

IBM InfoSphere Information Server  
Version 11 Release 3

*IBM InfoSphere Information Server  
Connectivity Guide for Accessing  
Greenplum Databases*





IBM InfoSphere Information Server  
Version 11 Release 3

*IBM InfoSphere Information Server  
Connectivity Guide for Accessing  
Greenplum Databases*



**Note**

Before using this information and the product that it supports, read the information in “Notices and trademarks” on page 35.

---

# Contents

<b>Greenplum databases</b> . . . . .	<b>1</b>
Configuring access to Greenplum databases . . . . .	1
Configuring ODBC access to Greenplum databases on Windows . . . . .	1
Greenplum parallel file distribution program (gpfdist) . . . . .	2
Designing jobs with Greenplum Connector stages . . . . .	2
Importing Greenplum metadata . . . . .	3
Defining a job that includes a Greenplum Connector stage . . . . .	3
Connecting to a Greenplum data source . . . . .	4
Reading data from a Greenplum database . . . . .	4
Writing data by using the Greenplum Connector stage . . . . .	6
Looking up data in a Greenplum database . . . . .	7
Reference . . . . .	10
Properties for the Greenplum connector . . . . .	10
Write modes . . . . .	16
Parallel execution . . . . .	18
Data type mappings from InfoSphere DataStage to Greenplum . . . . .	19
Troubleshooting the Greenplum Connector stage . . . . .	20
The connector was not able to start the gpfdist program . . . . .	20
Connection with the gpfdist program fails . . . . .	20
Data truncation with Greenplum geometric or multiple format data types . . . . .	21
External table has more gpfdist instances than available primary segments . . . . .	21
Jobs using key columns fail with ERROR: operator does not exist . . . . .	22
Jobs fail with permission error . . . . .	22
<b>Appendix A. Product accessibility</b> . . . . .	<b>23</b>
<b>Appendix B. Reading command-line syntax</b> . . . . .	<b>25</b>
<b>Appendix C. How to read syntax diagrams</b> . . . . .	<b>27</b>
<b>Appendix D. Contacting IBM</b> . . . . .	<b>29</b>
<b>Appendix E. Accessing the product documentation</b> . . . . .	<b>31</b>
<b>Appendix F. Providing feedback on the product documentation</b> . . . . .	<b>33</b>
<b>Notices and trademarks</b> . . . . .	<b>35</b>
<b>Index</b> . . . . .	<b>41</b>



---

## Greenplum databases

Use the Greenplum connector to access Greenplum databases.

Use the Greenplum connector to perform the following operations:

- Read data from or write data to or look up data in Greenplum databases.
- Import metadata from Greenplum databases by using InfoSphere Metadata Asset Manager (IMAM).

### Related tasks:

“Configuring access to Greenplum databases”

You can configure access to a Greenplum database by configuring an ODBC data source definition (DSN) for Greenplum and by adding the directory location of the Greenplum parallel file distribution program (gpfdist) to the system path. The Greenplum Connector stage uses ODBC to connect and to execute statements and uses the gpfdist program to exchange data with the Greenplum database.

### Related reference:

“Troubleshooting the Greenplum Connector stage” on page 20

Several common errors are specific to the Greenplum Connector stage.

---

## Configuring access to Greenplum databases

You can configure access to a Greenplum database by configuring an ODBC data source definition (DSN) for Greenplum and by adding the directory location of the Greenplum parallel file distribution program (gpfdist) to the system path. The Greenplum Connector stage uses ODBC to connect and to execute statements and uses the gpfdist program to exchange data with the Greenplum database.

## Configuring ODBC access to Greenplum databases on Windows

To connect to a Greenplum database, you must first configure an ODBC data source definition (DSN) for the database by using the IBM Greenplum Wire Protocol ODBC driver.

### Before you begin

- Ensure that the ODBC driver for Greenplum libraries is installed.

### Procedure

1. Start the Microsoft ODBC Data Source Administrator.
  - On a 32-bit Windows computer, click **Start > Control panel > Administrative Tools > Data Sources (ODBC)**
  - On a 64-bit Windows computer, navigate to `C:\Windows\SysWOW64\odbcad32.exe`.

