

数据库和应用迁移 UGO(UGO) 25.1.0

常见问题

文档版本 01
发布日期 2025-01-30



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 产品咨询	1
1.1 资源冻结/释放/删除	1
1.2 源数据库忽略迁移的 schema 有哪些?	2
1.3 迁移项目中, 语法转换与迁移验证的区别?	3
1.4 Oracle 数据库迁移到目标库, 数据库结构有什么变化?	3
1.5 部分功能无法正常使用, 是为什么?	3
1.6 迁移到目标库后, 生成的 dsc_ora_ext 这个 Schema 的作用是什么?	3
1.7 如何解决 Oracle 系统参数配置太低导致采集失败或速度慢的问题?	4
1.8 如何解决收集失败并提示“SNAPSHOT TOO OLD”的问题?	4
1.9 自建 GaussDB 内核版本号与 UGO 界面显示 GaussDB 版本号对应关系	4
1.10 什么是 VPCEP 资源配额?	5
2 数据库连接	7
2.1 创建数据库评估项目, 无法连接到 Oracle 源库	7
2.2 通过 sys 用户连接源库失败	7
2.3 GaussDB 如何创建不同兼容类型数据库	7
2.4 测试连接到数据库报错	9
2.5 SQL Server 测试连接失败, 提示错误信息 ErrorCode=0, SQLState=08S01	10
3 评估项目	12
3.1 连接方式该如何选择?	12
3.2 采集源库的方法	12
3.3 数据库权限检查	13
3.3.1 Oracle 为源	13
3.3.2 MySQL 为源	14
3.3.3 GoldenDB 为源	15
3.3.4 Microsoft SQL Server 为源	16
3.3.5 PostgreSQL 为源	16
3.4 对象采集失败的原因有哪些?	16
3.5 GaussDB 中没有 Schema 名称的对象语句验证失败, 该如何设置搜索路径?	17
3.6 原生兼容、转换兼容、部分兼容和不兼容分别是什么意思?	17
3.7 迁移风险(10大SQL风险语句)和风险SQL摘要有什么关系?	18
3.8 改造点统计是什么, 怎么统计的?	18
3.9 评估项目出现对象收集错误并显示“Closed Connection”	18

3.10 评估项目停止并报错 ErrorCode=4036.....	19
3.11 评估项目停止并报错 ErrorCode=17002.....	20
3.12 评估任务连接成功，但预检查失败.....	21
3.13 采集对源数据库的影响.....	21
4 迁移项目.....	23
4.1 创建迁移项目，为什么没有可选的评估项目？	23
4.2 “转换配置”的使用场景是什么？	23
4.3 对象校正时 SQL 修改保存失败.....	23
4.4 ERROR: syntax error at or near "@"的解决方法.....	24
4.5 对象验证失败的原因有哪些？	24
4.6 GaussDB 数据库设置 GUC 参数对迁移的影响.....	25
4.6.1 GaussDB（Oracle 兼容模式）下 GUC 参数设置.....	25
4.6.2 GaussDB（MySQL 兼容模式）数据库 GUC 参数.....	28
4.7 如何设置 GaussDB GUC 参数.....	30
4.8 Oracle 迁移至 GaussDB 完成后 index 总数查询.....	31
4.9 ctid,xc_node_id,tableoid 在 GaussDB 中已存在。	33
4.10 异构数据库迁移至 GaussDB 分布式后 select 查询无 order by 数据结果集不一致.....	35

1 产品咨询

1.1 资源冻结/释放/删除

UGO 资源为什么被释放了？

用户在华为云购买产品后，如果没有及时的进行续费或充值，将进入宽限期。如宽限期满仍未续费或充值，将进入保留期。在保留期内资源将停止服务。保留期满仍未续费或充值，存储在云服务中的数据将被删除、云服务资源将被释放。请参见[资源停止服务或逾期释放说明](#)。

UGO 资源为什么被冻结了？

资源冻结的类型有多种，最常见类型为欠费冻结。

单击了解[资源冻结的类型、冻结后对续费的影响](#)。

怎样将资源解冻？

欠费冻结：用户可通过续费或充值来解冻资源，恢复UGO正常使用。

欠费冻结的UGO允许续费、释放 或删除。

冻结、解冻、释放资源时对业务的影响

- 资源冻结时：
 - 资源将被限制访问和使用，仅允许查看已有项目，修改、删除、创建以及SQL语句转换等功能不可使用。
 - 资源被冻结后，可以手动进行删除。
- 资源解冻时：资源将被解除限制，用户可以进行创建等操作。
- 资源释放时：资源将被释放，实例将被删除。

怎样续费？

购买的UGO到期后，请在管理控制台[续费管理](#)页面进行续费操作。详细操作请参考[续费管理](#)。

资源被释放了能否恢复?

资源被删除后,将无法恢复,请谨慎操作。

1.2 源数据库忽略迁移的 schema 有哪些?

Oracle 为源

不采集的 Schema 列表:

'APEX_050000'、'AUDSYS'、'ADAMS'、'ANONYMOUS'、
'AURORA\$ORB\$UNAUTHENTICATED'、'AWR_STAGE'、'APEX_030200'、
'APEX_040200'、
'APEX_PUBLIC_USER'、'APPQOSSYS'、'BLAKE'、'CLARK'、'CSMIG'、
'CTXSYS'、'DBSNMP'、'DIP'、'DMSYS'、
'DSSYS'、'DEMO'、'DVSYS'、'DVF'、
'DBSFUSER'、'EXFSYS'、'FLOWS_FILES'、
'GGSYS'、'GSMADMIN_INTERNAL'、'GSMCATUSER'、'GSMUSER'、
'JONES'、'LBACSYS'、'MDDATA'、'MDSYS'、'MGMT_VIEW'、
'OLAPSYS'、'ORACLE_OCM'、'ORDDATA'、'ORDPLUGINS'、
'ORDSYS'、'OUTLN'、'OWBSYS'、'OWBSYS_AUDIT'、'OJVMSYS'、
'PERFSTAT'、'REMOTE_SCHEDULER_AGENT'、
'SI_INFORMTN_SCHEMA'、'SPATIAL_CSW_ADMIN_USR'、
'SPATIAL_WFS_ADMIN_USR'、'SYS'、'SYSMAN'、'SPATIAL_CSW_ADMIN_USR'、
'SYSBACKUP'、
'SYSKM'、'SYSDG'、'SYSRAC'、'SYS\$UMF'、'SYSTEM'、
'TRACESVR'、'TMSYS'、'WMSYS'、'XDB'、'XS\$NULL'、'GSMROOTUSER'。

UGO不采集以下表名称:

- 系统生成的表,比如: SYS_EXPORT_SCHEMA_%, 不采集。
- SYS_PLSQL 开头的系统对象不采集。
- 以 BIN\$ 开头被删除对象不采集。
- 嵌套表不采集。
- DBA_DATAPUMP_JOBS 视图或 USER_DATAPUMP_JOBS 视图中记录的由数据泵作业自动生成的表不采集。

MySQL 为源

'INFORMATION_SCHEMA'、'MYSQL'、'PERFORMANCE_SCHEMA'、'SYS'。

PostgreSQL 为源

'information_schema', 以及以'pg_'作为前缀的系统Schema, 如: 'pg_catalog'、'pg_toast'、'pg_temp_1'等。

Microsoft SQL Server 为源

'GUEST'、'INFORMATION_SCHEMA'、'SYS'、'DB_OWNER'、'DB_ACCESSADMIN'、'DB_SECURITYADMIN'、'DB_DDLADMIN'、'DB_BACKUPOPERATOR'、'DB_DATAREADER'、'DB_DATAWRITER'、'DB_DENYDATAREADER'、'DB_DENYDATAWRITER'。

