Informatica® Data Integration - Free & PayGo
February 2025

Tasks

Informatica Data Integration - Free & PayGo Tasks
February 2025

© Copyright Informatica LLC 2022, 2025

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

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

Informatica, Informatica Cloud, Informatica Intelligent Cloud Services, PowerCenter, PowerExchange, and the Informatica logo 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.

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

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

Publication Date: 2025-02-07

Table of Contents

Preface 6

Informatica Resources. 6

Informatica Documentation. 6

Informatica Intelligent Cloud Services web site. 6

Informatica Intelligent Cloud Services Communities. 6

Informatica Intelligent Cloud Services Marketplace. 6

Data Integration connector documentation. 7

Informatica Knowledge Base. 7

Informatica Intelligent Cloud Services Trust Center. 7

Informatica Global Customer Support. 7

Chapter 1: Data integration tasks. 8

Data filters. 9

Simple data filters. 9

Advanced data filters. 10

Data filter operators. 10

Data filter variables. 11

Rules and guidelines for data filters. 12

Field metadata. 12

Field expressions. 13

Creating a field expression. 13

Transformation language components for expressions. 14

Expression syntax. 14

String and numeric literals. 14

Rules and guidelines for expressions. 15

Adding comments to expressions. 15

Reserved words. 16

Advanced session properties. 17

Parameter files. 23

Schedules. 23

Repeat frequency. 24

Time zones and schedules. 25

Daylight Savings Time changes and schedules. 25

Creating a schedule. 26

Running a task on a schedule. 26

Email notification. 27

Preprocessing and postprocessing commands. 27

Preprocessing and postprocessing SQL commands. 28

Preprocessing and postprocessing operating system commands. 28

Monitoring a job. 28

Stopping a job. 28
Guidelines for sources and targets in data integration tasks. 29
Rules and guidelines for flat file sources and targets. 29
Rules and guidelines for database sources and targets. 29

Chapter 2: Mapping tasks. 31
Mapping task templates. 31
Related objects. 32
Advanced relationships. 33
SQL ELT optimization. 33
Simultaneous task runs. 34
Schema change handling. 34
Dynamic schema handling options. 35
Dynamic schema change handling rules and guidelines. 36
Schema mismatch handling. 37
Mapping task configuration. 37
Creating a mapping task. 37
Configuring sources. 38
Configuring targets 40
Configuring input parameters. 42
Configuring persistent values. 43
Configuring runtime options. 43
Viewing and editing mapping task details. 47
Sequence values. 47
Running a mapping task. 48

Chapter 3: Data transfer tasks. 49
Task operations. 49
Data transfer task sources. 50
Source filters. 50
Sort conditions. 50
Custom queries. 50
Second sources. 51
Lookup condition. 51
Second source filters. 52
Data transfer task targets. 52
Database target truncation. 53
Update columns. 53
Field mapping. 53
Field data types. 54
Configuring a data transfer task. 54
Defining the data transfer task. 54
Configuring the source. 55

Configuring a second source. 55

Configuring the target. 56

Configuring the field mapping. 57

Configuring runtime options. 57

Running a data transfer task. 57

Chapter 4: Data loader tasks. 58

Step 1. Connect to your source. 59

Guidelines for customizing your source. 60

Selecting source objects. 61

Defining the source format. 61

Selecting source fields. 61

Configuring filters. 62

Selecting primary key fields. 63

Selecting watermark fields. 64

Step 2. Connect to your target. 65

Incremental loading. 65

Dropping and re-creating target tables. 66

Step 3. Run the task. 66

Running the task on a schedule. 67

Configuring email notifications. 67

Handling runtime errors. 67

Configuring the task location. 67

Runtime environment. 68

Index. 69

Preface

Use *Tasks* to learn how to set up and run Data Integration tasks manually or on a schedule.

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

Data integration tasks

A data integration task is a process that you configure to analyze, extract, transform, and load data. You can run individual tasks manually or set tasks to run on a schedule.

You can use the following tasks to integrate data:

Mapping tasks

Mapping tasks process data based on the data flow logic defined in a mapping.

A mapping reads data from one or more sources, transforms the data based on logic that you define, and writes it to one or more targets. Create a mapping when you need to augment or manipulate your data before you load it to a target. For example, if you need to aggregate data, calculate values, perform complex joins, normalize data, or route data to different targets, you can create a mapping to do this.

A mapping task runs the data flow logic that you've defined in the mapping. Choose this task type after you've created a mapping so that you can run the data flow logic defined in the mapping.

Data transfer tasks

Data transfer tasks move data from one or two sources to a target. You can also choose to sort and filter the data before you load it to the target.

Choose this task type when you want to transfer data from a source object, optionally add fields from a second source object, and write the data to a new or existing target object without changing the source data. For example, if you want to move customer records from an on-premises database table to a table in your cloud data warehouse, create a data transfer task.

Data loader tasks

Data loader tasks provide secure data loading from multi-object sources to corresponding objects in your cloud data warehouse. They can load data incrementally and provide support for schema drift.

To optimize performance, data loading occurs in parallel batches. To fine-tune the data, you can exclude certain objects and fields and also apply some simple filters. If your source data changes frequently, you can load only new and changed records each time the task runs.

Choose this task type when you need to ingest data as-is from multiple objects into your cloud data warehouse. For example, if you need to repeatedly load all the data from files an Amazon S3 bucket to corresponding tables in Snowflake Data Cloud, create a data loader task.

When you create a task, Data Integration walks you through the required steps. The options and properties that display depend on the task type

Data filters

A data filter is a filter that you apply on a source to limit the number of rows processed by the task. Create simple or advanced data filters to filter the source data before it enters the data flow.

You can create the following type of data filters in a data transfer task:

- Simple
- Advanced

You can create a set of data filters for each object included in the task. Each set of data filters acts independently of the other sets.

Simple data filters

You can create one or more simple data filters.

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

For example, you load rows from the Account Salesforce object to a database table. However, you want to load only accounts that have greater than or equal to $100,000 in annual revenue and that have more than 500 employees. You configure the following simple data filters:

Field	Operator	Field Value
AnnualRevenue	>=	100000
NumberOfEmployees	>	500

Configuring simple data filters

You configure simple data filters in the task wizard.

1. To create a simple data filter, perform one of the following actions:
   - In a synchronization task, replication task, or masking task, on the **Data Filters** page, click **New**. The **Data Filter** dialog box appears.
   - In a mapping task, in the **Filters** area on the **Sources** page, select **Simple** and then click **New**.
   - In a data transfer task, in the **Filters** area on the **Source** or **Second Source** page, select **Simple** and then click **New**.
2. If necessary, specify the object on which to create the data filter.

   You create separate data filters for each source object included in the task.
3. Enter the filter condition based on the field, operator, and field value.
4. Click **OK**.
5. Create additional simple data filters as needed.

   To delete a data filter, click the **Delete** icon next to the data filter.
6. Click **Next**.

Advanced data filters

Create an advanced data filter to create complex expressions that use AND, OR, or nested conditions.

When you create an advanced data filter, you enter one expression that contains all filters. The expression that you enter becomes the WHERE clause in the query used to retrieve records from the source.

For mappings in SQL ELT mode, use expression syntax that is valid in your cloud data warehouse. For more information, see the documentation for your cloud data warehouse. For other types of mappings, use Informatica transformation language expression syntax.

For example, you create a mapping to load rows from the Account Salesforce object to a database table. However, you want to load records where the billing state is California or New York and the annual revenue is greater than or equal to $100,000. You configure the following advanced filter expression:

```
(BillingState = 'CA' OR BillingState = 'NY') AND (AnnualRevenue >= 100000)
```

When you create a data filter on a Salesforce object, the corresponding task generates a SOQL query with a WHERE clause. The WHERE clause represents the data filter. The SOQL query must be less than 20,000 characters. If the query exceeds the character limit, the following error appears:

```
Salesforce SOQL limit of 5000 characters has been exceeded for the object: <Salesforce
object>. Please exclude more fields or decrease the filters.
```

Note: Filter conditions are not validated until runtime.

Configuring advanced data filters

Configure advanced data filters in the task wizard. You can convert simple data filters to an advanced data filter, but you can't convert an advanced data filter to simple data filters.

1. In a data transfer task, in the **Filters** area on the **Source** or **Second Source** page, select **Advanced**.

   To convert all simple data filters to one advanced data filter, select **Advanced**.
2. If necessary, specify the object on which to create the data filter.

   You create separate data filters for each source object included in the task.
3. Enter the filter expression.

   Click the field name to add the field to the expression.
4. Click **OK**.

   To delete a data filter, click the **Delete** icon next to the data filter.
5. Click **Next**.

Data filter operators

You can use specific operators with each field type.

For mappings in SQL ELT mode, use operators that are valid in your cloud data warehouse's expression language. For more information, see the documentation for your cloud data warehouse. For other types of mappings, use the Informatica transformation language operators that are listed in the following table.

The following table shows the operators you can use for each field type:

Field type	Operators
Boolean	=, !=, Is Null, Is Not Null
Currency	=, !=, <, <=, >, >=, Is Null, Is Not Null
Date	=, !=, <, <=, >, >=, Is Null, Is Not Null
Datetime	=, !=, <, <=, >, >=, Is Null, Is Not Null
Double	=, !=, <, <=, >, >=, Is Null, Is Not Null
ID	=, !=, Is Null, Is Not Null
Int	=, !=, <, <=, >, >=, Is Null, Is Not Null
Reference	=, !=, Is Null, Is Not Null
String	=, !=, LIKE'_%', LIKE'%_', LIKE'%_%', Is Null, Is Not Null, <, <=, >, >=
Textarea	=, !=, LIKE'_%', LIKE'%_', LIKE'%_%', Is Null, Is Not Null, <, <=, >, >=
All other field types	=, !=, Is Null, Is Not Null

Data filter variables

Data filter variables represent the date or time that a task previously ran. Use data filter variables to help capture the source data that changed since the last task run. You can use data filter variables in simple and advanced data filter conditions.

You can use the following data filter variables:

Variable	Description
$LastRunDate	The start date in GMT time zone of the last task run that was successful or ended with a warning. Does not include time. For example, 2018-09-24. Can be used as a value for filter where the field type is DATE.
$LastRunTime	The start date and time in GMT time zone of the last task run that was successful or ended with a warning. For example, 2018-09-24 15:23:23. Can be used as a value for filter where the field type is DATETIME. You can't use the $LastRunTime variable with DATE fields .

You can't use data filter variables in expressions in mappings SQL ELT mode.

For example, you can include the following simple filter condition:

```
LastModifiedDate > $LastRunTime
```

Note: Consider time zone differences when comparing dates across time zones. The date and time of the $LastRunDate and $LastRunTime variables are based on the time zone set in Informatica Intelligent Cloud Services. The date and time of the actual job is based on the GMT time zone for Salesforce sources and the database server for database sources. The difference in the time zones may yield unexpected results.

Rules and guidelines for data filters

Use the following rules and guidelines for data filters:

- When you create a data filter in a mapping in SQL ELT mode, use expression syntax that is valid in your cloud data warehouse. For more information, see the documentation for your cloud data warehouse. For other types of mappings, use Informatica transformation language expression syntax.
- Data filters must contain valid SQL or SOQL operators.
- You cannot include simple and advanced data filters in the same task.
- When you convert a simple data filter to an advanced data filter, you cannot convert the advanced data filter back to a simple data filter.
- A task fails if the fields included in the data filter no longer exist or if the data types of the fields change. If a data type changes, edit the task.
- You can select Equals, Not Equals, Is Null, or Is Not Null operators on fields of the Other data type.
- Applications do not apply filters with Equals, Starts With, or Ends With operators and string fields that contain data that starts or ends with a single quotation mark. To filter these records, use the Contains operator.
- You can only use IS NULL and LIKE operators in data filters for fields of the Text, Ntext, and Image data types.
- If you specify a date and no time for a date/time filter, Data Integration uses 00:00:00 (12:00:00 a.m.) as the time.
- The list of available operators in a simple data filter depends on the data type of the field included in the data filter. Some operators do not apply to all fields included in data filters.
- When you enter more than one simple data filter, applications filter rows that meet the requirements of all data filters.
- When you use a parameter in a data filter, start the data filter with the parameter. For example, use $ $Sales=100000 instead of 100000=$$Sales.

Field metadata

You can view and edit field metadata such as the type, precision, and scale for parameterized source, target, and lookup objects with certain connection types. You can view and edit field metadata in mapping tasks .

To view and edit field metadata in a mapping task, expand the **Field Data Types** area on the appropriate page of the mapping task editor. You configure source and lookup field metadata on the **Sources** page and target field metadata on the **Targets** page.

If you edit field metadata in the task and the field metadata changes after the task is saved, Data Integration uses the updated metadata. Typically, the is the desired behavior. However, if the task uses a flat file connection and you want to retain the metadata used at design time, enable the **Retain existing fields at runtime option**.

To see if a connector supports field metadata configuration, see the help for the appropriate connector.

Field expressions

When you configure a data transfer task, you can configure the field mapping. The field mapping defines how source fields are mapped to target fields. You can specify an expression for each field mapping.

You can map multiple source fields to the same target field. For example, you can map SourceFieldA and SourceFieldB to TargetFieldC.

Data Integration might suggest operations when you map multiple source fields to a single target field. For example, if you map multiple text fields to a target text field, Data Integration concatenates the source text fields by default. You can change the default expression.

Data Integration provides a transformation language that includes SQL-like functions to transform source data. Use these functions to write expressions, which modify data or test whether data matches the conditions that you specify.

For more information about functions and the Data Integration transformation language, see *Function Reference*.

Creating a field expression

Create a field expression in a task wizard.

1. On the **Field Mappings** page, select the target field for which you want to add an expression.
2. Perform one of the following actions based on your task type:
   - Click **Add or Edit Expression**.
   - Click **Actions** > **Edit Expression**, or click the field name.

   By default, the **Field Expression** dialog box shows the source field as the expression, which indicates that the target contains the same value as the source.
3. Enter the new field expression.

   To include source fields and system variables in the expression, you can select them from the **Source Fields** and **System Variables** tabs to insert them into the expression or you can add them to the expression manually.
4. Click **Validate Mapping** to validate the field mappings.
5. Click **Save**.

Validating expressions in field mappings

Use the following rules and guidelines when you validate an expression in a field mapping:

- When you validate mappings, Data Integration performs the following validations:
  - Verifies that the source and target fields in the task exist in the source or target. If the field does not exist, an error appears.
  - Verifies that all column data types are string and all field expressions contain string operations when the source and target are flat files.
  - Verifies that the correct parameters are used for each function and that the function is valid.
- The expression validator does not perform case-sensitive checks on field names.

- The expression validator verifies that the data type of a field in an expression matches the data type expected by the containing function. However, the expression validator does not check for incompatible data types between the following sets of objects:
  - Source and target fields of tasks.
  - Source field in a lookup condition and the lookup field.
  - Output of an expression or lookup and the target field.
    The expression or lookup with these incompatible data types may validate successfully, but, at runtime, the task fails and an error appears.
- If you map a string source field to a number target field, the validation succeeds. Data Integration tries to convert the string to a number using the atoi (ASCII to Integer) C function.
- The expression validator does not validate lookups.

Transformation language components for expressions

The transformation language includes the following components to create simple or complex expressions:

- Fields. Use the name of a source field to refer to the value of the field.
- Literals. Use numeric or string literals to refer to specific values.
- Functions. Use these SQL-like functions to change data in a task.
- Operators. Use transformation operators to create expressions to perform mathematical computations, combine data, or compare data.
- Constants. Use the predefined constants to reference values that remain constant, such as TRUE.

