

多活容灾服务 - 数据库双活 (Oracle) 用户操作手册

天翼云科技有限公司

| | |
|-----------------------------|----|
| 1. 产品介绍 | 4 |
| 1.1. 产品定义 | 4 |
| 1.1.1. 数据库双活 | 4 |
| 1.1.2. 产品架构 | 4 |
| 1.1.3. 产品优势 | 5 |
| 2. 应用场景 | 5 |
| 3. 数据库双活最佳实践 | 6 |
| 4. 安装部署 | 6 |
| 4.1. 网络配置 | 6 |
| 4.2. 工作节点安装与配置 | 8 |
| 4.2.1. JAVA 环境配置 | 9 |
| 4.2.2. linux 系统安装节点包 | 9 |
| 4.2.3. 进程列表 | 11 |
| 4.2.4. 进程查询及停止命令 | 12 |
| 4.3. 源端 Oracle 同步配置 | 12 |
| 4.3.1. 源端 Oracle 同步 OS 侧配置 | 12 |
| 4.3.2. 源端 Oracle 同步 DB 侧配置 | 14 |
| 4.3.2.1. 10g、11g 版本 | 14 |
| 4.3.2.2. 12c 及以上版本 | 18 |
| 4.4. 目标端 Oracle 同步配置 | 22 |
| 4.4.1. 目标端 Oracle 同步 DB 侧配置 | 22 |
| 4.4.1.1. 10g、11g 版本 | 22 |
| 4.4.1.2. 12c 及以上版本 | 23 |
| 4.5. 数据同步用户最小权限 | 24 |
| 4.5.1. 源端 Oracle 抽取用户最小权限 | 24 |
| 4.5.2. 目标端 Oracle 装载用户最小权限 | 28 |
| 5. 整体流程介绍 | 35 |
| 6. 激活节点 | 36 |
| 6.1. 界面 | 36 |
| 6.2. 激活操作 | 37 |
| 7. 数据库节点 | 41 |
| 7.1. 购买数据库双活许可 | 41 |
| 7.2. 新建数据库节点 | 42 |
| 7.3. 数据库节点界面 | 49 |
| 8. 同步规则 | 50 |
| 8.1. 创建同步规则 | 50 |
| 8.2. 同步规则界面 | 74 |
| 9. 数据检查 | 76 |
| 9.1. 创建表比较规则 | 77 |
| 9.2. 比较规则界面 | 86 |
| 10. 对象比较 | 88 |
| 11. 对象修复 | 91 |
| 12. 附录 | 92 |
| 12.1. 错误代码说明 | 92 |
| 12.2. 错误代码告警说明 | 96 |
| 12.3. 表比对全局变量参数 | 98 |

版本变更说明

| 版本号 | 变更日期 | 变更内容 |
|-------|------------|---------|
| 1.0.0 | 2025-12-20 | 初始版本发布。 |
| | | |
| | | |
| | | |

1. 产品介绍

1.1. 产品定义

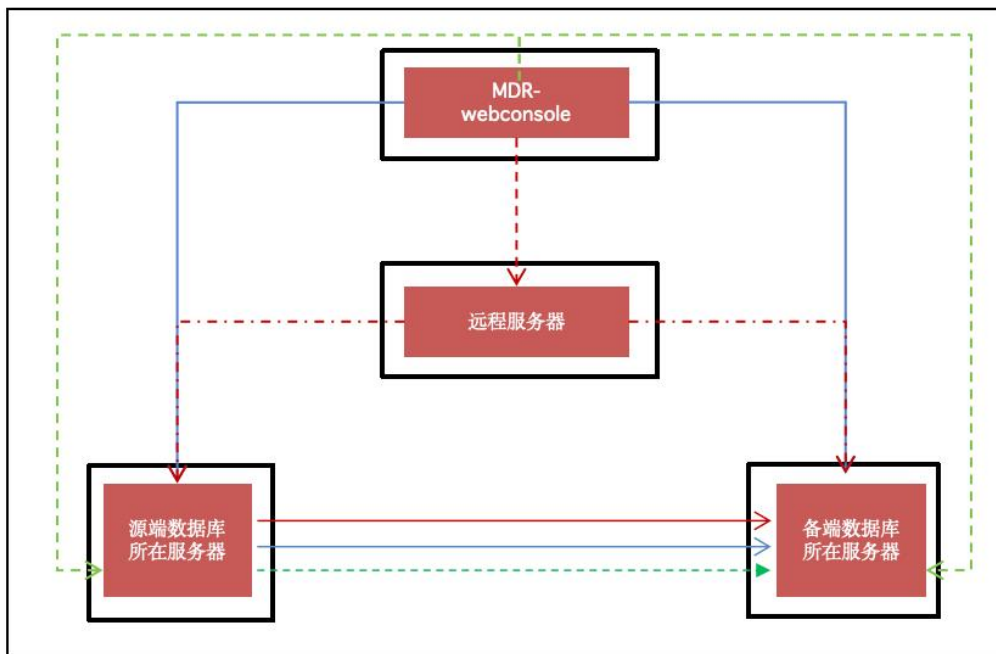
1.1.1. 数据库双活

多活容灾服务 MDR-数据库双活提供数据库语义级的同构数据库双活复制软件服务。

采用数据库之间语义级的数据实时复制与同步；基于数据库事务日志分析技术，在数据库高并发事务场景下实现数据实时同步；于目标端同步写入时序，严格确保源端和目标端的数据库事务级最终一致性；提供了备库接管、反向同步等功能。

1.1.2. 产品架构

典型架构包括三个角色：工作机、灾备机和控制机。



- 控制机：MDR-console 指提供 web 界面控制台服务的主机

webconsole: 统一数据管理平台服务，实现用户对节点资源、规则任务的管理和使用。

- 远程服务器工作节点（同步主机）：用于连接数据库，从源数据库抽取

数据并写入到目标库中。

Stream node：指节点包，用于与控制机及各数据库通信。

Stream node: 即实现对用户源端数据库读取数据然后通过 IP 网络发送给目标数据库的程序，实时捕获源数据库的事务日志通过 IP 网络以序列化传输方式发送给目标数据库并装载写入。

- 源数据库：指正常情况下用户生产系统的源端数据库服务器，或 RDS 数据库。
- 从节点：指正常情况下用户灾备系统的目标端数据库服务器，或 RDS 数据库。

1.1.3. 产品优势

1. 生产数据库无需停机

基于日志分析的数据库双活实时同步，动态多线程并发装载，能平滑地将原生产系统中的数据复制至目标数据库。

2. 数据对比和报表

帮助用户验证数据复制准确性，支持各类数据库对象、表比对，支持源端备端数据比对。

3. 生产系统无感知

采用远程部署，生产端不需要安装任何的软件，通过远程节点来进行数据的抽取和日志解析等工作；源、备端程序服务部署在同已远程节点上，大幅提高增量数据传输效率。

4. 平台兼容和版本迭代

实现跨数据库版本的数据同步，数据库版本升级迭代；实现跨操作系统平台的数据同步。

2. 应用场景

数据库实时容灾、应急接管、异地保护

- 核心业务实时容灾

- 建设两地三中心，预防区域性灾难
- 远距离同步使用窄带宽加密传输，保障数据安全性



3. 数据库双活最佳实践

1. 双活工作节点：1 个工作节点可作为 1 种数据库的同步主机。1 个工作节点可用于多个数据库的同步，建议 10 个以内；同时要考虑需要同步数据库的数据量和期望的响应速度，硬件配置有所不同。
2. 下表列出了双活工作节点应具备的最低硬件要求。在实际业务中，硬件配置的规划需考虑数据规模及所期望的数据库响应速度。

| 项目 | 配置说明 |
|-------|-----------------------|
| 服务器架构 | 支持 X86 |
| 硬盘 | 至少预留 500GB 空间，用于缓存数据。 |
| CPU | 推荐 16 核 32 线程及其以上。 |
| 内存 | 推荐 64G 内存及其以上。 |
| 网络要求 | 百兆以上以太网。 |

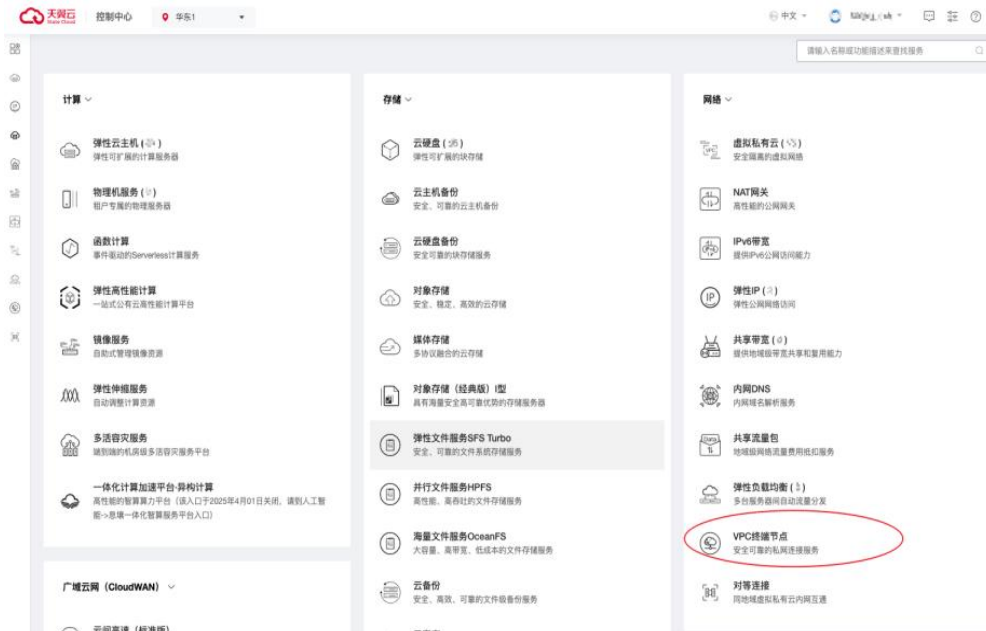
4. 安装部署

4.1. 网络配置

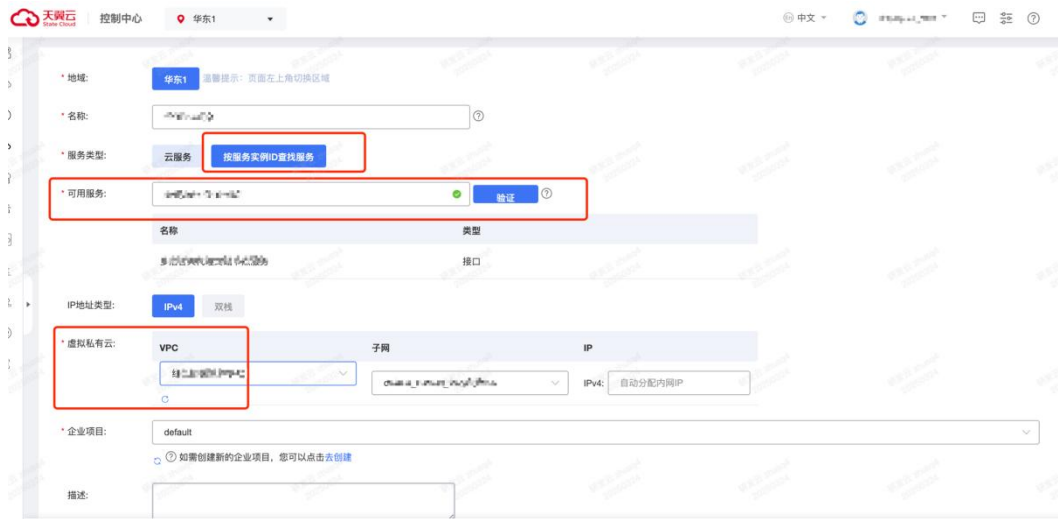
场景 1：若同步资源为天翼云内资源时，需手动配置其需同步资源所在的虚拟私有云（VPC），并通过部署 VPC 终端节点（VPCEP）实现 MDR 网络代理与目标

VPC 的安全互联。

1. 登录天翼云，进入[控制中心](#)。
2. 单击控制中心顶部的📍，选择“区域”。
3. 在服务列表选择“网络” - “VPC 终端节点”。



4. 点击右上角“创建终端节点”按钮，进入创建 VPC 终端节点页面。
5. 在进行节点添加之前，需要把云主机所在的 VPC，进行终端节点连接配置，截图如下：



服务类型选择“按服务实例 ID 查找服务”。其中，可用服务处填写 MDR 在不同资源池内的代理 VPC 终端节点服务 ID（为 MDR 侧提供固定 ID，不同资源池 ID 不一样）。不同资源池对应的代理 VPC 终端节点服务 ID 如下：

| 资源池名称 | 终端节点服务 ID |
|-------|-----------|
|-------|-----------|

| | |
|------|--------------------|
| 华东 1 | endpser-bjs8nmhm5m |
| 西南 1 | endpser-fnc13o1uao |
| 华南 2 | endpser-x6xhocvz79 |
| 西南 2 | endpser-ikzxim4cpv |
| 华北 2 | endpser-lmmnp90xgx |

虚拟私有云选择需要进行添加的 ECS 节点所在的 VPC。

注意：此链接对于租户侧不收费，费用都在终端节点服务端侧（MDR）结算。

- 租户配置终端节点成功后，点击详情页可查看节点 IP。此节点 IP 就是后续安装 i2Stream 客户端时，需要进行配置填写的 IP。



场景 2：若同步资源为云下、其他时，可参考如下：

- 云下或者它云场景，需要联系技术专家针对客户实际场景进行方案解决。
- 主要网络打通方案参考：
 - 云下通过公网与 MDR 打通
 - 云下通过专线：<https://www.ctyun.cn/document/10026762>
 - 云下通过 VPN：<https://www.ctyun.cn/document/10000057/10012487>
 - 云下通过 SD-WAN：<https://www.ctyun.cn/document/10390094/10028932>

4.2. 工作节点安装与配置

节点包安装步骤及注意事项：

- 所有数据库复制链路共用同一节点包，需对源端和目标端的机器安装节点包；
- 节点包集成同步所需的所有程序，包含 iadumper、iaowork、iatrack、iaback、

ialoader、iamonitor 和 iadiff 等。

4.2.1. JAVA 环境配置

工作节点软件同步程序运行依赖 Java 环境，建议安装 jdk17 及以上版本。

具体步骤如下：

1. 请检查确保当前机器具备 JDK17 的环境。

```
# java -version
```

2. 如未安装，请下载 JAVA 软件包进行安装。

```
[root@ecm-4f98 i2stream]# rpm -ivh jdk-17.0.16_linux-x64_bin.rpm
警告: jdk-17.0.16_linux-x64_bin.rpm: 头V3 RSA/SHA256 Signature, 密钥 ID 8d8b756f: NOKEY
准备中...##### [100%]
正在升级/安装...
 1: jdk-17-2000:17.0.16-12##### [100%]
[root@ecm-4f98 i2stream]#
```

```
[root@node123 sqllib]# java -version
java version "17.0.15" 2025-04-15 LTS
Java(TM) SE Runtime Environment (build 17.0.15+9-LTS-241)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.15+9-LTS-241, mixed mode, sharing)
[root@node123 sqllib]#
```

4.2.2. linux 系统安装节点包

用户需要准备适配的 OS 以完成安装节点包和进程启动，具体步骤如下：

1. 将安装包下载至服务器，MD5 完整性校验通过后 (linux 命令参考: md5sum 文件名)，解压该安装包。

```
# tar -zxvf i2Stream-node-<i2-version>.<os-version>.tar.gz
```

```
[root@ecm-4f98 i2stream]# pwd
/i2stream
[root@ecm-4f98 i2stream]# ls
i2stream-helper-9.1.3-2511071707.el7.x86_64.tar.gz  i2stream-node-9.1.3-2508261709.el7.x86_64.tar.gz
[root@ecm-4f98 i2stream]# tar -xvf i2stream-node-9.1.3-2508261709.el7.x86_64.tar.gz
i2stream/
i2stream/oracle/
i2stream/oracle/bin/
i2stream/oracle/bin/dumpredo
i2stream/oracle/bin/dumptxn
i2stream/oracle/bin/dumpdict
i2stream/oracle/bin/dumpasm
i2stream/oracle/bin/dumptab
i2stream/oracle/bin/precord
```

注意：安装包的解压需在 root 用户下进行命令操作。

2. iahelper 进程启动

iahelper 服务主要用于连接控制台，使得控制台可发现该节点。节点包安装完成后只需启动 iahelper 进程，其余同步进程在创建规则时自动拉起。

Linux/Aix 操作系统下启动 iahelper 进程的同时，会启动 iahelper watchdog 监视进程，监视进程会根据用户在激活双活工作节点时，选择的双活工作节点类型，

来开启对应的服务，启动服务的具体步骤如下：

- 1) 编辑/etc/hosts 文件，增加 nodeproxy-mdr 解析。

```
# vim /etc/hosts
```

添加或修改条目来映射域名到 IP 地址：

```
VPCE 的节点 IP nodeproxy-mdr
```

```
[root@ecm-4f98 bin]# cat /etc/hosts
127.0.0.1    localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
[root@ecm-4f98 bin]# vi /etc/hosts
[root@ecm-4f98 bin]# cat /etc/hosts
127.0.0.1    localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
192.168.0.9 nodeproxy-mdr
[root@ecm-4f98 bin]#
```

- 2) 进入到 MDR 工作节点的安装路径，其中<NODE_INSTALLATION_PATH>为安装包路径。

```
# cd <NODE_INSTALLATION_PATH>/bin/
```

- 3) 执行 ./iahelper nodeproxy-mdr，iahelper 首次启动需要输入密码，初始密码规定应为 8-16 位，需包含大小写字母，数字及特殊符号。

```
[root@ecm-4f98 bin]# pwd
/i2stream/i2stream/bin
[root@ecm-4f98 bin]# ls
create_helper_serv.sh  iadiff-patchsql.bat  iamonitor_dbclient_sqlmapper.sh  stop_iadiff.bat
iadefug                iadiff-patchsql.sh  setenv_libs_javahome.sh          stop_iadiff.sh
iadiff                 iadiff-run-class.sh start_iadiff.bat                  stop_ialoader.bat
iadiff.bat             iadiff-stop.bat     start_iadiff.sh                   stop_ialoader.sh
iadiff-console.bat    iadiff-stop.sh      start_ialoader.bat                stop_iamonitor.bat
iadiff-console.sh     iadiff-uuids.bat    start_ialoader.sh                 stop_iamonitor.sh
iadiff-console-test.sh iadiff-uuids.sh     start_iamonitor.bat                start_iamonitor.sh
iadiff-ipclient.sh    ia-helper            start_iamonitor.sh
[root@ecm-4f98 bin]# ./iahelper nodeproxy-mdr

Please customize the password for IaHelper.

1. Please enter the password of node:
*****
2. Please re-enter the password of node:
*****
The node password was initialized successfully.
[root@ecm-4f98 bin]#
```

