Informatica®

Informatica® Cloud Data Integration

Google Cloud Storage V2 Connector

Informatica Cloud Data Integration Google Cloud Storage V2 Connector
May 2024

© Copyright Informatica LLC 2018, 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: 2024-05-28

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 5
- 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 Google Cloud Storage V2 Connector 7

Google Cloud Storage V2 Connector assets 7

Administration of Google Cloud Storage V2 Connector 8

Introduction to Google Cloud Storage 8

Chapter 2: Google Cloud Storage V2 connections 11

Google Cloud Storage V2 connection properties 11

Proxy server settings 12

- Configure proxy settings for NTLM authentication 12

Chapter 3: Mappings for Google Cloud Storage 14

Google Cloud Storage V2 sources in mappings 14

Google Cloud Storage V2 targets in mappings 17

Google Cloud Storage file formatting options 19

- Fixed-width file formats 21
- Rules and guidelines for file formatting options 22

Directory source in Google Cloud Storage sources 22

Informatica encryption for Google Cloud Storage V2 sources and targets 23

- Importing encrypted source files 23

Data compression in Google Cloud Storage V2 sources and targets 24

Handling dynamic schemas 25

SQL ELT optimization 25

Mappings in advanced mode example 26

Directory-level partitioning for mappings in advanced mode 27

- Reading from partition columns 28
- Writing to partition columns 29
- Rules and guidelines for reading from and writing to a partition folder 30

Wildcard characters for mappings in advanced mode 30

Incrementally loading files for mappings in advanced mode 31

Rules and guidelines for mappings and mapping tasks. . . . 32
Rules and Guidelines for mappings in advanced mode. . . . 33
Troubleshooting a mapping task. . . . 34

Chapter 4: Migrating a mapping. . . . 36

Use the same object path for the migrated mapping. . . . 36
Use a different object path for the migrated mapping. . . . 36
Migration options. . . . 37
Rules and guidelines for migrating a mapping. . . . 38

Chapter 5: Upgrading to Google Cloud Storage V2 Connector. . . . 39

Connection switching example. . . . 40
Advanced properties retained after the switch. . . . 42

Appendix A: Data type reference. . . . 43

Flat Google Cloud Storage file data types and transformation data types. . . . 43
Avro Google Cloud Storage file data types and transformation data types. . . . 44
JSON Google Cloud Storage file data types and transformation data types. . . . 45
ORC Google Cloud Storage file data types and transformation data types. . . . 46
Parquet Google Cloud Storage file data types and transformation data types. . . . 47

Index. . . . 49

Preface

Use *Google Cloud Storage V2 Connector* to learn how to read from or write to Google Cloud Storage by using Cloud Data Integration. Learn to create a connection, develop and run mappings, mapping tasks, dynamic mapping tasks, and data transfer 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 Google Cloud Storage V2 Connector

You can use Google Cloud Storage V2 Connector to securely read data from or write data to Google Cloud Storage.

You can use Google Cloud Storage V2 objects as sources and targets in mappings and mapping tasks. You can switch mappings to advanced mode to include transformations and functions that enable advanced functionality. The advanced cluster can be hosted on the Google Cloud Platform.

Use Google Cloud Storage V2 Connector to read or write Avro, flat, and Parquet file formats for mappings. You can read or write Avro, flat, ORC, Parquet, and JSON for mappings in advanced mode.

You can read and write primitive data types for Avro, Parquet, JSON, and ORC files. You can read and write hierarchical data types only for Avro, Parquet, and JSON in mappings in advanced mode.

When you run a task or mapping, the Secure Agent uses the Google Cloud Storage API to perform the specified operation and reads data from or writes data to Google Cloud Storage files.

Google Cloud Storage V2 Connector assets

Create assets in Data Integration to integrate data using Google Cloud Storage V2 Connector.

You can insert data into a Google Cloud Storage target. You cannot perform update, upsert, or delete operations on a Google Cloud Storage target.

When you use Google Cloud Storage V2 Connector, you can include the following Data Integration assets:

- Data transfer task
- Dynamic mapping task
- Mapping
- Mapping task

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

Administration of Google Cloud Storage V2 Connector

Before you use Google Cloud Storage V2 Connector, you must complete the following prerequisite tasks:

1. Ensure that you have a Google service account to access Google Cloud Storage.
2. Ensure that you have the client_email, project_id, and private_key values for the service account. You will need to enter these details when you create a Google Cloud Storage connection in Data Integration.
3. Ensure that you have enabled the Google Cloud Storage JSON API for your service account.

   Google Cloud Storage V2 Connector uses the Google API to integrate with Google Cloud Storage.
4. Verify that you have read and write access to the Google Cloud Storage bucket that contains the source file and target file.
5. When you read data from or write data to a Google Cloud Storage file in a mapping, you must have the required permissions to run the mapping successfully.
6. To use the Informatica Encryption method, perform the following tasks:
   - Ensure that you add the following firewall rules for the VPC network on the Google Cloud Platform advanced cluster:
     - List of Secure Agent IP address ranges in the Source Filter section.
     - Google Cloud Platform cluster IP address ranges in the Source Filter section.
     - Port number `0-65535` and `udp:443` under Protocols and Ports section.
     - Ensure that the VPC network tag in the virtual machine that hosts the Secure Agent has the same value as the source tag and target tag of the firewall rules.
   - Ensure that the Informatica crypto library license is enabled.

Introduction to Google Cloud Storage

Google Cloud Storage is a web service that allows global storage and retrieval of large volumes of data at any time.

You can use Google Cloud Storage to stream multimedia, store custom data analytics pipelines, or distribute large data objects to users through direct download.

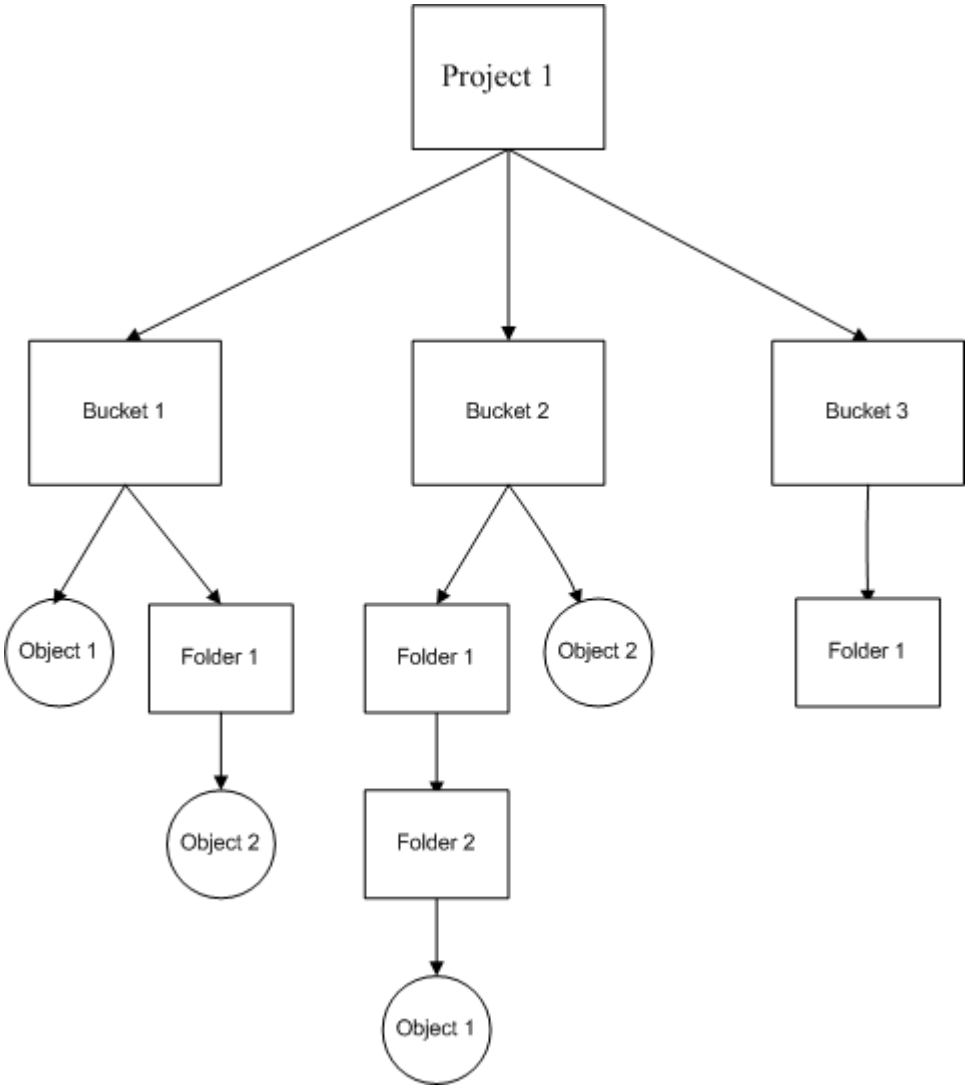
You can write data to Google Cloud Storage for data backup. In the event of a database failure, you can read the data from Google Cloud Storage and restore it back to the database.

Google Cloud Storage offers different storage classes based on factors such as data availability, latency, and price.

Google Cloud Storage comprises the following components:

- Projects
- Buckets
- Objects

The following image shows how data can be organized in Google Cloud Storage:



You can use the following components to read data from and write data to Google Cloud Storage:

Projects

In Google Cloud Storage, all resources are stored within a project. Project is a top-level container that stores billing details and user details. You can create multiple projects. A project has a unique project name, project ID, and project number.

Buckets

Each bucket acts like a container that stores data. You can use buckets to organize and access data. You can create more than one bucket but you cannot nest buckets.

You can create multiple folders within a bucket and you can also nest folders.

You can define access control lists to manage objects and buckets. An access control list consists of permission and scope entries. Permission defines the access to perform a read or write operation. Scope defines a user or a group who can perform the operation.

Objects

Objects comprise the data that you upload to Google Cloud Storage. You can create objects in a bucket. Objects consist of object data and object metadata components. The object data is a file that you store

in Google Cloud Storage. The object metadata is a collection of name-value pairs that describe object qualities.

CHAPTER 2

Google Cloud Storage V2 connections

Create a Google Cloud Storage V2 connection to securely read data from or write data to Google Cloud Storage files.

You can use a Google Cloud Storage V2 connection to specify sources and targets in mappings and mapping tasks.

Google Cloud Storage V2 connection properties

When you create a Google Cloud Storage V2 connection, configure the connection properties.

