ROMA Connect

FAQs

Issue 01

Date 2025-07-22

Contents

1 Instance Management

1.1 Region and AZ

Definition

A region and availability zone (AZ) identify the location of a data center. You can create resources in a specific region and AZ.

- Regions are divided based on geographical location and network latency. Public services, such as Elastic Cloud Server (ECS), Elastic Volume Service (EVS), Object Storage Service (OBS), Virtual Private Cloud (VPC), Elastic IP (EIP), and Image Management Service (IMS), are shared within the same region. Regions are classified into universal regions and dedicated regions. A universal region provides universal cloud services for common tenants. A dedicated region provides specific services for specific tenants.

- An AZ contains one or more physical data centers. Each AZ has independent cooling, fire extinguishing, moisture-proof, and electricity facilities. Within an AZ, computing, network, storage, and other resources are logically divided into multiple clusters. AZs within a region are interconnected using high-speed optical fibers to support cross-AZ high-availability systems.

Figure 1-1 shows the relationship between regions and AZs.

Figure 1-1 Regions and AZs

Selecting a Region

Selecting an AZ

When deploying resources, consider your applications' requirements on disaster recovery (DR) and network latency.

- For high DR capability, deploy resources in different AZs within the same region.

- For low network latency, deploy resources in the same AZ.

1.2 Why Can't I View the Subnet and Security Group Information When Creating an Instance?

ROMA Connect has strict permissions for user resources.

You may not be able to view the VPC, subnet, security group, or EIP due to insufficient **Server Administrator** and **VPC Administrator** permissions.

For details about how to add the permissions, see **Modifying User Group Permissions**.

1.3 Can I Modify the VPC and Subnet After an Instance Is Created?

No.

Different ROMA Connect instances can use the same or different VPCs and subnets based on site requirements.

To change the VPC and subnet of an instance, delete the instance and create a new one.

1.4 Does ROMA Connect Support Access from Another VPC/Subnet?

Yes, ROMA Connect supports access across VPCs and subnets.

- In one VPC, cross-subnet access is allowed.

- In the case of two VPCs, connect them using a peering connection. For details about how to create and use a VPC peering connection, see **VPC Peering Connection**.

1.5 Why Do Different Users View Different Data of the Same Instance?

ROMA Connect has strict permissions for user resources.

By default, you can view and manage only the integration applications and resources created by yourself.

With integration application authorization, you can share applications and resources with other users under the same account.

2 Data Integration

2.1 Common Data Integration Tasks

2.1.1 Which Types of Data Are Supported by FDI Databases?

FDI databases support the following types of data:

- Kafka: string
- MySQL: integer, bigint, number, varchar, char, float, date, timestamp, datetime
- Oracle: integer, bigint, number, varchar, char, float, date, timestamp
- SQL Server: integer, bigint, varchar, char, float, date, datetime
- PostgreSQL: integer, number, varchar, char, float, date, and timestamp
- MongoDB: integer, double, string, boolean, date, long, object, and array
- DIS: integer, JSON, arraystring, boolean, long, double, bigdecimal, and date
- MRS Hive: integer, bigint, string, float, timestamp, varchar, char, and date
- MRS HBase: string

📖 NOTE

Do not use varchar, char, and date for the MRS Hive database, because data will be truncated once the maximum data source length is reached.

2.1.2 What Is a Trace Number? Can Data Be Traced by Using a Trace Number?

A trace number identifies a task scheduling. It can be used by a backend to trace task status, but not tracing data.

2.1.3 Can I Clear the Destination Tables of an FDI Task?

Destination tables can be cleared only if the DWS, MySQL, Oracle, PostgreSQL/openGauss, GaussDB(for MySQL), and SQL Server data sources are selected as the destination when you create an FDI task.

2.1.4 Can a Task Collect Data from One Table to Another Table?

A common task can collect data only from a single table at the source to a single table at the destination. A composite task can collect data from multiple tables at the source to multiple tables at the destination, at a time.

2.1.5 Is the FDI Task Created by One User Visible to Another User Under the Same Account?

FDI tasks are isolated by user. If permissions are not granted to a user, the user cannot view the FDI tasks created by another user under the same account. If a user wants to view the FDI tasks of other users, the user must be granted permissions through integration application authorization.

