

解决方案实践

快速搭建 DataX 数据库迁移环境

文档版本 1.0.0
发布日期 2024-10-12



版权所有 © 华为技术有限公司 2024。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。

安全声明

漏洞处理流程

华为公司对产品漏洞管理的规定以“漏洞处理流程”为准，该流程的详细内容请参见如下网址：

<https://www.huawei.com/cn/psirt/vul-response-process>

如企业客户须获取漏洞信息，请参见如下网址：

<https://securitybulletin.huawei.com/enterprise/cn/security-advisory>

目录

1 方案概述	1
2 资源和成本规划	3
3 实施步骤	5
3.1 准备工作.....	5
3.2 快速部署.....	8
3.3 开始使用.....	14
3.4 快速卸载.....	24
4 附录	26
5 修订记录	27

1 方案概述

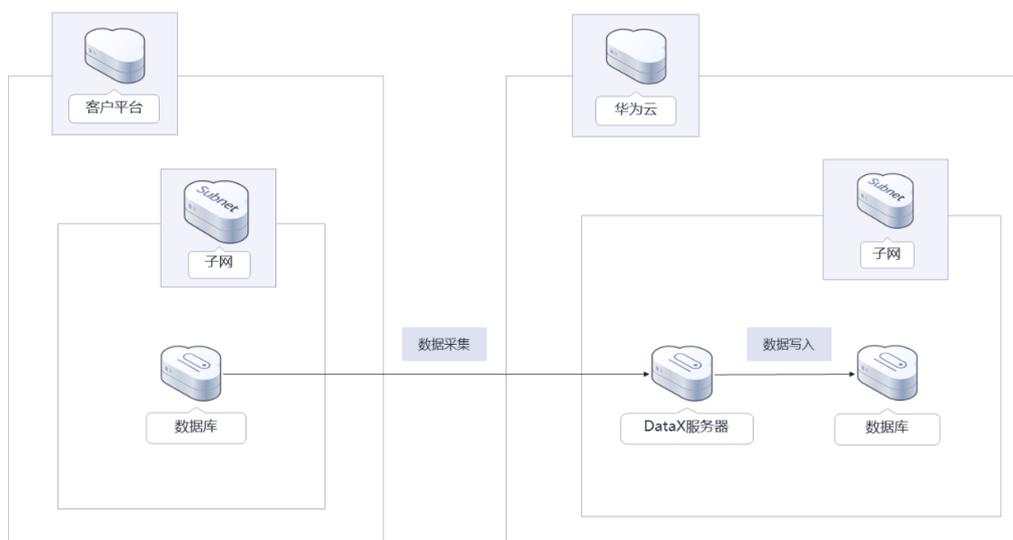
应用场景

该方案帮助您基于DataX快速实现异构数据源之间的高效同步迁移上云。它是一款开源、高效、易用的异构数据源离线同步工具，致力于实现包括关系型数据库(MySQL、Oracle等)、HDFS、Hive、ODPS、HBase、FTP 等各种异构数据源之间稳定高效的数据同步功能。

方案架构

该解决方案基于华为云部署DataX数据库迁移环境，帮助您在华为云上快速实现异构数据源之间的高效同步迁移。解决方案架构图如下：

图 1-1 方案架构图



部署该方案中需要使用的资源：

- 创建一台Flexus云服务器X实例，部署数据库迁移的DataX同步环境。
- 创建一个弹性公网IP EIP，并绑定到云服务器，用于提供访问公网和被公网访问能力。

- 创建安全组，通过配置安全组规则，为云服务器提供安全防护。

方案优势

- 并行执行
采用了分布式架构和多线程技术，能够并行地从多个数据源抽取数据，并将数据加载到目标数据源中，大大提高了数据同步的效率。
- 易配置管理
DataX 提供了直观的 JSON 配置文件格式，用户只需编写简单的配置文件，指定数据源、目标数据源、数据转换规则等信息，即可启动数据同步任务。
- 数据转换处理灵活
在数据同步过程中，DataX 提供了丰富的数据转换功能。可以进行字段映射、数据清洗、格式转换、数据过滤等操作，确保数据在不同数据源之间的一致性和准确性。

约束与限制

- 该解决方案部署前，需注册华为账号并开通华为云，完成实名认证，且账号不能处于欠费或冻结状态。如果计费模式选择“包年包月”，请确保账户余额充足以便一键部署资源的时候可以自动支付；或者在一键部署的过程进入[费用中心](#)，找到“待支付订单”并手动完成支付。
- 如果选用IAM委托权限部署资源，请确保使用的华为云账号有IAM的足够权限，具体请参考[创建rf_admin_trust委托](#)；如果使用华为主账号或admin用户组下的IAM子账户可不选委托，将采用当前登录用户的权限进行部署。

2 资源和成本规划

该解决方案主要部署如下资源，以下费用仅供参考，具体请参考华为云官网[价格详情](#)，实际收费以账单为准。

表 2-1 资源和成本规格-云服务器部署(包年包月)

华为云服务	配置示例	每月预估花费
Flexus云服务器X实例	<ul style="list-style-type: none">区域：华北-北京四计费模式：包年包月规格：Flexus云服务器X实例 性能模式（关闭） x1.2u.4g 2vCPUs 4GiB镜像：CentOS 7.6 64bit系统盘：GPSSD 100GB数据盘：GPSSD 100GB购买量：1	248.00元
弹性公网IP EIP	<ul style="list-style-type: none">按需计费：0.82元/GB区域：华北-北京四计费模式：按需计费线路：动态BGP公网带宽：按流量计费购买数量：1	0.82 元/GB
合计		248.00元 + EIP流量费用

