



Informatica™

Informatica® Cloud Data Intergration  
Spring 2018 April

# Amazon Redshift Connector Guide

© Copyright Informatica LLC 2017, 2019

This software and documentation are provided only under a separate license agreement containing restrictions on use and disclosure. No part of this document may be reproduced or transmitted in any form, by any means (electronic, photocopying, recording or otherwise) without prior consent of Informatica LLC.

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation is subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License.

Informatica, the Informatica logo, Informatica Cloud, and PowerCenter are trademarks or registered trademarks of Informatica LLC in the United States and many jurisdictions throughout the world. A current list of Informatica trademarks is available on the web at <https://www.informatica.com/trademarks.html>. Other company and product names may be trade names or trademarks of their respective owners.

Portions of this software and/or documentation are subject to copyright held by third parties. Required third party notices are included with the product.

See patents at <https://www.informatica.com/legal/patents.html>.

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](mailto: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: 2019-02-04

# Table of Contents

<b>Chapter 1: Introduction to Amazon Redshift Connector.....</b>	<b>5</b>
Amazon Redshift Connector Overview. . . . .	5
Amazon Redshift Supported Task Types and Object Types. . . . .	6
Introduction to Amazon Redshift. . . . .	6
Amazon Redshift Connector Example. . . . .	6
Administration of Amazon Redshift Connector. . . . .	6
Configure Amazon Redshift for SSL. . . . .	7
Create Minimal Amazon S3 Bucket Policy. . . . .	8
IAM Authentication. . . . .	9
<b>Chapter 2: Amazon Redshift Connections.....</b>	<b>10</b>
Amazon Redshift Connections Overview. . . . .	10
Amazon Redshift connection properties. . . . .	10
<b>Chapter 3: Amazon Redshift Sources and Targets.....</b>	<b>12</b>
Amazon Redshift Sources. . . . .	12
Amazon Redshift Staging Directory for Amazon Redshift Sources. . . . .	12
Server-side Encryption for Amazon Redshift Sources. . . . .	13
Client-side Encryption for Amazon Redshift Sources. . . . .	14
Unload Command. . . . .	14
Partitioning. . . . .	15
Amazon Redshift Targets. . . . .	16
Amazon Redshift Staging Directory for Amazon Redshift Targets. . . . .	16
Analyze Target Table. . . . .	17
Data Encryption in Amazon Redshift Targets. . . . .	17
Retain Staging Files. . . . .	18
Copy Command. . . . .	18
Field Mappings. . . . .	20
Vacuum Tables. . . . .	20
Working with Large Tables. . . . .	20
Octal Values as DELIMITER and QUOTE. . . . .	21
Success and Error Files. . . . .	21
<b>Chapter 4: Amazon Redshift Pushdown Optimization.....</b>	<b>23</b>
Amazon Redshift Pushdown Optimization Overview. . . . .	23
Pushdown Optimization Supported Functions and Transformations. . . . .	24
Configuring Amazon Redshift ODBC Connection. . . . .	26
Configuring Amazon Redshift ODBC Connection on Windows. . . . .	26
Configuring Amazon Redshift ODBC Connection on Linux. . . . .	29
Creating an ODBC Connection. . . . .	30

Cross-Schema Pushdown Optimization. . . . .	31
Configuring Cross-Schema Optimization for an Amazon Redshift Mapping Task. . . . .	32
Rules and Guidelines for Functions in Pushdown Optimization. . . . .	32
<b>Chapter 5: Synchronization Tasks with Amazon Redshift. . . . .</b>	<b>34</b>
Amazon Redshift Sources in Synchronization Tasks. . . . .	34
Amazon Redshift Targets in Synchronization Tasks. . . . .	36
Amazon Redshift Lookups in Synchronization Tasks. . . . .	39
Synchronization Task Example. . . . .	39
<b>Chapter 6: Mappings and Mapping Tasks with Amazon Redshift. . . . .</b>	<b>40</b>
Amazon Redshift Objects in Mappings. . . . .	40
Amazon Redshift Sources in Mappings. . . . .	40
Amazon Redshift Targets in Mappings. . . . .	42
Amazon Redshift Lookups in Mappings. . . . .	45
Amazon Redshift Objects in Template-Based Mapping Tasks. . . . .	46
Amazon Redshift Sources in Mapping Tasks. . . . .	46
Amazon Redshift Targets in Mapping Tasks. . . . .	47
<b>Chapter 7: Data Type Reference. . . . .</b>	<b>50</b>
Data Type Reference Overview. . . . .	50
Amazon Redshift and Transformation Data Types. . . . .	50
<b>Chapter 8: Troubleshooting. . . . .</b>	<b>52</b>
Troubleshooting Overview. . . . .	52
Troubleshooting for Amazon Redshift Connector. . . . .	52
Troubleshooting Amazon Redshift Connection. . . . .	53
<b>Index. . . . .</b>	<b>54</b>