- 4) 创建/root/ia/system.conf 文件，配置 tenant 指定租户 ID，其中 53f03e0d84324fc5a6e1bf32c90fb94b 可替换成租户的 ID。

```
# tenant=53f03e0d84324fc5a6e1bf32c90fb94b
```

```
[root@ecm-4f98 bin]# cd /root/ia
[root@ecm-4f98 ia]# ls
active.ca.csr  active.csr.conf  active.private.key  active.public.key
[root@ecm-4f98 ia]# vi system.conf
[root@ecm-4f98 ia]# cat system.conf
tenant=53f03e0d84324fc5a6e1bf32c90fb94b
[root@ecm-4f98 ia]#
```

- 5) 重启 ia-helper。再次执行 ./iahelper nodeproxy-mdr，注册双活工作节点至控

制机。

```
[root@ecm-4f98 bin]# pwd
/i2stream/i2stream/bin
[root@ecm-4f98 bin]# ls
create_helper_serv.sh  iadiff-patchsql.bat  iamonitor_dbclient_sqlmapper.sh  stop_iadiff.bat
iadefbug               iadiff-patchsql.sh  setenv_libs_javahome.sh         stop_iadiff.sh
iadiff                 iadiff-run-class.sh start_iadiff.bat                 stop_ialoader.bat
iadiff.bat            iadiff-stop.bat    start_iadiff.sh                 stop_ialoader.sh
iadiff-console.bat   iadiff-stop.sh    start_ialoader.bat             stop_iamonitor.bat
iadiff-console.sh    iadiff-uuids.bat  start_ialoader.sh              stop_iamonitor.sh
iadiff-console-test.sh iadiff-uuids.sh   start_iamonitor.bat
iadiff-ipccclient.sh iahelper           start_iamonitor.sh
[root@ecm-4f98 bin]# ./iahelper nodeproxy-mdr
[root@ecm-4f98 bin]# Iahelper Started, ver 9.1.3-2511071707, port 26837
[root@ecm-4f98 bin]# █
```

6) 执行如下命令可检查 iahelper 进程是否启动成功。

```
# ps -ef | grep iahelper
```

```
[root@ecm-4f98 bin]# ps -ef | grep iahelper
root      29114      1   0 16:59 ?        00:00:00 ./iahelper nodeproxy-mdr
root      29303  27524   0 17:01 pts/0    00:00:00 grep --color=auto iahelper
[root@ecm-4f98 bin]# █
```

首次启动 iahelper 服务后，会自动在启动进程用户的家目录 (\$HOME) 下生成 ia 的文件夹。

- ia 文件夹内的 proxy.conf 记录端口号，可通过修改参数 ports 的值变更端口。默认为 26803。
- ia 文件夹内的 watchdog.lck 监视当前源机节点相关服务，主要监视服务掉线重连。
- 待节点在页面被激活后，ia 文件夹内会自动生成 active.conf 文件，其中参数 <I2CTRLCENTER_IP>就会被写入至此文件中的 webip 参数中，方便再次启动 iahelper 服务后，只需输入 ./iahelper 即可启动服务，无需再增加控制台的 IP 地址。后续可使用如下方式启动：

```
# ./iahelper 或 # <NODE_INSTALLATION_PATH>/bin/iahelper。
```

4.2.3. 进程列表

oracle→oracle 链路的将会运行的进程如下表：

| iahelper | iamonitor | iawork | iatrack | iaback | IaDumper | IaLoader | MysqlReplication |
|----------|-----------|--------|---------|--------|----------|----------|------------------|
| √ | √ | √ | √ | √ | NA | NA | NA |

Iahelper 启动后，其他进程会在创建规则时自动拉起，无需手动启动。在节

点上通过执行 jps 命令可以查看各进程运行状态:

```
[root@ecm-4f98 ~]# jps
31905 Jps
30279 IaMonitor
[root@ecm-4f98 ~]#
```

4.2.4. 进程查询及停止命令

下方为进程相关命令:

1. 查询在双活工作节点是否启动了 ia-helper、ia-work、ia-track、ia-back 等服务并停止。

```
# ps -ef |grep 'ia-helper'|grep -v 'grep'|awk '{print $2}'|xargs kill -9
```

```
# ps -ef |grep 'ia-work'|grep -v 'grep'|awk '{print $2}'|xargs kill -9
```

```
# ps -ef |grep 'ia-track'|grep -v 'grep'|awk '{print $2}'|xargs kill -9
```

```
# ps -ef |grep 'ia-back'|grep -v 'grep'|awk '{print $2}'|xargs kill -9
```

2. 可执行 jps 命令查看任务进程号。

```
[root@oracle11g bin]# jps
16880 IaMonitor
10258 Jps
16967 IaDumper
17111 IaProducer
10200 iadiff
5178 IaConsumer
```

4.3. 源端 Oracle 同步配置

4.3.1. 源端 Oracle 同步 OS 侧配置

若节点服务器未部署 Oracle 数据库, 则安装 Oracle Client 的步骤为**必配置项**。

1. 确保当前服务器具备 **Oracle Client 19 版本** (建议使用版本) 的环境, 如未安装, 请参考以下步骤进行安装:

2. 下载 Oracle Instant Client 19 版本, Oracle 官网下载地址如下, 请根据源端 Oracle 数据库的版本以及当前服务器节点的操作系统进行下载对应版本的 Oracle Instant Client, 下载 Basic Package (RPM)对应的 rpm 包:

```
https://www.oracle.com/database/technologies/instant-client/linux-x86-64-downloads.html
```

(1) 安装 Oracle Instant Client 的操作系统依赖:

```
# dnf -y install libaio-devel libnsl
```

(2) 安装下载的 Oracle Instant Client 19 版本的 rpm 包:

```
# rpm -ivh oracle-instantclient<oracle_version>-basic-<oracle_version>.x86_64.rpm
```

(3) 在启动 i2Stream 的系统用户下的 .bash_profile 文件中添加环境变量:

```
# vi ~/.bash_profile
```

(4) 在最后一行, 添加环境变量内容如下:

```
export LD_LIBRARY_PATH=<oracle_lib_path>:$LD_LIBRARY_PATH  
ulimit -c unlimited
```

➤ 说明:

根据以下命令进行查询安装的路径: `find / -name libclntsh.so.*`, 一般默认的路径为 `/usr/lib/oracle/19.28/client64/lib/`

➤ 举例说明:

执行以上的查找命令, 显示出来的结果是: `/usr/lib/oracle/19.28/client64/lib/libclntsh.so.19.1`, 则取到前面的 lib 文件夹路径进行填写。此时 `<oracle_lib_path>` 为 `/usr/lib/oracle/19.28/client64/lib`。

(5) 保存后, 生效此文件。

```
# source ~/.bash_profile
```

3. 源备库 Oracle Client 端字符集需与 Oracle Server 保持一致, 备端字符集应为源端字符集的超集或保持一致。如同步数据包含中文, 应确认 Oracle Server 端字符集支持中文。Oracle Server 端字符集应由客户修改。

➤ 注意: 远程部署时, 要求部署远程节点的服务器 Oracle Client 端字符集与源库 Oracle Server 保持一致。

(1) 源备登录数据库确认 Oracle Server 字符集

```
> SELECT * FROM NLS_Database_Parameters;
```

➤ 说明:

上述 SQL 查询结果, NLS_LANGUAGE 指定服务器消息的语言, 影响提

示信息是中文还是英文。

NLS_TERRITORY 指定服务器的日期和数字格式, NLS_CHARACTERSET 是数据库字符集。

(2) 在 .bash_profile 文件中修改 Oracle Client 端字符集与 Oracle Server 保持一致

```
$ vi ~/.bash_profile
```

(3) 根据步骤 a 的查询结果, 在文件最后一行添加字符集设置

```
export NLS_LANG=LANGUAGE_TERRITORY.CHARACTERSET
```

➤ 说明:

例 NLS_LANGUAGE 为 AMERICAN, NLS_TERRITORY 为 AMERICA, NLS_CHARACTERSET 为 AL32UTF8, 则下述配置为 export NLS_LANG=AMERICAN_AMERICA.AL32UTF8。

(4) 保存后, 生效此文件

```
$ source ~/.bash_profile
```

(5) 确认配置生效, 查询字符集

```
$ echo $NLS_LANG
```

4. 完成 Oracle 节点服务器操作系统的环境配置。

➤ 注意:

1. 节点软件在运行前, 依赖于启动用户环境变量下的 LD_LIBRARY_PATH 下的路径, 否则会导致软件的进程或规则无法启动。
2. Oracle Server 已包含 Oracle Client, 本地部署无需另行安装 Oracle Client。

4.3.2. 源端 Oracle 同步 DB 侧配置

由于数据库版本不同, 配置的命令不同, 请根据数据库版本进行 DB 侧配置。

4.3.2.1. 10g、11g 版本

1. 源库建议开启归档模式, 查看当前归档模式是否开启 (建议开启归档模式, 不开启归档也可正常使用, 但同步规则发生异常后, 不支持断点续传的功能。只能通过重新全同步解决) :

```
SQL> archive log list
```

(1) 显示的 Database log mode 的结果需要和以下的信息保持一致:

```
Database log mode      Archive Mode
```

(2) 如若不一致 (显示模式为: No Archive Mode) , 开启归档模式的步骤如下:

① 首先需要关闭当前数据库的状态: `SQL> shutdown immediate`

② 此时将数据库启动到 mount 状态 (开启归档模式需要将数据库状态启动为 mount 才可以开启) :

```
SQL> startup mount
```

```
SQL> alter database archivelog;
```

③ 此时再将数据库启动为 open 状态: `SQL> alter database open;`

④ 可再次通过以下命令确认是否开启归档模式: `SQL> archive log list`

2. 建议检查源库审计日志级别, 当源库审计日志级别设置较高时, 会产生较多审计日志, 占用服务器资源, 可通过如下命令确认审计日志级别: `SQL> show parameter audit_trail;`

(1) 返回参数 VALUE 对应审计日志级别, 包含 None、DB、OS 等

- None 为默认值, 即不做审计
- DB 意指将 audit trail 记录在数据库的审计相关表中, 例如: aud\$
- OS 意指将 audit trail 记录在操作系统文件中

(2) 若返回值为 None 和 DB 以外的级别时, 可能产生较多审计日志

3. 需要对源备数据库的连接用户进行必要的配置, 若具体操作步骤如下:

(1) 源库均创建 i2 用户: `SQL> create user i2 identified by i2;`

(2) 源库应赋予 i2 用户权限:

```
SQL> grant select on sys.link$ to i2;
```

```
SQL> grant select on sys.seq$ to i2;
```

```
SQL> grant select on sys.user$ to i2;
```

```
SQL> grant dba to i2;
```

```
SQL*Plus: Release 11.2.0.4.0 Production on Sun Dec 14 17:46:13 2025
Copyright (c) 1982, 2013, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.4.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL> create user i2 identified by i2;

User created.

SQL> grant select on sys.link$ to i2;

Grant succeeded.

SQL> grant select on sys.seq$ to i2;

Grant succeeded.

SQL> grant select on sys.user$ to i2;

Grant succeeded.

SQL> grant dba to i2;

Grant succeeded.

SQL> |
```

- 说明：若 dba 用户权限受阻，则可以更改为其余 SQL 语句，详见下方源库 Oracle 最小权限。

4. 将源库改为强制写日志：

- (1) SQL 命令如下：`SQL> alter database force logging;`
- (2) 执行查询 SQL：`select force_logging from v_$database;`

5. 源库增加附加日志，又叫补充日志，Oracle 官网对附加日志描述中提及到对源库的影响较小，用户可放心。

- (1) 源库必须开启最小附加日志：

➤ 说明：

对于数据量较大的库，该命令可能要耗费较长时间来完成，请耐心等待。如果需要加快配置，可以将数据库关闭然后启动于 mount 状态，再执行以下 SQL 语句：`SQL> alter database add supplemental log data;`

```
SQL> alter database force logging;
Database altered.
SQL> select force_logging from v_$database;
FOR
---
YES
SQL> alter database add supplemental log data;
Database altered.
SQL> █
```

(2) 如果是主键映射复制方式可针对数据库或表补充开启 ALL,PRIMARY,UNIQUE 或 FOREIGN 标识键附加日志。缺省情况下, Oracle 不启用以上任何附加日志。当使用 ALL,PRIMARY,UNIQUE 或 FOREIGN 标识键附加日志时, 最小附加日志默认开启。

① 数据库开启命令如下:

1) 全字段补充日志, 即所有字段的值不论是否被修改都记录:

```
SQL> alter database add supplemental log data (ALL) columns;
```

2) 唯一键补充日志, 即只有唯一索引的字段被更新时, 才会记录该字段被修改前的值:

```
SQL> alter database add supplemental log data (UNIQUE) columns;
```

3) 主键补充日志, 即在更新命令的重做记录中添加被修改行的主键字段的旧值, 无论是否被修改, 都记录:

```
SQL> alter database add supplemental log data (PRIMARY KEY) columns;
```

4) 通过如下 SQL 语句查询确认数据库级修改结果:

➤ 说明: supplemental_log_data_min 意指最小附加日志, supplemental_log_data_pk 意指主键附加日志, supplemental_log_data_ui 意指唯一键附件日志, supplemental_log_data_all 意指全字段附加日志。

```
SQL> select supplemental_log_data_min, supplemental_log_data_pk, supplemental_log_data_ui, supplemental_log_data_all from v_$database;
```

(3) 针对表开启命令如下:

➤ 说明: 通常目标端数据库是只读的, 如果遇到目标端数据库必须修改的情况, 那么目标端增量 urp 转 irp (更新转插入) 时必须在源端数据库开全字段附加日志, 否则插入的数据只有更新的字段。

① 全字段补充日志, 即所有字段的值不论是否被修改都记录:

```
SQL>
```

```
alter table <table_name> add supplemental log data (ALL) columns;
```

② 唯一键补充日志，即只有唯一索引的字段被更新时，才会记录该字段被修改前的值：

```
SQL> alter table <table_name> add supplemental log data (UNIQUE) columns;
```

③ 主键补充日志，即在更新命令的重做记录中添加被修改行的主键字段的旧值，无论是否被修改，都记录：

```
SQL> alter table <table_name> add supplemental log data (PRIMARY KEY) columns;
```

④ 通过如下 SQL 语句查询确认表级修改结果：

- 说明：LOG_GROUP_TYPE 查询结果对应附加日志类型，<table_name>处应填写具体的表名。

```
SQL> select TABLE_NAME,LOG_GROUP_TYPE FROM DBA_LOG_GROUPS  
WHERE TABLE_NAME='<table_name>;'
```

6. 若规则需同步 JOB 对象，需于源库执行如下语句开启 sys.job\$ 系统表的字段附加日志。

- 说明：若不配置，在增量同步下，当源库的 job 执行时，程序无法捕获该 job 的 nextval 变更时间，这个会导致在备库接管时，启动 job 后会立即额外执行一次补偿性任务，即可能会导致重复执行该 job。如果数据依赖 job 的生成，则可能会导致数据重复处理。

```
SQL> ALTER TABLE SYS.JOB$ ADD SUPPLEMENTAL LOG GROUP  
I2_SYS_JOB (JOB, LOWNER) ALWAYS;
```

7. 完成 Oracle 源库数据库的配置。

4.3.2.2. 12c 及以上版本

1. 源库建议开启归档模式，查看当前归档模式是否开启：

```
SQL> archive log list
```

- 说明：建议开启归档模式，不开启归档也可正常使用，但同步规则发生异常后，不支持断点续传的功能。只能通过重新全同步解决。

(1) 显示的 Database log mode 的结果需要和以下的信息保持一致：

| Database log mode | Archive Mode |
|-------------------|--------------|
|-------------------|--------------|

(2) 如若不一致（显示模式为：No Archive Mode），开启归档模式的步骤如

下，开启归档模式需要将数据库状态启动为 mount 才可以开启。

① 首先需要关闭当前数据库的状态：`SQL> shutdown immediate`

② 此时将数据库启动到 mount 状态：

```
SQL> startup mount
```

```
SQL> alter database archivelog
```

③ 此时再将数据库启动为 open 状态：

```
SQL> alter database open;
```

④ 可再次通过以下命令确认是否开启归档模式：

```
SQL> archive log list
```

2. 软件需要对源库进行必要的配置，具体操作步骤如下：

(1) 确保源库当前进入的是 CDB 根容器内：

```
alter session set container = CDB$ROOT;
```

(2) 查看源库当前 oracle12c 版本及其以后版本的数据库有哪些 PDB 容器：

```
SQL> show pdbs
```

(3) 源库指定的 PDB 容器若未打开，需要将数据库打开，其中<PDB_NAME>指的是 PDB 容器名。

```
SQL> alter pluggable database <PDB_NAME> open
```

(4) 源库进入到需要进行数据库复制的 PDB 容器：

```
SQL> alter session set container = <PDB_NAME>
```

(5) 源库创建 i2 用户：

```
SQL> create user i2 identified by i2;
```

(6) 仅创建同步规则时，源库赋予 i2 用户权限：

```
SQL> grant select on sys.link$ to i2;
```

```
SQL> grant select on sys.user$ to i2;
```

```
SQL> grant select on sys.seq$ to i2;
```

```
SQL> grant select on sys.undo$ to i2;
```

```
SQL> grant dba to i2;
```

说明：若 dba 用户权限受阻，则可以更改为其余 SQL 语句，详见源库 Oracle 最小权限。

(7) 将源库改为强制写日志:

SQL 命令如下: SQL> alter database force logging;

执行查询 SQL: select force_logging from v_\$database;

(8) 源库进入到 CDB 根容器内:

```
alter session set container = CDB$ROOT;
```

(9) SQL 命令如下:

```
SQL> alter database force logging;
```

(10) 执行查询 SQL:

```
select force_logging from v_$database;
```

(11) 源库增加附加日志, 又叫补充日志 (Oracle 官网对附加日志描述中提及到对源库的影响较小, 用户可放心。)

① 源库进入到 CDB 根容器内:

```
alter session set container = CDB$ROOT;
```

② 源库必须开启最小附加日志 (说明: 对于数据量较大的库, 该命令可能要耗费较长时间来完成, 请耐心等待。如果需要加快配置, 可以将数据库关闭然后启动于 mount 状态, 再执行以下 SQL 语句。)

```
SQL> alter database add supplemental log data;
```

③ 如果是主键映射复制方式可针对数据库或表补充开启 ALL, PRIMARY, UNIQUE 或 FOREIGN 标识键附加日志 (缺省情况下, Oracle 不启用以上任何附加日志。当使用 ALL, PRIMARY, UNIQUE 或 FOREIGN 标识键附加日志时, 最小附加日志默认开启)