1.3 迁移项目中，语法转换与迁移验证的区别？

语法转换是将源库对象的SQL脚本转化成目标库的SQL脚本，并没有在目标数据库中执行。

迁移验证是将转换好的目标库SQL发送到目标数据库中执行。

1.4 Oracle 数据库迁移到目标库，数据库结构有什么变化？

- Oracle的schema将对应成PostgreSQL的schema。
- Oracle的schema将对应成MySQL的database。

1.5 部分功能无法正常使用，是什么原因？

- 用户未开通对应的权限。
UGO详细的常用操作与对应的授权项，请参考[权限管理](#)。
如何开通对应权限，请参考[创建用户并授权使用UGO](#)。
- 账号冻结或受限。
此时仅允许查看已有项目，修改、删除、创建以及SQL语句转换等功能不可使用。
界面会给出相应提示，您可以直接单击“提交工单”进行处理；也可以在管理控制台右上角，选择“[工单 > 新建工单](#)”，联系客服进行处理。

1.6 迁移到目标库后，生成的 dsc_ora_ext 这个 Schema 的作用是什么？

dsc_ora_ext是UGO为了让目标库实现Oracle源库中某些特定的功能而编写的自定义Schema。

若源库为MySQL，则自定义Schema名称为dsc_mys_ext。

例如：针对Oracle的某些系统函数，UGO自定义实现相同功能的目标库函数，并在迁移时将Oracle的系统函数调用转换成UGO自定义的函数调用。

1.7 如何解决 Oracle 系统参数配置太低导致采集失败或速度慢的问题？

如果Oracle SGA系统参数配置低导致收集失败或缓慢，执行以下命令：

```
show parameter SGA;
```

样本输出：

NAME	TYPE	VALUE
sga_max_size	big integer	796M
sga_target	big integer	0

联系数据库管理员，提高SGA参数的配置。

```
ALTER SYSTEM SET sga_max_size=5G scope=spfile;
```

```
ALTER SYSTEM SET sga_target=5G scope=spfile;
```

📖 说明

上述参数值仅为参考，若SGA的值配置过大，会导致源库资源使用过高，请咨询数据库管理员后，将其替换为合适的值。

参数详细说明，可参见[Oracle官网文档](#)。

1.8 如何解决收集失败并提示“SNAPSHOT TOO OLD”的问题？

如果收集失败或提示错误“ORA-01555: snapshot too old”，需要检查源Oracle数据库的UNDO参数。执行以下命令：

```
show parameter undo;
```

样本输出：

NAME	TYPE	VALUE
undo_management	string	AUTO
undo_retention	integer	28800

联系数据库管理员，提高UNDO_RETENTION参数的配置。

```
ALTER SYSTEM SET UNDO_RETENTION =N;
```

将N替换为一个合适的值。

1.9 自建 GaussDB 内核版本号与 UGO 界面显示 GaussDB 版本号对应关系

登录自建GaussDB数据库，执行以下命令，查询版本号。

```
select version ();
```

查询出的版本号，与UGO界面上显示的GaussDB版本号对应关系如[表1-1](#)所示。

表 1-1 版本号对应关系

GaussDB内核版本号	UGO显示GaussDB版本号
V500R002C10	GaussDB V2.0-2.7 企业版
503.0.x	GaussDB V2.0-3.1 企业版
503.1.x	GaussDB V2.0-3.2 企业版
503.2.x	GaussDB V2.0-3.3 企业版
505.0	GaussDB V2.0-8.0 企业版
505.1	GaussDB V2.0-8.100 企业版
505.2	GaussDB V2.0-8.200 企业版
506.0	GaussDB V2.0-9.0 企业版

1.10 什么是 VPCEP 资源配额？

VPCEP 资源配额

在创建“对象迁移”任务中，使用数据库实例连接的方式进行测试连接和创建迁移任务的时候，UGO会使用VPC终端节点（VPC Endpoint）服务进行网络打通。

对于VPC终端节点（VPC Endpoint）服务，一个用户在单个区域中创建终端节点服务的数量默认限制为20个，一个用户在单个区域中购买终端节点的数量默认限制为50个。如果受当前资源配额限制，无法满足业务使用需求，您可以申请扩大配额。

如何申请扩大配额

步骤1 登录管理控制台。

步骤2 在页面右上角，选择“资源 > 我的配额”。系统进入“服务配额”页面，单击“申请扩大配额”。

图 1-1 申请扩大配额



步骤3 在“新建工单”页面，根据您的需求，填写相关参数。其中，“问题描述”项请填写需要调整的内容和申请原因。

图 1-2 新建工单

支持与服务

工单管理

我的工单

新建工单

我的授权

自助工具

支持计划

专业服务

新建工单

选择问题所属产品 选择问题类型 3 新建工单

我想解决 配额申请 的相关问题

* 区域

* 问题描述

文件上传 添加文件

联系方式

工单进展提醒 手机 (139****2871); 邮件 (-); 抄送邮箱 (-)

我已阅读并同意《用户须知》

提交 取消

步骤4 填写完毕后，勾选协议并单击“提交”。

---结束

2 数据库连接

2.1 创建数据库评估项目，无法连接到 Oracle 源库

请检查如下设置：

- 源数据库与UGO服务的网络是否打通，目前UGO连接源库仅支持公网连接；
- 检查网络是否稳定（包括丢包率、时延等指标），如果到源库的网络质量不好，可能导致评估任务失败；
- 检查源库的防火墙，访问白名单限制是否对UGO服务放行；
- 检查数据库连接信息是否正确。

2.2 通过 sys 用户连接源库失败

sys用户是Oracle的超级管理员用户，一般Oracle数据库限制了sys用户的远程登录，建议使用其他数据库用户。

2.3 GaussDB 如何创建不同兼容类型数据库

请确保创建数据库时使用的兼容模式与数据库评估时选择并确认的 GaussDB 兼容模式一致。

创建 Oracle 兼容模式数据库

UGO 对于 Oracle、PostgreSQL 和 SQL Server 迁移到 GaussDB 的自动转换方案均基于 GaussDB 的 Oracle兼容模式设计，在进行这些链路的迁移时，请参照以下步骤创建 Oracle 兼容模式数据库：

步骤1 以具有创建数据库权限的用户登录 GaussDB 数据库。

步骤2 执行以下命令，创建兼容 Oracle 模式 GaussDB 数据库。

集中式版：

```
create database databasename dbcompatibility = 'A';
```

分布式版：

```
create database databasename dbcompatibility = 'ORA';
```

步骤3 执行以下命令，查看是否创建成功。

```
select * from pg_database where datname = 'databasename';
```

- 集中式版：若datcompatibility兼容性一列为A，则Oracle兼容数据库创建成功。
- 分布式版：若datcompatibility兼容性一列为ORA，则Oracle兼容数据库创建成功。

----结束

说明

UGO 不支持 PostgreSQL 迁移至 GaussDB PG兼容模式。

创建 MySQL 兼容模式数据库

当源数据库为 MySQL 或 GoldenDB，迁移到 GaussDB 时，请参照以下步骤创建 MySQL 兼容模式数据库：

步骤1 以具有创建数据库权限的用户登录 GaussDB 数据库。

步骤2 执行以下命令，创建兼容 MySQL 模式 GaussDB 数据库。

集中式版B兼容：

```
create database databasename dbcompatibility = 'B';
```

集中式版M兼容：

```
create database databasename dbcompatibility = 'M';
```

分布式版：

```
create database databasename dbcompatibility = 'MySQL';
```

步骤3 执行以下命令，查看是否创建成功。

```
select * from pg_database where datname = 'databasename';
```

- 集中式版B兼容：若datcompatibility兼容性一列为B，则MySQL兼容数据库创建成功。
- 集中式版M兼容：若datcompatibility兼容性一列为M，则M兼容数据库创建成功。
- 分布式版：若datcompatibility兼容性一列为MySQL，则MySQL兼容数据库创建成功。

