



Informatica®

Informatica® Cloud Data Integration

Open Table Connector

© Copyright Informatica LLC 2024, 2025

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: 2025-01-10

Table of Contents

Preface.	4
Informatica Resources.	4
Informatica Documentation.	4
Informatica Intelligent Cloud Services web site.	4
Informatica Intelligent Cloud Services Communities.	4
Informatica Intelligent Cloud Services Marketplace.	4
Data Integration connector documentation.	4
Informatica Knowledge Base.	5
Informatica Intelligent Cloud Services Trust Center.	5
Informatica Global Customer Support.	5
Chapter 1: Introduction to Open Table Connector.	6
Open Table Connector assets.	6
Chapter 2: Connections for Open Table.	7
Prerequisites.	7
Create minimal IAM policies.	7
Install the JDBC driver.	9
Connect to Open Table.	9
Before you begin.	9
Open Table formats with associated catalog and storage types.	10
Connection details.	10
Catalog types.	10
Storage types.	11
Chapter 3: Mappings and mapping tasks with Open Table Connector.	13
Open Table sources in mappings.	13
Open Table targets in mappings.	15
Create a target table at runtime.	17
Open Table lookups in mappings.	18
Using time travel queries.	20
Dynamic schema handling.	20
Rules and guidelines for dynamic schema handling.	21
Rules and Guidelines for Open Table tasks.	21
Chapter 4: Data type reference.	22
Open Table and transformation data types.	22
Index.	24

Preface

Use *Open Table Connector* to learn how to read from or write to Apache Iceberg or Delta Lake tables by using Cloud Data Integration. Learn to create an Open Table connection, develop and run mappings, and mapping tasks in Cloud Data Integration.

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 Open Table Connector

You can use Open Table Connector to connect to a catalog from Data Integration to read from and write data to Open Table formats.

You can create an Open Table connection and use it in mappings in advanced mode to securely read from or write data to Apache Iceberg or Delta Lake Open Table formats managed by the AWS Glue Catalog or Hive Metastore catalog.

Open Table Connector assets

Create assets in Data Integration to integrate data using Open Table Connector.

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

- Mapping
- Mapping task

Important: You can use Open Table Connector only in mappings in advanced mode. You can run a task based on a mapping in advanced mode.

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

CHAPTER 2

Connections for Open Table

Create an Open Table connection to securely read from or write data to Open Table formats available in a catalog.

You can use an Open Table connection to specify sources, targets, and lookups in mappings and mapping tasks.

Prerequisites

Before you create an Open Table connection, complete the prerequisites.

If you use AWS Glue Catalog and Amazon S3 Storage to interact with Apache Iceberg or Delta Lake tables, you need to have access to the following AWS services that manage the tables on AWS:

- **AWS Glue Catalog:** AWS Glue Catalog manages the metadata associated with the Apache Iceberg or Delta Lake tables.
- **Amazon S3 Storage:** Amazon S3 stores the Apache Iceberg or Delta Lake tables containing actual records in columnar format, organized in partitioned directories.
- **Amazon Athena JDBC driver:** Amazon Athena JDBC driver connects to the AWS Glue Catalog to access Apache Iceberg or Delta Lake tables metadata and perform SQL queries on data stored in Amazon S3 storage.

You need to create separate policies to access these services.

If you use Hive Metastore catalog and Microsoft Azure Delta Lake Storage Gen2 to interact with Apache Iceberg tables, you need to have access to the following services that manage the tables on Microsoft Azure Delta Lake Storage Gen2:

- **Hive Metastore catalog:** Hive Metastore catalog manages the metadata associated with the Apache Iceberg tables.
- **Microsoft Azure Delta Lake Storage Gen2:** Microsoft Azure Delta Lake Storage Gen2 stores the Apache Iceberg tables containing actual records in columnar format, organized in partitioned directories.
- **Hive JDBC driver:** Hive JDBC driver connects to the Hive Metastore catalog to access Apache Iceberg tables metadata and perform SQL queries on data stored in Microsoft Azure Delta Lake Storage Gen2.

Create minimal IAM policies

You need to create IAM policies with the minimum required permissions to interact with Apache Iceberg or Delta Lake tables managed by AWS Glue Catalog. For more information on configuring these policies, refer to the AWS documentation.

