Informatica®

Informatica® Cloud Data Integration

Oracle Connector

Informatica Cloud Data Integration Oracle Connector
July 2024

© Copyright Informatica LLC 2017, 2024

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

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

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

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

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

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

NOTICES

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

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

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

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

Publication Date: 2024-07-15

Table of Contents

Preface 5

Informatica Resources 5

- Informatica Documentation 5
- Informatica Intelligent Cloud Services web site 5
- Informatica Intelligent Cloud Services Communities 5
- Informatica Intelligent Cloud Services Marketplace 6
- Data Integration connector documentation 6
- Informatica Knowledge Base 6
- Informatica Intelligent Cloud Services Trust Center 6
- Informatica Global Customer Support 6

Chapter 1: Introduction to Oracle Connector 7

Oracle Connector assets 7

Oracle Connector example 8

Chapter 2: Connections for Oracle 9

Prerequisites 9

- SSL configuration 9
- Kerberos authentication 11

Connect to Oracle 12

- Before you begin 12
- Connection details 13
- Authentication types 13
- Advanced settings 14

Configuring SSL with the serverless runtime environment 16

Oracle connection rules and guidelines 17

Chapter 3: Synchronization tasks with Oracle Connector 18

Oracle sources in synchronization tasks 18

Oracle targets in synchronization tasks 19

- Configuring an update override for the target 19

Advanced properties for Oracle sources and targets in synchronization tasks 19

Oracle lookups in synchronization tasks 20

Data filters 20

- Simple data filters 20
- Advanced data filters 20

Rules and guidelines for data filters 20

Chapter 4: Mappings and mapping tasks with Oracle Connector 22

Oracle sources in mappings 22

Key range partitioning. . . . 24
Configuring key range partitioning. . . . 24
Oracle targets in mappings. . . . 25
Configuring an update override for the target. . . . 26
Enabling array update and upsert operations for an Oracle target. . . . 26
Oracle lookups in mapping. . . . 27
SQL ELT optimization. . . . 28
Full SQL ELT optimization. . . . 28
Source SQL ELT optimization. . . . 29
SQL ELT optimization functions. . . . 29
Supported functions for Expression transformation. . . . 30
Configuring SQL ELT optimization. . . . 31
Cross-schema SQL ELT Optimization. . . . 31
Rules and guidelines for Oracle sources and targets in tasks. . . . 32

Chapter 5: Replication tasks. . . . 34

Oracle sources in replication tasks. . . . 34
Oracle targets in replication tasks. . . . 35
Full load. . . . 35
Rules and guidelines for Oracle sources and targets in replication tasks. . . . 35

Chapter 6: Troubleshooting. . . . 37

Troubleshooting an Oracle connection. . . . 37
Troubleshooting a synchronization task. . . . 38
Troubleshooting a replication task. . . . 38

Appendix A: Data type reference. . . . 39

Oracle and transformation data types. . . . 39
Rules and guidelines for data types. . . . 40

Index. . . . 41

Preface

Use *Oracle Connector* to learn how to read from or write to Oracle by using Cloud Data Integration. Learn to create an Oracle connection, develop mappings, and run synchronization, mapping, dynamic mapping, replication, and data transfer tasks in Data Integration. You can also learn how to configure SQL ELT optimization and cross-schema SQL ELT optimization using an Oracle 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 Oracle Connector

You can use Oracle Connector to connect to Oracle databases from Data Integration to read and write data.

You can also use Oracle Connector to connect to the following databases:

- Oracle Database Cloud Service and Oracle databases that are enabled for advanced security
- Oracle Autonomous Database
- Azure Oracle private endpoint on a virtual network

You can use Oracle objects as sources and targets in synchronization tasks, mappings, mapping tasks, replication tasks, PowerCenter tasks, and data transfer tasks.

You can configure SSL authentication to establish one-way or two-way secure communication with the Oracle server. You cannot configure the SSL authentication when you use the Hosted Agent.

You can create, update, or delete records when you use Oracle Connector to write to Oracle databases. When you create a synchronization task, mapping, or mapping task, you can use the Oracle bulk API to perform insert operations.

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

Oracle Connector assets

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

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

- Data transfer task
- Dynamic mapping task
- Mapping
- Mapping task
- PowerCenter task
- Replication task
- Synchronization task

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

Oracle Connector example

You are an operations analyst who manages the customer support website of your organization. You want to get the complete statistics and elaborate reports about the support your organization gives to the customers.

You can use Oracle Connector to read data and maintain a database. You can configure a synchronization task or mapping task to read data from Oracle. You can then write the data to data visualization tools such as Tableau to visually analyze the data.

CHAPTER 2

Connections for Oracle

Create an Oracle connection to connect to Oracle databases to read and write data.

You can use Oracle connections in synchronization tasks, mappings, mapping tasks, replication tasks, PowerCenter tasks, and data transfer tasks.

Prerequisites

You can use Oracle Connector to connect to an SSL-enabled Oracle database with Oracle database authentication or Kerberos authentication.

To connect to an SSL-enabled Oracle database, see "SSL configuration" on page 9.

To connect to Oracle databases with Kerberos authentication, see "Kerberos authentication" on page 11.

SSL configuration

Before you use a secure Oracle connection with Oracle database authentication or Kerberos authentication to connect to an SSL-enabled Oracle database, the organization administrator needs to perform the prerequisite tasks.

1. Create a truststore certificate.
2. Create a keystore certificate. Applicable only when Client authentication is enabled in the Oracle database.

Adding the server certificate to the truststore

Add the server certificate to the client's truststore to establish a secure Oracle connection.