In mappings in SQL ELT mode, you use your cloud data warehouse's native expression components to create expressions. You don't use the Informatica transformation language to create expressions.

Expression syntax

You can create a simple expression that only contains a field, such as ORDERS, or a numeric literal, such as 10. You can also write complex expressions that include functions nested within functions, or combine different fields using the transformation language operators.

Note: Although the transformation language is based on standard SQL, there are differences between the two languages.

String and numeric literals

You can include numeric or string literals.

Enclose string literals within single quotation marks. For example:

```
'Alice Davis'
```

String literals are case sensitive and can contain any character except a single quotation mark. For example, the following string is not allowed:

```
'Joan's car'
```

To return a string containing a single quotation mark, use the CHR function:

```
'Joan' || CHR(39) || 's car'
```

Do not use single quotation marks with numeric literals. Just enter the number you want to include. For example:

```
.05
```

or

```
$$Sales_Tax
```

Rules and guidelines for expressions

Use the following rules and guidelines when you write expressions:

- For each source field, you can perform a lookup or create an expression. You cannot do both.
- You cannot use strings in numeric expressions.
  For example, the expression `1 + '1'` is not valid because you can only perform addition on numeric data types. You cannot add an integer and a string.
- You cannot use strings as numeric parameters.
  For example, the expression `SUBSTR(TEXT_VAL, '1', 10)` is not valid because the SUBSTR function requires an integer value, not a string, as the start position.
- You cannot mix data types when using comparison operators.
  For example, the expression `123.4 = '123.4'` is not valid because it compares a decimal value with a string.
- You can pass a value from a field, literal string or number, or the results of another expression.
- Separate each argument in a function with a comma.
- Except for literals, the transformation language is not case sensitive.
- The colon (:), comma (,), and period (.) have special meaning and should be used only to specify syntax.
- Data integration tasks treat a dash (-) as a minus operator.
- If you pass a literal value to a function, enclose literal strings within single quotation marks. Do not use quotation marks for literal numbers. Data integration tasks treat any string value enclosed in single quotation marks as a character string.
- Do not use quotation marks to designate fields.
- You can nest multiple functions within an expression. Data integration tasks evaluate the expression starting with the innermost function.
- When you use a parameter in an expression, use the appropriate function to convert the value to the necessary data type. For example, you might use the following expression to define a quarterly bonus for employees:

  ```
  IIF((EMP_SALES < TO_INTEGER($$SalesQuota), 200, 0)
  ```

Adding comments to expressions

You can use the following comment specifiers to insert comments in expressions:

- Two dashes:

  ```
  -- These are comments
  ```
- Two forward slashes:

  ```
  // These are comments
  ```

Data integration tasks ignore all text on a line preceded by comment specifiers. For example, to concatenate two strings, enter the following expression with comments in the middle of the expression:

```
 -- This expression concatenates first and last names for customers:
FIRST_NAME -- First names from the CUST table
|| // Concat symbol
LAST_NAME // Last names from the CUST table
// Joe Smith Aug 18 1998
```

Data integration tasks ignore the comments and evaluates the expression as follows:

```
FIRST_NAME || LAST_NAME
```

You cannot continue a comment to a new line:

```
-- This expression concatenates first and last names for customers:
FIRST_NAME -- First names from the CUST table
|| // Concat symbol
LAST_NAME // Last names from the CUST table
Joe Smith Aug 18 1998
```

In this case, data integration tasks do not validate the expression because the last line is not a valid expression.

Reserved words

Some keywords, such as constants, operators, and system variables, are reserved for specific functions. These include:

- :EXT
- :INFA
- :LKP
- :MCR
- :SD
- :SEQ
- :SP
- :TD
- AND
- DD_DELETE
- DD_INSERT
- DD_REJECT
- DD_UPDATE
- FALSE
- NOT
- NULL
- OR
- PROC_RESULT
- SPOUTPUT
- TRUE
- WORKFLOWSTARTTIME

The following words are reserved for Informatica Intelligent Cloud Services:

- ABORTED
- DISABLED
- FAILED
- NOTSTARTED
- STARTED

- STOPPED
- SUCCEEDED

Note: You cannot use a reserved word to name a field. Reserved words have predefined meanings in expressions.

Advanced session properties

Advanced session properties are optional properties that you can configure in mapping tasks. Use caution when you configure advanced session properties. The properties might not be appropriate for use with all tasks.

You can configure the following types of advanced session properties:

- General
- Performance
- Advanced
- Error handling

Note: Mappings in SQL ELT mode don't use advanced session properties.

General options

The following table describes the general options:

General options	Description
Write Backward Compatible Session Log File	Writes the session log to a file.
Session Log File Name	Name for the session log. Use any valid file name. You can customize the session log file name in one of the following ways: - Using a static name. A static log file name is a simple static string with or without a file extension. If you use a static name, the log file name is appended with a sequence number each time the task runs, for example samplelog.1, samplelog.2. When the maximum number of log files is reached, the numbering sequence begins a new cycle. - Using a dynamic name. A log file name is dynamic when it includes a parameter defined in a parameter file or a system variable. You can include any of the following system variables: - $CurrentTaskName. Replaced with the task name. - $CurrentTaskId. Replaced with the task ID. - $CurrentTime. Replaced with the current time. - $CurrentRunId. Replaced with the run ID for the current job. If you use a dynamic name, the file name is unique for every task run. The Maximum Number of Log Files property is not applied. To purge old log files, delete the files manually.
Session Log File Directory	Directory where the session log is saved. Use a directory local to the Secure Agent to run the task. By default, the session log is saved to the following directory: `<Secure Agent installation directory>/apps/Data_Integration_Server/logs`

General options	Description
$Source Connection Value	Source connection name.
$Target Connection Value	Target connection name.
Source File Directory	Source file directory path. Use for flat file connections only.
Target File Directory	Target file directory path. Use for flat file connections only.
Treat Source Rows as	When the task reads source data, it marks each row with an indicator that specifies the target operation to perform when the row reaches the target. Use one of the following options: - Insert. All rows are marked for insert into the target. - Update. All rows are marked for update in the target. - Delete. All rows are marked for delete from the target. - Data Driven. The task uses the Update Strategy object in the data flow to mark the operation for each source row.
Commit Type	Commit type to use. Use one of the following options. - Source. The task performs commits based on the number of source rows. - Target. The task performs commits based on the number of target rows. When you do not configure a commit type, the task performs a target commit.
Commit Interval	Interval in rows between commits. When you do not configure a commit interval, the task commits every 10,000 rows.
Commit on End of File	Commits data at the end of the file.
Rollback Transactions on Errors	Rolls back the transaction at the next commit point when the task encounters a non-fatal error. When the task encounters a transformation error, it rolls back the transaction if the error occurs after the effective transaction generator for the target.
Java Classpath	This option is not used.

Performance settings

The following table describes the performance settings:

Performance settings	**Description**
DTM Buffer Size	Amount of memory allocated to the task from the DTM process. By default, a minimum of 12 MB is allocated to the buffer at run time. Use one of the following options: - Auto. Enter Auto to use automatic memory settings. When you use Auto, configure **Maximum Memory Allowed for Auto Memory Attributes**. - A numeric value. Enter the numeric value that you want to use. The default unit of measure is bytes. Append KB, MB, or GB to the value to specify a different unit of measure. For example, 512MB. You might increase the DTM buffer size in the following circumstances: - When a task contains large amounts of character data, increase the DTM buffer size to 24 MB. - When a source contains a large binary object with a precision larger than the allocated DTM buffer size, increase the DTM buffer size so that the task does not fail.
Incremental Aggregation	Performs incremental aggregation for tasks.
Reinitialize Aggregate Cache	This option is not used.
Enable High Precision	Processes the Decimal data type to a precision of 28.
Session Retry on Deadlock	The task retries a write on the target when a deadlock occurs.
SQL ELT Optimization	Type of SQL ELT optimization. Use one of the following options: - None. The task processes all transformation logic for the task. - To Source. The task pushes as much of the transformation logic to the source database as possible. - To Target. The task pushes as much of the transformation logic to the target database as possible. - Full. The task pushes as much of the transformation logic to the source and target databases as possible. The task processes any transformation logic that it cannot push to a database. - $$PushdownConfig. The task uses the SQL ELT optimization type specified in the user-defined parameter file for the task. When you use $$PushdownConfig, ensure that the user-defined parameter is configured in the parameter file. When you use SQL ELT optimization, do not use the **Error Log Type** property. For more information, see the help for the appropriate connector. The SQL ELT optimization functionality varies depending on the support available for the connector. For more information, see the help for the appropriate connector.
Create Temporary View	Allows the task to create temporary view objects in the database when it pushes the task to the database. Use when the task includes an SQL override in the Source Qualifier transformation or Lookup transformation.
Create Temporary Sequence	This option is not used.

Performance settings	Description
Enable cross-schema SQL ELT optimization	Enables SQL ELT optimization for tasks that use source or target objects associated with different schemas within the same database. To see if cross-schema SQL ELT optimization is applicable to the connector you use, see the help for the relevant connector. This property is enabled by default.
Allow SQL ELT Optimization for User Incompatible Connections	Indicates that the database user of the active database has read permission on idle databases. If you indicate that the database user of the active database has read permission on idle databases, and it does not, the task fails. If you do not indicate that the database user of the active database has read permission on idle databases, the task does not push transformation logic to the idle databases.
Session Sort Order	Order to use to sort character data for the task.

Advanced options

The following table describes the advanced options:

Advanced options	Description
Constraint Based Load Ordering	This option is not used.
Cache Lookup() Function	This option is not used.
Default Buffer Block Size	Size of buffer blocks used to move data and index caches from sources to targets. By default, the task determines this value at run time. Use one of the following options: - Auto. Enter Auto to use automatic memory settings. When you use Auto, configure Maximum Memory Allowed for Auto Memory Attributes. - A numeric value. Enter the numeric value that you want to use. The default unit of measure is bytes. Append KB, MB, or GB to the value to specify a different unit of measure. For example, 512MB. The task must have enough buffer blocks to initialize. The minimum number of buffer blocks must be greater than the total number of Source Qualifiers, Normalizers for COBOL sources, and targets. The number of buffer blocks in a task = DTM Buffer Size / Buffer Block Size. Default settings create enough buffer blocks for 83 sources and targets. If the task contains more than 83, you might need to increase DTM Buffer Size or decrease Default Buffer Block Size.
Line Sequential Buffer Length	Number of bytes that the task reads for each row. Data Integration dynamically increases the maximum line sequential buffer length from the default of 1024 bytes.

Advanced options	Description
Maximum Memory Allowed for Auto Memory Attributes	Maximum memory allocated for automatic cache when you configure the task to determine the cache size at run time. You enable automatic memory settings by configuring a value for this attribute. Enter a numeric value. The default unit is bytes. Append KB, MB, or GB to the value to specify a different unit of measure. For example, 512MB. If the value is set to zero, the task uses default values for memory attributes that you set to auto.
Maximum Percentage of Total Memory Allowed for Auto Memory Attributes	Maximum percentage of memory allocated for automatic cache when you configure the task to determine the cache size at run time. If the value is set to zero, the task uses default values for memory attributes that you set to auto.
Additional Concurrent Pipelines for Lookup Cache Creation	Restricts the number of pipelines that the task can create concurrently to pre-build lookup caches. You can configure this property when the **Pre-build Lookup Cache** property is enabled for a task or transformation. When the **Pre-build Lookup Cache** property is enabled, the task creates a lookup cache before the Lookup receives the data. If the task has multiple Lookups, the task creates an additional pipeline for each lookup cache that it builds. To configure the number of pipelines that the task can create concurrently, select one of the following options: - Auto. The task determines the number of pipelines it can create at run time. - Numeric value. The task can create the specified number of pipelines to create lookup caches.
Custom Properties	Configure custom properties for the task. You can override the custom properties that the task uses after the job has started. The task also writes the override value of the property to the session log.
Pre-build Lookup Cache	Allows the task to build the lookup cache before the Lookup receives the data. The task can build multiple lookup cache files at the same time to improve performance. Configure one of the following options: - Always allowed. The task can build the lookup cache before the Lookup receives the first source row. The task creates an additional pipeline to build the cache. - Always disallowed. The task cannot build the lookup cache before the Lookup receives the first row. When you use this option, configure the **Configure the Additional Concurrent Pipelines for Lookup Cache Creation** property. The task can pre-build the lookup cache if this property is greater than zero.
DateTime Format String	Date time format for the task. You can specify seconds, milliseconds, or nanoseconds. To specify seconds, enter `MM/DD/YYYY HH24:MI:SS`. To specify milliseconds, enter `MM/DD/YYYY HH24:MI:SS.MS`. To specify microseconds, enter `MM/DD/YYYY HH24:MI:SS.US`. To specify nanoseconds, enter `MM/DD/YYYY HH24:MI:SS.NS`. By default, the format specifies microseconds, as follows: MM/DD/YYYY HH24:MI:SS.US.
Pre 85 Timestamp Compatibility	This option is not used.

Error handling

The following table describes the error handling options:

Error handling options	Description
Stop on Errors	The number of non-fatal errors the task can encounter before it stops the job. Non-fatal errors include reader, writer, and transformation errors. Enter the number of non-fatal errors you want to allow before stopping the job. The task maintains an independent error count for each source, target, and transformation. If you specify 0, non-fatal errors do not cause the job to stop. Optionally, you can use the $PMSessionErrorThreshold service process variable to set this threshold. Configure this variable as a DTM custom property for the Data Integration Server. You can override the value in a parameter file. For more information, see the following KB article: HOW TO: Set the session error threshold for a mapping task using $PMSessionErrorThreshold in CDI.
Override Tracing	Overrides tracing levels set on an object level.
On Stored Procedure Error	This option is not used.
On Pre-Session Command Task Error	Determines the behavior when a task that includes pre-session shell commands encounters errors. Use one of the following options: - Stop Session. The task stops when errors occur while executing pre-session shell commands. - Continue Session. The task continues regardless of errors. By default, the task stops.
On Pre-Post SQL Error	Determines the behavior when a task that includes pre-session or post-session SQL encounters errors: - Stop Session. The task stops when errors occur while executing pre-session or post-session SQL. - Continue. The task continues regardless of errors. By default, the task stops.
Error Log Type	Specifies the type of error log to create. You can specify flat file or no log. Default is none. You cannot log row errors from XML file sources. You can view the XML source errors in the session log. Do not use this property when you use the **SQL ELT Optimization** property.
Error Log File Directory	Specifies the directory where errors are logged. By default, the error log file directory is $PMBadFilesDir\.
Error Log File Name	Specifies error log file name. By default, the error log file name is PMError.log.
Log Row Data	Specifies whether or not to log transformation row data. When you enable error logging, the task logs transformation row data by default. If you disable this property, n/a or -1 appears in transformation row data fields.
Log Source Row Data	Specifies whether or not to log source row data. By default, the check box is clear and source row data is not logged.
Data Column Delimiter	Delimiter for string type source row data and transformation group row data. By default, the task uses a pipe (|) delimiter. **Tip:** Verify that you do not use the same delimiter for the row data as the error logging columns. If you use the same delimiter, you may find it difficult to read the error log file.

Parameter files

A parameter file is a list of user-defined parameters and their associated values.