表 2-2 资源和成本规格-云服务器部署(按需计费)

华为云服务	配置示例	每月预估花费
Flexus云服务器X实例	<ul style="list-style-type: none">● 按需计费：0.42元/小时● 区域：华北-北京四● 计费模式：按需计费● 规格：Flexus云服务器X实例 性能模式（关闭） x1.2u.4g 2vCPUs 4GiB● 镜像：CentOS 7.6 64bit● 系统盘：GPSSD 100GB● 数据盘：GPSSD 100GB● 购买时长：720小时● 购买量：1	301.68元
弹性公网IP EIP	<ul style="list-style-type: none">● 按需计费：0.82元/GB● 区域：华北-北京四● 计费模式：按需计费● 线路：动态BGP● 公网带宽：按流量计费● 购买数量：1	0.82 元/GB
合计		301.68元 + EIP流量费用

3 实施步骤

- 3.1 准备工作
- 3.2 快速部署
- 3.3 开始使用
- 3.4 快速卸载

3.1 准备工作

当您首次使用华为云时注册的账号，则无需执行该准备工作，如果您使用的是IAM用户账户，请确认您是否在admin用户组中，如果您不在admin组中，则需要为您的账号[授予相关权限](#)，并完成以下准备工作。

创建 rf_admin_trust 委托（可选）

步骤1 进入华为云官网，打开[控制台管理](#)界面，鼠标移动至个人账号处，打开“统一身份认证”菜单。

图 3-1 控制台管理界面



图 3-2 统一身份认证菜单



步骤2 进入“委托”菜单，搜索“rf_admin_trust”委托。

图 3-3 委托列表



- 如果委托存在，则不用执行接下来的创建委托的步骤
- 如果委托不存在时执行接下来的步骤创建委托

步骤3 单击步骤2界面中的“创建委托”按钮，在委托名称中输入“rf_admin_trust”，委托类型选择“云服务”，输入“RFS”，单击“完成”。

图 3-4 创建委托

委托 / 创建委托

* 委托名称

* 委托类型 普通账号
将账号内资源的操作权限委托给其他华为云账号。
 云服务
将账号内资源的操作权限委托给华为云服务。

* 云服务

* 持续时间

描述

0/255

步骤4 单击“立即授权”。

图 3-5 委托授权

授权

是否立即为当前创建的委托进行授权?

步骤5 在搜索框中输入“Tenant Administrator”并勾选搜索结果，单击“下一步”。

图 3-6 选择策略



步骤6 选择“所有资源”，并单击“确定”完成配置。

图 3-7 设置最小授权范围



步骤7 “委托”列表中出现“rf_admin_trust”委托则创建成功。

图 3-8 委托列表



----结束

3.2 快速部署

本章节主要帮助用户快速部署“快速搭建DataX数据库迁移环境”解决方案。

表 3-1 参数说明

参数名称	类型	是否可选	参数解释	默认值
subnet_id	string	必填	子网ID。如使用已有虚拟私有云VPC，请填写已有VPC子网ID；如使用新建VPC须填写new，新建VPC名称同云服务器名称，子网名称： <code>{ecs_name}-subnet</code> 。 获取已有VPC子网ID 。	new
security_group_name	string	必填	安全组名称。该模板使用新建安全组。取值范围：1-64个字符，支持数字、字母、中文、_(下划线)、-(中划线)、.(点)。	creating-a-database-migration-environment-with-datax
ecs_name	string	必填	云服务器名称。不允许重名。取值范围：1-56个字符组成，支持字母、数字、中文、下划线()、中划线(-)、英文句号(.)。	creating-a-database-migration-environment-with-datax
ecs_flavor	string	必填	云服务器实例规格，支持弹性云服务器ECS及华为云Flexus云服务器X实例。Flexus云服务器X实例规格ID命名规则为x1.?u.?g，例如2vCPUs4GiB规格ID为x1.2u.4g，具体华为云Flexus云服务器X实例规格请参考控制台。具体规格请参考官网 弹性云服务器规格清单 。	x1.2u.4g
system_disk_size	number	必填	云服务器系统盘大小，磁盘类型默认为通用型SSD。以GB为单位，取值范围为40~1,024，不支持缩盘。	100
data_disk_size	number	必填	云服务器数据盘大小，磁盘类型默认为通用型SSD。以GB为单位，取值范围为10-32,768。	100
ecs_password	string	必填	云服务器初始密码。取值范围：长度为8-26位，密码至少必须包含大写字母、小写字母、数字和特殊字符(!@\$\$%^_+=+[{ }];,./?)中的三种，密码不能包含用户名或用户名的逆序。管理员帐户为root。	空
charging_mode	string	必填	计费模式，默认自动扣费，取值为prePaid(包年包月)或postPaid(按需计费)	postPaid

参数名称	类型	是否可选	参数解释	默认值
charging_unit	string	必填	包年或包月，有效值为“year”或“month”。当charging_mode（计费模式）为prePaid时，此选项为必填项。	month
charging_period	number	必填	包年包月时长，当charging_unit取值为“year”，取值范围为1-3；取值为“month”，取值范围为1-9。当charging_mode（计费模式）为prePaid时，此选项为必填项。	1