Use the following keytool command to add the server certificate to the client's truststore:

```
keytool -import -trustcacerts -alias ca -file <server certificate with path> -keystore
<name of truststore to be generated with extension> -storepass <password for truststore>
-storetype <store type>
```

For example, consider you have a server certificate `oratls_server.cert` in the following location: `C:\SSL\oracle`

1. Run the following command to create the truststore `truststore.jks` with the truststore password "password":

```
C:\SSL\oracle> keytool -import -trustcacerts -alias ca -file oratls_server.cert -keystore
truststore.jks -storepass password -storetype JKS
```

2. Run the following command to create the PKCS12 truststore `truststore.p12` with truststore password "password":

```
C:\SSL\oracle> keytool -import -trustcacerts -alias ca -file oratls_server.cert -keystore
truststore.p12 -storepass password -storetype PKCS12
```

Creating a keystore certificate

Create a keystore certificate when client authentication is enabled in the Oracle server. You must create a keystore certificate that contains all the client certificates to establish an Oracle connection.

Perform the following steps to create a keystore certificate:

1. Download and install the Oracle client from the Oracle website.
2. Run the following command to create an Oracle wallet:

```
orapki wallet create -wallet <Path where wallet is to be created> -auto_login -pwd
<wallet password>
```

3. Run the following command to create a self-signed client certificate to the Oracle wallet:

```
orapki wallet add -wallet <Path where wallet is to be created> -dn "CN=<common
name>, OU=<organization unit>, O=<organization>, L=<locality>, ST=<state>,
C=<country>" -keysize <key size in bits> -self_signed-validity <number of days> -pwd
<wallet password>
```

   The command runs and creates the pkcs12 certificate at the specified location.

   You must specify the values of the `CN=<common name>, OU=<organization unit>, O=<organization>, L=<locality>, ST=<state>, C=<country>, keysize <key size in bits>, self_signed -validity <number of days>, and pwd <wallet password>` from the server certificate.

4. Run the following orapki command to export the self-signed client certificate:

```
orapki wallet export -wallet <wallet path> -dn "CN=<common name>, OU=<organization
unit>, O=<organization>, L=<locality>, ST=<state>, C=<country>" -cert <Name of the
exported certificate with path>
```

   The **-dn** command identifies the client certificate uniquely as the server wallet contains multiple client certificates installed.

5. Install the self-signed client certificate in the server Oracle wallet.

   Note: The client authentication fails if you do not add the self-signed client certificate to the server database Oracle wallet.

6. Add the server certificate as a trusted certificate to the Oracle wallet.

   Run the following command to add the server certificate:

```
orapki wallet add -wallet <wallet path> -trusted_cert -cert <Name of the server
certificate with path> -pwd <wallet password>
```

   Note: You must use the same wallet password for all orapki commands.

   Example Tasks

   Perform the following tasks to create a keystore certificate:

   1. Run the following command to create an Oracle wallet:

```
C:\app\client\ksuwalka\product\12.1.0\client_1\BIN>orapki wallet create -wallet
C:\app\client\ksuwalka\product\12.1.0\client_1\owm\wallet -auto_login -pwd oracle4u
```

   2. Run the following command to create a self-signed client certificate to the Oracle wallet:

```
C:\app\client\ksuwalka\product\12.1.0\client_1\BIN>orapki wallet add -wallet C:\app
\client\ksuwalka\product\12.1.0\client_1\owm\wallet -dn "CN=inw1pc07_kriti, OU=DEV,
O=infa,L=blr, ST=ka, C=IN" -keysize 2048 -self_signed -validity 3650 -pwd oracle4u
```

The `ewallet.p12` certificate is created in the following location: `C:\app\client\ksuwalka\product\12.1.0\client_1\owm\wallet`

3. Run the following orapki command to export the self-signed client certificate:
```
C:\app\client\ksuwalka\product\12.1.0\client_1\BIN>orapki wallet export -wallet
C:\app\client\ksuwalka\product\12.1.0\client_1\owm\wallet -dn "CN=inw1pc07_kriti,
OU=DEV, O=infa,L=blr, ST=ka, C=IN" -cert C:\Users\ksuwalka\Desktop
\client_inw1pc07.cert
```
4. Add the server certificate as a trusted certificate to the Oracle wallet. Run the following command to add the server certificate:
```
C:\app\client\ksuwalka\product\12.1.0\client_1\BIN>orapki wallet add -wallet C:\app
\client\ksuwalka\product\12.1.0\client_1\owm\wallet -trusted_cert -cert C:\SSL\oracle
\oratls_server.cert -pwd oracle4u
```
You can now use the keystore `C:\app\client\ksuwalka\product\12.1.0\client_1\owm\wallet\ewallet.p12` with keystore password `oracle4u`.

Kerberos authentication

You can use Kerberos authentication to connect to Oracle databases by placing the required configuration files on the Secure Agent machine. You can also use Kerberos authentication to connect to SSL-enabled Oracle databases.

When you configure Kerberos authentication to connect to Oracle, 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.

Configuring Kerberos authentication

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

1. To configure the Java Authentication and Authorization Service configuration file (JAAS), perform the following tasks:
   a. Create a JAAS configuration file on the Secure Agent machine.
   b. Add the following entries to the JAAS configuration file:
```
JDBC_DRIVER_01 {
com.sun.security.auth.module.Krb5LoginModule required useTicketCache=true;
};
```
2. 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>
```

3. Set the following environment variables on the Secure Agent machine.
   For more information about the required environment variables, see "Setting environment variables" on page 12.
4. Restart the Secure Agent.
5. To generate the credential cache file on the Secure Agent machine and use Kerberos authentication to connect to Oracle, perform the following tasks:
   a. On the Secure Agent machine, run the following command and specify the Oracle user name and realm name:

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

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

Setting environment variables

To use Kerberos authentication to connect to Oracle, you need to set the required environment variables on the Secure Agent machine.

Set the following environment variables:

- `setenv KRB5CCNAME <Absolute path and file name of the credentials cache file>`
- `setenv KRB5_CONFIG <Absolute path of the Kerberos configuration file>\krb5.conf`
- `setenv JAASCONFIG <Absolute path of the JAAS config file>\<File name>.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 Oracle connection.

To add the environment variables when you configure a connection and use Kerberos authentication, you need to add the *KRB5_CONFIG*, *KRB5CCNAME*, and *JAASCONFIG* properties in the **Metadata Advanced Connection Properties** field in an Oracle 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>;JAASCONFIG=<Absolute path of the JAAS config file>\<File name>.conf
```

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

Connect to Oracle

Let's configure the Oracle connection properties to connect to Oracle databases.