The following table describes the Google Cloud Storage 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	The Google Cloud Storage V2 connection type.
Runtime Environment	Name of the runtime environment where you want to run the tasks. Select a Secure Agent, Hosted Agent, or serverless runtime environment.
Service Account ID	The client_email value in the JSON file that you download after you create a service account.
Service Account Key	The private_key value in the JSON file that you download after you create a service account.
Project ID	The project_id value in the JSON file that you download after you create a service account. If you created multiple projects with the same service account, enter the ID of the project that contains the bucket that you want to connect to.

Property	Description
Is Encrypted File[1]	Specifies whether a file is encrypted. Select this option when you import an encrypted file from Google Cloud Storage. Default is unselected.
Bucket Name	The Google Cloud Storage bucket name that you want to connect to. When you select a source object or target object in a mapping, the Package Explorer lists files and folder available in the specified Google Cloud Storage bucket. If you do not specify a bucket name, you can select a bucket from the Package Explorer to select a source or target object.
Optimize Object Metadata Import	Optimizes the import of metadata for the selected object without parsing other objects, folders, or sub-folders available in the bucket. Directly importing metadata for the selected object can improve performance by reducing the overhead and time taken to parse each object available in the bucket. Default is not selected.

[1] Applies only to mappings in advanced mode.

Proxy server settings

If your organization uses an outgoing proxy server to connect to the Internet, the Secure Agent connects to Informatica Intelligent Cloud Services through the proxy server.

You can configure the Secure Agent to use the proxy server on Windows and Linux. You can use the unauthenticated or authenticated proxy server. The proxy settings applies to connections used in mappings and in mappings in advanced mode.

To configure the proxy settings for the Secure Agent, perform the following tasks:

- Configure the Secure Agent through the Secure Agent Manager on Windows or shell command on Linux.

  For instructions, see "Configure the proxy settings on Windows" or "Configure the proxy settings on Linux" in *Getting Started* in the Data Integration help.
- Configure the JVM options for the DTM in the Secure Agent properties. For instructions, see the Proxy server settings Knowledge Base article.

To configure the proxy settings for the serverless runtime environment, see "Using a proxy server" in *Runtime Environments* in the Administrator help.

Configure proxy settings for NTLM authentication

You can use a proxy server that uses NTLM authentication to connect to Google Cloud Storage.

To configure the proxy settings for NTLM authentication, perform the following steps:

1. In Administrator, select **Runtime Environments**.
2. Select the Secure Agent for which you want to configure from the list of available Secure Agents.
3. In the upper-right corner, click **Edit**.
4. In the **System Configuration Details** section, select the **Type** as **DTM** for the Data Integration Server.

5. Edit the **JVMOption1** and add the following value:

   ```
   -Dhttp.auth.ntlm.domain=<domain name>
   ```

6. Select the **Type** as **Platform** for the Data Integration Server.
7. Edit the **INFA_DEBUG** property and add the following value:

   ```
   -Dhttp.auth.ntlm.domain=<domain name>
   ```

8. Click **Save**.
9. Restart the Secure Agent.

CHAPTER 3

Mappings for Google Cloud Storage

When you configure a mapping, you describe the flow of data from the source to the target.

A mapping defines reusable data flow logic that you can use in mapping tasks.

When you create a mapping, you define the Source and Target transformations to represent a Google Cloud Storage V2 object. Use the Mapping Designer in Data Integration to add the Source or target transformations in the mapping canvas and configure the Google Cloud Storage V2 source and target properties.

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

You can use Monitor to monitor the jobs.

Google Cloud Storage V2 sources in mappings

To read data from a Google Cloud Storage file, configure a Google Cloud Storage object as the Source transformation in a mapping.

Specify the name and description of the Google Cloud Storage source. Configure the source and advanced properties for the source object.

The following table describes the properties that you can configure for a Google Cloud Storage source:

Property	Description
Connection	Name of the Google Cloud Storage V2 connection. Select a source connection, or click **New Parameter** to define a new parameter for the source connection. If you want to overwrite the parameter at runtime, select the **Allow parameter to be overridden at run time** option when you create a parameter.
Source Type	Type of the Google Cloud Storage source object available. You can read data to a single Google Cloud Storage source object.
Object	Name of the source object for the mapping. **Note:** Ensure that the column names in the source object do not begin with special characters to successfully read data from the source.

Property	Description
Parameter	A parameter file where you define values that you want to update without having to edit the task. Select a 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. When you define the value for the source object parameter, ensure that you provide a valid file name. If you want to overwrite the parameter at runtime, select the **Allow parameter to be overridden at run time** option when you create a parameter. When the task runs, the agent uses the parameters from the file that you specify in the task advanced session properties.
Format	Specifies the file format that the Google Cloud Storage V2 Connector uses to read data from Google Cloud Storage. You can select the following file format types: - Flat - Avro - Parquet - JSON - Orc[1] - None[2] **Note:** If you select **None** is as the format type, Google Cloud Storage V2 Connector reads data from Google Cloud Storage files in binary format. Open the **Formatting Options** dialog box to configure the formatting options for the file. For more information about format options, see " Google Cloud Storage file formatting options" on page 19.
Filter	This attribute is not applicable for Google Cloud Storage V2 Connector.
Sort	This attribute is not applicable for Google Cloud Storage V2 Connector.

[1] Applies only to mappings in advanced mode.

[2] Doesn't apply to mappings in advanced mode.

The following table describes the advanced properties that you can configure for a Google Cloud Storage source:

Property	Description
Google Cloud Storage Path	Optional. Overrides the bucket name or folder path of the Google Cloud Storage file that you specified in the connection. Use the following format: `gs://<bucket name>` or `gs://<bucket name>/<folder name>` For example, `gs://dlake_bkt/customer/sales` **Note:** You cannot specify wildcard characters in the Google Cloud Storage path.
Source File Name	Optional. Overrides the Google Cloud Storage source file name that you specified in the Source transformation. **Note:** Does not apply when you configure **Is Directory** option to read multiple files from a directory.
Is Directory	Select this property to read all the files available in the folder specified in the **Google Cloud Storage Path** property. **Note:** If you do not provide the **Google Cloud Storage Path** value during run time, the Secure Agent considers the value of the **Google Cloud Storage Path** that you specify when you select a Google Cloud Storage source file in the Source transformation.

Property	Description
Incremental File Load[1]	Indicates whether you want to incrementally load files when you use a directory as the source for mappings in advanced mode. When you incrementally load files, the mapping task reads and processes only files in the directory that have changed since the mapping task last ran. For more information, see "Incrementally loading files for mappings in advanced mode" on page 31.
Allow Wildcard Characters[1]	Indicates whether you want to use wildcard characters for the directory sources. If you select this option, you can use the question mark (?) and asterisk (*) wildcard characters in the folder path or file name. For details, see "Wildcard characters for mappings in advanced mode" on page 30.
Encryption Type	Method to decrypt data. You can select one of the following encryption types: - Informatica Encryption - None Default is **None**.
Compression Format	Method to read compressed data from Google Cloud Storage. You can read the compressed data in the following formats: - None - Gzip Select **None** to read from the compressed Avro or Parquet file. Select **Gzip** to read from the compressed Flat file. Default is **None**.

[1]Applies only to mappings in advanced mode.

You can set the tracing level in the advanced properties session to determine the amount of details that logs contain.

The following table describes the tracing levels that you can configure:

Tracing Level	Description
Terse	The Secure Agent logs initialization information, error messages, and notification of rejected data.
Normal	The Secure Agent logs initialization and status information, errors encountered, and skipped rows due to transformation row errors. Summarizes session results, but not at the level of individual rows.
Verbose Initialization	In addition to normal tracing, the Secure Agent logs additional initialization details, names of index and data files used, and detailed transformation statistics.
Verbose Data	In addition to verbose initialization tracing, the Secure Agent logs each row that passes into the mapping. Also notes where the Secure Agent truncates string data to fit the precision of a column and provides detailed transformation statistics. When you configure the tracing level to verbose data, the Secure Agent writes row data for all rows in a block when it processes a transformation.

Google Cloud Storage V2 targets in mappings

To write data to a Google Cloud Storage V2 target, configure a Google Cloud Storage object as the Target transformation in a mapping.

Specify the name and description of the Google Cloud Storage V2 target. You can configure the target and advanced properties for the target object.

The following table describes the properties that you can configure for a Google Cloud Storage V2 target:

Property	Description
Connection	Name of the Google Cloud Storage V2 target connection. Select a source connection, or click **New Parameter** to define a new parameter for the source connection. If you want to overwrite the parameter at runtime, select the **Allow parameter to be overridden at run time** option when you create a parameter. When the task runs, the agent uses the parameters from the file that you specify in the task advanced session properties.
Target Type	Type of the Google Cloud Storage target objects available. You can write data to a single target object or parameterize Google Cloud Storage target object. You cannot write data to multiple objects.
Object	Name of the Google Cloud Storage file. **Note:** Ensure that the column names in the target object do not begin with special characters to successfully write data to the target.
Parameter	Select an existing parameter for the target object or click **New Parameter** to define a new parameter for the target object. The **Parameter** property appears only if you select Parameter as the target type. If you want to overwrite the parameter at runtime, select the **Allow parameter to be overridden at run time** option when you create a parameter. When the task runs, the agent uses the parameters from the file that you specify in the task advanced session properties.
Create New at Runtime	Creates a target. Enter a name and path for the target object and select the source fields that you want to use. By default, all source fields are used. The target name can contain alphanumeric characters. You can use the following special characters in the file name: `., _, @, $, %` If you specify the path, the Secure Agent creates the target object in the path you specify in this property. The Secure Agent creates the target object in the following path: `<bucket_name>/<path_name>/<target_object_name>` You cannot parameterize the target at runtime. When you create a mapping in advanced mode, you can use the **CreateTarget** option only for Avro, JSON, and Parquet format.

Property	Description
Format	Specifies the file format that the Google Cloud Storage V2 Connector uses to write data to Google Cloud Storage. You can select the following file format types: - Flat - Avro - Parquet - JSON - Orc[1] - None[2] **Note:** If you select **None** is as the format type, Google Cloud Storage V2 Connector writes data to Google Cloud Storage files in binary format. Open the **Formatting Options** dialog box to configure the formatting options for the file. For more information about format options, see " Google Cloud Storage file formatting options" on page 19.
Operation	Select **Insert** as the target operation. You cannot perform update, upsert, or delete operations on a Google Cloud Storage object.
[1] Applies only to mappings in advanced mode. [2] Doesn't apply to mappings in advanced mode.	

