Informatica

# Informatica® PowerExchange for Amazon Redshift

10.5.6

# User Guide

Informatica PowerExchange for Amazon Redshift User Guide
10.5.6
May 2024



Publication Date: 2024-05-24

# Table of Contents

# Preface

Use the *Informatica® PowerExchange® for Amazon Redshift User Guide* to learn how to read from or write to Amazon Redshift by using the Developer tool. Learn to create a connection, develop and run mappings and dynamic mappings in the native environment and in the Hadoop and Databricks environments.

# Informatica Resources

Informatica provides you with a range of product resources through the Informatica Network and other online portals. Use the resources to get the most from your Informatica products and solutions and to learn from other Informatica users and subject matter experts.

## Informatica Network

The Informatica Network is the gateway to many resources, including the Informatica Knowledge Base and Informatica Global Customer Support. To enter the Informatica Network, visit https://network.informatica.com.

As an Informatica Network member, you have the following options:

- Search the Knowledge Base for product resources.
- View product availability information.
- Create and review your support cases.
- Find your local Informatica User Group Network and collaborate with your peers.

## Informatica Knowledge Base

Use the Informatica Knowledge Base to find product resources such as how-to articles, best practices, video tutorials, and answers to frequently asked questions.

To search the Knowledge Base, visit https://search.informatica.com. If you have questions, comments, or ideas about the Knowledge Base, contact the Informatica Knowledge Base team at KB_Feedback@informatica.com.

## Informatica Documentation

Use the Informatica Documentation Portal to explore an extensive library of documentation for current and recent product releases. To explore the Documentation Portal, visit https://docs.informatica.com.

If you have questions, comments, or ideas about the product documentation, contact the Informatica Documentation team at infa_documentation@informatica.com.

## Informatica Product Availability Matrices

Product Availability Matrices (PAMs) indicate the versions of the operating systems, databases, and types of data sources and targets that a product release supports. You can browse the Informatica PAMs at https://network.informatica.com/community/informatica-network/product-availability-matrices.

## Informatica Velocity

Informatica Velocity is a collection of tips and best practices developed by Informatica Professional Services and based on real-world experiences from hundreds of data management projects. Informatica Velocity represents the collective knowledge of Informatica consultants who work with organizations around the world to plan, develop, deploy, and maintain successful data management solutions.

You can find Informatica Velocity resources at http://velocity.informatica.com. If you have questions, comments, or ideas about Informatica Velocity, contact Informatica Professional Services at ips@informatica.com.

## Informatica Marketplace

The Informatica Marketplace is a forum where you can find solutions that extend and enhance your Informatica implementations. Leverage any of the hundreds of solutions from Informatica developers and partners on the Marketplace to improve your productivity and speed up time to implementation on your projects. You can find the Informatica Marketplace at https://marketplace.informatica.com.

## Informatica Global Customer Support

You can contact a Global Support Center by telephone or through the Informatica Network.

To find your local Informatica Global Customer Support telephone number, visit the Informatica website at the following link:
https://www.informatica.com/services-and-training/customer-success-services/contact-us.html.

To find online support resources on the Informatica Network, visit https://network.informatica.com and select the eSupport option.

CHAPTER 1

# Introduction to PowerExchange for Amazon Redshift

This chapter includes the following topics:



## PowerExchange for Amazon Redshift Overview

You can use PowerExchange for Amazon Redshift to read data from or write data to Amazon Redshift. You can also use PowerExchange for Amazon Redshift to read data from Amazon Redshift views.

Amazon Redshift views contain information about the functioning of the Amazon Redshift system. You can run a query on views like you run a query on database tables. You can select multiple schemas for Amazon Redshift objects.

You can use Amazon Redshift objects as sources and targets in mappings. When you use Amazon Redshift objects in mappings, you must configure properties specific to Amazon Redshift. You can validate and run mappings in native or non-native environments.

You can configure HTTPS proxy to connect to Amazon Redshift. You can also configure an SSL connection to connect to Amazon Redshift.

The Data Integration Service uses the Amazon driver to communicate with Amazon Redshift.

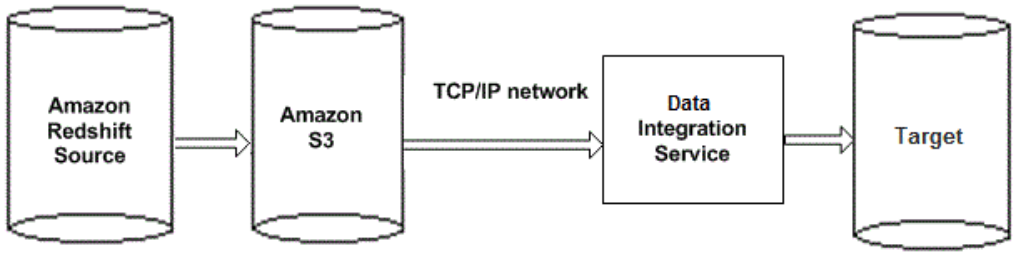
### Example

You work for an organization that stores purchase order details, such as customer ID, item codes, and item quantity in an on-premise MySQL database. You need to analyze purchase order details and move data from the on-premise MySQL database to an affordable cloud-based environment. Create a mapping to read all the purchase records from the MySQL database and write them to Amazon Redshift for data analysis.

# Data Integration Service and Amazon Redshift Integration

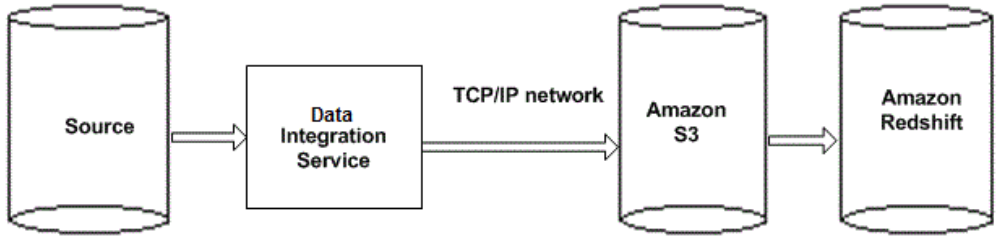
The Data Integration Service uses the Amazon Redshift connection to connect to Amazon Redshift.

The following image shows how Informatica connects to Amazon Redshift to read data:




When you run the Amazon Redshift mapping, the Data Integration Service reads data from Amazon Redshift based on the workflow and Amazon Redshift connection configuration. The Data Integration Service connects and reads data from Amazon Simple Storage Service (Amazon S3) through a TCP/IP network. The Data Integration Service then stores data in a staging directory on the Informatica machine. Amazon S3 is a storage service in which you can copy data from source and simultaneously move data to any target. The Data Integration Service issues a copy command that copies data from Amazon S3 to the target.

The following image shows how Informatica connects to Amazon Redshift to write data:




When you run the Amazon Redshift mapping, the Data Integration Service writes data to Amazon Redshift based on the workflow and Amazon Redshift connection configuration. The Data Integration Service stores data in a staging directory on the Informatica machine. The Data Integration Service then connects and writes data to Amazon Simple Storage Service (Amazon S3) through a TCP/IP network. Amazon S3 is a storage service in which you can copy data from source and simultaneously move data to Amazon Redshift clusters. The Data Integration Service issues a copy command that copies data from Amazon S3 to the Amazon Redshift target table.

# Introduction to Amazon Redshift

Amazon Redshift is a cloud-based petabyte-scale data warehouse service that organizations can use to analyze and store data.

Amazon Redshift uses columnar data storage, parallel processing, and data compression to store data and to achieve fast query execution. Amazon Redshift uses a cluster-based architecture that consists of a leader

node and compute nodes. The leader node manages the compute nodes and communicates with the external client programs. The leader node interacts with the client applications and communicates with compute nodes. A compute node stores data and runs queries for the leader node. Any client that uses a PostgreSQL driver can communicate with Amazon Redshift.

# Amazon Redshift Spectrum Overview

Amazon Redshift Spectrum enables you to run complex Amazon Redshift SQL queries on a large amount data of different formats stored in Amazon S3.

Amazon Redshift Spectrum resides on Amazon Redshift servers independent of the Amazon Redshift cluster. When you run queries using Amazon Redshift Spectrum, the queries run faster and uses less Amazon Redshift cluster processing capacity as Amazon Redshift Spectrum pushes all the compute-intensive tasks to the Amazon Redshift Spectrum layer.

## External Schema and External Table

To use Amazon Redshift Spectrum, you can read data from an external table within an external schema that references a database in an external data catalog.

The metadata of the external database and external table are stored in the external data catalog. You must provide Amazon Redshift authorization to access the data catalog and the data files in Amazon S3.

You can read data from a single external table, multiple external tables, or from a standard Amazon Redshift table that is joined to an external table.

Multiple Amazon Redshift clusters can contain multiple external tables. You can run a query for the same data on Amazon S3 from any Amazon Redshift cluster in the same region. When you update the data in Amazon S3, the data is immediately available in all the Amazon Redshift clusters.

When you add an external table as a source in a mapping, the external table name is displayed in the `spectrum_schemaname` format in the **Select Source Object** dialog box.

You can use the following data types when you read from an external table:

- Bigint (INT8)
- Boolean (BOOL)
- Char (CHARACTER)
- Date
- Decimal (NUMERIC)
- Double Precision (FLOAT8)
- Integer (INT, INT4)
- Real (FLOAT4)
- Smallint (INT2)
- Timestamp
- Varchar (CHARACTER VARYING)

**Rules and guidelines for external tables**

Consider the following rules and guidelines for external tables:

- You can only read data from an Amazon Redshift Spectrum external table.
- The Secure Agent does not remove the external table names from the list of target objects available in the Target transformation.

CHAPTER 2

# PowerExchange for Amazon Redshift Configuration

This chapter includes the following topics:



## PowerExchange for Amazon Redshift Configuration Overview

PowerExchange for Amazon Redshift is installed with Informatica Services. You enable PowerExchange for Amazon Redshift with a license key.

## Prerequisites

Before you can use PowerExchange for Amazon Redshift, perform the following tasks:

1. Verify that the domain has a Data Integration Service and a Model Repository Service.
2. Verify that you can connect to Amazon Redshift with an SQL client that uses the PostgreSQL driver. For example, you can use SQL Workbench/J to connect to Amazon Redshift.
3. To run mappings on Hortonworks, Amazon EMR, and MapR distributions that use non-Kerberos authentication, configure user impersonation.
   For information about configuring user impersonation, see the *Data Engineering Integration Guide*.
4. To run mappings on MapR secure clusters, configure the MapR secure clusters on all the nodes.
   For information about configuring MapR secure clusters, see the *IData Engineering Integration Guide*.

5. To run mappings on Spark engine and Blaze engine using a secure domain, you must import the CA certificates available in the Redshift Certificate Authority bundle and the `Baltimore CyberTrust Root` certificate file.
   To import the CA certificates available in the Redshift Certificate Authority bundle, perform the following steps:
   - Download the `redshift-keytool.jar` file.
   - Run the following command to import the CA certificates available in the Redshift Certificate Authority bundle into the Informatica TrustStore location:

     ```
     java -jar redshift-keytool.jar -k <infa_trust_store_location> -p
     <keystore_password>
     ```

   - Restart the Data Integration Service.

   To import the `Baltimore CyberTrust Root` certificate file, perform the following steps:
   - Download the `Baltimore CyberTrust Root` certificate file.
   - Provide the read, write, and execute permissions to the certificate file.
   - Run the following command to import the certificate file into the Informatica TrustStore location:

     ```
     <INFA_HOME>/java/jre/bin/keytool -keystore <infa_trust_store_location> -
     importcert -alias <Alias_Name> -file <BaltimoreCyberTrustRoot certificate file
     path>/<certificate_filename> -storepass <Truststore_Password>
     ```

   - Restart the Data Integration Service.
6. To enable Metadata Access Service to import metadata from Amazon Redshift, see the following KB article:
   https://knowledge.informatica.com/s/article/HOW-TO-Enable-Metadata-Access-Service-to-import-metadata-from-Amazon-S3-and-Amazon-Redshift?language=en_US

### Configure Databricks Connection Advanced Properties

Verify that a Databricks connection is created in the domain. If you want to read NULL values from or write NULL values to an Amazon Redshift table, configure the following advanced properties in the Databricks connection:

- `infaspark.flatfile.reader.nullValue=True`
- `infaspark.flatfile.writer.nullValue=True`

For more information about product requirements and supported platforms, see the Product Availability Matrix.

# IAM Authentication

Optional. You can configure Amazon Identity and Access Management (IAM) authentication when the Data Integration Service runs on an Amazon Elastic Compute Cloud (EC2) system. Use IAM authentication for secure and controlled access to Amazon Redshift resources when you run a session.

Use IAM authentication when you want to run a mapping on an EC2 system.

Perform the following steps to configure IAM authentication:

1. Create a minimal Amazon IAM Policy. For more information, see "Create a Minimal Amazon IAM Policy" on page 14.

2. Create the Amazon EC2 role. Associate the minimal Amazon IAM policy while creating the EC2 role. The Amazon EC2 role is used when you create an EC2 system in the Redshift cluster. For more information about creating the Amazon EC2 role, see the AWS documentation.
3. Create an EC2 instance. Assign the Amazon EC2 role that you created in step #2 to the EC2 instance.
4. Create the Amazon Redshift Role ARN for secure access to Amazon Redshift resources. Associate the minimal Amazon IAM policy while creating the Amazon Redshift role. You can use the Amazon Redshift Role ARN in the UNLOAD and COPY commands. For more information about creating the Amazon Redshift Role ARN, see the AWS documentation.
5. Add the Amazon Redshift Role ARN to the Amazon Redshift cluster to successfully perform the read and write operations. For more information about adding the Amazon Redshift Role ARN to the Amazon Redshift cluster, see the AWS documentation.
6. Install the Data Integration Service on the EC2 system.

You can use AWS IAM authentication when you run a mapping in the EMR cluster. To use Amazon IAM authentication in the EMR cluster, you must create the Amazon EMR Role. Create a new Amazon EMR Role or use the default Amazon EMR Role. You must assign both the Amazon EMR Role and Amazon Redshift Role to the EMR cluster for secure access to Amazon Redshift resources.

**Note:** Before you configure IAM Role with EMR cluster, you must install the Informatica Services on an EC2 instance with the IAM Roles assigned.

# Create a Minimal Amazon IAM Policy

Create an Amazon IAM policy and define the required permissions to stage the data in Amazon S3 when you want to read data from and write data to Amazon Redshift.