----结束

说明

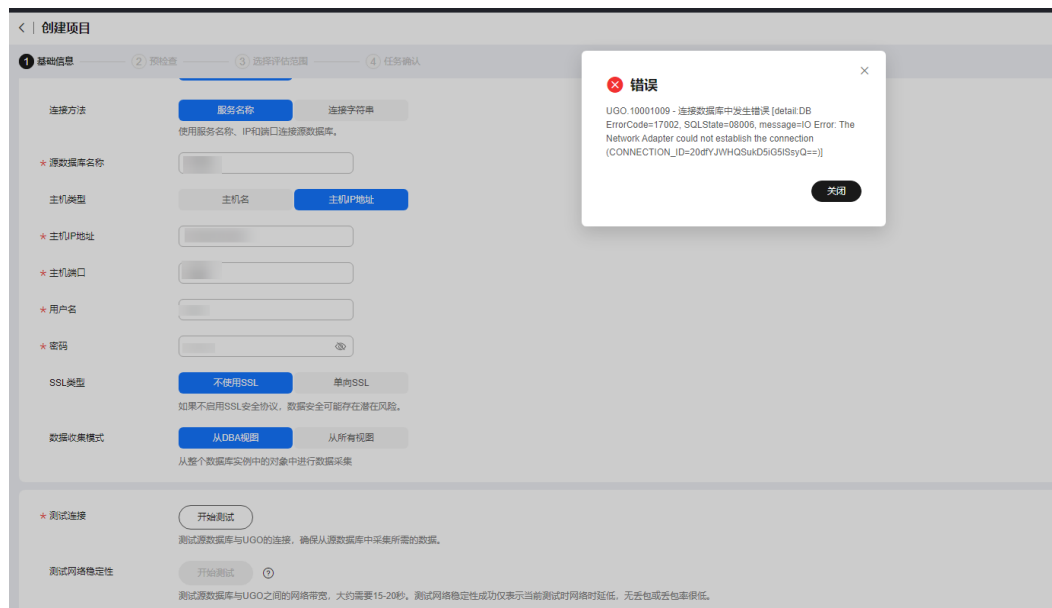
源库为 GoldenDB 迁移 GaussDB 集中式版时，暂时仅支持迁移到 B 兼容模式。

2.4 测试连接到数据库报错

问题现象

测试连接到数据库报错，提示：“连接数据库中发生错误”。

图 2-1 报错信息



可能原因

1. 数据库连接信息不正确。
2. 数据库用户没有连接权限。
3. 数据库与ugo服务的网络不通。
4. 数据库防火墙限制ugo服务访问。
5. 数据库已存在的连接达到数据库设置的最大连接数限制。

解决办法

1. 更正数据库连接信息。
2. 给用户授予连接权限。
3. 打通数据库与ugo服务之间的网络。
4. 数据库的防火墙访问白名单限制对UGO服务放行。
5. 调整数据库设置的最大连接数。

2.5 SQL Server 测试连接失败，提示错误信息 ErrorCode=0, SQLState=08S01

问题描述

当源数据库为SQL Server，并选择“不使用SSL”时，测试连接失败，弹窗提示 ErrorCode=0, SQLState=08S01，同时查看相关日志，发现有如下相关错误信息：Caused by: The server selected protocol version TLS10 is not accepted by client preferences [TLS13, TLS12]。

问题原因

即便指定了非SSL链接，SQL Server底层协议仍会自生成一个证书对登录数据包内容进行加密，较低版本可能会使用TLS1.0协议，而当前使用的JRE因为不够安全的原因，默认禁用了TLS1.0。

解决方法

按照以下步骤，修改JVM配置：

步骤1 以root用户登录到UGO服务器。

步骤2 确认Java运行环境路径，下文以JAVA_HOME代称。

步骤3 修改<JAVA_HOME>/lib/security/java.security文件。

```
vi <JAVA_HOME>/lib/security/java.security
```

步骤4 删除jdk.tls.disabledAlgorithms配置项中的TLS v1协议，并保存。

图 2-2 删除对 TLS v1 协议的禁用

```
# Disabled algorithms will not be negotiated for SSL/TLS connections, even
# if they are enabled explicitly in an application.
#
# For PKI-based peer authentication and key exchange mechanisms, this list
# of disabled algorithms will also be checked during certification path
# building and validation, including algorithms used in certificates, as
# well as revocation information such as CRLs and signed OCSP Responses.
# This is in addition to the jdk.certpath.disabledAlgorithms property above.
#
# See the specification of "jdk.certpath.disabledAlgorithms" for the
# syntax of the disabled algorithm string.
#
# Note: The algorithm restrictions do not apply to trust anchors or
# self-signed certificates.
#
# Note: This property is currently used by the JDK Reference implementation.
# It is not guaranteed to be examined and used by other implementations.
#
# Example:
# jdk.tls.disabledAlgorithms=MD5, SSLv3, DSA, RSA keySize < 2048
jdk.tls.disabledAlgorithms=SSLv3, TLSv1, TLSv1.1, RC4, DES, MD5withRSA, \
DH keySize < 1024, EC keySize < 224, 3DES_EDE_CBC, anon, NULL, \
include jdk.disabled.namedCurves
```

步骤5 重启UGO服务，<UGO_HOME>为UGO安装目录。

```
python3 <UGO_HOME>/bin/ugoserver.py restart  
----结束
```

3 评估项目

3.1 连接方式该如何选择？

- 服务名称连接：您需要填写源库的数据库名称、IP地址或主机名、用户名等参数。UGO将根据这些参数构造连接字符串。
- 连接字符串连接：JDBC URL。源库URI，应遵循源库的格式规范。您需要填写数据库名称、主机名或IP地址等参数。

3.2 采集源库的方法

Oracle 为源库

Oracle源库对象的DDL信息如无特殊说明，都是通过DBMS_METADATA.get_ddl系统包函数获取的。

对象的基本信息从系统视图中获取，有DBA视图与ALL视图两种采集模式，以下以ALL视图采集模式进行说明：

INDEX：采集索引的时候，先从ALL_INDEXES视图里面查出所有的约束信息，然后把由主键或者唯一键约束自动创建的索引过滤掉（如果先创建了索引，然后创建约束，并且约束名和索引名相同，这种情况下索引也会被认为是由约束创建的，不会被采集）。

PostgreSQL 为源库

以下是UGO对各个对象使用的采集方法：

- SCHEMA：使用 "CREATE SCHEMA " + schema名; 的拼接方式。
- TABLE：通过查询pg_catalog.pg_attribute、pg_catalog.pg_class、pg_catalog.pg_namespace、pg_inherits系统视图，获取表名、字段名、字段类型、以及表分区信息，拼接成CREATE TABLE语句。生成列、表约束、外键约束等信息未考虑。
- INDEX：通过查询pg_indexes视图的indexdef字段获取CREATE INDEX语句。
- VIEW：通过查询pg_views视图的definition字段获取视图的子查询，然后拼接成CREATE VIEW语句。

- **FUNCTION/PROCEDURE:** 通过pg_get_functiondef方法来获取CREATE FUNCTION 和 CREATE PROCEDURE语句。
请注意pg_get_functiondef方法无法处理聚合函数，因此自定义聚合函数当前无法采集。
- **TRIGGER:** 通过pg_get_functiondef方法获取触发器函数的定义，再通过pg_get_triggerdef获取CREATE TRIGGER语句，然后拼接成完整的触发器定义语句。

由于PostgreSQL数据库中并未公开获取所有对象类型定义DDL的方法，因此部分对象的DDL将采用查询元数据拼接的方式获取。这可能导致采集的DDL信息和原始DDL不一致，甚至缺少信息的情况。