2.1.6 How Do I Configure FDI to Connect to MRS Hive of Other Tenants Through a Public Network?

Bind all MRS IP addresses to EIPs, and replace all private IP addresses in the MRS configuration file with EIPs. Then, add the **dfs.client.use.datanode.hostname** configuration item to the **hdfs-site.xml** file and set its value to **true**.

2.1.7 Will Collected Data Be Updated After a File Is Parsed from OBS to the RDS Database?

If the same file is collected and the destination database has a primary key, FDI updates data based on the primary key each time the file is updated on OBS.

If multiple files are collected and the destination database has same primary keys, FDI updates data with the value in the earliest file each time. By default, FDI obtains files from OBS based on the timestamp in descending order. The file uploaded at the latest time is parsed first, and the file uploaded at the earliest time is parsed last. As a result, after the task execution is complete, the value in the database is still the value in the earliest uploaded file.

2.1.8 Why Are the Column Values Incorrect When the CSV File on the Source OBS Is Parsed to the Relational Database?

This occurs during task configuration if the sequence of filling in metadata of the source OBS is different from that of the columns in the CSV file. Therefore, modify the task configuration and fill in the metadata based on the sequence of the columns in the CSV file.

2.1.9 Can Data Be Integrated into the Destination When the Data Types of Source and Destination Fields of MRS Hive Do Not Match?

If the field data type of MRS Hive at the destination does not match that of the source, for example, the data type of a source field is **datetime** and the data type of a destination field is **date**, data can be integrated to the destination, but the

value of the field whose data type does not match will be **NULL**. Other fields are properly integrated.

2.1.10 Is the Mapping Between MRS Hive or MRS HBase and MongoDB Case-sensitive When It Is Manually Entered?

The mapping is case-sensitive. The capitalization rule of source and destination fields must be the same as that of the selected fields. If they are different, the MongoDB database will not be impacted, but MRS Hive and MRS HBase will run abnormally.

2.1.11 Does MRS Hive Support Partitioning?

MRS Hive supports partitioning.

When entering mapping information, you can comply with the following format:

```
{
"partion_col":{
"origin_col": "Partition field name of the source table",
"target_col": "Partition field name of the destination"
},
"col_seq":{
"index_0": "Source table field 1",
"index_1": "Source table field 2",
"index_2": "Source table field 3"
}
}
```

Partitioning is optional for MRS Hive.
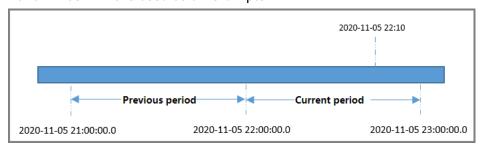
If there is no partition, comply with the following format when entering mapping information:

```
{
"col_seq":{
"index_0": "Source table field 1",
"index_1": "Source table field 2",
"index_2": "Source table field 3"
}
}
```

2.1.12 How Do I Set the Custom Period for the API Data Source at the Source?

When you create a common FDI task, if you set the source as an API and enable incremental migration for the task, scheduling period can be customized. The following lists several example configurations.

- If **Time Interval** is set to **Hour**, the configuration scenarios are as follows:

  2020-11-05 22:10 is used as an example.

a. **Period** is set to **Current period** and **Right Periodic Boundary** is set to **Open interval**.

The values of **Start Time Field** and **End Time Field** are the start time and end time of the current period, respectively.

The time interval for obtaining data is [2020-11-05 22:00:00.0, 2020-11-05 23:00:00.0].

b. **Period** is set to **Current period** and **Right Periodic Boundary** is set to **Closed interval**.

The values of **Start Time Field** and **End Time Field** are the start time and the end time subtracted by 1 ms of the current period, respectively.

The time interval for obtaining data is [2020-11-05 22:00:00.0, 2020-11-05 22:59:59.999].

c. **Period** is set to **Previous period** and **Right Periodic Boundary** is set to **Open interval**.

The values of **Start Time Field** and **End Time Field** are the start time and end time of the previous period, respectively.

The time interval for obtaining data is [2020-11-05 21:00:00.0, 2020-11-05 22:00:00.0].

d. **Period** is set to **Previous period** and **Right Periodic Boundary** is set to **Closed interval**.

The values of **Start Time Field** and **End Time Field** are the start time and the end time subtracted by 1 ms of the previous period, respectively.

The time interval for obtaining data is [2020-11-05 21:00:00.0, 2020-11-05 21:59:59.999].