# CHAPTER 1

## Introduction to Amazon Redshift Connector

This chapter includes the following topics:

- [Amazon Redshift Connector Overview, 5](#)
- [Amazon Redshift Supported Task Types and Object Types, 6](#)
- [Introduction to Amazon Redshift, 6](#)
- [Amazon Redshift Connector Example, 6](#)
- [Administration of Amazon Redshift Connector, 6](#)

## 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 that you use Amazon Redshift V2 Connector if you want to create a mass ingestion task to transfer files from any source that mass ingestion task supports to an Amazon Redshift target. For more information about using Amazon Redshift V2 Connector, see the Amazon Redshift V2 Connector documentation.

# Amazon Redshift Supported Task Types and Object Types

The following table lists the tasks and object types that Amazon Redshift Connector supports:

Task Type	Source	Target
Synchronization	Yes	Yes
Mapping	Yes	Yes

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.

## 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 9](#)
- 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.

## 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 (Beijing)
- 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:

- Step 1: Create Minimal Amazon S3 Bucket Policy. For more information, see [“Create Minimal Amazon S3 Bucket Policy” on page 8](#).
- Step 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.
- Step 3: Create an EC2 instance. Assign the Amazon EC2 role that you created in step #2 to the EC2 instance.
- Step 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.
- Step 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.
- Step 6: Install Secure Agent on the EC2 system.

## CHAPTER 2

# Amazon Redshift Connections

This chapter includes the following topics:

- [Amazon Redshift Connections Overview, 10](#)
- [Amazon Redshift connection properties, 10](#)

## 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. <b>Note:</b> 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 3

# Amazon Redshift Sources and Targets

This chapter includes the following topics:

- [Amazon Redshift Sources, 12](#)
- [Amazon Redshift Targets, 16](#)
- [Working with Large Tables, 20](#)
- [Octal Values as DELIMITER and QUOTE, 21](#)
- [Success and Error Files, 21](#)

## 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<MMddHHmssSSS+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.

### Encryption Type

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. 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. To enable client-side encryption, perform the following tasks:

1. Provide the master symmetric key when you create an Amazon Redshift connection. Ensure that you provide a 256-bit AES encryption key in Base64 format.
2. Update the security policy JAR files on each Secure Agent machine in the runtime environment. Update the `local_policy.jar` and the `US_export_policy.jar` files in the following directory: <Secure Agent installation directory>\jre\lib\security. You can download the JAR files supported by your JAVA environment from the Oracle website.
3. On the **Schedule** page, select **Client Side Encryption** as the encryption type in the advanced source properties.

## 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.

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

To enable client-side encryption, you must provide a master symmetric key in the connection properties. The 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:

1. Provide the master symmetric key when you create an Amazon Redshift connection. Ensure that you provide a 256-bit AES encryption key in Base64 format.
2. Update the security policy JAR files on each Secure Agent machine in the runtime environment.

Update the `local_policy.jar` and the `US_export_policy.jar` files in the following directory: <Secure Agent installation directory>\jre\lib\security. You can download the JAR files for your JAVA environment from the Oracle website.

3. On the **Schedule** page, select **Client Side Encryption** as the encryption type in the advanced target properties.

## 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 ACCEPTINVCHARS, 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.