Before you begin

Check out the "Prerequisites" on page 9 to learn about the authentication requirements before you configure a connection.

Connection details

The following table describes the Oracle 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	Type of connection. Select Oracle from the list.
Runtime Environment	The name of the runtime environment where you want to run the tasks. Select a Secure Agent, Hosted Agent, or serverless runtime environment.
Oracle Subtype	The Oracle connection subtype that you can use to connect to Oracle on-premises or Oracle Autonomous Database. Select one of the following options: - Oracle ADB. Connects to Oracle Autonomous Database. - Oracle On-premise. Connects to Oracle on-premises.

Authentication types

You can configure one of the following authentication modes to connect to Oracle databases:

- Oracle Database Authentication: Uses your Oracle user name and password to connect to Oracle.
- Kerberos: Uses Kerberos authentication to connect to Oracle.
  When you choose this option on Windows, ensure that the user account that starts the Secure Agent service is available in the Oracle database. You don't need to enter your credentials to access Oracle.

  Note: You can't configure Kerberos authentication when you use a Hosted Agent or serverless runtime environment.

Select the required authentication type and then configure the authentication-specific parameters.

Default is Oracle Database Authentication.

Oracle Database authentication

The following table describes the basic connection properties for Oracle Database authentication:

Property	Description
User Name	User name for the database login. The user name can't contain a semicolon.
Password	Password for the database login. The password can't contain a semicolon.
Host	Name of the machine that hosts the database server.

Property	Description
Port	Network port number used to connect to the database server. Default is 1521.
Service Name	Service name or System ID (SID) that uniquely identifies the Oracle database. Specify the SID in the following format to connect to Oracle databases: `SID:<ORACLE_SID>`
Schema	Schema used for the Oracle connection.
Code Page	The code page of the database server.

Kerberos authentication

The following table describes the basic connection properties for Kerberos authentication:

Property	Description
Host	Name of the machine that hosts the database server.
Port	Network port number used to connect to the database server. Default is 1521.
Service Name	Service name or System ID (SID) that uniquely identifies the Oracle database. Specify the SID in the following format to connect to Oracle databases: `SID:<ORACLE_SID>`
Schema	Schema used for the Oracle connection.
Code Page	The code page of the database server.

Advanced settings

The following table describes the advanced connection property for Oracle Database authentication:

Property	Description
Encryption Method	The method that the Secure Agent uses to encrypt the data exchanged between the Secure Agent and the database server. Default is No Encryption. This property doesn't apply if you use the Hosted Agent.
Crypto Protocol Version	Cryptographic protocols to use when you enable SSL encryption. Not applicable when you use the Hosted Agent or the serverless runtime environment.
Validate Server Certificate	Validates the certificate that is sent by the database server. If you specify the `HostNameInCertificate` parameter, the Secure Agent also validates the host name in the certificate.

Property	Description
Trust Store	The location and name of the truststore file. For the serverless runtime environment, specify the following certificate path in the serverless agent directory: `/home/cldagnt/SystemAgent/serverless/configurations/ssl_store/<TrustStore_filename>`
Trust Store Password	The password to access the contents of the truststore file.
Host Name in Certificate	Host name of the machine that hosts the secure database. If you specify a host name, the Secure Agent validates the host name included in the connection with the host name in the SSL certificate.
Key Store	The location and the file name of the keystore. For the serverless runtime environment, specify the following certificate path in the serverless agent directory: `/home/cldagnt/SystemAgent/serverless/configurations/ssl_store/<KeyStore_filename>`
Key Store Password	The password for the keystore file required for secure communication.
Key Password	The password for the individual keys in the keystore file required for secure communication.
Connection Retry Period	Number of seconds the Secure Agent attempts to reconnect to the Oracle database if the connection fails. If the Secure Agent can't connect to the database in the retry period, the operation fails. Used for all operations. Default is 0.

Property	Description

Metadata Advanced Connection Properties	Additional properties for the JDBC driver to fetch the metadata. Enter properties in the following format:
	`<parameter name>=<parameter value>`
	If you enter more than one property, separate each key-value pair with a semicolon.
	For example, enter the following property to configure the connection timeout when you test a connection:
	`LoginTimeout=<value_in_seconds>`
	Note: The default connection timeout is 270 seconds.
	To connect to an Oracle database enabled for advanced security, you can specify the Oracle advanced security options for the JDBC driver.
	For example, `EncryptionTypes=AES256;`
	`EncryptionLevel=accepted;DataIntegrityLevel=accepted;`
	`DataIntegrityTypes=SHA1`

Runtime Advanced Connection Properties	Additional properties for the ODBC driver to run mappings.
	If you specify more than one property, separate each key-value pair with a semicolon.
	For example, `charset=sjis;`
	`readtimeout=180`
	To connect to an Oracle database enabled for advanced security, you can specify the Oracle advanced security options for the ODBC driver.
	For example, `EncryptionTypes=AES256;EncryptionLevel=1;`
	`DataIntegrityLevel=1;DataIntegrityTypes=SHA1;`
	`DataIntegrityTypes=SHA1`

Configuring SSL with the serverless runtime environment

You can use the serverless runtime environment with Oracle Connector to connect to an SSL-enabled Oracle database.