The following table describes the advanced properties that you can configure for a Google Cloud Storage V2 target:

Property	Description
Google Cloud Storage Path	Overrides the Google Cloud Storage path where the target file exists. Use the following format: `gs://<bucket name>` or `gs://<bucket name>/<folder name>` For example, `gs://dlake_bkt/customer/sales` **Note:** You cannot specify wildcard characters in the Google Cloud Storage path.
Target File Name	Optional. Overrides the Google Cloud Storage target object name specified in the Target transformation.
Encryption Type	Method to encrypt data. You can select one of the following encryption types: - Informatica Encryption - None Default is **None**.
Compression Format[1]	Compresses data when you write to Google Cloud Storage. You can compress the data in the following formats: - None - Gzip - Deflate - Snappy Default is **None**. In advanced mode, you can compress data only when you run a mapping that writes data to a Google Cloud Storage file in Parquet, JSON, or Delimited file formats.

Property	Description
Forward Rejected Rows	This property is not applicable for Google Cloud Storage V2 Connector.
[1]Applies only to mappings in advanced mode.	

Google Cloud Storage file formatting options

When you select the format of a Google Cloud Storage file for a source or a target, you can configure the formatting options.

The following table describes the Google Cloud Storage V2 file formatting options that you can configure:

Option	Description
Schema Source	The schema of the source or target file. You can select one of the following options to specify a schema: - Read from data file. Google Cloud Storage V2 Connector imports the schema from the file in Google Cloud Storage. Applicable only to mappings in advanced mode. - Import from schema file. Imports schema from a schema definition file in your local machine.
Schema File	The schema definition file on the agent machine from where you want to upload the schema. This property appears only if you select `Import from schema file` in the **Schema Source** property. You can use the following file formats: - Avro - Flat - JSON - Parquet You can't upload a schema file when you select the **Create Target** option to write data to Google Cloud Storage.

The following table describes the formatting options for delimited and fixed width flat file types:

Option	Description
Flat File Type	The type of flat file. Select one of the following options: - Delimited. Reads a flat file that contains column delimiters. - Fixed Width. Reads a flat file with fields that have a fixed length. You must select the file format in the **Fixed Width File Format** option. If you do not have a fixed-width file format, click **New** > **Components** > **Fixed Width File Format** to create one.
Delimiter	Character used to separate columns of data. You can configure parameters such as comma, tab, colon, semicolon, or others. Default is comma (,). If you specify a multibyte character as a delimiter in the source object, the mapping fails. **Note:** To set a tab as a delimiter, you must type the tab character in any text editor. Then, copy and paste the tab character in the **Delimiter** field.

Option	Description
Escape Character	Character immediately preceding a column delimiter character embedded in an unquoted string, or immediately preceding the quote character in a quoted string. Default is backslash (\).
Qualifier	Quote character that defines the boundaries of data. You can set the qualifier as a single quote or double quote. Default is double quote (").
Qualifier Mode	Specify the qualifier behavior for the target object. You can select one of the following options: - MINIMAL. Applies qualifier to data enclosed within a delimiter value or a special character. - ALL. Applies qualifier to all data. - NON_NUMERIC. Not applicable. - ALL_NON_NULL. Not applicable. Default is MINIMAL.
Disable escape char when a qualifier is set	Check to disable the escape character when a qualifier value is already set.
Code Page[1]	The code page that the Secure Agent must use to read or write data. You can select from the following code pages: - UTF-8. Select for Unicode and non-Unicode data. - MS Windows Latin 1. Select for ISO 8859-1 Western European data. - Shift-JIS. Select for double-byte character data. - ISO 8859-15 Latin 9 (Western European). - ISO 8859-2 Eastern European. - ISO 8859-3 Southeast European. - ISO 8859-5 Cyrillic. - ISO 8859-9 Latin 5 (Turkish). - IBM EBCDIC International Latin-1. Default is UTF-8.
Header Line Number	Specify the line number that you want to use as the header when you read data from the Google Cloud Storage. You can also read a file that doesn't have a header. To read data from a file with no header, specify the value of the **Header Line Number** field as 0. To read data from a file with a header, set the value of the **Header Line Number** field to a value that is greater than or equal to one. Default is 1. Ensure that the value of the **Header Line Number** field is lesser than or equal to the value of the **First Data Row** field. When you create a mapping in advanced mode, set the value of the header line number to 0, 1, or empty to run the mapping successfully. This property is applicable during runtime and data preview to read a file.
First Data Row[1]	Specify the line number from where you want to read data. You must enter a value that is greater or equal to one. To read data from the header, the value of the **Header Line Number** and the **First Data Row** fields must be the same. Default is 2. This property is applicable when you preview data in a read and write operation. Also applies when you run the mapping to read the data.

Option	Description
Target Header	Select whether you want to write data to a flat file with or without a header. Default is With Header. This property is not applicable when you read data from a Google Cloud Storage source.
Distribution Column[1]	Specify the name of the column that is used to create multiple target files during runtime. This property is not applicable when you read data from a Google Cloud Storage source.
Maximum Rows To Preview	Not applicable.
Row Delimiter	Not applicable.
[1] Applies only to mappings.	

The following table describes the formatting options for JSON files:

Property	Description
Number of Rows to Sample[1]	Specify the number of rows to read to find the best match to populate the metadata. Default is 1.
Memory available to process data (in MB)[1]	The memory that the parser uses to read the JSON sample schema and process it. The default value is 2. Set the value to the file size that you want to read. If the file size is more than 2 MB, you might encounter an error.
Read multiple-line JSON files	Not applicable.
[1]Applies only to mappings in advanced mode.	

Fixed-width file formats

You can use a fixed-width flat file as a source or target in mappings and mapping tasks.

When you configure a Source or Target transformation and select the fixed-width flat file type, you must select the most appropriate fixed-width file format to use based on the data in the fixed-width flat file.

See the following exceptions before you use a fixed-width flat file:

- You cannot use a fixed-width flat file as a source or target for mappings in advanced mode and data transfer tasks.
- When you create a flat file target at runtime and select a fixed width file format, the Secure Agent ignores the fixed-width column boundaries that you specified for the fixed-width flat file format and applies the additional fixed width attributes for the new target object.
- When you use a fixed-width flat file as a source or target, you cannot edit the metadata for the fields.
- When you write a column of Numeric data type from fixed-width flat file source to an empty fixed-width flat file target that use the same fixed-width file format, the Secure Agent appends a null character to the value in Numeric column in the target.

- When you use a Secure Agent installed on a Linux machine and create a fixed-width file format, ensure that the sample file uses the `\n` character as the new line symbol, and that source files use the same symbol.
- When you use a Secure Agent installed on a Windows machine and create a fixed-width file format, ensure that the sample file uses the `\r\n` character as the new line symbol, and that source files use the same symbol.
- When you create a fixed-width file format, ensure that the sample flat file only uses UTF-8 character set encoding.

Rules and guidelines for file formatting options

You must set the appropriate formatting options when you select the Flat, Avro, JSON, ORC, or Parquet format types. Use the following guidelines when you configure file formatting options:

- You must use the same qualifier and delimiter in all the fields and columns of a schema file.
- If you select the delimited format type and select Import from schema file as the value of the Schema Source formatting option, you can only upload a schema file in the JSON format.
- When you use a flat file, use the following format for the schema file:

```
{"Columns":[{"Name":"empid","Type":"string","Precision":"255","Scale":"0"},
{"Name":"empname","Type":"string","Precision":"255","Scale":"0"},
{"Name":"dept_id","Type":"string","Precision":"255","Scale":"0"},
{"Name":"salary","Type":"number","Precision":"28","Scale":"9"},
{"Name":"no_of_days_in_current_position","Type":"bigint","Precision":"19","Scale":"0"}]}
```

- If you select the Avro, JSON, or Parquet format type and select **Read from data file** as the value of the **Schema Source** formatting option, you cannot view the schema of the file.
- When you read data from or write data to a file of the Avro or Parquet format, you must increase the Java heap size based on the amount of data you want to read or write. Increase the Java heap size in the JVM options for type DTM in the **System Configuration Details** section of the Secure Agent. Otherwise, the mapping task fails.
- When you select a file of JSON format that contains a large amount of data, you cannot preview data.
- When you read data from a flat file that contains a newline character, you must enclose the data within double quotation marks. Otherwise, the Secure Agent displays incorrect number of success rows in the session log.
- When you run a mapping that uses the **None** file format and set the **Is Directory** advanced property, the mapping reads only one file from the directory.

Directory source in Google Cloud Storage sources

You can select the **Is Directory** option under the advanced properties for a Google Cloud Storage source object to read all the files in a Google Cloud Storage folder and sub-folder.

Use the following rules and guidelines to configure the **Is Directory** option:

- You cannot read files available in a sub-folder within a sub-folder.
- All the source files in the directory must contain the same schema.
- When you read files in a Google Cloud Storage folder in Delimited format, all the files must have data in the same format. For example, delimiters, header fields, and escape characters must be same.

- When you run a mapping to read data from the files in a Google Cloud Storage folder and sub-folder using the Is Directory source advanced property, ensure that the folder and the sub-folder does not contain a file with the same schema. Otherwise, the mapping fails.
- The source files in the directory must have the same format and metadata. The file format, delimiters, header fields, escape characters, and compression format must be same for all the files. Otherwise, the mapping results are unpredictable.

Informatica encryption for Google Cloud Storage V2 sources and targets

You can download a flat source file that is encrypted using the Informatica crypto libraries in the local machine or staging location and decrypt the source files. You can encrypt the data when you write data to a flat file target.

Informatica encryption is applicable only when you run mappings on the Secure Agent machine installed on the Google Cloud Platform virtual machine.

When you configure a mapping, you can enable Informatica encryption for delimited flat files, fixed-width files, and binary files.

When you configure a mapping in advanced mode, you can enable Informatica encryption only for delimited flat files.

To read a source file that is encrypted or to encrypt the data when you write to a target file using the Informatica crypto libraries, perform the following tasks:

1. Ensure that the organization administrator has permission to the Informatica crypto libraries license when you create a Google Cloud Storage V2 connection.
2. Select **Informatica Encryption** as the encryption type in the advanced source or target properties.
3. Ensure that you encrypt or decrypt the file within the same organization ID.

Importing encrypted source files

When you read an Informatica encrypted source file and select the **Informatica Encryption** as the encryption type, Secure Agent fails to import the encrypted file.