Use a parameter file to define values that you want to update without having to edit the task. You update the values in the parameter file instead of updating values in a task. The parameter values are applied when the task runs.

You can use a parameter file to define parameter values in mapping tasks.

Define parameter values for connections in the following transformations:

- Source
- Target
- Lookup
- SQL

Define parameter values for objects in the following transformations:

- Source
- Target
- Lookup

Also, define values for parameters in data filters, expressions, and lookup expressions.

Note: Not all connectors support parameter files. To see if a connector supports runtime override of connections and data objects, see the help for the appropriate connector.

You enter the parameter file name and location when you configure the task.

You can't use a parameter file in a mapping task that is based on a mapping in SQL ELT mode.

Schedules

You can run tasks manually or you can use schedules to run them at a specific time or interval such as hourly, daily, or weekly.

To use a schedule, you associate the task with a schedule when you configure the task. You can use an existing schedule or create a new schedule. If you want to create a schedule, you can create the schedule from the task's **Schedule** or **Runtime Options** page during task configuration.

When you create a schedule, you specify the date and time. You can configure a schedule to run associated assets throughout the day between 12:00 a.m. and 11:55 p.m. Informatica Intelligent Cloud Services might add a small schedule offset to the start time, end time, and all other time configurations. As a result, scheduled tasks and taskflows might start later than expected. For example, you configure a schedule to run hourly until noon, and the schedule offset for your organization is 10 seconds. Informatica Intelligent Cloud Services extends the end time for the schedule to 12:00:10 p.m., and the last hourly task or taskflow starts at 12:00:10 p.m. To see the schedule offset for your organization, check the **Schedule Offset** organization property.

You can monitor scheduled tasks from the **All Jobs** page in Monitor. Scheduled tasks do not appear on the **My Jobs** page.

When you copy a task that includes a schedule, the schedule is not associated with the new task. To associate a schedule with the new task, edit the task.

If you remove a task from a schedule as the task runs, the job completes. Data Integration cancels any additional runs associated with the schedule.

Repeat frequency

The repeat frequency determines how often tasks run. You can set the repeat frequency to every N minutes, hourly, daily, weekly, biweekly, or monthly.

The following table describes the repeat frequency options:

Option	Description
Does not repeat	Tasks run as scheduled and do not repeat.
Every N minutes	Tasks run on an interval based on a specified number of minutes. You can configure the following options: - Repeat frequency. Select a frequency in minutes. Options are 5, 10, 15, 20, 30, 45. - Days. Days of the week when you want tasks to run. You can select one or more days of the week. - Time range. Hours of the day when you want tasks to start. Select All Day or configure a time range. You can configure a time range between 00:00-23:55. - Repeat option. The range of days when you want tasks to run. You can select Repeat Indefinitely or configure an end date and time.
Hourly	Tasks run on an hourly interval based on the start time of the schedule. You can configure the following options: - Repeat frequency. Select a frequency in hours. Options are 1, 2, 3, 4, 6, 8, 12. - Days. Days of the week when you want tasks to run. You can select one or more days of the week. - Time range. Hours of the day when you want tasks to start. Select All Day or configure a time range. You can configure a time range between 00:00-23:55. - Repeat option. The range of days when you want tasks to run. You can select Repeat Indefinitely or configure an end date and time.
Daily	Tasks run daily at the start time configured for the schedule. You can configure the following options: - Repeat frequency. The frequency at which you want tasks to run. Select Every Day or Every Weekday. - Repeat option. The range of days when you want tasks to run. You can select Repeat Indefinitely or configure an end date and time.
Weekly	Tasks run on a weekly interval based on the start time of the schedule. You can configure the following options: - Days. Days of the week when you want tasks to run. You can select one or more days of the week. - Repeat option. The range of days when you want tasks to run. You can select Repeat Indefinitely or configure an end date and time. If you do not specify a day, the schedule runs regularly on the same day of the week as the start date.

Option	Description
Biweekly	Tasks run every two weeks based on the start time of the schedule. You can configure the following options: - Days. Days of the week when you want tasks to run. You can select one or more days of the week. You must select at least one day. - Repeat option. The range of days when you want tasks to run. You can select Repeat Indefinitely or configure an end date and time. If you configure a biweekly schedule to start at 5 p.m. on a Tuesday and run tasks every two weeks on Mondays, the schedule begins running tasks on the following Monday.
Monthly	Tasks run on a monthly interval based on the start time of the schedule. You can configure the following options: - Day. Day of the month when you want tasks to run. You can configure one of the following options: - Select the exact date of the month, between 1-28. If you want the task to run on days later in the month, use the <n> <day of the week> option. - Select the <n> <day of the week>. Options for <n> include First, Second, Third, Fourth, and Last. Options for <day of the week> includes Day, and Sunday-Saturday. Tip: With the Day option, you can configure tasks to run on the First Day or the Last Day of the month. - Repeat option. The range of days when you want tasks to run. You can select Repeat Indefinitely or configure an end date and time.

Time zones and schedules

Informatica Intelligent Cloud Services stores time in Coordinated Universal Time (UTC). When you log in, Informatica Intelligent Cloud Services converts the time and displays it in the time zone associated with your user profile.

When you create a schedule, you select the time zone for the scheduler to use. You can select a time zone that is different from your time zone or your organization time zone.

The time zone for system-configured schedules is UTC.

Daylight Savings Time changes and schedules

Informatica Intelligent Cloud Services applies Daylight Savings Time changes to all tasks except biweekly tasks.

When Daylight Savings time goes into effect, tasks scheduled to run between 2:00 a.m. and 2:59 a.m., do not run the day that the time changes from 2:00 a.m. to 3:00 a.m. If a task is scheduled to run biweekly at 2 a.m., it will run at 3 a.m. the day of the time change and at 2 a.m. for the next run.

Daylight Savings Time does not trigger additional runs for tasks that are scheduled to run between 1:00 a.m. - 1:59 a.m. when Standard Time begins. For example, a task is scheduled to run every day at 1:30 a.m. When the time changes from 2 a.m. to 1 a.m., the task does not run again at 1:30 a.m.

Tip: To ensure that Informatica Intelligent Cloud Services does not skip any scheduled runs near the 2 a.m. time change, do not schedule jobs to run between 12:59 a.m. and 3:01 a.m.

Creating a schedule

You can create a schedule in Data Integration when you configure a task. You can also create a schedule in Administrator.

The following procedure describes how to create a schedule when you access the **Schedule** page from Data Integration during task configuration.

1. Select **Run this task on a schedule**, and then click **New**.
2. Configure the following properties:

Property	Description
Schedule Name	Name of the schedule. Each schedule name must be unique within the organization. Schedule names can contain alphanumeric characters, spaces, and the following special characters: _ . + - Maximum length is 100 characters. Schedule names are not case sensitive.
Description	Description of the schedule. Maximum length is 255 characters.
Starts	Date and time when the schedule takes effect. The date format is MM/DD/YYYY. Time appears in the 24-hour format. Click the calendar button to select the start date. The start date and time can affect the repeat frequency for tasks and taskflow jobs that repeat at regular intervals. For example, if the start date is November 10 and the repeat frequency is monthly, the schedule runs associated assets on the tenth day of each month. If the start time is 3:10 and the repeat frequency is hourly, the assets run every hour at 10 minutes past the hour. Default is the current date, current time, and time zone of the user that creates the schedule.
Time Zone	Select the time zone for the schedule to use. The time zone can differ from the organization time zone or user time zone.
Repeats	Repeat frequency for the schedule. Select one of the following options: - Does Not Repeat - Every N Minutes - Hourly - Daily - Weekly - Monthly Default is Does Not Repeat.

3. Click **Save** to save the schedule and return to the task configuration page.

Running a task on a schedule

Associate a task with a schedule on the **Schedule** or **Runtime Options** page when you configure the task. You can use an existing schedule or create a schedule.

1. Select **Run this task on a schedule**.
2. To specify whether to use an existing schedule or a new schedule, perform one of the following tasks:
   - To use an existing schedule, select the schedule that you want to use.

- To create a schedule to use for the task, click **New**, and then configure the schedule properties.

3. Click **Save**.

Email notification

You can configure email notification for a task. When you configure custom email notification, Data Integration uses the custom email notification instead of the email notification options configured for the organization.

To configure email notification options, perform the following steps in the task wizard:

1. Specify whether to use the default email notification options that have been set for your organization or create custom email notification for the task. Configure email notification using the following options:

Field	Description
Use Default Email Notification Options for my Organization	Use the email notification options configured for the organization.
Use Custom Email Notification Options for this Task	Use the email notification options configured for the task. You can send email to different addresses based on whether the task failed, completed with errors, or completed successfully. Use commas to separate a list of email addresses. When you select this option, email notification options configured for the organization are not used.

2. Click **Save**.

Preprocessing and postprocessing commands

You can run preprocessing and postprocessing commands to perform additional jobs. The task runs preprocessing commands before it reads the source. It runs postprocessing commands after it writes to the target.

You can use the following types of commands:

- SQL commands. Use SQL commands to perform database tasks.
- Operating system commands. Use shell and DOS commands to perform operating system tasks.

If any command in the preprocessing or postprocessing scripts fail, the task fails.

Preprocessing and postprocessing SQL commands

You can run SQL commands before or after a task. For example, you can use SQL commands to drop indexes on the target before the task runs, and then recreate them when the task completes. Data Integration does not validate the SQL.

Use the following rules and guidelines when creating the SQL commands:

- Use any command that is valid for the database type. However, Data Integration does not allow nested comments, even if the database allows them.
- Use a semicolon (;) to separate multiple statements. Data Integration issues a commit after each statement.
- Data Integration ignores semicolons within comments. If you need to use a semicolon outside of comments, you can escape it with a backslash (\).

Preprocessing and postprocessing operating system commands

Data Integration can perform operating system commands before or after the task runs. For example, use a preprocessing shell command to archive a copy of the target flat file before the task runs on a UNIX machine.

You can use the following types of operating system commands:

- UNIX. Any valid UNIX command or shell script.
- Windows. Any valid DOS or batch file.

Enter multiple preprocessing or postprocessing commands as a single line without spaces.

If the Secure Agent is on a Windows machine, separate commands with an ampersand (&). If the Secure Agent is on a Linux machine, separate commands with a semicolon (;).

Monitoring a job

You can monitor tasks or taskflows that are currently running, have completed, or have stopped.

Monitor jobs on the following pages:

- Monitor the jobs that you initiated on the **My Jobs** page in Data Integration.
- Monitor running jobs in your organization on the **Running Jobs** page in Monitor.
- Monitor all jobs in your organization on the **All Jobs** page in Monitor.

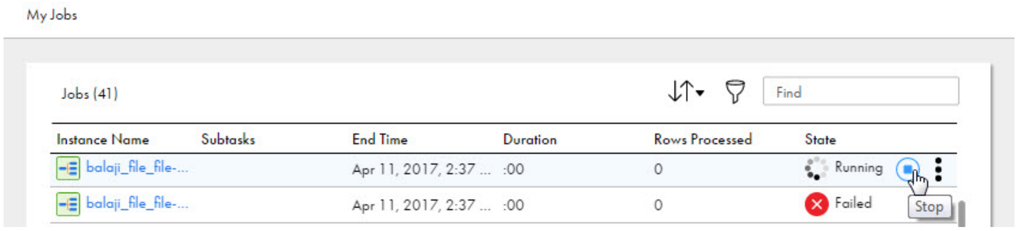
For more information about monitoring jobs, see *Monitor*.

Stopping a job

A job is an instance of a mapping, task, or taskflow. You can stop a running job on the **All Jobs**, **Running Jobs**, or **My Jobs** page.

1. Open Monitor and select **All Jobs** or **Running Jobs**, or open Data Integration and select **My Jobs**.

2. In the row that contains the job that you want to stop, click the **Stop** icon.



My Jobs

Jobs (41)

Instance Name	Subtasks	End Time	Duration	Rows Processed	State
balaji_file_file-...		Apr 11, 2017, 2:37 ...	:00	0	Running
balaji_file_file-...		Apr 11, 2017, 2:37 ...	:00	0	Failed

To view details about the stopped job, click the job name.

Guidelines for sources and targets in data integration tasks

Use the following rules and guidelines for sources and targets in data integration tasks:

- Field names must contain 65 characters or less.
- Field names must contain only alphanumeric or underscore characters. Spaces are not allowed.
- Field names cannot start with a number.
- Each field name must be unique within each source and target object.
- Data Integration truncates data if the scale or precision of a numeric target column is less than the scale or precision of the corresponding source column.

Rules and guidelines for flat file sources and targets

Use the following rules and guidelines for flat file sources and targets:

- All date columns in a flat file source must have the same date format.
- The flat file cannot contain empty column names. If a file contains an empty column name, the following error appears:

  ```
  Invalid header line: Empty column name found.
  ```
- Column names in a flat file must contain printable tab or ASCII characters (ASCII code 32-126). If the file contains a character that is not valid, the following error appears:

  ```
  Invalid header line: Non-printable character found. The file might be binary or might
  have invalid characters in the header line.
  ```
- You can use a tab, space, or any printable special character as a delimiter. The delimiter can have a maximum of 10 characters. The delimiter must be different from the escape character and text qualifier.
- For flat file sources and targets with multibyte data on Linux, the default locale must be UTF-8.

Rules and guidelines for database sources and targets

Use the following rules and guidelines for database sources and targets:

- You can use database tables as targets. You can use database tables, aliases, and views as sources.
- Relational targets must meet the minimum system requirements.

- The database user account for each database target connection must have DELETE, INSERT, SELECT, and UPDATE privileges.

CHAPTER 2

Mapping tasks

Use the mapping task to process data based on the data flow logic defined in a mapping.

When you create a mapping task, you select the mapping for the task to use. The mapping must already exist before you can create a mapping task for it. Alternatively, you can create a mapping task using a template.

If the mapping includes parameters, you can define the parameters when you configure the task or define the parameters when you run the task. You can use user-defined parameters for data filters, expressions, and lookup expressions in a mapping task. You define user-defined parameters in a parameter file associated with the task.

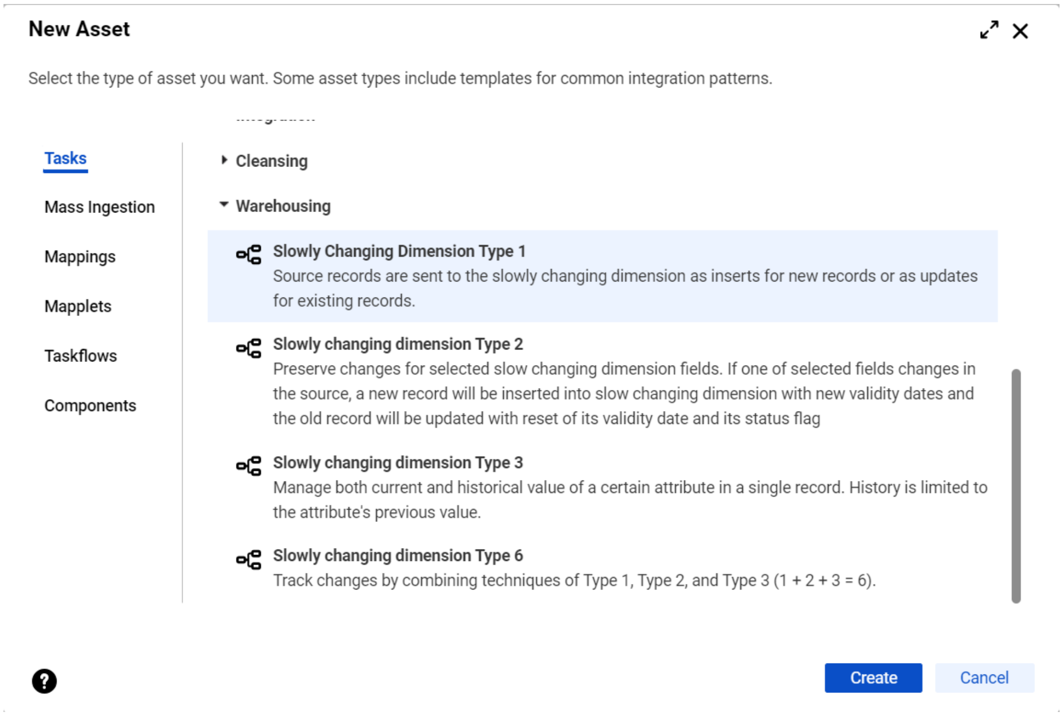
At run time, a mapping task processes task data based on the data flow logic from the mapping, the parameters defined in the task, and the user-defined parameters defined in a parameter file, when available.