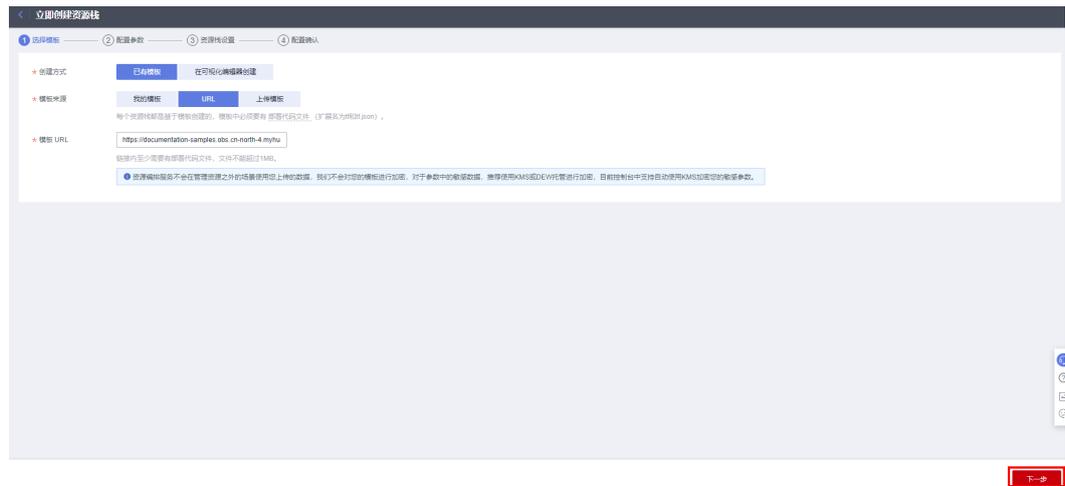
步骤1 登录[华为云解决方案实践](#)，选择“快速搭建DataX数据库迁移环境”，单击“一键部署”，跳转至解决方案创建资源栈界面。

图 3-9 解决方案实施库



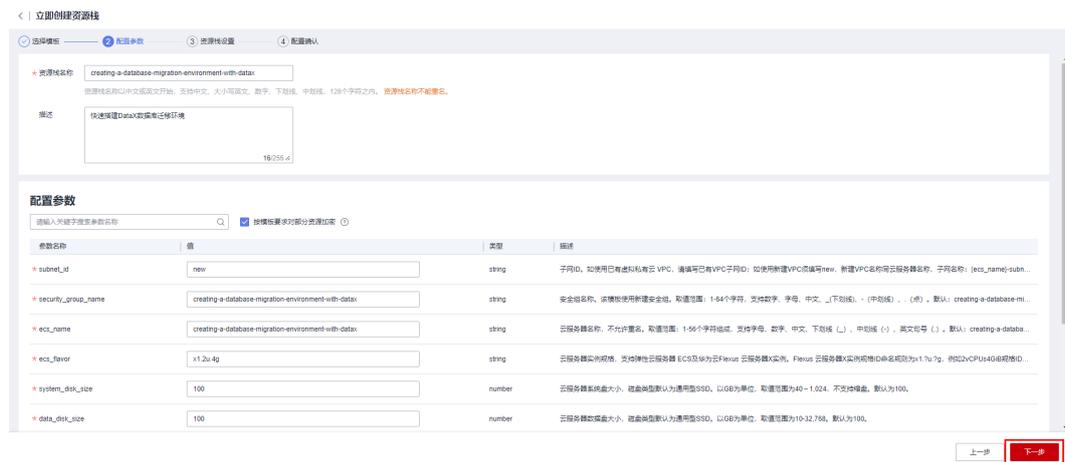
步骤2 在选择模板界面中，单击“下一步”。

图 3-10 选择模板



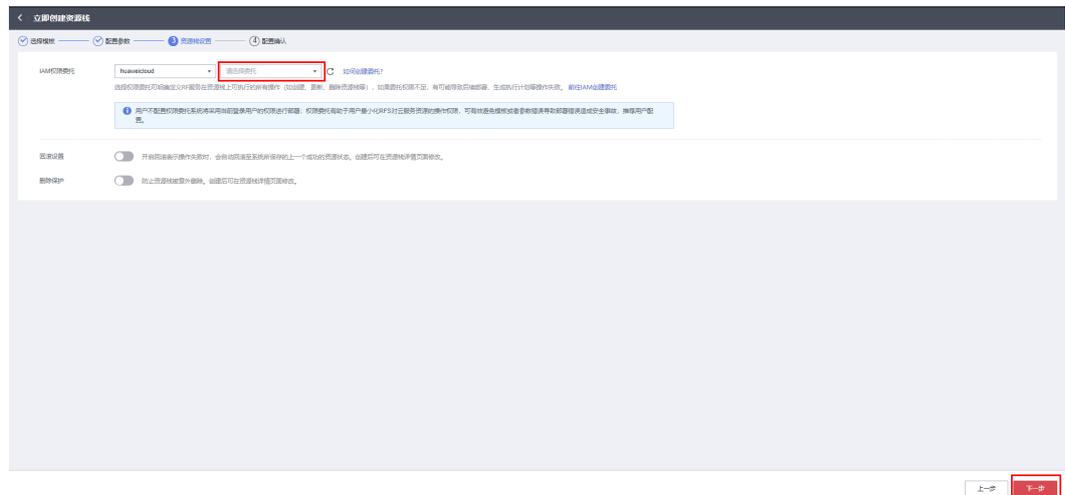
步骤3 在配置参数界面中，参考表 表1 参数说明 完成自定义参数填写，单击“下一步”。