To import the data successfully, perform the following tasks:

1. Select the **Is Encrypted File** option in the Google Cloud Storage V2 connection.
2. Import an encrypted file from Google Cloud Storage and select **Delimited** as the **Format**.
3. Select **Import from Schema File** as the **Schema Source** in the formatting options.
4. Upload a schema file in the JSON format in the **Schema File** property to override the schema of the encrypted file.
5. Select **Informatica Encryption** as the encryption type in the advanced source properties.

Alternatively, you can select a dummy source file that contains the same metadata as in the Informatica encrypted source file from where you want to read data. To override the file name of the dummy source file, enter the file name of the Informatica encrypted source file in the **Source File Name** advanced source property. Then, select **Informatica Encryption** as the encryption type in the advanced source properties.

Data compression in Google Cloud Storage V2 sources and targets

You can decompress the data when you read data from a Google Cloud Storage V2 source and compress the data when you write data to a Google Cloud Storage V2 target.

Configure the compression format in the **Compression Format** option under the advanced source and target properties.

The following table lists the supported compression formats in the source for different file formats:

Compression format	Avro File	Flat File	JSON File	Parquet File
Gzip	No	Yes	No	Yes
None	Yes	Yes	Yes	Yes

Note: Select the None compression format if you want to use Deflate or Snappy compression format for Avro and Parquet file formats.

The following table lists the supported compression formats in the target for different file formats:

Compression format	Avro File	Flat File	JSON File	Parquet File
Deflate	Yes	No	No	No
Gzip	No	Yes	No	Yes
None	Yes	Yes	Yes	Yes
Snappy	Yes	No	No	Yes

To read a compressed file from Google Cloud Storage V2, the compressed file must have specific extensions. If the extensions used to read the compressed file are not valid, the Secure Agent does not process the file. The following table describes the extensions that are appended based on the compression format that you use:

Compression format	File Name Extension
Deflate	`.deflate`
Gzip	`.GZ`
Snappy	`.snappy`

Use the following guidelines when you configure data compression:

- Data compression is supported at the file level. You cannot use data compression for a directory.
- When **Is Directory** property is selected at source, the files within the directory are read sequentially.

- When you download a compressed Gzip file for the Google Cloud Platform console, uncompressed file is downloaded by default. To download the compressed file, you need to remove the content encoding metadata of the object manually. Select **Edit object metadata** of the object and remove `Gzip` from the **Content-Encoding** field.
- When you configure Gzip compression format in the target and the mapping fails with a Java heap space error, update the staging optimization memory in the JVMOptions property to `-Xmx2048m` and `-Xms512m`. Google Cloud Storage requires a buffer size of 15 MB to upload the compressed files.

Handling dynamic schemas

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

Configure schema change handling in the Advanced Options section on the **Runtime Options** tab when you configure the task. You can configure asynchronous or dynamic schema change handling.

When you configure dynamic schema change handling, you can choose from the following options to refresh the schema:

Keep Existing File Format

Data Integration fetches the most recent target schema at runtime and does not apply upstream schema changes to the target file.

Drop Current and Recreate

Data Integration updates the target schema to match the incoming schema on every task run.

Restrictions

Consider the following restrictions when you configure dynamic schema handling:

- When you select the **Keep Existing File Format** option while configuring dynamic schema handling in a mapping that creates a new target at runtime and the target already exists, the changes in the source are propagated to the target.
- You cannot use the **Drop Current and Recreate** option in mappings in advanced mode.
- When you select the **Keep Existing File Format** option while configuring dynamic schema handling in a mapping that creates a new target at runtime and the target already exists, the changes in the source are propagated to the target when the object name contains special characters.

SQL ELT optimization

You can enable full SQL ELT optimization when you want to load data from Google Cloud Storage sources to your data warehouse in Google BigQuery. While loading the data to Google BigQuery, you can transform the data as per your data warehouse model and requirements. When you enable full SQL ELT optimization on a mapping task, the mapping logic is pushed to the GCP environment to leverage GCP commands. For more information, see the help for Google BigQuery V2 Connector.

If your use case involves loading data to any other supported cloud data warehouse, see the connector help for the applicable cloud data warehouse.

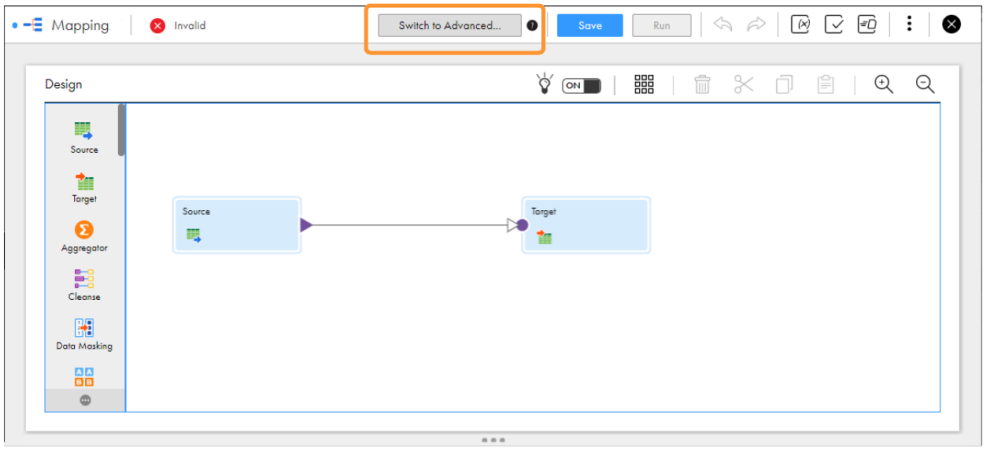
Mappings in advanced mode example

You work for one of the largest pharmaceutical company that maintains millions of records in their pharmacy database. The company has more than 10,000 employees working at 500 locations across the globe. The company has a very large IT infrastructure and about 20 TB of information gets downloaded on daily basis from the Internet.

To avoid performance, scalability, and high cost challenges, the company plans to port its entire data from its operational data stores to Google Cloud Storage within a short span of time. Create a mapping in advanced mode to achieve faster performance when you read data from the operational data stores and write data to the Google Cloud Storage target.

1. In Data Integration, click **New** > **Mappings** > **Mapping**.
2. In the Mapping Designer, click **Switch to Advanced**.

   The following image shows the **Switch to Advanced** button in the Mapping Designer:



3. In the **Switch to Advanced** dialog box, click **Switch to Advanced**.

   The Mapping Designer updates the mapping canvas to display the transformations and functions that are available in advanced mode.
4. Enter a name, location, and description for the mapping.
5. On the Source transformation, specify a name and description in the general properties.
6. On the **Source** tab, perform the following steps to provide the source details to read data from the source:
   a. In the **Connection** field, select the required source connection.
   b. In the **Source Type** field, select the type of the source.
   c. In the **Object** field, select the required object.
   d. In the **Advanced Properties** section, provide the appropriate values.
7. In the Target transformation, specify a name and description in the general properties.
8. On the **Target** tab, perform the following steps to provide the target details to write data to the Google Cloud Storage target:
   a. In the **Connection** field, select the Google Cloud Storage V2 target connection.
   b. In the **Target Type** field, select the type of the target.
   c. In the **Object** field, select the required object.

d. In the **Operation** field, select the required operation.

e. In the **Advanced Properties** section, provide appropriate values for the advanced target properties.

9. On the **Fields** tab, map the source fields to the target fields.
10. Click **Save** > **Run** to validate the mapping.

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

Directory-level partitioning for mappings in advanced mode

When you create a mapping in advanced mode, you can read from and write to partition columns.

You can organize tables or data sets into partitions to group the same type of data based on a column or partition key. You can select one or more partition columns in a table or data set.

To read from partition columns, select a partition directory and identify the partition columns. To write to partition columns, you can add partition columns from the list of fields and change the partition order, if required.

You can read data from or write data to partition columns for the following file formats:

- Avro
- Parquet
- ORC
- JSON

When you run a mapping on a Google Cloud Platform cluster to read data from or write data to partition folders, you must configure the following INFA_DEBUG property in the **System Configuration Details** section for the Secure Agent:

Property	Value
Service	Data Integration Server
Type	Platform
Name	INFA_DEBUG
Value	GoogleStorage_RandomizeLocalFileName=true

Importing partition folders

Consider the following rules and guidelines when you import partition folders:

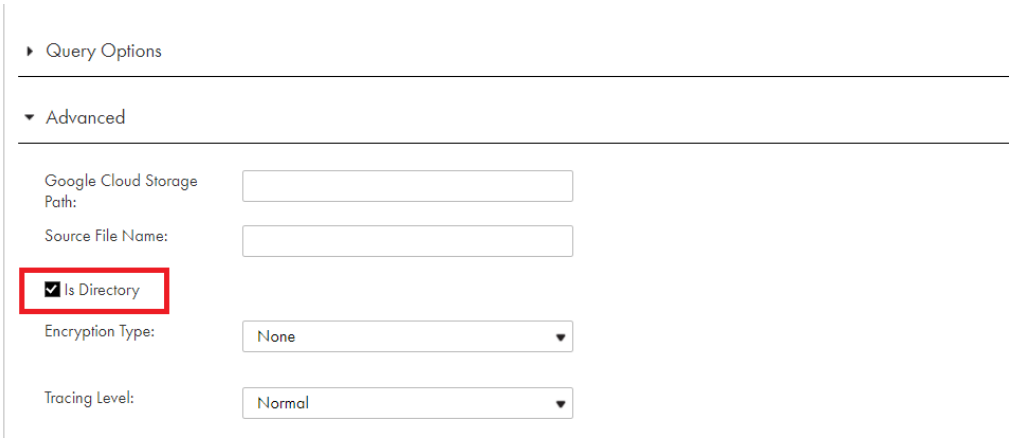
- When you read or write data to a partition folder, you can import a directory that contains both partition folder and files. If the directory contains only files but no partition folders, a validation error occurs.
- To import a partition directory that contains only partition folders, ensure that the partition directory contains files and data. Otherwise, a validation error is encountered.
- You must import a directory that contains only partition folders and select the **Is Directory** option in the advanced source property.

- If you import a partition directory that has a partition folder but no files in the partition folder, a validation error is encountered.
- When you import a directory that has a partition folder, the data type for the partition column is imported as a String.
- When you import a Google Cloud Storage object that has partition columns, the partition fields are listed at the end of the list.