### **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 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 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 4

# Amazon Redshift Pushdown Optimization

This chapter includes the following topics:

- [Amazon Redshift Pushdown Optimization Overview, 23](#)
- [Pushdown Optimization Supported Functions and Transformations, 24](#)
- [Configuring Amazon Redshift ODBC Connection, 26](#)
- [Cross-Schema Pushdown Optimization, 31](#)
- [Rules and Guidelines for Functions in Pushdown Optimization, 32](#)

## 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 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()	-
AES_DECRYPT()	-	IS_NUMBER()	-	REPLACESTR()	-
AES_ENCRYPT()	-	IS_SPACES()	-	REVERSE()	-
ASCII()	-	ISNULL()	S	ROUND(DATE)	-
AVG()	S	LAST()	-	ROUND(NUMBER)	X
CEIL()	X	LAST_DAY()	X	RPAD()	X
CHOOSE()	-	LEAST()	-	RTRIM()	X
CHR()	X	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()	-	SYSTEMSTAMP()	S
DATE_DIFF()	X	MEDIAN()	-	TAN()	S
DECODE()	X	METAPHONE()	-	TANH()	-
DECODE_BASE64()	-	MIN()	S	TO_BIGINT	X



Function	Pushdown	Function	Pushdown	Function	Pushdown
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
IN()	S	RATE()	-	VARIANCE()	S
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 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

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.

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.

### 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:

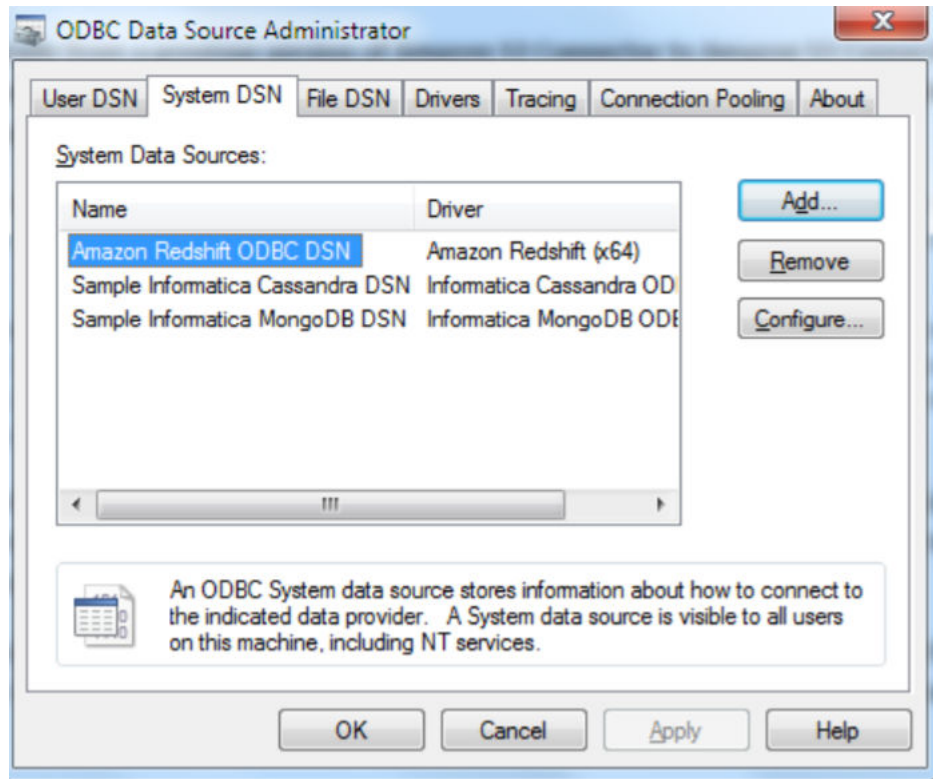
1. Download the Amazon Redshift ODBC drivers from the AWS website.  
You must download the Amazon Redshift ODBC 64-bit driver.
2. Install the Amazon Redshift ODBC drivers on the machine where the Secure Agent is installed.

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.