Before you configure a secure Oracle connection using the serverless runtime environment, complete the following prerequisite tasks to add the SSL certificates to the serverless runtime location:

1. Create the following structure for the serverless agent configuration in AWS or Azure: `<Supplementary file location>/serverless_agent_config`
2. Add the truststore and keystore certificates in the Amazon S3 bucket or Azure container in the following location in your AWS or Azure account: `<Supplementary file location>/serverless_agent_config/SSL`
3. Copy the following code snippet to a text editor:

```
version: 1
agent:
  agentAutoApply:
    general:
      sslStore:
        - fileCopy:
            sourcePath: SSL/<TrustStore_filename>
```

```
- fileCopy:
    sourcePath: SSL/<KeyStore_filename>
```

where the source path is the directory path of the certificate files in AWS or Azure.

4. 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 SSL certificates are copied from the AWS or Azure location to the serverless agent directory.
5. In the Oracle connection properties, specify the following certificate path in the serverless agent directory in the **Trust Store** and **Key Store** fields: `/home/cldagnt/SystemAgent/serverless/configurations/ssl_store/<cert_filename>`

Oracle connection rules and guidelines

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

- An Oracle table name can have a maximum of 30 characters.
- When you run a task that contains an Oracle database in the Public schema, ensure that the schema does not contain too many objects. If the schema contains too many objects, the task times out. You can remove some objects from the Oracle database or move the objects into another database schema.
- When you run a task for an Oracle database target, ensure that the UTF-8 characters do not exceed the maximum length of the varchar or char fields. Data Integration might truncate UTF-8 characters if they exceed the maximum length of the varchar or char fields.
- If the data that you write from a flat file to Oracle contains Portugese characters, ensure that the code page for the Oracle server is not set to MS Windows Latin 1. Select the code page as ISO 8859-15 Latin 1 when you create an Oracle connection.
- Schema name is case sensitive when the name of the schema contains hyphens.

CHAPTER 3

Synchronization tasks with Oracle Connector

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

You can configure a synchronization task using the Synchronization Task wizard. You can use Oracle objects as sources, targets, or lookup objects.

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.

Oracle sources in synchronization tasks

When you configure a synchronization task to use an Oracle source, you can read data from a single object, multiple related objects, or a saved query. To read data from multiple sources, you must define relationships based on key columns or create a user-defined join condition. To optimize performance, you can configure a filter in the **Data Filters** tab. You can add multiple Oracle objects that have an explicit relationship defined in Oracle.

You can also create a data filter, where you can select an object, field, operator, and enter a filter value based on which the synchronization task runs. You can also configure an advanced filter to define a more complex filter condition, which can include multiple conditions using the AND or OR logical operators.

The following table describes the Oracle source properties:

Property	Description
Connection	Name of the active source connection.
Source Type	Type of the source object available. You can choose from the following source types: - Single - Multiple - Saved query
Source Object	Name of the source object.
Display source fields in alphabetical order	Displays source fields in alphabetical order. By default, fields appear in the order returned by the source system.

Oracle targets in synchronization tasks

You can use a single object as a target in a synchronization task.

The following table describes the Oracle target properties:

Property	Description
Connection	Name of the active target connection.
Target Object	Name of target objects available in the connection.
Truncate Target	When you enable the **Truncate Target** option, the Secure Agent truncates the table. Default is false.
Enable Target Bulk Load	Uses Oracle bulk API to write data in bulk mode to Oracle. Default is unselected.
Display target fields in alphabetical order	Displays target fields in alphabetical order. By default, fields appear in the order returned by the target system.

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**.

Advanced properties for Oracle sources and targets in synchronization tasks

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

The following table describes the advanced properties that you can configure for an Oracle source or target:

Property	Description
Preprocessing Commands	Enter the pre-processing script that should be executed before running the synchronization task.
Post-processing Commands	Enter the post-processing script that should be executed after running the synchronization task.
Parameter File Name	Name of the file that contains the parameters to be used in filters or expressions.
Maximum Number of Log Files	Enter the maximum number of log files that the Secure Agent generates until the number reaches the limit that you set. Default is 10.

Oracle lookups in synchronization tasks

When you configure field mappings in a synchronization task, you can create a lookup to an Oracle object. When you use an Oracle object as a lookup, you do not need to configure specific Oracle properties.

Data filters

You can create simple or advanced data filters. You can also create a set of data filters for each object included in a replication task or synchronization task. Each set of data filters act independently of the other sets.

Simple data filters

You can create one or more simple data filters. When you create multiple simple data filters, the associated task creates an AND operator between the filters and loads rows that apply to all simple data filters.

Advanced data filters

You can create an advanced data filter to create complex expressions that use AND, OR, or nested conditions. The expression that you enter becomes the WHERE clause in the query used to retrieve records from the source.

Rules and guidelines for data filters

Use the following rules and guidelines when you configure data filters:

- If you include UTF-8 characters in a data filter on an Oracle source, the data filter does not work.

- If you create a simple data filter on a date field in an Oracle source, the date field must use the default Oracle format **DD-MON-YY**. To create a filter on a date field with a non-default date format, use an advanced data filter.
  For example, to read an Oracle date field that has the date format '**yyyy/mm/dd:hh:mi:ssam**', use the following advanced filter:

  “DATE_FIELD” > to_date('1998/05/31:12:00:00AM', 'yyyy/mm/dd:hh:mi:ssam')

CHAPTER 4

Mappings and mapping tasks with Oracle Connector

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

Oracle sources in mappings

To read data from an Oracle database, configure an Oracle object as the Source transformation in a mapping.

Enable partitioning when you configure the Source transformation in the Mapping Designer to optimize the performance of the mapping task.

Specify the name and description of the Oracle source. Configure the source, query options, and advanced properties for the source object.

