Informatica® Cloud Data Integration

ODBC Connector

Informatica Cloud Data Integration ODBC Connector
April 2024

Table of Contents

Preface

Use *ODBC Connector* to learn how to read from or write to ODBC-compliant databases. Learn to create an ODBC connection, develop mappings, and run mapping, synchronization, and data transfer tasks. Learn how to configure SQL ELT optimization , cross-schema SQL ELT optimization, and cross-database SQL ELT optimization using an ODBC connection.

Informatica Resources

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

Informatica Documentation

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

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

Informatica Intelligent Cloud Services web site

You can access the Informatica Intelligent Cloud Services web site at http://www.informatica.com/cloud. This site contains information about Informatica Cloud integration services.

Informatica Intelligent Cloud Services Communities

Use the Informatica Intelligent Cloud Services Community to discuss and resolve technical issues. You can also find technical tips, documentation updates, and answers to frequently asked questions.

Access the Informatica Intelligent Cloud Services Community at:

https://network.informatica.com/community/informatica-network/products/cloud-integration

Developers can learn more and share tips at the Cloud Developer community:

https://network.informatica.com/community/informatica-network/products/cloud-integration/cloud-developers

Informatica Intelligent Cloud Services Marketplace

Visit the Informatica Marketplace to try and buy Data Integration Connectors, templates, and mapplets:

https://marketplace.informatica.com/

Data Integration connector documentation

You can access documentation for Data Integration Connectors at the Documentation Portal. To explore the Documentation Portal, visit https://docs.informatica.com.

Informatica Knowledge Base

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

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

Informatica Intelligent Cloud Services Trust Center

The Informatica Intelligent Cloud Services Trust Center provides information about Informatica security policies and real-time system availability.

You can access the trust center at https://www.informatica.com/trust-center.html.

Subscribe to the Informatica Intelligent Cloud Services Trust Center to receive upgrade, maintenance, and incident notifications. The Informatica Intelligent Cloud Services Status page displays the production status of all the Informatica cloud products. All maintenance updates are posted to this page, and during an outage, it will have the most current information. To ensure you are notified of updates and outages, you can subscribe to receive updates for a single component or all Informatica Intelligent Cloud Services components. Subscribing to all components is the best way to be certain you never miss an update.

To subscribe, on the Informatica Intelligent Cloud Services Status page, click **SUBSCRIBE TO UPDATES**. You can choose to receive notifications sent as emails, SMS text messages, webhooks, RSS feeds, or any combination of the four.

Informatica Global Customer Support

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

To find online support resources on the Informatica Network, click **Contact Support** in the Informatica Intelligent Cloud Services Help menu to go to the **Cloud Support** page. The **Cloud Support** page includes system status information and community discussions. Log in to Informatica Network and click **Need Help** to find additional resources and to contact Informatica Global Customer Support through email.

The telephone numbers for Informatica Global Customer Support are available from the Informatica web site at https://www.informatica.com/services-and-training/support-services/contact-us.html.

CHAPTER 1

Introduction to ODBC Connector

You can use ODBC Connector to read data from and write data to any application that is ODBC compliant. You can also use ODBC Connector to read from or write data to Oracle Database Cloud Service.

Use an ODBC connection in mappings synchronization tasks, and mapping tasks to connect to sources, targets and lookups. You can write data to a flat file target.

You can switch mappings to advanced mode to include transformations and functions that enable advanced functionality.

Example

You want to migrate sales data from DB2 to Salesforce. You do not have a DB2 Connector to read sales data from DB2 source. You can use ODBC Connector to migrate sales data from DB2 to Salesforce.

ODBC Connector assets

Create assets in Data Integration to integrate data using ODBC Connector.

When you use ODBC Connector, you can include the following Data Integration assets:

- Data transfer task
- Mapping
- Mapping task
- Synchronization task

For more information about configuring assets and transformations, see *Mappings*, *Transformations*, and *Tasks* in the Data Integration documentation.

CHAPTER 2

Connections for ODBC

Create an ODBC connection to securely read data from and write data to any ODBC-compliant database. You can use an ODBC connection to specify sources, targets, and lookups in mappings, synchronization tasks, and mapping tasks.

When you create the system DSN, you need to specify the data source name and the connection string. Choose a database driver that is compatible with the database to which you want to connect.

Prerequisites

You can use an ODBC connection to connect to ODBC-compliant databases including Informix, Microsoft Access, dBase, DB2, Teradata, Netezza, Greenplum, Microsoft Excel, and Microsoft Azure SQL Data Warehouse.

Before you create an ODBC connection to read from or write to any ODBC-compliant database, be sure to complete the prerequisites.

Configure the ODBC driver

To use an ODBC connection, you'll need to install the ODBC client and ODBC driver on your machine.

- ODBC client is a front-end application installed in your machine. You use the ODBC client application to connect to databases.
- ODBC driver is a back-end application installed on a computer that is used to store data for access by several users. An ODBC driver processes ODBC function calls, submits SQL requests to a specific data source, and returns results to the client application.

An ODBC client can access any database for which there is an ODBC driver.

Configure the DB2 ODBC driver on Linux

Before you establish an ODBC connection to connect to DB2 on Linux, configure the ODBC driver.

1. To get the DB2 ODBC 64-bit driver, contact Informatica Global Customer Support.
2. Install the DB2 ODBC driver on the Secure Agent machine.
3. Configure the `odbc.ini` file properties in the following format:
   ```
   [ODBC_DB2]
   Driver=/root/ODBC_Drivers/DWdb228.so
   Description=<Description of the data source>
   Database=<Name of the database>
   ```

```
IpAddress=<IP address for the DB2 server>
LogonID=<Login id for the DB2 database>
Password=<Password for the DB2 database>
TcpPort=50000
```

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

Property	Description
Driver	Location of the DB2 ODBC driver file.
Description	Description of the data source.
Database	Name of the DB2 database.
IpAddress	IP address of the DB2 server.
LogonID	Login id for the DB2 database.
Password	Password for the DB2 database.
TcpPort	Port number of the DB2 server.

5. From the command line, run the following command to export the `odbc.ini` file:

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

6. Restart the Secure Agent.

Configure the DB2 ODBC driver on Windows

Before you establish an ODBC connection to connect to DB2 on Windows, configure the ODBC driver.

1. To get the DB2 ODBC 64-bit driver, contact Informatica Global Customer Support.

2. Install the DB2 ODBC driver on the Secure Agent machine.

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**.
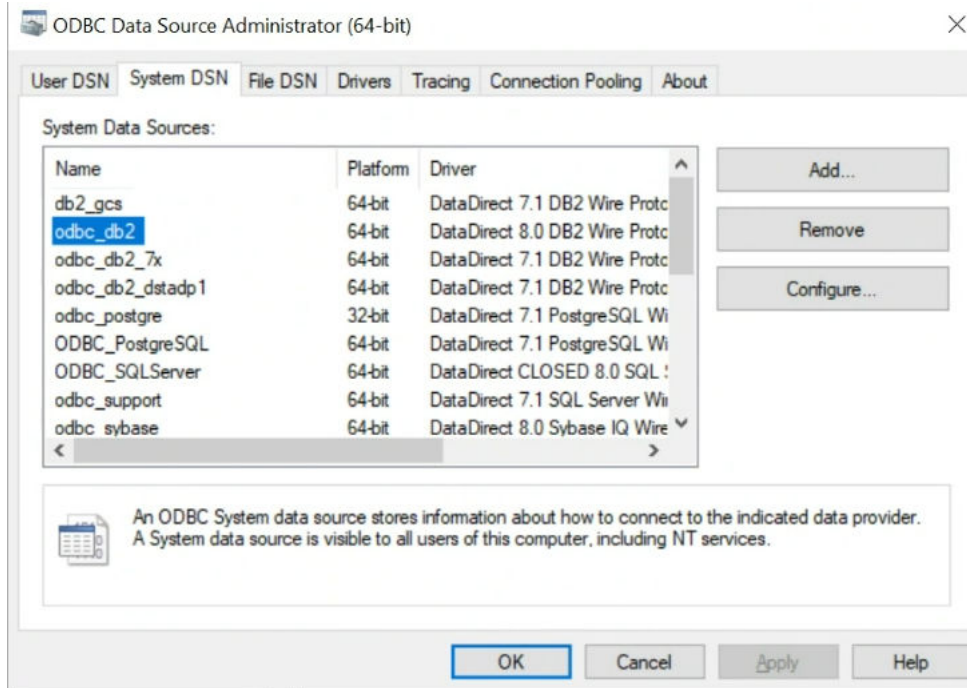
6. Select the **odbc_db2** system data source, and click **Add**.
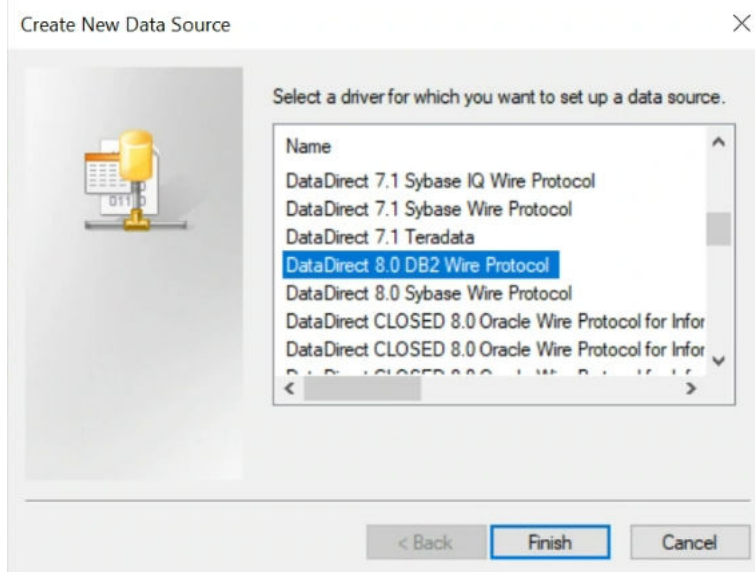
The **Create New Data Source** dialog box appears.

7. Select **DataDirect 8.0 DB2 Wire Protocol**.

8. Click **Finish**.

The **ODBC DB2 Wire Protocol Driver Setup** dialog box appears.

9. Specify the following connection properties:

Property	Description
Data Source Name	Name of the data source.
Description	Description of the data source.
Ip Address	IP address for the DB2 server.
Tcp Port	Port number of the DB2 server.
Location Name	Not applicable.

Property	Description
Collection	Not applicable.
Database Name	Name of the DB2 database.

10. Click **Test Connect** to test the connection that you configured.
The **Logon to DB2 Wire Protocol** dialog box appears.

11. Specify the credentials of the DB2 database.

12. Click **OK**.

Configure the Microsoft Azure ODBC driver

Before you establish an ODBC connection to connect to Microsoft Azure SQL Data Warehouse, configure the ODBC driver.

1. Download the Microsoft ODBC drivers from the Microsoft Azure website, and then install the drivers for Windows and Linux operating systems.

2. On Linux, you need to set the ODBCINI and LD_LIBRARY_PATH environmental variables for the driver and create the DSN entries. Add the path of the `odbc.ini` file to the ODBCINI environment variable.

        setenv ODBCINI "/data/home/adputf_9/cloud_td/ODBCINI/odbc.ini"

3. To set the LD_LIBRARY_PATH environment variable, use the following syntax:

        setenv LD_LIBRARY_PATH "/opt/microsoft/msodbcsql/lib64/libmsodbcsql-11.0.so.2270.0"

4. Add entries for the Microsoft Azure SQL Data Warehouse data sources in the `odbc.ini` file.
The following section shows a sample entry in the `odbc.ini` file:

        [Sample Azure DW ODBC DSN]
        [SD_Azure_DW]
        Driver=/opt/microsoft/msodbcsql/lib64/libmsodbcsql-11.0.so.2270.0
        Description=Microsoft ODBC Driver 11 for SQL Server
        Server=dghhgx2ad3.database.windows.net
        Database=INFASQLDW_DEV
        LogonID=infadwadmin
        Password=

```
QuotedId=Yes
AnsiNPW=Yes
EncryptionMethod=1
SeedBeforeConnect=1
EnableQuotedIdentifiers=1
ValidateServerCertificate=0
DriverUnicodeType=1
```

5.   Restart the Secure Agent.

Configure the Netezza ODBC driver on Linux

Before you establish an ODBC connection to connect to Netezza on Linux, configure the ODBC driver.