Minimum policy for Amazon Athena

The following sample policy shows the minimal Amazon IAM policy to access Amazon Athena:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": [
        "athena:CreatePreparedStatement",
        "athena:GetPreparedStatement",
        "athena:GetWorkGroup",
        "athena:GetTableMetadata",
        "athena:StartQueryExecution",
        "athena:GetQueryResultsStream",
        "athena:ListDatabases",
        "athena:GetQueryExecution",
        "athena:GetQueryResults",
        "athena:GetDatabase",
        "athena:ListTableMetadata",
        "athena:GetDataCatalog",
        "athena>DeletePreparedStatement"
      ],
      "Resource": [
        "arn:aws:athena:*:*:workgroup/*",
        "arn:aws:athena:*:*:datacatalog/*"
      ]
    },
    {
      "Sid": "VisualEditor1",
      "Effect": "Allow",
      "Action": [
        "athena:ListDataCatalogs",
        "athena:GetQueryExecution",
        "athena:ListWorkGroups",
        "athena:GetPreparedStatement"
      ],
      "Resource": "*"
    }
  ]
}
```

Minimum policy for AWS Glue

The following sample policy shows the minimal Amazon IAM policy to access AWS Glue Catalog:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "glue:*"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}
```

Minimum policy for AWS S3

The following sample policy shows the minimal Amazon IAM policy to read from or write data to an Amazon S3 bucket:

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



```

        "Sid": "VisualEditor0",
        "Effect": "Allow",
        "Action": [
            "s3:PutObject",
            "s3:GetObject",
            "s3:ListBucket",
            "s3:DeleteObject"
        ],
        "Resource": "*"
    },
    {
        "Effect": "Allow",
        "Action": [
            "s3:ListBucket",
            "s3:GetBucketLocation",
            "s3:ListAllMyBuckets",
            "s3:GetBucketAcl"
        ],
        "Resource": [
            "*"
        ]
    }
]
}

```

Install the JDBC driver

Before you use Open Table Connector, you need to copy the Amazon Athena or Hive JDBC driver on the Windows or Linux machine where you installed the Secure Agent.

1. Download the latest Amazon Athena or Hive JDBC driver from the website.
2. Navigate to the following directory on the Secure Agent machine: <Secure Agent installation directory>/ext/connectors/thirdparty/
3. Create the following folder: `informatica.opentableformat/common`
4. Add the JDBC driver to the folder.
5. Restart the Secure Agent.

Connect to Open Table

Let's configure the Open Table connection properties to connect to AWS Glue Catalog or Hive Metastore.

Before you begin

Before you get started, you will need to add the Amazon Athena or Hive JDBC driver to the Secure Agent machine and configure the authentication-specific prerequisites.

Permanent IAM Credentials authentication requires the access key and secret key values of the IAM user. Keep the access key and secret key handy before creating the connection. For more information about creating an access key and secret key, see the AWS documentation.

To configure Service Principal authentication, you need the Azure account name, client secret, client ID, and tenant ID for your application registered in the Azure Active Directory. Keep the Azure account name, client secret, client ID, and tenant ID handy before creating the connection. For more information about Azure account name, client secret, client ID, and tenant ID for your application, see the Microsoft Azure documentation.

Check out [“Prerequisites” on page 7](#) to learn more about how to configure policies and role to access Apache Iceberg or Delta Lake tables.

Open Table formats with associated catalog and storage types

You can choose the Open Table format that you want to use and its associated catalog type and storage type to interact with data.

The following table summarizes the Open Table formats that you can use, their catalog types, storage types, and the authentication options available for each storage type:

Open Table format	Catalog type	Storage type	Authentication type
Apache Iceberg	AWS Glue Catalog	Amazon S3	Permanent IAM Credentials authentication
	Hive Metastore	Microsoft Azure Delta Lake Storage Gen2	Service Principal authentication
Delta Lake	AWS Glue Catalog	Amazon S3	Permanent IAM Credentials authentication

Connection details

The following table describes the Open Table 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	Open Table
Runtime Environment	The name of the runtime environment where you want to run tasks. Select a Secure Agent, Hosted Agent, or serverless runtime environment.
Open Table Format	The Open Table format that you want to use to read from or write data to a catalog. Select Apache Iceberg or Delta Lake from the list.

Catalog types

You can select AWS Glue Catalog or Hive Metastore as the catalog type to manage the metadata of the Open Table format that you selected.

Select the catalog type that your Open Table format uses and then configure the catalog specific parameters.

AWS Glue Catalog

If the Apache Iceberg or Delta Lake Open Table format uses AWS Glue Catalog as the catalog type, configure the properties specific to AWS Glue Catalog.

