Informatica® Cloud Data Integration

MySQL Connector

Informatica Cloud Data Integration MySQL Connector
April 2024

Table of Contents

Preface

Use *MySQL Connector* to learn how to read from or write to MySQL databases by using Cloud Data Integration. Learn to create a MySQL connection, develop mappings, and run synchronization, mapping, dynamic mapping, replication, and data transfer tasks in Data Integration.

Informatica Resources

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

Informatica Documentation

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

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

Informatica Intelligent Cloud Services web site

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

Informatica Intelligent Cloud Services Communities

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

Access the Informatica Intelligent Cloud Services Community at:

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

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

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

Informatica Intelligent Cloud Services Marketplace

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

https://marketplace.informatica.com/

Data Integration connector documentation

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

Informatica Knowledge Base

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

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

Informatica Intelligent Cloud Services Trust Center

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

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

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

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

Informatica Global Customer Support

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

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

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

CHAPTER 1

Introduction to MySQL Connector

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

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

- Amazon Aurora MySQL
- Amazon Aurora MySQL and Azure MySQL private endpoints on virtual networks

You can use MySQL objects as sources and targets in synchronization tasks, mappings, mapping tasks, replication tasks, PowerCenter tasks, and data transfer tasks. You can create, update, or delete records when you use MySQL Connector to write to the MySQL or Amazon Aurora MySQL databases.

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

MySQL Connector assets

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

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

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

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

Administration of MySQL Connector

To use MySQL Connector, you must download and install the JDBC and ODBC drivers on your Windows or Linux machine where the Secure Agent is installed.

Before you use MySQL Connector, complete the following tasks:

1. Install the MySQL JDBC driver.
2. Install the MySQL ODBC driver.

Installing MySQL JDBC driver

Before you use MySQL Connector, you must install the MySQL JDBC driver and MySQL ODBC driver on the Windows or Linux machine where you installed the Secure Agent.

Perform the following steps to install the MySQL JDBC drivers on Windows or Linux:

1. Click the following link to download the MySQL JDBC driver: https://downloads.mysql.com/archives/c-j/
2. Select the required **Product Version**.
3. Select the **Operating System** as **Platform Independent**.
4. Download the **.zip** or **.tar** file based on your Windows or Linux system.
5. Extract the downloaded file and copy the `mysql-connector-java-<version>.jar` file.
6. Paste the `mysql-connector-java-<version>.jar` file in the following directory on the Secure Agent machine:
   - On Windows: `<Secure Agent installation directory>\apps\Data_Integration_Server\ext\drivers`
   - On Linux: `<Secure Agent installation directory>/apps/Data_Integration_Server/ext/drivers`
7. Restart the Secure Agent.

Installing MySQL ODBC driver

Before you use MySQL Connector, you must install the MySQL ODBC driver on the Windows or Linux machine where you installed the Secure Agent.

Installing MySQL ODBC driver on Windows

Perform the following steps to install the MySQL ODBC driver on Windows:

1. Click the following link to download the MySQL ODBC driver: https://downloads.mysql.com/archives/c-odbc/
2. Select the required **Product Version**.
3. Select the **Operating System** as **Microsoft Windows**.
4. Select the **OS Version** as **Windows (x86, 64-bit)**.
5. Download the ZIP Archive `mysql-connector-odbc-noinstall-<version>-winx64.zip` file.

   Note: Do not use the MSI installer because it removes the existing MySQL ODBC driver versions from the system.
6. Extract the .zip file.
7. Open the Command Prompt as an administrator and navigate to the extracted .zip file location.

8. Run the following command in the command prompt: Install.bat

After you install the MySQL ODBC driver, check if the installed MySQL ODBC Unicode Driver name appears as one of the available ODBC drivers under the ODBC Data Source Administrator (64-bit) dialog box:

The following image shows the **MySQL ODBC 8.0 Unicode Driver** driver name in the **ODBC Data Source Administrator (64-bit)** dialog box when you try to create a new data source:
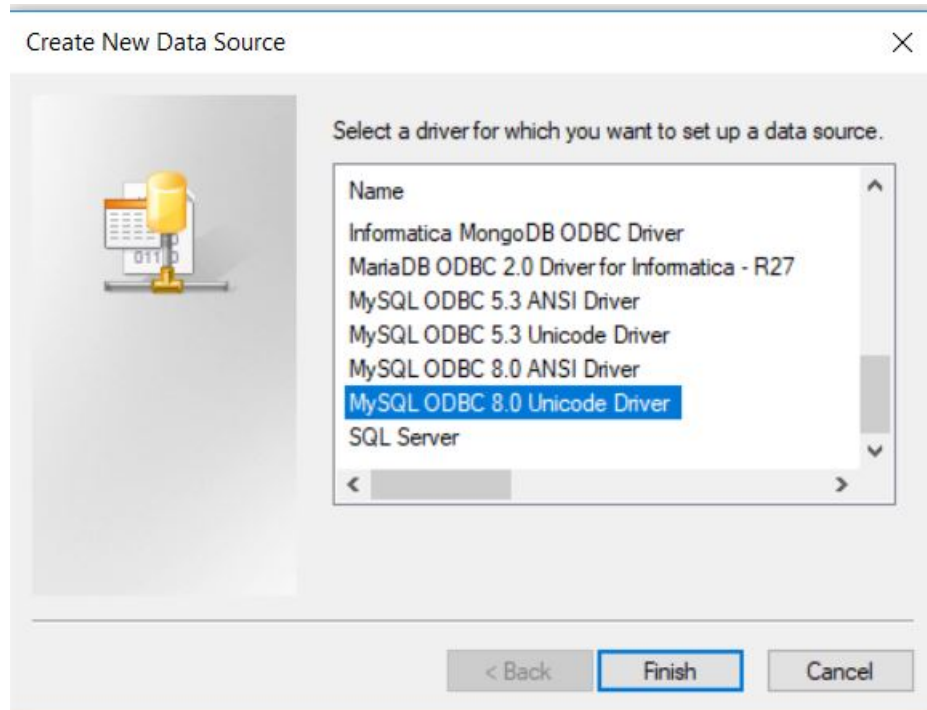
Installing MySQL ODBC driver on Linux

Perform the following steps to install the MySQL ODBC driver on Linux:

1. Click the following link to download the MySQL ODBC driver:
    https://downloads.mysql.com/archives/c-odbc/

2. Select the required **Product Version**.

3. Select the **Operating System** as **Linux- Generic**.

4. Select the **OS Version** as **All**.

5. Download the **Linux - Generic (glibc 2.12) (x86, 64-bit)** file.

6. Extract the downloaded file to a local directory in your system.

    Note: When you extract the file, ensure that you copy all files from the downloaded driver package.

7. Edit the `<Secure Agent installation directory>/odbcinst.ini` file and add the following values:

    ```
    [MySQL ODBC 8.0 Unicode Driver]
    Description = ODBC for MySQL
    Driver = <Extracted folder path>/lib/libmyodbc8w.so
    Setup = <Extracted folder path>/lib/libmyodbc8w.so
    FileUsage = 1
    ```

    In the **Driver** and **Setup** fields, you must specify the file path of the `libmyodbc8w.so` file that you extracted to the local directory in your system.

Prerequisites to configure SSL with serverless runtime environment

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

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

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

2. Add the truststore and keystore certificates in the Amazon S3 bucket or Azure container in the following location in your AWS or Azure account: `<Supplementary file location>/serverless_agent_config/SSL`

3. Copy the following code snippet to a text editor:
   ```
   version: 1
   agent:
    agentAutoApply:
       general:
         sslStore:
           - fileCopy:
               sourcePath: SSL/<TrustStore_filename>
           - fileCopy:
               sourcePath: SSL/<KeyStore_filename>
     dataIntegrationServer:
       autoDeploy:
         mysql:
           jdbc:
             drivers:
               - fileCopy:
                   sourcePath: <Driver directory>/<MySQL_JDBC_filename>
           odbc:
             drivers:
               - fileCopy:
                   sourcePath: <Driver directory>/<MySQL_ODBC_ZIP_filename>
   ```
   where the source path is the directory path of the certificate files in AWS or Azure.

4. Ensure that the syntax and indentations are valid, and then save the file as `serverlessUserAgentConfig.yml` in the following AWS or Azure location: `<Supplementary file location>/serverless_agent_config`
   When the .yml file runs, the SSL certificates are copied from the AWS or Azure location to the serverless agent directory.

5. In the MySQL connection properties, specify the following certificate path in the serverless agent directory in the **Trust Store** and **Key Store** fields: `/home/cldagnt/SystemAgent/serverless/configurations/ssl_store/<cert_filename>`

Rules and guidelines for MySQL sources and targets

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

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

- You cannot include lookup fields of Text or Ntext data type because MySQL connections use a UTF-8 code page.