1.   Add the path of the `odbc.ini` file to the ODBCINI environment variable.
     For example,

     ```
     setenv ODBCINI "/data/home/qamercury/cloud_td/ODBCINI/odbc.ini"
     ```

2.   To set the ODBCINST environment variable, use the following syntax:

     ```
     setenv ODBCINST /data/home/qamercury/cloud_td/ODBCINI/odbcinst.ini
     ```

3.   To set the LD_LIBRARY_PATH environment variable, use the following syntax:

     ```
     setenv LD_LIBRARY_PATH  ".:/export/qa_adp/thirdparty/netezza/linux.64/
     lib64:$LD_LIBRARY_PATH"
     ```

4.   Add entries for the Netezza data sources in the `odbc.ini` file.
     The following format shows a sample entry in the `odbc.ini` file:

     ```
     [Sample Netezza ODBC DSN]
     Driver=/export/qa_adp/thirdparty/netezza/linux.64/lib64/libnzodbc.so
     Description=NetezzaSQL ODBC
     Servername=adaptersnz2.informatica.com
     Port=5480
     Database=ADPQA_DB
     Username=adpqa
     Password=adpqa
     StripCRLF=false
     ReadOnly=false
     ShowSystemTables=false
     DateFormat=1
     NumericAsChar=false
     DebugLogging=true
     ```

5.   Restart the Secure Agent after you configure the environment variables.

Configure the SAP IQ ODBC driver on Linux

Before you establish an ODBC connection to connect to SAP IQ on Linux, configure the ODBC driver.

1.   To get the SAP IQ ODBC 64-bit driver, contact Informatica Global Customer Support.

2.   Install the SAP IQ ODBC driver on the Secure Agent machine.

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

     ```
     [SAPIQ_ODBC]
     Driver=<SAP IQ location>/lib64/DWsyiq28.so
     DriverUnicodeType=1
     UserID=<Username for the SAP IQ database>
     Password=<Password for the SAP IQ database>
     CommLinks=tcpip(host=hostname;port=2638)
     ServerName=<Server name of the SAP IQ database>
     DatabaseName=<Database name>
     AutoStop=no
     Charset=UTF-8
     Language=EN
     ```

4. Run the following command to export the `odbc.ini` file:

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

5. Restart the Secure Agent.

Configure the SAP IQ ODBC driver on Windows

Before you establish an ODBC connection to connect to SAP IQ on Windows, configure the ODBC driver.

1. To get the SAP IQ ODBC 64-bit driver, contact Informatica Global Customer Support.

2. Install the SAP IQ ODBC driver on the Secure Agent machine.

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**.
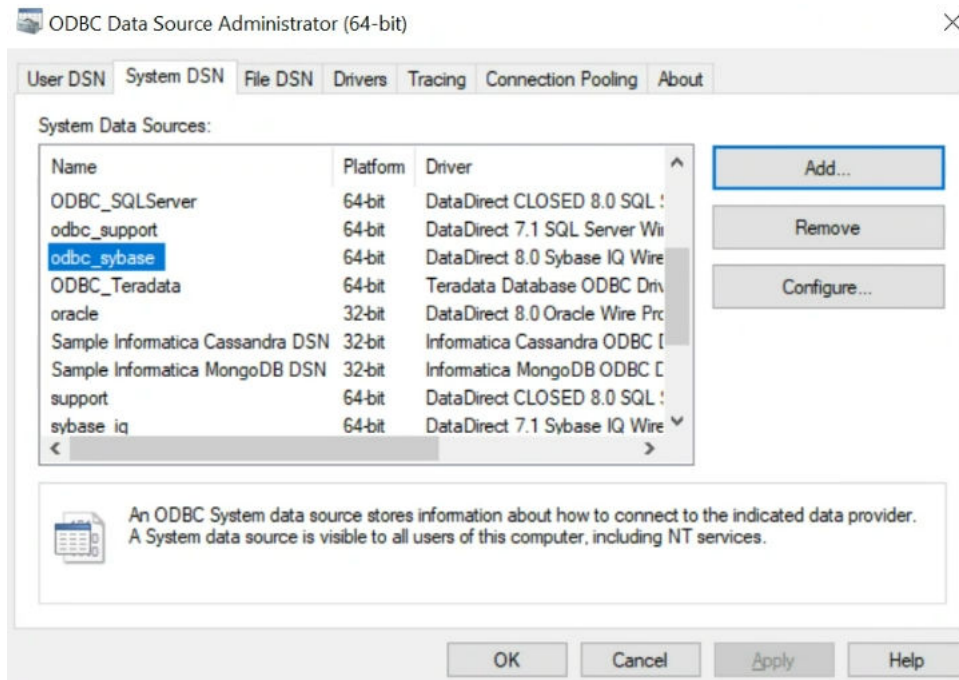
6. Select the **odbc_sybase** system data source, and click **Add**.

The **Create New Data Source** dialog box appears.
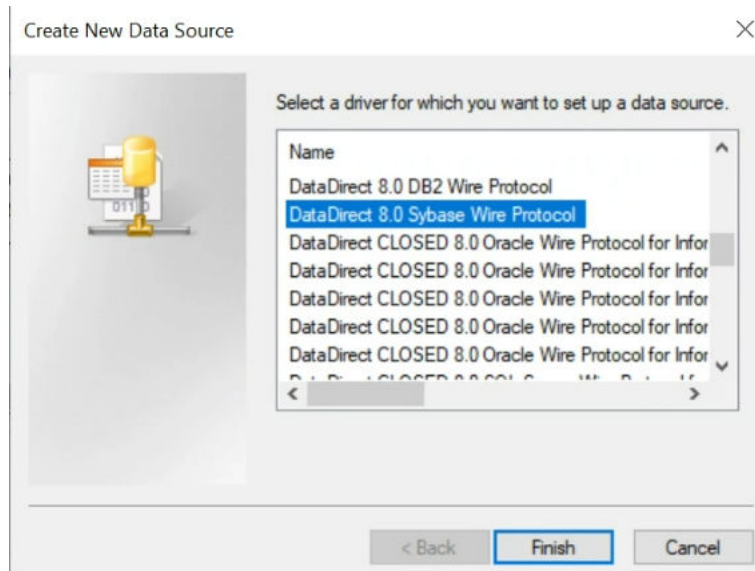
7. Select **DataDirect 8.0 Sybase Wire Protocol**.

8. Click **Finish**.
   The **ODBC Configuration for SQL Anywhere** dialog box appears.
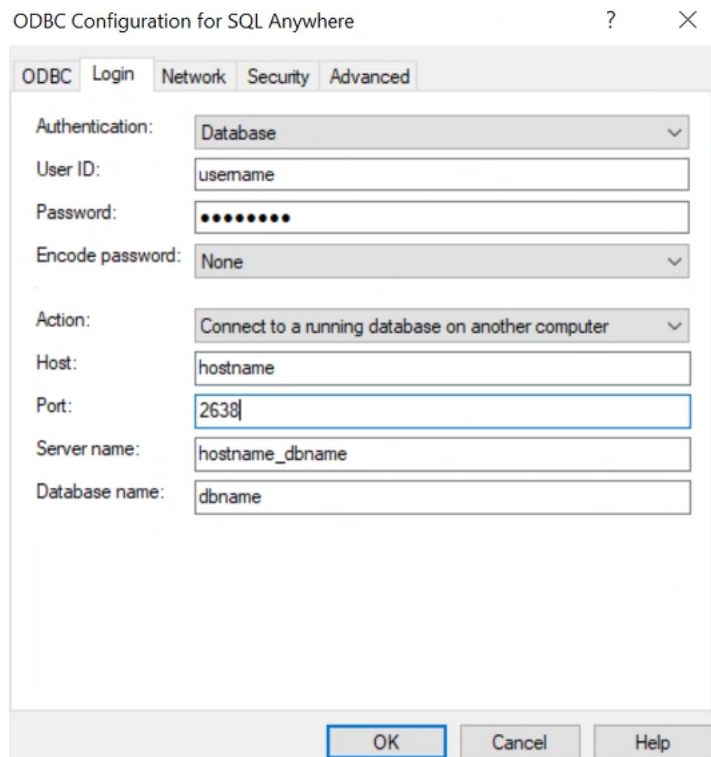
9. Specify the following connection properties:

Property	Description
Authentication	The authentication mode to access the database.
User ID	User name to access the SAP IQ database.
Password	Password to access the SAP IQ database.
Encode password	Not applicable.
Action	The method to connect the SAP IQ database.
Host	Host name of the SAP IQ server.
Port	Port number of the SAP IQ server.
Server Name	Domain name of the SAP IQ account.
Database Name	Name of the SAP IQ database.

10. Click **OK**.

Configure the Teradata ODBC driver on Linux

Before you establish an ODBC connection to connect to Teradata on Linux, configure the ODBC driver.

1. Add the path of the `odbc.ini` file to the ODBCINI environment variable. For example,

    `setenv ODBCINI "/data/home/adputf_9/cloud_td/ODBCINI/odbc.ini"`

2. To set the LD_LIBRARY_PATH environment variable, use the following syntax:

    `setenv LD_LIBRARY_PATH "/opt/teradata/client/<Version>/lib64"`

3. Add entries for the Teradata data sources in the `odbc.ini` file.
   The following section shows a sample entry in the `odbc.ini` file:

```
[Sample Teradata ODBC DSN]
[ODBC Data Sources]
<DSN_NAME>=tdata.so

[<DSN_NAME>]
Driver=<Teradata_ClientHome>/lib64/tdata.so
Description=DataDirect 7.1 Teradata
AccountString=
AuthenticationDomain=
AuthenticationPassword=
AuthenticationUserid=
CharacterSet=ASCII
DBCName=<Teradata Server>
Database=
EnableDataEncryption=0
EnableExtendedStmtInfo=0
EnableLOBs=1
EnableReconnect=0
IntegratedSecurity=0
LoginTimeout=20
LogonID=
MapCallEscapeToExec=0
MaxRespSize=8192
```

```
Password=
PortNumber=1025
PrintOption=N
ProcedureWithSplSource=Y
ReportCodePageConversionErrors=0
SecurityMechanism=
SecurityParameter=
ShowSelectableTables=1
TDProfile=
TDRole=
TDUserName=
```

4. Restart the Secure Agent after you configure the environment variables.

Prepare for Kerberos authentication

You can use Kerberos authentication to connect to DB2 or SAP Sybase ASE databases by placing the required configuration files on the Secure Agent machine. You can also use Kerberos authentication to connect to SSL-enabled DB2 or SAP Sybase ASE databases.

When you configure Kerberos authentication to connect to DB2 or SAP Sybase ASE, consider the following guidelines:

- You can't use the Hosted Agent or serverless runtime environment.
- Ensure that the Secure Agent and database server that you use are registered in the KDC server.
- You can't add more than one KDC to a krb5.conf file.
- You can't generate a credential cache file for more than one Kerberos principal user.

Configure Kerberos authentication

Before you use Kerberos authentication to connect to DB2 or SAP Sybase ASE on Linux or Windows, the organization administrator needs to perform the prerequisite tasks.

1. To configure the krb5.conf file, perform the following tasks:

   a. Create a krb5.conf file on the Secure Agent machine.

   b. Add the details of the Key Distribution Center (KDC) and admin server to the krb5.conf file in the following format:

   ```
   [libdefaults]
   default_realm = <Realm name>
   forwardable = true
   ticket_lifetime = 24h

   [realms]
   <REALM NAME> = {
   kdc = <Location where KDC is installed>
   admin_server = <Location where KDC is installed>
                                   }
    [domain_realm]
   <domain name or host name> = <Domain name or host name of Kerberos>
   <domain name or host name> = <Domain name or host name of Kerberos>
   ```

2. Set the following environment variables on the Secure Agent machine.
   For more information about the required environment variables, see <u>"Set environment variables" on page 18</u>.

3. Restart the Secure Agent.

4. To generate the credential cache file on the Secure Agent machine and use Kerberos authentication to connect to the selected ODBC subtype, perform the following tasks:

    a. On the Secure Agent machine, run the following command and specify the user name and realm name of the selected ODBC subtype:

    ```
    Kinit <user name>@<realm_name>
    ```

    b. When prompted, enter the password for the Kerberos principal user.

Set environment variables

To use Kerberos authentication to connect to DB2 or SAP Sybase ASE, you need to set the required environment variables on the Secure Agent machine.

- `setenv KRB5CCNAME <Absolute path and file name of the credentials cache file>`
- `setenv KRB5_CONFIG <Absolute path of the Kerberos configuration file>\krb5.conf`

After you set the environmental variables, you need to restart the Secure Agent.