The following table describes the property to configure AWS Glue Catalog:

Property	Description
Athena JDBC URL	Enter the JDBC URL in the following format: <code>jdbc:athena://Region=<AWS_Region>;OutputLocation=<S3_Location></code> For example, <code>jdbc:athena://Region=us-west1;OutputLocation=s3://working/dir.</code>

Hive Metastore

If the Apache Iceberg Open Table format uses Hive Metastore as the catalog type, configure the properties specific to Hive Metastore.

The following table describes the properties to configure Hive Metastore:

Property	Description
Hive Metastore URI	The Hive thrift server URL to connect to Hive Metastore.
Hive JDBC URL	The JDBC URL to connect to Hive4 server.
Hive User Name	The user name of your Hive account to connect to Hive Metastore.
Hive Password	The password of your Hive account to connect to Hive Metastore.

Storage types

You can choose Amazon S3 or Microsoft Azure Data Lake Storage Gen2 as the storage type to store the Open Table format tables.

Select the storage type and configure the storage specific authentication parameters.

Amazon S3 Storage

If you use AWS Glue Catalog as the catalog type, configure the properties specific to Amazon S3 Storage.

Select Permanent IAM Credentials authentication as the authentication type to access Open Table formats in Amazon S3 Storage.

Permanent IAM Credentials authentication

The following table describes the properties to configure Permanent IAM Credentials authentication:

Property	Description
Access Key	The key to access the AWS Glue Catalog.
Secret Key	The secret key to access the AWS Glue Catalog. The secret key is associated with the access key and uniquely identifies the account.

Microsoft Azure Data Lake Storage Gen2

If you use Hive Metastore as the catalog type, configure the properties specific to Microsoft Azure Data Lake Storage Gen2.

Select Service Principal authentication as the authentication type to access Open Table formats in Microsoft Azure Data Lake Storage Gen2.

Service Principal authentication

The following table describes the properties to configure Service Principal authentication:

Property	Description
Azure Account Name	The name of the Microsoft Azure Data Lake Storage Gen2 account to stage the files.
Azure Client Secret	The client secret for your application.
Azure Client ID	The client ID of your application. Enter the application ID or client ID for your application registered in the Azure Active Directory.
Azure Tenant ID	The directory ID or tenant ID for your application.

CHAPTER 3

Mappings and mapping tasks with Open Table Connector

Use the Data Integration Mapping Designer to define and configure a mapping in advanced mode for Open Table sources and targets.

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

When you run a mapping in advanced mode, Data Integration analyzes the data logic defined in the mapping to automatically assign data logic to run on an advanced cluster.

You can use Monitor to monitor the job.

Open Table sources in mappings

When you configure a mapping in advanced mode to use an Open Table source, you can configure the source properties.

Specify the name and description of the Open Table source. Configure the source and advanced source properties for the Open Table object.

The following table describes the Open Table 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 run time, select the Allow parameter to be overridden at run time option.
Source Type	Type of the Open Table source object. You can choose from the following source types: <ul style="list-style-type: none">- Single Object. Select to specify a single Open Table object.- Multiple Objects. This property does not apply to Source transformation.- Parameter. Select to specify a parameter name. You can configure the source object in a mapping task associated with a mapping that uses this Source transformation.

Property	Description
Parameter	<p>A parameter file where you define values that you want to update without the need 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 run time, select the Allow parameter to be overridden at run time option.</p> <p>When the task runs, the Secure Agent uses the parameters from the file that you specify in the advanced session properties.</p>
Object	Source object for the mapping.

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

Property	Description
Filter	<p>Filter value in a read operation. Click Configure to add conditions to filter records and reduce the number of rows that the Secure Agent reads from the source.</p> <p>You can specify the following filter conditions:</p> <ul style="list-style-type: none"> - Not Parameterized. Use a basic filter to specify the object, field, operator, and value to select specific records. - Completely Parameterized. Use a parameter to represent the field mapping. - Advanced. Use an advanced filter to define a more complex filter condition that uses the Open Table query format.
Sort	Not applicable

