

Cloud Data Migration

Getting Started

Issue 01
Date 2022-09-30



Copyright © Huawei Technologies Co., Ltd. 2022. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Technologies Co., Ltd.

Trademarks and Permissions



HUAWEI and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

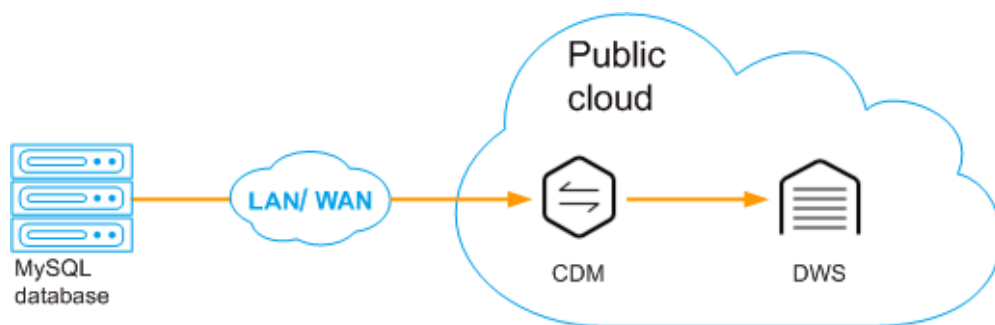
Contents

1 Overview.....	1
2 Step 1: Creating a Cluster.....	2
3 Step 2: Creating Links.....	5
4 Step 3: Creating and Executing a Job.....	11
5 Step 4: Viewing Job Execution Result.....	14

1 Overview

This section describes how to use CDM to migrate the tables in the on-premises MySQL database to DWS, thereby helping you get familiar with CDM. [Figure 1-1](#) shows the specific scenario.

Figure 1-1 Migrating data from a local MySQL database to DWS



The procedure of using CDM is as follows:

1. [Creating a CDM Cluster](#)
2. [Creating Links](#)
3. [Creating and Executing a Job](#)
4. [Viewing Job Execution Result](#)

2 Step 1: Creating a Cluster

Scenario

This section describes how to create a CDM cluster to migrate data between an on-premises MySQL database and DWS.

Prerequisites

You have obtained the region, VPC, subnet, and security group of the data warehouse cluster.

Procedure

Step 1 Log in to the CDM management console.

Step 2 Click **buy CDM Cluster** and set the following parameters:

- **Region:** Select the region where the CDM cluster resides. Resources in different regions cannot communicate with each other. The region must be the same as that of the data warehouse cluster.
- **AZ:** An AZ is a physical region where resources use independent power supply and networks. AZs are physically isolated but interconnected through an internal network. In this case, select **AZ2**.
- **Name:** The cluster name must start with a letter and contains 4 to 64 characters consisting of letters, digits, hyphens (-), and underscores (_). It cannot contain special characters. For example, **cdm-aff1**.
- **Instance Type:** Select an instance flavor based on your service data volume.
 - **cdm.large:** 8 vCPUs and 16 GB of memory. The maximum and assured bandwidths are 3 Gbit/s and 0.8 Gbit/s. Up to 16 jobs can be executed concurrently.
 - **cdm.xlarge:** 16 vCPUs and 32 GB of memory. The maximum and assured bandwidths are 10 Gbit/s and 4 Gbit/s. Up to 32 jobs can be executed concurrently. This flavor is suitable for migrating terabytes of data that requires a bandwidth of 10GE.
 - **cdm.4xlarge:** 64 vCPUs and 128 GB of memory. The maximum and assured bandwidths are 40 Gbit/s and 36 Gbit/s. Up to 64 jobs can be executed concurrently.

- **VPC:** Select the VPC where DWS resides.
- **Subnet:** You are advised to use the same subnet as that of DWS.
- **Security Group:** You are advised to use the security group as that of DWS.
You can select a subnet and security group that are different from those of DWS. In this case, configure the security group rules to allow the CDM cluster to properly access DWS.
- Retain the default values for other parameters.

