，用户可以释放集群的公网域名。

释放公网域名的操作步骤如下：

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，单击“专属集群 > 集群列表”。

**步骤3** 在集群列表中找到所需要的集群，然后单击集群名称，进入“集群详情”页面。

**步骤4** 在“连接信息”区域下，显示公网域名，单击“释放”。

**步骤5** 在“释放域名”确认窗口中，单击“是”。

----结束

## 10.4.2 为 DWS 集群绑定和解绑弹性负载均衡

### 概述

用户使用客户端连接DWS集群时，如果用户仅连接一个CN节点地址，通过该CN节点内网IP或弹性公网IP连接时，只能连接到固定的CN节点上，存在CN单点问题。如果通过内网域名连接，域名解析服务会对每个客户端随机选择一个内网/公网IP地址，其解析机制并不能保证负载均衡，同样也存在CN单点问题。因此引入了弹性负载均衡服务（Elastic Load Balance，下称ELB）解决集群访问的单点问题。

弹性负载均衡（ELB）是将访问流量根据转发策略分发到后端多台弹性云服务器的流量分发控制服务，可以通过流量分发扩展应用系统对外的服务能力，提高应用程序的容错能力。了解更多，请参见《弹性负载均衡用户指南》。

利用ELB健康检查机制可将集群的CN请求快速转发到正常工作的CN节点上，当有CN故障节点时，也能第一时间切换流量到健康节点上，最大化降低集群访问故障。

#### 说明

- 该特性仅8.1.1.200及以上集群版本支持。
- 为保证集群负载均衡和高可用，**避免业务中断风险**，客户生产业务下，要求集群创建后进行ELB绑定。
- ELB不支持跨库访问。

### 约束与限制

- 绑定ELB时，需创建和DWS集群相同的区域、VPC以及企业项目的ELB进行绑定。
- DWS的ELB功能仅支持独享型ELB规格，不支持共享型ELB绑定操作。

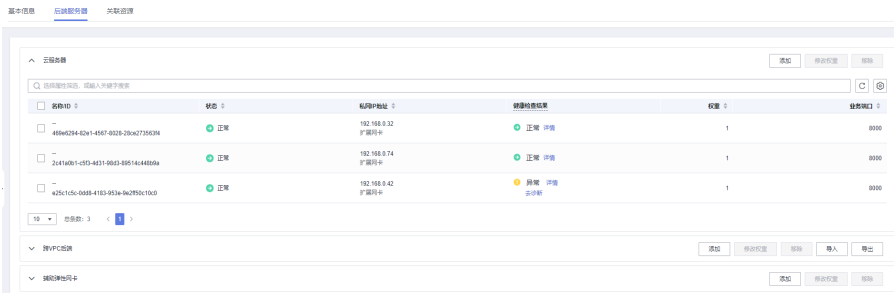
#### 须知

部分Region因为独享型ELB规格没有上线，所以不支持负载均衡能力。详情可跳转至ELB控制台查看是否支持独享型ELB。

- 绑定ELB时，需创建TCP网络型且有内网IP的ELB进行绑定。
- 创建ELB时，ELB规格需要用户按本身业务访问流量评估，建议选取最大规格。DWS侧仅是绑定关联ELB，并不改变ELB规格。
- 创建ELB时，仅需创建ELB，无需创建ELB服务的监听器与后端服务器组，DWS会自动创建所需要的ELB监听器与后端服务器组。
- 创建ELB时，不能存在与数据库相同端口的监听器，否则会导致ELB绑定失败。
- 绑定ELB时，DWS侧默认为ELB服务配置“ROUND\_ROBIN”转发策略，并设置10秒的健康检查间隔，50秒超时时间以及3次重试次数。用户如果需要修改此ELB默认参数时，请充分评估影响。
- 解绑ELB时，DWS侧会清除ELB中集群相关信息，但并不会删除用户ELB。
- 如需要公网IP或者域名访问ELB集群时，请通过ELB服务管理页面进行EIP绑定或者域名操作。
- 如果集群为IPv4集群，则仅支持绑定IPv4的ELB；如果集群为IPv6双栈集群，则仅支持绑定IPv6双栈的ELB。暂不支持交叉场景的ELB绑定。

绑定 ELB

- 步骤1 登录DWS管理控制台。
- 步骤2 单击“专属集群 > 集群列表”。默认显示用户所有的集群列表。
- 步骤3 在集群列表中，单击指定集群名称进入“集群详情”页面。
- 步骤4 单击“绑定弹性负载均衡”，选择需要绑定到此集群的ELB，若不存在ELB，则可在ELB服务页面创建完成后，在DWS侧刷新进行重新绑定即可。
- 步骤5 绑定命令下发成功后，返回集群管理页面，集群列表将显示“弹性负载均衡绑定中”任务信息，绑定需要一定时间，请耐心等待。
- 步骤6 进入弹性负载均衡服务控制台，选择“弹性负载均衡>后端服务器组”，单击绑定的ELB名称，切换到“后端服务器组”页签，检查集群CN节点是否被正确绑定到ELB中。



- 步骤7 绑定成功后，进入“集群详情”页面中的“连接信息”区域，可以查看ELB提供的IP地址，后续连接DWS集群使用此地址。

----结束

解绑 ELB

- 步骤1 登录DWS管理控制台。
- 步骤2 单击“专属集群 > 集群列表”。默认显示用户所有的集群列表。
- 步骤3 在集群列表中，单击指定集群名称进入“集群详情”页面。
- 步骤4 单击“解绑弹性负载均衡”，进行ELB解绑操作。
- 步骤5 解绑命令下发成功后，返回集群管理页面，集群列表显示“弹性负载均衡服务解绑中”任务信息，解绑需要一定时间，请耐心等待。
- 步骤6 进入负载均衡服务控制台，单击解绑的ELB名称，切换到“后端服务器”页签，检查集群CN节点是否已被删除。

----结束

10.4.3 增删 DWS 集群 CN 节点

概述

当用户集群创建后，实际需要的CN数量会随着业务需求而发生变化，因此管理CN节点功能的实现使用户可以根据实际需求动态调整集群CN数量。具体操作如下：

- 增加CN节点

- **删除CN节点**

 **说明**

- 该特性仅8.1.1及以上集群版本支持。
- 在线增删CN以及多CN并行增加仅8.1.3.300及以上集群版本支持（不包含8.2.0版本）。

## 约束与限制

- 资源发放时默认CN数量为3，用户可以根据实际发放节点数量调整CN数量，范围为2~20。
- 增删CN节点过程中不允许执行其他运维操作。
- 增加CN过程会消耗大量的CPU和IO资源，因此会对用户作业性能影响较大，用户应该尽可能在停止业务情况下或业务轻载的情况下执行增加CN操作。
- 增加CN节点时发生故障且回滚失败，可以重试增加CN操作。删除CN节点不支持回滚。
- 增加CN失败只能重试增加CN，其他运维操作不允许执行；删除CN失败，只能重试删除CN，其他运维操作不允许执行。
- 删除CN节点过程中如果有DDL操作，例如创建schema，function等，这些DDL可能会因为找不到被删除的CN节点而报错，业务重试即可成功。
- 删除CN节点过程中，如果只有一个节点异常，只能删除这个异常的节点；如果有两个及以上节点异常，不能删除CN节点，需要先处理恢复异常的节点后，才能删除CN节点。

## 增加 CN 节点

**步骤1** 登录DWS管理控制台。

**步骤2** 在“专属集群 > 集群列表”页面，找到需要增加CN节点的集群。

**步骤3** 在指定集群所在行的“操作”列，选择“更多>管理CN节点>增加CN节点”。

**步骤4** 在跳出页面根据用户需求选择是否在指定节点上添加CN。

- 选择“否”，则可以选择调整后的CN数量。
- 选择“是”，选择在指定节点。

---

**须知**

- 进行增加CN节点操作需要集群处于“可用”或者“不均衡”任意一种状态。
- 调整后数量，不能超过节点数。
- 新增CN数量，不能超过已部署CN数。

---

**步骤5** 确认无误后，单击“确定”。

----结束

## 删除 CN 节点

- 步骤1 登录DWS管理控制台。
- 步骤2 在“专属集群 > 集群列表”页面，找到需要删除CN节点的集群。
- 步骤3 在指定集群所在行的“操作”列，选择“更多>管理CN节点>删除CN节点”。
- 步骤4 在跳出页面根据用户需求勾选需要删除的CN节点，确认无误后，输入“DELETE”或单击“一键输入”，并单击“确定”，删除此CN节点。

### 须知

- 删除CN节点时需至少保留两个CN节点。
- 多AZ集群删除CN时需要在每个可用分区保留一个正常的CN节点，若节点故障则可删除。
- 删除CN节点时集群需要处于“可用”、“节点故障”或者“非均衡”任意一种状态。
- 如果CN节点已经绑定弹性IP，则禁止删除。
- 如果存在异常节点，则只允许删除异常状态的CN节点。
  - 存在一个节点故障，则只允许删除该CN节点。
  - 存在两个或两个以上节点故障，则不允许删除CN节点。

----结束

## 10.4.4 管理 DWS 数据库连接

### 操作场景

数据库默认支持一定数量的连接，管理员用户可以通过管理数据库的连接，了解当前数据库的连接性能或增加连接限制，使更多用户或应用程序可以同时连接到数据库。

### 支持的连接数规格

集群支持的连接数与集群节点规格有关：

表 10-13 支持连接数规格

| 参数                        | 参数描述                    | CN连接数 | DN连接数                            |
|---------------------------|-------------------------|-------|----------------------------------|
| max_connections           | 允许和数据库连接的最大并发连接数。       | 800   | max(VCPU核数/单节点DN数量*120+24, 5000) |
| max_pool_size             | CN的连接池与其它某个CN/DN的最大连接数。 |       |                                  |
| max_prepared_transactions | 设置可以同时处于预备状态的事务的最大数。    |       |                                  |

查看最大连接数

- 方式一：集群创建成功后，用户可在DWS管理控制台上单击指定集群名称，切换至“参数修改”模块查看数据库参数max\_connections的取值。
- 方式二：使用SQL客户端工具连接集群中的数据库后，通过SQL命令的方式查看数据库参数max\_connections的取值。  
SHOW max\_connections;

界面显示的结果与以下信息类似，表示数据库默认支持的最大连接数为200。

```
max_connections

200
(1 row)
```

查看已使用的连接数

- 步骤1 使用SQL客户端工具连接集群中的数据库。
- 步骤2 支持查看如表10-14所示的连接数场景。

须知

除了创建的时候用双引号引起的数据库和用户名称外，以下命令中用到的数据库名称和用户名称，其中包含的英文字母必须使用小写。

表 10-14 查看连接数介绍

| 描述               | 命令                                                                                                                                                                                                                                                          |
|------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 查看指定用户的会话连接数上限。  | <div>执行如下命令查看连接到指定用户dbadmin的会话连接数上限。</div> <div>SELECT ROLNAME,ROLCONNLIMIT FROM PG_ROLES WHERE ROLNAME='dbadmin';</div> <div>查询结果类似如下信息，其中“-1”表示没有对用户dbadmin设置连接数的限制。</div> <div>rolname   rolconnlmit<br/>-----+-----<br/>dwsadmin   -1<br/>(1 row)</div> |
| 查看指定用户已使用的会话连接数。 | <div>执行如下命令查看指定用户dbadmin已使用的会话连接数。</div> <div>SELECT COUNT(*) FROM V\$SESSION WHERE USERNAME='dbadmin';</div> <div>查询结果类似如下信息，其中，“1”表示dbadmin已使用的会话连接数。</div> <div>count<br/>-----<br/>1<br/>(1 row)</div>                                                  |

| 描述                | 命令                                                                                                                                                                                                                                                    |
|-------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 查看指定数据库的会话连接数上限。  | <p>执行如下命令查看连接到指定数据库gaussdb的会话连接数上限。</p> <pre>SELECT DATNAME,DATCONNLIMIT FROM PG_DATABASE WHERE DATNAME='gaussdb';</pre> <p>查询结果类似如下信息，其中“-1”表示没有对数据库gaussdb设置连接数的限制。</p> <pre>datname   datconnlimit -----+----- gaussdb            -1 (1 row)</pre> |
| 查看指定数据库已使用的会话连接数。 | <p>执行如下命令查看指定数据库gaussdb上已使用的会话连接数。</p> <pre>SELECT COUNT(*) FROM PG_STAT_ACTIVITY WHERE DATNAME='gaussdb';</pre> <p>查询结果类似如下信息，其中，“1”表示数据库gaussdb上已使用的会话连接数。</p> <pre>count ----- 1 (1 row)</pre>                                                     |
| 查看所有用户已使用会话连接数。   | <p>执行如下命令查看所有用户已使用的会话连接数。</p> <pre>SELECT COUNT(*) FROM PG_STAT_ACTIVITY;</pre> <pre>count ----- 10 (1 row)</pre>                                                                                                                                     |

----结束

## 10.5 DWS 资源负载管理

### 10.5.1 概述

数据库的系统资源（CPU资源、内存资源、IO资源和存储资源）是有限的，DWS在同时运行多种类型的业务（如数据加载、批量分析、实时查询等）时，各类型业务之间可能会竞争资源，从而出现资源性能瓶颈，导致吞吐量下降，造成整体的查询性能低下。那么对系统的资源进行合理的分配，能避免产生资源不合理利用而导致的系统运行效率下降等问题。

DWS提供了资源管理功能，用户可以根据自身业务将资源按需划分成不同的资源池，不同资源池之间资源互相隔离。再通过关联数据库用户将其关联至不同的资源池，用户SQL查询时将根据“用户-资源池”的关联关系将查询转至资源池中执行。通过指定资源池上可并行运行的查询数、单查询内存上限以及资源池可使用的内存和CPU资源，从而实现对不同业务之间的资源限制和隔离，满足数据库混合负载需求，保证执行查询时有充足的系统资源，确保查询性能。当集群转换为逻辑集群后，可以选择指定的逻辑集群进行同样的资源池的创建、修改、删除。

**须知**

- 该特性仅8.0.0及以上集群版本支持。
- 离线扩容过程中不支持资源管理功能，如开启了资源管理计划，请在进行离线扩容操作之前停止。

## 打开或关闭资源管理功能

资源管理配置包括资源管理开关、全局最大并发数。这里的全局最大并发数指的是单个CN上的最大并发数，如果您通过资源管理开关关闭了资源管理功能，那么所有的资源管理功能将不可用。

## 资源管理功能

DWS的资源管理根据系统资源管控类型可分为：

- 计算资源管理：主要由资源池功能实现，对计算资源进行隔离和限制，防止异常SQL查询导致集群级异常，包括：并发管理、内存管理、CPU管理以及异常规则。详情请参见[17.2-资源池](#)。
- 存储空间管理：从用户和schema两方面实现对存储空间的管理，防止磁盘满和数据库只读，详细请参见[空间管理](#)。
- 资源管理计划：按计划进行资源管理自动配置，应对复杂多变的负载场景，实现更为灵活的资源管理，详细请参见[17.3.3-导入导出资源管理计划](#)。

DWS的资源管理根据管控时机可分为：

- 运行前管理  
查询运行前，对查询进行运行前管理，如果资源充足则查询可以运行，否则查询需要排队，等待其他查询释放资源后才能运行。运行前管理包含**并发管理**和运行前**内存管理**。
- 运行中管理  
查询执行过程中，对查询使用资源进行管控，防止低质量SQL导致集群异常。运行中管理包含运行中**内存管理**、**CPU管理**、**空间管理**和**异常规则**。

## 简单查询和复杂查询

DWS提供了精细化的资源管理功能，在查询进行资源管理前，根据查询预期执行时间和资源消耗，将查询划分为执行时间长、资源消耗多的复杂查询和执行时间短、资源消耗少的简单查询。简单查询和复杂查询的划分和资源消耗相关，因此根据估算内存对查询进行划分：

- 简单查询：估算内存小于32MB。
- 复杂查询：估算内存大于等于32MB。

混合负载场景下，复杂查询可能会长时间占用大量资源，虽然简单查询执行时间短、消耗资源少，但是因为资源耗尽，简单查询不得不在资源池中等待复杂查询执行完成。为提升执行效率、提高系统吞吐量，DWS的“短查询加速”功能，实现对简单查询的单独管理。

- 开启短查询加速后，简单查询与复杂查询分开管理，简单查询无需与复杂查询竞争资源。

- 关闭短查询加速后，简单查询与复杂查询执行相同的资源管理操作。

虽然单个简单作业资源消耗少，但是大量简单作业并发运行还是会占用大量资源，因此短查询加速开启情况下，需要对简单查询进行并发管理；资源管理可能会影响查询性能，影响系统吞吐量，因此简单查询不进行资源管理，异常规则也不生效。

#### 说明

基于估算内存的查询划分一方面依赖估算内存的准确性，另一方面查询执行时间和CPU消耗可能与内存消耗不成正比，因此对于性能不敏感、业务明确的资源池可以通过关闭短查询加速实现对简单作业的资源管理和异常处理。

## 10.5.2 资源池

### 10.5.2.1 功能介绍

DWS资源池的具体功能包括：并发管理、内存管理、CPU管理以及异常规则。

#### 并发管理

并发，即资源池中的最大查询并发数。并发管理作为运行前管理，用于限制查询并发运行的数量，通过限制查询并发数降低资源争抢，保证资源的有序高效利用。

在资源池页面“短查询配置”一栏，您可以通过开关决定是否开启短查询加速功能。如果需要对简单语句并发数（默认值为-1，0 或-1表示不控制）进行修改，可选择打开短查询加速。

并发管理规则如下：

- 短查询加速开启，复杂查询受资源池并发控制，简单查询受短查询并发控制。
- 短查询加速关闭，复杂查询和简单查询均受资源池并发控制，短查询并发控制无效。

#### 内存管理

内存资源，即资源池所占用的内存百分比。

内存管理的目的：防止数据库系统占用内存过高导致内存溢出（OOM）和实现资源池之间的内存隔离和限制。为满足这两个目的，资源管理从以下两方面进行内存管理：

- 全局内存管理

为防止数据库系统使用内存过大导致OOM，设置数据库系统全局内存上限（max\_process\_memory），对数据库全局内存进行管理。全局内存管理包含运行前管理和运行中管理，运行中管理防止实际使用内存超限，运行前管理防止查询执行过程中报错，具体如下：

- 运行前管理：

一方面慢车道运行的所有查询估算内存都会进行统计，另一方面数据库系统实际使用内存会进行反馈，当实际使用内存大于统计内存时，对统计内存进行调整。查询运行前，判断全局剩余内存能否满足查询运行，满足情况下查询可以直接运行，否则查询需要排队，等待其他查询释放资源后运行。

- 运行中管理：

查询执行过程中实际使用的内存也会进行统计，查询在申请内存时判断内存使用是否超限，内存超限查询报错，已用内存释放。

- 资源池内存管理

资源池内存管理属于专属限额的管理方式，即资源池分配多少内存就只能使用多少内存，空闲出来的内存其他资源池不能使用。

资源池内存分配采用百分比方式，取值范围0~100。0表示资源池不进行内存管理，100表示资源池进行内存管理且可使用全局所有内存。

所有资源池分配的内存百分比之和不能超过100。资源池内存管理仅管控慢车道查询，且只包含运行前管理，处理逻辑与全局内存运行前管理类似。资源池慢车道查询运行前，进行估算内存统计，当统计内存大于资源池内存时，查询需要排队，等待资源池内其他查询运行结束释放资源后才能运行。

## CPU 管理

目前支持“共享配额”和“专属限额”两种CPU管理方式：

- CPU共享配额：在CPU系统繁忙的情况下，不同资源池按照共享配额配置分配CPU资源；在CPU系统空闲的情况下，该配置不生效。
- CPU专属限额：限制资源池使用的CPU核数上限，无论CPU系统繁忙或者空闲，该资源池上的作业都无法突破该限额配置。

在资源池页面“资源配置”一栏，您可以修改当前资源池的CPU共享配额和专属限额。

共享配额和专属限额的管理方式各有优劣，共享配额的管理方式可以实现CPU资源的充分利用，但是资源池之间隔离不彻底，可能影响查询性能；专属限额的管理方式可以实现CPU资源的绝对隔离，但是在资源池CPU资源空闲时，会造成资源的浪费。

### 说明

CPU专属限额仅8.1.3及以上集群版本支持。

## 异常规则

异常规则用于异常查询的识别和快速处理，防止“低质量SQL”长时间占用大量资源，导致其他查询阻塞或性能下降。

在资源池页面“关联异常规则”一栏，可浏览当前资源池中已绑定的异常规则，绑定新异常规则，解绑已有异常规则。目前支持的异常处理规则如[表1 异常规则参数](#)所示。

### 说明

- 8.2.1及以上集群版本支持降级异常规则，所有异常规则都支持降级行为，降级后仅网络资源抢占降至低优先级，即在正常查询无网络请求时，才会调度降级查询的网络请求。
- 关联、解绑异常规则仅8.2.0及以上集群版本支持。

表 10-15 异常规则参数

| 参数项            | 描述                                                                                                       | 取值范围（0表示不约束）            | 操作        |
|----------------|----------------------------------------------------------------------------------------------------------|-------------------------|-----------|
| 阻塞时间           | 作业的阻塞时间，包括全局并发排队以及局部并发排队的总时间，单位秒。<br>例如，如果配置“阻塞时间”为300秒，那么当该资源池中的用户执行的某个作业在阻塞300秒后将被终止。                  | 1~2147483647的整数。0表示不约束。 | 终止、降级或不约束 |
| 执行所消耗时间        | 已经执行的作业从开始执行到当前所消耗的时间，单位为秒。<br>例如，如果配置“执行所消耗时间”为100秒，那么当该资源池中的用户执行的某个作业在执行超过100秒后将被终止。                   | 1~2147483647的整数。0表示不约束。 | 终止、降级或不约束 |
| 所有DN上CPU总时间    | 作业在所有DN上执行时所耗费的CPU总时间，单位为秒。                                                                              | 1~2147483647的整数。0表示不约束。 | 终止、降级或不约束 |
| 检查倾斜率的时间间隔     | 检查作业执行的CPU倾斜率的时间间隔，单位为秒，需同“所有DN上CPU时间的倾斜率”一起设置。                                                          | 1~2147483647的整数。0表示不约束。 | 终止、降级或不约束 |
| 所有DN上CPU总时间倾斜率 | 作业在DN上执行时的CPU时间的倾斜率，依赖于“检查倾斜率的时间间隔”的设置。                                                                  | 1~100的整数。0表示不约束。        | 终止、降级或不约束 |
| 单DN算子下盘大小      | 作业在单个DN上最大下盘的数据量，单位MB。<br><b>说明</b><br>该异常规则仅8.2.0及以上集群版本支持。                                             | 1~2147483647的整数。0表示不约束。 | 终止、降级或不约束 |
| DN平均消耗CPU占比    | 作业在所有DN上执行时的平均CPU使用率，检测周期不强依赖“检查倾斜率的时间间隔”，若配置将使用该检查间隔，否则系统默认30秒间隔。<br><b>说明</b><br>该异常规则仅8.2.0及以上集群版本支持。 | 1~100的整数。0表示不约束。        | 终止、降级或不约束 |
| 单个DN上最大带宽      | 作业在单个DN上最大可占用的网络带宽，单位MB。<br><b>说明</b><br>该异常规则仅8.2.1及以上集群版本支持。                                           | 1~2147483647的整数。0表示不约束。 | 终止、降级或不约束 |

### 10.5.2.2 添加资源池

**步骤1** 登录DWS管理控制台。

**步骤2** 在集群列表中单击需要访问“资源管理”页面的集群名称。

**步骤3** 切换至“资源管理”页签。

**步骤4** 单击资源池列旁的“添加资源池”按钮添加资源池。

#### 说明

最多可创建63个资源池。

**步骤5** 参见表10-16填写资源池的名称和相关资源配置。

表 10-16 资源池参数

| 参数项          | 描述                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | 默认值     |
|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------|
| 名称           | 资源池的名称。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | -       |
| CPU资源<br>(%) | <ul style="list-style-type: none"><li>共享配额：关联在当前资源池的用户在执行作业时可以使用CPU时间比例，取值范围为1-99的整数。</li><li>专属限额：限定资源池中数据库用户在执行作业时使用的最大CPU核数占总核数的百分比，取值范围为0-100的整数，0表示不限制。</li></ul> <b>说明</b> <ul style="list-style-type: none"><li>所有资源池的总和不能超过99%。当配置CPU共享配额后，如果当前只有一个资源池时，该参数不生效。</li><li>共享配额非绝对限制，只有在发生CPU竞争时才生效。例如，资源池A和B被绑定在CPU1运行，当A和B均运行时参数生效，只有A运行则参数不生效。</li><li>所有资源池的CPU专属限额总和最大不能超过100%，系统默认的CPU专属限额(%)为0。</li><li>CPU专属限额仅8.1.3及以上集群版本支持。</li><li>用户在CPU低核数场景下，不建议分配多个专属资源池。因为CPU专属限额分配比最终实际分配核心最小为1，且在CPU核数量较少的情况下，如果先建立的资源池已占用完CPU剩余数量，后建立的资源池会共用上一个资源池的CPU，可能导致CPU配比与实际不符。<br/>举例说明：集群的CPU核为2，建立3个资源池，CPU分配比为15%、25%、60%。其中第一个物理核心分配为1（最小值），第二个分配为1（最小值），第三个分配为1（CPU核已无剩余，共用上一个CPU）。</li></ul> | -       |
| 内存资源<br>(%)  | 资源池所占用的内存百分比。<br><b>内存和查询并发支持单独管控和联合管控，联合管控时必须同时满足并发和内存要求时作业才能下发。</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | 0（不限制）  |
| 存储资源<br>(MB) | 可使用的永久表空间大小。<br><b>该值是资源池下所有DN的表空间总值，单DN节点可用空间=设置值 / DN节点数。</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | -1（不限制） |

| 参数项    | 描述                                                                            | 默认值         |
|--------|-------------------------------------------------------------------------------|-------------|
| 复杂语句并发 | 资源池中的最大查询并发数。<br>内存和查询并发支持单独管控和联合管控，联合管控时必须同时满足并发和内存要求时作业才能下发。                | 10          |
| 网络带宽权重 | 网络调度时权重值。取值范围为1~2147483647的整数，默认配置为-1。<br><b>注意</b><br>网络带宽权重仅8.2.1及以上集群版本支持。 | -1<br>(不限制) |

**步骤6** 核对信息后，单击“确定”，添加资源池完成。

----结束

10.5.2.3 修改资源池

在资源管理中，您可以对某个资源池的参数进行修改。

- 步骤1** 登录DWS管理控制台。
- 步骤2** 在集群列表中单击需要访问“资源管理”页面的集群名称。
- 步骤3** 切换至“资源管理”页签。
- 步骤4** 在“资源池”列表中切换需要修改的资源池名称，包括“短查询配置”、“资源配置”、“关联异常规则”、“关联用户”。
- 步骤5** 修改短查询配置。修改为相应取值，单击右侧“保存”。

| 参数项    | 描述                                    | 取值 |
|--------|---------------------------------------|----|
| 短查询加速  | 短查询加速开关，默认打开。                         | 打开 |
| 简单语句并发 | 短查询作业为执行估算内存小于32MB的查询作业，默认值“-1”表示不管控。 | 10 |

- 步骤6** 修改资源配置。
1. 单击右侧“编辑”，详情请参见[表10-17](#)修改相应参数。

表 10-17 资源池参数

| 参数项 | 描述      | 默认值 |
|-----|---------|-----|
| 名称  | 资源池的名称。 | -   |

| 参数项       | 描述                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | 默认值         |
|-----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| CPU资源 (%) | <ul style="list-style-type: none"> <li>- 共享配额：关联在当前资源池的用户在执行作业时可以使用CPU时间比例，取值范围为1-99的整数。</li> <li>- 专属限额：限定资源池中数据库用户在执行作业时使用的最大CPU核数占总核数的百分比，取值范围为0-100的整数，0表示不限制。</li> </ul> <p><b>说明</b></p> <ul style="list-style-type: none"> <li>- 所有资源池的总和不能超过99%。当配置CPU共享配额后，如果当前只有一个资源池时，该参数不生效。</li> <li>- 共享配额非绝对限制，只有在发生CPU竞争时才生效。例如，资源池A和B被绑定在CPU1运行，当A和B均运行时参数生效，只有A运行则参数不生效。</li> <li>- 所有资源池的CPU专属限额总和最大不能超过100%，系统默认的CPU专属限额（%）为0。</li> <li>- CPU专属限额仅8.1.3及以上集群版本支持。</li> <li>- 用户在CPU低核数场景下，不建议分配多个专属资源池。因为CPU专属限额分配比最终实际分配核心最小为1，且在CPU核数量较少的情况下，如果先建立的资源池已占用完CPU剩余数量，后建立的资源池会共用上一个资源池的CPU，可能导致CPU配比与实际不符。</li> </ul> <p>举例说明：集群的CPU核为2，建立3个资源池，CPU分配比为15%、25%、60%。其中第一个物理核心分配为1（最小值），第二个分配为1（最小值），第三个分配为1（CPU核已无剩余，共用上一个CPU）。</p> | -           |
| 内存资源 (%)  | <p>资源池所占用的内存百分比。</p> <p><b>内存和查询并发支持单独管控和联合管控，联合管控时必须同时满足并发和内存要求时作业才能下发。</b></p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | 0<br>(不限制)  |
| 存储资源 (MB) | <p>可使用的永久表空间大小。</p> <p><b>该值是资源池下所有DN的表空间总值，单DN节点可用空间=设置值 / DN节点数。</b></p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           | -1<br>(不限制) |
| 复杂语句并发    | <p>资源池中的最大查询并发数。</p> <p><b>内存和查询并发支持单独管控和联合管控，联合管控时必须同时满足并发和内存要求时作业才能下发。</b></p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | 10          |
| 网络带宽权重    | <p>网络调度时权重值。取值范围为1~2147483647的整数，默认配置为-1。</p> <p><b>注意</b><br/>网络带宽权重仅8.2.1及以上集群版本支持。</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            | -1<br>(不限制) |

## 📖 说明

CPU专属限额仅8.1.3及以上集群版本支持。

2. 单击“确定”。

**步骤7 关联异常规则。**

1. 单击左侧“关联异常规则”按钮。
2. 从当前异常规则列表中，勾选需要关联的异常规则，一次可勾选多个。
3. 单击“确定”。
4. 如果需要解绑异常规则，则单击“解除关联”按钮即可。

**说明**

- 关联、解绑异常规则仅8.2.0及以上集群版本支持。
- 默认异常规则，当用户未关联资源池，或者用户所关联的资源池未配置异常规则时默认生效；当用户所关联的资源池关联所指定规则时，以所关联规则为准。
  - 默认异常规则仅8.2.0及以上集群版本支持，升级到8.2.0及以上集群版本时默认异常规则不会生效，用户可自行创建所需规则。
  - 8.2.1集群版本支持降级异常规则，所有异常规则都支持降级行为，降级后仅网络资源抢占降至低优先级：在正常查询无网络请求时，才会调度降级查询的网络请求。
  - 同一个资源池最多关联16组异常规则。
- 同一个资源池可以关联多组规则，资源池关联的多组不同的异常规则以“或”的关系生效，满足其中一组的所有条件即可生效。例如，资源池关联两组规则，一组指定elapsedtime=2400，另一组指定elapsedtime=1200，memsize=2000，那么执行中的作业，满足执行时间达到1200秒且内存使用达到2000MB时，或者执行时间达到2400秒时，都会触发规则终止作业。

**步骤8 关联用户。**

1. 单击左侧“关联用户”。
2. 从当前用户列表中，勾选需要关联的用户，一次可勾选多个。
3. 单击“确定”。
4. 如果需要解除关联用户，则单击“解除关联”按钮即可。

**说明**

- 一个数据库用户只有被添加到某个资源池中之后，该用户运行作业所使用的资源才能被管控。
- 一个数据库用户只能被添加至一个资源池中，从资源池中移除的用户可以再次添加至其他资源池。
- 用户绑定列表里锁状态有“未锁定”、“锁定”和“未知”三种状态（在8.5.0.100之前的管控面版本未区分用户锁状态，仅显示为“未知”）。已锁定的用户无法勾选关联，勾选按钮置灰；“未知”状态用户可以被勾选，但是否绑定成功取决于用户实际的锁定状态。
- 数据库管理员用户不可关联。
- 当用户没有指定关联资源池时，会被默认关联到default\_pool，资源使用受default\_pool限制。default\_pool在开启资源管理功能后由系统自动创建。

---结束

## 10.5.2.4 删除资源池

**步骤1** 登录DWS管理控制台。

**步骤2** 在集群列表中单击需要访问“资源管理”页面的集群名称。

**步骤3** 切换至“资源管理”页签。

**步骤4** 在左侧“资源池”中单击需要删除的资源池名称。

**步骤5** 单击右侧的“删除资源池”。

#### 说明

删除资源池时如果资源池中有关联的数据库用户，则资源池不支持删除，请解除关联后重试。

----结束

## 10.5.3 资源管理计划

### 10.5.3.1 资源管理计划操作

#### 资源管理计划简介

资源管理计划功能是DWS为客户提供的资源管理高级特性，客户可以通过创建资源管理计划，向计划中添加多个时间阶段，并为每个阶段配置不同的资源配比。当启动计划时，计划就会按照配置在不同的时间段自动切换资源配置。如果客户在不同的时间段运行不同的业务，而这些业务又会占用不同比重的资源，使用资源管理计划功能可以很好的帮助客户实现不同时间段的资源配置自动切换。

#### 添加资源管理计划

**步骤1** 登录DWS管理控制台。

**步骤2** 在集群列表中单击需要访问“资源管理”页面的集群名称。

**步骤3** 进入“基本信息”页面，左导航栏单击“资源管理”。

**步骤4** 切换至“资源管理计划”页签，并单击页面右侧的“添加”按钮添加计划。

**步骤5** 填写资源管理计划的计划名称，并单击“确认”按钮。

#### 须知

- 创建资源管理计划之前必须提前规划并创建好资源池，请参见[17.2.3-添加资源池](#)。
- 最多支持创建10个资源管理计划。

----结束

#### 启动/停止资源管理计划

**步骤1** 登录DWS管理控制台。

**步骤2** 在集群列表中单击需要访问“资源管理”页面的集群名称。

**步骤3** 进入“基本信息”页面，左导航栏单击“资源管理”。

**步骤4** 切换至“资源管理计划”页签，单击“启动/停止”按钮。

**步骤5** 确认无误后，在弹窗中单击“确定”按钮启动/停止计划。

#### 须知

- 每个集群只能启动一个计划。
- 一个计划至少要有两个阶段才能启动。

----结束

### 查看资源管理计划执行日志

**步骤1** 登录DWS管理控制台。

**步骤2** 在集群列表中单击需要访问“资源管理”页面的集群名称。

**步骤3** 进入“基本信息”页面，左导航栏单击“资源管理”。

**步骤4** 切换至“资源管理计划”页签，在计划执行日志区域单击“查看”按钮查看计划阶段的执行日志。

----结束

### 删除资源管理计划

**步骤1** 登录DWS管理控制台。

**步骤2** 在集群列表中单击需要访问“资源管理”页面的集群名称。

**步骤3** 进入“基本信息”页面，左导航栏单击“资源管理”。

**步骤4** 切换至“资源管理计划”页签，单击“删除”按钮删除当前资源管理计划。

#### 须知

正在运行的资源管理计划不支持删除。

----结束

### 10.5.3.2 资源管理计划阶段

#### 前提条件

添加和修改资源管理计划阶段需满足以下条件：

- 所有资源池的CPU共享配额总和不超过99%。
- 所有资源池的CPU专属限额总和不超过100%。

#### 说明

- CPU限额仅8.1.3及以上集群版本支持。

### 添加资源管理计划阶段

**步骤1** 登录DWS管理控制台。

- 步骤2** 在集群列表中单击需要访问“资源管理”页面的集群名称。
- 步骤3** 进入“基本信息”页面，左导航栏单击“资源管理”。
- 步骤4** 切换至“资源管理计划”页签，单击计划阶段区域的“添加”按钮，在添加阶段页面填写阶段名称并配置资源信息，确认成功后单击“确定”按钮。

#### 须知

- 添加阶段时必须停止正在运行的资源管理计划，否则无法添加。
- 每个计划最多支持添加48个阶段。
- 一个计划中所有阶段的切换时间不能相同。
- 时间、日期、月份共同组成执行时间表达式，不允许配置不存在的日期表达式，例如2.30。

----结束

## 修改资源管理计划阶段

- 步骤1** 登录DWS管理控制台。
- 步骤2** 在集群列表中单击需要访问“资源管理”页面的集群名称。
- 步骤3** 进入“基本信息”页面，左导航栏单击“资源管理”。
- 步骤4** 切换至“资源管理计划”页签，单击计划阶段操作列的修改按钮。
- 步骤5** 在修改计划阶段页面可修改阶段的切换时间，资源配置等信息。

#### 说明

网络带宽权重仅8.2.1及以上集群版本支持。

----结束

## 手动切换资源管理计划阶段

运行中的计划如果需要提前切换至某个阶段，可以通过手动切换功能切换至目标阶段。

- 步骤1** 登录DWS管理控制台。
- 步骤2** 在集群列表中单击需要访问“资源管理”页面的集群名称。
- 步骤3** 进入“基本信息”页面，左导航栏单击“资源管理”。
- 步骤4** 切换至“资源管理计划”页签，单击计划概览处的“切换”按钮，选择要切换的目标阶段。



----结束

## 导入/导出资源管理计划阶段

- 步骤1** 登录DWS管理控制台。
- 步骤2** 在集群列表中单击需要访问“资源管理”页面的集群名称。
- 步骤3** 进入“基本信息”页面，左导航栏单击“资源管理”。
- 步骤4** 切换至“资源管理计划”页签，在计划阶段模块单击“导入/导出”按钮导入/导出资源管理计划阶段。

### 须知

- 正在运行的资源管理计划不支持导入。
- 导入时需要先创建资源池。

----结束

## 删除资源管理计划阶段

- 步骤1** 登录DWS管理控制台。
- 步骤2** 在集群列表中单击需要访问“资源管理”页面的集群名称。
- 步骤3** 进入“基本信息”页面，左导航栏单击“资源管理”。
- 步骤4** 切换至“资源管理计划”页签，单击计划阶段操作列的“删除”按钮。

----结束

### 须知

删除阶段时必须停止资源管理计划，否则无法删除。

## 10.5.4 空间管理

### 空间管理简介

存储资源无节制的使用可能导致磁盘满，进而导致集群异常、业务中断。磁盘满问题具有业务恢复难度大、恢复时间长的特点，通过引入数据库只读，极大概率降低了磁盘满问题的发生，但是数据库只读同样会导致业务中断，影响业务连续性。为解决数据库只读问题，DWS提供了多维度的存储资源管理能力，一方面在schema维度实现了schema空间管理，用于限制schema使用的永久空间大小；一方面在用户维度实现了永久空间、临时空间和算子空间管理，防止单用户业务异常导致数据库只读。

- schema维度：schema空间管理模块可查询集群下数据库和模式空间信息，并支持修改模式空间总值。
- 用户维度：用户空间管理用于限定不同用户可以使用的空间限额，防止用户使用存储空间过大导致业务执行受阻。DWS通过在创建用户时指定空间大小的方式实现对存储资源的管理，支持管理的存储空间类型包括：
  - 永久表存储空间（PREM SPACE）  
用于限制用户创建的永久表(非临时表)占用的空间限额。
  - 临时表存储空间（TEMP SPACE）  
用于限制用户创建的临时表占用的空间限额。
  - 算子落盘空间（SPILL SPACE）  
查询执行过程中，如果实际使用内存大于估算内存，则查询可能产生落盘，将这种查询执行过程中落盘占用的存储空间称为算子落盘空间。用户算子落盘空间管理用于限制用户查询执行过程中算子落盘占用的空间限额。

#### 说明

- 该特性仅8.1.1及以上集群版本支持。
- DWS管理面目前仅支持模式空间管理。

### 操作步骤

**步骤1** 登录DWS管理控制台。

**步骤2** 在集群列表中单击需要访问“资源管理”页面的集群名称。

**步骤3** 进入“基本信息”页面，左导航栏单击“资源管理”。

**步骤4** 切换至“schema空间管理”模块，切换需要查看的数据库。

**步骤5** 在需要修改空间限额模式的所在行操作列，单击“编辑”按钮，修改合适的空间限额。

**步骤6** 单击“确定”提交。

#### 说明

- 空间限额仅对普通用户有效，数据库系统管理员用户不受限制（因此当显示已用空间等于空间限额时，真实使用空间可能已超出设置的值）。
- 单DN限额=总限额/DN节点数，所以设置值可能与最终显示值存在细微差异。

----结束

## 10.6 修改 DWS 集群 GUC 参数

集群创建成功后，用户可以根据实际需要修改集群的数据库参数。在DWS管理控制台，您可以设置一些常用的数据库参数，详情请参见[修改参数](#)。也可以查看历史参数修改记录，详情请参见[查看参数修改历史](#)。单击“导出”按钮可导出集群相关参数配置。如需查看或设置其他数据库参数，您可以通过SQL命令的方式，详情请参见《数据仓库服务数据库开发指南》中的“配置GUC参数”章节。

### 前提条件

只有当集群无运行中的任务时，才能修改参数。

### 修改参数

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，单击“专属集群 > 集群列表”。

**步骤3** 在集群列表中找到所需要的集群，单击集群名称，进入“集群详情”页面。

**步骤4** 单击“参数修改”页签，并在“参数列表”模块修改相应的参数值，然后单击“保存”。

在参数列表上方单击下拉列表，可根据筛选条件（常用配置、功能控制配置、集群内存配置、集群磁盘配置、集群网络配置、SQL调优配置、SQL兼容性配置、所有配置）查找所需修改的参数，其中选择“常用配置”则显示集群所在区域高频次修改的参数前20条结果，如果没有统计数据，则使用自定义的配置。选择“所有配置”为可筛选配置的全集，即可查看所有参数。参数详情请参见[参数说明](#)。

单击搜索框，可根据“参数名称”和“是否重启集群”查找相应参数。

**步骤5** 在“修改预览”窗口，确认修改无误后，单击“保存”。

**步骤6** 用户可根据修改参数所在行的“是否重启集群”列，判断集群是否进行重启操作。

#### 说明

- 若修改参数无需进行重启集群操作，则参数修改后立即生效。
- 若修改参数需进行重启集群操作，参数修改任务下发成功后页面显示修改后的参数值，待集群重启成功后修改的参数值生效；同时集群状态会显示“待重启”并禁用部分运维操作，重启集群后状态会恢复正常。

----结束

### 查看参数修改历史

由于修改参数的操作可能较频繁，如果多次修改后无法确认哪些参数已经生效，可按如下流程检查参数修改记录。

#### 操作步骤

**步骤1** 登录DWS管理控制台。

- 步骤2 在左侧导航栏中，单击“专属集群 > 集群列表”。
- 步骤3 在集群列表中找到所需要的集群，单击集群名称，进入“集群详情”页面。
- 步骤4 单击“参数修改”页签，并在页面顶部切换到“修改历史”模块。

说明

- 如果修改参数无需进行重启集群操作，则参数修改后立即生效，修改状态为“已同步”。
- 如果修改参数需进行重启集群操作，则修改记录的状态为“需重启生效”，此时单击左侧的下拉展开可以看到具体哪些参数未生效。在重启集群后，该记录的状态会更新为“已同步”。

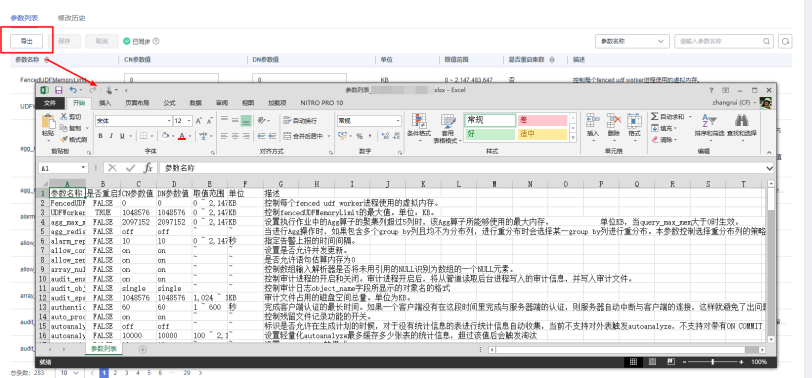
步骤5 该页面默认查询一段时间之内的修改记录，同时也可以在上角的输入框中输入要查询的参数，可以查找该参数所有的修改。

----结束

导出参数列表

- 步骤1 登录DWS管理控制台。
- 步骤2 在左侧导航栏中，单击“专属集群 > 集群列表”。
- 步骤3 在集群列表中找到所需要的集群，单击集群名称，进入“集群详情”页面。
- 步骤4 单击“参数修改”页签，并在页面顶部单击“导出”按钮。可导出集群相关配置参数。

图 10-2 导出参数配置



----结束

## 功能控制配置参数说明

表 10-18 功能控制配置参数说明

| 参数名称                          | 参数描述                                                                                                                                                                                                                                                            | 取值范围                  |
|-------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| audit_enabled                 | 控制审计进程的开启和关闭。审计进程开启后，将从管道读取后台进程写入的审计信息，并写入审计文件。                                                                                                                                                                                                                 | on或者off               |
| audit_space_limit             | 审计文件占用的磁盘空间总量。单位为KB。                                                                                                                                                                                                                                            | 1,024 ~ 1,073,741,824 |
| autoanalyze                   | 标识是否允许在生成计划的时候，对于“统计信息完全缺失”或“修改量达到analyze阈值”的表进行统计信息自动收集，当前不支持对外表触发autoanalyze，不支持对带有“ON COMMIT [DELETE ROWS DROP]”选项的临时表触发autoanalyze，如需收集，需用户手动执行analyze操作。如果在auto analyze某个表的过程中数据库发生异常，当数据库正常运行之后再执行语句有可能仍提示需要收集此表的统计信息。此时需要用户对该表手动执行一次analyze操作，以同步统计信息数据。 | on或者off               |
| autovacuum_max_workers        | 设置能同时运行的自动清理线程的最大数量。其中0表示不会自动进行autovacuum。                                                                                                                                                                                                                      | 0 ~ 128               |
| autovacuum_max_workers_hstore | 设置hstore表automerage的线程数量，该值不能大于autovacuum_max_workers。<br>修改本参数需要同步调整autovacuum_max_workers为原值加上autovacuum_max_workers_hstore的大小。                                                                                                                               | 0 ~ 128               |
| autovacuum_naptime            | 设置两次自动清理操作的时间间隔。单位为秒（s）。                                                                                                                                                                                                                                        | 1 ~ 2,147,483         |
| autovacuum_vacuum_cost_delay  | 设置在自动VACUUM操作里使用的开销延迟数值。                                                                                                                                                                                                                                        | -1 ~ 100              |
| behavior_compat_options       | 数据库兼容性行为配置项，该参数的值由若干个配置项用逗号隔开构成。此参数选项中strict_concat_functions和strict_text_concat_td不能同时设置。                                                                                                                                                                      | -                     |
| checkpoint_segments           | 设置周期内所保留的最少WAL日志段文件数量。每个日志文件大小为16MB。                                                                                                                                                                                                                            | 1 ~ 2,147,483,646     |

| 参数名称                          | 参数描述                                                                                                                               | 取值范围               |
|-------------------------------|------------------------------------------------------------------------------------------------------------------------------------|--------------------|
| ddl_lock_timeout              | 通过该参数单独指定阻塞DDL语句锁等待的时间，当申请的锁等待时间超过设定值时，系统会报错。                                                                                      | 0 ~ 2,147,483,647  |
| enable_resource_record        | 是否开启资源记录功能。                                                                                                                        | on或者off            |
| enable_resource_track         | 是否开启资源监控功能。                                                                                                                        | on或者off            |
| enable_track_record_subsql    | 设置是否开启子语句记录归档功能。开启时，存储过程、匿名块内部的子语句会被记录归档到相应的INFO表（GS_WLM_SESSION_INFO）。此参数为会话级参数，可在与CN的连接会话中设置生效，仅影响该会话连接中的语句；也可在CN和DN上同时设置，能全局生效。 | on或者off            |
| enable_user_metric_persistent | 设置是否开启用户历史资源监控转存功能。开启时，对于PG_TOTAL_USER_RESOURCE_INFO视图中数据，会定期采样保存到PG_WLM_USER_RESOURCE_HISTORY系统表中。                                | on或者off            |
| enable_view_update            | 用于设置是否开启视图更新功能。                                                                                                                    | on或者off            |
| extra_float_digits            | 调整浮点值显示的数据位数，浮点类型包括float4、float8 以及几何数据类型。参数值加在标准的数据位数上（FLT_DIG或DBL_DIG中合适的）。                                                      | -15 ~ 3            |
| failed_login_attempts         | 输入密码错误的次数达到该参数所设置的值时，账户将会被自动锁定。配置为0时表示不限制密码输入错误的次数。                                                                                | 0 ~ 1,000          |
| instr_unique_sql_count        | 控制是否收集Unique SQL，以及收集数量限制。                                                                                                         | 0 ~ 2,147,483,647  |
| job_queue_processes           | 表示系统可以并发执行的job数目。该参数为postmaster级别，通过gs_guc设置，需要重启gaussdb才能生效。                                                                      | 0 ~ 1,000          |
| lockwait_timeout              | 控制单个锁的最长等待时间。当申请的锁等待时间超过设定值时，系统会报错。单位为毫秒（ms）。                                                                                      | 0 ~ 2,147,483,647  |
| max_active_statements         | 设置全局的最大并发数量。此参数只应用到CN，且针对一个CN上的执行作业。设置为-1和0表示对最大并发数不做限制。                                                                           | -1 ~ 2,147,483,647 |

| 参数名称                           | 参数描述                                                                                                                                                                                                                                                                                                                                           | 取值范围               |
|--------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------|
| max_files_per_node             | 限制单个节点上单个SQL打开的文件最大数量。                                                                                                                                                                                                                                                                                                                         | -1 ~ 2,147,483,647 |
| max_prepared_transactions      | 设置可以同时处于“预备”状态的事务的最大数目。增加此参数的值会使DWS比系统默认设置需要更多的System V共享内存。                                                                                                                                                                                                                                                                                   | 0 ~ 536,870,911    |
| max_process_memory_auto_adjust | 设置是否开启max_process_memory参数的自动调整功能。                                                                                                                                                                                                                                                                                                             | on或者off            |
| object_mtime_record_mode       | 用于设置PG_OBJECT系统表中mtime字段的更新行为。 <ul style="list-style-type: none"><li>• default: 表示默认行为包括ALTER、COMMENT、GRANT/REVOKE和TRUNCATE操作会更新mtime字段。</li><li>• none: 表示不更新mtime字段。</li><li>• disable_acl: 表示GRANT/REVOKE操作不更新mtime字段。</li><li>• disable_truncate: 表示TRUNCATE操作不更新mtime字段。</li><li>• disable_partition: 表示分区表相关ALTER操作不更新mtime字段。</li></ul> | -                  |
| plog_merge_age                 | 该参数用于控制性能日志数据输出的周期。                                                                                                                                                                                                                                                                                                                            | 0 ~ 2,147,483,647  |
| random_function_version        | 控制analyze在进行数据采样时选取的random函数版本。                                                                                                                                                                                                                                                                                                                | 0 ~ 1              |
| resource_track_cost            | 设置对语句进行资源监控的最小执行代价。值为-1时，不进行资源监控。值大于或等于0时，执行语句的代价大于或者等于10并且超过这个参数值就会进行资源监控。                                                                                                                                                                                                                                                                    | -1 ~ 2,147,483,647 |
| resource_track_duration        | 设置资源监控实时视图中记录的语句执行结束后进行归档的最小执行时间，单位为秒。值为0时，资源监控实时视图中记录的所有语句都进行历史信息归档。值大于0时，资源监控实时视图中记录的语句的执行时间超过所设置的值时进行历史信息归档。                                                                                                                                                                                                                                | 0 ~ 2,147,483,647  |

| 参数名称                    | 参数描述                                                                                                                                                                                                                                                                                                                                                                                              | 取值范围              |
|-------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------|
| resource_track_level    | 设置当前会话的资源监控的等级。该参数只有当参数enable_resource_track为on时才有效。 <ul style="list-style-type: none"><li>• none：不开启资源监控功能。</li><li>• query：开启query级别资源监控功能，开启此功能会把SQL语句的计划信息（类似explain输出信息）记录到top SQL中。</li><li>• perf：开启perf级别资源监控功能，开启此功能会把包含实际执行时间和执行行数的计划信息（类似explain analyze输出信息）记录到top SQL中。</li><li>• operator：开启operator级别资源监控功能，开启此功能不仅会把包含实际执行时间和执行行数的信息记录到top SQL中，还会把算子级别执行信息刷新到top SQL中。</li></ul> | -                 |
| security_enable_options | 控制安全模式下可以解禁的操作。 <ul style="list-style-type: none"><li>• grant_to_public：表示允许安全模式下使用grant to public功能。</li><li>• grant_with_grant_option：表示允许安全模式下使用with grant option功能。</li><li>• foreign_table_options：表示允许安全模式下使用外表操作功能，不需要显式赋予用户useft权限。</li></ul>                                                                                                                                             | -                 |
| session_timeout         | Session闲置超时时间，单位为秒，0表示关闭超时限制。                                                                                                                                                                                                                                                                                                                                                                     | 0 ~ 86,400        |
| space_once_adjust_num   | 空间管控和空间统计功能中，控制慢速构建与细粒度校准操作中每次处理的文件个数阈值。0表示不启用慢速构建和细粒度校准功能。文件个数阈值影响数据库资源，建议合理设置。<br><b>说明</b><br>该参数仅8.1.3及以上集群版本支持。                                                                                                                                                                                                                                                                              | 0 ~ 2,147,483,647 |
| statement_timeout       | 当语句执行时间超过该参数设置的时间（从服务器收到命令时开始计时）时，该语句将会报错并退出执行。单位为毫秒（ms）。                                                                                                                                                                                                                                                                                                                                         | 0 ~ 2,147,483,647 |
| timezone                | 设置显示和解释时间类型数值时使用的时区。                                                                                                                                                                                                                                                                                                                                                                              | -                 |

| 参数名称                       | 参数描述                                                                                                                                                                                                                               | 取值范围      |
|----------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| topsql_retention_time      | 设置历史TopSQL中gs_wlm_session_info和gs_wlm_operator_info表中数据的保存时间，单位为天。设置此GUC参数启用数据保存功能前，请先清理gs_wlm_session_info和gs_wlm_operator_info表中的数据。 <ul style="list-style-type: none"><li>值为0时，表示数据永久保存。</li><li>值大于0时，表示数据能够保存的对应天数。</li></ul> | 0 ~ 3,650 |
| user_metric_retention_time | 设置用户历史资源监控数据的保存天数。该参数仅在enable_user_metric_persistent为on时有效。                                                                                                                                                                        | 0 ~ 3,650 |
| view_independent           | 用于设置是否开启视图与表、函数、同义词的解耦功能。基表恢复后目前已支持自动关联重建。                                                                                                                                                                                         | on或者off   |
| wlm_memory_feedback_adjust | 是否启用动态负载管理的内存负反馈。参数值格式如下所示： <ul style="list-style-type: none"><li>on：表示开启内存负反馈。</li><li>off：表示关闭内存负反馈。</li></ul>                                                                                                                   | on或者off   |

## 集群内存配置参数说明

表 10-19 集群内存配置参数说明

| 参数名称                 | 参数描述                                                                                                                         | 取值范围                      |
|----------------------|------------------------------------------------------------------------------------------------------------------------------|---------------------------|
| comm_usable_memory   | 单个DN内TCP代理通信库或SCTP通信库缓存最大可使用内存。单位为KB。                                                                                        | 102,400 ~ 1,073,741,823   |
| cstore_buffers       | 设置列存和OBS、HDFS外表列存格式（orc、parquet、carbondata）所使用的共享缓冲区的大小。单位为KB。                                                               | 16,384 ~ 1,073,741,823    |
| maintenance_work_mem | 设置在维护性操作（比如VACUUM、CREATE INDEX、ALTER TABLE ADD FOREIGN KEY等）中可使用的最大的内存。该参数的设置会影响VACUUM、VACUUMFULL、CLUSTER、CREATE INDEX的执行效率。 | 1,024 ~ 2,147,483,647     |
| max_process_memory   | 设置一个数据库节点可用的最大物理内存。单位：KB。默认值：物理内存 * 0.8 / (1 + 集群最大主DN数)。                                                                    | 2,097,152 ~ 2,147,483,647 |

| 参数名称                   | 参数描述                                                                                                                                                                                                                        | 取值范围                    |
|------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------|
| query_max_mem          | 设置执行作业所能够使用的最大内存。如果设置的query_max_mem值大于0，当作业执行时所使用内存超过该值时，将报错退出。                                                                                                                                                             | 0 ~ 2,147,483,647       |
| session_history_memory | 设置历史查询视图的内存大小。单位：KB。                                                                                                                                                                                                        | 10,240 ~ 2,147,483,647  |
| shared_buffers         | 设置DWS使用的共享内存大小。增加此参数的值会使DWS比系统默认设置需要更多的System V共享内存。单位：8KB。                                                                                                                                                                 | 16 ~ 1,073,741,823      |
| udf_memory_limit       | 控制每个CN、DN执行UDF时可用的最大物理内存量。单位：KB。                                                                                                                                                                                            | 204,800 ~ 2,147,483,647 |
| work_mem               | 设置内部排序操作和Hash表在开始写入临时磁盘文件之前使用的内存大小。ORDER BY，DISTINCT和merge joins都要用到排序操作。Hash表在散列连接、散列为基础的聚集、散列为基础的IN子查询处理中都要用到。对于复杂的查询，可能会同时并发运行好几个排序或者散列操作，每个都可以使用此参数所声明的内存量，不足时会使用临时文件。同样，好几个正在运行的会话可能会同时进行排序操作。因此使用的总内存可能是work_mem的好几倍。 | 64 ~ 2,147,483,647      |