1) 数据库开启命令如下 (全字段补充日志, 即所有字段的值不论是否被修改都记录) :

```
SQL> alter database add supplemental log data (ALL) columns;
```

a. 主键补充日志, 即在更新命令的重做记录中添加被修改行的主键字段的旧值, 无论是否被修改, 都记录:

```
SQL> alter database add supplemental log data (PRIMARY KEY) columns;
```

b. 通过如下 SQL 语句查询确认数据库级修改结果:

```
SQL> select supplemental_log_data_min, supplemental_log_data_
```

```
pk,supplemental_log_data_ui,supplemental_log_data_all from v_$databa  
se;
```

- 说明： supplemental_log_data_min 意指最小附加日志， supplemental_log_data_pk 意指主键附加日志， supplemental_log_data_ui 意指唯一键附件日志， supplemental_log_data_all 意指全字段附加日志

2) 针对表开启开启命令如下：

- 说明：通常目标端数据库是只读的，如果遇到目标端数据库必须修改的情况，那么目标端增量 urp 转 irp（更新转插入）时必须要在源端数据库开全字段附加日志，否则插入的数据只有更新的字段。

- a. 全字段补充日志，即所有字段的值不论是否被修改都记录：

```
SQL> alter table <table_name> add supplemental log data (ALL)
```

```
columns;
```

- b. 唯一键补充日志，即只有唯一索引的字段被更新时，才会记录该字段被修改前的值：

```
SQL> alter table <table_name> add supplemental log data (UNI
```

```
QUE) columns;
```

- c. 主键补充日志，即在更新命令的重做记录中添加被修改行的主键字段的旧值，无论是否被修改，都记录：

```
SQL> alter table <table_name> add supplemental log data (PRI
```

```
MARY KEY) columns;
```

- d. 通过如下 SQL 语句查询确认表级修改结果：

- 说明： LOG_GROUP_TYPE 查询结果对应附加日志类型， <table_name>处应填写具体的表名

```
SQL> select TABLE_NAME,LOG_GROUP_TYPE FROM DBA_L
```

```
OG_GROUPS WHERE TABLE_NAME='<table_name>';
```

(12) 若规则需同步 JOB 对象，需于源库执行如下语句开启 sys.job\$系统表的字段附加日志。

```
SQL> ALTER TABLE SYS.JOB$ ADD SUPPLEMENTAL LOG GROUP I
```

```
2_SYS_JOB (JOB, LOWNER) ALWAYS;
```

- 说明：若不配置，在增量同步下，当源库的 job 执行时，程序无法捕获该 job 的 nextval 变更时间，这个会导致在备库接管时，启动 job 后会立即额外执行一次补偿性任务，即可能会导致重复执行该 job。如果数据依赖 job 的生成，则可能会导致数据重复处理。

(13) 完成 Oracle 源数据库的配置。

4.4. 目标端 Oracle 同步配置

4.4.1. 目标端 Oracle 同步 DB 侧配置

由于数据库版本不同，配置的命令不同，请根据数据库版本进行 DB 侧配置。

4.4.1.1. 10g、11g 版本

1. 建议检查备库审计日志级别，当备库审计日志级别设置较高时，会产生较多审计日志，占用服务器资源，可通过如下命令确认审计日志级别：

```
SQL> show parameter audit_trail;
```

- (1) 返回参数 VALUE 对应审计日志级别，包含 None、DB、OS 等

- ① None 为默认值，即不做审计
- ② DB 意指将 audit trail 记录在数据库的审计相关表中，例如：aud\$
- ③ OS 意指将 audit trail 记录在操作系统文件中

- (2) 若返回值为 None 和 DB 以外的级别时，可能产生较多审计日志

2. 软件需要对备数据库的连接用户进行必要的配置，若具体操作步骤如下：

- (1) 备库均创建 i2 用户：

```
SQL> create user i2 identified by i2;
```

- (2) 备库赋予 i2 用户权限：

```
SQL> grant select on sys.link$ to i2;
```

```
SQL> grant all on sys.seq$ to i2;
```

```
SQL> grant select on sys.user$ to i2;
```

```
SQL> grant dba to i2;
```

- 说明：

1. 同步规则中定义为源库的数据库节点，在反向同步规则中将变更为目标库，则原先被赋予的 select 权限不足，可能造成源库回切和反向同步任务的失败；

2. 故同步规则在执行备端接管操作之后，源库回切之前，应重新对反向同步规则中备库 i2 用户赋权。

```
[oracle@ecm-9b6c ~]$ sqlplus / as sysdba
SQL*Plus: Release 11.2.0.4.0 Production on Sun Dec 14 18:37:51 2025
Copyright (c) 1982, 2013, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.4.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL> create user i2 identified by i2;

User created.

SQL> grant select on sys.link$ to i2;

Grant succeeded.

SQL> grant all on sys.seq$ to i2;

Grant succeeded.

SQL> grant select on sys.user$ to i2;

Grant succeeded.

SQL> grant dba to i2;

Grant succeeded.

SQL> █
```

➤ 说明：若 dba 用户权限受阻，则可以更改为其余 SQL 语句，详见备库 Oracle 最小权限。

3. 完成 Oracle 源备数据库的配置。

4.4.1.2. 12c 及以上版本

1. 软件需要对备库进行必要的配置，具体操作步骤如下：

(1) 确保备库当前进入的是 CDB 根容器内：

```
alter session set container = CDB$ROOT;
```

(2) 查看备库当前 oracle12c 版本及其以后版本的数据库有哪些 PDB 容器：

```
SQL> show pdbs;
```

(3) 备库指定的 PDB 容器若未打开，需要将数据库打开：

```
SQL> alter pluggable database <PDB_NAME> open;
```

➤ 说明：<PDB_NAME>指的是 PDB 容器名。

(4) 备库进入到需要进行数据库复制的 PDB 容器：

```
SQL> alter session set container = <PDB_NAME>;
```

(5) 备库创建 i2 用户:

```
SQL> create user i2 identified by i2;
```

(6) 备库赋予 i2 用户权限:

```
SQL> grant select on sys.link$ to i2;
```

```
SQL> grant select on sys.user$ to i2;
```

```
SQL> grant all on sys.seq$ to i2;
```

```
SQL> grant all on sys.undo$ to i2;
```

```
SQL> grant dba to i2;
```

➤ 说明:

- 同步规则中定义为源库的数据库节点，在反向同步规则中将变更为目标库，则原先被赋予的 select 权限不足，可能造成源库回切和反向同步任务的失败；故同步规则在执行备端接管操作之后，源库回切之前，应重新对反向同步规则中备库 i2 用户赋权。
- 若 dba 用户权限受阻，则可以更改为其余 SQL 语句，详见备库 Oracle 最小权限。

(7) 完成 Oracle 备库数据库的配置。

4.5. 数据同步用户最小权限

4.5.1. 源端 Oracle 抽取用户最小权限

MDR 数据库数据同步通过源端数据库指定用户进行数据抽取，目标端数据库指定用户进行数据装载。假定源库指定连接用户为 i2，则需为 i2 用户授予数据同步的最小权限为:

```
grant connect to i2;
```

```
grant select any dictionary to i2;
```

```
grant select on dba_triggers to i2;
```

```
grant select on dba_tab_privs to i2;
```

```
grant select on dba_role_privs to i2;
```

```
grant select on v_$transportable_platform to i2;
```

```
grant select on v_$database to i2;
```

```
grant select_catalog_role to i2;
```

```
grant select on sys.link$ to i2;
```

```
grant select on sys.user$ to i2;
```

```
grant select on sys.seq$ to i2;
```

```
grant select on sys.undo$ to i2;
```

```
grant select on sys.access$ to i2;
```

```
grant select on sys.aq$_queue_table_affinities to i2;
```

```
grant select on sys.attrcol$ to i2;
```

```
grant select on sys.attribute$ to i2;
```

```
grant select on sys.aud$ to i2;
```

```
grant select on sys.ccol$ to i2;
```

```
grant select on sys.cdef$ to i2;
```

```
grant select on sys.clu$ to i2;
```

```
grant select on sys.col$ to i2;
```

```
grant select on sys.coltype$ to i2;
```

```
grant select on sys.com$ to i2;
```

```
grant select on sys.con$ to i2;
```

```
grant select on sys.deferred_stg$ to i2;
```

```
grant select on sys.defsubpart$ to i2;
```

```
grant select on sys.dependency$ to i2;
```

```
grant select on sys.disk_and_fixed_objects to i2;
```

```
grant select on sys.external_tab$ to i2;
```

```
grant select on sys.hist_head$ to i2;
```

```
grant select on sys.icol$ to i2;
```

```
grant select on sys.idl_ub1$ to i2;
```

```
grant select on sys.ind$ to i2;
```

```
grant select on sys.indcompart$ to i2;
```

```
grant select on sys.ind_online$ to i2;
```

```
grant select on sys.indpart$ to i2;
```

```
grant select on sys.indsubpart$ to i2;
```

```
grant select on sys.javaobj$ to i2;
```

```
grant select on sys.link$ to i2;
```

```
grant select on sys.lob$ to i2;  
grant select on sys.lobcomppart$ to i2;  
grant select on sys.lobfrag$ to i2;  
grant select on sys.mlog$ to i2;  
grant select on sys.mlog_refcol$ to i2;  
grant select on sys.ntab$ to i2;  
grant select on sys.obj$ to i2;  
grant select on sys.job$ to i2;  
grant select on sys.objauth$ to i2;  
grant select on sys.partcol$ to i2;  
grant select on sys.partlob$ to i2;  
grant select on sys.partobj$ to i2;  
grant select on sys.procedure$ to i2;  
grant select on sys.procedureinfo$ to i2;  
grant select on sys.profile$ to i2;  
grant select on sys.profname$ to i2;  
grant select on sys.recyclebin$ to i2;  
grant select on sys.seg$ to i2;  
grant select on sys.seq$ to i2;  
grant select on sys.settings$ to i2;  
grant select on sys.snap$ to i2;  
grant select on sys.source$ to i2;  
grant select on sys.subpartcol$ to i2;  
grant select on sys.sum$ to i2;  
grant select on sys.syn$ to i2;  
grant select on sys.sysauth$ to i2;  
grant select on sys.system_privilege_map to i2;  
grant select on sys.tab$ to i2;  
grant select on sys.tabcompart$ to i2;  
grant select on sys.table_privilege_map to i2;  
grant select on sys.tabpart$ to i2;
```

```
grant select on sys.tabsubpart$ to i2;  
grant select on system.aq$_queues to i2;  
grant select on system.aq$_queue_tables to i2;  
grant select on sys.trigger$ to i2;  
grant select on sys.ts$ to i2;  
grant select on sys.type$ to i2;  
grant select on sys.type_misc$ to i2;  
grant select on sys.user$ to i2;  
grant select on sys.view$ to i2;  
grant select on sys.v_$archived_log to i2;  
grant select on sys.v_$asm_diskgroup to i2;  
grant select on sys.v_$database to i2;  
grant select on sys.v_$datafile to i2;  
grant select on sys.v_$log to i2;  
grant select on sys.v_$logfile to i2;  
grant select on sys.v_$standby_log to i2;  
grant select on sys.v_$locked_object to i2;  
grant select on sys.v_$session to i2;  
grant select on sys.v_$version to i2;  
grant select on sys.v_$tempfile to i2;  
grant select on sys.v_$datafile to i2;  
grant select on sys.all_constraints to i2;  
grant select on sys.all_tab_cols to i2;  
grant select on sys.dba_clu_columns to i2;  
grant select on sys.dba_data_files to i2;  
grant select on sys.dba_db_links to i2;  
grant select on sys.dba_directories to i2;  
grant select on sys.dba_errors to i2;  
grant select on sys.dba_extents to i2;  
grant select on sys.dba_free_space to i2;  
grant select on sys.dba_jobs to i2;
```

```
grant select on sys.dba_log_group_columns to i2;  
grant select on sys.dba_log_groups to i2;  
grant select on sys.dba_objects to i2;  
grant select on sys.dba_part_indexes to i2;  
grant select on sys.dba_refresh to i2;  
grant select on sys.dba_role_privs to i2;  
grant select on sys.dba_scheduler_job_args to i2;  
grant select on sys.dba_scheduler_jobs to i2;  
grant select on sys.dba_segments to i2;  
grant select on sys.dba_sequences to i2;  
grant select on sys.dba_sys_privs to i2;  
grant select on sys.dba_tablespaces to i2;  
grant select on sys.dba_temp_files to i2;  
grant select on sys.dba_users to i2;  
grant select on sys.gv_$instance to i2;  
grant select on sys.v_$instance to i2;  
grant select on sys.gv_$session to i2;  
grant select on sys.gv_$process to i2;  
grant select on sys.gv_$lock to i2;  
grant select any table to i2;  
grant select on sys."_CURRENT_EDITION_OBJ" to i2;  
grant datapump_exp_full_database to i2;  
grant datapump_imp_full_database to i2;  
grant execute on dbms_metadata to i2;  
grant execute on xdb.dbms_csx_admin to i2;  
grant unlimited tablespace to i2;
```

4.5.2. 目标端 Oracle 装载用户最小权限

MDR 数据库数据同步通过源端数据库指定用户进行数据抽取, 目标端数据库指定用户进行数据装载。假定目标库指定连接用户为 i2, 则需为 i2 用户授予

数据同步的最小权限为:

```
grant connect to i2;
```

```
grant select on sys.link$ to i2;
```

```
grant select on sys."_CURRENT_EDITION_OBJ" to i2;
```

```
grant all on sys.user$ to i2;
```

```
grant all on sys.seq$ to i2;
```

```
grant all on sys.undo$ to i2;
```

```
grant create session to i2;
```

```
grant alter session to i2;
```

```
grant drop tablespace to i2;
```

```
grant become user to i2;
```

```
grant drop rollback segment to i2;
```

```
grant select any table to i2;
```

```
grant insert any table to i2;
```

```
grant update any table to i2;
```

```
grant drop any index to i2;
```

```
grant select any sequence to i2;
```

```
grant create role to i2;
```

```
grant execute any procedure to i2;
```

```
grant alter profile to i2;
```

```
grant create any directory to i2;
```

```
grant create any library to i2;
```

```
grant execute any library to i2;
```

```
grant alter any indextype to i2;
```

```
grant drop any indextype to i2;
```

```
grant dequeue any queue to i2;
```

```
grant execute any evaluation context to i2;
```

```
grant export full database to i2;
```

```
grant create rule to i2;
```

```
grant alter any sql profile to i2;
```

```
grant administer any sql tuning set to i2;
```

grant change notification to i2;

grant drop any edition to i2;

grant drop any mining model to i2;

grant alter any mining model to i2;

grant alter any cube dimension to i2;

grant create cube to i2;

grant drop any cube build process to i2;

grant read any file group to i2;

grant alter user to i2;

grant create any synonym to i2;

grant drop any trigger to i2;

grant grant any privilege to i2;

grant create any materialized view to i2;

grant drop any library to i2;

grant grant any role to i2;

grant alter resource cost to i2;

grant grant any object privilege to i2;

grant create synonym to i2;

grant create any job to i2;

grant create job to i2;

grant manage any queue to i2;

grant create any table to i2;

grant insert any cube dimension to i2;

grant drop any cube to i2;

grant drop any measure folder to i2;

grant alter rollback segment to i2;

grant delete any table to i2;

grant alter database to i2;

grant force any transaction to i2;

grant alter any procedure to i2;

grant drop any materialized view to i2;

grant under any type to i2;
grant alter any library to i2;
grant create dimension to i2;
grant debug any procedure to i2;
grant create rule set to i2;
grant alter any rule set to i2;
grant analyze any dictionary to i2;
grant alter any edition to i2;
grant create any assembly to i2;
grant alter any cube to i2;
grant select any cube to i2;
grant audit system to i2;
grant create any cluster to i2;
grant create any index to i2;
grant drop any sequence to i2;
grant drop any context to i2;
grant create evaluation context to i2;
grant audit any to i2;
grant create cluster to i2;
grant execute any program to i2;
grant execute any class to i2;
grant drop any table to i2;
grant create any cube dimension to i2;
grant create any measure folder to i2;
grant create cube build process to i2;
grant create sequence to i2;
grant create trigger to i2;
grant restricted session to i2;
grant create tablespace to i2;
grant alter tablespace to i2;
grant create user to i2;

grant lock any table to i2;
grant create view to i2;
grant drop any view to i2;
grant create type to i2;
grant execute any operator to i2;
grant create any dimension to i2;
grant alter any dimension to i2;
grant create any outline to i2;
grant administer database trigger to i2;
grant resumable to i2;
grant flashback any table to i2;
grant create any rule set to i2;
grant execute any rule set to i2;
grant import full database to i2;
grant execute any rule to i2;
grant create any edition to i2;
grant create assembly to i2;
grant alter any assembly to i2;
grant create cube dimension to i2;
grant create any cube build process to i2;
grant update any cube dimension to i2;
grant drop user to i2;
grant create any view to i2;
grant create any procedure to i2;
grant drop profile to i2;
grant drop any type to i2;
grant create any operator to i2;
grant global query rewrite to i2;
grant create any context to i2;
grant drop any outline to i2;
grant enqueue any queue to i2;

grant create table to i2;
grant create procedure to i2;
grant create any sequence to i2;
grant alter any role to i2;
grant analyze any to i2;
grant create indextype to i2;
grant under any table to i2;
grant drop any dimension to i2;
grant select any dictionary to i2;
grant create any evaluation context to i2;
grant drop any evaluation context to i2;
grant create any rule to i2;
grant create mining model to i2;
grant update any cube build process to i2;
grant execute any type to i2;
grant create rollback segment to i2;
grant create public synonym to i2;
grant create public database link to i2;
grant drop any role to i2;
grant alter any type to i2;
grant create database link to i2;
grant update any cube to i2;
grant alter system to i2;
grant comment any table to i2;
grant alter any index to i2;
grant drop public database link to i2;
grant create profile to i2;
grant alter any materialized view to i2;
grant under any view to i2;
grant execute any indextype to i2;
grant alter any outline to i2;

grant administer resource manager to i2;

grant manage scheduler to i2;

grant manage file group to i2;

grant create any mining model to i2;

grant select any mining model to i2;

grant delete any measure folder to i2;

grant backup any table to i2;

grant administer sql management object to i2;

grant drop any cluster to i2;

grant drop public synonym to i2;

grant drop any procedure to i2;

grant drop any directory to i2;

grant create any indextype to i2;

grant advisor to i2;

grant delete any cube dimension to i2;

grant create measure folder to i2;

grant drop any synonym to i2;

grant alter any sequence to i2;

grant force transaction to i2;

grant create any type to i2;

grant alter any operator to i2;

grant on commit refresh to i2;

grant debug connect session to i2;

grant drop any rule set to i2;

grant manage any file group to i2;

grant execute any assembly to i2;

grant execute assembly to i2;

grant comment any mining model to i2;

grant select any cube dimension to i2;