- When you use a saved query to read data from a MySQL source, you must not use the full outer join operation.
- When you use a saved query to read data from a MySQL source that contains Float columns, the Secure Agent imports the Float data type as Real data type.
- When you use a saved query to read data from a MySQL source that contains Double columns, the Secure Agent imports the Double data type as Decimal data type.
- You cannot use the `select * from <table_name>` command in a saved query. You must provide the explicit column list.
- When you read time data from a MySQL source, the synchronization task converts the time to the Coordinated Universal Time (UTC) time zone. Consider the time zone change if you create data filters and field mapping expressions based on time.
- If a task writes time data from a MySQL source to a flat file, the synchronization task converts the time data to a Date/Time data type. The date is the current date and time is the time specified in the source. You can use a string function in an expression to remove the date before loading data to the flat file.
- When you configure partitioning in a mapping to read data from MySQL, do not define a partition for the Date column. If the configured partition is for a Date column, the mapping fails because of a third-party issue.
- When you run a synchronization task to write data to a MySQL target, the Secure Agent writes one record with o value to the target even if the primary key column of the target is not connected to any source port.
- When you run a mapping task to create a target, the target is successfully created at runtime. If you drop the created target table and run the task again, the task fails.
- When you run a MySQL mapping to read from a stored procedure, ensure that the precision of the Float data type does not exceed 18.
- Consider the following rules and guidelines when you add objects with a hyphen in the column names in the Source and Target transformations:
  - When the source column name that you read has hyphens and you use the **Create New at Runtime** option to create a target, the hyphens in the table name are converted to underscores in the target column.
  - If you add a new column to the object in an existing mapping and the column name has a hyphen, you cannot manually add these columns in the mapping.
  - If you parameterize the target object, the manual field map option in the **Field Mapping** tab is not applicable. You must select Automatic, Completely Parameterized, or Partially Parameterized as the field map options for the mapping to work successfully.
- You cannot use SQL ELT optimization to push the transformation logic to MySQL.

CHAPTER 2

MySQL connections

Create a MySQL connection to connect to MySQL so that the Secure Agent can read data from and write data to MySQL.

You create a MySQL connection on the **Connections** page. Use the connection when you create the synchronization tasks, mappings, mapping tasks, replication tasks, and data transfer tasks.

MySQL connection properties

When you set up a MySQL connection, configure the connection properties.

The following table describes the MySQL connection properties:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	Type of connection. Select MySQL from the list.
Runtime Environment	The name of the runtime environment where you want to run the tasks. Specify a Secure Agent, Hosted Agent, or serverless runtime environment.
User Name	User name for the database login. The user name can't contain a semicolon.
Password	Password for the database login. The password can't contain a semicolon.
Host	Name of the machine that hosts the database server.
Port	Network port number used to connect to the database server. Default is 3306.

Property	Description
Database Name	Name of the MySQL database that you want to connect to. **Note:** The database name is case-sensitive. Maximum length is 64 characters. Database name can contain alphanumeric and underscore characters.
Code Page	The code page of the database server.

Metadata Advanced Connection Properties	Additional properties for the JDBC driver to fetch the metadata. Enter properties in the following format:
	`<parameter name>=<parameter value>`
	If you enter more than one property, separate each key-value pair with a semicolon.
	For example, enter the following property to configure the connection timeout when you test a connection:
	`connectTimeout=<value_in_miliseconds>`
	Note: The default connection timeout is 270000 miliseconds.

Runtime Advanced Connection Properties	Additional properties for the ODBC driver to run mappings. If you specify more than one property, separate each key-value pair with a semicolon.

SSL properties

You can configure a MySQL connection to use SSL to securely communicate with the MySQL database.

Note: You can enable SSL for a MySQL connection only when you use the 8.x MySQL JDBC and ODBC drivers. Ensure that both the MySQL JDBC and ODBC drivers are of 8.x version.

To configure SSL, you must first download and install the MySQL ODBC and JDBC drivers, version 8.x. For information about installing the MySQL ODBC and JDBC drivers, version 8.x, see the Knowledge Base article: 561573

After you install the drivers, in the MySQL connection properties, enable SSL and specify the TLS protocols that you want to use for the secure communication.

When you enable SSL for the MySQL connection, you must configure the SSL properties for both the MySQL JDBC and ODBC drivers. Configure the required SSL properties for the JDBC driver, so that the Secure Agent can access metadata securely from MySQL. Also, configure the required SSL properties for the ODBC driver, so that the Secure Agent runs mappings to securely read from or write data to MySQL.

