Informatica

Informatica® Cloud Data Intergration

Amazon Redshift Connectors

Informatica Cloud Data Intergration Amazon Redshift Connectors
April 2020

DISCLAIMER: Informatica LLC provides this documentation "as is" without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of noninfringement, merchantability, or use for a particular purpose. Informatica LLC does not warrant that this software or documentation is error free. The information provided in this software or documentation may include technical inaccuracies or typographical errors. The information in this software and documentation is subject to change at any time without notice.

NOTICES

This Informatica product (the "Software") includes certain drivers (the "DataDirect Drivers") from DataDirect Technologies, an operating company of Progress Software Corporation ("DataDirect") which are subject to the following terms and conditions:

1. THE DATADIRECT DRIVERS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT.
2. IN NO EVENT WILL DATADIRECT OR ITS THIRD PARTY SUPPLIERS BE LIABLE TO THE END-USER CUSTOMER FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, CONSEQUENTIAL OR OTHER DAMAGES ARISING OUT OF THE USE OF THE ODBC DRIVERS, WHETHER OR NOT INFORMED OF THE POSSIBILITIES OF DAMAGES IN ADVANCE. THESE LIMITATIONS APPLY TO ALL CAUSES OF ACTION, INCLUDING, WITHOUT LIMITATION, BREACH OF CONTRACT, BREACH OF WARRANTY, NEGLIGENCE, STRICT LIABILITY, MISREPRESENTATION AND OTHER TORTS.

The information in this documentation is subject to change without notice. If you find any problems in this documentation, report them to us at infa_documentation@informatica.com.

Informatica products are warranted according to the terms and conditions of the agreements under which they are provided. INFORMATICA PROVIDES THE INFORMATION IN THIS DOCUMENT "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT.

Publication Date: 2020-06-22

Table of Contents

Preface

Use *Amazon Redshift Connectors* to learn about *Amazon Redshift* and *Amazon Redshift V2 Connectors.* Use Part I to know the overview and the functionality comparison between Amazon Redshift V2 and Amazon Redshift connectors. Use Part II and Part III to learn about the functionality available for Amazon Redshift connectors.

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.

Part I: Introduction to Amazon Redshift connectors

This part contains the following chapters:

CHAPTER 1

Amazon Redshift connectors

This chapter includes the following topics:

Amazon Redshift connectors overview

You can use Amazon Redshift connectors to read data from and write data to Amazon Redshift. Use the connectors to create sources and targets that represent records in Amazon Redshift

When you use Amazon Redshift connectors to create and run a Data Integration task, the Secure Agent reads from and writes data to Amazon Redshift based on the taskflow and Amazon Redshift connection configuration. The Secure Agent connects reads data from and writes data to Amazon Simple Storage Service (Amazon S3) through a TCP/IP network. The Secure Agent uses the Amazon driver to communicate with Amazon Redshift.

You can move data from any data source to Amazon Redshift.

Use the following connectors to create connections and integrate data to and from Amazon Redshift:

Amazon Redshift V2 Connector

This is the recommended connector to connect to Amazon Redshift. Use Amazon Redshift V2 Connector to create a mass ingestion task, a mapping, or a mapping task.

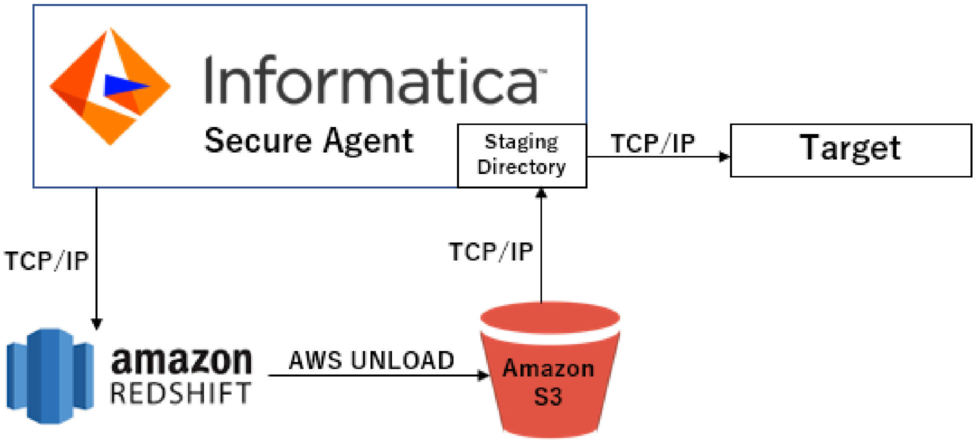
Amazon Redshift Connector

This is an older version of Amazon Redshift Connector. Informatica recommends that you use Amazon Redshift Connector to read or write data only when you want to use a synchronization task.

Secure Agent and Amazon Redshift integration

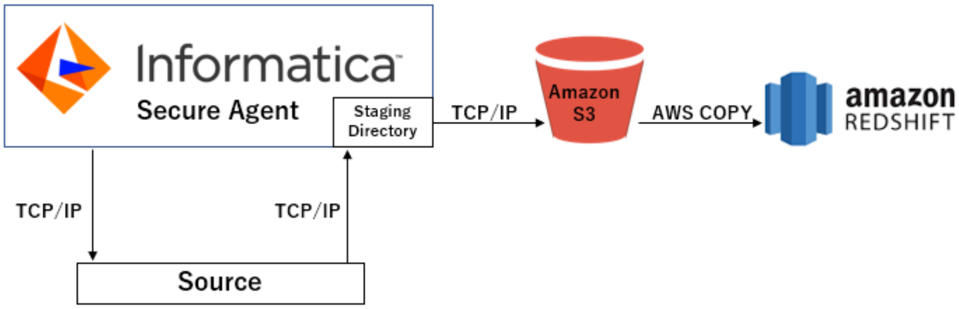
The Secure Agent uses the Amazon Redshift connection to connect to Amazon Redshift.

The following image shows how the Secure Agent connects to Amazon Redshift to read data:



The Secure Agent connects to Amazon Redshift and issues UNLOAD command to read data from Amazon Redshift to the Amazon Simple Storage Service (Amazon S3) bucket specified in the connection properties. The Secure Agent then stores data in a staging directory on the Secure Agent system over a TCP/IP network.

The following image shows how the Secure Agent connects to Amazon Redshift to write data:



The Secure Agent reads data from a staging directory on the Secure Agent system and writes data to Amazon Simple Storage Service (Amazon S3) through a TCP/IP network. The Secure Agent then issues COPY command to write data from Amazon S3 to the Amazon Redshift target table.

CHAPTER 2

Connector comparison

Based on your requirements to integrate or ingest data, you can use either Amazon Redshift Connector or Amazon Redshift V2 Connector to create data integration tasks.

The functionality to create integration tasks and configure read and write operations differ in both connectors. Informatica recommends to use Amazon Redshift V2 connector as the new features and enhancements are provided for Amazon Redshift V2 Connector.

Mapping functionality

The following table compares the mapping functionality supported by Amazon Redshift connectors:

Mapping functionality	Amazon Redshift Connector	Amazon Redshift V2 Connector
Hosted agent	Yes	Yes
Proxy server	Yes	Yes
Synchronization task	Yes	No
Mass ingestion task	No	Yes
Mapping task	Yes	Yes (Preferred)
Elastic mappings	No	Yes
Read from Oracle CDC Sources	No	Yes

Mapping functionality	Amazon Redshift Connector	Amazon Redshift V2 Connector
Pushdown optimization	Use an ODBC connection with **ODBC Subtype**=`Redshift` to enable source or full pushdown optimization between Amazon Redshift source and target.	- Use an Amazon S3 V3 source connection and an Amazon Redshift V2 target connection to enable full pushdown optimization between Amazon S3 source and Amazon Redshift target. - Use an ODBC connection with **ODBC Subtype**=`Redshift` to pushdown optimization between Amazon Redshift source and target. **Note:** Pushdown optimization does not apply to elastic mappings.
Lookup transformation	Cache, uncached, and connected	Cache, uncached, connected, and unconnected **Note:** Unconnected lookups do not apply to elastic mappings.

The following table lists the Amazon Redshift functionality supported by the Amazon Redshift connectors:

Amazon Redshift Functionality	Amazon Redshift Connector	Amazon Redshift V2 Connector
VPC endpoints	Yes	Yes
Redshift Spectrum	No	Yes

Source functionality

When you import an object from Amazon Redshift to read data, you can configure the advance source properties to determine the read operation behavior. For example, you can read data in an encrypted format or you can configure partitioning for optimal performance.

The following table lists the source functionality you can use when you read data from an Amazon Redshift source for Amazon Redshift and Amazon Redshift V2 connectors:

Feature	Amazon Redshift Connector	Amazon Redshift V2 Connector
Staging directory[1]	Yes	Yes[2]
Server-side encryption	Yes	Yes
Server-side encryption with KMS	Yes	Yes[2]
Client-side encryption	Yes	Yes[2]
Unload command	Yes	Yes
Partitioning	Yes	Yes[2]
Working with large tables	Yes	Yes
Octal Values as DELIMITER and QUOTE	Yes	Yes

Feature	Amazon Redshift Connector	Amazon Redshift V2 Connector
Success and error files	Yes	Yes
Import objects from different schema	No	Yes

[1] Does not apply to elastic mappings. However, you must specify a staging directory on Amazon Redshift in elastic configurations. For more information, see *Administrator*.

[2] Does not apply to elastic mappings.

Target functionality

When you import an object from Amazon Redshift to write data, you can configure the advance target properties to determine the write operation behavior. For example, you can write data in an encrypted format or you can retain Amazon S3 staging files after the write operation is complete.

The following table lists the target functionality you can use when you write data to an Amazon Redshift target:

Target functionality	Amazon Redshift mapping	Amazon Redshift V2 mapping
Staging directory[1]	Yes	Yes[2]
Server-side encryption	Yes	Yes
Client-side encryption	Yes	Yes[2]
Analyze target table	Yes	Yes
Retain staging files	Yes	Yes
Copy Command	Yes	Yes
Vacuum Tables	Yes	Yes
Recovery and restart processing	No	Yes[2]
Preserve record order on write	No	Yes[2]
Working with Large Tables	Yes	Yes
Octal Values as DELIMITER and QUOTE	Yes	Yes
Success and Error Files	Yes	Yes[2]

Target functionality	Amazon Redshift mapping	Amazon Redshift V2 mapping
Import objects from different schema	No	Yes

[1] Does not apply to elastic mappings. However, you must specify a staging directory on Amazon Redshift in elastic configurations. For more information, see *Administrator*.

[2] Does not apply to elastic mappings.

CHAPTER 3

Connector use cases

This chapter includes the following topics:

Synchronization task use case

You work for an e-commerce organization that stores sales order details in a MySQL database. Your organization needs to move the data from the MySQL database to an Amazon Redshift target.

Use Amazon Redshift Connector to create a synchronization task to write to an Amazon Redshift target.

Mapping and mapping task use case

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 to know the items ordered in a particular state and move data from the on-premise MySQL database to state-wise target tables in an affordable cloud-based environment.

Use Amazon Redshift V2 Connector to create a parameterized mapping to state-wise read purchase records from the MySQL database and write them to multiple Amazon Redshift targets to prepare an upcoming marketing campaign for all states.

Mapping task with Oracle CDC sources use case

our organization needs to replicate real-time changed data from a mission-critical Oracle production system to minimize intrusive, non-critical work, such as offline reporting or analytical operations system.

Use Amazon Redshift V2 Connector to capture changed data from the Oracle CDC source and write the changed data to an Amazon Redshift target table. Add the Oracle CDC sources in mappings, and then run the associated mapping tasks to write the changed data to the target.

Elastic mapping use case

You work for an organization that stores large amount of purchase order details, such as customer ID, item codes, and item quantity in Amazon S3. You need to port the data from Amazon S3 to another cloud-based environment to quickly analyze the purchase order details and to increase future revenues.

Use Amazon Redshift V2 Connector to create an elastic mapping that runs on the elastic cluster to achieve faster performance when you read all the purchase records from Amazon S3 and write the records to an Amazon Redshift target.

Mass ingestion task use case

You work for an organization that stores purchase order details data, such as customer ID, item codes, and item quantity in an on-premise flat file system. You need to move the files that contains the purchase order details data from an on-premise flat file system to a cloud-based environment for data analysis.

Use Amazon Redshift V2 Connector to create a mass ingestion task to move all the files that contains the purchase order details data from a flat file system to an Amazon Redshift target at once, instead of moving single row of data separately.

Part II: Data Integration with Amazon Redshift V2 Connector

This part contains the following chapters:

CHAPTER 4

Introduction to Amazon Redshift V2 Connector

This chapter includes the following topics:

Amazon Redshift V2 Connector overview

You can use Amazon Redshift V2 Connector to securely read data from and write data to Amazon Redshift. Amazon Redshift V2 sources and targets represent records in Amazon Redshift.

You can create an Amazon Redshift V2 connection and use the connection in mass ingestion tasks, mappings, and mapping tasks. You can also use the Amazon Redshift V2 connection in elastic mappings. For more information about elastic mappings, see *Administrator* and *Mappings*.

When you run an Amazon Redshift V2 mass ingestion task, mapping, or mapping task, the Secure Agent writes data to Amazon Redshift based on the workflow and Amazon Redshift V2 connection configuration. The Secure Agent 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 a source and simultaneously move data to Amazon Redshift clusters. The Secure Agent issues a copy command that copies data from Amazon S3 to the Amazon Redshift target table.

You can move data from any data source to Amazon Redshift. The Data Integration uses the Amazon driver to communicate with Amazon Redshift.

Create a mass ingestion task to transfer files from any source that mass ingestion task supports to an Amazon Redshift target. Create a mapping task to process data based on the data flow logic defined in a mapping or integration template.

You can create a mapping task to capture changed data from the Oracle CDC source and write the changed data to an Amazon Redshift target table.

Amazon Redshift V2 Connector supports Hosted Agent.

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. With Amazon Redshift Spectrum, you can directly run queries to read Amazon S3 data files without the need to load or transform the data.

You can run queries for the large amount of Amazon S3 data files without the need to scale the specified Amazon Redshift cluster.

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 must create an external table within an external schema that references a database in an external data catalog. You can create the external table for Avro, ORC, Parquet, RCFile, SequenceFIile, and Textfile file formats.

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 create an external database in Amazon Redshift. You can read data from a single external table, multiple external table, or from a standard Amazon Redshift table that is joined to the 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 create an external table, you must specify the Amazon S3 location from where you want to read the data. You can create the external tables by defining the structure of the Amazon S3 data files and registering the external tables in the external data catalog. Then, you can run queries or join the external tables.

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

Note: You can only read data from the Amazon Redshift Spectrum external table. You cannot insert or update data in the Amazon Redshift Spectrum external table.

When you create an external table using Athena or Glu data catalogs, ensure that you create the external tables using the data types that Amazon Redshift V2 Connector supports.

The following lists the data types that Amazon Redshift V2 Connector supports when you create an external table:

- Bigint (INT8)
- Boolean (BOOL)
- Char (CHARACTER)
- Date

  Note: Applicable when you create an external table for the ORC, Parquet, and Textfile file formats.
- Decimal (NUMERIC)
- Double Precision (FLOAT8)
- Integer (INT, INT4)
- Real (FLOAT4)
- Smallint (INT2)
- Timestamp
- Varchar (CHARACTER VARYING)

For more information on how to create an external table, see the AWS documentation.

Administration of Amazon Redshift V2 Connector

As a user, you can use Amazon Redshift V2 Connector after the organization administrator ensures that users have access to the Secure Agent directory that contains the success and error files. The directory path must be the same on each Secure Agent machine in the runtime environment. The organization administrator must also perform the following tasks:

- Get the Amazon Redshift JDBC URL.
- Manage Authentication. Use either of the following two methods:
  - Create an Access Key ID and Secret Access Key.
    Provide the values for access key ID and secret access key when you configure the Amazon Redshift V2 connection. For more information about creating an access key ID and secret access key, see the AWS documentation.
  - Configure AWS Identity and Access Management (IAM) Authentication to enhance security.
    If you use IAM authentication, do not provide access key ID and secret access key explicitly in the Amazon Redshift V2 connection. Instead, you must create an Redshift Role Amazon Resource Name (ARN), add the minimal Amazon S3 bucket policy to the Redshift Role ARN, and add the Redshift Role ARN to the Redshift cluster.

    Provide the Redshift Role ARN in the AWS_IAM_ROLE option in the UNLOAD and COPY commands when you create a task.

    If you specify both, access key ID and secret access key in the connection properties and AWS_IAM_ROLE in the UNLOAD and COPY commands, AWS_IAM_ROLE takes the precedence.

    You must add IAM EC2 role and IAM Redshift role to the customer master key when you use IAM authentication and server-side encryption using customer master key.

    Hosted Agent does not support IAM authentication. For more information about how to configure IAM authentication for Amazon Redshift V2 Connector, see “IAM authentication ” on page 24
- Configure Amazon Redshift for SSL if you want to support an SSL connection.

- Create a master symmetric key if you want to enable client-side encryption.
- Create an AWS Key Management Service (AWS KMS)-managed customer master key if you want to enable server-side encryption.
- Create minimal Amazon S3 bucket policy for Amazon Redshift V2 Connector.
- To access the data catalog and the data files in Amazon S3 by using Amazon Redshift Spectrum, ensure that the Amazon Redshift cluster has the required authorization.
- Configure a CDC source if you want to create a mapping to capture changed data from the CDC source, and then run the associated mapping tasks to write the changed data to an Amazon Redshift target.
  To create a mapping with a CDC source, ensure that you have the PowerExchangeClient and CDC licenses.
- To run elastic mappings successfully, ensure that the Redshift cluster and the elastic cluster reside in the same virtual private cloud (VPC).
- To use Amazon Resource Name (ARN) for cross-account access, ensure that you follow prerequisites described by AWS. For more information, see the Amazon documentation.

Configure Amazon Redshift for SSL

You can configure the Secure Agent to support an SSL connection to Amazon Redshift.

1. Download the Amazon Redshift certificate from the following location:
   https://s3.amazonaws.com/redshift-downloads/redshift-ssl-ca-cert.pem.
2. Run the following command to add the certificate file to the key store: `${JAVA_HOME}/bin/keytool -keystore {JAVA_HOME}/lib/security/cacerts -import -alias <string_value> -file <certificate_filepath>`.
3. In Administrator, select **Runtime Environments**.
4. Select the Secure Agent from the list of Secure Agents.
5. In the upper-right corner, click **Edit**.
6. In the **System Configuration Details** section, change the **Type** to **DTM**.
7. Click the **Edit Agent Configuration** icon next to **JVMOption1** and add the following command: `-Djavax.net.ssl.trustStore=<keystore_name>`.
8. Click the **Edit Agent Configuration** icon next to **JVMOption2** and add the following command:`-Djavax.net.ssl.trustStorePassword=<password>`.
9. Add the following parameter to the JDBC URL that you specify in the Amazon Redshift V2 connection properties: `ssl=true`. For example, `jdbc:redshift://mycluster.xyz789.us-west-2.redshift.amazonaws.com:5439/dev?ssl=true`.
10. Click **OK** to save your changes.

Create minimal Amazon S3 bucket policy

The minimal Amazon S3 bucket policy restricts user operations and user access to particular Amazon S3 buckets by assigning an AWS IAM policy to users. You can configure the AWS IAM policy through the AWS console.

In elastic mappings, you can use different AWS accounts within the same AWS region. Make sure that the Amazon S3 bucket policy confirms access to the AWS accounts used in elastic mappings.

You can use the following minimum required permissions to successfully read data from and write data to Amazon Redshift resources:

- PutObject
- GetObject
- DeleteObject
- ListBucket

Sample Policy:

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

Note: The **Test Connection** does not validate the IAM policy assigned to users. The Amazon S3 bucket name is available in the advanced properties for source and target.

IAM authentication

Optional. You can configure IAM authentication when the Secure Agent is installed on an Amazon Elastic Compute Cloud (EC2) system. Use IAM authentication for secure and controlled access to Amazon Redshift resources when you run mappings and mapping tasks.

Use IAM authentication when you want to run the mappings and mapping tasks on Secure agent installed on an EC2 system.

Perform the following steps to configure IAM authentication:

1. Create minimal Amazon S3 bucket policy. For more information, see "Create minimal Amazon S3 bucket policy" on page 23
2. Create roles.
    - Create the Amazon EC2 role. Associate the minimal Amazon S3 bucket policy while creating the EC2 role. The Amazon EC2 role is used when you create or launch an EC2 instance. For more information about creating the Amazon EC2 role, see the AWS documentation.
    - Create the Amazon Redshift Role ARN for secure access to Amazon Redshift resources. Associate the minimal Amazon S3 bucket policy while creating the Amazon Redshift role. For more information about creating the Amazon Redshift Role ARN, see the AWS documentation.

      Note: Use the same Amazon Redshift Role ARN in the UNLOAD and COPY commands.
3. Assign the Amazon EC2 role that you created in step #2 to the EC2 instance.
4. Assign the Amazon Redshift Role ARN to the Amazon Redshift cluster to successfully perform read and write operations using UNLOAD and COPY commands. For more information about adding the Amazon Redshift Role ARN to the Amazon Redshift cluster, see the AWS documentation.

5. Install Secure Agent on the EC2 instance.

Amazon Redshift Spectrum prerequisite tasks

To read data from an Amazon Redshift Spectrum external table, you must provide the required authorization to 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.

CHAPTER 5

Amazon Redshift V2 connections

This chapter includes the following topics:

Amazon Redshift V2 connections overview

Amazon Redshift V2 connection enables you to read data from or write data to Amazon Redshift. You can use Amazon Redshift V2 connections to specify sources or targets in mappings and mapping tasks. You can use Amazon Redshift V2 connections to specify targets in mass ingestion tasks.