Use the following minimum required permissions to stage the data in Amazon S3:

- PutObject
- GetObject
- DeleteObject
- ListBucket

You can use the following sample Amazon IAM policy:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:DeleteObject",
                "s3:ListBucket",
                      ],
            "Resource": [
                "arn:aws:s3:::<bucket_name>/*",
                "arn:aws:s3:::<bucket_name>"
            ]
        }
    ]
}
```

## Amazon Redshift Supported Regions

The Amazon S3 bucket and Amazon Redshift cluster must reside in the same region to run a session successfully.

The supported regions are:

- Asia Pacific (Mumbai)
- Asia Pacific (Seoul)
- Asia Pacific (Singapore)
- Asia Pacific (Sydney)
- Asia Pacific (Tokyo)
- AWS GovCloud (US)
- Canada (Central)
- China (Beijing)
- China (Ningxia)
- EU (Frankfurt)
- EU (Ireland)
- EU (London)
- EU (Paris)
- South America (Sao Paulo)
- US East (N. Virginia)
- US East (Ohio)
- US West (N. California)
- US West (Oregon)

# Configure Databricks Cluster

Set the access key ID and secret access key values under Spark Config in your Databricks cluster configuration to access Amazon S3 storage. You must specify one key value pair per line and each key value pair must be separated by a single space.

```
spark.hadoop.fs.s3a.awsAccessKeyId xxyyzz
spark.hadoop.fs.s3a.awsSecretAccessKey xxxyyyzzz
```

### Access using IAM role

Optional. Create an IAM role associated with the AWS account of the Databricks deployment. Amazon S3 bucket must belong to the same account associated with the Databricks deployment. If the bucket belongs to a different AWS account, then, the Cross-Account bucket policy must be enabled to access the bucket.

### Server-side S3 encryption (AES-256)

Optional. Set the `server-side-encryption-algorithm` property under Spark Config in your Databricks cluster configuration:

```
spark.hadoop.fs.s3a.server-side-encryption-algorithm AES256
```

### Server-side encryption using SSE-KMS

Optional. Set the following properties under Spark Config in your Databricks cluster configuration:

```
spark.hadoop.fs.s3a.server-side-encryption-kms-master-key-id arn:aws:kms:us-west-XX:key/
XXXYYYYYYY
spark.hadoop.fs.s3a.server-side-encryption-algorithm aws:kms
spark.hadoop.fs.s3a.impl com.databricks.s3a.S3AFileSystem
```

# Amazon Redshift Spectrum Prerequisite Tasks

To read data from an Amazon Redshift Spectrum external table, you must provide the required authorization to the Amazon Redshift cluster to access the data catalog and the data files in Amazon S3.

1. Create an AWS Identity and Access Management (IAM) role to authorize the Amazon Redshift cluster access to the external data catalog and data files in Amazon S3.
2. Associate the IAM Role with the specified Amazon Redshift cluster.
3. Create an external schema.
4. Provide Amazon Redshift Role ARN for the IAM Role in the external schema.
5. Create an external table within the external schema and specify the Amazon S3 location from where you want to read the data. For more information about creating external tables, see the AWS documentation.

**Note:** The Amazon Redshift cluster and the Amazon S3 bucket that contains the data files must belong to the same region. The Amazon Redshift cluster must be of version 1.0.1294 or later.

# Configuring Proxy Settings

You can configure proxy to connect to Amazon Redshift in the native environment, on the Spark engine, or on the Databricks Spark engine. For more information on how to configure proxy, see the following Knowledge base articles:

- For the native environment, see KB 562908.
- For Spark engine, see KB 000186916.
- For Databricks Spark engine, see KB 000186919.

CHAPTER 3

# Amazon Redshift Connections

This chapter includes the following topics:



## Amazon Redshift Connection Overview

Amazon Redshift connection enables you to read data from or write data to Amazon Redshift.

You can use Amazon Redshift connections to create data objects and run mappings. The Developer tool uses the connection when you create a data object. The Data Integration Service uses the connection when you run mappings.

You can use AWS Identity and Access Management (IAM) authentication to securely control access to Amazon S3 resources. If you have valid AWS credentials and you want to use IAM authentication, you do not have to specify the access key and secret key when you create an Amazon Redshift connection.

You can create an Amazon Redshift connection from the Developer tool or the Administrator tool. The Developer tool stores connections in the domain configuration repository. Create and manage connections in the connection preferences.

When you run a mapping that reads data from an Amazon Redshift source and writes data to an Amazon Redshift target on the Spark engine, the mapping fails if the AWS credentials such as Access Key or Secret Key are different for source and target.

# Amazon Redshift Connection Properties

When you set up an Amazon Redshift connection, you must configure the connection properties.

The following table describes the Amazon Redshift connection properties:

| Property | Description |
|---|---|
| Name | The name of the connection. The name is not case sensitive and must be unique within the domain. You can change this property after you create the connection. The name cannot exceed 128 characters, contain spaces, or contain the following special characters:~ ` ! $ % ^ & * ( ) - + = { [ } ] \| \ : ; " ' < , > . ? / |
| ID | String that the Data Integration Service uses to identify the connection. The ID is not case sensitive. It must be 255 characters or less and must be unique in the domain. You cannot change this property after you create the connection. Default value is the connection name. |
| Description | The description of the connection. The description cannot exceed 4,000 characters. |
| Location | The domain where you want to create the connection. |
| Type | The connection type. Select Amazon Redshift in the Database. |

The **Details** tab contains the connection attributes of the Amazon Redshift connection. The following table describes the connection attributes:

| Property | Description |
|---|---|
| Authentication Type | The authentication method to log in to Amazon Redshift. You can select one of the following authentication methods:<br>- Default. Uses the username and password to connect to Amazon Redshift.<br>- Redshift IAM Authentication via AssumeRole. Uses AssumeRole for IAM authentication to connect to Amazon Redshift. |
| Username | User name of the Amazon Redshift account. |
| Password | Password for the Amazon Redshift account. |
| Access Key ID | Amazon S3 bucket access key ID to stage data in the S3 bucket using unload and copy commands. |
| Secret Access Key | Amazon S3 bucket secret access key ID to stage the data in the S3 bucket. |
| Master Symmetric Key | Optional. Provide a 256-bit AES encryption key in the Base64 format when you enable client-side encryption. You can generate a key using a third-party tool.<br>If you specify a value, ensure that you specify the encryption type as client-side encryption in the advanced target properties. |
| JDBC URL | Amazon Redshift connection URL. |

| Property | Description |
|---|---|
| Cluster Region | Optional. The AWS cluster region in which the bucket you want to access resides.<br>Select a cluster region if you choose to provide a custom JDBC URL that does not contain a cluster region name in the **JDBC URL** connection property.<br>If you specify a cluster region in both **Cluster Region** and **JDBC URL** connection properties, the Data Integration Service ignores the cluster region that you specify in the **JDBC URL** connection property.<br>To use the cluster region name that you specify in the **JDBC URL** connection property, select **None** as the cluster region in this property.<br>Select one of the following cluster regions:<br>- Asia Pacific (Mumbai)<br>- Asia Pacific (Seoul)<br>- Asia Pacific (Singapore)<br>- Asia Pacific (Sydney)<br>- Asia Pacific (Tokyo)<br>- AWS GovCloud (US)<br>- Canada (Central)<br>- China (Beijing)<br>- China (Ningxia)<br>- EU (Ireland)<br>- EU (Frankfurt)<br>- EU (London)<br>- EU (Paris)<br>- South America (Sao Paulo)<br>- US East (Ohio)<br>- US East (N. Virginia)<br>- US West (N. California)<br>- US West (Oregon)<br>Default is None.<br>You can only read data from or write data to the cluster regions supported by AWS SDK used by PowerExchange for Amazon Redshift. |
| Customer Master Key ID | Optional. Specify the customer master key ID generated by AWS Key Management Service (AWS KMS) or the Amazon Resource Name (ARN) of your custom key for cross-account access. You must generate the customer master key corresponding to the region where Amazon S3 bucket resides. You can specify any of the following values:<br>**Customer generated customer master key**<br>Enables client-side or server-side encryption.<br>**Default customer master key**<br>Enables client-side or server-side encryption. Only the administrator user of the account can use the default customer master key ID to enable client-side encryption. |
| IAM Assume Role ARN | The Amazon Resource Number (ARN) of the IAM role assumed by the user to use the dynamically generated temporary security credentials.<br>Set the value of this property if you want to use the temporary security credentials to access Amazon Redshift.<br>For more information about how to get the ARN of the IAM role, see the AWS documentation. |
| Cluster Identifier | The unique identifier of the cluster that hosts Amazon Redshift for which you are requesting the security credentials.<br>Specify the cluster name. |
| Database Name | Name of the Amazon Redshift database. |

| Property | Description |
|---|---|
| Auto Create DBUser | Select to create a new Amazon Redshift database user at run time.<br>Default is disabled.<br>**Note:** Use this option to create a new Amazon Redshift database user. If you create a new user directly using the AWS command line, the user is not created in Amazon Redshift. |
| Database Group | Name of the database group.<br>Specify the database user and group details after you log in to the Amazon Redshift database.<br>**Note:** If you do not specify a group, the user is added to a public group. |
| Expiration Time | The time duration that the password for the Amazon Redshift database user expires.<br>Specify a value between 900 seconds and 3600 seconds.<br>Default is 900. |
| Use EC2 Role to Assume Role | Select to enable the EC2 role to assume another IAM role specified in the IAM AssumeRole ARN option. |
| IAM User Access Key | The access key of the IAM user that has permissions to assume the IAM AssumeRole ARN. |
| IAM User Secret Access Key | The secret access key of the IAM user that has permissions to assume the IAM Assume Role ARN. |

# Creating an Amazon Redshift Connection

Create an Amazon Redshift connection before you create an Amazon Redshift data object.

1. In the Developer tool, click **Window** > **Preferences**.
2. Select **Informatica** > **Connections**.
3. Expand the domain in the **Available Connections**.
4. Select the connection type **Database** > **Amazon Redshift**, and click **Add**.
5. Enter a connection name and an optional description.
6. Select Amazon Redshift as the connection type.
7. Click **Next**.
8. Configure the connection properties.
9. Click **Test Connection** to verify the connection to Amazon Redshift.

   If a default Metadata Access Service is not set, a message appears to configure the Metadata Access Service. Click **OK** and set one Metadata Access Service as default. After you set a default Metadata Access Service, the connection to Amazon Redshift is tested. If the Metadata Access Service does not exist, contact the Informatica administrator to create a new Metadata Access Service in the domain.
10. Click **Finish**.

CHAPTER 4

# PowerExchange for Amazon Redshift Data Objects

This chapter includes the following topics:



## Amazon Redshift Data Object Overview

An Amazon Redshift data object is a physical data object that uses Amazon Redshift as a source or target. An Amazon Redshift data object represents the data in an Amazon Redshift data source.

You can configure the data object read and write operation properties that determine how data can be read from Amazon sources and loaded to Amazon Redshift targets. You first create a connection to create an Amazon Redshift data object. Create a data object operation for the Amazon Redshift data object. Then, you can add the data object read or write operation to a mapping.

**Note:** When you import an Amazon Redshift table in the Developer tool, ensure that the table names and the column names are in lowercase. When you import an Amazon Redshift table in the Developer tool, you can add nullable columns in the table. However, you must not add the nullable columns as primary keys in the table.

## Amazon Redshift Data Object Properties

Specify the data object properties when you create the data object.

The following table describes the properties that you configure for the Amazon Redshift data objects:

| Property | Description |
|---|---|
| Name | Name of the Amazon Redshift data object. |
| Location | The project or folder in the Model Repository where you want to store the Amazon Redshift data object. |
| Connection | Name of the Amazon Redshift connection. |

# Amazon Redshift Data Object Read Operation

Create a mapping with an Amazon Redshift data object read operation to read data from Amazon Redshift.

You can encrypt data, specify the location of the staging directory, and securely unload the results to Amazon Redshift.

## Amazon Redshift Staging Directory for Amazon Redshift Sources

The Data Integration Service creates a staging file in the directory that you specify in the source properties. The Data Integration Service reads the data from the Amazon Redshift source and writes the data to the staging directory before it writes data to Amazon S3.

The Data Integration Service deletes the staged files from the staging directory after it writes the data to Amazon S3. Specify a staging directory in the mapping properties with an appropriate amount of disk space for the volume of data that you want to process. Specify a directory on the machine that hosts the Data Integration Service.

The Data Integration Service creates subdirectories in the staging directory. Subdirectories use the following naming convention:

```
<staging directory>/infaRedShiftStaging<MMddHHmmssSSS+xyz>
```

## Data Encryption in Amazon Redshift Sources

To protect data, you can encrypt the data when you read the data from a source.

Select the type of the encryption in the **Encryption Type** field in data object read operation properties. The Unload command creates staging files on Amazon S3 for server-side encryption with the AWS-managed encryption keys and AWS Key Management Service key.

Use the customer master key ID generated by AWS Key Management Service in the Unload command for server-side encryption.

You can select the following types of encryption:

**None**

The data is not encrypted.

**SSE-S3**

If you select the **SSE-S3** encryption type, the Unload command creates the staging files in the Amazon S3 bucket and Amazon S3 encrypts the file using AWS-managed encryption keys for server-side encryption.

**SSE-KMS**

If you select **Server Side Encryption With KMS** as the encryption type, the Unload command creates the staging files in the Amazon S3 bucket and Amazon S3 encrypts the file using AWS KMS-managed customer master key or Amazon Resource Name (ARN) for server-side encryption.

The AWS KMS-managed customer master key that you specify in the connection property must belong to the same region where Amazon S3 is hosted.

For example, if Amazon S3 is hosted in the **US West (Oregon)** region, you must use the AWS KMS-managed customer master key enabled in the same region when you select the **SSE-KMS** encryption type.

**CSE-SMK**

If you select the **CSE-SMK** encryption type, Amazon Redshift uploads the data to the Amazon S3 server by using the master symmetric key and then loads the data by using the copy command with the encrypted option and a private encryption key for additional security.

You must provide a master symmetric key ID in the connection property to enable **CSE-SMK** encryption type.

**Note:** PowerExchange for Amazon Redshift does not support the server-side encryption with the master symmetric key and client-side encryption with the customer master key.

The following table lists the encryption type support for various environments:

| Encryption Type | Native Environment | Blaze Environment | Spark Environment | Databricks Environment |
|---|---|---|---|---|
| Server-side Encryption | Yes | Yes | Yes | Yes |
| Client-side Encryption | Yes | No | No | No |

## Read from Amazon Redshift without staging in Amazon S3

You can also read data from an Amazon Redshift source without staging the source data on the S3 bucket by enabling the following two flags in JVM command line options:

- `DenableDirectRead=true`
- `DfetchSize=100`

Perform the following steps to set the flags in the Data Integration Service:

1. Select **Properties** > **Execution Options**.
2. Set the `Launch Job Options` to `In the service process`.

**Note:** If the `Launch Job Options` is set to `In separate local processes`, then provide the flag value in the `Custom properties` by selecting **Processes** > **Custom Properties** in the Data Integration Service.

# Unload Command

You can use the Unload command to extract data from Amazon Redshift and create staging files on Amazon S3. The Unload command uses a secure connection to load data into one or more files on Amazon S3.

You can specify the Unload command options directly in the **Unload Options** field. Enter the options in uppercase and use a semicolon to separate the options. For example,