Mapping task templates

Use a mapping task template to run a mapping task without creating a mapping beforehand.

Each mapping task template is based upon a mapping template. Use a mapping task template when the mapping on which the mapping task template is based suits your needs. When you select a mapping task template, Data Integration creates a copy of the template for you to use. When you define the mapping task in the task wizard, you save a copy of the mapping template on which the mapping task template is based.

Templates are divided into three categories: Integration, Cleansing, and Warehousing, as shown in the following image:



New Asset

Select the type of asset you want. Some asset types include templates for common integration patterns.

Slowly Changing Dimension Type 1
Source records are sent to the slowly changing dimension as inserts for new records or as updates for existing records.

Slowly changing dimension Type 2
Preserve changes for selected slow changing dimension fields. If one of selected fields changes in the source, a new record will be inserted into slow changing dimension with new validity dates and the old record will be updated with reset of its validity date and its status flag

Slowly changing dimension Type 3
Manage both current and historical value of a certain attribute in a single record. History is limited to the attribute's previous value.

Slowly changing dimension Type 6
Track changes by combining techniques of Type 1, Type 2, and Type 3 (1 + 2 + 3 = 6).

The templates range from simple templates that you can use to copy data from one source to another, to complex templates that you can use for data warehousing-related tasks.

Related objects

When a mapping includes a source that is a parameter and is configured for multiple objects, you can join related objects in the task.

You can join related objects based on existing relationships or custom relationships. Data Integration restricts the type of relationships that you can create based on the connection type.

Use the following relationships to join related objects:

Existing relationships

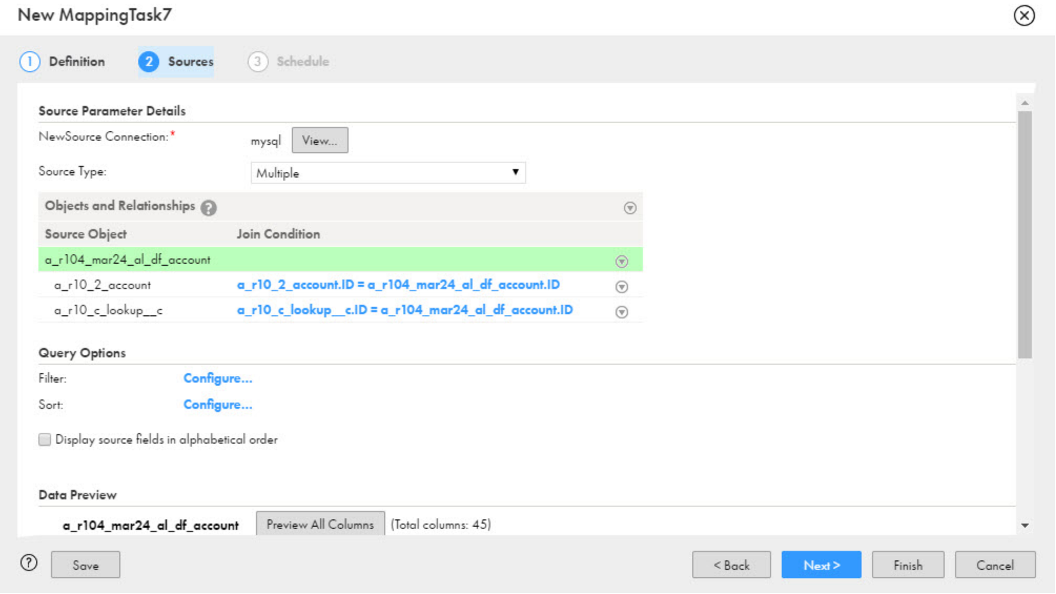
> You can use relationships defined in the source system to join related objects. You can join objects with existing relationships for Salesforce, database, and some Data Integration Connectors connection types.
>
> After you select a primary object, you select a related object from a list of related objects.

Custom relationships

> You can use custom relationships to join multiple source objects. You can create custom relationships for the database connection type.
>
> When you create a custom relationship for database objects, you create an inner, left outer, or right outer join on the source fields that you select.

To join source objects, you add the primary source object in the **Objects and Relationships** table. Then you add related objects, specify keys for the primary and related objects, and configure the join type and operator.

For more information about related source objects, see the Source Transformation section in *Transformations*.



Source Parameter Details

NewSource Connection:* mysql

Source Type: Multiple

Objects and Relationships

Source Object	Join Condition
a_r104_mar24_al_df_account	
a_r10_2_account	a_r10_2_account.ID = a_r104_mar24_al_df_account.ID
a_r10_c_lookup__c	a_r10_c_lookup__c.ID = a_r104_mar24_al_df_account.ID

Query Options

Filter: Configure...

Sort: Configure...

□ Display source fields in alphabetical order

Data Preview

a_r104_mar24_al_df_account (Total columns: 45)

Advanced relationships

You can create an advanced relationship for database sources when the source object in the mapping is a parameter and configured for multiple sources. You cannot create an advanced relationship between source objects that have been joined using a custom relationship.

When you create an advanced relationship, the wizard converts any relationships that you defined to an SQL statement that you can edit.

To create an advanced relationship, you add the primary source object in the Objects and Relationships table. Then you select fields and write the SQL statement that you want to use. Use an SQL statement that is valid for the source database. You can also add additional objects from the source.

SQL ELT optimization

SQL ELT optimization pushes some of the transformation logic in a mapping to source or target databases for execution, which can improve task performance. By default, SQL ELT optimization is enabled for all mapping tasks.

When you run a task configured for SQL ELT optimization, the task converts the transformation logic to an SQL query. The task sends the query to the database, and the database executes the query.

The amount of transformation logic that you can push to the database depends on the database, transformation logic, and task configuration. The task processes all transformation logic that it cannot push to a database.

Use the **SQL ELT Optimization** advanced session properties to configure SQL ELT optimization for a task. You can configure the SQL ELT optimization type, how to run the task if SQL ELT optimization is not possible, whether to allow the task to create temporary views in the database, and other options.

SQL ELT Optimization advanced session properties aren't displayed for mapping tasks that are based on mappings in SQL ELT mode because mappings in SQL ELT mode are automatically configured to push transformation logic to the cloud data warehouse.

Note: SQL ELT optimization functionality varies depending on the support available for the connector. For more information, see the help for the appropriate connector.

Simultaneous task runs

You can run multiple instances of a mapping task at the same time.

You might want to enable simultaneous task runs to load target files concurrently.

You can use multiple task instances in a Parallel Paths step of a taskflow or in two different taskflows that run in parallel.

To enable simultaneous task runs, select the **Allow the mapping task to be executed simultaneously** on the **Runtime Options** page when you configure the task.

Use caution when you configure mapping tasks to run simultaneously. Mapping features that change each time the task runs, such as in-out parameters and sequence generator values, might produce unexpected results when you run the task instances simultaneously.

Schema change handling

You can choose how Data Integration handles changes that you make to the schema of some data object types.

By default, if you make changes to the schema, Data Integration does not pick up the changes automatically. If you want Data Integration to refresh the data object schema every time the mapping task runs, you can enable dynamic schema handling.

A schema change includes one or more of the following changes to the data object:

- Fields are added.
- Fields are deleted.
- Fields are renamed.
- Field data type, precision, or scale is updated.

Data Integration automatically refreshes the schema for relational objects every time the task runs. If you want to dynamically refresh the schema for other object types, enable dynamic schema change handling in the **Advanced Options** area of the **Runtime Options** page when you configure the task.

The following table describes the schema change handling options:

Option	Description
Asynchronous	Default. Data Integration refreshes the schema when you edit the mapping or mapping task, and when Informatica Intelligent Cloud Services is upgraded.
Dynamic	Data Integration refreshes the schema every time the task runs. Applicable for source, target, and lookup objects of certain connector types. For some connector types, Data Integration can only refresh the schema if the data object is a flat file. If you select this option, the file object format must be delimited. Not applicable to hierarchical data. To see if a connector supports dynamic schema change handling, see the help for the appropriate connector.

If you update fields in the source object and you enable dynamic schema handling, be sure to update the Target transformation field mapping. Data Integration writes Null to the target fields that were previously mapped to the renamed or deleted source fields. If you use a target created at run time, update the target object name so that Data Integration creates a new target when the task runs. The task fails if Data Integration tries to alter a target created in a previous task run.

Dynamic schema handling options

When you enable dynamic schema change handling, you can select how Data Integration applies schema changes from upstream transformations to the target object. If the mapping contains more than one target, select the schema change handling for each target.

To select target schema options, the target field mapping must be automatic.

When you configure target schema options for objects that are created at runtime, Data Integration creates the target the first time you run the task. In subsequent task runs, Data Integration updates the target based on the schema change option that you select.

The schema change handling options available are based on the target connection. To see if a connector supports dynamic schema change handling, see the help for the appropriate connector.

The following table describes the schema handling options that you can select for each target type:

Schema handling option	Target type	Description
Keep Existing File Format	File	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	Database and file	For database targets, Data Integration drops the existing target table and creates a new target table with the schema from the upstream transformations on every run. For file targets, Data Integration updates the target schema to match the incoming schema on every task run.

Schema handling option	Target type	Description
Alter and Apply Changes	Database	Data Integration updates the target schema with additive changes to match the schema from the upstream transformations. It does not delete columns from the target.
Don't Apply DDL Changes	Database	Data Integration fetches the target schema at runtime and does not apply upstream schema changes to the target table.

Data Integration does not pass field constraints to the target. For example, the source contains fields S1 and S2 configured with the NOT NULL constraint. The target contains fields T1 and T2 also configured with the NOT NULL constraint. You select the **Alter and Apply Changes** schema handling option. When you run the task, fields S1 and S2 are written to the target with no constraints.

Dynamic schema change handling rules and guidelines

Enable dynamic schema change handling so that Data Integration refreshes the data object schema every time the mapping task runs.

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

- Changes to the object schema take precedence over changes to the field metadata in the mapping. For example, you add a field to the source object and then edit the metadata of an existing field in the mapping. At run time, Data Integration adds the new field and does not edit the existing field.
- Data Integration resolves parameters before picking up the object schema.
- Data Integration treats renamed fields as deleted and added columns. If you rename a field, you might need to update transformations that reference the renamed field. For example, if you rename a field that is used in the lookup condition, the lookup cannot find the new field and the task fails.
- When you rename, add, or delete fields, you might need to update the field mapping. For example, if you delete all the previously mapped fields in a target object, you must remap at least one field or the task fails.
- Data Integration writes Null values to a target field in the following situations:
  - You rename a target field with automatic field mapping, and the field name does not match a source field.
  - You rename a source field with manual field mapping, and you do not remap the field to the target.
- If you delete a field from a source or lookup object and a downstream transformation references the field, the task fails.
- If you change a source or lookup field type, the task might fail if the new field type results in errors downstream. For example, if you change an integer field in an arithmetic expression to a string field, the expression is not valid and the task fails.
- If you change a target field type, Data Integration converts the data from the incoming field to the new target field type. If the conversion results in an error, Data Integration drops the row. For example if you change a string type to a date type where the string does not contain a date, Data Integration drops the row.

Schema mismatch handling

You can choose how Data Integration responds when it encounters a mismatch between a file and the selected schema.

When you run a mapping through a mapping task, you can configure an advanced option that specifies how Data Integration handles mismatched schemas:

- Skip mismatched files and continue. When Data Integration finds a schema mismatch, it stops searching for other errors in the same file and writes that error to the log. Data Integration doesn't process any other records from that file and continues with the next file.
- Stop on first mismatched file. Data Integration stops all processing when it finds a schema mismatch error, and it writes the error to the log. Data Integration does not roll backfiles processed before the error was found, and it doesn't process the file containing the error.

When you run a mapping in the Mapping Designer, Data Integration evaluates every file and skips the entire file when it encounters a schema mismatch.

Schema mismatch handling doesn't apply to mapping tasks that are based on mappings in SQL ELT mode.

Mapping task configuration

Complete the following steps to create a mapping task:

1. Create the mapping task.
2. Configure the source.
3. Configure the target.
4. Define input parameters.
5. Optionally, define persistent values.
6. Optionally, configure runtime options.

As you configure the task, you can click **Save** to save your work at any time.

Creating a mapping task

Define general properties of the task on the **General** page.

1. To create a mapping task, click **New** > **Tasks**, and then complete one of the following steps:
   - To create a mapping task based on a mapping, select **Mapping Task** and click **Create**.
   - To create a mapping task using a template, expand the appropriate template category, select the template you want to use, and then click **Create**.
2. To edit a mapping task, on the **Explore** page, navigate to the mapping task. In the row that contains the task, click **Actions** and select **Edit**.

3. Configure the following fields:

Field	Description
Name	Name of the task. Task names can contain alphanumeric characters, spaces, and the following special characters: _ . + - Maximum length is 100 characters. Task names are not case sensitive.
Location	Project or folder in which the task resides. If the **Explore** page is currently active and a project or folder is selected, the default location for the asset is the selected project or folder. Otherwise, the default location is the location of the most recently saved asset.
Description	Description of the task. Maximum length is 4000 characters.
Runtime Environment	Runtime environment that contains the Secure Agent to run the task.
Mapping	Mapping associated with the task. To select the mapping, click **Select** and navigate to the mapping that you want to use. To open the mapping, click **View**.

4. Click **Next**.

Configuring sources

Configure source and lookup source parameters on the **Sources** page. The **Sources** page displays differently based on the basis for the task. If the mapping does not include source or lookup parameters, the **Sources** page does not appear.

You can add a single source object or multiple source objects based on the connection type and mapping configuration.

If the mapping specifies a connection parameter, and after you create the mapping task you edit the mapping to change the source, you might need to edit the task. Check the specific connection in the mapping task to determine if you need to reset it, and then validate the task again.

1. For each source or lookup source parameter, configure the following details as required:

Parameter detail	Description
Connection	Select a connection. To create a connection, click **New**. To edit a connection, click **View**, and in the **View Connection** dialog box, click **Edit**. **Tip:** Hover over the source connection tool tip to view more information about the connection.
Source type	Source type. The options available depend on the connection type.