图 3-11 配置参数



步骤4 （可选，如果使用华为主账号或admin用户组下的IAM子账户可不选委托）在资源设置界面中，在权限委托下拉框中选择“rf_admin_trust”委托，单击“下一步”。

图 3-12 资源栈设置



步骤5 在配置确认界面中，单击“创建执行计划”。

图 3-13 配置确认

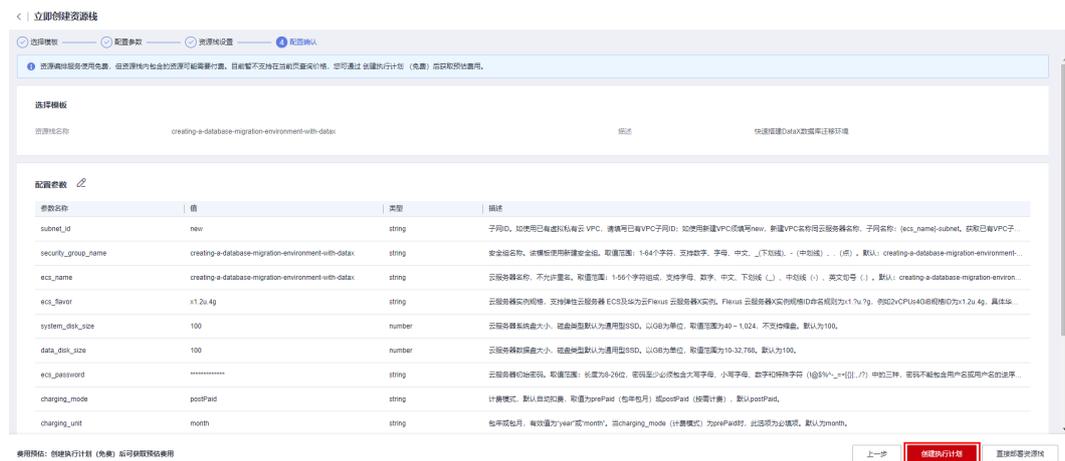


图 3-14 创建执行计划

创建执行计划

通过执行计划，可以预览您的资源变更信息。

* 执行计划名称: executionPlan_20241010_1018_4b39

描述: 请输入对执行计划的描述

0/255

确定 取消

步骤7 单击“部署”，并且在弹出的执行计划确认框中单击“执行”。

图 3-15 执行计划

creating-a-database-migration-environment-with-datax

删除 更新模板或参数

基本信息 资源 输出 事件 模板 执行计划

执行计划名称ID	状态	操作按钮	创建时间	描述	操作
executionPlan_20241010_1018_4b39 c13a8c57-1c44-4480-9d25-2228b0c5d20c	创建成功, 待部署	部署费用明细	20241010 10:19:20 GMT+08:00	-	部署 删除

图 3-16 执行计划确认



步骤8 （可选）如果计费模式选择“包年包月”，在余额不充足的情况下（所需总费用请参考[资源和成本规格-云服务器部署\(包年包月\)](#)）请及时登录[费用中心](#)，手动完成待支付订单的费用支付。

步骤9 待“事件”中出现“Apply required resource success”，表示该解决方案已经部署完成。大约等待10-15分钟，DataX数据库同步环境即可搭建完成。

图 3-17 部署完成



----结束

3.3 开始使用

验证 DataX 迁移服务正常运行

📖 说明

- 有关DataX中间件说明请参考[DataX开源地址](#)。
- 该解决方案使用的资源版本：Java版本：JDK-22。
- 请参考[添加安全组规则](#)，配置对应数据库服务器IP地址白名单，以便能正常使用服务。
- 注意：该服务分为新建VPC和已有VPC两个版本，保证DataX服务器和目标数据库服务器在一个子网下，实现私网的连接与访问，提高迁移数据的安全性。

步骤1 登录[Flexus云服务器X实例](#)控制台，选择创建的云服务器，单击远程登录，进入Linux云服务器。

图 3-18 登录云服务器



图 3-19 登录 Flexus 云服务器



步骤2 在Linux云服务器中输入账号和密码后回车。

步骤3 输入命令：`cd /opt/datax`进入该目录，执行`python3 bin/datax.py job/job.json`查看回显，如回显为下图红框所示，则表示datax服务成功搭建。