Alternatively, you can add the environment variables when you create an ODBC connection with the subtype as **DB2** or **SAP Sybase ASE**.

Enter the *KRB5_CONFIG* and *KRB5CCNAME* details in the **Kerberos Connection Properties** field in the ODBC connection.

For example, add the properties in the following format:

```
KRB5_CONFIG=<Absolute path of the Kerberos configuration file>
\krb5.conf;KRB5CCNAME=<Absolute path of the credential cache file>/<File name>
```

Note: Ensure that you separate each key-value pair with a semicolon.

Connect to ODBC

Let's configure the ODBC connection properties to connect to ODBC-compliant databases.
The ODBC connection provides subtypes to connect to a specific database and perform operations. You can use the following ODBC subtype based on the database:

ODBC Subtype	Description
Azure DW	Select Azure DW to enable SQL ELT optimization when you read from or write to Microsoft Azure SQL Data Warehouse.
DB2	Select DB2 to read from or write to DB2. You can also enable SQL ELT optimization when you read from or write to DB2. If you select DB2, the **Authentication Mode** property appears from which you can use Kerberos authentication to connect to DB2 databases. For more information about how to configure Kerberos authentication, see "Prepare for Kerberos authentication" on page 17.
Google BigQuery	Select Google BigQuery to enable SQL ELT optimization when you read from or write to Google BigQuery.
PostgreSQL	Select PostgreSQL to enable SQL ELT optimization when you read from or write to PostgreSQL.

ODBC Subtype	Description
Redshift	Select Redshift to enable SQL ELT optimization when you read from or write to Amazon Redshift.
SAP IQ	Select SAP IQ to read data from the SAP IQ database.
SAP Sybase ASE	Select SAP Sybase ASE to read from or write to Sybase ASE databases. If you select SAP Sybase ASE, the **Authentication Mode** property appears from which you can use Kerberos authentication to connect to Sybase ASE databases. For more information about how to configure Kerberos authentication, see "Prepare for Kerberos authentication" on page 17.
Snowflake	Select Snowflake to enable SQL ELT optimization when you read from or write to Snowflake.
Teradata	Select Teradata to enable SQL ELT optimization when you read from or write data to Teradata. You can also enable SQL transformation in a mapping to call a stored procedure in Teradata or to process SQL saved queries against the Teradata database. **Note:** If you want to use an SSL-enabled ODBC Teradata connection, ensure that the **SSL Mode** option under **WebSocket** is set to an appropriate value while configuring the Teradata ODBC driver.
Other	Select Other to enable SQL ELT optimization when you read from or write to Microsoft Access, Microsoft Excel, or Netezza.

Before you begin

Before you get started, you'll need to install the ODBC driver and ODBC client on the Secure Agent machine to establish an ODBC connection.

To use Kerberos authentication when you connect to DB2 or SSL-enabled DB2 databases, you'll also need to configure the `krb5.conf` file and set the required environment variables on the Secure Agent machine.

Check out "Prerequisites" on page 8 to learn more about the configuration prerequisites.

Connection details

The following table describes the basic connection properties:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	ODBC

Property	Description
Runtime Environment	The name of the runtime environment where you want to run tasks.
	Select a Secure Agent, Hosted Agent, or serverless runtime environment.
	For more information about how to configure a serverless environment, see "Use the serverless runtime environment" on page 22.
ODBC Subtype	The ODBC connection subtype that you need to select to connect to a specific database. For information on the supported subtypes and their capabilities, see "Connect to ODBC" on page 18.
Authentication Mode	The authentication method to connect to DB2 or SAP Sybase ASE.
	This property appears only if you select the ODBC subtype as **DB2** or **SAP Sybase ASE**.
	Select one of the following authentication modes from the list:
	- Database. Uses the user name and password to connect to the selected ODBC subtype.
	- Kerberos. Uses Kerberos authentication to connect to the selected ODBC subtype.
	When you choose this option on Windows, ensure that the user account that starts the Secure Agent service is available in the endpoint of the selected ODBC subtype. You don't need to enter your credentials to access the database.
	Note: You can't configure Kerberos authentication when you use a Hosted Agent or serverless runtime environment.
	Default is Database.
Kerberos Connection Properties	Additional connection properties to use Kerberos authentication to connect to DB2 or SAP Sybase ASE.
	This property appears only if you select the ODBC subtype as **DB2** or **SAP Sybase ASE** and authentication mode as **Kerberos**.
	If you specify more than one property, separate each key-value pair with a semicolon.
	For example, if you don't set the required environment variables on the Secure Agent machine before you use Kerberos authentication, add the *KRB5_CONFIG* and *KRB5CCNAME* properties in the following format:

	`KRB5_CONFIG=<Absolute path of the Kerberos configuration file>`
	`\krb5.conf;KRB5CCNAME=<Absolute path of the credential cache file>/<File`
	`name>`

User Name	The user name to connect to the required database.
Password	The password to connect to the required database.
	The password cannot contain a semicolon.
Data Source Name	The data source name of the ODBC object.

Property	Description
Schema	The schema name of the ODBC object.
Code Page	The code page of the database server or flat file defined in the connection. Select one of the following code pages: - MS Windows Latin 1. Select for ISO 8859-1 Western European data - UTF-8. Select for Unicode data - Shift-JIS. Select for double-byte character data - ISO 8859-15 Latin 9 (Western European) - ISO 8859-2 Eastern European - ISO 8859-3 Southeast European - ISO 8859-5 Cyrillic - ISO 8859-9 Latin 5 (Turkish) - IBM EBCDIC International Latin-1 - Japanese Extended UNIX Code (incl. JIS X 0212) - Japanese EUC (with \<-> Yen mapping) - Japanese EUC (Packed Format) - IBM EBCDIC Japanese - IBM EBCDIC Japanese CP939 - Japanese EBCDIC Fujitsu - HITACHI KEIS Japanese - NEC ACOS JIPSE Japanese - UNISYS Japanese - MITSUBISHI MELCOM Japanese - Japanese EBCDIC-Kana Fujitsu - HITACHI KEIS-Kana Japanese - NEC ACOS JIPSE-Kana Japanese - UNISYS-Kana Japanese - MITSUBISHI MELCOM-Kana Japanese - EBCDIC Japanese - EBCDIK Japanese - PC Japanese SJIS-78 syntax (IBM-942) - PC Japanese SJIS-90 (IBM-943) - EBCDIC Japanese Katakana SBCS - EBCDIC Japanese Katakana (w/ euro) - EBCDIC Japanese Latin-Kanji (w/ euro) - EBCDIC Japanese Extended (DBCS IBM-1390 combined with DBCS IBM-1399) - EBCDIC Japanese Latin (w/ euro update) - EBCDIC Japanese Katakana SBCS (w/ euro update) - MS Taiwan Big-5 w/ HKSCS extensions - MS Windows Traditional Chinese, superset of Big 5 - Taiwan Big-5 (w/ euro update) - Taiwan Big-5 (w/o euro update) - PC Chinese GBK (IBM-1386) - Chinese EUC - Simplified Chinese (GB2312-80) - Hong Kong Supplementary Character Set - ISO 8859-8 Hebrew - PC Hebrew (old) - PC Hebrew (w/o euro update) - PC Hebrew (w/ euro update) - MS Windows Hebrew (older version) - MS Windows Hebrew (w/o euro update) - Lotus MBCS encoding for Windows Hebrew - EBCDIC Hebrew (updated with sheqel, control characters) - EBCDIC Hebrew (w/ euro) - EBCDIC Hebrew (updated w/ euro and new sheqel, control characters) - Israeli Standard 960 (7-bit Hebrew encoding)

Advanced settings

The following table describes the advanced connection properties:

Property	Description
Driver Manager for Linux	When you create a new ODBC connection on Linux platform, you can select a driver manager for the Linux Secure Agent. Select one of the following driver managers: - Data Direct - unixODBC2.3.0 - unixODBC2.3.4 The default driver manager is UnixODBC2.3.0. To connect to Teradata, you can use only Data Direct as the driver manager on Linux.

Configure an ODBC connection for Siebel

You can use an ODBC connection to connect to Siebel.

- On the Secure Agent machine, use the ODBC Administrator to configure a system DSN.

- In the Data Integration organization, configure an ODBC connection to use the system DSN and an SQL authenticated login.

Rules and guidelines for an ODBC connection

Consider the following rules and guidelines when you create an ODBC connection:

- ODBC connections support system DSNs, not user DSNs.

- Consider to use a predefined connection instead of an ODBC connection for databases. For example, use the Oracle connection type to connect to an Oracle database.

- The Snowflake ODBC driver is not applicable for SUSE Linux.

Use the serverless runtime environment

You can use a serverless runtime environment hosted on AWS or Azure to connect to ODBC-compliant databases.

Before you configure an ODBC connection using the serverless runtime environment, perform the following tasks:

- Add the ODBC drivers in the Amazon S3 bucket or Azure container in your AWS or Azure account.

- Configure the .yml serverless configuration file.

Add the ODBC drivers in the Amazon S3 bucket or Azure container in your AWS or Azure account

Perform the following steps to use a serverless runtime environment in an ODBC connection:

1. Create the following structure for the serverless agent configuration in AWS or Azure:
    `<Supplementary file location>/serverless_agent_config`

2. Add the ODBC drivers in the Amazon S3 bucket or Azure container in the following location in your AWS or Azure account: `<Supplementary file location>/serverless_agent_config/ODBC`

Configure the .yml serverless configuration file

Perform the following steps to configure the .yml serverless configuration file in the serverless runtime environment:

1. Copy the following code snippet to a text editor and specify the driver file names and DSN entries:

```
version: 1
agent:
  dataIntegrationServer:
    autoDeploy:
      odbc:
        drivers:
          - fileCopy:
              sourcePath: ODBC/<Driver_filename>
          - fileCopy:
              sourcePath: ODBC/<Driver_filename>
        dsns:
          - name: "<Name of the ODBC database>"
            entries:
              - key: Driver
                value: <Driver_filename>
              - key: Description
                value: "<Description of the driver>"
```

where the source path is the directory path of the ODBC drivers in AWS or Azure.

Note: The DSN entries vary based on the driver you want to add to the serverless runtime location.

The following example shows the DNS entries for the Microsoft SQL Server driver:

```
version: 1
agent:
  dataIntegrationServer:
    autoDeploy:
      odbc:
        drivers:
          - fileCopy:
              sourcePath: ODBC/DWdb227.so
          - fileCopy:
              sourcePath: ODBC/DWdb227.so
        dsns:
          - name: "<SQL server>"
            entries:
              - key: Driver
                value: DWsqls227.so
              - key: Description
                value: "SQL Server 2014 Connection for ODL"
              - key: HostName
                value: INVW16SQL19
              - key: PortNumber
                value: 1433
              - key: Database
                value: adapter_semantic
              - key: QuotedId
                value: No
              - key: AnsiNPW
                value: Yes
```

2. Ensure that the syntax and indentations are valid, and then save the file as
   `serverlessUserAgentConfig.yml` in the following AWS or Azure location: `<Supplementary file`
   `location>/serverless_agent_config`
   When the .yml file runs, the ODBC drivers are copied from the AWS or Azure location to the
   serverless agent directory and the DNS entries are updated in the `odbc.ini` file.

For more information about how to configure and use the serverless environment, see "Serverless runtime
environment setup" in *Runtime Environments* in the Administrator help.

CHAPTER 3

Synchronization tasks with ODBC Connector

Use a Synchronization task to synchronize data between a source and target.

You can configure a synchronization task using the Synchronization Task wizard.

When you create a task, you can associate it with a schedule to run it at specified times or on regular intervals. Or, you can run it manually. You can monitor tasks that are currently running in the activity monitor and view logs about completed tasks in the activity log.

ODBC sources in synchronization tasks

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

The following table describes the ODBC source properties:

Property	Description
Connection	Name of the source connection.
Source Type	Type of the source object. Select Single, Multiple or Saved Query.
Source Object	Name of the source object. Select the source object for a single source or multiple related sources.
Display source fields in alphabetical order	Displays source fields in alphabetical order instead of the order returned by the source system.