**Note:** On Windows, InfoSphere® Information Server is a 32-bit application. Even on a 64-bit Windows computer, the connector is running as a 32-bit application. Therefore you must use the 32-bit version of the ODBC Data Source Administrator as the Greenplum connector will not be able to locate DSN definitions created in the 64-bit ODBC Data Source Administrator.

2. On the System DSN page, click **Add**.

3. On the Create New Data Source page, select the IBM Greenplum Wire Protocol driver and click **Finish**. For information about configuring the driver options, see the Greenplum Wire Protocol Driver chapter in the *DataDirect Connect Series for ODBC User's Guide*.

## Greenplum parallel file distribution program (gpfdist)

The Greenplum Connector stage exchanges data with the Greenplum server by using the Greenplum file distribution program, which is called gpfdist.

The gpfdist program runs on the database client, and it must be installed on the InfoSphere Information Server engine tier computer. For data to be transferred by using the gpfdist protocol, a network route must be present to enable bidirectional access by using an IP address and optionally the presence of a DNS server to facilitate the name resolution. The connector invokes a gpfdist process on every physical computer node and creates the external table. The host of the external table data is identified by the fastname entry in the parallel engine configuration file (\$APT\_CONFIG\_FILE). In order for the connector to invoke gpfdist on each engine tier, the location of gpfdist(%GPHOME\_LOADERS%\bin) must be in the system path. In addition, the location of gpfdist dependent libraries (%GPHOME\_LOADERS%\lib) must be in the system library path. On Windows, the system environment variable *PATH* is updated in the Advanced system settings. On Linux, the *PATH* environment variable is updated in the dsenv script.

**Note:** On Windows, the Greenplum installer adds %GPHOME\_LOADERS%\bin and %GPHOME\_LOADERS%\lib to the *PATH* system environment variable. Verify that these directories are in the *PATH*.

For manually adding %GPHOME\_LOADERS%\bin and %GPHOME\_LOADERS%\lib to the *PATH* system environment variable see the topic on setting the library path environment variable.

---

## Designing jobs with Greenplum Connector stages

You can use Greenplum Connector stages in your jobs to read data from Greenplum databases or write data to Greenplum databases or look up data in the contexts of those jobs.

### Procedure

1. Define a job that includes a Greenplum Connector stage.
2. Define a connection to a Greenplum data source.
3. To set up the Greenplum Connector stage as a source stage to read data from the Greenplum database, complete the following steps:
  - a. Configure the Greenplum Connector stage as a source.
  - b. Set up column definitions on the output link.
4. To set up the Greenplum Connector stage as a target stage to write data to the Greenplum database, complete the following steps:
  - a. Configure the Greenplum Connector stage as a target.
  - b. Set up column definitions on the input link.
5. To set up the Greenplum Connector stage to look up data in a Greenplum database, complete the following step:
  - a. Configure normal lookup operations or configure sparse lookup operations.

**Related concepts:**

“Greenplum databases,” on page 1  
Use the Greenplum connector to access Greenplum databases.

## Importing Greenplum metadata

Before you use the Greenplum connector to read, write, or look up data, you can use InfoSphere® Metadata Asset Manager to import the metadata that represents tables and views in a Greenplum database. The imported metadata is then saved in the metadata repository.

### Before you begin

- Configure the ODBC access to Greenplum databases on Linux and UNIX or on Windows as required.
- Ensure that you have the SELECT privilege for the following system schemas:
  - pg\_catalog
  - information\_schema

### About this task

By using the Greenplum connector, you can import metadata about the following types of assets:

- The host computer that contains the Greenplum database.
- The database.
- Database schemas.
- Database tables, system tables, external tables and views. All imported tables are stored in the metadata repository as database tables.
- Database columns.

### Procedure

Import metadata by using InfoSphere Metadata Asset Manager. For more information about importing metadata by using InfoSphere Metadata Asset Manager, see the online product documentation in IBM Knowledge Center or the IBM® InfoSphere Information Server Guide to Managing Common Metadata.

## Defining a job that includes a Greenplum Connector stage

Before you can read, write, or look up data from a Greenplum data source, you must create a job that includes a Greenplum Connector stage. Then, you add any additional stages that are required and create the necessary links.