```
DELIMITER=\036;PARALLEL=ON;NULL=text;AWS_IAM_ROLE=arn:aws:iam::<account ID>:role/<role-name>
```

Even if you specify the delimiter as a semicolon, you must use a semicolon to separate the options. For example,

```
DELIMITER=;;PARALLEL=ON;NULL=text;AWS_IAM_ROLE=arn:aws:iam::<account ID>:role/<role-name>
```

You can create a property file to specify the Unload command options. You must place the property file in any location in the machine on which the Data Integration Service runs. Enter each option in a separate line in uppercase letters. For example:

```
DELIMITER = \036
PARALLEL = ON
NULL=text
AWS_IAM_ROLE=arn:aws:iam::<account ID>:role/<role-name>
```

Include the property file path in the **Unload Options** field. For example: `C:\Temp\Redshift\unloadoptions.txt`

It is recommended to use octal representation of non-printable characters as DELIMITER.

## Unload Command Options

The Unload command options extract data from Amazon Redshift and load data to staging files on Amazon S3 in a particular format. PowerExchange for Amazon Redshift supports ADDQUOTES, AWS_IAM_ROLE, DELIMITER, MAXFILESIZE, NULL, PARALLEL, and REGION unload command options.

To add options to the Unload command, use the **Unload Options** option. The ESCAPE option is set by default. You can set the following options:

**ADDQUOTES**

The Unload command can read data values that contain the delimiter. The Unload command adds quotation marks to each data field. If double quote (") is a part of data, use ESCAPE to read the double quote as a regular character.

Use the ADDQUOTES Unload command to read data that contains special characters.

**AWS_IAM_ROLE**

Specify the Amazon Redshift Role Resource Name (ARN) to run the session on Data Integration Service installed on an Amazon EC2 system in the following format: `AWS_IAM_ROLE=arn:aws:iam::<account ID>:role/<role-name>`

For example: `arn:aws:iam::123123456789:role/redshift_read`

**DELIMITER**

A single ASCII character to separate fields in the input file. Default is \036, the octal representation of the non-printable character and the record separator.

You can use characters such as pipe (|), tilde (~), or a tab (\t). The delimiter you specify should not be a part of the data. If the delimiter is a part of data, use ESCAPE to read the delimiter character as a regular character.

**MAXFILESIZE**

You can use the MAXFILESIZE Unload command option to limit the size of the files unloaded from a Redshift table to Amazon S3.

Specify the value in decimals between 5 MB and 6.2 GB.

Enter the value of the MAXFILESIZE Unload command option in the following format: `MAXFILESIZE=50MB`.

**NULL**

You can use the NULL Unload command option to replace the null values in an Amazon Redshift source table with the string that you specify using the NULL Unload command option.

Enter the value of the NULL Unload command option in the following format: `NULL=text`. Do not add spaces when you enter the string value. For more information about the NULL Unload command, see the AWS documentation.

**Note:** Applicable when you run a mapping in the native environment.

**PARALLEL**

The Unload command writes data in parallel to multiple files, according to the number of slices in the cluster. Default is On.

If you turn the Parallel option off, the Unload command writes data serially. The maximum size of a data file is 6.5 GB. Do not use `PARALLEL OFF` if you run a mapping on the Blaze engine.

**REGION**

You can use the REGION attribute when the Amazon S3 staging bucket is not in the same region as the cluster region. If Amazon Redshift resides in the US East (N. Virginia) region, you can use an Amazon S3 bucket residing in the Asia Pacific (Mumbai) region to create staging files. For example, `REGION = ap-south-1`.

## Amazon Redshift Data Object Read Operation Properties

The Data Integration Service reads data from Amazon Redshift based on the data object read operation.

The Developer tool displays advanced properties for the Amazon Redshift data object operation in the Advanced view.
The following table describes the Advanced properties for an Amazon Redshift data object read operation:

| Property | Description |
|---|---|
| S3 Bucket Name | Amazon S3 bucket name for staging the data.<br>You can also specify the bucket name with the folder path. If you provide an Amazon S3 bucket name that is in a different region than the Amazon Redshift cluster, you must configure the REGION attribute in the Unload command options.<br>Required if you configure a read operation enabled for staging data. Optional when you do not configure staging for the read operation.<br>**Note:** The S3 bucket name doesn't appear in the logs of the data preview table in Redshift. |
| Enable Compression | Compresses the staging files into the Amazon S3 staging directory.<br>The mapping performance improves when the Data Integration Service compresses the staging files.<br>**Note:** Applicable when you run a mapping in the native environment, on the Spark engine, or on the Databricks Spark engine. |

| Property | Description |
|---|---|
| Staging Directory Location | Amazon Redshift staging directory.<br>Specify a directory on the machine that hosts the Data Integration Service.<br>**Note:** Applicable when you run a mapping in the native environment. |
| Unload Options | Unload command options.<br>Add options to the Unload command to write data from an Amazon Redshift object to an S3 bucket. You can add the following options:<br>- DELIMITER<br>- REGION<br>- PARALLEL<br>- NULL<br>- AWS_IAM_ROLE<br>- ADDQUOTES<br>Specify a directory on the machine that hosts the Data Integration Service.<br>Do not use `PARALLEL OFF` if you run a mapping on the Blaze engine. |
| Treat NULL Value as NULL | Retains the null values when you read data from Amazon Redshift.<br>**Note:** Applicable when you run a mapping in the native environment. |
| Encryption Type | Method you want to use to encrypt data.<br>Select one of the following values:<br>- None<br>- SSE-S3<br>- SSE-KMS<br>- CSE-SMK<br>For more information, see "Data Encryption in Amazon Redshift Sources" on page 22. |
| Download S3 Files in Multiple Parts | Downloads large Amazon S3 objects in multiple parts.<br>When the file size of an Amazon S3 object is greater than 8 MB, you can choose to download the object in multiple parts in parallel.<br>**Note:** Applicable when you run a mapping in the native environment. |
| Multipart Download Threshold Size | Maximum size of an Amazon S3 object in bytes.<br>When you download large Amazon S3 objects, the large objects are broken into multiple parts. Default is 5 MB.<br>**Note:** Applicable when you run a mapping in the native environment. |
| Pre-SQL | The pre-SQL commands to run a query before you read data from Amazon Redshift. You can also use the UNLOAD or COPY command. The command you specify here is processed as a plain text.<br>**Note:** Applicable when you run a mapping in the native environment, on the Spark engine, or on the Databricks Spark engine. |
| Post-SQL | The post-SQL commands to run a query after you read data to Amazon Redshift. You can also use the UNLOAD or COPY command. The command you specify here is processed as a plain text.<br>**Note:** Applicable when you run a mapping in the native environment, on the Spark engine, or on the Databricks Spark engine. |

| Property | Description |
|---|---|
| Select Distinct | Selects unique values.<br>The Data Integration Service includes a `SELECT DISTINCT` statement in the query if you choose this option. Amazon Redshift ignores trailing spaces. Therefore, the Data Integration Service might extract fewer rows than expected.<br>**Note:** Applicable when you run a mapping in the native environment, on the Spark engine, or on the Databricks Spark engine. |
| SQL Query | Overrides the default SQL query.<br>Enclose column names in double quotes. The SQL query is case sensitive. Specify an SQL statement supported by the Amazon Redshift database.<br>When you specify the columns in the SQL query, ensure that the column name matches the source column name that are mapped in the mapping.<br>When you use the date and timestamp data types, you need to specify the format `YYYY-MM-DD HH24:MI:SS.US` to take care of the data conversion.<br>For example, for a Redshift table `sample` that contains the fields `d_date` of the date data type and `t_timestamp` of the timestamp data type, specify the SQL query in the following format:<br>`select (to_char(cast ( sample.d_date as timestamp ),'YYYY-MM-DD HH24:MI:SS.US')), (to_char(sample.t_timestamp,'YYYY-MM-DD HH24:MI:SS.US')) FROM sample;`<br>**Note:** Applicable when you run a mapping in the native environment, on the Spark engine, or on the Databricks Spark engine. |

# Amazon Redshift Data Object Write Operation

Create a mapping to move data to an Amazon Redshift table. Change the connection to an Amazon Redshift connection, and define the write operation properties to write data to Amazon Redshift.

You can perform insert, update, delete, and upsert operations on an Amazon Redshift target.

**Note:** If the distribution key column in a target table contains null values and you configure a mapping with an upsert operation for the same target table, the mapping might create duplicate rows. To avoid creating duplicate rows, you must perform one of the following tasks:

- Replace the null value with a non-null value when you load data.
- Do not configure the column as a distribution key if you expect null values in the distribution key column.
- Remove the distribution key column from the target table temporarily when you load data.

## Amazon Redshift Staging Directory for Amazon Redshift Targets

The Data Integration Service creates a staging file in the directory that you specify in the target properties. The Data Integration Service writes the data to the staging directory before it writes data to Amazon Redshift.

The Data Integration Service deletes the staged files from the staging directory after it writes the data to Amazon S3. Specify a staging directory in the mapping properties with an appropriate amount of disk space for the volume of data that you want to process. Specify a directory on the machine that hosts the Data Integration Service.

The Data Integration Service creates subdirectories in the staging directory. Subdirectories use the following naming convention:

```
<staging directory>/infaRedShiftStaging<MMddHHmmssSSS+xyz>
```

# Analyze Target Table

To optimize query performance, you can configure a mapping to analyze the target table. Target table analysis updates statistical metadata of the database tables.

You can use the Analyze Target Table option to extract sample rows from the table, analyze the samples, and save the column statistics. Amazon Redshift then updates the query planner with the statistical metadata. The query planner uses the statistical metadata to build and choose optimal plans to improve the efficiency of queries.

You can run the Analyze Target Table option after you load data to an existing table by using the Copy command. If you load data to a new table, the Copy command performs an analysis by default.

# Data Encryption in Amazon Redshift Targets

To protect data, you can enable server-side encryption or client-side encryption to encrypt the data that you insert in Amazon Redshift.

If you enable both server-side and client-side encryption for an Amazon Redshift target, then the client-side encryption is used for data load.

You can encrypt data by using the master symmetric key or customer master key. Do not use the master symmetric key and customer master key together. Customer master key is a user managed key generated by AWS Key Management Service (AWS KMS) to encrypt data. Master symmetric key is a 256-bit AES encryption key in the Base64 format that is used to enable client-side encryption. You can generate master symmetric key by using a third-party tool.

The following table lists the encryption type support for various environments:

| Encryption Type | Native Environment | Blaze Environment | Spark Environment | Databricks Environment |
|---|---|---|---|---|
| Server-side Encryption | Yes | Yes | Yes | Yes |
| Client-side Encryption | Yes | No | No | No |

## Server-side Encryption for Amazon Redshift Targets

If you want Amazon Redshift to encrypt data while uploading the `.csv` files to Amazon Redshift, you must enable server-side encryption. To enable server-side encryption, select **S3 Server Side Encryption** in the data object operation advanced properties.

For a server-side encryption, you can also use AWS KMS-managed customer master key to encrypt data when you run a mapping in the native environment, on the Spark engine, or on the Databricks Spark engine. Perform the following steps to enable server-side encryption with AWS KMS-managed customer master key:

1. Provide the customer master key ID when you create an Amazon Redshift connection.
2. Select **S3 Server Side Encryption** as the encryption type in the advanced properties of the data object operation.

**Note:** PowerExchange for Amazon Redshift supports the **Server Side Encryption With KMS** encryption type on the following distributions:

- Amazon EMR version 5.20.
- Cloudera CDH version 5.16 and 6.1.

### Client-side Encryption for Amazon Redshift Targets

Client-side encryption is a technique to encrypt data before transmitting the data to the Amazon Redshift server.

When you enable client-side encryption for Amazon Redshift targets, the Data Integration Service fetches the data from the source, writes the data to the staging directory, encrypts the data, and then writes the data to an Amazon S3 bucket. The Amazon S3 bucket then writes the data to Amazon Redshift.

If you enable both server-side and client-side encryption for an Amazon Redshift target, then the client-side encryption is used for data load.

To enable client-side encryption, you must provide a master symmetric key in the connection properties. The Data Integration Service encrypts the data by using the master symmetric key. The master symmetric key is a 256-bit AES encryption key in the Base64 format. PowerExchange for Amazon Redshift uploads the data to the Amazon S3 server by using the master symmetric key and then loads the data to Amazon Redshift by using the copy command with the Encrypted option and a private encryption key for additional security. To enable client-side encryption, perform the following tasks:

1. Provide the master symmetric key when you create an Amazon Redshift connection. Ensure that you provide a 256-bit AES encryption key in Base64 format.
2. Download the `local_policy.jar` and the `US_export_policy.jar` files for your JAVA environment from the Oracle website. Replace the existing `local_policy.jar` and the `US_export_policy.jar` files in the following directory: `<JAVA_HOME>\lib\security`.
3. Select **S3 Client Side Encryption** in the Data Object Operation Advanced properties.

## Retain Staging Files

You can retain staging files on Amazon S3 after the Data Integration Service writes data to the target. You can retain files to create a data lake of your organizational data on Amazon S3. The files you retain can also serve as a backup of your data.

When you configure the write operation, you can provide a file prefix or directory prefix to save the staging files. After you provide the prefixes, the Data Integration Service creates files within the directories at Amazon S3 location specified in the target connection. Configure one of the following options for the **Prefix for Retaining Staging Files on S3** property:

- Provide a directory prefix and a file prefix. For example, backup_dir/backup_file. The Data Integration Service creates the following directories and files:

  - `backup_dir_<year>_<month>_<date>_<timestamp_inLong>`

  - `backup_file.batch_<batch_number>.csv.<file_number>.<encryption_if_applicable>`

- Provide a file prefix. For example, backup_file. The Data Integration Service creates the following directories and files:

  - `<year>_<month>_<date>_<timestamp_inLong>`

  - `backup_file.batch_<batch_number>.csv.<file_number>.<encryption_if_applicable>`

- Do not provide a prefix. The Data Integration Service does not save the staging files.

If you run a mapping on the Blaze mode, you must only provide a directory prefix. The file prefix is ignored. The file creation is dynamic on the Blaze engine.

# Copy Command

You can use the Copy command to append data in a table. The Copy command uses a secure connection to load data from flat files in an Amazon S3 bucket to Amazon Redshift.

You can specify the Copy command options directly in the **Copy Options** field. Enter the options in uppercase and use a semicolon to separate the options. For example,