When you configure a synchronization task to use an ODBC 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 ODBC advanced properties:

Advanced Property	Description
Preprocessing Commands	Enter pre-processing script that should be executed before running the synchronization task.
Post-processing Commands	Enter post-processing script that should be executed after running the synchronization task.
Parameter File Name	File that contains the parameters to be used in filters or expressions. Ensure that you have saved the parameter file in the `<Secure Agent installation directory>/apps/ Data_Integration_Server/data/userparameters` directory.
Maximum Number of Log Files	The maximum number of files to retain for each of the following logs: - Session log - Error log - Import log The default value is 10.
Execution Mode	Sets the amount of detail that appears in the log file. Select Standard or Verbose. Default is Standard. Select Verbose for troubleshooting.

Rules and guidelines for ODBC sources

Consider the following rules and guidelines when you configure an ODBC source:

- ODBC source connections do not detect primary-foreign-key relationships. Therefore, these tables do not appear related when you add multiple source objects.

- If you add multiple MySQL tables as the source for a task and you use an ODBC connection to connect to MySQL, you must use a user-defined join. If you create the relationship and run the task, the task fails.

- You cannot use the `select * from <table_name>` command in a saved query. You must provide the explicit column list.

ODBC targets in synchronization tasks

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

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

You can configure target objects to perform the following operations:

- Insert

- Update

- Upsert

- Delete

The following table describes the ODBC target properties:

Property	Description
Connection	Name of the target connection.
Target Object	Name of the target object.
Truncate Target	Truncates an ODBC target before writing data to the target. Select True or False. **Note:** When you truncate a target through an ODBC connection, it executes a DELETE FROM statement to truncate the table. When you write to DB2 and you enable Truncate Target option for the target object, the Secure Agent runs the truncate command instead of the delete command, which enhances the performance of the write operation. If the truncate command fails, the Secure Agent runs the delete command.
Display target fields in alphabetical order	Displays target fields in alphabetical order instead of the order returned by the source system.

When you configure a synchronization task to use ODBC targets, you can configure advanced target properties. Advanced target properties appear on the **Schedule** page of the Synchronization Task wizard.

The following table describes the ODBC advanced target properties:

Advanced Property	Description
Preprocessing Commands	Enter pre-processing script that should be executed before running the synchronization task.
Post-processing Commands	Enter post-processing script that should be executed after running the synchronization task.
Parameter File Name	File that contains the parameters to be used in filters or expressions. Ensure that you have saved the parameter file in the Secure Agent `/main/rdtmDir/userparameters` directory.
Maximum Number of Log Files	The maximum number of files to retain for each of the following logs: - Session log - Error log - Import log
Execution Mode	Sets the amount of detail that appears in the log file. Select Standard or Verbose. Default is Standard. Select Verbose for troubleshooting.

Rules and guidelines for data filters

Use the following rules and guidelines for data filters:

- Do not use data filter variables with ODBC source connections.

- When you run a task that contains a simple data filter on a date field from a Microsoft Access, MySQL, or Oracle database and the connection type is an ODBC connection, the task fails with an error similar to one of the following errors:

```
RR_4035 SQL Error [ FnName: Execute -- [Microsoft][ODBC Microsoft Access Driver] Data
type mismatch in criteria expression.].
```

```
RR_4035 SQL Error [ FnName: Execute -- [Oracle][ODBC][Ora]ORA-01843: not a valid
month ].
```

By default, a simple data filter applies a double quotation mark to escape the column name in an ODBC connection, which causes the error. To resolve the error, create an advanced data filter and apply the correct escape character to the column names.

- If you change the data type of a Microsoft Access or ODBC source column to binary, varbinary, longvarbinary, or ntext in a synchronization task, you cannot create a simple data filter on the column. You can create an advanced data filter.

- In synchronization tasks, use a user-defined join.

CHAPTER 4

Mappings and mapping tasks with ODBC Connector

Use the Data Integration Mapping Designer to create a mapping. When you create a mapping, you configure a source or target to represent an ODBC object.

In advanced mode, the Mapping Designer updates the mapping canvas to include transformations and functions that enable advanced functionality.

Describe the flow of data from source and target along with the required transformations before the agent writes data to the target. When you create a mapping task, select the mapping that you want to use. Use the Mapping Task wizard to create a mapping task. Validate and run the mapping to read data from sources and write to a target. The mapping task processes data based on the data flow logic you define in the mapping.

ODBC sources in mappings

In a mapping, you can configure a Source transformation to represent a single ODBC source, multiple ODBC sources, ODBC query or ODBC parameter.

The following table describes the ODBC source properties that you can configure in a source transformation:

Property	Description
Connection	Name of the source connection. You can select an existing connection, create a new connection, or define parameter values for the source connection property. If you want to overwrite the source connection properties at runtime, select the **Allow parameter to be overridden at run time** option. Specify the parameter file directory and name in the advanced session properties.
Source Type	Type of source object. Select Single Object, Multiple Objects, Query or Parameter.

Property	Description
Parameter	A parameter file where you define values that you want to update without having to edit the task. Select an existing parameter for the source object or click **New Parameter** to define a new parameter for the source object. The **Parameter** property appears only if you select parameter as the source type. If you want to overwrite the parameter at runtime, select the **Allow parameter to be overridden at run time** option. When the task runs, the Secure Agent uses the parameters from the file that you specify in the advanced session properties.
Object	Name of the source object. Select the source object for the task.
Objects and Relationships	Adds multiple objects. Click on Add Source Object. **Note:** The Objects and Relationships property appears only if you select Multiple Objects as the source type.
Query	Click on Define Query and enter a valid custom query. **Note:** The Query property appears only if you select Query as the source type.
Parameter	The parameter for the source object. Create or select the parameter for the source object. **Note:** The parameter property appears only if you select Parameter as the source type.
Filter	Filters records and reduces the number of rows that the Secure Agent reads from the source. Add conditions in a read operation to filter records from the source.
Sort	Sorts records based on the conditions you specify.
Select distinct rows only	Eliminates duplicate rows. Select to eliminate duplicate rows. Default is false.
Tracing Level	Sets the amount of detail that appears in the log file. Select Normal, Verbose Initialization or Verbose Data. Default is normal.
Pre SQL	Pre-SQL command that must be run before reading data from the source.
Post SQL	Post-SQL command that must be run after reading data from the source.
Output is Deterministic	Specify only when the source output does not change between session runs.
Output is Repeatable	Specify only when the order of the source output is same between the session runs. Select Never or Always.
SQL Override	The SQL statement to override the default query generated from the specified source type to read data from the ODBC source.

Key range partitioning

You can configure key range partitioning when you use a mapping task to read data from ODBC sources. With key range partitioning, the Secure Agent distributes rows of source data based on the field 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 partitions.

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. If the mapping includes multiple sources, use the same number of key ranges for each source.

When you define key range partitioning for a column, the Secure Agent reads the rows that are within the specified partition range. For example, if you configure two partitions for a column with the ranges as 10 through 20 and 30 through 40, the Secure Agent does not read the rows 20 through 30 because these rows are not within the specified partition range.

Consider the following rules when you configure partitioning:

- You can configure a partition key for fields of the following data types:
  - String
  - Any type of number data type. However, you cannot use decimals in key range values.
  - Date/time type. Use the following format: MM/DD/YYYY HH24:MI:SS
- You cannot use key range partitions when a mapping includes any of the following transformations:
  - Web Services
  - XML to Relational
- Do not define a partition using the Date data type column in an ODBC mapping that reads from DB2.
- When you specify the datetime and datetime2 columns with subseconds as the key range values in a partitioning, the records are not loaded to the target. You must delete the value in subseconds from the values specified in the partitioning and run the mapping.

Configure key range partitioning

Perform the following steps to configure key range partitioning for ODBC sources:

1. In the Source Properties, click the **Partitions** tab.
2. Select the required partition key from the list.
3. Click **Add New Key Range** to define the number of partitions and the key ranges based on which the Secure Agent must partition data.

   Use a blank value for the start range to indicate the minimum value. Use a blank value for the end range to indicate the maximum value.

   The following image displays the **Partitions** tab:
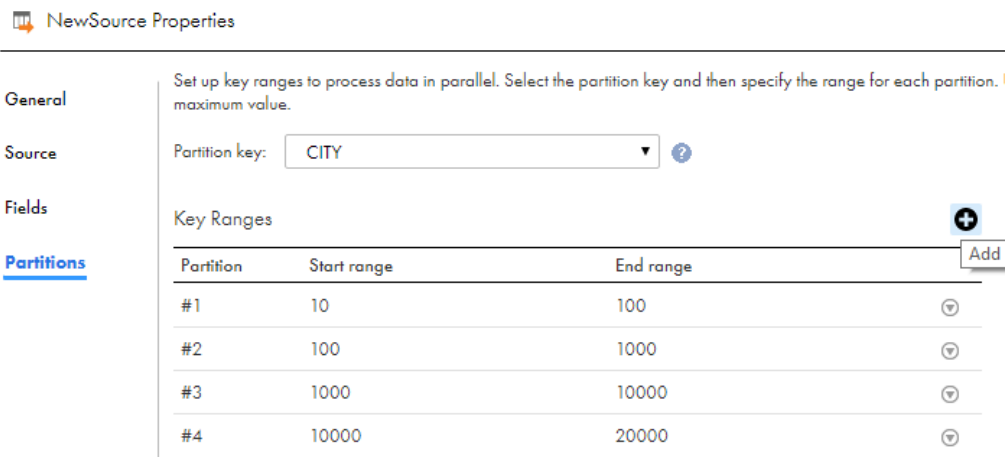
ODBC targets in mappings

In a mapping, you can configure a Target transformation to represent a single ODBC target.

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

Property	Description
Connection	Name of the target connection. You can select an existing connection, create a new connection, or define parameter values for the target connection property. If you want to overwrite the target connection properties at runtime, select the **Allow parameter to be overridden at run time** option.
Target Type	Type of target object. Select Single Object or Parameter.
Parameter	A parameter file where you define values that you want to update without having to edit the task. Select an existing parameter for the target object or click **New Parameter** to define a new parameter for the target object. The **Parameter** property appears only if you select parameter as the target type. If you want to overwrite the target object at runtime, select the **Allow parameter to be overridden at run time** option. When the task runs, the Secure Agent uses the parameters from the file that you specify in the advanced session properties.
Object	The target object for the task. Select the target object.
Parameter	The parameter for the target object. Create or select the parameter for the target object. **Note:** The parameter field appears only if you select parameter as the target type.
Operation	The target operation. Select the target operation. Select Insert, Update, Upsert, Delete or Data Driven. **Note:** Irrespective of the type of write operation you select, you must ensure that you have the privileges for the Delete, Insert, and Update operations for the mapping to run successfully.
Truncate Target	Truncates the database target table before inserting new rows. Select one of the following options: - True. Truncates the target table before inserting all rows. - False. Inserts new rows without truncating the target table Default is False. **Note:** The truncate target property is not applicable for virtual tables.
Forward Rejected Rows	Determines whether the transformation passes rejected rows to the next transformation or drops rejected rows. By default, the mapping task forwards rejected rows to the next transformation.
Update Override	An update SQL statement that updates the data in an ODBC target table. The update SQL statement you specify overrides the default update statements that the Secure Agent generates to update the target based on key columns. You can define an update override statement to update target tables based on both key or non-key columns. In the override statement, you must enclose all reserved words in quotation marks.

Configuring an update override for the target

To override the default update SQL statement that the Secure Agent generates, you can specify an SQL statement in the **Update Override** field of the advanced target properties.

1. Next to the **Update Override** field, click **Configure**.

2. In the **Update Override SQL Editor** dialog box, enter the update SQL statement that the Secure Agent must use.

3. Click **Generate SQL** to generate an SQL query.

4. Click **Format SQL** to format the SQL query you entered.

   You can modify the generated SQL query in the SQL editor based on your requirement.

5. Click **OK**.

ODBC lookups in mappings

You can create lookups for objects in ODBC connection. You can retrieve data from an ODBC lookup object based on the specified lookup condition.

When you configure a lookup in ODBC, you select the lookup connection and lookup object. You also define the behavior when a lookup condition returns more than one match.

Note: You can't configure a Lookup transformation in a data transfer task.

The following table describes the ODBC lookup object properties that you can configure in a Lookup transformation:

Lookup Object Properties	Description
Connection	Name of the lookup connection. You can select an existing connection, create a new connection, or define parameter values for the lookup connection property. If you want to overwrite the lookup connection properties at runtime, select the **Allow parameter to be overridden at run time** option. Specify the parameter file directory and name in the advanced session properties.
Source Type	Type of the ODBC lookup object available. Select one of the following lookup object types: - Single Object - Query - Parameter When the lookup source is large, you can use a custom query to reduce the number of columns to query.
Lookup Object	Name of the lookup object for the mapping.
Parameter	The parameter for the lookup object. Create or select the parameter for the lookup object. **Note:** The parameter property appears only if you select parameter as the source type.
Multiple Matches	Select one of the following options: - Error, if more than 1 output value - Randomly pick 1 output value If you want to overwrite the parameter at runtime, select the **Allow parameter to be overridden at run time** option. When the task runs, the Secure Agent uses the parameters from the file that you specify in the advanced session properties.