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

Property	Description
Iceberg Spark Properties	<p>The properties that you want to configure for the Iceberg tables.</p> <p>Enter the properties in the following format:</p> <pre><parameter name>=<parameter value></pre> <p>If you enter more than one property, enter each property in a new line.</p> <p>For example,</p> <pre>s3.access-points.<bucket name>=<S3 bucket ARN></pre> <pre>BucketRegion=<bucket-region-name></pre> <p>Note: The S3 bucket ARN property is mandatory when the source and target are in different regions. For more information, see the Create the ARN for your AWS S3 bucket KB article.</p>
Delta Spark Properties	<p>The properties that you want to configure for the Delta Lake tables.</p> <p>Enter the properties in the following format:</p> <pre><parameter name>=<parameter value></pre> <p>If you enter more than one property, enter each property in a new line.</p> <p>For example,</p> <pre>BucketRegion=<bucket-region-name></pre>
Tracing Level	<p>Sets the amount of detail that appears in the log file.</p> <p>You can choose terse, normal, verbose initialization, or verbose data. Default is normal.</p>

Open Table targets in mappings

When you configure a mapping in advanced mode to use an Open Table target, configure the target properties.

Specify the name and description of the Open Table target. Configure the target and advanced target properties for the Open Table object in a Target transformation.

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

Property	Description
Connection	<p>Name of the target connection.</p> <p>You can select an existing connection, create a new connection, or define parameter values for the source connection property.</p> <p>If you want to overwrite the target connection properties at run time, select the Allow parameter to be overridden at run time option.</p>
Target Type	<p>Type of the Open Table target object.</p> <p>You can choose from the following source types:</p> <ul style="list-style-type: none"> - Single Object. Select to specify a single Open Table object. - Parameter. Select to specify a parameter name. You can configure the target object in a mapping task associated with a mapping that uses this Target transformation.

Property	Description
Parameter	<p>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.</p> <p>The Parameter property appears only if you select parameter as the target type.</p> <p>If you want to overwrite the target object at run time, select the Allow parameter to be overridden at run time option.</p> <p>When the task runs, the Secure Agent uses the parameters from the file that you specify in the advanced session properties.</p>
Object	<p>Name of the target object.</p> <p>You can select an existing object from the list or create a target object at run time. For information on creating the target object at run time, see the "Create a target table at runtime" on page 17 topic.</p> <p>Note: You cannot create a target object at runtime if you configure Hive Metastore catalog in the connection.</p>
Operation	<p>Type of the target operation.</p> <p>Select one of the following operations:</p> <ul style="list-style-type: none"> - Insert - Update - Upsert - Delete <p>Note: You cannot configure the Data Driven operation on an Open Table target.</p>
Update Columns	<p>Select the unique key column as the condition field for the update, upsert, or delete operations in an Open Table target.</p>

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

Property	Description
Iceberg Spark Properties	<p>The properties that you want to configure for the Iceberg tables.</p> <p>Enter the properties in the following format:</p> <pre><parameter name>=<parameter value></pre> <p>If you enter more than one property, enter each property in a new line.</p> <p>For example,</p> <pre>BucketRegion=<bucket-region-name></pre> <p>Note: When you configure update, upsert, or delete target operations and the source and target are in different regions, you need to set the bucket region property.</p>
Delta Spark Properties	<p>The properties that you want to configure for the Delta Lake tables.</p> <p>Enter the properties in the following format:</p> <pre><parameter name>=<parameter value></pre> <p>If you enter more than one property, enter each property in a new line.</p> <p>For example,</p> <pre>BucketRegion=<bucket-region-name></pre> <p>Note: When the source and target are in different regions, you need to set the bucket region property.</p>
UpdateMode	<p>Loads data to the target based on the mode you specify.</p> <p>This property applies when you select the Update or Upsert operation.</p> <p>Select one of the following modes:</p> <ul style="list-style-type: none"> - Update As Update. Updates records in the target table if the specified unique key column value matches with the incoming column value. - Update Else Insert. Updates records in the target table if the specified unique key column value matches with the incoming column value. If the unique key column value does not match, the mapping inserts a new row with the records.

Create a target table at runtime

You can use an existing target or create a target to hold the results of a mapping. If you choose to create the target, the agent creates the target if it does not exist already when you run the task.

To specify the target properties, perform the following tasks:

1. Select the Target transformation in the mapping.
2. To specify the target, click the **Target** tab.
3. Select the target connection.
4. For the target type, choose **Single Object** or **Parameter**.
5. Specify the target object or parameter.
6. To specify a target object, perform the following tasks:
 - a. Click **Select** and choose a target object. You can select an existing target object or create a new target object at runtime.
 - b. To create a target object at runtime, select **Create New at Runtime**.
 - c. In the **Object Name** field, enter the name of the target table that you want to create.
 - d. In the **Catalog Name with Database Name** field, enter the AWS Glue catalog name and database name in the following format: <CatalogName>/<DatabaseName>.