### Procedure

1. In the InfoSphere DataStage® and QualityStage® Designer client, select **File > New** from the menu.
2. In the **New** window, select the **Parallel Job** icon, and then click **OK**.
3. Add the Greenplum Connector stage to the job:
  - a. In the palette, select the **Database** category.
  - b. From the list of available stage types, drag the **Greenplum Connector stage** to the canvas.
  - c. Optional: Rename the Greenplum Connector stage. Choose a name that indicates the role of the stage in the job.
4. Create the necessary links and additional stages for the job:

- For a job that reads data from a Greenplum database, create the next stage in the job, and then create an output link from the Greenplum Connector stage to the next stage.
  - For a job that writes data to a Greenplum database, create one or more links from other stages in the job to the Greenplum Connector stage.
  - For a job that looks up Greenplum data source, add a Lookup stage to the job design canvas, and then create a reference link from the Greenplum Connector stage to the Lookup stage.
5. Save the job.

## Connecting to a Greenplum data source

To access Greenplum data sources, you must define a connection by using the properties in the **Connection** section on the Properties page.

### Before you begin

- Create the ODBC driver and data source definition to Greenplum.
- Define a job that includes a Greenplum Connector stage.

### Procedure

1. To open the stage editor, on the job design canvas, double-click the **Greenplum Connector** stage icon.
2. In the **Data source** property, specify the ODBC data source name (DSN) as defined in the ODBC driver manager or the `odbc.ini` file for a Greenplum database.
3. Optional: In the **Database** property, specify the name of the Greenplum database on the server that is defined in the ODBC data source.
4. In the **User Name** and **Password** properties, specify the credentials of the user to use for the connection.
5. Click **Save** to save the details.

## Reading data from a Greenplum database

You can configure a Greenplum Connector stage to connect to a Greenplum data source and read data from it.

### Before you begin

- Define a job that contains a Greenplum Connector stage.
- Define a connection to a Greenplum data source.

### About this task

The following figure shows an example of using the Greenplum Connector stage to read data. In this example, the Greenplum Connector stage reads data from a Greenplum data source, and the Sequential File stage writes the data to a file. When you configure the Greenplum Connector stage to read data, you create an output link, which in this example transfers rows from the Greenplum Connector stage to the Sequential File stage.

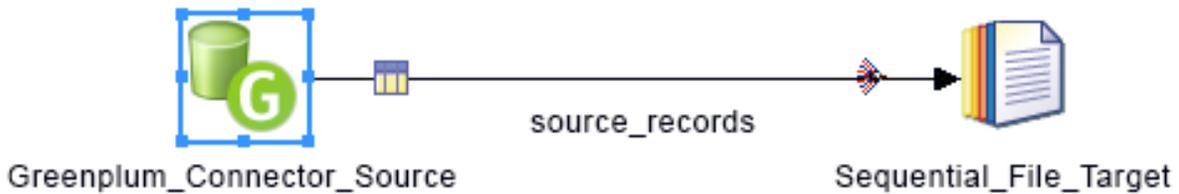


Figure 1. Example of using the Greenplum Connector stage to read data from a data source

## Configuring the Greenplum connector as a source

To configure a Greenplum Connector stage to read or look up rows in a Greenplum table or view, you must specify the source table or view.

### Procedure

1. From the job design canvas, double-click the Greenplum Connector stage.
2. Click the **Output** tab.
3. On the Properties page, in the **Usage** section specify the settings for the read operation.
4. Enter the name of the target table in the **Table name** field. Use the syntax *table\_name* or *schema\_name.table\_name* or *database\_name.schema\_name.table\_name*, where *schema\_name* is the owning schema of the table and *database\_name* is the owning database of the schema and table. When *schema\_name* is not specified, the connector uses the default schema of the currently connected user.
5. Click **OK**, and then save the job.

## Setting up column definitions on a link

Column definitions, which you set on a link, specify the format of the data records that the connector reads from a database or writes to a database.