Parameter detail	Description
Object	Select an object or enter a query. If a list of objects doesn't appear, click **Select**. The **Select Source Object** dialog box displays up to 200 objects. If the object you want to use doesn't appear, enter a search string to reduce the number of objects. **Tip:** Hover over the source object tool tip to view more information about the object.
Display Technical Field Names Instead of Labels	Displays technical names instead of business names. Not available for all connection types.
Add Currently Processed File Name	Adds the source file name to each row. Data Integration adds the CurrentlyProcesedFileName field to the source at run time. Available for parameterized source objects with flat file connections.

2. Configure formatting options if required.

   The properties available vary based on the connection type. For information about a particular connector's properties, see the help for the appropriate connector.

3. Optionally, expand the **Data Preview** area to preview the data.

   To display fields in alphabetical order, click **Display fields in alphabetical order**. By default, fields appear in the order returned by the source system.
   To download the preview results, click **Download**.

4. For parameterized source objects, configure data filters if necessary.
5. For parameterized source objects, configure sort options if necessary.
6. Configure field metadata if required.

   You can configure field metadata for sources with certain connection types. To see if a connector supports field metadata configuration, see the help for the appropriate connector. You can't update field metadata in mapping tasks that are based on mappings in SQL ELT mode.

   To edit field metadata, expand the **Field Data Types** area and select the row that contains the field that you want to edit. Configure the following attributes:

Data type attribute	Description
Retain existing fields at runtime	When enabled, the task uses the field metadata that is configured in the task. If field metadata changes after a task is saved, Data Integration uses the updated field metadata. Typically, this is the desired behavior. However, if the task uses a flat file connection and you want to retain the metadata used at design time, enable this option.
Type	The data type of the field.

Data type attribute	Description
Precision	Total number of digits in a number. For example, the number 123.45 has a precision of 5. The precision must be greater than or equal to 1.
Scale	Number of digits to the right of the decimal point of a number. For example, the number 123.45 has a scale of 2. Scale must be greater than or equal to 0. The scale of a number must be less than its precision. The maximum scale for a numeric data type is 65535. Not editable for all data types.

7. Optionally, configure advanced options for the selected connection.

   The options available depend on the connection.
   When you edit advanced options, you can restore default values for individual properties or for all advanced properties. Default values are defined in the mapping that the task is based on.

   For information about a particular connector's properties, see the help for the appropriate connector.
8. Click **Next**.

Configuring targets

The **Targets** page displays differently depending on the basis for the task.

The **Targets** page displays when the mapping includes parameters for target connections or target objects. The properties that you need to specify are based on the type of parameter.

1. For each target parameter, configure the following details:

Target parameter detail	Description
Connection	Select a connection. To create a connection, click **New**. To edit a connection, click **View**, and in the **View Connection** dialog box, click **Edit**. **Tip:** Hover over the target connection tool tip to view more information about the connection.
Object	Select a target object. If a list of objects does not appear, click **Select**. The **Select Target Object** dialog box displays up to 200 objects. If the object you want to use does not display, enter a search string to reduce the number of objects that display. **Tip:** Hover over the target object tool tip to view more information about the object.
Operation	Target operation, either insert, update, upsert, delete, or data driven. Not editable for all connection types.

Target parameter detail	Description
Truncate target	Truncates the target object before inserting new rows. Applies to insert and data driven operations. Not available for all connection types.
Enable target bulk load	Uses the database bulk API to perform an insert operation. Use the bulk API to write large amounts of data to the database with a minimal number of API calls. Loading in bulk mode can improve performance, but it limits the ability to recover because no database logging occurs. Applies to insert operations. Not available for all connection types.
Display technical names instead of labels	Displays technical names instead of business names. Not available for all connection types.

2. Configure formatting options if required.

   The properties available vary based on the connection type. For information about a particular connector's properties, see the help for the appropriate connector.

3. Optionally, expand the **Data Preview** area to preview the data.

   To display fields in alphabetical order, click **Display mapplet fields in alphabetical order**. By default, fields appear in the order returned by the target system.
   To download the preview results, click **Download**.

4. Configure field metadata if required.

   You can configure field metadata for targets with certain connection types. To see if a connector supports field metadata configuration, see the help for the appropriate connector. You can't update field metadata in mapping tasks that are based on mappings in SQL ELT mode.

   To edit field metadata, expand the **Field Data Types** area and select the row that contains the field that you want to edit. Configure the following attributes:

Data type attribute	Description
Retain existing fields at runtime	When enabled, the task uses the field metadata that is configured in the task. If field metadata changes after a task is saved, Data Integration uses the updated field metadata. Typically, this is the desired behavior. However, if the task uses a flat file connection and you want to retain the metadata used at design time, enable this option.
Type	The data type of the field.

Data type attribute	Description
Precision	Total number of digits in a number. For example, the number 123.45 has a precision of 5. The precision must be greater than or equal to 1.
Scale	Number of digits to the right of the decimal point of a number. For example, the number 123.45 has a scale of 2. Scale must be greater than or equal to 0. The scale of a number must be less than its precision. The maximum scale for a numeric data type is 65535. Not editable for all data types.

5. Optionally, configure advanced options for the selected connection.

   The options available depend on the connection.
   When you edit advanced options, you can restore default values for individual properties or for all advanced properties. Default values are defined in the mapping that the task is based on.

   For information about a particular connector's properties, see the help for the appropriate connector.
6. Click **Next**.

Configuring input parameters

The **Input Parameters** page displays differently depending on the basis for the task. If the mapping does not contain input parameters, the page doesn't display.

The **Input Parameters** page displays input parameters such as filter and lookup conditions, field mappings, and expressions. It also displays stored procedure connections. It doesn't display source, target, or lookup connections and objects.

1. Configure the parameters that display.

   Depending on the mapping data flow, you might need to configure some parameters before you configure other parameters. For more information, see *Mappings*.

   Warning: If the mapping contains a filter parameter, the task might display as valid even if you haven't configured the filter value. You must configure all parameters before the task is valid and you can run it.
2. If necessary, configure stored procedure details.

   This displays when a stored procedure requires a connection. Select a connection or optionally create a new one.
3. If necessary, configure lookup conditions.

   This displays when the lookup condition is parameterized.
4. If necessary, configure mapplet details.

   This displays when a mapplet requires a connection and the mapplet is configured to display on this page.

   For some connection types, you can select **Display technical field names instead of labels** to display technical names instead of business names.

   To display fields in alphabetical order, click **Display mapplet fields in alphabetical order**.
5. If necessary, configure field mappings.

This displays when the field mapping is parameterized. Use the **Object** list to display fields from different objects.

Map fields in one of the following ways:

- To automatically map fields with the same name, click **Automap** > **Exact Field Name**. Or, to map fields with similar names, click **Automap** > **Smart Map**.
  You can undo all automapped field mappings by clicking **Smart Map** > **Undo Automap**. To unmap a single field, select the field to unmap and click **Actions** > **Unmap**.
- To manually map fields, select and drag the source fields to the applicable target fields.

6. Click **Next**.

Configuring persistent values

The **Persistent Values** page displays if the mapping contains persisted values such as in-out parameters or sequences.

1. For each sequence value, edit or reset the current value.

   Note: If the Sequence transformation is configured to reset the value for each run, editing the current value in the mapping task has no effect.
2. Configure in-out parameters as necessary.

   In-out parameters don't apply to mapping tasks that are based on mappings in SQL ELT mode.
3. Click **Next**.

Configuring runtime options

On the **Runtime Options** page, configure optional runtime options for the task. Expand each area to see the options you can configure.

1. Specify whether to run the task on a schedule or without a schedule.

   Choose one of the following options:
   - If you want to run the task on a schedule, click **Run on a schedule**. Select the schedule you want to use or click **New** to create a schedule.
   - If you want to run the task without a schedule, click **Do not run on a schedule**.
2. Configure email notification options for the task.
3. Optionally, in the **Advanced Options** area, configure the advanced options that are displayed for your connection:

Field	Description
Pre-Processing Commands	Commands to run before the task.
Post-Processing Commands	Commands to run after the task completes.

Field	Description
Maximum Number of Log Files	Number of session log files to retain. By default, Data Integration stores each type of log file for 10 runs before it overwrites the log files for new runs. **Note:** If a dollar sign ($) is present in a custom session log file name, for example, MyLog_$CurrentTime, the file name is dynamic. If you customize the session log file name using a dynamic name, this property doesn't apply. To purge old log files, delete the files manually.
Schema Change Handling	Determines how Data Integration picks up changes to the object schema. Select one of the following options: - Asynchronous. Data Integration refreshes the schema when you update the mapping or mapping task, and after an upgrade. - Dynamic. Data Integration refreshes the schema every time the task runs. Default is Asynchronous.
Dynamic Schema Handling	Determines how Data Integration applies schema changes from upstream transformations to the target object. Available when the schema change handling is dynamic and the field mapping is automatic. For each target, select how Data Integration updates the target schema. The options available are based on the target connection. For more information, see "Schema change handling" on page 34 or the help for the appropriate connector.

4. Optionally, to create a parameter file based on the parameters and default values specified in the mapping on which the task is based, click **Download Parameter File Template** within the parameter file location.

   For more information about parameter file templates, see *Mappings*.

5. Optionally, if the mapping task contains parameters, you can use parameter values from a parameter file. Choose one of the following options:
   - To use a parameter file on a local machine, select **Local**. Enter the following information:

Field	Description
Parameter File Directory	Path for the directory that contains the parameter file, excluding the parameter file name. The Secure Agent must be able to access the directory. You can use an absolute file path or a path relative to one of the following $PM system variables: - $PMRootDir - $PMSourceFileDir - $PMLookupFileDir - $PMCacheDir - $PMSessionLogDir - $PMExtProcDir - $PMTempDir By default, Data Integration uses the following parameter file directory: `<Secure Agent installation directory>/apps/Data_Integration_Server/data/userparameters`
Parameter File Name	Name of the file that contains the definitions and values of user-defined parameters used in the task. You can provide the file name or the relative path and file name in this field.

- To use a cloud-hosted file, select **Cloud Hosted**. Enter the following information about the file:

Field	Description
Connection	Connection where the parameter file is stored. You can use the following connection types: - Amazon S3 - Google Storage V2 - Azure Data Lake Store Gen2
Object	Name of the file that contains the definitions and values of user-defined parameters used in the task.

6. Choose whether to run the task in standard or verbose execution mode, if this option is visible.

   If you select verbose mode, the mapping generates additional data in the logs that you can use for troubleshooting. Select verbose execution mode only for troubleshooting purposes. Verbose execution mode impacts performance because of the amount of data it generates.

7. Optionally, configure SQL ELT optimization.

The following table describes the SQL ELT optimization properties:

Property	**Description**
SQL ELT Optimization Type	Type of SQL ELT optimization. Use one of the following options: - None. The task processes all transformation logic for the task. - To Source. The task pushes as much of the transformation logic to the source database as possible. - To Target. The task pushes as much of the transformation logic to the target database as possible. - Full. The task pushes all of the transformation logic to the source and target databases. The source and target schema must be the same. - $$PushdownConfig. The task uses the SQL ELT optimization type specified in the user-defined parameter file for the task. When you use $$PushdownConfig, ensure that the user-defined parameter is configured in the parameter file. When you use SQL ELT optimization, do not use the **Error Log Type** advanced session property. Default is Full. **Note:** SQL ELT optimization functionality varies depending on the support available for the connector. For more information, see the help for the appropriate connector. SQL ELT optimization doesn't apply to mapping tasks that are based on mappings in SQL ELT mode because mappings in SQL ELT mode are automatically configured to push transformation logic to the cloud data warehouse.
Optimization Context Type	Provides context about the mapping configuration for SQL ELT optimization. If you select an option other than None, Data Integration constructs a single query for SQL ELT optimization by combining multiple targets in the mapping based on the target configurations. If you select None, the query is not optimized. If Data Integration cannot apply the selected context, Data Integration uses the default SQL ELT optimization behavior. Select one of the following options: - None - SCD Type 2 merge - Multi-insert Default is None. For more information, see the help for the appropriate connector.
SQL ELT Optimization Fallback Option	If full SQL ELT optimization is not available for the connection, choose how Data Integration handles SQL ELT optimization. Choose one of the following options: - Partial SQL ELT. Default. Data Integration pushes as much transformation logic as possible to the source and target database. The task processes any transformation logic that it can't push to a database. - Non SQL ELT. The task runs with no SQL ELT optimization. - Fail Task. Data Integration fails the task. Default is disabled.

Property	Description
Create Temporary View	Allows the task to create temporary view objects in the database when it pushes the task to the database. Use when the task includes an SQL override in the Source Qualifier transformation or Lookup transformation. Default is enabled. Disabled when the SQL ELT optimization type is None.
Create Temporary Sequence	This option is not used.

8. Choose to enable cross-schema SQL ELT optimization.

   Cross-schema SQL ELT optimization doesn't apply to mapping tasks that are based on mappings in SQL ELT mode.

9. If you want to run multiple instances of the task at the same time, enable simultaneous runs of the mapping task.

   Some mapping features might produce unexpected results in simultaneous task runs.

10. Click **Finish**.

Viewing and editing mapping task details

You can view details about a mapping task, such as the mapping or runtime environment used by the task.

The **Task Details** page includes the following information:

- The runtime environment used to run the mapping task.
- The date the task was created and the user who created the task.
- The last time the task was updated and the user who updated the task.
- The date of the last run.
- The name and image of the mapping on which the task is based and the date the mapping was last updated.
- Pre-processing and post-processing commands.

To view details for a mapping task, perform the following steps:

1. On the **Explore** page, navigate to the task.
2. In the row that contains the task, click **Actions** and select **View**.

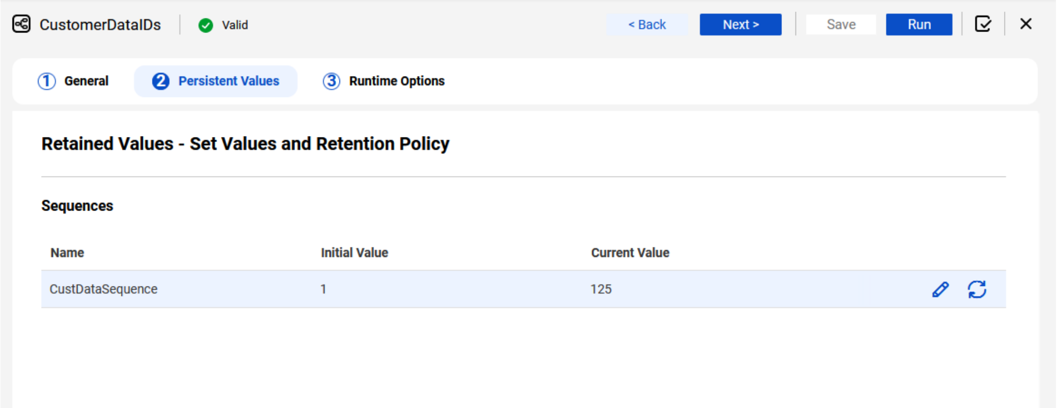
   On the **Task Details** page, you can click **Edit** to modify the mapping task.

Sequence values

When you run a mapping task that includes a Sequence transformation in the mapping, you can change the beginning value for the sequence.

To change the beginning value, you change the **Current Value** field in the **Sequences** page in the mapping task wizard. The **Current Value** field shows the first value the task will generate in the sequence, based on the last value generated in the last task execution.

For example, the last time you ran the CustDataIDs task, the last value generated was 124. The next time the task runs, the first number in the sequence is 125 because the Sequence transformation is configured to increment by 1. If you want the sequence to begin with 200, you change the **Current Value** to 200.