## 集群磁盘配置参数说明

表 10-20 集群磁盘配置参数说明

| 参数名称               | 参数描述                                                                           | 取值范围               |
|--------------------|--------------------------------------------------------------------------------|--------------------|
| sql_use_spacelimit | 限制单个SQL在单个DN上，触发落盘操作时，落盘文件的空间大小，管控的空间包括普通表、临时表以及中间结果集落盘占用的空间。单位为KB。其中-1表示没有限制。 | -1 ~ 2,147,483,647 |
| temp_file_limit    | 限制一个会话中，触发落盘操作时，单个落盘文件的空间大小。例如一次会话中，排序和哈希表使用的临时文件，或者游标占用的临时文件。                 | -1 ~ 2,147,483,647 |

## 集群网络配置参数说明

表 10-21 集群网络配置参数说明

| 参数名称            | 参数描述                                                             | 取值范围          |
|-----------------|------------------------------------------------------------------|---------------|
| comm_max_stream | TCP代理通信库或SCTP通信库支持的最大并发数据流数。该参数值必须大于并发数*每并发平均stream算子数*（smp的平方）。 | 1 ~ 65,535    |
| max_connections | 允许和数据库连接的最大并发连接数。此参数会影响集群的并发能力。                                  | 100 ~ 262,143 |
| max_pool_size   | CN的连接池与其它某个CN/DN的最大连接数。                                          | 1 ~ 65,535    |

## SQL 调优配置参数说明

表 10-22 SQL 调优配置参数说明

| 参数名称                         | 参数描述                                                                                                    | 取值范围          |
|------------------------------|---------------------------------------------------------------------------------------------------------|---------------|
| agg_redistribute_enhancement | 当进行Agg操作时，如果包含多个group by列且均不为分布列，进行重分布时会选择某一group by列进行重分布。本参数控制选择重分布列的策略。                              | on或者off       |
| best_agg_plan                | 参数用于控制优化器生成哪种hashagg的计划。                                                                                | 0 ~ 3         |
| cost_model_version           | 控制应用场景中估算时cost使用的模型。该参数的影响范围主要涵盖：表达式distinct估算、HashJoin代价模型、行数估算、重分布时分布键的选择及Aggregate的行数估算等。            | 0 ~ 4         |
| default_statistics_target    | 默认analyze的比例。                                                                                           | -100 ~ 10,000 |
| enable_codegen               | 标识是否允许开启代码生成优化，目前代码生成使用的是LLVM优化。on表示允许开启代码生成优化。off表示不允许开启代码生成优化。                                        | -             |
| enable_extrapolation_stats   | 标识对于日期类型是否允许基于历史统计信息使用推理估算的逻辑。使用该逻辑对于未及时收集统计信息的表可以增大估算准确的可能性，但也存在错误推理导致估算过大的可能性，需要对于日期类型数据定期插入的场景开启此开关。 | on或者off       |

| 参数名称                    | 参数描述                                                                                                                                                                                                                                                                                                                                                       | 取值范围        |
|-------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| hashjoin_spill_strategy | 选择hashjoin下盘策略： <ul style="list-style-type: none"><li>0: 当内表较大，并且多次下盘无法分开时，尝试外表是否可以放到数据库可用内存建立哈希表。如果内外表均很大，执行NestLoop。</li><li>1: 当内表较大，并且多次下盘无法分开时，尝试外表是否可以放到数据库可用内存建立哈希表。如果内外表均很大，强制执行HashJoin。</li><li>2: 当内表较大，并且多次下盘无法分开时，强制执行HashJoin。</li><li>3: 当内表较大，并且多次下盘无法分开时，尝试外表是否可以放到数据库可用内存建立哈希表。如果内外表均很大，则报错。</li><li>4: 当内表较大，并且多次下盘无法分开时，则报错。</li></ul> | 0 ~ 6       |
| max_streams_per_query   | 控制查询计划中Stream节点的数目。                                                                                                                                                                                                                                                                                                                                        | -1 ~ 10,000 |
| qrw_inlist2join_optmode | 控制是否使用inlist-to-join查询重写。 <ul style="list-style-type: none"><li>disable: 关闭inlist2join查询重写。</li><li>cost_base: 基于代价的inlist2join查询重写。</li><li>rule_base: 基于规则的inlist2join查询重写，即强制使用inlist2join查询重写。</li><li>任意正整数: inlist2join查询重写阈值，即list内元素个数大于该阈值，进行inlist2join查询重写。</li></ul>                                                                           | -           |
| query_dop               | 用户自定义的查询并行度。[1,64]: 打开固定SMP功能，系统会使用固定并行度。[-64,-1]: 打开SMP自适应功能，并限制自适应选取的最大并行度。                                                                                                                                                                                                                                                                              | -64 ~ 64    |

| 参数名称         | 参数描述                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | 取值范围 |
|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|
| rewrite_rule | <p>标识开启的可选查询重写规则。有部分查询重写规则是可选的，开启它们并不能总是对查询效率有提升效果。在特定的客户场景中，通过此GUC参数对查询重写规则进行设置，使得查询效率最优。</p> <ul style="list-style-type: none"><li>• none：不使用任何可选查询重写规则。</li><li>• lazyagg：使用Lazy Agg查询重写规则（消除子查询中的聚集运算）。</li><li>• magicset：使用Magic Set查询重写规则（从主查询中下推条件到提升的子链接）。</li><li>• uniquecheck：使用Unique Check重写规则（允许目标列不含聚集函数的表达式子链接场景提升，需在子链接按关联列聚集后目标列值唯一才能开启，建议专业调优人员使用）。</li><li>• disablerep：使用禁止复制表的子链接提升规则（针对复制表禁止子链接提升）。</li><li>• projection_pushdown：使用Projection Pushdown重写规则（子查询中消除父查询不使用的列）。</li><li>• or_conversion：使用OR转换重写规则（消除执行效率低下的关联OR条件）。</li><li>• plain_lazyagg：使用Plain Lazy Agg查询重写规则（消除单子查询中的聚集操作）。该选项仅8.1.3.100及以上集群版本支持。</li><li>• eager_magicset：使用eager_magicset查询重写规则（从主查询中下推条件到子查询）。该选项仅8.2.0及以上集群版本支持。</li><li>• casewhen_simplification：使用casewhen语句改写的重写规则，开启后对(case when xxx then const1 else const2)=const1场景进行改写。该选项仅8.3.0及以上集群版本支持。</li><li>• outer_join_quality_imply：left outer join和right outer join存在等值关联条件时，外表关联列上的表达式条件下推到内表的关联列上。该选项仅8.3.0及以上集群版本支持。</li><li>• inlist_merge：使用inlist_or_inlist查询重写规则，支持基表相同列的or语句</li></ul> | -    |

| 参数名称 | 参数描述                                                                                                                                                                                                                                                                                                                                     | 取值范围 |
|------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|
|      | <p>合并，开启后对(where a in (list1) or a in (list2))进行合并改写，合并后可支持inlist2join。该选项仅8.3.0及以上集群版本支持。</p> <ul style="list-style-type: none"><li>subquery_qual_pull_up: 对于无法提升的子查询，若子查询与其它表进行关联的关联列在子查询内部有过滤条件，支持从子查询中提取过滤条件并传递到关联条件另一侧。目前只支持不存在类型转换的var op const形式，例如a &gt; 2。打开该开关时，则认为outer_join_quality_imply也是打开的。该选项仅9.1.0及以上集群版本支持。</li></ul> |      |

## SQL 兼容性配置参数说明

表 10-23 SQL 兼容性配置参数说明

| 参数名称               | 参数描述                                                                                                                                                                                                                           | 取值范围 |
|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|
| full_group_by_mode | <p>用于控制disable_full_group_by_mysql语法开关打开后的两种不同行为。</p> <ul style="list-style-type: none"><li>nullpadding: 表示对于非聚集列而言，对该列NULL值进行填充，取该列非NULL值，结果集可能为不同行。</li><li>notpadding: 表示对于非聚集列而言，不处理NULL值，取该行整行数据，非聚集列结果集为随机的一行。</li></ul> | -    |

## 10.7 管理 DWS 标签

### 10.7.1 标签简介

标签（Tag）是用户自定义的、用于标识云资源的键值对，它可以帮助用户对云资源进行分类和搜索。

标签由标签“键”和标签“值”组成。

- 一个标签“键”可以拥有多个标签“值”。
- 一个云资源不能拥有重复的标签“键”。

在DWS中，当用户创建集群后，如果想对集群添加如项目名称、业务类别、背景信息等相关信息的标识，用户可以通过添加标签来实现。如果用户在其他云服务中使用了

标签，那么建议用户为同一个业务所使用的云资源创建相同的标签键值对以保持一致性。

DWS支持以下标签：

- 资源标签  
在DWS中创建的非全局的标签。
- 预定义标签  
在标签管理服务（简称TMS）中创建的预定义标签，属于全局标签。  
有关预定义标签的更多信息，请参见《标签管理服务用户指南》。

在DWS中，支持对以下资源添加标签：

- 集群  
支持在创建集群时或在集群创建成功后，对集群添加标签。并且，在集群列表中，可通过标签进行搜索。  
每个集群最多可以添加 20 个标签。  
当用户对某一集群添加了标签，并对该集群创建快照，使用快照恢复集群时，该集群的标签不会恢复，需要用户重新添加标签。  
当删除集群时，也会删除与集群关联的非预定义标签，而预定义标签则需要前往标签管理服务进行删除。

## 10.7.2 标签管理

本章节介绍如何根据标签搜索集群，以及如何为集群添加标签、修改标签和删除标签。

### 为集群添加标签

- 步骤1** 在“专属集群 > 集群列表”页面，单击需要添加标签的集群的名称，切换至“标签”页签。
- 步骤2** 单击“添加标签”，弹出“添加标签”对话框。
- 步骤3** 在“添加标签”对话框中配置标签参数，当配置了标签键值后，标签值不能为空值。

表 10-24 标签配置参数

| 参数 | 参数说明                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | 样例值     |
|----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------|
| 键  | <p>您可以选择：</p> <ul style="list-style-type: none"><li>在输入框的下拉列表中选择预定义标签键或集群已有的资源标签键。</li></ul> <p><b>说明</b><br/>如果添加预定义标签，用户需要预先在标签管理服务中创建好预定义标签，然后在“键”的下拉框中进行选择。用户可以通过单击“查看预定义标签”进入标签管理服务的“预定义标签”页面，然后单击“创建标签”来创建新的预定义标签。更多信息请参见《标签管理服务用户指南》中的“管理 &gt; 预定义标签 &gt; 创建预定义标签”章节。</p> <ul style="list-style-type: none"><li>在输入框中输入标签键名称。输入标签键的最大长度为128个字符，不能为空字符串，不能以 _sys_ 开头，且首尾字符不能为空格。</li></ul> <p>只能包含字母、数字、中文和空格，以及 _ . : = + - @。</p> <p><b>说明</b><br/>同一集群中的键名不能重复。</p> | key01   |
| 值  | <p>您可以选择：</p> <ul style="list-style-type: none"><li>在输入框的下拉列表中选择预定义标签值或集群的资源标签值。</li><li>在输入框中输入标签值。输入标签值的最大长度为255个字符，首尾字符不能为空格，可以为空字符串。</li></ul> <p>只能包含字母、数字、中文和空格，以及 _ . : = + - @。</p>                                                                                                                                                                                                                                                                                  | value01 |

步骤4 单击“确定”。

----结束

根据标签搜索集群

对于已经添加过标签的集群，用户可以通过设置标签过滤条件进行搜索，以便快速找到集群。

- 步骤1 登录DWS管理控制台。
- 步骤2 单击“专属集群 > 集群列表”。
- 步骤3 单击集群列表上方的搜索框，下划到“资源标签”筛选处。
- 步骤4 在“资源标签”区域，单击要搜索的标签键，然后再选择对应的标签值。再次单击搜索框，可继续添加标签筛选条件。
- 标签搜索仅支持“资源标签”区域下拉列表中已存在的键和值。如果没有可用的标签键和值，请先为集群创建标签，具体参见[为集群添加标签](#)。
- 步骤5 单击“搜索”，在集群列表中将显示搜索到的集群。
- 结束

## 修改集群的标签

- 步骤1 在“专属集群 > 集群列表”页面，单击需要修改标签的集群的名称，然后选择“标签”页签。
  - 步骤2 在所要修改的标签所在行操作列单击“编辑”按钮，弹出“编辑”对话框。
  - 步骤3 在对话框的“值”中输入修改后的键值。
  - 步骤4 单击“确定”。
- 结束

## 删除集群的标签

- 步骤1 在“专属集群 > 集群”页面，单击需要删除标签的集群的名称，然后单击“标签”页签。
  - 步骤2 在所要删除的标签所在行，单击“操作”列中的“删除”，弹出“删除”确认对话框。
  - 步骤3 确认无误后，输入“DELETE”或单击“一键输入”，并单击“确定”，删除此标签。
- 结束

## 10.8 重置 DWS 数据库系统管理员密码

DWS支持对数据库管理员账号进行重置密码。当数据库管理员忘记密码或者在登录时连续输错密码的次数达到上限导致账号被锁时，可以在“专属集群 > 集群列表”页面重置密码，重置密码后账号即可自动解锁。输错密码的次数上限默认为10次，可以在集群的“参数修改”页面通过参数“failed\_login\_attempts”进行设置，设置方法请参见[修改DWS集群GUC参数](#)。

### 重置密码

- 步骤1 登录DWS管理控制台。
- 步骤2 单击“专属集群 > 集群列表”。
- 步骤3 在指定集群的“操作”列，选择“更多 > 重置密码”，弹出“重置密码”页面。
- 步骤4 填写并确认新密码，单击“确定”。

密码复杂度要求如下：

- 密码长度为12～32个字符。
- 不能与用户名或倒序的用户名相同。
- 密码至少包含大写字母、小写字母、数字和特殊字符四类中的三类，其中可输入的特殊字符为：(~!?,;:\_){}[]/<>@#%^&\*+|\=-)。
- 弱密码检查。
- 新密码不能与当前密码或当前密码的倒序相同。
- 不能将历史密码重复使用。

#### 说明

若集群默认数据库管理员用户被删除或者重命名，将导致重置密码操作失败。

----结束

## 10.9 启停和删除 DWS 集群

### 重启集群

当集群处于非均衡或不能正常工作时，可能需要通过重启集群进行恢复。当您修改完配置，例如修改集群安全设置、参数修改相关配置，未立即重启集群的情况下，您也可以通过手动重启集群使配置生效。

#### 对系统的影响

- 重启期间集群将无法提供服务。因此，在重启前，请确定集群中没有正在运行的任务，并且所有数据都已经保存。  
如果集群正在处理业务数据，如导入数据、查询数据、创建快照或恢复快照时，一旦重启集群，有可能会造成文件损坏或重启失败。因此，建议停止所有集群任务后，再重启集群。  
您可以参考[在Cloud Eye查看DWS集群监控](#)查看集群的“会话数”和“活跃SQL数”指标，查看是否有活跃事务。
- 重启集群所需时间与集群的规模和业务有关，正常情况下大约需要3分钟左右，不超过20分钟。
- 如果重启失败，将有可能导致集群不可用，建议联系技术支持人员进行处理或稍后重试。

#### 操作步骤

**步骤1** 登录DWS管理控制台。

**步骤2** 单击“专属集群 > 集群列表”。

**步骤3** 在需要重启的集群的“操作”列，选择“更多 > 重启”。

#### 说明

集群列表操作栏功能键位置不固定，如果该节点部分功能不支持，“更多”里的功能键可能会前移和“更多”按钮并列。

**步骤4** 在弹出框单击“是”。

此时集群的“任务信息”变为“重启中”。当“集群状态”重新变为“可用”时，表示重启已成功。

----结束

### 停止集群

当用户的集群不再使用时，可通过停止集群功能来关闭集群，方便业务下线。

### 说明

- 如果当前控制台界面不支持该特性，请联系技术支持人员。
- 集群停止后实例关机，ECS场景下基础资源（vCPU、内存）不再保留，当再次启动云服务时，可能由于资源不足无法正常开机，请耐心等待，稍后再试。

### 操作步骤

**步骤1** 登录DWS管理控制台。

**步骤2** 单击左侧导航栏的“专属集群 > 集群列表”。

**步骤3** 在“专属集群”页面，在需要停止的专属集群的所在行“操作”列，选择“更多 > 停止”。

**步骤4** 在弹出框单击“是”。

此时集群的“任务信息”变为“停止中”。当“集群状态”变为“已停止”时，表示停止集群已成功。

----结束

## 启动集群

已停止的集群想要恢复集群使用时，可通过启动集群功能触发集群启动，恢复集群业务。

### 说明

如果当前控制台界面不支持该特性，请联系技术支持人员。

### 操作步骤

**步骤1** 登录DWS管理控制台。

**步骤2** 单击左侧导航栏的“专属集群 > 集群列表”。

**步骤3** 在“专属集群”页面，在需要启动的专属集群的所在行“操作”列，选择“更多 > 启动”。

**步骤4** 在弹出框单击“是”。

此时集群的“任务信息”变为“启动中”。当“集群状态”变为“可用”时，表示启动集群已成功。

----结束

## 删除集群

当用户不再需要使用某个集群时，可以参考本章节删除该集群。

### 对系统的影响

成功删除的集群无法恢复，同时集群中的用户数据、自动快照也会自动删除且无法再访问。删除集群时不会删除手动快照。

### 操作步骤

**步骤1** 登录DWS管理控制台。

- 步骤2** 单击管理控制台左上角的 ，选择区域。
- 步骤3** 在“专属集群 > 集群列表”页面，查找到所要删除的集群。
- 步骤4** 在集群所在行的“操作”列，单击“更多 > 删除”。
- 步骤5** 在弹出对话框中，再次进行删除确认，可以选择是否执行以下操作：
- 立即创建集群快照  
如果集群状态无异常，可以单击“立即创建集群快照”，跳转至快照列表页面，单击“创建快照”，为待删除的集群创建一个最新的快照，详情请参见[手动快照](#)。等快照创建完成后，请再返回“专属集群”页面执行删除集群的操作。
  - 关联资源名称
    - 释放与集群绑定的弹性公网IP  
如果集群绑定了弹性公网IP，建议勾选“弹性公网IP”，将待删除集群的弹性IP资源释放。
    - 自动快照
    - 手动快照  
如果用户创建了手动快照，可勾选“手动快照”进行删除。
- 步骤6** 确认无误后，输入“DELETE”或单击“一键输入”，并单击“确定”，删除此集群。此时集群列表中集群状态会显示为“删除中”，并显示集群删除进度。
- 如果待删除集群使用了自动创建的安全组，且该自动创建的安全组没有被别的集群使用，删除集群时，该安全组也会被一并删除。

----结束

## 10.10 管理企业项目

企业项目是一种云资源管理方式。企业管理（Enterprise Management）提供面向企业客户的云上等综合管理服务。区别于管理控制台独立操控、配置云产品的方式，企业管理控制台以面向企业资源管理为出发点，帮助企业以公司、部门、项目等分级管理方式实现企业云上的管理。

### 绑定企业项目

在DWS中，用户可以在创建集群时为集群选择所属的企业项目，从而将DWS集群与企业项目进行关联，详情请参见[创建DWS存算一体集群](#)。在选择“企业项目”的下拉列表中，将显示用户在企业项目服务中已创建的项目。系统还内置了一个缺省的企业项目“default”，如果用户没有为集群选择企业项目，将使用缺省项目“default”。

在集群创建过程中，如果集群与企业项目绑定成功，则集群创建成功，如果绑定失败，系统会发送告警，集群就会创建失败。

在为DWS集群创建快照时，会同时保存集群与企业项目的关联关系，在恢复集群时，也会恢复企业项目的关联关系。

当删除DWS集群时，DWS集群与企业项目的关联关系就会被自动删除。

## 查看企业项目

在集群创建成功后，您可以在集群列表和集群详情页面查看集群关联的企业项目。用户只能查询到有访问权限的项目下的集群资源。

- 在集群管理页面的集群列表中，查看集群所属的企业项目。
- 在集群列表中，找到所需要的集群，然后单击集群名称，进入“集群详情”页面，可以查看与集群关联的企业项目。单击企业项目的名称，可以跳转到企业管理的控制台页面对该企业项目进行查看或编辑。
- 同时，在企业管理的控制台上，查询指定项目中的资源列表时，也可以查询到DWS服务的资源。

## 按企业项目搜索集群

登录DWS管理控制台，单击“专属集群 > 集群列表”，在集群列表上方单击搜索框，下划至“企业项目”，然后输入所需搜索的项目名称，单击搜索按钮即可查看与该项目关联的所有集群。

## 将集群迁入或迁出企业项目

一个DWS集群只能关联一个企业项目。当集群创建成功后，可以在企业管理的控制台上，执行迁出操作，将DWS集群从当前所属的企业项目中迁出到另一个企业项目中；或者执行迁入操作，在指定的企业项目中迁入另一个企业项目中的DWS集群。迁入迁出后，DWS集群与新的企业项目进行关联，DWS集群与原企业项目的关联关系将被自动解除。

# 11 DWS 集群运维

---

## 11.1 在 Cloud Eye 查看 DWS 集群监控

### 功能说明

用户需要查看集群的监控指标信息时，可以通过云监控（Cloud Eye）确认集群各项指标情况。通过监控集群运行时的各项指标，用户可以识别出数据库集群状态异常的时间段，然后在数据库日志中，分析可能存在问题的活动，从而优化数据库性能。本章节定义了数据仓库服务上报云监控服务的监控指标的命名空间、监控指标列表和维度定义。用户可以通过云监控服务提供的管理控制台或API接口来检索数据仓库服务产生的监控指标和告警信息，详情请参见云监控服务的《用户指南》和《API参考》。

### 命名空间

SYS.DWS

### 集群监控指标

通过云监控提供的数据库服务相关监控指标，用户可以获取有关集群运行状况和性能的信息，并深入了解节点级别的对应信息。

数据库服务相关监控指标，具体请参见[表11-1](#)。

表 11-1 数据仓库服务支持的监控指标

| 指标ID                             | 指标名称     | 指标含义                                                                                                           | 取值范围      | 单位      | 进制  | 测量对象（维度） | 监控周期（原始指标） |
|----------------------------------|----------|----------------------------------------------------------------------------------------------------------------|-----------|---------|-----|----------|------------|
| dws001_shared_buffer_hit_ratio   | 缓存命中率    | 该指标是指在系统中请求数据时，请求到的数据已经存在于缓存中的比率。即缓存中已经存在的数据量与请求的总数据量之比。缓存命中率越高，说明系统的缓存利用率越高，请求数据时需要从磁盘或网络中读取的次数就越少，系统响应速度就越快。 | 0% ~ 100% | %       | 不涉及 | 数据仓库集群   | 4 分钟       |
| dws002_in_memory_sort_ratio      | 内存中排序比率  | 该指标用于表示排序算法所使用的额外内存空间与排序数据所占用的内存空间的比值。例如，在归并排序中，归并缓冲区的大小通常是排序数据大小的一部分，因此归并排序的内存中排序比率通常在10%到50%之间。              | 0% ~ 100% | %       | 不涉及 | 数据仓库集群   | 4 分钟       |
| dws003_physical_reads            | 文件读取次数   | 该指标用于表示读取数据库文件的累计次数。                                                                                           | > 0       | count   | 不涉及 | 数据仓库集群   | 4 分钟       |
| dws004_physical_writes           | 文件写入次数   | 该指标用于表示写入数据库文件的累计次数。                                                                                           | > 0       | count   | 不涉及 | 数据仓库集群   | 4 分钟       |
| dws005_physical_reads_per_second | 每秒文件读取次数 | 该指标用于表示每秒读取数据库文件的次数。                                                                                           | ≥ 0       | count/s | 不涉及 | 数据仓库集群   | 4 分钟       |

| 指标ID                              | 指标名称     | 指标含义                                                                                                             | 取值范围      | 单位      | 进制        | 测量对象<br>(维度) | 监控周期<br>(原始指标) |
|-----------------------------------|----------|------------------------------------------------------------------------------------------------------------------|-----------|---------|-----------|--------------|----------------|
| dws006_physical_writes_per_second | 每秒文件写入次数 | 该指标用于表示每秒写入数据库文件的次数。                                                                                             | ≥ 0       | count/s | 不涉及       | 数据仓库集群       | 4 分钟           |
| dws007_db_size                    | 数据量大小    | 该指标用于表示数据库中所有数据总量大小。                                                                                             | ≥0MB      | MB      | 1024(IEC) | 数据仓库集群       | 4 分钟           |
| dws008_active_sql_count           | 活跃SQL数   | 该指标用于表示数据库当前活动的SQL个数。                                                                                            | ≥ 0       | count   | 不涉及       | 数据仓库集群       | 4 分钟           |
| dws009_session_count              | 会话数      | 该指标用于表示数据库当前访问的连接个数。                                                                                             | ≥ 0       | count   | 不涉及       | 数据仓库集群       | 4 分钟           |
| dws010_cpu_usage                  | CPU使用率   | 该指标用于统计每个集群节点的CPU使用率。                                                                                            | 0% ~ 100% | %       | 不涉及       | 数据仓库节点       | 1 分钟           |
| dws011_mem_usage                  | 内存使用率    | 该指标用于统计每个集群节点的内存使用率，以百分比为单位。<br><b>说明</b><br>由于控制台界面升级到新版本(8.3.0.202)后，内存使用率统计包括了缓存Cache所占的内存大小，该指标显示值比升级前会有所提升。 | 0% ~ 100% | %       | 不涉及       | 数据仓库节点       | 1 分钟           |
| dws012_iops                       | IOPS     | 该指标用于统计每个集群节点每秒处理的I/O请求数量。                                                                                       | ≥ 0       | count/s | 不涉及       | 数据仓库节点       | 1 分钟           |
| dws013_bytes_in                   | 网络输入吞吐量  | 该指标用于统计每秒从网络输入每个集群节点的数据总量。                                                                                       | ≥ 0字节/秒   | byte/s  | 1024(IEC) | 数据仓库节点       | 1 分钟           |

| 指标ID                                  | 指标名称       | 指标含义                         | 取值范围           | 单位     | 进制        | 测量对象<br>(维度) | 监控周期<br>(原始指标) |
|---------------------------------------|------------|------------------------------|----------------|--------|-----------|--------------|----------------|
| dws014_by<br>tes_out                  | 网络输出吞吐量    | 该指标用于统计从每个节点每秒发送到网络的数据总量。    | ≥ 0字节/秒        | byte/s | 1024(IEC) | 数据仓库节点       | 1分钟            |
| dws015_dis<br>k_usage                 | 磁盘利用率      | 该指标用于统计每个集群节点的磁盘使用情况。        | 0% ~ 100%      | %      | 不涉及       | 数据仓库节点       | 1分钟            |
| dws016_dis<br>k_total_size            | 磁盘总大小      | 该指标用于统计每个集群节点的数据磁盘总大小。       | 100GB ~ 2000GB | GB     | 1024(IEC) | 数据仓库节点       | 1分钟            |
| dws017_dis<br>k_used_size             | 磁盘使用量      | 该指标用于统计每个集群节点的数据磁盘已用大小。      | 0GB ~ 3600GB   | GB     | 1024(IEC) | 数据仓库节点       | 1分钟            |
| dws018_dis<br>k_read_thr<br>oughput   | 磁盘读吞吐量     | 该指标用于统计集群节点数据磁盘每秒读取的数据量。     | ≥ 0字节/秒        | byte/s | 1024(IEC) | 数据仓库节点       | 1分钟            |
| dws019_dis<br>k_write_thr<br>oughput  | 磁盘写吞吐量     | 该指标用于统计集群节点数据磁盘每秒写入的数据量。     | ≥ 0字节/秒        | byte/s | 1024(IEC) | 数据仓库节点       | 1分钟            |
| dws020_av<br>g_disk_sec_<br>per_read  | 磁盘读耗时      | 该指标用于统计集群节点数据磁盘每次读取数据的平均耗时。  | > 0秒           | Second | 不涉及       | 数据仓库节点       | 1分钟            |
| dws021_av<br>g_disk_sec_<br>per_write | 磁盘写耗时      | 该指标用于统计集群节点数据磁盘每次写入数据的平均耗时。  | > 0秒           | Second | 不涉及       | 数据仓库节点       | 1分钟            |
| dws022_av<br>g_disk_que<br>ue_length  | 磁盘平均队列长度   | 该指标用于统计每个集群节点数据磁盘平均的I/O队列长度。 | ≥ 0            | count  | 不涉及       | 数据仓库节点       | 1分钟            |
| dws_024_d<br>n_diskio_u<br>til        | DN实例I/O使用率 | 该指标用于统计集群DN实例磁盘平均I/O使用率。     | 0% ~ 100%      | %      | 不涉及       | 数据仓库实例       | 1分钟            |

维度

| Key             | Value    |
|-----------------|----------|
| datastore_id    | 数据仓库集群ID |
| dws_instance_id | 数据仓库节点ID |

查看集群监控和节点监控信息

- 步骤1 登录DWS管理控制台，进入“专属集群 > 集群列表”页面。
- 步骤2 查看集群信息：在集群列表中，在指定集群所在行，单击“查看监控指标”，打开云监控管理控制台。默认显示的是集群监控信息。

用户可以根据需要，选择指定的监控指标名称以及时间范围，从而显示监控指标的性能曲线。

 说明

集群列表操作栏功能键位置不固定，如果该节点部分功能不支持，“更多”里的功能键可能会前移和“更多”按钮并列。

----结束

多节点间的监控指标对比

- 步骤1 在云监控管理控制台的左侧导航栏中，选择“监控看板 > 我的看板”，单击需要添加监控视图的看板名称，进入我的看板中，然后单击“添加监控视图”，系统弹出“添加监控视图”窗口。
- 步骤2 在“添加监控视图”界面时，可以选择“曲线图”或者“条形图”来显示视图，确认无误后单击“确定”。

例如选择“曲线图”，指标展示选择“多指标一个视图”，对比DWS三个节点间的CPU使用率，参数设置如下表所示。

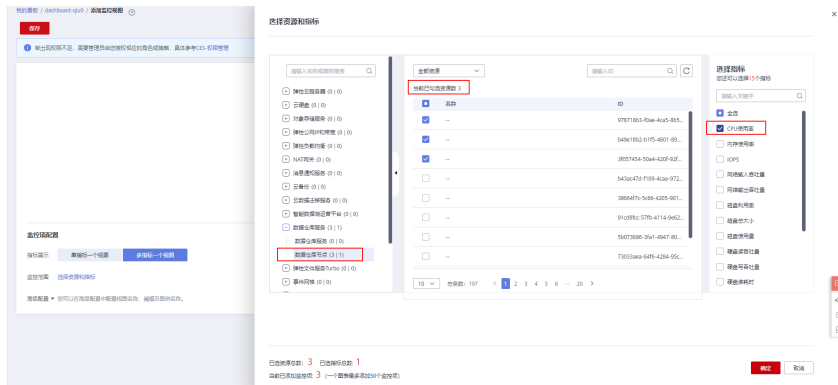


表 11-2 配置样例

| 参数名  | 样例值                                                                     |
|------|-------------------------------------------------------------------------|
| 资源类型 | 数据仓库服务                                                                  |
| 维度   | 数据仓库节点                                                                  |
| 监控对象 | dws-demo-dws-cn-cn-2-1<br>dws-demo-dws-cn-cn-1-1<br>dws-demo-dws-dn-1-1 |
| 监控指标 | CPU使用率                                                                  |

**步骤3** 单击“保存”，完成监控视图的添加。

在所选的我的看板上可以查看新添加监控视图的监控走势图，单击放大按钮，可放大查看详细的指标对比数据。

----结束

## 创建告警规则

通过设置DWS的告警规则，用户可以自定义监控目标和通知策略，及时了解DWS运行状态，从而起到预警作用。

设置DWS的告警规则包括设置告警规则名称、监控对象、监控指标、告警阈值、监控周期和是否发送通知等参数。本节介绍设置DWS规则的具体方法。

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中选择“专属集群 > 集群列表”。

**步骤3** 在指定集群所在行的“操作”列中，单击“查看监控指标”，进入云监控管理控制台，查看数据仓库服务监控信息。

确认待查看“监控信息”的集群状态必须为“可用”，否则不能进行创建告警操作。

**步骤4** 在云监控管理控制台的左侧导航树上，选择“告警 > 告警规则”。

**步骤5** 在“告警规则”界面，单击右上方“创建告警规则”。

**步骤6** 在“创建告警规则”界面，根据界面提示配置参数。

1. 配置规则名称和描述。
2. 配置告警内容，根据界面提示配置参数。

表 11-3 配置告警内容

| 参数名称 | 参数说明              | 取值样例   |
|------|-------------------|--------|
| 资源类型 | 配置告警规则监控的云服务资源名称。 | 数据仓库服务 |

| 参数名称 | 参数说明                                                                                   | 取值样例   |
|------|----------------------------------------------------------------------------------------|--------|
| 维度   | 用于指定告警规则对应指标的维度名称。可以选择“数据仓库节点”或“数据仓库服务”。                                               | 数据仓库节点 |
| 监控范围 | 告警规则适用的资源范围，选择“指定资源”，然后勾选具体的监控对象，可以是一个或多个。对于DWS而言，在弹窗中选择您创建的集群实例ID或节点ID。               | 指定资源   |
| 触发规则 | 根据需要可选择关联模板、导入已有模板或自定义创建。                                                              | 自定义创建  |
| 模板   | 当告警内容选择从模板导入时，该参数有效。<br>选择需要导入的模板。如果当前没有告警模板，可以单击“创建自定义告警模板”创建一个符合您需求的告警模板。            | -      |
| 告警策略 | 当告警内容选择从自定义创建时，该参数有效。<br>设置触发告警的告警策略。例如，CPU使用率连续三个周期原始值≥80%。<br>数据仓库服务支持的监控指标请参见表11-1。 | -      |
| 告警级别 | 根据告警的严重程度不同等级，可选择紧急、重要、次要、提示。                                                          | 重要     |

### 3. 配置告警通知，根据界面提示配置参数。

表 11-4 配置告警通知

| 参数名称 | 参数说明                                                                                                   | 取值样例 |
|------|--------------------------------------------------------------------------------------------------------|------|
| 发送通知 | 配置是否发送邮件、短信、HTTP和HTTPS通知用户。<br>选择“是”（推荐选择），会发送通知；选择“否”，不会发送通知。                                         | 是    |
| 生效时间 | 该告警规则仅在生效时间内发送通知消息。<br>如生效时间为00:00-8:00，则该告警规则仅在00:00-8:00发送通知消息。                                      | -    |
| 通知对象 | 需要发送告警通知的主题名称。<br>当发送通知选择“是”时，需要选择已有的主题名称，若此处没有需要的主题则需先创建主题，该功能会调用消息通知服务（SMN）。<br>创建主题请参见《消息通知服务用户指南》。 | -    |

| 参数名称 | 参数说明                               | 取值样例 |
|------|------------------------------------|------|
| 触发条件 | 可以选择“出现告警”、“恢复正常”两种状态，作为触发告警通知的条件。 | -    |

- 配置完成后，单击“立即创建”，完成告警规则的创建。

告警规则添加完成后，当监控指标触发设定的阈值时，云监控会在第一时间通过消息通知服务实时告知您资源异常，以免因此造成业务损失。

----结束

## 11.2 查看并订阅 DWS 集群告警

### 11.2.1 告警管理简介

#### 概述

告警管理包含查看告警规则、告警规则配置与告警信息订阅功能。其中，告警规则可以提供过去一周的告警信息统计与告警信息明细，方便用户自行查看租户下的告警。该特性除了以默认值的形式提供一套DWS告警最佳实践外，还允许用户根据自己的业务特点，个性化修改告警阈值。告警管理通过消息通知服务（Simple Message Notification，简称SMN）发送DWS告警通知，用户可订阅告警启用通知。

#### 说明

- 该特性仅8.1.1.200及以上集群版本支持。
- 告警管理目前暂不支持按照企业项目划分告警。

#### 进入告警管理页面

**步骤1** 登录DWS 管理控制台。

**步骤2** 在左侧导航栏，选择“监控 > 告警”，切换至“告警”页签。

**步骤3** 进入数据仓库告警展示页面。该页面分为三个区域：

- 存量告警统计**  
最近7天的存量告警统计值（按告警级别分类），以柱状图的形式展示。用户可通过存量告警统计图，对过去一周告警发生的数量和分布有清晰的了解。
- 当日告警**  
当天的存量告警统计值（按级别分类），以列表的形式展示。重点向用户强调当天未处理的告警数量，帮助用户快速掌握目前告警的数量和分布。
- 告警详情**  
最近7天的所有告警（包括已处理和未处理）的明细信息，以表格的形式展示。可查看近7天内所有告警的告警名称、告警级别、告警源、集群名称、定位信息、详细信息、产生日期、状态等信息，帮助用户快速发现和定位问题。

 说明

告警展示页面的数据来自EventService微服务，该微服务最多可以提供30天的告警缓存数据。

----结束

告警类别和告警

 说明

告警策略触发条件以用户实际配置为准。

表 11-5 DMS 告警源触发阈值告警

| 告警类别 | 告警名称                | 告警级别           | 告警描述                                                                                                                                                                                 |
|------|---------------------|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 默认   | DWS集群节点CPU使用率超阈值    | 紧急             | DMS告警模块在指定周期内，检测到集群任意节点的CPU使用率（系统+用户）超过当前设定阈值，且抑制条件不满足时，DMS告警模块将触发该告警；检测到集群任意节点的CPU使用率（系统+用户）低于当前设定阈值，且抑制条件不满足时，DMS告警模块将消除该告警。                                                       |
| 默认   | DWS集群节点数据盘使用率超阈值    | >85% 紧急，>80%重要 | DMS告警模块在指定周期内，检测到集群任意节点的数据盘（/var/chroot/DWS/data[n]）使用率超过当前设定阈值，且抑制条件不满足时，DMS告警模块将触发该告警；检测到集群任意节点的数据盘（/var/chroot/DWS/data[n]）使用率低于当前设定阈值，且抑制条件不满足时，DMS告警模块将消除该告警。                   |
| 默认   | DWS集群节点数据盘I/O利用率超阈值 | 紧急             | DMS告警模块在指定周期内，检测到集群任意节点的数据盘（/var/chroot/DWS/data[n]）I/O利用率（util）超过当前设定阈值，且抑制条件不满足时，DMS告警模块将触发该告警；检测到集群任意节点的数据盘（/var/chroot/DWS/data[n]）I/O利用率（util）低于当前设定阈值，且抑制条件不满足时，DMS告警模块将消除该告警。 |

| 告警类别 | 告警名称              | 告警级别 | 告警描述                                                                                                                                                                                 |
|------|-------------------|------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 默认   | DWS集群节点数据盘时延超阈值   | 重要   | DMS告警模块在指定周期内，检测到集群任意节点的数据盘（/var/chroot/DWS/data[n]）I/O延时（await）超过当前设定阈值，且抑制条件不满足时，DMS告警模块将触发该告警；检测到集群任意节点的数据盘（/var/chroot/DWS/data[n]）I/O延时（await）低于当前设定阈值，且抑制条件不满足时，DMS告警模块将消除该告警。 |
| 默认   | DWS集群查询语句触发下盘量超阈值 | 紧急   | DMS告警模块在指定周期内，检测到集群中执行的SQL语句触发结果集下盘，下盘量超过当前设定阈值，且抑制条件不满足时，DMS告警模块将触发该告警；该告警为针对每个SQL语句下盘量的告警，因此无法自动消除，需要用户在处理完该告警所涉及的SQL语句后手动消除该告警项。                                                  |
| 默认   | DWS集群查询语句堆积数量超阈值  | 紧急   | DMS告警模块在指定周期内，检测到集群中处于等待状态的SQL语句数量超过当前设定阈值时，DMS告警模块将触发该告警；检测到集群中处于等待状态的SQL语句数量低于当前设定阈值时，DMS告警模块将消除该告警。                                                                               |
| 默认   | DWS集群默认资源池队列阻塞    | 紧急   | DMS告警模块在指定周期内，检测到集群的默认资源池队列发生阻塞，且抑制条件不能满足时，DMS告警模块将触发该告警；检测到集群的默认资源池队列不再发生阻塞，DMS告警模块将消除该告警。                                                                                          |
| 默认   | DWS集群的sql探针耗时超阈值  | 紧急   | DMS告警模块在指定周期内，检测到任意集群的某个主机上出现sql探针耗时超过阈值，且抑制条件不能满足时，DMS告警模块将触发该告警；检测到任意集群的某个主机上不再出现sql探针耗时超过阈值时，DMS告警模块将消除该告警。<br><b>说明</b><br>该告警仅8.1.1.300及以上集群版本支持，历史版本需要联系技术支持人员升级。              |

| 告警类别 | 告警名称                         | 告警级别 | 告警描述                                                                                                                                                                                 |
|------|------------------------------|------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 默认   | DWS集群中存在持有表锁过长的vacuum full操作 | 重要   | <p>DMS告警模块在指定周期内，检测到集群中存在长时间运行的vacuum full操作，并且阻塞了其他操作。其他业务SQL存在锁等待情况，且抑制条件不能满足时，DMS告警模块将触发该告警；检测到集群的vacuum full操作没有造成锁等待，DMS告警模块将消除该告警。</p> <p><b>说明</b><br/>该告警请联系技术支持人员升级后支持。</p> |
| 默认   | DWS集群节点实例内存使用率超阈值            | 紧急   | <p>DMS告警模块在指定周期内，检测到任意集群的某个节点上出现实例内存使用率超阈值，且抑制条件不能满足时，DMS告警模块将触发该告警；检测到任意集群的某个节点上不再出现实例内存使用率超阈值时，DMS告警模块将消除该告警。</p> <p><b>说明</b><br/>该告警请联系技术支持人员升级后支持。</p>                          |
| 默认   | DWS集群节点动态内存使用率超阈值            | 紧急   | <p>DMS告警模块在指定周期内，检测到任意集群的某个节点上出现动态内存使用率超阈值，且抑制条件不能满足时，DMS告警模块将触发该告警；检测到任意集群的某个节点上不再出现动态内存使用率超阈值时，DMS告警模块将消除该告警。</p> <p><b>说明</b><br/>该告警请联系技术支持人员升级后支持。</p>                          |
| 默认   | DWS集群资源池磁盘使用率超阈值             | 紧急   | <p>DMS告警模块在指定周期内，检测到集群资源池磁盘使用率超过当前设定阈值，且抑制条件不满足时，DMS告警模块将触发该告警；检测到集群资源池磁盘使用率低于当前设定阈值时，DMS告警模块将消除该告警。</p> <p><b>说明</b><br/>该告警请联系技术支持人员升级后支持。</p>                                     |
| 默认   | DWS集群会话数量使用率超阈值              | 紧急   | <p>DMS告警模块在指定周期内，检测到集群会话数量使用率超过当前设定阈值，且抑制条件不满足时，DMS告警模块将触发该告警；检测到集群会话数量使用率低于当前设定阈值时，DMS告警模块将消除该告警。</p> <p><b>说明</b><br/>该告警请联系技术支持人员升级后支持。</p>                                       |

| 告警类别 | 告警名称              | 告警级别 | 告警描述                                                                                                                                               |
|------|-------------------|------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| 默认   | DWS集群活跃会话数量使用率超阈值 | 紧急   | <p>DMS告警模块在指定周期内，检测到集群活跃会话数量使用率超过当前设定阈值，且抑制条件不满足时，DMS告警模块将触发该告警；检测到集群活跃会话数量使用率低于当前设定阈值时，DMS告警模块将消除该告警。</p> <p><b>说明</b><br/>该告警请联系技术支持人员升级后支持。</p> |
| 默认   | DWS集群数据库死锁数量超阈值   | 紧急   | <p>DMS告警模块在指定周期内，检测到集群数据库死锁数量超过当前设定阈值，且抑制条件不满足时，DMS告警模块将触发该告警；检测到集群数据库死锁数量低于当前设定阈值时，DMS告警模块将消除该告警。</p> <p><b>说明</b><br/>该告警请联系技术支持人员升级后支持。</p>     |
| 默认   | DWS集群数据库会话使用率超阈值  | 紧急   | <p>DMS告警模块在指定周期内，检测到集群数据库会话使用率超过当前设定阈值，且抑制条件不满足时，DMS告警模块将触发该告警；检测到集群数据库会话使用率低于当前设定阈值时，DMS告警模块将消除该告警。</p> <p><b>说明</b><br/>该告警请联系技术支持人员升级后支持。</p>   |

### 11.2.2 订阅告警通知

用户可通过订阅DWS的告警通知，在特定告警级别告警时收到短信、电子邮件或应用等方式的通知消息。

#### 创建订阅

- 步骤1 登录DWS管理控制台。
- 步骤2 在左侧导航树，选择“监控 > 告警”，切换至“订阅”页签。
- 步骤3 在页面左上角单击“创建订阅”按钮。
- 步骤4 在“订阅设置”区域，设置订阅基本信息及告警过滤。

### 订阅设置

订阅基本信息及告警过滤设置

★ 是否开启

?

★ 订阅名称

请输入订阅名称

?

集群

请选择集群

▼

表 11-6 订阅参数

| 参数名  | 参数解释                                                                                                                        |
|------|-----------------------------------------------------------------------------------------------------------------------------|
| 是否开启 | 设置是否开启告警订阅。<br>关闭后停止发送已订阅告警的通知消息，但不会删除该订阅。                                                                                  |
| 订阅名称 | 设置订阅告警的名称。 <ul style="list-style-type: none"><li>名称只能包含大写字母、小写字母、数字、-和_，且必须由大写字母、小写字母或数字开头。</li><li>名称长度为1～256字符。</li></ul> |
| 集群   | 选择订阅的集群：只支持一个集群订阅多个告警。                                                                                                      |

**步骤5** “订阅告警列表”区域显示系统根据订阅设置筛选出的告警。在“消息通知主题名称”下拉框中，选择一个消息通知主题。

如需创建新主题，请单击“**创建新主题**”按钮，系统将跳转到消息通知服务控制台页面。

 说明

所选择的消息通知主题，必须已授予DWS 服务向该主题发布消息的权限。如果所选主题尚未给DWS授权，请前往消息通知服务的主题管理页面设置主题策略授权。设置主题策略时，“可发布消息的服务”需勾选“DWS”。

**步骤6** 确认无误后，单击“确定”，完成创建订阅。

----结束

修改订阅

- 步骤1** 登录DWS 管理控制台。

**步骤2** 在左侧导航树，选择“监控 > 告警”，切换至“订阅”页签。

**步骤3** 在指定订阅名称所在行操作列，单击“编辑”按钮，进入“编辑订阅”页面。

**步骤4** 在编辑订阅页面，选择要更改的参数项进行修改。具体修改方法参见创建订阅中的[步骤4～5](#)。

步骤5 确认无误后，单击“确定”。

----结束

删除订阅

步骤1 登录DWS 管理控制台。

步骤2 在左侧导航树，选择“监控 > 告警”，切换至订阅页签。

步骤3 在指定订阅名称所在行操作列，单击“删除”按钮，弹出确认删除对话框。

步骤4 单击“是”，删除该订阅。

----结束

11.3 查看并订阅 DWS 集群事件

11.3.1 事件通知概述

概述

DWS 使用SMN（Simple Message Notification，消息通知服务）发送DWS 事件的通知，订阅DWS 事件即可启用通知。在订阅中，用户需要指定一个或多个事件筛选条件。每当发生与所有筛选条件匹配的事件时，DWS 就会通过该订阅发送通知。筛选条件包含事件类别（例如：管理、监控或安全）、事件级别（例如：正常或警告）和事件源类型（例如：集群或快照）。

支持的事件类别和事件

事件是用户集群状态发生变化的记录。它可以是由用户操作触发的（比如审计事件），也有可能是集群服务状态变化引起的（比如集群修复成功或集群修复失败）。以下为当前DWS支持的事件和事件类别列表。

- 下表显示了事件源类型为集群的事件

表 11-7 事件源类型为集群的事件

| 事件类别 | 事件名称                 | 事件级别 | 事件      |
|------|----------------------|------|---------|
| 管理   | createClusterFail    | 警告   | 集群创建失败。 |
| 管理   | createClusterSuccess | 正常   | 集群创建成功。 |
| 管理   | createCluster        | 正常   | 开始创建集群。 |
| 管理   | extendCluster        | 正常   | 开始扩容集群。 |
| 管理   | extendClusterSuccess | 正常   | 集群扩容成功。 |
| 管理   | extendClusterFail    | 警告   | 集群扩容失败。 |
| 管理   | deleteClusterFail    | 警告   | 集群删除失败。 |

| 事件类别 | 事件名称                              | 事件级别 | 事件               |
|------|-----------------------------------|------|------------------|
| 管理   | deleteClusterSuccess              | 正常   | 集群删除成功。          |
| 管理   | deleteCluster                     | 正常   | 开始删除集群。          |
| 管理   | restoreClusterFail                | 警告   | 集群恢复失败。          |
| 管理   | restoreClusterSuccess             | 正常   | 集群恢复成功。          |
| 管理   | restoreCluster                    | 正常   | 开始恢复集群。          |
| 管理   | restartClusterFail                | 警告   | 集群重启失败。          |
| 管理   | restartClusterSuccess             | 正常   | 集群重启成功。          |
| 管理   | restartCluster                    | 正常   | 开始重启集群。          |
| 管理   | configureMRSExtDataSources        | 正常   | 开始配置集群的MRS外部数据源。 |
| 管理   | configureMRSExtDataSourcesFail    | 警告   | 配置集群的MRS外部数据源失败。 |
| 管理   | configureMRSExtDataSourcesSuccess | 正常   | 配置集群的MRS外部数据源成功。 |
| 管理   | deleteMRSExtDataSources           | 正常   | 开始删除集群MRS外部数据源。  |
| 管理   | deleteMRSExtDataSourcesFail       | 警告   | 删除集群MRS外部数据源失败。  |
| 管理   | deletedMRSExtDataSourcesSuccess   | 正常   | 删除集群MRS外部数据源成功。  |
| 管理   | bindEipToCluster                  | 正常   | 集群绑定EIP。         |
| 管理   | bindEipToClusterFail              | 警告   | 集群绑定EIP失败。       |
| 管理   | unbindEipToCluster                | 正常   | 集群解绑EIP。         |
| 管理   | unbindEipToClusterFail            | 警告   | 集群解绑EIP失败。       |
| 管理   | refreshEipToCluster               | 正常   | 集群刷新EIP。         |
| 管理   | refreshEipToClusterFail           | 警告   | 集群刷新EIP失败。       |
| 安全   | resetPasswordFail                 | 警告   | 集群密码重置失败。        |
| 安全   | resetPasswordSuccess              | 正常   | 集群密码重置成功。        |
| 安全   | updateConfiguration               | 正常   | 开始更新集群安全参数。      |
| 安全   | updateConfigurationFail           | 警告   | 更新集群安全参数失败。      |
| 安全   | updateConfigurationSuccess        | 正常   | 更新集群安全参数成功。      |

| 事件类别 | 事件名称                 | 事件级别 | 事件           |
|------|----------------------|------|--------------|
| 监控   | repairCluster        | 正常   | 节点故障，开始修复集群。 |
| 监控   | repairClusterFail    | 警告   | 集群修复失败。      |
| 监控   | repairClusterSuccess | 正常   | 集群修复成功。      |

- 下表显示了事件源类型为快照的事件

表 11-8 事件源类型为快照的事件

| 事件类别 | 事件名称                | 事件级别 | 事件      |
|------|---------------------|------|---------|
| 管理   | deleteBackup        | 正常   | 快照删除成功。 |
| 管理   | deleteBackupFail    | 警告   | 快照删除失败。 |
| 管理   | createBackup        | 正常   | 开始创建快照。 |
| 管理   | createBackupSuccess | 正常   | 快照创建成功。 |
| 管理   | createBackupFail    | 警告   | 快照创建失败。 |

- 下表显示了事件源类型为容灾的事件

表 11-9 事件源类型为容灾的事件

| 事件类别 | 事件名称                          | 事件级别 | 事件      |
|------|-------------------------------|------|---------|
| 管理   | beginCreateDisasterRecovery   | 正常   | 开始创建容灾。 |
| 管理   | createDisasterRecoverySuccess | 正常   | 创建容灾成功。 |
| 管理   | createDisasterRecoveryFail    | 警告   | 创建容灾失败。 |
| 管理   | beginStartDisasterRecovery    | 正常   | 开始启动容灾。 |
| 管理   | startDisasterRecoverySuccess  | 正常   | 启动容灾成功。 |
| 管理   | startDisasterRecoveryFail     | 警告   | 启动容灾失败。 |
| 管理   | beginStopDisasterRecovery     | 正常   | 开始停止容灾。 |
| 管理   | stopDisasterRecoverySuccess   | 正常   | 停止容灾成功。 |
| 管理   | stopDisasterRecoveryFail      | 警告   | 停止容灾失败。 |

| 事件类别 | 事件名称                                         | 事件级别 | 事件           |
|------|----------------------------------------------|------|--------------|
| 管理   | beginSwitchoverDisasterRecovery              | 正常   | 开始灾备切换。      |
| 管理   | switchoverDisasterRecoverySuccess            | 正常   | 灾备切换成功。      |
| 管理   | switchoverDisasterRecoveryFail               | 警告   | 灾备切换失败。      |
| 管理   | beginDeleteDisasterRecovery                  | 正常   | 开始删除容灾。      |
| 管理   | deleteDisasterRecoverySuccess                | 正常   | 删除容灾成功。      |
| 管理   | deleteDisasterRecoveryFail                   | 警告   | 删除容灾失败。      |
| 管理   | disasterRecoveryAbnormal                     | 警告   | 容灾运行异常。      |
| 管理   | beginFailoverDisasterRecovery                | 正常   | 开始异常切换。      |
| 管理   | failoverDisasterRecoverySuccess              | 正常   | 异常切换成功。      |
| 管理   | failoverDisasterRecoveryFail                 | 警告   | 异常切换失败。      |
| 管理   | beginRecoveryDisaster                        | 正常   | 开始恢复容灾。      |
| 管理   | recoveryDisasterSuccess                      | 正常   | 恢复容灾成功。      |
| 管理   | recoveryDisasterFail                         | 警告   | 恢复容灾失败。      |
| 管理   | emptyDisasterRecovery                        | 警告   | 当前容灾对象内无容灾表。 |
| 管理   | switchoverContinueAsFailoverDisasterRecovery | 警告   | 灾备切换降级为异常切换。 |

- 下表显示了事件源类型为数据迁移的事件

表 11-10 事件源类型为数据迁移的事件

| 事件类别 | 事件名称                                     | 事件级别 | 事件        |
|------|------------------------------------------|------|-----------|
| 数据迁移 | dataMigrationApplicationDetectedAbnormal | 警告   | 作业任务状态异常。 |
| 数据迁移 | dataMigrationApplicationReturnNormal     | 正常   | 作业任务恢复正常。 |
| 数据迁移 | dataMigrationCreateApplication           | 正常   | 创建作业任务。   |

| 事件类别 | 事件名称                                         | 事件级别 | 事件          |
|------|----------------------------------------------|------|-------------|
| 数据迁移 | dataMigrationCreateCluster                   | 正常   | 开始创建数据迁移实例。 |
| 数据迁移 | dataMigrationCreateClusterFailed             | 警告   | 数据迁移实例创建失败。 |
| 数据迁移 | dataMigrationCreateClusterSuccess            | 正常   | 数据迁移实例创建成功。 |
| 数据迁移 | dataMigrationCreateConnection                | 正常   | 创建连接配置。     |
| 数据迁移 | dataMigrationCreateMapping                   | 正常   | 创建表映射配置。    |
| 数据迁移 | dataMigrationDeleteApplication               | 正常   | 开始删除作业任务。   |
| 数据迁移 | dataMigrationDeleteApplicationFailed         | 警告   | 作业任务删除失败。   |
| 数据迁移 | dataMigrationDeleteApplicationSuccess        | 正常   | 作业任务删除成功。   |
| 数据迁移 | dataMigrationDeleteCluster                   | 正常   | 开始删除数据迁移实例。 |
| 数据迁移 | dataMigrationDeleteClusterApplication        | 正常   | 开始删除作业任务。   |
| 数据迁移 | dataMigrationDeleteClusterApplicationFailed  | 警告   | 作业任务删除失败。   |
| 数据迁移 | dataMigrationDeleteClusterApplicationSuccess | 正常   | 作业任务删除成功。   |
| 数据迁移 | dataMigrationDeleteClusterFailed             | 警告   | 数据迁移实例删除失败。 |
| 数据迁移 | dataMigrationDeleteClusterSuccess            | 正常   | 数据迁移实例删除成功。 |
| 数据迁移 | dataMigrationDeleteConnection                | 正常   | 删除连接配置。     |
| 数据迁移 | dataMigrationDeleteMapping                   | 正常   | 删除表映射配置。    |
| 数据迁移 | dataMigrationDialsConnection                 | 正常   | 测试连接配置。     |
| 数据迁移 | dataMigrationModifyConnection                | 正常   | 修改连接配置。     |

| 事件类别 | 事件名称                                 | 事件级别 | 事件        |
|------|--------------------------------------|------|-----------|
| 数据迁移 | dataMigrationModifyMapping           | 正常   | 修改表映射配置。  |
| 数据迁移 | dataMigrationStartApplication        | 正常   | 开始启动作业任务。 |
| 数据迁移 | dataMigrationStartApplicationFailed  | 警告   | 作业任务启动失败。 |
| 数据迁移 | dataMigrationStartApplicationSuccess | 正常   | 作业任务启动成功。 |
| 数据迁移 | dataMigrationStopApplication         | 正常   | 开始停止作业任务。 |
| 数据迁移 | dataMigrationStopApplicationFailed   | 警告   | 作业任务停止失败。 |
| 数据迁移 | dataMigrationStopApplicationSuccess  | 正常   | 作业任务停止成功。 |