Reading from partition columns

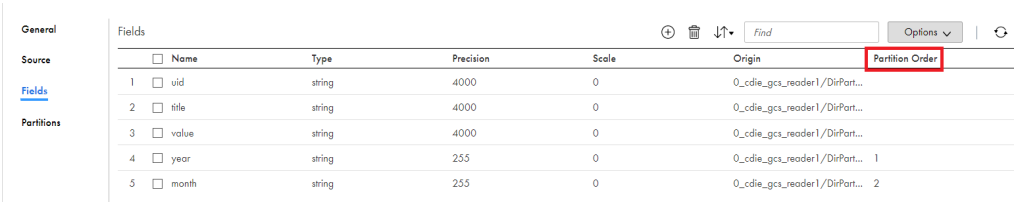
You can select the directory source type and view the list of partition columns. You can also view the order in which the fields were selected for partitioning.

1. Select a directory from the list of source objects.
2. Select the **Is Directory** option in the Advanced Source Properties.



3. Select the **Fields** tab to view the number of partitions.
   The **partitionOrder** column appears for the list of partitioned fields.

   The following image shows an example of the **partitionOrder** column:



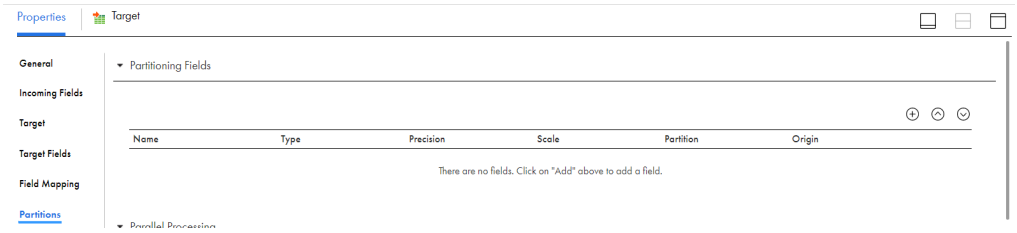
   The **partitionOrder** column specifies whether a column is partitioned.

   In the above example, two partition columns are available. The partition order values 1 and 2 signify the order in which the `year` and `month` fields were selected for partitioning.

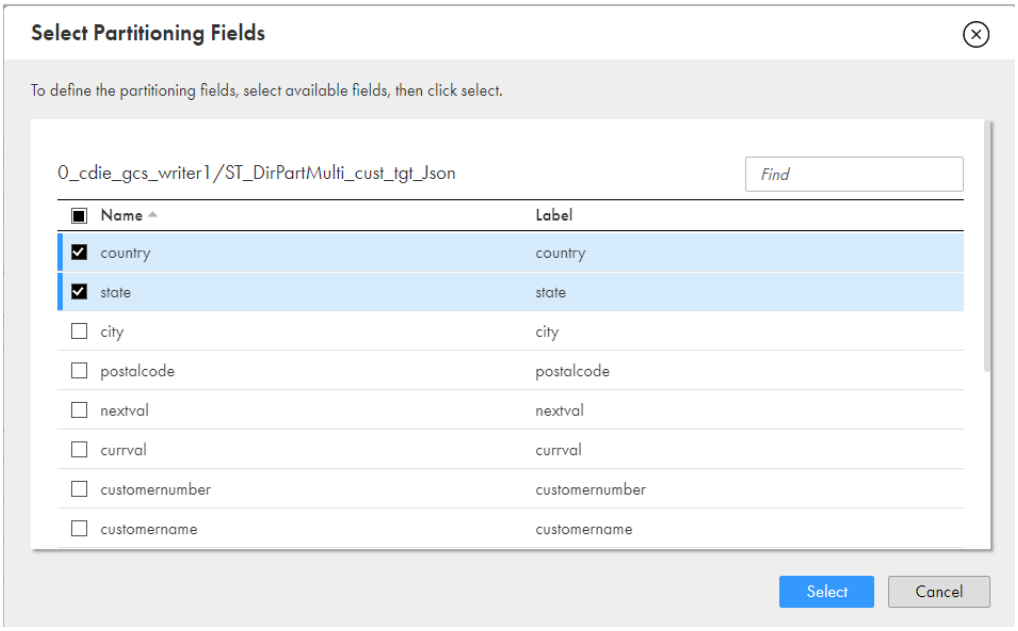
Writing to partition columns

You can add partition columns for a target and change the partition order of the columns.

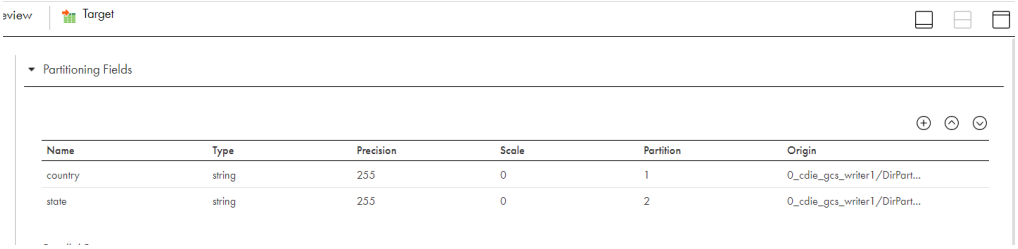
1. Click the ⊕ icon in the **Partitions** tab to add the partition columns for a target.
   The following image shows the **Partitions** tab where you can add the partition columns:



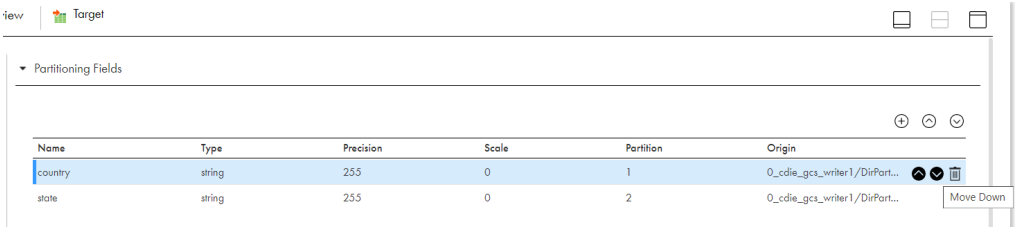
2. On the **Partitions** tab, select the partitioning fields from the list of available fields:



3. Click **Select**.
   The Partitions tab shows the partition columns that you selected:



Note: You can change the partition order using the up and down arrows as shown in the following image:



Rules and guidelines for reading from and writing to a partition folder

Consider the following rules and guidelines when you read from and write to a partition folder:

- You can read or write data to partition folders with the Avro, Parquet, JSON, and ORC files.
- You cannot write data to partition folders with Avro, JSON, ORC, and Parquet files if the source contains hierarchical data.
- You can configure a Filter transformation on a partition column for a Google Cloud Storage source.
- When you pass a timestamp value in a partition column, the value gets encoded. For example, `03:26:01` gets encoded as `03%3A26%3A01`.
- When you pass a special character in a partition column, the value gets encoded. For example, `#"` is encoded as `%23%22`
- When the partition column contains special characters, the special characters gets replaced with underscore.
- You cannot use the **Edit Metadata** option with partition columns.
- You cannot use the **View Schema** option for a partition directory at source and target side.
- You cannot use the **Import from Schema File** option for partition directory at source because the schema file does not have information for partition columns.
- You cannot use the **Data Preview** option with partition columns.
- You cannot select the partition columns in a mapping task if the target object is parameterized.
- When you create a target, you can add partition fields and arrange the partition columns in an order. You cannot add partition fields and arrange the partition columns in an order for an existing target.
- When you create a target, the **Label** column in the **Partitions** tab denotes the partition column name.
- If a partition column name contains more than 74 characters, the partition column name is truncated to only 74 characters and the Secure Agent writes NULL values or `_HIVE_DEFAULT_PARTITION_` to the partition column.
- The value of the partition folder name formed using the combination of the partition column name and value must not exceed 1024 characters. Otherwise, the mapping fails.
  If a partition folder name that has more than 1024 characters, you can perform one of the following task to truncate the partition folder name:
  - Increase the precision of the source field to 256. The mapping runs successfully and the Secure Agent truncates the data in the target field to 255 characters.
  - Specify a `substr` function through an Expression transformation on the partition column and truncate the value of the partition column.

Wildcard characters for mappings in advanced mode

When you create a mapping in advanced mode to read data from an Avro, flat, JSON, ORC, or Parquet file, you can use wildcard characters to specify the source file name.

To use wildcard characters for the source file name, enable the **Is Directory** and **Allow Wildcard Characters** options in the advanced source properties.

When you read an Avro, JSON, ORC, Parquet, or flat file, you can use the `?` and `*` wildcard characters to define one or more characters in a search.

You can use the following wildcard characters:

? (Question mark)

The question mark character (?) allows one occurrence of any character. For example, if you enter the source file name as `a?b.txt`, the Secure Agent reads data from files with the following names:

- `a1b.txt`
- `a2b.txt`
- `aab.txt`
- `acb.txt`

*** (Asterisk)**

The asterisk mark character (*) allows zero or more than one occurrence of any character. If you enter the source file name as `a*b.txt`, the Secure Agent reads data from files with the following names:

- `aab.txt`
- `a1b.txt`
- `ab.txt`
- `abc11b.txt`

Incrementally loading files for mappings in advanced mode

You can incrementally load source files in a directory to read and process only the files that have changed since the last time the mapping task ran.

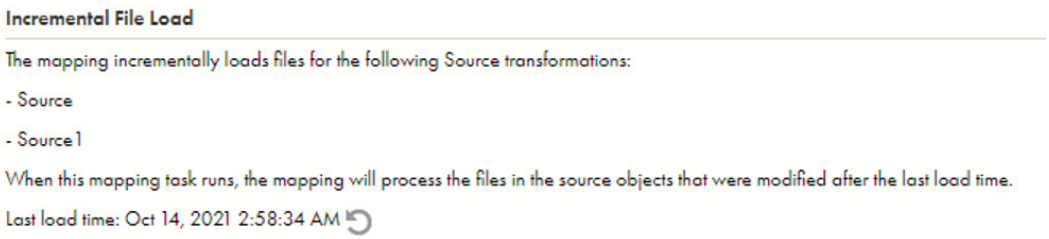
You can incrementally load files only from mappings in advanced mode. Ensure that all of the source files exist in the same Cloud environment.

To incrementally load source files from a Google Cloud Storage directory, select the **Is Directory** option and the **Incremental File Load** option in the advanced source properties of the Google Cloud Storage V2 source object.