如果采集的DDL无法满足您的使用要求，您可以使用pg_dump或者第三方数据库客户端工具获取对象DDL，然后通过[SQL语句转换](#)进行SQL转换。

3.3 数据库权限检查

3.3.1 Oracle 为源

检查项	描述	操作建议
DBMS_METADATA权限	检查用户是否具有从Oracle数据库字典中检索元数据的权限，该权限用来获取Schema对象的DDL。	赋权语句： GRANT SELECT_CATALOG_ROLE TO <user>; GRANT SELECT ANY DICTIONARY TO <user>; 或 GRANT DBA TO <user>; 若已授予角色，但是预检查未通过，请通过下述语句检查授予角色是否为用户的DEFAULT_ROLE。 SELECT GRANTED_ROLE, DEFAULT_ROLE FROM DBA_ROLE_PRIVS WHERE GRANTEE='<user>' AND GRANTED_ROLE IN ('SELECT_CATALOG_ROLE', 'DBA'); 若查询结果为NO，则需要执行以下语句赋权： ALTER USER <user> DEFAULT ROLE SELECT_CATALOG_ROLE; 或 ALTER USER <user> DEFAULT ROLE DBA;
动态视图权限	检查用户对各种动态性能视图的访问权限，该权限用来获取数据库基础信息。	赋予DBA权限即可。

检查项	描述	操作建议
DDL对象数量检查	检查该用户是否至少有一个Schema对象能访问并且存在DDL需要评估。	用户至少需要能够访问一个存在DDL的Schema对象，否则无法创建评估任务。注意：Oracle系统Schema中的对象不会被采集。
DBMS_METADATA SQL 格式化参数设置	检查用户是否可以执行DBMS_METADATA的SQL格式化命令，如果检查结果为警告，评估项目可以创建成功，但是采集出的sql格式可能存在问题，导致评估和转换失败。	请检查是否可以执行 DBMS_METADATA.SET_TRANSFORM_PARAMETER(dbms_metadata.SESSION_TRANSFORM, '\CONSTRAINTS_AS_ALTER\', false) 和 DBMS_METADATA.SET_TRANSFORM_PARAMETER(dbms_metadata.SESSION_TRANSFORM, '\SQLTERMINATOR\', true)语句。

3.3.2 MySQL 为源

检查项	描述	操作建议
mysql数据库进行权限检查	UGO默认使用mysql数据库进行测试连接，并需要访问mysql数据库进行权限检查。	创建用户。 CREATE USER <user> IDENTIFIED BY passwd; GRANT SELECT ON mysql.* TO <user>;
DDL对象数量检查	检查该用户是否至少有一个Schema对象能访问并且存在DDL需要评估。	用户至少需要能够访问一个存在DDL的Schema对象，否则无法创建评估任务。注意：MySQL的数据库对象不在对象数量的统计范围之内。
process权限	process权限用于指定用户可以查看information_schema所有的表。	赋权语句。 GRANT PROCESS ON *.* TO <user>;
采集对象权限	检查用户是否有对象采集的权限。	赋权语句。 GRANT SELECT ON <schema_name>.* TO <user>; GRANT SHOW VIEW ON <schema_name>.* TO <user>; GRANT TRIGGER ON <schema_name>.* TO <user>; 说明 <schema_name>.*可以指定授权全局范围*.*。

检查项	描述	操作建议
存储过程和存储函数	从MySQL8.0.20版本开始，针对存储过程和存储函数，需要授予SHOW_ROUTINE权限。	赋权语句 GRANT SHOW_ROUTINE ON *.* TO <user>;

3.3.3 GoldenDB 为源

检查项	描述	操作建议
GoldenDB系统表mysql.user的权限	该权限用于采集USER、ROLE对象的信息。	1.创建用户。 CREATE USER <user> IDENTIFIEDBY passwd; GRANT SELECT ON mysql.user TO <user>;
process权限	process权限用于指定用户可以查看information_schema所有的表。	赋权语句。 GRANT PROCESS ON *.* TO db-user;
采集对象权限	检查用户是否有对象采集的权限。	赋权语句。 GRANT SELECT ON <schema_name>.* TO <user>; GRANT SHOW VIEW ON <schema_name>.* TO <user>; GRANT TRIGGER ON <schema_name>.* TO <user>; 说明 <ul style="list-style-type: none"> • <schema_name>.*可以指定授权全局范围*.*。 • GoldenDB CN节点中创建的用户默认可以连接上所有的库，但是也同样需要对要采集的库授予SELECT权限，否则将无法采集。
存储过程和存储函数	从MySQL8.0.20版本开始，针对存储过程和存储函数，还需要授予SHOW_ROUTINE权限。	赋权语句 GRANT SHOW_ROUTINE ON *.* TO <user>;

3.3.4 Microsoft SQL Server 为源

表 3-1 SQL Server 为源检查项

检查项	描述	操作建议
DDL 查询权限检查	检查用户是否有获取数据库对象的权限。	为用户授予获取数据库对象权限。 赋权语句如下： GRANT SELECT ON OBJECT ;
DDL对象数量检查	检查该用户是否至少有一个Schema对象能访问并且存在DDL需要评估。	用户可能没有访问表的权限，请授予View Definition权限。赋权语句如下： GRANT VIEW DEFINITION ON DATABASE :: <database> TO <user>; 且用户至少需要有一个Schema对象，否则无法创建评估项目。注意：SQL Server系统Schema中的对象不在对象数量的统计范围之内。
View Definition权限检查	查询用户是否有View Definition的权限	用户无法查询元信息，请授予View Definition权限。 赋权语句如下： GRANT VIEW DEFINITION ON DATABASE :: <database> TO <user>;

3.3.5 PostgreSQL 为源

检查项	描述	操作建议
DDL对象数量检查	检查该用户是否至少有一个Schema对象能访问并且存在DDL需要评估。	用户至少需要能够访问一个模式对象，否则无法创建评估任务。注意：PostgreSQL的系统对象不在对象数量的统计范围之内。

📖 说明

评估项目创建项目的预检查阶段不会检查源数据库的权限，只检查源数据库中至少需要一个对象需要评估。

3.4 对象采集失败的原因有哪些？

可能的原因如下：

- 网络到源数据库已关闭
一此时需要恢复UGO与源数据库之间的网络连接，并重新创建评估工程。
- 目标库验证失败

- 建表语句如果指定namespace，请检查目标库是否存在该namespace，如果不存在请在目标库创建该namespace。
- 创建视图、函数等失败请检查是否依赖目标库table，如果依赖table请先确保table创建成功。

3.5 GaussDB 中没有 Schema 名称的对象语句验证失败，该如何设置搜索路径？

在GaussDB目标库中执行以下示例后，显示表“aa”不存在。

```
create schema sch1;  
  
create table sch1.aa(col int);  
  
select * from aa;
```

实际显示的错误消息为：-ERROR: 表“aa”不存在。

```
LINE 1: select * from aa;
```

^

```
SQL state: 42P01
```

```
Character: 15
```

这是由于搜索路径中不存在sch1。

您可以通过执行以下语句在搜索路径中添加sch1。

```
set search_path = "$user",public,sch1;
```

然后重新执行select语句。

```
select * from aa;
```

此时执行没有任何错误。

说明

要知道当前的search_path，请执行语句。

```
show search_path;
```

3.6 原生兼容、转换兼容、部分兼容和不兼容分别是什么意思？

原生兼容、转换兼容、部分兼容和不兼容分别是什么意思？

- 原生兼容：指与目标数据库语法完全兼容，无需转换。
- 转换兼容：指经UGO转换，语法的特性/功能与源数据库一致。
- 部分兼容：指经UGO转换，语法的部分特性/功能与源数据库仍有差异，存在风险，可在启动语法转换前通过“转换配置”来选择适合客户业务情况的转换方案。