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 V2 connection.

Create an Amazon Redshift V2 connection on the **Connections** page and associate it with a mapping, mapping task, or mass ingestion task. Define the source and target properties to read or write data to Amazon Redshift.

Note: If you enable both HTTP and SOCKS proxies, SOCKS proxy is used by default. If you want to use HTTP proxy instead of SOCKS proxy, set the value of the **DisableSocksProxy** property to true in the System property.

Amazon Redshift V2 connection properties

When you set up an Amazon Redshift V2 connection, you must configure the connection properties.

The following table describes the Amazon Redshift V2 connection properties:

Connection property	Description
Runtime Environment	Name of the runtime environment where you want to run the tasks.
Username	User name of the Amazon Redshift account.
Password	Password for the Amazon Redshift account.
AWS Access Key ID	Access key to access the Amazon S3 bucket.
AWS Secret Access Key	Secret access key to access the Amazon 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.
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. **Note:** Cross-account access is not applicable to elastic mappings. You must generate the customer master key ID for the same region where your Amazon S3 bucket resides. You can either specify the customer-generated customer master key ID or the default customer master key ID.

Connection property	Description
JDBC URL	The URL of the Amazon Redshift V2 connection. Enter the JDBC URL in the following format: `jdbc:redshift://<database_name><cluster_name>.<region_name>.redshift.amazonaws.com:<port_number>/<database_name>`
Cluster Region	Optional. The AWS cluster region in which the bucket you want to access resides. 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. If you specify a cluster region in both **Cluster Region** and **JDBC URL** connection properties, the Secure Agent ignores the cluster region that you specify in the **JDBC URL** connection property. To use the cluster region name that you specify in the **JDBC URL** connection property, select **None** as the cluster region in this property. Select one of the following cluster regions: - None - Asia Pacific(Mumbai) - Asia Pacific(Seoul) - Asia Pacific(Singapore) - Asia Pacific(Sydney) - Asia Pacific(Tokyo) - Asia Pacific(Hong Kong) - AWS GovCloud (US) - AWS GovCloud (US-East) - Canada(Central) - China(Bejing) - China(Ningxia) - EU(Ireland) - EU(Frankfurt) - EU(Paris) - EU(Stockholm) - South America(Sao Paulo) - Middle East(Bahrain) - US East(N. Virginia) - US East(Ohio) - US West(N. California) - US West(Oregon) Default is **None**. You can only read data from or write data to the cluster regions supported by AWS SDK used by the connector.

Configuring proxy settings

If your organization uses an outgoing proxy server to connect to the internet, the Secure Agent connects to Informatica Intelligent Cloud Services through the proxy server.
Contact your network administrator for the correct proxy settings.

Proxy settings do not apply to elastic mappings.

Configuring proxy settings on Windows

To configure the proxy server settings for the Secure Agent on a Windows machine, you can configure the proxy server settings through the Secure Agent or the JVM options of the Secure Agent.

Configuring proxy settings through Secure Agent Manager

To configure the proxy server settings through the Secure Agent Manager, perform the following steps:

1. Click **Start** > **All Programs** > **Informatica Cloud Secure Agent** > **Informatica Cloud Secure Agent** to launch the Secure Agent Manager.

   The Secure Agent Manager displays the Secure Agent status.
2. Click **Proxy** in the Secure Agent Manager page.
3. Click **Use a Proxy Server** to enter proxy server settings.
4. Configure the following proxy server details:

Field	Description
Proxy Host	Required. Host name of the outgoing proxy server that the Secure Agent uses.
Proxy Port	Required. Port number of the outgoing proxy server.

5. Click **OK**.

   The Secure Agent Manager restarts the Secure Agent to apply the settings.

Configuring the proxy settings through JVMOptions

1. Log in to Informatica Intelligent Cloud Services.
2. Open Administrator and select **Runtime Environments**.
3. Select the Secure Agent for which you want to configure a proxy server.
4. On the upper-right corner of the page, click **Edit**.
5. In the **System Configuration Details** section, select the **Type** as **DTM** for the Data Integration Service.
   - Add the following parameters in any **JVMOption** field and specify appropriate values for each parameter:

Parameter	Description
-Dhttps.proxyHost=	Host name of the outgoing HTTPS proxy server.
-Dhttps.proxyPort=	Port number of the outgoing HTTPS proxy server.

   For example,

```
JVMOption1=-Dhttps.proxyHost=<proxy_server_hostname>
```

```
JVMOption2=-Dhttps.proxyPort=8081
```

6. Click **Save**.

   The Secure Agent restarts to apply the settings.

Configuring proxy settings on Linux

The Secure Agent installer configures the proxy server settings for the Secure Agent based on settings configured in the browser. You can update the proxy server settings defined for the Secure Agent from the command line.

To configure the proxy server settings for the Secure Agent on a Linux machine, use a shell command that updates the `proxy.ini` file. Contact the network administrator to determine the proxy settings.

1. Navigate to the following directory:

   ```
   <Secure Agent installation directory>/apps/agentcore
   ```

2. Update the `proxy.ini` file.
   - To update the `proxy.ini` file for an unauthenticated proxy, enter the following command:

     ```
     consoleAgentManager.bat configureProxy <proxy host> <proxy port>
     ```

3. Restart the Secure Agent.

CHAPTER 6

Amazon Redshift V2 sources and targets

This chapter includes the following topics:

Amazon Redshift V2 sources

You can use an Amazon Redshift V2 object as a source in a mapping.

When you configure the advanced source properties, configure properties specific to Amazon Redshift V2. You can encrypt data, retain the staging files on Amazon S3, and securely unload the results of a query to files on Amazon Redshift.

The following table lists the Amazon Redshift V2 source features that you can use in mappings:

Feature	Mapping	Elastic Mapping
Staging directory	Yes	No However, you must specify a staging directory on Amazon Redshift in elastic configurations. For more information, see *Administrator*.
Server-side encryption	Yes	Yes
Server-side encryption with KMS	Yes	No
Client-side encryption	Yes	No
Unload command	Yes	Yes
Partitioning	Yes	No
Working with large tables	Yes	Yes
Octal Values as DELIMITER and QUOTE	Yes	Yes
Success and error files	Yes	Yes

Feature	Mapping	Elastic Mapping
Import objects from different schema	Yes	Yes
Custom query	Yes	No

Amazon Redshift staging directory for Amazon Redshift V2 sources

The Secure Agent creates a staging file in the directory that you specify in the source properties. The Secure Agent reads the data from Amazon Redshift V2 source and writes the data to the staging directory before writing the data to Amazon S3.

The Secure Agent deletes the staged files from the staging directory after writing 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 Secure Agent.

Data encryption in Amazon Redshift V2 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 under the Amazon Redshift V2 advanced source properties on the **Schedule** page. 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 the **SSE-KMS** 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 or ARN 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: Amazon Redshift V2 Connector does not support the server-side encryption with the master symmetric key and client-side encryption with the customer master key.

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

Note: The NULL Unload command option does not apply to elastic mappings.

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

If you run the Unload command as a pre-SQL or post-SQL command, specify the `ALLOWOVERWRITE` option to overwrite the existing objects.

By default, the UNLOAD property field is empty.

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. You can delimit the data with a particular character or load data to multiple files in parallel.

To add options to the Unload command, use the **Unload Options** option.

You can set the following options:

DELIMITER

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

ESCAPE

You can add an escape character for CHAR and VARCHAR columns in delimited unload files before the delimiter character is specified for the unloaded data. By default, the escape option is **ON**. To disable the escape option, specify **OFF** as the value of the escape option. For example, `ESCAPE = OFF`.

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`.

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.

NULL

You can use 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: The NULL Unload command option does not apply to elastic mappings.

AWS_IAM_ROLE

Specify the Amazon Redshift Role Resource Name (ARN) to run the mapping on Secure Agent 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`

Source partitioning

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

You can define the partition type as key range partitioning. Configure key range partitioning to partition Amazon Redshift data based on the value of a fields or set of fields. With key range partitioning, the Secure Agent distributes rows of source data based the fields that you define as partition keys. The Secure Agent compares the field value to the range values for each partition and sends rows to the appropriate partition.

Use key range partitioning for columns that have an even distribution of data values. Otherwise, the partitions might have unequal size. For example, a column might have 10 rows between key values 1 and 1000 and the column might have 999 rows between key values 1001 and 2000.

With key range partitioning, a query for one partition might return rows sooner than another partition. Or, one partition can return rows while the other partitions are not returning rows. This situation occurs when the rows in the table are in a similar order as the key range. One query might be reading and returning rows while the other queries are reading and filtering the same rows.

Note: You can configure a partition key only of the Integer and String data types.

When you configure more than two partitions in a mapping, the Secure Agent ignore the values that you specify in the start range for the first partition and end range for the last partition. The Secure Agent uses the start range value for the first partition as less than 10 and the end range value for the last partition as greater than the value you specify for the last partition.

For example, if you configure three partitions in a mapping and specify the start range value for the first partition as 5 and the end range value for the last partition as 90, the mapping runs successfully. However, the Secure Agent ignores the values that you specify and uses the start range value for the first partition as less than 10 and the end range value for the last partition as greater than 90.

Amazon Redshift V2 targets

You can use an Amazon Redshift V2 object as a target in a mapping, mappingtask, or mass ingestion task. You can also create an Amazon Redshift V2 target based on the input source.

When you configure the advanced target properties, configure properties specific to Amazon Redshift V2. You can encrypt data, update statistical metadata of the database tables to improve the efficiency of queries, load data into Amazon Redshift from flat files in an Amazon S3 bucket, and use vacuum tables to recover disk space and sort rows in tables.

Note: If the distribution key column in a target table contains null values and you configure a task with an upsert operation for the same target table, the task 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. You can use the Pre-SQL and Post-SQL properties to remove and then add the distribution key column in the target table.

The following table lists the Amazon Redshift V2 target features that you use in mappings:

Feature	Mapping	Elastic Mapping
Staging directory	Yes	No However, you must specify a staging directory on Amazon Redshift in elastic configurations. For more information, see *Administrator*.
Server-side encryption	Yes	Yes
Client-side encryption	Yes	No
Analyze target table	Yes	Yes
Retain staging files	Yes	Yes
Copy Command	Yes	Yes
Vacuum Tables	Yes	Yes
Recovery and restart processing	Yes	No
Preserve record order on write	Yes	No
Working with Large Tables	Yes	Yes
Octal Values as DELIMITER and QUOTE	Yes	Yes
Success and Error Files	Yes	No
Import objects from different schema	Yes	Yes

Amazon Redshift staging directory for Amazon Redshift V2 targets

The Secure Agent creates a staging file in the directory that you specify in the target properties. The Secure Agent writes the data to the staging directory before writing the data to Amazon Redshift.

The Secure Agent deletes the staged files from the staging directory after writing 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 Secure Agent.

The Secure Agent creates subdirectories in the staging directory. Subdirectories use the following naming convention: `<staging directory>/infaRedShiftStaging<MMddHHmmssSSS+xyz>`

Data encryption in Amazon Redshift V2 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.

Server-side encryption for Amazon Redshift V2 targets

If you want Amazon Redshift to encrypt data while uploading and staging the `.csv` files to Amazon S3, you must enable server-side encryption.

To enable server-side encryption, select **S3 Server Side Encryption** in the advanced target properties and specify the **Customer Master key ID** in the connection properties.

You can configure the customer master key ID generated by AWS Key Management Service (AWS KMS) in the connection properties for server-side encryption. You must add IAM EC2 role and IAM Redshift role to the customer master key when you use IAM authentication and server-side encryption using customer master key.

If you select the server-side encryption in the advanced target properties and do not specify the customer master key ID in the connection properties, Amazon S3-managed encryption keys are used to encrypt data.

Client-side encryption for Amazon Redshift V2 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 V2 targets, the Secure Agent 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.

Note: If you enable both server-side and client-side encryption for an Amazon Redshift V2 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 and select **S3 Client Side Encryption** in the advanced target properties.

The Secure Agent encrypts the data by using the master symmetric key. The master symmetric key is a 256-bit AES encryption key in the Base64 format. Amazon Redshift V2 Connector 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.