When you incrementally load files from a Google Cloud Storage directory, the job loads files that have changed from the last load time to five minutes before the job started running. For example, if you run a job at 2:00 p.m, the job loads files changed before 1:55 p.m. The five-minute buffer ensures that the job loads only complete files, because uploading objects on Google Cloud Storage can take a few minutes to complete.

When you configure a mapping task, the **Incremental File Load** section lists the Source transformations that incrementally load files and the time that the last job completed loading the files. By default, the next job that runs checks for files modified after the last load time.

The following image shows the **Incremental File Load** section in the **Persisted Task Settings** page of the mapping task:



You can also override the load time that the mapping uses to look for changed files in the specified source directory. You can reset the incremental file load settings to perform a full load of all the changed files in the directory, or you can configure a time that the mapping uses to look for changed files.

A mapping in advanced mode that incrementally loads a directory that contains complex file formats such as Parquet and Avro, the mapping fails if there are no new or changed files in the source since the last run.

For more information on incremental loading, see Reprocessing incrementally-loaded source files in *Tasks* in the Data Integration documentation.

Rules and guidelines for mappings and mapping tasks

Use the following guidelines when you create mappings:

- When you use the Snappy compression format to write data to Google Cloud Storage, the mapping retains a `snappy-1.1.8****-libsnappyjava.so` file in the temp directory on the agent machine after it runs successfully.
- When you override the source or target file using the **Source File Name** or **Target File Name** advanced property, ensure that the file contains valid header columns.
- When you create a target based on a Avro source file and the source contains fields of Null or BigInt data type, the Secure Agent creates a field of the Integer data type in the Avro target file.
- When you create a Google Cloud Storage target at runtime and the source contains an empty table or view, the mapping fails, but the Secure Agent creates an empty file in Google Cloud Storage.
- When you read data from a Google Cloud Storage source and write data to a Google Cloud Storage target and the source and target connections use different project IDs associated with the same Google service account, the mapping fails.
- When you parameterize the object, you can override it with a file located within the specified object path and its subfolders. For example, if you define the object path as `bucket name/folder1/file1.txt` in a mapping task, you can replace the file with another one from the directory `bucket name/folder1` or any of its subfolders. However, if you want to override it with a file from a different directory, you need to parameterize the Google Cloud storage path and file name in the advanced properties of the source or target, and then provide a distinct object path in the parameter file.

Rules and Guidelines for mappings in advanced mode

Consider the following guidelines when you create a mapping in advanced mode:

- When you read or write hierarchical data to a Google Cloud Storage file in Avro, JSON, or Parquet format, you cannot preview data.
- Gzip compression does not apply when you write to a Google Cloud Storage file in Avro or ORC format.
- When you read data from Google Cloud Storage, you cannot preview data the data in the mapping.
- When you write data to a Google Cloud Storage file in ORC format, you cannot preview data.
- You cannot read files available in a sub-folder.
- You cannot use a multi-character delimiter in a mapping.
- You cannot write to the same object when you use multiple Google Cloud Storage targets in a mapping.
- You cannot override input parameters from the parameter file in a mapping. Instead, you can use in-out parameters to override them. When you configure in-out parameters, you need to completely parameterize the values. You cannot use partial parameterization.
- When write data to Google Cloud Storage flat file and select **Informatica Encryption** as the **Encryption Type**, ensure that you do not set the **Compression Format** to **Gzip**. Otherwise, the mapping fails with the following error:
  `java.lang.RuntimeException`
- When you use Informatica encryption to decrypt or encrypt files of large size, ensure that you have at least twice the disk space available in the advanced cluster.
  For example, if the size of a file is 75 GB, you must have at least 150 GB of disk space available in the advanced cluster to successfully encrypt or decrypt the file.
- When you set the qualifier mode to Minimal and use an escape character, the escape characters are not escaped and quoted in the target. To resolve this issue, set the qualifier mode to All.
- When you set the qualifier mode to All and do not specify a value for the qualifier, \00 (Null) is considered as the qualifier.
- When a column name in the Google Cloud Storage source file starts with a number and you create a Google Cloud Storage target at runtime, the corresponding target column is prefixed by an underscore character (_).
- When a column name in the source file contains special characters and you create a target at runtime, the Secure Agent replaces the special characters with underscore (_) character in the target file.
- When you perform update, upsert, or delete operations, the Secure Agent does not display the number of **Rows Processed** in **My Jobs** page.
- When you read data from a Google Cloud Storage flat file source and create a Google Cloud Storage target at runtime in Avro or ORC format, ensure that the column names in the source does not contain unicode characters in Hindi or Kannada.
- When you import a Google Cloud Storage source file in Parquet format to read Float values in an array and write the data to a Google Cloud Storage target file in Parquet format, precision loss is encountered in the target.
- When you import a Google Cloud Storage source file in Avro format to read Float values in an array and write the data to a Google Cloud Storage target file in Avro format, precision loss is encountered in the target.
- When you import a Google Cloud Storage JSON file and the file contains a column of Double data type, the data preview displays the Double values in scientific notation.

- When you import a Google Cloud Storage source file and the file name contains special characters, the Secure Agent replaces special characters with underscore (_) character.
- When you import a Google Cloud Storage source file and a file with the same name exist in another bucket, the Secure Agent imports the file with `1_` prefix.
  For example, if the source file name in Google Cloud Storage is `accounts-1.csv`, the Secure Agent imports the file as `1_accounts_1_csv`.
- When you write data to an existing target in Google Cloud Storage or create a new target at runtime, the Secure Agent creates a new folder with the following format and creates part files and success file in the folder:
  `<target_file_name>_<unique object ID>`

  For example, if you run a mapping to write data to `CustomerTgt.csv` file, the Secure Agent creates a folder `CustomerTgt.csv_233434` and creates the `SUCCESS.csv` file and `part-00000.csv` file. The Secure Agent writes data to the `part-00000.csv` file.

  In the subsequent re-run of the mapping, the Secure Agent creates a new part file in the same folder.
- When you write data to an existing target, the Secure Agent creates a success file in the created folder. If there are multiple success files in a bucket, the Secure Agent creates the file with `1_` prefix.
- A mapping configured with the following settings fails:
  - If you do not select **Optimize Object Metadata Import** in the connection properties.
  - If you select **Partition Directory** as the source object and the source advanced property **Is Directory** is enabled in the mapping.
  - If you provide the directory partition folder created using the connection in the source advanced property by overriding the directory partition path.

  To run the mapping successfully, select the connection property **Optimize Object Metadata Import** and refresh the metadata, or select the directory partition folder created with the connection as the source object, and run the mapping again.

Troubleshooting a mapping task

Time zone for the Date and Timestamp data type fields in Parquet or Avro file formats defaults to the Secure Agent host machine time zone.

When you run a mapping in advanced mode to read from or write to fields of the Date and Timestamp data types in the Parquet or Avro file formats, the time zone defaults to the Secure Agent host machine time zone.

To change the Date and Timestamp to the UTC time zone, you can either set the Spark properties globally in the Secure Agent directory for all the tasks in the organization that use this Secure Agent, or you can set the Spark session properties for a specific task from the task properties:

To set the properties globally, perform the following tasks:

1. Add the following properties to the `<Secure Agent installation directory>/apps/At_Scale_Server/41.0.2.1/spark/custom.properties` directory:
   - infacco.job.spark.driver.extraJavaOptions=-Duser.timezone=UTC
   - infacco.job.spark.executor.extraJavaOptions=-Duser.timezone=UTC
2. Restart the Secure Agent.

To set the properties for a specific task, navigate to the Spark session properties in the task properties, and perform the following steps:

- Select the session property name as `spark.driver.extraJavaOptions` and set the value to `-Duser.timezone=UTC`.
- Select `spark.executor.extraJavaOptions` and set the value to `-Duser.timezone=UTC`.

Data corruption occurs in the target for data of double data type.

When you read data of the double data type from a Google Cloud Storage JSON file and write data to a Google Cloud Storage flat file target, data corruption occurs in the target for the corresponding data of the Double data type.

Workaround: Change the data type of the target column from flat_string data type to flat_number data type and increase the precision to 38 and the scale to 15.

When you run the mapping, the Secure Agent writes the data of double data type to the target column of decimal data type with trailing zeros and without data loss.

CHAPTER 4

Migrating a mapping

You can configure a connection and mapping in one environment and then migrate and run the mapping in another environment.

You can also migrate mappings configured in advanced mode. After the migration, you can change the connection properties from the Administrator service, but you do not need to modify the mapping. Data Integration uses the configured runtime attributes from the earlier environment to run the mapping successfully in the new environment.

Consider a scenario where you develop a mapping in the development organization (Org 1) and you then migrate and run the mapping in the production organization (Org 2). After you migrate, you might want to use the same or a different connection endpoint or object path in Org 2. Based on your requirement, follow the guidelines in this section before you plan the migration.

Use the same object path for the migrated mapping

If you want the migrated mapping in Org 2 to use the same object path as in Org 1, you must maintain the same folder path and file name in the Google Cloud Storage account for Org 2.

For example, if you have two different accounts, Account1 used for Org 1 and Account2 used for Org 2, the folder path and the file name must be the same in both the accounts:

Account1: folder1/filename1

Account2: folder1/filename1

In this scenario, you do not need to override the folder path and file name in the advanced properties.

Use a different object path for the migrated mapping

After you migrate the mapping, you can use a different object path to run the mapping from the new environment.

In this scenario, before you migrate the mapping, you can change the object metadata, runtime attributes, or the connection attributes to reflect the object path in the migrated environment. You do not have to edit or update the mapping in the new environment.

As a rule, when you specify the folder path and file name in the advanced properties, connection, or object properties, Data Integration honors the attributes in the following order of precedence:

1. **Runtime advanced attributes**. The advanced properties such as the folder path and file name in the Source or Target transformation in a mapping.
2. **Connection attributes**. The bucket name attribute in the connection properties.
3. **Object metadata**. The object selected in the Source or Target transformation in a mapping.

Migration options

When you migrate, you can choose from one of the following options to update the object path:

Option 1. Update the connection properties to reference the new object

When you import the mapping into Org 2, in the **Review Connections** section, you can change the existing connection to map to the connection that has access to the specified folder path and file name in Org 2.

Option 2. Override the properties from the advanced properties

Before the migration, specify the required folder path and file name for the object from Org 2 in the advanced properties of the Org 1 mapping.

After the migration, when you run the mapping, the Secure Agent uses the configured advanced parameters to override the object specified in the mapping imported from Org 1.