The following table describes the properties that you can configure for an Oracle source:

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 the Oracle source object available. You can choose from the following source types: - Single - Multiple - Query - 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 for the mapping. You can specify a custom query for a source object.
Filter	Configure a simple filter or an advanced filter to remove rows at the source. You can improve efficiency by filtering early in the data flow. A simple filter includes a field name, operator, and value. Use an advanced filter to define a more complex filter condition, which can include multiple conditions using the AND or OR logical operators.
Sort	Select the fields and type of sorting to use. To sort data for a parameterized source, you must use a parameter for the sort options.
Select distinct rows only	Select this option to read only distinct rows.

The following table describes the advanced properties that you can configure for an Oracle source:

Advanced Property	Description
Tracing Level	Amount of detail that appears in the log for this transformation. You can choose terse, 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.
SQL Override	The SQL statement to override the default query generated from the specified source type to read data from the Oracle source.
Output is deterministic	Relational source or transformation output that does not change between session runs when the input data is consistent between runs. When you configure this property, the Secure Agent does not stage source data for recovery if transformations in the pipeline always produce repeatable data.
Output is repeatable	Relational source or transformation output that is in the same order between session runs when the order of the input data is consistent. When output is deterministic and output is repeatable, the Secure Agent does not stage source data for recovery.
Schema Name	Overrides the schema name of the source object.

Key range partitioning

You can configure key range partitioning when you use a mapping task to read data from Oracle sources. With key range partitioning, the Secure Agent distributes rows of source data based on the fields that you define as partition keys. The Secure Agent compares the field value to the range values for each partition and sends rows to the appropriate 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
- 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.

Configuring key range partitioning

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

1. In the Source Properties, click the **Partitions** tab.
2. In the **Partition Key** field, select the required partition key from the list.
3. In the **Key Ranges** section, 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.

Oracle targets in mappings

To write data to an Oracle target, configure an Oracle object as the Target transformation in a mapping. Specify the name and description of the Oracle target. Configure the target and advanced properties for the target object.

The following table describes the properties that you can configure for an Oracle target:

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 the Oracle target object available. You can choose from the following target types: - Single - 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	You can select an existing object from the list or create a target at run time.
Operation	Select the target operation. You can perform the following operations on an Oracle target: - Insert - Update - Upsert - Delete - Data Driven
Truncate target	When you enable the **Truncate Target** option, the Secure Agent truncates the table. Default is false.
Enable Target Bulk Load	Uses Oracle bulk API to insert data in bulk mode. When you create a mapping task, you can use the Oracle bulk API to perform insert operation.
Pre SQL	Pre-SQL command to run against the target database before writing data to the target.
Post SQL	Post-SQL command to run against the target database after writing data to the target.
Update Override	An update SQL statement that updates the data in an Oracle 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. You cannot validate the update SQL statement in the SQL transformation. In the override statement, you must enclose all reserved words in quotation marks.

Property	Description
Reject File Directory	The directory that stores the rejected files. Specify the directory where you want to store the rejected files.
Reject File Name	Name of the rejected file that is stored in the reject file directory.
Schema Name	Overrides the schema name of the target object.

If you select the **Forward Rejected Rows** option, the Secure Agent flags the rows for reject and writes them to the reject file. If you do not select the **Forward Rejected Rows** option, the Secure Agent drops rejected rows and writes them to the session log file. The Secure Agent does not write the rejected rows to the reject file.

Note: If there is an error row when you run a task to write data to an Oracle target in bulk mode, the Secure Agent writes the batch that contains the error row in normal mode.

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**.

Enabling array update and upsert operations for an Oracle target

You can configure an Oracle mapping to update or upsert data in batches of arrays.

When you configure an array update or upsert, the Secure Agent clubs all records together into arrays and writes those arrays of data into the Oracle target in batches instead of updating or upserting data row by row. Array update and upsert operations reduce the network traffic and optimizes performance.

To configure an array update or upsert, you must set the following custom properties:

EnableArrayUpsert

Before you upsert data into an Oracle target, configure the **EnableArrayUpsert** custom property in the Secure Agent properties or in the Oracle mapping task properties, and set its value to yes.

EnableArrayUpdate

Before you update data into an Oracle target, configure the **EnableArrayUpdate** custom property in the Secure Agent properties or in the Oracle mapping task properties, and set its value to yes.

Note: You cannot array update or upsert data to an Oracle target when you select the data driven operation in an Oracle target transformation.

Oracle lookups in mapping

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

When you configure a lookup in Oracle, you select the lookup connection and lookup object.

You can also set the default column value for the return field in a Lookup transformation in an Oracle mapping.

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

Property	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 Oracle 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.
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 lookup object or click **New Parameter** to define a new parameter for the lookup 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.
Lookup Object	Name of the Oracle lookup object.
Multiple Matches	The behavior when the lookup condition returns multiple matches. You can select one of the following options: - Return first row - Return last row - Return any row - Return all rows - Report error
SQL Override	The SQL statement to override the default query that creates lookup data from an Oracle source.
Schema Name	Overrides the schema name of the lookup object.

SQL ELT optimization

When you read data from an Oracle 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 a 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.

SQL ELT optimization improves mapping performance when the database can process the transformation logic faster than the Secure Agent. The Secure Agent also reads less data from the database.

The amount of transformation logic that the Secure Agent pushes to the database depends on the database, the transformation logic, and the mapping task. The Secure Agent processes all transformation logic that it cannot push to a database.

When you configure SQL ELT optimization for the 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.

Note: When you push 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.

The Secure Agent can push the following transformation logic to an Oracle source or target:

Transformations	Supported SQL ELT optimization Type
Aggregator	Source, Full
Expression	Source, Full
Filter	Source, Full
Joiner	Source, Full
Sorter	Source, Full
Union	Source, Full
Router	Full

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 in the **SQL ELT Optimization** section.