图 3-20 执行 datax 运行命令

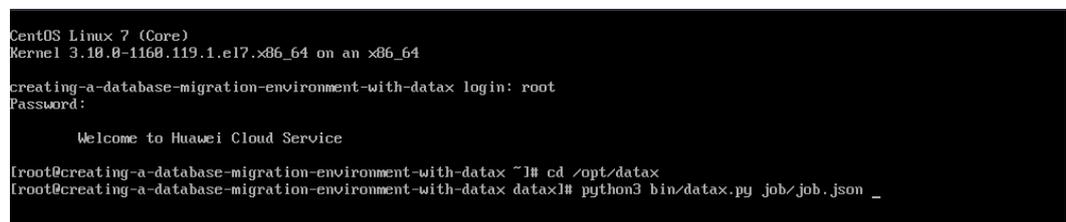


图 3-22 生成模板

```
[root@ ~]# -dev datax]#  
[root@ ~]# -dev datax]# python3 bin/datax.py -r mysqlreader -w sqlserverwriter  
  
DataX (DATAX-OPENSOURCE-3.0), From Alibaba !  
Copyright (C) 2010-2017, Alibaba Group. All Rights Reserved.  
  
Please refer to the mysqlreader document:  
https://github.com/alibaba/DataX/blob/master/mysqlreader/doc/mysqlreader.md  
  
Please refer to the sqlserverwriter document:  
https://github.com/alibaba/DataX/blob/master/sqlserverwriter/doc/sqlserverwriter.md  
  
Please save the following configuration as a json file and use  
python {DATAX_HOME}/bin/datax.py {JSON_FILE_NAME}.json  
to run the job.  
  
{  
  "job": {  
    "content": [  
      {  
        "reader": {  
          "name": "mysqlreader",  
          "parameter": {  
            "column": [],  
            "connection": [  
              {  
                "jdbcUrl": [],  
                "table": []  
              }  
            ],  
            "password": "",  
            "username": "",  
            "where": ""  
          }  
        },  
        "writer": {  
          "name": "sqlserverwriter",  
          "parameter": {  
            "column": [],  
            "connection": [  
              {  
                "jdbcUrl": "",  
                "table": []  
              }  
            ],  
            "password": "",  
            "postSql": [],  
            "preSql": [],  
            "username": ""  
          }  
        }  
      }  
    ],  
    "setting": {  
      "speed": {  
        "channel": ""  
      }  
    }  
  }  
}
```

步骤2 模板字段简介

1. job
 - 配置文件的主要对象，包含整个数据迁移任务的配置信息。
2. setting
 - 位于job对象下，用于设置数据迁移任务的全局参数，如并发通道数（channel），该参数决定了DataX在执行数据迁移时使用的并行线程数。
3. content
 - 位于job对象下，是一个数组，包含多个数据迁移任务的具体配置。每个任务都包含reader和writer两个部分。
4. reader

- 位于content数组的每个对象中，用于配置数据源读取器。
 - name: 读取器的名称，如mysqlreader表示MySQL数据源读取器。
 - parameter: 读取器的参数配置，包含数据源连接信息、要读取的字段等。
 - username: 数据源的用户名。
 - password: 数据源的密码。
 - column: 要读取的字段列表。
 - connection: 数据源连接信息，包括JDBC URL和要读取的表名等。
- 5. writer
 - 位于content数组的每个对象中，用于配置目标数据库写入器。
 - name: 写入器的名称，如sqlserverwriter表示SQL Server目标数据库写入器。
 - parameter: 写入器的参数配置，包含目标数据库连接信息、要写入的字段等。
 - username: 目标数据库的用户名。
 - password: 目标数据库的密码。
 - column: 要写入的字段列表，应与数据源读取器中的字段列表对应。
 - postSql: 写入数据到目的表后，会执行这里的标准语句。
 - preSql: 写入数据到目的表前，会先执行这里的标准语句，例如：清空表数据。
 - connection: 目标数据库连接信息，包括JDBC URL和要写入的表名等。

图 3-23 模板示例:

```
{
  "job": {
    "setting": {
      "speed": {
        "channel": "4"
      }
    },
    "content": [
      {
        "reader": {
          "name": "mysqlreader",
          "parameter": {
            "username": "mysql_user",
            "password": "mysql_password",
            "column": ["id", "name", "age"],
            "connection": [
              {
                "jdbcUrl": ["jdbc:mysql://localhost:3306/source_db"],
                "table": ["source_table"]
              }
            ]
          }
        },
        "writer": {
          "name": "sqlserverwriter",
```

```
"parameter": {  
  "username": "sqlserver_user",  
  "password": "sqlserver_password",  
  "column": ["id", "name", "age"],  
  "connection": [  
    {  
      "jdbcUrl": "jdbc:sqlserver://localhost:1433;databaseName=target_db",  
      "table": ["target_table"]  
    }  
  ]  
}
```

----结束

数据库 MySQL 迁移 MySQL 操作验证

须知

- 源端数据库服务器 和 目标数据库服务器，分别需对DataX服务器开放对应数据库 3306访问端口，请参考[修改安全组规则](#)，配置IP地址白名单，以便DataX服务可以连接源端和目标端数据库。
- 目标端数据库如果和DataX数据库在同一子网下，模板中目标端数据库可配置私网。
- 注意：DataX迁移数据库时，reader、writer插件要匹配数据库的版本，DataX服务对应数据库的reader插件路径在DataX工作目录下的plugin/reader/{reader}/libs下，writer插件路径在DataX工作目录下的plugin/writer/{writer}/libs下。
- 本次以MySQL-8.0.39版本示例测试数据库迁移同步。

步骤1 迁移前的源端和目标端的数据库、表信息展示。

图 3-24 源端数据库、表数据

```
mysql> use mysql_origin;  
Database changed  
mysql> select * from mysql_origin_table;  
+----+-----+-----+  
| id | name   | score |  
+----+-----+-----+  
| 1  | Michael | 60.00 |  
| 2  | David   | 80.00 |  
| 3  | Emma   | 67.00 |  
| 4  | Amelia  | 93.00 |  
| 5  | James  | 83.00 |  
+----+-----+-----+  
5 rows in set (0.00 sec)  
  
mysql> █
```

图 3-25 目标端数据库、表数据

```
mysql> use mysql_target;
Database changed
mysql> select * from mysql_target_table_2;
Empty set (0.00 sec)

mysql> █
```