Option 3. Parameterize the properties in the mapping

You can choose to parameterize the advanced attributes, such as the folder path and file name before the migration. You can configure input parameters, in-out parameters, and parameter files in the mapping. When you use a parameter file, you can save the parameter file on a local machine or in a cloud-hosted directory. After you migrate the mapping, do not edit or update the mapping. If you have used in-out parameters for the advanced attributes such as for the folder path and file name, you can update these from the parameter file.

Parameterizing only the advanced properties, but not the object in the mapping

If you want to parameterize only the advanced properties and use them at runtime, select a placeholder object in the object properties in the mapping and then specify an override to this placeholder object from the advanced properties. Ensure that the placeholder object contains the same metadata as the corresponding table that you specify as an override. When you run the mapping, the value specified in the advanced property overrides the placeholder object.

Parameterizing both the object and the advanced properties

If you want to keep both the Google Cloud Storage object type and the advanced fields parameterized, you must leave the **Allow parameter to be overridden at runtime** option unselected in the input parameter window while adding the parameters, and then select the required object at the task level. When you run the task, the values specified in the advanced properties take precedence.

Rules and guidelines for migrating a mapping

Consider the following rules and guidelines when you use the same or a different object path for the migrated mapping :

- The following table lists the transformation, object type, and the fields in the advanced properties of a mapping that you can retain when you migrate to the new environment:

Transformations	Object Type	Advanced Fields
Source	Single object	Folder path and file name
Target	Single object	

- After you migrate the mapping from Org1 to Org2, you must not edit the mapping.
- For an existing target object in Org2, you must specify the folder path and file name in the advanced properties. If the object does not exist in Org2 or the connection does not have access to the object that you used in Org1, the mapping fails with a 403 error.
- When you use a parameter file in a cloud-hosted directory and you use different Google Cloud Storage V2 connections in Org1 and Org2, you must specify a bucket name in the **Bucket Name** connection property. Otherwise, the mapping fails.
- Ensure that you specify a valid folder and file name in the advanced properties. Otherwise, the mapping fails.
- You cannot configure encryption in a mapping that you migrate to the new environment.
- Before you specify a Google Cloud Storage folder path and enable the Is Directory option to read files from a Google Cloud Storage folder and subfolder, specify the following value for the INFA_DEBUG property in the System Configuration Details section for the Secure Agent : `-DMetadataFromRuntimeOverrideGCSv2=false`
- When you create a mapping in advanced mode, ensure that you do not select the **Allow Wildcard Characters** option or specify wildcard characters for the source file name. Else, the mapping fails.
- When you create a mapping in advanced mode, ensure that you do not read from and write to partition columns. Else, the mapping fails.

CHAPTER 5

Upgrading to Google Cloud Storage V2 Connector

If you are accessing Google Cloud Storage using the Google Cloud Storage connection, you can upgrade to the newer Google Cloud Storage V2 Connector. You can replace the source or target connection type in existing mappings and mapping tasks that use the Google Cloud Storage connection with the Google Cloud Storage V2 connection.

After you replace the connection in an existing mapping, re-import the Google Cloud Storage object and remap the fields in the mapping. The configured advanced source and target properties in the fields that are common between the two connectors are retained in the new connector. You can run the mapping successfully using the configured values from the old connector. You can additionally configure features that the enhanced Google Cloud Storage V2 Connector offers.

The encryption type in the advanced target properties is not retained. You must manually specify the encryption type after you switch the connection.

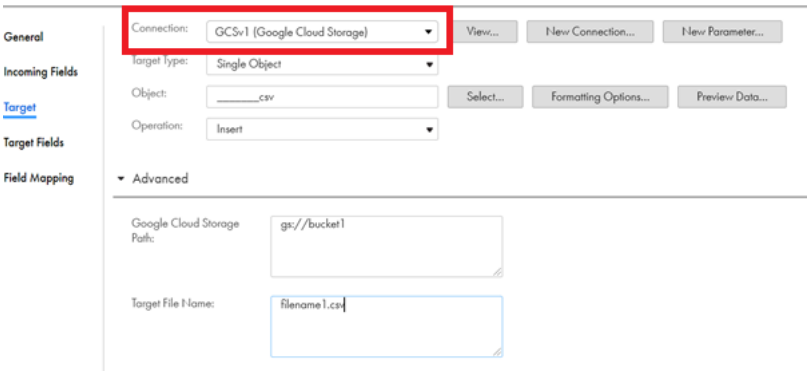
Note: If you are using the Google Cloud Storage V1 connection in mappings to read from or write data to Google Cloud Storage, Informatica recommends you to use the Google Cloud Storage V2 connection to make use of the features that the enhanced connector offers. To get the license for Google Cloud Storage V2 Connector, contact Global Customer support.

Connection switching example

You want to upgrade your existing Google Cloud Storage V1 mapping that uses the Google Cloud Storage connection to the Google Cloud Storage V2 connection.

1. Open the existing Google Cloud Storage V1 mapping that you want to upgrade to Google Cloud Storage V2.

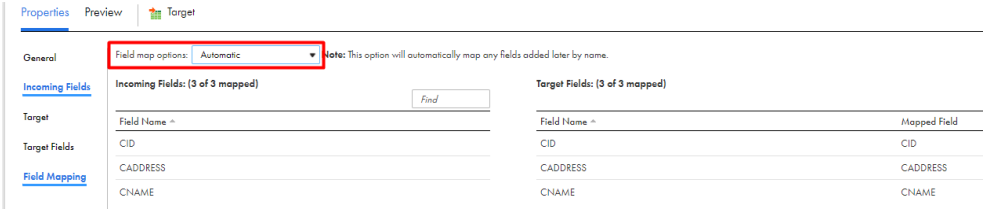
   The following image shows an existing mapping that uses the Google Cloud Storage connection and contains the configured advanced properties in the Target transformation:



   The configured target object path in this example is: `gs://bucket1/__________.csv`

2. To retain the mapped fields from the field mapping when you switch the connection, on the **Field Mapping** tab, choose from the following **Field Map Options** menu in the Google Cloud Storage V1 mapping:

   - To retain the fields automatically mapped after the switch, select **Automatic**.

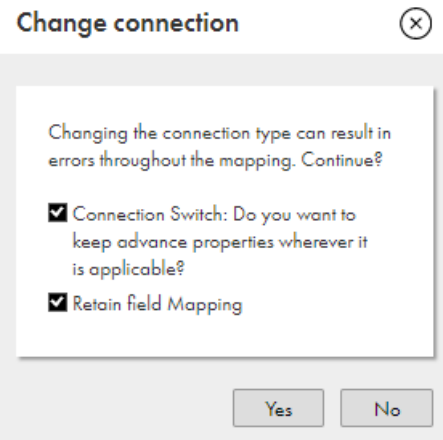


   - To manually map the retained fields after the switch, select **Manual**.

     Note: When you select manual, after switching the connection, you have the option to automap the retained fields using the previous mapping.

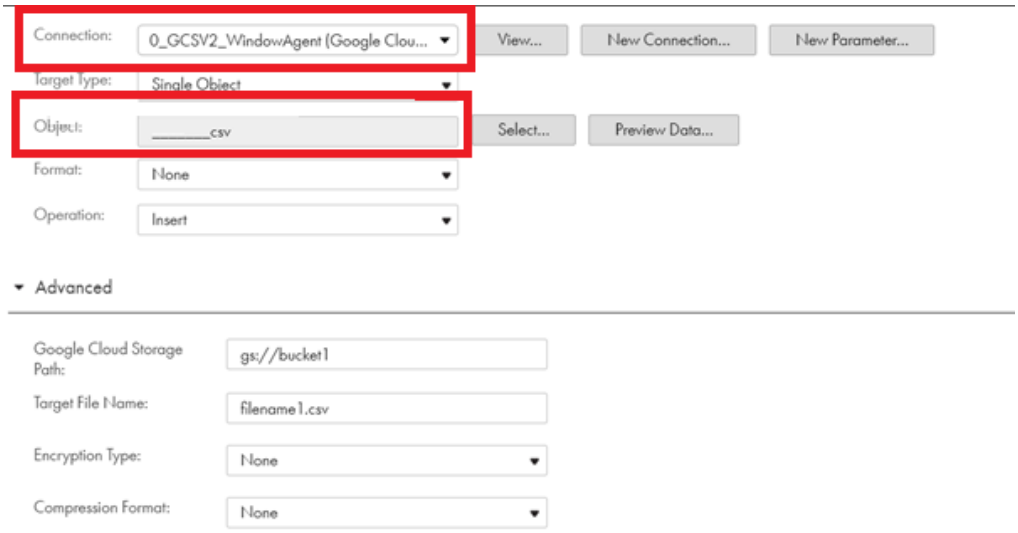
3. To switch the connection, in the **Connection** field, change the connection from Google Cloud Storage V1 to Google Cloud Storage V2.
4. In the **Change Connection** dialog box, select the following properties, and click **Yes**:
   - **Connection switch**. Switches to the connection that you select.
   - **Retain field mapping**. Retains the configured field mappings from Google Cloud Storage V1.

The following image shows the options that you must select:



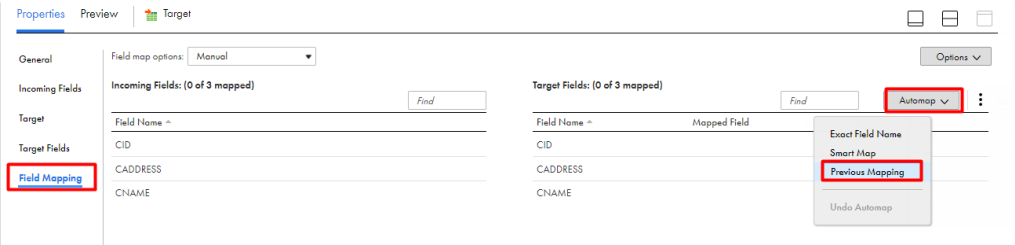
5. Use the same object path in the mapping as Google Cloud Storage V1.