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>
```

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.

To add options to the Copy command, use the **CopyOptions Property File** option.

You can set the following options:

DELIMITER

A single ASCII character to separate fields in the input file. You can use characters such as pipe (|), tilde (~), or a tab (\t). The delimiter must not be a part of the data. Default is \036, the octal representation of the non-printable character and record separator.

ACCEPTINVCHARS

Loads data into VARCHAR columns even if the data contains UTF-8 characters that are not valid. When you specify ACCEPTINCHARS, the Secure Agent 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 Secure Agent 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. Default is question mark (?).

QUOTE

Specifies the quote character to use with comma separated values. Default is \037, the octal representation of the non-printable character, 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. Use the COMPUPDATE option in an insert operation 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.

Default is OFF.

TRUNCATECOLUMN

Truncates the data of the VARCHAR and CHAR data types column before writing the data to the target. If the size of the data that you want to write to the target is larger than size of the target column, the Secure Agent truncates the data before writing data to the target column.

By default, the TRUNCATECOLUMNS option is OFF. To enable the TRUNCATECOLUMNS option, specify ON as the value of the TRUNCATECOLUMNS option. For example, TRUNCATECOLUMNS=ON.

AWS_IAM_ROLE

Specify the Amazon Redshift Role Resource Name (ARN) to run the task on Secure Agent 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`

Analyze target table

To optimize query performance, you can configure a task 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.

Retain staging files

You can retain staging files on Amazon S3 after the Secure Agent 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 create a target connection, you can configure a file prefix or directory prefix to save the staging files. After you provide the prefixes, the Secure Agent 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 Secure Agent 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 Secure Agent creates the following directories and files:
  - `<year>_<month>_<date>_<timestamp_inLong><3 digit of random number>00<ProcessID><PartitionId>`
  - `backup_file.batch_<batch_number>.csv.<file_number>.<encryption_if_applicable>`
- Do not provide a prefix. The Secure Agent does not save the staging files.

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.

Reindex

Analyzes the distribution of the values in the interleaved sort key columns to configure the entire Vacuum table operations for a better performance.

Recovery and restart processing

When you run a mapping task to capture changed data from a CDC source and write the changed data to an Amazon Redshift target table, Amazon Redshift V2 Connector supports recovery and restart processing.

If a mapping task fails or is stopped before completing the task, the Secure Agent uses the recovery information stored in the `infa_recovery_table` table on the target system to resume the extraction of changed data from the point of interruption. This functionality prevents changed data loss and inconsistencies between the source and target.

To enable recovery and restart processing, set the **Recovery Strategy** advanced session property to **Resume from last checkpoint** on the **Schedule** page when you create or edit a mapping task. With this setting, the mapping task can resume processing changed data from the point of interruption.

In special situations, you can specify a restart point for a mapping task. Typically, the first time you start a mapping task, you specify a restart point that corresponds to the target materialization time so that no change records are skipped. The default restart point is the end of log (EOL), which is the current point of CDC processing in the log. You can specify a restart point that corresponds to the extraction processing starting from the earliest available record in the log or from a specific date and time. When you use a time-based restart point, extraction processing starts in the log that contains the first unit-of-work (UOW) that has an end time later than the restart time.

When you specify a restart point, consider the following points:

- The restart point applies to all sources in the mapping that is associated with the mapping task.
- If you set a restart point that is too early, it might correspond to a expired log file. In this case, the value of the restart point is considered as the earliest available record in the available log files.
- If you set a restart point that is later than the latest record in the log files, an error message is issued.

Note: Restart information is associated with a mapping task, a specific source and target combination. If you change the source object in a mapping, you must either create a new mapping task for the mapping or increment the restart revision number for the existing mapping task. To increment the restart revision number, navigate to the **CDC Runtime** page for the mapping task, open the **Select Restart Point** dialog box, and click **OK**. If you do not take one of these actions, the mapping task will fail the next time you run it.

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 target 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 Secure Agent 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

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

Success and error files

The Secure Agent generates success and error files after you run a mapping. Success and error files are `.csv` files that contain row-level details.

The Secure Agent generates a success file after you run a mapping. The success file contains an entry for each record that successfully writes into Amazon Redshift. Each entry contains the values that are written for all the fields of the record. Use this file to understand the data that the Secure Agent writes to the Amazon S3 bucket and then to the Amazon Redshift target.

The error file contains an entry for each data error. Each entry in the file contains the values for all fields of the record and the error message. Use the error file to understand why the Secure Agent does not write data to the Amazon Redshift target.

The Secure Agent 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 mapping properties for success files:

- You must provide the file path where you want the Secure Agent to generate the success rows file.
- The success rows file uses the following naming convention: `<timestamp>success`

Consider the following guidelines when you configure the mapping properties for error files:

- You must provide the file path where you want the Secure Agent to generate the error rows file.
- The success rows file uses the following naming convention: `<timestamp>error`

  Note: The insert and upsert tasks error rows file follows the same naming convention.
- When you define a error file directory, you can use the variable `$PMBadFileDir`. When you use the `$PMBadFileDir` variable, the application writes the file to the following Secure Agent directory: `<Secure Agent installation directory>/apps/Data_Integration_Server/data/error`.

CHAPTER 7

Mappings and mapping tasks with Amazon Redshift V2 Connector

This chapter includes the following topics:

Amazon Redshift V2 objects in mappings

When you create a mapping, you can configure a Source or Target transformation to represent an Amazon Redshift V2 object.

Create a mapping task to process data based on the data flow logic defined in a mapping or integration template. You can also create a mapping task to capture changed data from the Oracle CDC source and write the changed data to an Amazon Redshift target table.

Note: When you select an Amazon Redshift V2 object that contains a boolean data type and preview the data, the Secure Agent truncates the value of the boolean data type and displays only the first letter of the boolean value.

When you create a mapping if you use a simple filter, you must specify the filter condition in the `YYYY-MM-DD HH24:MI:SS.MS` format. If you use an advanced filter, you must specify the filter condition in the `date_time_fix.f_timestamp < to_date('2012-05-24 09:13:57','YYYY-MM-DD HH24:MI:SS.MS')` format.

Amazon Redshift V2 sources in mappings

In a mapping, you can configure a Source transformation to represent an Amazon Redshift V2 source.

The following table describes the Amazon Redshift V2 source properties that you can configure in a Source transformation:

Property	Description
Connection	Name of the source connection. Select a source connection, or click **New Parameter** to define a new parameter for the source connection.
Source type	Type of the source object. Select any of the following source object: - Single Object - Multiple Objects - Query - Parameter **Note:** You cannot override source query object and multiple objects at runtime using parameter files in a mapping. Multiple objects do not support advanced relationships.
Object	Name of the source object. You can select single or multiple source objects.
Parameter	Select an existing parameter for the source object or click **New Parameter** to define a new parameter for the source object. The **Parameter** property appears only if you select Parameter as the source type. If you want to overwrite the parameter at runtime, select the **Overwrite Parameter** option.

The following table describes the Amazon Redshift V2 advanced source properties that you can configure in a Source transformation:

Property	Description
S3 Bucket Name	Amazon S3 bucket name for staging the data. 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.
Enable Compression	Compresses the staging files into the Amazon S3 staging directory. The task performance improves when the Secure Agent compresses the staging files. Default is selected.
Staging Directory Location	Location of the local staging directory. When you run a task in Secure Agent runtime environment, specify a directory path that is available on the corresponding Secure Agent machine in the runtime environment. Specify the directory path in the following manner: `<staging directory>` For example, `C:\Temp`. Ensure that you have the write permissions on the directory. Does not apply to elastic mappings.

Property	Description
Unload Options	Unload command options. Add options to the Unload command to extract data from Amazon Redshift and create staging files on Amazon S3. Provide an Amazon Redshift Role Amazon Resource Name (ARN). You can add the following options: - DELIMITER - ESCAPE - PARALLEL - NULL - AWS_IAM_ROLE - REGION For example: DELIMITER = \036;ESCAPE = OFF;NULL=text;PARALLEL = ON;AWS_IAM_ROLE=arn;aws;iam;;<account ID>;role/<role-name>;REGION = ap-south-1 You cannot use the NULL option in an elastic mapping. Specify a directory on the machine that hosts the Secure Agent. **Note:** If you do not add the options to the Unload command manually, the Secure Agent uses the default values.
Treat NULL Value as NULL	Retains the null values when you read data from Amazon Redshift.
Encryption Type	Encrypts the data in the Amazon S3 staging directory. You can select the following encryption types: - None - SSE-S3 - SSE-KMS - CSE-SMK You can only use **SSE-S3** encryption in an elastic mapping. Default is None.
Download S3 Files in Multiple Parts	Downloads large Amazon S3 objects in multiple parts. 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. Default is 5 MB. Does not apply to elastic mappings.
Multipart Download Threshold Size	The maximum threshold size to download an Amazon S3 object in multiple parts. Default is 5 MB. Does not apply to elastic mappings.
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.
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.
Select Distinct	Selects unique values. The Secure Agent includes a `SELECT DISTINCT` statement if you choose this option. Amazon Redshift ignores trailing spaces. Therefore, the Secure Agent might extract fewer rows than expected. **Note:** If you select the source type as query or use the **SQL Query** property and select the **Select Distinct** option, the Secure Agent ignores the **Select Distinct** option.

Property	Description
SQL Query	Overrides the default SQL query. Enclose column names in double quotes. The SQL query is case sensitive. Specify an SQL statement supported by the Amazon Redshift database. When you specify the columns in the SQL query, ensure that the column name in the query matches the source column name in the mapping.
Tracing Level	Use the verbose tracing level to get the amount of detail that appears in the log for the Source transformation.

Configuring key range partition

Configure key range partition to partition Amazon Redshift data based on field values.

1. In **Source Properties**, click the **Partitions** tab.
2. Select the required **Partition Key** from the list.
3. Click **Add New key Range** to add partitions.
4. Specify the **Start range** and **End range**.

Amazon Redshift V2 targets in mappings

To write data to Amazon Redshift, configure an Amazon Redshift V2 object as the target in a mapping.

When you enable the source partition, the Secure Agent uses the pass-through partitioning to write data to Amazon Redshift to optimize the mapping performance at run time. Specify the name and description of the Amazon Redshift V2 target. Configure the target and advanced properties for the target object

The following table describes the target properties that you can configure in a Target transformation:

Property	Description
Connection	Name of the target connection. Select a target connection, or click **New Parameter** to define a new parameter for the target connection.
Target Type	Type of the target object. Select Single Object or Parameter.
Object	Name of the target object. You can select a single target object.
Parameter	Select an existing parameter for the target object or click **New Parameter** to define a new parameter for the target object. The **Parameter** property appears only if you select Parameter as the target type. If you want to overwrite the parameter at runtime, select the **Overwrite Parameter** option.
Operation	Type of the target operation. Select Insert, Update, Upsert, Delete, Data Driven. **Note:** Select **Data Driven** if you want to create a mapping to capture changed data from a CDC source.

Property	Description
Data Driven Condition	Enables you to define expressions that flag rows for an insert, update, delete, or reject operation. **Note:** Appears only when you select **Data Driven** as the operation type. However, you must leave the field empty as the rows in the CDC source tables are already marked with the operation types.
Update Columns	Select columns you want to use as a logical primary key for performing update, upsert, and delete operations on the target. **Note:** This field is not required if the target table already has a primary key.
Create Target	Creates a new target. When you create a new target, enter a value of the following fields: - **Name**: Enter a name for the target object. - **Path**: Provide a schema name and create a target table within the schema. By default, the field is empty. The Secure Agent converts the target table names that you specify in the **Create Target** field into lower case. When you create a target, you can view and edit the metadata of the target object in the **Target Fields** tab. You can edit the data type, precision and define primary key of the columns in the target objects. To edit the metadata, click **Options** > **Edit Metadata** in the **Target Fields** tab. **Note:** When you create a target if the source table contains column of Text data type, the Secure Agent displays the following error message: `Unsupported datatype - 'text' for column 'LONGTXTAREA__C'` You must edit the data type of the source column in the Source transformation. **Note:** You cannot parameterize the target at runtime.

The following table describes the Amazon Redshift V2 advanced target properties:

Property	Description
S3 Bucket Name	Amazon S3 bucket name for writing the files to Amazon Redshift target. 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 Copy command options.
Enable Compression	Compresses the staging files before writing the files to Amazon Redshift. The task performance improves when the Secure Agent compresses the staged files. Default is selected.
Staging Directory Location	Location of the local staging directory. When you run a task in Secure Agent runtime environment, specify a directory path that is available on the corresponding Secure Agent machine in the runtime environment. Specify the directory path in the following manner: `<staging directory>` For example, `C:\Temp`. Ensure that you have the write permissions on the directory. Does not apply to elastic mappings.
Batch Size	Minimum number of rows in a batch. Enter a number greater than 0. Default is 2000000. Does not apply to elastic mappings.

Property	Description
Max Errors per Upload Batch for INSERT	Number of error rows that causes an upload insert batch to fail. Enter a positive integer. Default is 1. If the number of errors is equal to or greater than the property value, the Secure Agent 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.
Require Null Value For Char and Varchar	Replaces the string value with NULL when you write data to Amazon Redshift columns of Char and Varchar data types. Default is an empty string. **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.
WaitTime In Seconds For S3 File Consistency	Number of seconds to wait for the Secure Agent to make the staged files consistent with the list of files available on Amazon S3. Default is 0. Does not apply to elastic mappings.
Copy Options	Copy command options. Add options to the Copy command to write data from Amazon S3 to the Amazon Redshift target when the default delimiter comma (,) or double-quote (") is used in the data. Provide the Amazon Redshift Role Amazon Resource Name (ARN). You can add the following options: - DELIMITER - ACCEPTINVCHARS - QUOTE - COMPUPDATE - AWS_IAM_ROLE - REGION For example: `DELIMITER = \036;ACCEPTINVCHARS = #;QUOTE = \037 COMPUPDATE = ON;AWS_IAM_ROLE=arn:aws:iam::<account ID>:role/<role-name>;REGION = ap-south-1` Specify a directory on the machine that hosts the Secure Agent. **Note:** If you do not add the options to the Copy command manually, the Secure Agent uses the default values.
S3 Server Side Encryption	Indicates that Amazon S3 encrypts data during upload. Provide a customer master key ID in the connection property to enable this property. Default is not selected.
S3 Client Side Encryption	Indicates that the Secure Agent encrypts data using a private key. Provide a master symmetric key ID in the connection property to enable this property. If you enable both server-side and client-side encryptions, the Secure Agent ignores the server-side encryption. Does not apply to elastic mappings.
Analyze Target Table	Runs an ANALYZE command on the target table. The query planner on Amazon Redshift updates the statistical metadata to build and choose optimal plans to improve the efficiency of queries.

Property	Description
Vacuum Target Table	Recovers disk space and sorts the row in a specified table or all tables in the database. You can select the following recovery options: - None - Full - Sort Only - Delete Only - Reindex Default is None.
Prefix to retain staging files on S3	Retains staging files on Amazon S3. 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`.
Success File Directory	Directory for the Amazon Redshift success file. Specify a directory on the machine that hosts the Secure Agent. Does not apply to elastic mappings.
Error File Directory	Directory for the Amazon Redshift error file. Specify a directory on the machine that hosts the Secure Agent. Does not apply to elastic mappings.
Treat Source Rows As	Overrides the default target operation. Default is **INSERT**. Select one of the following override options: **NONE** By default, none is enabled. The Secure Agent considers the task operation that you select in the **Operation** target property. **INSERT** Performs insert operation. If enabled, the Secure Agent inserts all rows flagged for insert. If disabled, the Secure Agent rejects the rows flagged for insert. **DELETE** Performs delete operation. If enabled, the Secure Agent deletes all rows flagged for delete. If disabled, the Secure Agent rejects all rows flagged for delete. **UPDATE and UPSERT** 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: - Update as Update: The Secure Agent updates all rows as updates. - Update else Insert: The Secure Agent updates existing rows and inserts other rows as if marked for insert. Amazon Redshift V2 Connector does not support the Upsert operation in the Upgrade Strategy transformation. To use an Update Strategy transformation to write data to an Amazon Redshift target, you must select **Treat Source Rows As** as **None**. By default, the Secure Agent performs the task operation based on the value that you specify in the **Operation** target property. However, if you specify an option in the **Treat Source Rows As** property, the Secure Agent ignores the value of that you specify in the **Operation** target property or in the Update Strategy transformation.

Property	Description
TransferManager Thread Pool Size	Number of threads to write data in parallel. Default is 10. Does not apply to elastic mappings.
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.
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.
Preserve record order on write	Retains the order of the records when you read data from a CDC source and write data to an Amazon Redshift target. 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. Does not apply to elastic mappings.
Minimum Upload Part Size	Minimum size of the Amazon Redshift object to upload an object. Default is 5 MB. Does not apply to elastic mappings.
Number of files per batch	Calculates the number of the staging files per batch. If you do not provide the number of files, Amazon Redshift V2 Connector calculates the number of the staging files. Does not apply to elastic mappings.
Schema Name	Overrides the default schema name.
Target table name	Overwrites the default target table name.
Recovery Schema Name	Schema that contains recovery information stored in the `infa_recovery_table` table on the target system to resume the extraction of the changed data from the last checkpoint. Does not apply to elastic mappings.
Forward Rejected Rows	This property is not applicable for Amazon Redshift V2 Connector.

Amazon Redshift lookup transformation

Use a Lookup transformation to retrieve data based on a specified lookup condition.

Use an Amazon Redshift V2 Lookup transformation to look up data in an Amazon Redshift object. For example, the source table includes the customer code, but you want to include the customer name in the target table to make summary data easy to read. You can use the Amazon Redshift V2 Lookup transformation to look up the customer name in another Amazon Redshift object.

You can add the following lookups to an Amazon Redshift object when you configure field mappings in a mapping task:

- Cached

- Uncached
- Connected
- Unconnected

Use the JDBC URL specified in the connection properties to create lookups. You can not use unconnected lookups in elastic mappings.

Note: A Lookup transformation in an elastic mapping, with the **On Multiple Matches** property configured as `Report Error`, runs successfully without displaying an error message.

For more information about the Lookup transformation, see *Transformations*.

Amazon Redshift V2 objects in mapping tasks

When you configure a mapping task, you can configure advanced properties for Amazon Redshift V2 targets.

Amazon Redshift V2 sources in mapping tasks

For Amazon Redshift V2 source connections used in template-based mapping tasks, you can configure advanced properties in the Sources page.

You can configure the following source advanced properties:

Property	Description
S3 Bucket Name	Amazon S3 bucket name for staging the data. 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.
Enable Compression	Compresses the staging files into the Amazon S3 staging directory. The task performance improves when the Secure Agent compresses the staging files. Default is selected.
Staging Directory Location	Location of the local staging directory. When you run a task in Secure Agent runtime environment, specify a directory path that is available on the corresponding Secure Agent machine in the runtime environment. Specify the directory path in the following manner: `<staging directory>` For example, `C:\Temp`. Ensure that you have the write permissions on the directory. Does not apply to elastic mappings.

Property	Description
Unload Options	Unload command options. Add options to the Unload command to extract data from Amazon Redshift and create staging files on Amazon S3. Provide an Amazon Redshift Role Amazon Resource Name (ARN). You can add the following options: - DELIMITER - ESCAPE - PARALLEL - NULL - AWS_IAM_ROLE - REGION For example: DELIMITER = \036;ESCAPE = OFF;NULL=text;PARALLEL = ON;AWS_IAM_ROLE=arn;aws;iam;;<account ID>;role/<role-name>;REGION = ap-south-1 You cannot use the NULL option in an elastic mapping. Specify a directory on the machine that hosts the Secure Agent. **Note:** If you do not add the options to the Unload command manually, the Secure Agent uses the default values.
Treat NULL Value as NULL	Retains the null values when you read data from Amazon Redshift.
Encryption Type	Encrypts the data in the Amazon S3 staging directory. You can select the following encryption types: - None - SSE-S3 - SSE-KMS - CSE-SMK You can only use **SSE-S3** encryption in an elastic mapping. Default is None. For more information about the encryption types, see "Data encryption in Amazon Redshift V2 sources" on page 32
Download S3 Files in Multiple Parts	Downloads large Amazon S3 objects in multiple parts. 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. Default is 5 MB. Does not apply to elastic mappings.
Multipart Download Threshold Size	The maximum threshold size to download an Amazon S3 object in multiple parts. Default is 5 MB. Does not apply to elastic mappings.
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.
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.
Select Distinct	Selects unique values. The Secure Agent includes a `SELECT DISTINCT` statement if you choose this option. Amazon Redshift ignores trailing spaces. Therefore, the Secure Agent might extract fewer rows than expected. **Note:** If you select the source type as query or use the **SQL Query** property and select the **Select Distinct** option, the Secure Agent ignores the **Select Distinct** option.

Property	Description
SQL Query	Overrides the default SQL query. Enclose column names in double quotes. The SQL query is case sensitive. Specify an SQL statement supported by the Amazon Redshift database. When you specify the columns in the SQL query, ensure that the column name in the query matches the source column name in the mapping.
Tracing Level	Use the verbose tracing level to get the amount of detail that appears in the log for the Source transformation.

Amazon Redshift V2 targets in mapping tasks

For Amazon Redshift V2 target connections used in mapping tasks, you can configure advanced target properties in the **Targets** page of the Mapping Task wizard.

You can configure the following Amazon Redshift V2 advanced target properties:

Property	Description
S3 Bucket Name	Amazon S3 bucket name for writing the files to Amazon Redshift target. 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 Copy command options.
Enable Compression	Compresses the staging files before writing the files to Amazon Redshift. The task performance improves when the Secure Agent compresses the staged files. Default is selected.
Staging Directory Location	Location of the local staging directory. When you run a task in Secure Agent runtime environment, specify a directory path that is available on the corresponding Secure Agent machine in the runtime environment. Specify the directory path in the following manner: `<staging directory>` For example, `C:\Temp`. Ensure that you have the write permissions on the directory. Does not apply to elastic mappings.
Batch Size	Minimum number of rows in a batch. Enter a number greater than 0. Default is 2000000. Does not apply to elastic mappings.
Max Errors per Upload Batch for INSERT	Number of error rows that causes an upload insert batch to fail. Enter a positive integer. Default is 1. If the number of errors is equal to or greater than the property value, the Secure Agent 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.

Property	Description
Require Null Value For Char and Varchar	Replaces the string value with NULL when you write data to Amazon Redshift columns of Char and Varchar data types. Default is an empty string. **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.
WaitTime In Seconds For S3 File Consistency	Number of seconds to wait for the Secure Agent to make the staged files consistent with the list of files available on Amazon S3. Default is 0. Does not apply to elastic mappings.
Copy Options	Copy command options. Add options to the Copy command to write data from Amazon S3 to the Amazon Redshift target when the default delimiter comma (,) or double-quote (") is used in the data. Provide the Amazon Redshift Role Amazon Resource Name (ARN). You can add the following options: - DELIMITER - ACCEPTINVCHARS - QUOTE - COMPUPDATE - AWS_IAM_ROLE - REGION For example: `DELIMITER = \036;ACCEPTINVCHARS = #;QUOTE = \037 COMPUPDATE = ON;AWS_IAM_ROLE=arn:aws:iam::<account ID>:role/<role-name>;REGION = ap-south-1` Specify a directory on the machine that hosts the Secure Agent. **Note:** If you do not add the options to the Copy command manually, the Secure Agent uses the default values.
S3 Server Side Encryption	Indicates that Amazon S3 encrypts data during upload. Provide a customer master key ID in the connection property to enable this property. Default is not selected.
S3 Client Side Encryption	Indicates that the Secure Agent encrypts data using a private key. Provide a master symmetric key ID in the connection property to enable this property. If you enable both server-side and client-side encryptions, the Secure Agent ignores the server-side encryption. Does not apply to elastic mappings.
Analyze Target Table	Runs an ANALYZE command on the target table. The query planner on Amazon Redshift updates the statistical metadata to build and choose optimal plans to improve the efficiency of queries.

Property	Description
Vacuum Target Table	Recovers disk space and sorts the row in a specified table or all tables in the database. You can select the following recovery options: - None - Full - Sort Only - Delete Only - Reindex Default is None. For more information about the vacuum tables, see "Vacuum tables" on page 39.
Prefix to retain staging files on S3	Retains staging files on Amazon S3. 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`.
Success File Directory	Directory for the Amazon Redshift success file. Specify a directory on the machine that hosts the Secure Agent. Does not apply to elastic mappings.
Error File Directory	Directory for the Amazon Redshift error file. Specify a directory on the machine that hosts the Secure Agent. Does not apply to elastic mappings.
Treat Source Rows As	Overrides the default target operation. Default is **INSERT**. Select one of the following override options: **NONE** By default, none is enabled. The Secure Agent considers the task operation that you select in the **Operation** target property. **INSERT** Performs insert operation. If enabled, the Secure Agent inserts all rows flagged for insert. If disabled, the Secure Agent rejects the rows flagged for insert. **DELETE** Performs delete operation. If enabled, the Secure Agent deletes all rows flagged for delete. If disabled, the Secure Agent rejects all rows flagged for delete. **UPDATE and UPSERT** 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: - Update as Update: The Secure Agent updates all rows as updates. - Update else Insert: The Secure Agent updates existing rows and inserts other rows as if marked for insert. Amazon Redshift V2 Connector does not support the Upsert operation in the Upgrade Strategy transformation. To use an Update Strategy transformation to write data to an Amazon Redshift target, you must select **Treat Source Rows As** as **None**. By default, the Secure Agent performs the task operation based on the value that you specify in the **Operation** target property. However, if you specify an option in the **Treat Source Rows As** property, the Secure Agent ignores the value of that you specify in the **Operation** target property or in the Update Strategy transformation.

Property	Description
TransferManager Thread Pool Size	Number of threads to write data in parallel. Default is 10. Does not apply to elastic mappings.
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.
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.
Preserve record order on write	Retains the order of the records when you read data from a CDC source and write data to an Amazon Redshift target. 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. Does not apply to elastic mappings.
Minimum Upload Part Size	Minimum size of the Amazon Redshift object to upload an object. Default is 5 MB. Does not apply to elastic mappings.
Number of files per batch	Calculates the number of the staging files per batch. If you do not provide the number of files, Amazon Redshift V2 Connector calculates the number of the staging files. Does not apply to elastic mappings.
Schema Name	Overrides the default schema name.
Target table name	Overwrites the default target table name.
Recovery Schema Name	Schema that contains recovery information stored in the `infa_recovery_table` table on the target system to resume the extraction of the changed data from the last checkpoint. Does not apply to elastic mappings.
Forward Rejected Rows	This property is not applicable for Amazon Redshift V2 Connector.

Mapping task with Oracle CDC sources example

Your organization needs to replicate real-time changed data from a mission-critical Oracle production system to minimize intrusive, non-critical work, such as offline reporting or analytical operations system. You can use Amazon Redshift V2 Connector to capture changed data from the Oracle CDC source and write the changed data to an Amazon Redshift target table. Add the Oracle CDC sources in mappings, and then run the associated mapping tasks to write the changed data to the target.

1. In Data Integration, click **New** > **Mapping** > **Create**.

   The **New Mapping** dialog box appears.

2. Enter a name and description for the mapping.
3. On the Source transformation, specify a name and description in the general properties.
4. On the **Source** tab, select the configured Oracle CDC connection and specify the required source properties.
5. On the Target transformation, specify a name and description in the general properties.
6. On the **Target** tab, perform the following steps to configure the target properties:
   a. In the **Connection** field, select the Amazon Redshift V2 connection.
   b. In the **Target Type** field, select the type of the target object.
   c. In the **Object** field, select the required target object.
   d. In the **Operation** field, select **Data Driven** to properly handle insert, update, and delete records from the source.
   e. In the **Data Driven Condition** field, leave the field empty.
   f. In the **Advanced Properties** section, provide the values of the required target properties. You must select the **Preserve record order on write** check box and enter the value of the **Recovery Schema Name** property.
7. On the **Field Mapping** tab, map the incoming fields to the target fields. You can manually map an incoming field to a target field or automatically map fields based on the field names.
8. In the **Actions** menu, click **New Mapping Task**.

   The **New Mapping Task** page appears.
9. In the **Definition** tab, enter the task name and select the configured mapping.
10. In the **CDC Runtime** tab, specify the required properties.

    For more information about the **CDC Runtime** properties, see the help for Oracle CDC Connector.
11. In the **Schedule** tab, specify the following properties in the **Advanced Session Properties** section:
    a. In the **Commit on End of File** field, select the value of the property as **No**.
    b. In the **Commit Type** field, select the value of the property as **Source**.
    c. In the **Recovery Strategy** field, select the value of the property as **Resume from last checkpoint**.
12. Click **Save** > **Run** the mapping.

    Alternatively, you can create a schedule that runs the mapping task on a recurring basis without manual intervention. You can define the schedule to minimize the time between mapping task runs.

In **Monitor**, you can monitor the status of the logs after you run the task.

Elastic mapping example

You work for an organization that stores large amount of purchase order details, such as customer ID, item codes, and item quantity in Amazon S3. You need to port the data from Amazon S3 to another cloud-based environment to quickly analyze the purchase order details and to increase future revenues.

Create an elastic mapping that runs on the elastic cluster to achieve faster performance when you read all the purchase records from Amazon S3 and write the records to an Amazon Redshift target.

1. In Data Integration, click **New** > **Mappings** > elastic cluster > **Create**.

   The **New Mapping** dialog box appears.

2. Enter a name, location, and description for the mapping.
3. On the Source transformation, specify a name and description in the general properties.
4. On the **Source** tab, perform the following steps to provide the source details to read data from the Amazon S3 source:
    a. In the **Connection** field, select the Amazon S3 V2 source connection.
    b. In the **Source Type** field, select the type of the source.
    c. In the **Object** field, select the required object.
    d. In the **Advanced Properties** section, provide the appropriate values.
5. On the **Fields** tab, map the Amazon S3 source fields to the target fields.
6. On the Target transformation, specify a name and description in the general properties.
7. On the **Target** tab, perform the following steps to provide the target details to write data to the Amazon Redshift target:
    a. In the **Connection** field, select the Amazon Redshift V2 target connection.
    b. In the **Target Type** field, select the type of the target.
    c. In the **Object** field, select the required object.
    d. In the **Operation** field, select the required operation.
    e. In the **Advanced Properties** section, provide appropriate values for the advanced target properties.
8. Map the Amazon S3 source and the Amazon Redshift target.
9. Click **Save** > **Run** to validate the mapping.

    In Monitor, you can monitor the status of the logs after you run the task.

CHAPTER 8

Mass ingestion tasks with Amazon Redshift V2 Connector

This chapter includes the following topics:

Mass ingestion task overview

Use mass ingestion tasks to transfer a large number of files of any file type between on-premises and cloud repositories, and to track and monitor file transfers.

Create an Amazon Redshift V2 connection and use the connection to perform a mass ingestion task. When you create a mass ingestion task, select the target connection and specify which files you want to move from the source to the Amazon Redshift target.

Example

You work for an organization that stores purchase order details data, such as customer ID, item codes, and item quantity in an on-premise flat file system. You need to move the files that contains the purchase order details data from an on-premise flat file system to a cloud-based environment for data analysis.

You can create a mass ingestion task to move all the files that contains the purchase order details data from a flat file system to an Amazon Redshift target at once, instead of moving single row of data separately.

Before you begin

Before you create mass ingestion tasks, verify that the following conditions exist:

- The organization has the following licenses:
  - Mass Ingestion
  - Mass Ingestion Runtime
- The Mass Ingestion application is running on the Secure Agent.
- Source and target connections exist, based on the sources from where you want to transfer files and the targets to where you want to transfer files.

Amazon Redshift V2 targets in mass ingestion tasks

In a mass ingestion task, you can configure the Amazon Redshift V2 target properties to transfer files from any source that mass ingestion tasks supports to an Amazon Redshift target.

The following table describes the Amazon Redshift V2 target properties that you can configure in a mass ingestion task:

Target Property	Description
Connection Type	Type of the target connection. Select **Amazon Redshift V2** as the connection type.
Connection	Select the connection from a list of configured connections.

Amazon Redshift V2 Connector provides the following options that you must select to perform the copy command method:

- **Define Redshift Copy Command Properties**. Select this option to define the Amazon Redshift copy command properties.
- **Enter Custom Redshift Copy Command**. Select this option to provide a custom Amazon Redshift copy command that the mass ingestion task uses.

The following table describes the advanced target properties that you can configure in a mass ingestion task if you select the **Define Redshift Copy Command Properties** option:

Property	Description
Target Table Name	Name of the table in Amazon Redshift to which the files are loaded.
Schema	The Amazon Redshift schema name. Default is the schema that is used while creating connection.
Truncate Target Table	Truncate the target table before loading.

Property	Description
Analyze Target Table	The analyze command collects statistics about the contents of tables in the database to help determine the most efficient execution plans for queries.
Vacuum Target Table	You can select to vacuum the target table to recover disk space and sorts rows in a specified table in the database. You can select the following recovery options: - Full. Sorts the specified table and recovers disk space occupied by rows marked for deletion by previous update and delete operations. - Sort. Sorts the specified table without recovering space freed by deleted rows. - Delete. Recovers disk space occupied by rows marked for deletion by previous update and delete operations, and compresses the table to free up used space.
Copy Options	Select the format with which to copy data. The following options are available: - DELIMITER. A single ASCII character to separate fields in the input file. You can use characters such as pipe (|), tilde (~), or a tab (\t). The delimiter you specify cannot be a part of the data. - QUOTE. Specifies the quote character used to identify `nvarchar` characters and skip them. - COMPUPDATE. Overrides current compression encoding and applies compression to an empty table. - AWS_IAM_ROLE. Specify the Amazon Redshift Role Resource Name to run on an Amazon EC2 system. - IGNOREHEADER. Select to ignore headers. For example, if you specify `IGNOREHEADER 0`, the task processes data from row 0. - DATEFORMAT. Specify the format for date fields. - TIMEFORMAT. Specify the format for time fields.

The following table describes the advanced target properties that you can configure in a mass ingestion task if you select the **Enter Custom Redshift Copy Command** option:

Property	Description

Copy Command	Amazon Redshift COPY command appends the data to any existing rows in the table.
	If the Amazon S3 staging directory and the Amazon Redshift target belongs to different regions, you must specify the region in the COPY command.
	For example,
	`copy public.messages`
	`from '{{FROM-S3PATH}}' credentials`
	`'aws_access_key_id={{ACCESS-KEY-ID}};aws_secret_access_key={{SECRET-ACCESS-KEY-ID}}'`
	`MAXERROR 0 REGION '' QUOTE '"' DELIMITER ',' NULL '' CSV;`
	Where `public` is the schema and `messages` is the table name.
	For more information about the COPY command, see the AWS documentation.

The following table describes the Amazon Redshift advanced target properties that you can configure in a mass ingestion task after you select one of the copy command methods:

Property	Description
Pre SQL	SQL command to run before the mass ingestion task runs the COPY command.
Post SQL	SQL command to run after the mass ingestion task runs the COPY command.

Property	Description
S3 Staging Directory	Specify the Amazon S3 staging directory. You must specify the Amazon S3 staging directory in `<bucket_name/folder_name>` format. The staging directory is deleted after the mass ingestion task runs.
Upload to Redshift with no Intermediate Staging	Upload files from Amazon S3 to Amazon Redshift directly from the Amazon S3 source directory with no additional, intermediate staging. If you select this option, ensure that the Amazon S3 bucket and the Amazon S3 staging directory belong to the same region. If you do not select this option, ensure that the Amazon S3 staging directory and Amazon Redshift target belong to the same region.
File Compression	Determines whether or not files are compressed before they are transferred to the target directory. The following options are available: - None. Files are not compressed. - GZIP. Files are compressed using GZIP compression.
File Encryption Type	Type of Amazon S3 file encryption to use during file transfer. The following options are available: - None. Files are not encrypted during transfer. - S3 server-side encryption. Amazon S3 encrypts the file using AWS-managed encryption keys. - S3 client-side encryption. Ensure that unrestricted policies are implemented for the AgentJVM, and that the master symmetric key for the connection is set. **Note:** Client-side encryption does not apply to tasks where Amazon S3 is the source.
S3 Accelerated Transfer	Select whether to use Amazon S3 Transfer Acceleration on the S3 bucket. To use Transfer Acceleration, accelerated transfer must be enabled for the bucket. The following options are available: - Disabled. Do not use Amazon S3 Transfer Acceleration. - Accelerated. Use Amazon S3 Transfer Acceleration. - Dualstack Accelerated. Use Amazon S3 Transfer Acceleration on a dual-stack endpoint.
Minimum Upload Part Size	Minimum upload part size when uploading a large file as a set of multiple independent parts, in megabytes. Use this option to tune the file load to Amazon S3.
Multipart Upload Threshold	Multipart download minimum threshold to determine when to upload objects in multipleparts in parallel.

Custom Amazon Redshift Copy command

If you select to use an Amazon Redshift target connection, you can create a custom copy command that the mass ingestion task triggers to load files to Amazon Redshift.

You must specify credentials and variables in the command in the following format:

```
<ID> = <variable>
```

You can use the following credential IDs and variables for the custom copy command:

ID	Variable and Description
aws_access_key_id	Variable: {{ACCESS-KEY-ID}} Description: The AWS access key ID.
aws_secret_access_key	Variable: {{SECRET-ACCESS-KEY-ID}} Description: The AWS secret access key.
master_symmetric_key	Variable: {{MASTER-KEY}} Description: The master symmetric key.
from	Variable: {{FROM-S3PATH}} Description: The Amazon S3 folder.

You can also specify the format with which to copy data. The following options are available:

- DELIMITER. A single ASCII character to separate fields in the input file. You can use characters such as pipe (|), tilde (~), or a tab (\t). The delimiter you specify cannot be a part of the data.
- QUOTE. Specifies the quote character to use with comma separated values.
- COMPUPDATE. Overrides current compression encoding and applies compression to an empty table.
- AWS_IAM_ROLE. Specify the Amazon Redshift Role Resource Name to run on an Amazon EC2 system.
- IGNOREHEADER. Select to ignore headers. For example, if you specify `IGNOREHEADER 0`, the task processes data from row 0.
- DATEFORMAT. Specify the format for date fields.
- TIMEFORMAT. Specify the format for time fields.

For more information, see the Amazon Redshift copy command documentation at http://docs.aws.amazon.com/redshift/latest/dg/r_COPY.html.

The following code provides an example of a custom copy command:

```
copy "public"."test_str_tgt" ("col1" , "col2") from '{{FROM-S3PATH}}'
credentials 'aws_access_key_id={{ACCESS-KEY-ID}};aws_secret_access_key={{SECRET-ACCESS-
KEY-ID}}'
MAXERROR 0
QUOTE '"'
DELIMITER ','
DATEFORMAT AS 'YYYY-MM-DD HH24:MI:SS'
ROUNDEC
TIMEFORMAT AS 'YYYY-MM-DD HH24:MI:SS'
NULL ''
CSV
MANIFEST
```

Creating a mass ingestion task

You can create a mass ingestion task to transfer files from any source that mass ingestion task supports to an Amazon Redshift target.

1. In Data Integration, click **New** > **Tasks**.

2. Select **Mass Ingestion** and then click **Create**.

   The **Definition** tab appears.

3. In the **Definition** tab, configure the following properties:

Property	Description
Task Name	Name of the mass ingestion task. The names of mass ingestion tasks must be unique within the organization. Task names can contain alphanumeric characters, spaces, and underscores. Names must begin with an alphabetic character or underscore. Task names are not case sensitive.
Location	Project folder in which the task resides.
Description	Optional description of the task. Maximum length is 1024 characters.
Runtime Environment	Runtime environment that contains the Secure Agent used to run the task. The Mass Ingestion application must run on the Secure Agent.

4. Click **Next**.

   The **Source** tab appears.

5. On the **Source Details** page, select connection from a list of configured connections in the **Connection Type** field.

   You can select one of the following sources that mass ingestion task supports:

   - Local folder
   - Advanced FTP
   - Advanced FTPS
   - Advanced SFTP
   - Amazon S3

6. Click **View** to view the connection details.
7. Click **Test** to test the connection in the **View Connection** dialog.
8. Click **Next**.

   The **Target** tab appears.

9. On the **Target Details** section, select the **Connection Type** as **Amazon Redshift V2** and configure the Amazon Redshift V2 target properties.
10. Click **View** to view the connection details.
11. Click **Test** to test the connection in the **View Connection** dialog.
12. Click **Next**.

    The **Schedule** tab appears where you can select whether to run the task on a schedule or without a schedule.

13. Click **Run this task on schedule** to run a task on a schedule and select the schedule you want to use.

    If you want to remove a task from a schedule, click **Do not run this task on a schedule**.

14. Click **Finish** to save and close the task wizard.

You can edit, run, or view the mass ingestion task on the **Explore** page after you create the mass ingestion task.

Viewing mass ingestion task details

You can view details about a mass ingestion task, including the source and target connections and the associated schedule.

1. On the **Explore** page, navigate to the task.
2. In the row that contains the task, click **Actions** and select **View**.

   The **Task Details** page appears with task, source, target, and schedule details.
3. You can edit or run the task that you selected to view. On the **Task Details** page, click **Edit** to modify the task or click **Run** to run the task.

Running a mass ingestion task

You can run a mass ingestion task in the following ways:

1. To run a mass ingestion task manually, on the **Explore** page, navigate to the task. In the row that contains the task, click **Actions** and select **Run**.

   Alternatively, you can run the task manually from the **Task Details** page. To access the **Task Details** page, click **Actions** and select **View**. In the **Task Details** page, select **Run**.
2. To run a mass ingestion task on a schedule, edit the task in the mass ingestion task wizard to associate the task with a schedule.

CHAPTER 9

Amazon Redshift pushdown optimization

This chapter includes the following topics:

Amazon Redshift pushdown optimization overview

You can use pushdown optimization to push the transformation logic to the target databases. The amount of transformation logic that you can push to the database depends on the database, transformation logic, and task configuration. The task processes all transformation logic that it cannot push to a database.

You can configure pushdown optimization by using any of the following connections:

- **Amazon Redshift V2 connection**: Use this connection to pushdown a mapping with Amazon S3 as source and Amazon Redshift as target. When data is read from the source to the target, this connection uses the AWS commands.
- **ODBC connection**: Use this connection to pushdown a mapping with Amazon Redshift as source and target. When data is read from the source to the target, this connection uses the database commands.

Amazon Redshift pushdown with Amazon S3 source and Amazon Redshift V2 target

You can set pushdown optimization for the Amazon Redshift V2 connection that uses Amazon Redshift drivers to enhance the mapping performance. You can configure Full pushdown only when you read data from an Amazon S3 source and write to an Amazon Redshift target.

Note: Full pushdown optimization supports only insert operations for Amazon Redshift.

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 Web Services and stores its data in Amazon S3. The organization plans to implement a business intelligence service to build visualization and perform real-time analysis. You can load data from Amazon S3 to Amazon Redshift by configuring the transformations using the AWS commands, to support the adequate data warehouse model and the consuming requirements.

Create an Amazon S3 V2 connection to read data form the Amazon S3 source. Create an Amazon Redshift V2 connection and use pushdown optimization to write data to the Amazon Redshift target. Using the Amazon Redshift V2 connection with pushdown optimization enhances the performances and reduces the cost involved.

Configuring optimization for an Amazon Redshift V2 mapping task

Perform the following steps to configure pushdown optimization for an Amazon Redshift V2 mapping task:

1. Create an Amazon S3 V2 connection and an Amazon Redshift V2 connection.
2. Create a mapping to read data from an Amazon S3 source and write data to an Amazon Redshift target.
3. Create a mapping task.
    a. Select the configured mapping.
    b. In the **Advanced Session Properties** on the **Schedule** tab, add **Pushdown Optimization** and set the value to **Full**.
    c. Save the task and click **Finish**.

When you run the mapping task, the transformation logic is pushed to the Amazon Redshift database.

Pushdown optimization supported functions and transformations

The following table summarizes the availability of pushdown functions in an Amazon Redshift database. Columns marked with an X indicate that the function can be pushed to the Amazon Redshift database by using full pushdown optimization. Columns marked with a dash (-) symbol indicate that the function cannot be pushed to the database.

Function	Pushdown	Function	Pushdown	Function	Pushdown
ABORT()	-	INITCAP()	X	REG_MATCH()	-
ABS()	X	INSTR()	X	REG_REPLACE	-
ADD_TO_DATE()	X	IS_DATE()	-	REPLACECHR()	X
AES_DECRYPT()	-	IS_NUMBER()	-	REPLACESTR()	X
AES_ENCRYPT()	-	IS_SPACES()	-	REVERSE()	-
ASCII()	-	ISNULL()	-	ROUND(DATE)	-
AVG()	-	LAST()	-	ROUND(NUMBER)	X
CEIL()	X	LAST_DAY()	X	RPAD()	X

Function	Pushdown	Function	Pushdown	Function	Pushdown
CHOOSE()	-	LEAST()	-	RTRIM()	X
CHR()	X	LENGTH()	X	SET_DATE_PART()	-
CHRCODE()	-	LN()	X	SIGN()	X
COMPRESS()	-	LOG()	X	SIN()	X
CONCAT()	X	LOOKUP	-	SINH()	-
COS()	X	LOWER()	X	SOUNDEX()	-
COSH()	-	LPAD()	X	SQRT()	X
COUNT()	-	LTRIM()	X	STDDEV()	-
CRC32()	-	MAKE_DATE_TIME()	-	SUBSTR()	X
CUME()	-	MAX()	-	SUM()	-
DATE_COMPARE()	X	MD5()	-	SYSTIMESTAMP()	X
DATE_DIFF()	X	MEDIAN()	-	TAN()	X
DECODE()	X	METAPHONE()	-	TANH()	-
DECODE_BASE64()	-	MIN()	-	TO_BIGINT	X
DECOMPRESS()	-	MOD()	X	TO_CHAR(DATE)	X
ENCODE_BASE64()	-	MOVINGAVG()	-	TO_CHAR(NUMBER)	X
EXP()	X	MOVINGSUM()	-	TO_DATE()	X
FIRST()	-	NPER()	-	TO_DECIMAL()	X
FLOOR()	X	PERCENTILE()	-	TO_FLOAT()	X
FV()	-	PMT()	-	TO_INTEGER()	X
GET_DATE_PART()	X	POWER()	X	TRUNC(DATE)	X
GREATEST()	-	PV()	-	TRUNC(NUMBER)	X
IIF()	-	RAND()	-	UPPER()	X
IN()	-	RATE()	-	VARIANCE()	-
INDEXOF()	-	REG_EXTRACT()	-		

The following table lists the pushdown operators that can be used in an Amazon Redshift database. Columns marked with an X indicate that the operator can be pushed to the Amazon Redshift database by using full pushdown optimization.

Operator	Pushdown
+	X
-	X
*	X
/	X
%	X
||	X
>	X
<	X
=	X
>=	X
<=	X
!=	X
AND	X
OR	X
NOT	X

The following table lists the transformation logic that is supported by pushdown optimization:

Transformations	Pushdown
Expression	Full
Filter	Full

Feature support for pushdown optimization

You must configure an Amazon S3 V2 connection with basic or IAM authentication when you enable pushdown optimization in a mapping task.

When you configure pushdown optimization, the mappings support the following advance properties for an Amazon S3 V2 source:

- Client-side encryption

- Data compression
  - GZIP only
- File formatting
  - Delimiter
  - Qualifier
  - Code Page
  - Header Line Number
  - First Data Row
  - Source Type
  - Folder Path
- File source type
- File name
- Format type
  - Avro
  - Parquet
  - ORC
  - JSON
  - Delimited

When you configure pushdown optimization, the mappings support the following advance properties for an Amazon Redshift V2 target:

- Analyze target table
- COPY command
  - Region
  - Truncatecolumn
  - AWS_IAM_Role, only for Parquet and ORC files
- Pre-SQL
- Post-SQL
- Require null value for Char and Varchar
- Schema name
- Target table name
- Truncate target table before data upload
- Vacuum table
- Treat Source Rows As (only INSERT supported)

Note: If you configure source and target advanced properties that are not supported, the mappings run in the Informatica runtime environment.

Rules and guidelines for functions in pushdown optimization

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

- To push TO_DATE() and TO_CHAR() to Amazon Redshift, you must define the string and format arguments.
- If you use the NS format as part of the ADD_TO_DATE() function, the agent 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 agent 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

  - - DDD

  - - HH24

  - - MI

  - - MM

  - - MS

  - - SS

  - - US

  - - YYYY
- To copy data from Amazon S3 to Amazon Redshift, you must use multiple data files by splitting large files. For more information, see the Amazon documentation.
- If the Amazon S3 bucket region and the Amazon Redshift region are different, specify the **REGION** attribute in the **COPY** command to enable full pushdown optimization.

  Note: Does not apply to ORC and Parquet files.
- If the data has values that are greater than the precision values, specify the attribute **TRUNCATECOLUMNS=ON.**

  Note: Does not apply to ORC and Parquet files.
- For the ORC and Parquet file types, specify **AWS_IAM_ROLE** in the COPY command, to enable full pushdown optimzation.
- You can use **REPLACESTR()** only to replace a single string value with a new string value.
  Syntax

