

CloudTable Service

Best Practices

Issue 01
Date 2026-02-27



Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2026. 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 Cloud Computing Technologies Co., Ltd.

Trademarks and Permissions



HUAWEI and other Huawei trademarks are the property 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 Cloud 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 Importing Data..... | 1 |
| 1.1 Using a DLI Flink Job to Synchronize MRS Kafka Data to a CloudTable HBase Cluster in Real Time..... | 1 |
| 1.2 Using a DLI Flink Job to Synchronize MRS Kafka Data to a CloudTable ClickHouse Cluster in Real Time..... | 5 |

1 Importing Data

1.1 Using a DLI Flink Job to Synchronize MRS Kafka Data to a CloudTable HBase Cluster in Real Time

This section describes the best practices of real-time data synchronization. You can use DLI Flink jobs to synchronize MRS Kafka data to HBase in real time.

- For details about DLI, see [DLI Service Overview](#).
- For details about Kafka, see [MapReduce Service Overview](#).

Figure 1-1 Data synchronization process



Constraints

- Kerberos authentication is not enabled for the MRS cluster.
- To ensure network connectivity, the security group, region, VPC, and subnet of the MRS cluster must be the same as those of the CloudTable cluster.
- You need to add the elastic resource network segment subnet associated with the DLI queue to the inbound rules of both the MRS and CloudTable security groups to establish a data source connection. For details, see [Creating an Enhanced Datasource Connection](#).
- You must enable the network connectivity between the upstream and downstream of DLI. For details, see [Testing Address Connectivity](#).

Procedure

The general procedure is as follows:

1. [Step 1: Create a CloudTable HBase Cluster](#)
2. [Step 2: Create a Flink Job in the MRS Cluster to Generate Data](#)
3. [Step 3: Create a DLI Flink Job to Synchronize Data](#)

4. [Step 4: Verify the Result](#)

Preparations

- You have signed up for a HUAWEI ID and enable Huawei Cloud services. For details, see [Signing Up for a HUAWEI ID and Enabling Huawei Cloud Services](#). The account cannot be in arrears or frozen.
- You have created a VPC and subnet. For details, see [Creating a VPC and Subnet](#).

Step 1: Create a CloudTable HBase Cluster

1. Log in to the CloudTable console and [create an HBase cluster](#).
2. [Install the client](#).
3. Start the Shell to access the cluster. Run the `bin/hbase shell` command to start the shell to access the cluster.
4. Create the **order** table.

```
create 'order', {NAME => 'detail'}
```

Step 2: Create a Flink Job in the MRS Cluster to Generate Data

1. Create an [MRS cluster](#).
2. Log in to Manager and choose **Cluster > Flink > Dashboard**.
3. Click the link on the right of **Flink WebUI** to access the Flink web UI.
4. On the Flink web UI, create a Flink task to generate data.
 - a. Click **Create Job** on the **Job Management** page. The **Create Job** page is displayed.
 - b. Set parameters and click **OK** to create a Flink SQL job. To modify a SQL statement, click **Develop** in the **Operation** column and add the following command on the SQL page:

NOTE

ip:port: IP address and port number

- To obtain the IP address, log in to Manager of the cluster and choose **Cluster > Kafka > Instance**. On the displayed page, view the **Management IP Address** of Broker.
- To obtain the port number, click **Configurations**. On the configuration page that is displayed, search for **port** and obtain the port number (which is the PLAINTEXT protocol port number listened by the Broker service).
- You are advised to add multiple IP addresses and port numbers to the **properties.bootstrap.servers** parameter to prevent job running failures caused by unstable network or other reasons.

SQL statement examples

```
CREATE TABLE IF NOT EXISTS `lineorder_hbase` (  
  `order_id` string,  
  `order_channel` string,  
  `order_time` string,  
  `pay_amount` double,  
  `real_pay` double,  
  `pay_time` string,  
  `user_id` string,  
  `user_name` string,  
  `area_id` string
```

```
) WITH (  
'connector' = 'kafka',  
'topic' = 'test_flink',  
'properties.bootstrap.servers' = 'ip:port',  
'value.format' = 'json',  
'properties.sasl.kerberos.service.name' = 'kafka'  
);  
CREATE TABLE lineorder_datagen (  
  `order_id` string,  
  `order_channel` string,  
  `order_time` string,  
  `pay_amount` double,  
  `real_pay` double,  
  `pay_time` string,  
  `user_id` string,  
  `user_name` string,  
  `area_id` string  
) WITH (  
'connector' = 'datagen',  
'rows-per-second' = '1000'  
);  
INSERT INTO  
lineorder_hbase  
SELECT  
*  
FROM  
lineorder_datagen;
```

- c. Return to the **Job Management** page and click **Start** in the **Operation** column. If the job status is **Running**, the job is successfully executed.