   The following image shows the switched connection with the same object path: `gs://bucket1/_____.csv`



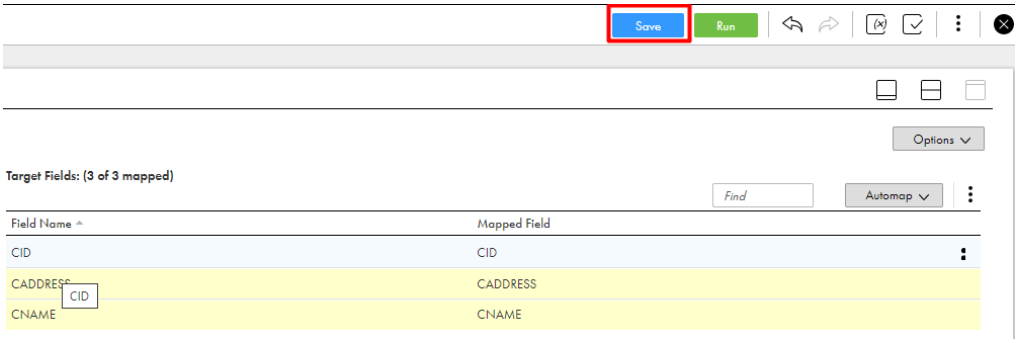
   The configured target advanced properties from the Google Cloud Storage V1 mapping reflect in the Target transformation.

6. If you had selected **Manual** on the **Field Mapping** tab in the Google Cloud Storage V1 mapping and you want to reflect the field mappings in Google Cloud Storage V2, on the **Field Mapping** tab, select **Automap**, and then select **Previous Mapping**.



Note: If you had selected **Automatic** in Google Cloud Storage V1, you do not have to perform this task.

In the following image, the configured mapped fields from the Google Cloud Storage V1 mapping reflects in the Google Cloud Storage V2 mapping:



7. Click **Save**.

Advanced properties retained after the switch

The following table lists the configured advanced target properties from Google Cloud Storage Connector that are retained in Google Cloud Storage V2 Connector after the switch:

- Google Cloud Storage Path
- Target File Name

Note: Even after you replace the connection with the Google Cloud Storage V2 connection, the object field displays the object that you selected in the earlier connection. You need to re-import the Google Cloud Storage object and remap the fields in the mapping.

APPENDIX A

Data type reference

Data Integration uses the following data types in mappings and mapping tasks with Google Cloud Storage:

Google Cloud Storage native data types

Google Cloud Storage data types appear in the **Fields** tab for Source and Target transformations when you choose to edit metadata for the fields.

Transformation data types

Set of data types that appear in the remaining transformations. They are internal data types based on ANSI SQL-92 generic data types, which Data Integration uses to move data across platforms. Transformation data types appear in all remaining transformations in a mapping, synchronization task, or mapping task.

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.

Flat Google Cloud Storage file data types and transformation data types

Flat Google Cloud Storage file data types map to transformation data types that the Secure Agent uses to move data across platforms.

The following table lists the Google Cloud Storage data types that the Secure Agent supports and the corresponding transformation data types:

Flat Google Cloud Storage Data Type	Transformation Data Type	Description
BigInt	BigInt	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807 Precision of 19, scale of 0
Number	Decimal	Precision 1 to 28 digits, scale 0 to 28

Flat Google Cloud Storage Data Type	Transformation Data Type	Description
Nstring	Text	1 to 104,857,600 characters Fixed-length or varying-length string.
STRING	String	1 to 104,857,600 characters Default precision is 256. You can increase the value up to 104857600 characters.

Note: BigInt, Number, and Nstring data types are applicable only when you select **Import from Schema File** as the **Schema Source** in the file formatting options.

Avro Google Cloud Storage file data types and transformation data types

Avro Google Cloud Storage file data types map to transformation data types that the Secure Agent uses to move data across platforms.

The following table lists the Avro Google Cloud Storage file data types that the Secure Agent supports and the corresponding transformation data types:

Avro Google Cloud Storage File Data Type	Transformation Data Type	Range and Description
ARRAY[1]	Array	Unlimited number of characters
BOOLEAN	Integer	TRUE (1) or FALSE (0)
BYTES	Binary	Precision 4000
DOUBLE	Double	Precision 15
FLOAT	Double	Precision 15
INT	Integer	-2,147,483,648 to 2,147,483,647 Precision 10, scale 0
LONG	Bigint	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807 Precision 19, scale 0
MAP[1]	Map	Unlimited number of characters
NULL	Integer	-2,147,483,648 to 2,147,483,647 Precision 10, scale 0
RECORD[1]	Struct	Unlimited number of characters

Avro Google Cloud Storage File Data Type	Transformation Data Type	Range and Description
STRING	String	1 to 104,857,600 characters Default precision is 256. You can increase the value up to 104857600 characters.
UNION[1]	Corresponding data type in a union of ["primitive_type|complex_type", "null"] or ["null", "primitive_type|complex_type"].	Dependent on primitive or complex data type.

[1]Applies only to mappings in advanced mode.

Note: Google Cloud Storage V2 Connector does not support the following Avro complex data types:

- Enum
- Fixed

JSON Google Cloud Storage file data types and transformation data types

JSON Google Cloud Storage file data types map to transformation data types that the Secure Agent uses to move data across platforms.

The following table lists the JSON Google Cloud Storage file data types that the Secure Agent supports and the corresponding transformation data types:

JSON Google Cloud Storage File Data Type	Transformation Data Type	Range and Description
ARRAY[1]	Array	Unlimited number of characters
BIGINT	Bigint	Precision of 19 digits, scale of 0
BOOLEAN	Integer	TRUE (1) or FALSE (0)
DOUBLE	Double	Precision 15
INTEGER	Integer	-2,147,483,648 to 2,147,483,647 Precision of 10, scale of 0

JSON Google Cloud Storage File Data Type	Transformation Data Type	Range and Description
OBJECT[1]	Struct	Unlimited number of characters
STRING	String	1 to 104,857,600 characters Default precision is 256. You can increase the value up to 104857600 characters.

[1]Applies only to mappings in advanced mode.

Note: The following JSON complex data types are not applicable for Google Cloud Storage V2 Connector:

- Date/Timestamp
- Enum
- Union

ORC Google Cloud Storage file data types and transformation data types

ORC Google Cloud Storage file data types map to transformation data types that the Secure Agent uses to move data across platforms.

The following table lists the ORC Google Cloud Storage file data types that the Secure Agent supports and the corresponding transformation data types:

ORC Google Cloud Storage File Data Type	Transformation Data Type	Range and Description
BigInt	BigInt	-9223372036854775808 to 9,223,372,036,854,775,807
Boolean	Integer	TRUE (1) or FALSE (0)
Char	String	1 to 104,857,600 characters
Date	Date/Time	Jan 1, 1753 A.D. to Dec 31, 4712 A.D. (precision to microsecond)
Double	Double	Precision of 15 digits
Float	Double	Precision of 15 digits
Integer	Integer	-2,147,483,648 to 2,147,483,647
SmallInt	Integer	-32,768 to 32,767

ORC Google Cloud Storage File Data Type	Transformation Data Type	Range and Description
String	String	1 to 104,857,600 characters
Timestamp	Date/Time	1 to 19 characters Precision 19 to 26, scale 0 to 6
TinyInt	Integer	-128 to 127
Varchar	String	1 to 104,857,600 characters

Parquet Google Cloud Storage file data types and transformation data types

Parquet Google Cloud Storage file data types map to transformation data types that the Secure Agent uses to move data across platforms.

The following table lists the Google Cloud Storage file data types that the Secure Agent supports and the corresponding transformation data types:

Parquet Google Cloud Storage File Data Type	Transformation	Description
BIGINT	Bigint	Precision of 19 digits, scale of 0.
BOOLEAN	Integer	TRUE (1) or FALSE (0)
DATE	Date/Time	January 1, 0001 to December 31, 9999.
DECIMAL	Decimal	Precision 1 to 28 digits, scale 0 to 28. **Note:** You cannot use decimal values with precision greater than 28.
DOUBLE	Double	Precision of 15 digits.
FLOAT	Double	Precision of 15 digits.
Int32	Integer	-2,147,483,648 to 2,147,483,647 Precision of 10, scale of 0
Int64	Bigint	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807 Precision of 19, scale of 0
Int96	Date/Time	Jan1,0001 to Dec 31, 9999. Precision of 29, scale of 9.
Map[1]	Map	Unlimited number of characters.

Parquet Google Cloud Storage File Data Type	Transformation	Description
Struct[1]	Struct	Unlimited number of characters.
String	String	-1 to 104,857,600 characters.
Time	Date/Time	Time of the day. Precision to microsecond.
Timestamp	Date/Time	January 1, 0001 00:00:00 to December 31, 9999 23:59:59.997. Precision to microsecond. **Note:** You cannot set precision to nanoseconds.
group(LIST)[1]	Array	Unlimited number of characters.

[1]Applies only to mappings in advanced mode.

Note: The following Parquet complex data types are not applicable for Google Cloud Storage V2 Connector:

- Byte_Array
- Enum
- Fixed Length Byte Array
- Union

The Parquet schema that you specify to read or write a Parquet file must be in lowercase. Parquet does not support case-sensitive schema.

INDEX

A

Avro Google Cloud Storage file data types
  transformation data types 44

C

Cloud Application Integration community
  URL 5
Cloud Developer community
  URL 5
connections
  Google Cloud Storage V2 11

D

data encryption
  Informatica encryption 23
Data Integration community
  URL 5
data type reference
  overview 43
directory source
  Google Cloud Storage sources 22
dynamic schema handling 25

G

Google Cloud Storage
  access control lists 8
  components 8
  features 8
  introduction 8
  native data types 43
Google Cloud Storage components
  buckets 8
  objects 8
  projects 8
Google Cloud Storage connections
  overview 11
Google Cloud Storage Connector
  administration 8
Google Cloud Storage V2
  connection properties 11
Google Cloud Storage V2 Connector
  overview 7
  supported task and object types 7

I

incrementally load files
  overview 31
Informatica Global Customer Support
  contact information 6
Informatica Intelligent Cloud Services
  web site 5

J

JSON Google Cloud Storage file data types
  transformation data types 45

M

maintenance outages 6
mapping
  Google Cloud Storage V2 sources 14
  Google Cloud Storage V2 targets 17
mapping advanced properties
  Google Cloud Storage V2 source 14
  Google Cloud Storage V2 target 17
mapping properties
  Google Cloud Storage V2 source 14
  Google Cloud Storage V2 targets 17
mappings in advanced mode
  example 26
  rules and guidelines 33

O

ORC file data types
  transformation data types 46

P

Parquet Google Cloud Storage file data types
  transformation data types 47

R

rules and guidelines
  Avro format types 32
  file format types 22
  parquet format types 32

S

status
  Informatica Intelligent Cloud Services 6
system status 6

T

transformation
  data types 43
troubleshooting
  mapping task 34
trust site
  description 6

U

upgrade notifications 6

W

web site 5
wildcard character
  overview 30