Note: SSL is not applicable when you use the Hosted Agent. You can configure SSL when you use the Secure Agent or the serverless runtime environment.

The following table describes the MySQL connection SSL properties:

Connection property	Description
Use SSL	Determines whether the Secure Agent establishes a secure connection to the MySQL database. When you select this option and the database server supports SSL, the Secure Agent establishes an encrypted connection. If the MySQL database server cannot configure SSL, the connection either fails or the Secure Agent establishes an unencrypted connection depending on whether you enable or disable the **Require SSL** checkbox. If you do not select the **Use SSL** checkbox, the Secure Agent attempts to establish an unencrypted connection.
Verify Server Certificate	If you select **Use SSL** and select this option, the client validates the server certificate that is sent by the database server.
Require SSL	Applicable only if you select **Use SSL**. If you select the **Require SSL** checkbox, and the MySQL database supports SSL, the Secure Agent establishes an SSL connection. If you select the **Require SSL** checkbox, and the MySQL database cannot configure SSL, the Secure Agent attempts to establish an SSL connection but fails. If you clear the **Require SSL** checkbox, and the MySQL database cannot configure SSL, the Secure Agent establishes an unencrypted connection.
TLS Protocols	The TLS protocols used for the secure communication when you select **Use SSL**. You can select from the following protocols: - TLSv1 - TLSv1.1 - TLSv1.2 Default is TLSv1.2. The TLSv1 and TLSv1.1 protocols are not applicable.

The following table describes the MySQL connection properties for the JDBC driver version 8.x when you enable **Use SSL**:

Connection property	Description

Trust Certificate Key Store	The path and file name of the truststore file. You must prefix the file path with file colon (`file:`).
	For example, `file:C:\SSL\mysql_new\truststore`
	For the serverless runtime environment, specify the following certificate path in the serverless agent directory:
	`/home/cldagnt/SystemAgent/serverless/configurations/ssl_store/`
	`<TrustStore_filename>`

Trust Certificate Key Store Password	The password for the truststore file.

Connection property	Description

Client Certificate Key Store	The path and file name of the keystore file. You must prefix the file path with file colon (`file:`).
	For example, `file:C:\SSL\mysql_new\keystore`
	For the serverless runtime environment, specify the following certificate path in the serverless agent directory:
	`/home/cldagnt/SystemAgent/serverless/configurations/ssl_store/`
	`<KeyStore_filename>`

Client Certificate Key Store Password	The password to access the keystore file.

JDBC Cipher Suites	Colon-separated cipher suite values in RFC format.
	For example:
	
	`TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256:`
	`TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256`

The following table describes the MySQL connection properties for the ODBC driver version 8.x when you enable **Use SSL**:

Connection property	Description
SSL Certificate Authority	The path and name of the CA certificate. For example, `C:\SSL\mysql_new\ca.pem`
SSL Certificate	The path and name of the client certificate. For example, `C:\SSL\mysql_new\client-cert.pem`
SSL Key	The path and the name of the private key of the client. For example, `C:\SSL\mysql_new\client-key.pem`

SSL Cipher	Colon-separated cipher-suite values in OpenSSL format.
	For example:
	
	`ECDHE-ECDSA-AES128-GCM-SHA256:`
	`ECDHE-ECDSA-AES256-GCM-SHA384:`
	`ECDHE-RSA-AES128-GCM-SHA256:`

Verify Server's Identity	Verifies the host name in the certificate while verifying the server CA certificate. This property is applicable only when you enable **Verify Server Certificate** in the SSL properties.

CHAPTER 3

Synchronization tasks with MySQL Connector

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

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

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

MySQL sources in synchronization tasks

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

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

The following table describes the MySQL source properties:

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

MySQL targets in synchronization tasks

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

The following table describes the MySQL target properties:

Property	Description
Connection	Name of the MySQL target connection.
Target Object	Name of target objects available in the connection.
Truncate Target	When you enable the **Truncate Target** option, the Secure Agent truncates the table before it writes the data to the target. Default is false.
Display target fields in alphabetical order	Displays target fields in alphabetical order. By default, fields appear in the order returned by the target system.