- If **Time Interval** is set to **Day**, the following configuration scenarios are available:

2020-11-06 22:10 is used as an example.

a. **Period** is set to **Current period** and **Right Periodic Boundary** is set to **Open interval**.

The values of **Start Time Field** and **End Time Field** are the start time and end time of the current period, respectively.

The time interval for obtaining data is [2020-11-06 00:00:00.0, 2020-11-07 00:00:00.0].

b. **Period** is set to **Current period** and **Right Periodic Boundary** is set to **Closed interval**.

The values of **Start Time Field** and **End Time Field** are the start time and the end time subtracted by 1 ms of the current period, respectively.

The time interval for obtaining data is [2020-11-06 00:00:00.0, 2020-11-06 23:59:59.999].

c. **Period** is set to **Previous period** and **Right Periodic Boundary** is set to **Open interval**.

The values of **Start Time Field** and **End Time Field** are the start time and end time of the previous period, respectively.

The time interval for obtaining data is [2020-11-05 00:00:00.0, 2020-11-06 00:00:00.0].

d. **Period** is set to **Previous period** and **Right Periodic Boundary** is set to **Closed interval**.

The values of **Start Time Field** and **End Time Field** are the start time and the end time subtracted by 1 ms of the previous period, respectively.

The time interval for obtaining data is [2020-11-05 00:00:00.0, 2020-11-05 23:59:59.999].

- If **Time Interval** is set to **Week**, the following configuration scenarios are available:

  2020-11-06 22:10 is used as an example.

  a. **Period** is set to **Current period** and **Right Periodic Boundary** is set to **Open interval**.

  The values of **Start Time Field** and **End Time Field** are the start time and end time of the current period, respectively.

  The time interval for obtaining data is [2020-11-02 00:00:00.0, 2020-11-09 00:00:00.0].

  b. **Period** is set to **Current period** and **Right Periodic Boundary** is set to **Closed interval**.

  The values of **Start Time Field** and **End Time Field** are the start time and the end time subtracted by 1 ms of the current period, respectively.

  The time interval for obtaining data is [2020-11-02 00:00:00.0, 2020-11-08 23:59:59.999].

  c. **Period** is set to **Previous period** and **Right Periodic Boundary** is set to **Open interval**.

  The values of **Start Time Field** and **End Time Field** are the start time and end time of the previous period, respectively.
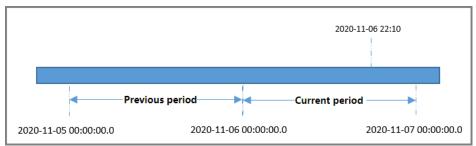
  The time interval for obtaining data is [2020-10-26 00:00:00.0, 2020-11-02 00:00:00.0].
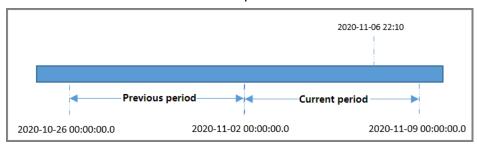
  d. **Period** is set to **Previous period** and **Right Periodic Boundary** is set to **Closed interval**.

  The values of **Start Time Field** and **End Time Field** are the start time and the end time subtracted by 1 ms of the previous period, respectively.

  The time interval for obtaining data is [2020-10-26 00:00:00.0, 2020-11-01 23:59:59.999].

- If **Time Interval** is set to **Month**, the following configuration scenarios are available:

  2020-11-06 22:10 is used as an example.

  a. **Period** is set to **Current period** and **Right Periodic Boundary** is set to **Open interval**.

     The values of **Start Time Field** and **End Time Field** are the start time and end time of the current period, respectively.

     The time interval for obtaining data is [2020-11-01 00:00:00.0, 2020-12-01 00:00:00.0].

  b. **Period** is set to **Current period** and **Right Periodic Boundary** is set to **Closed interval**.

     The values of **Start Time Field** and **End Time Field** are the start time and the end time subtracted by 1 ms of the current period, respectively.

     The time interval for obtaining data is [2020-11-01 00:00:00.0, 2020-11-30 23:59:59.999].

  c. **Period** is set to **Previous period** and **Right Periodic Boundary** is set to **Open interval**.

     The values of **Start Time Field** and **End Time Field** are the start time and end time of the previous period, respectively.
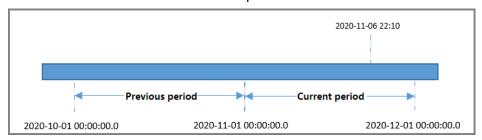
     The time interval for obtaining data is [2020-10-01 00:00:00.0, 2020-11-01 00:00:00.0].

  d. **Period** is set to **Previous period** and **Right Periodic Boundary** is set to **Closed interval**.