Lookup Object Properties	Description
Filter	Not applicable.
Sort	Not applicable.
SQL Override	The SQL statement to override the default query that creates lookup data from an ODBC source.

Calling a stored procedure

You can use an ODBC connection in a mapping to call a stored procedure or process saved queries when you connect to Teradata.

When you configure the ODBC connection, you must select the ODBC subtype as Teradata. You can then use the ODBC connection in a SQL transformation to call a stored procedure or to process saved queries.

You can use the SQL transformation to process SQL queries midstream in a pipeline. You can configure the SQL transformation to process the following types of SQL statements:

Stored procedure

Stored procedures reside in the database and run within the database. When you configure the SQL transformation to process a stored procedure, it passes input parameters to the stored procedure. The stored procedure passes the return value or values to the output fields of the transformation.

SQL Query

You can configure the SQL transformation to process a saved query that you create in Data Integration or you can enter a query in the SQL editor.

You can also parameterize the ODBC connection with the Teradata ODBC subtype in an SQL transformation.

For more information about SQL transformations, see *Transformations* in the Data Integration documentation.

Rules and guidelines for calling a stored procedure

Consider the following rules and guidelines for calling a stored procedure using the ODBC subtype as Teradata:

- You can't configure an unconnected stored procedure using the SQL transformation.
- When you use an SQL transformation to call a stored procedure in Teradata, ensure that the stored procedure definitions do not contain keywords, special characters, and Unicode characters.
- You can't process a stored function in an SQL transformation.
- You can't configure the input or in-out parameter in an entered query that you define in the SQL editor.

ODBC mapping example

You work in the human resource management team of a manufacturing organization and you need to migrate employee data from DB2 to Informix database. Create a mapping task to use the insert operation. You use the following objects in the ODBC mapping:

Source Object

The source object for the mapping task is Employee. Use the ODBC connection to connect to DB2 database and read data from the DB2 Employee object. Use the Employee object as a single source in the mapping task.

Target Object

The target for the mapping task is an Informix table. The target includes the EMPID, EMPNAME, AGE, DOB, DEPTID_EMP, LOCATION, DESIGNATION, and SALARY fields.

Mapping

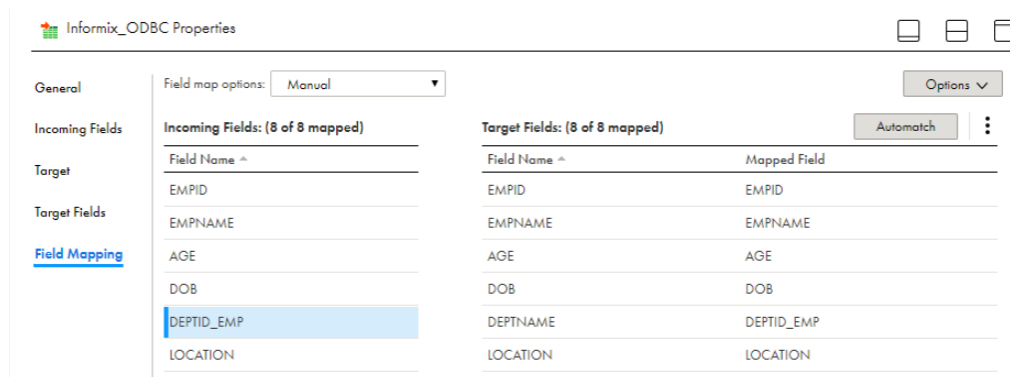Map the fields of the Employee source to the Informix target.

The following image shows the ODBC mapping:

The following image shows the mapped fields of the DB2 source with the Informix target object:

When you run the mapping, the agent writes the Employee details to the Informix table.

The following image shows the Employee details that the agent writes based on the fields you mapped in the mapping task:

Rules and guidelines for ODBC objects

Consider the following rules and guidelines for ODBC objects used as sources and targets in mappings and mapping tasks

- When you create or edit a task with an ODBC connection, database tables from other schema in the database might appear in the wizard. The wizard does not filter tables based on the schema specified for the ODBC connection.

- Even though you can use an ODBC connection to read or write unicode data, ensure that source or target table names and field names do not contain Unicode (UTF-8) characters.

- The data preview area might not display data from an ODBC connection if the database table or column name is also a database key word.

- If you use an ODBC connection for an Oracle database target, ensure that Oracle table columns with the following data types do not exceed the specified maximum precision: char(1999), varchar(3999), nvarchar(3998), and nchar(3998).

- You can't use an ODBC connection to perform update or delete operation on an Excel target.

- If you use an ODBC connection for an Excel source or target file, ensure that named ranges are defined in the Excel file.

- Do not use an ODBC connection to perform upserts on a MySQL database. Use a MySQL connection to perform upserts.

- When you use an ODBC connection to include multiple MySQL tables in mapping tasks, use an advanced relationship instead of an existing or custom relationship.

- When you use a saved query, do not specify a star (*) in the projection list in a saved query. For example, in `SELECT * from EMP_MVIEW`, instead of star, you must specify the columns explicitly in the query.

- When you use an ODBC connection with the subtype as DB2, you cannot perform the following operations:

  - Override the default update SQL statement to write to a DB2 target.

  - Configure an SQL transformation to call a stored procedure.

- When you import data from an SAP Sybase ASE database, the Text data type changes to String with a character length of 255 characters. If the source column contains more than 255 characters, the remaining characters are truncated. To resolve the truncation issue, change the precision of the source column according to your requirement.

CHAPTER 5

SQL ELT optimization

When you use an ODBC connection in a mapping to read data from a source, transform the data, and write the data to a target, you can configure SQL ELT optimization to push the transformation logic to the source or target database system. If the source and target databases are the same, you can configure full SQL ELT optimization for improved performance.

When the Secure Agent applies SQL ELT optimization, it pushes transformation logic to the database. The Secure Agent translates the transformation logic into SQL queries and sends the SQL queries to the database. The database runs the SQL queries to process the transformations. The amount of transformation logic that the Secure Agent pushes to the database depends on the database, the transformation logic, and the mapping configuration. The Secure Agent processes all transformation logic that it cannot push to a database.

SQL ELT optimization improves mapping performance because the database processes the transformation logic faster than the Secure Agent. The amount of data that the Secure Agent needs to read from the database is reduced. When you push the transformation logic to the database, ensure that the database has enough resources to process the queries faster. Otherwise, there could be a performance degradation.

You cannot configure SQL ELT optimization for a mapping in advanced mode.

SQL ELT optimization types

The Secure Agent applies SQL ELT optimization to a mapping when you select the SQL ELT Optimization type in the advanced session property.

You can select the following SQL ELT optimization types:

- None. Select no SQL ELT optimization type for the mapping.
- To Source. The Secure Agent tries to push as much transformation logic as it can to the source database.
- Full. The Secure Agent pushes all transformation logic in the mapping to the target database.

Source SQL ELT optimization

When you configure source SQL ELT optimization for a mapping, the Secure Agent analyzes the optimized mapping from the source to the target or until it reaches a downstream transformation that it cannot push to the source database.

The Secure Agent generates and executes a SELECT statement for each source that has transformation logic pushed down. Then, it reads the results of this SQL query and processes the remaining transformations in the mapping.

You can configure a mapping to use source SQL ELT optimization if the source and target reside in different databases. For example, if a mapping contains a Teradata source and an Oracle target, you can configure source SQL ELT optimization to push some transformation logic for processing to the Teradata source.

Full SQL ELT optimization

When the Secure Agent applies full SQL ELT optimization, it pushes all the transformation logic in the mapping to the target database. You can configure full SQL ELT optimization only when the source and target are in the same database.

When you configure full SQL ELT optimization, the Secure Agent attempts to push all transformation logic in the mapping to the target database. If the Secure Agent cannot push all transformation logic to the database, it performs both source-side and target-side SQL ELT optimization.

When you run a mapping configured for full SQL ELT optimization, the Secure Agent analyzes the mapping from the source to the target or until it reaches a downstream transformation it cannot push to the target database. It generates and executes SQL statements against the source or target based on the transformation logic it can push to the database.

Working with databases

You can configure SQL ELT optimization for the following databases when you use the ODBC connection in the mapping:

- Amazon Redshift
- DB2
- Google BigQuery
- Microsoft Azure SQL Data Warehouse
- Netezza
- PostgreSQL
- Snowflake
- Teradata

You need to select the appropriate ODBC subtype to connect to a database in the ODBC connection properties.

The following table provides the ODBC subtype that you must select in the ODBC connection to connect to a specific database:

Supported database	ODBC Subtype
Amazon Redshift	Redshift
DB2	DB2
Google BigQuery	Google BigQuery
Microsoft Azure SQL Data Warehouse	Azure DW

Supported database	ODBC Subtype
Netezza[1]	Other
PostgreSQL	PostgreSQL
Snowflake	Snowflake
Teradata[1]	Teradata

[1]If you connect to Netezza or Teradata from Linux, you must select **Data Direct** as the **Driver Manager for Linux** in the ODBC connection properties.

SQL ELT optimization transformations

When you configure SQL ELT optimization, the Secure Agent tries to push the configured transformation to the database.

The following table shows the supported SQL ELT optimization types for each database to which you can push the transformation:

Transformations	Amazon Redshift	DB2	Google BigQuery	Microsoft Azure SQL Data Warehouse	Netezza	PostgreSQL	Snowflake	Teradata
Aggregator	Source, Full	Full	Source, Full	Full	Source, Full	Source, Full	Source, Full	Source, Full [2]
Expression	Source, Full	Full	Source, Full	Full	Source, Full	Source, Full	Source, Full	Source, Full [2]
Filter	Source, Full	Source, Full	Source, Full	Full	Source, Full	Source, Full	Source, Full	Source, Full
Joiner	Source, Full	Source, Full	Source, Full	Full	Source, Full	Source, Full	Source, Full	Source, Full
Lookup	Source, Full[1]	Source, Full	Source, Full	-	-	Source, Full	Source, Full[1]	-
Sorter	Source, Full	Source, Full	Source, Full	Full	Source, Full	Source, Full	Source, Full	Source, Full
Union	Source, Full	Source, Full	Source, Full	Full	Source, Full	Source, Full	Source, Full	Source, Full
Router	Full	Source, Full	Full	Full	Full	-	Full	Full

Transformations	Amazon Redshift	DB2	Google BigQuery	Microsoft Azure SQL Data Warehouse	Netezza	PostgreSQL	Snowflake	Teradata
Update Strategy	-	Source, Full	Source, Full	-	-	-	-	-
Sequence Generator	-	-	-	-	-	-	Source, Full	-

[1]You can also configure an unconnected Lookup transformation for Amazon Redshift and Snowflake.

[2]To enable Expression or Aggregator transformation in a Teradata mapping task, see the "Enabling SQL ELT optimization for Expression and Aggregator transformations" topic in the Teradata Connector guide.

SQL ELT optimization functions

When you enable SQL ELT optimization, the Secure Agent converts the expression in the transformation by determining equivalent functions in the database. If there is no equivalent function in the database, the Secure Agent processes the transformation logic.