Advanced properties for MySQL sources and targets in synchronization tasks

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

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

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

MySQL lookups in synchronization tasks

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

CHAPTER 4

Mappings and mapping tasks with MySQL Connector

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

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

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

MySQL sources in mappings

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

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

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

The following table describes the source properties that you can configure for a MySQL source:

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

Property	Description
Parameter	A parameter file where you define values that you want to update without having to edit the task. Select an existing parameter for the source object or click **New Parameter** to define a new parameter for the source object. The **Parameter** property appears only if you select parameter as the source type. If you want to overwrite the parameter at runtime, select the **Allow parameter to be overridden at run time** option. When the task runs, the Secure Agent uses the parameters from the file that you specify in the advanced session properties.
Object	Name of the MySQL source object.
Filter	Configure a simple filter or an advanced filter to remove rows at the source. You can improve efficiency by filtering early in the data flow. A simple filter includes a field name, operator, and value. Use an advanced filter to define a more complex filter condition, which can include multiple conditions using the AND or OR logical operators.
Sort	Select the fields and type of sorting to use. To sort data for a parameterized source, you must use a parameter for the sort options.
Select distinct rows	Select this option to read only distinct rows.

The following table describes the advanced source properties that you can configure for a MySQL source:

Property	Description
Tracing level	Amount of detail that appears in the log for this transformation. You can choose terse, normal, verbose initialization, or verbose data. Default is normal.
Pre SQL	Pre-SQL command to run before reading data from the source.
Post SQL	Post-SQL command that must be run after reading data from the source.
SQL Override	The SQL statement to override the default query generated to read data from the MySQL source. Ensure that the list of selected columns, data types, and the order of the columns that appear in the SQL override query matches the columns, data types, and order in which they appear in the source object.
Output is Deterministic	Relational source or transformation output that does not change between session runs when the input data is consistent between runs. When you configure this property, the Secure Agent does not stage source data for recovery if transformations in the pipeline always produce repeatable data.
Output is repeatable	Relational source or transformation output that is in the same order between session runs when the order of the input data is consistent. When output is deterministic and output is repeatable, the Secure Agent does not stage source data for recovery.

Key range partitioning

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

Use key range partitioning for columns that have an even distribution of data values. Otherwise, the partitions might have unequal size. For example, a column might have 10 rows between key values 1 and 1000 and the column might have 999 rows between key values 1001 and 2000. If the mapping includes multiple sources, use the same number of key ranges for each source.

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

Consider the following rules when you configure partitioning:

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

Configuring key range partitioning

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

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

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

MySQL targets in mappings

To write data to a MySQL database, configure a MySQL object as the Target transformation in a mapping.

Specify the name and description of the MySQL target. Configure the target and advanced properties for the target object.

The following table describes the target properties that you can configure for a MySQL target:

Property	Description
Connection	Name of the target connection. You can select an existing connection, create a new connection, or define parameter values for the target connection property. If you want to overwrite the target connection properties at runtime, select the **Allow parameter to be overridden at run time** option.
Target Type	Type of the MySQL target object available. You can choose from the following source types: - Single - Parameter
Parameter	A parameter file where you define values that you want to update without having to edit the task. Select an existing parameter for the target object or click **New Parameter** to define a new parameter for the target object. The **Parameter** property appears only if you select parameter as the target type. If you want to overwrite the target object at runtime, select the **Allow parameter to be overridden at run time** option. When the task runs, the Secure Agent uses the parameters from the file that you specify in the advanced session properties.
Object	Select an existing object from the list or create a new object at run time.
Operation	Select the target operation. You can perform the following operations on a MySQL target: - Insert - Update - Upsert - Delete - Data Driven
Truncate target	When you enable the **Truncate Target** option, the Secure Agent truncates the table before running the task. Default is not selected.
Update Override	An update SQL statement that updates the data in a MySQL target table. The update SQL statement you specify overrides the default update statements that the Secure Agent generates to update the target based on key columns. You can define an update override statement to update target tables based on both key or non-key columns. In the override statement, you must enclose all reserved words in quotation marks.

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

Configuring an update override for the target

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

1. Next to the **Update Override** field, click **Configure**.
2. In the **Update Override SQL Editor** dialog box, enter the update SQL statement that the Secure Agent must use.
3. Click **Generate SQL** to generate an SQL query.
4. Click **Format SQL** to format the SQL query you entered.