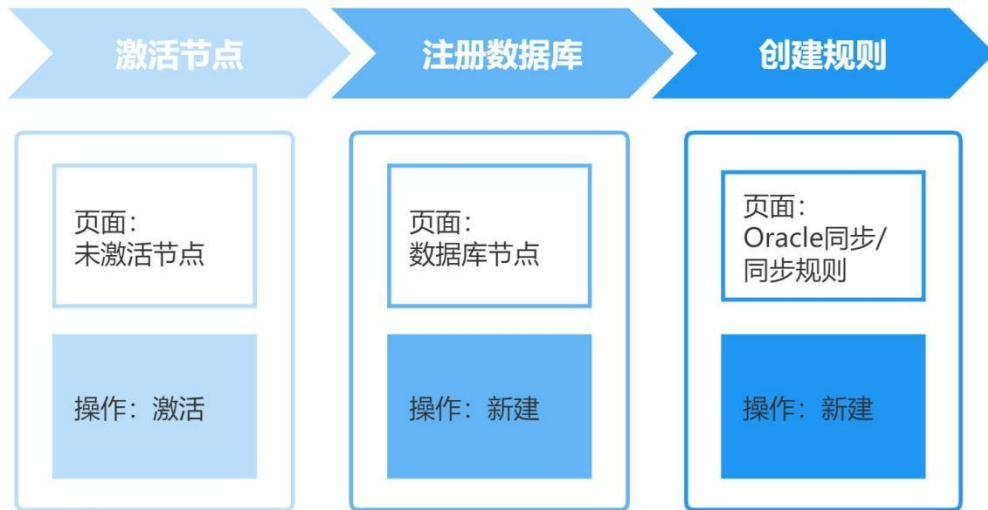
grant alter any trigger to i2;

grant create external job to i2;

```
grant create operator to i2;  
grant alter any table to i2;  
grant create any trigger to i2;  
grant create library to i2;  
grant merge any view to i2;  
grant administer sql tuning set to i2;  
grant insert any measure folder to i2;  
grant drop any operator to i2;  
grant drop any sql profile to i2;  
grant create any sql profile to i2;  
grant drop any cube dimension to i2;  
grant create any cube to i2;  
grant manage tablespace to i2;  
grant alter any cluster to i2;  
grant create materialized view to i2;  
grant query rewrite to i2;  
grant alter any evaluation context to i2;  
grant alter any rule to i2;  
grant drop any rule to i2;  
grant select any transaction to i2;  
grant drop any assembly to i2;  
grant flashback archive administer to i2;
```

5. 整体流程介绍

数据库双活整体流程包括：安装节点包并激活双活工作节点、注册数据库节点、创建同步规则三个部分，其中数据比较等为可选环节，非必要。整体数据同步流程示意如下：



6. 激活节点

本章介绍了控制台页面激活双活工作节点的过程，该操作为创建数据库双活同步规则界面操作的第一步。

● 使用条件

1. 至少有一台装有 MDR 节点软件的服务器。
2. 该服务器需要开启对应的 ia-helper 服务，且能与控制台正常通讯。

6.1. 界面

进入多活容灾服务控制台，点击操作平台的菜单栏：“资源同步管理”-“数据源”-“双活工作节点”，如下图所示：

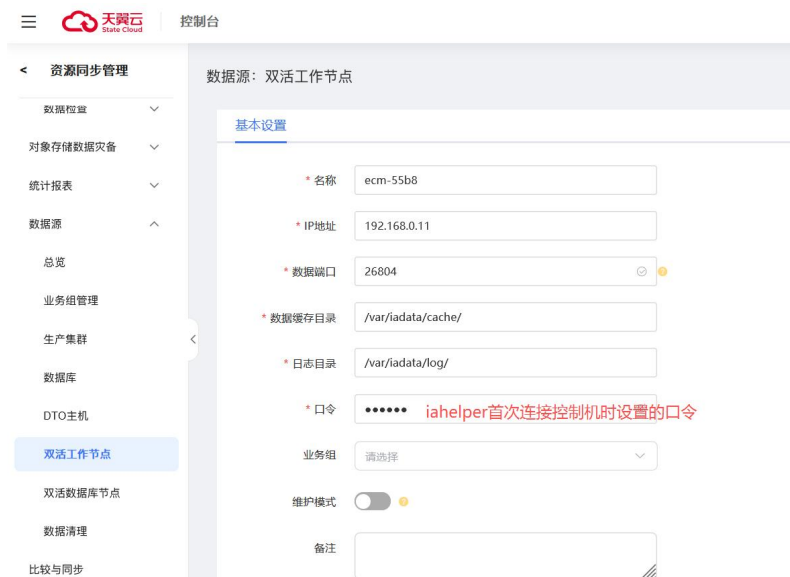


节点信息栏说明：

| 元件名称 | 注解 | 说明 |
|-------|------------------------------|-----------------------|
| 名称 | 自动识别并显示此双活工作节点上的hostname 的名称 | |
| 节点状态 | 未激活 | 此双活工作节点尚未激活 |
| | 残留节点 | 此双活工作节点此前已被激活，仍残留部分信息 |
| IP 地址 | 显示此双活工作节点的 IP 地址 | |
| 数据端口 | 显示此双活工作节点连接控制台所使用的端口号 | |
| 操作系统 | 显示此双活工作节点的操作系统版本 | |
| 软件版本 | 显示此双活工作节点中 MDR 的软件版本号 | |

6.2. 激活操作

正确启动 iahelper 服务，且能与控制台正常通讯的双活工作节点，会自动显示在“双活工作节点”页面，需要对其进行“激活”操作。单击操作列“激活”按钮，页面跳转至下图“双活工作节点·节点激活”设置页面，进行激活配置，如下图：



控制台

资源同步管理

数据源：双活工作节点

基本设置

* 名称

* IP地址

* 数据端口

* 数据缓存目录

* 日志目录

* 口令 iahelper首次连接控制机时设置的口令

业务组

维护模式

备注

“双活工作节点·节点激活”设置页面说明:

| 元件名称 | 注解 | 说明 |
|--------|---|--|
| 名称 | 用户自定义的双活工作节点的名称，便于管理，支持中文和英文字符，区分和识别当前双活工作节点的名称 | |
| IP 地址 | 自动识别并显示当前服务器的 IP 地址 | 如果此服务器含有多个 IP 地址，则会界面上会一一显示，IP 地址会自动用逗号“，”隔离开。需要保留 1 个 IP 地址用来通讯，删除其他地址。 |
| 数据端口 | 会自动显示当前双活工作节点所使用的数据端口。双活工作节点有 10 个服务程序，同一个节点最多会需要运行所有这些服务，它们会使用由数据端口开始的 10 个端口。通常情况采用默认端口即可，但如果端口被占用，比如多套软件同时运行的场景的情况，则需要调整，调整此参数需要重启节点上的服务 | 所有节点的端口默认都由该端口递增获得，您也可以通过点击下方节点按钮后面的修改图标来自定义节点端口。 <数据端口>指的是界面上用户自定义的数据端口的填写。 用户自定义的数据端口必须在 1-65535 之间，否则会提示“端口不合法”。此数据端口的值需要用户在底层修改监听端口的配置文件 |
| 数据缓存目录 | 存放当前服务器的数据缓存信息的绝对路径。路径以文件分割符（“/” | 确保当前服务器中运行 MDR 软件服务的用户对数据缓存目 |

| | | |
|------|---|---|
| | 或“\”) 结尾 | 录有读写权限。 |
| 日志目录 | 存放当前服务器的日志信息的绝对路径 | |
| 口令 | 一般默认即可。自动获取并显示当前双活工作节点的 UUID，用户也可以根据自己需求进行自定义更改，此 UUID 主要作用是作为区分不同的节点的标识。 | 用户若修改了此 ipctoken，则需要确保当前注册的双活工作节点 UUID 是唯一的，和其余的双活工作节点的 UUID 不一致。 |
| 业务组 | 支持双活工作节点关联业务组。 | |
| 维护模式 | 开启维护模式，该节点将跳过告警监控，忽略异常 | |
| 备注 | 用户可以自定义填写此双活工作节点的备注 | |
| 确定 | 点击“确认”，完成双活工作节点的设置并提交 | |
| 取消 | 点击“取消”，取消激活该双活工作节点的操作 | |

● 注意

1. 数据缓存目录和日志目录必须以文件分割符 (“/”或“\”) 否则会提示“结尾输入的数据缓存目录不正确”，从而无法添加双活工作节点。
2. MDR-Active 的数据缓存目录**强烈建议**不要放置在共享目录中，否则会极大降低装载效率。
3. 节点类型错误不会导致节点服务出错，但会影响 WEB 界面的状态监控。

4. 用户若修改数据端口需要**重启节点服务程序**。
5. 每个操作系统用户可以注册一个节点，每个节点有自己的 UUID，数据库双活通过 UUID 区分不同的节点。
6. ia-helper 服务程序通过配置目录下的 active.conf 文件判断自身节点是否已经注册。当需要重新注册节点时，**请确保该文件已被删除**。
7. ia-helper 会主动连接控制机，当无法激活或找不到双活工作节点时，首先检查网络是否通畅，然后尝试重新启动 ia-helper 程序。如若还是检测不到，需要注意是否是版本差异问题，强烈建议节点版本和控制机版本保持一致。
8. 需要确保操作系统内，运行服务的用户能对节点的缓存目录和日志目录有读写权限。

双活工作节点信息栏说明：

| 元件名称 | 注解 | | 说明 |
|-------|--------------------------------------|----------------|-------------------------------------|
| 名称 | 显示用户激活此双活工作节点时自定义的名称，便于管理，支持中文和英文字符。 | | |
| 节点状态 | 在线 | 当前双活工作节点为在线状态。 | 取决于 ia-helper 服务的状态。 |
| | 离线 | 当前双活工作节点为离线状态。 | |
| 进程状态 | | 显示进程 | 显示此双活工作节点类型对应的服务进程名和状态。绿色为正常，红色为异常。 |
| IP 地址 | 显示当前双活工作节点的 IP 地址 | | |
| 操作系统 | 显示当前双活工作节点的操作系统及版本号 | | |
| 软件版本 | 显示此双活工作节点中 MDR 的软件版本号 | | |
| 所有者 | 将此双活工作节点添加搭配 MDR-Active 的用户 | | |

| | | |
|------|------------------------|--|
| 维护模式 | 开启维护模式，该节点将跳过告警监控，忽略异常 | |
|------|------------------------|--|

双活工作节点操作栏说明：

| 元件名称 | 注解 | 说明 |
|------|-----------------------------------|----------------|
| 修改 | 修改当前双活工作节点的配置信息，同节点激活时的配置步骤一致。 | 跳转至双活工作节点页面 |
| 调试信息 | 下载当前双活工作节点的日志文件 | |
| 授权 | 将当前双活工作节点的不同操作权限授予其他MDR-Active 用户 | |
| 更新公钥 | 更新节点的公钥 | |
| 重启进程 | 重新节点进程 | 支持单一/部分/全部进程重启 |
| 删除 | 删除当前双活工作节点 | |

7. 数据库节点


新建数据库节点为创建数据库双活同步规则界面操作的第二步。

- 前提条件

1. 相应节点已注册为双活工作节点。
2. 对应的数据库已完成相应配置。

7.1. 购买数据库双活许可

- 操作步骤

1. 登录天翼云，进入[控制中心](#)。
2. 单击控制中心顶部的 ，选择“区域”。
3. 在服务列表选择“计算” - “多活容灾服务”，进入[多活容灾服务控制台](#)。
4. 点击左侧菜单栏 - “资源同步”模块，进入资源管理模块页面。
5. 点击左侧菜单栏 - “数据库双活”，点击“许可”，进入许可页面。
6. 点击右上角“购买数据库双活许可”按钮，弹出购买许可弹窗。按需购买许可。



7. 填写数据库类型、购买数量和时长，勾选已阅读并同意相关协议后，点击“购买”按钮，完成许可支付。



7.2. 新建数据库节点

● 前提条件

1. 默认 sys 用户创建的表不能同步。

2. 配置在数据源节点的 DB 用户作为同步链路用户时, 其创建的表将无法被同步。

● 操作步骤

1. 点击左侧菜单栏 - “数据源” - “双活数据库节点”。点击“新建”按钮, 页面跳转至 “数据库节点·新建” 页面, 可进行数据库节点配置, 如下图:



“Oracle 数据库节点·新建”页面说明:

| 元件名称 | 注解 | 说明 |
|-------|----------------------------------|----|
| 名称 | 用户自定义的数据库节点的名称, 便于管理, 支持中文和英文字符。 | |
| 工作模式 | 单机 | |
| 工作节点 | 选择添加至控制台内的工作节点 | |
| 数据库类型 | 选择 Oracle | |
| 角色 | 选择“源库”或者“备库” | |

| | | |
|-------|---|--|
| 软件许可 | <p>1、许可为 Oracle 类型</p> <p>2、选择描述为数据库双活的许可，需包含"Oracle"且可用数量\geq1；</p> | <p>1、如果用户尚未通过“数据库双活”→“许可”购买有效 License，该选项下拉框显示为空；</p> <p>2、如果已经购买有效 License，下拉框将显示所有可用的 License。</p> |
| 业务组 | 数据库节点所属的业务组。 | 在“资源管理”->“业务组管理”中管理业务组。 |
| 备注 | 可自定义该数据库节点的备注 | |
| 维护模式 | 默认关闭。打开开关，该节点相关的告警会被忽略 | |
| 用户名 | Oracle 数据库远程用户 | |
| 密码 | Oracle 数据库远程用户的密码 | |
| 服务名 | Oracle 数据库对应的服务名。 | <p>1、在 Oracle 数据库机器上使用 grid 或 oracle 用户执行以下命令进行查看：lsnrctl status，找到对应的 Service 进行填写。</p> <p>2、在 Oracle 数据库，执行语句查询： SELECT name FROM v\$services;</p> |
| 数据字符集 | 默认不填写。 | 仅在源机和目标机的数据字符不一致的时候，需要进行选择 |
| 端口 | Oracle 数据库端口号，默认 | |

| | | | | |
|------|----|----------------------------|--------------------------------|---|
| | | | 为 1521, 管理员可以根据实际情况自行修改。 | |
| 部署方式 | 单机 | 数据库 IP | Oracle 数据库的 IP 地址 | |
| | | 线程号 | 数据库实例的线程 ID, 可以在实例所在的机器上进行查询。 | |
| | 集群 | 数据库配置 | 单击“添加”按钮, 可填写多个节点 IP | |
| | | 数据库 IP | 各 Oracle 数据库节点的 IP 地址。每行仅可填写一个 | |
| | | 线程号 | 数据库实例的线程 ID, 可以在实例所在的机器上进行查询。 | <p>1、RAC ADG 主库-单机 ADG 环境, 若选用单机 ADG 备库作为同步规则源库, 源库也需创建为集群, 数据库 IP 填写为相同值, 线程号需执行以下查询语句, 查看 thread# 返回:</p> <pre>SQL>select GROUP#,THREAD#,STATUS from v\$log;</pre> <p>2、非上述源库环境, 则进入到数据库中下输入命令, 查看 thread# 返回:</p> <pre>SQL> select instance_name, host_name, thread# from v_\$instance;</pre> <p>找到对应的 thread# 的值即为此线程号。</p> |
| | 禁用 | 只针对集群选项设置, 默认为否, 表示节点数据库正常 | | |

| | | | | |
|---------|----------------|---------|--|---|
| | | | 工作。选择时，软件将不分析该节点的数据库日志 | |
| | | 操作 | 可删除当前的数据库配置 | |
| 读取日志线程数 | | | 1、默认 1，单线程抽取 2、可配置为多线程抽取，加快日志读取速度 3、最大值 128，需结合工作节点可用线程数和数据库实际压力调整此值 | 如已创建相关的规则，修改后，需在规则操作“停止规则并停止解析”后再启动规则，使修改后的配置生效 |
| 日志读取方式 | 文件系统/裸设备 | | 此项配置需要同数据库配置一致，非 ASM 单实例的 Oracle 会采用文件系统，Oracle10g 可能会出现裸设备的情况。 | |
| | | os 认证 | 开启 os 认证，则不需要输入 ASM 密码。 | 强烈建议不开启，在复杂环境中，os 认证的会导致有些情况无法读取到更新的日志，从而产生较大延迟。 |
| | ASM 实例/ASM 块设备 | ASM 服务名 | 此 Oracle 数据库对应的 ASM 服务名。 | 1、进入 grid 用户下，使用 lsnrctl status 命令，找到对应的 Service，查看对应 ASM 服务名。 2、ASM 环境监听 ASM 的服务和 Oracle 的服务，ASM 服务名一般为+ASM 等格式 |
| | | ASM 用户名 | sys | 指的是 grid 用户下 ASM 管理的用户名，建议使用 sys 用户。亦可自建用户。 自建用户的操作如下： su - grid |

| | | | |
|--------|------|--|---|
| | | | <pre>sqlplus / as sysasm SQL> create user i2 identified by Info1234; SQL> grant sysasm to i2;</pre> |
| | | ASM 密 码 sys 用户对应的 ASM 密码 | 如若 sys 用户或自建用户的密码忘记, 则可根据客户的允许后, 采用强制修改密码的操作进行修改。 强制修改密码的操作如下: <pre>su - grid asmcmd ASMCMDB> orapwusr --modify 用户名</pre> |
| | | ASM 端 口 ASM 管理监听端口, 默认为 1521, 管理员可以根据实际情况自行修改。 | 1、界面测试连接失败时, 可于底层数据库测试连接, 验证填写内容是否正确: 2、su - grid 3、sqlplus ASM 用户名/ASM 用户密码@数据库 ip:ASM 端口/ASM 服务名 as sysasm |
| | | 文件打开方式 | 分为普通和 DIRECT。默认 DIRECT。 1、块设备一般选 DIRECT, 个别环境下, 文件系统的库也需要选择 DIRECT 2、如果 iatrack 日志持续报 can' t find xid 错误, 可以尝试 DIRECT, 并重新提交规则, 重新同步。 |
| 远程文件代理 | 代理端口 | 用户根据 logreader 命令行的启动数值进行填写 | 1、采用远程部署, 源库启动 logreader 服务的数据库节点必须开启此按钮, 且正确填写 |
| | 压缩方式 | 根据压缩方式的不同, 分为 | |

| | | | |
|----------|--|--|---|
| | | “不压缩”、“LZO”和“ZIP”。 | 相关配置 |
| 中继节点 | | 同一源库的数据需同步至多个备库时，源库需开启此开关 | 只有工作节点勾选了“中继节点”，才可在下拉框中被选择。 |
| 会话过滤 | | 默认不开启。此选项针对双向同步，双向同步时需要勾选源库和备库的会话过滤。 | |
| ADG 环境配置 | | <p>1、分为三种：“非 ADG 数据库”、“ADG 环境中的数据库容器 (CDB)”和“ADG 环境中的可插拔数据库 (PDB)”。</p> <p>2、通常情况下配置为普通数据库即可，即非 ADG。</p> | <p>1、其余配置项专门针对于用户对容器 DB 做 ADG，且将 ADG 备库做为 ACTIVE 源库的情况，在 ADG 备库的 PDB 实例中，对某些视图的查询结果是错误的，因此 ACTIVE 需要 CDB 的配置，以便在 CDB 实例中做相应查询。</p> <p>2、此种情况下，需要先配置 CDB 库节点，不占用 license，不可以用在同步规则中；然后再配置 PDB 库节点，同时选择其所属 CDB。</p> |
| 测试连接 | | 测试界面上所有填写数据的数据库节点数据判断是否可以连通。 | <p>1、连接成功，页面会返回“连接成功”。</p> <p>2、若返回错误码，请检查对应的数据库监听等各项配置及填写信息正确与否。</p> |
| 提交 | | 单击“提交”，完成 Oracle 数据库节点的新建 | |
| 取消 | | 取消本次新建数据库节点操作，返回数据库节点页面 | |