- 不兼容：指UGO不支持转换，目标库不兼容该语法。
其中原生兼容、转换兼容和部分兼容统称为UGO支持转换项。

3.7 迁移风险（10大SQL风险语句）和风险SQL摘要有什么关系？

迁移风险和风险SQL摘要之间没有任何关系。

迁移风险（10大SQL风险语句）：指过去7天内占用源数据库CPU和内存较高，且执行时间较长的前10条SQL语句。

风险SQL摘要：从目标数据库的角度，概述迁移风险。对于目标数据库不直接支持的某些功能，应用程序需要特别关注该风险。

风险分为不同的类别，如功能风险，性能风险等。

- 功能风险：例如，目标数据库不直接支持的数据类型（如取值范围）。具体可以从data_type_mismatch, table_def_mismatch, sequence_limitation方面概述。
- 性能风险：影响库性能的风险。例如，GaussDB不支持分区间隔，具体可以从分布和分区方面概述。

3.8 改造点统计是什么，怎么统计的？

源语法中的每个子句或关键字均可视为迁移中一个改造点。

成功改造点的数据可统计为原生支持项、UGO支持项和迁移风险项。

失败改造点的数据可统计为不支持转换项。

3.9 评估项目出现对象收集错误并显示“Closed Connection”

问题描述

执行数据库评估项目，“项目状态”中的“对象收集错误”显示错误“Closed Connection”。

图 3-1 错误显示



Schema名称	对象类型	对象名称	错误
CDATA10	TABLE	BASE_SERVICE_EVENT_T	Closed Connection
CDATA10	TABLE	BASE_SITEMAP_T	Closed Connection
CDATA10	TABLE	BASE_TABLE_BACKUP_IGNORE...	Closed Connection

问题原因

UGO数据库采集源库DDL信息时设定采集时间60s，如果数据库对象过大或因数据库性能、网络连接等原因导致60s未返回数据，会导致数据采集失败。

解决方法

方法一：手动提交未采集到的对象sql

- 步骤1 登录UGO控制台。
- 步骤2 单击左侧导航栏“结构迁移 > 数据库评估”。
- 步骤3 单击待查看评估项目的“项目名称”，进入“源数据库分析”页签。
- 步骤4 单击“查看对象详情”，选中具体失败Schema。
- 步骤5 单击“编辑SQL”，手动导入SQL并保存。
- 步骤6 返回项目列表，单击“恢复”。

----结束

方法二：通过页面修改采集时间

- 步骤1 单击失败项目“操作”列的“恢复”。

图 3-2 恢复操作



- 步骤2 修改数据库连接超时时间。

- 步骤3 单击“确定”。

----结束

3.10 评估项目停止并报错 ErrorCode=4036

问题描述

评估项目运行时，“项目状态”显示“已停止”并报错ErrorCode=4036。

图 3-3 报错信息



问题原因

Pga内存问题，数据库报错“pga-memory-used-by-the-instance-exceeds-pga-aggregate-limit”。

解决方法

- 步骤1 使用oracle用户登录oracle源库。
- 步骤2 执行以下命令，查看当前pga_aggregate_limit。
show parameter pga_aggregate_limit;
- 步骤3 执行以下命令，手动修改pga_aggregate_limit大小。
alter system set pga_aggregate_limit = 0;
- 步骤4 [登录UGO控制台](#)，“恢复”项目。
----结束

3.11 评估项目停止并报错 ErrorCode=17002

问题描述

评估项目运行时，“项目状态”显示“已停止”并报错ErrorCode=17002。

问题原因

在业务繁忙时或同时启动多个数据库评估任务时，数据库连接数将会达到上限，UGO获取不到源库的有效连接，停止采集任务。

解决方法

- 步骤1 使用oracle用户登录源库。
- 步骤2 执行以下命令，查看目前数据库的最大连接数。
show parameter processes;
- 步骤3 执行以下命令，创建spfile。
create spfile from pfile;
- 步骤4 执行以下命令，重启oracle。
shutdown immediate;
startup
- 步骤5 执行以下命令，修改源库连接数大小。
alter system set processes = 2000 scope = spfile;
- 步骤6 执行以下命令，重启oracle并查看当前数据库最大连接数是否修改成功。
shutdown immediate;
startup


```
show parameter processes;
```

步骤7 登录UGO控制台，“恢复”项目。

----结束

3.12 评估任务连接成功，但预检查失败

问题现象

用户给创建评估项目的连接用户授予了DBA权限，评估任务测试连接成功，但预检查失败。

问题原因

赋给用户给了DBA权限，但是是非默认的DBA角色，需要修改为默认DBA角色。

解决办法

步骤1 登录到源数据库中，执行以下SQL语句查询，当前用户是否为默认角色。

```
select * from dba_role_privs where grantee='user_name';
```

📖 说明

- 若查询结果的dba的default_role是no，就是非默认DBA角色。
- user_name为连接源端数据库的用户。

步骤2 执行以下SQL语句，使其变为默认角色。

```
ALTER USER user_name DEFAULT ROLE dba;
```

步骤3 修改完成后在预评估界面重新开始预检查。

----结束

3.13 采集对源数据库的影响

- 当有UGO评估任务连接源库进行对象采集时，源库Oracle执行select或alter等语句，执行时间会增加约0.1s。
- 对源库Oracle性能的影响，如表所示。

表 3-2 对源库性能的影响

源库类型	源库配置	CPU使用率	内存使用率	I/O使用率	应用查询影响
无应用程序的Oracle数据库	CPU核数: 48 内存: 188GB 存储: 511GB	0.04% -> 17.03%	1.5% -> 2.56%	0.1% -> 2.8%	不涉及
TPC-H中模拟的有应用程序的Oracle数据库	CPU核数: 48 内存: 188GB 存储: 511GB	0.04% -> 19.63%	1.52% -> 2.54%	0.1% -> 10.2%	提高5%

 说明

- 实验室测试数据，具体影响依据实际情况而定。
- TPC-H是数据库业界广泛接受和使用的基准测试，由一套面向业务的临时查询和并发数据修改组成。查询和填充数据库的数据具有广泛的行业相关性。

4 迁移项目

4.1 创建迁移项目，为什么没有可选的评估项目？

未创建完整的数据评估项目。

创建迁移项目，至少需要有一个已确定目标数据库的评估项目。

4.2 “转换配置”的使用场景是什么？

在进行迁移之前，识别到源数据库应用系统中存在有多种迁移方案可选择，用户可选择最优配置方案（依赖于源库系统以及用户需求）再进行迁移。

4.3 对象校正时 SQL 修改保存失败

用户在迁移的对象校验页面进行SQL修改后保存，出现如图所示报错：

图 4-1 报错显示



查询长度已超过1MB的最大限制。

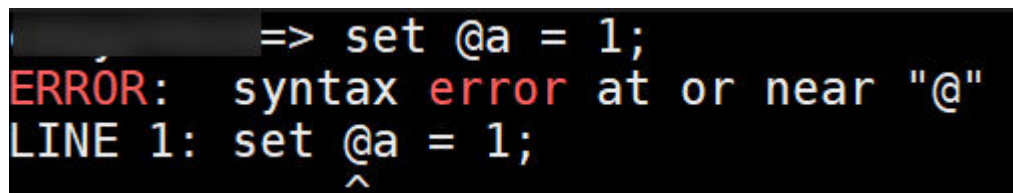
调整查询长度后正常。

4.4 ERROR: syntax error at or near "@"的解决方法

问题现象

GaussDB 集中式版 V2.0-3.2、3.3、8.0 企业版输入set @a = 1;后出现如下报错信息。

图 4-2 报错现象



```
=> set @a = 1;  
ERROR: syntax error at or near "@"  
LINE 1: set @a = 1;  
                ^
```

问题原因

未设定enable_set_variables参数。

解决方法

方法一：对指定数据库生效

步骤1 以root用户连接数据库。

步骤2 执行以下命令，切换enable_set_variables参数。

```
alter database dbname set b_format_behavior_compat_options =  
'enable_set_variables';
```

----结束

方法二：对指定的实例生效

步骤1 以root用户连接数据库实例。

步骤2 切换到数据库实例用户。

```
su - omm
```

📖 说明

"omm" 为安装数据库实例用户，请以实际数据库实例用户为准。

步骤3 执行以下命令，设置数据库参数。

```
gs_guc reload -Z datanode -N all -I all -c  
"b_format_behavior_compat_options='enable_set_variables';
```