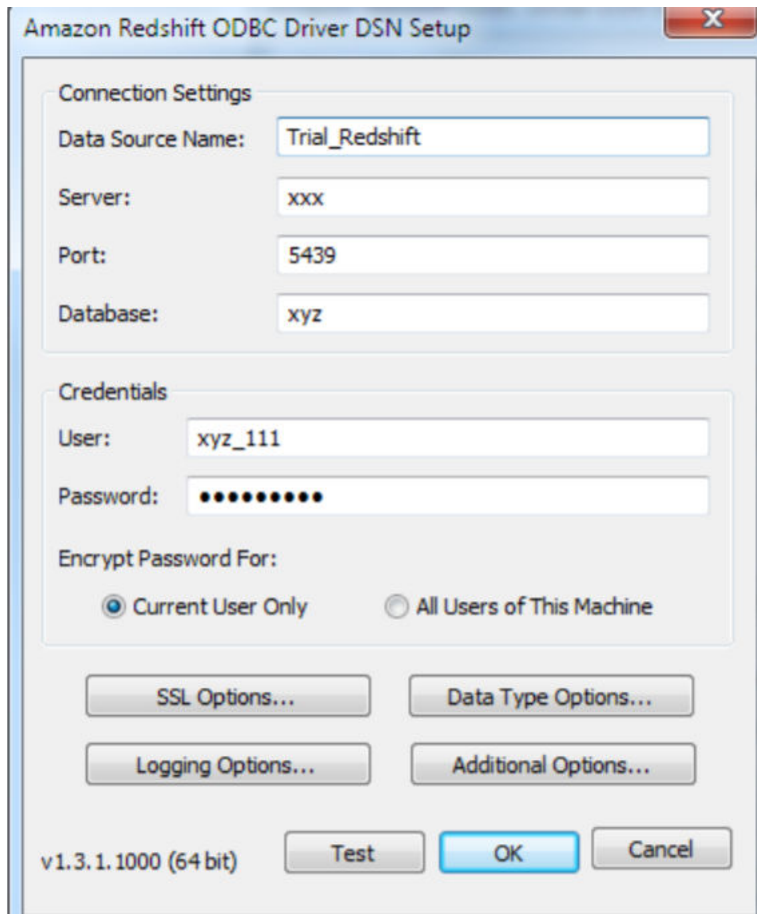
5. Click **System DSN**.

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



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:



- 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.

- 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: - <b>Current User Only</b> - <b>All Users of This Machine</b> Default is <b>Current User Only</b> .

- Click **Test** to test the connection in the **Amazon Redshift ODBC Driver DSN Setup** box.
- 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 30](#)