步骤2 迁移Json模板配置（读取源端数据库表中的id，name部分字段信息，迁移到目的端数据库）

图 3-26 Json 迁移模板

```
{
  "job": {
    "content": [
      {
        "reader": {
          "name": "mysqlreader",
          "parameter": {
            "column": ["id", "name"],
            "connection": [
              {
                "jdbcUrl": ["jdbc:mysql://[redacted]:3306/mysql_origin"],
                "table": ["mysql_origin_table"]
              }
            ],
            "password": "[redacted]",
            "username": "root",
          }
        },
        "writer": {
          "name": "mysqlwriter",
          "parameter": {
            "column": ["id", "name"],
            "connection": [
              {
                "jdbcUrl": "jdbc:mysql://1[redacted]:3306/mysql_target",
                "table": ["mysql_target_table_2"]
              }
            ],
            "password": "[redacted]",
            "username": "root"
          }
        }
      }
    ],
    "setting": {
      "speed": {
        "channel": "1"
      }
    }
  }
}
```

步骤3 输入命令：cd /opt/datax进入该目录，执行python3 bin/datax.py mysql2mysql.json，查看服务迁移结果。

图 3-27 查看执行结果

```
2024-10-12 10:10:29.422 [job-0] INFO OriginalConfPretreatmentUtil - table:mysql_target_table_2 all columns:[
id,name
]
2024-10-12 10:10:29.450 [job-0] INFO OriginalConfPretreatmentUtil - Write data [
INSERT INTO ss (id,name) VALUES(?,?)
], which jdbcUrl like:jdbc:mysql://10.10.10.10:3306/mysql_target?yearIsDateType=false&zeroDateTimeBehavior=convertToNull&rewriteBatchedStatements=true&tinyIntIsBit=false
2024-10-12 10:10:29.457 [job-0] INFO JobContainer - JobContainer starts to do prepare ...
2024-10-12 10:10:29.457 [job-0] INFO JobContainer - DataX Reader.Job [mysqlreader] do prepare work .
2024-10-12 10:10:29.458 [job-0] INFO JobContainer - DataX Writer.Job [mysqlwriter] do prepare work .
2024-10-12 10:10:29.460 [job-0] INFO JobContainer - DataX Reader.Job [mysqlreader] splits to [1] tasks.
2024-10-12 10:10:29.461 [job-0] INFO JobContainer - DataX Writer.Job [mysqlwriter] splits to [1] tasks.
2024-10-12 10:10:29.485 [job-0] INFO JobContainer - Job set Channel.Number to 5 channels.
2024-10-12 10:10:29.487 [job-0] INFO JobContainer - Scheduler starts [1] taskGroups.
2024-10-12 10:10:29.490 [job-0] INFO JobContainer - Running by standalone Mode.
2024-10-12 10:10:29.494 [taskGroup-0] INFO TaskGroupContainer - taskGroup[0] start [1] channels for [1] tasks.
2024-10-12 10:10:29.496 [taskGroup-0] INFO Channel - Channel set byte speed limit to -1, No tps activated.
2024-10-12 10:10:29.520 [taskGroup-0] INFO TaskGroupContainer - taskGroup[0] TaskId[0] attemptCount[0] is started
2024-10-12 10:10:29.522 [0-0-0-reader] INFO CommonDbmsReader$Task - Begin to read record by Sql: [select id,name from mysql_origin_table
] jdbcUrl:jdbc:mysql://10.10.10.10:3306/mysql_origin?yearIsDateType=false&zeroDateTimeBehavior=convertToNull&tinyIntIsBit=false&rewriteBatchedStatements=true].
2024-10-12 10:10:29.520 [taskGroup-0] INFO CommonDbmsReader$Task - Finished read record by Sql: [select id,name from mysql_origin_table
] jdbcUrl:jdbc:mysql://10.10.10.10:3306/mysql_origin?yearIsDateType=false&zeroDateTimeBehavior=convertToNull&tinyIntIsBit=false&rewriteBatchedStatements=true].
2024-10-12 10:10:29.921 [taskGroup-0] INFO TaskGroupContainer - taskGroup[0] completed its tasks.
2024-10-12 10:10:29.516 [job-0] INFO StandaloneJobContainerCommunicator - Total 5 records, 32 bytes | Speed 38/s, 0 records/s | Error 0 records, 0 bytes | All Task WaitWriterTime 0.000s | All Task Ma
nitReaderTime 0.025s | Percentage 100.00%
2024-10-12 10:10:39.517 [job-0] INFO AbstractScheduler - Scheduler accomplished all tasks.
2024-10-12 10:10:39.517 [job-0] INFO JobContainer - DataX Writer.Job [mysqlwriter] do post work.
2024-10-12 10:10:39.517 [job-0] INFO JobContainer - DataX Reader.Job [mysqlreader] do post work.
2024-10-12 10:10:39.517 [job-0] INFO JobContainer - DataX jobId [0] completed successfully.
2024-10-12 10:10:39.518 [job-0] INFO HookInvoker - no hook invoked, because base dir not exists or is a file: /opt/datax/hook
2024-10-12 10:10:39.520 [job-0] INFO JobContainer -
[total cpu info] =>
averageCpu          | maxDeltaCpu          | minDeltaCpu
-1.00%              | -1.00%              | -1.00%

[total gc info] =>
NAME                | totalGCCount         | maxDeltaGCCount     | minDeltaGCCount     | totalGCtime         | maxDeltaGCtime      | minDeltaGCtime
GI Young Generation | 1                    | 1                    | 1                    | 0.009s              | 0.009s              | 0.009s
GI Old Generation  | 0                    | 0                    | 0                    | 0.000s              | 0.000s              | 0.000s
GI Concurrent GC    | 2                    | 2                    | 2                    | 0.003s              | 0.003s              | 0.003s

2024-10-12 10:10:39.520 [job-0] INFO JobContainer - Perffrace not enable!
2024-10-12 10:10:39.520 [job-0] INFO StandaloneJobContainerCommunicator - Total 5 records, 32 bytes | Speed 38/s, 0 records/s | Error 0 records, 0 bytes | All Task WaitWriterTime 0.000s | All Task Ma
nitReaderTime 0.025s | Percentage 100.00%
2024-10-12 10:10:39.521 [job-0] INFO JobContainer -
任务启动时间      : 2024-10-12 10:10:29
任务结束时间      : 2024-10-12 10:10:39
任务总计耗时     : 11s
任务平均速率     : 38/s
记录写入速度     : 0rec/s
退出异常次数     : 0
退出成功次数     : 0
```