### 11.3.2 订阅事件通知

用户通过订阅DWS 的事件通知，这样便能在特定集群或快照发生管理、监控或安全事件时收到短信、电子邮件或应用等方式的通知消息。

#### 创建订阅

- 步骤1 登录DWS 管理控制台。
- 步骤2 在左侧导航树，选择“管理 > 事件管理”。
- 步骤3 在“事件管理”页面中选择“订阅 > 创建订阅”。
- 步骤4 在“订阅设置”区域，设置订阅基本信息及事件过滤。

“订阅事件列表”区域显示系统根据订阅设置筛选出的事件。

表 11-11 订阅参数

| 参数名  | 参数解释                                                                                                                        |
|------|-----------------------------------------------------------------------------------------------------------------------------|
| 是否开启 | 设置是否开启事件订阅。<br>关闭后停止发送已订阅事件的通知消息，不会删除该订阅。                                                                                   |
| 订阅名称 | 设置订阅事件的名称。 <ul style="list-style-type: none"><li>名称只能包含大写字母、小写字母、数字、-和_，且必须由大写字母、小写字母或数字开头。</li><li>名称长度为1～256字符。</li></ul> |
| 事件类别 | 选择订阅事件的类别：管理，监控或安全。                                                                                                         |

| 参数名   | 参数解释             |
|-------|------------------|
| 事件级别  | 选择事件的告警级别：正常或警告。 |
| 事件源类型 | 选择事件源类型：集群、快照。   |

**步骤5** 在“消息通知主题名称”下拉框中，选择一个消息通知主题。

- 所选择的消息通知主题，必须已授予DWS 服务向该主题发布消息的权限。  
如果所选主题尚未给DWS 授权，请前往消息通知服务的主题管理页面设置主题策略授权。详细操作请参见《消息通知服务用户指南》中的“主题管理 > 设置主题策略”。设置主题策略时，“可发布消息的服务”需勾选“DWS”。
- 如需创建新主题，请单击“创建新主题”，系统将跳转到消息通知服务控制台页面，请参见《消息通知服务用户指南》中的“主题管理 > 创建主题”章节创建新主题。

**步骤6** 单击“确定”，完成创建订阅。

----结束

## 修改订阅

**步骤1** 在DWS 管理控制台的“管理 > 事件管理”页面中，单击“订阅”页签。

**步骤2** 在指定订阅名称所在行的“操作”列，单击“编辑”，进入“编辑订阅”页面。

**步骤3** 在“编辑订阅”页面，选择要更改的参数项进行修改。具体修改方法参见创建订阅中的[步骤4](#)~[步骤6](#)。

----结束

## 删除订阅

**步骤1** 在DWS 管理控制台的“管理 > 事件管理”页面中，单击“订阅”页签。

**步骤2** 在指定订阅名称所在行的“操作”列，单击“删除”，弹出确认删除对话框。

**步骤3** 单击“是”，删除该订阅。

----结束

## 11.3.3 查看事件

介绍用户如何查找集群或快照发生的事件。

**步骤1** 登录DWS 管理控制台。

**步骤2** 在左侧导航树，选择“管理 > 事件管理”，进入事件管理页面。

在事件列表中，默认显示当前所有集群或快照已发生的事件。

在表头的字段名称“时间”旁，单击，可以将事件按时间的顺序或倒叙进行排列。

在事件列表上方，可单击搜索框，在下拉列表中根据筛选条件（时间、事件、事件级别、事件源、事件源类型、事件类别）查找相关事件。

----结束

## 11.4 备份与恢复 DWS 集群

### 11.4.1 快照概述

快照是对DWS集群在某一时间点的一次全量数据或增量数据的备份，记录了当前数据库的数据以及集群的相关信息，其中包括节点数量、节点规格和数据库管理员用户名等。快照创建方式包括手动创建快照和自动创建快照，详情参见[手动快照](#)和[自动快照](#)。

当选择将快照恢复到新集群时，DWS会根据快照记录的集群信息来创建新集群，然后从快照记录的数据中还原数据库信息。从快照恢复数据到新集群的详细步骤请参见[恢复快照到新集群](#)。

当选择将快照恢复到原集群时，DWS会清除当前集群的数据，然后从快照记录的数据中将数据库信息还原到当前集群。从快照恢复数据到原集群的详细步骤请参见[恢复快照到当前集群](#)。

其中，快照备份和恢复速率如下（此速率为实验室测试环境下数据，介质为SSD本地盘，仅供用户参考。在实际使用中，由于磁盘、网络、带宽等因素可能会产生一定的差异）：

- 备份速率：200 MB/s/DN
- 恢复速率：125 MB/s/DN

### 约束与限制

- 对于有数据可靠性要求的用户，尤其是针对业务方无法通过上游重新导入进行数据恢复的场景，集群需要备份操作，避免人为或其他因素造成数据丢失后无法恢复。
- Schema粒度快照支持集群版本如下所示（如果当前控制台界面不支持该功能，请联系技术支持人员）。
  - 9.1.0.100及以上集群版本。
  - 8.3.0.110及以上的8.3.0.xxx集群版本。
  - 8.2.1.230及以上的8.2.1.2xx集群版本。
- OBS快照存储空间：
  - DWS提供的免费存储空间等于集群存储空间，即单节点存储空间大小 \* 节点数。
- 快照服务依赖及部署说明：
  - 快照管理功能依赖于OBS、NFS的备份介质。
  - 备份设备为NFS的备份介质时，使用挂载盘的方式，依赖高性能弹性文件服务 SFS-Turbo。具体配置步骤请参见[11.1.3.2-自动快照策略](#)配置备份介质。
  - 当前快照恢复到新集群时仅支持OBS介质。
- DWS根据快照创建的新集群与生成快照的原始集群具有相同的配置，即节点的数量和规格、内存、磁盘与原集群一致。

- 根据快照创建新集群时，如果没有指定其他值，则参数默认与生成快照时的备份信息保持一致。
- 快照生成期间，请避免进行Vacuum Full操作，否则可能会导致集群只读。
- 创建快照时因备份数据会降低磁盘I/O性能，建议在业务相对空闲的时期进行快照操作。
- 快照期间会保留一些中间文件，需额外占用磁盘空间，因此请避开业务高峰期并保证磁盘容量在70%以下。

## 11.4.2 手动快照

### 11.4.2.1 创建集群粒度的手动快照

#### 前提条件

集群快照是DWS集群在某一时间点的完整备份，记录了这一时刻指定集群的所有配置数据和业务数据。用户根据业务需要备份集群数据时，可以在“快照管理”页面创建集群的快照。

手动快照可以随时创建，在创建成功后会一直保存，直到在DWS控制台将此快照删除。由于创建手动快照采用全量备份的方式，因此备份时间较长。

#### 说明

- 手动创建的集群粒度快照支持备份到OBS服务、NFS介质。
- 待创建集群粒度快照的集群状态必须为“可用”、“待重启”或“非均衡”中的任意一种，当集群版本低于8.1.3.101时，“只读”状态的集群也可创建集群粒度的快照。

#### 系统影响

正在创建快照的集群，暂时无法提供完整服务，如无法执行重启、扩容、重置密码、修改集群配置信息等操作。

#### 说明

为了保证快照数据的完整性，建议创建快照时暂停写入新数据。

#### 操作步骤

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，单击“**管理 > 快照管理**”（或在集群列表中，单击指定集群的名称，进入“**集群详情**”页面，左导航栏切换至“**快照**”页签）。

**步骤3** 进入“快照管理”页面，单击右上角的“**创建快照**”，也可在集群管理页面选择指定集群所在行操作列“**更多 > 创建快照**”。

**步骤4** 在快照创建页面完善快照信息：

- “**集群名称**”：选择一个指定的DWS 集群。只有状态为“可用”的集群会在下拉列表中显示。
- “**快照名称**”：填写快照的名称。快照名称长度为4~64个字符，必须以字母开头，不区分大小写，可以包含字母、数字、中划线或者下划线，不能包含其他的特殊字符。

- “快照级别”：选择快照的级别，创建集群粒度的快照，快照级别需要选择为“cluster”。
- “快照描述”：填写快照的描述信息。此参数为可选参数。快照描述的字符长度为0~256，不支持特殊字符!<>'=&"。

**步骤5** 确认无误后，单击“立即创建”，集群开始创建快照。

创建快照时，正在创建快照的集群任务状态变为“创建快照中”。同时正在创建的快照，状态显示为“创建中”。快照创建完成后，快照的状态为“可用”。

#### 说明

如果快照大小明显大于集群当前存储数据的大小，可能存在数据仅标记为删除但未真实清理回收，建议执行vacuum并重新创建快照，详情请参见[如何清理与回收DWS存储空间?](#)。

----结束

## 11.4.2.2 创建 schema 粒度的手动快照

### Schema 粒度快照概述

schema快照是DWS集群在某一时间点某些schema集的数据备份，记录了这一时刻集群指定schema的所有业务数据。用户根据业务需要备份集群的schema数据时，可以在“快照管理”页面创建集群的schema粒度快照。

手动创建的细粒度快照可以随时创建，在创建成功后会一直保存，直到在DWS控制台将此快照删除。

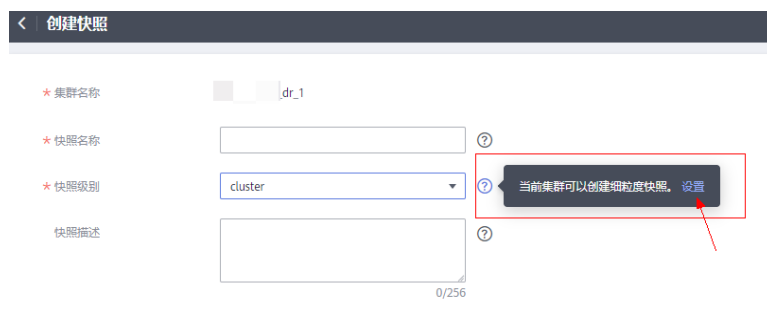
#### 说明

- 如果当前控制台界面不支持该功能，请联系技术支持人员。
- 手动创建的集群的schema快照支持备份到OBS服务、NFS的介质。
- 待创建schema粒度快照的集群状态必须为“可用”或“非均衡”。

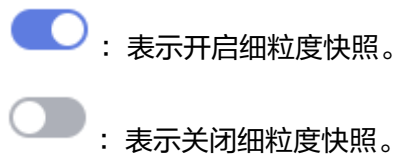
### 前提条件

用户需手动打开细粒度快照的开关：

- 步骤1** 在左侧导航栏中，单击“管理 > 快照管理”（或在集群列表中，单击指定集群的名称，进入“集群详情”页面，左导航栏切换至“快照”页签）。
- 步骤2** 进入“快照管理”页面，单击右上角的“创建快照”，也可在集群管理页面选择指定集群所在行操作列“更多 > 创建快照”。
- 步骤3** 在快照级别栏单击，并单击“设置”。



**步骤4** 在快照列表页面单击“细粒度快照开关”，打开或关闭细粒度快照。



快照管理

策略列表

细粒度快照开关: ☒

删除策略

| 快照名称        | 快照状态 | 备份方式 | 快照类型 | 快照名称 | 快照描述    | 快照创建时间                       | 操作       |
|-------------|------|------|------|------|---------|------------------------------|----------|
| 20220901152 | 可用   | 普通   | 增量   | 085  | cluster | 2022/09/01 23:29:56 GAT-0800 | 查看 删除 更多 |
| 20220901072 | 可用   | 普通   | 增量   | 085  | cluster | 2022/09/01 15:29:27 GAT-0800 | 查看 删除 更多 |
| 20220903032 | 可用   | 普通   | 增量   | 085  | cluster | 2022/09/01 07:29:27 GAT-0800 | 查看 删除 更多 |
| 20220903012 | 可用   | 普通   | 增量   | 085  | cluster | 2022/09/01 23:29:27 GAT-0800 | 查看 删除 更多 |
| 20220902072 | 可用   | 普通   | 增量   | 085  | cluster | 2022/09/01 15:29:27 GAT-0800 | 查看 删除 更多 |
| 20220919332 | 可用   | 普通   | 增量   | 085  | cluster | 2022/09/01 07:29:27 GAT-0800 | 查看 删除 更多 |
| 20220919152 | 可用   | 普通   | 增量   | 085  | cluster | 2022/09/19 23:29:27 GAT-0800 | 查看 删除 更多 |
| 20220919102 | 可用   | 普通   | 增量   | 085  | cluster | 2022/09/19 20:22:58 GAT-0800 | 查看 删除 更多 |
| 20220919072 | 可用   | 普通   | 增量   | 085  | cluster | 2022/09/19 15:29:27 GAT-0800 | 查看 删除 更多 |

**说明**

- 开启细粒度快照开关后，创建的自动或手动快照可用于进行表恢复。

----结束

系统影响

正在创建快照的集群，暂时无法提供完整服务，如无法执行重启、扩容、重置密码、修改集群配置信息等操作。

**说明**

为了保证快照数据的完整性，建议创建快照时暂停写入新数据。

操作步骤

- 步骤1** 登录DWS 管理控制台。
- 步骤2** 在左侧导航栏中，单击“**管理 > 快照管理**”（或在集群列表中，单击指定集群的名称，进入“**集群详情**”页面，左导航栏切换至“**快照**”页签）。
- 步骤3** 进入“快照管理”页面，单击右上角的“**创建快照**”，也可在集群管理页面选择指定集群所在行操作列“**更多 > 创建快照**”。
- 步骤4** 在快照创建页面完善快照信息：
  - “**集群名称**”：选择一个指定的DWS 集群。只有状态为“可用”的集群会在下拉列表中显示。
  - “**快照名称**”：填写快照的名称。快照名称长度为4~64个字符，必须以字母开头，不区分大小写，可以包含字母、数字、中划线或者下划线，不能包含其他的特殊字符。
  - “**快照级别**”：选择快照的级别，创建Schema粒度的快照，快照级别需选择为“schema”。
  - “**快照描述**”：填写快照的描述信息。此参数为可选参数。快照描述的字符长度为0~256，不支持特殊字符!<>'=&"。
- 步骤5** 快照级别选择为schema后，页面下方将会显示schema级别的快照选择栏。

- 数据库选项下拉框中选择具体的数据库。
- 在schema选择栏，用户可以勾选具体需要备份的schema，可在schema列表右上角的搜索框中，输入指定schema名称并单击🔍查询。支持模糊搜索名称的部分字符。

#### 📖 说明

- schema粒度的快照每次备份只能备份同一个数据库里的schema。
- 单次备份可备份的schema数量默认为50。

**步骤6** 确认无误后，单击“立即创建”，集群开始创建快照。

创建快照时，正在创建快照的集群任务状态变为“创建快照中”。同时正在创建的快照，状态显示为“创建中”。快照创建完成后，快照的状态为“可用”。

#### 📖 说明

如果快照大小明显大于集群可用存储空间大小，可能存在数据仅标记为删除但未真实清理回收，建议执行清理并重新创建快照，详情请参见[如何清理与回收DWS存储空间？](#)。

----结束

### 11.4.2.3 删除手动快照

用户需要删除状态为“不可用”的快照，或者需要释放快照所占用的存储空间时，可以在“快照管理”中删除不再使用的快照。

#### ⚠️ 注意

成功删除的快照无法恢复，请谨慎操作。

## 操作步骤

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，单击“**管理 > 快照管理**”（或在集群列表中，单击指定集群的名称，进入“**集群详情**”页面，左导航栏切换至“**快照**”页签）。默认显示用户所有的快照。

**步骤3** 在快照列表中，在需要删除的快照所对应的“操作”列，单击“更多 > 删除”按钮。

### 说明

用户可以对手动快照进行删除，自动快照不支持删除。

**步骤4** 确认无误后，输入“DELETE”，并单击“确定”，删除此快照。

----结束

## 11.4.3 自动快照

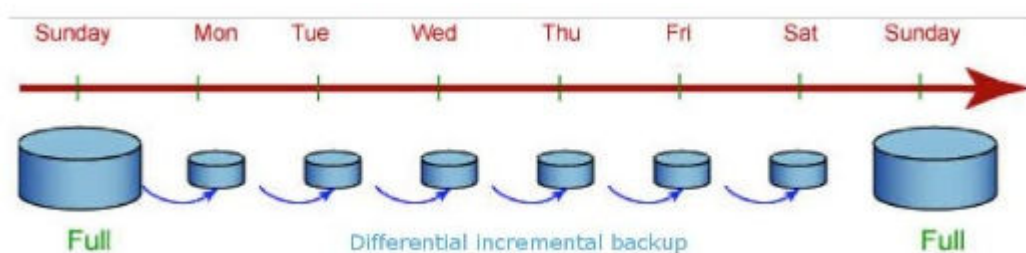
### 11.4.3.1 自动快照概述

自动快照采用差异增量备份，第一次创建自动快照为全量备份，以后每间隔一段时间做一次全量备份，全量备份作为基础版本。两次全量备份之间都是做增量备份，增量备份基于前一次备份所发生的更改进行记录。

在恢复快照时，DWS会将最近一次的全量备份到本次备份之间的所有备份一起用于恢复集群，因此不会产生数据丢失。

为了保证每个增量快照都能够正常进行数据恢复，如果增量快照的保存时间超过了保留天数，DWS不会立即删除过期的增量快照，而是仍然保留这部分快照，以便提供给后续其他增量快照恢复集群时使用，直到完成下一次全量快照，DWS才会一并删除已过期的上一次的全量自动快照和相关的增量快照。

图 11-1 快照备份过程



集群创建成功后，自动快照默认处于启用状态。当集群启用了自动快照时，DWS将按照设定的时间和周期以及快照类型自动创建快照，默认为每8小时做一次增量备份，每周做一次全量备份。用户也可以对集群设置自动快照策略，并根据自身需求，对集群设置一个或多个自动快照策略。如果用户未配置全量备份策略，则每15次增量备份做一次全量备份。DWS管理控制台设置集群的自动快照策略，具体操作请参见[设置自动快照策略](#)。

自动快照有保留期，可设置为1~31天，默认为7天，系统会在保留期结束时删除到期快照。该保留期为用户可见快照的保留期，如果存在增量快照未到期，为保证快照可用，前置增量快照和全量快照不会立即物理删除，已到期的快照为隐藏状态，用户不

可见。待全部增量快照都到期后，前置隐藏的快照才实际物理删除。如果需要将自动快照保留更长时间，可以创建一份副本作为手动快照。自动快照会保留至保留期结束，而手动快照在手动删除前将会一直保留。如何复制自动快照，详情见[复制自动快照](#)。

### 11.4.3.2 设置自动快照策略

用户可根据自身需求，选择快照类型，并对集群设置一个或至多3个自动集群级快照策略和一个或至多20个schema级快照策略。

自动快照策略开启后，系统将按照设定的时间和周期以及快照类型自动创建快照。

#### 操作步骤

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，选择“专属集群 > 集群列表”。

**步骤3** 在集群列表中，单击指定集群的名称，进入“集群详情”页面。

**步骤4** 切换至“快照”页签，再单击二级页签中的“策略列表”。在“策略列表”页面，会展示当前集群所有的策略（包括默认的自动快照策略）。

**步骤5** （可选）单击“自动快照”开关，开启自动快照策略。

-  表示开启自动创建快照策略，默认为开启状态，保留天数默认为7天。
-  表示关闭自动创建快照策略，关闭后设置的快照策略不会触发自动快照。

**步骤6** 设置自动快照的保留天数以及修改当前集群使用的备份设备，各参数配置原则如[表 11-12](#)所示。

表 11-12 自动快照参数说明

| 参数名                | 参数解释                                                                              |
|--------------------|-----------------------------------------------------------------------------------|
| 保留天数               | 设置自动创建的快照的保留天数，可设置范围为1~31天。<br><b>说明</b><br>用户不允许手动删除自动创建的快照，自动快照保留天数超期后，系统会自动删除。 |
| 备份设备               | 下拉框，支持OBS、NFS。                                                                    |
| 备份NFS文件系统地址（NFS参数） | NFS的共享地址IP。填写弹性文件服务共享路径IP地址，挂载成功后默认在集群实例的“/var/chroot/nfsbackup”目录下创建挂载目录。        |

**步骤7** 自动创建快照策略后，可以对参数进行设置，各参数配置原则如[表11-13](#)所示。

- 快照类型为集群级或Schema级全量快照时，快照策略可选择一次性和周期性，其中Schema级别全量快照需要选择对应的数据库，且一次支持备份单个或者至多20个Schema：
  - 设置集群级或Schema级全量周期性快照策略，可指定星期或日期，选择触发时间点。

**警告**

- 29号、30号、31号为月末日期，会存在漏备，请谨慎选择。具体策略和执行请以实际月份日期为准。
  - 快照策略时间为本地时间。
- 
- 设置集群级或Schema级全量一次性快照策略，可指定具体日期和触发时间。
  - 快照类型为集群级或Schema级增量快照时，快照策略只能选择周期性。  
设置集群级或Schema级增量周期性快照策略，可以指定星期或日期，并且可以选择触发时间点，或设置开始时间以及时间间隔。

**表 11-13** 快照策略参数说明

| 参数名       | 参数解释                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 快照策略名称    | 策略名称要求在4位到92位之间，必须以字母开头，不区分大小写，可以包含字母、数字、中划线或者下划线，不能包含其他特殊字符，并且名称唯一。                                                                                                                                                                                                                                                                                                                                                                                                          |
| 快照级别      | 支持集群和Schema级别快照。                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| 快照类型      | 可选择全量和增量。<br><b>说明</b> <ul style="list-style-type: none"><li>• 15次增量快照会触发一次全量快照。</li><li>• 增量快照恢复是以全量快照为基础，增量快照为辅恢复数据到增量快照时间点的全部数据。</li><li>• 增量快照记录基于前一次快照所发生的更改，备份时间快；全量快照是对整个集群的数据进行备份，时间较长。在恢复快照期间，DWS会将最近一次的全量快照到本次快照之间的所有快照一起用于恢复集群。</li><li>• Schema级增量自动策略支持版本如下：<ul style="list-style-type: none"><li>- 9.1.0.100及以上集群版本。</li><li>- 8.3.0.110及以上的8.3.0.xxx集群版本。</li><li>- 8.2.1.230及以上的8.2.1.2xx集群版本。</li></ul></li><li>• Schema级策略不支持修改策略级别、数据库及Schema属性。</li></ul> |
| 快照策略      | 可选择周期性或一次性。<br><b>说明</b><br>只有快照类型选择全量时，才可选择一次性快照策略。                                                                                                                                                                                                                                                                                                                                                                                                                          |
| 一次性快照策略设置 | 可设置在未来某一个具体时间点创建一次全量备份，时间为本地时间。                                                                                                                                                                                                                                                                                                                                                                                                                                               |

| 参数名    | 参数解释                                                                                                                                                                                                                                                                                                                             |
|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 快照执行周期 | <p>可根据需求设置周期性的快照触发策略：</p> <ul style="list-style-type: none"><li>● 天数设置：可以指定星期或指定日期，选择星期后不可再选择日期。在选择日期时，若当前月份没有所选日期，直接顺延到之后的月份。</li><li>● 时间设置：可以选择指定时间，含义为指定周期性策略当天具体的触发时间。增量快照类型可以指定开始时间以及间隔，可选间隔集群级为4-24小时（Schema级间隔为1-24小时），表示从某个时间开始每隔多少小时做一次快照。</li></ul> <p><b>须知</b><br/>当增量数据量较大时，如果设置的备份周期太长会出现备份慢的情况，建议可以适当增加备份频率。</p> |

**步骤8** 确认无误后，单击“确定”。

**步骤9** （可选）用户可根据自身需求在指定自动快照策略中单击“修改”按钮对已经开启的自动创建快照策略进行修改。

**步骤10** （可选）用户可单击“预览快照策略”按钮，预览当前集群使用中的快照策略，显示集群未来7次的快照情况。如果集群没有设置全量策略，系统将默认使用每进行15次增量快照进行一次全量快照的策略。

#### 须知

由于集群的操作不可预期，需注意：

- 关闭自动快照开关，快照预览策略页面将不显示设置的自动策略。
- 关闭快照列表的细粒度快照开关，快照预览策略页面将不显示Schema级别自动策略。
- 参考策略预览时间，集群将在预设时间1小时内触发快照任务。
- 当集群在扩容、升级、修改快照介质等操作之后，进行下一次自动快照时，系统默认做全量自动快照。
- 当集群使用周期性策略时，上次自动快照结束后4小时内不允许再次自动备份，不符合要求的触发时间将被跳过。
- 当多个策略的下次触发时间发生冲突时，优先级关系为集群级 > Schema级，一次性>周期性、全量>增量。
- 当进行备份恢复时，支持从任意一种快照类型恢复资源的全量数据，无论快照类型是全量快照还是增量快照。
- 当使用Schema级自动增量策略时，与集群级全量和增量策略交替使用时会在满足条件时自动将schema增量快照转为全量快照，建议用户合理搭配策略周期使用。

----结束

### 11.4.3.3 复制自动快照

用户可以对自动快照进行复制，生成手动快照以便长期保留。

## 复制自动快照

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，选择“管理 > 快照管理”。

默认显示用户所有的快照。用户可以对自动快照进行复制。

**步骤3** 在快照列表中，在需要复制快照所对应的“操作”列，单击“更多 > 复制”按钮。

- “快照名称”：填写新快照的名称。  
快照名称长度为4~64个字符，必须以字母开头，不区分大小写，可以包含字母、数字、中划线或者下划线，不能包含其他的特殊字符。
- “快照描述”：填写快照的描述信息。  
此参数为可选参数。快照描述的字符长度为0~256，不支持特殊字符!<>'=&"。

**步骤4** 单击“确定”，开始为集群复制快照。

系统提示“快照复制下发成功”。快照复制完成后，复制快照的状态为“可用”。

### 说明

如果快照大小明显大于集群当前存储数据的大小，可能存在数据仅标记为删除但未真实清理回收，建议清理回收存储空间后重新创建快照，参见[如何清理与回收DWS存储空间?](#)。

----结束

### 11.4.3.4 删除自动快照

自动快照不支持手动删除，仅DWS系统能够删除自动快照。

DWS会在如下情况下删除自动快照：

- 自动快照保留期结束。
- 删除集群。

### 注意

为防止用户误执行删除集群操作，DWS将提供以下策略（该功能仅在8.2.0及以上集群版本支持）：

- 若最新一次的快照为自动快照，则系统会将自动快照保留一天时间。
- 若最新一次的快照为手动快照，则会删除相应集群已有的自动快照。

## 11.4.4 查看快照信息

用户创建快照后，可以在“快照管理”页面查看集群快照的信息。

### 查看快照信息

**步骤1** 登录DWS 管理控制台。

**步骤2** 在左侧导航栏中，选择“管理 > 快照管理”。

在快照列表中，默认显示用户所有的快照。

**步骤3** 在快照列表中，查看快照的“快照名称”、“快照状态”、“集群名称”、“备份方式”、“快照类型”、“快照介质”和“快照创建时间”。

用户也可以在快照列表的右上角，输入快照名称或集群名称并单击，搜索指定的快照。支持模糊搜索名称的部分字符。

快照状态如表11-14所示。

表 11-14 快照状态说明

| 状态    | 说明            |
|-------|---------------|
| “可用”  | 表示当前快照工作正常。   |
| “创建中” | 表示当前快照正在创建中。  |
| “不可用” | 表示当前快照无法提供服务。 |

备份方式如表11-15所示。

表 11-15 备份方式

| 类型 | 说明                                                            |
|----|---------------------------------------------------------------|
| 手动 | 用户通过DWS管理控制台或者API接口手动创建的快照。用户可以删除手动快照。                        |
| 自动 | 用户为集群开启快照自动备份策略后，系统自动创建的快照。用户不能删除自动快照，自动快照的保留天数超期后，系统会自动进行删除。 |

快照类型如表11-16所示。

表 11-16 快照类型

| 类型 | 说明        |
|----|-----------|
| 全量 | 该快照为全量备份。 |
| 增量 | 该快照为增量备份。 |

快照介质如表11-17所示。

表 11-17 快照介质

| 快照介质 | 说明                                           |
|------|----------------------------------------------|
| OBS  | 策略中配置备份设备为：OBS，则创建出的快照为OBS快照，备份数据保存在OBS服务器上。 |

| 快照介质 | 说明                                           |
|------|----------------------------------------------|
| NFS  | 策略中配置备份设备为：NFS，则创建出的快照为NFS快照，备份数据保存在NFS服务器上。 |

快照级别如表11-18所示。

表 11-18 快照级别

| 快照级别    | 说明                                         |
|---------|--------------------------------------------|
| cluster | 表示当前快照为集群粒度的快照，备份某一时刻集群所有的配置数据和业务数据。       |
| schema  | 表示当前快照为Schema粒度的快照，备份某一时刻集群某些schema级的业务数据。 |

----结束

根据表名查询快照信息

前提条件

打开细粒度快照开关之后创建的快照才支持细粒度搜索。

说明

仅8.2.1.230及以上集群版本支持根据表名查询快照信息。

操作步骤

- 步骤1 登录DWS管理控制台。
- 步骤2 在左侧导航栏中，选择“专属集群 > 集群列表”。
- 步骤3 在集群列表中，单击指定集群的名称，进入“集群详情”页面。
- 步骤4 切换至“快照>快照列表”页签，开启细粒度快照开关。并单击右边“高级搜索”按钮。

出现数据库和Schema搜索框和表名输入框，可组成“database.schema.table”三元组，用于查询指定数据库模式表下的细粒度快照信息。快照信息详情请参见步骤3。

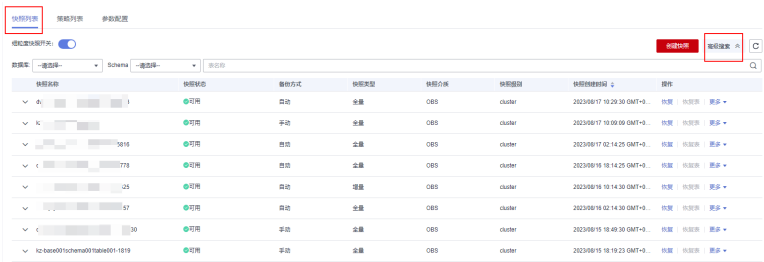


表 11-19 三元组对应含义

| 元组名      | 含义    |
|----------|-------|
| database | 数据库名。 |
| schema   | 模式名。  |
| table    | 表名。   |

----结束

## 11.4.5 恢复快照

### 11.4.5.1 恢复快照约束

#### 集群粒度快照恢复

集群粒度恢复分为两个主要步骤：

1. 恢复数据：即备份工具将备份集中的数据并行的恢复到各个实例的数据目录中，实例包括主CN，主DN等。
2. 重建备DN：即主DN恢复完之后，备DN使用全量Build的方式重建出来，这个过程也是并行执行。

#### 说明

- 综上恢复过程要比备份过程耗时要久一些，通常需要备份时间1.5~2倍的耗时。
- 集群级恢复后参数与备份时一致，恢复到新集群时需保证和原集群规格一致，如果原集群有规格变更操作，新集群需与原集群变更前的规格保持一致，如果新集群规格小的话可能会导致恢复失败。

### 11.4.5.2 恢复快照到新集群

#### 操作场景

用户需要查看集群过去某个时刻的快照数据时，必须先恢复快照到新集群。

恢复快照到新集群时，恢复时长是由快照备份的数据量所决定的。如果快照备份的数据量大，恢复就比较慢。如果快照备份的数据量小，恢复就相对比较快。

自动快照是增量备份，在恢复快照时，DWS会将最近一次的全量备份到本次快照之间的所有快照一起用于恢复集群。自动快照的备份频率是可以设置的，如果一周只备份一次，当增量数据量较大时，就会出现备份慢的情况，因此，建议适当地增加备份频率。

**须知**

- 恢复快照到新集群的功能目前仅支持备份设备为OBS的快照。
- 恢复快照时默认创建一个与原始集群规格相同、节点数也相同的新集群。
- 当前恢复到新集群操作可保证原业务不中断。
- 暂不支持细粒度恢复绝对表空间或相对表空间下的表。
- 不支持恢复逻辑集群与资源池到新集群。

**前提条件**

- 恢复到新集群时创建所需的资源小于或等于用户剩余配额。
- 用户进行恢复快照到新集群操作时，请确保快照状态为“可用”状态。

**操作步骤**

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，选择“**管理 > 快照管理**”（或在集群列表中，单击指定集群的名称，进入“**集群详情**”页面，左导航栏切换至“**快照**”页签）。默认显示用户所有的快照。

**步骤3** 在快照列表中，找到所需恢复的快照，在该快照所在行的“操作”列，单击“恢复”按钮。

**步骤4** 进入“恢复快照”页面，选择“新集群”并填写新集群的配置参数。

- 选择恢复到单AZ集群。
- 选择恢复到多AZ集群。

**说明**

- 多AZ恢复集群只支持8.2.0.100以上集群版本。
- 目前只支持存算一体集群。
- 当前region的可用分区数量 $\geq 3$ 。
- 节点数量和CN数量须满足3的倍数。
- 多AZ恢复集群只支持DN数量 $\leq 2$ 的规格。

恢复快照时，参数支持重新定义，具体请参见[表11-20](#)。其他参数默认与快照中的备份信息保持一致，具体请参见[表11-13](#)。

**表 11-20** 新集群参数配置说明

| 配置类型 | 配置名称                                                                                                                                                |
|------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
| 基础配置 | 可用分区、节点规格、集群名称、数据库端口、虚拟私有云、子网、安全组、公网访问、企业项目。                                                                                                        |
| 高级配置 | 当配置为“自定义”时，设置以下参数： <ul style="list-style-type: none"><li>● 备份设备：下拉框，支持OBS和NFS。</li><li>● 标签：用于标识集群的键值对，有关标签的更多信息，请参见<a href="#">标签简介</a>。</li></ul> |

**步骤5** 单击“立即恢复”，进入“规格确认”页面。

**步骤6** 规格确认无误后，单击“提交”，开始恢复快照到新集群。

等待新集群状态变为“可用”，表示快照已恢复成功。

集群快照恢复成功后，新集群的内网地址和弹性IP地址（如果“公网访问”设置为“自动分配”）会重新分配。

----结束

## 查看恢复详情

**步骤1** 登录DWS管理控制台。

**步骤2** 选择“专属集群 > 集群列表”，默认显示用户所有的集群列表。

**步骤3** 集群列表中，集群状态为“恢复中”，单击下方“查看详情”按钮。

**步骤4** 进入“任务详情”页面，用户可查看集群此时的恢复快照进度。

### 说明

- 任务详情预估时长仅作为参考，实际时长依赖当前数据量大小。
- “执行恢复”阶段单击“查看”可查看内核恢复过程。需要注意的是，任务详情的任务时间与内核执行时间由于任务调度和集群重启等原因，存在一定的时间间隔。

----结束

### 11.4.5.3 恢复快照到当前集群

## 操作场景

恢复到原集群时，可以指定快照恢复到本集群。通常在集群故障或者需要将数据回滚到指定快照版本时，用户使用该功能。

### 须知

- 该功能仅8.1.3.200及以上集群版本支持。
- 该功能目前支持备份设备为OBS的快照。
- 快照状态为“可用”时，用户才可以进行快照恢复。
- 不支持恢复逻辑集群与资源池到当前集群。

## 操作步骤

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，选择“管理 > 快照管理”（或在集群列表中，单击指定集群的名称，进入“集群详情”页面。左导航栏切换至“快照”页签）。默认显示用户所有的快照。

**步骤3** 在快照列表中，找到所需恢复的快照，在该快照所在行的“操作”列，单击“恢复”按钮。

**步骤4** 选择对应的快照进行就地恢复。

#### 说明

快照就地恢复期间，集群不可用。

----结束

### 11.4.5.4 恢复单表到原集群

#### 操作场景

集群粒度的快照和schema粒度的快照，可以指定快照备份集中的单表恢复到原集群。通常用户在业务操作中，对某个表中的数据进行了错误操作或者误删，需要对此表数据回滚到指定快照时，使用该功能。

#### 须知

- 如果当前控制台界面不支持该功能，请联系技术支持人员。
- 恢复单表到原集群的功能支持备份设备为OBS的快照。
- 该功能目前支持备份级别为“cluster”以及“schema”的快照。
- 集群状态和快照状态都为“可用”时，用户才可以进行单表恢复。
- 不支持状态为“只读”的集群进行单表恢复。
- 暂不支持细粒度恢复绝对表空间或相对表空间下的表。

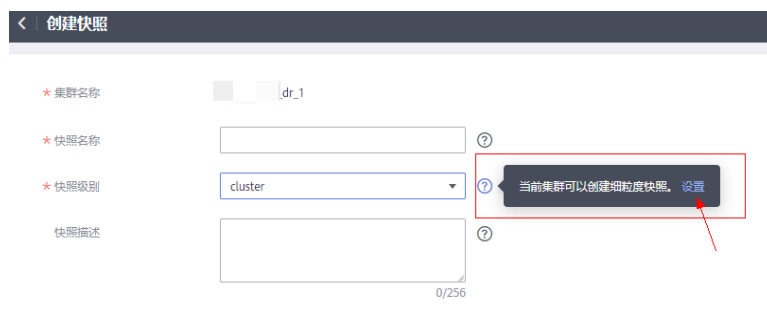
#### 前提条件

用户需手动打开细粒度快照的开关：

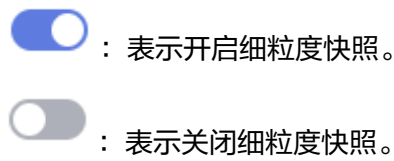
**步骤1** 在左侧导航栏中，单击“管理 > 快照管理”（或在集群列表中，单击指定集群的名称，进入“集群详情”页面，左导航栏切换至“快照”页签）。

**步骤2** 进入“快照管理”页面，单击右上角的“创建快照”，也可在集群管理页面选择指定集群所在行操作列“更多 > 创建快照”。

**步骤3** 在快照级别栏单击，并单击“设置”。



**步骤4** 在快照列表页面单击“细粒度快照开关”，打开或关闭细粒度快照。



| 快照名称        | 快照状态 | 备份方式 | 快照类型 | 快照名称 | 快照策略    | 快照创建时间                        | 操作         |
|-------------|------|------|------|------|---------|-------------------------------|------------|
| 20220918152 | 成功   | 自动   | 增量   | OBS  | cluster | 2022/09/18 22:29:56 GMT+08:00 | 恢复 快照策略 更多 |
| 20220918152 | 成功   | 自动   | 增量   | OBS  | cluster | 2022/09/18 22:29:56 GMT+08:00 | 恢复 快照策略 更多 |
| 20220918152 | 成功   | 自动   | 增量   | OBS  | cluster | 2022/09/18 22:29:56 GMT+08:00 | 恢复 快照策略 更多 |
| 20220918152 | 成功   | 自动   | 增量   | OBS  | cluster | 2022/09/18 22:29:56 GMT+08:00 | 恢复 快照策略 更多 |
| 20220918152 | 成功   | 自动   | 增量   | OBS  | cluster | 2022/09/18 22:29:56 GMT+08:00 | 恢复 快照策略 更多 |
| 20220918152 | 成功   | 自动   | 增量   | OBS  | cluster | 2022/09/18 22:29:56 GMT+08:00 | 恢复 快照策略 更多 |
| 20220918152 | 成功   | 自动   | 增量   | OBS  | cluster | 2022/09/18 22:29:56 GMT+08:00 | 恢复 快照策略 更多 |
| 20220918152 | 成功   | 自动   | 增量   | OBS  | cluster | 2022/09/18 22:29:56 GMT+08:00 | 恢复 快照策略 更多 |
| 20220918152 | 成功   | 自动   | 增量   | OBS  | cluster | 2022/09/18 22:29:56 GMT+08:00 | 恢复 快照策略 更多 |
| 20220918152 | 成功   | 自动   | 增量   | OBS  | cluster | 2022/09/18 22:29:56 GMT+08:00 | 恢复 快照策略 更多 |

**说明**

- 开启细粒度快照开关后，创建的自动或手动快照可用于进行表恢复。

----结束

操作步骤

- 步骤1** 登录DWS管理控制台。
- 步骤2** 在左侧导航栏中，选择“管理 > 快照管理”。
- 步骤3** 快照列表中，在指定快照所在行“操作”列，单击“恢复表”，进入恢复表页面。
- 步骤4** 在恢复表页面需要输入以下内容：
  - 数据库：集群粒度的快照需要选择指定的数据库，schema粒度的快照此选项为备份时所选数据库，详情请参见[创建集群粒度的手动快照](#)和[创建schema粒度的手动快照](#)。
  - 源Schema：要恢复的表所在的schema。
  - 源表：要恢复的表的名称。
  - 目标Schema：恢复的新表所在的schema。
  - 目标表：恢复的新表的名称。

**注意**

- 输入的表名需要满足DWS数据库名称约束：长度不超过63个字符，以字母或下划线开头，中间字符可以是字母、数字、下划线，区分大小写。
- 确保要恢复的源表在备份集中存在，否则会导致恢复操作失败。
- 输入的目标表若在数据库中已存在，此表将会被覆盖，用户请谨慎操作。

**步骤5** 确认无误后，单击“立即恢复”按钮，集群开始对选择的表进行恢复。

----结束

### 11.4.5.5 恢复单表/多表到新集群

#### 操作场景

集群粒度的快照和Schema粒度的快照，可以指定快照备份集中的单表/多表恢复到新集群。通常用户在业务操作中，对某个表中的数据进行了错误操作或者误删，需要对此表找回数据时使用该功能，找到最新包含该表数据的快照，将数据恢复到一个新集群中。在不破坏原集群表数据的同时，对比新老集群的数据，针对性恢复数据。

须知

- 该功能仅9.1.0及以上集群版本支持。仅支持OBS介质。
- 低版本的集群细粒度快照，可跨版本恢复到9.1.0集群版本的新集群中。
- 9.1.0版本集群细粒度快照，可恢复到9.1.0版本的异构新集群中，即新老集群在节点数和规格上都可以不一致。
- 仅支持细粒度的单表/多表快照恢复到新集群。

#### 前提条件

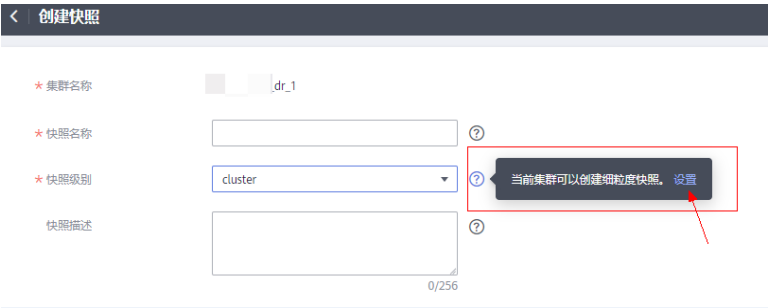
用户需手动打开细粒度快照的开关：

- 步骤1

在左侧导航栏中，单击“管理 > 快照管理”（或在集群列表中，单击指定集群的名称，进入“集群详情”页面，左导航栏切换至“快照”页签）。
- 步骤2

进入“快照管理”页面，单击右上角的“创建快照”，也可在集群管理页面选择指定集群所在行操作列“更多 > 创建快照”。
- 步骤3

在快照级别栏单击，并单击“设置”。



- 步骤4

在快照列表页面单击“细粒度快照开关”，打开或关闭细粒度快照。

- 

: 表示开启细粒度快照。
- 

: 表示关闭细粒度快照。



说明

- 开启细粒度快照开关后，创建的自动或手动快照可用于进行表恢复。

----结束

操作步骤

- 步骤1** 登录DWS管理控制台。
- 步骤2** 在左侧导航栏中，选择“**管理 > 快照管理**”（或在集群列表中，单击指定集群的名称，进入“**集群详情**”页面。左导航栏切换至“**快照**”页签）。默认显示用户所有的快照。
- 步骤3** 在快照列表中，找到所需恢复的快照，在该快照所在行的“**操作**”列，单击“**恢复**”按钮。
- 步骤4** 恢复级别选择“**表级**”。
- 步骤5** 选择恢复新集群的基本信息，详情请参见[创建DWS存算一体集群](#)。

说明

- 选择节点规格和节点数量时，在支持细粒度异构恢复下，新建集群可以跟原集群规格和节点数不一致。
- 单表/多表恢复新集群，集群版本仅可选9.1.0及以上集群版本。

- 步骤6** 单表/多表选择。下拉框选择**数据库名称**，数据库配置选择“**自定义**”时，可以设置以下配置参数，如果选择“**默认配置**”，以下参数将使用它们的默认值，配置完成后勾选**恢复表列表**（可多选）。

说明

恢复到新集群会新建数据库，如果新建的数据库与快照的数据库在配置上存在差异，可能会导致恢复失败。请在恢复之前确认原数据库的配置，如果跟默认配置不一致，可自定义修改。

表 11-21 数据库自定义参数

| 参数名称 | 参数描述                                                                                           | 取值范围                         | 默认值       |
|------|------------------------------------------------------------------------------------------------|------------------------------|-----------|
| 模板名  | 即从哪个模板创建新数据库。DWS采用从模板数据库复制的方式来创建新的数据库。初始时，DWS包含两个模板数据库template0、template1，以及一个默认的用户数据库gaussdb。 | 已有数据库的名称、template0、template1 | template0 |

| 参数名称 | 参数描述                                                                                                                                                                                                                                                                                                                            | 取值范围                                        | 默认值           |
|------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------|---------------|
| 字符编码 | <ul style="list-style-type: none"><li>指定数据库使用的字符编码，可以是字符串（如“SQL_ASCII”）、整数编号。</li><li>不指定时，默认使用模板数据库的编码。模板数据库template0和template1的编码默认与操作系统环境相关。<ul style="list-style-type: none"><li>template1不允许指定字符编码，因此若要创建数据库时指定字符编码，请使用template0创建数据库。</li><li>如果需要指定encoding，需要和template参数搭配使用，且template取值为template0。</li></ul></li></ul> | （常用取值）<br>GBK、UTF8、<br>Latin1、<br>SQL_ASCII | SQL_A<br>SCII |
| 字符集  | 指定新数据库使用的字符集。例如，通过 <code>lc_collate = 'zh_CN.gbk'</code> 设定该参数。该参数的使用会影响到对字符串的排序顺序（如使用ORDER BY执行，以及在文本列上使用索引的顺序）。默认是使用模板数据库的排序顺序。                                                                                                                                                                                               | 有效的排序类型                                     | C             |
| 字符分类 | 指定新数据库使用的字符分类。例如，通过 <code>lc_ctype = 'zh_CN.gbk'</code> 设定该参数。该参数的使用会影响到字符的分类，如大写、小写和数字。默认是使用模板数据库的字符分类。                                                                                                                                                                                                                        | 有效的字符分类                                     | C             |
| 类型   | 指定兼容的数据库的类型。                                                                                                                                                                                                                                                                                                                    | ORA、TD、<br>MySQL                            | ORA           |

**步骤7** 单击“下一步：确认配置”。

**步骤8** 确认无误后，单击“立即恢复”按钮，集群开始对选择的表进行恢复新集群。

----结束

## 11.4.6 快照参数配置

快照参数配置可以指定创建快照和快照恢复的参数信息，以便对快照功能进行优化调整。

### 说明

- 该功能仅8.2.0及以上集群版本支持（对于8.2.0之前的集群版本，仅支持个别参数的配置）。
- 配置参数后该集群所有的快照、恢复都将以前所配置的参数下发命令。
- 9.1.0.210及以上集群版本新增支持thread-count配置，Roach启动多线程备份冷热表和v3表时，使用的并发线程数，默认为cpu核数的2倍。

操作步骤

- 步骤1 登录DWS管理控制台。
- 步骤2 在左侧导航栏中，选择“专属集群 > 集群列表”。
- 步骤3 在集群列表中，单击指定集群的名称，进入“集群详情”页面。
- 步骤4 切换至“快照”页签，再单击二级页签中的“参数配置”。在“参数配置”页面，会展示当前集群所有的可配参数。
- 步骤5 根据用户不同的参数要求填入合适的参数值，详情请参见[表11-22](#)。
- 步骤6 单击保存即可。
- 结束

快照参数详情

表 11-22 快照参数说明

| 参数名称              | 参数类型 | 参数描述                                                                                                                           | 默认值       |
|-------------------|------|--------------------------------------------------------------------------------------------------------------------------------|-----------|
| parallel-process  | 备份参数 | 指定Roach备份时每个节点的并发进程。<br><b>说明</b><br>该参数支持8.2.0之前版本集群配置。                                                                       | 为本节点主DN个数 |
| compression-type  | 备份参数 | 压缩算法类型： <ul style="list-style-type: none"><li>zlib</li><li>LZ4</li></ul> <b>说明</b><br>该参数支持8.2.0之前版本集群配置。                      | LZ4       |
| compression-level | 备份参数 | 压缩级别，取值范围为0~9： <ul style="list-style-type: none"><li>0代表快速或无压缩。</li><li>9代表慢速或最大压缩。</li></ul> <b>说明</b><br>该参数支持8.2.0之前版本集群配置。 | 6         |
| buffer-size       | 备份参数 | 指定Roach上传介质的buffer大小，取值范围为256~16384，单位是MB。                                                                                     | 256       |
| buffer-block-size | 备份参数 | 指定Roach读取数据文件的数据块大小，取值范围为5242880~268435456，单位是Byte。                                                                            | 67108864  |

| 参数名称                | 参数类型 | 参数描述                                                                                                                                                                  | 默认值              |
|---------------------|------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------|
| cpu-cores           | 备份参数 | Roach启动多线程并发时，可以使用的CPU核数。                                                                                                                                             | 为节点CPU逻辑核数总和的1/2 |
| master-timeout      | 备份参数 | 指定Roach master节点和agent节点的通信超时时间，取值范围为600~3600，单位是s。                                                                                                                   | 3600             |
| max-backup-io-speed | 备份参数 | 指定Roach备份时的IO流控，取值范围为0~2048，单位是MB/s。必须大于buffer-block-size，0表示不限制。                                                                                                     | 0                |
| backup-mode         | 备份参数 | 全量备份的模式： <ul style="list-style-type: none"><li>0：一阶段的备份</li><li>1：两阶段的备份</li></ul>                                                                                    | 0                |
| cbm-parse-mode      | 备份参数 | 增量备份的模式： <ul style="list-style-type: none"><li>0：一次扫描cbm（内存占用大，性能快）</li><li>1：多次扫描cbm（内存占用稳定，性能差）</li></ul>                                                           | 0                |
| thread-count        | 备份参数 | Roach启动多线程备份冷热表和v3表时，使用的并发线程数，最大取值为核数的16倍。<br><b>说明</b><br>该参数仅9.1.0.210及以上集群版本支持配置。                                                                                  | 默认是cpu核数的2倍      |
| dump-options        | 备份参数 | 指定细粒度备份时支持的备份选项，支持备份权限和注释。 <ul style="list-style-type: none"><li>权限：enable_handle_acl。</li><li>注释：enable_handle_comment。</li></ul> <b>说明</b><br>该参数仅9.1.0及以上集群版本支持配置。 | 不备份权限和注释         |
| parallel-process    | 恢复参数 | 指定Roach备份时每个节点的并发进程，默认当前节点主DN数 + 1                                                                                                                                    | 1                |
| cpu-cores           | 恢复参数 | Roach启动多线程并发时，可以使用的CPU核数。                                                                                                                                             | 默认是1/2 cpu核数     |

| 参数名称          | 参数类型 | 参数描述                                                                                                                                                                                                                                                                                      | 默认值         |
|---------------|------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| logging-level | 恢复参数 | 日志级别： <ul style="list-style-type: none"><li>● FATAL（致命）：导致系统停止工作的、无法恢复的故障。该级别为最严重级别。</li><li>● ERROR（错误）：重大错误。</li><li>● WARNING（警告）：异常情况。系统在该情况下可能会继续处理任务。</li><li>● INFO（提示）：日志记录过程中的提示性日志。</li><li>● DEBUG（调试）：用于调试的详细信息。</li><li>● DEBUG2（调试2）：最详细的日志信息，通常会过滤不显示。该级别严重程度最轻。</li></ul> | INFO        |
| thread-count  | 恢复参数 | Roach启动多线程恢复冷热表和v3表时，使用的并发线程数，最大取值为核数的16倍。<br><b>说明</b><br>该参数仅9.1.0.210及以上集群版本支持配置。                                                                                                                                                                                                      | 默认是cpu核数的2倍 |

### 11.4.7 停止快照

用户创建快照开始后，可以根据需求在“快照管理”页面手动停止本次快照。

 说明

- 停止快照功能仅8.1.3.200及以上集群版本支持。
- 若快照即将完成，停止快照命令将不再生效，快照会正常结束。

#### 前提条件

停止快照仅支持快照状态显示为“创建中”的快照。在快照创建启动时与快照即将创建完成时无法进行停止快照操作。

#### 操作步骤

**步骤1** 登录DWS管理控制台。

- 步骤2 在左侧导航栏中，选择“管理 > 快照管理”（或在集群列表中，单击指定集群的名称，进入“集群详情”页面。左导航栏切换至“快照”页签）。默认显示用户所有的快照。
- 步骤3 在快照列表中，选择指定“创建中”快照所在行的“操作”列，单击“停止快照”按钮。
- 步骤4 在弹出的提示框中，单击“是”将停止本次快照，快照停止后状态将变为“不可用”状态。

| 快照管理        |      |      |      |      |      |         |                          |
|-------------|------|------|------|------|------|---------|--------------------------|
|             |      |      |      |      |      |         |                          |
| 快照名称        | 快照状态 | 集群名称 | 备份方式 | 快照类型 | 快照介值 | 快照组别    | 快照创建时间                   |
| test_stop_4 | 不可用  |      | 手动   | 全量   | OBS  | cluster | 2022/07/27 11:15:37 G... |
|             | 可用   |      | 手动   | 全量   | OBS  | cluster | 2022/07/27 10:23:48 G... |
|             | 可用   |      | 手动   | 全量   | OBS  | cluster | 2022/07/27 10:17:11 G... |
|             | 可用   |      | 手动   | 全量   | OBS  | cluster | 2022/07/27 10:06:03 G... |

----结束

## 11.5 DWS 集群节点扩缩容

### 11.5.1 查看巡检

#### 背景信息

DWS支持在变更操作前（扩/缩容、经典规格变更、升级）进行集群巡检，即在对应变更页面单击“立即巡检”按钮后会巡检当前集群的健康状态和集群指标是否满足变更要求，检查通过后可开始变更。如果巡检不通过，可打开巡检详情，查看不通过的巡检项，根据详情说明处理不通过的巡检项。巡检项检查标准详情请参见表11-23。

#### 说明

- 该特性仅8.1.1及以上集群版本支持。
- 如果用户无法处理不通过的巡检项，请联系技术支持人员。

#### 注意事项

- 集群已安装巡检插件并且插件版本为8.3.1.100及以上。
- 当前巡检结果24小时内有效，在有效期可执行变更操作，超过后需要重新进行巡检。
- 变更前集群24小时内没有巡检成功过，在变更操作（扩/缩容、经典规格变更、升级）开始前需要先完成一次巡检，并保证巡检检查通过。

#### 查看巡检详情

- 步骤1 登录DWS 管理控制台。
- 步骤2 在集群列表中单击指定集群名称。
- 步骤3 进入“集群详情”页面，切换至“巡检管理”页签。
- 步骤4 单击指定任务名称旁边的下拉按钮，查看巡检结果（巡检状态、执行进度、巡检结果、通过率等），并单击所在行“查看详情”按钮查看具体巡检项详情。

 说明

在变更界面创建巡检任务后也可在当前页面查看巡检进度，巡检详情以及终止巡检。

----结束

终止巡检任务

- 步骤1 登录DWS 管理控制台。
- 步骤2 在集群列表中单击指定集群名称。
- 步骤3 进入“集群详情”页面，切换至“**巡检管理**”页签。
- 步骤4 单击巡检中任务所在行操作列的“**终止任务**”按钮终止该巡检任务。