Retained Values - Set Values and Retention Policy

Sequences

Name	Initial Value	Current Value
CustDataSequence	1	125

Note: If the Sequence transformation is configured to reset the value for each run, editing the current value in the mapping task has no effect.

Running a mapping task

Run a mapping task after you configure all of the required settings.

You can run a mapping task in the following ways:

- Manually. To run a mapping task manually, on the **Explore** page, navigate to the task. In the row that contains the task, click **Actions** and select **Run**.
  You can also run a mapping task manually, with or without advanced options, from the **Task Details** page. To access the **Task Details** page, click **Actions** and select **View**.
- On a schedule. To run a mapping task on a schedule, edit the task in the mapping task wizard to associate the task with a schedule.

When a mapping task that processes change data from a CDC source is stopped, you can click **Restart** on the job details page to restart the task from the beginning or to resume from where it left off.

CHAPTER 3

Data transfer tasks

Use a data transfer task to transfer data from a source to a target. For example, you might use a data transfer task to transfer data from an on-premises database to a cloud data warehouse.

When you configure a data transfer task, you can augment the source data with data from a lookup source. Based on the source connection that you use, you can also sort and filter the data before you load it to the target.

To see if a data transfer task is applicable to the connectors you are using, see the help for the relevant connectors.

Task operations

When you configure a data transfer task, you specify the task operation. The operations available are based on the target that you select.

You can select the following operations:

Insert

Inserts all source rows into the target. If Data Integration finds a source row that exists in the target, the row fails.

If you write data to a flat file target, Data Integration truncates the flat file before it inserts the source rows into the file.

Update

Updates rows in the target that exist in the source. If Data Integration finds a row in the source that does not exist in the target, the row fails.

Upsert

Updates all rows in the target that also exist in the source and inserts all new source rows in to the target.

If a source field contains a NULL value and the corresponding target field contains a value, Data Integration retains the existing value in the target field.

Delete

Deletes all rows from the target that exist in the source.

Data transfer task sources

You can select a single source object to transfer data from or define a custom query.

The type of source and the formatting and advanced options that you can configure for the source depend on the source connection that you select. For example, you can create a custom query for database sources that support SQL queries. For a flat file source, you can configure formatting options such as the formatting type. For Salesforce sources, you can configure advanced options such as the SOQL filter condition, row limit, and bulk query.

For information about the options that you can configure for a source connection, see the help for the appropriate connector.

You can preview the data in the source. The preview returns the first 10 rows. To preview rows in alphabetical order, select **Display fields in alphabetical order**. Data Integration does not change the order of rows in the actual source. You can also download the preview results as a CSV file.

Source filters

Apply filter conditions to filter the source data that you transfer to the target.

You can apply the following types of filter conditions:

Simple

Select the source field, and configure the operator and value to use in the filter.

When you define more than one filter condition, the task evaluates them in the order that you specify. The task evaluates the filter conditions using the AND logical operator to join the conditions. It returns rows that match all the filter conditions.

Advanced

Create a filter expression using the expression editor. You enter one expression that contains all filters. You can use source fields and built-in functions in the expression.

For more information about advanced filters, see "Advanced data filters" on page 10.

Sort conditions

For certain source types, you can sort the source data to provided sorted data to the target.

When you sort data, you select one or more source fields to sort by. If you apply more than one sort condition, Data Integration sorts fields in the listed order.

To see if a connector supports sorting, see the help for the appropriate connector.

Custom queries

Create a custom query when you want to use a database source that you can't configure with a single object and lookup source. You might create a custom query to join multiple source tables.

To use a custom query as a source, select **Query** as the source type and then click **Select**. When you define the query, use SQL that is valid for the source database. You can use database-specific functions in the query.

When you create a custom query, enter an SQL SELECT statement to select the source columns you want to use. Data Integration uses the SQL statement to retrieve source column information.

Data Integration ensures that custom query column names are unique. If an SQL statement returns a duplicate column name, Data Integration adds a number to the duplicate column name as follows:

```
<column_name><number>
```

Tip: Test the SQL statement you want to use on the source database before you create a custom query. Data Integration doesn't display specific error messages for invalid SQL statements.

Second sources

When you configure a data transfer task, you can add a second source to use as a lookup source. Configure the lookup source on the **Second Source** page.

The task queries the lookup source based on the lookup condition that you specify and returns the result of the lookup to the target.

Select a second source when you want to augment the source data with a related value or values from the lookup source. For example, the source is an orders table that contains a customer ID field. You might retrieve the customer name and address from the lookup source so that you can include them in the target object. The task returns all fields from the lookup source.

To optimize performance, the task caches the lookup source. The cache remains static and does not change as the task runs. The task deletes the cache files after the task completes.

You can preview the data in the lookup source. The preview returns the first 10 rows. To preview rows in alphabetical order, select **Display fields in alphabetical order**. Data Integration does not change the order of rows in the actual source. You can also download the preview results to a CSV file.

You can also filter the data from both sources before writing it to the target.

Lookup condition

When you select a second source to use as a lookup source, you must configure one or more lookup conditions.

A lookup condition defines when the lookup returns values from the lookup source. When you configure a lookup condition, you compare the value of one or more fields from the original source with values in the lookup source.

A lookup condition includes an incoming field from the original source, a field from the lookup source, and an operator. To avoid possible naming conflicts, the data transfer task applies the prefix `SRC_` to the fields from the original source. If this results in a naming conflict for any field from the original source, the task applies the prefix `IN_SRC_` to the field from the original source.

For example, you might configure the following lookup condition when the original source contains the CustID field, the lookup source contains the CustomerID field, and you want to return values from the lookup source when the customer IDs match:

Lookup Field	**Operator**	**Incoming Field**
CustomerID	Equals	SRC_CustID

You can use the following operators in a lookup condition:

- Equals

- Not Equals
- Less Than
- Less Than or Equals
- Greater Than
- Greater Than or Equals

When you enter multiple conditions, the task evaluates the lookup conditions using the AND logical operator to join the conditions. It returns rows that match all the lookup conditions.

When you include multiple conditions, to optimize performance, enter the conditions in the following order:

1. Equals
2. Less Than, Less Than or Equals, Greater Than, Greater Than or Equals
3. Not Equals

The lookup condition matches null values. When an input field is NULL, the task evaluates the NULL equal to null values in the lookup.

If the lookup condition has multiple matches, the task returns any row.

Second source filters

You can apply filter conditions to filter the combined data.

You can configure the following types of filters:

Simple

To configure a filter condition, select a source field and configure the operator and value to use in the filter. You can select a field from either source. Fields from the original source are prefixed with the characters `SRC_` or `IN_SRC_`.

When you define more than one filter condition, the task evaluates them in the order that you specify. The task evaluates the filter conditions using the AND logical operator to join the conditions. It returns rows that match all the filter conditions.

Advanced

Create a filter expression using the expression editor. You enter one expression that contains all filters. You can use source fields and built-in functions in the expression.

For more information about advanced filters, see "Advanced data filters" on page 10.

Data transfer task targets

You can use a single object as a target for a data transfer task. Select a target object or create a new target object at run time.

The task operations that you can select depend on the target connection that you use. For more information about task operations for different target types, see the help for the appropriate connector.

You can preview the data in the target. The preview returns the first 10 rows. You can download the preview results to a CSV file. To preview rows in alphabetical order, select **Display fields in alphabetical order**. Data Integration does not change the order of rows in the actual target.

Database target truncation

You can configure a data transfer task to truncate a database target table before writing new data to the table when the task uses an Insert operation. By default, Data Integration inserts new rows without truncating the target table.

Update columns

Update columns are columns that uniquely identify rows in the target table. Add update columns when the database target table does not contain a primary key and the data transfer task uses an update, upsert, or delete operation.

When you run the data transfer task, the task uses the field mapping to match rows in the source to the database table. If the data transfer task matches a source row to multiple target rows, it performs the specified task operation on all matched target rows.

Field mapping

Configure field mapping in a data transfer task to map source fields to target fields. Configure field mapping on the **Field Mapping** page of the data transfer task wizard.

You must map at least one source field to a target field. If the task uses multiple sources, fields from the original source are prefixed with the characters `SRC_` or `IN_SRC_`.

You can configure the following field mapping options:

Options

Configure which fields to display. Click **Options** and select from the following display options:

- Show All
- Show Mapped
- Show Unmapped

Automap

Data Integration automatically links fields with the same name or similar name. Click **Automap** and select from the following mapping options:

- Exact Field Name. Data Integration matches fields of the same name.
- Smart Map. Data Integration matches fields with similar names. For example, if you have an incoming field `Cust_Name` and a target field `Customer_Name`, Data Integration automatically links the `Cust_Name` field with the `Customer_Name` field.
- Undo Automap. Data Integration clears fields mapping with Smart Map or Exact Field Name but does not clear manually mapped fields.

Actions

Additional field link options. Provides the following options:

- Map Selected. Links the selected incoming field with the selected target field.
- Unmap Selected. Clears the link for the selected field.
- Clear Mapping. Clears all field mappings.

After you map a field, if you want to configure a field expression, click the mapped field name. You can include fields and built-in functions in the expression but not user-defined functions.

When you create a target at run time, Data Integration maps the source fields to the target fields. You cannot unmap or edit the source fields mapped to the target but you can add fields to the target. You can also edit the mapped field expression and metadata and reorder the added fields. You cannot reorder the fields copied from the source.

Field data types

When you create a data transfer task, Data Integration assigns a data type to each field in the source and target. When you add a field to a target that you create at run time, you select the data type.

Configuring a data transfer task

To configure a data transfer task, perform the following steps:

1. Define the data transfer task.
2. Configure the source.
3. Optionally, configure a second source to use as a lookup source.
4. Configure the target.
5. Configure field mappings.
6. Configure runtime options.

As you work through the data transfer task wizard, you can click **Save** to save your work at any time. Use the **Validation** panel to validate the task. When you have completed the wizard, click **Exit** to close the task wizard.

Defining the data transfer task

1. To create a new data transfer task click **New** > **Tasks**. Select **Data Transfer Task** and click **Create**.

   To edit a data transfer task, on the **Explore** page, navigate to the data transfer task. In the row that contains the task, click **Actions** and select **Edit**.

2. Configure the following properties:

Property	Description
Name	Name of the data transfer task.
Location	Project and folder in which the task resides.

Property	Description
Description	Description of the data transfer task. Maximum length is 4000 characters.
Runtime Environment	Runtime environment that contains the Secure Agent to run the task.

3. Click **Next**.

Configuring the source

You can configure a single object or custom query as the source of a Data Integration task.

1. On the **Source** page, select the source connection.

   To create a connection, click **New**. To edit a connection, click **View**, and the in **View Connection** dialog box, click **Edit**.
2. Select the source type. To use a single source, select **Single**. To use a custom query, select **Query**.

   You can use a custom query when you use a database connection.
3. Select the source object or enter the query.
4. For file sources, configure formatting options.
5. Configure data filters.
6. Configure sort conditions.

   Not all connectors support sorting. The **Sort** area appears if the source connector supports sorting.
7. If preview data does not appear automatically, expand the **Data Preview** area to preview the source data.
8. Click **Next**.

Configuring a second source

Optionally, configure a second source to use as a lookup source.

1. On the **Second Source** page, select **Yes** to add a second source to the task.

   If you do not want to configure a second source, select **No**.
2. If you add a second source, perform the following steps to configure the source:
   a. In the **Source Details** area, select **Augment Data with Lookup**.
   b. Select the source connection and source object.

      To create a connection, click **New**. To edit a connection, click **View**, and the in **View Connection** dialog box, click **Edit**.
   c. For file sources, configure formatting options.
   d. If preview data does not appear automatically, expand the **Data Preview** area to preview the source data.
   e. Configure one or more lookup conditions.
   f. Optionally, configure data filters for the combined sources.
3. Click **Next**.

Configuring the target

You can write data to a single target. You can select an existing target object or create a new object at run time. If you create a target at run time, the task operation is Insert.

1. On the **Target** page, configure the following properties as required:

Property	Description
Connection	Select a connection. To create a connection, click **New**. To edit a connection, click **View**, and the in **View Connection** dialog box, click **Edit**. To configure advanced properties for the connection, click **Advanced Options**. Not available for all connections types. For information about a particular connector's properties, see the help for the appropriate connector.
Object	Select a target object. Click **Select**. The **Select Target Object** dialog box displays up to 200 objects. If the object you want to use does not display, enter a search string to reduce the number of objects that display.
Target operation	Select one of the following task operations: - Insert - Update - Upsert - Delete - Data Driven The operations available depend on the target connection.
Formatting Options	For flat file connections only. Select a delimiter and text qualifier. Optionally, select an escape character. If you choose Other for the delimiter, the delimiter cannot be an alphanumeric character or a double quotation mark.
Truncate Target	Database targets with the Insert task operation only. Truncates a database target table before inserting new rows. - True. Truncates the target table before inserting all rows. - False. Inserts new rows without truncating the target table. Default is False.
Enable Target Bulk Load	Select this option to write data in bulk mode. The default value is false.

2. For database targets, configure update columns if necessary.
3. If preview data does not appear automatically, expand the Data Preview area to preview the target data.
4. Click **Next**.

Configuring the field mapping

Configure field mappings to define the data that the data transfer task writes to the target. Configure field mappings on the **Field Mapping** page.

1. To match fields with the same name, click **Automap** > **Exact Field Name**. Or, to match fields with similar names, click **Automap** > **Smart Map**.

   You can also select and drag the source fields to the applicable target fields.
2. To configure the field expression, click the mapped field. In the **Field Expression** window, enter the expression you want to use and click **OK**.
3. If you create a target at run time and want to add target fields, click **Add**. Configure the following field properties:

Property	Description
Name	Name of the field.
Type	The data type of the data in the column.
Precision	Total number of digits in a number. For example, the number 123.45 has a precision of 5. The precision must be greater than or equal to 1.
Scale	The number of digits to the right of the decimal point of a number. For example, the number 123.45 has a scale of 2. Scale must be greater than or equal to 0. The scale of a number must be less than its precision. The maximum scale for a numeric data type is 65535.

4. Click **Next**.

Configuring runtime options

Configure runtime options on the **Runtime Options** page.

You can run a data transfer task manually or on a schedule.

1. To run a data transfer task on a schedule, select **Run on a schedule** and then select the schedule.

   Note: You must create the schedule in Administrator before you can select it in the task.
2. Configure email notification options for the task.
3. Click **Save**.

Running a data transfer task

You can run a data transfer task in one of the following ways:

- Manually. To run a data transfer task manually, on the **Explore** page, navigate to the task. In the row that contains the task, click **Actions** and select **Run**.
  You can also run a data transfer task manually from the **Task Details** page. To access the **Task Details** page, click **Actions** and select **View**.
- On a schedule. To run a data transfer task on a schedule, edit the task in the data transfer task wizard to associate the task with a schedule.

CHAPTER 4

Data loader tasks

Create a data loader task to load large data sets to a cloud data warehouse quickly and securely.

To start creating a task, click **New** in navigation menu on the left or **Get Started** on the **Welcome** page.

You can create a data loader task in just a few steps:

1. Connect to your source.

   Choose the location you want to read data from. You can refine your source data by choosing the objects to read and the fields and records you want to process. You can also select primary key fields and the fields which indicate whether data has changed since the last task run.

2. Connect to your target.

   Choose the location you want to load data to. You can also decide whether to keep or re-create the target objects when you re-run the task.