Figure 2-1 Creating a cluster

The screenshot shows the 'Configure' step of a cluster creation wizard. It includes the following fields and options:

- Region:** CN North-Beijing1
- Project:** Project
- AZ:** AZ1, AZ2
- Name:** cdm-df33
- Version:** 2.9.1.100
- Instance Type:**

Flavor	vCPUs/Memory	Assured/Maximum Ba...	Concurrent Extractors
<input checked="" type="radio"/> cdm.large	8 vCPUs 16 GB	0.8/3 Gbps	20
<input type="radio"/> cdm.xlarge	16 vCPUs 32 GB	4/10 Gbps	100

Selected Instance Specifications: cdm.large | 8 vCPUs | 16 GB
- Quantity:** 1
- VPC:** vpc-css (View VPC)
- Subnet:** subnet-f129(192.168.0.0/24)
- Security Group:** mrs_mrs_MJ6_Nmz (View Security Group)

Figure 2-2 Creating another cluster

The screenshot shows the 'Configure' step for another cluster. It includes the following fields and options:

- Subnet:** subnet-7630(172.16.0.0/24)
- Security Group:** rds-dayu-demo (View Security Group)
- Enterprise Project:** [Empty]
- Auto Shutdown:** Auto Shutdown and Scheduled Startup/Shutdown cannot be enabled at the same time.
- Scheduled Startup:**
- Scheduled Shutdown:**
- Notification:**

Step 3 Check the current configuration and click **Buy Now** to go to the page for confirming the order.

NOTE

You cannot modify the flavor of an existing cluster. If you require a higher flavor, create a cluster with your desired flavor.

Step 4 Click **Submit**. The system starts to create a CDM cluster. You can view the creation progress on the **Cluster Management** page.

----End

3 Step 2: Creating Links

Scenario

Before migrating the local MySQL database to DWS, create two links:

1. MySQL link: used to connect to the on-premises MySQL database.
2. DWS link: used to connect to the DWS database.

CDM needs to access the on-premises data source. Therefore, before creating links, bind an EIP to the CDM cluster.

Prerequisites

- Your on-premises MySQL database can be accessed using the public IP address.
- You have obtained the IP address, port, database name, username, and password for connecting to the MySQL database. In addition, the user has the read, write, and delete permissions on the MySQL database.
- You have a DWS instance and have obtained the IP address, port number, database name, username, and password for accessing DWS. In addition, the user has the read, write, and delete permissions for the DWS database.
- You have uploaded the MySQL database driver by referring to [Managing Drivers](#).

Binding an EIP to a CDM Cluster

Step 1 Log in to the CDM management console.

Step 2 Locate a CDM cluster and click **Bind EIP** in the **Operation** column. In the displayed dialog box, select an EIP.

If no EIP is available, create one as prompted.

Step 3 Click **OK**.

----End

Creating a MySQL Link

- Step 1** In the left navigation pane, choose **Cluster Management**. Locate the **cdm-aff1** cluster created in [Step 1: Creating a Cluster](#).
- Step 2** In the **Operation** column, click **Bind EIP**, and select and bind an EIP to the cluster.

 **NOTE**

If SSL encryption is configured for the access channel of a local data source, CDM cannot connect to the data source using the EIP.

- Step 3** Click **Job Management** in the **Operation** column of the CDM cluster. On the displayed page, click the **Links** tab and then **Create Link**. The **Select Connector** page is displayed.

Figure 3-1 Selecting a connector



- Step 4** Select **MySQL** and click **Next**. On the page that is displayed, configure MySQL link parameters.

Figure 3-2 Creating a MySQL link

* Name [Configuration Guide](#)

* Connector

Database Type

* Database Server

* Port

* Database Name

* Username

* Password

Use Local API

Use Agent

Driver Version [mysql-connector-java-5.1.48.jar](#) [Upload](#) | [Copy from SFTP](#)

[Hide Advanced Attributes](#)

Fetch Size

Commit Size

Link Attributes

Reference Sign

Batch Size

Click **Show Advanced Attributes** and set optional parameters. For details, see [Link to Relational Databases](#). Retain the default values of the optional parameters and configure the mandatory parameters according to [Table 3-1](#).

Table 3-1 MySQL link parameters

Parameter	Description	Example Value
Name	Unique link name	mysqllink

Parameter	Description	Example Value
Database Server	IP address or domain name of the MySQL database server	192.168.1.110
Port	MySQL database port	3306
Database Name	Name of the MySQL database	sqoop
Username	User who has the read, write, and delete permissions on the MySQL database	admin
Password	Password of the user	-
Use Local API	Whether to use the local API of the database for acceleration. (The system attempts to enable the local_infile system variable of the MySQL database.)	Yes
Use Agent	Whether to extract data from the data source through an agent	Yes
local_infile Character Set	When using local_infile to import data to MySQL, you can configure the encoding format.	utf8
Driver Version	A driver version that adapts to MySQL	-
Agent	Click Select to select the agent created in Connecting to an Agent .	-
Fetch Size	(Optional) Displayed when you click Show Advanced Attributes . Number of rows obtained by each request. Set this parameter based on the data source and the job's data size. If the value is either too large or too small, the job may run for a long time.	1000
Commit Size	(Optional) Displayed when you click Show Advanced Attributes . Number of records submitted each time. Set this parameter based on the data destination and the job's data size. If the value is either too large or too small, the job may run for a long time.	1000
Link Attributes	Custom attributes of the link	useCompression=true