----结束

巡检项检查标准说明

表 11-23 巡检项详情

| 变更操作             | 巡检项名称             | 检查标准                                                                                                                           |
|------------------|-------------------|--------------------------------------------------------------------------------------------------------------------------------|
| 扩/扩容、经典变更<br>前巡检 | CheckTimeZone     | 检查集群内各节点时区，如果时区一致则检查通过，否则检查项不通过。                                                                                               |
|                  | CheckSpaceUsage   | 如果使用率超过warning阈值（默认为70%）报warning，超过NG阈值（默认为90%）则检查项不通过。集群路径下检查“GAUSSHOME/PGHOST/GPHOME/GAUSSLOG/tmp/data”路径的剩余空间，不满足阈值则检查项不通过。 |
|                  | CheckClusterState | CM进程不存在，检查不通过；检查fenced UDF状态，如果为down则报warning；检查集群状态为Normal，则检查项通过，否则检查项不通过。                                                   |
|                  | CheckEnvProfile   | 检查节点环境变量（\$GAUSSHOME、\$LD_LIBRARY_PATH、\$PATH），如果环境变量存在并配置正确则检查项通过，否则检查项不通过。                                                   |
|                  | CheckReadonlyMode | 检查集群中所有含CN节点上default_transaction_read_only值，如果为off则检查通过，否则不通过。                                                                 |
|                  | CheckCatchup      | 检查gaussdb进程堆栈是否能搜索到CatchupMain函数，如果搜索不到则检查项通过，否则检查项不通过。                                                                        |
|                  | CheckCollector    | 查看信息收集是否成功，如果收集成功则检查项通过，否则检查项不通过。                                                                                              |

| 变更操作 | 巡检项名称                    | 检查标准                                                                       |
|------|--------------------------|----------------------------------------------------------------------------|
|      | CheckTrust               | 检查各个节点的是否互信，如果有节点未互信则检查不通过，否则检查通过。                                         |
|      | CheckBalanceState        | 如果集群Balanced属性为Yes，则检查项通过，否则Warning，如果查询失败则检查项不通过。                         |
|      | CheckCnNumberSame        | 如果“/opt/dws/xml/cluster.xml和cm_ctl query -Cv”查询出来的CN数量不一致，检查通过，否则检查不通过。    |
|      | CheckCMPParam            | 如果enable_transaction_read_only为on、coordinator_heartbeat_timeout各节点一致，检查通过。 |
|      | CheckUtilslib            | 如果\$GAUSSHOME/utilslib目录存在，则检查不通过，否则检查通过。                                  |
|      | CheckPgxcgroup           | 检查pgxc_group表中in_redistribution为Y的个数，如果数量为0则检查项通过，否则检查项不通过。                |
|      | CheckLockState           | 检查集群是否被锁，如果集群未锁则检查通过，否则检查项不通过。                                             |
|      | CheckDBConnection        | 检查能否连接数据库，如果连接成功则检查项通过，否则检查项不通过。                                           |
|      | CheckGUConsistent        | 检查各CN/DN实例的guc参数是否一致，如果全部一致则检查通过，否则检查不通过。                                  |
|      | CheckTDDate              | 若存在TD模式数据库下的orc表，且包含date类型的列则检查不通过。                                        |
|      | CheckPgxcRedistb         | 检查数据库中是否存在重分布残留的临时表，如果不存在则检查通过，否则检查不通过。                                    |
|      | CheckMetaData            | 检查系统表元数据是否一致，如果一致，则检查通过，否则检查不通过。                                           |
|      | CheckGUCSetting          | postgresql.conf中GUC参数和pg_settings中参数是否一致，一致则检查通过，否则不通过。                    |
|      | CheckProacl              | 检查pg_proc系统表中proacl中是否存在纯数字用户名，如果存在，检查不通过，否则检查通过。                          |
|      | CheckMetaDataConsistency | 检查系统表在CN和DN之间数据是否一致，一致则检查通过，否则检查不通过。                                       |
|      | CheckReturnType          | 用户自定义函数包含非法返回类型，检查不通过，否则检查通过。                                              |

| 变更操作 | 巡检项名称                     | 检查标准                                                                                                                                                                                            |
|------|---------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|      | CheckUltraWide Table      | 如果存在超过996列的表，检查不通过，否则检查通过。                                                                                                                                                                      |
|      | CheckDataRedis Schema     | 如果数据库中存在data_redis schema且属主用户名不是redisuser，则检查不通过，否则检查通过。                                                                                                                                       |
|      | CheckDiskSpace Limited    | 用户的磁盘空间是否受限，如果受限，检查不通过，否则检查通过。                                                                                                                                                                  |
|      | CheckTableCollate         | 如果数据库中存在PCK表或者列存分区表使用collate字段，检查不通过，否则检查通过。                                                                                                                                                    |
|      | CheckDefaultOrientation   | 检测GUC参数，数据库设置和用户设置default_orientation均为行存则通过，否则检查不通过。                                                                                                                                           |
|      | CheckReplicationUuid      | 不存在使用default uuid的复制表则通过，否则检查不通过。                                                                                                                                                               |
|      | CheckUserStatementTimeout | 检测用户级的statement_timeout设置，未设置或设置为0则通过，否则检查不通过。                                                                                                                                                  |
|      | CheckJsonb                | 执行SQL语句“select attrelid::regclass from pg_attribute a join pg_type t on a.atttypid = t.oid and t.typname = 'jsonb' group by 1”，如果使用jsonb类型检查不通过，否则检查通过。                                         |
|      | CheckLengthOfIndex        | 执行SQL语句SELECT “ length(pg_get_indexdef(indexrelid)) FROM pg_index order by 1 desc limit 1”，如果结果大于192*1024检查不通过，否则检查通过。                                                                          |
|      | CheckLengthOfTable        | 执行SQL语句“select c.oid from pg_class c,pg_namespace n where c.relnamespace=n.oid and relkind='r' and n.nspname not in ('cstore') and length(n.nspname  '.'  c.relname)>=64;”，如果结果不为空检查不通过，否则检查通过。 |
|      | CheckUseWorkloadManager   | 执行sql语句“show use_workload_manager”，如果结果为on，检查通过，否则检查不通过。8.1.3.320及以后的版本不再校验此项参数，即此后的版本检查通过。                                                                                                     |
|      | CheckNecessary Schema     | 检查必要“schema(public)”是否都存在，如果缺失则检查不通过，否则检查通过。                                                                                                                                                    |

| 变更操作 | 巡检项名称                     | 检查标准                                                                                                                                              |
|------|---------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|
|      | CheckCMPParam Consistency | 获取主备CM节点上的cm.conf文件，对比参数设置，如果参数配置一致则检查通过，否则检查不通过。                                                                                                 |
|      | CheckSQLCompatibility     | mysql兼容性模式下，重分布临时表带索引会导致重分布慢。在对应的业务库中“SHOW sql_compatibility;”，如果值是mysql，且behavior_compat_options没有设置枚举值disable_including_all_mysql则检查不通过，否则检查通过。 |
|      | CheckBinaryUpgrade        | 检查“/DWS/manager/upgrade_backup/”下是否有对应的备份文件，如果存在备份的文件，则检查不通过，否则检查通过。                                                                              |
|      | CheckColdTableSpace       | 在所有库中检查是否存在冷热表，如果存在则检查不通过，否则检查通过。                                                                                                                 |
|      | CheckXFS                  | 查看“/etc/os-release”文件获取版本信息，如果是EulerOS且版本在4.19.87及以下则涉及XFS bug，检查不通过，否则检查通过。                                                                      |
|      | CheckGTMConfigConsistency | 获取主备GTM配置文件参数，如果参数设置一致则检查通过，否则检查不通过。                                                                                                              |
|      | CheckColversion           | 检查如果存在没有标记为1.0列存表，且当前默认列存为2.0，则检查不通过，否则检查通过。                                                                                                      |
|      | CheckTopSqlSize           | 检查topsql表大小，如果超过50G，则检查不通过，否则检查通过。                                                                                                                |
|      | CheckDeltaTable           | 检查是否存在delta表，如果存在delta表则该项检查不通过，否则检查通过。                                                                                                           |
|      | CheckMaxDataNode          | 检查comm_max_datanode的值，如果不等于实际主datanode个数，检查不通过，否则检查通过。                                                                                            |
|      | CheckSSHIP                | 获取主备CM节点上的cm.conf文件，对比参数设置，如果参数配置一致则检查通过，否则检查不通过。                                                                                                 |
|      | CheckTimeZoneLink         | 在沙箱内执行“ll /etc/localtime”，如果链接指向的文件路径包含“/var/chroot”则检查不通过，否则检查通过。                                                                                |
|      | CheckSpecialFile          | 检查程序目录“(GAUSSHOME)”下文件是否存在特殊字符以及非Ruby用户的文件，如果不存在则检查通过，否则检查不通过。                                                                                    |
|      | CheckSysSchemaTable       | 如果系统schema下有用户自建的表检查不通过，否则检查通过。                                                                                                                   |

| 变更操作  | 巡检项名称                    | 检查标准                                                                          |
|-------|--------------------------|-------------------------------------------------------------------------------|
| 升级前巡检 | CheckClusterParams       | 集群配置类参数（ip类、Port类、路径类参数）在postgresql.conf或者pgxc_node的配置，必须和静态配置文件一致。不一致则检查不通过。 |
|       | CheckCNNum               | 检查集群CN实例个数，如果CN个数大于2，小于10则检查项通过，否则检查项不通过。                                     |
|       | CheckDDL                 | 启动事务创建/删除schema和表，如果事务能够提交，检查通过，否则检查不通过。                                      |
|       | CheckTimeZone            | 检查集群内各节点时区，如果时区一致则检查通过，否则检查项不通过。                                              |
|       | CheckXidEpoch            | 检查xid消耗量，如果大于等于2的32次方，检查不通过。                                                  |
|       | CheckCnNumberSame        | 如果“/opt/dws/xml/cluster.xml和cm_ctl query -Cv”查询出来的cn数量不一致，检查通过，否则检查不通过。       |
|       | CheckGaussVer            | 检查各节点\$GAUSSHOM/bin目录下二进制文件版本是否一致。如果不一致，检查不通过。                                |
|       | CheckPsort               | 如果有psort索引，检查不通过。                                                             |
|       | CheckCatchup             | 检查gaussdb进程堆栈是否能搜索到CatchupMain函数，若搜索不到则检查项通过，否则检查项不通过。                        |
|       | CheckClusterState        | CM进程不存在，检查不通过；检查fenced UDF状态，如果为down则报warning；检查集群状态为Normal，则检查项通过，否则检查项不通过。  |
|       | CheckMetaDataConsistency | 检查系统表在CN和DN之间数据是否一致，一致则检查通过，否则检查不通过。                                          |
|       | CheckDependSystemObj     | 如果用户自建的对象依赖系统对象，检查不通过，否则检查通过。                                                 |
|       | CheckPgKeywords          | 如果表、列、函数、数据类型名为新增的保留关键字，检查不通过，否则检查通过。                                         |
|       | CheckReadOnlyMode        | 检查集群中所有含CN节点上default_transaction_read_only值，若为off则检查通过，否则不通过。                 |
|       | CheckMetaData            | 检查系统表元数据是否一致，如果一致，则检查通过，否则检查不通过。                                              |
|       | CheckGUCSetting          | postgresql.conf中GUC参数和pg_settings中参数是否一致，一致则检查通过，否则不通过。                       |

| 变更操作 | 巡检项名称                | 检查标准                                                                                                                           |
|------|----------------------|--------------------------------------------------------------------------------------------------------------------------------|
|      | CheckPgxcgroup       | 检查pgxc_group表中in_redistribution为Y的个数，如果数量为0则检查项通过，否则检查项不通过。                                                                    |
|      | CheckCmserverStandby | 如果集群cm_server属性为standby。则检查项通过，否则Warning。                                                                                      |
|      | CheckSpaceUsage      | 如果使用率超过warning阈值（默认为70%）报warning，超过NG阈值（默认为90%）则检查项不通过。集群路径下检查“GAUSSHOME/PGHOST/GPHOME/GAUSSLOG/tmp/data”路径的剩余空间，不满足阈值则检查项不通过。 |
|      | CheckEnvProfile      | 检查节点环境变量（\$GAUSSHOME、\$LD_LIBRARY_PATH、\$PATH），如果环境变量存在并配置正确则检查项通过，否则检查项不通过。                                                   |
|      | CheckBalanceState    | 如果集群Balanced属性为Yes，则检查项通过，否则Warning，如果查询失败则检查项不通过。                                                                             |
|      | CheckTDDate          | 若存在TD模式数据库下的orc表，且包含date类型的列则检查不通过。                                                                                            |
|      | CheckCatalog         | 检查pg_catalog中是否有用户自定义的数据库对象，如果有检查不通过，否则检查通过。                                                                                   |
|      | CheckPgauthid        | 检查pg_authid中，oid的最高位如果等于1，检查不通过，否则检查通过。                                                                                        |
|      | CheckSysdate         | 检查表、视图、存储过程中是否使用sysdate视图，如果有使用，检查不通过，否则检查通过。                                                                                  |
|      | CheckFilesNumber     | 当GAUSSHOME/PGHOST/GPHOME/tmp文件数量大于10000时，检查不通过，否则检查通过。                                                                         |
|      | CheckKeyFilesExist   | 检查upgrade_version文件、conf文件、control文件，data下关键目录是否存在，若有文件存在，检查通过，否则检查不通过。                                                        |
|      | CheckReturnType      | 用户自定义函数包含非法返回类型，检查不通过，否则检查通过。                                                                                                  |
|      | CheckTrust           | 检查各个节点的是否互信，若有节点未互信则检查不通过，否则检查通过。                                                                                              |
|      | CheckEnumGUCValue    | 检查pg_postgres.conf部分参数是否有引号，如果缺失单引号，则检查不通过，否则检查通过。                                                                             |

| 变更操作 | 巡检项名称                    | 检查标准                                                                                                                                                                                                                             |
|------|--------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|      | CheckSpecialFile         | 检查程序目录“(GAUSSHOME)”下文件是否存在特殊字符以及非Ruby用户的文件，如果不存在则检查通过，否则检查不通过。                                                                                                                                                                   |
|      | CheckNecessary Schema    | 检查必要“schema(public)”是否都存在。                                                                                                                                                                                                       |
|      | CheckUserDefinedDataType | 连接所有数据库，执行SQL语句“select count(*) from pg_type t,pg_namespace n,PG_ATTRIBUTE a where t.typtype=n.oid and t.oid=a.atttypid and t.typname='time_stamp' and n.nspname='information_schema' and a.atttypid> 16384;”，结果为空则检查通过，否则检查不通过。 |
|      | CheckCMParm Consistency  | 获取主备CM节点上的cm.conf文件，对比参数设置，如果参数配置一致则检查通过，否则检查不通过。                                                                                                                                                                                |
|      | CheckLightProxy          | 检查参数enable_light_proxy，如果为off，且behavior_compat_options不包含枚举值enable_force_add_batch则检查不通过，否则检查通过。                                                                                                                                 |
|      | CheckSSHIP               | 获取主备CM节点上的cm.conf文件，对比参数设置，如果参数配置一致则检查通过，否则检查不通过。                                                                                                                                                                                |
|      | CheckSysSchemaTable      | 如果系统schema下有用户自建的表则检查不通过，否则检查通过。                                                                                                                                                                                                 |
|      | CheckTimeZone Link       | 在沙箱内执行“ll /etc/localtime”，如果链接指向的文件路径包含“/var/chroot”则检查不通过，否则检查通过。                                                                                                                                                               |

## 11.5.2 节点管理

### 节点管理概述

节点管理包含查看当前集群的节点列表、添加或删除用于扩容的新节点等功能，在节点管理页面可以清晰的查看各节点的状态、节点规格、所属可用分区以及资源状态等

信息。此外，单击指定节点“节点别名”列文字后的  图标，可对节点别名进行修改（如果节点没有别名，则可为节点增加别名）。

#### 说明

- 该特性仅8.1.1.200及以上集群版本支持。

## 添加节点

添加节点主要用于大规模扩容场景，通过节点管理功能可以提前分批次添加准备好用于扩容的节点，添加节点过程中不影响集群业务。例如需要新扩容180个节点，可分3批各添加60个，如果其中有一部分添加失败，可再次添加失败数量的节点，等180个节点添加成功后，再使用这些添加好的节点进行扩容。

### 注意事项

- 添加节点功能只能在管理侧任务列表中没有其他任务运行时使用。
- 添加节点的存储大小必须和该集群已有的节点保持一致。
- 添加成功的节点称为“空闲节点”，此节点主要用于扩容场景下，建议需要使用时添加，并及时扩容到集群中。
- 在反亲和部署模式下，每次添加节点的数量只能是集群安全环大小的整数倍，如集群安全环大小为3，则添加节点时会强制数量为3的倍数。
- 在反亲和部署模式下，如果因异常下电或其它原因导致某个空闲节点故障，那么该节点所在服务器组内其他节点也会变成不可用状态，此时建议删除故障节点后重新添加。
- 在反亲和部署模式下，如果某个节点添加失败回滚，那么该节点所在服务器组正在创建的其他节点也将被回滚。

### 操作步骤

**步骤1** 登录DWS管理控制台。

**步骤2** 选择“专属集群 > 集群列表”。默认显示用户所有的集群列表。

**步骤3** 在集群列表中，单击指定集群名称进入“集群详情”页面，单击左侧的“节点管理”进入节点管理页签。

**步骤4** 单击“添加节点”按钮，填写需要增加的空闲节点数量，单击“下一步：确认”。如果原子网IP不够，支持跨子网添加空闲节点。

**步骤5** 确认无误后，单击“提交”按钮，跳转到节点管理页面，开始添加节点。添加失败的节点会自动回滚并记录在失败列表中。

----结束

## 删除节点

### 注意事项

- 删除节点功能只能在管理侧任务列表中没有其他任务运行时使用。
- 删除节点功能只支持删除资源状态为“空闲”的节点，此节点还未扩容至集群，可以删除，已使用的节点不支持删除。
- 在反亲和部署模式下，删除节点只能以集群的安全环为单位，如集群安全环大小为3，则选择某个节点删除时会自动选择并提示删除同一个安全环的其他节点。

### 操作步骤

**步骤1** 登录DWS管理控制台。

**步骤2** 选择“专属集群 > 集群列表”。默认显示用户所有的集群列表。

**步骤3** 在集群列表中，单击指定集群名称进入“集群详情”页面，单击“节点管理”进入节点管理页签。

**步骤4** 在“节点管理”页面选中要删除的节点，单击“删除节点”按钮。

**步骤5** 确认无误后单击“确定”，删除成功后，该节点将不再显示到节点管理页面。

----结束

## 11.5.3 节点变更

### 11.5.3.1 集群扩容

用户需要更多的计算资源或存储资源以满足业务需要时，可以在管理控制台对已有集群，通过新增节点进行扩容操作。

#### 说明

- 存算一体集群扩容时使用与当前集群相同的存储规格。
- 如果子网IP数量不足，允许执行跨子网扩容操作。

如果您是因集群存储容量不足而扩容集群，建议您在扩容前先执行VACUUM清理和回收存储空间，DWS数据仓库中保存的数据在删除后，可能没有释放占用的磁盘空间形成脏数据，导致磁盘浪费。如果执行VACUUM后，已使用存储容量仍然占用过高，您再进行扩容。VACUUM的语法请参见《数据仓库服务SQL语法参考》中“SQL语法参考 > DDL语法 > VACUUM”章节。

#### 扩容对系统的影响

- 扩容前，需关闭创建了临时表的客户端连接，因为在扩容过程中及扩容成功之前创建的临时表将会失效，操作临时表也会失败。但是扩容后创建的临时表不受影响。
- 正在扩容的集群将禁用重启集群、停止集群、启动集群、规格变更、增删CN、扩容集群、缩容集群、创建快照、重置数据库管理员密码等功能。
- 离线扩容过程中，集群会自动重启，因此集群会有一段时间变为“不可用”状态，重启成功后集群恢复到“可用”状态。然后在扩容结束阶段，如果用户选择了自动重分布，系统会将集群中用户数据在全部节点重新动态分布，否则需要用户自主开始数据重分布。
- 离线扩容过程中，应该停止所有业务或运行少量查询语句。表重分布期间会对表加共享锁，所有插入、更新、删除操作和表DDL操作都会长时间阻塞，会出现等锁超时情况。一旦表重分布完成，用户可以正常访问该表。在重分布执行过程中，应当避免执行超过20分钟的查询（在重分布执行时申请写锁的默认时间为20分钟）。否则可能导致重分布出现等待加锁超时失败的问题。
- 扩容后，如果集群创建新快照，将包含扩容节点上的数据。
- 如果集群扩容失败，数据库会在后台自动执行扩容回滚操作，集群会恢复到扩容前的节点个数。
  - 如果回滚成功，集群仍可以正常使用，用户可以重新执行“扩容”操作，如果仍扩容失败，请及时联系技术支持人员进行处理。
  - 如果因为某些异常原因后台回滚失败，则集群可能会变为“不可用”状态，此时无法再执行“扩容”或重启集群的操作，请及时联系技术支持人员进行处理。

## 前提条件

- 请确定需要扩容的集群处于“可用”、“只读”或者“非均衡”状态。
- 请确定计划扩容的节点数小于等于用户节点数的剩余配额，否则系统无法进行扩容操作。

## 扩容集群

### 说明

- 离线扩容期间集群将变为只读状态，请谨慎操作。
- 扩容期间集群会有闪断，请谨慎操作。
- 为保证用户的数据安全建议在开始扩容操作之前创建手动快照。如何创建快照请参见[手动快照](#)。

**步骤1** 登录DWS管理控制台。

**步骤2** 单击“专属集群 > 集群列表”。

默认显示用户所有的集群列表。

**步骤3** 在集群列表中，在指定集群所在行的“操作”列，选择“更多 > 节点变更 > 扩容”。系统将显示扩容页面。

在扩容开始前如果集群满足巡检条件，需单击“立即巡检”按钮先完成一次巡检，并保证巡检检查通过，通过后可进行下一步变更操作，详情请参见[查看巡检](#)。

- 如果原子网IP不够，可以跨子网扩容。

**步骤4** 在“增加节点数”选择一个扩容后的节点数。

- 扩容操作增加的是DN节点，如果需要增加CN节点可参见[增删DWS集群CN节点](#)。
- 扩容后的节点数量，在原节点数量的基础上，须至少增加3个节点，最多可增加的节点个数为节点剩余配额的最大值。并且，此处设置的扩容后的节点数量不能超过256个节点。
- 扩容增加的节点规格，默认与集群当前各节点的规格相同。
- 扩容后的集群与原集群的虚拟私有云和安全组也相同。

**步骤5** 设置高级配置。

- 选择“默认配置”：“自动重分布”默认开启，“重分布模式”默认为离线模式。
- 选择“自定义”，您可以设置以下高级配置参数进行扩容操作：
  - “自动重分布”：支持打开自动重分布。自动重分布开启，扩容阶段结束后将立即执行数据重分布；如果选择关闭此功能则只进行扩容添加节点，需在“更多>节点变更>重分布”中选择执行数据重分布。
  - “重分布并发配置”：自动重分布开启，支持设置并发数量。可配置并发数在1~200之间，默认值为4。
  - “重分布模式”：“离线模式”。

**步骤6** 确认无误后勾选“我已确认”，单击“下一步：确认”。

**步骤7** 单击“提交”。

- 提交扩容申请后，集群的“任务信息”显示为“节点扩容”，扩容需要时间请耐心等待。

- 扩容过程中，集群会自动重启，因此会有一段时间“集群状态”显示为“不可用”，重启成功后“集群状态”会变成“可用”。
- 扩容结束后，集群将重新分布数据，重分布过程中会存在一段时间，“集群状态”为“只读”。
- 只有“集群状态”显示为“可用”且“任务信息”不再有“节点扩容”的状态，才表示扩容成功，用户可以开始使用集群。
- 如果集群的“任务信息”显示为“扩容失败”，表示集群扩容失败。

----结束

## 使用空闲节点扩容

对于大规模的集群扩容，基于可靠性考虑，可通过提前准备好需要扩容的ECS节点执行[添加节点](#)，在扩容时选择使用空闲节点扩容。

### 说明

- 在大规模集群扩容时建议关闭自动重分布，有利于扩容阶段失败重试，从而增加可靠性。
- 扩容完成后，再手动执行[重分布](#)操作，保证重分布阶段也可进行多次失败重试。

### 注意事项

- 集群内必须提前添加好一定数量的可用节点才可以使用空闲节点扩容。
- 在反亲和部署模式下，使用空闲节点扩容的节点数量只能是安全环的整数倍。
- 提交扩容前需完成扩容准备操作，即准备即将进行扩容任务配置的工作，请耐心等待一段时间。

### 操作步骤

**步骤1** 登录DWS管理控制台。

**步骤2** 选择“专属集群 > 集群列表”。默认显示用户所有的集群列表。

**步骤3** 集群列表中，在指定集群所在行的“操作”列，选择“更多 > 节点变更 > 扩容”。

在扩容开始前如果集群满足巡检条件，需单击“**立即巡检**”按钮先完成一次巡检，并保证巡检检查通过，通过后可进行下一步变更操作，详情请参见[查看巡检](#)。

如果集群内存在空闲节点，系统将显示“是否从已添加节点扩容”的页面，否则直接显示常规扩容页面。

**步骤4** 单击“**扩容准备**”按钮，等待扩容准备完成。

**步骤5** 扩容页面参数用户可根据自身需求设置，详情请参见[扩容集群](#)。

配置好扩容和重分布参数后，勾选“我已确认”，单击“下一步：确认”。

**步骤6** 确认无误后，单击“提交”开始扩容。

----结束

## 查看扩容详情

**步骤1** 登录DWS管理控制台。

**步骤2** 选择“专属集群 > 集群列表”，默认显示用户所有的集群列表。

**步骤3** 集群列表中，在指定集群所在行的“任务信息”列，单击“查看详情”。

**步骤4** 进入“扩容详情”页面，用户可查看集群此时的扩容状态。

图 11-2 查看扩容详情

任务详情

集群名称: hwx1118786-逻辑0327-01      任务名称: 节点扩容      运行状态: 运行中

|      | 任务信息           | 预计时间 | 开始时间                      | 结束时间                      | 状态  | 详情 |
|------|----------------|------|---------------------------|---------------------------|-----|----|
| 添加资源 | 添加资源前准备 [3/3]  | --   | 2024/03/27 15:24:08 GM... | 2024/03/27 15:24:23 GM... | 已完成 | -- |
|      | 创建虚拟机 [3/3]    | --   | 2024/03/27 15:24:28 GM... | 2024/03/27 15:29:54 GM... | 已完成 | -- |
|      | 添加资源后置处理 [3/3] | --   | 2024/03/27 15:29:34 GM... | 2024/03/27 15:30:09 GM... | 已完成 | -- |
| 构建资源 | 等待添加资源完成       | --   | 2024/03/27 15:24:08 GM... | 2024/03/27 15:30:43 GM... | 已完成 | -- |
|      | 构建资源           | --   | 2024/03/27 15:30:44 GM... | 2024/03/27 15:41:13 GM... | 已完成 | -- |
|      | 构建后置处理         | 1分钟  | 2024/03/27 15:41:14 GM... | --                        | 运行中 | -- |

----结束

11.5.3.2 集群重分布

11.5.3.2.1 重分布

集群重分布是集群扩容后的一项重要且耗时的任务，集群扩容前业务数据都集中分布在老节点上，扩容之后会新增空的可用节点，此时将老节点的数据均匀分布到新扩容节点的过程称为数据“重分布”，数据“重分布”后将大大提升业务响应速率。

默认情况下，在扩容之后将自动调起重分布任务，为了增强扩容重分布整个流程的可靠性，可以选择在扩容时关闭自动重分布功能，在扩容成功之后再手动使用重分布功能执行重分布任务，在这种分段模式下，扩容和重分布都可以做到失败重试。

当前重分布支持**离线重分布**模式，默认情况下，提交重分布任务时将选择离线重分布模式。

在重分布开始前或重分布暂停过程中，支持对未重分布的表按照schema、表维度设置重分布优先级。

须知

- 重分布功能8.1.1.200及以上集群版本支持。
- 只有在扩容之后，集群任务信息为“待重分布”状态时才能手动使用“重分布”功能，其他时段该功能不可使用。
- 在扩容阶段也可以选择重分布模式的高级配置，详情参见**设置高级配置**。
- 重分布队列的排序依据表的relpage大小进行，为确保relpage大小正确，建议在重分布之前对需要重分布的表执行analyze操作。

离线重分布

注意事项

- 离线重分布模式下数据库不支持DDL和DCL操作，正在重分布的表只支持简单的DQL操作。
- 表重分布期间会对表加共享锁，所有插入、更新、删除操作和表DDL操作都会长时间阻塞，会出现等锁超时情况。在重分布执行过程中，用户应当避免执行超过20分钟的查询（在重分布执行时申请写锁的默认时间为20分钟）。否则可能导致重分布出现等待加锁超时失败的问题。

#### 操作步骤

**步骤1** 登录DWS管理控制台。

**步骤2** 选择“专属集群 > 集群列表”。默认显示用户所有的集群列表。

**步骤3** 集群列表中，在指定集群所在行的“操作”列，选择“更多 > 节点变更 > 重分布”。系统将显示“重分布”页面。

**步骤4** 在“重分布”页面选择默认的离线模式，单击“下一步：确认”提交重分布任务。

----结束

### 11.5.3.2.2 查看重分布详情

在“重分布详情”页面可以查看到当前集群的重分布模式、重分布进度、数据表重分布详情等监控信息，可以对重分布进行暂停和恢复，重分布暂停状态可设置重分布优先级，修改重分布并发数等操作。

#### 说明

查看重分布详情功能8.1.1.200及以上集群版本支持，其中数据表重分布进度详情仅8.2.1及以上集群版本支持。

#### 注意事项

查看重分布详情功能只有集群处于重分布中、重分布失败或者重分布暂停状态下才能使用，其他时段该功能不可使用，且相关信息展示会有一定延迟。

#### 操作步骤

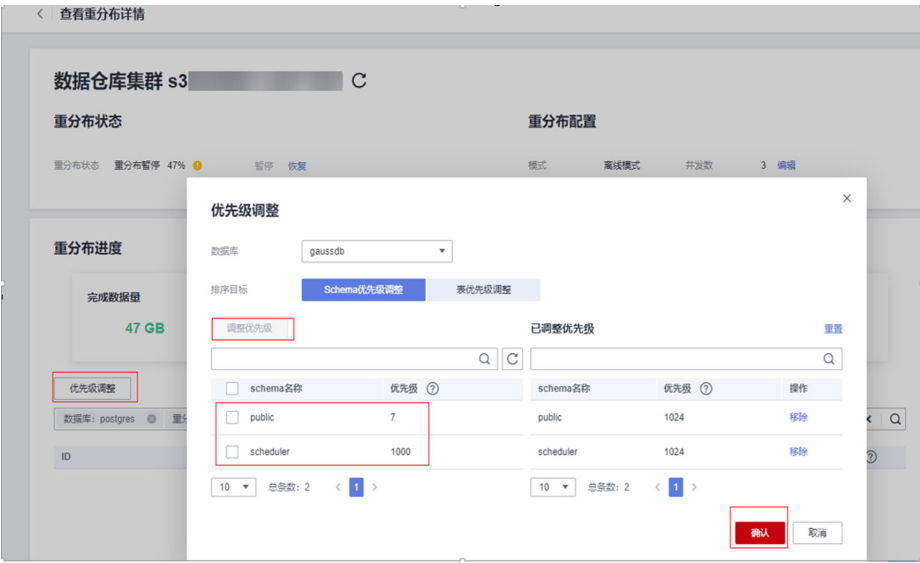
**步骤1** 登录DWS管理控制台。

**步骤2** 选择“专属集群 > 集群列表”。默认显示用户所有的集群列表。

**步骤3** 集群列表中，在指定集群所在行的“任务信息”列，单击“查看详情”。

**步骤4** 重分布中会显示重分布状态、重分布配置、重分布整体进度以及指定数据库下所有数据库表的重分布详情。用户在查看数据表重分布详情时，**必须指定一个数据库**，且支持按照表的重分布状态和表名作为条件的搜索，由于重分布机制的原因，若某数据库下所有表完成重分布，页面将不会再查询到详细信息。

**步骤5** 重分布暂停状态下，可设置重分布优先级（schema或表维度），设置了排序优先级后，会按照设置的重分布顺序依次进行重分布。用户也可在重分布开始前设置重分布优先级。



**步骤6** 重分布过程中可动态调整重分布并发数。

**说明**

8.1.0及以下集群版本不支持动态调整，需先暂停重分布。



**步骤7** 重分布完成后会显示重分布进度（完成数据量、剩余数据量、完成表数量、剩余表数量、重分布过程中的平均速率）。



----结束

## 11.6 DWS 集群规格变更

### 11.6.1 弹性变更规格

#### 概述

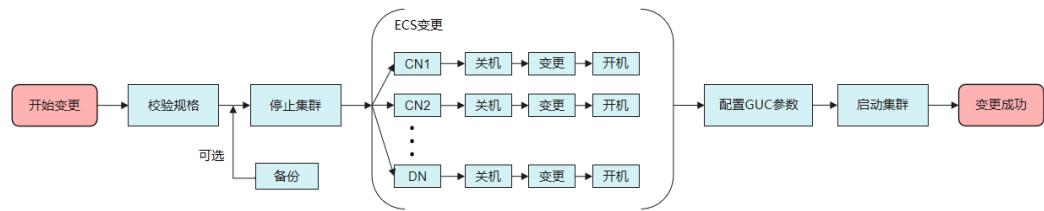
在使用集群过程中，当用户的业务量过大，则需要更多的资源（CPU、内存、磁盘等）来支撑庞大的业务量，如果用户当前使用的集群资源不足，则需要新建集群，或将当前集群删除后创建一个资源更多的集群，这样不仅消耗成本，整个流程也会十分耗时。而且如果用户创建的集群自身资源较多，但业务量并没有那么多的情况下，会导致资源冗余，成本过高的出现。

因此“弹性变更规格”功能应运而生，相比于扩容节点而言，该功能更适合阶段性峰值或只对计算能力（CPU、内存）变化有诉求的业务场景，在业务峰值来临之前您可以通过弹性变更规格快速提升集群计算能力，在业务峰值过后再快速的将集群配置降低，做到最大程度的节约成本。

基于ECS弹性云服务器底层基础能力，升降目标集群虚拟机节点的CPU、内存这两种资源配置，如下图所示：

- 弹性变更规格整个流程需要停止集群，所以需要用户合理安排变更时间窗口，防止集群停止影响业务。
- 在实际变更过程中，所有节点的变更并行执行，所以不会由于节点过多而造成变更时间变长，整个变更时间在5~10分钟左右，请耐心等待。

图 11-3 弹性变更原理图



### 说明

- 弹性变更规格仅8.1.1.300及以上集群版本支持，历史版本需要联系技术支持人员升级使用。

## 注意事项

- 变更规格降配即选择比集群当前规格低的目标规格，进行此操作可能会影响集群性能，请谨慎评估业务后进行操作。
- 变更规格前需检查当前Region目标规格的ECS资源、租户CPU配额是否充足，有满足条件的规格可供变更。
- 规格变更是支持重入的，若存在部分节点变更失败，可以再次提交变更任务，执行变更流程。

## 约束与限制

- ECS弹性云服务器同一类规格之间可以进行升/降配弹性变更，例如dwsx2.2xlarge.m7可变更到dwsx2.4xlarge.m7，但不支持变更到dwsx2.4xlarge.m6。
- 弹性变更规格需要停止虚拟机，仅支持离线变更，变更时间大约需要5~10分钟。

## 操作步骤

**步骤1** 登录DWS管理控制台。

**步骤2** 选择“专属集群 > 集群列表”。默认显示用户所有的集群列表。

**步骤3** 集群列表中，在需要变更的集群所在行的“操作”列，选择“更多 > 规格变更 > 弹性变更规格”，系统将显示“规格变更”页面。

**步骤4** 在“弹性变更规格”页面，选择目标规格，您可根据需求选择是否打开自动备份开关。

**步骤5** 确认无误后勾选“我已确认”，单击“下一步：确认”。

**步骤6** 单击“提交”按钮，提交集群变更规格任务。

**步骤7** 返回集群列表，集群将显示“弹性变更规格中”，请耐心等待5~10分钟左右。

----结束

## 11.6.2 EVS 集群磁盘扩容

### 磁盘扩容概述

随着客户业务的发展，磁盘空间往往最先出现资源瓶颈，在其他资源尚且充足的情况下，执行传统扩容操作不仅耗时久，还伴随着资源浪费问题。通过磁盘扩容可快速缓解存储资源瓶颈现象，操作过程中无需暂停业务，并且不会造成CPU、内存等资源浪费。用户可在没有其他业务情况下选择磁盘扩容操作，扩容成功后如果磁盘空间仍不足可以继续磁盘扩容，若扩容失败用户可尝试重新进行磁盘扩容操作。

### 说明

- 磁盘扩容功能仅8.1.1.203及以上版本支持，并且创建集群规格需要为存算一体SSD云盘类型。
- 集群状态为“可用”、“待重启”、“只读”、“非均衡”、“节点故障”或者“不可用”的情况下才能下发磁盘扩容。

### 注意事项

- 热数据存储只支持磁盘扩容，不支持缩容。
- 热数据存储扩容时，请选择在业务低峰期进行存储扩容。
- 若集群是只读或者磁盘满导致的不可用状态，单击“磁盘扩容”后，页面会弹出提示信息，磁盘扩容完成后，请耐心等待集群状态恢复为可用。

### 操作步骤

**步骤1** 登录DWS 管理控制台。

**步骤2** 单击“专属集群 > 集群列表”。默认显示用户所有的集群列表。

**步骤3** 集群列表中，在指定集群所在行的“操作”列，选择“更多 > 规格变更 > 弹性磁盘扩容”。系统将显示“磁盘扩容”页面。

**步骤4** 在“磁盘扩容”页面，选择磁盘扩容容量，集群变配过程中增大或减小磁盘大小的间隔大小为步长，在操作时需要按照对应规格的存储步长来选择。单击“下一步：规格确认”按钮。

|        |                                                          |
|--------|----------------------------------------------------------|
| 集群名称   | test-no-delete                                           |
| 热数据存储  | 100 GB / 节点                                              |
| * 本次扩容 | <div><div>—</div><div>10</div><div>+</div></div> GB / 节点 |
| 扩容后存储  | 110 GB / 节点                                              |
| 原节点数量  | 3                                                        |
| 规格详情   | 12 vCPUs   96 GB 内存   330 GB 热数据存储                       |

温馨提示

1. 热数据存储只支持扩容，不支持缩容。

2. 热数据存储扩容时，请选择在业务低峰期进行存储扩容；集群状态为只读状态时，请耐心等待集群状态恢复为可用。

☐ 我已确认

**步骤5** 确认无误后，单击“提交”按钮提交磁盘扩容任务。

**步骤6** 单击返回集群列表，显示磁盘扩容进度信息。

----结束

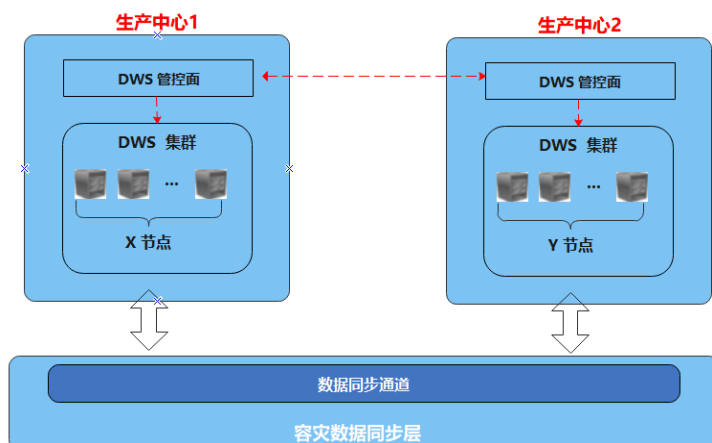
## 11.7 DWS 集群容灾管理

### 11.7.1 DWS 集群容灾场景介绍

#### 概览

容灾，即在另一个可用分区（Region内）部署一个同构的DWS灾备集群，如果生产集群所处的地理位置发生自然灾害，或者集群内部出现了故障从而导致生产集群无法正常对外提供读写服务，那么灾备集群可以切换为生产集群，从而保障业务连续性。架构图如下所示：

图 11-4 容灾架构图



### 说明

- Region内容灾特性仅8.1.1及以上集群版本支持。

## 容灾特点

- 多形态容灾
  - 支持Region内容灾。
  - 多种数据同步方式：基于直连互信作为同步层，以实现更多场景下的容灾。
- TCO低
  - 部署异构（逻辑同构）。
  - 容灾级别：集群级。
- 可视化
  - 自动化，一键式容灾演练、恢复。

## 约束与限制

- 加密集群不支持容灾。
- 非细粒度灾备集群在数据同步期间，无法提供读写服务。
- 灾备集群在容灾任务停止或者异常但灾备集群正常的情况下，可以提供读服务，灾备切换成功后可以提供读写服务。
- 容灾创建后，生产集群快照功能正常使用，但是灾备集群禁用快照功能，以及生产、灾备集群均禁用恢复功能。
- 同步资源池仅支持在8.3.0及以上集群版本的集群级容灾。
- 使用冷热表后，冷数据同步依赖OBS。
- 容灾不支持同步配置的外部数据源数据。
- 容灾管理为同一租户下的双集群容灾。
- 创建容灾需要生产集群与灾备集群类型、版本号一致且逻辑同构。
- Region内容灾生产集群和灾备集群需要在同一个VPC内。

- Region内容灾在生产集群与容灾集群切换后绑定的弹性负载均衡（ELB）会自动切换到新的生产集群，过程中会有短暂的连接中断，请注意在变更时间窗内尽量避免业务语句写入。
- Region内容灾在生产集群与容灾集群，原生产集群所在的EIP和内网域名，连接IP地址不会自动切换，业务系统中如果有使用EIP、域名或者IP方式连接，需要切换到新集群。

## 11.7.2 创建与启动 DWS 集群容灾

### 创建 Region 内集群级容灾

#### 前提条件

集群处于可用状态或者非均衡状态才可进行创建容灾操作。

#### 操作步骤

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，选择“管理 > 容灾管理”。

**步骤3** 在“容灾管理”页面，单击“创建容灾”。

**步骤4** 选择“容灾类型”。

- “容灾类型”：选择“Region内容灾”类型。
- “容灾名称”：容灾名称在4位到64位之间，不区分大小写，必须以字母开头，可以包含字母、数字、中划线或者下划线，不能包含其他的特殊字符。

**步骤5** 选择生产集群信息。

- “集群名称”：提供下拉列表选择已创建的生产集群。
- “可用分区”：生产集群的可用区，用户在选择生产集群后自动显示其对应的可用区信息。

**步骤6** 选择灾备集群信息。

- “可用分区”：选择灾备集群工作区域下的关联可用分区。

#### 说明

灾备集群可用分区支持与生产集群可用分区相同，3AZ集群在其中的任一AZ都可以作为灾备集群选中。

- “集群名称”：用户选择灾备集群可用分区后，会自动过滤出符合逻辑同构要求的灾备集群列表，如果没有符合条件的灾备集群，可单击“创建灾备集群”创建与生产集群配置相同的集群作为灾备集群。

**步骤7** 设置高级配置。选择“自定义”时，可以设置以下高级配置参数，若选择“默认配置”，以下参数将使用它们的默认值。

- “容灾同步周期”：容灾同步周期决定了生产集群间隔多长时间向灾备集群同步增量数据，请结合实际业务数据量设置合理值。

#### 说明

容灾同步周期默认值为30分钟。

**步骤8** 单击“确定”，开始为集群创建容灾。

此时容灾的“容灾状态”显示为“创建中”，创建需要时间请耐心等待。创建成功后“容灾状态”显示为“未启动”。

----结束

## 启动容灾

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，选择“管理 > 容灾管理”。

**步骤3** 在容灾列表中，在指定容灾所在行的“操作”列，单击“启动”按钮。

**步骤4** 在弹出框单击“确定”。

此时容灾的“容灾状态”显示为“启动中”，启动需要时间请耐心等待。启动成功后“容灾状态”显示为“运行中”。

### 说明

- 容灾状态为“未启动”、“启动失败”和“已停止”时可以执行启动容灾操作。
- 启动容灾后，生产集群和灾备集群将无法进行恢复、扩容、升级、重启、节点变更、节点管理、更新密码等操作，此外，灾备集群将无法进行备份操作，请谨慎操作。

----结束

## 查看容灾信息

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，选择“管理 > 容灾管理”。

**步骤3** 在容灾列表中找到所需要的容灾，然后单击容灾名称，进入容灾“基本信息”页面。

在容灾“基本信息”页面，可以查看如下相关信息：

- 容灾信息：用户可查看容灾ID、容灾名称、容灾类型、容灾创建时间、容灾启动时间以及容灾状态。
- 生产集群信息：用户可查看生产集群ID、集群名称、可用分区、已用存储容量、集群容灾状态、最近容灾成功时间等相关信息。
- 灾备集群信息：用户可查看灾备集群ID、集群名称、可用分区、已用存储容量、集群容灾状态、最近容灾成功时间等相关信息。
- 容灾配置：用户可查看并修改容灾同步周期。

----结束

## 更新容灾配置

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，选择“管理 > 容灾管理”。

**步骤3** 在容灾列表中找到所需要的容灾，然后单击容灾名称，进入容灾“基本信息”页面。

**步骤4** 在“容灾配置”模块，单击“**修改**”按钮可更新容灾的配置信息。

 **说明**

- 容灾状态为“未启动”或“已停止”时，可以执行容灾配置修改操作。
- 新的配置在容灾重新启动后生效。

----结束

## 容灾状态下如何进行集群扩容？

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，选择“专属集群 > 集群列表”。

**步骤3** 在集群列表中，若需要容灾扩容的集群“任务信息”为“容灾未启动”，执行**步骤5**和**步骤7**。

**步骤4** （可选）如果需要容灾扩容的集群“任务信息”不是“容灾未启动”时，则需要删除容灾，请参见**删除容灾**。

**步骤5** 在生产、灾备集群所在行的“操作”列，选择“更多>扩容”，确认节点规格后执行扩容操作。

**步骤6** 创建容灾，请参见**创建与启动DWS集群容灾**。

**步骤7** 启动容灾，请参见**启动容灾**。

 **说明**

扩容后的生产集群和灾备集群需逻辑同构，即：扩容后生产、灾备集群的DN数量保持一致。

----结束

## 11.7.3 切换 DWS 集群容灾

### 灾备切换

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，选择“管理 > 容灾管理”。

**步骤3** 在容灾列表中，在指定容灾所在行的“操作”列，选择“**更多 > 灾备切换**”按钮。

**步骤4** 在弹出框单击“确定”。

此时容灾的“容灾状态”显示为“灾备切换中”，灾备切换需要时间请耐心等待。

切换成功后“容灾状态”更新为切换前的状态，如：切换前容灾状态为“运行中”，切换后还是“运行中”。

#### 说明

- “灾备切换”按钮用于在容灾正常情况下主备倒换操作。
- 容灾状态为“运行中”时可以执行灾备切换操作。
- 灾备切换需要一定时间，在此期间，原生产集群将不可用。
- 不同场景下进行灾备切换，RPO（Recovery Point Object，灾难发生后系统和数据必须恢复到的时间点要求）说明如下：
  - 生产集群在“可用”的状态下，RPO=0。
  - 生产集群在“不可用”的状态下，无法保证RPO=0，但数据至少可恢复到生产集群“最近容灾成功时间”，详情请参见[查看容灾信息](#)。

----结束

## 异常切换

### 操作场景

生产集群不可用灾备集群正常，容灾状态为“异常”可以执行异常切换操作。

### 操作步骤

- 步骤1** 登录DWS管理控制台。
- 步骤2** 在左侧导航栏中，选择“管理 > 容灾管理”。
- 步骤3** 在容灾列表中，在指定容灾所在行的“操作”列，单击“**更多> 异常切换**”按钮。
- 步骤4** 在弹出框单击“确定”。

此时容灾的“容灾状态”显示为“异常切换中”，异常切换需要时间请耐心等待。

切换成功后“容灾状态”更新为切换前的状态，如：切换前容灾状态为“异常”，切换后还是“异常”。

#### 说明

- “异常切换”按钮用于容灾异常或者生产集群故障情况下主备切换操作。
- 容灾异常切换仅8.1.2及以上集群版本支持。
- 异常切换会将灾备集群升为主，若原生产集群故障后存在部分数据未同步到灾备集群，那灾备集群升主后将缺少这些数据，切换时请确认容灾最后同步时间，谨慎操作。

----结束

## 容灾恢复

### 操作场景

容灾恢复是主备集群进行异常切换后的恢复操作，容灾恢复前需确认原生产集群已恢复正常。

### 操作步骤

- 步骤1** 登录DWS管理控制台。
- 步骤2** 在左侧导航栏中，选择“管理 > 容灾管理”。
- 步骤3** 在容灾列表中，在指定容灾所在行的“操作”列，选择“**更多 > 容灾恢复**”按钮。

**步骤4** 在弹出框中，“数据同步模式”可选择“增量”或“全量”。

 **说明**

重建容灾过程中的数据同步模式，推荐选择“增量”同步。

**步骤5** 单击“确定”。

此时容灾的“容灾状态”显示为“容灾恢复中”，容灾恢复需要时间请耐心等待。

容灾恢复成功后“容灾状态”更新为运行中的状态。

 **说明**

- 容灾恢复仅8.1.2及以上集群版本支持。
- 容灾恢复会删除灾备集群数据与新生产集群重新建立容灾关系。

----结束

## 11.7.4 停止与删除 DWS 集群容灾

### 停止容灾

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，选择“管理 > 容灾管理”。

**步骤3** 在容灾列表中，在指定容灾所在行的“操作”列，单击“**停止**”按钮。

**步骤4** 在弹出框单击“确定”。

此时容灾的“容灾状态”显示为“停止中”，停止需要时间请耐心等待。停止成功后“容灾状态”显示为“已停止”

 **说明**

- 容灾状态为“运行中”和“停止失败”时可以执行停止容灾操作。
- 停止后，将无法进行数据同步，请谨慎操作。

----结束

### 删除容灾

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，选择“管理 > 容灾管理”。

**步骤3** 在容灾列表中，在指定容灾所在行的“操作”列，单击“**删除**”按钮。

**步骤4** 在弹出框单击“确定”。

此时容灾的“容灾状态”显示为“删除中”，删除需要时间请耐心等待。

 **说明**

- 容灾状态为“创建失败”、“未启动”、“启动失败”、“已停止”、“停止失败”和“异常”时可以执行删除容灾操作。
- 删除后，将无法进行数据同步，且不可恢复，请谨慎操作。

----结束

## 11.8 DWS 集群升级

DWS支持用户通过console自助完成集群升级操作，详情请参见[集群升级](#)。

DWS对集群进行运维操作前，会通过短信进行提示，在运维时间内请谨慎操作集群。

升级过程中会自动重启集群，在此期间集群将短时间无法提供服务。

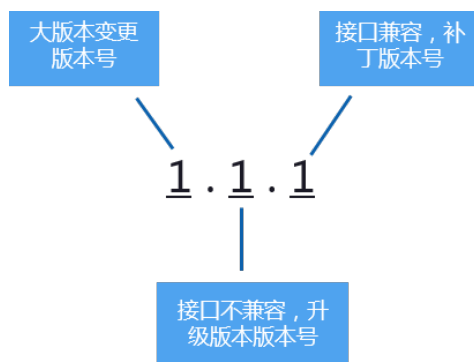
### 说明

- 集群升级到8.1.3及以上版本后，会进入升级观察期，可以观察新业务的运行情况，若发现问题，可快速回退到老版本。
- 集群从9.0.3版本升级到9.1.0版本时，由于9.1.0版本下发的缓存路径disk\_cache\_base\_paths值发生更改，两个磁盘缓存路径中的一个路径对应的disk cache无法从磁盘恢复，在升级后访问时，性能可能会有所下降，但随着查询操作进行，性能会恢复到正常水平。
- 升级集群对原有集群数据和规格都没有影响。

## 升级版本说明

集群版本说明如下图所示：

图 11-5 版本说明



- **服务补丁升级：**表示集群版本X.X.X最后一位数字的升级更新，例如，集群版本从1.1.0升级到1.1.1。
  - 持续时间：整个升级过程将花费不到10分钟。
  - 业务影响：在此期间，如果升级源版本至8.1.3及以上版本，则支持在线补丁，补丁升级期间不要求用户停止业务，业务存在秒级闪断。如果在8.1.3以下版本业务会中断1至3分钟，建议在业务低峰期进行。
- **服务升级：**表示集群版本X.X.X前面两位数字的升级更新，例如，集群版本从1.1.0升级到1.2.0。
  - 持续时间：整个升级过程将花费不到30分钟。
  - 业务影响：在此期间，如果升级源版本至8.1.1及以上版本，则支持在线升级，升级期间不要求用户停止业务，业务存在秒级闪断，建议在业务低峰期进行。
- **热补丁升级：**表示在当前集群版本的基础上扩展一位版本号（格式为0001-9999）进行升级操作。
  - 持续时间：单个热补丁升级过程中花费不到10分钟。

- 业务影响：热补丁升级过程中不影响业务，卸载热补丁后当前热补丁修复的问题可能再次出现。

## 集群升级

### 前提条件

8.1.1及以上集群版本支持用户自助通过console下发集群升级操作。

### 操作步骤

**步骤1** 登录DWS管理控制台。

**步骤2** 在集群列表中单击指定集群名称。

**步骤3** 进入“集群详情”页面，切换至“**升级管理**”页签。

**步骤4** 根据想要升级的操作，在“**类型**”一栏中选择升级或热补丁。

**步骤5** 在升级管理页面，通过目标版本下拉菜单选择集群要升级的版本。

#### 说明

- 在升级操作前如果集群满足巡检条件，需单击“**巡检**”按钮先完成一次巡检，并保证巡检检查通过，通过后可进行下一步变更操作，详情请参见[查看巡检](#)。
- 集群安装热补丁后无法建立容灾。

**步骤6** 单击“**升级**”按钮，根据提示单击“**确定**”按钮，下发升级操作。

**步骤7** 查看集群是否升级成功：

- 如果集群升级成功：8.1.3及以上集群版本，集群升级任务完成后进入业务观察期，若用户验证业务无问题，通过升级管理页面单击“**提交**”按钮，完成集群升级；如果用户验证集群使用性能等有影响，可通过集群升级页面单击“**回滚**”按钮，下发升级回退操作。

#### 说明

- 8.1.3以前集群版本下发升级任务后，完成集群升级前不支持回滚或提交操作。
- 用户下发升级任务成功后，如果没有进行提交操作，则在未提交版本期间会产生wlm线程占用系统存储空间并对性能有一定影响。
- 如果集群升级失败：可单击“**回滚**”操作，回退到原集群版本；或单击“**重试**”按钮重复下发升级操作。

----结束

## 11.9 DWS 集群日志管理

### 11.9.1 DWS 集群支持的日志类型说明

DWS提供数据库审计日志、管理控制台审计日志和其他日志，便于用户进行业务日志查询、问题分析以及掌握产品安全及性能情况。

## 数据库审计日志

DWS通过**安全设置**功能记录数据库执行的DML和DDL等操作信息，用户可以根据审计日志记录对集群状态异常场景进行故障定位和分析，也可以对数据库的历史操作记录进行行为分析和安全审计等，提高DWS的安全性。

关于开启和查看数据库审计日志，请参考[查看DWS数据库审计日志](#)。

## 管理控制台审计日志

DWS通过云审计服务（Cloud Trace Service，CTS）记录DWS管理控制台的关键操作事件，比如创建集群、创建快照、扩容集群、重启集群等。记录的日志可用于支撑安全分析、合规审计、资源跟踪和问题定位等常见应用场景。

关于开启和查看管理控制台相关事件的审计日志信息，请参考[查看DWS管理控制台操作日志](#)。

## 其他日志

DWS通过对接云日志服务（Log Tank Service，以下简称LTS），用户可以在LTS云日志服务查看采集的集群日志或进行日志转储。

当前支持的日志类型为：CN节点日志、DN节点日志、操作系统messages日志、审计日志、cms日志、gtm日志、roach客户端日志、roach服务端日志、升级日志和扩容日志。

### 11.9.2 转储 DWS 数据库审计日志

DWS记录数据库中的连接和用户活动相关信息。这些审计日志信息有助于用户监控数据库以确保安全、进行故障排除及定位历史操作记录。DWS的审计日志默认存储于数据库中，用户可以将审计日志转储到OBS中，便于负责监控数据库中活动的用户查看这些日志信息，目前支持开启审计日志转储和内核审计日志转储，两者功能隔离，可以分别开启或关闭。内核审计日志转储可直接将数据库老化的本地审计日志转储到OBS。

#### 说明

- 非OBS场景下，此功能不可用。
- 内核审计日志转储仅9.1.0.100及以上集群版本支持。
- 经典变更规格、增删CN等变更CN场景会导致数据丢失，建议在执行变更期间关闭审计日志转储。
- CN节点故障可能导致丢失该节点数据。
- 开启审计日志转储后，如果保存的审计日志大小超过1G就会转储，这可能会导致查询结果异常，用户请谨慎操作。
- 开启审计日志转储目录分区版本支持：
  - 针对8.1.3.x集群仅8.1.3.322及以上集群版本支持。
  - 针对8.2.0.x集群仅8.2.0.106及以上集群版本支持。
  - 8.2.1及以上集群版本支持。
  - 历史版本需联系技术支持人员升级使用，升级后用户需手动打开。

## 前提条件

DWS集群创建成功后，可以为集群开启审计日志转储，将审计日志转储到OBS中，方便查看。**开启审计日志转储前需满足如下条件：**

## 开启审计日志转储

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，单击“专属集群 > 集群列表”。

**步骤3** 在集群列表中，单击需要开启审计日志转储的集群的名称，然后在左侧导航栏单击“安全设置”。

**步骤4** 在“审计配置”区域中，开启审计日志转储。

每个区域的每个项目首次开启审计日志转储功能时，系统将提示您需创建名称为“DWSAccessOBS”的委托，委托创建成功后，DWS可以将审计日志转储至OBS中。

默认情况下，只有华为云账号或拥有Security Administrator权限的用户才具备查询委托和创建委托的权限。账号中的IAM用户，默认没有查询委托和创建委托的权限，此时需联系有权限的用户在当前页面完成对DWS的委托授权。

- 转储目录分区：支持审计日志转储时可采用OBS外表读取，转储至OBS时审计日志会采用csv格式且使用gz格式压缩。
- OBS桶：存储审计数据的OBS桶名称。如果没有可选择的OBS桶，可以单击“查看OBS桶”进入OBS管理控制台创建新的OBS桶，具体操作请参见《对象存储服务用户指南》中的“控制台指南 > 管理桶 > 创建桶”章节。
- OBS路径：在OBS中存储审计文件的自定义目录。多级目录可用“/”进行分隔，不能以“/”开头。路径取值范围：1~50个字符。如果填写的OBS路径不存在时，系统会先创建该OBS路径再进行转储。
- 转储周期（分）：根据用户配置的时间，周期性的将数据转储到OBS中。取值范围：5~43200。单位为分钟。

**步骤5** 单击“应用”。

“配置状态”显示为“应用中”，表示系统正在保存配置。

等待一段时间后当“配置状态”显示为“已同步”，表示已保存配置并生效。

----结束

## 开启内核审计日志转储

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，单击“专属集群 > 集群列表”。

**步骤3** 在集群列表中，单击需要开启内核审计日志转储的集群的名称，然后在左侧导航栏单击“安全设置”。

**步骤4** 在“审计配置”区域中，开启内核审计日志转储。

每个区域的每个项目首次内核开启审计日志转储功能时，系统将提示您需创建名称为“DWSAccessOBS”的委托，委托创建成功后，DWS可以将审计日志转储至OBS中。

默认情况下，只有华为云账号或拥有Security Administrator权限的用户才具备查询委托和创建委托的权限。账号中的IAM用户，默认没有查询委托和创建委托的权限，此时需联系有权限的用户在当前页面完成对DWS的委托授权。