----结束

4.5 对象验证失败的原因有哪些？

可能的原因如下：

- 目标库验证失败
 - 建表语句如果指定tablespace, 请检查目标库是否存在该tablespace, 如果不存在请在目标库创建该tablespace。
 - 创建视图、函数等失败请检查是否依赖目标库table, 如果依赖table请先确保table创建成功。
 - 是否选择了正确的转换配置进行转换, 部分转换会将源库语法进行保留, 目标库不兼容语法则会迁移失败, 应根据需求选择合理的转换配置。

4.6 GaussDB 数据库设置 GUC 参数对迁移的影响

4.6.1 GaussDB (Oracle 兼容模式) 下 GUC 参数设置

当使用Oracle兼容模式, 可以设置GUC参数提高GaussDB的兼容性, 迁移涉及的GUC参数和影响范围可参见[表4-1](#)。

注意

GaussDB数据库提供了许多运行参数, 配置这些参数可以影响数据库系统的行为。在修改这些参数时请确保用户理解了这些参数对数据库的影响, 否则可能会导致无法预料的结果。可参考[云数据库GaussDB GUC参数使用说明](#)。

表 4-1 GaussDB (Oracle 兼容模式) 的 GUC 参数

兼容 Oracle行为配置项	兼容性配置项	兼容性配置项名字	支持的数据库	是否建议打开	参数详情参考
behavior_compat_options	display_leading_zero	浮点数显示配置项-小数点前0显示	集中式版/分布式版	是	集中式版: 参数详情 分布式版: 参数详情
	end_month_calculate	add_months函数计算逻辑配置项	集中式版/分布式版	是	
	bind_procedure_searchpath	未指定模式名的存储过程中的数据库对象的搜索路径配置项。	集中式版/分布式版	是	
	correct_to_number	控制to_number()结果兼容性的配置项	集中式版/分布式版	是	

兼容 Oracle 行为配置项	兼容性配置项	兼容性配置项名字	支持的数据库	是否建议打开	参数详情参考
	unbind_divide_bound	控制对整数除法的结果进行范围校验	集中式版/分布式版	是	
	convert_string_digit_to_numeric	字符串转换为 numeric 配置项	集中式版/分布式版	是	
	return_null_string	控制函数 lpad() 和 rpad() 结果为空字符串"的显示配置项。	集中式版/分布式版	是	
	merge_update_multi	控制在使用 MERGE INTO ... WHEN MATCHED THEN UPDATE 和 INSERT ... ON DUPLICATE KEY UPDATE, 当目标表中一条目标数据与多条源数据冲突时 UPDATE 行为。	集中式版/分布式版	是	
	plstmt_implicit_savepoint	控制存储过程中更新语句的执行是否拥有独立的子事务。	集中式版/分布式版	是	
	hide_tailing_zero	numeric 显示配置项。	集中式版/分布式版	是	
	rownum_type_compat	控制 ROWNUM 的类型。	集中式版	是	
	aformat_null_test	控制 rowtype 类型判空逻辑。	集中式版/分布式版	是	
	aformat_regexp_match	控制正则表达式函数的匹配行为。	集中式版/分布式版	是	
	compat_cursor	控制隐式游标状态兼容行为。	集中式版	是	

兼容 Oracle 行为配置项	兼容性配置项	兼容性配置项名字	支持的数据库	是否建议打开	参数详情参考
	proc_outparam_override	控制存储过程出参的重载行为。	集中式版/分布式版	是	
	proc_implicit_for_loop_variable	控制存储过程中 FOR_LOOP 查询语句行为。	集中式版	是	
	allow_procedure_compile_check	控制制存储过程中 select 语句和 open cursor 语句的编译检查。	集中式版	是	
	plsql_security_definer	创建存储过程时默认为定义者权限。	集中式版/分布式版	是	
	plpgsql_dependency	创建函数，存储过程，包支持未定义的对象，可以新建成功。	集中式版	是	
plsql_compile_check_options	plsql_expression_check	开启 plsql_expression_check 参数，需要同时开启 plpgsql_dependency 参数。	503.1 集中式版	是	
a_format_version	10c	数据库平台兼容性行为配置项	集中式版/分布式版	是	
a_format_dev_version	s1	数据库平台迭代小版本兼容性行为配置项。	503.0 集中式版/分布式版	是	
	s2	数据库平台迭代小版本兼容性行为配置项。	503.1 集中式版/分布式版	是	

兼容 Oracle 行为配置项	兼容性配置项	兼容性配置项名字	支持的数据库	是否建议打开	参数详情参考
	s3	数据库平台迭代小版本兼容性行为配置项。	503.2 集中式版/分布式版	是	
	s4	数据库平台迭代小版本兼容性行为配置项	505.0 集中式版/分布式版	是	
	s5	数据库平台迭代小版本兼容性行为配置项。	505.1 集中式版/分布式版	是	
	s6	数据库平台迭代小版本兼容性行为配置项。	505.2 集中式版/分布式版	是	
sql_beta_feature	a_style_coerce	影响decode函数	集中式版/分布式版	是	

📖 说明

a_format_dev_version参数设置：

- GaussDB V2.0-3.1 企业版，设置为 s1。
- GaussDB V2.0-3.2 企业版，设置为 s2。
- GaussDB V2.0-3.3 企业版，设置为 s3。
- GaussDB V2.0-8.0 企业版，设置为 s4。
- GaussDB V2.0-8.100 企业版，设置为 s5。
- GaussDB V2.0-8.200 企业版和 GaussDB V2.0-9.0 企业版，设置为 s6。

版本号对应参考[自建GaussDB内核版本号与UGO界面显示GaussDB版本号对应关系](#)。

4.6.2 GaussDB (MySQL 兼容模式) 数据库 GUC 参数

当使用MySQL兼容模式，可以设置GUC参数提高GaussDB的兼容性，迁移涉及的GUC参数和影响范围可参见[表1 GaussDB \(MySQL兼容模式\) 的GUC参数](#)。

注意

GaussDB数据库提供了许多运行参数，配置这些参数可以影响数据库系统的行为。在修改这些参数时请确保用户理解了这些参数对数据库的影响，否则可能会导致无法预料的结果。

表 4-2 GaussDB（MySQL 兼容模式）的 GUC 参数

兼容MySQL行为配置项	兼容性配置项	兼容性配置项名字	支持的数据库	是否建议打开
b_format_behavior_compat_options	enable_set_variables	仅限集中式版会话变量生效。	集中式版	是
b_format_version	5.7	数据库平台兼容性行为配置项。	505.0 集中式版/分布式版	是
b_format_dev_version	s1	数据库平台迭代小版本兼容性行为配置项。	505.0 集中式版/分布式版	是
	s2	数据库平台迭代小版本兼容性行为配置项。	505.1 集中式版/分布式版	是

说明

b_format_dev_version参数设置：

- GaussDB V2.0-8.0 企业版，设置为 s1。
- GaussDB V2.0-8.100 企业版，设置为 s2。

版本号对应参考[自建GaussDB内核版本号与UGO界面显示GaussDB版本号对应关系](#)。

- S1兼容性配置项所影响的内容：
 - NEW()函数、last_day()函数、date_sub(date, INTERVAL expr unit)、datediff(expr1, expr2)、day()/dayofmonth()、dayname()
 - dayofweek()、dayofyear()、extract(unit FROM date)、from_days(days)、from_unixtime(unix_timestamp[,format])、get_format({DATE | TIME | DATETIME | TIMESTAMP}, {'EUR' | 'USA' | 'JIS' | 'ISO' | 'INTERNAL'})、hour()、makedate(year,dayofyear)、microsecond()、minute()、month()、monthname()、period_add(period, month_number)
 - TIMESTAMPDIFF、yearweek(date[, mode])、year()、weekofyear(date)、weekday()、week(date[, mode])、utc_timestamp()、utc_time()、period_diff(p1,p2)、second()
 - quarter()、str_to_date(str, format)、subdate(expr, days)、time_format(time, format)、ifnull(expr1, expr2)、database()、current_date、current_time、current_timestamp