```
DELIMITER=\036;ACCEPTINVCHARS=#;QUOTE=
\037;COMPUPDATE=ON;AWS_IAM_ROLE=arn:aws:iam::<account ID>:role/<role-name>
```

Even if you specify the delimiter as a semicolon, you must use a semicolon to separate the options. For example,

```
DELIMITER=;;ACCEPTINVCHARS=#;QUOTE=\037;COMPUPDATE=ON;AWS_IAM_ROLE=arn:aws:iam::<account
ID>:role/<role-name>
```

You can create a property file to specify the Copy command options. Enter the options in uppercase in separate lines. You must place the property file in any location in the machine on which the Data Integration Service runs. For example:

```
DELIMITER = \036
ACCEPTINVCHARS = #
QUOTE = \037
COMPUPDATE = ON
AWS_IAM_ROLE=arn:aws:iam::<account ID>:role/<role-name>
```

The property file contains the Copy command options. Include the property file path in the **Copy Options** field. For example: `C:\Temp\Redshift\copyoptions.txt`

It is recommended to use octal representation of non-printable characters as DELIMITER and QUOTE.

## Copy Command Options

The Copy command options read data from Amazon S3 and write data to Amazon Redshift in a particular format. You can apply compression to data in the tables or delimit the data with a particular character. PowerExchange for Amazon Redshift only supports DELIMITER, ACCEPTINVCHARS, QUOTE, REGION, COMPUPDATE, and AWS_IAM_ROLE copy command options.

To add options to the Copy command, use the **Copy Options** option. You can set the following options:

**DELIMITER**

A single ASCII character to separate fields in the input file. Default is \036, the octal representation of the non-printable character and the record separator.

You can use characters such as pipe (|), tilde (~), or a tab (\t). The delimiter must not be a part of the data.

**ACCEPTINVCHARS**

Loads data into VARCHAR columns even if the data contains UTF-8 characters that are not valid. Default is question mark (?).

When you specify ACCEPTINCHARS, the Data Integration Service replaces UTF-8 character that is not valid with an equal length string consisting of the character specified in ACCEPTINVCHARS. If you have specified '|' in ACCEPTINVCHARS, the Data Integration Service replaces the three-byte UTF-8 character with '|||'.

If you do not specify ACCEPTINVCHARS, the COPY command returns an error when it encounters an UTF-8 character that is not valid. You can use the ACCEPTINVCHARS option on VARCHAR columns.

**QUOTE**

Specifies the quote character to use with comma separated values. Default is \037, the octal representation of the non-printable character and the unit separator.

**REGION**

You can use the REGION attribute when the Amazon S3 staging bucket is not in the same region as the cluster region. If Amazon Redshift resides in the US East (N. Virginia) region, you can use an Amazon S3 bucket residing in the Asia Pacific (Mumbai) region to create staging files. For example, `REGION = ap-south-1`.

**COMPUPDATE**

Overrides current compression encoding and applies compression to an empty table. Default is OFF.

Use the COMPUPDATE option in an insert task when the rows in a table are more than 100,000. The behavior of COMPUPDATE depends on how it is configured:

- If you do not specify COMPUPDATE, the COPY command applies compression if the target table is empty and all columns in the table have either RAW or no encoding.
- If you specify COMPUPDATE ON, the COPY command replaces the existing encodings if the target table is empty and the columns in the table have encodings other than RAW.
- If you specify COMPUPDATE OFF, the COPY command does not apply compression.

**AWS_IAM_ROLE**

Specify the Amazon Redshift Role Resource Name (ARN) to run the session on Data Integration Service installed on an Amazon EC2 system in the following format: `AWS_IAM_ROLE=arn:aws:iam::<account ID>:role/<role-name>`

For example: `arn:aws:iam::123123456789:role/redshift_write`

## Vacuum Tables

You can use vacuum tables to recover disk space and sorts rows in a specified table or all tables in the database.

After you run bulk operations, such as delete or load, or after you run incremental updates, you must clean the database tables to recover disk space and to improve query performance on Amazon Redshift. Amazon Redshift does not reclaim and reuse free space when you delete and update rows.

Vacuum databases or tables often to maintain consistent query performance. You can recover disk space for the entire database or for individual tables in a database. You must run vacuum when you expect minimal activity on the database or during designated database administration schedules. Long durations of vacuum might impact database operations. Run vacuum often because large unsorted regions result in longer vacuum times.

You can enable the vacuum tables option when you configure the advanced target properties. You can select the following recovery options:

**None**

Does not sort rows or recover disk space.

**Full**

Sorts the specified table or all tables in the database and recovers disk space occupied by rows marked for deletion by previous update and delete operations.

**Sort Only**

Sorts the specified table or all tables in the database without recovering space freed by deleted rows.

**Delete Only**

Recovers disk space occupied by rows marked for deletion by previous update and delete operations, and compresses the table to free up used space.

## Octal Values as DELIMITER and QUOTE

In addition to printable ASCII characters, you can use octal values for printable and non-printable ASCII characters as DELIMITER and QUOTE.

To use a printable character as DELIMITER or QUOTE, you can either specify the ASCII character or the respective octal value. However, to use a non-printable character as DELIMITER or QUOTE, you must specify the respective octal value.

Example for a printable character:

`DELIMITER=#` or `DELIMITER=\043`

Example for a non-printable character, file separator:

`QUOTE=\034`

Octal values 000-037 and 177 represent non-printable characters and 040-176 represent printable characters.

The following table lists the recommended octal values, for QUOTE and DELIMITER in the Copy command and as DELIMITER in the Unload command, supported by Amazon Redshift:

| Command Option | Recommended Octal Values |
|---|---|
| COPY QUOTE | 001-010, 016-037, 041-054, 057, 073-100,133, 135-140, 173-177 |
| COPY DELIMITER | 001-011, 013, 014, 016, 017, 020-046, 050-054, 057, 073-133, 135-177 |
| UNLOAD DELIMITER | 001-011, 013, 014, 016, 017, 020-041, 043-045, 050-054, 056-133, 135-177 |

## Preserve Record Order on Write

You can retain the order number of the changed record when you capture the changed record from a CDC source to a target table. This property enables you to avoid inconsistencies between the CDC source and target.

When you modify a single record in a row several times in a CDC source, enable the **Preserve record order on write** option in the advanced write operation property to retain the order number of the changed record when you write the changed record to the target table.

For example, you have a record in the following CDC source table in which you have performed multiple of operations:

| Emp ID | Emp Name | Emp Description | RowType | RowID |
|---|---|---|---|---|
| 1 | John | L1 | Insert | 1 |
| 1 | John | L2 | Update | 2 |
| 1 | John | L3 | Update | 3 |

Here, assume that the `RowID` shows the order of the changed record in the CDC source table.

The Data Integration Service writes the following changed record along with the order number in the target table:

| Emp ID | Emp Name | Emp Description | RowType | RowID |
|---|---|---|---|---|
| 1 | John | L3 | Update | 3 |

## Amazon Redshift Data Object Write Operation Properties

Amazon Redshift data object write operation properties include run-time properties that apply to the Amazon Redshift data object.

The Developer tool displays advanced properties for the Amazon Redshift data object operation in the **Advanced** view. The following table describes the Advanced properties for an Amazon Redshift data object write operation:

| Property | Description |
|---|---|
| S3 Bucket Name | Amazon S3 bucket name for staging the data.<br>You can also specify the bucket name with the folder path. Use an S3 bucket in the same region as your Amazon Redshift cluster.<br>**Note:** Redshift data preview does not log the staging data for an S3 bucket. |
| Enable Compression | Compresses the staging files before writing to Amazon Redshift.<br>By default, the compression is enabled.<br>Mapping performance improves when the Data Integration Service compresses the staged files.<br>**Note:** Applicable when you run a mapping in the native environment, on the Spark engine, or on the Databricks Spark engine. |
| Staging Directory Location | Amazon Redshift staging directory.<br>Specify a directory on the machine that hosts the Data Integration Service.<br>**Note:** Applicable when you run a mapping in the native environment. |
| Batch Size | Minimum number of rows in a batch. Enter a number greater than 0.<br>Default is 2000000.<br>**Note:** Applicable when you run a mapping in the native environment. |
| Max Errors per Upload Batch for INSERT | The number of error rows that causes an upload insert batch to fail.<br>Enter a positive integer. Default is 1.<br>If the number of errors is equal to or greater than the property value, the Data Integration Service writes the entire batch to the error file. |
| Truncate Target Table Before Data Load | Deletes all the existing data in the Amazon Redshift target table before loading new data.<br>**Note:** Truncate table and create table should be from the same user or group. If you enable the auto create DBuser checkbox, and the user belongs to a different group, the mapping fails. |
| Require Null Value For Char And Varchar | Replaces the string value with NULL when uploading data to Amazon Redshift columns of Char and Varchar data types.<br>Default is an empty string.<br>**Note:** When you run a mapping to write null values to a table that contains a single column of the Int, Bigint, numeric, real, or double data type, the mapping fails. You must provide a value other than the default value in the **Require Null Value For Char And Varchar** property. |

| Property | Description |
|---|---|
| WaitTime In Seconds For S3 File Consistency | The number of seconds to wait for the Data Integration Service to make the staged files consistent with the list of files available on Amazon S3.<br>Default is 0.<br>**Note:** Applicable when you run a mapping in the native environment or on the Blaze engine. |
| Copy Options | Copy command options.<br>Enables you to add additional options to the copy command for writing data from an Amazon S3 source to an Amazon Redshift target when the default delimiter comma (\036) or double-quote (\037) is used in the data.<br>You can add the following options:<br>- DELIMITER<br>- ACCEPTINVCHARS<br>- QUOTE<br>- REGION<br>- COMPUPDATE<br>- AWS_IAM_ROLE<br>Specify a directory on the machine that hosts the Data Integration Service. |
| S3 Server Side Encryption | Indicates that Amazon S3 encrypts data during upload and decrypts data at the time of access. |
| S3 Client Side Encryption | Indicates that the Data Integration Service encrypts data by using a private encryption key.<br>If you enable both server side and client side encryption, the Data Integration Service ignores the server side encryption.<br>**Note:** Applicable when you run a mapping in the native environment. |
| Analyze Target Table | Runs an `ANALYZE` command on the target table.<br>The query planner on Amazon Redshift updates the statistical metadata to build and choose optimal plans to improve the efficiency of queries. |
| Vacuum Target Table | Recovers disk space and sorts rows in a specified table or all tables in the database.<br>Default is None.<br>You can select the following recovery options:<br>**None**<br>Does not sort rows or recover disk space.<br>**Full**<br>Sorts the specified table or all tables in the database and recovers disk space occupied by rows marked for deletion by previous update and delete operations.<br>**Sort Only**<br>Sorts the specified table or all tables in the database without recovering space freed by deleted rows.<br>**Delete Only**<br>Recovers disk space occupied by rows marked for deletion by previous update and delete operations, and compresses the table to free up used space. |

| Property | Description |
|---|---|
| Prefix To Retain For Staging Files On S3 | Retains staging files on Amazon S3.<br>Provide both a directory prefix and a file prefix separated by a slash (/) or only a file prefix to retain staging files on Amazon S3. For example, `backup_dir/backup_file` or `backup_file`.<br>If you run a mapping on the Blaze engine, the file prefix is not retained because the file creation is dynamic on the Blaze engine. You must specify only a directory prefix. For example, `backup_dir`. |
| Success File Directory | Directory for the Amazon Redshift success file.<br>Specify a directory on the machine that hosts the Data Integration Service.<br>**Note:** Applicable when you run a mapping in the native environment. |
| Error File Directory | Directory for the Amazon Redshift error file.<br>Specify a directory on the machine that hosts the Data Integration Service.<br>**Note:** Applicable when you run a mapping in the native environment. |
| Treat Source Rows As | Overrides the Amazon Redshift target.<br>Default is **INSERT**.<br>Select one of the following options:<br>**None**<br>You can use an Update Strategy transformation to write data to an Amazon Redshift target when you select this option.<br>**Note:** You can configure an Update Strategy transformation for an Amazon Redshift target in the native environment.<br>**INSERT**<br>If enabled, the Data Integration Service inserts all rows flagged for insert. If disabled, the Data Integration Service rejects the rows flagged for insert. By default, the insert operation is enabled.<br>**DELETE**<br>If enabled, the Data Integration Service deletes all rows flagged for delete. If disabled, the Data Integration Service rejects all rows flagged for delete.<br>To perform a delete operation, you must map the primary key column and at least one column other than primary key column.<br>**UPDATE and UPSERT**<br>Performs update and upsert operations. To perform an update operation, you must map the primary key column and at least one column other than primary key column. You can select the following data object operation attributes:<br>- Update as Update: The Data Integration Service updates all rows as updates.<br>- Update else Insert: The Data Integration Service updates existing rows and inserts other rows as if marked for insert. |
| Minimum Upload Part Size | Minimum size of the Amazon Redshift object to upload.<br>Default is 5 MB.<br>**Note:** Applicable when you run a mapping in the native environment. |
| Transfer Manager Thread Pool Size | The number of threads to write data in parallel.<br>Default is 10.<br>**Note:** Applicable when you run a mapping in the native environment. |

| Property | Description |
|---|---|