## 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:

- Download the Amazon Redshift ODBC drivers from the AWS website.  
You must download the Amazon Redshift ODBC 64-bit driver.
- Install the Amazon Redshift ODBC drivers on the machine where the Secure Agent is installed.
- 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
```

- 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.

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

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

- 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 30](#)

## 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:

- In Administrator, click **Connections**.
- 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:

Description:

Type:

**ODBC Connection Properties**

Runtime Environment:

User Name:

Password:

Data Source Name:

Schema:

Code Page:

ODBC Subtype:

- 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 <b>ODBC</b> .

- 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 <b>ODBC Subtype</b> field as <b>Redshift</b> .

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:

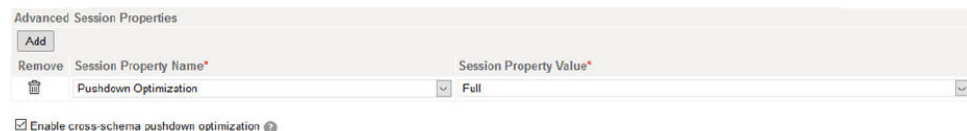
- a. Select the configured Amazon Redshift mapping.

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:



- 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 5

# Synchronization Tasks with Amazon Redshift

This chapter includes the following topics:

- [Amazon Redshift Sources in Synchronization Tasks, 34](#)
- [Amazon Redshift Targets in Synchronization Tasks, 36](#)
- [Amazon Redshift Lookups in Synchronization Tasks, 39](#)
- [Synchronization Task Example, 39](#)

## 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: <ul style="list-style-type: none"> <li>- DELIMITER</li> <li>- PARALLEL</li> <li>- ESCAPE</li> <li>- AWS_IAM_ROLE</li> </ul> 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 <b>UnloadOptions Property File</b> field. When you run a task in the Hosted Agent runtime environment, specify options directly in the <b>UnloadOptions Property File</b> 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: <ul style="list-style-type: none"> <li>- SSE-S3</li> <li>- SSE-KMS</li> </ul> Default is <b>SSE-S3</b> . For more information, see <a href="#">"Encryption Type" on page 13</a> .
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.
Infra Advanced Filter	SQL filter command to divide the source database into multiple segments.
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	<p>Amazon Redshift staging directory.</p> <p>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.</p> <p>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.</p>
Batch Size	<p>Minimum number of rows in a batch. Enter a number greater than 0.</p> <p>Default is 2000000.</p>
Max Redshift Errors per Upload Batch for INSERT	<p>Number of errors within a batch that causes a batch to fail. Enter a positive integer.</p> <p>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.</p> <p>Default is 1.</p>
Truncate Target Table Before Data Load	<p>Truncates an Amazon Redshift target before writing data to the target.</p>
Null value for CHAR and VARCHAR data types	<p>String value used to represent null values in CHAR and VARCHAR fields in Amazon Redshift targets, such as NULL or a space character.</p> <p>Default is an empty string.</p>
Wait time in seconds for file consistency on S3	<p>Number of seconds to wait for the Secure Agent to make the staging files available.</p> <p>Default is 5.</p>
CopyOptions Property File	<p>Copy command options.</p> <p>Add options to the Copy command to write data from an Amazon S3 bucket to Amazon Redshift target. You can add the following options:</p> <ul style="list-style-type: none"> <li>- DELIMITER</li> <li>- ACCEPTINVCHARS</li> <li>- QUOTE</li> <li>- COMPUPDATE</li> <li>- AWS_IAM_ROLE</li> </ul> <p>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 <b>CopyOptions Property File</b> field.</p> <p>When you run a task in the Hosted Agent runtime environment, you must specify options directly in the <b>CopyOptions Property File</b> field.</p>
Turn on S3 Server Side Encryption	<p>Indicates that Amazon S3 encrypts data during upload and decrypts data at the time of access.</p>
Turn on S3 Client Side Encryption	<p>Indicates that the Secure Agent encrypts data by using a private encryption key.</p> <p>If you enable both server side and client side encryption, the runtime environment ignores the server side encryption.</p>
Vacuum Target Table	<p>Recovers disk space and sorts rows in a specified table or all tables in the database.</p> <p>You can select the following recovery options:</p> <ul style="list-style-type: none"> <li>- None</li> <li>- Full</li> <li>- Sort Only</li> <li>- Delete Only</li> <li>- Reindex</li> </ul> <p>Default is None.</p>

Property	Description
Analyze Target Table	<p>Improve the efficiency of the read and write operations.</p> <p>The query planner on Amazon Redshift updates the statistical metadata to build and choose optimal plans to improve the efficiency of queries.</p>
Prefix for Retaining Staging files on S3	<p>Retains staging files on Amazon S3.</p> <p>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, <code>backup_dir/backup_file</code> or <code>backup_file</code>.</p>