The following table summarizes the SQL ELT optimization type for the available SQL ELT optimization functions for supported databases:

Functions	Amazon Redshift	DB2	Google BigQuery	Microsoft Azure SQL Data Warehouse	Netezza	PostgreSQL	Snowflake	Teradata
ABS()	Source, Full	Full	Source, Full	Source, Full	Source, Full	Source, Full	Source, Full	Source, Full
ADD_TO_ DATE()	Source, Full	Full	Source, Full	Source, Full	-	-	Source, Full	Full
ASCII()	-	Full	-	Source, Full	-	Source, Full	Source, Full	-
AVG()	Source, Full	Full	Source, Full	Source, Full	Source, Full	Source, Full	Source, Full	Source, Full
CEIL()	Source, Full	Full	Source, Full	Source, Full	-	Source, Full	Source, Full	Full
CHR()	Source, Full	Full	-	Source, Full	-	Source, Full	Source, Full	-

Functions	Amazon Redshift	DB2	Google BigQuery	Microsoft Azure SQL Data Warehouse	Netezza	PostgreSQL	Snowflake	Teradata
CONCAT()	Source, Full	Full	Source, Full	Source, Full	-	Source, Full	Source, Full	Full
COS()	Source, Full	Full	Source, Full	Source, Full	Source, Full	-	Source, Full	Source, Full
COSH()	-	Full	-	-	-	-	Source, Full	Full
COUNT()	Source, Full	Full	Source, Full	Source, Full	Source, Full	Source, Full	Source, Full	Source, Full
DATE_COMPARE()	Source, Full	Full	Source, Full	-	Source, Full	-	Source, Full	Source, Full
DATE_DIFF()	Source, Full	-	-	Source, Full	-	-	Source, Full	-
DECODE()	Source, Full	Full	-	Source, Full	Source, Full	-	Source, Full	Source, Full
EXP()	Source, Full	Full	-	Source, Full	Source, Full	-	Source, Full	Source, Full
FIRST()	-	-	-	Source, Full	-	-	-	-
FLOOR()	Source, Full	Full	Source, Full	Source, Full	-	Source, Full	Source, Full	Full
GET_DATE_PART()	Source, Full	Full	Source, Full	Source, Full	-	-	Source, Full	Full
IIF()	Source, Full	Full	-	Source, Full	Source, Full	-	Source, Full	Source, Full
IN()	Source, Full	Full	-	-	Source, Full	-	-	Source, Full
INITCAP()	Source, Full	Full	-	-	-	-	Source, Full	-
INSTR()	Source, Full	Full	Source, Full	Source, Full	-	-	Source, Full	Full
ISNULL()	Source, Full	Full	Source, Full	Source, Full	Source, Full	-	Source, Full	Source, Full
LAST()	-	-	-	Source, Full	-	-	-	-

Function s	Amazon Redshift	DB2	Google BigQuer y	Microsof t Azure SQL Data Warehou se	Netezza	PostgreSQ L	Snowfla ke	Teradata
LAST_DA Y()	Source, Full	Full	Source, Full	-	-	-	Source, Full	-
LENGTH( )	Source, Full	Full	Source, Full	Source, Full	-	Source, Full	Source, Full	Full
LN()	Source, Full	-	-	-	-	-	Source, Full	Full
LOG()	-	Full	-	Source, Full	-	-	Source, Full	Full
LOWER()	Source, Full	Full	Source, Full	Source, Full	Source, Full	Source, Full	Source, Full	Source, Full
LPAD()	Source, Full	Full	-	-	-	Source, Full	Source, Full	-
LTRIM()	Source, Full	Full	Source, Full	Source, Full	-	Source, Full	Source, Full	Full
MAKE_D ATE_TIM E()	-	-	-	Source, Full	-	-	-	-
MAX()	Source, Full	Full	Source, Full	Source, Full	Source, Full	Source, Full	Source, Full	Source, Full
MAX(NU MBER,DA TE,STRIN G)	-	-	-	-	-	-	-	-
MIN(NU MBER,DA TE,STRIN G)	-	-	-	-	-	-	-	-
MEDIAN( )	-	-	-	-	-	-	Source, Full	-
MIN()	Source, Full	Full	Source, Full	Source, Full	Source, Full	Source, Full	Source, Full	Source, Full
MD5()	Source, Full	-	-	Source, Full	-	-	Source, Full	-
MOD()	Source, Full	Full	Source, Full	Source, Full	-	Source, Full	Source, Full	Full

Function s	Amazon Redshift	DB2	Google BigQuer y	Microsof t Azure SQL Data Warehou se	Netezza	PostgreSQ L	Snowfla ke	Teradata
POWER()	Source, Full	Full	Source, Full	Source, Full	Source, Full	Source, Full	Source, Full	Source, Full
REPLACE CHR()	-	-	Source, Full	Source, Full	-	-	-	-
REPLACE STR()	-	-	Source, Full	Source, Full	-	-	Source, Full	-
ROUND(NUMBER)	Source, Full	Full	Source, Full	Source, Full	-	Source, Full	Source, Full	Full
RPAD()	Source, Full	Full	-	-	-	Source, Full	Source, Full	-
RTRIM()	Source, Full	Full	Source, Full	Source, Full	-	Source, Full	Source, Full	Full
SET_DAT E_PART()	-	-	-	Source, Full	-	-	-	-
SIGN()	Source, Full	Full	-	Source, Full	-	-	Source, Full	Full
SIN()	Source, Full	Full	Source, Full	Source, Full	Source, Full	-	Source, Full	Source, Full
SINH()	-	Full	-	Source, Full	-	-	Source, Full	-
SOUNDE X()	-	Full	-	Source, Full	-	-	-	-
SQRT()	Source, Full	Full	Source, Full	Source, Full	Source, Full	-	Source, Full	Source, Full
STDDEV( )	Source, Full	Full	-	Source, Full	-	-	Source, Full	Full
SUBSTR( )	Source, Full	Full	Source, Full	Source, Full	-	Source, Full	Source, Full	Full
SUM()	Source, Full	Full	Source, Full	Source, Full	Source, Full	Source, Full	Source, Full	Source, Full
SYSTIME STAMP()	Source, Full	Full	Source, Full	Source, Full	-	Source, Full	-	Full
TAN()	Source, Full	Full	Source, Full	Source, Full	Source, Full	-	Source, Full	Source, Full

Functions	Amazon Redshift	DB2	Google BigQuery	Microsoft Azure SQL Data Warehouse	Netezza	PostgreSQL	Snowflake	Teradata
TANH()	-	Full	-	Source, Full	-	-	Source, Full	Full
TO_BIGINT	Source, Full	Full	Source, Full	Source, Full	-	Source, Full	Source, Full	Full
TO_CHAR(DATE)	Source, Full	Full	Source, Full	Source, Full	-	Source, Full	-	Full
TO_CHAR(NUMBER)	Source, Full	Full	Source, Full	Source, Full	-	Source, Full	Source, Full	Full
TO_DATE()	Source, Full	Full	Source, Full	Source, Full	-	Source, Full	Source, Full	Full
TO_DECIMAL()	Source, Full	Full	Source, Full	Source, Full	Source, Full	Source, Full	Source, Full	Full
TO_FLOAT()	Source, Full	Full	Source, Full	Source, Full	Source, Full	Source, Full	Source, Full	Full
TO_INTEGER()	Source, Full	Full	Source, Full	Source, Full	Source, Full	Source, Full	Source, Full	Full
TO_NUMBER()	-	-	-	-	-	-	Source, Full	-
TRUNC(DATE)	Source, Full	-	Source, Full	-	-	-	Source, Full	-
TRUNC(NUMBER)	Source, Full	Full	Source, Full	Source, Full	-	-	Source, Full	Full
UPPER()	Source, Full	-	Source, Full	Source, Full	Source, Full	Source, Full	Source, Full	Source, Full
VARIANCE()	Source, Full	-	-	Source, Full	-	-	Source, Full	Full

SQL ELT optimization variables

When you use SQL ELT optimization, the Secure Agent converts the expression in the transformation by determining equivalent variables in the database. If there is no equivalent variable in the database, the Secure Agent processes the transformation logic.

The following table summarizes the SQL ELT optimization type for the available variables for supported databases:

Variables	Amazon Redshift	DB2	Google BigQuery	Microsoft Azure SQL Data Warehouse	Netezza	PostgreSQL	Snowflake	Teradata
SESSSTARTTIME()	-	-	Full	-	-	-	-	Full
SYSDATE()	-	Full	Source, Full	Source, Full	-	-	-	Full

Configuring SQL ELT optimization

To optimize a mapping, add the mapping to a task, and then configure SQL ELT optimization in the mapping task.

Before you configure SQL ELT optimization, complete the following tasks:

- Download and configure the database-specific ODBC drivers on Windows or Linux on the Secure Agent machine.
- In Data Integration, select the appropriate ODBC subtype, such as Azure DW, DB2, Google BigQuery, Redshift, PostgreSQL, Snowflake, or Teradata in the ODBC connection properties.
- To configure SQL ELT optimization to any other supported databases, for example, Microsoft Access or Netezza, select the **ODBC subtype** as **Other** in the ODBC connection properties.
- If you connect to Netezza or Teradata from Linux, you must select **Data Direct** as the **Driver Manager for Linux** in the ODBC connection properties.

Configure source or full SQL ELT optimization in the **SQL ELT Optimization** section.

1. In the **Schedule** tab of the mapping task, navigate to the **SQL ELT Optimization** section.
2. From the **SQL ELT Optimization** list, select the required type of SQL ELT optimization.

   For information about other advanced session properties related to SQL ELT optimization, such as **Create Temporary View**, **Create Temporary Sequence**, and **Allow SQL ELT for User Incompatible Connections**, see the topic "Advanced Session Properties" under Tasks > Mapping Tasks in the Data Integration documentation.

Cross-schema SQL ELT optimization

You can enable cross-schema SQL ELT optimization for tasks that use source or target objects associated with different schemas within the same database. The Source and Target transformations must use two separate ODBC connections to run cross-schema SQL ELT optimization.

The DSN entry in the ODBC connection of the source, target, or lookup object must be the same. You can override the schema in the connection.

You can configure cross-schema SQL ELT optimization for the following databases when you use the ODBC connection:

- Amazon Redshift
- Microsoft Azure SQL Data Warehouse
- PostgreSQL
- Snowflake
- Teradata

To configure cross-schema SQL ELT optimization in a mapping configuration task, perform the following tasks:

1. In the **SQL ELT Optimization** section, on the **Schedule** tab, select the type of SQL ELT optimization.
2. In the **Advanced Session Properties** section, select the **Enable cross-schema SQL ELT Optimization** check box.

For more information about configuring cross-schema SQL ELT optimization to read from or write data to objects associated with different schemas within the same database and for examples, see the help for the relevant connector.

Cross-database SQL ELT optimization

You can use an ODBC connection in a Snowflake mapping to enable cross-database SQL ELT optimization to run queries on data spread across multiple databases.

You can configure cross-database SQL ELT optimization in the mapping task. You must ensure that the source and target transformations in the mapping must use two different ODBC connections that point to different database DSN entries.

You can configure cross-database SQL ELT optimization for the following database when you use the ODBC connection:

- Snowflake

To configure cross-database SQL ELT optimization in a mapping configuration task, perform the following tasks:

1. On the **Schedule** tab of the mapping configuration task, navigate to the **Advanced Session Properties** section.
2. Click **Add** to add a new session property.
3. From the **Session Property Name** list, select **Allow SQL ELT across Databases** and set the **Session Property Value** to **Yes**.

For more information about configuring cross-database SQL ELT optimization to read from or write data to objects associated with different schemas within the same database and for examples, see the help for the relevant connector.

Verify the SQL ELT query in the session log

To verify that the SQL ELT optimization was applied during running the mapping, you can check the session log for the job. In Monitor, view the log for jobs.

Check the queries in the session logs to verify if the mapping applied SQL ELT optimization.

For example, the following query is generated in the session log for a mapping enabled with full SQL ELT optimization:

```
SQL_1_1_1> OPT_63051 [2019-06-24 17:13:14.772] Optimizer generated SQL statement for target [dbo.EMPINFO_JNR_TGT]:
    INSERT INTO dbo.EMPINFO_JNR_TGT(EMPID, EMPNAME, JOBCODE, DEPTNUM, DEPTNAME, LOCATION) SELECT dbo.EMPINFO.EMPID, dbo.EMPINFO.EMPNAME,
    dbo.EMPINFO.JOBCODE, dbo.EMPINFO.DEPTNUM, dbo.DEPTINFO.DEPTNAME, dbo.DEPTINFO.LOCATION FROM (dbo.DEPTINFO INNER JOIN dbo.EMPINFO ON
    (dbo.EMPINFO.DEPTNUM = dbo.DEPTINFO.DEPTNUM))
SQL_1_1_1> OPT_63056 [2019-06-24 17:13:14.899]
Time SQL execution completed: Mon Jun 24 17:13:14 2019
```

In the example, the generated SQL includes both the `Insert Into` and `Select` queries pushed down to the database as a single statement.

If there is a failure in pushing down the mapping or while generating the query, the session log provides the details of the error. You can use the details to troubleshoot the error.

When you do not enable SQL ELT optimization in a mapping, separate select and insert statements are generated for the read and write operations.