Parameter	Description	Example Value
Reference Sign	Delimiter used to separate referenced table names or column names This parameter is left blank by default.	'
Batch Size	Number of rows written each time. It should be less than Commit Size. When the number of rows written reaches the value of Commit Size, the rows will be committed to the database.	100

Step 5 Click **Save**. The **Link Management** page is displayed.

NOTE

If an error occurs during the saving, the security settings of the MySQL database are incorrect. In this case, you need to enable the EIP of the CDM cluster to access the MySQL database.

----End

Creating a DWS Link

Step 1 Click **Job Management** in the **Operation** column of the CDM cluster. On the displayed page, click the **Links** tab and then **Create Link**. The **Select Connector** page is displayed.

Figure 3-3 Selecting a connector



Step 2 Select **Data Warehouse Service** and click **Next** to configure the DWS link parameters. Set the mandatory parameters listed in **Table 3-2** and retain the default values for the optional parameters.

Table 3-2 DWS link parameters

Parameter	Description	Example Value
Name	Unique link name	dwslink
Database Server	IP address or domain name of the DWS database server	192.168.0.3
Port	DWS database port	8000
Database Name	Name of the DWS database	db_demo
Username	User who has the read, write, and delete permissions on the DWS database	dbadmin
Password	Password of the user	-
Use Agent	Whether to extract data from the data source through an agent	Yes
Agent	Click Select to select the agent created in Connecting to an Agent .	-

Step 3 Click **Save**.

----**End**

4 Step 3: Creating and Executing a Job

Scenario

This section describes how to create a table migration job to migrate data tables from an on-premises MySQL database to DWS.

Procedure

- Step 1** On the **Cluster Management** page, locate the **cdm-aff1** cluster created in [Step 1: Creating a Cluster](#).
- Step 2** Click **Job Management** in the **Operation** column of the CDM cluster.
- Step 3** Choose **Table/File Migration > Create Job**, and configure the required job information.

Figure 4-1 Creating a job

The screenshot displays the 'Job Configuration' window. At the top, there is a 'Job Name' field with the value 'mysql2dws'. Below this are two main configuration panels: 'Source Job Configuration' and 'Destination Job Configuration'. The 'Source Job Configuration' panel includes a 'Source Link Name' dropdown set to 'mysql_link', a 'Use SQL Statement' toggle set to 'Yes', and 'Schema/Table Space' and 'Table Name' input fields. The 'Destination Job Configuration' panel includes a 'Destination Link Name' dropdown set to 'dws_link', a 'Schema/Table Space' input field, an 'Auto Table Creation' dropdown set to 'Non-auto Creation', a 'Table Name' input field, a 'Clear Data Before Import' dropdown set to 'Do not clear', and an 'Import Mode' dropdown set to 'COPY'. At the bottom of the window, there are 'Cancel' and 'Next' buttons.

- **Job Name:** Enter a unique job name, for example, **mysql2dws**.
- **Source Job Configuration**
 - **Source Link Name:** Select the **mysqllink** link created in [Step 2: Creating Links](#).

- **Use SQL:** Select **No**.
- **Schema/Tablespace:** Select the MySQL database from which the table is to be exported.
- **Table Name:** Select the table from which data is to be exported.
- Retain the default values of other optional parameters. For details, see [From a Common Relational Database](#).
- **Destination Job Configuration**
 - **Destination Link Name:** Select the **dwslink** link created in [Step 2: Creating Links](#).
 - **Schema/Tablespace:** Select the database to which data is to be imported.
 - **Auto Table Creation:** Select **Auto creation**. If the table specified by **Table Name** does not exist, CDM automatically creates the table in the DWS database.
 - **Table Name:** Select the table to which data is to be imported.
 - **Advanced Attributes > Extend Field Length:** Select **Yes**. The encoding methods for Chinese characters stored in MySQL and DWS are different, and the required lengths are different as well. A Chinese character may occupy three bytes in UTF-8 encoding. If this parameter is set to **Yes**, the length of the fields of the character type will be set to three times of its original length when a table is automatically created. This prevents errors caused by insufficient lengths of the characters in DWS tables.
 - Retain the default values for other optional parameters. For details, see [To DWS](#).