7.3. 数据库节点界面

完成数据库节点配置，点击“提交”后，页面自动返回“数据库节点”页面，如下图所示：



双活数据库节点页面说明如下：

| 元件名称 | 注解 | |
|--------|------------------------------------|----------------------------------|
| 名称 | 显示用户创建此数据库节点时自定义的名称，便于管理，支持中文和英文字符 | |
| 节点状态 | 在线 | 当前数据库节点为在线状态，可访问 |
| | 离线 | 当前数据库节点为离线状态，不可访问 |
| | 未知 | 无法获取到数据库节点的信息 |
| | 异常 | 访问数据库节点异常，即按照页面填写的数据库配置信息访问数据库有误 |
| 类型 | 显示用户创建此数据库节点时此数据库节点的数据库类型。 | |
| 日志读取方式 | 显示当前获取到此数据库的日志读取方式。 | |
| 工作节点 | 显示此数据库节点对应的双活工作节点 | |
| 版本 | 显示此数据库节点中数据库软件的版本号。 | |
| 所有者 | 显示将此数据库节点添加至 MDR-Active 控制台的用户 | |
| 维护模式 | 默认为关。开启维护模式，该节点将跳过告警监控，忽略异常。 | |

双活数据库节点操作栏说明：

| 元件名称 | 注解 | 说明 |
|------|-------------------|-------------|
| 修改 | 修改当前数据库节点的配置信息。可修 | 跳转至数据库节点修改界 |

| | | |
|----|---|---|
| | 改信息和节点激活时的配置项一致。 | 面 |
| 删除 | 删除当前数据库节点。跳转至删除确认界面，点击“确定”，完成删除操作，点击“取消”，可取消删除操作。 | 若当前数据库节点存在关联的同步规则，是无法删除的；若执意删除，需要先删除数据库节点对应的规则，再删除此数据库节点。 |

8. 同步规则

● 前提条件

1. 源端和备端数据库进程所在的双活工作节点对应的节点状态都必须为在线状态，对应的进程状态都为正常。
2. 源端备端数据库节点对应的状态必须为正常状态。

8.1. 创建同步规则

● 操作步骤

1. 点击左侧菜单栏 - “数据库双活” - “同步规则”，进入同步规则界面。
2. 同步规则界面中，单击“新建”，进入 Oracle 同步规则配置界面。

数据库双活：同步规则

基本设置 全量设置 增量设置 压缩加密 高级设置 全量对象过滤设置 增量DDL过滤设置

规则提交后立即启动

名称

源端类型

源端数据库

备端类型

备端数据库

映射方式

行映射方式

共享进程

维护模式

业务组

备注

| 元件名称 | 注解 | 说明 |
|-------|--|--|
| 名称 | 用户自定义的同步规则名称，便于管理，支持中文和英文字符，区分和识别当前任务的名称 | |
| 源端类型 | 选择 oracle | |
| 源端数据库 | 用户根据已添加至数据库节点中的 oracle 数据库，选择作为源端的机器 | |
| 备端类型 | 选择 oracle | |
| 备端数据库 | 用户根据已添加至数据库节点中的 oracle 数据库，选择作为备端的机器 | |
| 类型 | 非整库映射 添加 | 1、点击添加按钮添加库，系统默认目标端与源端相同库名； 2、如果目标端想同步为不同的库 |
| | | 源备两端库之间的数据同步，支持不同库名的映 |

| | | | | |
|-----|--|------|--|---|
| | | | 名，可自行修改目标端对应的同步库名 | 射。 库名输入区分大小写，即数据库中库名为大写的，请在库映射中填写大写，如果是小写库名，则输入小写。 |
| | | 选择用户 | 通过搜索快速选择同步用户 | |
| | | 编辑 | 1、适用于大批量用户的添加。 2、编辑格式为： {源端数据库类型，源端数据库节点名，源端数据库用户，源端数据库名}{备端数据库类型，备端数据库节点名，备端数据库用户，备端数据库名} 多个库需采用换行处理 | |
| 表映射 | | 添加 | 点击添加，手动填写库、表进行同步 | 1.源备两端数据表之间的数据同步，可以全表同步，也可以抽取源端表中部分字段同步到目标端。表映射支持不同用户名、不同表名及不同字段名进行映射。 2.表映射表名支持正则表达式， |
| | | 选择表 | 通过搜索快速选择同步表 | |
| | | 编辑 | 1、适用于大批量用户的添加。2、编辑格式为： 源端用户,源端表名,备端用户,备端表名 多个表需采用换行处理。 | |

| | | | |
|----|------|----------------|--|
| | | | <p>包含元字符\、?、^、\$、*、+、.点、 、[a-z] 及 {n,m} 等，但不支持中文。如表名填写为正则表达式时，备端表名将始终与源端保持一致，不支持不同表名映射。</p> <p>3、使用表映射功能，不需要事先在目标端创建需要同步的数据表。</p> <p>用户名、表名、列名的输入区分大小写</p> |
| | 整库映射 | 源备两端的整库同步 | |
| 提交 | | 完成规则配置项后，可提交规则 | |
| 取消 | | 取消本次新建比较规则的操作 | |

3. 全量设置页面各项配置如下:

全量同步

全量同步数据源

全量装载方式

全量导出线程数

全量装载线程数

全量同步前清除用户

表覆盖策略

全量错误处理方式

尝试并发同步分区表

并发同步表 表名可不填

| 源端用户 | 表名 | 操作 |
|------|----|----|
|------|----|----|



| 元件名称 | 注解 | 说明 |
|---------|---|-----------------------|
| 全量同步 | 1、用户根据自己需求选择是否进行全同步 2、如果关闭此开关，则需要用户自行输入“起始 SCN”号来决定。 3、关闭此开关，点击获取 SCN，则自动填写为当前数据库 SCN 号 | 反向同步规则为仅增量的规则，该开关默认关闭 |
| 全量同步数据源 | 1、非必选，默认使用规则源库 全量 2、当全量和增量为不同的源库， | |

| | | |
|------------|--|--------------|
| | 即全量需使用非规则源库时, 再配置该项 | |
| 全量装载方式 | <p>1、下拉框可选直路径和普通方式。</p> <p>2、Oracle 加载数据的方法, 选择直路径加载数据快, 普通路径加载数据稳定。</p> <p>3、不是所有的表都可以用直路径装载, 全同步过程中, active 会对不支持直路径的表自动采用普通方式装载。</p> <p>4、如果遇到直路径装载错误, 需要修复表, 可以在修复表之前将规则调整为普通装载方式。</p> | |
| 全备导出线程数 | <p>1、默认为 1, 通过增加线程数, 同时导出多张数据表, 可以提高全量同步的性能;</p> <p>2、管理员可以根据实际业务量进行修改, 控制机界面取消输入限制和校验, 底层限制提高至 128。</p> | 最大线程数修改为 128 |
| 全备装载线程数 | <p>1、默认为 1, 通过增加线程数, 同时装载多张数据表时, 可以提高全量同步的性能;</p> <p>2、管理员可以根据实际业务量进行修改, 控制机界面取消输入限制和校验, 底层限制提高至 128。</p> | |
| 全备前是否先清除用户 | <p>默认为否;</p> <p>选择“是”将在同步前清除备端</p> | |

| | | | |
|-----------|--|-------------------------------|---|
| | | 中与源端同名的用户，此时需将备端用户退出才可完成同步规则。 | |
| 表覆盖策略 | 保留表数据 | 保留目标表结构和原始数据，将源端表数据添加至表中 | |
| | truncate 表 | 删除目标表的原始数据但是保留表结构 | |
| | 删除到回收站 | 删除目标端中表结构和内容，保留至回收站，之后可恢复 | |
| | 彻底删除 | 彻底删除目标端中表结构和内容 | |
| 尝试并发同步分区表 | 勾选此选项，数据库双活将尝试按分区并发处理分区表 | | 1、规则界面若仅打开“尝试并发同步分区表”开关，未填写并发同步表设置，程序也会自动并发导出表 |
| 并发同步表 | <p>1、此处可指定需要并发处理的表，注意区分大小写，表名可支持正则表达式，包含元字符\、?、^、\$、*、+、.点、 、[a-z]及{n,m}等，但不支持中文。</p> <p>2、此项功能对于大 LOB 表导出有帮助，但装载速度可能会下降，并发装载的表会自动变为普通方式装载。对于特别大的表，如果全同步导出过程中发生快照过老的错误，要尝试加大 undo 表空间，调整 Undoretention 来避免，如果仍旧不能解决问题，也可以配置到并发装载表。以上两个选项要结合 min_table_piece 配置使用</p> | | <p>2、新建规则后，程序会自动记录表的导出时间，供下次并发全同步或者修复表判断用；</p> <p>3、表导出时间记录在本地缓存目录的 tablestat 目录，文件以 UUID 组织。</p> <p>4、新建规则 min_table_piece 默认值为 102400，如果表占用的数据块数超过此参数值，则将表拆分成多片，并行处理；重新全同步，该参数意指表导出花费的时间，超过参数设置的表会分片处理，默认值 300s</p> |

4. 增量设置页面各项配置如下：

增量同步

增量装载线程数

备端对删除的表进行备份

转换更新主键操作

同步LOB数据

补全更新操作LOB数据

序列校准

定期在源库产生空事务

存储设置

源端磁盘缓存上限(MiB)

事务最大内存占用(KiB)

最大跟踪文件大小(MiB)

备端保留增量导出数据时长 (0-8760)小时

| 元件名称 | 注解 | 说明 |
|---------------|--|--------------------------|
| 增量装载线程数 | 默认为 1，通过增加线程数，同时装载多张数据表，可以提高增量同步的性能 | |
| 备端对删除的表进行备份 | 1、默认为否；该选项用于目标端对删除的表进行备份，防止误删除。 2、备份的表名类似于 _i2_<原表名>。 | |
| 转换更新主键操作 | 默认为不转换。 该项针对 Oracle->kafka，下拉框可选项“删除+插入” | |
| 是否同步 LOB 数据 | 1、默认同步，可修改 | |
| 补全更新操作 LOB 数据 | 1、默认为否。 2、是，在 Lob 表更新时，不论更新操作本身是否修改 lob 字段，都会 | 关闭此开关且开启全字段附加，Lob 表更新任意字 |

| | | | |
|------|------------|--|---|
| | | 从数据中取出所有 Lob 数据，并将表中的 Lob 数据同步至备端 | 段均不附加其余 Lob 字段 打开此开关但不开启全字段附加，更新 LOB 表任意字段均附加所有 Lob 字段 |
| | 序列校准 | <p>1、默认为否。</p> <p>2、选择“否”，只同步序列水位线，等到接管后再调整序列值。</p> <p>3、选择“是”，源备两端的序列值时刻保持同步。</p> | |
| | 定期在源库产生空事务 | <p>1、默认为否。</p> <p>2、10g 的库在没有业务的情况下，不会产生事务，基于规则的表对比会受此影响，不能正常对比；</p> <p>3、另外，生产是 RAC 的情况下，单个节点静止会影响日志解析。此功能打开将有助于解决上述问题。</p> | |
| 存储设置 | 源端内存缓存上限 | 用于设定 iatrack 进程的内存占用上限 | |
| | 源端磁盘缓存上限 | 用于设定源端软件所可以占用源端磁盘资源的最大上限，一旦超过这个限制，软件会自动停止运行 | |
| | 事务最大内存上限 | 用于设定 iatrack 进程每个事务的内存占用最大值，超过此值的事务将会形成临时事务文件存储在磁盘上 | |
| | 最大跟踪文件大小 | 用于设定 iatrack 进程产生的单个缓存文件大小 | |

| | | | |
|------|--------------|--|-----------------|
| | 增量装载队列内存上限 | 增量装载队列内存可配置的功能 | |
| | 备端保留增量导出数据时长 | <p>1、默认 0,即备端不保留增量导出数据 TF 文件，不生成 TF 文件的备份目录 Back/ActDataBak，备库装载完成数据后，存放 TF 文件的目录 Back/ActData 下的数据文件将直接删除</p> <p>2、填写该项后，备端缓存目录下将生成备份目录： 规则 uuid/Back/ActDataBak，用于存放备端导出数据的备份。正常存放 TF 文件的目录为 规则 uuid/Back/ActData，当该目录生成的 TF 文件大于 4 时，将生成 ActDataBak 目录，该目录仅保留最近的 4 个 TF 文件，其余按照 TF 文件序号顺序，待完成装载后，从 0 开始移动到 ActDataBak 目录下</p> <p>3、TF 文件移动到 ActDataBak 目录时刻，即为数据保留时长的起始时间，每当 ActDataBak 备份目录新增 TF 文件时，均会校验该目录文件保留时长，超过页面设置时间的将做清除处理</p> | |
| 错误处理 | 插入时记录冲突 | 源端插入数据，同步到目标端时，与目标端数据表里的数据重复，处理的方法是把目标端的重复数据删除，然后重新插入同步的数据 | |
| | | 转换为 | 把同步的数据，以更新的方式替换 |

| | | | | |
|--|----------|-------------|--|--|
| | | 更新 | 目标端重复的数据 | |
| | | 报错 | 不做处理，在规则日志中报错 | |
| | | 按指定字段更新大值插入 | 主键表在同步时，如果备库的主键数据通过其他方式已经存在，源库插入的主键数据同步到备库会与已经存在的主键产生冲突。他可以通过设置指定字段的大小值获胜策略，对指定的字段数据进行比对。如果插入的数据获胜，则更新到备库中，否则保留备库中的数据不变。 | |
| | | 按指定字段更新小值插入 | 主键表在同步时，如果备库的主键数据通过其他方式已经存在，源库插入的主键数据同步到备库会与已经存在的主键产生冲突。他可以通过设置指定字段的大小值获胜策略，对指定的字段数据进行比对。如果插入的数据获胜，则更新到备库中，否则保留备库中的数据不变。 | |
| | | 执行指定存储过程 | 主键表在同步时，如果备库的主键数据通过其他方式已经存在，源库插入的主键数据同步到备库会与已经存在的主键产生冲突。可以通过设置在发生插入冲突时执行指定存储过程。 | |
| | 更新时未找到记录 | 插入 | 源端更新数据，同步到目标端时，在目标端数据表里没有找到相匹配的数据而无法更新，处理的方法是插入同步的数据 | |
| | | 报错 | 不做处理，在规则日志中报错 | |

| | | | | |
|-----------|------------|---|--|-------------------------|
| | 删除时未找到记录 | 忽略 | 源端删除数据，同步到目标端时，在目标端数据表里没有找到相匹配的数据而无法删除，处理的方法是忽略不计 | |
| | | 报错 | 在规则日志中报错 | |
| | 装载错误处理方式 | 继续装载 | 忽略同步错误的的数据，继续同步此表及其它数据表 | |
| | | 停止装载 | 同步数据出现错误，停止所有同步 | |
| | | 停止此表装载 | 同步数据出现错误，停止同步此表，继续同步其它数据表 | |
| 上传 DML 错误 | | 是否上传增量 DML 装载错误，可以在规则装载信息，增量失败 DML 中查看。 | | |
| DML 追踪设置 | 是否保留删除记录 | | 1、默认为否。 2、选择是，增量时源端删除的数据，备库表仍保留，不执行删除操作。 | |
| | 自增列字段名称 | | 1、自定义名称，可为空 2、填写字段名称后，将在备库同步表和审计表中增加此字段 3、此值默认为 1，备库表每增加一行数据，此数值自增 1 | 仅支持备库为 12c 及以上版本 |
| | 操作产生时间字段名称 | 仅时间 | 1、自定义名称，可为空 2、填写字段名称后，将在备库同步表和审计表中增加此字段 3、数据类型为 VARCHAR2(9)，数据为 hh:mm:ss | 1、操作产生时间指源库 DML 操作产生的时间 |
| 仅日期 | | 1、自定义名称，可为空 2、填写字段名称后，将在备库同步 | | |

| | | | | |
|----------|------------|--|---|--|
| DML 追踪设置 | | | 表和审计表中增加此字段 3、数据类型为 VARCHAR2(11),数据为 YYYY-MM-DD | |
| | | 时间和日期 | 1、自定义名称, 可为空 2、填写字段名称后, 将在备库同步表和审计表中增加此字段 3、数据类型为 DATE, 数据为 YYYY-MM-DD hh:mm:ss | |
| | 操作装载时间字段名称 | 仅时间 | 1、自定义名称, 可为空, 填写字段名称后 2、将在备库同步表和审计表中增加此字段 3、数据类型为 VARCHAR2(9),数据为 hh:mm:ss | 1、操作装载时间指备库 DML 操作装载的时间 |
| | | 仅日期 | 1、自定义名称, 可为空 2、填写字段名称后, 将在备库同步表和审计表中增加此字段 3、数据类型为 VARCHAR2(11),数据为 YYYY-MM-DD | |
| | 时间和日期 | 自定义名称, 可为空, 填写字段名称后, 将在备库同步表和审计表中增加此字段, 数据类型为 DATE,数据为 YYYY-MM-DD hh:mm:ss | | |
| | 操作类型字段名称 | 自定义名称, 可为空, 填写字段名称后, 将在备库同步表和审计表中增加此字段, 数据类型为 VARCHAR2(20) | | |
| | 标记字段值 | 插入 | 自定义名称, 默认为 I, 不建议为空 | 1、操作类型字段用于标识 DML 操作,包含插入、更新、删除及更新主键,填写该项后标记字段值方生效。 |
| | | 更新 | 自定义名称, 默认为 U, 不建议为 | |

| | | | | |
|----|----------|-------|--|--|
| | | | 空 | |
| | | 删除 | 自定义名称，默认为 D，不建议为空 | |
| | | 更新主键 | 自定义名称，默认为 UPK，不建议为空 | |
| | 是否启用审计表 | 审计表前缀 | 同步表名增加前缀，用于审计表名 | <p>1、默认关闭，开启此开关后，在备库生成审计表</p> <p>2、审计表包含同步表本身的字段，DML 追踪设置填写的字段，此外还增加了 SCN VARCHAR2(20)和 BEFORE_AFTER CHAR(1)两列</p> <p>，</p> <p>BEFORE_AFTER 用于记录表更新前后的值，更新前 BEFORE_AFTER 列值为 B，更新后此值为 A</p> <p>3、修改此 DML 追踪设置后，规则需重新全量，除此之外支持增量时生成审计表。</p> |
| | | 审计表后缀 | 同步表名增加后缀，用于审计表名 | |
| 调试 | 保留装载失败事务 | | <p>1、默认为否。</p> <p>2、选择“是”，则在备端保留装载</p> | |

| | | | |
|--|--|-------|--|
| | | 失败事务。 | |
|--|--|-------|--|

5. 压缩加密页面各项配置如下:

基本设置 全量设置 增量设置 **压缩加密** 高级设置 全量对象过滤设置 增量DDL过滤设置

传输压缩

传输加密

加密类型

| 元件名称 | | 注解 | 说明 |
|------|------|---|---|
| 传输压缩 | 压缩级别 | 1、打开传输压缩开关后，可配置源端传输数据设置的数据压缩级别，1~9。 2、1-4 级为 zip 压缩，5-9 级为 lzma 压缩 3、压缩后数据与原始数据的压缩比，级别 1 大概在 30 ~ 40，2 大概在 15 ~ 30，其它参考具体压缩方式结果 | 1、设置压缩可减少缓存和带宽的占用，但会加大 CPU 占用 2、日志解释： total lines : 所有表的总行数 recv/origin : 收到的字节数, 解压之后的字节数 compression: 压缩率 elapsed time: 用时 |
| 传输加密 | 加密类型 | AES 和 SM4 | 支持 AES 和 SM4 两种加密方式 |
| | 密钥 | 填写对应类型的密钥 | |

6. 高级设置页面各项配置如下:

备库保留存在用户的密码

备端外部表存储位置

忽略外键依赖

表结构变更通知

truncate频率限制

备表新增主键列

* 表和索引的initrans值

全局自定义配置

| key | value | 操作 |
|-----|-------|----|
|-----|-------|----|



暂无数据

| 元件名称 | 注解 | 说明 |
|-------------|----------------------------------|----|
| 备库保留存在用户的密码 | 1、默认为否。 2、选择“是”，即备库将保留存在用户的密码 | |
| 备端外部表存储位置 | 如果源端存在外部表，则需为其定备端存储位置 | |

| | | |
|-----------------|--|----------------------------------|
| 忽略外键依赖 | <p>1、默认为否。</p> <p>2、针对主键映射有外键依赖的表，配合“保留已存在的表“的选项，勾选此选项，在目标端手动创建表然后删除外键约束，目标端对该表拆分队列装载。</p> <p>3、不勾选即外键依赖的相关表装载会进同一队列。</p> | |
| 表结构变更通知 | <p>1. 表结构变更时，以告警方式通知；</p> <p>2. 表结构变更包含建表、删表、truncate 表、表重命名、添加字段、删除字段、重命名字段、修改字段类型、添加约束、删除约束、约束重命名、启用/禁用约束、添加分区、分区和子分区重命名、分区和子分区添加/删除键值、移动分区和子分区、分裂分区和子分区和子分区、融合分区和子分区、交换分区和子分区、合并分区和子分区、删除子分区、子分区添加/删除键值</p> | |
| truncate 频率限制 | 开启后, truncate 操作的执行频率由配置文件的相关参数控制。 | |
| 备表新增主键列 | 备库表新增加主键列 | |
| 表和索引的 intrans 值 | 定义表和索引的 intrans 值 | |
| 全局自定义配置 | 配置全局自定义值 | |
| 内存限制 | 全量导出内存限制 | 默认 8G |
| | 增量解析内存限制 | 默认 8G |
| | 装载内存限制 | 默认 8G |
| 表空间 | 备端表空间 | 1. 用于存放目标端数据库的数据文 1. 如果备端相比较源 |

| | | | |
|---------|---------|---|---|
| 设置 | 存储位置 | <p>件的位置。</p> <p>2、如果源备端表空间一致，此项不必填写</p> <p>3、如果备端较源端缺少表空间时，此项需填写为节点起服务的用户拥有操作权限的路径；当备库为 ASM 存储的数据库环境时，此项需填写为 +DATA 起始的路径</p> | <p>端缺少表空间时，也可事先手动创建好表空间，否则备端在缺少表空间且未填写此项时，装载数据时将会因为找不到表空间而出错。</p> <p>2. 查询表空间语句： select</p> |
| | 表空间名称转换 | <p>用于配置源库、目标库的表空间名称映射</p> <p>此处按实际路径，必须以/或\结尾</p> | <p>TABLESPACE_NAME from dba_tablespaces;</p> |
| 表过滤设置 | 添加 | 点击填写不进行同步的数据表 | |
| | 编辑 | <p>1、适用于大量数据表过滤。</p> <p>2、编辑文本，格式为：“用户名”，“表名”</p> <p>多个表过滤需采用换行处理</p> | 用户名和表名输入区分大小写。 |
| | 按列匹配过滤表 | <p>1、默认关闭</p> <p>2、打开此开关，意指过滤指定用户下包含指定列的所有表</p> <p>3、点击添加，页面增加一行配置项。支持填写多行</p> <p>4、包含列：每行可填写多个字段，用英文逗号隔开。如填写 user:col1,col2，意指过滤 user 中同时包含 col1 和 col2 的所有表</p> | |
| ETL 表设置 | 操作类型 | <p>1、可以选择数据的插入、更新、删除、全部。</p> <p>2、全部即包含了插入、更新和删除操作。</p> | |

| | | | |
|-----------|-----------------|--|--|
| | 用户名、表名 | <p>1、指定具体是针对哪张表进行数据处理。</p> <p>2、用户名和表名区分大小写</p> | |
| 处 理 方式 | 忽略 | 表示对符合条件的捕获数据进行拦截，不会传输到备端去装载 | |
| | 标 记 删除 | 表示对于源端的删除数据，备端仅仅在该表的附加字段上写入 1 作为删除标识。 | 此功能已失效 |
| | 传 输 文件 | <p>1、表示当该表的某个字段内容是某个文件的绝对路径字符串时，程序会把该文件自动拷贝传输到备端的相对应的目录下。</p> <p>2、字段名需要在[字段或条件]中设置</p> | |
| | 装 载 前 执 行 | <p>1、在装载备库表前执行存储过程</p> <p>2、源端执行 DML，备端调试表装载源库表数据</p> <p>3、源库表数据作为存储过程的变量输入，调用存储过程，执行存储过程后，定义为 d\$<col name>的变量发生变化，修改后的数据同步至备库表</p> | <p>1、备库表装载数据时，可以调用预先指定好的存储过程。同步每一条数据均会执行一次存储过程</p> <p>2、表装载数据的操作包括表全量复制时的数据插入操作，增量同步时的插入、更新或删除操作。</p> <p>2、存储过程需要预先在备库执行，名称可以自定义，前两个参数名称和类型必须为 result int 和 returnMsg varchar2</p> <p>3、可在备库创建调试</p> |

| | | | |
|-------------|-------|-----------------|---|
| | | | <p>表，该表用于保存调试数据</p> <p>4、通过预设参数传递字段值的字段类型，目前只支持常用的时间、数字及字符类型 long lob 及其他一些特殊类型，暂不支持。</p> <p>5、规则在重新启动时，会首先调入备库的存储过程。规则运行过程中，修改存储过程不会立即生效，需要重启规则。</p> <p>6、表字段名大写时，存储过程参数定义\$后的字段名也需要大写</p> <p>7、表字段类型为 char，存储过程定义对应参数需设置为 varchar(2)，表字段 nchar，存储对应 nvarchar2</p> |
| | | 装 载 后 执 行 | <p>1、在装载备库表后执行存储过程</p> <p>2、源端执行 DML，备端同步表装载数据</p> <p>3、源库表数据作为存储过程的变量输入，调用存储过程，执行存储过程后的数据同步至调试表</p> |
| ETL 表 设置 | 字段或条件 | | <p>1.设定忽略数据时的数据条件，目前只支持 NUMBER 数据类型、CHAR、</p> <p>1.ETL 过滤需要将过滤列加到附加日志组，并</p> |

| | | | |
|--|--|--|---|
| | | <p>NCHAR、VARCHAR2 字符类型、DATE、TIMESTAMP 日期类型及函数进行条件判断。</p> <p>2.NUMBER 数据类型支持逻辑判断字符>、=、<、>=、<=、<>、!=、^=及 and、or、is null、is not null、in(xx,xx)、'+、'-、'*、\'</p> <p>3.字符类型支持逻辑判断字符>、=、<、>=、<=、<>、!=、^=及 and、or、is null、is not null、in(xx,xx)、like、not like</p> <p>4.日期类型支持>、=、<、>=、<=、<>、!=及 and、or、is null、is not null、sysdate，条件中仅支持 'YYYY-MM-DD HH24:MI:SS' 及“YYYY-MM-DD”格式，填写规范如下：</p> <p>如 下 列 名 >to_date('2022-09-0900:00:00','YY-YY-MM-DD HH24:MI:SS')</p> <p>列 名 <to_date('2009-12-01','YYYY-MM-DD')</p> <p>5.支持函数 mod, to_date, trunc, substr, upper, lower</p> <p>6.处理方式选择“装载前执行”或“装载后执行”时，需填写存储过程名</p> | <p>配置为必须写日志，否则 URP 过滤会失效。</p> <p>2.用户名和表名区分大小写。条件里的字段名也区分大小写，条件里的逻辑字符不要用中文字符格式。当设定了[插入]或[全部]过滤条件时，该条件也会在全同步中进行对源端数据过滤。</p> <p>3.当设定了[更新]过滤条件时，尽量不要在条件中设置多个不同字段的判断条件。因为，当源端只更新了其中列的某些字段而其他字段没有更新时，程序会因为未捕获到条件列所包含的所有字段数据，所以判断条件会失效。</p> <p>4、对于 URP 操作，8.2 版本更新如下：</p> <p>(1. 当一条数据满足过滤条件，同步到备库。随后被更新为不符合同步条件的值，备库该行数据应在备库被删除掉。这时，软件会将</p> |
|--|--|--|---|

| | | | |
|---------|--|----------------------------|---|
| | | | <p>URP 操作转换为 DRP 操作，让备库执行删除操作</p> <p>(2. 当一条数据初始未满足字段过滤条件，未同步至备库。随后更新为符合同步条件的数据，这时，备库不存在这条数据，所以，软件会将该 URP 操作转为 IRP 操作，在备库插入更新好的数据。如果 URP 字段值不全，则软件会到源库去回查获取完整数据。</p> <p>以上实现的最终目的是确保两边数据一致性</p> |
| 表延迟约束装载 | <p>1、未配置延迟约束装载的表，默认在每个 DML 语句后检查约束</p> <p>2、配置此参数后，当执行批量的 DML 语句时，会延迟约束检查，直至事务提交或调用此约束，方验证数据是否满足约束，不满足约束条件的事务将被回滚。</p> | | |
| 字段拼接设置 | <p>用户根据对字段的要求进行自行填写操作，操作类型分为：追加、前缀、常量、拼接、加、减、乘、除。会根据选择的操作不同，显示填写对应所需要的信息。</p> | <p>仅 Oracle-Kafka 链路支持</p> | |

| | | |
|--------|---|---|
| 带宽设置 | <p>1、用户可以配置带宽策略，对全同步和增量均有效。</p> <p>2、先选择时间范围，星期支持多选，后配置带宽参数，随后点击“添加”，即可完成一条带宽策略的配置。</p> <p>3、支持添加多条带宽控制策略，但时间不允许重复。</p> | |
| 运行时间范围 | <p>1、用户可以配置规则增量运行的时间</p> <p>2、先选择时间范围，随后点击“添加”，即可完成一条策略的配置。</p> <p>3、支持添加多条运行策略，但时间不允许重复。</p> <p>4、配置此项的规则，需在提交任务后，在规则“操作” -> “更多”中点击启动调度，此配置方可生效。</p> | <p>1、配置未生效时，规则的导出状态显示[停止调度]，启动后，状态会变更为[调度中]</p> |
| 确定 | 完成规则配置项后，可提交规则 | |
| 取消 | 取消本次新建同步规则的操作 | |

7. 全局对象过滤页面中，全量同步时，对所有勾选的对象进行同步。管理员可以根据需要在添加规则时自行设置，未勾选的对象将被过滤，不再进行全量同步。



8. 增量 DDL 过滤页面中，增量同步时，对所有勾选的 DDL 操作都进行同步。管理员可以根据需要在添加规则时自行设置，不勾选的 DDL 操作，在增量同步时将被过滤，不再进行同步。各配置项如下图所示：



8.2. 同步规则界面

同步规则信息栏说明:

| 元件名称 | 注解 | 说明 |
|------|---|-----------------------------|
| 名称 | 显示用户创建此同步规则时自定义的名称, 便于管理, 支持中文和英文字符 | |
| 数据源 | 显示此同步规则中源端对应的数据库节点名 | 鼠标悬停在库名称一栏时, 可以显示库的基本信息。 |
| 目标端 | 显示此同步规则中源端对应的数据库节点名 | 鼠标悬停在库名称一栏时, 可以显示库的基本信息。 |
| 导出 | 离线: 节点掉线, 可能的原因包括节点宕机、iahelper 进程退出。 停止: 此条规则全量导出停止。 完成: 已完成全同步导出过程。 | 显示获取源机数据库的数据, 并同步至目标机节点的状态。 |
| 导入 | 离线: 节点掉线, 可能的原因包括节点宕机、iahelper 进程退出。 停止: 此条规则全量导入停止。 完成: 已完成全同步导入过程。 | |
| 解析 | 离线: 可能的原因包括节点宕机、iahelper 进程退出。 进行中: 完成全同步, 正在进行增量操作。 停止: 此条规则增量解析停止。 准备: 正在进行增量准备。 | |
| 装载 | 离线: 包括节点宕机、iahelper 进程退出。 进行中: 同步之后进行在增量操作。 停止: 此条规则装载停止。 | 显示此同步规则中数据传至目标机节点的状态。 |

| | | |
|----------|---|--|
| | 准备：正在进行增量准备。 | |
| 解析时间 | 日志解析进程处理完的日志数据的时间，即数据库产生日志的时间，此时间与生产库的机器时间差值代表了日志解析的延迟。 | |
| 装载时间 | 数据装载进程装载完成的数据的时间，此时间为数据的时间，而非目标机的机器时间 | |
| 延迟/峰值(秒) | 显示此同步规则的延迟的时间，单位是秒。 | 1、延迟的计算方法是‘目标机机器时间’ - ‘装载的数据时间’ - ‘机器时间差’。 2、数据库双活会在同步开始时计算机器时间差、并在同步过程中每 10 分钟校正 1 次，保证延迟时间的准确性。 |
| 所有者 | 显示将此同步规则添加数据库双活控制台的用户。 | |
| 维护模式 | 默认为关。开启维护模式，该规则将跳过告警监控，忽略异常。 | |

同步规则操作栏说明：

| 操作 | 注解 | 说明 |
|----|---|--------------------------|
| 继续 | 当规则处于“停止”时，单击“继续”可以继续当前同步规则，继续暂停前的操作。 点击继续运行后，规则状态会先变为全量同步，检查所有的表是否已经同步，之后再进入增量状态。 | |
| 停止 | 停止当前同步规则，分为“停止规则”和“停止规则并停止日志解析”。 | 不停止日志解析，将持续产生缓存数据，请注意缓存空 |

| | | |
|----|------|--|
| | | 间。 |
| 修改 | | 规则在运行过程中不允许修改，只能点击查看，停止规则后可以修改，跳转至同步规则修改界面修改当前同步规则的配置信息。 |
| 日志 | | 1、规则日志信息分为正常的日志消息、注意、警告、一般错误、严重错误五种，管理员可以根据需要进行全部查看或选择性查看。 |
| 更多 | 重新同步 | 重新开始源目标端的全量同步，规则状态从全量同步开始，全量同步结束后进入增量同步状态 |
| 授权 | | 将当前同步规则的不同操作权限授予不同用户。 |
| 删除 | | 删除此同步规则。 |

同步规则左上菜单说明：

| 元件名称 | 注解 | 说明 |
|--------|----------------------|----|
| 新建 | 跳转至同步规则新建界面 | |
| 删除 | 通过单击复选框可以批量删除同步规则 | |
| 刷新 | 刷新当前同步规则的状态 | |
| 规则导出导入 | 支持 json 介质格式导入导出规则配置 | |

9. 数据检查

● 前提条件

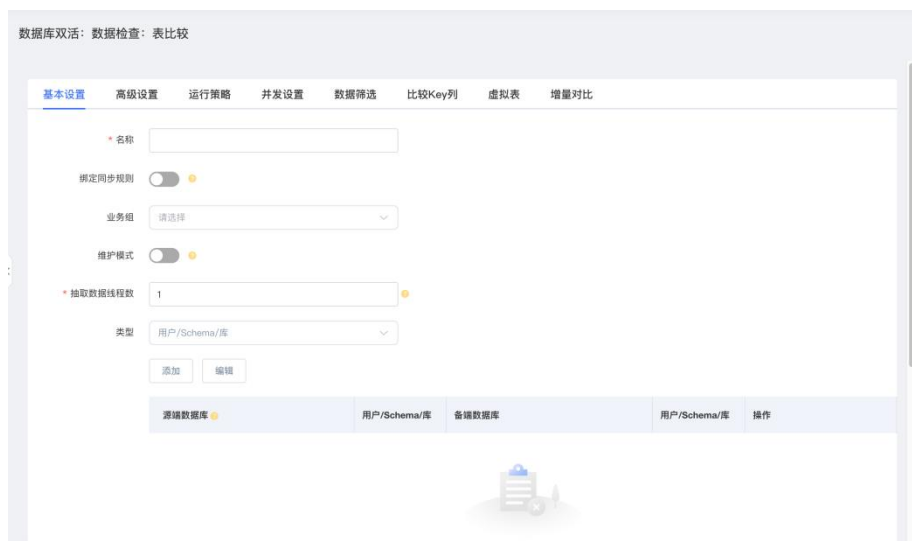
1. 数据库双活进程所在的双活工作节点对应的节点状态都必须为在线状态，对应的进程状态都为正常。

2. 源端、目标端的数据库节点对应的状态必须为正常状态。
3. 确保源端的数据库节点和目标端的数据库节点在注册时，选择的双活工作节点必须为同一个，且此双活工作节点启动了 iadiff 的进程。