Full SQL ELT optimization is ideal when the source and target are in the same connections. For example, if a mapping contains an Oracle source and an Oracle target, configure full SQL ELT optimization to push all the transformation logic for processing from an Oracle source database to an Oracle target database.

Source SQL ELT optimization

When the Secure Agent applies source SQL ELT optimization, it analyzes the mapping from source to target or until it reaches a downstream transformation it can't push to the source database.

The Secure Agent generates and executes a SELECT statement based on the transformation logic for each transformation it can push to the database. Then, it reads the results of this SQL query and processes the remaining transformations.

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 an Oarcle source and a Microsoft SQL Server target, you can configure source SQL ELT optimization to push some transformation logic for processing to the Oracle source.

SQL ELT optimization functions

When you use SQL ELT optimization, the Secure Agent converts the expression in the transformation by determining equivalent functions and pushes down the expression to process in the database. If there is no equivalent function in the database, the Secure Agent processes the transformation logic.

The following table summarizes the availability of SQL ELT optimization functions in Oracle:

Functions	SQL ELT optimization Type
ABS()	Source, Full
AVG()	Source, Full
COS()	Source, Full
COUNT()	Source, Full
DATE_COMPARE()	Source, Full
DECODE()	Source, Full
EXP()	Source, Full
IIF()	Source, Full
IN()	Source, Full
ISNULL()	Source, Full
LOWER()	Source, Full
MAX()	Source, Full
MIN()	Source, Full
POWER()	Source, Full
SIN()	Source, Full
SQRT()	Source, Full

Functions	SQL ELT optimization Type
SUM()	Source, Full
TAN()	Source, Full
UPPER()	Source, Full

Supported functions for Expression transformation

When you configure SQL ELT optimization using an Expression transformation, the Secure Agent tries to push the configured Expression transformation to Oracle.

The following table summarizes the availability of SQL ELT optimization functions in an Expression transformation in Oracle:

Functions	SQL ELT optimization Type
ADD_TO_DATE()	Full
CEIL()	Full
CONCAT()	Full
COSH()	Full
DATE_COMPARE()	Full
FLOOR()	Full
GET_DATE_PART()	Full
INSTR()	Full
LENGTH()	Full
LOG()	Full
LTRIM()	Full
MOD()	Full
ROUND(NUMBER)	Full
RTRIM()	Full
SIGN()	Full
SINH()	Full
STDDEV()	Full
SUBSTR()	Full

Functions	SQL ELT optimization Type
SYSDATE()	Full
SYSTIMESTAMP()	Full
TANH()	Full
TO_BIGINT	Full
TO_CHAR(DATE)	Full
TO_CHAR(NUMBER)	Full
TO_DATE()	Full
TO_DECIMAL()	Full
TO_FLOAT()	Full
TO_INTEGER()	Full
TRUNC(NUMBER)	Full
VARIANCE()	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.

1. In the **Runtime Options** page 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.

Cross-schema SQL ELT Optimization

You can use cross-schema SQL ELT Optimization for a mapping task to read from or write data to Oracle objects associated with different schemas within the same database.

To use cross-schema SQL ELT Optimization, create two Oracle connections and specify the schema in each connection. Ensure that the schema in the source connection is different from the schema in the target connection, but both the schemas must belong to the same database. When you configure SQL ELT Optimization for the mapping task, enable cross-schema SQL ELT Optimization in the advanced session properties. By default, the check box is selected.

Configuring cross-schema SQL ELT Optimization

Consider the following steps to configure cross-schema SQL ELT Optimization for an Oracle mapping task:

1. Create the following two Oracle connections, each defined with a different schema:
   a. Create an `orc_1` Oracle connection and specify `CQA_SCHEMA1` schema in the connection properties.
   b. Create `orc_2` Oracle connection and specify `CQA_SCHEMA2` schema in the connection properties.

2. Create an Oracle mapping, `m_Orc_pdo_acrossSchema`. Perform the following tasks:
   a. Add a Source transformation and include an Oracle source object and connection `orc1` to read data using `CQA_SCHEMA1`.
   b. Add a Target transformation and include an Oracle target object and connection `orc2` to write data using `CQA_SCHEMA2`.
3. Create an Oracle mapping task, and perform the following tasks:
   a. Select the configured Oracle mapping, `m_Orc_pdo_acrossSchema`.
   b. On the **Runtime Options** tab, in the **SQL ELT Optimization** section, set the SQL ELT Optimization value to **Full**.
   c. In the **Advanced Session Properties** section, select the **Enable cross-schema SQL ELT Optimization** check box.
   d. Save the task and click **Finish**.

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

Rules and guidelines for Oracle sources and targets in tasks

Use the following rules and guidelines for Oracle sources and targets in a task:

- You can use database tables, synonyms, or views as sources and targets.
- When you create a saved query to read data from an Oracle database, you must omit the final semicolon from the SQL statement.
- You can't use the `select * from <TABLE_NAME>` SQL command in a saved query. You must provide the explicit column list.
- If there is an error row when you run a task to write data to an Oracle target in bulk mode, the Secure Agent writes the batch that contains the error row in normal mode.
- Ensure that the Oracle target table name does not contain mixed case identifiers.
- When you run a mapping task to create a target, the target is successfully created at runtime. If you drop the created target table and run the task again, the task fails.
- When you add objects with a hyphen in the column names in the Source and Target transformations, consider the following rules and guidelines:
  - When the source column name that you read has hyphens and you use the **Create New at Runtime** option to create a target, the hyphens in the table name are converted to underscores in the target column.
  - If you add a new column to the object in an existing mapping and the column name has a hyphen, manually adding these columns in the mapping is not applicable.
  - If you parameterize the target object, the manual field map option in the **Field Mapping** tab is not applicable. You must select Automatic, Completely Parameterized, or Partially Parameterized as the field map options for the mapping to work successfully.
- To write data in a sequence to an Oracle target in bulk mode, you can specify the following query syntax while creating an Oracle table:

```
CREATE TABLE <TABLE_NAME> (<COLUMN_NAME> <COLUMN_DATATYPE GENERATED BY DEFAULT ON NULL AS
COLUMN_ALIAS_NAME>, <COLUMN_NAME2> <COLUMN_DATATYPE2>,...<COLUMN_NAME30>
<COLUMN_DATATYPE30>)
```

Ensure that the columns that you want to use in bulk mode must be of the integer data type.