  ```
  REPLACESTR ( CaseFlag, InputString, OldString, NewString).
  ```
- To push **SUBSTR()** to Amazon Redshift, you must define an integer value for the length argument.
- You cannot enable full pushdown optimization for a mapping task when the task contains a mapping with a single transformation connected to multiple transformations downstream and vice-versa.

Amazon Redshift pushdown through ODBC

Use an ODBC connection to enable full or source pushdown optimization when you want to read data from an Amazon Redshift source, write to an Amazon Redshift target, and if you want to run the mapping logic entirely within Amazon Redshift.

When you run a task configured for pushdown optimization, the task converts the transformation logic to an SQL statement. The task sends the SQL statement to the database, and the database executes the SQL statement.

You can set the pushdown optimization for the ODBC connection type that uses Amazon ODBC Redshift drivers to enhance the mapping performance. You must create a data source name in the ODBC datasource administrator.

After you create an Amazon Redshift ODBC connection, select the value of the **Pushdown Optimization** property as **Full** or **To Source** in the advanced session properties. You can check the session log to verify that the pushdown optimization has taken place.

Note: Amazon Redshift does not support upsert operations in a full 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 Web Services and stores its data in Amazon Redshift, a petabytescale 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 Amazon Redshift to the business intelligence service. You can use Amazon Redshift V2 Connector to read data from Amazon Redshift. To read this large amount of data, you can use source pushdown for the ODBC connection type. Using the ODBC connection type with pushdown optimization enhances the performance.

Configuring Amazon Redshift ODBC connection

Amazon Redshift supports Amazon ODBC Redshift drivers on Windows and Linux systems. You must install the Amazon ODBC Redshift 64-bit driver based on your system requirement.

Note: Informatica certifies Amazon Redshift ODBC driver version, `AmazonRedshiftODBC-64-bit-1.4.8.1000-1.x86_64`, to use for pushdown optimization.

Configuring Amazon Redshift ODBC connection on Windows

Before you establish an ODBC connection to connect to Amazon Redshift on Windows, you must configure the ODBC connection.

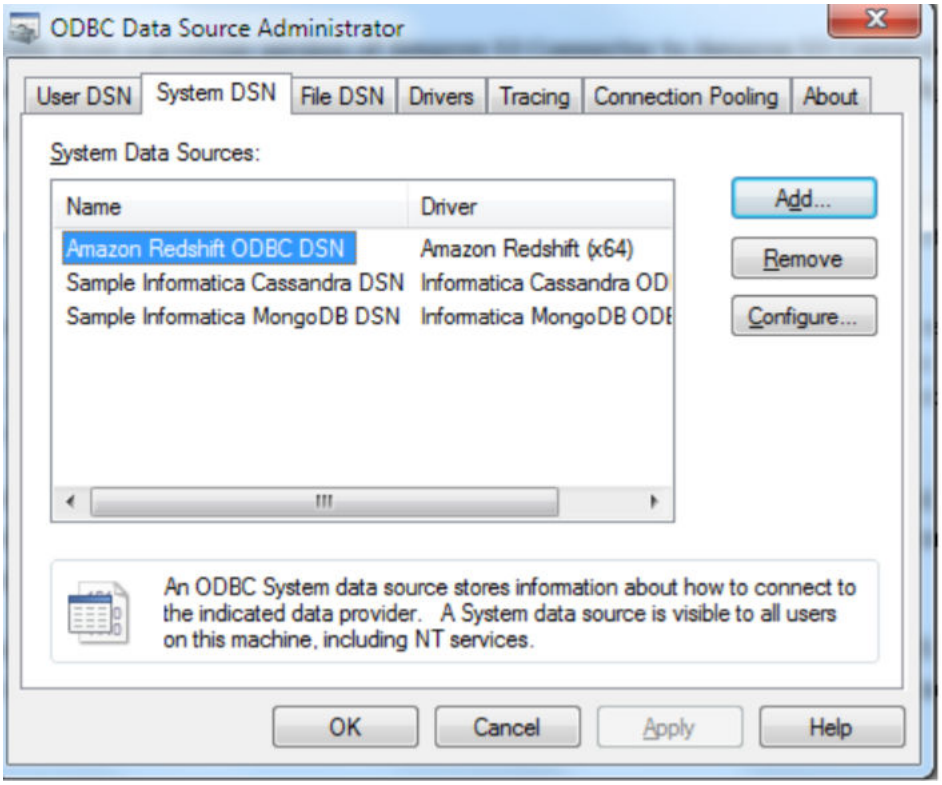
Perform the following steps to configure an ODBC connection on Windows:

1. Download the Amazon Redshift ODBC drivers from the AWS website.

   You must download the Amazon Redshift ODBC 64-bit driver.
2. Install the Amazon Redshift ODBC drivers on the machine where the Secure Agent is installed.
3. Open the folder in which ODBC data source file is installed.
4. Run the `odbcad32.exe` file.

   The **ODBC Data Source Administrator** dialog box appears.
5. Click **System DSN**.

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

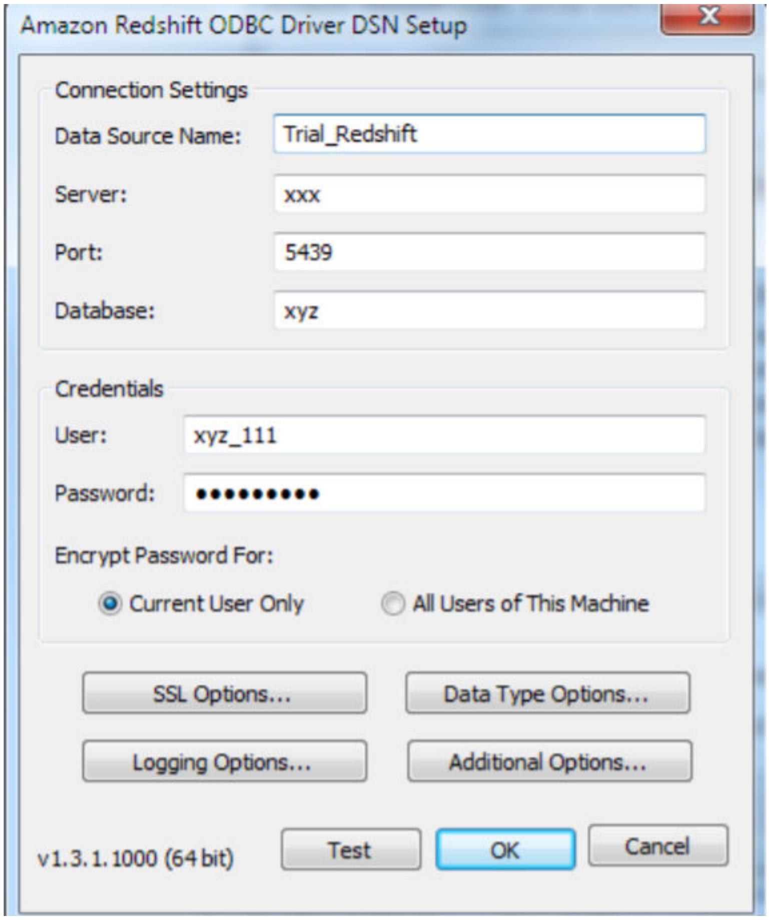


Name	Driver
Amazon Redshift ODBC DSN	Amazon Redshift (x64)
Sample Informatica Cassandra DSN	Informatica Cassandra OD
Sample Informatica MongoDB DSN	Informatica MongoDB ODE

An ODBC System data source stores information about how to connect to the indicated data provider. A System data source is visible to all users on this machine, including NT services.

6. Click **Configure**.

The **Amazon Redshift ODBC Driver DSN Setup** dialog box displays. The following image shows the **Amazon Redshift ODBC Driver DSN Setup** dialog box where you can configure the **Connection Settings** and **Credentials** section:



Amazon Redshift ODBC Driver DSN Setup

Connection Settings

Data Source Name: Trial_Redshift

Server: xxx

Port: 5439

Database: xyz

Credentials

User: xyz_111

Password: ●●●●●●●●●

Encrypt Password For:

● Current User Only ○ All Users of This Machine

7. 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.

8. Specify the following credentials in the **Credentials** section:

Property	Description
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: - **Current User Only** - **All Users of This Machine** Default is **Current User Only**.

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

The Amazon Redshift ODBC connection is configured successfully on Windows.

After you configure the Amazon Redshift ODBC connection, you must create an ODBC connection to connect to Amazon Redshift.

For more information about how to create an ODBC connection to connect to Amazon Redshift, see "Creating an ODBC connection" on page 75

Configuring Amazon Redshift ODBC connection on Linux

Before you establish an ODBC connection to connect to Amazon Redshift on Linux, you must configure the ODBC connection.

Perform the following steps to configure an ODBC connection on Linux:

1. Download the Amazon Redshift ODBC drivers from the AWS website.

   You must download the Amazon Redshift ODBC 64-bit driver.
2. Install the Amazon Redshift ODBC drivers on the machine where the Secure Agent is installed.
3. 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
```

4. 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.

Property	Description
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.

5. Run the following command to export the `odbc.ini` file.

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

6. Restart the Secure Agent.

   The Amazon Redshift ODBC connection on Linux is configured successfully.

After you configure the Amazon Redshift ODBC connection, you must create an ODBC connection to connect to Amazon Redshift.
For more information about how to create an ODBC connection to connect to Amazon Redshift, see "Creating an ODBC connection" on page 75

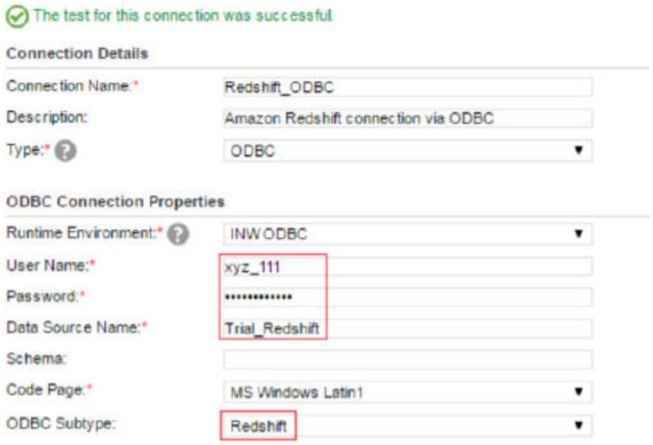
Creating an ODBC connection

You must create an ODBC connection to connect to Amazon Redshift after you configure the ODBC connection.

Perform the following steps to create an Amazon Redshift ODBC connection on the **Connections** page:

1. In Administrator, click **Connections**.
2. In the upper right corner, click **New Connections**.

   The **New Connection** page appears. The following image shows the **New Connection** page:



The test for this connection was successful

Connection Details

Connection Name:*	Redshift_ODBC
Description:	Amazon Redshift connection via ODBC
Type:*	ODBC

ODBC Connection Properties

Runtime Environment:*	INW ODBC
User Name:*	xyz_111
Password:*	••••••••••••
Data Source Name:*	Trial_Redshift
Schema:	
Code Page:*	MS Windows Latin1
ODBC Subtype:	Redshift

3. Configure the following connection details in the **Connection Details** section:

Property	Description
Connection Name	Name of the ODBC connection.
Description	Description of the connection.
Type	Type of the connection. Select the type of the connection as **ODBC**.

4. Configure the following connection details in the **Connection Properties** section:

Property	Description
Runtime Environment	The name of the runtime environment where you want to run the tasks.
User Name	User name of the Amazon Redshift account.
Password	Password for the Amazon Redshift account.
Data Source Name	Enter the name of the ODBC data source name that you created for the Amazon Redshift database.
Schema	Amazon Redshift schema name.
Code Page	Select the code page that the Secure Agent must use to read or write data.
ODBC Subtype	Enter the value of the **ODBC Subtype** field as **Redshift**.

The Amazon Redshift ODBC connection is created successfully.

Cross-Schema pushdown optimization

You can configure cross-schema pushdown optimization for a mapping task that uses a Amazon Redshift ODBC connection to read or write data to Amazon Redshift objects of different schemas in the same database.

To use cross-schema pushdown optimization, create Amazon Redshift ODBC connections and specify the schema for the source and target connections. The source and target schemas must be different but must belong to the same database. Configure pushdown optimization for the mapping task and enable cross-schema pushdown optimization in the advanced session properties. By default, the **Enable cross-schema pushdown optimization** check box is selected.

Configuring cross-schema optimization for an Amazon Redshift V2 mapping task

Perform the following steps to configure cross-schema pushdown optimization for an Amazon Redshift V2 mapping task:

1. Create Amazon Redshift ODBC source and target connections, each defined with a different schema.

For example,

- Create a `rs_odbc1` Amazon Redshift ODBC connection and specify `CQA_SCHEMA1` schema in the connection properties.
- Create a `rs_odbc2` Amazon Redshift ODBC connection and specify `CQA_SCHEMA2` schema in the connection properties.

2. Create an Amazon Redshift V2 mapping.

   For example, create a `m_rs_pdo_crossSchema` Amazon Redshift V2 mapping.

3. Add a Source transformation. Include an Amazon Redshift V2 source object and connection to read data using the schema specified in the connection.

   For example, add a Source transformation. Include an Amazon Redshift V2 source object and connection `rs_odbc1` to read data using `CQA_SCHEMA1`.

4. Add a Target transformation. Include an Amazon Redshift V2 target object and connection to write data using the schema specified in the connection.

   For example, add a Target transformation. Include an Amazon Redshift V2 target object and connection `rs_odbc2` to write data using `CQA_SCHEMA2`.

5. Create an Amazon Redshift V2 mapping task, and perform the following tasks:

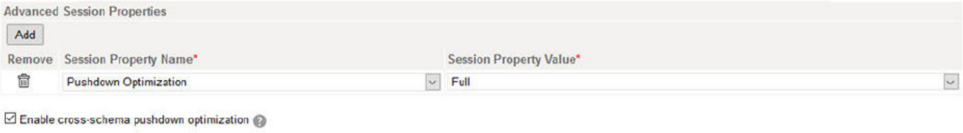
   a. Select the configured Amazon Redshift V2 mapping.

      For example, select the `m_rs_pdo_crossSchema` Amazon Redshift V2 mapping.

   b. In the **Advanced Options** on the **Schedule** tab, add **Pushdown Optimization** and set the value to **Full**.

   c. Select **Enable cross-schema pushdown optimization**.

      The following image shows the configured **Enable cross-schema pushdown optimization** property:



Pushdown Optimization: Full

☑ Enable cross-schema pushdown optimization

   d. Save the task and click **Finish**.

   When you run the mapping task, the Secure Agent reads data from the Amazon Redshift source object associated with the `CQA_SCHEMA1` schema and writes data to the Amazon Redshift V2 target object associated with `CQA_SCHEMA2` schema.

Pushdown optimization supported functions and transformations

The following table summarizes the availability of pushdown functions in an Amazon Redshift database. Columns marked with an X indicate that the function can be pushed to the Amazon Redshift database by using source-side or full pushdown optimization. Columns marked with S indicate that the function can be pushed to the Amazon Redshift database only by using source-side pushdown optimization. Columns marked with a dash (-) symbol indicate that the function cannot be pushed to the database.

Function	Pushdown	Function	Pushdown	Function	Pushdown
ABORT()	-	INSTR()	X	REG_REPLACE	-
ABS()	X	IS_DATE()	-	REPLACECHR()	-
ADD_TO_DATE()	X	IS_NUMBER()	-	REPLACESTR()	-

Function	Pushdown	Function	Pushdown	Function	Pushdown
AES_DECRYPT()	-	IS_SPACES()	-	REVERSE()	-
AES_ENCRYPT()	-	ISNULL()	S	ROUND(DATE)	-
ASCII()	-	LAST()	-	ROUND(NUMBER)	X
AVG()	S	LAST_DAY()	X	RPAD()	X
CEIL()	X	LEAST()	-	RTRIM()	X
CHOOSE()	-	LENGTH()	X	SET_DATE_PART()	-
CHRCODE()	-	LN()	X	SIGN()	X
COMPRESS()	-	LOG()	-	SIN()	X
CONCAT()	X	LOOKUP	-	SINH()	-
COS()	X	LOWER()	X	SOUNDEX()	-
COSH()	-	LPAD()	X	SQRT()	X
COUNT()	S	LTRIM()	X	STDDEV()	S
CRC32()	-	MAKE_DATE_TIME()	-	SUBSTR()	X
CUME()	-	MAX()	S	SUM()	S
DATE_COMPARE()	X	MD5()	-	SYSTIMESTAMP()	S
DATE_DIFF()	X	MEDIAN()	-	TAN()	S
DECODE()	X	METAPHONE()	-	TANH()	-
DECODE_BASE64()	-	MIN()	S	TO_BIGINT	X
DECOMPRESS()	-	MOD()	S	TO_CHAR(DATE)	S
ENCODE_BASE64()	-	MOVINGAVG()	-	TO_CHAR(NUMBER)	X
EXP()	X	MOVINGSUM()	-	TO_DATE()	X
FIRST()	-	NPER()	-	TO_DECIMAL()	X
FLOOR()	X	PERCENTILE()	-	TO_FLOAT()	X
FV()	-	PMT()	-	TO_INTEGER()	X
GET_DATE_PART()	X	POWER()	X	TRUNC(DATE)	S
GREATEST()	-	PV()	-	TRUNC(NUMBER)	S
IIF()	X	RAND()	-	UPPER()	X

Function	Pushdown	Function	Pushdown	Function	Pushdown
IN()	S	RATE()	-	VARIANCE()	S
INDEXOF()	-	REG_EXTRACT()	-		
INITCAP()	X	REG_MATCH()	-		

The following table lists the pushdown operators that can be used in an Amazon Redshift database. Columns marked with an X indicate that the operator can be pushed to the Amazon Redshift database by using source-side, or full pushdown optimization.

Operator	Pushdown
+	X
-	X
*	X
/	X
%	X
||	X
>	X
=	X
>=	X
<=	X
!=	X
AND	X
OR	X
NOT	X
^=	X

The following table lists the transformation logic that the Secure Agent can push to an Amazon Redshift source or target:

Transformations	Pushdown
Aggregator	Source, Full
Expression	Source, Full

Transformations	Pushdown
Filter	Source, Full
Joiner	Source, Full
Sorter	Source, Full
Union	Source, Full
Router	Full

Rules and guidelines for functions in pushdown optimization

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

- To push TRUNC(DATE) to Amazon Redshift, you must define the date and format arguments. Otherwise, the agent does not push the function to Amazon Redshift .
- The aggregator functions for Amazon Redshift accept only one argument, a field set for the aggregator function. The filter condition argument is ignored. In addition, verify that all fields mapped to the target are listed in the GROUP BY clause.
- The Update Override ODBC advanced target property is not applicable when you use an ODBC connection to connect to Amazon Redshift.
- To push TO_DATE() to Amazon Redshift, you must define the string and format arguments.
- To push TO_CHAR() to Amazon Redshift, you must define the date and format arguments.
- Do not specify a format for SYSTIMESTAMP() to push the SYSTIMESTAMP to Amazon Redshift. The Amazon Redshift database returns the complete time stamp.
- 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 agent 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 agent 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

  - - DDD

  - - HH24

  - - MI

- - MM
- - MS
- - SS
- - US
- - YYYY

CHAPTER 10

Data type reference

This chapter includes the following topics:

Data type reference overview

Data Integration uses the following data types in mappings, mapping tasks, and mass ingestion tasks with Amazon Redshift:

Amazon Redshift native data types

Amazon Redshift data types appear in the source and target transformations when you choose to edit metadata for the fields.

Transformation data types

Set of data types that appear in the transformations. They are internal data types based on ANSI SQL-92 generic data types, which the Secure Agent uses to move data across platforms. Transformation data types appear in all transformations in a mapping.

When Data Integration reads source data, it converts the native data types to the comparable transformation data types before transforming the data. When Data Integration writes to a target, it converts the transformation data types to the comparable native data types.

Amazon Redshift and transformation data types

The following table lists the Amazon Redshift data types that Data Integration supports and the corresponding transformation data types:

Amazon Redshift Data Type	Transformation Data Type	Description
Bigint	Bigint	Signed eight-byte integer.
Boolean	Small Integer	Logical Boolean (true/false).

Amazon Redshift Data Type	Transformation Data Type	Description
Char	String	Fixed-length character string.
Date	Timestamp	Calendar date (year, month, day).
Decimal	Decimal	Exact numeric of selectable precision.
Double Precision	Double	Double precision floating-point number.
Integer	Integer	Signed four-byte integer.
Real	Double	Single precision floating-point number.
Smallint	Small Integer	Signed two-byte integer.
Timestamp	Timestamp	Date and time (without time zone).
Varchar	String	Variable-length character string with a user-defined limit.

CHAPTER 11

Troubleshooting

This chapter includes the following topics:

Troubleshooting overview

Use the following sections to troubleshoot errors in Amazon Redshift V2 Connector.

Troubleshooting for Amazon Redshift V2 Connector

How to configure AWS IAM authentication for Amazon Redshift V2 Connector?

For information about configuring AWS IAM authentication, see https://kb.informatica.com/h2l/HowTo%20Library/1/0972-ConfiguringAWSIAMforAmazonRedshiftandAmazonRedshiftV2Connectors-H2L.pdf

Part III: Data Integration with Amazon Redshift Connector

This part contains the following chapters:

CHAPTER 12

Introduction to Amazon Redshift Connector

This chapter includes the following topics:

Amazon Redshift Connector overview

You can use Amazon Redshift Connector to securely read data from or write data to Amazon Redshift. Amazon Redshift sources and targets represent records in Amazon Redshift.

You can create an Amazon Redshift connection and use the connection in synchronization tasks, mappings and mapping tasks. When you use Amazon Redshift objects in synchronization tasks, mappings, and mapping tasks, you must configure properties specific to Amazon Redshift.

Create an Amazon Redshift connection to specify the location of Amazon Redshift sources, lookups, and targets you want to include in a task. When you run an Amazon Redshift synchronization task, mapping, or mapping task, the agent writes data to Amazon Redshift based on the workflow and Amazon Redshift connection configuration. The agent 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 a source and simultaneously move data to Amazon Redshift clusters. The agent issues a copy command that copies data from Amazon S3 to the Amazon Redshift target table.

You can also read data from or write data to the Amazon Redshift cluster that reside in a Virtual Private Cloud (VPC). When you read data from or write data to Amazon Redshift, you can specify the Hosted Agent or the Secure Agent.

You can move data from any data source to Amazon Redshift. Data Integration that uses the Amazon driver to communicate with Amazon Redshift.

Note: Informatica recommends to use Amazon Redshift V2 Connector as the new features and enhancements are provided for Amazon Redshift V2 Connector.

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 the Amazon driver can communicate with Amazon Redshift.

Amazon Redshift Connector 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 an Amazon Redshift target for data analysis.

Administration of Amazon Redshift Connector

As a user, you can use Amazon Redshift Connector after the organization administrator ensures that users have access to the Secure Agent directory that contains the success and error files. This directory path must be the same on each Secure Agent machine in the runtime environment. The organization administrator must also perform the following tasks:

- Get the Amazon Redshift JDBC URL.
- Manage Authentication. Use either of the following two methods:
  - Create an Access Key ID and Secret Access Key.
    Provide the values for access key ID and secret access key when you configure the Amazon Redshift connection. For more information about creating an access key ID and secret access key, see the AWS documentation.
  - Configure AWS Identity and Access Management (IAM) Authentication to enhance security.
    If you use IAM authentication, do not provide access key ID and secret access key explicitly in the Amazon Redshift connection. Instead, you must create an Redshift Role Amazon Resource Name (ARN), add the minimal Amazon S3 bucket policy to the Redshift Role ARN, and add the Redshift Role ARN to the Redshift cluster.

    Provide the Redshift Role ARN in the AWS_IAM_ROLE option in the UNLOAD and COPY commands when you create a task.

    If you specify both, access key ID and secret access key in the connection properties and AWS_IAM_ROLE in the UNLOAD and COPY commands, AWS_IAM_ROLE takes the precedence.

    You must add IAM EC2 role and IAM Redshift role to the customer master key when you use IAM authentication and server-side encryption using customer master key.

    Hosted Agent does not support IAM authentication. For more information about how to configure IAM authentication for Amazon Redshift Connector, see “IAM Authentication” on page 89

- Configure Amazon Redshift for SSL if you want to support an SSL connection.
- Create a master symmetric key if you want to enable client-side encryption.
- Create an AWS Key Management Service (AWS KMS)-managed customer master key if you want to enable server-side encryption.
- Create minimal Amazon S3 bucket policy for Amazon Redshift Connector.
- When you create a temporary table for an upsert, update, or delete operation in the local staging area, you must create the temporary table in the following format:

  ```
  RecordName + "_" + time-stamp + ProcessID + PartitionId
  ```

  Note: By default, you have the permission to create the temporary tables as you have the PUBLIC group membership. To deny the permission, revoke the TEMP permission from the PUBLIC group and allow the TEMP permission to specific or groups of individuals.

Configure Amazon Redshift for SSL

You can configure the Secure Agent to support an SSL connection to Amazon Redshift.