- OBS桶：存储内核审计数据的OBS桶名称。如果没有可选择的OBS桶，可以单击“查看OBS桶”进入OBS管理控制台创建新的OBS桶，具体操作请参见《对象存储服务用户指南》中的“控制台指南 > 管理桶 > 创建桶”章节。
- 内核审计日志OBS路径：在OBS中存储内核审计文件的自定义目录。多级目录可用“/”进行分隔，不能以“/”开头。路径取值范围：1~50个字符。如果填写的OBS路径不存在时，系统会先创建该OBS路径再进行转储。

**步骤5** 单击“应用”。

“配置状态”显示为“应用中”，表示系统正在保存配置。

等待一段时间后当“配置状态”显示为“已同步”，表示已保存配置并生效。

**步骤6** 开启内核审计日志转储后，可以通过pg\_query\_audit函数查看转储的审计日志，详情请参见[使用pg\\_query\\_audit函数查看数据库审计日志](#)。

或选择日志存放的OBS桶和文件夹进入查看具体的日志文件，可参见[步骤6](#)。

----结束

## 修改审计日志转储

开启审计日志转储后，可以对转储配置进行修改。例如，修改日志存放的OBS桶和路径，转储周期等。

修改审计日志转储具体操作如下：

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，单击“专属集群 > 集群列表”。

**步骤3** 在集群列表中，单击您想要修改审计日志转储的集群的名称，然后在左侧导航栏单击“安全设置”。

**步骤4** 在“审计配置”区域中，修改审计日志转储配置。

**步骤5** 单击“应用”。

“配置状态”显示为“应用中”，表示系统正在保存配置。

等待一段时间后当“配置状态”显示为“已同步”，表示已保存配置并生效。

----结束

## 查看转储的审计日志

开启审计日志转储后，可以通过OBS查看转储的审计日志。

查看转储的审计日志具体操作如下：

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，单击“专属集群 > 集群列表”。

**步骤3** 在集群列表中，单击需要查看审计日志转储记录的集群的名称，然后在左侧导航栏单击“安全设置”。

**步骤4** 在“审计配置”区域中，单击“查看转储记录”。

**步骤5** 在“审计日志转储记录”弹出框中，单击“查看OBS桶”，进入OBS管理控制台。

**步骤6** 选择日志存放的OBS桶和文件夹进入查看具体的日志文件。

可以选择将日志文件下载，解压并打开查看。审计日志文件字段说明如下。

**表 11-24** 审计日志文件字段说明

| 名称              | 类型                       | 描述                                  |
|-----------------|--------------------------|-------------------------------------|
| begintime       | timestamp with time zone | 操作的执行开始时间。                          |
| endtime         | timestamp with time zone | 操作的执行结束时间。                          |
| operation_type  | text                     | 操作类型，具体类型见 <a href="#">表11-25</a> 。 |
| audit_type      | text                     | 审计类型，具体类型见 <a href="#">表11-26</a> 。 |
| result          | text                     | 操作结果。                               |
| username        | text                     | 执行操作的用户名。                           |
| database        | text                     | 数据库名称。                              |
| client_conninfo | text                     | 客户端连接信息，即gsql, jdbc或odbc等。          |
| object_name     | text                     | 操作对象名称。                             |
| object_details  | text                     | 操作对象详情。                             |
| command_text    | text                     | 操作的执行命令。                            |
| detail_info     | text                     | 执行操作详细信息。                           |
| transaction_xid | text                     | 事务ID。                               |
| query_id        | text                     | 查询ID。                               |
| node_name       | text                     | 节点名称。                               |
| thread_id       | text                     | 线程ID。                               |
| local_port      | text                     | 本地端口。                               |
| remote_port     | text                     | 远端端口。                               |
| result_rows     | text                     | 操作结果行数。                             |
| error_code      | text                     | 错误码。                                |

表 11-25 operation\_type 操作类型项

| 操作类型            | 描述                                                                                                                                                    |
|-----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| audit_switch    | 表示对用户打开和关闭审计日志操作场景进行审计。                                                                                                                               |
| login_logout    | 表示对用户登录和登出操作场景进行审计。                                                                                                                                   |
| system          | 表示对系统的启停、实例切换操作场景进行审计。                                                                                                                                |
| sql_parse       | 表示对SQL语句解析场景进行审计。                                                                                                                                     |
| user_lock       | 表示对用户锁定和解锁操作的场景进行审计。                                                                                                                                  |
| grant_revoke    | 表示对用户权限授予和回收操作场景进行审计。                                                                                                                                 |
| violation       | 表示对用户访问存在越权的场景进行审计。                                                                                                                                   |
| ddl             | 表示对DDL操作场景进行审计，因为DDL操作会根据操作对象进行更细粒度控制，仍然沿用审计开关 audit_system_object，即由audit_system_object控制对哪些对象的DDL操作进行审计（此处不配置ddl，只要配置了 audit_system_object，审计也会生效）。 |
| dml             | 表示对DML操作场景进行审计。                                                                                                                                       |
| select          | 表示对SELECT操作场景进行审计。                                                                                                                                    |
| internal_event  | 表示对内部事件操作场景进行审计。                                                                                                                                      |
| user_func       | 表示对用户自定义函数、存储过程、匿名块操作场景进行审计。                                                                                                                          |
| special_func    | 表示对特殊函数调用操作场景进行审计，特殊函数包括：pg_terminate_backend和pg_cancel_backend。                                                                                      |
| copy            | 表示对COPY操作场景进行审计。                                                                                                                                      |
| set             | 表示对SET操作场景进行审计。                                                                                                                                       |
| transaction     | 表示对事务操作场景进行审计。                                                                                                                                        |
| vacuum          | 表示对VACUUM操作场景进行审计。                                                                                                                                    |
| analyze         | 表示对ANALYZE操作场景进行审计。                                                                                                                                   |
| cursor          | 表示对游标操作的场景进行审计。                                                                                                                                       |
| anonymous_block | 表示对匿名块操作场景进行审计。                                                                                                                                       |
| explain         | 表示对EXPLAIN操作场景进行审计。                                                                                                                                   |
| show            | 表示对SHOW操作场景进行审计。                                                                                                                                      |
| lock_table      | 表示对锁表操作场景进行审计。                                                                                                                                        |
| comment         | 表示对COMMENT操作场景进行审计。                                                                                                                                   |
| preparestmt     | 表示对PREPARE、EXECUTE、DEALLOCATE操作场景进行审计。                                                                                                                |

| 操作类型        | 描述                           |
|-------------|------------------------------|
| cluster     | 表示对CLUSTER操作场景进行审计。          |
| constraints | 表示对CONSTRAINTS操作场景进行审计。      |
| checkpoint  | 表示对CHECKPOINT操作场景进行审计。       |
| barrier     | 表示对BARRIER操作场景进行审计。          |
| cleanconn   | 表示对CLEAN CONNECTION操作场景进行审计。 |
| seclabel    | 表示对安全标签操作进行审计。               |
| notify      | 表示对通知操作进行审计。                 |
| load        | 表示对加载操作进行审计。                 |

表 11-26 audit\_type 审计类型项

| 审计类型                                                                                                     | 描述                                                                                                                                                                                  |
|----------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| audit_open/audit_close                                                                                   | 表示审计类型为打开和关闭审计日志操作。                                                                                                                                                                 |
| user_login/user_logout                                                                                   | 表示审计类型为用户登录/退出成功的操作和用户。                                                                                                                                                             |
| system_start/system_stop/<br>system_recover/system_switch                                                | 表示审计类型为系统的启停、实例切换操作。                                                                                                                                                                |
| sql_wait/sql_parse                                                                                       | 表示审计类型为SQL语句解析。                                                                                                                                                                     |
| lock_user/unlock_user                                                                                    | 表示审计类型为用户锁定和解锁成功的操作。                                                                                                                                                                |
| grant_role/revoke__role                                                                                  | 表示审计类型为用户权限授予和回收的操作。                                                                                                                                                                |
| user_violation                                                                                           | 表示审计类型为用户访问存在越权的操作。                                                                                                                                                                 |
| ddl_数据库对象                                                                                                | 表示审计类型为DDL操作，因为DDL操作由会根据操作对象进行更细粒度控制，仍然沿用审计开关audit_system_object，即由audit_system_object控制对哪些对象的DDL操作进行审计（此处不配置ddl，只要配置了audit_system_object，审计也会生效）。<br>例如：ddl_sequence表示审计类型为序列相关操作。 |
| dml_action_insert/<br>dml_action_delete/<br>dml_action_update/<br>dml_action_merge/<br>dml_action_select | 表示审计类型为INSERT、DELETE、UPDATE、MERGE等DML操作。                                                                                                                                            |
| internal_event                                                                                           | 表示审计类型为内部事件。                                                                                                                                                                        |

| 审计类型                                                                                                                                                                      | 描述                                                             |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------|
| user_func                                                                                                                                                                 | 表示审计类型为用户自定义函数、存储过程、匿名块操作。                                     |
| special_func                                                                                                                                                              | 表示审计类型为特殊函数调用操作，特殊函数包括：pg_terminate_backend和pg_cancel_backend。 |
| copy_to/copy_from                                                                                                                                                         | 表示审计类型为COPY相关操作。                                               |
| set_parameter                                                                                                                                                             | 表示审计类型为SET操作。                                                  |
| trans_begin/trans_commit/<br>trans_prepare/<br>trans_rollback_to/<br>trans_release/trans_savepoint/<br>trans_commit_prepare/<br>trans_rollback_prepare/<br>trans_rollback | 表示审计类型为事务相关操作。                                                 |
| vacuum/vacuum_full/<br>vacuum_merge                                                                                                                                       | 表示审计类型为VACUUM相关操作。                                             |
| analyze/analyze_verify                                                                                                                                                    | 表示审计类型为ANALYZE相关操作。                                            |
| cursor_declare/cursor_move/<br>cursor_fetch/cursor_close                                                                                                                  | 表示审计类型为游标相关操作。                                                 |
| codeblock_execute                                                                                                                                                         | 表示审计类型为匿名块。                                                    |
| explain                                                                                                                                                                   | 表示审计类型为EXPLAIN操作。                                              |
| show                                                                                                                                                                      | 表示审计类型为SHOW操作。                                                 |
| lock_table                                                                                                                                                                | 表示审计类型为锁表操作。                                                   |
| comment                                                                                                                                                                   | 表示审计类型为COMMENT操作。                                              |
| prepare/execute/deallocate                                                                                                                                                | 表示审计类型为PREPARE、EXECUTE或DEALLOCATE操作。                           |
| cluster                                                                                                                                                                   | 表示审计类型为CLUSTER操作。                                              |
| constraints                                                                                                                                                               | 表示审计类型为CONSTRAINTS操作。                                          |
| checkpoint                                                                                                                                                                | 表示审计类型为CHECKPOINT操作。                                           |
| barrier                                                                                                                                                                   | 表示审计类型为BARRIER操作。                                              |
| cleanconn                                                                                                                                                                 | 表示审计类型为CLEAN CONNECTION操作。                                     |
| seclabel                                                                                                                                                                  | 表示审计类型为安全标签操作。                                                 |
| notify                                                                                                                                                                    | 表示审计类型为通知操作。                                                   |
| load                                                                                                                                                                      | 表示审计类型为加载操作。                                                   |

----结束

## 关闭审计日志转储/内核审计日志转储

审计日志转储/内核审计日志转储开启后，如果不需要将审计日志转储/内核审计日志转储到OBS中，可以将审计日志转储/内核审计日志转储关闭。

**步骤1** 登录DWS管理控制台。

**步骤2** 在左侧导航栏中，单击“专属集群 > 集群列表”。

**步骤3** 在集群列表中，单击需要关闭审计日志转储/内核审计日志转储的集群的名称，然后在左侧导航栏单击“安全设置”。

**步骤4** 在“审计配置”区域中，关闭审计日志转储/内核审计日志转储开关。

**步骤5** 单击“应用”。

“配置状态”显示为“应用中”，表示系统正在保存配置。

等待一段时间后当“配置状态”显示为“已同步”，表示已保存配置并生效。

----结束

## 11.9.3 查看 DWS 数据库审计日志

数据库审计日志在集群的“安全设置”页面中进行设置，仅“可用”和“非均衡”状态的集群支持修改安全配置，同时集群的任务信息不能处于“节点扩容”、“配置中”、“升级中”、“重分布中”和“重启中”等状态。

### 前提条件

- 审计功能总开关audit\_enabled已开启。（audit\_enabled默认值为ON，若关闭请参考[修改DWS集群GUC参数](#)设置为OFF）。
- 已配置需要审计的审计项。各审计项及其开启办法，请参考[设置数据库审计日志](#)。
- 数据库正常运行，并且对数据库执行了一系列增、删、改、查操作，保证在查询时段内有审计结果产生。
- 数据库各个节点审计日志单独记录。
- 只有拥有AUDITADMIN属性的用户才可以查看审计记录。

### 设置数据库审计日志

**步骤1** 登录DWS管理控制台。

**步骤2** 单击“专属集群 > 集群列表”。

**步骤3** 在集群列表中，单击指定集群的名称，然后在左侧导航栏单击“安全设置”。

默认显示“配置状态”为“已同步”，表示页面显示的是数据库当前最新结果。

**步骤4** 在“审计配置”区域中，根据需要设置以下操作的审计开关：

 说明

审计日志保留策略默认为“空间优先”，表示当单个节点的审计日志超过1G后，将自动淘汰审计日志。该功能避免因审计日志占用磁盘空间高导致节点故障或性能低。

各审计项的详细信息如表11-27所示。

表 11-27 审计项

| 审计项    | 说明                                                                                                                                                                                                                                                                                                                                      |
|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 越权访问操作 | 表示是否记录用户的越权访问操作，默认关闭。                                                                                                                                                                                                                                                                                                                   |
| DQL操作  | 可选SELECT操作进行审计。<br><b>说明</b><br>8.1.1.100及以上集群版本支持“DQL操作”审计项。                                                                                                                                                                                                                                                                           |
| DML操作  | 表示是否对数据表的INSERT、UPDATE和DELETE操作进行记录，默认关闭。<br><b>说明</b><br>8.1.1.100及以上版本支持各审计项的细粒度拆分，并增加COPY，MERGE选项。                                                                                                                                                                                                                                   |
| DDL操作  | 表示是否对指定数据库对象的CREATE、DROP和ALTER操作进行记录。除“DATABASE”、“SCHEMA”和“USER”默认启用记录，其他默认关闭。<br><b>说明</b><br>8.1.1.100及以上版本新增“TABLE”、“DATA SOURCE”和“NODE GROUP”操作，默认启用。                                                                                                                                                                               |
| 其他操作   | 表示对其他的操作进行记录，其中TRANSACTION，CURSOR操作，默认勾选，VACUUM，ANALYZE，USER FUNCTION，SPECIAL FUNCTION，PREPARE STATEMENT默认关闭。<br><b>说明</b> <ul style="list-style-type: none"><li>8.1.1.100及以上版本支持“其他操作”审计项。</li><li>建议TRANSACTION审计项保持勾选，否则事务内语句将不会审计。</li><li>建议CURSOR审计项保持勾选，否则CURSOR内SELECT语句将不会被审计。需注意Data Studio客户端会自动给SELECT语句封装CURSOR。</li></ul> |

DWS除了支持表11-27的审计功能，默认还开启了如表11-28所示的关键审计项。

表 11-28 关键审计项

| 参数名   | 说明                    |
|-------|-----------------------|
| 关键审计项 | 记录用户登录成功、登录失败和注销的信息。  |
|       | 记录数据库启动、停止、恢复和切换审计信息。 |
|       | 记录用户锁定和解锁功能信息。        |

| 参数名 | 说明               |
|-----|------------------|
|     | 记录用户权限授予和权限回收信息。 |
|     | 记录SET操作的审计功能。    |

**步骤5** 设置是否开启审计日志转储功能。

关于审计日志转储功能的更多信息，请参见[开启审计日志转储](#)。

**步骤6** 单击“应用”。

“配置状态”显示为“应用中”，表示系统正在保存配置。

等待一段时间后当“配置状态”显示为“已同步”，表示已保存配置并生效。

此外，单击刷新图标，可刷新配置信息。

----结束

## 查看数据库审计日志方式

方式一：由于审计日志会占用一定磁盘空间，为了防止本地磁盘文件过大，DWS支持审计日志转储，用户可以开启审计日志转储功能，将审计日志转储到OBS（用户需创建用于存储审计日志的OBS桶）中进行查看或下载，详细内容请参考[开启审计日志转储](#)。

方式二：通过依赖于云日志服务LTS的“集群日志管理”功能查看采集的审计数据库日志或进行日志下载，详细内容请参考[查看集群日志](#)。

方式三：数据库的审计日志默认存储于数据库中，连接集群后使用pg\_query\_audit函数进行查看。详细内容请参考[使用pg\\_query\\_audit函数查看数据库审计日志](#)。

## 使用 pg\_query\_audit 函数查看数据库审计日志

**步骤1** 使用SQL客户端工具成功连接集群，连接方式请参考[连接DWS集群](#)。

**步骤2** 使用函数pg\_query\_audit查询当前CN节点的审计日志：

```
pg_query_audit(timestampz starttime,timestampz endtime,audit_log)
```

参数starttime和endtime分别表示审计记录的开始时间和结束时间，audit\_log表示所查看的审计日志信息所在的物理文件路径，当不指定audit\_log时，默认查看连接当前实例的审计日志信息。

例如，查看指定时间段当前CN节点审计记录。

```
SELECT * FROM pg_query_audit('2021-02-23 21:49:00','2021-02-23 21:50:00');
```

查询结果如下：

| begin time                | end time                  | operation_type | audit_type                                   | result                                     | username    | database |
|---------------------------|---------------------------|----------------|----------------------------------------------|--------------------------------------------|-------------|----------|
| client_conninfo           | object_name               | command_text   | detail_info                                  |                                            |             |          |
| transaction_xid           | query_id                  | node_name      | session_id                                   | local_port                                 | remote_port |          |
| 2021-02-23 21:49:57.76+08 | 2021-02-23 21:49:57.82+08 | login_logout   | user_login                                   | ok                                         | dbadmin     | gaussdb  |
| gsq[local]                | gaussdb                   | login db       | login db(gaussdb)                            | successfully, the current user is: dbadmin | 0           | 0        |
|                           |                           | coordinator1   | 140324035360512.667403397820909.coordinator1 | 27777                                      |             |          |

该条记录表明，用户dbadmin在2021-02-23 21:49:57.82+08登录数据库gaussdb。其中client\_conninfo字段在log\_hostname启动且IP连接时，字符@后显示反向DNS查找得到的主机名。

### 步骤3 使用函数pgxc\_query\_audit可以查询所有CN节点的审计日志：

**pgxc\_query\_audit(timestampz starttime,timestampz endtime)**

例如，查看指定时间段所有CN节点审计记录。

```
SELECT * FROM pgxc_query_audit('2021-02-23 22:05:00','2021-02-23 22:07:00') where audit_type = 'user_login' and username = 'user1';
```

查询结果如下：

| begin time                 | end time                   | operation_type | audit_type | result      | username | database | client_conninfo | object_name | command_text | detail_info                                                    | transaction_xid |
|----------------------------|----------------------------|----------------|------------|-------------|----------|----------|-----------------|-------------|--------------|----------------------------------------------------------------|-----------------|
| query_id                   | node_name                  | session_id     | local_port | remote_port |          |          |                 |             |              |                                                                |                 |
| 2021-02-23 22:06:22.219+08 | 2021-02-23 22:06:22.271+08 | login_logout   | user_login | ok          | user1    | gaussdb  | gsq@[local]     | gaussdb     | login db     | login db(gaussdb) successfully, the current user is: user1   0 | 0               |
| 2021-02-23 22:05:51.697+08 | 2021-02-23 22:05:51.749+08 | login_logout   | user_login | ok          | user1    | gaussdb  | gsq@[local]     | gaussdb     | login db     | login db(gaussdb) successfully, the current user is: user1   0 | 0               |
|                            |                            |                |            |             |          |          |                 |             |              |                                                                |                 |
|                            |                            |                |            |             |          |          |                 |             |              |                                                                |                 |

查询结果显示，用户user1在CN1和CN2的成功登录记录。

### 步骤4 查询审计多个对象名的记录。

SET audit\_object\_name\_format TO 'all';

```
SELECT object_name,result,operation_type,command_text FROM pgxc_query_audit('2022-08-26 8:00:00','2022-08-26 22:55:00') where command_text like '%student%';
```

查询结果如下：

| object_name                                                      | result | operation_type | command_text                                                                                                                                       |
|------------------------------------------------------------------|--------|----------------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| student                                                          | ok     | ddl            | CREATE TABLE student(stuNo int, stuName TEXT);                                                                                                     |
| studentscore                                                     | ok     | ddl            | CREATE TABLE studentscore(stuNo int, stuscore int);                                                                                                |
| ["public.student_view01","public.studentscore","public.student"] | ok     | ddl            | CREATE OR REPLACE VIEW student_view01 AS SELECT * FROM student t1 where t1.stuNo in (select stuNo from studentscore t2 where t1.stuNo = t2.stuNo); |
| ["public.student_view01","public.student","public.studentscore"] | ok     | dml            | SELECT * FROM student_view01;                                                                                                                      |

查询结果object\_name显示多个对象名，包括：表名、视图名及视图所涉及的基表名。

----结束

## 11.9.4 查看 DWS 管理控制台操作日志

### 开启审计服务

使用云审计服务前需要开启云审计服务，开启云审计服务后系统会自动创建一个追踪器，系统记录的所有操作将关联在该追踪器中。目前，一个云账户系统仅支持创建一个追踪器。

**步骤1** 登录管理控制台，选择“服务列表 > 管理与监管 > 云审计服务”，进入云审计服务信息页面。

**步骤2** 单击左侧导航树的“追踪器”，进入追踪器信息页面。

**步骤3** 开启云审计服务。

如果您是首次使用云审计服务，在追踪器列表中还没有已创建的追踪器，则请参考《云审计服务用户指南》中的“入门 > 开启云审计服务”开启云审计服务。

如果您已开通云审计服务了，开通时系统已为您自动创建了一个管理事件追踪器，管理事件追踪器只能有一个且不可删除。您也可以自行创建数据事件追踪器，详细内容请参见《云审计服务用户指南》中的“管理追踪器 > 创建追踪器”。

----结束

## 关闭审计日志

如果用户想关闭审计日志，需要在云审计服务中停用追踪器。

**步骤1** 登录管理控制台，选择“服务列表 > 管理与监管 > 云审计服务”，进入云审计服务信息页面。

**步骤2** 通过停用追踪器，关闭审计日志。如需重新开启审计日志，只要启用追踪器即可。

有关停用/启用追踪器的更多信息，请参考《云审计服务用户指南》中的“管理追踪器 > 停用/启用追踪器”章节。

----结束

## 支持审计的关键操作列表

通过云审计服务，您可以记录与DWS服务相关的操作事件，便于日后的查询、审计和回溯。

### 说明

- 自动快照的创建、删除走系统内部调度，非用户操作，不记录审计日志。
- 因DWS集群操作事件较多，下表仅展示部分高频支持审计的DWS操作。

表 11-29 云审计服务支持审计的 DWS 操作列表

| 操作名称    | 资源类型    | 事件名称                |
|---------|---------|---------------------|
| 创建集群    | cluster | createCluster       |
| 删除集群    | cluster | deleteCluster       |
| 集群巡检    | cluster | createInspection    |
| 终止巡检    | cluster | AbortInspection     |
| 扩容集群    | cluster | growCluster         |
| 空闲节点扩容  | cluster | resizeWithFreeNodes |
| 集群重分布   | cluster | redistributeCluster |
| 查询重分布详情 | cluster | queryRedisInfo      |
| 磁盘扩容    | cluster | executeDiskExpand   |

| 操作名称      | 资源类型    | 事件名称                        |
|-----------|---------|-----------------------------|
| 集群规格变更    | cluster | flavorResize                |
| 重启集群      | cluster | restartCluster              |
| 集群主备切换    | cluster | activeStandbySwitchover     |
| 重置密码      | cluster | resetPassword               |
| 集群修复      | cluster | repairCluster               |
| 创建集群连接信息  | cluster | createClusterConnection     |
| 修改集群连接信息  | cluster | modifyClusterConnection     |
| 删除集群连接信息  | cluster | deleteClusterConnection     |
| 调整集群大小    | cluster | resizeCluster               |
| 绑定或解绑弹性IP | cluster | bindOrUnbindEIP             |
| 创建或绑定Elb  | cluster | createOrBindElb             |
| 解绑Elb     | cluster | unbindElb                   |
| 添加CN      | cluster | addCN                       |
| 删除CN      | cluster | deleteCN                    |
| 集群升级      | cluster | clusterUpdateMgr            |
| 集群缩容      | cluster | shrinkCluster               |
| 添加资源管理计划  | cluster | addWorkloadPlan             |
| 删除资源池     | cluster | deleteWorkloadQueueInfo     |
| 添加资源池     | cluster | addWorkloadQueueInfo        |
| 修改集群guc参数 | cluster | updateClusterConfigurations |
| 解除只读      | cluster | cancelReadonly              |
| 修改运维时间窗   | cluster | modifyMaintenanceWindow     |
| 批量增加CN节点  | cluster | batchCreateCn               |
| 批量删除CN节点  | cluster | batchDeleteCn               |
| 批量添加标签    | cluster | batchCreateResourceTag      |
| 批量删除标签    | cluster | batchDeleteResourceTag      |
| 创建逻辑集群    | cluster | createLogicalCluster        |
| 删除逻辑集群    | cluster | deleteLogicalCluster        |
| 编辑逻辑集群    | cluster | editLogicalCluster          |
| 重启逻辑集群    | cluster | restartLogicalCluster       |

| 操作名称     | 资源类型             | 事件名称                             |
|----------|------------------|----------------------------------|
| 转换到逻辑集群  | cluster          | switchLogicalCluster             |
| 启动集群     | cluster          | startCluster                     |
| 停止集群     | cluster          | stopCluster                      |
| 修改集群安全组  | cluster          | changeSecurityGroup              |
| 修改集群时区   | cluster          | modifyClusterTimezone            |
| 创建快照     | backup           | createBackup                     |
| 删除快照     | backup           | deleteBackup                     |
| 恢复集群     | backup           | restoreCluster                   |
| 复制快照     | backup           | copySnapshot                     |
| 删除快照策略   | backup           | deleteBackupPolicy               |
| 更新快照策略   | backup           | updateClustersBackupPolicy       |
| 创建容灾     | disasterRecovery | createDisasterRecovery           |
| 删除容灾     | disasterRecovery | deleteDisasterRecovery           |
| 启动容灾     | disasterRecovery | startDisasterRecoveryAction      |
| 停止容灾     | disasterRecovery | stopDisasterRecoveryAction       |
| 灾备切换     | disasterRecovery | switchoverDisasterRecoveryAction |
| 异常切换     | disasterRecovery | failoverDisasterRecoveryAction   |
| 恢复容灾     | disasterRecovery | recoveryDisaster                 |
| 更新容灾配置   | disasterRecovery | updateRecoveryDisaster           |
| 查询容灾详情   | disasterRecovery | disasterRecoveryOperate          |
| 设置安全参数   | configurations   | updateConfigurations             |
| 创建扩展资源   | dataSource       | createExtDataSource              |
| 删除扩展资源   | dataSource       | deleteExtDataSource              |
| 更新扩展资源   | dataSource       | updateExtDataSource              |
| 创建MRS数据源 | dataSource       | createExtDataSource              |
| 删除MRS数据源 | dataSource       | deleteExtDataSource              |
| 更新MRS数据源 | dataSource       | updateExtDataSource              |

## 查看审计日志

**步骤1** 登录管理控制台，选择“服务列表 > 管理与监管> 云审计服务”，进入云审计服务信息页面。

**步骤2** 单击左侧导航树的“事件列表”，进入事件列表信息页面。

**步骤3** 单击事件列表上方的搜索框，设置对应的操作事件条件。

当前事件列表支持十个维度的组合查询，详细信息如下：

- “事件名称”：选择该选项时，还需选择某个具体的事件名称。
- “云服务”：选择“DWS”。
- “资源类型”：选择“所有资源类型”，或者指定具体的资源类型。
- “资源名称”：选择该选项时，还需选择或手动输入某个具体的资源名称。
- “资源ID”：选择该选项时，还需选择或者手动输入某个具体的资源ID。
- “操作用户”：在下拉框中选择某一具体的操作用户，此操作用户指用户级别，而非租户级别。
- “事件ID”：选择该选项时，还需选择或者手动输入某个具体的事件ID。
- “事件级别”：可选项为“所有事件级别”、“normal”、“warning”、“incident”，只可选择其中一项。
- “企业项目ID”：选择该选项时，还需选择或者手动输入某个具体的企业项目ID。
- “访问密钥ID”：选择该选项时，还需选择或者手动输入某个具体的访问密钥ID。

**步骤4** 单击“查询”，查看对应的操作事件。

**步骤5** 单击需要查看的事件名称，弹出一个窗口，显示了该操作事件结构的详细信息。

关于云审计服务事件结构的关键字段详解，请参见《云审计服务用户指南》中的“云审计服务事件参考 > 事件结构”和“云审计服务事件参考 > 事件样例”章节。

----结束

## 11.9.5 查看 DWS 集群其他日志

### 概述

集群日志服务用于采集集群日志并报送云日志服务（Log Tank Service，以下简称 LTS），用户可以在 LTS 云日志服务查看采集的集群日志或进行日志转储。

当前支持的日志类型为：CN 节点日志、DN 节点日志、操作系统 messages 日志、审计日志、cms 日志、gtm 日志、roach 客户端日志、roach 服务端日志、升级日志和扩容日志。

#### 说明

- 集群日志管理仅 8.1.1.300 及以上版本支持。
- cms 日志、gtm 日志、roach 客户端日志、roach 服务端日志、扩容日志、升级日志仅 8.3.0 及以上版本支持。

开启日志服务对接

- 步骤1 登录DWS管理控制台。
- 步骤2 单击“专属集群 > 集群列表”。默认显示用户所有的集群列表。
- 步骤3 在集群列表中，单击指定集群名称进入“集群详情”页面，左侧导航栏单击“日志服务”进入页签。
- 步骤4 在“日志服务”页签，单击“云日志服务对接”开关，首次开启时，将弹出提示框，确认无误后，单击“是”，进行授权委托。

说明

- 若已开启过LTS云日志服务对接，并授权该服务创建委托，再次开启时则无需授权操作。
- 默认情况下，只有华为云账号或拥有Security Administrator权限的用户才具备查询委托和创建委托的权限。账号中的IAM用户，默认没有查询委托和创建委托的权限，此时需联系有权限的用户在当前页面完成对DWS的委托授权。
- 开启LTS云日志服务对接时，还需授予用户LTS服务相关权限策略（LTS Admin、LTS Administrator、LTS FullAccess和LTS ReadOnlyAccess）。

----结束

查看集群日志

- 步骤1 登录DWS管理控制台。
- 步骤2 单击“专属集群 > 集群列表”。默认显示用户所有的集群列表。
- 步骤3 在集群列表中，单击指定集群名称进入“集群详情”页面，左侧导航栏单击“日志服务”进入页签。
- 步骤4 在“日志服务”页面，单击指定日志类型后的“查看日志”按钮，即可跳转到LTS云日志服务查看日志。

| 云日志服务对接 | 日志类型             | 描述信息       | 操作   |
|---------|------------------|------------|------|
|         | messages         | 操作系统日志     | 查看日志 |
|         | expand           | 扩展日志       | 查看日志 |
|         | roach-controller | roach控制器日志 | 查看日志 |
|         | audit            | 审计日志       | 查看日志 |
|         | gms              | gms日志      | 查看日志 |
|         | roach-agent      | roach客户端日志 | 查看日志 |
|         | cms              | cms日志      | 查看日志 |
|         | DN               | dms-DN日志   | 查看日志 |
|         | upgrade          | 升级日志       | 查看日志 |
|         | DN               | dms-DN日志   | 查看日志 |

----结束

关闭日志服务对接

- 步骤1 登录DWS管理控制台。
- 步骤2 单击“专属集群 > 集群列表”。默认显示用户所有的集群列表。
- 步骤3 在集群列表中，单击指定集群名称进入“集群详情”页面，左侧导航栏单击“日志服务”进入页签。
- 步骤4 单击“云日志服务对接”开关，弹出“关闭云日志服务”提示框。

**步骤5** 确认无误后，单击提示框中的“确认”按钮，即可关闭日志采集。

----结束

## 11.10 DWS 集群状态异常操作

### 解除只读

当集群进入只读状态时，无法进行数据库写入相关操作，用户可以在管理控制台解除集群的只读状态。触发只读状态可能是由于磁盘使用率过高，因此需要对集群数据进行清理，详情请参见《数据仓库服务故障排除》中“磁盘使用率高&集群只读处理方案”章节。

#### 说明

- 解除只读支持1.7.2及以上集群版本。
- 8.2.0集群版本开始支持只读状态下用户可以通过开启读写事务DROP/TRUNCATE TABLE的方式清理磁盘空间。

#### 对系统的影响

- 集群状态为“只读”时，可进行解除只读操作。
- 集群处于“只读”状态时，应停止写入任务，避免磁盘被写满造成数据丢失的风险。
- 解除只读操作成功后，尽快清理数据，避免集群过一段时间再次进入“只读”状态。

#### 操作步骤

**步骤1** 登录DWS管理控制台。

**步骤2** 单击“专属集群 > 集群列表”。

默认显示用户所有的集群列表。

**步骤3** 在指定集群所在行“集群状态”列，在“只读”状态下单击“解除只读”。

**步骤4** 在弹出框中，单击“确定”，再次进行解除只读确认，对集群进行解除只读操作。

----结束

### 集群主备恢复

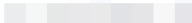
当集群状态为“非均衡”时会出现某些节点主实例增多，从而负载压力较大。这种情况下集群状态是正常的，但整体性能要低于均衡状态。可参考本章节进行集群主备恢复操作将集群状态切换为“可用”状态。

#### 说明

- 集群主备恢复仅8.1.1.202及以上集群版本支持。
- 集群主备恢复将会短暂中断业务，中断时间根据用户自身业务量所决定，建议用户在业务低峰期执行此操作。

**步骤1** 登录DWS管理控制台。

- 步骤2** 在“专属集群 > 集群列表”页面，找到非均衡状态的集群。
- 步骤3** 在指定集群所在行“集群状态”列，“非均衡”状态下单击“一键修复”。

| 集群名称                                                                                  | 集群状态          | 任务信息 ① | 节点规格        | 近期事件 | 操作                                                                   |
|---------------------------------------------------------------------------------------|---------------|--------|-------------|------|----------------------------------------------------------------------|
| ▼  1 | 非均衡 ②<br>一键修复 | --     | dwsxlarge.4 | 2    | <a href="#">监控面板</a>   <a href="#">查看监控指标</a>   <a href="#">更多</a> ▼ |

- 步骤4** 在弹出对话框中，确认业务处于低峰期，然后再进行主备恢复确认，确认无误后单击“是”后页面右上角显示主备恢复请求下发成功。
- 步骤5** 此时集群进行主备恢复操作，集群的任务信息变为“主备恢复中”，当集群主备恢复成功后，集群状态变为“可用”。

----结束

## 11.11 Vacuum 回收 DWS 空间

### 11.11.1 智能运维概览

DWS提供智能运维功能，帮助用户快速高效地执行运维任务。智能运维会根据集群负载情况，选择合理时间窗、并发度完成用户指定的任务。在运维任务执行过程中，智能运维会时刻关注用户业务的变化，及时调整运维任务执行策略，以减轻对用户业务的影响。智能运维支持周期型和单次型任务的创建，执行时间窗可按照不同用户业务负载定制化。

智能运维具备一定的高可用性，在集群异常的情况下，智能运维将重新执行失败的运维任务，若由于集群异常导致运维任务部分步骤无法完成，智能运维将尝试跳过失败的步骤，以节省用户运维时间窗开销。

智能运维界面主要由以下部分构成：

- 运维任务公共配置：
  - 用户表VacuumFull运维任务最大并发数：配置应用于每个用户表VacuumFull运维任务，建议依据剩余磁盘空间和时间窗内IO负载情况合理配置。配置范围为1~24，建议配置值为5。
  - 小CU阈值：根据小CU阈值来判定是不是小CU表，小于阈值为小CU表，小于等于该值为触发Vacuum的条件，越大触发越敏感，默认推荐值为1000。
  - 小CU占比：小CU表占全部CU比率，大于等于该值为触发Vacuum的条件，越小触发越敏感，默认推荐值为50%。
- 进行中的运维任务：显示正在运行的运维任务信息（目前仅支持Vacuum运维任务，用户数据膨胀，磁盘内存不足可进行Vacuum操作）。
  - 用户频繁创建、删除表，导致系统表膨胀严重，可对系统表执行Vacuum。
  - 用户频繁执行UPDATE、DELETE语句，导致用户表膨胀严重，可对用户表执行Vacuum/Vacuum Full。
- 运维详情：运维详情包括两部分，运维计划和运维状态。运维计划将展示全部运维任务基本信息，运维状态将展示运维任务运行状态信息。

 说明

- 该特性仅8.1.3及以上版本支持。其中小CU阈值、小CU占比仅9.1.0.200及以上集群版本支持，如果版本不符合将不展示该字段。
- VacuumFull运维任务完成后系统会自动进行Analyze操作，用户无需手动。
- 运维任务公共配置模块仅8.1.3及以上集群版本支持，历史版本需要联系技术支持人员升级使用。

## 11.11.2 管理运维计划

### 前提条件

- 对于自动Vacuum运维任务，用户配置时间窗时，应避开业务高峰期。否则自动Vacuum将有可能与用户业务产生死锁冲突。
- Vacuum Full运维任务对于用户表并发度最高为24，最低为0；对于系统表并发度最高为1，最低为0。并发度不支持用户自定义，但可根据系统io\_util自动调节：
  - io\_util将0%~60%划分为两个区间段：
    - 0%~30%区间段，io\_util每减少15%，并发度增加2。
    - 30%~60%区间段，io\_util值每减少15%，并发度增加1。
  - 60%~70%并发度不变。
  - 70%以上并发度减1，如果仍维持70%以上IO，则并发度持续减到0。
- 调度器会在时间窗内，扫描列存小CU膨胀情况，若发现列存CU膨胀（列存CU的膨胀依据是CU平均记录数小于1000），则调度器会将该列存表优先级提前。列存CU膨胀检测条件不受膨胀率、可回收空间限制。
- 优先级表最多支持添加100个任务表。
- 调度器自动vacuum功能依赖于统计信息，若统计信息不准确，可能会影响调度器自动vacuum执行顺序以及统计结果。
- 调度器不支持带有空格和单引号的表，包括数据库名，schema名，表名。如果vacuum在扫描过程中遇到带有单引号或空格的表，将自动跳过。对于优先级表中有空格或单引号的表，也自动跳过。

### 运维任务公共配置

#### 注意事项

- 该配置应用于每个用户表VacuumFull运维任务中，其中正在运行中的任务不影响，周期性任务新配置在下次任务执行时生效。

#### 操作步骤

**步骤1** 登录DWS管理控制台。

**步骤2** 在集群列表中单击指定集群名称。

**步骤3** 进入“集群详情”页面，切换至“智能运维”页签。

**步骤4** 在页面上方“运维任务公共配置”模块，修改下列公共配置值。

- **用户表VacuumFull运维任务最大并发数：**配置应用于每个用户表VacuumFull运维任务，建议依据剩余磁盘空间和时间窗内IO负载情况合理配置。配置范围为1~24，用户可根据剩余磁盘空间和IO负载，合理设置最大并发数，建议配置值为

5. 其中正在运行中的任务不影响，如果是周期性任务，新设置的并发数在下次运维任务运行时生效。

- **小CU阈值**：根据小CU阈值来判定是不是小CU表，小于阈值为小CU表，小于等于该值为触发Vacuum的条件，越大触发越敏感，默认推荐值为1000。
- **小CU占比**：小CU表占全部CU比率，大于等于该值为触发Vacuum的条件，越小触发越敏感，默认推荐值为50%。

步骤5 确认无误后，单击“保存”完成配置。

----结束

添加运维计划

- 步骤1 登录DWS管理控制台。
- 步骤2 在集群列表中单击指定集群名称。
- 步骤3 进入“集群详情”页面，切换至“智能运维”页签。
- 步骤4 在运维详情部分切换至运维计划模块，单击“添加运维任务”按钮。
- 步骤5 在弹出的添加运维任务边栏，进行运维任务的基础配置。

表 11-30 运维任务基础配置项

| 配置项名称 | 配置项描述                                                                                                                                                                                                                                                                                                                                                       | 样例                                |
|-------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------|
| 运维任务  | Vacuum（目前仅支持Vacuum运维任务）。                                                                                                                                                                                                                                                                                                                                    | Vacuum                            |
| 任务简述  | 智能任务的简要描述。                                                                                                                                                                                                                                                                                                                                                  | 该运维任务可帮助用户定期调用系统Vacuum命令，以实现空间回收。 |
| 备注    | 备注信息。                                                                                                                                                                                                                                                                                                                                                       | -                                 |
| 调度模式  | 支持以下三种调度模式： <ul style="list-style-type: none"><li>● <b>自动</b>：智能运维将在指定时间窗内扫描数据库，依据用户业务负载、用户表可回收空间，自动下发表级Vacuum任务。</li><li>● <b>指定目标</b>：用户可选择指定Vacuum目标，智能运维将在指定时间窗内，自动下发表级Vacuum任务。</li><li>● <b>优先级模式</b>：用户可配置优先Vacuum目标，若时间窗剩余，智能运维将自动扫描其他可Vacuum的表，并下发表级Vacuum任务。</li></ul> <b>说明</b><br>建议使用指定目标方式进行Vacuum和VacuumFull，同时不建议对列存宽表进行VacuumFull，以防止内存膨胀。 | 指定目标                              |

| 配置项名称        | 配置项描述                                                                                                                                                                                                                                                                                                                                                                                                                               | 样例                                  |
|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------|
| 自动 Vacuum 目标 | <p>支持：系统表 Vacuum 或用户表 Vacuum Full。</p> <ul style="list-style-type: none"><li>对于系统表 Vacuum，会持有系统表四级锁（share update exclusive 锁），不会阻塞用户业务，仅会阻塞系统表 DDL 进程。</li><li>对于用户表 Vacuum Full，会持有系统表的八级锁（access exclusive 锁），期间所有访问会被阻塞，并等待 Vacuum Full 结束，建议用户合理安排调度时间，在业务负载低峰期执行 Vacuum Full 操作，避免锁表影响业务。</li></ul> <p><b>注意</b><br/>Vacuum Full 操作相当于开辟一块和表数据大小相同的空间（表真实数据大小约为表总大小 * (1 - 脏页率)），因此表空间会先增后降，请提前计算好 Vacuum Full 所需要的空间再行处理。</p> | 用户表 Vacuum Full                     |
| 优先 Vacuum 目标 | 用户可配置优先 Vacuum 目标，其中一行对应一张表，每张表以数据库名、模式名、表名表示，以空格进行分割。                                                                                                                                                                                                                                                                                                                                                                              | -                                   |
| 高级配置         | <p>选择“自定义”时，可以设置 Vacuum 膨胀率和目标表可回收空间高级配置参数（满足其中一个条件即触发自动 Vacuum），如果选择“默认配置”，将使用它们的默认值。</p> <p><b>说明</b><br/>Vacuum 膨胀率：在数据库中频繁执行 UPDATE、DELETE 等操作后被删除或更新的行不会从表中物理删除，仅从数据库中被逻辑删除，在完成 VACUUM 之前这些过期数据仍然存储在磁盘中，从而导致表膨胀。当膨胀率达到运维任务中用户设置的百分比后，就会自动触发 Vacuum。</p>                                                                                                                                                                        | 默认配置（Vacuum 膨胀率 80%；目标表可回收空间 100GB） |

**步骤6** 单击“下一步：定时配置”，进行运维任务的“定时配置”。

选择运维类型：

- 单次型任务：需要设置起止时间，智能运维将在设置的时间段内运行运维任务。
- 周期型任务：需要配置时间窗，时间窗配置提供每日、每周、每月三种类型供用户选择。智能运维将自动分析用户注册的时间段，在每个时间段内运行运维任务。

**步骤7** 单击“下一步：配置确认”，确认无误后单击“配置确认”提交。

----结束

## 修改运维计划

- 步骤1 登录DWS管理控制台。
- 步骤2 在集群列表中单击指定集群名称。
- 步骤3 进入“集群详情”页面，切换至“智能运维”页签。
- 步骤4 在运维详情部分切换至运维计划模块，在指定运维任务所在行操作列单击“修改”按钮。
- 步骤5 弹出修改运维任务边栏，修改运维任务功能与添加运维任务功能保持一致，详情请参见[添加运维计划](#)（修改配置下次运行时生效）。
- 步骤6 确认修改后单击“确定”提交。

----结束

## 查看运维任务详情

- 步骤1 登录DWS管理控制台。
- 步骤2 在集群列表中单击指定集群名称。
- 步骤3 进入“集群详情”页面，切换至“智能运维”页签。
- 步骤4 在运维详情部分切换至运维计划模块，在指定运维任务所在行操作列单击“详情”按钮。
- 步骤5 弹出运维任务详情边栏，用户可查看确认。

----结束

### 11.11.3 查看运维任务

- 步骤1 登录DWS管理控制台。
- 步骤2 在集群列表中单击指定集群名称。
- 步骤3 进入“集群详情”页面，切换至“智能运维”页签。
- 步骤4 在运维详情部分切换至运维状态模块。
- 步骤5 单击指定运维任务名称旁箭头，将显示运维任务运行的详细信息。
  - 运维任务：Vacuum
  - 状态：等待中、运行中、已完成、失败。
  - 完成度
  - 时间窗剩余
  - 执行时间窗（本地时间）
  - 正在Vacuum的表数量
  - 准备Vacuum的表数量

- 完成Vacuum的表数量
- 失败Vacuum的表数量
- Vacuum的总表数量
- 此次Vacuum清理空间
- Vacuum详细信息列表

#### 说明

- 其中正在Vacuum的表，准备Vacuum的表，已完成Vacuum的表和Vacuum失败的表最多各显示100张。
- 如果集群只读，智能运维任务则无法执行INSERT语句，可能会遇到任务状态一直为运行中，此时运行中为历史状态，代表在设置时间内未完成。如果手动暂停该任务并且未调度，任务状态可能一直处于等待中，请尽快解除只读并联系技术支持人员更新任务状态。

----结束

# 12 常见问题

---

## 12.1 产品咨询

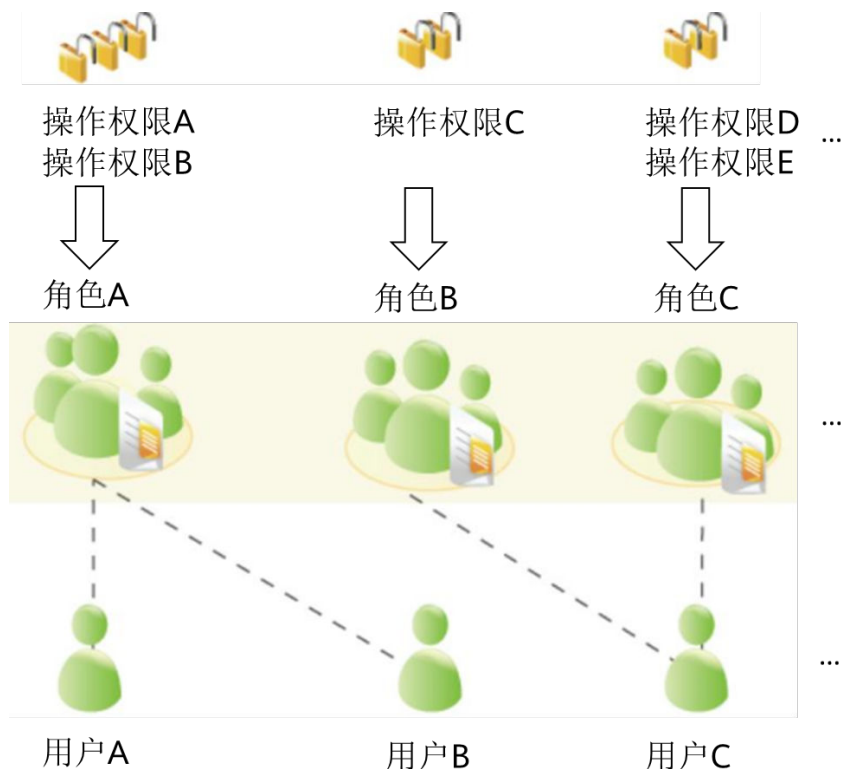
### 12.1.1 DWS 用户和角色是什么关系？

用户和角色在整个集群范围内是共享的，但是其数据并不共享。即用户可以连接任何数据库，但当连接成功后，任何用户都只能访问连接请求里声明的那个数据库。

- 角色（ROLE）本质上是一组权限的集合，通常情况下使用ROLE来组织权限，使用用户进行权限的管理和业务操作。
- 角色之间的权限可以继承，用户组的所有用户可自动继承对应角色的权限。
- 数据库中USER与ROLE的关系为：USER的权限来自于ROLE。
- 用户组包含了具有相同权限的用户集合。
- 用户可以看作是具有登录权限的角色。
- 角色可以看作是没有登录权限的用户。

DWS提供的权限包括“管理面”各组件的操作维护权限，在实际应用时需根据业务场景为各用户分别配置不同权限。为了提升权限管理的易用性，“管理面”引入角色的功能，通过选取指定的权限并统一授予角色，以权限集合的形式实现了权限集中查看和管理。

集中权限管理中权限、角色和用户的关系如下图所示。



DWS提供多种权限，根据业务场景实际需要选择指定的权限授予不同角色，可能是一个或者多个权限对应一个角色。

通过GRANT把角色授予用户后，用户即具有了角色的所有权限。推荐使用角色进行高效权限分配。只对自己的表有所有权限，对其他用户放在属于各自模式下的表无权限。

- 角色A：授予操作权限A和B，用户A和用户B通过分配角色A取得对应的权限。
- 角色B：授予操作权限C，用户C通过分配角色B取得对应的权限。
- 角色C：授予操作权限D和E，用户C通过分配角色C取得对应的权限。

### 12.1.2 如何查看 DWS 数据库用户的创建时间?

### 方式一：

在创建DWS数据库用户时，如果指定了用户的生效时间（VALID BEGIN）与用户创建时间一致，且之后未修改过该用户生效时间的情况下，就可以使用视图PG\_USER查看用户列表，并通过valbegin字段查看该用户的生效时间，即该用户的创建时间。

示例：

创建用户jerry指定生效时间为当前创建时间。

```
CREATE USER jerry PASSWORD 'password' VALID BEGIN '2022-05-19 10:31:56';
```

通过查询视图PG\_USER查看用户列表。valbegin字段显示了jerry的生效时间，即jerry的创建时间。

```
SELECT * FROM PG_USER;
username | usesysid | usecreatedb | usesuper | usecatupd | userepl | passwd | valbegin | valuntil |
respool | parent | spacelimit | useconfig | nodegroup | tempspacelimit |
spillspacelimit
```

```
+-----+-----+-----+-----+-----+-----+
Ruby | 10 | t | | t | t | t | ***** | | | default_pool | 0
dbadmin | 16393 | f | | f | f | f | ***** | | | default_pool | 0
jack | 451897 | f | | f | f | f | ***** | | | default_pool | 0
emma | 451910 | f | | f | f | f | ***** | | | default_pool | 0
jerry | 457386 | f | | f | f | f | ***** | 2022-05-19 10:31:56+08 | | default_pool
0 | | | | | | | | | |
(5 rows)
```

### 方式二：

通过系统表PG\_AUTH\_HISTORY的passwordtime字段查看用户最初的密码创建时间，即该用户的创建时间。需要有系统管理员权限才可以访问此系统表。

```
SELECT rolid, min(passwordtime) as create_time FROM pg_auth_history group by rolid order by rolid;
```

### 示例：

通过查询视图PG\_USER获取用户jerry的OID为457386，查询passwordtime字段获取到用户jerry的创建时间为2022-05-19 10:31:56。

```
SELECT rolid, min(passwordtime) as create_time FROM pg_auth_history group by rolid order by rolid;
rolid | create_time
+-----+-----+
10 | 2022-02-25 09:53:38.711785+08
16393 | 2022-02-25 09:55:17.992932+08
451897 | 2022-05-18 09:42:26.897855+08
451910 | 2022-05-18 09:46:33.152354+08
457386 | 2022-05-19 10:31:56.037706+08
(5 rows)
```

## 12.1.3 如何选择 DWS 区域和可用分区

### 什么是区域、可用分区？

**区域**和**可用分区**用来描述数据中心的位置，您可以在特定的区域、可用分区创建资源。

- **区域（Region）**指物理的数据中心。每个区域完全独立，这样可以实现最大程度的容错能力和稳定性。资源创建成功后不能更换区域。
- **可用分区（AZ，Availability Zone）**是同一区域内，电力和网络互相隔离的物理区域，一个可用分区不受其他可用分区故障的影响。一个区域内可以有多个可用分区，不同可用分区之间物理隔离，但内网互通，既保障了可用分区的独立性，又提供了低价、低时延的网络连接。

### 如何选择区域？

建议就近选择靠近您或者您的目标用户的区域，这样可以减少网络时延，提高访问速度。

### 如何选择可用分区？

是否将资源放在同一可用分区内，主要取决于您对容灾能力和网络时延的要求。

- 如果您的应用需要较高的容灾能力，建议您将资源部署在同一区域的不同可用分区内。
- 如果您的应用要求实例之间的网络延时较低，则建议您将资源创建在同一可用分区内。

## 区域和终端节点

当您通过API使用资源时，您必须指定其区域终端节点。请向企业管理员获取区域和终端节点信息。

### 12.1.4 数据在 DWS 中是否安全？

安全。在大数据时代，数据是用户的核心资产。将继续秉承多年来向社会做出的“上不碰应用，下不碰数据”的承诺，保证用户核心资产的安全。这是对用户和社会的承诺，也是及其伙伴商业成功的保障和基石。

数据仓库服务工程师对整个数据仓库系统进行了电信系统级别的安全增强，大量地采用了多年来在电信行业里积累的各种经验和知识，特别是针对数据安全，用户隐私方面的技术和专利。因此，数据仓库服务是一款符合电信级质量要求的产品，满足各级政府、金融机构、电信运营商对数据安全和用户隐私的要求，并在以上各行业被广泛使用。数据仓库服务还获得了如下安全认证：

- 网络安全实验室ICSL的认证：该认证是遵从英国当局颁布的网络安全标准设立的。
- 隐私和安全管理当局PSA的官方认证：该认证满足欧盟对数据安全和隐私的要求。

## 业务数据安全

数据仓库服务构建在的基础软件设施之上，包括云主机弹性云服务器和对象存储服务OBS。

DWS用户的业务数据是直接存放在集群的云主机当中，集群的云主机对DWS用户本身不可见，只向用户提供数据仓库访问服务，用户以及的运维管理员均无法登录DWS集群云主机进行操作。

DWS集群云主机操作系统进行了严格的安全加固，包括内核安全加固、系统最新补丁、权限控制、端口管理、协议与端口防攻击等。

DWS提供完整的密码策略、身份认证、会话管理、用户权限管理和数据库审计等安全措施。

## 快照数据安全

DWS的备份数据是以快照的形式存储在OBS上。OBS上的数据支持访问权限控制，密钥访问，数据加密。DWS的快照数据仅用于数据的备份和恢复，无法被外界任何用户访问操作，包括DWS用户本身。DWS系统管理员可以通过DWS Console的快照管理和公有云账单看到快照数据在OBS的空间使用情况。

## 网络访问安全

DWS的如下网络安全部署设计使租户之间实现100%的二三层网络隔离，满足政务，金融用户的高等级安全隔离需要。

- DWS部署在租户专属的云主机环境中，不和任何其他租户共享，从物理上隔绝了数据因为计算资源共享而被泄露的可能性。
- DWS集群的虚拟机通过虚拟私有云隔离，避免被其他租户发现和入侵。
- 网络划分为业务平面和管理平面，两个平面采用物理隔离的方式进行部署，保证业务、管理各自网络的安全性。
- 安全组规则保护，租户可以通过自定义安全组的功能，配置安全域的访问规则，提供灵活的网络安全性配置。
- 外部应用软件访问数据仓库服务支持SSL网络安全协议。
- 支持数据从OBS导入的加密传输。

### 12.1.5 可以修改 DWS 集群的安全组吗？

DWS集群创建成功后可以在当前的安全组中添加、删除或修改安全组规则。

**修改已有安全组规则：**

1. 登录DWS控制台。
2. 在左侧导航树，选择“集群 > 集群列表”。
3. 在集群列表中找到所需要的集群，然后单击集群名称。
4. 在“集群详情”页面中，找到“安全组”参数，单击安全组名称进入安全组详情页面，您可以对安全组进行设置。

### 12.1.6 DWS 的脏页是如何产生的？

#### 产生原因

DWS采用多版本控制技术（Multi-Version Concurrency Control，简称MVCC）的并发控制机制保证多个事务访问数据库时的一致性和并发性，其优点是读写互不阻塞，缺点则是会造成磁盘膨胀的问题，而MVCC机制是产生脏页的主要原因。

具体表现为如下场景：

- 当对表执行delete操作时，删除的数据只是在逻辑上被标记为已删除，并未真正从磁盘页面中移除。
- 当对表执行update操作时，DWS将待更新的原数据进行逻辑上的删除标记，同时插入新数据。

对于表中的delete、update操作，被标记为已删除的数据在数据库内部统称为废弃元组，废弃元组在整张表中的占比即为脏页率。因此当表的脏页率高时，则认为表内部被标记为已删除的数据占比高。

#### 处理方案

针对表的脏页率过高的问题，DWS提供了查询脏页率的系统视图，具体使用请参见《数据仓库服务开发指南》中“PGXC\_STAT\_TABLE\_DIRTY”章节。

为了解决脏页率高导致磁盘空间膨胀的问题，DWS提供了VACUUM的功能，可以有效清理delete、update操作后标记的已删除数据，具体请参见《数据仓库服务SQL语法参考》中“VACUUM”章节。