Pre-SQL	<p>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.</p>
Post-SQL	<p>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.</p>
Target Table Name	<p>You can override the default target table name.</p>
Part Size	<p>Specifies the part size of an object.</p> <p>Default is 5 MB.</p>
TransferManager Thread Pool Size	<p>Specifies the number of the threads to write data in parallel.</p> <p>Default is 10.</p>
Number of Files per Batch	<p>Provide the number of files to calculate the number of the target staging file per batch.</p> <p>If you do not provide a value, the number of the target staging file is calculated internally.</p>
Success File Directory	<p>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: <code>&lt;Secure Agent installation directory&gt;/apps/Data_Integration_Server/data/success</code></p> <p>The Hosted Agent does not create a success rows file. Leave the <b>Success File Directory</b> field blank when you run a task in the Hosted Agent runtime environment.</p>
Error File Directory	<p>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: <code>&lt;Secure Agent installation directory&gt;/apps/Data_Integration_Server/data/error</code></p> <p>When you specify the default error file directory you can download the error file from the schedule tab.</p> <p>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.</p> <p>The Hosted Agent does not create an error rows file. Leave the <b>Error File Directory</b> field blank when you run a task in the Hosted Agent runtime environment.</p>

# 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.

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

## 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 6

# Mappings and Mapping Tasks with Amazon Redshift

This chapter includes the following topics:

- [Amazon Redshift Objects in Mappings, 40](#)
- [Amazon Redshift Objects in Template-Based Mapping Tasks, 46](#)

## 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	<p>Filter value in a read operation. Click <b>Configure</b> to add conditions to filter records and reduce the number of rows that the Secure Agent reads from the source.</p> <p>You can specify the following filter conditions:</p> <ul style="list-style-type: none"> <li>- <b>Not parameterized.</b> Use a basic filter to specify the object, field, operator, and value to select specific records.</li> <li>- <b>Completely parameterized.</b> Use a parameter to represent the field mapping.</li> <li>- <b>Advanced.</b> Use an advanced filter to define a more complex filter condition that uses the Amazon Redshift query format.</li> </ul>
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	<p>Amazon S3 bucket name for the Amazon Redshift target data.</p> <p>Use an S3 bucket in the same region as your Amazon Redshift cluster.</p>
Enable Compression	<p>Compresses staging files before writing the files to Amazon Redshift.</p> <p>Task performance improves when the runtime environment compresses the staging files.</p> <p>Default is selected.</p>
Staging Directory Location	<p>Amazon Redshift staging directory.</p> <p>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.</p> <p>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.</p>
UnloadOptions Property File	<p>Unload command options.</p> <p>Add options to the unload command to write data from an Amazon Redshift object to an S3 bucket. You can add the following options:</p> <ul style="list-style-type: none"> <li>- DELIMITER</li> <li>- PARALLEL</li> <li>- ESCAPE</li> <li>- AWS_IAM_ROLE</li> </ul> <p>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 <b>UnloadOptions Property File</b> field.</p> <p>When you run a task in the Hosted Agent runtime environment, specify options directly in the <b>UnloadOptions Property File</b> field.</p>
Turn on S3 Client Side Encryption	<p>Indicates that the Secure Agent encrypts data by using a private encryption key.</p>
Encryption Type	<p>Select the source encryption type. You can select from the following encryption types:</p> <ul style="list-style-type: none"> <li>- SSE-S3</li> <li>- SSE-KMS</li> </ul> <p>Default is <b>SSE-S3</b> For more information, see <a href="#">"Encryption Type" on page 13</a> .</p>

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.
Infra Advanced Filter	SQL filter command to divide the source database into multiple segments.
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 200000.
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: <ul style="list-style-type: none"> <li>- DELIMITER</li> <li>- ACCEPTINVCHARS</li> <li>- QUOTE</li> <li>- COMPUPDATE</li> <li>- AWS_IAM_ROLE</li> </ul> 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 <b>CopyOptions Property File</b> field. When you run a task in the Hosted Agent runtime environment, you must specify options directly in the <b>CopyOptions Property File</b> 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: <ul style="list-style-type: none"> <li>- None</li> <li>- Full</li> <li>- Sort Only</li> <li>- Delete Only</li> <li>- Reindex</li> </ul> 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, <code>backup_dir/backup_file</code> or <code>backup_file</code> .