Step 3: Create a DLI Flink Job to Synchronize Data

1. Create elastic resource pools and queues. For details, see [Creating an Elastic Resource Pool and Creating Queues Within It](#) in the .
2. Create a data source connection. For details, see [Creating an Enhanced Datasource Connection](#).
3. Test the connectivity between DLI and the upstream MRS Kafka and between DLI and the downstream CloudTable HBase.
 - a. After the elastic resource pool and queue are created, choose **Resources > Queue Management**. On the displayed page, test the address connectivity. For details, see [Testing Address Connectivity](#).
 - b. To obtain the upstream IP address and port number, go to Manager of the cluster and choose **Cluster > Kafka > Instance**. On the displayed page, view the **Management IP Address** of Broker. Click **Configurations**. On the configuration page that is displayed, search for **port** and obtain the port number (which is the PLAINTEXT protocol port number listened by the Broker service).
 - c. Obtain the downstream IP address and port.
 - i. To obtain the IP address, go to the **Details** page. Obtain the domain name from **ZK Link (Intranet)** under **Cluster Information**. Run the following command to resolve the IP address:

```
ping Access domain name
```
 - ii. To obtain the port number, go to the **Details** page. Obtain the port from **ZK Link (Intranet)** under **Cluster Information**.
4. Create a Flink job. For details, see [Submitting a Flink Job Using DLI](#).
5. Select the Flink job created in [4](#), click **Edit** in the **Operation** column, and add SQL statements for data synchronization.

```
CREATE TABLE orders (  
  order_id string,  
  order_channel string,  
  order_time string,  
  pay_amount double,  
  real_pay double,  
  pay_time string,  
  user_id string,  
  user_name string,  
  area_id string  
) WITH (  
  'connector' = 'kafka',  
  'topic' = 'test_flink',  
  'properties.bootstrap.servers' = 'ip:port',  
  'properties.group.id' = 'testGroup_1',  
  'scan.startup.mode' = 'latest-offset',  
  'format' = 'json'  
);  
create table hbaseSink(  
  order_id string,  
  detail Row(  
    order_channel string,  
    order_time string,  
    pay_amount double,  
    real_pay double,  
    pay_time string,  
    user_id string,  
    user_name string,  
    area_id string)  
) with (  
  'connector' = 'hbase-2.2',  
  'table-name' = 'order',  
  'zookeeper.quorum' = 'ip:port',  
  'sink.buffer-flush.max-rows' = '1'  
);  
insert into hbaseSink select order_id,  
Row(order_channel,order_time,pay_amount,real_pay,pay_time,user_id,user_name,area_id) from orders;
```

6. Click **Format** and click **Save**.

NOTICE

Click **Format** to format the SQL code. Otherwise, new null characters may be introduced during code copy and paste, causing job execution failures.

7. On the DLI management console, choose **Job Management** > **Flink Jobs**.
8. Click **Start** in the **Operation** column to start the job created in 4. If the job status is **Running**, the job is successfully executed.

Step 4: Verify the Result

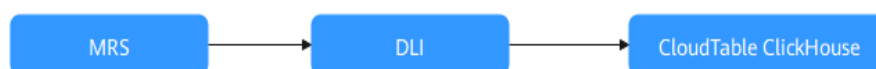
1. After the MRS Flink and DLI tasks are successfully executed, return to the command window of the HBase cluster and start the downstream HBase shell client.
scan 'order'
2. Check whether the data source is continuously updated.

1.2 Using a DLI Flink Job to Synchronize MRS Kafka Data to a CloudTable ClickHouse Cluster in Real Time

This section describes the best practices of real-time data synchronization. You can use DLI Flink jobs to synchronize data generated by MRS Kafka jobs to ClickHouse in real time.

- For details about DLI, see [DLI Service Overview](#).
- For details about Kafka, see [MapReduce Service Overview](#).

Figure 1-2 Data synchronization process



Constraints

- Kerberos authentication is not enabled for the MRS cluster.
- To ensure network connectivity, the security group, region, VPC, and subnet of the MRS cluster must be the same as those of the CloudTable cluster.
- You need to add the elastic resource network segment subnet associated with the DLI queue to the inbound rules of both the MRS and CloudTable security groups to establish a data source connection. For details, see [Creating an Enhanced Datasource Connection](#).
- You must enable the network connectivity between the upstream and downstream of DLI. For details, see [Testing Address Connectivity](#).

Procedure

The general procedure is as follows:

1. [Step 1: Create a CloudTable ClickHouse Cluster](#)
2. [Step 2: Create a Flink Job in the MRS Cluster to Generate Data](#)
3. [Step 3: Create a DLI Flink Job to Synchronize Data](#)
4. [Step 4: Verify the Result](#)

Preparations

- You have signed up for a HUAWEI ID and enable Huawei Cloud services. For details, see [Signing Up for a HUAWEI ID and Enabling Huawei Cloud Services](#). The account cannot be in arrears or frozen.
- You have created a VPC and subnet. For details, see [Creating a VPC and Subnet](#).