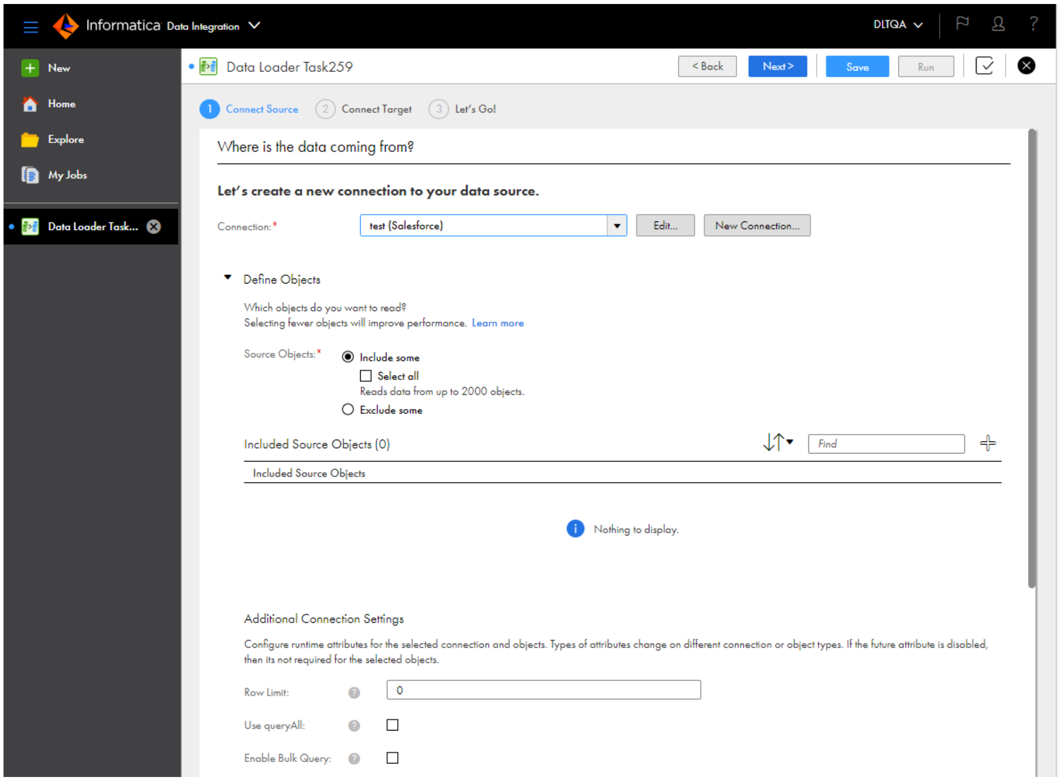
3. Save and run the task.

   You can run your task when you finish creating it, save it and run it later, or run it on a schedule. You can also choose whether to send notification emails related to the task run.

Step 1. Connect to your source

Configure a source connection to connect to your source. Configure the source connection on the **Connect Source** page. You can create a new connection or select an existing connection.

The following image shows the **Connect Source** page:



Where is the data coming from?

Let's create a new connection to your data source.

Connection:* test (Salesforce)

Define Objects

Which objects do you want to read?
Selecting fewer objects will improve performance. Learn more

Source Objects:* ● Include some
□ Select all
Reads data from up to 2000 objects.
○ Exclude some

Included Source Objects (0)

Included Source Objects

Nothing to display.

Additional Connection Settings

Configure runtime attributes for the selected connection and objects. Types of attributes change on different connection or object types. If the future attribute is disabled, then its not required for the selected objects.

Row Limit: 0

Use queryAll: □

Enable Bulk Query: □

To create a source connection, click **New Connection**. You can also select an existing connection. After you choose the connection, you might have to configure additional connection settings like **Row Limit** and **Use queryAll** that vary based on the connection type. For more information about connection properties, see *Connections* .

By default, Data Integration adds only the source objects you want to include. You can configure this feature using options in the Source Objects field.

For some connection types, Data Integration automatically detects primary key fields and watermark fields. Watermark fields identify the records that were added or changed since the last task run.

Click **Next** to customize the target connection.

You can customize the following source options:

- Which source objects to read
- The source format (file sources only)
- Which source fields to exclude
- Which records to include
- Primary key fields
- Watermark fields

For guidelines about customizing the source, see "Guidelines for customizing your source" on page 60.

Guidelines for customizing your source

To increase performance by removing unnecessary data, by default Data Integration adds only the source objects you include in the source location.

Use the following guidelines when customizing your source:

Remove unnecessary data from the data flow.

When you create a task, it's best to remove unnecessary source objects, fields, and records from the data flow. Removing unnecessary data decreases the time it takes to run a task and helps to minimize rejected records.

You can choose which source objects to read on the **Connect Source** page. You can also select the fields to exclude and configure filters to exclude unnecessary records.

To prevent duplicate target rows, configure primary key fields.

Primary key fields uniquely identify records in the source and target objects. When you re-run a task, the task uses the primary key fields so that it can update existing rows and insert new rows into the target tables. If the source objects don't have primary key fields defined, the task inserts rows into the target tables, but it cannot update existing rows, which can lead to duplicate rows in the target tables.

Data loader tasks can automatically detect primary key fields for most connection types. You can also select the primary key fields manually. Configure primary key field options on the **Connect Source** page.

To process only new and changed data, configure watermark fields.

Watermark fields are date/time or numeric fields that identify which records were added or changed. If the source objects don't have watermark fields defined, the task must process all records in the source objects each time the task runs, which increases the task processing time.

Data loader tasks can automatically detect watermark fields for most connection types. You can also select the watermark fields manually. Configure watermark field options on the **Connect Source** page.

The following table lists the possible primary key and watermark field configurations and the expected results:

Primary key and watermark field configuration	Result
Primary key fields configured. Watermark fields configured.	Changed records updated, new records inserted into the target tables (upsert). Recommended configuration for best performance.
Primary key fields configured. Watermark fields not required.	Changed records updated, new records inserted into the target tables (upsert). Impacts task performance because the task must perform a full scan on the source.
Primary key fields not required. Watermark fields configured.	All records inserted into the target tables. Records that already exist are duplicated.
Primary key fields not required. Watermark fields not required.	All records inserted into the target tables. Records that already exist are duplicated. Impacts task performance because the task must perform a full scan on the source. This is the least recommended configuration.

Selecting source objects

After you configure a new connection or select an existing connection, you select the objects to read.

To load the data as quickly as possible, configure the task to read only the source objects that you need to process. Reading data from fewer source objects decreases the time it takes to run the task.

You can configure the objects to read on the **Connect Source** page under **Define Objects**.

By default, the task reads data only from the objects you include in the source location. If you want to read data from most objects in the source location, choose **Include some** or **Exclude some**, then select the objects that you do or don't want to read. If you want to read data from all objects, choose **Select all**. If you choose **Select all** a data loader task can read up to 2000 objects.

If you choose to read all objects or exclude some objects, you'll need to enter the source path for some source types. If you entered a source path when you created the connection, the default source path is the same as the one you entered when you created the connection. Otherwise, the source path is empty by default. If you plan to read all objects or exclude objects in different locations, enter the path to the parent container.

To configure the objects to exclude or include, click the plus sign (**+**) icon in the **Excluded Source Objects** or **Included Source Objects** area. After you select objects, the **Excluded Source Objects** or **Included Source Objects** area displays the objects you excluded or included. To delete an object, click the **Delete** icon in the row that contains the object.

Source path syntax

The source path syntax varies based on the connection type. To find the source path syntax, click **Edit**.

For most source types, you enter the source path in the format `<container or bucket name>/<folder name>/<subfolder name>` or `<database name>/<schema name>`. If you enter a source path for an Amazon S3, Azure Blob Storage, or Azure Data Lake Storage Gen 2 source, you'll need to append a slash character at the end of each folder name to distinguish the folder from a file. So, for these sources, the source path format is `<container or bucket name>/<folder name/>/<subfolder name/>`, for example, `bucket1/folder1//folder2//folder3/`.

For more information about connection properties, see *Connections* .

Defining the source format

If you connect to a file source, you'll need to configure formatting options such as the file format, schema source, or delimiter.

Configure the formatting options on the **Connect Source** page under **Define Source Format**. If you don't see the **Define Source Format** area, then you don't have to configure formatting options for your source.

When you configure formatting options, you can preview the data to ensure that the formatting options are correct. You can select the fields to use in the data preview.

Selecting source fields

By default, a data loader task reads the fields only from the sources you include and loads them to each corresponding target. You can exclude unnecessary fields for one or more source objects so that they are not written to the target. Excluding unnecessary fields from the data flow helps to increase task performance.

Configure the fields to exclude on the **Connect Source** page under **Exclude Fields**. To exclude fields, click the plus sign (**+**) icon in the **Excluded Fields** area, and then select the source object and the fields to exclude.

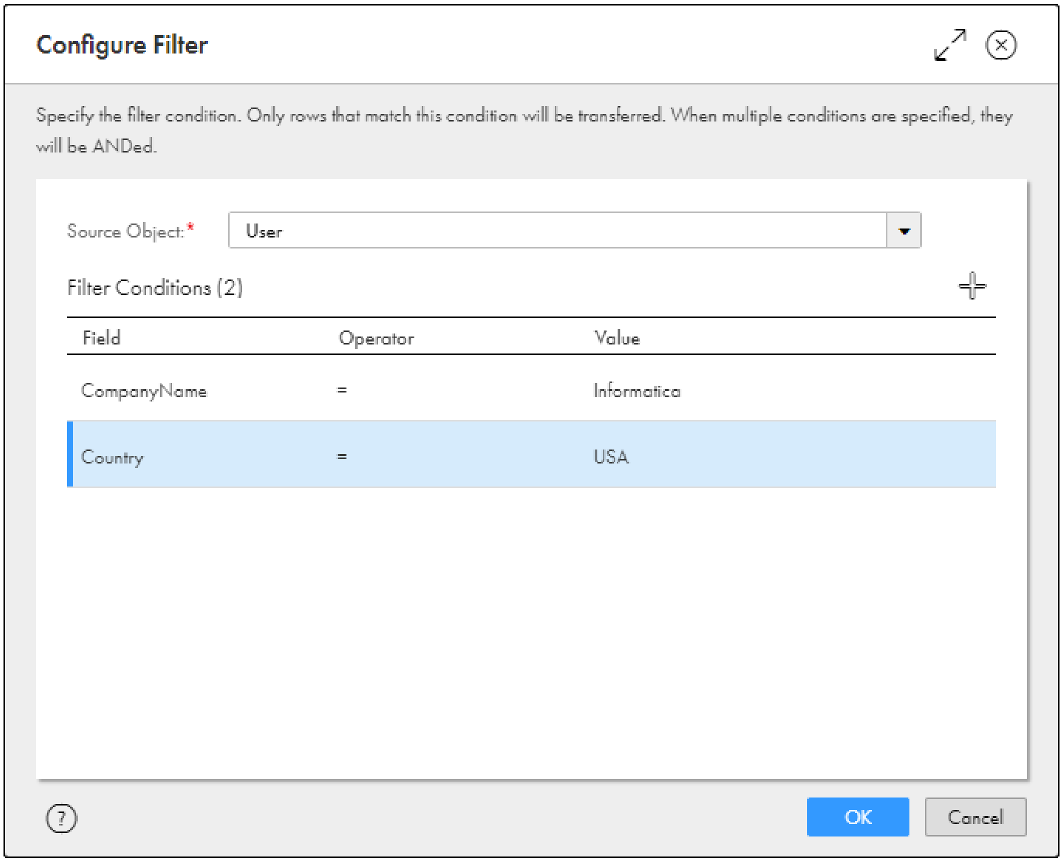
The **Excluded Fields** area displays the fields you excluded for each source object. To update the excluded fields for a source object, click the excluded fields for that source object. To delete all excluded fields for a source object, click the **Delete** icon in the row that contains the source object.

Configuring filters

You can configure filters for one or more source objects so that only records that match the filter conditions are written to the target. The task processes the records that match all of the filter conditions.

Configure filters on the **Connect Source** page under **Define Filters**. To add filters, click **+** in the **Filter Conditions** area. In the **Configure Filters** dialog box, select the source object and configure one or more filter conditions.

This image shows the **Configure Filter** dialog box:



Configure Filter

Specify the filter condition. Only rows that match this condition will be transferred. When multiple conditions are specified, they will be ANDed.

Source Object:* User

Filter Conditions (2)

Field	Operator	Value
CompanyName	=	Informatica
Country	=	USA

To add a filter condition, click the plus sign (**+**) icon. Each filter condition contains a field, operator, and value. You can use the following operators in a filter condition:

= (equals)

< (less than)

> (greater than)

<= (less than or equal to)

>= (greater than or equal to)

!= (not equals)

Enter date values in the format YYYY-MM-DD HH24:MI:SS.

If you add multiple filter conditions, the task evaluates the filter conditions in the order in which they appear in the **Configure Filters** dialog box.

For example, you want to extract records from the Orders table only when the city is New York and the order date is later than January 1, 1990. Select "Orders" as the source object, and configure the following filter conditions:

```
City = New York
```

```
OrderDate > 1990-01-01 00:00:00
```

The **Filter Conditions** area displays the filters you added for each source object. To update the filters for a source object, click the filters in the row that contains the source object. To delete all filters for a source object, click the **Delete** icon in the row that contains the source object.

Selecting primary key fields

Data loader tasks can automatically detect primary key fields for many source types. You can also choose your own primary key fields, or set them as not required.

Primary key fields uniquely identify each record in a source object. For example, in a CUSTOMERS table, the values in the CUSTOMER_ID column uniquely identify each customer. A source object can have one primary key field, a composite key that consists of several primary key fields, or no primary key fields.

Primary key fields are needed if you want to update rows when you re-run the task. If there are no primary key fields, the task inserts records into the target tables, but it cannot update rows.

Select primary key fields on the **Connect Source** page under **Define Primary Keys**.

You can select primary key fields in the following ways:

Automatically detect from source

The task identifies the primary key fields automatically in each source object that has them. This is the default option for many types of sources. This option is not available for file-based sources.

If you choose this option but there are no primary key fields in the source objects or the task can't detect the primary key fields, then the task inserts records into the target tables.

Primary key fields not required

Select this option if you want to perform a full load every time the task runs. For example, you want to read data from database tables that have primary key fields, but you don't want to update rows in the target when you re-run the task.

When you select this option, the task inserts records into the target tables.

Enter manually

Select this option to choose the primary key fields in each source object. If you select this option, you must also specify whether the source objects have the same primary key fields or different primary key fields.

If the source objects have the same primary key fields, select **The same across all sources**. Then, click **Choose** to select the primary key fields.

If the source objects have different primary key fields, select **Different across sources**. Then, click the plus sign (**+**) icon in the **Primary Key Fields** area and select the source object and the primary key fields. Repeat this process to select primary key fields for other source objects.

The **Primary Key Fields** area displays the primary key fields for each source object. To update the primary key fields for a source object, click the primary key fields for that source object. To delete all primary key fields for a source object, click the **Delete** icon in the row that contains the source object.

Selecting watermark fields

Data loader tasks can automatically detect watermark fields for many source types. You can also choose your own watermark fields, or set them as not required.

Watermark fields are date/time or numeric fields that identify the records that were added or changed since the last task run. For example, you might select the MODIFIED_DATE column as the watermark field for a source table. A source object can have one or no watermark fields.

If you want the task to load only new and changed data, you'll need to make sure that watermark fields are configured. If there are no watermark fields, the task processes all fields in the source objects.