- round(arg1, arg2)、localtime([precision])、localtimestamp、dbtimezone、timenow()、numtodsinterval(num, interval_unit)、numtoyminterval(num, interval_unit)
- new_time(date, timezone1,timezone2)、sysdate([precision])、add_months(d,n)、months_between(d1, d2)、convert_tz(dt, from_tz, to_tz)
- adddate(date, INTERVAL expr unit)、date_format(date, format)
- S2兼容性配置项所影响的内容：
 - 包含s1控制的兼容性行为。
 - 影响如下功能：
 - AUTO_INCREMENT自动增长列在导入数据或者进行BATCH INSERT执行计划的插入操作时，对于混合0、NULL和确定值的场景，确定值插入时会立即更新自增计数，后续的0/NULl将在确定值的基础上自增。
 - like操作符对于匹配串末尾是转义符的场景不再报错。
 - 字符集、字符序的排序规则优先级将发生变更。
 - 影响如下语法：
 - CREATE TABLE table_name LIKE source_table语法。
 - CREATE TABLE table_name LIKE source_table和“CREATE TABLE table_name (LIKE source_table)”语法不支持指定INCLUDING和EXCLUDING选项，缺省等同于指定INCLUDING ALL。
 - 支持LOAD DATA语法，gs_loader中部分与LOAD DATA语法一致的语法功能会发生变化。
 - 支持set names指定collate子句。
 - 涉及ALTER TABLE，RENAME TABLE等更改表名的语法，影响场景如：新表名对应字符串开头为“#MySQL50#”，且其后有其他字符，“#MySQL50#”将被忽略。

4.7 如何设置 GaussDB GUC 参数

GaussDB 集中式版 GUC 参数设置

GaussDB 集中式版提供两种方式设置 GUC 参数：

1. 通过gsqll、管控平台或其他客户端工具，连接到 GaussDB，执行以下SQL，完成 GUC 参数数据库级别设置：

```
ALTER DATABASE <database_name> SET behavior_compat_options =
'bind_procedure_searchpath,truncate_numeric_tail_zero,plsql_security_definer,proc_out
param_override,aformat_null_test,rownum_type_compat,allow_procedure_compile_ch
eck,proc_implicit_for_loop_variable,plstmt_implicit_savepoint,end_month_calculate,dis
able_rewrite_nesttable,plpgsql_dependency,display_leading_zero,correct_to_number,u
nbind_divide_bound,convert_string_digit_to_numeric,hide_tailing_zero,return_null_stri
ng,aformat_regexp_match,compat_cursor,enable_funcname_with_argname,tableof_el
em_constraints,merge_update_multi';
ALTER DATABASE <database_name> SET plsql_compile_check_options =
'plsql_expression_check';
```

```
ALTER DATABASE <database_name> SET sql_beta_feature = 'a_style_coerce';  
ALTER DATABASE <database_name> SET a_format_version = '10c';  
ALTER DATABASE <database_name> SET a_format_dev_version = 's6';
```

2. 登录到 GaussDB 后台，通过 gs_guc 工具，执行以下命令，完成 GUC 参数实例级别设置：

```
su - omm  
gs_guc reload -Z coordinator -Z datanode -N all -I all -c  
"behavior_compat_options='bind_procedure_searchpath,truncate_numeric_tail_zero,pls  
ql_security_definer,proc_outparam_override,aformat_null_test,rownum_type_compat,a  
llow_procedure_compile_check,proc_implicit_for_loop_variable,plstmt_implicit_savepoi  
nt,end_month_calculate,disable_rewrite_nesttable,plpgsql_dependency,display_leadin  
g_zero,correct_to_number,unbind_divide_bound,convert_string_digit_to_numeric,hide_t  
ailing_zero,return_null_string,aformat_regexp_match,compat_cursor,enable_funcname  
_with_argname,tableof_elem_constraints,merge_update_multi"  
gs_guc reload -Z coordinator -Z datanode -N all -I all -c  
"plsql_compile_check_options='plsql_expression_check'"  
gs_guc reload -Z datanode -N all -I all -c "sql_beta_feature='a_style_coerce'"  
gs_guc reload -Z coordinator -Z datanode -N all -I all -c "a_format_version='10c'"  
gs_guc reload -Z coordinator -Z datanode -N all -I all -c "a_format_dev_version='s6'"
```

📖 说明

以上示例 SQL 及命令均以 Oracle 为源、GaussDB 集中式版 V2.0 - 8.0 企业版为目标库。其他链路的 GUC 参数，请参考预检查功能中提示的参数名称和参数值，分别替换掉上述命令中等号前后的内容。

GaussDB 分布式版 GUC 参数设置

GaussDB 分布式版的云上实例不支持自行修改 GUC 参数配置，请联系 GaussDB 运维人员完成设置。

非云版本的 GaussDB 分布式版，仅支持通过后台登录后，使用 gs_guc 工具执行以下命令，完成 GUC 参数实例级别设置：

```
gs_guc reload -Z coordinator -Z datanode -N all -I all -c  
"behavior_compat_options='bind_procedure_searchpath,truncate_numeric_tail_zero,plsql_s  
ecurity_definer,proc_outparam_override,aformat_null_test,plstmt_implicit_savepoint,end_m  
onth_calculate,disable_rewrite_nesttable,display_leading_zero,correct_to_number,unbind_di  
vide_bound,convert_string_digit_to_numeric,hide_tailing_zero,return_null_string,aformat_re  
gexp_match,enable_funcname_with_argname,tableof_elem_constraints,merge_update_mu  
lti";  
gs_guc reload -Z coordinator -Z datanode -N all -I all -c "a_format_version='10c';"  
gs_guc reload -Z coordinator -Z datanode -N all -I all -c "a_format_dev_version='s6';"  
gs_guc reload -Z datanode -N all -I all -c "sql_beta_feature='a_style_coerce';"
```

📖 说明

以上示例 SQL 及命令均以 Oracle 为源、GaussDB 集中式版 V2.0 - 8.0 企业版为目标库。其他链路的 GUC 参数，请参考预检查功能中提示的参数名称和参数值，分别替换掉上述命令中等号前后的内容。

4.8 Oracle 迁移至 GaussDB 完成后 index 总数查询

问题现象

Oracle 为源库迁移至 GaussDB 后，目标库 GaussDB index 总数与源库 Oracle 查询数量不一致。

问题原因

UGO 采集时将主键和唯一约束以及系统生成的索引过滤掉，导致迁移后源库和目标库index数量不同。

解决办法

连接上Oracle数据库执行以下SQL语句查询index数量。

1. ugo评估页面显示的index个数查询语句：

```
SELECT count(*)
FROM (select owner, index_name, status from ALL_INDEXES WHERE OWNER in ('schema_name')) i l
LEFT OUTER JOIN
( WITH
cons_ind AS -- 主键和唯一键约束
(SELECT constraint_name
FROM ALL_CONSTRAINTS
WHERE owner in ('schema_name') AND constraint_type IN ('P', 'U') AND constraint_name =
index_name),
cons_col AS -- 约束的列信息
(SELECT ci.constraint_name, cc.position, cc.column_name
FROM cons_ind ci, ALL_CONS_COLUMNS cc
WHERE cc.owner in ('schema_name') AND cc.constraint_name = ci.constraint_name),
ind_col AS -- 索引的列信息
(SELECT ic.index_name, ic.column_position, ic.column_name
FROM cons_ind ci, ALL_IND_COLUMNS ic
WHERE ic.index_owner in ('schema_name') AND ic.index_name = ci.constraint_name),
not_match_ind AS -- 约束的列信息和索引的列信息不匹配
(SELECT cc.constraint_name, ic.index_name
FROM cons_col cc FULL OUTER JOIN ind_col ic
ON cc.constraint_name = ic.index_name
AND cc.position = ic.column_position
AND cc.column_name = ic.column_name
WHERE cc.constraint_name IS NULL OR ic.index_name IS NULL )

SELECT ci.constraint_name
FROM cons_ind ci
LEFT OUTER JOIN ( SELECT constraint_name FROM not_match_ind
WHERE constraint_name IS NOT NULL
UNION
SELECT index_name FROM not_match_ind
WHERE index_name IS NOT NULL ) nm
ON ci.constraint_name = nm.constraint_name
WHERE nm.constraint_name IS NULL ) fil_cons_not
ON il.index_name = fil_cons_not.constraint_name
WHERE fil_cons_not.CONSTRAINT_NAME is null;
```