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

5. Click **OK**.

MySQL lookups in mapping

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

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

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

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

Property	Description
Connection	Name of the lookup connection.
	You can select an existing connection, create a new connection, or define parameter values for the lookup connection property.
	If you want to overwrite the lookup connection properties at runtime, select the **Allow parameter to be overridden at run time** option.
	Specify the parameter file directory and name in the advanced session properties.
Source Type	Type of the MySQL lookup object available. You can choose from the following lookup object types:
	- Single Object
	- Parameter
Parameter	A parameter file where you define values that you want to update without having to edit the task. Select an existing parameter for the lookup object or click **New Parameter** to define a new parameter for the lookup object.
	The **Parameter** property appears only if you select parameter as the source type.
	If you want to overwrite the parameter at runtime, select the **Allow parameter to be overridden at run time** option.
	When the task runs, the Secure Agent uses the parameters from the file that you specify in the advanced session properties.
Lookup Object	Name of the MySQL lookup object.
Multiple Matches	The behavior when the lookup condition returns multiple matches.
	You can select one of the following options:
	- Return first row
	- Return last row
	- Return any row
	- Return all rows
	- Report error

Calling a stored procedure

You can use the SQL transformation to call a stored procedure in a MySQL database or to process SQL queries midstream in a pipeline.

Before you configure stored procedures, you must specify the following property in the **Runtime Advanced Connection Properties** in the MySQL connection for stored procedures to work: `NO_SSPS=1`

Note: Informatica recommends that you create a separate MySQL connection and specify the property when you configure stored procedures.

You can then configure the SQL transformation to process the following types of SQL statements:
Stored procedure

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

SQL Query

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

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

Rules and guidelines for calling a stored procedure

Consider the following rules and guidelines for calling a stored procedure:

- You can't define a stored procedure that contains Unicode characters.
- You can't configure an unconnected stored procedure using the SQL transformation.
- You can't process a stored function in an SQL transformation.
- You can't configure the input or in-out parameter in an entered query that you define in the SQL editor.

CHAPTER 5

Replication tasks with MySQL

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

A replication task can replicate data from one or more MySQL objects. When you configure the task, you can replicate all available objects through the selected connection, or you can select objects for replication by including or excluding a set of objects.

You can also exclude rows and columns from the replication task. Associate a schedule with a replication task to specify when and how often the task runs.

Note: The length of source tables and column names can contain at most 128 characters.

MySQL sources in replication tasks

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

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

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

MySQL targets in replication tasks

When you replicate data to a database target, the replication task truncates the table name in the target database if the source object name exceeds 128 characters.

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

Property	Description
Connection	Name of the target connection. Select a target connection or create a connection.
Target Prefix	String that prefixes the source object names to create names for the target objects in the target. **Note:** For more information, see "Truncating Table Names" in the help for *Tasks*.
Load Type	Replicates full load of data from MySQL to other applications.
Delete Options	This property is not applicable for MySQL Connector.

You can also configure the commit size, which defines the number of rows to commit. If you do not specify a value, the Secure Agent uses the default value. Default is 10000.

CHAPTER 6

Troubleshooting

Use the following sections to troubleshoot errors in MySQL Connector.

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

Troubleshooting a replication task

A data overflow error occurs when I run a replication task that writes MySQL data to a flat file.

When a replication task writes MySQL data with the BigInt data type to a flat file and the data length is greater than 10, a data overflow error occurs because the replication application converts BigInt to Int (10). To prevent the data overflow error, change the BigInt source data type to Decimal.

Data type reference

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

MySQL native data types

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

Transformation data types

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

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

MySQL and transformation data types

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

MySQL Data Type	Transformation Data Type	Description
Char	String	1 to 104,857,600 characters
Varchar	String	1 to 104,857,600 characters
Boolean	Integer	-2,147,483,648 to 2,147,483,647 (Precision 10, scale 0)
TinyInt	Integer	-2,147,483,648 to 2,147,483,647 (Precision 10, scale 0)
SmallInt	Integer	-2,147,483,648 to 2,147,483,647 (Precision 10, scale 0)
MediumInt	Integer	-2,147,483,648 to 2,147,483,647 (Precision 10, scale 0)
Int	Integer	-2,147,483,648 to 2,147,483,647 (Precision 10, scale 0) **Note:** Unsigned Int is not applicable.

MySQL Data Type	Transformation Data Type	Description
Bigint	Bigint	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807 (Precision 19, scale 0) **Note:** Unsigned Bigint is not applicable.
Float	Decimal	Precision 1 to 28, scale 0 to 28
Double	Double	Precision 15 **Note:** Unsigned Double is not applicable.
Decimal	Decimal	Precision 1 to 28, scale 0 to 28
Date	Date/Time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. (Precision to the nanosecond)
Datetime	Date/Time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. (precision to the nanosecond)
Timestamp	Date/Time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. (precision to the nanosecond)

INDEX