| Pre-SQL | The pre-SQL commands to run a query before you write data from Amazon Redshift. You can also use the UNLOAD or COPY command. The command you specify here is processed as a plain text.<br>**Note:** Applicable when you run a mapping in the native environment, on the Spark engine, or on the Databricks Spark engine. |
| Post-SQL | The post-SQL commands to run a query after you write data to Amazon Redshift. You can also use the UNLOAD or COPY command. The command you specify here is processed as a plain text.<br>**Note:** Applicable when you run a mapping in the native environment, on the Spark engine, or on the Databricks Spark engine. |
| Schema Name | Overrides the default schema name.<br>**Note:** Applicable when you run a mapping in the native environment, on the Spark engine, or on the Databricks Spark engine. |
| Preserve record order on write | Retains record order when the Data Integration Service writes data from a CDC source to the target.<br>Use this property when you create a mapping to capture the changed record from a CDC source. This property enables you to avoid inconsistencies between the CDC source and target. |
| Number of files per batch | Specifies the number of staging files for each batch.<br>If you do not provide the number of files, the Integration Service calculates the number of staging files.<br>**Note:** Applicable when you run a mapping in the native environment. |
| Target Table Name | Overwrites the default target table name.<br>**Note:** Applicable when you run a mapping in the native environment, on the Spark engine, or on the Databricks Spark engine. |
| Recovery Schema Name | Not applicable. |
| Target Schema Strategy | Target schema strategy for the Amazon Redshift target table.<br>You can select one of the following target schema strategies:<br>- RETAIN - Retain existing target schema<br>- CREATE - Create or replace table at run time<br>**Note:** When you select the **CREATE** option, you must provide the value of the **Schema Name** property to run the mapping successfully.<br>- Assign Parameter<br>**Note:** Applicable when you run a mapping in the native environment, on the Spark engine, or on the Databricks Spark engine.<br>For more information about Target Schema Strategy, see "Target Schema Strategy" on page 42 |

# Creating an Amazon Redshift Data Object

Create an Amazon Redshift data object to add to a mapping.

1. Select a project or folder in the **Object Explorer** view.
2. Click **File** > **New** > **Data Object**.

3. Select **AmazonRedshift Data Object** and click **Next**.

   The **AmazonRedshift Data Object** dialog box appears.
4. Enter a name for the data object.
5. Click **Browse** next to the **Location** option and select the target project or folder.
6. Click **Browse** next to the **Connection** option and select the Amazon Redshift connection from which you want to import the Amazon Redshift object.
7. To add a resource, click **Add** next to the **Selected Resources** option.

   The **Add Resource** dialog box appears.
8. Select the checkbox next to the Amazon Redshift object you want to add and click **OK**.
9. Click **Finish**.
   The data object appears under Data Objects in the project or folder in the **Object Explorer** view.

# Creating a Data Object Operation

You can create a data object read or write operation for an Amazon Redshift data object. You can then add the Amazon Redshift data object operation to a mapping.

1. Select the data object in the **Object Explorer** view.
2. Right-click and select **New** > **Data Object Operation**.

   The **Data Object Operation** dialog box appears.
3. Enter a name for the data object operation.
4. Select the type of data object operation. You can choose to create a read or write operation.
5. Click **Add**.

   The **Select Resources** dialog box appears.
6. Select the Amazon Redshift data object for which you want to create the data object operation and click **OK**.
7. Click **Finish**.

The Developer tool creates the data object operation for the selected data object.

In the **Data Object Operations** tab, you can select the **View Operation** link next to the data object read or write operation name to open the operation directly after you create the data object read or write operation.

# Creating an Amazon Redshift Target

You can create a new Amazon Redshift target based on the Source transformation or any transformation.

1. Select a project or folder in the **Object Explorer** view.
2. Select a source or transformation in the mapping.
3. Right-click and select **Create Target**.
   The **Create Target** dialog box appears.

4. Select **Others** and then select **AmazonRedshift Data Object** from the list in the **Data Object Type** section.
5. Click **OK**.
   The **New AmazonRedshift Data Object** dialog box appears.
6. Enter a name for the data object.
7. Click **Finish**.

The new target appears under the **Physical Data Objects** category in the project or folder in the **Object Explorer** view.

## Rules and Guidelines for Creating an Amazon Redshift Target

Use the following rules and guidelines when you create a new Amazon Redshift target:

- You cannot define a primary key on the target table.
- You must select the connection for the new target data object in the **Connection** property after you create a new target.
- You must import an Amazon Redshift data object in the Developer tool to read the tables from the newly created target.
- The Data Integration Service creates the new target table with the name specified in the **Native Name** property in the **Data Object Details** tab or the **Table Name** advanced properties of the data object write operation.
- When you create a new target, the Data Integration Service maps all the binary port from the source to the string port in the target.
- When you create a new target, the Data Integration Service creates all the columns of the target table in the following manner:
  - Creates all the columns of the target table as nullable.
  - Creates all the columns of the SmallInt, Int, and Boolean data types as Bigint data type.
  - Creates all the columns of the Date data type as Timestamp data type.
  - Creates all the columns of the String data type as Varchar data type.
- When you create a new target, you must select the value of the **Target Schema Strategy** property as **CREATE** and provide the value of the **Schema Name** property.

# Success and Error Files

The success file contain an entry for each record that successfully writes into Amazon Redshift. The error file contains an entry for each data error.

The Data Integration Service generates success and error files after you run a mapping. Success and error files are .csv files that contain row-level details. The Data Integration Service does not overwrite success or error files. Access the error rows files and success rows files directly from the directories where they are generated. You can manually delete the files that you no longer need.

Consider the following guidelines when you configure the data object operation properties for success files:

- Specify the Success File Directory in the data object operation properties. Specify a directory on the machine that hosts the Data Integration Service.

- The success rows file uses the following naming convention: `infa_rs_<operation>_<schema.table_name>.batch_<batch_number>_file_<file_number>_<timestamp>_success.csv`.

Consider the following guidelines when you configure the data object operation properties for error files:

- Specify the Error File Directory in the data object operation properties. Specify a directory on the machine that hosts the Data Integration Service.
- For insert tasks, the error rows file uses the following naming convention: `infa_rs_<operation>_<schema.table>.batch_<batch_number>_file_<file_number>_<timestamp>_error.csv`. For upsert tasks, the error rows file uses the following naming convention: `infa_rs_<operation>_<schema.table>_<timestamp_inLong>.batch_<batch_number>_file_<file_number>_<timestamp>_error.csv`.

## Sample Error File

If a target table has the fields f_integer, f_char, and f_varchar, and if a row is rejected, the Data Integration Service generates an error file in the following format:

| Errors Details | f_integer | f_char | f_varchar |
|---|---|---|---|
| "Query Start Time: 2014-03-24 11:41:30.629 Offending File: INSERT_bdt_with_composite_key.batch_0.csv.0.gz Line Number: 4 Column Name: f_char Column Type: char Offending Value: .....Furniture Values Intl LLC_upd_upd ERROR Reason: Multibyte character not supported for CHAR (Hint: try using VARCHAR). Invalid char: c3 a6" | "3" | ""æˆ'Furniture Values Intl LLC_upd_upd"" | ""001E000000SI3jIIAT"" |
| "Query Start Time: 2014-03-24 11:42:00.763 Offending File: INSERT_bdt_with_composite_key.batch_8.csv.0.gz Line Number: 80 Column Name: f_char Column Type: char Offending Value: .....Heitkamp Inc_upd_upd ERROR Reason: Multibyte character not supported for CHAR (Hint: try using VARCHAR). Invalid char: c3 a6" | "9999" | ""æˆ'Heitkamp Inc_upd_upd"" | ""001E000000SHd7ZIAT"" |

CHAPTER 5

# Amazon Redshift Mappings

This chapter includes the following topics:



## Amazon Redshift Mapping Overview

After you create the Amazon Redshift data object with a Amazon Redshift connection, you can develop a mapping. You can define the following types of objects in the mapping:

- A Read transformation of the Amazon Redshift data object to read data from Amazon Redshift in native or non-native run-time environment.
- A Write transformation of the Amazon Redshift data object to write data to Amazon Redshift in native or non-native run-time environment.

Validate and run the mapping. You can deploy the mapping and run it or add the mapping to a Mapping task in a workflow.

## Mapping Validation and Run-time Environments

You can validate and run mappings in the native environment or in a non-native environment, such as Hadoop or Databricks.

The Data Integration Service validates whether the mapping can run in the selected environment. You must validate the mapping for an environment before you run the mapping in that environment.

When you run a mapping, you can choose to run the mapping in the native environment or in a non-native environment, such as Hadoop or Databricks. Configure the run-time environment in the Developer tool to optimize mapping performance and process data that is greater than 10 terabytes. When you run mappings in the native environment, the Data Integration Service processes and runs the mapping. When you run

mappings in a non-native environment, the Data Integration Service pushes the processing to a compute cluster, such as Hadoop or Databricks.

You can run standalone mappings, mappings that are a part of a workflow in a non-native environment. When you select the Hadoop environment, the Data Integration Service pushes the mapping logic to the Blaze or Spark engine.

When you select the Databricks environment, the Integration Service pushes the mapping logic to the Databricks Spark engine, the Apache Spark engine packaged for Databricks.

**Note:** When the tracing level is none and you run a mapping on the Spark engine, the Data Integration Service does not log the PowerExchange for Amazon Redshift details in Spark logs.

# Amazon Redshift Dynamic Mapping Overview

You can use Amazon Redshift data objects as dynamic sources and targets in a mapping.

Use the Amazon Redshift dynamic mapping to accommodate changes to source, target, and transformation logics at run time. You can use an Amazon Redshift dynamic mapping to manage frequent schema or metadata changes or to reuse the mapping logic for data sources with different schemas. Configure rules, parameters, and general transformation properties to create the dynamic mapping.

If the data source for a source or target changes, you can configure a mapping to dynamically get metadata changes at runtime. If a source changes, you can configure the Read transformation to accommodate changes. If a target changes, you can configure the Write transformation accommodate target changes.

You do not need to manually synchronize the data object and update each transformation before you run the mapping again. The Data Integration Service dynamically determine transformation ports, transformation logic in the ports, and the port links within the mapping.

There are the two options available to enable a mapping to run dynamically. You can select one of the following options to enable the dynamic mapping:

- In the **Data Object** tab of the data object read or write operation, select the **At runtime, get data object columns from data source** option when you create a mapping.
  When you enable the dynamic mapping using this option, you can refresh the source and target schemas at the runtime.
- In the **Ports** tab of the data object write operation, select the value of the **Columns defined by** property as **Mapping Flow** when you configure the data object write operation properties.
  Additionally, specify a value of the **Target Schema Strategy** data object write operation property when you select the **Mapping Flow** option.

  When you enable the dynamic mapping using this option, you can add all the Source transformation or transformation ports to the target dynamically, retain an existing target table, or create a new target table if the table does not exist in the target.

**Note:** Dynamic mapping is applicable when you run the mapping in the native environment, on the Spark engine, or on the Databricks Spark engine.

For information about dynamic mappings, see the *Informatica Developer Mapping Guide*.

## Refresh Schema

You can refresh the source or target schema at the runtime when you enable a mapping to run dynamically. You can refresh the imported metadata before you run the dynamic mapping.

You can enable a mapping to run dynamically using the **At runtime, get data object columns from data source** option in the **Data Object** tab of the Read and Write transformations when you create a mapping.

When you add or override the metadata dynamically, you can include all the existing read and write data objects in a single mapping and run the mapping. You do not have to change the source schema to update the data objects and mappings manually to incorporate all the new changes in the mapping.

You can use the mapping template rules to tune the behavior of the execution of such pipeline mapping.

When the Source or Target transformation contains updated ports such as changes in the port names, data types, precision, or scale, the Data Integration Service fetches the updated ports and runs the mapping dynamically. You must ensure that at least one of the column name in the source or target table is the same as before refreshing the schema to run the dynamic mapping successfully.

Even though the original order of the source or target ports in the table changes, the Data Integration Service displays the original order of the ports in the table when you refresh the schemas at runtime.

If there are more columns in the source table as compared to the target table, the Data Integration Service does not map the extra column to the target table and loads null data for all the unmapped columns in the target table.