图 3-28 查看目的端数据库写入情况

```
mysql> use mysql_target;
Database changed
mysql> select * from mysql_target_table_2;
Empty set (0.00 sec)

mysql> select * from mysql_target_table_2;
+----+-----+
| id | name |
+----+-----+
| 1  | Michael |
| 2  | David |
| 3  | Emma |
| 4  | Amelia |
| 5  | James |
+----+-----+
5 rows in set (0.00 sec)

mysql>
```

----结束

数据库 MySQL 迁移 Oracle 操作验证

须知

- 源端数据库服务器需对DataX服务器开放对应数据库3306访问端口，目标端数据库服务器需对DataX服务器开放对应数据库1521访问端口，请参考[修改安全组规则](#)，配置IP地址白名单，以便DataX服务可以连接源端和目标端数据库。
- 目标端数据库如果和DataX数据库在同一子网下，模板中目标端数据库可配置私网。
- 注意：DataX迁移数据库时，reader、writer插件要匹配数据库的版本，DataX服务对应数据库的reader插件路径在DataX工作目录下的plugin/reader/{reader}}/libs下，writer插件路径在DataX工作目录下的plugin/writer/{writer}}/libs下。
- 本次以MySQL-8.0.39，Oracle-19c版本示例测试异构数据库迁移同步。

步骤1 迁移前的源端和目标端的数据库、表信息展示。

图 3-29 源端数据库、表数据

```
mysql> use mysql_origin;
Database changed
mysql> select * from mysql_origin_table;
+----+-----+-----+
| id | name  | score |
+----+-----+-----+
| 1  | Michael | 60.00 |
| 2  | David  | 80.00 |
| 3  | Emma  | 67.00 |
| 4  | Amelia | 93.00 |
| 5  | James  | 83.00 |
+----+-----+-----+
5 rows in set (0.00 sec)

mysql>
```

图 3-30 目标端数据库、表数据

```
SQL> conn oracle_target/123456;
Connected.
SQL> select * from oracle_target_table;

no rows selected

SQL>
```

步骤2 迁移Json模板配置（读取源端数据库表中的id，name部分字段信息，迁移到目的端Oracle数据库）

图 3-31 Json 迁移模板

```
{
  "job": {
    "content": [
      {
        "reader": {
          "name": "mysqlreader",
          "parameter": {
            "column": ["id", "name"],
            "connection": [
              {
                "jdbcUrl": ["jdbc:mysql://[redacted]:3306/mysql_origin"],
                "table": ["mysql_origin_table"]
              }
            ],
            "password": "[redacted]",
            "username": "root",
          }
        },
        "writer": {
          "name": "oraclewriter",
          "parameter": {
            "column": ["id", "name"],
            "connection": [
              {
                "jdbcUrl": "jdbc:oracle:thin:@//[redacted]:1521/orcl",
                "table": ["oracle_target_table"]
              }
            ],
            "password": "[redacted]",
            "username": "oracle_target"
          }
        }
      }
    ],
    "setting": {
      "speed": {
        "channel": "1"
      }
    }
  }
}
```

步骤3 输入命令：`cd /opt/datax`进入该目录，执行`python3 bin/datax.py mysql2oracle.json`，查看服务迁移结果。