- e. In the **Table Location** field, enter the location of the target table data.
- f. Specify the table properties to optimize the table configuration settings.
- g. Click **OK**.

The following image shows a sample target object:

Open Table lookups in mappings

In a mapping in advanced mode, you can configure a Lookup transformation to represent an Open Table object.

You can retrieve data from an Open Table lookup object based on the specified lookup condition.

The following table describes the Open Table lookup 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 target connection properties at run time, select the Allow parameter to be overridden at run time option.
Source Type	Type of the Open Table lookup object. Select one of the following lookup object types: - Single Object . Select to specify a single Open Table object. - Parameter . Select to specify a parameter name. You can configure the lookup object in a mapping task associated with a mapping that uses this Lookup transformation.

Property	Description
Parameter	<p>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.</p> <p>The Parameter property appears only if you select parameter as the source type.</p> <p>If you want to overwrite the lookup object at run time, select the Allow parameter to be overridden at run time option.</p> <p>When the task runs, the Secure Agent uses the parameters from the file that you specify in the advanced session properties.</p>
Lookup Object	Name of the Open Table lookup object.
Multiple Matches	<p>The behavior when the lookup condition returns multiple matches.</p> <p>You can select one of the following options:</p> <ul style="list-style-type: none"> - Return any row - Return all rows - Report error <p>For more information about configuring lookups, see the <i>Transformations</i> guide in the Data Integration documentation.</p>

The following table describes the Open Table lookup advanced properties that you can configure in a Lookup transformation:

Property	Description
Iceberg Spark Properties	<p>The properties that you want to configure for the Iceberg tables.</p> <p>Enter the properties in the following format:</p> <pre><parameter name>=<parameter value></pre> <p>If you enter more than one property, enter each property in a new line.</p> <p>For example,</p> <pre>s3.access-points.<bucket name>=<S3 bucket ARN> BucketRegion=<bucket-region-name></pre> <p>Note: The S3 bucket ARN property is mandatory when the source and target are in different regions. For more information, see the Create the ARN for your AWS S3 bucket KB article.</p>
Delta Spark Properties	<p>The properties that you want to configure for the Delta Lake tables.</p> <p>Enter the properties in the following format:</p> <pre><parameter name>=<parameter value></pre> <p>If you enter more than one property, enter each property in a new line.</p> <p>For example,</p> <pre>BucketRegion=<bucket-region-name></pre>

Using time travel queries

You can use the time travel feature to query or roll back historical data of a table.

When you create or delete Apache Iceberg or Delta Lake table items, it automatically generates a snapshot and replaces the old data with the snapshot data. You can utilize these snapshots to perform time travel queries and roll back data as it existed at a specific point in time or at a specific snapshot.

When you configure a source transformation in a mapping, you can use the advanced filter option in your Open Table object to configure a time travel query. The query fetches the data based on a timestamp or snapshot ID you specify.

You can use one of the following queries to retrieve the data:

- **Query by Timestamp:** You can query an Iceberg or Delta Lake table as it existed at a particular timestamp. For example, if you want to time travel to July 10, 1986 at 04:20:09, use the following time travel query:

```
TimestampValue = 1986-07-10 04:20:09.0 UTC
```

- **Query by Snapshot ID:** You can query an Iceberg or Delta Lake table by specifying a snapshot ID. Each snapshot has a unique identifier.

For example, if you want to time travel to a snapshot with ID 33444444553321, use the following time travel query:

```
SnapshotVersion = 33444444553321
```

Dynamic schema handling

You can choose how Data Integration handles changes that you make to the data object schemas. To refresh the schema every time the mapping task runs, you can enable dynamic schema handling in the task.

When you use Apache Iceberg tables as the Open Table format, you can configure dynamic schema handling.

The schema change includes the newly added fields within the data object.

You can configure schema change handling on the **Runtime Options** page when you configure the task.

The following table describes the schema change handling options:

Option	Description
Asynchronous	Default. Data Integration refreshes the schema when you edit the mapping or mapping task, and when Informatica Intelligent Cloud Services is upgraded.
Dynamic	Data Integration refreshes the schema every time the task runs. You can choose from the following options to refresh the schema: <ul style="list-style-type: none">- Alter and Apply Changes. Data Integration adds new fields from the source schema to the target schema.- Don't apply DDL Changes. Data Integration does not apply the schema changes to the target.

For more information, see the "Schema change handling" topic in *Tasks* in the Data Integration help.