### Procedure

1. From the job design canvas, double-click the connector icon.
2. Use one of the following methods to set up the column definitions:
  - Drag a table definition from the repository view to the link on the job canvas. Then, use the arrow buttons to move the columns between the **Available columns** and **Selected columns** lists.
  - On the **Columns** page, click **Load** and select a table definition from the metadata repository. Then, to choose which columns from the table definition apply to the link, move the columns from the **Available columns** list to the **Selected columns** list.
3. Configure the properties for the columns:
  - a. Right-click within the columns grid, and select **Properties** from the menu.
  - b. Select the properties to display, specify the order in which to display them, and then click **OK**.
4. Optional: Modify the column definitions. You can change the column names, data types, and other attributes. In addition, you can add, insert, or remove columns.
5. Optional: Save the new table definition in the metadata repository:

- a. On the **Columns** page, click **Save**, and then click **OK** to display the repository view.
- b. Navigate to an existing folder, or create a new folder in which to save the table definition.
- c. Select the folder, and then click **Save**.

## Writing data by using the Greenplum Connector stage

You can configure a Greenplum Connector stage to connect to a Greenplum data source and write data to it.

### Before you begin

- Define a job that contains a Greenplum Connector stage.
- Define a connection to a Greenplum data source.

### About this task

The following figure shows an example of using the Greenplum connector to write data. In this example, the Sequential File stage reads data from a file and then the Greenplum Connector stage writes data to the Greenplum data source.

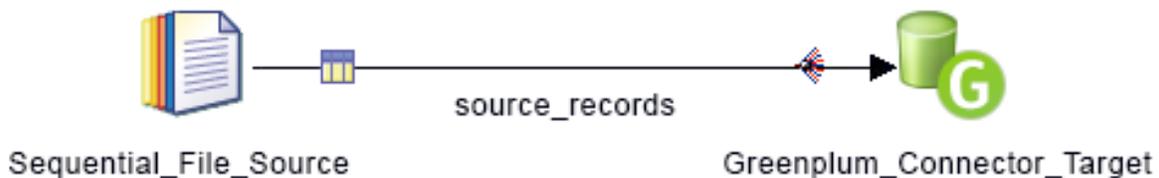


Figure 2. Example of using the Greenplum Connector stage to write data to a data source

### Configuring the Greenplum connector as a target

To configure a Greenplum Connector stage to write rows to a Greenplum table, you must specify the target table or view or define the SQL statements.

#### Procedure

1. On the job design canvas, double-click the Greenplum Connector stage icon.
2. Select the input link to edit.
3. Select the **Write mode** to define the type of SQL the Greenplum connector stage must execute.
4. Enter the name of the target table in the **Table name** field. Use the syntax *table\_name* or *schema\_name.table\_name* or *database\_name.schema\_name.table\_name*, where *schema\_name* is the owning schema of the table and *database\_name* is the owning database of the schema and table. When *schema\_name* is not specified, the connector uses the default schema of the currently connected user.
5. Click **OK**, and then save the job.

#### Related reference:

“Write modes” on page 16

When you configure the Greenplum connector as a target, you can use the **Write mode** property to specify the mode to use to write rows to the Greenplum

database.

## Setting up column definitions on a link

Column definitions, which you set on a link, specify the format of the data records that the connector reads from a database or writes to a database.

### Procedure

1. From the job design canvas, double-click the connector icon.
2. Use one of the following methods to set up the column definitions:
  - Drag a table definition from the repository view to the link on the job canvas. Then, use the arrow buttons to move the columns between the **Available columns** and **Selected columns** lists.
  - On the **Columns** page, click **Load** and select a table definition from the metadata repository. Then, to choose which columns from the table definition apply to the link, move the columns from the **Available columns** list to the **Selected columns** list.
3. Configure the properties for the columns:
  - a. Right-click within the columns grid, and select **Properties** from the menu.
  - b. Select the properties to display, specify the order in which to display them, and then click **OK**.
4. Optional: Modify the column definitions. You can change the column names, data types, and other attributes. In addition, you can add, insert, or remove columns.
5. Optional: Save the new table definition in the metadata repository:
  - a. On the **Columns** page, click **Save**, and then click **OK** to display the repository view.
  - b. Navigate to an existing folder, or create a new folder in which to save the table definition.
  - c. Select the folder, and then click **Save**.

## Looking up data in a Greenplum database

You can use the Greenplum connector to look up data from a Greenplum table by using a reference link to link the Greenplum Connector stage to a Lookup stage.