**Note:** If the Source transformation contains updated columns that do not match the Target transformation, the Data Integration Service does not link the new ports by default when you refresh the source or target schema. You must create a run-time link between the transformations to link ports at run time based on a parameter or link policy in the **Run-time Linking** tab.

For information about run-time linking, see the *Informatica Developer Mapping Guide*.

## Mapping Flow

You can add all the Source transformation or transformation ports to the target dynamically when enable a mapping to run dynamically using the **Mapping Flow** option. You can then use the dynamic ports in the Write transformation.

When you select the **Mapping Flow** option, the Data Integration Service allows the Target transformation to override ports of the Write transformation with all the updated incoming ports from the pipeline mapping and loads the target table with the ports at runtime.

The Data Integration Service creates the target tables dynamically based on the metadata of the incoming ports from the pipeline mapping.

To enable a dynamic mapping using the **Mapping Flow** option, select the value of the **Columns defined by** property as **Mapping Flow** in the **Ports** tab in the Write transformation.

You must specify a value of the **Target Schema Strategy** data object write operation property when you select the **Mapping Flow** option.

## Target Schema Strategy

You can choose to retain an existing target table or create a new target table in the target when you run a dynamic mapping.

You can select one of the following options in the **Target Schema Strategy** advanced properties for the data object write operation:

**RETAIN - Retain existing target schema**

The Data Integration Service retains the existing target schema.

**Note:** When you select **RETAIN** option and if the target table does not exist or the metadata of the source and target tables do not match, the mapping fails.

**CREATE - Create or replace table at run time**

The Data Integration Service creates a new table based on the data object or the mapping flow if the table does not exist in the target.

When the Data Integration Service creates a table based on the data object, the table contains columns that match the ports in the data object. When the Data Integration Service creates a table based on the mapping flow, the table contains columns that match generated ports in the Write transformation.

**Note:** When you select **CREATE** option and if the table already exists in the target, the mapping fails with an error message stating that the target table already exists.

**Assign Parameter**

You can assign a parameter to represent the value for the target schema strategy and then change the parameter at run time.

# Partitioning Overview

When you configure an Amazon Redshift mapping to read data from Amazon Redshift, you can configure partitioning to optimize the mapping performance at run time. The partition type controls how the Data Integration Service distributes data among partitions at partition points.

To configure partitioning, you must specify the value of **Maximum Parallelism** property in the Data Integration Service and on the Developer client. Specify a value greater than 1.

The following table summarizes the partition type options that you can configure for a read or write operation on the native environment:

| Operation Type | Supported Partition Type |
|---|---|
| Read | You can select from the following partition types:<br>- **None**. By default, the Data Integration Service creates a single partition.<br>- **Dynamic**. Not applicable.<br>**Note:** If you select the dynamic partition type, a warning message appears and the Data Integration Service creates a single partition for the mapping to run.<br>- **Key Range**. The Data Integration Service distributes rows of data based on a port or set of ports that you define as the partition key. The default number of partitions is 2. |
| Write | - **None**. By default, the Data Integration Service creates a single partition.<br>- **Dynamic**. By default, the Data Integration Service creates the same number of partitions for the target based on the number of partitions you specified for the source. |

# Key Range Partitioning

You can configure key range partitioning for a source operation that runs in the native environment.

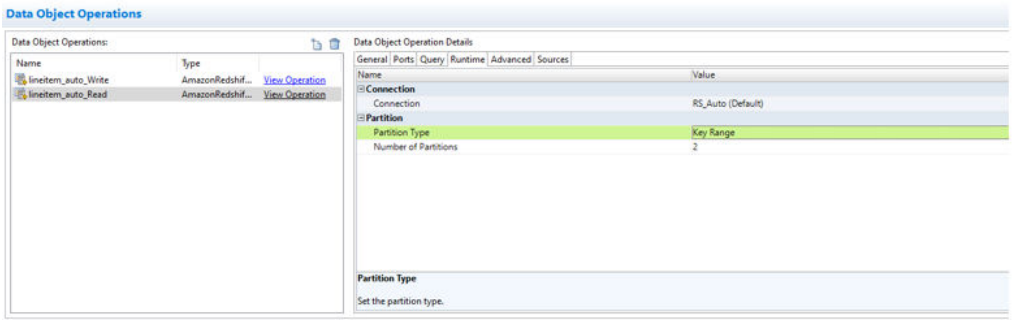
The Data Integration Service distributes rows of data based on a port or set of ports that you define as the partition key. You define a range of values for each port. The Integration Service uses the key and ranges to send rows to the appropriate partition.

For example, if you select key range partitioning, the Data Integration Service uses the key and ranges to create the WHERE clause when it selects data from the source. Therefore, you can have the Data Integration Service pass all rows that contain a specified value to one partition and all other rows to another partition.

## Configure Key Range Partitioning

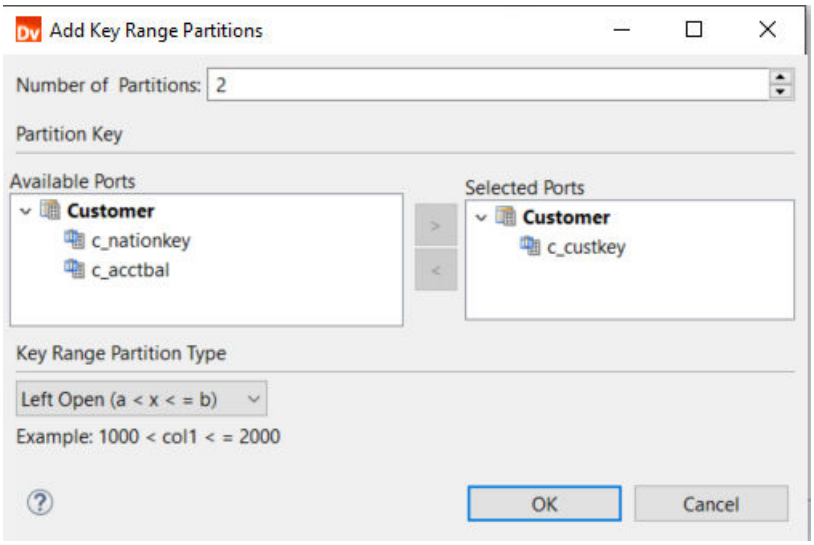
You can configure key range partitioning for a mapping to read data from an Amazon Redshift V2 source.

1. Open the Amazon Redshift V2 data object read operation for which you want to configure partitioning.
2. On the **Run-time** tab in Data Object Operation Details, select **Key Range** as the **Partition Type**.

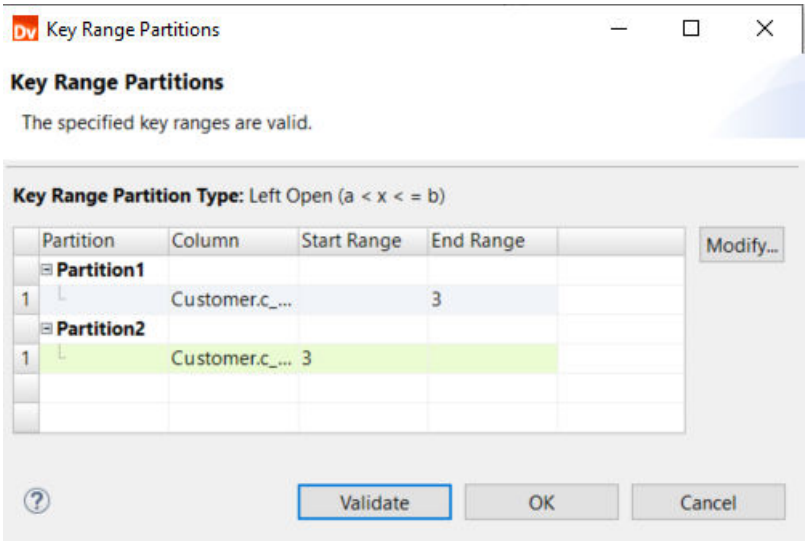



The default number of partitions created for key range partitioning is 2.

3. To configure key range partitioning, in the **Number of Partitions** field, click the **Open** button ( ).
4. In the Key Range Partitions dialog box, click **Add**.
5. Select the ports that you want to specify as the partition key, and click **OK**.



6. Specify the start and end ranges for each of the partitions and validate to verify if the specified key ranges are valid.




7. Click **OK**.

## Rules and Guidelines for Specifying Key Ranges

Use the following rules and guidelines when you create key ranges:

- You can leave the start and end range blank for a partition. When you leave the start range blank, the Integration Service uses the minimum data value as the start range. When you leave the end range blank, the Integration Service uses the maximum data value as the end range.
- Specify the key range for the partitions in a continuous sequence.
  For example, if you configure three partitions, where 1-3 is the range for the first partition, partition 2 must start with the end range of the first partition and partition 3 must start with the end range of the second partition:

  partition 1: 1-3

  partition 2: 3-5

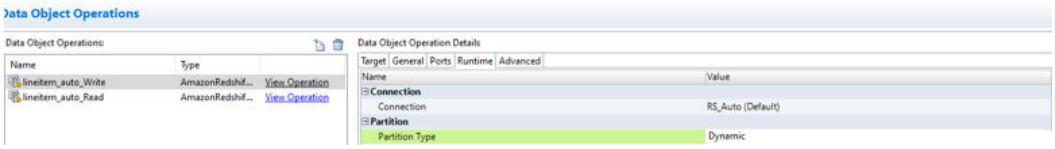
  partition 3: 5-7

# Dynamic Partitioning

When you configure a write operation on the native environment, you can configure dynamic partitioning to optimize the mapping performance.

When you select dynamic partitioning, the Data Integration Service determines the number of partitions to create at runtime based on the number of partitions specified for the source.

The following image shows the value of the partitioning type set to dynamic:



# Amazon Redshift Mapping Example

Your organization has a large amount of customers data from for all regions in flat files. You organization needs to analyze customer data in the US region in a short span of time. Create a mapping that reads all the customer records and write the records to Amazon Redshift table.

You can use the following objects in a Amazon Redshift mapping:

**Flat file input**

The input file is a flat file that contains the customer names and other details about customers.

Create a flat file data object. Configure the flat file connection and specify the flat file that contains the customer data as a resource for the data object. Drag the data object into a mapping as a read data object.

**Transformations**

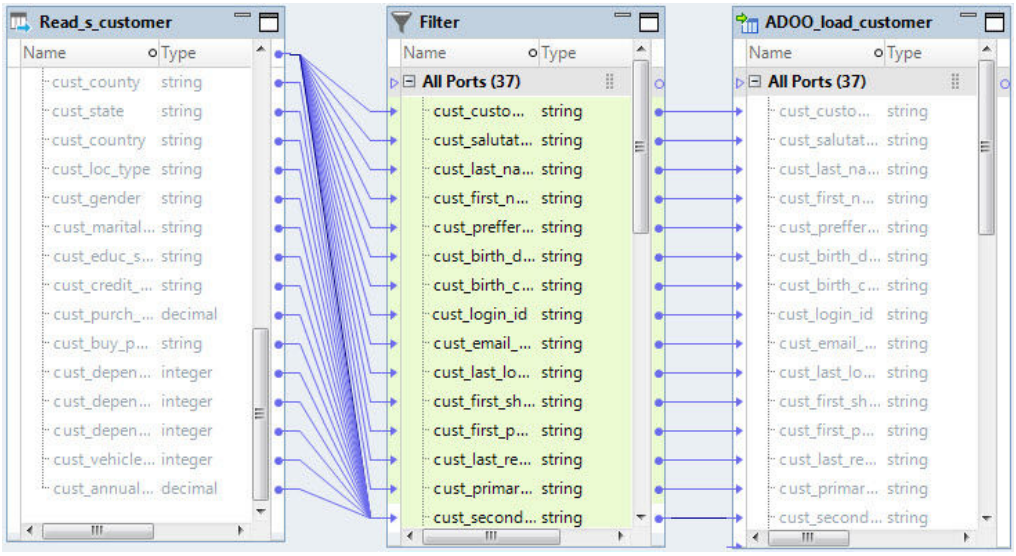
Add Filter transformation to get customer data in a particular region.

The Filter transformation filters the source data based on the value you specify for the region ID column. The Data Integration Service returns the rows that meet the filter condition.

**Amazon Redshift output**

Create an Amazon Redshift data object write operation. Configure the Amazon Redshift connection and specify the Amazon Redshift object as a target for the data object. Drag the data object into a mapping as a target data object.

The following image shows the Amazon Redshift mapping example:




When you run the mapping, the customer records are read from the flat file and written to the Amazon Redshift table.

# Amazon Redshift Dynamic Mapping Example

Your organization has a large amount of data that keeps changing. Your organization needs to incorporate all the updated data in a short span of time. Create a dynamic mapping, where you can refresh the source schema dynamically to fetch the updated data. Add all the dynamic ports to the target to override the metadata of the existing ports.

1. Import the Amazon Redshift read and write data objects.
2. Select a project or folder in the **Object Explorer** view.
3. Click **File** > **New** > **Mapping**.

   The **Mapping** dialog box appears.
4. Enter the name of the mapping in the **Name** field.
5. Click **Finish**.
6. Drag the data object into a mapping.

   The **AmazonRedshift Data Object Access** dialog box appears.
7. Select the **Read** option and click **OK**.
8. In the **Data Object** tab, select the **At runtime, get data object columns from data source** check box.
9. Drag the data object into a mapping.

   The **AmazonRedshift Data Object Acess** dialog box appears.
10. Select the **Write** option and click **OK**.
11. In the **Ports** tab, select the value of the **Columns defined by** as **Mapping Flow**.
12. In the **Advanced** tab, specify a value of the **Target Schema Strategy**.
13. Select all the source incoming ports and add the ports to the target.
14. Save and run the mapping.

CHAPTER 6

# Pushdown Optimization

This chapter includes the following topics:



## Amazon Redshift Pushdown Optimization Overview

You can use pushdown optimization to push transformation logic to source or target databases.

**Note:** You can run a mapping configured for pushdown optimization in the native environment.

Use pushdown optimization to improve mapping performance by using the database resources. When you run a mapping configured for pushdown optimization, the mapping converts the transformation logic to an SQL query. The mapping sends the query to the database, and the database executes the query.

Amazon Redshift supports source-side and full pushdown optimization for mappings. You can perform insert, update, or delete operation in a pushdown optimization.

**Example**: You work for a rapidly growing data science organization. Your organization develops software products to analyze financials, building financial graphs connecting people profiles, companies, jobs, advertisers, and publishers. The organization uses infrastructure based on Amazon Redshift Services and stores its data in Amazon Redshift database, a petabyte-scale data warehouse. The organization plans to implement a business intelligence service to build visualization and perform real-time analysis. Therefore, you need to port the vast amount of data stored in one database of Amazon Redshift to another Amazon Redshift database. Then, use MPP to run high-performance analytics.

You can use an ODBC connection to read this large amount of data from and write data to Amazon Redshift. Use full pushdown for the ODBC connection type to enhance the performance.

To read data from and write data to a Amazon Redshift object using the OBDC connection, perform the following steps:

1. Download and install the Amazon Redshift ODBC driver.

2. Configure a system DSN.
3. Create an ODBC connection to access the Amazon Redshift read and write data objects.
4. Import the Amazon Redshift read and write data objects.
5. Create and run a mapping.

# Installing the Amazon Redshift ODBC drivers

Download and install the Amazon Redshift ODBC drivers from the AWS website to connect to Amazon Redshift using an ODBC connection.

PowerExchange for Amazon Redshift supports Amazon ODBC Redshift drivers on Windows and Linux systems. Based on the operating system, download and install the Amazon Redshift ODBC drivers on the Developer client and server machines.

# Configuring a System DSN

To use an ODBC connection to connect to Amazon Redshift, you must configure a System DSN in the ODBC datasource administrator based on the operating system.

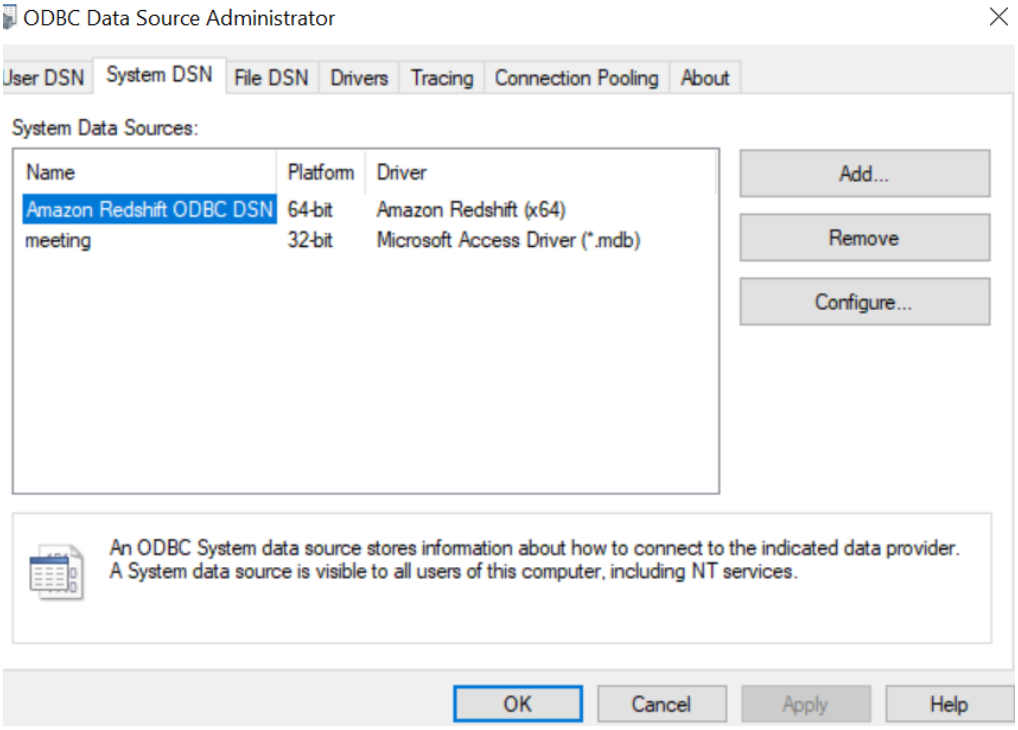
## Configuring a System DSN on Windows

Configure a System DSN on Windows to use ODBC connection to connect to Amazon Redshift.

1. Open and double-click the `odbcad32.exe` ODBC data source file from the following location: `C:\WINDOWS\system32`.

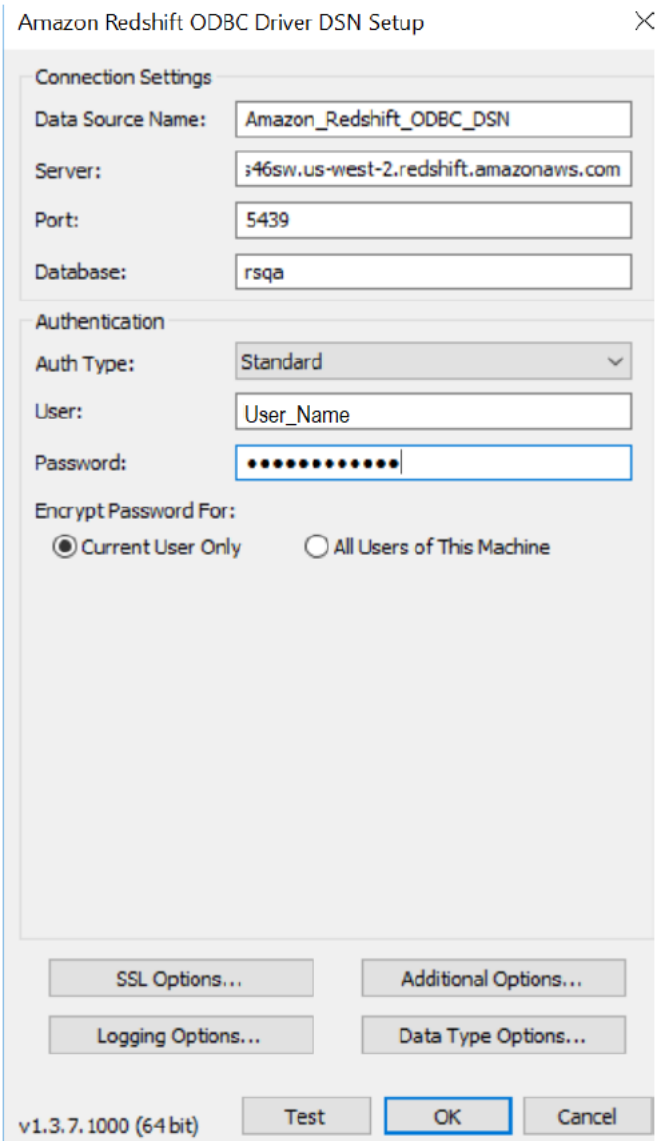
   The **ODBC Data Sources Administrator** box appears.
2. Click **System DSN**.

The **System DSN** tab appears. The following image shows the **System DSN** tab on the **ODBC Data Sources Administrator** box:




3. Click **Configure**.

The **Amazon Redshift ODBC Driver DSN Setup** box appears. The following image shows the **Amazon Redshift ODBC Driver DSN Setup** box where you can configure the connection settings and authentication:




4. Specify the following connection properties in the **Connection Settings** section:

| Property | Description |
|---|---|
| Data Source Name | Name of the data source. |
| Server | Location of the Amazon Redshift server. |
| Port | Port number of the Amazon Redshift server. |
| Database | Name of the Amazon Redshift database. |

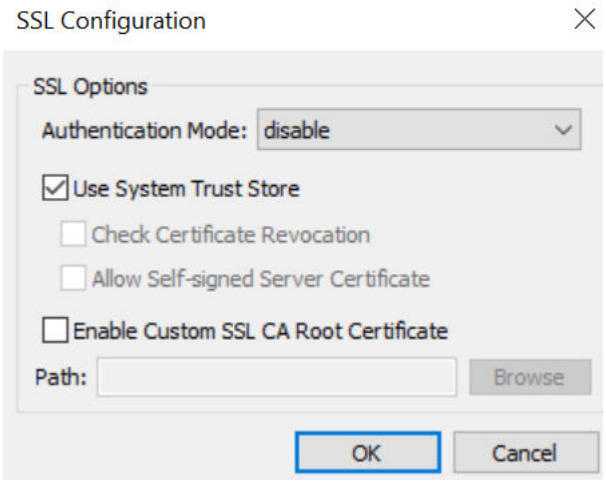
**Note:** You must specify the **Server**, **Port**, and **Database** values from the JDBC URL.

5. Specify the following authentication properties in the **Authentication** section:

| Property | Description |
|---|---|
| Auth Type | Type of the authentication.<br>Default is **Standard**. |
| User | User name to access the Amazon Redshift database. |
| Password | Password for the Amazon Redshift database. |
| Encrypt Password For | Encrypts the password for the following users:<br>- **Current User Only**<br>- **All Users of This Machine**<br>Default is **Current User Only**. |

6. Click **SSL Options** in the **Amazon Redshift ODBC Driver DSN Setup** box.

   The **SSL Configuration** box appears. The following image shows the **SSL Configuration** box:




7. Select **disable** to disable the authentication in the **Authentication Mode** field.
8. Click **OK** in the **SSL Configuration** box.

   The **SSL Configuration** box closes.
9. Click **Test** to test the connection in the **Amazon Redshift ODBC Driver DSN Setup** box.
10. Click **OK**.

## Configuring a System DSN on Linux

Configure a System DSN on Linux to use ODBC connection to connect to Amazon Redshift.

1. Configure the `odbc.ini` file properties in the following format:

```
[ODBC Data Sources]
driver_name=dsn_name

[dsn_name]
Driver=path/driver_file
```

```
Host=cluster_endpoint
Port=port_number
Database=database_name
```

2. Specify the following properties in the `odbc.ini` file:

| Property | Description |
|---|---|
| ODBC Data Sources | Name of the data source. |
| Driver | Location of the Amazon Redshift ODBC driver file. |
| Host | Location of the Amazon Redshift host. |
| Port | Port number of the Amazon Redshift server. |
| Database | Name of the Amazon Redshift database. |

**Note:** You must specify the **Host**, **Port**, and **Database** values from the JDBC URL.

3. Add the `odbc.ini` file path in your source file in the following format:

```
ODBCINI=/<odbc.ini file path>/odbc.ini
```

4. Restart the Data Integration Service.

# Creating an Amazon Redshift ODBC Connection

Create an Amazon Redshift ODBC connections to access Amazon Redshift read and write data objects.

1. In the Developer tool, click **New Connections**.

   The **Select a Connection Category** box appears.
2. Select **Connections**.
3. Specify the name, description, and location for the connection.
4. Select **Type** as **ODBC** and click **Next**.
5. Configure the following connection properties:

| Relational Connection Property | Description |
|---|---|
| User Name | Enter the user name to connect to the Amazon Redshift database. |
| Password | Enter the password to connect to the Amazon Redshift database. |
| Connection String | Enter the name of the ODBC data source that you created for teh Amazon Redshift database on the Data Integration Service host machine.<br>For example: `AmazonRSDW_ODBC_SERVER` |

| Relational Connection Property | Description |
|---|---|
| Code Page | Select the code page that the Data Integration Service uses to read or write data. |
| ODBC Provider | Select **AWS Redshift**. |

6. Test the connection and click **OK**.

   The Amazon Redshift ODBC connection is created successfully.

# Importing the Amazon Redshift Data Objects

Import the Amazon Redshift read data object from which you want to read data and the write data object to write data.

1. Select a project or folder in the **Object Explorer** view.
2. Click **File** > **New** > **Data Object**.
3. Select **Relational Data Object** and click **Next**.

   The **New Relational Data Object** dialog box appears.
4. Click **Browse** next to the **Connection** option and select the ODBC connection from which you want to import the Amazon Redshift resources.
5. Click **Create data from existing resource**.
6. To add a resource to the **Relational Data Object**, click **Browse** next to the **Resource** option.
7. Navigate to the resource to add it to the data object and click **Ok**.
8. Click **Browse** next to the **Location** option and select the target project or folder.
9. Click **Finish**.

The data object appears under **Data Object** in the project or folder in the **Object Explorer** view. You can also add resources to a relational data object after you create it.

# Creating a Mapping

After you import the read and write data objects, create a mapping to read data from and write data to Amazon Redshift.

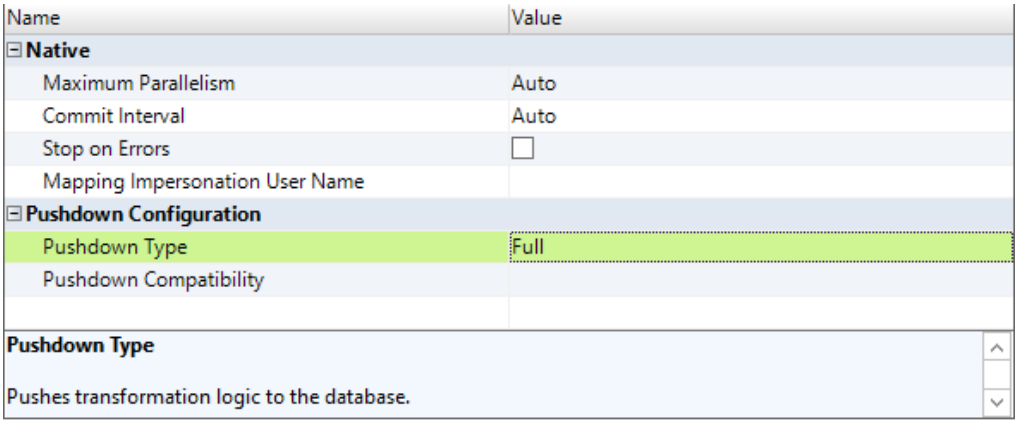
In a mapping, you define properties that determine how the Data Integration Service extracts data from or loads data to a data source. Configure pushdown optimization for the read and write data objects.

1. Select a project or folder in the **Object Explorer** view.
2. Click **Tasks** > **Create**.
3. Click **File** > **New** > **Mapping**.
4. Enter a mapping name and click **Finish**.

   An empty mapping appears in the editor.

5. Drag a data object to the editor and select **Read** to add the data object as a source.
6. Drag a data object to the editor and select **Write** to add the data object as a target.
7. On the **Properties** tab, select **Run-time**.
8. Select `Full` as pushdown type.

   The following image shows the pushdown optimization configuration:




9. Save and run the mapping.

# Supported Pushdown Optimization Functions and Operators

The following table summarizes the availability of pushdown functions in an Amazon Redshift database:

| Function | Pushdown | Function | Pushdown | Function | Pushdown |
|---|---|---|---|---|---|
| ABORT() | No | INITCAP() | Yes | REG_MATCH() | No |
| ABS() | Yes | INSTR() | Yes | REG_REPLACE | No |
| ADD_TO_DATE() | Yes | IS_DATE() | No | REPLACECHR() | No |
| AES_DECRYPT() | No | IS_NUMBER() | No | REPLACESTR() | No |
| AES_ENCRYPT() | No | IS_SPACES() | No | REVERSE() | No |
| ASCII() | No | ISNULL() | Yes | ROUND(DATE) | No |
| AVG() | Yes | LAST() | No | ROUND(NUMBER) | Yes |
| CEIL() | Yes | LAST_DAY() | Yes | RPAD() | Yes |
| CHOOSE() | No | LEAST() | No | RTRIM() | Yes |
| CHR() | Yes | LENGTH() | Yes | SET_DATE_PART() | No |

| Function | Pushdown | Function | Pushdown | Function | Pushdown |
|---|---|---|---|---|---|
| CHRCODE() | No | LN() | Yes | SIGN() | Yes |
| COMPRESS() | No | LOG() | No | SIN() | Yes |
| CONCAT() | Yes | LOOKUP | No | SINH() | No |
| COS() | Yes | LOWER() | Yes | SOUNDEX() | No |
| COSH() | No | LPAD() | Yes | SQRT() | Yes |
| COUNT() | Yes | LTRIM() | Yes | STDDEV() | Yes |
| CRC32() | No | MAKE_DATE_TIME() | No | SUBSTR() | Yes |
| CUME() | No | MAX() | Yes | SUM() | Yes |
| DATE_COMPARE() | Yes | MD5() | No | SYSTIMESTAMP() | Yes |
| DATE_DIFF() | Yes | MEDIAN() | No | TAN() | Yes |
| DECODE() | Yes | METAPHONE() | No | TANH() | No |
| DECODE_BASE64() | No | MIN() | Yes | TO_BIGINT | Yes |
| DECOMPRESS() | No | MOD() | YesS | TO_CHAR(DATE) | Yes |
| ENCODE_BASE64() | No | MOVINGAVG() | No | TO_CHAR(NUMBER) | Yes |
| EXP() | Yes | MOVINGSUM() | No | TO_DATE() | Yes |
| FIRST() | No | NPER() | No | TO_DECIMAL() | Yes |
| FLOOR() | Yes | PERCENTILE() | No | TO_FLOAT() | Yes |
| FV() | No | PMT() | No | TO_INTEGER() | Yes |
| GET_DATE_PART() | Yes | POWER() | Yes | TRUNC(DATE) | Yes |
| GREATEST() | No | PV() | No | TRUNC(NUMBER) | Yes |
| IIF() | Yes | RAND() | No | UPPER() | Yes |
| IN() | Yes | RATE() | No | VARIANCE() | Yes |
| INDEXOF() | No | REG_EXTRACT() | No | | |

You can use the following operators lists to use in an Amazon Redshift database:

- + , - , *, /, %
- >, = , >= , <=
- AND, OR, NOT
- ||, !=, ^=

The following table lists the transformation types that PowerExchange for Amazon Redshift supports for the source-side or full pushdown optimization:

| Transformation | Source-Side | Full |
|---|---|---|
| Aggregator | Yes | Yes |
| Expression | Yes | Yes |
| Filter | Yes | Yes |
| Joiner | Yes | Yes |
| Lookup | Yes | Yes |
| Router | No | Yes |
| Sorter | Yes | Yes<br>**Note:** If you select the **Distinct rows** option in the Sorter transformation to push the transformation using full pushdown optimization, the Data Integration Service does not ignore the `ORDER BY` clause.<br>If you select the **All rows** option in the Sorter transformation, the Data Integration Service ignores the `ORDER BY` clause. |
| Union | Yes | Yes |
| Update Strategy | No | Yes<br>**Note:** To push the Update Strategy transformation using full pushdown optimization, do not select the **Forward Rejected Rows** option in the Update Strategy transformation. Additionally, in the **Run-time** tab in the mapping, select the value of the **Pushdown Compatibility** property as **Rows do not have the same key**. |

# Rules and Guidelines for Functions in Pushdown Optimization

Use the following rules and guidelines when pushing functions to an Amazon Redshift database:

- To use pushdown optimization, you must set the value of the **Multiple Matches** Lookup transformation property to **Return all rows**.
- To push TRUNC(DATE) to Amazon Redshift, you must define the date and format arguments. Otherwise, the Data Integration Service does not push the function to Amazon Redshift.
- To push TO_DATE() to Amazon Redshift, you must define the string and format arguments. Otherwise, the Data Integration Service does not push the function to Amazon Redshift.

- For Amazon Redshift, when you define only a string argument for TO_DATE() and TO_CHAR(), the Data Integration Service considers the default date format present in the session property. The default date format in the session property is: MM/DD/YYYY HH24:MI:SS.US
- Do not specify a format for SYSTIMESTAMP() to push the SYSTIMESTAMP to Amazon Redshift. The Amazon Redshift database returns the complete time stamp.
- The aggregator functions for Amazon Redshift accept only one argument, a field set for the aggregation function. The filter condition argument is not honored. In addition, make sure that all fields mapped to the target are listed in the GROUP BY clause.
- To push INSTR() to Amazon Redshift, you must only define string, search_value, and start arguments. Amazon Redshift does not support occurrence and comparison_type arguments.
- The flag argument is ignored when you push TO_BIGINT and TO_INTEGER to Amazon Redshift.
- The CaseFlag argument is ignored when you push IN() to Amazon Redshift.
- If you use the NS format as part of the ADD_TO_DATE() function, the Data Integration Service does not push the function to Amazon Redshift.
- If you use any of the following formats as part of the TO_CHAR() and TO_DATE() functions, the Data Integration Service does not push the function to Amazon Redshift:
  - NS
  - SSSS
  - SSSSS
  - RR
- To push TRUNC(DATE), GET_DATE_PART(), and DATE_DIFF() to Amazon Redshift, you must use the following formats:
  - D
  - HH24
  - MI
  - MM
  - MS
  - SS
  - US
  - YYYY
- To push GET_DATE_PART() to Amazon Redshift, you must use the following formats:
  - D
  - DDD
  - HH24
  - MI
  - MM
  - MS
  - SS
  - US
  - YYYY

CHAPTER 7

# Amazon Redshift Lookup

This chapter includes the following topics:



## Amazon Redshift Lookup Overview

You can use an Amazon Redshift data object read operation to look up data in an Amazon Redshift table.

You can add an Amazon Redshift data object read operation as a lookup in a mapping. You can then configure a lookup condition to look up data from the Amazon Redshift table. You can configure the following types of lookup on an Amazon Redshift table:

**Cached Lookup**

You can configure a cached lookup operation to cache the lookup data on the Spark and Databricks Spark engine.

**Uncached Lookup**

You can configure an uncached lookup operation when you do not want the Data Integration Service to cache the lookup data in the native environment.

**Note:** When you configure an uncached lookup operation, the Data Integration Service does not honor the Amazon Redshift data object read advanced properties. However, you must provide the Amazon S3 bucket name in the Amazon Redshift data object read advanced properties to validate the read operation.

For more information about cached and uncached lookup, see *"Lookup Transformation" in the Developer Transformation Guide*.

# General Properties

The general properties display the name and description of the Amazon Redshift lookup.

The following table describes the general properties that you can view and edit for an Amazon Redshift lookup:

| Property | Description |
|---|---|
| Name | Name of the Amazon Redshift lookup. |
| Description | Description of the Amazon Redshift lookup. |
| Physical Data Object | Name of the Amazon Redshift data object read operation. |
| On multiple matches | Determines which row the Amazon Redshift lookup returns when it finds multiple rows that match the lookup condition.<br>You can select one of the following options:<br>- Return first row<br>- Return last row<br>- Return any row<br>- Return all rows<br>- Report error<br>**Note:** When you configure an uncached lookup operation in the native environment and select the **Return last row** option, the Data Integration Service returns the first row that matches the lookup condition and generates an error message in the session log. |