Step 4 Click **Next**. The **Map Field** tab page is displayed. CDM automatically maps table fields at the migration source and destination. Check whether the field mapping is correct.

- If the field mapping is incorrect, click the row where the field is located and drag the field to adjust the mapping.
- When importing data to DWS, you need to manually select the distribution columns of DWS. You are advised to select the distribution columns according to the following principles:
 - a. Use the primary key as the distribution column.
 - b. If multiple data segments are combined as primary keys, specify all primary keys as the distribution column.
 - c. In the scenario where no primary key is available, if no distribution column is selected, DWS uses the first column as the distribution column by default. As a result, data skew risks exist.
- If you want to convert the content of the source fields, perform the operations in this step. For details, see [Converting Fields](#). In this example, field conversion is not required.

Figure 4-2 Field mapping

Source Field				Destination Field		
Name	Example Value	Type	Operation	Name	Type	Operation
owner		string	☞ Q ☞	owner	VARCHAR(10485760)	☞
table_name		string	☞ Q ☞	table_name	VARCHAR(10485760)	☞

Step 5 Click **Next** and set task parameters. Generally, retain the default values of all parameters.

In this step, you can configure the following optional functions:

- **Retry Upon Failure:** If the job fails to be executed, you can determine whether to automatically retry. Retain the default value **Never**.
- **Group:** Select the group to which the job belongs. The default group is **DEFAULT**. On the **Job Management** page, jobs can be displayed, started, or exported by group.
- **Schedule Execution:** To configure scheduled jobs, see [Scheduling Job Execution](#). Retain the default value **No**.
- **Concurrent Extractors:** Enter the number of extractors to be concurrently executed. Retain the default value **1**.
- **Write Dirty Data:** Specify this parameter if data that fails to be processed or filtered out during job execution needs to be written to OBS for future viewing. Before writing dirty data, create an OBS link. Retain the default value **No** so that dirty data is not recorded.
- **Delete Job After Completion:** Retain the default value **Do not delete**.

Step 6 Click **Save and Run**. CDM starts to execute the job immediately.

----End

5 Step 4: Viewing Job Execution Result

Scenario

This section describes how to view a job's execution results and its historical information in the latest 90 days, including the number of written rows, read rows, written bytes, written files, and log information.

Procedure

- Step 1** On the **Cluster Management** page, locate the **cdm-aff1** cluster created in [Step 1: Creating a Cluster](#).
- Step 2** Click **Job Management** in the **Operation** column of the CDM cluster.
- Step 3** Locate the **mysql_dws** job created in [Step 3: Creating and Executing a Job](#) and view the running status of the job.

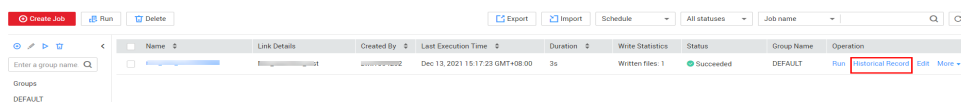
NOTE

The job status can be **New**, **Pending**, **Booting**, **Running**, **Failed**, or **Succeeded**.

Pending indicates that the job is waiting to be scheduled by the system, and **Booting** indicates that the data to be migrated is being analyzed.

- Step 4** Click **Historical Record** to view the number of written rows, number of read rows, number of written bytes, and number of written files.

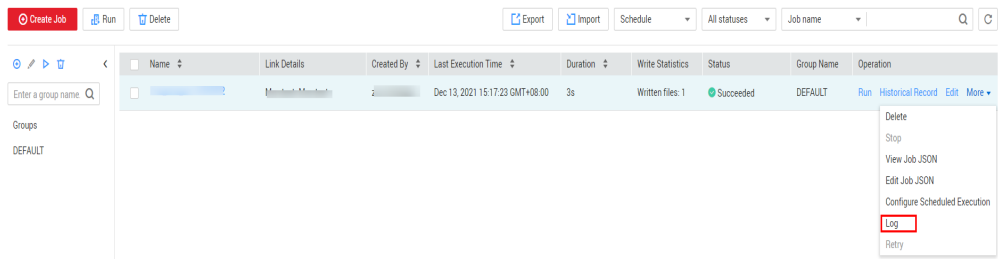
Figure 5-1 Viewing the historical records



- Step 5** Click **Log** to view the job logs.

Alternatively, in the **Operation** column, choose **More > Log** to view the latest logs of the job.

Figure 5-2 Viewing job logs



----End