- You can't write data to an Oracle target when you configure a source partitioning and run the mapping in bulk mode.
- When you read from Oracle, ensure that the Oracle table name or column name does not contain Unicode characters.
- If you want to override the source or target table name, don't enter the name in the <SchemaName>.<TableName> format in the **Table Name** field. Enter the schema name and table name values separately in the **Schema Name** and **Table Name** fields in the advanced properties section.

Consider the following rules and guidelines when you use Oracle Autonomous Database sources and targets:

- When you configure partitioning in bulk mode to write to Oracle Autonomous Database, use an existing target instead of creating a new target at run time. Otherwise, performance issue occurs.
- To run a mapping that contains more than three source partitions, you must set the **Service Name** property value to **low**.
  For more information about service names for Oracle Autonomous Database, see the Oracle Autonomous Database documentation.

CHAPTER 5

Replication tasks

You can replicate Oracle data to a target by using the replication task. You might replicate data to back up the data or perform offline reporting. You can replicate data in Oracle objects to databases or flat files.

A replication task can replicate data from one or more Oracle objects. When you configure the task, you can replicate all available objects through the selected connection, or you can select objects for replication by including or excluding a set of objects. You can also exclude rows and columns from the replication task. Associate a schedule with a replication task to specify when and how often the task runs.

When you replicate Oracle sources, you can replicate all current rows in the Oracle source. The names of source tables and fields can contain at most 30 characters. For example, the replication task creates a column name in an Oracle database based on the following 40-character field name:

```
TenLetters1234567890TenLettersXXXXXXXXXX
```

The replication task truncates the column name to the first 30 characters:

```
TenLetters1234567890TenLetters
```

If the truncation causes duplicate column names, the replication task replaces the last character of the duplicate column names with sequential numbers. The replication task also replaces the last character of duplicate table names from the same task.

Oracle sources in replication tasks

When you replicate a source object to a database, the replication task replicates the data to a database table with the same name as the source object.

The following table describes the Oracle source properties in a replication task:

Property	Description
Source Connection	Name of the active source connection. Select a source connection or create a new connection.
Objects to Replicate	Source objects that you can replicate. You can choose from the following options to replicate the objects: - All Objects - Include Objects - Exclude Objects
If an error occurs while processing an object	Determines the execution or termination of the process when an error occurs.

Oracle targets in replication tasks

When you replicate data to a database target, the replication task truncates the table name in the target database if the source object name exceeds 30 characters.

The following table describes the Oracle target properties in a replication task:

Property	Description
Connection	Name of the target connection. Select a target connection or create a new connection.
Target Prefix	String that prefixes the source object names to create names for the target objects in the target. **Note:** For more information, see "*Truncating Table Names*" in Tasks help.
Enable Target Bulk Load	Uses Oracle bulk API to replicate data in bulk mode.
Load Type	Replicates full load of data from Oracle to other applications.
Delete Options	Not Applicable.
Commit size	Used for the replication task for all the runs. Defines the number of rows to commit. If you do not specify a value, the Secure Agent uses the default value 10000.

Full load

For a full load, the replication task replicates the data for all rows of the source objects in the task. Each time the task runs, the replication task truncates the target database tables or flat file and performs a full data refresh from the source.

Rules and guidelines for Oracle sources and targets in replication tasks

Use the following rules and guidelines for Oracle sources and targets in a replication task:

- You cannot configure multiple replication tasks to replicate the same source object to the same target object.
  For example, you cannot configure two replication tasks to write Salesforce Account data to the SF_ACCOUNT Oracle database table.
- If you configure a replication task to replicate all objects from the Public schema and the schema contains synonyms, the task might fail. The task fails when the user account for the Oracle connection does not have the Select privilege on the synonyms in the Public schema. The task also fails if the synonym names do not begin with a letter or underscore (_).

- If you replicate timestamp data to an Oracle database, the replication task truncates fractional seconds to the second.
  For example, the replication task truncates 12:52:47.58 to 12:52:47.
- When you configure replications tasks for data that contains the Clob or Blob data types, the Blob data type is truncated to 32000 bytes and the Clob data type is truncated to 32000 characters.

CHAPTER 6

Troubleshooting

Use the following sections to troubleshoot errors in Oracle Connector.

For a list of common error messages and possible solutions, see the article, "Troubleshooting: Common Error Messages".

Troubleshooting an Oracle connection

The performance of a task that uses an Oracle connection is not optimal.

You can configure certain properties in the Secure Agent to optimize the performance of Oracle Connector.

The following table describes the properties that you can set:

Property	Description	Parameters
Array size	Add the OdbcOracleDefaultArraySize property to increase the array size. Set the value of the property to an appropriate array size, in bytes, based on the amount of data that you want to read or write. Default is 6000000 bytes. **Note:** This property does not optimize performance for custom SQL queries configured in the mapping.	- Service: Data Integration Server - Type: DTM - Name: OdbcOracleDefaultArraySize - Value: Array size in bytes. For example, 7000000.
Optimize Oracle write	Configure the Data Integration Server OptimizeOracleSQLServerWrite property to optimize the write operation. Default is No.	- Name: OptimizeOracleSQLServerWrite - Value: Yes
Enable array upsert	Add the EnableArrayUpsert property to optimize batch upserts using arrays to an Oracle target.	- Service: Data Integration Server - Type: DTM - Name: EnableArrayUpsert - Value: Yes
Enable array update	Add the EnableArrayUpdate property to optimize batch updates using arrays to an Oracle target.	- Service: Data Integration Server - Type: DTM - Name: EnableArrayUpdate - Value: Yes

ERROR: keytool error: java.security.KeyStoreException: TrustedCertEntry not applicable.