Rules and guidelines for dynamic schema handling

Consider the following rules and guidelines when you enable dynamic schema change handling:

- When you use Apache Iceberg Open Table format and create a target table at runtime in a mapping, ensure that the table name and database name contain only lowercase characters.

Rules and Guidelines for Open Table tasks

You need to follow certain rules and guidelines to configure tasks that uses Open Table Connector.

Consider the following rules and guidelines when you configure a task:

- A mapping fails at run time when you specify a simple filter with **Starts With**, **Ends With**, or **Contains** operators to retrieve data. To avoid this issue, configure an advanced filter with a `like` query to filter query results based on patterns instead of exact textual matches.
- When you configure an advanced data filter for a field of Timestamp data type, the filter ignores the time and considers only the date.
- When you use a simple filter on the date or timestamp data type column to read data, the task fails at run time with the following error:

```
An error occurred during mapping translation: Invalid expression string for filter condition
```

As a workaround, use the advance filter to filter date and timestamp data type columns from an Open Table source.

- You can't use uncached lookup in the Lookup transformation.
- When you run a mapping with an Open Table connection to read from or write data to Apache Iceberg tables available in AWS Glue Catalog, the mapping fails if the source and target are pointing to different AWS Glue catalogs available in different regions.
- When you run a mapping with an Upsert target operation and all the condition fields are not mapped in the field mapping, the mapping fails.
- When you run a mapping to read from a Snowflake table and write to an Apache Iceberg table managed by the Hive Metastore catalog, the mapping fails.
- When you configure a mapping with an Apache Iceberg table, the mapping fails in the following scenarios:
 - If the source table is managed by the AWS Glue Catalog and the target table is managed by the Hive Metastore catalog.
 - If the source table is managed by the Hive Metastore catalog and the target table is managed by the AWS Glue Catalog.
- When you configure a filter query in a mapping with an AND or OR operator to read data from an Apache Iceberg table managed by the Hive Metastore catalog, the query does not return any records.
- When you read data from an Apache Iceberg table managed by the Hive Metastore catalog in a mapping, the data preview at the source table and the preview job fail.

CHAPTER 4

Data type reference

Data Integration uses the following data types in Open Table mappings and mapping tasks:

- **Open Table native data types:** Open Table 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.

Open Table and transformation data types

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

Open Table data type	Transformation data type	Supported Open Table format	Description
Bigint	Bigint	Delta Lake	-9223372036854775808 to 9223372036854775807
Binary	Binary	Apache Iceberg, Delta Lake	Variable length binary data.
Boolean	String	Apache Iceberg, Delta Lake	True or false
Date	Date/Time	Apache Iceberg, Delta Lake	Calendar date without timezone and time. 0001-01-01 to 9999-12-31 Precision 29, scale 9
Decimal (Precision, Scale)	Decimal	Apache Iceberg, Delta Lake	The maximum value for precision is 38, and the maximum value for scale is 38.

Open Table data type	Transformation data type	Supported Open Table format	Description
Double	Double	Apache Iceberg, Delta Lake	4.94065645841246544e-324d to 1.79769313486231570e+308d, positive or negative.
Float	Double	Apache Iceberg, Delta Lake	1.40129846432481707e-45 to 3.40282346638528860e+38, positive or negative.
Int	Integer	Apache Iceberg, Delta Lake	2^{31} to $2^{31} - 1$.
Smallint	Integer	Delta Lake	-32768 to 32767
String	String	Apache Iceberg, Delta Lake	Variable length character data.
Timestamp	Date/Time	Apache Iceberg, Delta Lake	Timestamp without timezone. Jan 1, 0001 A.D. to Dec 31, 9999 A.D. (Precision to the microsecond)
Tinyint	Integer	Delta Lake	-128 to 127

Rules and guidelines for data types

Consider the following rules and guidelines when you configure Open Table data types that Data Integration supports and the corresponding transformation data types:

You cannot use Binary data type when you read or write data to Apache Iceberg table managed by the Hive Metastore catalog.

INDEX

C

Cloud Application Integration community
URL [4](#)
Cloud Developer community
URL [4](#)
connections
Open Table [9](#)

D

Data Integration community
URL [4](#)

I

Informatica Global Customer Support
contact information [5](#)
Informatica Intelligent Cloud Services
web site [4](#)

M

maintenance outages [5](#)

O

Open Table
connection properties [9](#)

S

status
Informatica Intelligent Cloud Services [5](#)
system status [5](#)

T

trust site
description [5](#)

U

upgrade notifications [5](#)

W

web site [4](#)