# Ports Properties

The ports properties display the input ports from the source in the mapping to the Amazon Redshift lookup. You can specify the ports to be available as output ports from the Amazon Redshift lookup. The ports properties display the data types, precision, and scale of the source port.

The following table describes the ports properties:

| Property | Description |
|---|---|
| Name | Name of the source port. |
| Type | Data type of the source port. |
| Precision | Maximum number of significant digits for numeric data types, or maximum number of characters for string data types. For numeric data types, precision includes scale. |
| Scale | Maximum number of digits after the decimal point of numeric values. |
| Output | Specify the ports that must be available as output ports from the Amazon Redshift lookup. |

| Property | Description |
|---|---|
| Description | Description of the port. |
| Input Rules | A set of rules that filter the ports to include or exclude in the transformation based on port names or data type. Configure input rules when you define dynamic ports. |

# Run-time Properties

Set the run-time properties to configure a cached or uncached lookup in a mapping.

When you enable lookup caching, the Data Integration Service caches the lookup values. The Data Integration Service queries the lookup source once, caches the values, and looks up values in the cache. Caching the lookup values can increase performance on large lookup tables. By default, the **Lookup caching enabled** check box is selected.

When you disable caching, the Data Integration Service does not caches the lookup values. The Data Integration Service queries the lookup source instead of building and querying the lookup cache. Each time a row passes, the Data Integration Service issues a SELECT statement to the lookup source for lookup values. Do not select the **Lookup caching enabled** check box to enable the uncached lookup.

# Lookup Properties

Specify the lookup properties to look up an Amazon Redshift table. You can configure a lookup condition to look up data from the Amazon Redshift table.

There are two types of option that you must select in the **Specify by** property to configure a lookup condition:

- **Value**: Select this option if you want to configure a lookup condition using the column name.
- **Parameter**: Select this option if you want to parameterize the lookup condition.

The following table describes the lookup properties that you can specify for an Amazon Redshift lookup if you select the **Value** option:

| Property | Description |
|---|---|
| Lookup Column | The name of the columns that you want to look up. |
| Operator | Operators that you can use to filter records.<br>You can select one of the following operators:<br>`=, !=, <=, >=, and` |
| Input Port | The input source port. |

The following table describes the lookup properties that you can specify for an Amazon Redshift lookup if you select the **Parameter** option:

| Property | Description |
|---|---|
| Parameter | The name of the parameter that you want to use to look up.<br>You can also create a new parameter.<br>Click **New** to create a new parameter. Enter the parameter name and specify an expression in the **New Parameter** dialog box. Click **Validate** to check if the expression that you specified is valid or not. |

# Adding an Amazon Redshift Data Object Read Operation as a Lookup in a Mapping

You can add an Amazon Redshift data object read operation as a lookup to look up data in an Amazon Redshift table.

1. Open a mapping from the **Object Explorer** view.
2. From the **Object Explorer** view, drag an Amazon Redshift data object read operation to the mapping editor.
   The **Add to Mapping** dialog box appears.
3. Select **Lookup** to add the data object read operation as an operation to the mapping.
4. Select the Amazon Redshift data object read operation and connect the lookup input ports and the lookup output ports.
5. In the **Properties** view, configure the following parameters:
   a. On the **General** tab, select the option that you want the Data Integration Service to return when it finds multiple rows that match the lookup condition.
   b. On the **Ports** tab, configure the output ports and input rules.
   c. On the **Run-time** tab, disable lookup caching if you want to enable uncached lookup.
   d. On the **Lookup** tab, configure the lookup condition.
6. When the mapping is valid, click **File** > **Save** to save the mapping to the Model repository.

APPENDIX A

# Amazon Redshift Datatype Reference

This appendix includes the following topics:



## Datatype Reference Overview

When you run the mapping to read data from or write data to Amazon Redshift, the Data Integration Service converts the transformation data types to comparable native Amazon Redshift data types.

## Amazon Redshift and Transformation Datatypes

The Amazon Redshift data types are the names and aliases that represent how the Data Integration Service stores the data types.

For example, Smallint is the Amazon Redshift data type name. The data type is stored as a 2-byte integer. Here, Smallint is the Amazon Redshift data type name and Int2 is the Amazon Redshift data type alias.

The following table compares the Amazon Redshift data types and the transformation data types:

| Amazon Redshift Data Type | Amazon Redshift Data Type Aliases | Description | Transformation Data Type |
|---|---|---|---|
| Bigint | Int8 | Signed eight-byte integer. | Bigint |
| Boolean | Bool | Logical Boolean (true/false). | Small Integer |
| Char | Character, Nchar, Bpchar | Fixed-length character string. | String |
| Date | NA | Calendar date (year, month, day). | Timestamp |

| Amazon Redshift Data Type | Amazon Redshift Data Type Aliases | Description | Transformation Data Type |
|---|---|---|---|
| Decimal | Numeric | Exact numeric of selectable precision. | Decimal |
| Double Precision | Float8, Float | Double precision floating-point number. | Double |
| Integer | Int, Int4 | Signed four-byte integer. | Integer |
| Real | Float4 | Single precision floating-point number. | Double |
| Smallint | Int2 | Signed two-byte integer. | Small Integer |
| Super | Super | Stores semi-structured data or documents as values.<br>Maximum value is 16 MB.<br>For more information on super data type, see the AWS documentation. | String |
| Timestamp | Timestamp without time zone | Date and time (without time zone). | Timestamp |
| Timestampz | Timestamp with time zone | Date and time (with time zone). | Timestamp |
| Varchar | Character Varying, Nvarchar, Text | Variable-length character string with a user-defined limit. | String |

**Note:** You can use the Timestampz data type when you run mappings in the native environment.

# Rules and guidelines for data types

The Amazon Redshift data types are the names and aliases that represent how the Data Integration Service stores the data types.

## Super

Consider the following guidelines for data that contains the super data type:

- You cannot condition in a simple filter or an uncached lookup for the super data type columns.
- By default, the precision of the super data type is 256. If you do not set the precision correctly in the mapping, the mapping fails due to data truncation or target overflow.
- When you map the following data types to the super data type in the target, the COPY command fails:
  - char
  - varchar
  - date
  - time
  - timestamp

- timestamptz

The issue occurs because the super column in a Amazon Redshift table always expects the string data to be enclosed within double quotes while loading the data using COPY command. When you map the Char and Varchar data types to the Super data type in the target, ensure that the incoming data contains double quotes (").

APPENDIX B

# Troubleshooting

This appendix includes the following topics:



## Troubleshooting Overview

Use the following sections to troubleshoot errors in PowerExchange for Amazon Redshift.

## Troubleshooting for PowerExchange for Amazon Redshift

**How to enable Metadata Access Service for PowerExchange for Amazon Redshift?**

You can optionally enable Metadata Access Service to import metadata from Amazon Redshift. For information on how to enable Metadata Access Service, see https://knowledge.informatica.com/s/article/HOW-TO-Enable-Metadata-Access-Service-to-import-metadata-from-Amazon-S3-and-Amazon-Redshift?language=en_US

**How to configure performance tuning and sizing guidelines for PowerExchange for Amazon Redshift on the Spark engine?**

For information about performance tuning and sizing guidelines, see https://docs.informatica.com/data-integration/powerexchange-adapters-for-informatica/h2l/1111-performance-tuning-and-sizing-guidelines-for-powerexchange-/abstract.html

**How to solve the following error that occurs while running an Amazon Redshift mapping on the Spark engine to write a table that contains more than 500 columns: "java.lang.StackOverflowError"**

For information about the issue, see https://kb.informatica.com/solution/23/Pages/67/544565.aspx?myk=544565

**How to solve the following error that occurs while running an Amazon Redshift mapping on the Spark engine to read from a table that contains more than 510 columns: "java.lang.StackOverflowError"**

For information about the issue, see https://kb.informatica.com/solution/23/Pages/62/516323.aspx?myk=516323

**How to solve the following error that occurs while running an Amazon Redshift mapping on the Spark engine to read or write data: "No space available in any of the local directories"**

For information about the issue, see https://kb.informatica.com/solution/23/Pages/62/516324.aspx?myk=516324

**How to solve the following error that occurs while running an Amazon Redshift mapping on the Spark engine to read or write data: "Container is running beyond physical memory limits in EMR cluster"**

For information about the issue, see https://kb.informatica.com/solution/23/Pages/62/516326.aspx?myk=516326

**How to solve the following error that occurs while running an Amazon Redshift mapping on the Spark engine to read data: "com.amazonaws.AmazonClientException: Unable to execute HTTP request: Read timed out"**

For information about the issue, see https://kb.informatica.com/solution/23/Pages/62/516327.aspx?myk=516327

**How to solve the out of disk space error that occurs when you use PoweExchange for Amazon Redshift to read and preview data?**

For information about the issue, see https://kb.informatica.com/solution/23/Pages/62/516321.aspx?myk=516321

**Mapping on the Spark engine fails with an error when you use the Amazon S3 staging bucket with a dot (.) in the bucket name for CDP 7.1 distribution.**

If you run a mapping on the Spark engine with such configurations, the mapping fails with the following error message:

```
Unable to execute HTTP request: Certificate for xxxx doesn't match any of the
subject alternative names
```

Perform the following steps to run the mapping successfully:

1. In the CDP cluster, go to **HDFS**.
2. Click **Configuration**.
3. In **Cluster-wide Advanced Configuration Snippet (Safety Valve) for core-site.xml**, add the property **fs.s3a.path.style.access** and set the value to **true**.
4. Restart the cluster.
5. Refresh the cluster configuration object.
6. Restart the Data Integration Service.

# INDEX

## R



## S



## T



## U