Select watermark fields on the **Connect Source** page under **Define Watermark Fields**.

You can select watermark fields in the following ways:

Automatically detect from source

The task identifies the watermark field automatically in each source object that has one. This is the default option.

If you choose this option but there are no watermark fields in the source objects or the task can't detect the watermark fields, then the task processes all records each time the task runs.

Watermark field not required

Select this option if you want the task to processes all records each time the task runs.

Enter manually

Select this option to choose the watermark field in each source object. If you select this option, you must also specify whether the source objects have the same watermark field or different watermark fields.

If the source objects have the same watermark field, select **The same across all sources**. Then, click **Choose** to select the watermark field.

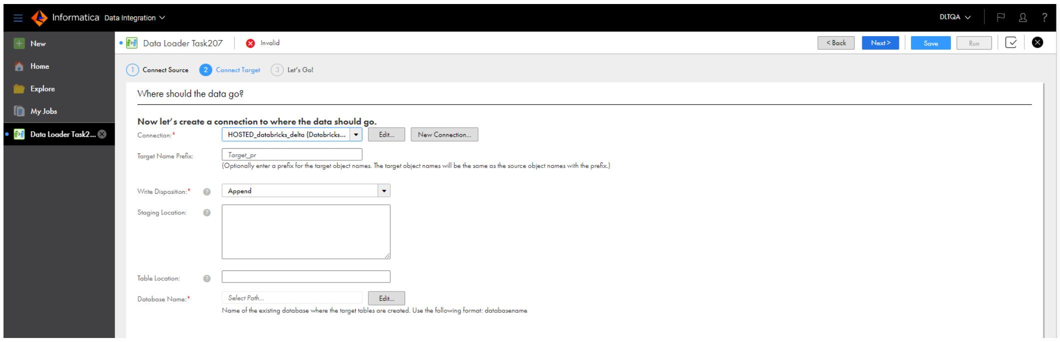
If the source objects have different watermark fields, select **Different across sources**. Then, click the plus sign (**+**) icon in the **Watermark Fields** area and select the source object and the watermark field. Repeat this process to select watermark fields for other source objects.

The **Watermark Fields** area displays the watermark field for each source object. To update the watermark field for a source object, click the watermark field for that source object. To delete the watermark field for a source object, click the **Delete** icon in the row that contains the source object.

Step 2. Connect to your target

Configure a target connection to connect to your target. Configure the target connection on the **Connect Target** page. You can create a new connection or select an existing connection.

The following image shows the **Connect Target** page:



Where should the data go?

Now let's create a connection to where the data should go.

Connection:* HOSTED_databricks_delta (Databricks...

Target Name Prefix: Target_pr

(Optionally enter a prefix for the target object names. The target object names will be the same as the source object names with the prefix.)

Write Disposition:* Append

Staging Location:

Table Location:

Database Name:* Select Path...

Name of the existing database where the target tables are created. Use the following format: databasename

To create a target connection, click **New Connection**. You can also select an existing connection. After you choose the connection, you might have to configure additional properties like **Write Disposition** and **Staging Location** that vary based on the connection type.

The first time you run a data loader task, the task creates one target table for each source object. By default, target table names are the same as the source object names. However, special characters in the source table names will be replaced with underscore characters. For example, if the source table name is `Orders$`, the corresponding target table will be named `Orders_`.

If you enter a target name prefix, the target object names will be the same as the source object names, but they'll be preceded with the prefix. For example, if a source table name is `Account`, and you enter the prefix `tgt_`, the target table name will be `tgt_Account`.

When you re-run a task, you can either load data to the target incrementally or drop and re-create the target tables.

Incremental loading

You can configure a data loader task to load data to the target incrementally. When the task loads data incrementally, only new and changed data is loaded to the target each time you re-run the task. Incremental loading increases task performance since fewer rows are loaded.

To configure the task to load data incrementally, select **Yes** under **Load to existing tables** on the **Connect Target** page.

When you load data incrementally, the task runs most efficiently when you configure the following options in the source:

Primary key fields are detected automatically or entered manually.

> The task uses the primary key fields to identify rows when updating or inserting data into the target tables. If there are no primary key fields, the task can insert data into the target tables, but it can't update existing rows. This can create duplicate rows in the target tables.

Watermark fields are detected automatically or entered manually.

> To load data incrementally, the source objects must contain watermark fields. The task uses the watermark fields to determine which records have been added or changed since the last task run. If there are no watermark fields, the task can't determine which records are new or changed. So, it processes all records in the source objects each time the task runs.

If the source object structure changes between task runs, the data loader task detects the changes and alters the target tables to match the source objects. Therefore, if you add a field to a source object or change a field's data type, the task adds the new column or updates the column data type in the corresponding target table. The task does not delete any existing target fields, however. If you want to delete unnecessary target fields, you can either delete the fields manually or configure the task to drop and re-create the target tables.

If the name or data type of the watermark column changes between task runs, or if you change the watermark column between task runs, the task performs a full load even if you've configured the target to load to existing tables.

Dropping and re-creating target tables

You can configure a data loader task to drop the target tables and re-create them every time the task runs. When you select this option, the task deletes the target tables created by the original task run and creates new tables using the schema of the source objects.

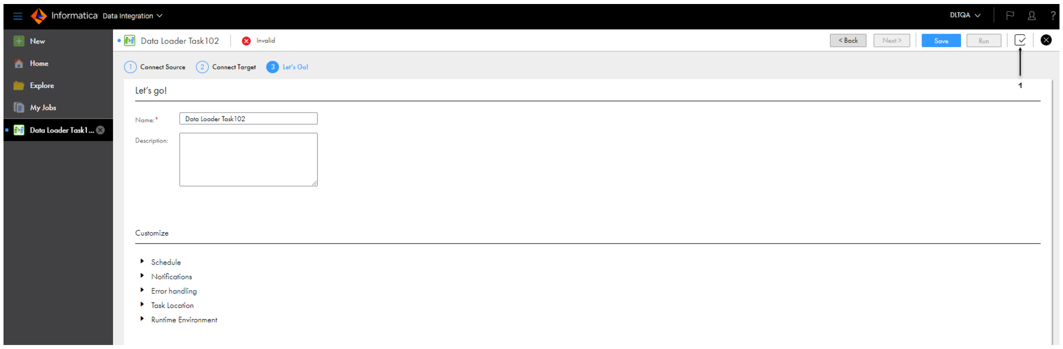
To configure the task to drop and re-create the target tables, select **No, create new tables every time** under **Load to existing tables** on the **Connect Target** page. To ensure that the task processes all records each time it runs, you also need to select **Watermark field not required** under **Define Watermark Fields** on the **Connect Source** page.

When you select these options, the task performs a full target load each time it runs.

Step 3. Run the task

After you configure the target, you give the task a name and decide how to run it. Configure these options on the **Let's Go** page.

This image shows the **Let's Go** page:



Name:* Data Loader Task102

Description:

1. Validation icon

Before you run the task, you can check it for errors using the validation panel. To open the validation panel, click the **Validation** icon. If there are validation errors, you can save the task, but you can't run it until you fix the errors. If there are no errors, you can run the task now, save it and run it later, or run it on a schedule.

To run the task now, click **Save** to save the task, and then click **Run**. To save it and run it later, click **Save**.

You can also configure the schedule, email notification options, error handling, and task location. The runtime environment is always the Informatica Cloud Hosted Agent, which is maintained by Informatica.

When you run the task, data loading occurs in parallel batches of five objects each. So, the first batch loads data from the first five source objects to the first five target objects. The next batch loads data from the second set of five source objects, and so on. You can monitor the task progress on the **My Jobs** page. For more information, see *Monitor*.

Running the task on a schedule

You can run your task on a schedule. For example, you might want to run the task every 12 hours or every Friday at noon.

Select a schedule for your task on the **Let's Go** page under **Schedule**. Click **Run on a schedule**, and then select the schedule.

If you want to create a new schedule, click **New Schedule**. To make the schedule easy to identify, give the schedule a meaningful name such as "Fridays at 12PM." For the start time, the date format is MM-DD-YYYY, and the time appears in the 12-hour format.

You can also create, view, edit, and delete schedules in the Administrator service. For more information, see *Organization Administration*.

Configuring email notifications

You can configure the task to send email notifications to users when the task completes successfully, completes with warnings, or fails. If you don't configure email notifications, the task doesn't send any.

Configure email notification options for your task on the **Let's Go** page under **Notifications**. You can send notifications to different addresses based on whether the task completed successfully, completed with warnings, or failed.

You can enter multiple email addresses in each field. If you enter multiple addresses, separate them with commas.

Handling runtime errors

If a task encounters a runtime error, you can stop the task or skip the error and continue. The default is to stop the task.

To determine how Data Loader handles runtime errors, set one of the following options:

- Stop the task.
- Skip the error object and continue the task.

Configuring the task location

You can select the project where your data loader task is stored.

Configure the task location on the **Let's Go** page under **Task Location**. By default, the task is stored in the currently open project, or, if no project is open, in the project where you saved your last task. If you haven't previously created a task, the task is stored in the Default project. To select a different location, click the project name and choose a different project.

If you're editing a task you saved before, the task location is read-only. You can move the task to a different project on the **Explore** page.

Runtime environment

The runtime environment is the execution platform for data loader tasks.

You can view the runtime environment on the **Let's Go** page under **Runtime Environment**. The runtime environment for all data loader tasks is the Informatica Cloud Hosted Agent.

The Informatica Cloud Hosted Agent connects to your sources and targets securely and does the data processing when you run a task. The hosted agent is maintained by Informatica. There is nothing to download, and there are no settings to configure.

INDEX

A

advanced data filters
  configuring 10
  deleting 10
advanced relationships
  in mapping tasks 33
AND
  reserved word 16

C

CHR function
  inserting single quotation mark 14
Cloud Application Integration community
  URL 6
Cloud Developer community
  URL 6
comments
  adding to field expressions 15
constants
  definition 14

D

data filters
  configuring 9
  configuring advanced filters 10
  configuring simple filters 9
  deleting 9
  operators 10
  rules for configuring 12
  variables 11
Data Integration community
  URL 6
data integration tasks
  rules for flat file sources and targets 29
data loader tasks
  configuring filters 62
  connecting to the source 59
  connecting to the target 65
  creating 58
  email notifications 66, 67
  handling runtime errors 67
  incremental loading 65
  primary keys 63
  running 66
  running on a schedule 67
  runtime environment 68
  scheduling 66
  selecting source fields 61
  selecting source objects 61
  source customization guidelines 60
  source format 61
data loader tasks *(continued)*
  source path syntax 61
  task location 67
  validating 66
  watermark fields 64
data transfer task
  configuration 54
  configuring a lookup source 55
  configuring a second source 55
  configuring the source 55
  configuring the target 56
  creation 54
  description 49
  field mapping 57
  lookup condition 51
  lookup source 51
  operations 49
  running 57
  runtime options 57
  scheduling 57
  second source 51
  second source filters 52
  sort conditions 50
  source filters 50
  targets 52
data transfer task sources 50
database sources and targets
  rules in synchronization tasks 29
Daylight Savings Time
  schedules 25
DD_DELETE constant
  reserved word 16
DD_INSERT constant
  reserved word 16
DD_REJECT constant
  reserved word 16
DD_UPDATE constant
  reserved word 16
deleting
  advanced data filters 10
  data filters 9
dynamic schema handling 34, 37

E

email notification
  configuring for tasks 27
email notifications
  configuring 67
expressions
  rules 15
:EXT reference qualifier
  reserved word 16

F

FALSE constant
  reserved word 16
field data type 54
field expressions
  comments, adding 15
  components 14
  creating 13
  literals 14
  reserved words 16
  rules 15
  rules for validating 13
  syntax 14
field mapping
  configuration 57
field metadata 12
filters
  configuring 62
flat files
  editing metadata in mapping tasks 12
  rules for sources and targets in data integration tasks 29
functions
  definition 14

I

in-out parameters
  in mapping tasks 43
:INFA reference qualifier
  reserved word 16
Informatica Cloud Hosted Agent 68
Informatica Global Customer Support
  contact information 7
Informatica Intelligent Cloud Services
  web site 6
input parameters
  in mapping tasks 42

J

jobs
  monitoring 28
  scheduling 23

L

$LastRunDate
  data filter variable 11
$LastRunTime
  data filter variable 11
literals
  definition 14
  single quotation mark requirement 14
  string and numeric 14
:LKP reference qualifier
  reserved word 16

M

maintenance outages 7
mapping tasks
  advanced session properties for mappings 17
  configuring 37
mapping tasks *(continued)*
  configuring schedules, email, and advanced options 43
  configuring sources 38
  creating 37
  editing 37
  in-out parameters 43
  input parameters 42
  joining related sources 32
  overview 31
  running 48
  Sequence configuration 43, 47
  SQL ELT optimization 33
  templates 31
  using parameter files 23
  viewing details 47
Masking tasks
  using parameter files 23
:MCR reference qualifier
  reserved word 16
monitoring jobs 28

N

NOT
  reserved word 16
NULL constant
  reserved word 16

O

operators
  definition 14
OR
  reserved word 16

P

parameter file
  configuring for mapping tasks 43
parameter files
  overview 23
parameters
  user defined 23
postprocessing commands
  SQL commands 28
  tasks 27
preprocessing and postprocessing commands
  configuring for mapping tasks 43
  rules for operating system commands 28
  rules for SQL commands 28
preprocessing commands
  SQL commands 28
  tasks 27
primary keys
  configuring 63
PROC_RESULT variable
  reserved word 16

Q

quotation marks
  inserting single using CHR function 14

R

repeat frequency
    schedules 24
reserved words
    list 16
rules for sources and targets 29
runtime environment
    data loader tasks 68

S

schedules
    configuring 26
    creating 26
    Daylight Savings Time 25
    repeat frequency 24
    time zones 25
scheduling
    mapping tasks 43
    taskflows 23
    tasks 23
:SD reference qualifier
    reserved word 16
:SEQ reference qualifier
    reserved word 16
simple data filters
    configuring 9
sort conditions
    data transfer task 50
source filters
    data transfer tasks 50, 52
sources
    configuring filters 62
    configuring for mapping tasks 38
    configuring primary keys 63
    configuring watermark fields 64
    customization guidelines 60
    formatting objects 61
    selecting fields 61
    selecting objects 61
    source path 61
    source path syntax 61
:SP reference qualifier
    reserved word 16
SPOUTPUT
    reserved word 16
SQL ELT optimization
    mapping tasks 33
status
    Informatica Intelligent Cloud Services 7
stopping a running job 28
string literals
    single quotation mark requirement 14
synchronization tasks
    rules for database sources and targets 29
    using parameter files 23
syntax
    for field expressions 14
system status 7

T

targets
    configuring 65
    configuring for mapping tasks 40
    dropping and re-creating 66
    incremental loading 65
task operations
    data transfer tasks 49
taskflows
    monitoring 28
    scheduling 23
tasks
    configuring email notification 27
    configuring field expressions 13
    creating schedules 26
    monitoring 28
    scheduling 23
    stopping a running job 28
:TD reference qualifier
    reserved word 16
time zones
    description 25
TRUE constant
    reserved word 16
trust site
    description 7

U

update columns
    data transfer task 53
upgrade notifications 7
user parameters 23

V

validation
    rules 13
variables
    for data filters 11

W

watermark fields
    configuring 64
web site 6
WORKFLOWSTARTTIME variable
    reserved word 16