     The values of **Start Time Field** and **End Time Field** are the start time and the end time subtracted by 1 ms of the previous period, respectively.

     The time interval for obtaining data is [2020-10-01 00:00:00.0, 2020-10-31 23:59:59.999].

2.1.13 Does SAP Support Reading Views by Page?

Yes, but data may be duplicate or lost. When SAP reads a view, the **/SAPDS/ RFC_READ_TABLE2** function is called, which may cause problems in pagination.

2.2 Composite Data Integration Tasks

2.2.1 How Do Source Database Restarts Affect Composite Tasks?

During service usage, the operations such as capacity expansion and restart may be required for databases such as Oracle. The impacts of these operations on composite tasks are as follows:

- If the database takes a short time to restart, the system retries a database connection and attempts to restore a task.

- If the database takes a long time to restart during a composite task, an error will be reported. After the database is restarted, you must restart the composite task.

2.2.2 How Do I Handle an Exception of the CDC Function (Using the XStream Mode) Caused by Unexpected Oracle Database Restart?

If the CDC function is abnormal because the Oracle database restarts unexpectedly, perform the following operations on the Oracle database to locate and rectify the fault:

Connect to the Oracle database as user **sys** and run the following command:

```
SELECT CAPTURE_NAME,status FROM ALL_CAPTURE;
```

If the status is **Aborted**, the outbound server of the CDC is abnormal. Run the following command and press **Enter**. Then, enter **/** and press **Enter** again to manually start the outbound server.

```
BEGIN
 DBMS_XSTREAM_ADM.START_OUTBOUND(server_name => 'dbzxout');
END;
```

3 Service Integration

3.1 Does APIC Support Multiple Backend Endpoints?

Yes, APIC supports the configuration of multiple backend endpoints through a load balance channel. You can add multiple Elastic Cloud Servers (ECSs) to a load balance channel.

3.2 How Do I Perform App Authentication in iOS System?

APIC provides SDKs and demos in multiple languages, such as Java, Python, C, PHP, and Go, for App authentication. If you want to use iOS or languages not supported, see **App Authentication Principles**.

3.3 How Can I Ensure the Security of Backend Services Invoked by APIC?

You can ensure the security of backend services invoked by APIC by using the following methods:

- Bind signature keys to APIs. After a signature key is bound to an API, APIC adds signature information to each request sent to the backend service. The backend service calculates the signature information in each request and checks whether the signature information is consistent with that on APIC.

- Encrypt requests using HTTPS. Before using this method, ensure that the required SSL certificates are available.

3.4 Do I Need to Publish an API Again After Modification?

Yes. After you modify the parameters of a published API, you must publish the API again to synchronize the modifications to the environment.

3.5 What Can I Do If an API Published in a Non-RELEASE Environment Is Inaccessible?

Add the x-stage request header with its value set to the environment name and then access the API again.

3.6 Can I Invoke Different Backend Services by Publishing an API in Different Environments?

Yes, you can invoke different backend services by publishing an API in different environments while specifying environment variables and backend parameters.

3.7 Can Applications Deployed in a VPC Call APIs?

Yes, applications deployed in a VPC can call APIs by default. If the domain name resolution fails, configure the DNS server on the current endpoint. For details, see **How Do I Switch to a Private DNS Server**. After the configuration, applications deployed in the VPC can call APIs.

3.8 What Is the Maximum Size of an API Request Packet?

The size of the API request body can be configured on the instance configuration parameter page (1 to 9536 MB), with the following restrictions:

- When App authentication is used, only API requests with a body less than or equal to 12 MB can be authenticated.
- The maximum size of the request body for a hosting API is 2 GB.
- The request body of function APIs cannot be calculated accurately due to the memory size limit of the internal sandbox.

3.9 Can I Modify a Deployed Custom Backend?

Configurations can be modified except for the integration application, backend request method, and request path. The modification takes effect only after the backend is deployed again.

3.10 How Does APIC Throttle Requests?

You can configure request throttling policies to control the number of APIs that can be accessed in a specified period to protect backend services. For details, see **Creating a Request Throttling Policy**.