1. Download the Amazon Redshift certificate from the following location: https://s3.amazonaws.com/redshift-downloads/redshift-ssl-ca-cert.pem.
2. Run the following command to add the certificate file to the key store: `${JAVA_HOME}/bin/keytool -keystore {JAVA_HOME}/lib/security/cacerts -import -alias <string_value> -file <certificate_filepath>`.
3. In Administrator, select **Runtime Environments**.
4. Select the Secure Agent for which you want to increase memory from the list of available Secure Agents.
5. In the upper-right corner, click **Edit**.
6. In the **System Configuration Details** section, change the **Type** to **DTM**.
7. Click the **Edit Agent Configuration** icon next to **JVMOption1** and add the following command: -`Djavax.net.ssl.trustStore=<keystore_name>`.
8. Click the **Edit Agent Configuration** icon next to **JVMOption2** and add the following command:-`Djavax.net.ssl.trustStorePassword=<password>`.
9. Add the following parameter to the JDBC URL you specified in your Amazon Redshift connection properties: `ssl=true`. For example, `jdbc:redshift://mycluster.xyz789.us-west-2.redshift.amazonaws.com:5439/dev?ssl=true`.
10. Click **OK** to save your changes.

Create minimal Amazon S3 bucket policy

The minimal Amazon S3 bucket policy restricts user operations and user access to particular Amazon S3 buckets by assigning an AWS IAM policy to users. You can configure the AWS IAM policy through the AWS console.

You can use the following minimum required actions for users to successfully read data from and write data to Amazon Redshift resources:

- PutObject
- GetObject
- DeleteObject
- ListBucket

- GetBucketPolicy

Sample Policy:

```
{

"Version": "2012-10-17", "Statement": [

{ "Effect": "Allow", "Action": [ "s3:PutObject", "s3:GetObject", "s3:DeleteObject",
"s3:ListBucket", "s3:GetBucketPolicy" ], "Resource":
[ "arn:aws:s3:::<specify_bucket_name>/*", "arn:aws:s3:::<specify_bucket_name>" ] }

]

}
```

Note: The **Test Connection** does not validate the IAM policy assigned to users. The Amazon S3 bucket name is available in the advanced properties for source and target.

You must make sure that the Amazon S3 bucket and Amazon Redshift cluster reside in the same region to run the synchronization tasks and mapping tasks a session successfully.

You can only read data from or write data to the regions supported by AWS SDK used by the Connector. The supported regions are:

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

IAM Authentication

Optional. You can configure IAM authentication when Secure Agent is installed on an Amazon Elastic Compute Cloud (EC2) system. Use IAM authentication for secure and controlled access to Amazon Redshift resources when you run synchronization tasks and mapping tasks.

Use IAM authentication when you want to run the synchronization and mapping tasks on Secure agent installed on an EC2 system. Perform the following steps to configure IAM authentication:

1. Create Minimal Amazon S3 Bucket Policy. For more information, see "Create minimal Amazon S3 bucket policy" on page 88.

2. Create the Amazon 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. 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 Secure Agent on the EC2 system.

CHAPTER 13

Amazon Redshift connections

This chapter includes the following topics:

Amazon Redshift connections overview

Create an Amazon Redshift connection to securely read data from or write data to Amazon Redshift. You can use Amazon Redshift connections to specify sources and targets in synchronization tasks, mappings, and mapping tasks.

Create a connection and associate it with a synchronization task, mapping, or mapping task. Define the source and target properties to read data from or write data to Amazon Redshift.

You can create an Amazon Redshift connection on the **Connections** page and use it in the Mapping Designer when you create a mapping or in the Synchronization Task wizard when you create a task. The connection becomes available to the entire organization.

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:

Connection property	Description
Runtime Environment	The name of the runtime environment where you want to run the tasks.
Username	User name of the Amazon Redshift account.
Password	Password for the Amazon Redshift account.
Schema	Amazon Redshift schema name. Default is public.

Connection property	Description
AWS Access Key ID	Optional. Amazon S3 bucket access key ID. To run tasks on Secure Agent installed on an EC2 system, you might leave the Access Key ID blank. To run tasks on Secure Agent that is not installed on an EC2 system, you must provide the Access Key ID.
AWS Secret Access Key	Optional. Amazon S3 bucket secret access key ID. To run tasks on Secure Agent installed on an EC2 system, you might leave the Secret Access Key blank. To run tasks on Secure Agent that is not installed on an EC2 system, you must provide the Secret Access Key.
Master Symmetric Key	Optional. Amazon S3 encryption key. Provide a 256-bit AES encryption key in the Base64 format.
Customer Master Key ID	Optional. Specify the customer master key ID or alias name generated by AWS Key Management Service (AWS KMS). You must generate the customer master key ID for the same region where Amazon S3 bucket reside. You can either specify the customer generated customer master key ID or the default customer master key ID.
JDBC URL	Amazon Redshift connection URL.
Number of bytes needed to support multibytes for varchar	Applicable to Create Target. Reads the Varchar precision of the source table and creates the target table with 1x/2x/3x/4x times of the source precision to successfully write multibyte characters in the target table. **Note:** You cannot create a target table if the Varchar precision exceeds 65535 that is maximum allowed.

Note: When you test a connection, Secure Agent validates Redshift connection. Validation of AWS Access key and AWS Secret key requires the Amazon S3 bucket name present in the advanced source and target properties. Therefore, Secure Agent validates AWS Access key and AWS Secret key when a synchronization or mapping task is run.

CHAPTER 14

Amazon Redshift sources and targets

This chapter includes the following topics:

Amazon Redshift sources

You can use an Amazon Redshift object as a source in a synchronization task, mapping, or mapping task. You can also use multiple related Amazon Redshift standard objects as sources in a synchronization task.

When you use Amazon Redshift source objects, you can select a standard object as the primary source, and then add child objects.

When you configure the advanced source properties, you configure properties specific to Amazon Redshift. You can encrypt data, retain the staging files on Amazon S3, and securely unload the results of a query to files on Amazon Redshift.

Amazon Redshift staging directory for Amazon Redshift sources

The agent creates a staging file in the directory that you specify in the source properties. The synchronization tasks, mapping and mapping tasks stage data in a staging directory before reading data from Amazon Redshift. The agent deletes the staging files from the staging directory when the task completes.

You cannot configure a directory on Hosted Agent. The Hosted Agent creates a directory to stage data at a temporary location and deletes the staging files from the temporary location when the task completes.

To improve task performance, enable compression for staging files. Specify a staging directory with an appropriate amount of disk space for the volume of data that you want to process. Specify a directory path that is available on each Secure Agent machine in the runtime environment.

The applications create subdirectories in the staging directory based on the time that the task runs. Subdirectories use the following naming convention:

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

Server-side encryption for Amazon Redshift sources

If you want Amazon Redshift to encrypt data while fetching the file from Amazon Redshift and staging the file to Amazon S3, you must enable server-side encryption.

You can configure the customer master key ID generated by AWS Key Management Service (AWS KMS) in the connection properties for server-side encryption. You must add IAM EC2 role and IAM Redshift role to the customer master key when you use IAM authentication and server-side encryption using customer master key.

If you select the server-side encryption in the advanced target properties, you must specify the customer master key ID in the connection properties.

Note: The staging files in the Amazon S3 are deleted after the task is complete.

Data encryption in Amazon Redshift sources

You can encrypt data using the customer master key ID generated by AWS Key Management Service (AWS KMS) for server-side encryption.

You can select the type of the encryption in the **Encryption Type** field under the Amazon Redshift advanced source properties on the **Schedule** page. 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:

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 the **SSE-KMS** 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 for server-side encryption.

The AWS KMS-managed customer master key specified 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.

If you enable the **Turn on S3 Client Side Encryption** property and select the **Encryption Type** as **SSE-S3**, the Amazon S3 encrypts the data using the master symmetric key for client-side encryption.

If you enable the **Turn on S3 Client Side Encryption** property and select the **Encryption Type** as **SSE-KMS**, the Amazon S3 encrypts the data using the customer master key ID generated by AWS Key Management Service for server-side encryption.

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

Client-side encryption for Amazon Redshift sources

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 sources, Amazon Redshift unloads the data in encrypted format, and then pushes the data to the Secure Agent. The Secure Agent writes the data to the target based on the task or mapping logic.

To enable client-side encryption, you must provide a master symmetric key in the connection properties and select the **Turn on S3 Client Side Encryption** option in the advanced target properties.

The Secure Agent encrypts the data by using the master symmetric key. The master symmetric key is a 256-bit AES encryption key in the Base64 format. Amazon Redshift Connector 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.

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 **UnloadOptions Property File** field. Enter the options in uppercase and delimit the options by using a new line. The Unload command has the following options and default values:

```
DELIMITER=\036
```

```
ESCAPE=OFF
```

```
PARALLEL=ON
```

```
AWS_IAM_ROLE=arn:aws:iam::<account ID>:role/<role-name>
```

When you run a task in the Secure Agent runtime environment, you can create a property file. The property file contains the Unload command options. Include the property file path in the **UnloadOptions Property File** field. For example:

```
C:\Temp\Redshift\unloadoptions.txt
```

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

If you run the Unload command as a pre-SQL or post-SQL command, specify the `ALLOWOVERWRITE` option to overwrite the existing objects.

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. You can delimit the data with a particular character or load data to multiple files in parallel.

To add options to the Unload command, use the **UnloadOptions Property File** option. You can set the following options:

DELIMITER

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

ESCAPE

You can add an escape character for CHAR and VARCHAR columns in delimited unload files before occurrences of the following characters:

- Linefeed \n
- Carriage return \r
- Delimiter character specified for the unloaded data
- Escape character \
- Single- or double-quote character

Default is OFF.

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.

AWS_IAM_ROLE

Specify the Amazon Redshift Role Resource Name (ARN) to run the task on agent 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`

ADDQUOTES

ADDQUOTES is implemented with the UNLOAD command by default. Do not specify the ADDQUOTES option in the advanced source properties. The Unload command adds quotation marks to each data field. With added quotation marks, the UNLOAD command can read data values that contain the delimiter. If double quote (") is a part of data, use ESCAPE to read the double quote as a regular character.

Partitioning

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

You can define the partition type as key range partitioning. Configure key range partitioning to partition Amazon Redshift data based on the value of a fields or set of fields. With key range partitioning, the Secure Agent distributes rows of source data based the fields that you define as partition keys. The Secure Agent compares the field value to the range values for each partition and sends rows to the appropriate partition.

Use key range partitioning for columns that have an even distribution of data values. Otherwise, the partitions might have unequal size. For example, a column might have 10 rows between key values 1 and 1000 and the column might have 999 rows between key values 1001 and 2000.

With key range partitioning, a query for one partition might return rows sooner than another partition. Or, one partition can return rows while the other partitions are not returning rows. This situation occurs when the rows in the table are in a similar order as the key range. One query might be reading and returning rows while the other queries are reading and filtering the same rows.

Amazon Redshift targets

You can use an Amazon Redshift object as a single target in a synchronization task, mapping, or mapping task. You can also create an Amazon Redshift target based on the input source. When you use Amazon Redshift target objects, you can select a standard object as the primary source.

You can insert, update, upsert, and delete data from Amazon Redshift targets. An update or insert task writes an entire batch to an Amazon Redshift target if no errors occur within the batch. If an error occurs within a batch, the Secure Agent writes the entire batch to the error rows file.

When you configure the advanced target properties, you configure properties specific to Amazon Redshift. You can encrypt data, update statistical metadata of the database tables to improve the efficiency of queries, load data into Amazon Redshift from flat files in an Amazon S3 bucket, and use vacuum tables to recover disk space and sort rows in tables.

If a mapping includes a flat file or an Amazon Redshift target, you can choose to use an existing target or create a new target at run time. You must specify Amazon Redshift target object names in lowercase letters.

Note: If the distribution key column in a target table contains null values and you configure a task with an upsert operation for the same target table, the task 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. You can use the Pre-SQL and Post-SQL properties to remove and then add the distribution key column in the target table.

Amazon Redshift staging directory for Amazon Redshift targets

The agent creates a staging file in the directory that you specify in the target properties. The synchronization tasks, mappings, and mapping tasks stage data in a staging directory before writing data to Amazon Redshift. You can configure the task to retain or delete staging files.

You cannot configure a directory on Hosted Agent. The Hosted Agent creates a directory to stage data at a temporary location and deletes the staging files from the temporary location when the task completes.

To improve task performance, enable compression for staging files. Specify a staging directory with an appropriate amount of disk space for the volume of data that you want to process. Specify a directory path that is available on each Secure Agent machine in the runtime environment.

The applications creates subdirectories in the staging directory based on the time that the task runs. Subdirectories use the following naming convention:

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

Analyze target table

To optimize query performance, you can configure a task 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.

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 Server Side Encryption as the encryption type in the advanced target properties on the **Schedule** page.

You can configure the customer master key ID generated by AWS Key Management Service (AWS KMS) in the connection properties for server-side encryption. You must add IAM EC2 role and IAM Redshift role to the customer master key when you use IAM authentication and server-side encryption using customer master key. If you select the server-side encryption in the advanced target properties and do not specify the customer master key ID in the connection properties, Amazon S3-managed encryption keys are used to encrypt data.

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 Secure Agent 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.

To enable client-side encryption, you must provide a master symmetric key in the connection properties and select the **Turn on S3 Client Side Encryption** option in the advanced target properties.

The Secure Agent encrypts the data by using the master symmetric key. The master symmetric key is a 256-bit AES encryption key in the Base64 format. Amazon Redshift Connector 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:

Retain staging files

You can retain staging files on Amazon S3 after the agent 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 create a target connection, you can configure a file prefix or directory prefix to save the staging files. After you provide the prefixes, the agent 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 agent 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 agent creates the following directories and files:
  - `<year>_<month>_<date>_<timestamp_inLong><3 digit of random number>00<ProcessID><PartitionId>`
  - `backup_file.batch_<batch_number>.csv.<file_number>.<encryption_if_applicable>`
- Do not provide a prefix. The agent does not save the staging files.

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 source to Amazon Redshift.

You can specify the Copy command options directly in the **CopyOptions Property File** field. Enter the options in uppercase and delimit the options by using a new line. The Copy command has the following options and default values:

```
DELIMITER=\036

ACCEPTINVCHARS=?

QUOTE=\037

COMPUPDATE=OFF

AWS_IAM_ROLE=arn:aws:iam::<account ID>:role/<role-name>
```

When you run a task in the Secure Agent runtime environment, you can create a property file. The property file contains the Copy command options. Include the property file path in the **CopyOptions Property File** 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.

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

DELIMITER

A single ASCII character to separate fields in the input file. You can use characters such as pipe (|), tilde (~), or a tab (\t). The delimiter must not be a part of the data. Default is \036, the octal representation of the non-printable character, record separator.

ACCEPTINVCHARS

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

QUOTE

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

COMPUPDATE

Overrides current compression encoding and applies compression to an empty table. Use the COMPUPDATE option in an insert operation 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.

Default is OFF.

AWS_IAM_ROLE

Specify the Amazon Redshift Role Resource Name (ARN) to run the task on agent 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`

Field mappings

The field mapping page displays key icons for primary key fields. When you configure field mappings, map all key fields and NOT NULL fields to successfully insert or upsert data to Amazon Redshift targets. Though Amazon Redshift enforces NOT NULL fields, it does not enforce key constraints.

The field mapping page displays key icons for primary key fields. Other Amazon Redshift key types are not marked. You must map a non-key field for update operation. If you use Amazon Redshift Identity fields in field mappings, map all available Identity fields or none. The Identity fields contain data that is automatically generated by Amazon Redshift.

You cannot map identity columns in a field map, if the identity column is not part of a key. If an identity column is part of a key, you must map the identity column in field map. However, you cannot set a value on the identity column from source.

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.

Reindex

Analyzes the distribution of the values in the interleaved sort key columns to configure the entire **Vacuum table** operations for a better performance.

Working with large tables

You can upload or download a large object as a set of multiple independent parts.

Amazon Redshift Connector uses the AWS TransferManager API to upload a large object in multiple parts to Amazon S3. While downloading a large object, the Secure Agent downloads the object in multiple parts from the Amazon S3.

When the file size is more than 5 MB, you can configure multipart upload to upload object in multiple parts in parallel. You can choose to download the object in multiple parts in parallel when the file size of an Amazon S3 object is greater than 12 MB.

You can configure **Enable Downloading S3 Files in Multiple Parts** option in the advanced source properties. You can configure the **Part Size** and **TransferManager Thread Pool Size** options in the advanced target properties.

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

Success and error files

The Secure Agent generates success and error files after you run a session. Success and error files are .csv files that contain row-level details. The Hosted Agent does not create success and error files after you run a session.

The Secure Agent generates a success file after you run a session. The success file contains an entry for each record that successfully writes into Amazon Redshift. Each entry contains the values that are written for all the fields of the record. Use this file to understand the data that the Secure Agent writes to the Amazon S3 bucket and then to the Amazon Redshift target.

The error file contains an entry for each data error. Each entry in the file contains the values for all fields of the record and the error message. Use the error file to understand why the Secure Agent does not write data to the Amazon Redshift target.

The Secure Agent 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 cannot access the error rows file from the **All Jobs** page. You can manually delete the files that you no longer need.

Consider the following guidelines when you configure the session properties for success files:

- By default, a success rows file is generated in the following directory: `<Secure Agent installation directory>/apps/Data_Integration_Server/data/success`. You can specify a different directory with the **Success File Directory** advanced target option.
- 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 session properties for error files:

- By default, an error rows file is generated in the following directory: `<Secure Agent installation directory>/apps/Data_Integration_Server/data/error`. You can specify a different directory with the **Error File Directory** advanced target option.
- When you define a error file directory, you can use the variable `$PMBadFileDir`. When you use the `$PMBadFileDir` variable, the application writes the file to the following Secure Agent directory: `<Secure Agent installation directory>/apps/Data_Integration_Server/data/error`.

- 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`.

CHAPTER 15

Synchronization tasks with Amazon Redshift

This chapter includes the following topics:

Amazon Redshift sources in synchronization tasks

You configure Amazon Redshift source properties on the **Source** page of the Synchronization Task wizard.

To optimize performance, you can configure a filter in the **Data Filters** page. Configure a simple filter or an advanced filter to remove rows at the source. You can improve efficiency by filtering early in the data flow. A simple filter includes a field name, operator, and value. Use an advanced filter to define a more complex filter condition, which can include multiple conditions using the AND or OR logical operators.

The following table describes the Amazon Redshift source properties:

Property	Description
Connection	Name of the source connection.
Source Type	Type of the source object. Select Single or Multiple.
Source Object	Name of the source object. Select the source object for a single source or multiple related sources.

When you configure a synchronization task to use an Amazon Redshift source, you can configure advanced source properties. Advanced source properties appear on the **Schedule** page of the Synchronization Task wizard.

The following table describes the Amazon Redshift advanced source properties:

Advanced Property	Description
S3 Bucket Name	Amazon S3 bucket name for the Amazon Redshift source data. Use an S3 bucket in the same region as your Amazon Redshift cluster.
Enable Compression	Compresses staging files before writing the files to Amazon Redshift. Task performance improves when the Secure Agent compresses the staging files. Default is selected.
Staging Directory Location	Amazon Redshift staging directory. When you run a task in Secure Agent runtime environment, specify a directory path that is available on each Secure Agent machine in the runtime environment. When you run a task in Hosted Agent runtime environment, leave the staging directory location blank. The Hosted Agent creates a directory at a temporary location.
UnloadOptions Property File	Unload command options. Add options to the Unload command to write data from an Amazon Redshift object to an S3 bucket. You can add the following options: - DELIMITER - PARALLEL - ESCAPE - AWS_IAM_ROLE When you run a task in the Secure Agent runtime environment, either specify the path of the property file that contains the unload options or specify the unload options directly in the **UnloadOptions Property File** field. When you run a task in the Hosted Agent runtime environment, specify options directly in the **UnloadOptions Property File** field.
Turn on S3 Client Side Encryption	Indicates that the Secure Agent encrypts data by using a private encryption key.
Encryption Type	Select the source encryption type. You can select from the following encryption types: - SSE-S3 - SSE-KMS Default is **SSE-S3**. For more information, see "Data encryption in Amazon Redshift sources" on page 94 .
Enable Downloading S3 Files in Multiple Parts	Downloads large Amazon S3 objects in multiple parts. When the file size of an Amazon S3 object is greater than 5 MB, you can choose to download the object in multiple parts in parallel.
Part Size	Specifies the part size of an object. Default is 5 MB.
Infa Advanced Filter	Not applicable for Amazon Redshift Connector.
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.
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.
SQL Query	Overrides the default query. Enclose column names in double quotes. The SQL query is case sensitive. Specify an SQL statement supported by the Amazon Redshift database.

Advanced Property	Description
Number of Sorted Ports	Number of columns used when sorting rows queried from the source. The agent adds an ORDER BY clause to the default query when it reads source rows. The ORDER BY clause includes the number of ports specified, starting from the top of the transformation. When you specify the number of sorted ports, the database sort order must match the session sort order. Default is 0.
Select Distinct	Selects unique values. The agent includes a SELECT DISTINCT statement if you choose this option. Amazon Redshift ignores trailing spaces. Therefore, the agent might extract fewer rows than expected.
Source Table Name	You can override the default source table name.

Amazon Redshift targets in synchronization tasks

You can use an Amazon Redshift object as a target in a synchronization task.

When you use Amazon Redshift target objects, you can select a standard object as the primary source.

You can configure Amazon Redshift target properties on the **Target** page of the Synchronization Task wizard.

The following table describes the Amazon Redshift target properties:

Property	Description
Connection	Name of the target connection.
Target Object	Name of the target object. Select the primary target object.
Create Target	Creates a target. Enter a name for the target object and select the source fields that you want to use. Default name is the source object name and by default, all source fields are used. Optionally, enter a file extension for the target object.

When you configure a synchronization task to use Amazon Redshift targets, you can configure advanced target properties.

The following table shows the Amazon Redshift advanced target properties:

Property	Description
S3 Bucket Name	Amazon S3 bucket name for the Amazon Redshift target data. Use an S3 bucket in the same region as your Amazon Redshift cluster.
Enable Compression	Compresses staging files before writing the files to Amazon Redshift. The performance of the synchronization task improves when the Secure Agent compresses the staging files. Default is selected.