VACUUM不会释放已经分配好的空间，如果要彻底回收已删除的空间，则需要使用VACUUM FULL。

## 说明

- VACUUM FULL可以清理并释放已删除数据的空间，进而提高数据库的性能和效率。但是执行VACUUM FULL命令更加耗费时间和资源，并且可能会导致一些表被锁定，因此VACUUM FULL命令需要在数据库负载较低的情况下使用。
- 为降低磁盘膨胀对数据库性能的影响，建议对查询脏页率超过80%的非系统表执行VACUUM FULL，用户也可根据业务场景自行选择是否执行VACUUM FULL。

# 12.2 数据库连接

## 12.2.1 如何与 DWS 进行通信？

业务应用与DWS进行通信的基本原则是确保业务应用所在网络与DWS的网络能连通，以下是当前主流连接场景，请根据实际情况进行选择。

表 12-1 如何与 DWS 进行通信

| 场景    | 通信方式                   | 支持的连接方式                                                                                              |
|-------|------------------------|------------------------------------------------------------------------------------------------------|
| 云上    | 业务应用与DWS在同一个区域内同一个VPC下 | <ul style="list-style-type: none"> <li>• gsql</li> <li>• Data Studio</li> <li>• JDBC/ODBC</li> </ul> |
|       | 业务应用与DWS在同一个区域内不同VPC下  |                                                                                                      |
|       | 业务应用与DWS在不同区域下         |                                                                                                      |
| 云下和云上 | 业务应用在云下数据中心，需要与DWS进行通信 |                                                                                                      |

## 业务应用与 DWS 在同一个区域内同一个 VPC 下

为保证业务低时延，建议将业务应用和DWS都部署在同一个区域内。例如业务应用部署在ECS，建议将DWS集群部署在跟ECS在同一个虚拟私有云（以下简称VPC）下，应用通过内网IP直接跟DWS进行通信。该场景下，在创建DWS集群时，选择DWS集群跟ECS保持在同一个区域和VPC内。

例如ECS部署在，则DWS选择在下，同时选择DWS跟ECS保持在同一个VPC1下，ECS的内网IP为192.168.120.1，DWS的内网IP为192.168.120.2，即可确保通过内网IP进行通信。

检查通信的基本要点是ECS出方向规则和DWS入方向规则，检查步骤如下：

### 步骤1 检查ECS出方向规则：

应确保ECS的安全组的出方向规则放通，如下。如果没有放通，请参见ECS的。

| 策略 ? | 协议端口 ? | 类型   | 目的地址 ?      |
|------|--------|------|-------------|
| 允许   | 全部     | IPv4 | 0.0.0.0/0 ? |

**步骤2 检查DWS入方向规则：**

DWS创建时如果没有专门设置安全组，系统默认的安全组入规则已放通所有IPv4地址、端口为8000的TCP类型访问。为确保安全，也支持只开放某个对应的IP，详情参见

|    |            |      |             |
|----|------------|------|-------------|
| 允许 | TCP : 8000 | IPv4 | 0.0.0.0/0 ? |
|----|------------|------|-------------|

**步骤3** 登录ECS，能ping通DWS的内网IP，表示网络连通，如果ping不通，请检查以上配置，如果ECS有防火墙，请一并检查防火墙配置。

----结束

使用gsql连接示例：

```
gsql -d gaussdb -h 192.168.120.2 -p 8000 -U dbadmin -W password -r
```

## 业务应用与 DWS 在同一个区域内不同 VPC 下

为保证业务低时延，建议将业务应用和DWS都部署在同一个区域内。例如业务应用部署在ECS，建议将DWS集群部署在跟ECS在同一个虚拟私有云（以下简称VPC）下，如果DWS集群选择了不同的VPC，则ECS与DWS无法直接连通。

例如ECS和DWS都部署在下，但是ECS在VPC1下，DWS在VPC2下，此时需要将VPC1和VPC2建立后，ECS即可通过DWS的内网IP进行访问。

检查通信的基本要点是ECS出方向规则、DWS入方向规则、两个VPC已建立对等连接，检查步骤如下：

**步骤1 检查ECS出方向规则：**

应确保ECS的安全组的出方向规则放通，如下。如果没有放通，请参见ECS的。

| 策略 ? | 协议端口 ? | 类型   | 目的地址 ?      |
|------|--------|------|-------------|
| 允许   | 全部     | IPv4 | 0.0.0.0/0 ? |

**步骤2 检查DWS入方向规则：**

DWS创建时如果没有专门设置安全组，系统默认的安全组入规则已放通所有IPv4地址、端口为8000的TCP类型访问。为确保安全，也支持只开放某个对应的IP，详情参见

|    |            |      |             |
|----|------------|------|-------------|
| 允许 | TCP : 8000 | IPv4 | 0.0.0.0/0 ? |
|----|------------|------|-------------|

**步骤3** 将ECS所在的VPC1和DWS所在的VPC2建立。

**步骤4** 登录ECS，能ping通DWS的内网IP，表示网络连通，如果ping不通，请检查以上配置，如果ECS有防火墙，请一并检查防火墙配置。

----结束

使用gsql连接示例：

```
gsql -d gaussdb -h 192.168.120.2 -p 8000 -U dbadmin -W password -r
```

## 业务应用与 DWS 在不同区域下

如果业务应用与DWS在不同区域下，例如ECS在、DWS在，此时需要将两个区域建立后才能通信。

## 业务应用在云下数据中心，需要与 DWS 进行通信

如果业务应用不在云上，在本地数据中心，此时需要与云上DWS进行通信，分为两种场景：

- **场景一：**云下业务应用通过DWS的公网IP进行通信。

使用gsql连接示例：

```
gsql -d gaussdb -h 公网IP -p 8000 -U dbadmin -W password -r
```

- **场景二：**云下业务无法访问外网，则需要通过进行通信。

## 12.2.2 DWS 是否支持第三方客户端以及 JDBC 和 ODBC 驱动程序？

推荐使用DWS客户端和驱动程序。与开源的PostgreSQL客户端和驱动程序相比，有两个主要的优点：

- **安全强化：**PostgreSQL驱动程序只支持MD5认证，但DWS驱动程序支持SHA256和MD5。
- **数据类型增强：**DWS驱动程序支持新的数据类型smalldatetime和tinyint。

DWS支持开源PostgreSQL客户端和JDBC和ODBC驱动程序。

兼容的客户端和驱动程序版本如下：

- PostgreSQL的psql 9.2.4或更高版本
- PostgreSQL JDBC驱动程序9.3-1103或更高版本
- PSQL ODBC 09.01.0200或更高版本

使用JDBC/ODBC连接DWS，可参见《数据仓库服务开发指南》的。

### 须知

- 建议使用官方推荐的方式连接数据库。参见。
- 其它客户端不能保证100%兼容性，需要客户自行验证。
- 使用其它客户端如果出现不兼容而报错，且不能替换客户端的情况，可尝试替换客户端中的libpq驱动。替换方法：下载并解压gsql客户端压缩包，获取gsql目录下的libpg.so，替换到客户端指定目录中。

## 12.2.3 无法连接 DWS 集群时怎么处理？

### 检查原因

基本原因可能有以下几种：

- 集群状态是否正常。
- 连接命令是否正确，用户名、密码、IP地址或端口无误。
- 安装客户端的操作系统类型、版本是否正确。
- 安装客户端的操作是否正确。

如果是在环境无法连接，还需要检查以下可能导致异常的原因：

- 弹性云服务器是否与集群在相同可用分区、虚拟私有云、子网和安全组。
- 安全组的出入规则是否正确。

如果是在互联网环境无法连接，还需要检查以下可能导致异常的原因：

- 用户网络是否与互联网可以正常连通。
- 用户网络防火墙策略是否限制了访问。
- 用户网络是否需要通过代理才能访问互联网。

## 联系服务人员

如果无法确定原因并解决问题，请提交工单反馈问题。您可以登录管理控制台，在右上方单击填写并提交工单。

### 12.2.4 为什么在互联网环境连接 DWS 后，解绑了 EIP 不会立即返回失败消息？

这是因为解绑了EIP后，会导致网络断开。但是此过程中，TCP协议层因keepalive等的设置，无法及时识别物理连接已经故障，导致gsql，ODBC和JDBC等客户端无法及时识别网络故障。

客户端等待数据库返回的时间与keepalive参数的设置相关，具体可以表示为：  
 $\text{keepalive\_time} + \text{keepalive\_probes} * \text{keepalive\_intvl}$ 。

因为keepalive参数涉及到网络的通信的稳定性，所以可根据具体的业务压力与网络状况进行调整。

如果是Linux环境，使用sysctl命令修改如下参数：

- net.ipv4.tcp\_keepalive\_time
- net.ipv4.tcp\_keeaplive\_probes
- net.ipv4.tcp\_keepalive\_intvl

以修改net.ipv4.tcp\_keepalive\_time参数值为例，执行如下命令将参数值修改为120秒：

***sysctl net.ipv4.tcp\_keepalive\_time=120***

如果是Windows环境，修改注册表“HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters”中的如下配置信息：

- KeepAliveTime
- KeepAliveInterval
- TcpMaxDataRetransmissions（相当于tcp\_keepalive\_probes）

说明

如果以上参数不在注册表“HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters”中，可以在注册表编辑器对应路径下右键单击“新建 > DWORD值”进行添加。

12.2.5 使用公网 IP 连接 DWS 集群时如何设置白名单？

用户可以登录VPC管理控制台手动创建一个安全组，然后回到DWS创建集群页面，单击“安全组”下拉列表旁边的  按钮，刷新后在“安全组”下拉列表中选择新建的安全组。

为了使DWS客户端可以连接集群，用户需要在新建的安全组中添加一条入规则，开放DWS集群的数据库端口的访问权限。

- 协议：TCP。
- 端口范围：8000。指定为创建DWS集群时设置的数据库端口，这个端口是DWS用于接收客户端连接的端口。
- 源地址：选中“IP地址”，然后指定为客户端主机的IP地址，例如“192.168.0.10/32”。

图 12-1 添加入方向规则

添加入方向规则 [教我设置](#)

安全组入方向规则为白名单（允许），放通入方向网络流量。

安全组 dws-

如您要添加多条规则，建议单击[导入规则](#)以进行批量导入。

| 优先级 ? | 策略 | 协议端口 ?      | 类型   | 源地址 ?                   | 描述 | 操作    |
|-------|----|-------------|------|-------------------------|----|-------|
| 1     | 允许 | TCP<br>8000 | IPv4 | IP地址<br>192.168.0.10/32 |    | 复制 删除 |

⊕ 增加1条规则

确定 取消

添加完成后，即设置白名单成功。

12.2.6 使用 API 调用和直连数据库两种方式有哪些差异？

API调用跟直连数据库两种连接方式的优劣可用从安全性、性能维度进行对比：

安全性

表 12-2 安全性对比

| 维度   | API 访问               | 直连数据库      |
|------|----------------------|------------|
| 通信加密 | 天然支持 HTTPS + TLS 加密。 | 需要手动配置SSL。 |

| 维度    | API 访问                   | 直连数据库                               |
|-------|--------------------------|-------------------------------------|
| 身份鉴权  | 支持更灵活的方式（AK/SK或Token认证）。 | 依赖数据库账号密码，权限粒度较粗（如Schema/Table 级别）。 |
| 访问控制  | 可在 API 网关层限制 IP、频率、黑名单等。 | 依赖数据库网络白名单（如 VPC、安全组）。              |
| 防注入攻击 | 参数化接口天然防SQL注入。           | 需要代码层防御SQL注入。                       |
| 审计与日志 | 集中记录API调用日志（请求来源、参数等）。   | 依赖数据库日志（SQL 语句、来源 IP 等）。            |

API 访问在安全性上更优，尤其是对外部系统暴露时，适合作为公共接口；直连数据库需依赖额外安全配置。

性能

表 12-3 性能对比

| 维度    | API 访问                     | 直连数据库               |
|-------|----------------------------|---------------------|
| 网络开销  | 需额外 HTTP 协议头，序列化/反序列化成本较高。 | 使用高效二进制协议（如libpq）。  |
| 延迟    | 通常更高（需经过 API 网关、应用服务器等）。   | 更低延迟（直接与数据库通信）。     |
| 吞吐量   | 受限于API服务器性能。               | 直接利用数据库高性能引擎。       |
| 长连接复用 | 通常短连接，需频繁建连。               | 支持连接池，减少建连开销。       |
| 复杂查询  | 可能无法直接传递复杂 SQL。            | 支持原生SQL，优化器可发挥最大性能。 |

直连数据库在性能上更优，尤其是对复杂查询或高吞吐场景；API 访问适用于轻量级、高频次简单请求。

适用场景

通过以上对比，两种方式适配不同场景：

- 直连数据库：适合大数据量操作，安全性不高，需要提供数据库名称账号、密码等。

- 通过API访问 适合低频访问，封装固定操作，安全性高，需要一个服务层，同时可以实现字段控制。

## 12.3 数据迁移

### 12.3.1 DWS 的 OBS 外表与 GDS 外表支持的数据格式有什么区别？

OBS与GDS外表支持格式文件区别如下：

OBS导入支持ORC、TEXT、JSON、CSV、CARBONDATA、PARQUET文件格式，导出支持ORC、CSV、TEXT、PARQUET文件格式，缺省值为TEXT。

GDS导入导出支持的文件格式：TEXT、CSV和FIXED，缺省值为TEXT。

### 12.3.2 数据如何存储到 DWS？

DWS支持多数据源高效入库，典型的入库方式如下所示。详细指导请参见《数据仓库服务数据库开发指南》中的“导入数据”章节。

- 从OBS导入数据。  
数据上传到OBS对象存储服务中，再从OBS中导入，支持CSV，TEXT格式数据。
- 通过INSERT语句直接插入数据。  
用户可以通过DWS提供的客户端工具（gsq）或者JDBC/ODBC驱动从上层应用向DWS写入数据。DWS支持完整的数据库事务级别的增删改(CRUD)操作。这是最简单的一种方式，这种方式适合数据写入量不太大，并发度不太高的场景。
- 通过COPY FROM STDIN方式导入数据。  
通过COPY FROM STDIN命令写数据到一个表。

### 12.3.3 DWS 可以存储多少业务数据？

数据仓库集群每个节点默认能够支持的存储容量，一个集群支持的节点数范围为3～256，集群总的存储容量随集群规模按比例扩充。

为增强可靠性，每个节点都有一个副本，副本会占用一半的存储空间，选择容量时副本容量会自动翻倍存储。

数据仓库系统会备份数据，生成索引、临时缓存文件、运行日志等内容，并占用存储容量。每个节点实际存储的数据，大致为总存储容量的一半。

### 12.3.4 如何使用 DWS 的\copy 导入导出？

由于云上DWS是全托管服务，用户无法登录后台，无法使用copy进行导入导出文件，所以云上将copy语法禁掉。云上推荐将数据文件放到obs上，使用obs外表进行入库，如果需要使用copy导入导出数据，可以参考如下方法：

1. 将数据文件放到客户端的机器上。
2. 使用gsq连接集群。
3. 执行如下命令导入数据，输入数据文件在客户端的目录信息和文件名，with中指定导入选项，跟正常copy一样，但是需要在copy前添加"\标识，入库成功后不会有消息提示。

```
\copy tb_name from '/directory_name/file_name' with(...);
```

4. 执行如下命令，使用默认参数直接导出数据到本地文件。  
`\copy table_name to '/directory_name/file_name';`
5. 使用copy\_option参数导出为CSV文件。  
`\copy table_name to '/directory_name/file_name' CSV;`
6. 使用with指定option参数，导出为CSV文件，分隔符为'|'。  
`\copy table_name to '/directory_name/file_name' with(format 'csv',delimiter '|');`

### 12.3.5 如何实现 DWS 不同编码库之间数据容错导入

要实现从数据库A（UTF8编码）至数据库B（GBK编码）的数据导入，常规方法导入数据时会出现字符集编码不匹配的错误，导致数据无法导入。

针对小批量数据导入的场景，可以通过\COPY命令来完成，具体方法如下：

**步骤1** 创建数据库A和B，其中数据库A的编码格式为UTF8，数据库B的编码格式为GBK。

```
postgres=> CREATE DATABASE A ENCODING 'UTF8' template = template0;
postgres=> CREATE DATABASE B ENCODING 'GBK' template = template0;
```

**步骤2** 查看数据库列表，可以看到已经创建的数据库A和B。

```
postgres=> \l
 List of databases
 Name | Owner | Encoding | Collate | Ctype | Access privileges
-----+-----+-----+-----+-----+-----
 a | dbadmin | UTF8 | C | C |
 b | dbadmin | GBK | C | C |
 gaussdb | Ruby | SQL_ASCII | C | C |
 postgres | Ruby | SQL_ASCII | C | C |
 template0 | Ruby | SQL_ASCII | C | C | =c/Ruby
 | | | | Ruby=CTc/Ruby
 template1 | Ruby | SQL_ASCII | C | C | =c/Ruby
 | | | | Ruby=CTc/Ruby
 xiaodi | dbadmin | UTF8 | C | C |
(7 rows)
```

**步骤3** 切换到数据库A，输入用户密码。创建表test01，并向表中插入数据。

```
postgres=> \c a
Password for user dbadmin:
SSL connection (protocol: TLSv1.3, cipher: TLS_AES_128_GCM_SHA256, bits: 128)
You are now connected to database "a" as user "dbadmin".

a=> CREATE TABLE test01
(
 c_customer_sk integer,
 c_customer_id char(5),
 c_first_name char(6),
 c_last_name char(8)
)
with (orientation = column,compression=middle)
distribute by hash (c_last_name);
CREATE TABLE
a=> INSERT INTO test01(c_customer_sk, c_customer_id, c_first_name) VALUES (3769, 'hello', 'Grace');
INSERT 0 1
a=> INSERT INTO test01 VALUES (456, 'good');
INSERT 0 1
```

**步骤4** 使用\COPY命令从utf8库中将数据以Unicode编码格式导出成文件test01.dat。

```
\copy test01 to '/opt/test01.dat' with (ENCODING 'Unicode');
```

**步骤5** 切换到数据库B，创建同名表test01。

```
a=> \c b
Password for user dbadmin:
SSL connection (protocol: TLSv1.3, cipher: TLS_AES_128_GCM_SHA256, bits: 128)
You are now connected to database "b" as user "dbadmin".
```

```
b=> CREATE TABLE test01
(
 c_customer_sk integer,
 c_customer_id char(5),
 c_first_name char(6),
 c_last_name char(8)
)
with (orientation = column,compression=middle)
distribute by hash (c_last_name);
```

**步骤6** 使用\COPY命令将文件test01.dat导入到数据库B。

```
\copy test01 from '/opt/test01.dat' with (ENCODING 'Unicode',COMPATIBLE_ILLEGAL_CHARS 'true');
```

#### 说明

- 容错性参数COMPATIBLE\_ILLEGAL\_CHARS指定导入时对非法字符进行容错处理，非法字符转换后入库。不报错，不中断导入。
- 此参数不支持BINARY格式，会报“cannot specify bulkload compatibility options in BINARY mode”错误信息。
- 此参数仅对COPY FROM导入有效。

**步骤7** 在数据库B里查看表test01中的数据；

```
b=> select * from test01;
c_customer_sk | c_customer_id | c_first_name | c_last_name
-----+-----+-----+-----
 3769 | hello | Grace |
 456 | good | |
(2 rows)
```

**步骤8** 通过以上操作就可完成将数据从数据库A（UTF8）至数据库B（GBK）的导入。

----结束

## 12.3.6 DWS 导入性能都和哪些因素有关联？

dws的导入性能受多方面因素影响，主要有以下几点：

1. 集群规格：磁盘io、网络吞吐、内存、cpu规格等。
2. 业务规划：表字段的类型、是否压缩、行存还是列存。
3. 数据存储：集群本地、OBS等。
4. 数据导入的方式选择等。

## 12.4 数据库使用

### 12.4.1 如何调整 DWS 分布列？

在数据仓库类型的数据库中，大表的分布列选择对于数据库和语句查询性能都有至关重要的影响。如果表的分布列选择不当，在数据导入后有可能出现数据分布倾斜，进而导致某些磁盘的使用明显高于其他磁盘，极端情况下会导致集群只读。对于Hash分表策略，存在数据倾斜情况下，查询时出现部分DN的I/O短板，从而影响整体查询性能。合理的选择分布列，并对已经创建的表，进行分布列的调整，对表查询的性能至关重要。

采用Hash分表策略之后需对表的数据进行数据倾斜性检查，以确保数据在各个DN上是均匀分布的。一般来说，不同DN的数据量相差5%以上即可视为倾斜，如果相差10%以上就必须调整分布列。

针对分布不均匀的表，尽可能通过调整分布列，以减少数据倾斜，避免带来潜在的数据库性能问题。

## 选择合适的分布列

Hash分布表的分布列选取至关重要，需要满足以下基本原则：

- 列值应比较离散，以便数据能够均匀分布到各个DN。例如，考虑选择表的主键为分布列，如在人员信息表中选择身份证号码为分布列。
- 在满足第一条原则的情况下尽量不要选取存在常量filter的列。
- 在满足前两条原则的情况，考虑选择查询中的连接条件为分布列，以便Join任务能够下推到DN中执行，且减少DN之间的通信数据量。
- 支持多分布列特性，可以更好地满足数据分布的均匀性要求。

## 如何调整

通过`select version();`语句查询当前数据库版本号，版本号不同，调整的方式不同：

```
test_lhy> select version();
 version

PostgreSQL 9.2.4 (GaussDB 8.1.1 build 7ab61a49) compiled at 2021-06-26 12:05:53 commit 2518 last mr 3356 release
(1 row)
```

- **8.0.x及以前版本，通过重建表时指定分布列来调整：**

**步骤1** 通过Data Studio或者Linux下使用gsqll访问数据库。

**步骤2** 创建新表。

### 📖 说明

以下步骤语句中，table1为原表名，table1\_new为新表名，column1和column2为分布列名称。

```
CREATE TABLE IF NOT EXISTS table1_new
(LIKE table1 INCLUDING ALL EXCLUDING DISTRIBUTION)
DISTRIBUTE BY
HASH (column1, column2);
```

**步骤3** 迁移数据到新表。

```
START TRANSACTION;
LOCK TABLE table1 IN ACCESS EXCLUSIVE MODE;
INSERT INTO table1_new SELECT * FROM table1;
COMMIT;
```

**步骤4** 查看表数据是否迁移成功，删除原表。

```
SELECT COUNT(*) FROM table1_new;
DROP TABLE table1;
```

**步骤5** 替换原表。

```
ALTER TABLE table1_new RENAME TO table1;
```

----结束

- **8.1.0及以后版本后，通过ALTER TABLE语法进行调整，以下为示例。**

**步骤1** 查询当前表定义，返回结果显示该表分布列为c\_last\_name。

```
SELECT pg_get_tabledef('customer_t1');
```

```
gaussdb=> select pg_get_tabledef ('customer_t1');
 pg_get_tabledef

SET search_path = public;
CREATE TABLE customer_t1 (
 c_customer_sk integer,
 c_customer_id character(5),
 c_first_name character(6),
 c_last_name character(8)
)
WITH (orientation=column, compression=middle, colversion=2.0, enable_delta=false)+
DISTRIBUTE BY HASH(c_last_name)
TO GROUP group_version1;
(1 row)
```

**步骤2** 更新分布列中的数据时报错。

```
UPDATE customer_t1 SET c_last_name = 'Jimmy' WHERE c_customer_sk = 6885;
```

```
gaussdb=> update customer_t1 set c_last_name = 'Jimmy' where c_customer_sk = 6885;
ERROR: Distributed key column can't be updated in current version
```

**步骤3** 将该表的分布列修改为不会更新的列，例如c\_customer\_sk。

```
ALTER TABLE customer_t1 DISTRIBUTE BY hash (c_customer_sk);
```

```
gaussdb=> alter table customer_t1 DISTRIBUTE BY hash (c_customer_sk);
ALTER TABLE
```

**步骤4** 重新执行更新旧的分布列的数据。更新成功。

```
UPDATE customer_t1 SET c_last_name = 'Jimmy' WHERE c_customer_sk = 6885;
```

```
gaussdb=> update customer_t1 set c_last_name = 'Jimmy' where c_customer_sk = 6885;
UPDATE 1
```

----结束

## 12.4.2 如何查看和设置 DWS 数据库的字符集编码格式

### 查看数据库字符集编码

使用server\_encoding参数查看当前数据库的字符集编码。例如，查看到数据库music的字符集编码为UTF8。

```
music=> SHOW server_encoding;
server_encoding

UTF8
(1 row)
```

### 设置数据库的字符集编码

#### 说明

DWS不支持修改已创建数据库的字符编码格式。

如果需要指定数据库的字符集编码格式，可按照下面的CREATE DATABASE语法格式，使用template0新建一个数据库。为了适应全球化的需求，使数据库编码能够存储与表示绝大多数的字符，建议创建Database的时候使用UTF8编码。

## CREATE DATABASE 语法格式

```
CREATE DATABASE database_name
[[WITH] { [OWNER [=] user_name] |
 [TEMPLATE [=] template] |
 [ENCODING [=] encoding] |
 [LC_COLLATE [=] lc_collate] |
 [LC_CTYPE [=] lc_ctype] |
 [DBCOMPATIBILITY [=] compatibility_type] |
 [CONNECTION LIMIT [=] connlimit] } [...]];
```

- **TEMPLATE [=] template**

模板名。即从哪个模板创建新数据库。DWS采用从模板数据库复制的方式来创建新的数据库。初始时，DWS包含两个模板数据库template0、template1，以及一个默认的用户数据库。

取值范围：已有数据库的名称。不指定时，系统默认拷贝template1。另外，不支持指定为数据库。

### 须知

目前不支持模板库中含有SEQUENCE对象。如果模板库中有SEQUENCE，则会创建数据库失败。

- **ENCODING [=] encoding**

指定数据库使用的字符编码，可以是字符串（如'SQL\_ASCII'）、整数编号。

不指定时，默认使用模板数据库的编码。模板数据库template0和template1的编码默认与操作系统环境相关。template1不允许修改字符编码，因此若要变更编码，请使用template0创建数据库。

常用取值：GBK、UTF8、Latin1。

### 须知

指定新的数据库字符集编码必须与所选择的本地环境中（LC\_COLLATE和LC\_CTYPE）的设置兼容。

当指定的字符编码集为GBK时，部分中文生僻字无法直接作为对象名。这是因为GBK第二个字节的编码范围在0x40-0x7E之间时，字节编码与ASCII字符@A-Z[\]^\_`a-z{}重叠。其中@[\]^\_`{}是数据库中的操作符，直接作为对象名时，会语法报错。例如“烤”字，GBK16进制编码为0x8240，第二个字节为0x40，与ASCII“@”符号编码相同，因此无法直接作为对象名使用。如果确实要使用，可以在创建和访问对象时，通过增加双引号来规避这个问题。

## 示例

创建一个UTF8编码的数据库music（本地环境的编码格式必须也为UTF8）。

```
CREATE DATABASE music ENCODING 'UTF8' template = template0;
```

## 12.4.3 如何处理 DWS 建表时 date 类型字段自动转换为 timestamp 类型的问题？

创建数据库时，可通过DBCOMPATIBILITY参数指定兼容的数据库的类型，DBCOMPATIBILITY取值范围：ORA、TD、MySQL。分别表示兼容Oracle、Teradata

和MySQL数据库。如果创建数据库时不指定该参数，则默认为ORA，在ORA兼容模式下，date类型会自动转换为timestamp(0)，只有在MySQL兼容模式下才支持date类型。

为解决以上问题，需要将兼容模式修改为MySQL。DWS不支持修改现有数据库的兼容模式，只能在创建数据库时指定兼容模式。DWS从8.1.1集群版本开始支持MySQL兼容模式类型，可参考如下示例进行操作：

```
gaussdb=> CREATE DATABASE mydatabase DBCOMPATIBILITY='mysql';
CREATE DATABASE
gaussdb=> \c mydatabase
Non-SSL connection (SSL connection is recommended when requiring high-security)
You are now connected to database "mydatabase" as user "dbadmin".
mydatabase=> create table t1(c1 int, c2 date);
NOTICE: The 'DISTRIBUTE BY' clause is not specified. Using round-robin as the distribution mode by default.
HINT: Please use 'DISTRIBUTE BY' clause to specify suitable data distribution column.
CREATE TABLE
```

若无法采用重建数据库更改兼容模式的方法，可通过修改字段类型来规避date类型会自动转换为timestamp类型。例如，将日期以string类型的形式插入表中，可参考如下示例操作：

```
gaussdb=> CREATE TABLE mytable (a date,b int);
CREATE TABLE
gaussdb=> INSERT INTO mytable VALUES(date '12-08-2023',01);
INSERT 0 1
gaussdb=> SELECT * FROM mytable;
 a | b
-----+---
2023-12-08 00:00:00 | 1
(1 row)
gaussdb=> ALTER TABLE mytable MODIFY a VARCHAR(20);
ALTER TABLE
gaussdb=> INSERT INTO mytable VALUES('2023-12-10',02);
INSERT 0 1
gaussdb=> SELECT * FROM mytable;
 a | b
-----+---
2023-12-08 00:00:00 | 1
2023-12-10 | 2
(2 rows)
```

## 12.4.4 DWS 是否需要定时对常用的表做 VACUUM FULL 和 ANALYZE 操作？

需要。

对于频繁增、删、改的表，需要定期执行VACUUM FULL和ANALYZE，该操作可回收已更新或已删除的数据所占据的磁盘空间，防止因数据膨胀和统计信息不准造成性能下降。

- 一般情况下，对表执行完大量增、改操作后，建议进行ANALYZE。
- 对表执行过删除操作后，建议进行VACUUM，一般不建议日常使用VACUUM FULL选项，但是可以在特殊情况下使用。例如，用户删除了一个表的大部分行之后，希望从物理上缩小该表以减少磁盘空间占用。VACUUM和VACUUM FULL具体的差异可以参考[VACUUM和VACUUM FULL](#)。

### 语法规则

指定某张表进行分析。

```
ANALYZE table_name;
```

对数据库中的所有表（非外部表）进行分析。

```
ANALYZE;
```

指定某张表进行VACUUM。

```
VACUUM table_name;
```

指定某张表进行VACUUM FULL。

```
VACUUM FULL table_name;
```

 说明

- 如果执行VACUUM FULL命令后所占用物理空间无变化（未减少），请确认是否有其他活跃事务（删除数据事务开始之前开始的事务，并在VACUUM FULL执行前未结束）存在，如果有，需等其他活跃事务退出后进行重试。
- 8.1.3及以上版本中Vacuum/Vacuum Full可在管理控制台操作，详情可参见《数据仓库服务用户指南》中“Vacuum回收DWS空间”章节。

VACUUM 和 VACUUM FULL

在DWS中，VACUUM的本质就是一个“吸尘器”，用于吸收“尘埃”。而尘埃其实就是旧数据，如果这些数据没有及时清理，那么将会导致数据库空间膨胀，性能下降，更严重的情况会导致异常退出。

VACUUM的作用：

- 空间膨胀问题：清除废旧元组以及相应的索引。包括提交的事务delete的元组（以及索引）、update的旧版本（以及索引），回滚的事务insert的元组（以及索引）、update的新版本（以及索引）、copy导入的元组（以及索引）。
- FREEZE：防止因事务ID回卷问题（Transaction ID wraparound）而导致的异常退出，将小于OldestXmin的事务号转化为freeze xid，更新表的relfrozenxid，更新库的relfrozenxid、truncate clog。
- 更新统计信息：VACUUM ANALYZE时，会更新统计信息，使得优化器能够选择更好的方案执行SQL语句。

VACUUM命令存在两种形式，VACUUM和VACUUM FULL，目前VACUUM对行存表有作用，对列存表无显著的作用，列存表只能依靠VACUUM FULL释放空间。具体区别见下表：

表 12-4 VACUUM 和 VACUUM FULL

| 差异项  | VACUUM                                                                                        | VACUUM FULL                                                 |
|------|-----------------------------------------------------------------------------------------------|-------------------------------------------------------------|
| 空间清理 | 如果删除的记录位于表的末端，其所占用的空间将会被物理释放并归还操作系统。而如果不是末端数据，会将表中或索引中 dead tuple（死亡元组）所占用的空间置为可用状态，从而复用这些空间。 | 不论被清理的数据处于何处，这些数据所占用的空间都将被物理释放并归还于操作系统。当再有数据插入后，分配新的磁盘页面使用。 |
| 锁类型  | 共享锁，可以与其他操作并行。                                                                                | 排他锁，执行期间基于该表的操作全部挂起。                                        |
| 物理空间 | 不会释放。                                                                                         | 会释放。                                                        |

| 差异项  | VACUUM             | VACUUM FULL                                       |
|------|--------------------|---------------------------------------------------|
| 事务ID | 不回收。               | 回收。                                               |
| 执行开销 | 开销较小，可以定期执行。       | 开销很大，建议确认数据库所占磁盘页面空间接近临界值再执行操作，且最好选择数据量操作较少的时段完成。 |
| 执行效果 | 执行后基于该表的操作效率有一定提升。 | 执行完后，基于该表的操作效率大大提升。                               |

## 12.4.5 如何导出 DWS 某张表结构？

### 使用 SQL 编辑器进行数据导出

建议使用SQL编辑器进行表数据导出，登录数据源，编辑框面板上方选择对应的数据库和模式，输入查询表数据的SQL语句，单击“导出”按钮。支持以下几种导出方式：

- **本地导出**：将查询SQL的所有结果导出到xlsx或csv文件，可直接在本地打开查看，最多支持导出20000条数据。
- **全量导出**：将查询SQL的所有结果导出到指定的OBS桶下的路径中，默认为csv文件。

## 12.4.6 DWS 是否有高效的删除表数据的方法？

有。删除大批量的日志数据时，使用delete语法需要花费更大的时间，此时可以通过TRUNCATE语法进行大批量删除操作，它的删除速度比delete快得多。

### 功能描述

清理表数据，TRUNCATE在清理表数据时，可以快速地从表中删除所有行。它和在目标表上进行无条件的DELETE有同样的效果，由于TRUNCATE不做表扫描，在大表上操作效果更明显，效率会提升很多。

### 功能特点

- TRUNCATE TABLE在功能上与不带WHERE子句DELETE语句相同：二者均删除表中的全部行。
- TRUNCATE TABLE比DELETE速度快且使用系统和事务日志资源少：
  - DELETE语句每次删除一行，并在事务日志中为所删除每行记录一项。
  - TRUNCATE TABLE通过释放存储表数据所用数据页来删除数据，并且只在事务日志中记录页的释放。
- TRUNCATE，DELETE，DROP三者的差异如下：
  - TRUNCATE TABLE，删除内容，释放空间，但不删除定义。
  - DELETE TABLE，删除内容，不删除定义，不释放空间。
  - DROP TABLE，删除内容和定义，释放空间。



## 12.4.8 如果 DWS 建表时没有指定分布列，数据会怎么存储？

### 说明

8.1.2及以上集群版本，可通过GUC参数default\_distribution\_mode来查询和设置表的默认分布方式。

如果建表时没有指定分布列，数据会以下几种场景来存储：

- 场景一

若建表时包含主键/唯一约束，则选取HASH分布，分布列为主键/唯一约束对应的列。

```
CREATE TABLE warehouse1
(
 W_WAREHOUSE_SK INTEGER PRIMARY KEY,
 W_WAREHOUSE_ID CHAR(16) NOT NULL,
 W_WAREHOUSE_NAME VARCHAR(20)
);
NOTICE: CREATE TABLE / PRIMARY KEY will create implicit index "warehouse1_pkey" for table "warehouse1"
CREATE TABLE

SELECT getdistributekey('warehouse1');
getdistributekey

w_warehouse_sk
(1 row)
```

- 场景二

若建表时不包含主键/唯一约束，但存在数据类型支持作分布列的列，则选取HASH分布，分布列为第一个数据类型支持作分布列的列。

```
CREATE TABLE warehouse2
(
 W_WAREHOUSE_SK INTEGER ,
 W_WAREHOUSE_ID CHAR(16) NOT NULL,
 W_WAREHOUSE_NAME VARCHAR(20)
);
NOTICE: The 'DISTRIBUTE BY' clause is not specified. Using 'w_warehouse_sk' as the distribution column by default.
HINT: Please use 'DISTRIBUTE BY' clause to specify suitable data distribution column.
CREATE TABLE

SELECT getdistributekey('warehouse2');
getdistributekey

w_warehouse_sk
(1 row)
```

- 场景三

若建表时不包含主键/唯一约束，也不存在数据类型支持作分布列的列，选取ROUNDROBIN分布。

```
CREATE TABLE warehouse3
(
 W_WAREHOUSE_ID CHAR(16) NOT NULL,
 W_WAREHOUSE_NAME VARCHAR(20)
);
NOTICE: The 'DISTRIBUTE BY' clause is not specified. Using 'w_warehouse_id' as the distribution column by default.
HINT: Please use 'DISTRIBUTE BY' clause to specify suitable data distribution column.
CREATE TABLE

SELECT getdistributekey('warehouse3');
getdistributekey

```

```
w_warehouse_id
(1 row)
```

### 12.4.9 如何将 DWS 联结查询的 null 结果替换成 0？

在执行outer join（left join、right join、full join）联结查询时，outer join在匹配失败的情况下结果集会补空，产生大量NULL值，可以在联结查询时将这部分null值替换为0。

可使用coalesce函数，它的作用是返回参数列表中第一个非NULL的参数值。例如：

```
SELECT coalesce(NULL,'hello');
coalesce

hello
(1 row)
```

有表course1和表course2，使用left join对两表进行联结查询：

```
SELECT * FROM course1;
stu_id | stu_name | cour_name
-----+-----+-----
20110103 | ALLEN | Math
20110102 | JACK | Programming Design
20110101 | MAX | Science
(3 rows)

SELECT * FROM course2;
cour_id | cour_name | teacher_name
-----+-----+-----
1002 | Programming Design | Mark
1001 | Science | Anne
(2 rows)

SELECT course1.stu_name,course2.cour_id,course2.cour_name,course2.teacher_name FROM course1 LEFT
JOIN course2 ON course1.cour_name = course2.cour_name ORDER BY 1;
stu_name | cour_id | cour_name | teacher_name
-----+-----+-----+-----
ALLEN | | |
JACK | 1002 | Programming Design | Mark
MAX | 1001 | Science | Anne
(3 rows)
```

使用coalesce函数将查询结果中的空值替换为0或其他非0值：

```
SELECT course1.stu_name,
coalesce(course2.cour_id,0) AS cour_id,
coalesce(course2.cour_name,'NA') AS cour_name,
coalesce(course2.teacher_name,'NA') AS teacher_name
FROM course1
LEFT JOIN course2 ON course1.cour_name = course2.cour_name
ORDER BY 1;
stu_name | cour_id | cour_name | teacher_name
-----+-----+-----+-----
ALLEN | 0 | NA | NA
JACK | 1002 | Programming Design | Mark
MAX | 1001 | Science | Anne
(3 rows)
```

### 12.4.10 如何查看 DWS 表是行存还是列存？

表的存储方式由建表语句中的ORIENTATION参数控制，row表示行存，column表示列存。

查看已创建的表是行存还是列存，可通过表定义函数PG\_GET\_TABLEDEF查询。

如下orientation=column表示为列存表。

目前暂不支持通过ALTER TABLE语句修改ORIENTATION参数，即行存表和列存表无法直接进行转换。

```
SELECT * FROM PG_GET_TABLEDEF('customer_t1');
 pg_get_tabledef

SET search_path = tpchobs; +
CREATE TABLE customer_t1 (+
 c_customer_sk integer, +
 c_customer_id character(5), +
 c_first_name character(6), +
 c_last_name character(8) +
) +
WITH (orientation=column, compression=middle, colversion=2.0, enable_delta=false)+
DISTRIBUTE BY HASH(c_last_name) +
TO GROUP group_version1;
(1 row)
```

## 12.4.11 DWS 列存表的常用信息查询

使用列存表时，一些常用信息查询SQL示例：

先创建列存分区表my\_table，并向表中插入数据。

```
CREATE TABLE my_table
(
 product_id INT,
 product_name VARCHAR2(40),
 product_quantity INT
)
WITH (ORIENTATION = COLUMN)
PARTITION BY range(product_quantity)
(
 partition my_table_p1 values less than(600),
 partition my_table_p2 values less than(800),
 partition my_table_p3 values less than(950),
 partition my_table_p4 values less than(1000));

INSERT INTO my_table VALUES(1011, 'tents', 720);
INSERT INTO my_table VALUES(1012, 'hammock', 890);
INSERT INTO my_table VALUES(1013, 'compass', 210);
INSERT INTO my_table VALUES(1014, 'telescope', 490);
INSERT INTO my_table VALUES(1015, 'flashlight', 990);
INSERT INTO my_table VALUES(1016, 'ropes', 890);
```

查看已创建的列存分区表：

```
SELECT * FROM my_table;
 product_id | product_name | product_quantity
-----+-----+-----
1013 | compass | 210
1014 | telescope | 490
1011 | tents | 720
1015 | flashlight | 990
1012 | hammock | 890
1016 | ropes | 890
(6 rows)
```

## 查询分区边界

```
SELECT relname, partstrategy, boundaries FROM pg_partition where parentid=(select parentid from
pg_partition where relname='my_table');
 relname | partstrategy | boundaries
-----+-----+-----
my_table | r |
my_table_p1 | r | {600}
my_table_p2 | r | {800}
my_table_p3 | r | {950}
```

```
my_table_p4 | r | {1000}
(5 rows)
```

## 查询列存表列数

```
SELECT count(*) FROM ALL_TAB_COLUMNS where table_name='my_table';
count

3
(1 row)
```

## 查询数据在各 DN 分布

```
SELECT table_skewness('my_table');
table_skewness

("dn_6007_6008 ",3,50.000%)
("dn_6009_6010 ",2,33.333%)
("dn_6003_6004 ",1,16.667%)
("dn_6001_6002 ",0,0.000%)
("dn_6005_6006 ",0,0.000%)
("dn_6011_6012 ",0,0.000%)
(6 rows)
```

## 查询某一有数据分布 DN 上分区 P1 所对应的 cudesc 和 delta 表名称

```
EXECUTE DIRECT ON (dn_6003_6004) 'select a.relname from pg_class a, pg_partition b where
(a.oid=b.reldeltarelid or a.oid=b.relcudescrid) and b.relname="my_table_p1";
relname

pg_delta_part_60317
pg_cudesc_part_60317
(2 rows)
```

## 12.4.12 DWS 查询时索引失效场景解析

对表建立索引可提高数据库查询性能，但有时会出现建立了索引，但查询计划中却发现索引没有被使用的情况。针对这种情况，本文将列举几种常见的场景和优化方法。

### 场景一：返回结果集很大

以行存表的Seq Scan和Index Scan为例：

- Seq Scan：按照表的记录的排列顺序从头到尾依次检索扫描，每次扫描要取到所有的记录。这也是最简单最基础的扫表方式，扫描的代价比较大。
- Index Scan：对于指定的查询，先扫描一遍索引，从索引中找到符合要求的记录的位置（指针），再定位到表中具体的Page去获取，即先走索引，再读表数据。

因此，根据两种扫描方式的特点可知，多数情况下，Index Scan要比Seq Scan快。但是如果获取的结果集占有所有数据的比重很大时（超过70%），这时Index Scan因为要先扫描索引再读表数据反而不如直接全表扫描的速度快。

### 场景二：未及时 ANALYZE

ANALYZE更新表的统计信息，如果表未执行ANALYZE或最近一次执行完ANALYZE后表进行过数据量较大的增删操作，会导致统计信息不准，该场景下也可能导致查询表时没有使用索引。

优化方法：对表执行ANALYZE更新统计信息。

### 场景三：过滤条件使用了函数或隐式类型转化

如果在过滤条件中使用了计算、函数、隐式类型转化，都可能导致无法选择索引。

例如创建表，并在a，b，c三列上都分别创建了索引。

```
CREATE TABLE test(a int, b text, c date);
```

- 在索引字段进行计算操作。

从下面的执行结果可以看出：where a = 101，where a = 102 - 1都使用了a列上的索引，但是where a + 1 = 102没有使用索引。

```
explain verbose select * from test where a = 101;
QUERY PLAN

id | operation | E-rows | E-distinct | E-memory | E-width | E-costs
---+-----+-----+-----+-----+-----+-----
1 | -> Streaming (type: GATHER) | 1 | | | 44 | 16.27
2 | -> Index Scan using index_a on public.test | 1 | | 1MB | 44 | 8.27

Predicate Information (identified by plan id)

2 --Index Scan using index_a on public.test
Index Cond: (test.a = 101)

Targetlist Information (identified by plan id)

1 --Streaming (type: GATHER)
Output: a, b, c
Node/s: dn_6005_6006
2 --Index Scan using index_a on public.test
Output: a, b, c
Distribute Key: a

===== Query Summary =====
System available mem: 3358720KB
Query Max mem: 3358720KB
Query estimated mem: 1024KB
(24 rows)

explain verbose select * from test where a = 102 - 1;
QUERY PLAN

id | operation | E-rows | E-distinct | E-memory | E-width | E-costs
---+-----+-----+-----+-----+-----+-----
1 | -> Streaming (type: GATHER) | 1 | | | 44 | 16.27
2 | -> Index Scan using index_a on public.test | 1 | | 1MB | 44 | 8.27

Predicate Information (identified by plan id)

2 --Index Scan using index_a on public.test
Index Cond: (test.a = 101)

Targetlist Information (identified by plan id)

1 --Streaming (type: GATHER)
Output: a, b, c
Node/s: dn_6005_6006
2 --Index Scan using index_a on public.test
Output: a, b, c
Distribute Key: a

===== Query Summary =====
System available mem: 3358720KB
Query Max mem: 3358720KB
Query estimated mem: 1024KB
(24 rows)
```

```
explain verbose select * from test where a + 1 = 102;
QUERY PLAN

id | operation | E-rows | E-distinct | E-memory | E-width | E-costs
+-----+-----+-----+-----+-----+-----+-----
1 | -> Streaming (type: GATHER) | 1 | | | 44 | 22.21
2 | -> Seq Scan on public.test | 1 | | 1MB | 44 | 14.21

Predicate Information (identified by plan id)

2 --Seq Scan on public.test
Filter: ((test.a + 1) = 102)

Targetlist Information (identified by plan id)

1 --Streaming (type: GATHER)
Output: a, b, c
Node/s: All datanodes
2 --Seq Scan on public.test
Output: a, b, c
Distribute Key: a

===== Query Summary =====

System available mem: 3358720KB
Query Max mem: 3358720KB
Query estimated mem: 1024KB
(24 rows)
```

优化方式：尽量使用常量代替表达式，或者常量计算尽量写在等号的右侧。

- 在索引字段上使用函数。

从下面的执行结果可以看出：在索引列上使用函数也会导致无法选择索引。

```
explain verbose select * from test where to_char(c, 'yyyyMMdd') =
to_char(CURRENT_DATE,'yyyyMMdd');
QUERY PLAN

id | operation | E-rows | E-distinct | E-memory | E-width | E-costs
+-----+-----+-----+-----+-----+-----+-----
1 | -> Streaming (type: GATHER) | 1 | | | 44 | 22.28
2 | -> Seq Scan on public.test | 1 | | 1MB | 44 | 14.28

Predicate Information (identified by plan id)

2 --Seq Scan on public.test
Filter: (to_char(test.c, 'yyyyMMdd')::text) = to_char(('2022-11-30'::pg_catalog.date)::timestamp
with time zone, 'yyyyMMdd')::text)

Targetlist Information (identified by plan id)

1 --Streaming (type: GATHER)
Output: a, b, c
Node/s: All datanodes
2 --Seq Scan on public.test
Output: a, b, c
Distribute Key: a

===== Query Summary =====

System available mem: 3358720KB
Query Max mem: 3358720KB
Query estimated mem: 1024KB
(24 rows)
explain verbose select * from test where c = current_date;
QUERY PLAN

```

```
id | operation | E-rows | E-distinct | E-memory | E-width | E-costs
-----+-----+-----+-----+-----+-----+-----
1 | -> Streaming (type: GATHER) | 1 | | | 44 | 16.27
2 | -> Index Scan using index_c on public.test | 1 | | 1MB | 44 | 8.27

Predicate Information (identified by plan id)

2 --Index Scan using index_c on public.test
 Index Cond: (test.c = '2022-11-30'::pg_catalog.date)

Targetlist Information (identified by plan id)

1 --Streaming (type: GATHER)
 Output: a, b, c
 Node/s: All datanodes
2 --Index Scan using index_c on public.test
 Output: a, b, c
 Distribute Key: a

===== Query Summary =====

System available mem: 3358720KB
Query Max mem: 3358720KB
Query estimated mem: 1024KB
(24 rows)
```

优化方法：尽量减少索引列上没有必要的函数调用。

- 数据类型隐式转化。

此类场景较常见，例如字段b的类型是text，过滤条件是where b = 2，在生成计划时，text类型会隐式转化为bigint类型，实际的过滤条件变成where b::bigint = 2，导致b列上的索引失效。

```
explain verbose select * from test where b = 2;
```

QUERY PLAN

```
id | operation | E-rows | E-distinct | E-memory | E-width | E-costs
-----+-----+-----+-----+-----+-----+-----
1 | -> Streaming (type: GATHER) | 1 | | | 44 | 22.21
2 | -> Seq Scan on public.test | 1 | | 1MB | 44 | 14.21
```

Predicate Information (identified by plan id)

```
2 --Seq Scan on public.test
 Filter: ((test.b)::bigint = 2)
```

Targetlist Information (identified by plan id)

```
1 --Streaming (type: GATHER)
 Output: a, b, c
 Node/s: All datanodes
2 --Seq Scan on public.test
 Output: a, b, c
 Distribute Key: a
```

===== Query Summary =====

```

System available mem: 3358720KB
Query Max mem: 3358720KB
Query estimated mem: 1024KB
(24 rows)
```

```
explain verbose select * from test where b = '2';
```

QUERY PLAN

```
id | operation | E-rows | E-distinct | E-memory | E-width | E-costs
-----+-----+-----+-----+-----+-----+-----
1 | -> Streaming (type: GATHER) | 1 | | | 44 | 16.27
2 | -> Index Scan using index_b on public.test | 1 | | 1MB | 44 | 8.27
```

Predicate Information (identified by plan id)

```

2 --Index Scan using index_b on public.test
 Index Cond: (test.b = '2'::text)

Targetlist Information (identified by plan id)

1 --Streaming (type: GATHER)
 Output: a, b, c
 Node/s: All datanodes
2 --Index Scan using index_b on public.test
 Output: a, b, c
 Distribute Key: a

===== Query Summary =====

System available mem: 3358720KB
Query Max mem: 3358720KB
Query estimated mem: 1024KB
(24 rows)
```

优化方法：索引条件上的常量尽可能使用和索引列相同类型的常量，避免发生隐式类型转化。

## 场景四：用 nestloop + indexscan 代替 hashjoin

此类语句的特征是两个表关联的时候，其中一个表上where条件过滤之后的结果集行数很小，同时，最终满足条件的结果集行数也很小。此时，使用nestloop+indexscan的效果往往要优于hashjoin。较优的执行计划如下：

可以看到第5层的Index Cond: (t1.b = t2.b)已经把join条件下推到了基表扫描上。

```
explain verbose select t1.a,t1.b from t1,t2 where t1.b=t2.b and t2.a=4;
id | operation | E-rows | E-distinct | E-memory | E-width | E-costs
-----+-----+-----+-----+-----+-----+-----
1 | -> Streaming (type: GATHER) | 26 | | | 8 | 17.97
2 | -> Nested Loop (3,5) | 26 | 1MB | 8 | 11.97
3 | -> Streaming(type: BROADCAST) | 2 | 2MB | 4 | 2.78
4 | -> Seq Scan on public.t2 | 1 | 1MB | 4 | 2.62
5 | -> Index Scan using t1_b_idx on public.t1 | 26 | 1MB | 8 | 9.05
(5 rows)

Predicate Information (identified by plan id)

4 --Seq Scan on public.t2
 Filter: (t2.a = 4)
5 --Index Scan using t1_b_idx on public.t1
 Index Cond: (t1.b = t2.b)
(4 rows)

Targetlist Information (identified by plan id)

1 --Streaming (type: GATHER)
 Output: t1.a, t1.b
 Node/s: All datanodes
2 --Nested Loop (3,5)
 Output: t1.a, t1.b
3 --Streaming(type: BROADCAST)
 Output: t2.b
 Spawn on: datanode2
 Consumer Nodes: All datanodes
4 --Seq Scan on public.t2
 Output: t2.b
 Distribute Key: t2.a
5 --Index Scan using t1_b_idx on public.t1
 Output: t1.a, t1.b
 Distribute Key: t1.a
(15 rows)
```

```
===== Query Summary =====
```

```

System available mem: 9262694KB
Query Max mem: 9471590KB
Query estimated mem: 5144KB
(3 rows)
```

如果优化器没有选择这种执行计划，可以通过以下方式优化：

```
set enable_index_nestloop = on;
set enable_hashjoin = off;
set enable_seqscan = off;
```

## 场景五：使用 hint 指定索引时指定的索引方式不对

DWS的plan hint当前支持指定的Scan方式有三种：tablescan、indexscan和indexonlyscan。

- tablescan：全表扫描，比如行存表的Seq Scan，列存表的CStore Scan。
- indexscan：先扫索引，再根据索引取表记录。
- indexonlyscan：覆盖索引扫描，所需的返回结果能被所扫描的索引全部覆盖。与index scan相比，index only scan所包含的字段集合，涵盖了查询语句中的字段，这样，提取出相应的index就不必再根据索引取表记录了。

因此，对于需要indexonlyscan的场景，如果hint指定了indexscan，该hint是无法生效的：

```
explain verbose select/*+ indexscan(test)*/ b from test where b = '1';
WARNING: unused hint: IndexScan(test)
```

QUERY PLAN

| id | operation                                       | E-rows | E-distinct | E-memory | E-width | E-costs |
|----|-------------------------------------------------|--------|------------|----------|---------|---------|
| 1  | -> Streaming (type: GATHER)                     | 1      |            |          | 32      | 16.27   |
| 2  | -> Index Only Scan using index_b on public.test | 1      |            | 1MB      | 32      | 8.27    |

Predicate Information (identified by plan id)

2 --Index Only Scan using index\_b on public.test  
Index Cond: (test.b = '1'::text)

Targetlist Information (identified by plan id)

1 --Streaming (type: GATHER)  
Output: b  
Node/s: All datanodes  
2 --Index Only Scan using index\_b on public.test  
Output: b  
Distribute Key: a

```
===== Query Summary =====
```

```

System available mem: 3358720KB
Query Max mem: 3358720KB
Query estimated mem: 1024KB
(24 rows)
```

```
explain verbose select/*+ indexonlyscan(test)*/ b from test where b = '1';
```

QUERY PLAN

| id | operation                                       | E-rows | E-distinct | E-memory | E-width | E-costs |
|----|-------------------------------------------------|--------|------------|----------|---------|---------|
| 1  | -> Streaming (type: GATHER)                     | 1      |            |          | 32      | 16.27   |
| 2  | -> Index Only Scan using index_b on public.test | 1      |            | 1MB      | 32      | 8.27    |

Predicate Information (identified by plan id)

```
2 --Index Only Scan using index_b on public.test
 Index Cond: (test.b = '1'::text)

Targetlist Information (identified by plan id)

1 --Streaming (type: GATHER)
 Output: b
 Node/s: All datanodes
2 --Index Only Scan using index_b on public.test
 Output: b
 Distribute Key: a

===== Query Summary =====

System available mem: 3358720KB
Query Max mem: 3358720KB
Query estimated mem: 1024KB
(24 rows)
```

优化方法：使用hint时正确指定indexscan和indexonlyscan。

## 场景六：全文检索 GIN 索引

为了加速文本搜索，进行全文检索时可以创建GIN索引：

```
CREATE INDEX idxb ON test using gin(to_tsvector('english',b));
```

创建GIN索引时，必须使用to\_tsvector的两参数版本，并且只有当查询时也使用了两参数版本，且参数值与索引中相同时，才会使用该索引：

### 说明

to\_tsvector()函数有两个版本，只输入一个参数的版本和输入两个参数的版本。输入一个参数时，系统默认采用default\_text\_search\_config所指定的分词器。创建索引时必须使用to\_tsvector的两参数版本，否则索引内容可能不一致。

```
explain verbose select * from test where to_tsvector(b) @@ to_tsquery('cat') order by 1;
QUERY PLAN
```

| id | operation                   | E-rows | E-distinct | E-memory | E-width | E-costs |
|----|-----------------------------|--------|------------|----------|---------|---------|
| 1  | -> Streaming (type: GATHER) | 2      |            |          | 44      | 22.23   |
| 2  | -> Sort                     | 2      | 16MB       |          | 44      | 14.23   |
| 3  | -> Seq Scan on public.test  | 1      | 1MB        |          | 44      | 14.21   |

Predicate Information (identified by plan id)

```
3 --Seq Scan on public.test
 Filter: (to_tsvector(test.b) @@ "'cat' '::tsquery)
```

Targetlist Information (identified by plan id)

```
1 --Streaming (type: GATHER)
 Output: a, b, c
 Merge Sort Key: test.a
 Node/s: All datanodes
2 --Sort
 Output: a, b, c
 Sort Key: test.a
3 --Seq Scan on public.test
 Output: a, b, c
 Distribute Key: a
```