### About this task

A *reference link* represents a table lookup operation. You can use a reference link as an input link to a Lookup stage and as an output link from other types of stages, such as the Greenplum Connector stage.

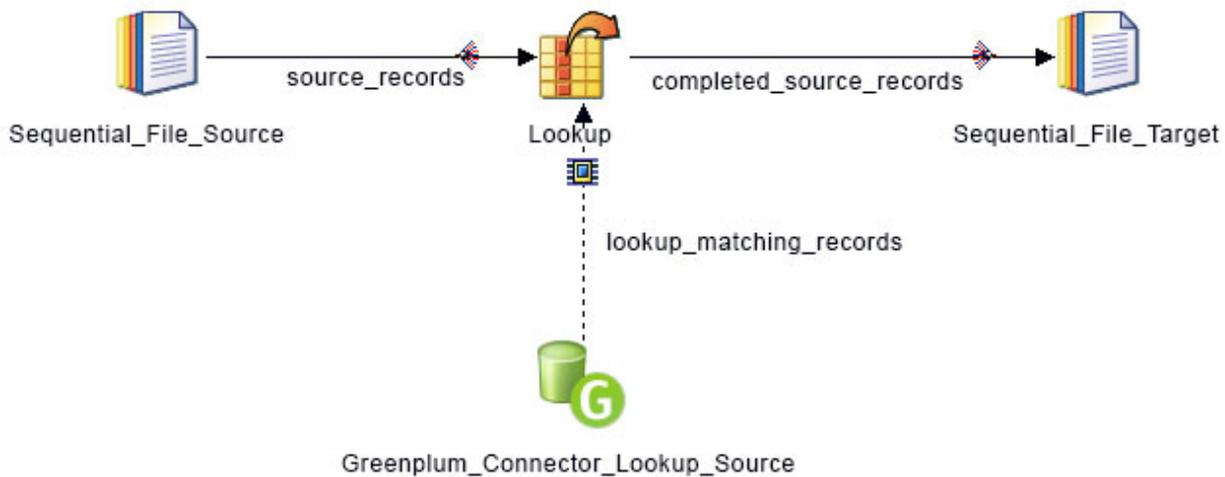


Figure 3. Example of using the Greenplum Connector stage with a Lookup stage.

### Configuring normal lookup operations

You can configure a Greenplum Connector stage to retrieve a set of records from a Greenplum database and provide them to a Lookup stage in the same job. The Lookup stage then performs a normal (in-memory) lookup operation on those records.

#### Before you begin

- Set up the column definitions on a link to specify the format of the records that the Greenplum Connector stage reads from a Greenplum server.
- Configure the Greenplum Connector stage as a source for the reference data.
- Add a Lookup stage to the job design canvas, and then create a reference link from the Greenplum Connector stage to the Lookup stage.

#### About this task

In a normal lookup, the connector runs the specified SELECT statement only one time; therefore, the SELECT statement cannot include any input parameters. The Lookup stage searches the result set data that is provided by the connector and looks for matches for the parameter sets that arrive in the form of records on the input link to the Lookup stage. A normal lookup is also known as an in-memory lookup because the lookup is performed on the cached data in memory. You can use a normal lookup when the target table is small enough that all of the rows in the table can fit in memory.

#### Procedure

1. Double-click the Greenplum Connector stage.
2. From the **Lookup Type** list, select **Normal**, and then click **OK**.
3. Double-click the Lookup stage.

4. To specify the key columns, drag the required columns from the input link to the reference link. The columns from the input link contain values that are used as input values for the lookup operation.
5. Map the input link and reference link columns to the output link columns and specify conditions for a lookup failure:
  - a. Drag or copy the columns from the input link and reference link to your output link.
  - b. To define conditions for a lookup failure, click the **Constraints** icon in the menu.
  - c. In the Lookup Failure column, select a value, and then click **OK**. If you select **Reject**, you must have a reject link from the Lookup stage and a target stage in your job configuration to capture the rejected records.
  - d. Click **OK**.
6. Save, compile, and run the job.

**Related tasks:**

“Configuring the Greenplum connector as a source” on page 5

To configure a Greenplum Connector stage to read or look up rows in a Greenplum table or view, you must specify the source table or view.

“Setting up column definitions on a link” on page 5

Column definitions, which you set on a link, specify the format of the data records that the connector reads from a database or writes to a database.