Property	Description
Staging Directory Location	Amazon Redshift staging directory. When you run a task in Secure Agent runtime environment, specify a directory path that is available on each Secure Agent machine in the runtime environment. When you run a task in Hosted Agent runtime environment, leave the staging directory blank. The Hosted Agent creates a directory at a temporary location.
Batch Size	Minimum number of rows in a batch. Enter a number greater than 0. Default is 2000000.
Max Redshift Errors per Upload Batch for INSERT	Number of errors within a batch that causes a batch to fail. Enter a positive integer. If the number of errors is equal to or greater than the property value, the Secure Agent writes the entire batch to the error rows file. Default is 1.
Truncate Target Table Before Data Load	Truncates an Amazon Redshift target before writing data to the target.
Null value for CHAR and VARCHAR data types	String value used to represent null values in CHAR and VARCHAR fields in Amazon Redshift targets, such as NULL or a space character. Default is an empty string.
Wait time in seconds for file consistency on S3	Number of seconds to wait for the Secure Agent to make the staging files available. Default is 5.
CopyOptions Property File	Copy command options. Add options to the Copy command to write data from an Amazon S3 bucket to Amazon Redshift target. You can add the following options: - DELIMITER - ACCEPTINVCHARS - QUOTE - COMPUPDATE - AWS_IAM_ROLE When you run a task in the Secure Agent runtime environment, either specify the path of the property file that contains the copy options or specify the copy options directly in the **CopyOptions Property File** field. When you run a task in the Hosted Agent runtime environment, you must specify options directly in the **CopyOptions Property File** field.
Turn on S3 Server Side Encryption	Indicates that Amazon S3 encrypts data during upload and decrypts data at the time of access.
Turn on S3 Client Side Encryption	Indicates that the Secure Agent encrypts data by using a private encryption key. If you enable both server side and client side encryption, the runtime environment ignores the server side encryption.
Vacuum Target Table	Recovers disk space and sorts rows in a specified table or all tables in the database. You can select the following recovery options: - None - Full - Sort Only - Delete Only - Reindex Default is None.

Property	Description
Analyze Target Table	Improve the efficiency of the read and write operations. The query planner on Amazon Redshift updates the statistical metadata to build and choose optimal plans to improve the efficiency of queries.
Prefix for Retaining Staging files on S3	Retains staging files on Amazon S3. 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`.
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.
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.
Target Table Name	You can override the default target table name.
Part Size	Specifies the part size of an object. Default is 5 MB.
TransferManager Thread Pool Size	Specifies the number of the threads to write data in parallel. Default is 10.
Number of Files per Batch	Provide the number of files to calculate the number of the target staging file per batch. If you do not provide a value, the number of the target staging file is calculated internally.
Success File Directory	Directory for the Amazon Redshift success rows file. Specify a directory path that is available on each Secure Agent machine in the runtime environment. By default, Data Integration writes the success file to the following directory: `<Secure Agent installation directory>/apps/Data_Integration_Server/data/success` The Hosted Agent does not create a success rows file. Leave the **Success File Directory** field blank when you run a task in the Hosted Agent runtime environment.
Error File Directory	Directory for error rows file. Specify a directory path that is available on each Secure Agent machine in the runtime environment. By default, Data Integration writes the error rows file to the following directory: `<Secure Agent installation directory>/apps/Data_Integration_Server/data/error` When you specify the default error file directory you can download the error file from the schedule tab. When the Error File Directory is other than the default error directory, you cannot download the error file from the Schedule tab. You must go to the specified directory to retrieve the error file. The Hosted Agent does not create an error rows file. Leave the **Error File Directory** field blank when you run a task in the Hosted Agent runtime environment.

Amazon Redshift lookups in synchronization tasks

When you configure field mappings in a synchronization task, you can create an uncached lookup to an Amazon Redshift object. Use the JDBC URL specified in the connection properties to create an uncached lookup.

Note: Amazon Redshift Connector does not support un-connected lookup transformation.

Synchronization task example

You work for an e-commerce organization that stores sales order details in a MySQL database. Your organization needs to move the data from the MySQL database to an Amazon Redshift target.

Configure a synchronization task to write to Amazon Redshift.

You perform the following synchronization tasks:

Define the synchronization task.

Configure a synchronization task to use the insert operation.

Use a MySQL source object.

The source for the mapping is a MySQL connection that contains the sales order details. The MySQL object is a single source in the synchronization task. You can include the Customer ID, Item_codes, Item_quantity, and Price columns. Specify *sales_order_details* as the resource for the source object.

Create an Amazon Redshift target object.

Select the fields *Customer_ID*, *Item_codes*, *Item_quantity*, and *Price* from the source object that you want to insert into the target object. Provide a name *sales_order_details* for the target object and specify the connection type as MySQL. The synchronization task writes the data to Amazon Redshift. You can also use an existing target object.

Configure a field mapping.

Map all the fields under *sales_order_details* source data to all the fields in the target *sales_order_details*. The synchronization application writes the mapped source data to Amazon Redshift.

Configure the advanced target properties.

In the advanced target properties, you choose properties that are specific to Amazon Redshift. Specify an Amazon S3 bucket name for the Amazon Redshift target data. Use an S3 bucket in the same region as your Amazon Redshift cluster. You can also specify options for the copy command, and turn on server side and client side encryption. Click **Save** and **Finish** the task.

Open Amazon Redshift to visualize the exported data.

Schedule the task.

You can schedule the task for each requirement and save. You can select the synchronization task from the **Explore** page and run the task. In **Monitor**, you can monitor the status of the logs after you run the task.

CHAPTER 16

Mappings and mapping tasks with Amazon Redshift

This chapter includes the following topics:

Amazon Redshift objects in mappings

When you create a mapping, you can configure a Source or Target transformation to represent an Amazon Redshift object.

Amazon Redshift sources in mappings

In a mapping, you can configure a Source transformation to represent a single Amazon Redshift source or multiple Amazon Redshift sources.

You can use multiple related Amazon Redshift standard objects as a source. You can select a standard object as the primary source, then you add one or more child objects.

The following table describes the Amazon Redshift source properties that you can configure in a Source transformation:

Property	Description
Connection	Name of the source connection.
Source type	Type of the source object. Select Single Object, Multiple Objects, Query, or Parameter.
Object	Name of the source object. Select the source object for a single source.

The following table describes the Amazon Redshift query options that you can configure in a Source transformation:

Property	Description
Filter	Filter value in a read operation. Click **Configure** to add conditions to filter records and reduce the number of rows that the Secure Agent reads from the source. You can specify the following filter conditions: - **Not parameterized**. Use a basic filter to specify the object, field, operator, and value to select specific records. - **Completely parameterized**. Use a parameter to represent the field mapping. - **Advanced**. Use an advanced filter to define a more complex filter condition that uses the Amazon Redshift query format.
Sort	Not applicable.

The following table describes the Amazon Redshift source advanced properties that you can configure in a Source transformation:

Advanced Property	Description
S3 Bucket Name	Amazon S3 bucket name for the Amazon Redshift target data. Use an S3 bucket in the same region as your Amazon Redshift cluster.
Enable Compression	Compresses staging files before writing the files to Amazon Redshift. Task performance improves when the runtime environment compresses the staging files. Default is selected.
Staging Directory Location	Amazon Redshift staging directory. When you run a task in Secure Agent runtime environment, specify a directory path that is available on each Secure Agent machine in the runtime environment. When you run a task in Hosted Agent runtime environment, leave the staging directory blank. The Hosted Agent creates a directory at a temporary location.
UnloadOptions Property File	Unload command options. Add options to the unload command to write data from an Amazon Redshift object to an S3 bucket. You can add the following options: - DELIMITER - PARALLEL - ESCAPE - AWS_IAM_ROLE When you run a task in the Secure Agent runtime environment, either specify the path of the property file that contains the unload options or specify the unload options directly in the **UnloadOptions Property File** field. When you run a task in the Hosted Agent runtime environment, specify options directly in the **UnloadOptions Property File** field.
Turn on S3 Client Side Encryption	Indicates that the Secure Agent encrypts data by using a private encryption key.
Encryption Type	Select the source encryption type. You can select from the following encryption types: - SSE-S3 - SSE-KMS Default is **SSE-S3**.

Advanced Property	Description
Enable Downloading S3 Files in Multiple Parts	Downloads large Amazon S3 objects in multiple parts. When the file size of an Amazon S3 object is greater than 5 MB, you can choose to download the object in multiple parts in parallel.
Part Size	Specifies the part size of an object. Default is 5 MB.
Infa Advanced Filter	Not applicable for Amazon Redshift Connector.
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.
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.
SQL Query	Overrides the default query. Enclose column names in double quotes. The SQL query is case sensitive. Specify an SQL statement supported by the Amazon Redshift database.
Number of Sorted Ports	Number of columns used when sorting rows queried from the source. The agent adds an ORDER BY clause to the default query when it reads source rows. The ORDER BY clause includes the number of ports specified, starting from the top of the transformation. When you specify the number of sorted ports, the database sort order must match the session sort order. Default is 0.
Select Distinct	Selects unique values. The agent includes a SELECT DISTINCT statement if you choose this option. Amazon Redshift ignores trailing spaces. Therefore, the agent might extract fewer rows than expected.
Source Table Name	You can override the default source table name.
Tracing Level	Sets the amount of detail that appears in the log file. You can choose terse, normal, verbose initialization, or verbose data. Default is normal.

Configuring key range partitioning

Configure key range partitioning to partition Amazon Redshift data based on field values.

1. In **Source Properties**, click the **Partitions** tab.
2. Select the required **Partition Key** from the list.
3. Click **Add New key Range** to add partitions.
4. Specify the **Start range** and **End range**.

Amazon Redshift targets in mappings

In a mapping, you can configure a Target transformation to represent a single Amazon Redshift target. You can also create an Amazon Redshift target at runtime based on the input fields.

When you use an Amazon Redshift target object, select a standard object as the primary target, and then add a child object. You can use a custom object as a single target.

The following table describes the Amazon Redshift target properties that you can configure in a Target transformation:

Property	Description
Connection	Name of the target connection.
Target Type	Type of the target object. Select Single Object or Parameter.
Object	Name of the target object. Target object for a single target.
Operation	Target operation. Select Insert, Update, Upsert, or Delete.
Create Target	Creates a target. Enter a name for the target object and select the source fields that you want to use. Default name is the source object name and by default, all source fields are used. Optionally, enter a file extension for the target object.

The following table describes the Amazon Redshift target advanced properties that you can configure in a Target transformation:

Property	Description
S3 Bucket Name	Amazon S3 bucket name for the Amazon Redshift target data. Use an S3 bucket in the same region as your Amazon Redshift cluster.
Enable Compression	Compresses staging files before writing the files to Amazon Redshift. Task performance improves when the runtime environment compresses the staging files. Default is selected.
Staging Directory Location	Amazon Redshift staging directory. For Secure Agent runtime environment, specify a directory path that is available on each Secure Agent machine in the runtime environment. For Hosted Agent runtime environment, leave the staging directory blank. The Hosted Agent creates a directory at a temporary location.
Batch Size	Minimum number of rows in a batch. Enter a number greater than 0. Default is 2000000.
Max Redshift Errors per Upload Batch for INSERT	Number of errors within a batch that causes a batch to fail. Enter a positive integer. If the number of errors is equal to or greater than the property value, the runtime environment writes the entire batch to the error rows file. Default is 1.
Truncate Target Table Before Data Load	Truncates an Amazon Redshift target before writing data to the target.
Null value for CHAR and VARCHAR data types	String value used to represent null values in CHAR and VARCHAR fields in Amazon Redshift targets, such as NULL or a space character. Default is an empty string.

Property	Description
Wait time in seconds for file consistency on S3	Number of seconds to wait for the runtime environment to make the staging files available. Default is 5.
CopyOptions Property File	Copy command options. Add options to the Copy command to write data from an Amazon S3 bucket to Amazon Redshift target. You can add the following options: - DELIMITER - ACCEPTINVCHARS - QUOTE - COMPUPDATE - AWS_IAM_ROLE When you run a task in the Secure Agent runtime environment, either specify the path of the property file that contains the copy options or specify the copy options directly in the **CopyOptions Property File** field. When you run a task in the Hosted Agent runtime environment, you must specify options directly in the **CopyOptions Property File** field.
Turn on S3 Server Side Encryption	Indicates that Amazon S3 encrypts data during upload and decrypts data at the time of access.
Turn on S3 Client Side Encryption	Indicates that the runtime environment encrypts data by using a private encryption key. If you enable both server side and client side encryption, the runtime environment ignores the server side encryption.
Vacuum Target Table	Recovers disk space and sorts rows in a specified table or all tables in the database. You can select the following recovery options: - None - Full - Sort Only - Delete Only - Reindex Default is None.
Prefix for Retaining Staging Files on S3	Retains staging files on Amazon S3. 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`.
Analyze Target Table	Improve the efficiency of the read and write operations. The query planner on Amazon Redshift updates the statistical metadata to build and choose optimal plans to improve the efficiency of queries.
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.
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.
Target Table Name	You can override the default target table name.

Property	Description
Part Size	Specifies the part size of an object. Default is 5 MB.
TransferManager Thread Pool Size	Specifies the number of the threads to write data in parallel. Default is 10.
Number of Files per Batch	Provide the number of files to calculate the number of the target staging file per batch. If you do not provide a value, the number of the target staging file is calculated internally.
Success File Directory	Directory for the Amazon Redshift success rows file. Specify a directory path that is available on each Secure Agent machine in the runtime environment. By default, Data Integration writes the success file to the following directory: `<Secure Agent installation directory>/apps/Data_Integration_Server/data/success` The Hosted Agent does not create a success rows file. Leave the **Success File Directory** field blank when you run a task in the Hosted Agent runtime environment.
Error File Directory	Directory for the Amazon Redshift error rows file. Directory for error rows file. Specify a directory path that is available on each Secure Agent machine in the runtime environment. By default, Data Integration writes the error rows file to the following directory: `<Secure Agent installation directory>/apps/Data_Integration_Server/data/error` When you specify the default error file directory you can download the error file from the schedule tab. When the Error File Directory is other than the default error directory, you cannot download the error file from the Schedule tab. You must go to the specified directory to retrieve the error file. The Hosted Agent does not create an error rows file. Leave the **Error File Directory** field blank when you run a task in the Hosted Agent runtime environment.
Forward Rejected Rows	Determines whether the transformation passes rejected rows to the next transformation or drops rejected rows. By default, the mapping application forwards rejected rows to the next transformation.

When you edit a target task, selecting a different Amazon Redshift connection clears the advanced target properties. Enter the S3 bucket name and other advanced properties applicable to the selected Amazon Redshift connection.

Amazon Redshift lookups in mappings

In a mapping, you can configure a Lookup transformation to represent an Amazon Redshift object.

When you use an Amazon Redshift object as a lookup, you need to configure the Amazon S3 bucket name in Amazon Redshift properties.

When you use a cache lookup with Amazon Redshift connection, the lookup condition is ignored if the lookup condition contains a NULL value.

Note: Amazon Redshift Connector does not support un-connected lookup transformation.

Amazon Redshift objects in template-based mapping tasks

When you configure a mapping task based on an integration template, you can configure advanced properties for Amazon Redshift sources and targets.

Amazon Redshift sources in mapping tasks

For Amazon Redshift source connections used in template-based mapping tasks, you can configure advanced properties in the **Sources** page of the Mapping Task wizard.

You can configure the following advanced properties:

Advanced Property	Description
S3 Bucket Name	Amazon S3 bucket name for the Amazon Redshift target data. Use an S3 bucket in the same region as your Amazon Redshift cluster.
Enable Compression	Compresses staging files before writing the files to Amazon Redshift. Task performance improves when the runtime environment compresses the staging files. Default is selected.
Staging Directory Location	Amazon Redshift staging directory. When you run a task in Secure Agent runtime environment, specify a directory path that is available on each Secure Agent machine in the runtime environment. When you run a task in Hosted Agent runtime environment, leave the staging directory blank. The Hosted Agent creates a directory at a temporary location.
UnloadOptions Property File	Unload command options. Add options to the unload command to write data from an Amazon Redshift object to an S3 bucket. You can add the following options: - DELIMITER - PARALLEL - ESCAPE - AWS_IAM_ROLE When you run a task in the Secure Agent runtime environment, either specify the path of the property file that contains the unload options or specify the unload options directly in the **UnloadOptions Property File** field. When you run a task in the Hosted Agent runtime environment, specify options directly in the **UnloadOptions Property File** field.
Turn on S3 Client Side Encryption	Indicates that the Secure Agent encrypts data by using a private encryption key.
Encryption Type	Select the source encryption type. You can select from the following encryption types: - SSE-S3 - SSE-KMS Default is **SSE-S3** For more information, see "Data encryption in Amazon Redshift sources" on page 94 .
Enable Downloading S3 Files in Multiple Parts	Downloads large Amazon S3 objects in multiple parts. When the file size of an Amazon S3 object is greater than 5 MB, you can choose to download the object in multiple parts in parallel.

Advanced Property	Description
Part Size	Specifies the part size of an object. Default is 5 MB.
Infa Advanced Filter	Not applicable for Amazon Redshift Connector.
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.
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.
SQL Query	Overrides the default query. Enclose column names in double quotes. The SQL query is case sensitive. Specify an SQL statement supported by the Amazon Redshift database.
Number of Sorted Ports	Number of columns used when sorting rows queried from the source. The agent adds an ORDER BY clause to the default query when it reads source rows. The ORDER BY clause includes the number of ports specified, starting from the top of the transformation. When you specify the number of sorted ports, the database sort order must match the session sort order. Default is 0.
Select Distinct	Selects unique values. The agent includes a SELECT DISTINCT statement if you choose this option. Amazon Redshift ignores trailing spaces. Therefore, the agent might extract fewer rows than expected.
Source Table Name	You can override the default source table name.
Tracing Level	Sets the amount of detail that appears in the log file. You can choose terse, normal, verbose initialization, or verbose data. Default is normal.

Amazon Redshift targets in mapping tasks

For Amazon Redshift target connections used in template-based mapping tasks, you can configure advanced properties in the **Targets** page of the Mapping Task wizard.

You can configure the following advanced properties:

Property	Description
S3 Bucket Name	Amazon S3 bucket name for the Amazon Redshift target data. Use an S3 bucket in the same region as your Amazon Redshift cluster.
Enable Compression	Compresses staging files before writing the files to Amazon Redshift. Task performance improves when the runtime environment compresses the staging files. Default is selected.
Staging Directory Location	Amazon Redshift staging directory. Specify a directory on the machine that hosts the runtime environment.
Batch Size	Minimum number of rows in a batch. Enter a number greater than 0. Default is 2000000.

Property	Description
Max Redshift Errors per Upload Batch for INSERT	Number of errors within a batch that causes a batch to fail. Enter a positive integer. If the number of errors is equal to or greater than the property value, the runtime environment writes the entire batch to the error rows file. Default is 1.
Truncate Target Table Before Data Load	Truncates an Amazon Redshift target before writing data to the target.
Null value for CHAR and VARCHAR data types	String value used to represent null values in CHAR and VARCHAR fields in Amazon Redshift targets, such as NULL or a space character. Default is an empty string.
Wait time in seconds for file consistency on S3	Number of seconds to wait for the runtime environment to make the staging files available. Default is 5.
CopyOptions Property File	Path to the property file. Enables you to add options to the copy command to write data from Amazon S3 to an Amazon Redshift target. You can add the following options: - DELIMITER - ACCEPTINVCHARS - QUOTE - COMPUPDATE When you run a task in the Secure Agent runtime environment, either specify the path of the property file that contains the copy options or specify the copy options directly in the **CopyOptions Property File** field. When you run a task in the Hosted Agent runtime environment, you must specify options directly in the **CopyOptions Property File** field.
Turn on S3 Server Side Encryption	Indicates that Amazon S3 encrypts data during upload and decrypts data at the time of access.
Turn on S3 Client Side Encryption	Indicates that the runtime environment encrypts data by using a private encryption key. If you enable both server side and client side encryption, the runtime environment ignores the server side encryption.
Vacuum Target Table	Recovers disk space and sorts rows in a specified table or all tables in the database. You can select the following recovery options: - None - Full - Sort Only - Delete Only - Reindex Default is None.
Analyze Target Table	Improve the efficiency of the read and write operations. The query planner on Amazon Redshift updates the statistical metadata to build and choose optimal plans to improve the efficiency of queries.
Prefix for Retaining Staging Files on S3	Retains staging files on Amazon S3. 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`.

Property	Description
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.
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.
Target Table Name	You can override the default target table name.
Part Size	Specifies the part size of an object. Default is 5 MB.
TransferManager Thread Pool Size	Specifies the number of the threads to write data in parallel. Default is 10.
Number of Files per Batch	Provide the number of files to calculate the number of the target staging file per batch. If you do not provide a value, the number of the target staging file is calculated internally.
Success File Directory	Directory for the Amazon Redshift success rows file. Specify a directory path that is available on each Secure Agent machine in the runtime environment. By default, Data Integration writes the success file to the following directory: `<Secure Agent installation directory>/apps/Data_Integration_Server/data/success` The Hosted Agent does not create a success rows file. Leave the **Success File Directory** field blank when you run a task in the Hosted Agent runtime environment.
Error File Directory	Directory for the Amazon Redshift error rows file. Directory for error rows file. Specify a directory path that is available on each Secure Agent machine in the runtime environment. By default, Data Integration writes the error rows file to the following directory: `<Secure Agent installation directory>/apps/Data_Integration_Server/data/error` When you specify the default error file directory you can download the error file from the schedule tab. When the Error File Directory is other than the default error directory, you cannot download the error file from the Schedule tab. You must go to the specified directory to retrieve the error file. The Hosted Agent does not create an error rows file. Leave the **Error File Directory** field blank when you run a task in the Hosted Agent runtime environment.