For example, the session log of a read operation shows the following query:

```
READER_1_1_1> RR_4010 [2019-06-24 17:18:18.631] SQ instance [Source] SQL Query [SELECT "dbo"."EMPINFO"."EMPID",
"dbo"."EMPINFO"."EMPNAME", "dbo"."EMPINFO"."JOBCODE", "dbo"."EMPINFO"."DEPTNUM" FROM "dbo"."EMPINFO"]
READER_1_1_1> RR_4049 [2019-06-24 17:18:18.631] SQL Query issued to database : (Mon Jun 24 17:18:18 2019)
```

For example, the session log of a write operation shows the following query:

```
WRITER_1_*_1> WRT_8124 [2019-06-24 17:18:18.630] Target Table EMPINFO_JNR_TGT :SQL INSERT statement:
INSERT INTO "dbo"."EMPINFO_JNR_TGT"("EMPID","EMPNAME","JOBCODE","DEPTNUM","DEPTNAME","LOCATION")  VALUES ( ?, ?, ?, ?, ?, ?)
```

Rules and guidelines for SQL ELT optimization

Certain rules and guidelines apply when you configure SQL ELT optimization to specific databases.

Amazon Redshift

Consider the following rules and guidelines for SQL ELT optimization to an Amazon Redshift database:

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

- To push TO_CHAR() to Amazon Redshift, you must define the date and format arguments.
- Do not specify a format for SYSTIMESTAMP() to push the SYSTIMESTAMP to Amazon Redshift. The Amazon Redshift database returns the complete time stamp.
- To push INSTR() to Amazon Redshift, you must only define string, search_value, and start arguments. Amazon Redshift does not support occurrence and comparison_type arguments.
- The flag argument is ignored when you push TO_BIGINT and TO_INTEGER to Amazon Redshift.
- The CaseFlag argument is ignored when you push IN() to Amazon Redshift.
- If you use the NS format as part of the ADD_TO_DATE() function, the agent does not push the function to Amazon Redshift.
- If you use any of the following formats as part of the TO_CHAR() and TO_DATE() functions, the agent does not push the function to Amazon Redshift:
  - - NS
  - - SSSS
  - - SSSSS
  - - RR
- To push TRUNC(DATE), GET_DATE_PART(), and DATE_DIFF() to Amazon Redshift, you must use the following formats:
  - - D
  - - DDD
  - - HH24
  - - MI
  - - MM
  - - MS
  - - SS
  - - US
  - - YYYY
- When you push the DATE_DIFF() function to Amazon Redshift using a Redshift ODBC connection, the Secure Agent incorrectly returns the difference values. If the result is positive, the Secure Agent returns negative values and if the result is negative, the positive value is returned.
- When you select an Amazon Redshift ODBC connection as source and select an external table as a source object, the data preview fails.

DB2

Consider the following rules and guidelines for SQL ELT optimization to a DB2 database:
Mappings

When you configure an ODBC DB2 mapping, adhere to the following guidelines:

- You cannot use the ODBC DB2 connection to read or write unicode data.
- You cannot calculate and store data temporarily using the **Variable Field** in an Expression transformation.

Functions

When you push functions to DB2, adhere to the following guidelines:

- You cannot get a case-sensitive return value for the IN () function.

- To push the MOD() function, the argument that you pass must be of the Integer date type.
- When you push the SUBSTR() function, the value of the string argument must be greater than the value of the length argument.
- When you push the SOUNDEX() function that contains an empty string or a character string without an English alphabet, the function returns Z000.
- You cannot use microseconds or nanoseconds value when you push functions to the DB2 database.
- You cannot push the GET_DATE_PART() function when the format argument is NULL or the format argument is not specified.
- When you push the INSTR() function, you can only define the string, search_value, and start arguments.
- When you push the DECODE() function that contains the Null value in the argument, the Secure Agent ignores the Null value and the function returns the value of the default argument.
- You cannot push the LTRIM() or RTRIM() function that contains the trim_set argument.
- To push the TO_BIGINT(), TO_DECIMAL(), TO_FLOAT(), or TO_INTEGER function, the argument that you pass must be of the Numeric date type.
- To use the **NULL** expression in the Expression transformation, you must specify the expression in one of the following formats:
  - TO_CHAR(NULL)
  - TO_INTEGER(NULL)
- When you push the EXP() function that contains the Numeric or Double data type, the function might return a different decimal value for the data types as compared to a mapping that runs without SQL ELT optimization.
- To push TO_CHAR(date) and TO_DATE() to DB2, you must use the following formats:
  - YYYYMMDD
  - YYYYMMDD HH24MISS
  - YYYY-MM-DD HH24MISS
  - YYYYMMDD HH24:MI:SS
  - YYYY/MM/DD HH24:MI:SS
  - YYYY/MM/DD HH24MISS
  - YYYY-MM-DD HH24:MI:SS
  - YYYY-MM-DD-HH24.MI.SS.US
  - YYYY-MM-DD-HH24.MI.SS.MS
  - YYYY-MM-DD-HH24.MI.SS
- To push ADD_TO_DATE() and GET_DATE_PART() to DB2, you must use the following formats:
  - HH
  - HH24
  - HH12
  - MM
  - MON
  - MONTH
  - Y

- YY

- YYY

- YYYY

- D

- DD

- DDD

- DY

- DAY

- US

- SS

- MI

Transformations

- To join source tables, ensure that the source tables are from the same database and use the same connection.
- You cannot configure more than one Sorter transformation in a mapping.
- You cannot configure a Union transformation when one of the following conditions are true:
  - One of the sources to the Union transformation contains either a distinct union or sorter.
  - The input groups do not originate from the same relational database management system.
- You cannot configure a Lookup transformation when one of the following conditions are true:
  - The lookup connection is not compatible for SQL ELT optimization with the source connection.
  - The unconnected Lookup transformation is downstream from an Aggregator transformation.
- The mapping fails when the Union transformation is downstream from a transformation that required a view or sequence generator to be created in a database and the connections are on different databases.

SQL override

- The mapping fails when you use an ORDER BY clause in an SQL override query.
- When you define an SQL override query to override the custom query, you cannot extract distinct rows from the source table.
- When you configure an SQL override query in a Lookup transformation, ensure that you select all ports in the same order that appear in the Lookup transformation.

Target operations

- You cannot configure an update override for the target.
- You cannot perform an upsert operation on the target.

Google BigQuery

Consider the following rules and guidelines for SQL ELT optimization to a Google BigQuery database:

- The **Update Override** ODBC advanced target property is not applicable when you use an ODBC connection to configure SQL ELT optimization to write data to a Google BigQuery target.
- Update, upsert, and delete operations are not applicable when you use an ODBC connection to configure SQL ELT optimization to write data to a Google BigQuery target.

- When you configure SQL ELT optimization to write data to a Google BigQuery target, the **Truncate Target** option is not supported. You can configure a pre SQL in the source to delete data from the target table.

- When you configure SQL ELT optimization, ensure that the transformation does not contain a variable port.

- To push the ADD_TO_DATE() function to the Google BigQuery database, you must define the arguments of the Date data type.

- To push the DECODE() function to evaluate multiple columns and conditions for TRUE or FALSE, you must define a boolean expression instead of using TRUE or FALSE in the value argument.

- To push the GET_DATE_PART() function to the Google BigQuery database, you must define the arguments of the Date, DateTime, or Timestamp data type.

- To push the INSTR() function to the Google BigQuery database, you must use the following format: INSTR(string, search_value)

- To push the LAST_DAY() function to the Google BigQuery database, you must define the arguments of the Date data type.

- To push the MAX() function to the Google BigQuery database, you must define the arguments of the Number data type.

- To push the MIN() function to the Google BigQuery database, you must define the arguments of the Date, Number, or String data type.

- To push the ROUND(DATE) or TRUNC(DATE) function to the Google BigQuery database, you must define the arguments of the Timestamp data type.

- To push the TO_CHAR(DATE) function to the Google BigQuery database, you must define the arguments of the Timestamp data type.

- When you push the SYSTIMESTAMP() function to the Google BigQuery database, do not specify any format. The Google BigQuery database returns the complete timestamp.

- When you push the SYSDATE() function to the Google BigQuery database, you must map the output of the expression transformation to a column of Date data type in the Google BigQuery target.

- When you push the TO_DATE() function to the Google BigQuery database, you must configure the output field in the expression transformation to a column of Timestamp data type.

- When you push TO_DATE(string, format) or IS_DATE(string, format) to Google BigQuery and specify the SS, SS.MS, or SS.US format, the function returns the same value for the formats in seconds and subseconds.

- When you push TO_DATE(string, format) or IS_DATE(string, format) to Google BigQuery, you must use the following format arguments:

  - YYYY

  - YY

  - MONTH

  - MON

  - MM

  - DD

  - HH24

  - HH12

  - MI

  - SS

- SS.MS

- SS.US

- PM

- AM

- pm

- am

Note: If you specify HH12 in the format argument, you must specify AM, am, PM, or pm.

- When you push the TO_DATE() function to Google BigQuery using an ODBC connection and provide a constant in the expression, ensure that you specify the format argument. Otherwise, the mapping fails.

- When you push TO_CHAR() to Google BigQuery, you must use the following format arguments:

- YYYY

- YY

- MONTH

- MON

- MM

- Q

- DD

- DDD

- D

- DY

- HH

- HH24

- HH12

- MI

- SS

- SS.MS

- SS.US

- PM

- AM

- pm

- am

- T

Note: If you specify HH12 in the format argument, you must specify AM, am, PM, or pm.

- When you push ROUND(string, format) or TRUNC(string, format) to Google BigQuery, you must use the following format arguments:

- HH24

- MI

- SS

- DD

- MS

- When you push a function that returns a Boolean value, you must configure the output field in the expression transformation to a column of Integer data type.

- If you configure a Lookup condition, you must use only the equals to (=) operator. If you use any operator other than the equals to (=) operator, the mapping fails.

- When you configure the Lookup Source Filter or Lookup SQL Override property in a Lookup transformation, you must add the **Create Temporary View** property under the **Advanced Session Properties** tab when you create a mapping task and select **Yes** in the **Session Property Value** field.

- If the Lookup transformation name contains Unicode characters, the mapping fails.

- When you configure an unconnected Lookup transformation, the fields specified in the Lookup SQL Override property are matched with the lookup fields based on the field names.

- When you configure a Lookup transformation and select **Report error** in the **Multiple Matches** property, the mapping fails and the Secure Agent logs the following error in the session log file:
  ```
  FnName: Execute Direct - [Informatica] [BigQuery] (70) Invalid query: Scalar subquery
  produced more than one element
  ```

- If you specify a function in the Lookup SQL Override property, you must specify the alias name for the function with the lookup field as an argument.

- When you read data of date, datetime, or timestamp data type and write the data as a string to the target, you must add the **DateTime Format String** property under the **Advanced Session Properties** tab when you create a mapping task and specify **YYYY-MM-DD HH24:MI:SS** in the **Session Property Value** field.

- Ensure that the Data Source Name, User name, and the Driver Manager for Linux in the source and target ODBC connection are same. If the values of the Data Source Name, User name, and the Driver Manager for Linux are different in the source and target ODBC connection, the mapping fails with the following error:
  ```
  "SQL ELT optimization stops because the connections are not pushdown compatible."
  ```

- Ensure that you do not specify an in-out parameter of Date or Time data type. Otherwise, the mapping task fails.

- Ensure that you do not parameterize the mapping and use a parameter file to define values for fields, expressions, or data filters. Otherwise, the mapping task fails.

- Ensure that you do not completely parameterize the expression in the Expression transformation and use a parameter file to define values. Otherwise, the mapping task fails.

Microsoft Azure SQL Data Warehouse

Consider the following rules and guidelines for SQL ELT optimization to a Microsoft Azure SQL Data Warehouse database:

- When you use the Microsoft ODBC Driver 17, you cannot run mappings on Red Hat Enterprise Linux 8.

- When you read data that contains reserved keywords from Microsoft Azure Synapse SQL, ensure that you add the keywords to the `reswords.txt` file in the agent machine. The `reswords.txt` file is available in the following path:
  ```
  <Secure Agent installation directory>\downloads\package-ICSAgent_RXX.X\package\ICS\main
  \bin\rdtm
  ```

- You cannot use the ORDER BY clause in a source custom query unless you also specify a TOP, OFFSET, or FOR XML clause in the query.

- When you run a mapping configured with the **Create New at Runtime** option, and if you drop the target that was created on Microsoft Azure SQL Data Warehouse using the drop table tablename command, and rerun the same mapping, the Secure Agent instead of creating the target and then writing data to that target, does not trigger the create query and results in an error.