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: <code>&lt;Secure Agent installation directory&gt;/apps/Data_Integration_Server/data/success</code> The Hosted Agent does not create a success rows file. Leave the <b>Success File Directory</b> 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: <code>&lt;Secure Agent installation directory&gt;/apps/Data_Integration_Server/data/error</code> 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 <b>Error File Directory</b> 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.

For more information about, see "Lookup Transformation" in the *PowerCenter Transformation Guide*.

# 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: <ul style="list-style-type: none"><li>- DELIMITER</li><li>- PARALLEL</li><li>- ESCAPE</li><li>- AWS_IAM_ROLE</li></ul> 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 <b>UnloadOptions Property File</b> field. When you run a task in the Hosted Agent runtime environment, specify options directly in the <b>UnloadOptions Property File</b> 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: <ul style="list-style-type: none"><li>- SSE-S3</li><li>- SSE-KMS</li></ul> Default is <b>SSE-S3</b> . For more information, see <a href="#">"Encryption Type" on page 13</a> .
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.
Infra Advanced Filter	SQL filter command to divide the source database into multiple segments.
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	<p>Path to the property file.</p> <p>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:</p> <ul style="list-style-type: none"> <li>- DELIMITER</li> <li>- ACCEPTINVCHARS</li> <li>- QUOTE</li> <li>- COMPUPDATE</li> </ul> <p>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 <b>CopyOptions Property File</b> field.</p> <p>When you run a task in the Hosted Agent runtime environment, you must specify options directly in the <b>CopyOptions Property File</b> field.</p>
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	<p>Recovers disk space and sorts rows in a specified table or all tables in the database. You can select the following recovery options:</p> <ul style="list-style-type: none"> <li>- None</li> <li>- Full</li> <li>- Sort Only</li> <li>- Delete Only</li> <li>- Reindex</li> </ul> <p>Default is None.</p>
Analyze Target Table	<p>Improve the efficiency of the read and write operations.</p> <p>The query planner on Amazon Redshift updates the statistical metadata to build and choose optimal plans to improve the efficiency of queries.</p>
Prefix for Retaining Staging Files on S3	<p>Retains staging files on Amazon S3.</p> <p>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, <code>backup_dir/backup_file</code> or <code>backup_file</code>.</p>



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 <b>Success File Directory</b> 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 <b>Error File Directory</b> field blank when you run a task in the Hosted Agent runtime environment.

# CHAPTER 7

## Data Type Reference

This chapter includes the following topics:

- [Data Type Reference Overview, 50](#)
- [Amazon Redshift and Transformation Data Types, 50](#)

### 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 8

## Troubleshooting

This chapter includes the following topics:

- [Troubleshooting Overview, 52](#)
- [Troubleshooting for Amazon Redshift Connector, 52](#)
- [Troubleshooting Amazon Redshift Connection, 53](#)

### 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 NAbdluxKirJVjDas1zo3WONdQ/+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

## A

- administration
  - IAM authentication [9](#)
  - minimal Amazon S3 bucket policy [8](#)
- Amazon Redshift
  - connection properties [10](#)
  - mapping tasks [46](#)
  - mappings [40](#)
  - pushdown optimization overview [23](#)
  - SSL configuration [7](#)
  - troubleshooting connection [53](#)
- Amazon Redshift connections
  - administration [6](#)
- Amazon Redshift Connections
  - overview [10](#)
- Amazon Redshift Data Types
  - data types [50](#)
- Amazon Redshift ODBC connection
  - configuration [26](#)
- Amazon Redshift sources
  - mapping [40](#)
  - server-side encryption [13](#)
  - staging directory [12](#)
- Amazon Redshift targets
  - mappings [42](#)
  - staging directory [16](#)

## C

- Configuring Amazon Redshift ODBC connection
  - Linux [29](#)
- Configuring Amazon Redshift ODBC Connection
  - Windows [26](#)
- connections
  - Amazon Redshift [10](#)
- Creating an ODBC connection [30](#)
- cross-schema
  - pushdown optimization [31](#)

## D

- data type reference
  - overview [50](#)

## E

- encryption type [13](#)

## F

- field mapping
  - Amazon Redshift lookups [39](#)

## I

- IAM authentication
  - administration [9](#)
- introduction
  - amazon redshift introduction [6](#)

## M

- Mapping task
  - target properties [47](#)
  - targets [47](#)
  - source properties [46](#)
  - sources [46](#)
- mappings
  - Amazon Redshift [40](#)
  - Amazon Redshift lookups [45](#)

## O

- octal values
  - DELIMITER [21](#)
  - QUOTE [21](#)

## P

- pushdown optimization
  - functions [24](#)
  - transformations [24](#)

## S

- synchronization
  - example [39](#)
- Synchronization task
  - source [34](#)
  - source properties [34](#)
  - target [36](#)
  - target properties [36](#)
- synchronization tasks
  - Amazon Redshift lookups [39](#)