CHAPTER 17

Amazon Redshift pushdown optimization

This chapter includes the following topics:

Amazon Redshift pushdown optimization overview

You can use pushdown optimization to push transformation logic to source databases or target databases. Use pushdown optimization when using database resources can improve task performance.

When you run a task configured for pushdown optimization, the task converts the transformation logic to an SQL query. The task sends the query to the database, and the database executes the query.

Amazon Redshift Connector supports **Full** and **Source** pushdown optimization for the ODBC connection type that uses Amazon ODBC Redshift drivers for mapping.

Note: Amazon Redshift does not support upsert operation in a full 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 Web Services and stores its data in Amazon Redshift, a petabytescale 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 Amazon Redshift to the business intelligence service. You can use Amazon Redshift Connector to read data from Amazon Redshift. To read this large amount of data, you can use source pushdown for the ODBC connection type. Using the ODBC connection type with pushdown optimization enhances the performance.

Pushdown optimization supported functions and transformations

The following table summarizes the availability of pushdown functions in an Amazon Redshift database. Columns marked with an X indicate that the function can be pushed to the Amazon Redshift database by using source-side or full pushdown optimization. Columns marked with S indicate that the function can be pushed to the Amazon Redshift database only by using source-side pushdown optimization. Columns marked with a dash (-) symbol indicate that the function cannot be pushed to the database.

Function	Pushdown	Function	Pushdown	Function	Pushdown
ABORT()	-	INSTR()	X	REG_REPLACE	-
ABS()	X	IS_DATE()	-	REPLACECHR()	-
ADD_TO_DATE()	X	IS_NUMBER()	-	REPLACESTR()	-
AES_DECRYPT()	-	IS_SPACES()	-	REVERSE()	-
AES_ENCRYPT()	-	ISNULL()	S	ROUND(DATE)	-
ASCII()	-	LAST()	-	ROUND(NUMBER)	X
AVG()	S	LAST_DAY()	X	RPAD()	X
CEIL()	X	LEAST()	-	RTRIM()	X
CHOOSE()	-	LENGTH()	X	SET_DATE_PART()	-
CHRCODE()	-	LN()	X	SIGN()	X
COMPRESS()	-	LOG()	-	SIN()	X
CONCAT()	X	LOOKUP	-	SINH()	-
COS()	X	LOWER()	X	SOUNDEX()	-
COSH()	-	LPAD()	X	SQRT()	X
COUNT()	S	LTRIM()	X	STDDEV()	S
CRC32()	-	MAKE_DATE_TIME()	-	SUBSTR()	X
CUME()	-	MAX()	S	SUM()	S
DATE_COMPARE()	X	MD5()	-	SYSTIMESTAMP()	S
DATE_DIFF()	X	MEDIAN()	-	TAN()	S
DECODE()	X	METAPHONE()	-	TANH()	-
DECODE_BASE64()	-	MIN()	S	TO_BIGINT	X
DECOMPRESS()	-	MOD()	S	TO_CHAR(DATE)	S

Function	Pushdown	Function	Pushdown	Function	Pushdown
ENCODE_BASE64()	-	MOVINGAVG()	-	TO_CHAR(NUMBER)	X
EXP()	X	MOVINGSUM()	-	TO_DATE()	X
FIRST()	-	NPER()	-	TO_DECIMAL()	X
FLOOR()	X	PERCENTILE()	-	TO_FLOAT()	X
FV()	-	PMT()	-	TO_INTEGER()	X
GET_DATE_PART()	X	POWER()	X	TRUNC(DATE)	S
GREATEST()	-	PV()	-	TRUNC(NUMBER)	S
IIF()	X	RAND()	-	UPPER()	X
IN()	S	RATE()	-	VARIANCE()	S
INDEXOF()	-	REG_EXTRACT()	-		
INITCAP()	X	REG_MATCH()	-		

The following table lists the pushdown operators that can be used in an Amazon Redshift database. Columns marked with an X indicate that the operator can be pushed to the Amazon Redshift database by using source-side, or full pushdown optimization.

Operator	Pushdown
+	X
-	X
*	X
/	X
%	X
||	X
>	X
=	X
>=	X
<=	X
!=	X
AND	X

Operator	Pushdown
OR	X
NOT	X
^=	X

The following table lists the transformation logic that the Secure Agent can push to an Amazon Redshift source or target:

Transformations	Pushdown
Aggregator	Source, Full
Expression	Source, Full
Filter	Source, Full
Joiner	Source, Full
Sorter	Source, Full
Union	Source, Full
Router	Full

Configuring Amazon Redshift ODBC connection

Amazon Redshift supports Amazon ODBC Redshift drivers on Windows and Linux systems. You must install the Amazon ODBC Redshift 64-bit driver based on your system requirement.

Note: Informatica certifies Amazon Redshift ODBC driver version, `AmazonRedshiftODBC-64-bit-1.4.8.1000-1.x86_64`, to use for pushdown optimization.

Configuring Amazon Redshift ODBC connection on Windows

Before you establish an ODBC connection to connect to Amazon Redshift on Windows, you must configure the ODBC connection.

Perform the following steps to configure an ODBC connection on Windows:

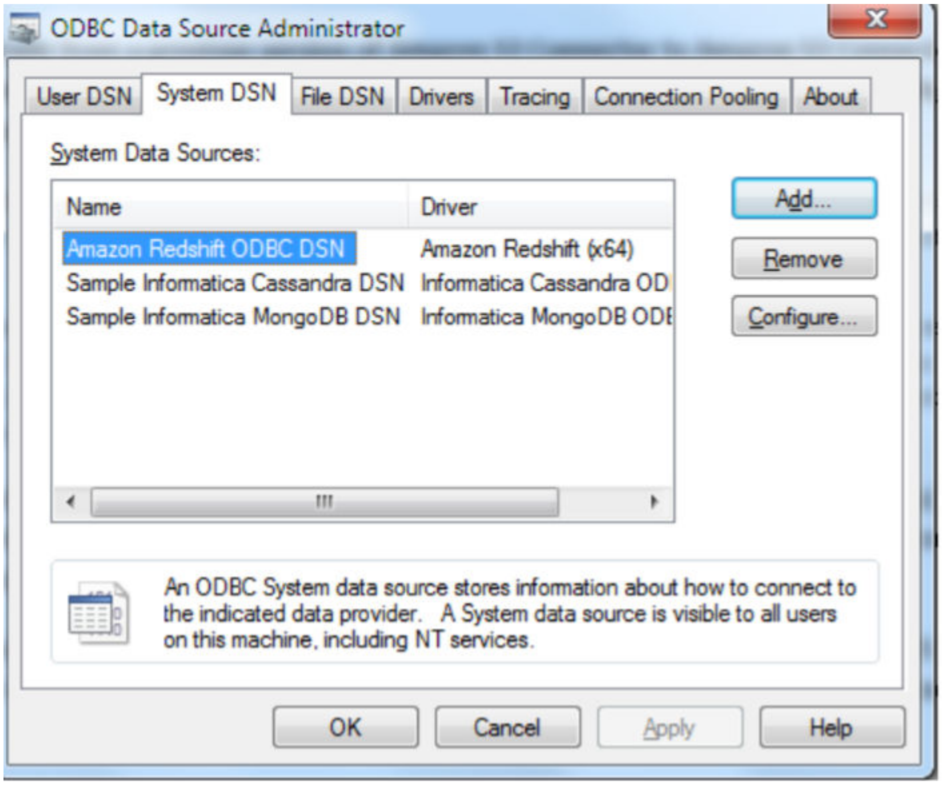
1. Download the Amazon Redshift ODBC drivers from the AWS website.

   You must download the Amazon Redshift ODBC 64-bit driver.
2. Install the Amazon Redshift ODBC drivers on the machine where the Secure Agent is installed.
3. Open the folder in which ODBC data source file is installed.
4. Run the `odbcad32.exe` file.

   The **ODBC Data Source Administrator** dialog box appears.

5. Click **System DSN**.

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

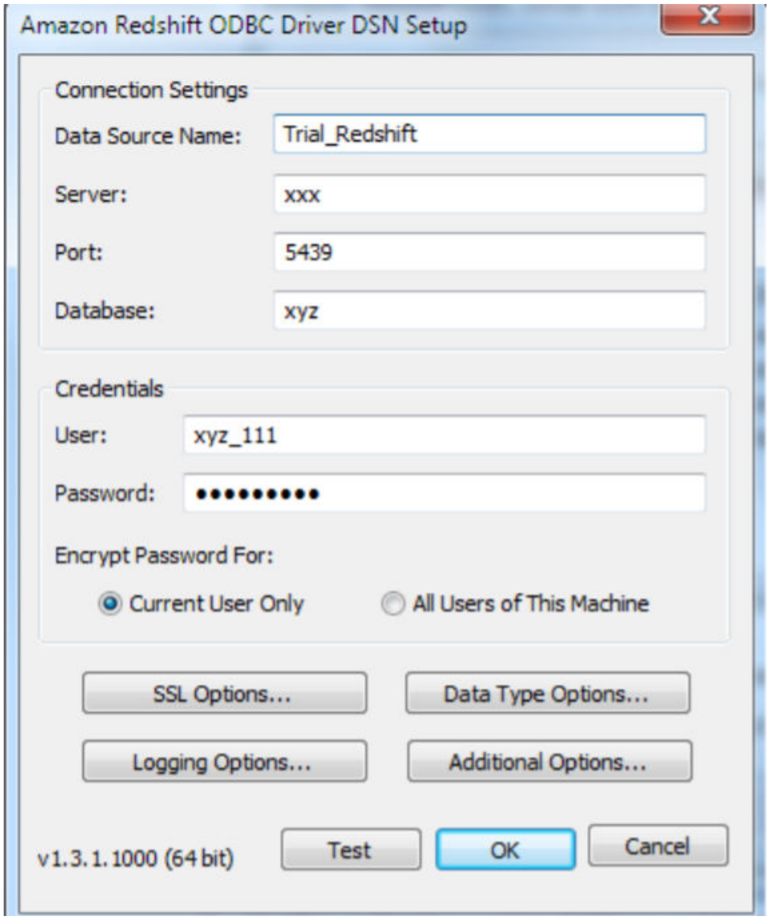


Name	Driver
Amazon Redshift ODBC DSN	Amazon Redshift (x64)
Sample Informatica Cassandra DSN	Informatica Cassandra OD
Sample Informatica MongoDB DSN	Informatica MongoDB ODI

An ODBC System data source stores information about how to connect to the indicated data provider. A System data source is visible to all users on this machine, including NT services.

6. Click **Configure**.

The **Amazon Redshift ODBC Driver DSN Setup** dialog box displays. The following image shows the **Amazon Redshift ODBC Driver DSN Setup** dialog box where you can configure the **Connection Settings** and **Credentials** section:



Amazon Redshift ODBC Driver DSN Setup

Connection Settings

Data Source Name: Trial_Redshift

Server: xxx

Port: 5439

Database: xyz

Credentials

User: xyz_111

Password: ●●●●●●●●●

Encrypt Password For:

● Current User Only ○ All Users of This Machine

7. 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.

8. Specify the following credentials in the **Credentials** section:

Property	Description
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: - **Current User Only** - **All Users of This Machine** Default is **Current User Only**.

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

The Amazon Redshift ODBC connection is configured successfully on Windows.

After you configure the Amazon Redshift ODBC connection, you must create an ODBC connection to connect to Amazon Redshift.

For more information about how to create an ODBC connection to connect to Amazon Redshift, see "Creating an ODBC connection" on page 75

Configuring Amazon Redshift ODBC connection on Linux

Before you establish an ODBC connection to connect to Amazon Redshift on Linux, you must configure the ODBC connection.

Perform the following steps to configure an ODBC connection on Linux:

1. Download the Amazon Redshift ODBC drivers from the AWS website.

   You must download the Amazon Redshift ODBC 64-bit driver.
2. Install the Amazon Redshift ODBC drivers on the machine where the Secure Agent is installed.
3. 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
```

4. 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.

Property	Description
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.

5. Run the following command to export the `odbc.ini` file.

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

6. Restart the Secure Agent.

   The Amazon Redshift ODBC connection on Linux is configured successfully.

After you configure the Amazon Redshift ODBC connection, you must create an ODBC connection to connect to Amazon Redshift.
For more information about how to create an ODBC connection to connect to Amazon Redshift, see "Creating an ODBC connection" on page 75

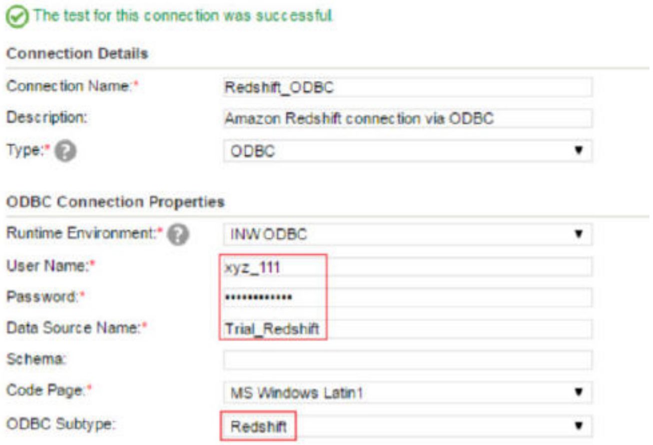
Creating an ODBC connection

You must create an ODBC connection to connect to Amazon Redshift after you configure the ODBC connection.

Perform the following steps to create an Amazon Redshift ODBC connection on the **Connections** page:

1. In Administrator, click **Connections**.
2. In the upper right corner, click **New Connections**.

   The **New Connection** page appears. The following image shows the **New Connection** page:



The test for this connection was successful

Connection Details

Connection Name:* Redshift_ODBC

Description: Amazon Redshift connection via ODBC

Type:* ODBC

ODBC Connection Properties

Runtime Environment:* INW ODBC

User Name:* xyz_111

Password:* ••••••••••••

Data Source Name:* Trial_Redshift

Schema:

Code Page:* MS Windows Latin1

ODBC Subtype: Redshift

3. Configure the following connection details in the **Connection Details** section:

Property	Description
Connection Name	Name of the ODBC connection.
Description	Description of the connection.
Type	Type of the connection. Select the type of the connection as **ODBC**.

4. Configure the following connection details in the **Connection Properties** section:

Property	Description
Runtime Environment	The name of the runtime environment where you want to run the tasks.
User Name	User name of the Amazon Redshift account.
Password	Password for the Amazon Redshift account.
Data Source Name	Enter the name of the ODBC data source name that you created for the Amazon Redshift database.
Schema	Amazon Redshift schema name.
Code Page	Select the code page that the Secure Agent must use to read or write data.
ODBC Subtype	Enter the value of the **ODBC Subtype** field as **Redshift**.

The Amazon Redshift ODBC connection is created successfully.

Cross-Schema pushdown optimization

You can configure cross-schema pushdown optimization for a mapping task that uses a Amazon Redshift ODBC connection to read or write data to Amazon Redshift objects of different schemas in the same database.

To use cross-schema pushdown optimization, create Amazon Redshift ODBC connections and specify the schema for the source and target connections. The source and target schemas must be different but must belong to the same database. Configure pushdown optimization for the mapping task and enable cross-schema pushdown optimization in the advanced session properties. By default, the **Enable cross-schema pushdown optimization** check box is selected.

Configuring cross-schema optimization for an Amazon Redshift mapping task

Perform the following steps to configure cross-schema pushdown optimization for an Amazon Redshift mapping task:

1. Create Amazon Redshift ODBC source and target connections, each defined with a different schema.

   For example,

   - Create a `rs_odbc1` Amazon Redshift ODBC connection and specify `CQA_SCHEMA1` schema in the connection properties.
   - Create a `rs_odbc2` Amazon Redshift ODBC connection and specify `CQA_SCHEMA2` schema in the connection properties.

2. Create an Amazon Redshift mapping.

   For example, create a `m_rs_pdo_crossSchema` Amazon Redshift mapping.

3. Add a Source transformation. Include an Amazon Redshift source object and connection to read data using the schema specified in the connection.

   For example, add a Source transformation. Include an Amazon Redshift source object and connection `rs_odbc1` to read data using `CQA_SCHEMA1`.

4. Add a Target transformation. Include an Amazon Redshift target object and connection to write data using the schema specified in the connection.

   For example, add a Target transformation. Include an Amazon Redshift target object and connection `rs_odbc2` to write data using `CQA_SCHEMA2`.

5. Create an Amazon Redshift mapping task, and perform the following tasks:

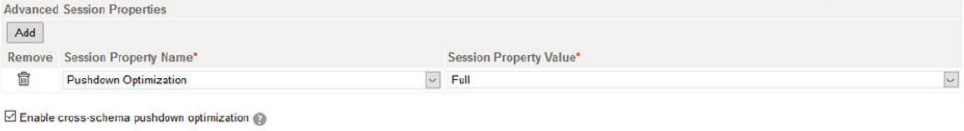
   a. Select the configured Amazon Redshift mapping.

      For example, select the `m_rs_pdo_crossSchema` Amazon Redshift mapping.

   b. In the **Advanced Options** on the **Schedule** tab, add **Pushdown Optimization** and set the value to **Full**.

   c. Select **Enable cross-schema pushdown optimization**.

      The following image shows the configured **Enable cross-schema pushdown optimization** property:



Advanced Session Properties

Add

Remove	Session Property Name*	Session Property Value*
🗑	Pushdown Optimization	Full

☑ Enable cross-schema pushdown optimization

   d. Save the task and click **Finish**.

   When you run the mapping task, the Secure Agent reads data from the Amazon Redshift source object associated with the `CQA_SCHEMA1` schema and writes data to the Amazon Redshift target object associated with `CQA_SCHEMA2` schema.

Rules and guidelines for functions in pushdown optimization

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

- To push TRUNC(DATE) to Amazon Redshift, you must define the date and format arguments. Otherwise, the agent does not push the function to Amazon Redshift.

- The aggregator functions for Amazon Redshift accept only one argument, a field set for the aggregator 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.
- Do not specify a format for SYSTIMESTAMP() to push the SYSTIMESTAMP to Amazon Redshift. The Amazon Redshift database returns the complete time stamp.
- 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 agent 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 agent 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
  - DDD
  - HH24
  - MI
  - MM
  - MS
  - SS
  - US
  - YYYY

CHAPTER 18

Data type reference

This chapter includes the following topics:

Data type reference overview

Data Integration uses the following data types in synchronization tasks, mappings, and mapping tasks with Amazon Redshift:

Amazon Redshift Native Data Types

Amazon Redshift data types appear in the Source and Target transformations when you choose to edit metadata for the fields.

Transformation Data Types

Set of data types that appear in the transformations. They are internal data types based on ANSI SQL-92 generic data types, which the runtime environment uses to move data across platforms. Transformation data types appear in all transformations in synchronization tasks, mappings, and mapping tasks.

When Data Integration reads source data, it converts the native data types to the comparable transformation data types before transforming the data. When Data Integration writes to a target, it converts the transformation data types to the comparable native data types.

Amazon Redshift and transformation data types

The following table lists the Amazon Redshift data types that the runtime environment supports and the corresponding transformation data types:

Amazon Redshift Data Type	Transformation Data Type	Description
Bigint	Bigint	Signed eight-byte integer.
Boolean	Small Integer	Logical Boolean (true/false).

Amazon Redshift Data Type	Transformation Data Type	Description
Char	String	Fixed-length character string.
Date	Timestamp	Calendar date (year, month, day).
Decimal	Decimal	Exact numeric of selectable precision.
Double Precision	Double	Double precision floating-point number.
Integer	Integer	Signed four-byte integer.
Real	Double	Single precision floating-point number.
Smallint	Small Integer	Signed two-byte integer.
Timestamp	Timestamp	Date and time (without time zone).
Varchar	String	Variable-length character string with a user-defined limit.

CHAPTER 19

Troubleshooting

This chapter includes the following topics:

Troubleshooting overview

Use the following sections to troubleshoot errors in Amazon Redshift Connector.

Troubleshooting for Amazon Redshift Connector

How to solve the task failure issue when you use ODBC connection to connect to Amazon Redshift to read UTF characters and Secure Agent is installed on Linux?

For information about the issue, see https://kb.informatica.com/solution/23/Pages/62/516325.aspx?myk=516325

What must be the maximum size of the local staging area, when the compression option for an Amazon Redshift connection to perform a read or write operation is enabled?

For information about the issue, see https://kb.informatica.com/faq/7/Pages/16/497540.aspx?myk=497540

How to configure AWS IAM Authentication for Amazon Redshift Connector?

For information about configuring AWS IAM authentication, see https://kb.informatica.com/h2l/HowTo%20Library/1/0972-ConfiguringAWSIAMforAmazonRedshiftandAmazonRedshiftV2Connectors-H2L.pdf

Does the result of an Amazon Redshift task vary based on whether you map the target field that contains identity and primary key to the same column or different column?

For information about the issue, see https://kb.informatica.com/faq/7/Pages/21/535693.aspx?myk=535693

Troubleshooting Amazon Redshift connection

When you run a task to write data to an Amazon Redshift, the task fails with the following error:

```
Amazon_RedshiftWriter_30007 [ERROR] Copy command on record 'public.basic_data_types'
failed due to [ERROR: S3ServiceException:The bucket you are attempting to access
must be addressed using the specified endpoint. Please send all future requests to
this endpoint.,Status 301,Error PermanentRedirect,Rid A8BA401CC765AC53,ExtRid
NAbd1uxKirJVjDas1zo3WONdQ/+6p674RYkO
```

This issue occurs because the Amazon Redshift user and cluster in the connection properties are in a different region from the S3 bucket in the task.

You must configure the task to use an S3 bucket in the same region as the user and cluster in the connection. You can also use a different connection to write to the S3 bucket.

INDEX