9.1. 创建表比较规则

表比较规则 · 新建的具体操作步骤如下：

1. 单击菜单栏“数据库双活” - “数据检查” - “表比较”，进入表比较规则界面。
2. 表比较规则界面中，单击“新建”，进入表比较规则配置界面。



3. 源端数据库选择已注册的源端数据库节点和目标端数据库节点。
4. 表比较规则配置界面开启绑定同步规则后界面变化为 oracle 规则配置界面

总览 / 实时同步 / 表比较 · 新建

基本设置 运行策略 压缩加密 高级设置 增量对比

* 名称

绑定同步规则

* 同步规则

* 源端数据库

* 备端数据库

业务组

自动修复表

修复关联表

维护模式

* 抽取数据线程数

类型

用户映射 表映射 按列匹配

排除表

| 源端用户 | 表名 | 操作 |
|------|----|----|
|------|----|----|

5. 基本设置页面各配置项如下:

| 元件名称 | 注解 | 说明 |
|--------|--|------------------|
| 名称 | 用户自定义的同步规则名称，便于管理，支持中文和英文字符，区分和识别当前任务的名称 | |
| 绑定同步规则 | 开启后，绑定 oracle 同步规则 | 仅可绑定 oracle 同步规则 |
| 同步规则 | 点击选择已创建运行的 oracle 同步规则 | |
| 源端数据库 | 用户根据已添加至数据库节点中的 oracle 数据库，选择作为源端的机器 | |
| 备端数据库 | 用户根据已添加至数据库节点中的 oracle 数据库，选择作为备端的机器 | |

| | | | | |
|---------|-------|-----------------|--|------------------------|
| 业务组 | | | | |
| 自动修复表 | | 不修复 | | |
| | | 完整修复 | | |
| | | 差异修复 | | |
| 抽取数据线程数 | | 对源端数据库的数据抽取的线程数 | | |
| 类型 | 非整库映射 | 添加 | <p>源备两端库之间的数据同步，支持不同库名的映射。</p> <p>1、点击添加按钮添加库，系统默认目标端与源端相同库名；</p> <p>2、如果目标端想同步为不同的库名，可自行修改目标端对应的同步库名</p> <p>库名输入区分大小写，即数据库中库名为大写的，请在库映射中填写大写，如果是小写库名，则输入小写。</p> | |
| | | 选择用户 | 通过搜索快速选择同步用户 | |
| | | 编辑 | <p>1、适用于大批量用户的添加。</p> <p>2、编辑格式为： {源端数据库类型，源端数据库节点名，源端数据库用户，源端数据库名}{备端数据库类型，备端数据库节点名，备端数据库用户，备端数据库名}</p> <p>多个库需采用换行处理</p> | |
| | 表映射 | 添加 | 点击添加，手动填写库、表进行同步 | 1.源备两端数据表之间的数据同步，可以全表同 |
| | | 选择表 | 通过搜索快速选择同步表 | |

| | | | |
|-----|------|-------------------|--|
| | | | <p>步，也可以抽取源端表中部分字段同步到目标端。表映射支持不同用户名、不同表名及不同字段名进行映射。</p> <p>2.表映射表名支持正则表达式，包含元字符\、?、^、\$、*、+、.点、 、[a-z]及{n,m}等，但不支持中文。如表名填写为正则表达式时，备端表名将始终与源端保持一致，不支持不同表名映射。</p> <p>3、使用表映射功能，不需要事先在目标端创建需要同步的数据表。</p> <p>用户名、表名、列名的输入区分大小写</p> |
| | 整库映射 | 源备两端的整库同步 | |
| 排除表 | 添加 | 点击添加，手动填写用户、表进行同步 | |

| | | | |
|----|----|--|--|
| | 编辑 | 1、适用于大批量用户的添加。 2、编辑格式为： 源端用户,备端用户 多个用户需采用换行处理 | |
| 提交 | | 完成规则配置项后，可提交规则 | |
| 取消 | | 取消本次新建比较规则的操作 | |

6. 高级设置页面各配置项如下：

| 元件名称 | | 注解 | 说明 |
|------------------|-----------------|---|---|
| 启用 count 类型 | 不启用 count 比对 | 默认不启用 count 比对。 | |
| | 仅 count 比对 | 表示仅比对源备端的行数，并不进行内容比对。 | |
| | 比较时 count | 表示发现 count 不一致时，直接停止比对。 | |
| 是否只比较 key 列 | | 默认不启用。开启则表示只比对 key 列这些列的内容。 | 比对 key 列可以知道哪些具体的行有差异，比较行数的方法确只能知道行数不同，至于具体是哪些行却不能知道。 |
| 最大差异行 | | 此处填写的是差异行数，表示比较时出现的差异数大于此数值时,则直接停止比较。 | |
| 是否产生 sqlpatch 文件 | | 默认关闭，开启则表示若有表比对不一致的情况，会输出一个可执行的 sql 文本。 | |
| 是否启用容差值比对 | 数值容差 | 表示可接受的比对的数值的值差异大小。 | |

| | | | |
|--|-------------|---|-----------------------------|
| | 时间容差 | 表示可接受的时间点的值的差异大小，仅填写整数值，单位是秒。 | |
| | 大表拆分类型 | 默认为不拆分。一般为增加比对效率进行的配置，通过对大表的拆分进行并发比对，增加比对效率 | |
| | 拆分后单个计划最大行数 | 表示的是拆分大表时，每次拆分为一个计划，每个计划配置的最大行数。 | |
| | 最大过期时间 | 表示的是分拆结果可以重复使用的时间。默认值 0 表示每次都拆分。 | 由于分拆比较还是比较耗时的，不需要每次比对都要去拆分。 |
| | 递归比对的最大次数 | 表示的是比对不一致时，循环反复再次比对的次数限制。 | |
| | 每次递归比对的时间间隔 | 默认是 0 毫秒，表示的是每次循环反复比对时之间的时间间隔，单位是毫秒。 | |
| | 源端忽略类型 | 表示的是比对时忽略的源端表的数据类型，多个类型用逗号分隔开，严格区分大小写。 | |
| | 备端忽略类型 | 表示的是比对时忽略的备端表的数据类型，多个类型用逗号分隔开，严格区分大小写。 | |
| | 源端忽略列名 | 表示的是比对时忽略的源端表的列名，多个类型用逗号分隔开，严格区分大小写。 | |
| | 备端忽略列名 | 表示的是比对时忽略的备端表 | |

| | | |
|-----------|-------------------------|--|
| | 的列名，多个类型用逗号分隔开，严格区分大小写。 | |
| 全局自定义配置参数 | 详见附录表比对全局变量参数。 | |

7. 运行策略页面各配置项如下：

| 元件名称 | | 注解 | 说明 |
|------|------------|------------------------|----|
| 策略类型 | 一次性任务-立即执行 | 表示的是，此表比较规则提交时，立即进行比对。 | |
| | 一次性任务-选择时间 | 表示的是，指定时间再启动表比对规则。 | |
| | 周期性任务 | 表示的是表比对增加周期性的时间配置。 | |

8. 数据筛选页面各配置项如下：

| 元件名称 | | 注解 | 说明 |
|--------------|--------|--|----|
| 全局筛选配置 | | 表示的是对库内所有的进行筛选配置。 | |
| 是否开启全局时间限制比对 | 关闭 | 默认关闭。表示的是对表内含有时间类型的字段进行的筛选比对。对库内所有的表均生效，没有时间类型的字段则不生效。 | |
| | 时间戳字段名 | 表示的是表内含有的时间戳字段名，需输入所有表通用的统一时间戳字段名。 | |

| | | | |
|--------|-------------|---|--|
| | 最新时间 | 表示的是设置当前的时间, 不设置则默认当前部署表比对程序所在机器的时间。 | |
| | 追溯时间间隔 | 表示的是当前时间减去此设置的追溯时间间隔值, 即为表比对筛选的时间范围。单位为秒。 | |
| 单表筛选配置 | | 表示的是对单表的数据筛选进行的配置。 | 优先级高于"全局筛选"。 |
| 排除表配置 | 源端用户/Schema | 用户根据需求选择需要筛选比对的源端用户或者 Schema 名。 | 表示的是单独配置某表, 排除对某表的筛选。 |
| | 源端表名 | 用户根据需求选择需要筛选比对的源端表名。 | |
| | 源端数据查询条件 | 用户自行填写源端数据库类型的 sql 语句中对应的查询条件的 where 语句。 | 查询条件的格式为: ['一级 where 语句'; '二级 where 语句'; '更多 where 语句']。 |
| | 备端用户/Schema | 用户根据需求选择需要筛选比对的备端用户或者 Schema 名。 | |
| | 备端表名 | 用户根据需求选择 | |

| | | | |
|-------|----------|--|---|
| | | 需要筛选比对的备端表名。 | |
| | 备端数据查询条件 | 用户自行填写备端数据库类型的 sql 语句中对应的查询条件的 where 语句。 | 查询条件的格式为: ['一级 where 语句'; '二级 where 语句'; '更多 where 语句']。如果不填写, 默认是和源库的查询语句的 where 保持一致, 请确认源备库的 sql 语句是否通用。 |
| 操作-删除 | | 删除此筛选条件。 | |

9. 比较 Key 列页面指的是人工手动配置哪些列是 Key 列。程序再根据配置的 Key 列进行筛选匹配, 各配置项如下:

| 元件名称 | 注解 | 说明 |
|-------------|---|-----------------------|
| 源端用户/Schema | 用户根据需求选择需要筛选比对的源端用户或者 Schema 名。 | 表示的是单独配置某表, 排除对某表的筛选。 |
| 源端表名 | 用户根据需求选择需要筛选比对的源端表名。 | |
| 备端用户/Schema | 用户根据需求选择需要筛选比对的备端用户或者 Schema 名。 | |
| 备端表名 | 用户根据需求选择需要筛选比对的备端表名。 | |
| 源备端的 Key 列 | 用户自行填写需要比对的表 Key 列名 (包含表的主键和索引), 源备端的 Key 列名需保持一致, 如需比对 | |

| | | |
|-------|------------------------|--|
| | 多个 Key 列，需要用分号； 隔开。 | |
| 操作-删除 | 删除此筛选条件。 | |

9.2. 比较规则界面

表比较规则信息栏说明：

| 元件名称 | 注解 | 说明 |
|------|--|----|
| 开始时间 | 显示此表比较规则开始比对的时间。 | |
| 名称 | 显示用户创建此表比较规则时自定义的名称，便于管理，支持中文和英文字符。 | |
| 源机状态 | 百分比：拆分进度+比较进度。 等待中：等待这个周期再进行比对。 比较完成：完成表比对的任务。 停止：此表比对的规则已停止。 比较异常：此表比对的规则存在异常，未能完成。 | |
| 备机状态 | 百分比：拆分进度+比较进度。 等待中：等待这个周期再进行比对。 比较完成：完成表比对的任务。 停止：此表比对的规则已停 | |

| | | |
|------|------------------------------|--|
| | 止。 比较异常：此表比对的规则存在异常，未能完成。 | |
| 数据源 | 显示此表比较规则中源端数据库节点名。 | |
| 目标端 | 显示此表比较规则中目标端数据库节点名。 | |
| 表差异数 | 界面上直接显示比较完成后的表差异数。 | |
| 类型 | 表、用户或者整库。根据此规则配置的选择显示。 | |
| 所有者 | 显示将此表比较规则添加MDRActive控制台的用户。 | |

表比较规则操作栏说明：

| 操作 | 注解 | 说明 |
|------|--|----------------------------|
| 比较结果 | 当规则处于“停止”时，单击“继续”可以继续当前表比较规则，继续暂停前的操作。 点击继续运行后，规则状态会先变为全量同步，检查所有的表是否已经同步，之后再进入增量状态。 | |
| 历史 | 停止当前表比较规则，分为“停止规则”和“停止规则并停止日志解析”。 | 不停止日志解析，将持续产生缓存数据，请注意缓存空间。 |
| 修改 | 规则在运行过程中不允许修改，只能点击查看，停止规 | 跳转至修改表比对规则的界面。 |

| | | |
|------|-----------------------------------|--|
| | 则可以修改，修改当前表比较规则的配置信息。 | |
| 重新比较 | 重新根据此表比对规则的配置进行比较。 | |
| 停止 | 停止当前的表比对的任务。 | |
| 授权 | 将当前表比对规则的不同操作权限授予不同 MDRActive 用户。 | |
| 删除 | 删除此表比对规则。 | |

10. 对象比较

数据库双活：数据检查：对象比较

在Oracle异构环境下，“对象比较”功能可在数据同步后，对比原表与目标库的数据（名称与数据），若发现不一致，您可直接通过“对象修复”功能进行处理，有效保障两端数据的一致性。

新建 刷新 删除 更多

| <input type="checkbox"/> | 名称 | 开始时间 | 状态 | 数据库 | 目标库 | 对象差异数 | 类型 | 所有者 | 维护模式 | 操作 |
|--------------------------|----------|---------------------|------|----------|-----------|-------|----|----------------------|--------------------------|------------|
| <input type="checkbox"/> | test对象比较 | 2025-12-24 10:42:03 | 比较完成 | oracle源库 | oracle目标库 | 0 | 用户 | 53f03e06843248c5a... | <input type="checkbox"/> | 比较结束 修改 更多 |

共 1 条 1

数据库双活：同步规则

基本设置 运行策略 对象过滤设置

* 名称

* 源端数据库

* 备端数据库

专用比较 通用比较

对比表的记录数

自动恢复对象

维护模式

类型

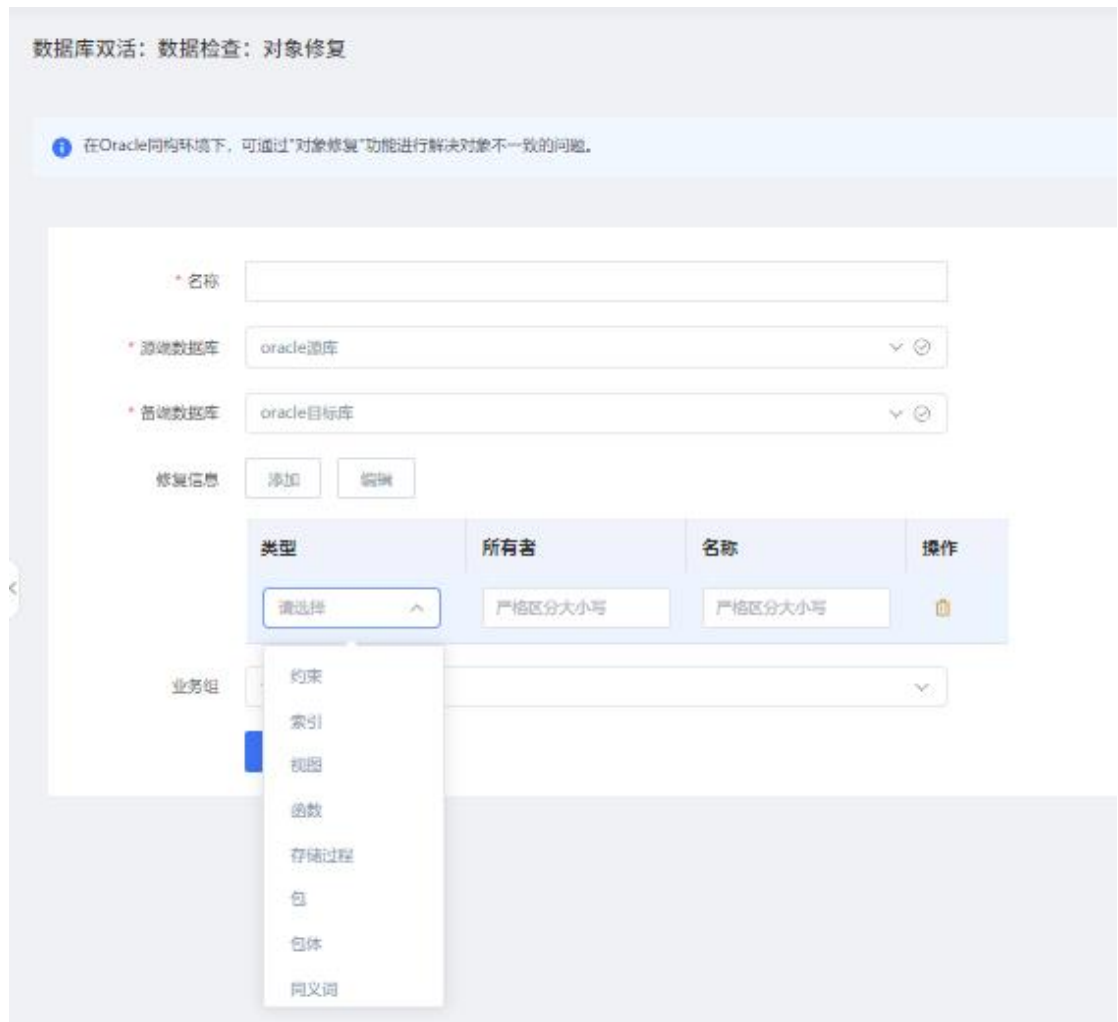
业务组

| 元件名称 | 注解 | 说明 |
|-------|--|----|
| 名称 | 用户自定义的对象比较规则名称，便于管理，支持中文和英文字符，区分和识别当前任务的名称 | |
| 源端数据库 | 用户根据已添加至数据库节点中的 Oracle 数据库，选择作为源端的机器 | |
| 备端数据库 | 用户根据已添加至数据库节点中的 Oracle 数据库，选择作为备端的机器 | |

| | | | | |
|---------|----|-------------|---|--|
| 对比表的记录数 | | | 支持对比表的记录数的比较，用户可根据需求选择是否开启 | |
| 自动修复对象 | | | 对象比较规则完成后，自动修复有差异的对象，用户可根据需求选择是否开启 | |
| 维护模式 | | | 开启按钮，将跳过告警监控，忽略异常 | |
| 映射方式 | 用户 | 添加 | 点击添加，手动填写用户进行对比 | 1、用户名输入区分大小写，即数据库中用户名为大写的，请在用户映射中填写大写，如果是小写用户名，则输入小写 |
| | | 选择用户 | 通过搜索快速选择用户进行比较 | |
| | | 编辑 | <p>1. 适用于大批量用户的添加。</p> <p>2. 编辑格式为：</p> <p>源端用户,备端用户</p> <p>多个用户需采用换行处理</p> | |
| | 整库 | 源备两端整库的对象比较 | | |
| 提交 | | | 点击提交，完成对象比较任务的设置 | |

| | | |
|----|--------------------|--|
| 取消 | 点击取消，退出此次对象比较任务的设置 | |
|----|--------------------|--|