===== Query Summary =====

```

System available mem: 3358720KB
Query Max mem: 3358720KB
```

```
Query estimated mem: 1024KB
(29 rows)
explain verbose select * from test where to_tsvector('english',b) @@ to_tsquery('cat') order by 1;
QUERY PLAN

id | operation | E-rows | E-distinct | E-memory | E-width | E-costs
---+-----+-----+-----+-----+-----+-----
1 | -> Streaming (type: GATHER) | 2 | 2 | | 44 | 20.03
2 | -> Sort | 2 | | 16MB | 44 | 12.03
3 | -> Bitmap Heap Scan on public.test | 1 | | 1MB | 44 | 12.02
4 | -> Bitmap Index Scan | 1 | | 1MB | 0 | 8.00

Predicate Information (identified by plan id)

3 --Bitmap Heap Scan on public.test
Recheck Cond: (to_tsvector('english':regconfig, test.b) @@ 'cat':tsquery)
4 --Bitmap Index Scan
Index Cond: (to_tsvector('english':regconfig, test.b) @@ 'cat':tsquery)

Targetlist Information (identified by plan id)

1 --Streaming (type: GATHER)
Output: a, b, c
Merge Sort Key: test.a
Node/s: All datanodes
2 --Sort
Output: a, b, c
Sort Key: test.a
3 --Bitmap Heap Scan on public.test
Output: a, b, c
Distribute Key: a

===== Query Summary =====

System available mem: 3358720KB
Query Max mem: 3358720KB
Query estimated mem: 2048KB
(32 rows)
```

优化方式：查询时使用to\_tsvector的两参数版本，且保证参数值与索引中相同。

### 12.4.13 如何使用 DWS 自定义函数改写 CRC32()函数

DWS目前未内置CRC32函数，但如果需要实现MySQL中的CRC32()函数功能，用户可使用DWS的自定义函数语句对其进行改写。

- 函数：CRC32(expr)
- 描述：用于计算循环冗余值。入参expr为字符串。如果参数为NULL，则返回NULL；否则，在计算冗余后返回32位无符号值。

DWS的自定义函数语句改写CRC32函数示例：

```
CREATE OR REPLACE FUNCTION crc32(text_string text) RETURNS bigint AS $$
DECLARE
 val bigint;
 i int;
 j int;
 byte_length int;
 binary_string bytea;
BEGIN
 IF text_string is null THEN
 RETURN null;
 ELSIF text_string = '' THEN
 RETURN 0;
 END IF;
 i = 0;
```

```
val = 4294967295;
byte_length = bit_length(text_string) / 8;
binary_string = decode(replace(text_string, E'\\', E'\\\\'), 'escape');
LOOP
 val = (val # get_byte(binary_string, i))::bigint;
 i = i + 1;
 j = 0;
 LOOP
 val = ((val >> 1) # (3988292384 * (val & 1)))::bigint;
 j = j + 1;
 IF j >= 8 THEN
 EXIT;
 END IF;
 END LOOP;
 IF i >= byte_length THEN
 EXIT;
 END IF;
 END LOOP;
 RETURN (val # 4294967295);
END
$$ IMMUTABLE LANGUAGE plpgsql;
```

验证改写后的结果：

```
select crc32(null),crc32(''),crc32('1');
crc32 | crc32 | crc32
-----+-----+-----
 | | 0 | 2212294583
(1 row)
```

有关自定义函数的更多用法，可参考《数据库仓库服务SQL语法参考》中的“CREATE FUNCTION”章节。

## 12.4.14 DWS 以 pg\_toast\_temp\*或 pg\_temp\*开头的 Schema 是什么？

查询Schema列表的时候，发现查询结果存在pg\_temp\*或pg\_toast\_temp\*的Schema，如下图所示。

SELECT \* FROM pg\_namespace;

| namespace                                 | nspowner | nspacl                                  | permissions | usedspace |
|-------------------------------------------|----------|-----------------------------------------|-------------|-----------|
| pg_toast                                  | postgres |                                         |             | 0         |
| ctore                                     | postgres |                                         |             | 0         |
| gs_logical_cluster                        | postgres |                                         |             | 0         |
| sys                                       | postgres |                                         |             | 0         |
| dbas_om                                   | postgres | { Ruby=IC/Ruby, Ruby=LP/Ruby, =U/Ruby } |             | 24576     |
| dbas_job                                  | postgres | { Ruby=IC/Ruby, Ruby=LP/Ruby, =U/Ruby } |             | 0         |
| pg_catalog                                | postgres | { Ruby=IC/Ruby, Ruby=LP/Ruby, =U/Ruby } |             | 13908896  |
| public                                    | postgres | { Ruby=IC/Ruby, Ruby=LP/Ruby, =U/Ruby } |             | 622592    |
| information_schema                        | postgres | { Ruby=IC/Ruby, Ruby=LP/Ruby, =U/Ruby } |             | 352256    |
| utl_file                                  | postgres | { Ruby=IC/Ruby, Ruby=LP/Ruby, =U/Ruby } |             | 0         |
| dbas_output                               | postgres | { Ruby=IC/Ruby, Ruby=LP/Ruby, =U/Ruby } |             | 0         |
| dbas_random                               | postgres | { Ruby=IC/Ruby, Ruby=LP/Ruby, =U/Ruby } |             | 0         |
| utl_row                                   | postgres | { Ruby=IC/Ruby, Ruby=LP/Ruby, =U/Ruby } |             | 0         |
| dbas_sql                                  | postgres | { Ruby=IC/Ruby, Ruby=LP/Ruby, =U/Ruby } |             | 0         |
| dbas_job                                  | postgres | { Ruby=IC/Ruby, Ruby=LP/Ruby, =U/Ruby } |             | 0         |
| scheduler                                 | postgres |                                         |             | 8192      |
| u1                                        | postgres |                                         |             | 0         |
| u2                                        | postgres |                                         |             | 0         |
| u3                                        | postgres |                                         |             | 0         |
| u4                                        | postgres |                                         |             | 0         |
| s1                                        | postgres |                                         |             | 0         |
| s2                                        | postgres |                                         |             | 0         |
| pg_temp_cn_5003_4_1_281471119284272       | postgres |                                         |             | 0         |
| pg_toast_temp_cn_5003_4_1_281471119284272 | postgres |                                         |             | 0         |
| vac_toast                                 | postgres |                                         |             | 0         |

这些Schema是在创建临时表时，该临时表通过每个会话独立的以pg\_temp开头的Schema来保证只对当前会话可见，因此，不建议用户在日常操作中手动删除以pg\_temp，pg\_toast\_temp开头的Schema。

临时表只在当前会话可见，本会话结束后会自动删除，这些相应的Schema也会被删除。

## 12.4.15 DWS 查询时结果不一致的常见场景和解决方法

DWS中使用SQL语句查询时会出现同一条查询语句返回结果不一致的问题，此类问题大部分都是由于语法使用不当或用法不合理导致，通过合理的业务使用可以避免此类问题。以下列举了几种常见的查询结果不一致的场景和对应的解决办法供参考：

## 窗口函数中使用排序后取部分结果

### 场景：

窗口函数row\_number()中使用排序后查询表t3的c列，两次查询结果不同。

```
SELECT * FROM t3 order by 1,2,3;
a | b | c
---+---+---
1 | 2 | 1
1 | 2 | 2
1 | 2 | 3
(3 rows)

SELECT c,rn FROM (select c,row_number() over(order by a,b) as rn from t3) where rn = 1;
c | rn
---+---
1 | 1
(1 row)
SELECT c,rn FROM (select c,row_number() over(order by a,b) as rn from t3) where rn = 1;
c | rn
---+---
3 | 1
(1 row)
```

### 原因分析：

如上所示，执行同一条语句：*select c,rn from (select c,row\_number() over(order by a,b) as rn from t3) where rn = 1;* 两次查询结果不同，因为在窗口函数的排序列a、b上存在重复值1、2且重复值在c列上的值不同，就会导致每次按照a，b列排序结果取第一条时，所取的数据是随机的，造成结果集不一致。

### 解决方法：

该场景需要将取值列c列也加到排序中，使排序结果获取的第一条数据固定。

```
SELECT c,rn FROM (select c,row_number() over(order by a,b,c) as rn from t3) where rn = 1;
c | rn
---+---
1 | 1
(1 row)
```

## 子视图/子查询中使用排序

### 场景：

创建表test和视图v后，子查询中使用排序查询表test，出现查询结果不一致。

```
CREATE TABLE test(a serial ,b int);
INSERT INTO test(b) VALUES(1);
INSERT INTO test(b) SELECT b FROM test;
...
INSERT INTO test(b) SELECT b FROM test;
CREATE VIEW v as SELECT * FROM test ORDER BY a;
```

### 问题SQL：

```
SELECT * FROM v limit 1;
a | b
---+---
3 | 1
(1 row)

SELECT * FROM (select * from test order by a) limit 10;
a | b
---+---
14 | 1
(1 row)
```

```
SELECT * FROM test order by a limit 10;
a | b
---+---
1 | 1
(1 row)
```

**原因分析：**

在子视图和子查询中，ORDER BY子句是无效的。

**解决方法：**

不建议在子视图和子查询中使用order by，若要保证结果有序，需在最外层查询中使用order by。

## 子查询 limit

**场景：**子查询中使用limit，两次查询结果不一致。

```
SELECT * FROM (select a from test limit 1) order by 1;
a

5
(1 row)

SELECT * FROM (select a from test limit 1) order by 1;
a

1
(1 row)
```

**原因分析：**

子查询中的limit会导致获取随机结果，从而最终查询结果为随机提取。

**解决方法：**

要保证最终查询结果的稳定，需避免在子查询中使用limit。

## 使用 string\_agg

**场景：**使用string\_agg查询表employee，出现查询结果不一致。

```
SELECT * FROM employee;
empno | ename | job | mgr | hiredate | sal | comm | deptno
-----+-----+-----+-----+-----+-----+-----+-----
7654 | MARTIN | SALEMAN | 7698 | 2022-11-08 00:00:00 | 12000 | 1400 | 30
7566 | JONES | MANAGER | 7839 | 2022-11-08 00:00:00 | 32000 | 0 | 20
7499 | ALLEN | SALEMAN | 7698 | 2022-11-08 00:00:00 | 16000 | 300 | 30
(3 rows)

SELECT count(*) FROM (select deptno, string_agg(ename, ',') from employee group by deptno) t1, (select
deptno, string_agg(ename, ',') from employee group by deptno) t2 where t1.string_agg = t2.string_agg;
count

2
(1 row)

SELECT count(*) FROM (select deptno, string_agg(ename, ',') from employee group by deptno) t1, (select
deptno, string_agg(ename, ',') from employee group by deptno) t2 where t1.string_agg = t2.string_agg;
count

1
(1 row)
```

**原因分析：**

String\_agg函数的作用是将组内的数据合并成一行，但是如果某用户的用法是string\_agg(ename, ','), 结果集就是不稳定的，因为没有指定组合的顺序。例如，上述语句中，对于select deptno, string\_agg(ename, ',') from employee group by deptno;

输出结果既可以是：

```
30 | ALLEN,MARTIN
```

也可能是：

```
30 | MARTIN,ALLEN
```

两个结果都是合理的，因此上述关联场景下，有可能出现t1这个subquery中的结果和t2这个subquery中的结果对于deptno=30时的输出结果不一致。

**解决方法：**

String\_agg中增加order by排序，保证按顺序拼接。

```
SELECT count(*) FROM (select deptno, string_agg(ename, ',' order by ename desc) from employee group by deptno) t1 ,(select deptno, string_agg(ename, ',' order by ename desc) from employee group by deptno) t2 where t1.string_agg = t2.string_agg;
```

## 数据库兼容模式

**场景：**在数据库中查询空串结果不一致。

database1 (TD兼容模式)：

```
td=# select " is null;
isnull

f
(1 row)
```

database2 (ORA兼容模式)：

```
ora=# select " is null;
isnull

t
(1 row)
```

**原因分析：**

查询空串结果不同是由于不同数据库兼容模式下空串与null语法有差异导致。

目前，DWS支持三种数据库兼容模式：Oracle、TD和MySQL，不同兼容模式下语法和行为存在差异，兼容性差异说明可参考《数据仓库服务开发指南》中“Oracle、Teradata和MySQL语法兼容性差异”。

不同兼容模式下的database表现出不同的兼容性行为属于正常现象。可以通过查看select datname, datcompatibility from pg\_database,确认数据库兼容性设置是否相同。

**解决方法：**

这种场景下只能将两个database的兼容性模式设置为一致的才能解决。Database的DBCOMPATIBILITY属性不支持ALTER，只能通过新建数据库的方法，在创建数据库时指定相同的DBCOMPATIBILITY属性解决。

## 数据库兼容性行为配置项 behavior\_compat\_options 设置不同

**场景：**add\_months函数计算结果不一致。

database1:

```
SELECT add_months('2018-02-28',3) from dual;
add_months

2018-05-28 00:00:00
(1 row)
```

database2:

```
SELECT add_months('2018-02-28',3) from dual;
add_months

2018-05-31 00:00:00
(1 row)
```

**原因分析：**

数据库兼容性配置项behavior\_compat\_options不同会导致部分行为不同，该参数选项可参考《数据仓库服务开发指南》中“GUC参数>其他选项>behavior\_compat\_options”中的相关选项描述。

此场景中behavior\_compat\_options配置项中的end\_month\_calculate参数控制add\_months函数计算逻辑配置项。设置end\_month\_calculate配置项时，如果param1的日期（Day字段）为月末，并且param1的日期（Day字段）比result月份的月末日期小，计算结果中的日期字段（Day字段）和result的月末日期保持一致。

**解决方法：**

需要将数据库中参数behavior\_compat\_options的兼容性配置项设置为一致。该参数类型为USERSET类型，可session级别设置或集群级修改。

## 自定义函数属性设置不合理

**场景：**自定义函数get\_count()并调用该函数出现结果不一致场景。

```
CREATE FUNCTION get_count() returns int
SHIPPABLE
as $$
declare
 result int;
begin
 result = (select count(*) from test); --test表是hash表
 return result;
end;
$$
language plpgsql;
```

调用该函数。

```
SELECT get_count();
get_count

2106
(1 row)

SELECT get_count() FROM t_src;
get_count

1032
(1 row)
```

**原因分析：**

由于该函数指定了SHIPPABLE的函数属性，因此生成计划时该函数会下推到DN上执行，该函数下推到DN后，由于函数定义中的test表是hash表，因此每个DN上只有该表的一部分数据，所以select count(\*) from test; 返回的结果不是test表全量数据的结果，而是每个DN上部分数据的结果，因此导致加上from表后函数返回预期发生变化。

**解决方法：**

以下两种方法任选其一即可（推荐第一种方法）：

1. 将函数改为不下推：ALTER FUNCTION get\_count() not shippable;
2. 将函数中用到的表改为复制表，这样每个DN上都是一份该表的全量数据，即使下推到DN执行，也能保证结果集符合预期。

## 使用 UNLOGGED 表

**场景：**

使用unlogged表后，在集群重启后，关联查询结果集异常，查看unlogged表缺少部分数据。

**原因分析：**

如果设置max\_query\_retry\_times为0，且在建表时指定UNLOGGED关键字，则创建的表为非日志表。在非日志表中写入的数据不会被写入到预写日志中，这样就会比普通表快很多。但是非日志表在冲突、执行操作系统重启、强制重启、切断电源操作或异常关机后会被自动截断，会造成数据丢失的风险。非日志表中的内容也不会被复制到备服务器中。在非日志表中创建的索引也不会被自动记录。因此当集群发生异常重启（进程重启、节点故障、集群重启）时，会导致部分内存中的数据未及时落盘，造成部分数据丢失，从而导致结果集异常。

**解决方法：**

unlogged表在集群异常情况下的安全性无法保证，一般不能作为业务表使用，更多的场景是作为临时表使用。当出现集群故障后，为了保证数据正常，需要重建unlogged表或将数据备份后重新导入数据库。

## 12.4.16 DWS 哪些系统表不能做 VACUUM FULL

从功能实现上，DWS的系统表都可以做VACUUM FULL，但是会上八级锁，涉及这些系统表的业务会被阻塞。

根据数据库版本不同，建议如下：

### 8.1.3 及以上版本

- 8.1.3及以上版本的集群，AUTO VACUUM默认是打开的（由GUC参数autovacuum控制），用户通过设置对应的GUC参数后，系统会自动触发所有系统表和用户的行存表进行VACUUM FULL，用户不需要手动执行vacuum。
  - autovacuum\_max\_workers = 0，系统表和普通表都不会触发。
  - autovacuum = off，普通表不会触发，但系统表会触发。
- 以上仅针对行存表的AUTO VACCUM触发，如果需要针对列存表做自动触发VACUUM，还需要用户在管理控制台上配置智能调度任务。

### 8.1.1 及历史版本

1. 以下系统表在做VACUUM FULL时会影响所有业务，请选择空闲时间窗或停止业务时操作。
  - pg\_statistic（统计信息，建议不要清理，会影响业务查询性能）
  - pg\_attribute
  - pgxc\_class
  - pg\_type
  - pg\_depend
  - pg\_class
  - pg\_index
  - pg\_proc
  - pg\_partition
  - pg\_object
  - pg\_shdepend
2. 以下系统表主要影响资源的监控和表大小的查询接口，不影响其他业务。
  - gs\_wlm\_user\_resource\_history
  - gs\_wlm\_session\_info
  - gs\_wlm\_instance\_history
  - gs\_respool\_resource\_history
  - pg\_relfilenode\_size
3. 其余系统表不占用空间，通常不做清理。
4. 建议日常运维活动中，每周监控以下系统表的大小，如果一定要回收这些空间，优先处理关键系统表。

语句如下：

```
SELECT c.oid,c.relname, c.relkind, pg_relation_size(c.oid) AS size FROM pg_class c WHERE c.relkind IN ('r') AND c.oid <16385 ORDER BY size DESC;
```

## 12.4.17 DWS 语句处于 idle in transaction 状态常见场景

在使用PGXC\_STAT\_ACTIVITY视图查询用户SQL相关信息时，查询结果中的state字段有时会显示“idle in transaction”。idle in transaction具体含义为：后端在事务中，但事务中没有语句在执行。该状态表示该条语句已经执行完成，因此query\_id为0，但是本事务还未提交或回滚。此状态下的语句已经执行完成，不占用CPU和IO等资源，会占用连接数，并发数等连接资源。

若业务中出现语句处于idle in transaction状态，可参考如下常见场景及对应的解决方法来处理：

### 场景一：事务开启后没有提交，语句处于 idle in transaction

手动BEGIN/START TRANSACTION开启事务，执行某语句后，不执行COMMIT/ROLLBACK，此时执行如下命令查看视图PGXC\_STAT\_ACTIVITY：

```
SELECT state, query, query_id FROM pgxc_stat_activity;
```

查看结果显示：该语句状态为idle in transaction。

| state               | query                                                      | query_id           |
|---------------------|------------------------------------------------------------|--------------------|
| active              |                                                            | 0                  |
| idle                |                                                            | 0                  |
| idle                |                                                            | 0                  |
| active              | WLM fetch collect info from data nodes                     | 73464968921613282  |
| active              | WLM calculate space info process                           | 0                  |
| active              | WLM monitor update and verify local info                   | 73464968921613276  |
| active              | WLM arbiter sync info by CCN and CNs                       | 0                  |
| idle in transaction | select count(1) from t group by a order by 1 desc limit 1; | 0                  |
| idle                |                                                            | 0                  |
| active              | select state,query,query_id from pgxc_stat_activity;       | 73464968921613283  |
| active              |                                                            | 0                  |
| idle                |                                                            | 0                  |
| idle                |                                                            | 0                  |
| active              | WLM fetch collect info from data nodes                     | 145522562959541153 |
| active              | WLM calculate space info process                           | 0                  |
| active              | WLM monitor update and verify local info                   | 145522562959541123 |
| active              | WLM arbiter sync info by CCN and CNs                       | 0                  |
| active              | SELECT * FROM pg_stat_activity                             | 73464968921613283  |
| idle                |                                                            | 0                  |

(19 rows)

**解决方法：**这种场景下需要手动对开启的事务执行COMMIT/ROLLBACK即可。

**场景二：存储过程中有 DDL 语句，该存储过程结束前，其他节点上 DDL 语句执行完后的状态是 idle in transaction**

先创建存储过程：

```
CREATE OR REPLACE FUNCTION public.test_sleep()
RETURNS void
LANGUAGE plpgsql
AS $$
BEGIN
 truncate t1;
 truncate t2;
 EXECUTE IMMEDIATE 'select pg_sleep(6)';
 RETURN;
END$$;
```

再执行如下命令查看PGXC\_STAT\_ACTIVITY视图：

```
SELECT coorname,pid,query_id,state,query,username FROM pgxc_stat_activity WHERE username='jack';
```

查看结果显示：truncate t2处于idle in transaction状态，coorname为coordinator2。说明cn2上该语句已经执行完成，该存储过程在执行下一条语句。

| coorname     | pid             | query_id          | state               | query                | username |
|--------------|-----------------|-------------------|---------------------|----------------------|----------|
| coordinator1 | 139767124588288 | 73464968921614213 | active              | select test_sleep(); | jack     |
| coordinator2 | 140055318353664 | 0                 | idle in transaction | truncate t2          | jack     |

(2 rows)

**解决方法：**此类场景是由于存储过程执行慢导致，等存储过程执行完成即可，也可考虑优化存储过程中执行时间较长的语句。

**场景三：大量 SAVEPOINT/RELEASE 语句处于 idle in transaction（8.1.0 之前集群版本）**

执行如下命令查看PGXC\_STAT\_ACTIVITY视图：

```
SELECT coorname,pid,query_id,state,query,username FROM pgxc_stat_activity WHERE username='jack';
```

结果显示SAVEPOINT/RELEASE语句处于idle in transaction。

| coorname     | pid             | query_id          | state               | query                 | username |
|--------------|-----------------|-------------------|---------------------|-----------------------|----------|
| coordinator1 | 140127877723904 | 77687093572141691 | active              | select test sleep1(); | jack     |
| coordinator2 | 139773127153408 | 0                 | idle in transaction | release s1            | jack     |
| coordinator3 | 140193352906496 | 0                 | idle in transaction | release s1            | jack     |

(3 rows)

**解决方法：**

SAVEPOINT和RELEASE语句是带EXCEPTION的存储过程执行时系统自动生成的（8.1.0之后的集群版本不再向CN下发SAVEPOINT），DWS带EXCEPTION的存储过程在实现上基于子事务实现，简单对应关系如下：

```
begin
 (Savepoint s1)
 DDL/DML
exception
 (Rollback to s1)
 (Release s1)
 ...
end
```

存储过程启动时如果有EXCEPTION，则会启动一个子事务，如果执行过程中出现EXCEPTION，则会回滚当前事务并进行异常的处理；如果没有出现EXCEPTION则会直接提交前面的子事务。

当此类存储过程较多且有嵌套时容易出现，与场景二类似，等整个存储过程执行完即可。如果RELEASE较多，说明存储过程触发了多个EXCEPTION，可分析存储过程逻辑是否合理。

12.4.18 DWS 如何实现行转列及列转行？

本节介绍DWS中如何使用SQL语句实现行转列、列转行。

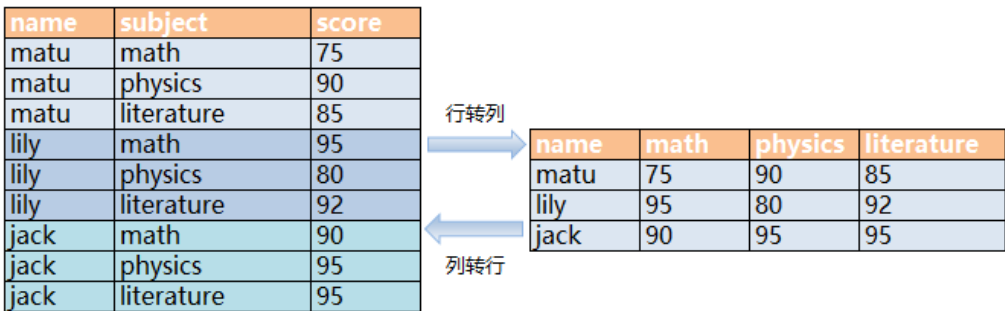
场景介绍

以学生成绩为例：

老师会按照学科录入成绩，每科老师都会单独录入每个学生对应学科的成绩，而每位学生只关注自己各科的成绩。如果把老师录入数据作为原始表，那么学生查看自己的成绩就要用到行转列；如果让学生自己填写各科的成绩并汇总，然后老师去查自己学科所有学生的成绩，那就是列转行。

行转列与列转行的示意图如下：

图 12-2 示意图



- 行转列  
将多行数据转换成一列显示，或将一列数据转换成多列显示。
- 列转行  
将一行数据转换成多行显示，或将多列数据转换成一列显示。

## 示例表

- 创建行存表students\_info并插入数据。

```
CREATE TABLE students_info(name varchar(20),subject varchar(100),score bigint) distribute by hash(name);
INSERT INTO students_info VALUES('lily','math',95);
INSERT INTO students_info VALUES('lily','physics',80);
INSERT INTO students_info VALUES('lily','literature',92);
INSERT INTO students_info VALUES('matu','math',75);
INSERT INTO students_info VALUES('matu','physics',90);
INSERT INTO students_info VALUES('matu','literature',85);
INSERT INTO students_info VALUES('jack','math',90);
INSERT INTO students_info VALUES('jack','physics',95);
INSERT INTO students_info VALUES('jack','literature',95);
```

查看表students\_info信息。

```
SELECT * FROM students_info;
name | subject | score
-----+-----+-----
matu | math | 75
matu | physics | 90
matu | literature | 85
lily | math | 95
lily | physics | 80
lily | literature | 92
jack | math | 90
jack | physics | 95
jack | literature | 95
```

- 创建列存表students\_info1并插入数据。

```
CREATE TABLE students_info1(name varchar(20), math bigint, physics bigint, literature bigint) with (orientation = column) distribute by hash(name);
INSERT INTO students_info1 VALUES('lily',95,80,92);
INSERT INTO students_info1 VALUES('matu',75,90,85);
INSERT INTO students_info1 VALUES('jack',90,95,95);
```

查看表students\_info1信息。

```
SELECT * FROM students_info1;
name | math | physics | literature
-----+-----+-----+-----
matu | 75 | 90 | 85
lily | 95 | 80 | 92
jack | 90 | 95 | 95
(3 rows)
```

## 静态行转列

静态行转列需要手动指定每一列的列名，如果存在则取其对应值，否则将赋其默认值0。

```
SELECT name,
sum(case when subject='math' then score else 0 end) as math,
sum(case when subject='physics' then score else 0 end) as physics,
sum(case when subject='literature' then score else 0 end) as literature FROM students_info GROUP BY name;
name | math | physics | literature
-----+-----+-----+-----
matu | 75 | 90 | 85
lily | 95 | 80 | 92
jack | 90 | 95 | 95
(3 rows)
```

## 动态行转列

8.1.2及以上集群版本可使用GROUP\_CONCAT生成列存语句。

```
SELECT group_concat(concat('sum(IF(subject = ', subject, ', score, 0)) AS ', name, ''))FROM students_info;
group_concat

sum(IF(subject = 'literature', score, 0)) AS "jack",sum(IF(subject = 'literature', score, 0)) AS
'lily',sum(IF(subject = 'literature', score, 0)) AS "matu",sum(IF(subject = 'math', score, 0)) AS "jack",sum(IF
(subject = 'math', score, 0)) AS "lily",sum(IF(subject = 'math', score, 0)) AS "matu",sum(IF(subject =
'physics', score, 0)) AS "jack",sum(IF(subject = 'physics', score, 0)) AS "lily",sum(IF(subject = 'physics
', score, 0)) AS "matu"
(1 row)
```

8.1.1及更低版本中可用LISTAGG生成列存语句。

```
SELECT listagg(concat('sum(case when subject = ', subject, ' then score else 0 end) AS ', subject, ''),',')
within GROUP(ORDER BY 1)FROM (select distinct subject from students_info);
listagg

--
sum(case when subject = 'literature' then score else 0 end) AS "literature",sum(case when subject =
'physics' then score else 0 end) AS "physics",sum(case when subject = 'math' then score else 0 end) AS
"math
"
(1 row)
```

再通过视图动态重建：

```
CREATE OR REPLACE FUNCTION build_view()
RETURNS VOID
LANGUAGE plpgsql
AS $$ DECLARE
sql text;
rec record;
BEGIN
sql := 'select LISTAGG(
CONCAT("sum(case when subject = ''', subject, '' then score else 0 end) AS ', subject, '"")
, "") within group(order by 1) from (select distinct subject from students_info);';
EXECUTE sql INTO rec;
sql := 'drop view if exists get_score';
EXECUTE sql;
sql := 'create view get_score as select name, ' || rec.LISTAGG || ' from students_info group by name';
EXECUTE sql;
END$$;
```

执行重建：

```
CALL build_view();
```

查询视图：

```
SELECT * FROM get_score;
name | literature | physics | math
-----+-----+-----+-----
matu | 85 | 90 | 75
lily | 92 | 80 | 95
jack | 95 | 95 | 90
(3 rows)
```

## 列转行

使用union all，将各科目（math、physics和literature）整合为一列，示例如下：

```
SELECT * FROM
(
 SELECT name, 'math' AS subject, math AS score FROM students_info1
 union all
 SELECT name, 'physics' AS subject, physics AS score FROM students_info1
 union all
 SELECT name, 'literature' AS subject, literature AS score FROM students_info1
)
order by name;
```

| name | subject    | score |
|------|------------|-------|
| jack | math       | 90    |
| jack | physics    | 95    |
| jack | literature | 95    |
| lily | math       | 95    |
| lily | physics    | 80    |
| lily | literature | 92    |
| matu | math       | 75    |
| matu | physics    | 90    |
| matu | literature | 85    |

(9 rows)

### 12.4.19 DWS 唯一约束和唯一索引有什么区别？

- 唯一约束和唯一索引概念上不同  
唯一约束确保一列或者一组列中包含的数据对于表中所有的行都是唯一的。如果没有声明DISTRIBUTE BY REPLICATION，则唯一约束的列集合中必须包含分布列。  
唯一索引用于限制索引字段值的唯一性，或者是多个字段组合值的唯一性。  
CREATE UNIQUE INDEX创建唯一索引。
- 唯一约束和唯一索引功能上不同  
约束主要是为了保证数据的完整性，索引主要是为了辅助查询。
- 唯一约束和唯一索引使用方法上不同
  - a. 唯一约束和唯一索引，都可以实现列数据的唯一，列值可以有NULL。
  - b. 创建唯一约束，会自动创建一个同名的唯一索引，该索引不能单独删除，删除约束会自动删除索引。唯一约束是通过唯一索引来实现数据的唯一。DWS行存表支持唯一约束，而列存表不支持。
  - c. 创建一个唯一索引，这个索引独立的、可以单独删除。目前，DWS只有B-Tree可以创建唯一索引。
  - d. 如果一个列上想有约束和索引，且两者可以单独的删除。可以先建唯一索引，再建同名的唯一约束。
  - e. 如果表的一个字段，要作为另外一个表的外键，这个字段必须有唯一约束（或是主键），如果只是有唯一索引，就会报错。

示例：创建两个列的复合索引，并不要求是唯一索引。

```
CREATE TABLE t (n1 number,n2 number,n3 number,PRIMARY KEY (n3));
CREATE INDEX t_idx ON t(n1,n2);
```

DWS支持多个唯一索引。

```
CREATE UNIQUE INDEX u_index ON t(n3);
CREATE UNIQUE INDEX u_index1 ON t(n3);
```

可以使用上述示例创建的索引t\_idx来创建唯一约束t\_uk，而且它只在列n1上唯一，也就是说唯一约束比索引更加严格。

```
ALTER TABLE t ADD CONSTRAINT t_uk UNIQUE USING INDEX u_index;
```

12.4.20 DWS 函数和存储过程有什么区别？

函数和存储过程是数据库管理系统中常见的两种对象，它们在实现特定功能时具有相同点，也有不同点。了解它们的特点和适用场景，对于合理设计数据库结构和提高数据库性能具有重要意义。

表 12-5 函数和存储过程的区别

| 函数                                                    | 存储过程                                                                    |
|-------------------------------------------------------|-------------------------------------------------------------------------|
| 两者都可以用于实现特定的功能。无论是函数还是存储过程，都可以封装一系列的SQL语句，以完成某些特定的操作。 |                                                                         |
| 两者都可以接收输入参数，并且根据参数的不同来进行相应的操作。                        |                                                                         |
| 函数的标识符为FUNCTION。                                      | 存储过程的标识符为PROCEDURE。                                                     |
| 函数必须返回一个具体的值，并且规定返回值的数值类型。                            | 存储过程可以没有返回值，也可以有返回值，甚至可以有多个返回值，可以通过输出参数返回结果，也可以直接在存储过程中使用SELECT语句返回结果集。 |
| 函数适用于需要返回单个值的情况，比如计算某个数值、字符串处理、返回表等。                  | 存储过程适用于需要执行DML操作的情况，比如批量插入、更新、删除数据等。                                    |

• 创建并调用函数

创建表emp并插入数据，查询表数据如下：

```
SELECT * FROM emp;
empno | ename | job | mgr | hiredate | sal | comm | deptno
-----+-----+-----+-----+-----+-----+-----+-----
7369 | SMITH | CLERK | 7902 | 1980-12-17 00:00:00 | 800.00 | | 20
7499 | ALLEN | SALESMAN | 7698 | 1981-02-20 00:00:00 | 1600.00 | 300.00 | 30
7566 | JONES | MANAGER | 7839 | 1981-04-02 00:00:00 | 2975.00 | | 20
7521 | WARD | SALESMAN | 7698 | 1981-02-22 00:00:00 | 1250.00 | 500.00 | 30
(4 rows)
```

创建函数emp\_comp，用于接受两个数字作为输入并返回计算值：

```
CREATE OR REPLACE FUNCTION emp_comp (
 p_sal NUMBER,
 p_comm NUMBER
) RETURN NUMBER
IS
BEGIN
 RETURN (p_sal + NVL(p_comm, 0)) * 24;
END;
/
```

使用SELECT命令调用函数：

```
SELECT ename "Name", sal "Salary", comm "Commission", emp_comp(sal, comm) "Total
Compensation" FROM emp;
Name | Salary | Commission | Total Compensation
-----+-----+-----+-----
SMITH | 800.00 | | 19200.00
```

```
ALLEN | 1600.00 | 300.00 | 45600.00
JONES | 2975.00 | | 71400.00
WARD | 1250.00 | 500.00 | 42000.00
(4 rows)
```

- **创建并调用存储过程**

创建表MATCHES并插入数据，查询表数据如下：

```
SELECT * FROM MATCHES;
matchno | teamno | playerno | won | lost
-----+-----+-----+-----+-----
1 | 1 | 6 | 3 | 1
7 | 1 | 57 | 3 | 0
8 | 1 | 8 | 0 | 3
9 | 2 | 27 | 3 | 2
11 | 2 | 112 | 2 | 3
(5 rows)
```

创建存储过程delete\_matches，用于删除给定球员参加的所有比赛：

```
CREATE PROCEDURE delete_matches(IN p_playerno INTEGER)
AS
BEGIN
 DELETE FROM MATCHES WHERE playerno = p_playerno;
END;
/
```

调用存储过程delete\_matches：

```
CALL delete_matches(57);
```

再次查询表MATCHES，由返回结果可知，playerno为57的数据已被删除：

```
SELECT * FROM MATCHES;
matchno | teamno | playerno | won | lost
-----+-----+-----+-----+-----
11 | 2 | 112 | 2 | 3
8 | 1 | 8 | 0 | 3
1 | 1 | 6 | 3 | 1
9 | 2 | 27 | 3 | 2
(4 rows)
```

## 12.4.21 如何删除 DWS 重复的表数据？

清理数据库脏数据时，可能会有多条重复数据只保留一条的场景，此场景可以使用聚合函数或窗口函数来实现。

### 构建表数据

**步骤1** 创建表t\_customer，向表中插入包含重复记录的数据：

```
CREATE TABLE t_customer (
 id int NOT NULL,
 cust_name varchar(32) NOT NULL COMMENT '名字',
 gender varchar(10) NOT NULL COMMENT '性别',
 email varchar(32) NOT NULL COMMENT 'email',
 PRIMARY KEY (id)
);

INSERT INTO t_customer VALUES ('1', 'Tom', 'Male', 'high_salary@sample.com');
INSERT INTO t_customer VALUES ('2', 'Jennifer', 'Female', 'good_job@sample.com');
INSERT INTO t_customer VALUES ('3', 'Tom', 'Male', 'high_salary@sample.com');
INSERT INTO t_customer VALUES ('4', 'John', 'Male', 'good_job@sample.com');
INSERT INTO t_customer VALUES ('5', 'Jennifer', 'Female', 'good_job@sample.com');
INSERT INTO t_customer VALUES ('6', 'Tom', 'Male', 'high_salary@sample.com');
```

**步骤2** 查询表t\_customer：

```
SELECT * FROM t_customer ORDER BY id;
```

| id | cust_name | gender | email                  |
|----|-----------|--------|------------------------|
| 1  | Tom       | Male   | high_salary@sample.com |
| 2  | Jennifer  | Female | good_job@sample.com    |
| 3  | Tom       | Male   | high_salary@sample.com |
| 4  | John      | Male   | good_job@sample.com    |
| 5  | Jennifer  | Female | good_job@sample.com    |
| 6  | Tom       | Male   | high_salary@sample.com |

#### ----结束

当客户的名字、性别、邮件都相同时，则判定它们为重复的记录。对于表 t\_customer，id 等于 1、3、6 的为重复数据，id 为 2、5 的也是重复数据，删除多余的数据的同时需要保留其中的一条。

#### 方法一：使用聚合函数 min(expr)

使用聚合函数通过子查询取出 id 最小的不重复行，然后通过 NOT IN 删除重复数据。

##### 步骤1 查询id最小的不重复行：

```
SELECT
 min(id) id,
 cust_name,
 gender,
 COUNT(cust_name) count
FROM t_customer
GROUP BY cust_name,gender
ORDER BY id;
```

| id | cust_name | gender | count |
|----|-----------|--------|-------|
| 1  | Tom       | Male   | 3     |
| 2  | Jennifer  | Female | 2     |
| 4  | John      | Male   | 1     |

通过查询结果可知，重复的数据行 id 为 3、5、6 的数据被过滤掉了。

##### 步骤2 使用 NOT IN 过滤重复数据行并删除：

```
DELETE from t_customer where id not in (
 SELECT
 min(id) id
 FROM t_customer
 GROUP BY cust_name,gender
);
```

##### 步骤3 查询删除重复数据后的表 t\_customer：

```
SELECT * FROM t_customer ORDER BY id;
```

| id | cust_name | gender | email                  |
|----|-----------|--------|------------------------|
| 1  | Tom       | Male   | high_salary@sample.com |
| 2  | Jennifer  | Female | good_job@sample.com    |
| 4  | John      | Male   | good_job@sample.com    |

由返回结果可知，重复数据已被删除。

#### ----结束

#### 方法二：使用窗口函数 row\_number()

通过 PARTITION BY 对列进行分区排序并生成序列号，然后将序号大于 1 的行删除。

**步骤1** 分区查询，对列进行分区排序并生成序号列：

```
SELECT
 id,
 cust_name,
 gender,
 ROW_NUMBER() OVER (PARTITION BY cust_name,gender ORDER BY id) num
FROM t_customer;
```

| id | cust_name | gender | num |
|----|-----------|--------|-----|
| 4  | John      | Male   | 1   |
| 1  | Tom       | Male   | 1   |
| 3  | Tom       | Male   | 2   |
| 6  | Tom       | Male   | 3   |
| 2  | Jennifer  | Female | 1   |
| 5  | Jennifer  | Female | 2   |

由返回结果可知，num>1的数据即为重复数据。

**步骤2** 删除num>1的数据：

```
DELETE FROM t_customer WHERE id in (
 SELECT id FROM (
 SELECT * FROM (
 SELECT ROW_NUMBER() OVER w AS row_num,id
 FROM t_customer
 WINDOW w AS (PARTITION BY cust_name,gender ORDER BY id)
 WHERE row_num > 1)
);
```

**步骤3** 查询删除重复数据后的表t\_customer：

```
SELECT * FROM t_customer ORDER BY id;
```

| id | cust_name | gender | email                  |
|----|-----------|--------|------------------------|
| 1  | Tom       | Male   | high_salary@sample.com |
| 2  | Jennifer  | Female | good_job@sample.com    |
| 4  | John      | Male   | good_job@sample.com    |

----结束

## 12.5 集群管理

### 12.5.1 如何清理与回收 DWS 存储空间？

DWS数据仓库中保存的数据在删除后，可能没有释放占用的磁盘空间形成脏数据，导致磁盘浪费、创建及恢复快照性能下降等问题，如何清理？

清理与回收存储空间对系统的影响如下：

- 删除无用的脏数据，释放存储空间。
- 数据库将进行大量读写操作，可能影响正常使用，建议选择空闲时间执行。
- 数据库的存储空间越大，即数据可能越多，清理的时间越长。

定期进行脏数据清理，可以清理与回收存储空间。不同集群版本下操作步骤有所不同，具体如下：

8.1.3 及以上版本：通过管理控制台“智能运维”功能进行启动清理

- 步骤1 登录DWS控制台。
- 步骤2 在集群列表中单击指定集群名称。
- 步骤3 进入“集群详情”页面，切换至“智能运维”页签。
- 步骤4 在运维详情部分切换至运维计划模块。单击“添加运维任务”按钮。

| 运维计划                            | 运维计划 |
|---------------------------------|------|
| 名称                              | 操作   |
| <input type="checkbox"/> 新增计划   | 新增计划 |
| <input type="checkbox"/> Vacuum | 删除计划 |
| <input type="checkbox"/> Vacuum | 删除计划 |
| <input type="checkbox"/> Vacuum | 删除计划 |
| <input type="checkbox"/> Vacuum | 删除计划 |
| <input type="checkbox"/> Vacuum | 删除计划 |
| <input type="checkbox"/> Vacuum | 删除计划 |
| <input type="checkbox"/> Vacuum | 删除计划 |
| <input type="checkbox"/> Vacuum | 删除计划 |
| <input type="checkbox"/> Vacuum | 删除计划 |

- 步骤5 弹出添加运维任务边栏，
- 运维任务选择“Vacuum”。
  - 调度模式选择“自动”，DWS将自动扫描Vacuum目标。
  - Vacuum目标选择系统表或用户表：
    - 如果用户业务UPDATE、DELETE较多，选择用户表。
    - 如果创建表、删除表较多，选择系统表。

添加运维任务

基础配置

定时配置

配置确认

\* 运维任务

Vacuum

任务描述

请输入任务描述

备注

\* 调度模式

自动

自动Vacuum目标

☒ 用户表VacuumFull ☐ 系统表Vacuum

高级配置

默认配置 自定义

自动Vacuum触发条件

☒ Vacuum膨胀率 30 %

☒ 目标表可回收空间 100 G...

下一步：定时配置

取消

- 步骤6 单击“下一步：定时配置”，配置Vacuum类型，推荐选择“周期型任务”，DWS将自动在自定义时间窗内执行Vacuum。

添加运维任务

基础配置

定时配置

配置确认

\* 运维类型

☐ 单次型任务

☒ 周期型任务

周期类型

☐ 每日

☐ 每周

☒ 每月

☒ 全选

☐ 1

☐ 2

☐ 3

☐ 4

☐ 5

☐ 6

☐ 7

☐ 8

☐ 9

☐ 10

☐ 11

☐ 12

☐ 13

☐ 14

☐ 15

☐ 16

☐ 17

☐ 18

☐ 19

☐ 20

☐ 21

☐ 22

☐ 23

☐ 24

☐ 25

☐ 26

☐ 27

☐ 28

☐ 29

☐ 30

☐ 31

每月

\* 周期时间窗

00:00:00

06:00:00

添加

上一步: 基础配置

下一步: 配置确认

取消

说明

对于自动Vacuum运维任务，系统对于用户表的处理方法实际采用的是VACUUM FULL操作。VACUUM FULL执行过程中，本身持有8级锁，会阻塞其他业务，导致锁冲突产生，业务本身会陷入锁等待，20分钟后超时报错。因此，在用户配置时间窗内，应尽量避免执行所有处理表的相关业务。

- 步骤7 确认无误后，单击“下一步：配置确认”，完成配置。
- 结束

8.1.2 及以前版本：手动执行 VACUUM FULL 进行清理

须知

1. VACUUM FULL操作会锁表，VACUUM FULL期间，该表的所有访问会阻塞，并等待VACUUM FULL结束，请合理安排调度时间，避免锁表影响业务。

2. VACUUM FULL是对当前表的有效数据抽出来重新整理，同时清理脏数据，该操作会临时占用额外的整理空间（这部分空间待整理完成后释放），因此空间会先增后降，请提前计算好VACUUM FULL所需要的空间再行处理（额外的整理空间大小=表大小\*（1 - 脏页率））。

步骤1 连接数据库，执行以下SQL语句查询脏页率超过30%的较大表，并且按照表大小从大到小排序。

```
SELECT schemaname AS schema, relname AS table_name, n_live_tup AS analyze_count,
pg_size_pretty(pg_table_size(relid)) as table_size, dirty_page_rate
FROM PGXC_GET_STAT_ALL_TABLES
WHERE schemaName NOT IN ('pg_toast', 'pg_catalog', 'information_schema', 'cstore', 'pmk')
AND dirty_page_rate > 30
ORDER BY table_size DESC, dirty_page_rate DESC;
```

**步骤2** 判断是否有回显结果。

- 是，对于表大小超过10G的表，则执行**步骤3**。
- 否，操作结束。

**步骤3** 将脏页Top5的表，进行VACUUM FULL清理（清理时，如果最高磁盘空间>70%，请串行清理）。

```
VACUUM FULL ANALYZE schema.table_name;
```

----结束

## 12.5.2 为什么 DWS 扩容后已使用存储容量比扩容前减少了很多？

### 原因分析

扩容前，如果您没有执行vacuum清理和回收存储空间，DWS数据仓库中之前被删除的数据，可能没有释放占用的磁盘空间形成脏数据，导致磁盘浪费。

而在扩容时，系统会做一次重分布，集群扩容时新节点添加完成后，原节点存储的业务数据明显多于新节点，此时系统自动在所有节点重新分布保存数据。在开始做重分布时，系统会自动执行一次vacuum，从而释放了存储空间，因此，扩容后已使用存储容量减少了很多。

### 处理方法

建议您定期做vacuum full清理与回收存储空间，防止数据膨胀。

如果执行vacuum后，已使用存储容量仍然占用过高，请分析现有集群规格是否满足业务需求，若不满足，建议您对集群进行扩容。

## 12.5.3 DWS 的磁盘空间/容量是如何统计的？

1. DWS的磁盘总容量统计：以3个数据节点为例，假设每个节点320G，总容量为960G。当存入一个1G的数据，DWS因为副本机制会将这1G的数据在两个节点中都各存一份，共占2G的空间，如果再加上元数据、索引等，实际1G的数据，存入DWS后占用的空间不止2G。所以总容量为960G的3节点集群，总量能存480G的数据。因为存储硬盘本身不贵，客户数据才珍贵。

客户在DWS控制台上创建集群的时候，页面已经是按照一个节点的真正容量空间来统计的。比如dwsx2.xlarge，在创建页面是160G，但实际这个节点的磁盘是有320G的，已经将这个320G显示为160G了，便于客户按实际落盘数据进行创建。

2. 单个节点磁盘的使用情况的确认：

同样的，以总容量为960G，3个数据节点为例，那么每个节点的磁盘容量就是320G。

登录DWS控制台，选择“监控>节点监控>概览”显示每个节点的磁盘以及其他资源占用情况。

#### 说明

- 节点管理看到的磁盘空间是DWS集群内所有的磁盘即系统盘、数据盘加到一起的容量，而在集群概览里看到的磁盘空间只是集群内能做表数据存储的可用空间，另外DWS集群中表是有备份的副本数的，表的备份数据也是需要占用磁盘存储的。
- 如果已确定由于磁盘空间不足导致集群状态只读，告警磁盘不可用的异常场景时，可参考“扩容集群”章节进行节点扩容处理。

## 12.5.4 DWS 添加云监控服务的告警规则时会话数阈值如何设置？

连接数据库后，执行以下SQL语句可以查看当前全局最大并发会话数。

```
show max_active_statements;
```

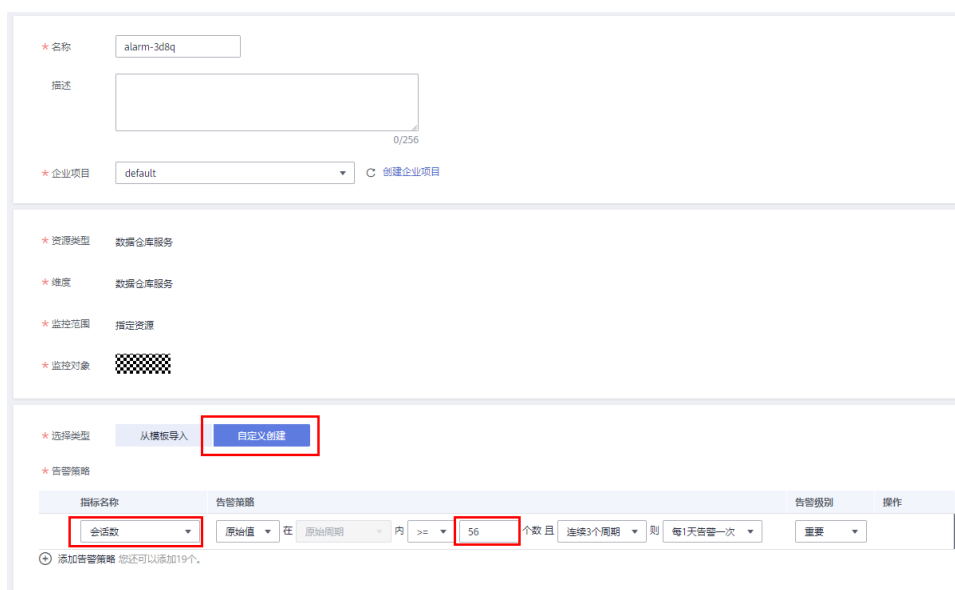
进入到云监控界面，根据查出的全局最大并发会话数，取70%-80%为阈值即可。例如，查询到max\_active\_statements为80，则阈值设置为80\*70%=56。

设置方法：

1. 在DWS控制台，选择“集群 > 集群列表”。
2. 单击集群所在行右侧的“查看监控指标”，进入云监控服务界面。
3. 单击左上角 ，单击集群名称所在行右侧“创建告警规则”。



4. “选择类型”选择“自定义创建”，指标名称选择“会话数”，告警策略填写“56”，告警级别为“重要”，单击“立即创建”。

告警规则创建表单截图。表单包含以下字段：

- \* 名称: alarm-3d8q
- \* 企业项目: default
- \* 资源类型: 数据仓库服务
- \* 维度: 数据仓库服务
- \* 监控范围: 指定资源
- \* 监控对象: 
- \* 选择类型: 从模板导入 (禁用), 自定义创建 (选中)
- \* 告警策略: 指标名称: 会话数, 告警策略: 原始值, 在: 原始周期, 内: >=, 56, 个数, 连续3个周期, 则: 每1天告警一次, 告警级别: 重要

## 12.5.5 DWS 增加 CN 和扩容集群分别在什么场景下使用？

### CN 并发介绍

CN全称为：协调节点（Coordinator Node），是和用户关系最密切也是DWS内部非常重要的一个组件。它负责提供外部应用接口、优化全局执行计划、向Datanode分发执行计划，以及汇总、处理执行结果。CN是外部应用的接口，CN的并发能力直接决定了业务的并发度。

单CN的并发能力受如下几个参数控制：

- **max\_connections**: 允许和数据库连接的最大并发连接数。此参数会影响集群的并发能力。默认值与集群规格有关，具体参见“管理数据库连接”章节。
- **max\_active\_statements**: 设置全局的最大并发数量。此参数只应用到CN，且针对一个CN上的执行作业。默认值60，最多允许60个作业同时运行，其余作业将会排队。

## 选择增加 CN 还是扩容集群

- 连接数不足：初次创建集群时，集群默认的CN节点数是3，能基本满足客户的连接需求。当集群属于高并发请求，各CN节点的连接数很大，或CN节点的CPU明显高于DN节点的CPU时，建议增加CN节点数量，具体参见“管理CN节点”章节。
- 存储容量和性能不足：随着您的业务规模扩张，对数据存储容量和性能有更高的要求时，或者集群整体CPU不足时，建议通过扩容集群进行集群节点的扩容。详情请参见“集群扩容”章节。

随着业务规模扩大，集群扩容到一定节点规模后，也必然需要增加相应的CN节点，满足DWS的分布式要求。简言之，增加CN，不一定需要扩容集群，但是扩容集群后，会伴随着增加CN节点的需求。

## 12.5.6 DWS 在 CPU 核数、内存相同的情况下，小规格多节点与大规格三节点集群如何选择？

- 小规格多节点：  
如果数据量不大、集群节点数量需要伸缩，但不能忍受太高的成本，可以选择小规格多节点集群。  
例如，规格为8核32G的小规格集群（dwsx2.h.2xlarge.4.c6），可以提供较强的计算能力，由于集群节点数量较多，可以处理集群的高并发请求。这种情况下，只需要保证节点间网速通畅，避免集群性能受限。
- 大规格三节点：  
如果需要处理大量数据、需要高性能计算，但可以承受较高的成本，可以选择大规格三节点集群。  
例如，规格为32核256G的大规格集群（dws2.m6.8xlarge.8），拥有更快的CPU处理能力和更大的内存容量，可以更快速地处理数据。但是三节点的集群节点数量有限，高并发场景下性能较差。

## 12.5.7 DWS 热数据存储和冷数据存储的区别？

热数据存储和冷数据存储最大的区别在于存储介质不同：

- 热数据存储是直接将频繁被查询或更新，对访问的响应时间要求很高的热数据存储**在DN数据盘中**。
- 冷数据存储将不更新，偶尔被查询，对访问的响应时间要求不高的冷数据存储**在OBS中**。

因为其存储介质的不同，决定了两者的成本、性能、以及适用场景，如[表12-6](#)所示：

表 12-6 冷热数据存储区别

| 存储名称  | 读取、写入速度 | 成本 | 容量     | 适用场景                                                |
|-------|---------|----|--------|-----------------------------------------------------|
| 热数据存储 | 快       | 高  | 固定，有限制 | 适用于那些数据量有限，需要频繁读取、更新的场景。                            |
| 冷数据存储 | 慢       | 低  | 大、无限制  | 适用于一些归档类业务，利用其低成本，大容量的特点，在完整保存数据的同时，还能节省成本，不占用本地空间。 |

## 12.5.8 DWS 缩容按钮置灰如何处理？

### 问题描述

用户进行缩容操作时，页面“缩容”按钮置灰，无法进入下一步缩容操作。

### 原因分析

在进行缩容操作前系统会提前检查集群是否满足缩容条件，缩容条件不满足，“缩容”按钮置灰。

### 解决办法

需排查集群配置信息，检查缩容是否满足以下条件：

- 集群按照环的方式配置，比如4个或5个主机组成一个环，这些主机上的DN主节点、备节点和从节点都部署在这些节点里，这些节点组成一个集群环，缩容的最小单元是一个集群环，集群至少有2个环才能支持缩容，缩容按照集群环从后往前缩容节点。
- 缩容节点不能包含GTM组件，CM Server组件，CN组件。
- 集群状态为Normal，无其他任务信息。
- 集群租户账号不能处于只读，冻结，受限状态。
- 集群非逻辑集群模式。
- 集群不能有空闲节点。

## 12.6 账户与权限

### 12.6.1 DWS 如何实现业务隔离

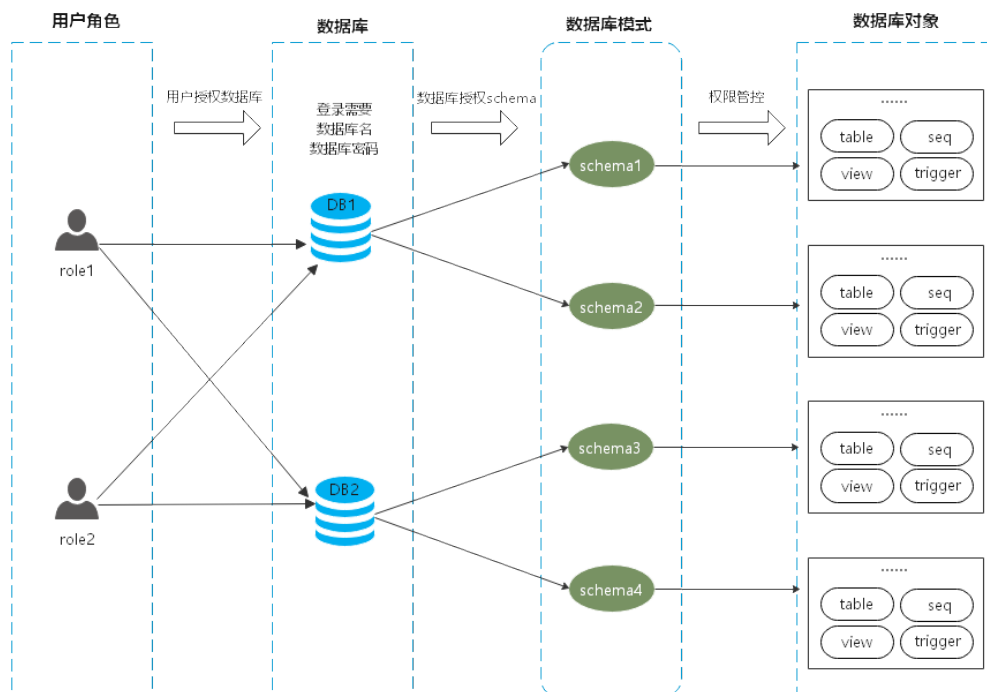
#### 业务隔离

DWS中可以使用Database和Schema实现业务的隔离，区别在于：

- Database之间无法直接互访，通过连接隔离实现彻底的权限隔离。各个Database之间共享资源极少，可实现连接隔离、权限隔离等。
- Schema隔离的方式共用资源较多，可以通过GRANT与REVOKE语法便捷地控制不同用户对各Schema及其下属对象的权限，从而赋予业务更多的灵活性。

从便捷性和资源共享效率上考虑，推荐使用Schema进行业务隔离。建议系统管理员创建Schema和Database，再赋予相关用户对应的权限。

图 12-3 权限控制



## DATABASE

数据库Database是数据库对象的物理集合，不同Database之间资源完全隔离（除部分共享对象之外）。即Database是对业务的物理隔离，不同Database的之间的对象不能相互访问。比如在Database A中无法访问Database B中的对象。因此登录集群的时候必须显示指定要连接的Database。