图 3-32 查看执行结果

```
2024-10-12 11:28:29.411 [job-0] INFO OriginalConfPretreatmentUtil - table:[mysql_origin_table] has columns:[id,name,score].
2024-10-12 11:28:29.735 [job-0] INFO OriginalConfPretreatmentUtil - table:[oracle_target_table] all columns:[ID,NAME]
2024-10-12 11:28:29.756 [job-0] INFO OriginalConfPretreatmentUtil - Write data [
INSERT INTO xs (id,name) VALUES(?,?)
] which jdbcUrl likes[jdbc:oracle:thin://[redacted]:1521/orcl]
2024-10-12 11:28:29.756 [job-0] INFO JobContainer - JobContainer starts to do prepare ...
2024-10-12 11:28:29.757 [job-0] INFO JobContainer - Datax Reader:Job [mysqlreader] do prepare work ...
2024-10-12 11:28:29.757 [job-0] INFO JobContainer - Datax Writer:Job [oraclewriter] do prepare work ...
2024-10-12 11:28:29.757 [job-0] INFO JobContainer - JobContainer starts to do split ...
2024-10-12 11:28:29.757 [job-0] INFO JobContainer - Job set Channel-Number to 1 channels.
2024-10-12 11:28:29.760 [job-0] INFO JobContainer - Datax Reader:Job [mysqlreader] splits to [1] tasks.
2024-10-12 11:28:29.760 [job-0] INFO JobContainer - Datax Writer:Job [oraclewriter] splits to [1] tasks.
2024-10-12 11:28:29.768 [job-0] INFO JobContainer - JobContainer starts to do schedule ...
2024-10-12 11:28:29.788 [job-0] INFO JobContainer - Scheduler starts [1] taskGroups.
2024-10-12 11:28:29.791 [job-0] INFO JobContainer - Running by standalone Mode.
2024-10-12 11:28:29.795 [taskGroup-0] INFO TaskGroupContainer - taskGroup[id:0] start [1] channels for [1] tasks.
2024-10-12 11:28:29.798 [taskGroup-0] INFO Channel - Channel set byte_speed_limit to -1, No bps activated.
2024-10-12 11:28:29.798 [taskGroup-0] INFO Channel - Channel set record_speed_limit to -1, No tps activated.
2024-10-12 11:28:29.820 [0-0-reader] INFO CommonHdfsReaderTask - begin to read record by sql:[select id,name from mysql_origin_table]
2024-10-12 11:28:29.823 [0-0-reader] INFO CommonHdfsReaderTask - Finished read record by sql:[select id,name from mysql_origin_table]
2024-10-12 11:28:30.021 [taskGroup-0] INFO TaskGroupContainer - taskGroup[0] attemptCount[1] is started, used:212ms
2024-10-12 11:28:30.021 [taskGroup-0] INFO TaskGroupContainer - taskGroup[0] is succeeded, used:212ms
2024-10-12 11:28:30.813 [job-0] INFO StandAloneJobContainerCommunicator - Total 5 records, 32 bytes | Speed 38/s, 0 records/s | Error 0 records, 0 bytes | All Task WaitWriterTime 0.000s | All Task WaitReaderTime 0.000s | Percentage 100.00%
2024-10-12 11:28:30.813 [job-0] INFO AbstractScheduler - Scheduler accomplished all tasks.
2024-10-12 11:28:30.814 [job-0] INFO JobContainer - Datax Writer:Job [oraclewriter] do post work.
2024-10-12 11:28:30.816 [job-0] INFO JobContainer - Datax Reader:Job [mysqlreader] do post work.
2024-10-12 11:28:30.816 [job-0] INFO JobContainer - Datax jobId [0] completed successfully.
2024-10-12 11:28:30.816 [job-0] INFO HookInvoker - No hook invoked, because base dir not exists or its a file: /opt/datax/hook
2024-10-12 11:28:30.818 [job-0] INFO JobContainer -
[total cpu info] =>
averageCpu | maxDeltaCpu | minDeltaCpu
1.00% | 1.00% | 1.00%
[total gc info] =>
NAME | totalGCCount | maxDeltaGCCount | minDeltaGCCount | totalGCTime | maxDeltaGCTime | minDeltaGCTime
G1 Young Generation | 1 | 0 | 0 | 0.019s | 0.019s | 0.019s
G1 Old Generation | 0 | 0 | 0 | 0.000s | 0.000s | 0.000s
G1 Concurrent GC | 2 | 2 | 2 | 0.002s | 0.002s | 0.002s
2024-10-12 11:28:30.818 [job-0] INFO JobContainer - PerfTrace not enable!
2024-10-12 11:28:30.819 [job-0] INFO StandAloneJobContainerCommunicator - Total 5 records, 32 bytes | Speed 38/s, 0 records/s | Error 0 records, 0 bytes | All Task WaitWriterTime 0.000s | All Task WaitReaderTime 0.000s | Percentage 100.00%
2024-10-12 11:28:30.820 [job-0] INFO JobContainer -
任务启动时刻 : 2024-10-12 11:28:28
任务结束时刻 : 2024-10-12 11:28:39
任务总记录数 : 116
任务平均速率 : 38/s
任务平均流量 : 0rec/s
退出记录总数 : 0
读写失败总数 : 0
```

图 3-33 查看目的端数据库写入情况

```
SQL> conn oracle_target/123456;
Connected.
SQL> select * from oracle_target_table;

no rows selected

SQL> select * from oracle_target_table;

-----
          ID
-----
NAME
-----
          1
Michael
          2
David
          3
Emma
          4
Amelia
          5
James
```

----结束

3.4 快速卸载

须知

- 如选择“新建VPC”一键部署该方案，目标数据库服务器和DataX服务器在同一VPC下，须对目标数据库服务器**切换其他VPC**后，即可按照下述步骤进行卸载。

步骤1 登录[资源编排服务 RFS](#)，进入“资源栈”，选择创建的资源栈名称，单击“删除”。

图 3-34 一键卸载



步骤2 在弹出的删除资源栈确认框中，输入"Delete"，单击“确定”，即可卸载解决方案。

图 3-35 删除资源栈确认



----结束

4 附录

名词解释

基本概念、云服务简介、专有名词解释

- 华为云Flexus云服务器X实例：Flexus云服务器X实例是新一代面向中小企业和开发者打造的柔性算力云服务器。Flexus云服务器X实例功能接近ECS，同时还具备独有特点，例如Flexus云服务器X实例具有更灵活的vCPU内存配比、支持热变配不中断业务变更规格、支持性能模式等。
- 弹性公网IP EIP：提供独立的公网IP资源，包括公网IP地址与公网出口带宽服务。可以与弹性云服务器、裸金属服务器、虚拟IP、弹性负载均衡、NAT网关等资源灵活地绑定及解绑。

5 修订记录

表 5-1 修订记录

发布日期	修订记录
2024-10-17	第一次正式发布。