2. 查询primary key+unique 约束总数：

```
SELECT count(*)
FROM ALL_CONSTRAINTS
WHERE owner in ('schema_name') AND constraint_type IN ('P', 'U') AND constraint_name =
index_name;
```

3. 系统生成的对象个数查询：

```
select owner,OBJECT_NAME, object_type as objectType, count(*) as count
from dba_objects
where object_type in ('INDEX')
and owner in ('schema_name')
and OBJECT_NAME not in( SELECT index_name
FROM (select owner, index_name, status from ALL_INDEXES WHERE OWNER in ('schema_name')) i l
LEFT OUTER JOIN
( WITH
cons_ind AS -- 主键和唯一键约束
(SELECT constraint_name
```

```

FROM ALL_CONSTRAINTS
WHERE owner in ('schema_name') AND constraint_type IN ('P', 'U') AND constraint_name =
index_name),
cons_col AS -- 约束的列信息
(SELECT ci.constraint_name, cc.position, cc.column_name
FROM cons_ind ci, ALL_CONS_COLUMNS cc
WHERE cc.owner in ('schema_name') AND cc.constraint_name = ci.constraint_name),
ind_col AS -- 索引的列信息
(SELECT ic.index_name, ic.column_position, ic.column_name
FROM cons_ind ci, ALL_IND_COLUMNS ic
WHERE ic.index_owner in ('schema_name') AND ic.index_name = ci.constraint_name),
not_match_ind AS -- 约束的列信息和索引的列信息不匹配
(SELECT cc.constraint_name, ic.index_name
FROM cons_col cc FULL OUTER JOIN ind_col ic
ON cc.constraint_name = ic.index_name
AND cc.position = ic.column_position
AND cc.column_name = ic.column_name
WHERE cc.constraint_name IS NULL OR ic.index_name IS NULL )
SELECT ci.constraint_name
FROM cons_ind ci
LEFT OUTER JOIN ( SELECT constraint_name FROM not_match_ind
WHERE constraint_name IS NOT NULL
UNION
SELECT index_name FROM not_match_ind
WHERE index_name IS NOT NULL ) nm
ON ci.constraint_name = nm.constraint_name
WHERE nm.constraint_name IS NULL ) fil_cons_not
ON il.index_name = fil_cons_not.constraint_name
WHERE fil_cons_not.CONSTRAINT_NAME is null)
and OBJECT_NAME not in (SELECT constraint_name
FROM ALL_CONSTRAINTS
WHERE owner in ('schema_name') AND constraint_type IN ('P', 'U') AND constraint_name =
index_name)
group by owner, object_type,OBJECT_NAME;

```

📖 说明

- 源库Oracle查询的index总数等于以上三部分数量之和。
- 'schema_name' 为迁移项目要迁移的schema_name，请以实际情况替换。

4.9 ctid,xc_node_id,tableoid 在 GaussDB 中已存在。

问题现象:

异构数据库迁移至GaussDB需要手动修改ctid,xc_node_id,tableoid,ctrd等列名称。

```

ugo=>
ugo=> create table test_row(tableoid int);
ERROR: column name "tableoid" conflicts with a system column name
ugo=>
ugo=> create table test_row1(xc_node_id int);
ERROR: column name "xc_node_id" conflicts with a system column name
ugo=>
ugo=>
ugo=> create table test_row2(ctid int);
ERROR: column name "ctid" conflicts with a system column name
ugo=>

```

问题原因

GaussDB中已存在系统的列名称ctid, xc_node_id, tableoid, ctrd, 如果源库也存在列名称ctid,xc_node_id,tableoid, ctrd, 迁移至GaussDB时需要手动修改创建表的列名称。

解决办法

步骤1 执行以下命令，连接GaussDB目标库。

```
gsql -U username -d database -p 4000 -W pwd
```

说明

“username”为数据库用户名；“database”为数据库名，“pwd”为用户名密码。

步骤2 连接上数据库，执行以下SQL语句，完成列名称“ctid”、“xc_node_id”、“tableoid”“ctrd”的修改。

```
create table test("CTRD" int);  
create table test("XC_NODE" int);  
create table test("TABLEO" int);  
create table test("CTRD" int);
```

----结束

4.10 异构数据库迁移至 GaussDB 分布式后 select 查询无 order by 数据结果集不一致

问题现象

图 4-3 查询 SQL 中无 order by

```
shiyang202418=>
shiyang202418=> select * from test_keep;
name | id | fatherid
-----+---+-----
C    | 3 | 1
E    | 4 | 2
E    | 4 | 2
A    | 2 | 3
D    | 5 | 1
B    | 2 | 1
(6 rows)

shiyang202418=>
shiyang202418=> select * from test_keep limit 1;
name | id | fatherid
-----+---+-----
A    | 2 | 3
(1 row)

shiyang202418=> select * from test_keep limit 1;
name | id | fatherid
-----+---+-----
C    | 3 | 1
(1 row)

shiyang202418=> select * from test_keep limit 1;
name | id | fatherid
-----+---+-----
C    | 3 | 1
(1 row)

shiyang202418=> select * from test_keep limit 1;
name | id | fatherid
-----+---+-----
C    | 3 | 1
(1 row)

shiyang202418=> select * from test_keep limit 1;
name | id | fatherid
-----+---+-----
B    | 2 | 1
(1 row)

shiyang202418=> select * from test_keep limit 1;
name | id | fatherid
-----+---+-----
C    | 3 | 1
(1 row)
```

问题原因

GaussDB 分布式数据中，数据是按照分布键均匀存储在各DN节点上，select查询无 order by时，数据随机从DN节点上读取，因此数据结果无法保证一致。

解决办法

查询的SQL语句中添加order by排序，order by排序的列为分布键。

图 4-4 查询 SQL 中存在 order by

```

shiyang202418=> \d+ test_keep
          Table "public.test_keep"
-----
Column | Type          | Modifiers | Storage | Stats target | Description
-----
name   | character_varying(10) |          |          |              |
id     | integer       |          |          |              |
fatherid | numeric(38,0) |          | main    |              |
Has OIDs: no
Distribute By: HASH(name)
Location Nodes: ALL_DATANODES
Options: orientation=row, compression=no

shiyang202418=>
shiyang202418=>
shiyang202418=> select * from test_keep order by name limit 1;
 name | id | fatherid
-----
A     | 2 | 3
(1 row)

shiyang202418=> select * from test_keep order by name limit 1;
 name | id | fatherid
-----
A     | 2 | 3
(1 row)

shiyang202418=> select * from test_keep order by name limit 1;
 name | id | fatherid
-----
A     | 2 | 3
(1 row)

shiyang202418=> select * from test_keep order by name limit 1;
 name | id | fatherid
-----
A     | 2 | 3
(1 row)

shiyang202418=> select * from test_keep order by name limit 1;
 name | id | fatherid
-----
A     | 2 | 3
(1 row)

shiyang202418=> select * from test_keep order by name limit 1;
 name | id | fatherid
-----
A     | 2 | 3
(1 row)

```