Step 1: Create a CloudTable ClickHouse Cluster

1. Log in to the CloudTable console and [create a ClickHouse cluster](#).
2. Download a client. In the navigation pane on the left, choose **Help**. In the right pane, click **Download Client** and **Client Verification File**.

3. [Install the client, verify the client](#), and connect to the cluster.

4. Create a Flink database.

```
create database flink;
```

Use the Flink database.

```
use flink;
```

5. Create the **flink.order** table.

```
create table flink.order(order_id String,order_channel String,order_time String,pay_amount  
Float64,real_pay Float64,pay_time String,user_id String,user_name String,area_id String) ENGINE =  
ReplicatedMergeTree('/clickhouse/tables/{shard}/flink/order', '{replica}')ORDER BY order_id;
```

6. Check whether the table is successfully created:

```
select * from flink.order;
```

Step 2: Create a Flink Job in the MRS Cluster to Generate Data

1. Create an [MRS cluster](#).
2. Log in to Manager and choose **Cluster > Flink > Dashboard**.
3. Click the link on the right of **Flink WebUI** to access the Flink web UI.
4. On the Flink web UI, create a Flink task to generate data.
 - a. Click **Create Job** on the **Job Management** page. The **Create Job** page is displayed.
 - b. Set parameters and click **OK** to create a Flink SQL job. To modify a SQL statement, click **Develop** in the **Operation** column and add the following command on the SQL page:

NOTE

Obtain the IP address and port.

- To obtain the IP address, log in to Manager of the cluster and choose **Cluster > Kafka > Instance**. On the displayed page, view the **Management IP Address** of Broker.
- To obtain the port number, click **Configurations**. On the configuration page that is displayed, search for **port** and obtain the port number (which is the PLAINTEXT protocol port number listened by the Broker service).
- You are advised to add multiple IP addresses and port numbers to the **properties.bootstrap.servers** parameter to prevent job running failures caused by unstable network or other reasons.

SQL statement examples

```
CREATE TABLE IF NOT EXISTS `lineorder_ck` (  
  `order_id` string,  
  `order_channel` string,  
  `order_time` string,  
  `pay_amount` double,  
  `real_pay` double,  
  `pay_time` string,  
  `user_id` string,  
  `user_name` string,  
  `area_id` string  
) WITH (  
  'connector' = 'kafka',  
  'topic' = 'test_flink',  
  'properties.bootstrap.servers' = 'ip:port',  
  'value.format' = 'json',  
  'properties.sasl.kerberos.service.name' = 'kafka'  
);  
CREATE TABLE lineorder_datagen (  
  `order_id` string,  
  `order_channel` string,
```

```
`order_time` string,  
`pay_amount` double,  
`real_pay` double,  
`pay_time` string,  
`user_id` string,  
`user_name` string,  
`area_id` string  
) WITH (  
  'connector' = 'datagen',  
  'rows-per-second' = '1000'  
);  
INSERT INTO  
lineorder_ck  
SELECT  
*  
FROM  
lineorder_datagen;
```

- c. Return to the **Job Management** page and click **Start** in the **Operation** column. If the job status is **Running**, the job is successfully executed.

Step 3: Create a DLI Flink Job to Synchronize Data

1. Create elastic resource pools and queues. For details, see [Creating an Elastic Resource Pool and Creating Queues Within It](#) in the .
2. Create a data source connection. For details, see [Creating an Enhanced Datasource Connection](#).
3. Test the connectivity between DLI and the upstream MRS Kafka and between DLI and the downstream CloudTable ClickHouse.
 - a. After the elastic resource pool and queue are created, choose **Resources > Queue Management**. On the displayed page, test the address connectivity. For details, see [Testing Address Connectivity](#).
 - b. To obtain the upstream IP address and port number, go to Manager of the cluster and choose **Cluster > Kafka > Instance**. On the displayed page, view the **Management IP Address** of Broker. Click **Configurations**. On the configuration page that is displayed, search for **port** and obtain the port number (which is the PLAINTEXT protocol port number listened by the Broker service).
 - c. To obtain the downstream IP address and port number, go to the **Details** page.
4. Create a Flink job. For details, see [Submitting a Flink Job Using DLI](#).
5. Select the Flink job created in [4](#), click **Edit** in the **Operation** column, and add SQL statements for data synchronization.

```
create table orders (  
  order_id string,  
  order_channel string,  
  order_time string,  
  pay_amount double,  
  real_pay double,  
  pay_time string,  
  user_id string,  
  user_name string,  
  area_id string  
) WITH (  
  'connector' = 'kafka',  
  'topic' = 'test_flink',  
  'properties.bootstrap.servers' = 'ip:port',  
  'properties.group.id' = 'testGroup_1',  
  'scan.startup.mode' = 'latest-offset',  
  'format' = 'json'
```