You must use Java 1.8 to create a **PKCS12** truststore certificate. You cannot create a **PKCS12** truststore certificate using Java 1.7.

Troubleshooting a synchronization task

"Error Log Table Name Prefix" occurs which specifies table name prefix for a relational error log.

Oracle has a 30 character limit for table names. The task fails if a table name exceeds 30 characters.

Troubleshooting a replication task

[DataDirect][ODBC Oracle Wire Protocol driver]Fractional truncation. Error in parameter <number>.

The database driver issues a warning message if a replication task replicates all Salesforce objects and tries to write a null value to a numeric field of an Oracle database.

An error message appears while configuring the target for the replication task.

You cannot configure multiple replication tasks to replicate the same source object to the same target object.

For example, you cannot configure two replication tasks to write Salesforce Account data to the SF_ACCOUNT Oracle database table. An error appears when you select the same target connection and target object that is used by another replication task.

If you replicate specific objects for the source connection, the following error appears: Duplicate replication task for <object name> object with a target prefix of <target prefix>.

An error occurs when you run two replication tasks to replicate two objects whose target prefixed name is the same. You must change the target connection, enter a different target prefix or remove the object from the task.

To write data from sources with the same name to different target objects in the same target connection, you must configure different target prefixes.

If you replicate all objects for the source connection, the following error appears: Duplicate replication task.

An error occurs when you run two replication tasks to replicate the same object from the same source to the same target with the same target prefix. You must change the target or enter a different target prefix.

Numeric data is loaded incorrectly when you run a replication task that writes Oracle data to a flat file.

When a replication task writes Oracle data with the Number data type to a flat file with the scale as 0, the number is rounded with a precision of 15. The replication application converts the Number data type to Double (15).

To prevent rounding, specify a scale other than 0 for the source data type. When you specify a non-zero scale, rounding will not occur because the replication application converts the Number data type to Decimal.

APPENDIX A

Data type reference

Data Integration uses the following data types in mappings and tasks with Oracle:

Oracle native data types

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

Transformation data types

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

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

Oracle and transformation data types

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

Oracle Data Type	Transformation Data Type	Description
Blob	Binary	1 to 104,857,599 characters
Char	String	1 to 104,857,600 characters
Clob	Text	1 to 104,857,599 characters
Date	Date/Time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D
Number	Double	Precision 15, scale 0
Number(p,s)	Decimal	Precision of 1 to 28, scale of 0 to 28
Timestamp	Date/Time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D (precision to the nanosecond)
Varchar	String	1 to 104,857,600 characters
Varchar2	String	1 to 104,857,600 characters

Oracle Data Type	Transformation Data Type	Description
nChar	String	1 to 104,857,600 characters
nVarchar2	String	1 to 104,857,600 characters

Rules and guidelines for data types

Consider the following rules and guidelines for certain Oracle data types:

- The default precision for reading or writing Clob and Blob data types is 32,000. If you configure a mapping to read or write data with Blob or Clob data types whose precision is more than 32,000, you must manually increase the precision of these data types in the mapping accordingly.
- When you read or write data that contains the Clob and Blob data types, ensure that the row size does not exceed 100 MB.
- Do not use a SQL transformation to read or write Clob and Blob data types where the row size exceeds 100 MB.
- When you run a saved query in a synchronization task, ensure that the precision of the Clob and Blob data types is not equal or more than 100 MB.
- When you use an Oracle source with a Salesforce target, if the source contains fields with the Number data type, change the field type to numeric. Values for fields with the Number data type do not load correctly in Salesforce. You can change the type on the **Field Mapping** page. If you have many fields with the Number data type, you can add the `oracle.use.varchar.for.number` custom property for the Secure Agent.
- Do not include NChar and Nvarchar2 columns in Oracle targets. Due to an ODBC driver limitation, the PowerCenter task truncates the last half of NChar and Nvarchar2 data before writing it to Oracle targets.

INDEX

C

Cloud Application Integration community
  URL 5
Cloud Developer community
  URL 5
configuring key range
  partitioning 24
connections
  Oracle 12

D

data filters
  advanced 20
  rules and guidelines 20
  simple 20
Data Integration community
  URL 5
data type reference
  overview 39

E

environment variables
  Kerberos authentication 12

I

Informatica Global Customer Support
  contact information 6
Informatica Intelligent Cloud Services
  web site 5

K

Kerberos authentication
  Oracle 11
keystore certificate
  creation 10

M

maintenance outages 6
mapping
  mapping task 22

O

Oracle
  connection properties 12
  data types 39
Oracle advanced properties
  Synchronization task 19
Oracle connections
  overview 9
Oracle Connector
  assets 7
  example 8
  overview 7
  Synchronization task 18
Oracle mapping
  source properties 22
  target properties 25
Oracle sources
  mapping 27

P

Partitioning
  key range 24

R

replication
  source properties 34
  target properties 35
Replication
  rules and guidelines 35
  sources and targets 35
replication task
  full load 35
rules and guidelines
  tasks 32

S

SQL ELT optimization
  functions 29, 30
SQL ELT Optimization
  Configuration 31
status
  Informatica Intelligent Cloud Services 6
Synchronization task
  sources 18
  targets 19
system status 6

T

troubleshooting
  Oracle connection 37
  Replication task 38
  Synchronization task 38
trust site
  description 6
trust store certificate
  creation 9

U

upgrade notifications 6

W

web site 5