11. 对象修复



| | |
|------|----|
| 元件名称 | 注解 |
|------|----|

| | | |
|-------|----|--|
| 名称 | | 用户自定义的对象修复规则名称，便于管理，支持中文和英文字符，区分和识别当前任务的名称 |
| 源端数据库 | | 用户根据已添加至数据库节点中的 Oracle 数据库，选择作为源端的机器。 |
| 备端数据库 | | 用户根据已添加至数据库节点中的 Oracle 数据库，选择作为备端的机器。 |
| 修复信息 | 添加 | 点击添加，手动填写对象修复信息 |
| | 编辑 | 适用于大批量对象信息的内容的编辑；编辑格式为：类型,所有者,对象修复的名称。多条对象修复需采用换行处理；对象名称需要注意大小写。 |
| 提交 | | 点击“提交”，完成对象修复任务设置 |
| 取消 | | 点击“取消”，退出此次对象修复任务新建页面 |

12. 附录

12.1. 错误代码说明

| 说明 | 错误代码 | 可能引起的原因和解决方案 |
|------------|------|--------------|
| ECONN_ERR | 3522 | 网络问题 |
| EMT_CANCEL | 3523 | 用户取消操作 |
| EVERSION | 3546 | 软件版本不兼容 |

| | | |
|----------------|------|-------------------------|
| ESEQ | 3554 | 发送和接收程序的缓存数据存在丢失情况，不能续传 |
| EBAD_MSG | 3565 | 未知的消息编码 |
| EBAD_PATH | 3571 | 错误的路径 |
| ECOMPRESS | 3581 | 数据压缩失败 |
| EUNCOMPRESS | 3582 | 数据解压失败 |
| IAERR_DB_CONN | 4000 | 数据库连接错误 |
| IAERR_CTL_NET | 4001 | 控制连接断开 |
| IAERR_LOG_SEQ | 4002 | 无法查找指定序号的 redolog |
| IAERR_WR_DISK | 4003 | 磁盘空间问题，无法写入增量数据 |
| IAERR_ORA_INST | 4004 | 个别 Oracle 实例无法连接 |
| IAERR_DMP_DICT | 4005 | 导出数据字典出错 |
| IAERR_DYN_EXIT | 4006 | 源端分析日志线程退出 |
| IAERR_ASM_CONN | 4007 | ASM 实例连接失败 |
| IAERR_DATA | 4008 | TF 文件中数据错误 |
| IAERR_TIMEOUT | 4009 | 等待超时 |
| IAERR_STARTED | 4010 | 已经启动 |
| IAERR_STOPPED | 4011 | 已经停止 |
| IAERR_LANGUAGE | 4012 | 源备端字符集不一致 |
| IAERR_LOAD | 4013 | 备端装载出错 |
| IAERR_UNKNOWN | 4014 | 状态未知 |
| IAERR_NO_RELAY | 4015 | 一对多复制必须启用 |

| | | |
|------------------------|------|---|
| | | 中继节点 |
| IAERR_OFFLINE | 4016 | iahelper 程序异常 |
| IAERR_PROC_ERR | 4017 | 控制程序出现异常 |
| IAERR_CONF | 4018 | 规则配置出错 |
| IAERR_SAVE_DICT | 4019 | 保存数据字典到文件 出错 |
| IAERR_LOAD_DICT | 4020 | 从文件中读取数据字典 出错 |
| IAERR_BLK_TIMEOUT | 4021 | 日志解析停滞不前， 需要引起注意 |
| IAERR_PROC_OFFLINE | 4022 | 进程离线 |
| IAERR_LOAD_TS | 4023 | 备库没有对应的表空间 |
| IAERR_LOAD_USER | 4024 | 备库没有对应的用户 |
| IAERR_MACHINE_CODE | 4025 | 获取机器识别码错误 |
| IAERR_JSON_STR | 4026 | 来自 WEB 页面的 JSON 数据格式错误， 可能是版本兼容问题 |
| IAERR_DML_ERR_TOOMUCH | 4027 | 邮件告警功能：DML 错误过多 |
| IAERR_DDL_ERR_TOOMUCH | 4028 | 邮件告警功能：DDL 错误过多 |
| IAERR_FAIL_TAB_TOOMUCH | 4029 | 邮件告警功能：出错的 表过多 |
| IAERR_LOAD_DELAY | 4030 | 邮件告警功能：装载 延迟超过阈值 |
| IAERR_SYNC_ERR | 4031 | 邮件告警功能：同步 规则状态异常 |

| | | |
|-------------------------|------|--|
| IAERR_RESTART_TRACK | 4032 | relay 进程重启, track 进程需要重新启动解析 |
| IAERR_BIG_TXN | 4033 | 数据库存在长期事务, 中断解析要小心 |
| IAERR_WARN_TIMEOUT | 4034 | relay 某条规则磁盘空间不足, 另外的规则同样展示错误 |
| IAERR_LD_OVERFLOW | 4035 | back 装载数据超过自身限制 |
| IAERR_TRACK_DELAY | 4041 | 解析延时超过设置值 |
| IAERR_REV_NO_RM | 4042 | 反向同步规则没有 rowid 缓存目录 |
| IAERR_ONECMP_TIMEOUT | 4043 | 一次性定时表对比任务超时 |
| IAERR_NODE_NOT_FOUND | 4070 | 未找到节点 |
| IAERR_RULE_IS_RUNNING | 4071 | 规则是运行状态, 不能再次启动此规则 |
| IAERR_TRACK_BUSY | 4072 | track 是运行或者停止状态, 不能进行相应操作 |
| IAERR_RULE_NOT_FOUND | 4073 | 没有找到相应规则 |
| IAERR_I2HUB_IS_STOPPING | 4074 | I2Hub 进程处于正在停止状态, 不能执行目前的操作 |
| IAERR_I2HUB_ERR | 4075 | I2Hub 进程执行 IPC 出错, 具体错误信息需要查看 I2Hub 日志 |
| IAERR_TS_FULL | 4076 | 目标端表空间不足 |

12.2. 错误代码告警说明

| 错误代码 | 告警级别 | 告警消息摘要 |
|------|-------|-----------------------|
| 3522 | warn | communicate lost |
| 3523 | info | cancel by user |
| 3546 | fatal | version not match |
| 3554 | fatal | TF seq problem |
| 3565 | err | unknown-err msg |
| 3571 | err | path not correct |
| 3581 | fatal | compress data fail |
| 3582 | fatal | unzip data fail |
| 4000 | warn | db connect err |
| 4001 | warn | network broken |
| 4002 | fatal | can' t find redo file |
| 4003 | err | work node disk full |
| 4004 | warn | oracle can' t visit |
| 4005 | fatal | dump dict err |
| 4006 | fatal | dyn-ld thread exit |
| 4007 | warn | asm connect err |
| 4008 | fatal | TF file' s data err |
| 4009 | warn | wait timeout |
| 4010 | info | has been started |
| 4011 | info | has been stopped |
| 4012 | err | db lang different |
| 4013 | err | load data failed |
| 4014 | err | unknown |
| 4015 | warn | no relay node |

| | | |
|------|-------|---|
| 4016 | warn | node offline |
| 4017 | warn | process offline |
| 4018 | fatal | rule configure err |
| 4019 | err | save dict err |
| 4020 | err | load dict err |
| 4021 | warn | analyze timeout |
| 4022 | warn | process offline |
| 4023 | err | load ts err |
| 4024 | err | load user err |
| 4025 | err | get machine code err |
| 4026 | err | json string format err |
| 4027 | warn | more dml failure |
| 4028 | warn | more ddl failure |
| 4029 | warn | more tables failure |
| 4032 | info | restart track |
| 4033 | warn | big transaction |
| 4034 | warn | timeout |
| 4035 | err | load overflow |
| 4041 | warn | track delay |
| 4042 | err | revert rule no ridmap |
| 4050 | err | rule is ok, but has error logs, please focus on rule err log |
| 4051 | info | startup iaproxy |
| 4052 | info | startup iawork |
| 4053 | info | startup iaback |
| 4054 | info | startup iarelay |

| | | |
|------|---------|----------------------------|
| 4055 | info | startup iatrack |
| 4056 | info | startup iamask |
| 4057 | info | startup consume |
| 4058 | warn | node state error |
| 4059 | warn | db state error |
| 4060 | warn | rule state change |
| 4061 | warn | task state change |
| 4063 | warn | load ddl |
| 4064 | report | cmp tab or obj info |
| 4065 | warn | load delay overtime |
| 4066 | info | startup iadbmon |
| 4067 | info | startup iamss |
| 4068 | info | startup iadiff |
| 4069 | info | startup ialoader |
| 4070 | info | startup iaQianbase |
| 4071 | warn | rule is running |
| 4072 | err | track delay overtime |
| 0 | recover | model state recover normal |

12.3. 表比对全局变量参数

| key | 含义 | 默认值 |
|-----------------------------------|-----------------------|--------|
| dkdiffengine.progress_batch_size | 打印当前执行步数频率 | 100000 |
| dkdiffengine.default_fetch_size | 查询 fetch size | 1000 |
| dkdbsource.left_diff_type_names | 左边比对数据类型(多个数据类型用逗号分隔) | 空 |
| dkdbsource.left_ignore_type_names | 左边忽略数据类型(多个数据类型用逗号分隔) | 空 |

| | | |
|--|---|-------|
| dkdbsource.right_diff_type_names | 右边比对数据类型(多个数据类型用逗号分隔) | 空 |
| dkdbsource.right_ignore_type_names | 右边忽略数据类型(多个数据类型用逗号分隔) | 空 |
| dkdbsource.diff.only.key.columns | 只比对 key 列 | false |
| dkdiffengine.recursion_max_steps | "递归比对"最大步数 | 1 |
| dkdiffengine.recursion_interval_step_delay | "递归" 每步之间延迟(单位毫秒) | 0 |
| dkapplication.execute.threads | 并发比对最大线程数 | 1 |
| database.select.disable.orderby | 是否不用 order by 值 0: 不 disable(使用 order by) 值 1: 有主键不 disable, 没有主键时 disable 值 2: disable | 0 |
| dkbinarydiffor.string.encode.binary | binary 数据 string 存储(某些数据库, 也许因为不支持二进制数据类型等原因, 用字符串保存二进制数据). | true |
| dkfilesink.auto.delete.exists.file | DKFileSink 自动删除(如果差异输出文件已经存在, 是否自动删除). 默认 false | false |
| debug.dkdatabase.using.connection.pool | 是否使用数据库连接池 | false |
| debug.databasevisit.using.thread.task | 数据库查询和使用线程分离 | true |
| debug.dkdiffengine.empty.comparison | 空比对(进行真正的比较, 主要用于调试时测试连接获取数据速度) | false |
| debug.dkdiffengine.empty.left.getrow | 空左边获取行数据(不从左边取数据) | false |
| debug.dkdiffengine.empty.right.getrow | 空右边获取行数据(不从右边取数据) | false |

| | | |
|---|---|---------|
| databasevisit.samekey.max.rows | 同 Key 行最大行数(为了避免内存溢出, 同 Key 的行数如果达到或者超过该值, 则抛出异常退出本地计划执行) | 2048 |
| dknumberdiffor.null.is.zero | 数值比对器配置 | false |
| dkautomatictablecomparison.ignore.notexists.columns.on.rightside | 是否忽略(数据库表比较)右边不存在的列。数据库比较, 如果左边的表的列名在右边表不存在, 则该列忽略(不参与比较) | true |
| databasevisit.task.queue.size | (数据库查询和使用线程分离)数据库访问任务队列 | 64 |
| dkdbsource.oracle_uniquekey_actas_primarykey_pattern | oracle(如果没有主键, 索引名称符合某个特征(索引名称正则表达式)的唯一索引可以看做主键, 比如, 如果索引名称 QBPK 结尾的匹配模式, 可以配置为: (\$i)*.QBPK\$) | 空 |
| dksink.max.binary.format.length | 大数据 (byte[], blob, clob) sink 最大显示长度, 超过长度部分截断并未尾显示省略号(...). 最大值 3997. | 1024 |
| databasevisit.max.where.keylist.size databasevisit.left.max.where.keylist.size databasevisit.right.max.where.keylist.size | 递归比对时, 如果前一步差异行数太多, 构建的 where 语句就会很长, 可能导致查询不能执行。如果前一步差异行数超过该配置数值时, 将拆分 where 条件产生多个查询 sql. | 512 |
| databasevisit.disable.where.keylist.size | 用于 disable 上述 16 的功能 | false |
| databasevisit.max.split.keylist.size databasevisit.left.max.split.keylist.size databasevisit.right.max.split.size | 递归比对时, 如果前一步差异行数太多, 用上述 16 配置产生的 sql 语句可能很多, 多次 keylist 条件查询 | 16*1024 |

| | | |
|--|---|-------|
| keylist.size | 表的速度可能反而没有整表无keylist 条件查询速度快。如果前一步差异行数超过该配置数值时, 将用整个查询(不添加 keylist 条件). | |
| databasevisit.disable.split. keylist.size | 用于 disable 上述 18 的功能 | false |
| database.breakpoint.retry.max.count | 执行查询过程中, 数据库连接中断, 断点重试最大次数 (0 表示不重试) | 10 |
| database.breakpoint.retry. delay.in.ms | 执行查询过程中, 数据库连接中断, 断点重试各次数之间延迟(单位 毫秒) | 6000 |
| dkdatabase.getconnection. retry.max.count | 获取数据库连接最大重试次数 | 2 |
| dkdatabase.getconnection. retry.delay.in.ms | 获取数据库连接重试各次数之间延迟(单位毫秒) | 6000 |
| debug.xmlfileinfo.xml.format.pretty | 设置 RunTime 文件格式友好打印 | true |
| debug.table.wheres. delete.split.file.after.merge | where 条件分拆大表多计划执行产生多个 json 文件, 最后会自动合并。是否删除分拆 json 文件 | true |
| split.table.schedule.cron | 大表比对时间可能比较长, 可以将其进行拆分, 然后利用多线程比对功能, 对大表进行并行比对。配置为空串是表示不进行大表拆分, 配置为 immediate 表示比对时进行拆分, 配置 cron 表达式表示按 cron 表达式配置定时拆分 | 空 |
| split.table.execute.threads | 大表拆分使用的线程数, 默认和 dkapplication.execute.threads 一样 | |

| | | |
|---|--|-------|
| split.table.single.segment.max.rows | 拆分后单个计划最大行数 | 500 万 |
| split.table.max.split.regions | 单个表最大拆分计划数 | 50 |
| split.table.result.expire.in.seconds | 拆分结果多长时间过期(单位秒) | 86400 |
| debug.split.table.result.detail.compute.force | 调试配置：拆分表结果每次都强制计算（默认配置下，如果表结构和总行数，最大最小值等变化不大，则不需要重新拆分），这种配置不利于测试 | false |
| left.extra.system.user.list | 左边系统数据库(schema)，多个用逗号分隔 | 空 |
| left.extra.system.user.prefix.list | 前缀 | 空 |
| left.extra.system.user.postfix.list | 后缀 | 空 |
| right.extra.system.user.list | 右边系统数据库 | 空 |
| right.extra.system.user.prefix.list | 前缀 | 空 |
| right.extra.system.user.postfix.list | 后缀 | 空 |
| left.extra.not.system.user.list | 左边系统数据库(schema)(排除配置) | 空 |
| left.extra.not.system.user.prefix.list | 前缀 | 空 |
| left.extra.not.system.user.postfix.list | 后缀 | 空 |
| right.extra.not.system.user.list | 右边系统数据库(排除配置) | 空 |
| right.not.extra.system.user.prefix.list | 前缀 | 空 |
| right.extra.not.system.user.postfix.list | 后缀 | 空 |
| left.extra.system.tablename.list | 左边系统表（表名 test1 匹配所有 schema，而表名 sys.test1 只匹配 schema sys) | 空 |
| left.extra.system.tablename.prefix.list | 前缀 | 空 |

| | | |
|---|---|-------|
| left.extra.system.tablename.postfix.list | 后缀 | 空 |
| right.extra.system.tablename.list | 右边系统表 | 空 |
| right.extra.system.tablename.prefix.list | 前缀 | 空 |
| right.extra.system.tablename.postfix.list | 后缀 | 空 |
| left.extra.not.system.tablename.list | 左边系统表(排除配置) | 空 |
| left.extra.not.system.tablename.prefix.list | 前缀 | 空 |
| left.extra.not.system.tablename.postfix.list | 后缀 | 空 |
| right.extra.not.system.tablename.list | 右边系统表(排除配置) | 空 |
| right.extra.not.system.tablename.prefix.list | 前缀 | 空 |
| right.extra.not.system.tablename.postfix.list | 后缀 | 空 |
| left.intermediate.tablename.postfix.list | 由于同步等各种需要，每个原表可能会产生一些中间表，这些中间表需要忽略。左边中间表前缀(多个之间用逗号分隔) | |
| right.intermediate.tablename.postfix.list | 右边中间表前缀(多个之间用逗号分隔) | |
| dkdbsource.oracle.bfile.as.text | oracle bfile 类型 bfile 按文本读取文件名; false: 读取文件内容, true: 只读取文件名; 默认 false | |
| dkfilesink.enable.sqlpatch.file | json file (sink) 时是否同时产生 sqlpatch 文件 | false |

| | | |
|--|---|---------|
| execute.patch.after.complete | 启用 dkfilesink.enable.sqlpatch.file 后，是否在比对完成后自动执行 | false |
| debug.with.gui.dbtype | 调试配置：统一比对界面联调时可以配置忽略界面的数据库类型，无论界面传什么数据库类型，都统一用配置的数据库类型 | 空 |
| dktable.access.include.systems | 是否比对系统表和系统 schema | false |
| dkmagic.plan.max.diffs | 最大差异数，如果单个计划最大差异数超过该值则退出比对 | 10000 |
| dkmagic.plan.number.tolerance | 数值容差 | 0.00001 |
| databasevisit.select.orderby.collation | 配置排序字典。实际配置时需要与此为前缀，比如配置 mysql 配置用 databasevisit.select.orderby.collation.MYSQL。配置后每个字符串字段排序后面会添加该值 | |
| dkdiff.enable.step.count.table | 是否启用 count table 阶段。启用 count table 阶段配置下，如果源备端 count（表行数）不一样，则不进行下一阶段比对 | false |
| dkdiff.only.step.count.table | 是否只比对 count。启用后值比对表的行数，不进行内容比对 | false |
| dkdiff.only.step.count.table.output | 如果配置，这只比对行数时用特殊格式输出 | |
| debug.enable.limit.max.threads | 是否限制最大线程数为 cpu 核心数 | false |
| debug.primarykey.mustbe.index | 调试配置：是否判断主键在索引里才有主键列作为比对 key | false |
| exception.ignore.db2.malformedinputexception | 是否忽略 Db2 字符串转码异常。如果忽略异常，rs.getString 返回 null | true |
| debug.dkequalsdiffor.timezone.offset | 有时区和无时区字段比对，调整有 | 0 |

| | | |
|--------------------------|--|--|
| | 时区字段时间(在 UTC 时区基础上调整)。单位: 秒 (可以为负数) | |
| db.qianbasextp.cert.path | qianbase xTP certs 目录, 需要包含 ca.crt, client.yfcdc.crt, client.yfcdc.key.pk8 | |