## SCHEMA

数据库里面通过Schema把数据库对象进行逻辑划分，在Database中，通过Schema实现对数据库对象的逻辑隔离。

通过权限管理实现在同一个session下对不同Schema下对象的访问和操作权限。Schema下则是各种应用程序会接触到的对象，比如表，索引，数据类型，函数，操作符等。

同一个Schema下，不能存在同名的数据库对象；但是不同Schema下的对象名可以重复。

```
gaussdb=> CREATE SCHEMA myschema;
CREATE SCHEMA
gaussdb=> CREATE SCHEMA myschema_1;
CREATE SCHEMA
```

```
gaussdb=> CREATE TABLE myschema.t1(a int, b int) DISTRIBUTE BY HASH(b);
CREATE TABLE
gaussdb=> CREATE TABLE myschema.t1(a int, b int) DISTRIBUTE BY HASH(b);
ERROR: relation "t1" already exists
gaussdb=> CREATE TABLE myschema_1.t1(a int, b int) DISTRIBUTE BY HASH(b);
CREATE TABLE
```

Schema实现了对业务的逻辑划分，反过来这些业务对象也对Schema形成一种依赖关系，因此当Schema下存在对象时，删除Schema的时候会报错，并提示具体的依赖信息。

```
gaussdb=> DROP SCHEMA myschema_1;
ERROR: cannot drop schema myschema_1 because other objects depend on it
Detail: table myschema_1.t1 depends on schema myschema_1
Hint: Use DROP ... CASCADE to drop the dependent objects too.
```

当删除Schema的时候加上CASCADE选项，把Schema以及依赖此Schema的选项连带删除。

```
gaussdb=> DROP SCHEMA myschema_1 CASCADE;
NOTICE: drop cascades to table myschema_1.t1
gaussdb=> DROP SCHEMA
```

## USER/ROLE

用户或角色是数据库服务器(集群)全局范围内的权限控制系统，是集群业务的所有者和执行者，用于各种集群范围内所有的对象权限管理。因此角色不特定于某个单独的数据库，但角色登录集群的时候必须要显式指定登录的用户名，以保证当前连接执行的操作者的透明性。同时数据库也会通过权限管理限定用户的访问和操作权限。

用户是权限的最终体现者，所有的权限管理最终都体现在用户对数据库对象的操作权限是否被允许。

## 权限管理

DWS中的权限管理分为三种场景：

- 系统权限

系统权限又称为用户属性，包括SYSADMIN、CREATEDB、CREATEROLE、AUDITADMIN和LOGIN。

系统权限一般通过CREATE/ALTER ROLE语法来指定。其中，SYSADMIN权限可以通过GRANT/REVOKE ALL PRIVILEGE授予或撤销。但系统权限无法通过ROLE和USER的权限被继承，也无法授予PUBLIC。

- 用户权限

将一个角色或用户的权限授予一个或多个其他角色或用户。在这种情况下，每个角色或用户都可视为拥有一个或多个数据库权限的集合。

当声明了WITH ADMIN OPTION，被授权的用户可以将该权限再次授予其他角色或用户，以及撤销所有由该角色或用户继承到的权限。当授权的角色或用户发生变更或被撤销时，所有继承该角色或用户权限的用户拥有的权限都会随之发生变更。

数据库系统管理员可以给任何角色或用户授予/撤销任何权限。拥有CREATEROLE权限的角色可以赋予或者撤销任何非系统管理员角色的权限。

- 数据对象权限

将数据库对象（表和视图、指定字段、数据库、函数、模式等）的相关权限授予特定角色或用户。GRANT命令将数据库对象的特定权限授予一个或多个角色。这些权限会追加到已有的权限上。

## SCHEMA 隔离应用示例

示例一：

Schema的owner默认拥有该Schema下对象的所有权限，包括删除权限；Database的owner默认拥有该Database下对象的所有权限，包括删除权限。因此建议对Database和Schema的创建要做比较严格的控制，一般建议使用管理员创建Database和Schema，然后把相关的权限控制赋予业务用户。

**步骤1** dbadmin在数据库testdb下把创建Schema的权限赋予普通用户user\_1。

```
testdb=> GRANT CREATE ON DATABASE testdb to user_1;
GRANT
```

**步骤2** 切换到普通用户user\_1。

```
testdb=> SET SESSION AUTHORIZATION user_1 PASSWORD '*****';
SET
```

用户user\_1在数据库testdb下创建名为myschema\_2的Schema。

```
testdb=> CREATE SCHEMA myschema_2;
CREATE SCHEMA
```

**步骤3** 切换到管理员dbadmin。

```
testdb=> RESET SESSION AUTHORIZATION;
RESET
```

管理员dbadmin在模式myschema\_2下创建表t1。

```
testdb=> CREATE TABLE myschema_2.t1(a int, b int) DISTRIBUTE BY HASH(b);
CREATE TABLE
```

**步骤4** 切换到普通用户user\_1。

```
testdb=> SET SESSION AUTHORIZATION user_1 PASSWORD '*****';
SET
```

普通用户user\_1删除管理员dbadmin在模式myschema\_2下创建的表t1。

```
testdb=> drop table myschema_2.t1;
DROP TABLE
```

----结束

示例二：

因为Schema的逻辑隔离的功能，访问数据库对象实际上要通过Schema和具体对象的两层校验。

**步骤1** 把表myschema.t1的权限赋予用户user\_1。

```
gaussdb=> GRANT SELECT ON TABLE myschema.t1 TO user_1;
GRANT
```

**步骤2** 切换到用户user\_1。

```
SET SESSION AUTHORIZATION user_1 PASSWORD '*****';
SET
```

查询表myschema.t1。

```
gaussdb=> SELECT * FROM myschema.t1;
ERROR: permission denied for schema myschema
LINE 1: SELECT * FROM myschema.t1;
```

**步骤3** 切换到管理员dbadmin。

```
gaussdb=> RESET SESSION AUTHORIZATION;
RESET
```

把myschema.t1的权限赋予用户user\_1。

```
gaussdb=> GRANT USAGE ON SCHEMA myschema TO user_1;
GRANT
```

步骤4 切换到普通用户user\_1。

```
gaussdb=> SET SESSION AUTHORIZATION user_1 PASSWORD '*****1',
SET
```

查询表myschema.t1。

```
gaussdb=> SELECT * FROM myschema.t1;
a | b
---+---
(0 rows)
```

----结束

12.6.2 DWS 数据库账户密码到期了，如何修改？

数据库账户密码到期修改方式如下：

- 数据库管理员dbadmin的密码，可登录DWS控制台，选择集群所在行右边的“更多 > 重置密码”进行修改。

图 12-4 dbadmin 重置密码



- 数据库账户（普通用户和管理员dbadmin）的密码有效期，也可以在连接数据库后使用ALTER USER进行修改：  
ALTER USER *username* PASSWORD EXPIRATION 90;

12.6.3 如何给 DWS 指定用户赋予某张表的权限？

给指定用户赋予某张表的权限主要通过以下语法实现，本章主要介绍常见的几种场景，包括只读（SELECT）、插入（INSERT）、改写（UPDATE）和拥有所有权限。

语法格式

```
GRANT { { SELECT | INSERT | UPDATE | DELETE | TRUNCATE | REFERENCES | TRIGGER | ANALYZE |
ANALYSE } [, ...]
| ALL [PRIVILEGES] }
ON { [TABLE] table_name [, ...]
| ALL TABLES IN SCHEMA schema_name [, ...] }
TO { [GROUP] role_name | PUBLIC } [, ...]
[WITH GRANT OPTION];
```

场景介绍

假设当前有用户u1~u5，在系统中有对应的同名Schema u1~u5，各用户的权限管控如下：

- u2作为只读用户，需要表u1.t1的SELECT权限。
- u3作为插入用户，需要表u1.t1的INSERT权限。
- u4作为改写用户，需要表u1.t1的UPDATE权限。
- u5作为拥有所有权限的用户，需要表u1.t1的所有权限。

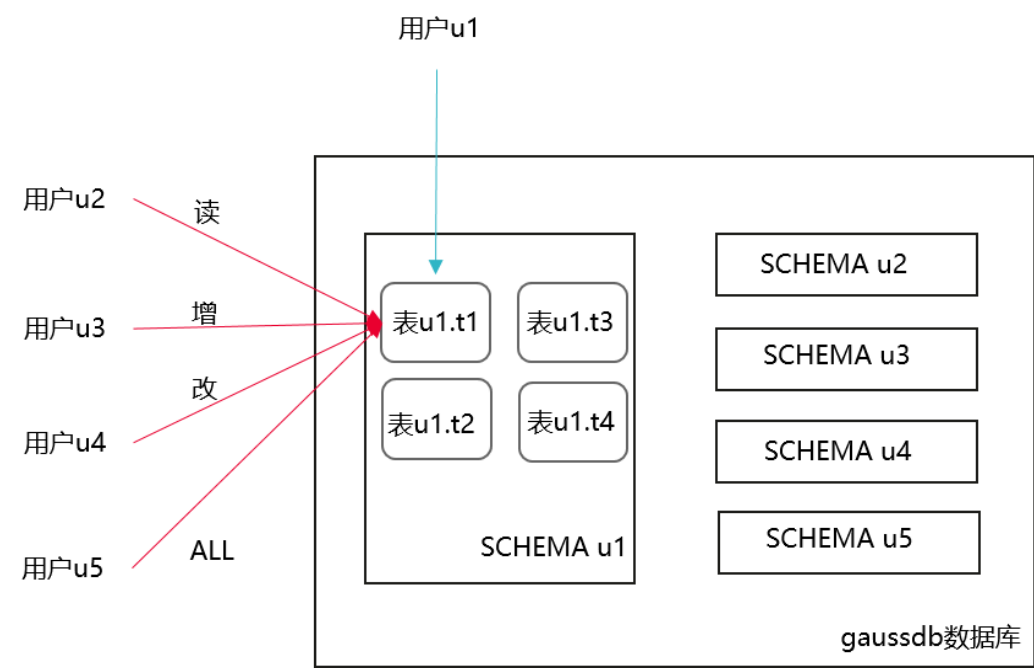


表 12-7 表 u1.t1 的表权限分类

| 用户名 | 用户类型 | Grant授权语句                                                                               | 查询 | 插入 | 修改 | 删除 |
|-----|------|-----------------------------------------------------------------------------------------|----|----|----|----|
| u1  | 所有者  | -                                                                                       | √  | √  | √  | √  |
| u2  | 只读用户 | GRANT SELECT ON u1.t1 TO u2;                                                            | √  | x  | x  | x  |
| u3  | 插入用户 | GRANT INSERT ON u1.t1 TO u3;                                                            | x  | √  | x  | x  |
| u4  | 改写用户 | GRANT SELECT,UPDATE ON u1.t1 TO u4;<br><b>须知</b><br>授予UPDATE权限必须同时授予SELECT权限，否则会出现信息泄露。 | √  | x  | √  | x  |

| 用户名 | 用户类型      | Grant授权语句                            | 查询 | 插入 | 修改 | 删除 |
|-----|-----------|--------------------------------------|----|----|----|----|
| u5  | 拥有所有权限的用户 | GRANT ALL PRIVILEGES ON u1.t1 TO u5; | √  | √  | √  | √  |

操作步骤

以下将演示不同权限的授权方法和验证过程。

- 步骤1

打开窗口1（即dbadmin连接会话窗口，后续不再提示），使用系统管理员dbadmin连接DWS数据库，创建用户u1~u5（系统默认会创建u1~u5的同名SCHEMA）。
- ```
CREATE USER u1 PASSWORD '{password}';
CREATE USER u2 PASSWORD '{password}';
CREATE USER u3 PASSWORD '{password}';
CREATE USER u4 PASSWORD '{password}';
CREATE USER u5 PASSWORD '{password}';
```
- 步骤2

在SCHEMA u1下创建表u1.t1。
- ```
CREATE TABLE u1.t1 (c1 int, c2 int);
```
- 步骤3

为表中插入两条数据。
- ```
INSERT INTO u1.t1 VALUES (1,2);
INSERT INTO u1.t1 VALUES (1,2);
```
- 步骤4

DWS中引入了SCHEMA层概念，如果有SCHEMA，需要先给用户赋予SCHEMA的使用权限。
- ```
GRANT USAGE ON SCHEMA u1 TO u2,u3,u4,u5;
```
- 步骤5

给只读用户u2赋予表u1.t1的查询权限。
- ```
GRANT SELECT ON u1.t1 TO u2;
```
- 步骤6

打开窗口2（即用户u2连接会话窗口，后续不再提示），使用用户u2连接DWS数据库，验证u2可以查询u1.t1表，但是不能写入和修改，此时u2为只读用户。
- ```
SELECT * FROM u1.t1;
INSERT INTO u1.t1 VALUES (1,20);
UPDATE u1.t1 SET c2 = 3 WHERE c1 =1;
```

```
gaussdb=> SELECT * FROM u1.t1;
 c1 | c2
----+----
 1 | 2
 1 | 2
(2 rows)

gaussdb=> INSERT INTO u1.t1 VALUES (1,20);
ERROR: permission denied for relation t1
gaussdb=> UPDATE u1.t1 SET c2 = 3 WHERE c1 =1;
ERROR: permission denied for relation t1
```

**步骤7** 切换回窗口1，分别给u3、u4、u5赋予对应的权限。

```
GRANT INSERT ON u1.t1 TO u3; --插入用户u3，可以插入数据
GRANT SELECT,UPDATE ON u1.t1 TO u4; --改写用户u4，可以修改表
GRANT ALL PRIVILEGES ON u1.t1 TO u5; --拥有所有权限的用户u5，可以对表进行查询、插入、改写和删除
```

**步骤8** 打开窗口3，使用用户u3连接DWS数据库，验证u3可以插入u1.t1，但是不能查询和修改，此时u3为插入用户。

```
SELECT * FROM u1.t1;
INSERT INTO u1.t1 VALUES (1,20);
UPDATE u1.t1 SET c2 = 3 WHERE c1 =1;
```

```
gaussdb=> SELECT * FROM u1.t1;
ERROR: permission denied for relation t1
gaussdb=> INSERT INTO u1.t1 VALUES (1,20);
INSERT 0 1
gaussdb=> UPDATE u1.t1 SET c2 = 3 WHERE c1 =1;
ERROR: permission denied for relation t1
```

**步骤9** 打开窗口4，使用用户u4连接DWS数据库，验证u4可以修改u1.t1，同时还可以查询，但是不能插入，此时u4为改写用户。

```
SELECT * FROM u1.t1;
INSERT INTO u1.t1 VALUES (1,20);
UPDATE u1.t1 SET c2 = 3 WHERE c1 =1;
```

```
gaussdb=> SELECT * FROM u1.t1;
 c1 | c2
----+----
 1 | 2
 1 | 2
 1 | 20
(3 rows)

gaussdb=> INSERT INTO u1.t1 VALUES (1,20);
ERROR: permission denied for relation t1
gaussdb=> UPDATE u1.t1 SET c2 = 3 WHERE c1 =1;
UPDATE 3
```

**步骤10** 打开窗口5，使用用户u5连接DWS数据库，验证u5可以查询、插入、修改和删除u1.t1，此时u5为拥有所有权限的用户。

```
SELECT * FROM u1.t1;
INSERT INTO u1.t1 VALUES (1,20);
UPDATE u1.t1 SET c2 = 3 WHERE c1 =1;
DELETE FROM u1.t1;
```

```
gaussdb=> SELECT * FROM u1.t1;
 c1 | c2
----+----
 1 | 3
 1 | 3
 1 | 3
(3 rows)

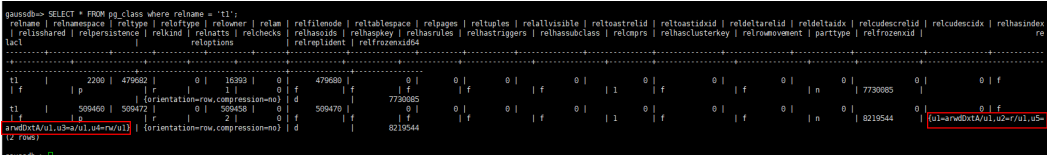
gaussdb=> INSERT INTO u1.t1 VALUES (1,20);
INSERT 0 1
gaussdb=> UPDATE u1.t1 SET c2 = 3 WHERE c1 =1;
UPDATE 4
gaussdb=> DELETE FROM u1.t1;
DELETE 4
```

**步骤11** 最后切换回窗口1，通过函数has\_table\_privilege分别查询每个用户的权限。

```
SELECT * FROM pg_class WHERE relname = 't1';
```

返回结果，查看relacl字段，该字段回显结果如下。*"rolename=xxxx/yyyy"* --表示rolename对该表有xxxx权限，且权限来自yyyy；

例如下图，与以上验证结果完全一致。



- u1=arwdDxtA/u1，表示u1为owner，拥有所有权限。
- u2=r/u1，表示u2拥有读权限。
- u3=a/u1，表示u3拥有插入权限。
- u4=rw/u1，表示u4拥有读和修改权限。
- u5=arwdDxtA/u1，表示u5拥有所有权限。

----结束

### 12.6.4 如何给 DWS 指定用户赋予某个 SCHEMA 的权限？

给某个用户授权某个SCHEMA的权限，包括三个场景（本章节针对SCHEMA层级权限，仅演示授权查询权限，如果需要其他权限，可以参考《常见问题》中“如何给指定用户赋予某张表的权限？”章节）：

- SCHEMA下当前某张表权限。
- SCHEMA下当前所有表的权限。
- SCHEMA下未来创建的表的权限。

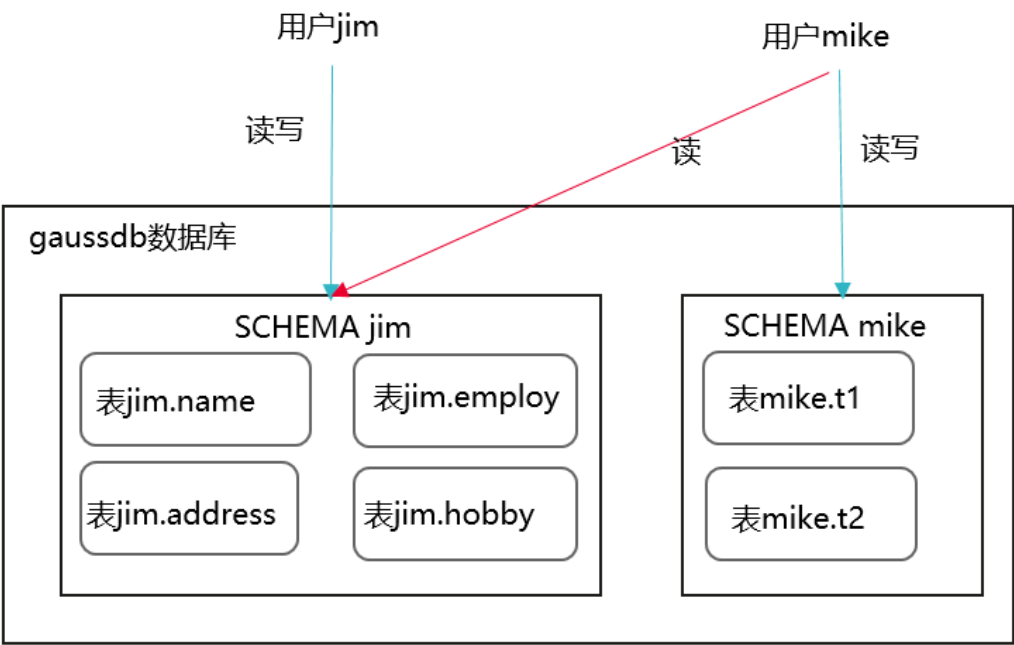


**注意**

不支持将外表的VACUUM，DROP，ALTER的权限赋予用户。

如图12-5所示，假设有两个用户jim和mike，对应的同名SCHEMA是jim和mike，用户mike需要访问SCHEMA jim的表（包括当前的某张表、所有表、未来创建的表）。

图 12-5 用户 mike 访问 SCHEMA jim 下的表



**步骤1** 打开窗口1（即dbadmin连接会话窗口，后续不再提示），使用系统管理员dbadmin连接DWS数据库，创建用户jim和mike（系统默认会创建jim和mike的同名SCHEMA）。

```
CREATE USER jim PASSWORD '{password}';
CREATE USER mike PASSWORD '{password}';
```

**步骤2** 在SCHEMA jim下创建表jim.name和jim.address。

```
CREATE TABLE jim.name (c1 int, c2 int);
CREATE TABLE jim.address (c1 int, c2 int);
```

**步骤3** 给用户mike赋予SCHEMA jim的访问权限。

```
GRANT USAGE ON SCHEMA jim TO mike;
```

**步骤4** （某张表权限）给用户mike赋予SCHEMA jim下某张表jim.name的查询权限。

```
GRANT SELECT ON jim.name TO mike;
```

**步骤5** 打开窗口2（即用户mike连接会话窗口，后续不再提示），使用用户mike连接DWS数据库，验证mike可以查询jim.name表，但是不能查询jim.address表。

```
SELECT * FROM jim.name;
SELECT * FROM jim.address;
```

| Result Information                                                                                                                       | SQL Details                                | Status           | Times |
|------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------|------------------|-------|
| [Statistics in some tables or columns(jim.name) are not collected.]<br>⌵                                                                 | SELECT * FROM jim.name                     | Run successfully | 268ms |
| "error_code":"DWS 50010001","error_msg":"sql error. STATE: 42501, message: permission denied to user 'mike' for relation 'jim.address'") | ERROR: SELECT<br>SELECT * FROM jim.address | Running failed   | 31ms  |

**步骤6** （所有表权限）切换回窗口1，使用dbadmin给用户mike赋予SCHEMA jim下所有表的查询权限。

```
GRANT SELECT ON ALL TABLES IN SCHEMA jim TO mike;
```

**步骤7** 切换回窗口2，再次验证mike可以查询所有表。

```
SELECT * FROM jim.name;
SELECT * FROM jim.address;
```

| Result Information                                                     | SQL Details               | Status           | Times |
|------------------------------------------------------------------------|---------------------------|------------------|-------|
| [Statistics in some tables or columns(jim.name) are not collected.]    | SELECT * FROM jim.name    | Run successfully | 13ms  |
| [Statistics in some tables or columns(jim.address) are not collected.] | SELECT * FROM jim.address | Run successfully | 18ms  |

**步骤8** 切换回窗口1，创建一张新的表jim.employ。

```
CREATE TABLE jim.employ (c1 int, c2 int);
```

**步骤9** 切换回窗口2，验证发现用户mike没有jim.employ的查询权限，说明mike虽然有SCHEMA jim下所有表的访问权限，但是对于赋权后新创建的表还是没有访问权限（即mike对SCHEMA jim未来的表权限是没有的）。

```
SELECT * FROM jim.employ;
```

| Result Information                                                                                                                       | SQL Details              | Status         | Times |
|------------------------------------------------------------------------------------------------------------------------------------------|--------------------------|----------------|-------|
| ["error_code":"DWS.S0010001","error_msg":"sql error. STATE: 42501, message: permission denied to user 'mike' for relation 'jim.employ'"] | SELECT * FROM jim.employ | Running failed | 17ms  |

**步骤10** （未来表权限）切换回窗口1，给用户mike赋予SCHEMA jim未来创建的表的访问权限，并创建一张新的表jim.hobby。

```
ALTER DEFAULT PRIVILEGES FOR ROLE jim IN SCHEMA jim GRANT SELECT ON TABLES TO mike;
CREATE TABLE jim.hobby (c1 int, c2 int);
```

**步骤11** 切换回窗口2，验证发现用户mike可以访问jim.hobby。但是对于之前的jim.employ还是没有访问权限（以上ALTER DEFAULT PRIVILEGES语句的授权范围只是未来创建的表，对于已经创建的表，还是要使用GRANT单张表重新授权），此时需要参考[步骤4](#)重新授权jim.employ即可。

```
SELECT * FROM jim.hobby;
```

| Result Information                                                   | SQL Details             | Status           | Times |
|----------------------------------------------------------------------|-------------------------|------------------|-------|
| [Statistics in some tables or columns(jim.hobby) are not collected.] | SELECT * FROM jim.hobby | Run successfully | 19ms  |

----结束

## 12.6.5 如何创建 DWS 数据库只读用户？

### 场景介绍

在业务开发场景中，数据库管理员通过SCHEMA来划分不同的业务，例如在金融行业中，负债业务属于SCHEMA s1，资产业务属于SCHEMA s2。

当前需要在数据库中创建一个只读用户user1，允许这个用户访问负债业务SCHEMA s1下所有的表（包括未来创建的新表），供日常读取，但是不允许进行数据插入、修改或删除。

### 实现原理

DWS有基于角色的用户管理，需要先创建一个的只读角色role1，再将对应的角色授权到实际的用户user1即可。

### 操作步骤

**步骤1** 使用系统管理员dbadmin连接DWS数据库。

**步骤2** 执行以下SQL语句创建角色role1。

```
CREATE ROLE role1 PASSWORD disable;
```

**步骤3** 执行以下SQL语句，为角色role1进行授权。

```
GRANT usage ON SCHEMA s1 TO role1; --赋予SCHEMA s1的访问权限;
GRANT select ON ALL TABLES IN SCHEMA s1 TO role1; --赋予SCHEMA s1下所有表的查询权限;
ALTER DEFAULT PRIVILEGES FOR USER tom IN SCHEMA s1 GRANT select ON TABLES TO role1; --赋予
SCHEMA s1未来创建的表的权限，其中tom为SCHEMA s1的owner
```

**步骤4** 执行以下SQL语句，将角色role1授权到实际用户user1。

```
GRANT role1 TO user1;
```

**步骤5** 如果访问的SCHEMA s1中包含有外表，还需要对只读用户user1进行使用外表的授权，执行以下命令。

```
ALTER USER user1 USEFT;
```

否则以只读用户查询外表时会报以下错误：“ERROR: permission denied to select from foreign table in security mode”。

**步骤6** 使用只读用户user1进行SCHEMA s1下所有表数据的日常读取。

----结束

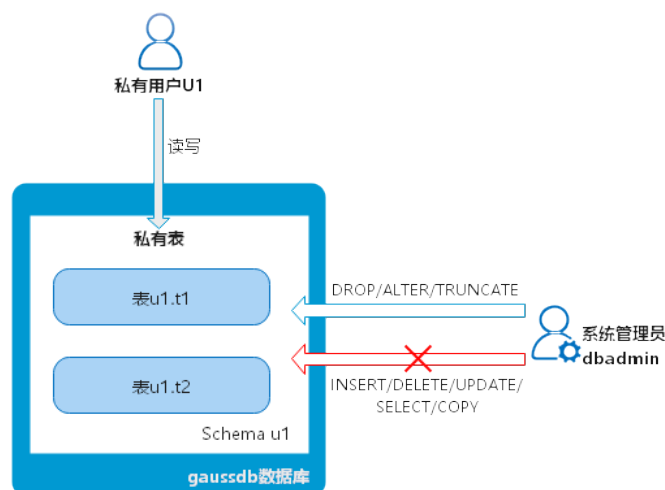
## 12.6.6 如何创建 DWS 数据库私有用户和私有表？

### 场景介绍

在业务场景中，普通用户创建的表，系统管理员dbadmin默认是有访问权限，并未完全私有。而在开启下的业务场景中，管理员dbadmin对普通用户的表没有访问权限，同时也没有控制权限（DROP、ALTER、TRUNCATE）。

若当前业务中需要创建一个私有用户和私有表（即私有用户创建的表），私有表只有私有用户本身可以访问，系统管理员dbadmin和其他普通用户均无权访问（进行INSERT、DELETE、UPDATE、SELECT、COPY操作），但同时需要满足：未经私有用户授权的情况下，可允许系统管理员dbadmin执行DROP/ALTER/TRUNCATE操作。那么可以通过创建INDEPENDENT属性的用户（私有用户）实现。

图 12-6 私有用户



### 实现原理

通过创建INDEPENDENT属性的用户来实现。

**INDEPENDENT | NOINDEPENDENT**: 定义私有、独立的角色。具有INDEPENDENT属性的角色，管理员对其进行的控制、访问的权限被分离，具体规则如下：

- 未经INDEPENDENT角色授权，管理员无权对其表对象进行增、删、查、改、拷贝、授权操作。
- 未经INDEPENDENT角色授权，管理员无权修改INDEPENDENT角色的继承关系。
- 管理员无权修改INDEPENDENT角色的表对象的属主。
- 管理员无权修改INDEPENDENT角色的数据库口令，INDEPENDENT角色需管理好自身口令，口令丢失无法重置。
- 管理员属性用户不允许定义修改为INDEPENDENT属性。

## 操作步骤

**步骤1** 使用系统管理员dbadmin连接DWS数据库。

**步骤2** 执行以下SQL语句创建私有用户u1。

```
CREATE USER u1 WITH INDEPENDENT IDENTIFIED BY 'password';
```

**步骤3** 切换到u1用户，创建测试表test，并插入数据。

```
CREATE TABLE test (id INT, name VARCHAR(20));
INSERT INTO test VALUES (1, 'joe');
INSERT INTO test VALUES (2, 'jim');
```

**步骤4** 切换到dbadmin用户，并执行以下SQL语句验证dbadmin用户是否可以访问私有用户u1创建的私有表test。

```
SELECT * FROM u1.test;
```

查询结果提示dbadmin无权访问，即私有用户和私有表创建成功。

```
gaussdb=> SELECT * FROM u1.test;
ERROR: SELECT permission denied to user "dbadmin" for relation "u1.test"
```

**步骤5** dbadmin用户执行DROP语句，可成功删除test表。

```
DROP TABLE u1.test;
```

```
gaussdb=> drop table u1.test;
DROP TABLE
```

----结束

## 12.6.7 DWS 如何 REVOKE 某用户的 connect on database 权限？

### 业务场景

某业务中需要撤销指定用户u1连接某数据库的权限，在执行**REVOKE CONNECT ON DATABASE gaussdb FROM u1**命令成功后，使用u1还能继续连接数据库，撤销权限没有生效。

### 原因分析

若直接使用**REVOKE CONNECT ON DATABASE gaussdb from u1**命令撤销u1用户的权限不会生效，因为数据库的CONNECT权限授予了PUBLIC，需指定关键字PUBLIC实现。

- DWS提供了一个隐式定义的拥有所有角色的组PUBLIC，所有创建的用户和角色默认拥有PUBLIC所拥有的权限。要撤销或重新授予用户和角色对PUBLIC的权限，可通过在GRANT和REVOKE指定关键字PUBLIC实现。
- DWS会将某些类型的对象上的权限授予PUBLIC。默认情况下，对表、表字段、序列、外部数据源、外部服务器、模式或表空间对象的权限不会授予PUBLIC。以下这些对象的权限会授予PUBLIC：
  - 数据库的CONNECT权限。
  - CREATE TEMP TABLE权限。
  - 函数的EXECUTE特权。
  - 语言和数据类型（包括域）的USAGE特权。
- 对象拥有者可以撤销默认授予PUBLIC的权限并专门授予权限给其他用户。

## 操作示例

撤销用户u1访问数据库gaussdb的权限：

**步骤1** 连接DWS集群的数据库gaussdb。

```
gsql -d gaussdb -p 8000 -h 192.168.x.xx -U dbadmin -W password -r
gaussdb=>
```

**步骤2** 创建用户u1。

```
gaussdb=> CREATE USER u1 IDENTIFIED BY 'xxxxxxx';
```

**步骤3** 确认用户u1可正常访问数据库gaussdb。

```
gsql -d gaussdb -p 8000 -h 192.168.x.xx -U u1 -W password -r
gaussdb=>
```

**步骤4** 使用管理员用户dbadmin连接数据库gaussdb，执行REVOKE命令撤销public的connect on database权限。

```
gsql -d gaussdb -h 192.168.x.xx -U dbadmin -p 8000 -r
gaussdb=> REVOKE CONNECT ON DATABASE gaussdb FROM public;
REVOKE
```

**步骤5** 验证结果，使用u1连接数据库，显示如下信息表示用户u1的connect on database权限已成功撤销。

```
gsql -d gaussdb -p 8000 -h 192.168.x.xx -U u1 -W password -r
gsq: FATAL: permission denied for database "gaussdb"
DETAIL: User does not have CONNECT privilege.
```

----结束

## 12.6.8 如何查看 DWS 某个用户有哪些表的权限？

**场景一：查看用户有哪些表的权限**，可使用information\_schema.table\_privileges系统表查看。例如

```
SELECT * FROM information_schema.table_privileges WHERE GRANTEE='user_name';
```

```
gaussdb=> SELECT * FROM information_schema.table_privileges WHERE GRANTEE='u2';
grantor | grantee | table_catalog | table_schema | table_name | privilege_type | is_grantable | with_hierarchy

u2 | u2 | gaussdb | u2 | t2 | INSERT | YES | NO
u2 | u2 | gaussdb | u2 | t2 | SELECT | YES | YES
u2 | u2 | gaussdb | u2 | t2 | UPDATE | YES | NO
u2 | u2 | gaussdb | u2 | t2 | DELETE | YES | NO
u2 | u2 | gaussdb | u2 | t2 | TRUNCATE | YES | NO
u2 | u2 | gaussdb | u2 | t2 | REFERENCES | YES | NO
u2 | u2 | gaussdb | u2 | t2 | TRIGGER | YES | NO
u2 | u2 | gaussdb | u2 | t2 | ANALYZE | YES | NO
u2 | u2 | gaussdb | u2 | t2 | VACUUM | YES | NO
u2 | u2 | gaussdb | u2 | t2 | ALTER | YES | NO
u2 | u2 | gaussdb | u2 | t2 | DROP | YES | NO
u1 | u2 | gaussdb | u1 | t1 | SELECT | NO | YES
(12 rows)
```

表 12-8 table\_privileges 字段

| 字段             | 数据类型           | 描述                                                                                          |
|----------------|----------------|---------------------------------------------------------------------------------------------|
| grantor        | sql_identifier | 赋权用户。                                                                                       |
| grantee        | sql_identifier | 被赋权用户。                                                                                      |
| table_catalog  | sql_identifier | 包含该表的数据库名。                                                                                  |
| table_schema   | sql_identifier | 包含该表的模式名。                                                                                   |
| table_name     | sql_identifier | 表名。                                                                                         |
| privilege_type | character_data | 被赋予的权限类型：SELECT，INSERT，UPDATE，DELETE，TRUNCATE，REFERENCES，ANALYZE，VACUUM，ALTER，DROP或TRIGGER。 |
| is_grantable   | yes_or_no      | 权限是否可赋予其他用户，YES表示可授予，NO表示不可授予。                                                              |
| with_hierarchy | yes_or_no      | 是否允许在表继承层级上的特定操作。当特定操作为SELECT时显示YES，否则为NO。                                                  |

如上图所示，表示用户u2拥有Schema u2下的t2的所有权限和Schema u1下的t1的SELECT权限。

需注意，在查询有哪些表权限时，information\_schema.table\_privileges只能查到当前用户被直接授予的权限，而函数has\_table\_privilege()除了能查询被直接授予的权限外还能查到间接的权限（即GRANT role to user获取的）。例如：

```
CREATE TABLE t1 (c1 int);
CREATE USER u1 password '*****';
CREATE USER u2 password '*****';
GRANT dbadmin to u2; // 间接通过角色成员关系赋予权限
GRANT SELECT on t1 to u1; // 直接授予权限

SET ROLE u1 password '*****';
SELECT * FROM public.t1; 直接授权可以访问表
c1

(0 rows)

SET ROLE u2 password '*****';
SELECT * FROM public.t1; //间接授权可以访问表
c1

(0 rows)

RESET role; //切换回到dbadmin
SELECT * FROM information_schema.table_privileges WHERE table_name = 't1'; //
information_schema.table_privileges仅能看到直接授权
 grantor | grantee | table_catalog | table_schema | table_name | privilege_type | is_grantable |
with_hierarchy
-----+-----+-----+-----+-----+-----+-----+
dbadmin | u1 | gaussdb | public | t1 | SELECT | NO | YES
(1 rows)

SELECT has_table_privilege('u2', 'public.t1', 'select'); // has_table_privilege还可以看到间接授权
has_table_privilege

```

```
t
(1 row)
```

场景二：查看用户是否有某张表的权限，可以通过以下方法。

步骤1 执行以下语句查询pg\_class系统表。

```
SELECT * FROM pg_class WHERE relname = 'tablename';
```

查看relacl字段，该字段回显结果如下，权限参数参见表12-9。

- "rolename=xxxx/yyyy" --表示rolename对该表有xxxx权限，且权限来自yyyy；
- "=xxxx/yyyy" -- 表示public对该表有xxxx权限，且权限来自yyyy。

例如下图：

joe=arwdDxtA，表示joe用户有所有权限（ALL PRIVILEGES）。

leo=arw/joe，表示leo用户拥有读、写、改权限，该权限来自joe授权。

```
postgres=# select * from pg_class where relname = 't';
 relname | relnamespace | reltype | reloftype | relowner | relam | relfilename | reltablespace | relpages | reltuples | relallvisible | reltoastrelid | reltoastidid | reldeltarelid | reldeltaid | relcudescrid | relcudescid | relhas
 index | relisshared | relpersist | relkind | relchecks | relhasoids | relhasrules | relhasrules | relhasrules | relhasrules | relhasrules | relhasrules | relhasrules | relhasrules | relhasrules | relhasrules | relhasrules | relhasrules
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
 t | 24587 | 98323 | 0 | 24585 | 0 | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
 (1 row)
```

表 12-9 权限的参数说明

| 参数       | 参数说明                |
|----------|---------------------|
| r        | SELECT（读）           |
| w        | UPDATE（写）           |
| a        | INSERT（插入）          |
| d        | DELETE              |
| D        | TRUNCATE            |
| x        | REFERENCES          |
| t        | TRIGGER             |
| X        | EXECUTE             |
| U        | USAGE               |
| C        | CREATE              |
| c        | CONNECT             |
| T        | TEMPORARY           |
| A        | ANALYZE ANALYSE     |
| arwdDxtA | ALL PRIVILEGES（用于表） |
| *        | 给前面权限的授权选项          |

步骤2 如果要查某用户对某张表是否有某种权限，也可以通过访问权限查询函数has\_table\_privilege进行查询。

```
SELECT * FROM has_table_privilege('用户名','表名','select');
```

例如，查询joe对表t1是否有查询权限。

```
SELECT * FROM has_table_privilege('joe','t1','select');
```

```
gaussdb=> select * from has_table_privilege('joe','t1','select');
has_table_privilege

t
(1 row)
```

----结束

## 12.6.9 DWS 数据库中的 Ruby 是什么用户？

在执行SELECT \* FROM pg\_user语句查看当前系统的用户时，看到Ruby用户且拥有很多权限。

Ruby用户为官方运维使用账户，DWS数据库创建后，默认生成Ruby账户，不涉及安全风险，请放心使用。

```
gaussdb> SELECT * FROM pg_user;
username | usersid | usecreatedb | usesuper | usecatupd | usecrepl | passwd | valbegin | valuntil | respool | parent | spacelimit | useconfig | nodegroup | temppacelimit | spillspa
celimit

dbadmin | 16384 | f | f | f | f | ***** | | | default_pool | 0 | | | | |
Ruby | 16385 | f | t | t | t | ***** | | | default_pool | 0 | | | | |
user_1 | 24594 | f | f | f | f | ***** | | | default_pool | 0 | | | | |
u1 | 24595 | f | f | f | f | ***** | | | default_pool | 0 | | | | |
u2 | 24597 | f | f | f | f | ***** | | | default_pool | 0 | | | | |
(5 rows)
```

## 12.7 数据库性能

### 12.7.1 为什么 DWS 使用一段时间后执行 SQL 很慢？

数据库在使用一段时间后，随着业务的增加使得表数据增加，或者对表数据经常进行增、删、改之后，引发数据膨胀和统计信息不准造成性能下降。

建议对于频繁增、删、改的表，定期执行vacuum full和analyze操作。操作步骤如下：

**步骤1** 默认每30000条数据收集100条做统计信息，当数据量大的时候，发现SQL时快时慢，一般是执行计划发生了变化，统计信息的需要调整采样率。set default\_statistics\_target可以提高采样率，对优化器生成最优计划有所帮助。

```
gaussdb=> set default_statistics_target=-2;
SET
```

**步骤2** 重新执行analyze。详细信息请参见。

```
gaussdb=> ANALYZE customer_t1;
ANALYZE
```

----结束

 说明

若用户想要知道是否是磁盘碎片的问题影响了数据库的性能，可以使用以下函数进行查询：

```
SELECT * FROM pgxc_get_stat_dirty_tables(30,100000);
```

## 12.7.2 为什么 DWS 的性能在极端场景下并未比单机数据库好

DWS中由于MPP架构的限制导致少部分PG系统方法、函数无法下推到DN节点来执行，仅能在CN端出现性能瓶颈。

### 原理解释：

- 一个操作能够并行执行是有条件的，需要逻辑上能够并行，比如做汇总（SUM），可以各个节点（DN）并行汇总后，最后的汇总一定是不能并行，要在某一个节点（CN）上执行，由于大部分的汇总工作已经在DN节点完成，CN端的工作是比较轻量的。
- 某些场景必须要集中执行，比如事务号，必须要保证全局唯一，该任务在系统里是通过GTM来实现的，因此，GTM也是全局唯一的组件（主备）。所有需要全局唯一的任务都是通过DWS中的GTM来完成，只是在设计上尽量避免阻塞在GTM上，因此GTM并没有太多瓶颈，而且有些场景下还可以GTM-Free和GTM-Lite。
- 从传统单机数据库的应用开发模式到并行数据库，为确保获得更好的性能，可能需要对业务进行少量改动，尤其是传统Oracle的存储过程互相嵌套的开发模式，如果要保证高性能，需要进行业务修改及对应的适配。

### 解决方案：

- 或者，请联系技术人员进行业务适配的修改调优。

## 12.7.3 DWS 业务读写阻塞，如何查看某个时间段的 SQL 执行记录？

当您的数据库集群出现读写阻塞时，可通过TopSQL功能查看某个时间段所执行的SQL语句，支持查看当前CN或者所有CN的sql语句。

TopSQL功能包括查看实时SQL语句和历史SQL语句：

- 实时SQL语句查询请参见：。
- 历史SQL语句查询请参见：。

## 12.7.4 DWS 的 CPU 资源隔离管控介绍

### CPU 资源管控概述

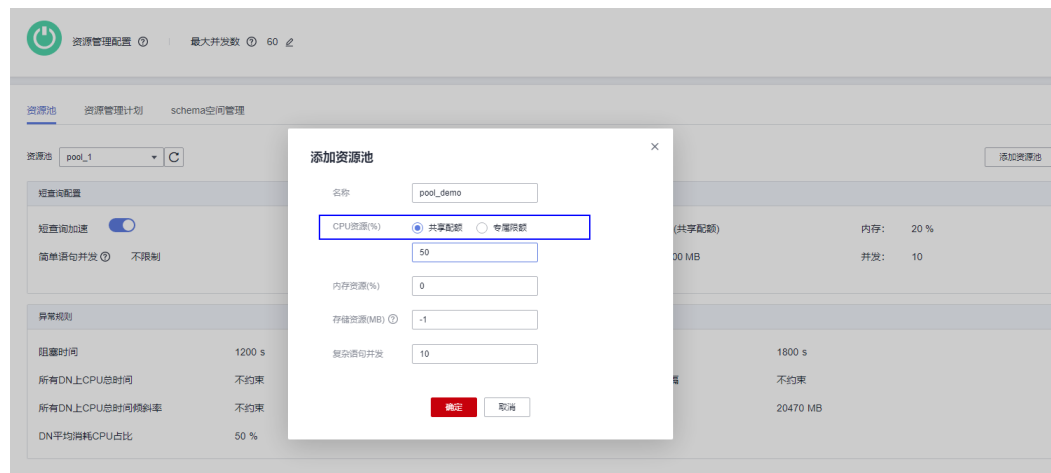
在不同的业务场景中，对数据库的系统资源（CPU资源、内存资源、IO资源和存储资源）进行合理的分配，保证执行查询时有充足的系统资源，确保查询性能，可以维持业务稳定性。

DWS的资源管理功能支持用户根据自身业务将资源按需划分成不同的资源池，不同资源池之间资源互相隔离。再通过关联数据库用户将其关联至不同的资源池，用户SQL查询时将根据“用户-资源池”的关联关系将查询转至资源池中执行。通过指定资源池上可并行运行的查询数、单查询内存上限以及资源池可使用的内存和CPU资源，从而实现不同业务之间的资源限制和隔离，满足数据库混合负载需求。

DWS主要利用cgroup（control group，控制组）进行CPU资源管控，涉及CPU、cpuacct、cpuset子系统。CPU共享配额管控基于CPU子系统的cpu.shares实现，该配

置方法的好处是：OS CPU没有占满的情况下，不触发CPU管控；CPU专属限额管控基于cpuset实现；cpuacct子系统主要用于CPU资源使用的监控。

在DWS控制台使用资源管理配置功能创建资源池时，根据业务需要对CPU资源管理的“共享配额”和“专属配额”进行配置。



## 共享配额

**共享配额：**关联在当前资源池的用户在执行作业时可以使用的CPU时间比例。

共享配额有两层含义：

- 共享：CPU是所有控制组共享的，其他控制组能够使用空闲的CPU资源。
- 配额：业务繁忙、CPU满负载情况下，控制组之间按照配额比例进行CPU抢占。

共享配额基于cpu.shares实现，只有在CPU满负载情况下生效，因此在CPU空闲情况下并不能保证控制组能够抢占到配额比例的CPU资源。CPU空闲并不能理解为没有CPU资源争抢，控制组内任务可以任意使用CPU。虽然CPU平均使用率可能不高，但是某个特定时刻还是可能存在CPU资源争抢的。

例如：10个CPU上运行10个作业，每个CPU上运行1个作业，这种情况下各作业在任意时刻请求CPU都可以瞬间得到响应，作业之间没有任何CPU资源的争抢；但假如10个CPU上运行20个作业，因为作业不会一直占用CPU，在某些时间可能等待IO、网络等，因此CPU使用率可能并不高，此时CPU资源看似空闲，但是在某个时刻可能出现2~N作业同时请求一个CPU的情况出现，即会导致CPU资源争抢，影响作业性能。

## 专属限额

**专属限额：**限定资源池中数据库用户在执行作业时可使用的最大CPU核数占总核数的百分比。

专属限额有两层含义：

- 专属：CPU是某个控制组专属的，其他已设置限额的控制组不能使用空闲的CPU资源。
- 限额：只能使用限额配置的CPU资源，其他控制组空闲的CPU资源，也不能抢占。

专属限额基于cpuset.cpu实现，通过合理的限额设置可以实现控制组之间CPU资源的绝对隔离，各控制组间任务互不影响。但因为CPU的绝对隔离，因此在控制组空闲时就

会导致CPU资源的极大浪费，因此限额设置不能太大。从作业性能来看并不是限额越大越好。

例如：10个作业运行在10个CPU上，CPU平均使用率5%左右；10个作业运行在5个CPU上，CPU平均使用率10%左右。通过上面共享配额的分析可知：虽然10个作业运行在5个CPU上CPU使用率很低，看似空闲，但是相对10个作业运行在10个CPU上还是存在某种程度的CPU资源争抢，因此10个作业运行在10个CPU上性能要好于运行在5个CPU上。但也不是越多越好，10个作业运行在20个CPU上，在任意一个时刻，总会至少10个CPU是空闲的，因此理论上10个作业运行在20个CPU上并不会比运行在10个CPU上性能更好。对于并发为N的控制组，分配cpus小于N的情况下，CPU越多作业性能越好；但是当分配CPUS大于N的情况下，性能就不会有任何提升了。

## CPU 资源管理应用场景

CPU共享配额和专属限额的管控方式各有优劣，共享配额能够实现CPU资源的充分利用，但是各控制组之间资源隔离不彻底，可能影响查询性能；专属限额的管控方式可以实现CPU资源的绝对隔离，但是在CPU资源空闲时会造成CPU资源的浪费。相对专属限额来说，共享配额拥有更高的CPU使用率和更高的整体作业吞吐量；相对共享配额来说，专属限额CPU隔离彻底，更满足性能敏感用户的使用诉求。

数据库系统中运行多种类型作业出现CPU争抢时，可根据不同场景，选择不同的CPU资源管控方式：

- 场景一：实现CPU资源的充分利用，不关注单一类型作业的性能，主要关注CPU整体吞吐量。  
应用建议：不建议进行用户之间的CPU隔离管控，无论哪一种CPU管控都会对CPU整体使用率产生影响。
- 场景二：允许一定程度的CPU资源争抢和性能损耗，在CPU空闲情况下实现CPU资源充分利用，在CPU满负载情况下需要各业务类型按比例使用CPU。  
应用建议：可以采用基于cpu.shares的共享配额管控方式，在实现满负载CPU隔离管控前提下，尽量提高CPU整体使用率。
- 场景三：部分作业对性能敏感，允许CPU资源的浪费。  
应用建议：可以采用基于cpuset.cpu的专属限额管控方式，实现不同类型作业之间的CPU绝对隔离。

## 12.7.5 为什么 DWS 普通用户比 dbadmin 用户执行的慢？

DWS在使用过程中会出现普通用户比dbadmin用户执行慢的场景主要有以下三种：

### 场景一：普通用户受资源管理的管控

普通用户在排队：waiting in queue/waiting in global queue/waiting in ccn queue.

1. 普通用户主要在waiting in queue/waiting in global queue时。

当前的活跃语句数超过max\_active\_statements限制导致的普通用户排队，由于管理员用户不受管控所以无需排队。可通过在DWS控制台修改max\_active\_statements参数值处理：

- a. 登录DWS控制台。
- b. 在左侧导航树，单击“集群 > 集群列表”。
- c. 在集群列表中找到所需要的集群，单击集群名称，进入“集群详情”页面。
- d. 单击“参数修改”页签，搜索“max\_active\_statements”参数，修改其参数值，单击“保存”，确认无误后再单击“保存”。

### 场景二：执行计划中的 or 条件对普通用户执行语句逐一判断耗时

执行计划中的or条件里有权限相关的判断，此场景多发生在使用系统视图时。例如以下SQL：

```
SELECT distinct(dtp.table_name),
 ta.table_catalog,
 ta.table_schema,
 ta.table_name,
 ta.table_type
from information_schema.tables ta left outer join DBA_TAB_PARTITIONS dtp
on (dtp.schema = ta.table_schema and dtp.table_name = ta.table_name)
where ta.table_schema = 'public';
```

一部分执行计划如下:

[illegible]

可以看到系统视图中的权限判断中多用or条件判断:

```
pg_has_role(c.reloader, 'USAGE'::text) OR has_table_privilege(c.oid, 'SELECT, INSERT, UPDATE, DELETE,
TRUNCATE, REFERENCES, TRIGGER'::text) OR has_any_column_privilege(c.oid, 'SELECT, INSERT, UPDATE,
REFERENCES'::text)
```

由于dbadmin用户pg has role总能返回true，因此or之后的条件无需继续判断；

而普通用户的or条件需要逐一判断，如果数据库中表个数比较多，最终会导致普通用户比dbadmin需要更长的执行时间。

这种场景如果输出结果集很少，可以考虑尝试设置`set enable_hashjoin = off; set enable_seqscan = off;`走index + nestloop的计划。

### 场景三：普通用户和管理员用户所分配资源池有差异

通过执行如下查询命令，查看用户所对应的资源池是否相同，如果不同，可在界面查看两个资源池上所分配的租户资源是否有差别。

```
SELECT * FROM pg_user;
```

### 12.7.6 DWS 中单表查询性能与哪些因素有关？

DWS采用Shared-nothing架构，数据是被分布式存储，因此分布键设计、单表存储数据量、分区数量都会影响单表的整体查询性能。

## 1. 分布键设计

DWS默认会选择主键的第一列作为分布键。如果同时设置主键和分布键，则主键必须包含分布键。分布键决定了数据在各个分区之间的分布情况，如果分布键很好地分布在各个分区中，则可以使查询性能变得更好。

分布列选择不当，在数据导入后有可能出现数据分布倾斜，进而导致某些磁盘的使用明显高于其他磁盘，极端情况下会导致集群只读。合理的选择分布键，对表查询的性能至关重要。此外，合适的分布键还可以使数据的索引更快地创建和维护。

## 2. 单表存储数据量

单表存储的数据量越大，查询性能就越差。当表中的数据量很大时，则需要考虑将数据进行分区存储。普通表若要转成分区表，需要新建分区表，然后把普通表中的数据导入到新建的分区表中。因此在初始设计表时，请根据业务提前规划是否使用分区表。

对表进行分区，一般需要遵循以下原则：

- 使用具有明显区间性的字段进行分区，比如日期、区域等字段。
- 分区名称应当体现分区的数据特征。比如关键字+区间特征。
- 将分区上边界的分区值定义为MAXVALUE，以防止可能出现的数据溢出。

## 3. 分区数量

利用分区，可以将表和索引划分为一些更小、更易管理的单元。大幅减少搜索空间，从而提升访问性能。

使用分区数量会影响查询的性能。如果分区数量太小，则可能会使查询性能下降。

DWS支持范围分区（Range Partitioning）和列表分区（List Partitioning）功能，即根据表的一列或者多列，将要插入表的记录分为若干个范围（这些范围在不同的分区里没有重叠），然后为每个范围创建一个分区，用来存储相应的数据。其中，列表分区（List Partitioning）仅8.1.3及以上集群版本支持。

因此，在设计数据仓库时，需要考虑这些因素并进行实验来确定最佳设计方案。

## 12.7.7 如何优化包含多个 CASE WHEN 条件的 SQL 查询？

在业务查询中，CASE WHEN语句常用来进行条件判断，但如果在SQL查询中存在大量冗余的CASE WHEN，例如：

```
SELECT
 SUM(CASE WHEN a > 1 THEN 1 ELSE 0 END) AS a1,
 SUM(CASE WHEN a > 2 THEN 1 ELSE 0 END) AS a2,
 ...
FROM test
WHERE dt = '20241225';
```

该语句冗长，执行时每个分支的CASE WHEN均需执行，导致查询时间成倍增加，影响查询性能。

DWS提供以下优化策略来解决此类问题。

### 使用临时结果集或者子查询

将复杂的CASE WHEN计算部分提取出来，放到一个临时的结果集中或者子查询中。这样可以减少在主查询中的重复计算逻辑。

例如，先创建一个子查询来计算中间结果：

```
SELECT
 sub.a1,
 sub.a2
FROM (
 SELECT
 sum(case when a > 1 then 1 else 0 end) AS a1,
 sum(case when a > 2 then 1 else 0 end) AS a2
 FROM test
 WHERE dt = '20241225'
) sub;
```

```
SELECT
 SUM(case_when_a1) as a1,
 SUM(case_when_a2) as a2,
 ...
FROM (
 SELECT
 CASE WHEN a > 1 THEN 1 ELSE 0 END AS case_when_a1,
 CASE WHEN a > 2 THEN 1 ELSE 0 END AS case_when_a2,
 ...
 FROM test
 WHERE dt = '20241225'
) AS subquery;
```

## 使用自定义函数

将CASE WHEN的逻辑封装成一个函数。这样在查询中只需要调用该函数，而不是多次编写相同的CASE WHEN逻辑。

例如，创建一个简单的函数count\_a\_gt\_value。

```
CREATE OR REPLACE FUNCTION count_a_gt_value(val INT)
RETURNS INT AS $$
DECLARE
 result INT;
BEGIN
 SELECT sum(CASE WHEN a > val THEN 1 ELSE 0 END)
 INTO result
 FROM test
 WHERE dt = '20241225';
 RETURN result;
END;
$$ LANGUAGE plpgsql;
```

再使用自定义函数count\_a\_gt\_value进行查询。

```
SELECT
 count_a_gt_value(1) AS a1,
 count_a_gt_value(2) AS a2
FROM test;
```

## 12.8 备份恢复

### 12.8.1 为什么 DWS 自动快照创建很慢，很长时间都没有创建好？

自动快照备份很慢，可能是由于所需备份的数据量较大。自动快照是增量备份，备份频率是可以设置的，如果一周只备份一次，当增量数据量较大时，就会出现备份慢的情况。建议您适当地增加备份频率。

其中快照备份和恢复速率如下所示（此速率为实验室测试环境下数据，介质为SSD本地盘，仅供用户参考。在实际使用中，由于磁盘、网络、带宽等因素可能会产生一定的差异）：

- 备份速率：200 MB/s/DN
- 恢复速率：125 MB/s/DN

### 12.8.2 DWS 快照是否与 EVS 快照功能相同？

数据仓库服务的快照功能与云硬盘快照不同。

数据仓库服务的快照针对DWS集群的所有配置数据和业务数据，用于还原创建快照时的集群数据；云硬盘快照是针对于云服务器数据盘或者系统盘的某个时段的业务数据。

## DWS 快照

数据仓库服务快照是对DWS集群在某一时间点的一次全量数据和增量数据的备份，记录了当前数据库的数据以及集群的相关信息，其中包括节点数量、节点规格和数据库管理员用户名称等。快照创建方式包括手动创建快照和自动创建快照。

从快照恢复到集群时，DWS会根据快照记录的集群信息来创建新集群，然后从快照记录的数据中还原数据库信息。

请参见《数据仓库服务用户指南》中的“备份与恢复DWS集群”章节。

## EVS 快照

云硬盘快照指的是云硬盘数据在某个时刻的完整拷贝或镜像，是一种重要的数据容灾手段，当数据丢失时，可通过快照将数据完整的恢复到快照时间点。

您可以创建快照，从而快速保存指定时刻云硬盘的数据。同时，您还可以通过快照创建新的云硬盘，这样云硬盘在初始状态就具有快照中的数据。

创建快照可以快速保存指定时刻云硬盘的数据，实现数据容灾：

- 当数据丢失时，可通过快照将数据完整的恢复到快照时间点。
- 通过快照创建新的云硬盘，新的云硬盘具有快照中的数据。

请参见《云硬盘产品介绍》中的“云硬盘快照（公测）”章节。