By default, a single API can be accessed for a maximum of 6000 times per second. If you do not create a request throttling policy, APIC executes the default request

throttling policy. On the **Instance Information** page of the ROMA Connect console, you can modify the **ratelimit_api_limits** parameter on the **Configuration Parameters** tab page to set the default request throttling policy for APIs.

3.11 What Are the Causes of an API Calling Failure?

API calling failures are classified into three types: failure of API calling in the same VPC, failure of API calling between different VPCs, and failure of API calling through a public network.

- If the API fails to be called in the VPC, check whether the domain name is the same as that automatically allocated by the API. If the domain name is different from that automatically allocated by the API, the API fails to be called.

- If APIs fail to be called between VPCs, check whether the two VPCs are connected. If the two VPCs are not connected, you can establish VPC peering connections to enable communication between the two VPCs.

  For details on how to create and use VPC peering connections, see **VPC Peering Connection**.

- If APIs fail to be called through a public network, the possible causes are as follows:

  – The API is not bound to an EIP. As a result, the API lacks a valid address for public network access, and therefore the API fails to be called through a public network. Bind the EIP and call the API again. For details, see .

  – The inbound rule is incorrectly configured. As a result, the API fails to be called through a public network. For details about inbound rules, see **Preparing Required Resources**.

  – The request header "host: Group domain name" is not added during API calling. As a result, the API fails to be called through a public network. The API can be called as long as the message header is added.

4 Message Integration

4.1 Is There a Maximum Length of Messages Produced to a Topic?

Yes, 10 MB.

4.2 What Kafka Server Versions Does MQS Support?

MQS supports Kafka 1.1.0 and 2.7.

5 Device Integration

5.1 What Communication Protocols Are Supported by LINK?

LINK allows device access using MQTT, Modbus, or OPC UA.

5.2 Why Can't My Device Be Connected to LINK?

The possible causes are as follows:

- Device information not configured

  On the device, check whether the connection address, connection port, client ID, username, and password are correct.

- Device disabled

  Check the device status on ROMA Connect. A disabled device cannot be connected to LINK. To connect the device to LINK, change the device status from **Disabled** to **Enabled**.

- Instance public access not enabled

  Enable public access on ROMA Connect. For details, see **Viewing Details of an Instance**.

- No EIP bound to the instance

  On ROMA Connect, check whether an EIP has been bound to the instance. Instances not bound with an EIP cannot be accessed through public networks.

- No port configured for the security group inbound rule

  On VPC, check whether a port is configured for the security group inbound rule. If no, devices cannot be accessed.

5.3 Why Does the Rule Engine Data Destination Fail to Receive Messages Sent by a Device?

The possible causes are as follows:

- An incorrect inbound rule is added to your security group.

  Click **Instance Information** and locate the target security group. Click the security group name to access its details page. Then, add an inbound rule based on the parameters listed in **Table 5-1**.

  Table 5-1 Security group parameters

Parameter	Port
Port for accessing MQS through an intranet	9092
Port for accessing MQS through an intranet in SSL mode	9093
Port for accessing MQS through a public network	9094, 9095, and 9096
Port for accessing MQS through a public network in SSL mode	9095, 9096, and 9097
Port for accessing MQS through RESTful APIs	9292

- The topic is incorrectly configured.

  You need to use a topic that has permission to publish messages. Check whether your topic has the publish permission. You can find the name of the topic where the fault occurs in the **Topic Name** column, and then check whether the topic has the publish permission in the **Topic Type** column. If a topic does not have the publish permission, the device cannot send messages to the destination of the rule engine.

  To solve this problem, you can configure a topic of the Publish type on the device so that the device can send messages to the destination.

- The first default rule of the security group is deleted.

  When a security group is created, the system automatically generates an inbound rule that allows the destination of the rule engine to receive messages from all devices. If you delete this rule without creating other inbound rules, the entire security group will lack inbound rules. As a result, the destination of the rule engine cannot receive messages from the device.

  You can add inbound rules to solve this problem. If the rule engine is required to receive information from a specific device, you can add an inbound rule and set the port range to the port of the specific device. If the rule engine is required to receive information from all devices, you can add an inbound rule and set the port range to **All**.

- The topic is deleted.

  The rule engine needs to forward messages to a specified topic. If the topic is deleted unexpectedly, messages cannot be forwarded to the topic.

  To solve this problem, create a topic and configure the topic in the rule engine. When you create the topic, you need to specify the publish or subscribe permission for the topic.