- The **Update Override** ODBC advanced target property is not applicable when you use an ODBC connection to connect to Microsoft Azure SQL Data Warehouse.

- Upsert operations for SQL ELT optimization are not applicable when you use an ODBC connection.

- The `datetimeoffset` datatype is applicable only in passthrough mappings.

- The Microsoft Azure SQL Data Warehouse aggregate functions accept only one argument, which is a field set for the aggregate function. The agent ignores any filter condition defined in the argument. In addition, ensure that all fields mapped to the target are listed in the `GROUP BY` clause.

- To push the `TO_CHAR()` function to the Microsoft Azure SQL Data Warehouse database, you must define the date and format arguments.

- When you push the `SYSTIMESTAMP()` and `SYSDATE()` functions to the Microsoft Azure SQL Data Warehouse database, do not specify any format. The Microsoft Azure SQL Data Warehouse database returns the complete time stamp. `SYSDATE` works without brackets `()` only, if used it shows as invalid expression.

- You cannot push the `TO_BIGINT()` or `TO_INTEGER()` function with more than one argument to the Microsoft Azure SQL Data Warehouse database.

- When you push the `REPLACECHR()` or `REPLACESTR()` function to the Microsoft Azure SQL Data Warehouse database, the agent ignores the caseFlag argument.
  For example, both `REPLACECHR(false, in_F_CHAR, 'a', 'b')` and `REPLACECHR(true, in_F_CHAR, 'a', 'b')` return the same value.

- To push `INSTR()` to Microsoft Azure SQL Data Warehouse database, you must only define string, search_value, and start arguments. Microsoft Azure SQL Data Warehouse does not support occurrence and comparison_type arguments.

- Microsoft Azure SQL Data Warehouse connector supports the following date formats with the `TO_DATE()` function:

  - `YYYY-MM-DD HH24:MI:SS.NS`

  - `YYYY-MM-DD HH12:MI:SS.NSAM`

  - `MON DD YYYY HH12:MI:SS.NSAM`

  - `MON DD YYYY HH24:MI:SS.NS`

  - `DD MON YYYY HH12:MI:SS.NSAM`

  - `DD MON YYYY HH24:MI:SS.NS`

  - `MM/DD/YY HH12:MI:SS.NSAM`

  - `MM/DD/YY HH24:MI:SS.NS`

  - `MM/DD/YYYY HH12:MI:SS.NSAM`

  - `MM/DD/YYYY HH24:MI:SS.NS`

  - `HH24:MI:SS.NS`

  - `HH12:MI:SS.NSAM`

- To push the `SET_DATE_PART()` function to the Microsoft Azure SQL Data Warehouse database, you must use the following date data types as arguments:

  - `datetime`

  - `datetimeoffset`

  - `datetime2`

  - `smalldatetime`

You can use the following formats for date data types:

- `YYYY, YY, YY, Y`

- `MM, MON, MONTH`

- `D, DD, DD, DY, DAY`

- `HH, HH12, HH24`

- `MI`

- `MS`

- `SS`

> **Note:** `NS` and `US` formats are not applicable to `SET_DATE_PART()`.

- To push the `ADD_TO_DATE()` function to the Microsoft Azure SQL Data Warehouse database, you must use the following date data types as arguments:

  - `date`

  - `datetime`

  - `datetimeoffset`

  - `datetime2`

  - `smalldatetime`

  - `time`

  You can use the following formats for date data types:

  - `YYYY, YY, YY, Y`

  - `MM, MON, MONTH`

  - `D, DD, DD, DY, DAY`

  - `HH, HH12, HH24`

  - `MI`

  - `MS`

  - `SS`

  - `NS`: applicable to `datetimeoffset`, `datetime2`, and `time`

  - `US`

- To push the `MAKE_DATE_TIME()` function to the Microsoft Azure SQL Data Warehouse database, you must use the following date data types as arguments:

  - `date`

  - `datetime`

  - `datetimeoffset`

  - `datetime2`

  - `smalldatetime`

  - `time`

  You can use year, month, day, hour, minute, second, and nanosecond with appropriate return date types.

Netezza

Consider the following rules and guidelines for SQL ELT optimization to a Netezza database:

- When you configure SQL ELT optimization in a mapping to write data that contains numeric fields to a Netezza target, the Secure Agent rounds large-range numeric data (38,10) to the first decimal place.
- You cannot push down data of the Timestamp data type to a Netezza target database.

Snowflake

Consider the following rules and guidelines for SQL ELT optimization to a Snowflake database:

Use the following rules and guidelines when you configure SQL ELT optimization to a Snowflake database:
Update override property

> The update override property is applicable for all ODBC subtypes in the ODBC connection, except Snowflake.

Common fields in multiple sources

> When you use a Snowflake ODBC connection in a mapping enabled with SQL ELT optimization to read data from two Snowflake sources that have fields with the same name and you define a filter condition for one of the common fields, the mapping fails.

Create Temporary View session property

> Enable the **Create Temporary View** property in the session properties of the mapping task before you configure the following properties:

- Upsert, update, or delete operation.
- Filter or joiner in the query options of the source.
- Push down a custom SQL query from the source.
- Unconnected lookup.

Not Null constraint

> When you run a mapping to write data to a Snowflake target, and you define the primary key for the target table but do not explicitly define the Not Null constraint, the upsert, delete, or update operation fails. You must add the Not Null constraint for the target table and then run the upsert, delete, or update operation.

Sequence Generator transformation

> When you configure a Sequence Generator transformation in a mapping, adhere to the following guidelines:

- Add the **Create Temporary Sequence** advanced session property and set the session property value to **Yes**.
- In the **Output** fields of the Sequence Generator transformation, do not map the CURRVAL field to an input field in a Target transformation or other downstream transformation.
- When we use the Sequence Generator transformation in a mapping and the target operation is upsert, update, or delete, the SQL ELT query is not generated. You must select the target operation as Insert to push down the Sequence Generator transformation to Snowflake.
- When you configure cross-database SQL ELT optimization for a Snowflake ODBC mapping that includes a Sequence Generator transformation, SQL ELT optimization fails.

Lookup

When you configure a lookup, adhere to the following guidelines:

- When you configure a connected lookup, you can select the **Return All Rows** multiple matches option in the lookup object properties. If you select any other option other than **Return All Rows**, the SQL ELT query is not generated.

- When you configure an unconnected lookup, you must select the **Report error** multiple matches option in the unconnected lookup object properties for the SQL ELT optimization to work. Ensure that you enable the **Create Temporary View** property in the session properties of the mapping task.

- When you configure an unconnected lookup in a mapping configured for SQL ELT optimization using a Snowflake ODBC connection, and if there are multiple matches in the data, the Secure Agent processes the records, but does not log an error when it finds multiple matches.

Create New at Runtime option

You can configure the **Create New at Runtime** option for a Target transformation in a mapping configured for SQL ELT optimization. When you use the **Create New at Runtime** option, adhere to the following guidelines:

- As the SQL identifiers are always enclosed in double quotes, you must explicitly enable the **AddQuotesAlwaysPDO** flag and set the value to **Yes** in the custom properties in the advanced session properties of the mapping task .

- If you enable the truncate target option in the target mapping, you must add the **AddQuotesAlways** DTM property and set the value to **Yes**. To configure this property, in the Secure Agent properties, navigate to the **Custom Configuration Details** section, select **Data Integration Server** as the service, **Type** as **DTM**, add the **AddQuotesAlways** property, and set the value to **Yes**.

- When you use the **Create New at Runtime** option, the TIMESTAMP_LTZ, TIMETAMP_TZ, Boolean, and Time data types are not supported.

- When you run a mapping configured with the **Create New at Runtime** option, and if you drop the target that was created on Snowflake using the drop table tablename command, and rerun the same mapping, the Secure Agent instead of creating the target and then writing data to that target, does not trigger the create query and results in an error.

- When you use the **Create New at Runtime** option, and if the name you specify for the target already exists in Snowflake, the Secure Agent inserts the data to the existing target table.

- When you use the **Create New at Runtime** option, but later delete the created target table and re-run the mapping task, the Secure Agent fails to create the target table.

Functions

When you push functions to Snowflake, adhere to the following guidelines:

- The Snowflake aggregate functions accept only one argument, which is a field set for the aggregate function. The agent ignores any filter condition defined in the argument. In addition, ensure that all fields mapped to the target are listed in the GROUP BY clause.

- When you push the SYSTIMESTAMP() function to the Snowflake database, do not specify any format. The Snowflake database returns the complete time stamp.

- You cannot push the TO_BIGINT() or TO_INTEGER() function with more than one argument to the Snowflake database.

- When you push the REPLACESTR() function to the Snowflake database, the agent ignores the caseFlag argument. The REPLACESTR() function must include four parameters for SQL ELT optimization to work.

- When you push the MD5 function, it returns NULL if any input is NULL.

- You cannot use millisecond and microsecond values when you push functions to the Snowflake database.
- You must use only the following supported date and time formats:
  - Y
  - YY
  - YYY
  - YYYY
  - MM
  - MON
  - MONTH
  - D
  - DD
  - DDD
  - DY
  - DAY
  - HH
  - MI
  - SS
  - NS

  For information on date and time-related functions, see the Snowflake documentation.

Teradata

Consider the following rules and guidelines for full SQL ELT optimization to a Teradata database:

- You cannot push the LTRIM(), RTRIM(), or ROUND(NUMBER) function that contains more than one argument to the Teradata database.
- You cannot use the upsert operation in a Joiner transformation.
- When you configure an SQL override query using full SQL ELT optimization, you must map all fields that you specify in the SQL override query to the Teradata target object.
- You can push the STDDEV() and VARIANCE() functions to the Teradata database only in an Aggregator transformation.
- You cannot use a ORDER BY clause in a custom query or SQL override query, unless you also specify the TOP clause in the query.

CHAPTER 6

Data type reference

Data Integration uses the following data types in ODBC mappings, mapping tasks, and data transfer tasks:

- ODBC native data types appear in the Source transformation and Target transformation when you choose to edit metadata for the fields.
- Transformation data types. Set of data types that appear in the transformations. These are internal data types based on ANSI SQL-92 generic data types, which the Secure Agent uses to move data across platforms. They appear in all transformations in a mapping.

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

ODBC data types and transformation data types

The following table compares ODBC data types such as Microsoft Access or Excel that Data Integration supports and the corresponding transformation data types:

ODBC Data Type	Transformation Data Type	Description
Bigint	Bigint	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807 characters; precision 19, scale 0
Binary	Binary	1 to 104,857,600 bytes
Bit	String	1 to 104,857,600 characters
Char(L)	String	1 to 104,857,600 characters
Date	Date/Time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. (precision to the nanosecond)
Datetime	Date/Time	Jan 1, 1753 00:00:00 to Dec 31, 9999 23:59:59.997
Decimal(P, S)	Decimal	Precision 1 to 28, scale 0 to 28
Double	Double	Precision 15
Float	Double	Precision 15

ODBC Data Type	Transformation Data Type	Description
Integer	Integer	-2,147,483,648 to 2,147,483,647 characters; precision 10, scale 0
Long Varbinary	Binary	1 to 104,857,600 bytes
Nchar	Nstring	1 to 104,857,600 characters
Ntext	Ntext	1 to 104,857,600 characters
Numeric	Decimal	Precision 1 to 28, scale 0 to 28
Nvarchar	Nstring	1 to 104,857,600 characters
Real	Real	Precision 7, scale 0
Smallint	Smallint	Precision 5, scale 0
Text	Text	1 to 104,857,600 characters
Time	Date/Time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. (precision to the nanosecond)
Timestamp	Date/Time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. (precision to the nanosecond)
Tinyint	Small Integer	Precision 5, scale 0
Varbinary	Binary	1 to 104,857,600 bytes
Varchar(L)	String	1 to 104,857,600 characters

Note: When the Secure Agent runs in Unicode data movement mode, the column precision that you specify for ODBC data types determines the number of characters the Secure Agent reads and writes.

Troubleshoot

When you run an ODBC MySQL mapping that uses the MySQL ODBC 3.51 driver to write data to a MySQL target that contains a column with the Double data type, the mapping fails with the following error:

```
Column [A1_NUMBER_18_0] has scale greater than precision.
```

To resolve the error, upgrade to the MySQL ODBC 5.1 driver.

Note: Before you configure an ODBC connection in Data Integration, you must create a system Data Source Name (DSN).

INDEX