## Configuring sparse lookup operations

You can configure a Greenplum Connector stage to perform a sparse (direct) lookup operation on a Greenplum table.

### Before you begin

- Set up the column definitions on a link to specify the format of the records that the Greenplum Connector stage reads from a Greenplum server.
- Configure the Greenplum Connector stage as a source for the reference data.
- Add a Lookup stage to the job design canvas, and then create a reference link from the Greenplum Connector stage to the Lookup stage.

### About this task

In a sparse lookup, the connector runs the specified SELECT statement one time for each parameter set that arrives in the form of a record on the input link to the Lookup stage. The specified input parameters in the statement must have corresponding columns defined on the reference link. Each input record includes a set of parameter values that are represented by key columns. The Greenplum Connector stage sets the parameter values on the bind variables in the SELECT statement, and then the Greenplum Connector stage runs the statement.

The result of the lookup is routed as one or more records through the reference link from the Greenplum Connector stage back to the Lookup stage and from the Lookup stage to the output link of the Lookup stage. A sparse lookup is also known as a direct lookup because the lookup is performed directly on the data source.

You can use the sparse lookup method when the target table is too large to fit in memory.

## Procedure

1. Double-click on the Greenplum Connector stage.
2. From the **Lookup Type** list, select **Sparse**.
3. On the Columns tab, define the key columns to use from the database to which the connector is connected.
4. On the Properties page, configure the properties on the **Properties** tab.
  - a. If you set the **Generate SQL** to **Yes**, specify the **Table name** and the Key columns details in the Columns page.
  - b. If you set the **Generate SQL at runtime** to **No** when you configured the connector as a source, specify a value for the **Select statement** property. In the select part of the SELECT statement, list the columns to return to the Lookup stage. Ensure that the columns in the select list match the columns on the reference link. Each column name in the WHERE clause of the statement must begin with the word *ORCHESTRATE* and a period. The word *ORCHESTRATE* can be all uppercase or all lowercase letters. For example, you can specify the following SELECT statement: `select Field002,Field003 from MY_TABLE where Field001 = ORCHESTRATE.Field001`. The column names that follow the word *ORCHESTRATE* should match the key columns on the reference link.
5. Click **OK** to save the changes.
6. Map the input link and reference link columns to the output link columns and specify conditions for a lookup failure:
  - a. Drag or copy the columns from the input link and reference link to your output link.
  - b. To define conditions for a lookup failure, click the **Constraints** icon in the menu.
  - c. In the Lookup Failure column, select a value, and then click **OK**. If you select **Reject**, you must have a reject link from the Lookup stage and a target stage in your job configuration to capture the rejected records.
  - d. Click **OK**.
7. Save, compile, and run the job.

### Related tasks:

“Configuring the Greenplum connector as a source” on page 5

To configure a Greenplum Connector stage to read or look up rows in a Greenplum table or view, you must specify the source table or view.

“Setting up column definitions on a link” on page 5

Column definitions, which you set on a link, specify the format of the data records that the connector reads from a database or writes to a database.

---

## Reference

To use the Greenplum connector successfully, you might need more information. The following section covers in detail specific functional features and usage scenarios of the Greenplum connector.

### Properties for the Greenplum connector

Use these options to manage how the connector reads and writes data.

#### Run before and after SQL statements property

Use the **Run before and after SQL statements** property to configure the connector to run semicolon-separated SQL statements before or after processing data. You can

configure the connector to run SQL statements before or after processing any data in a job or to run SQL statements once before or after processing the data on each node.

## Usage

Running SQL statements before or after processing data is useful when you need to perform operations that prepare data source objects for data access. For example, you might use an SQL statement to create a target table and add an index to it. The SQL statements that you specify are performed once for the whole job, before any data is processed.

After the connector runs the statements that are specified in the **Before SQL statement** property or **After SQL statement** property, the connector explicitly commits the current transaction. For example, if you specify a DML statement such as INSERT in the **Before SQL statement** property, the results of the DML statement are visible to individual nodes.

To run SQL statements on each node that the connector is configured to run on, use the **Before SQL (node) statement** property or the **After SQL (node) statement** property. The connector runs the specified SQL statements once before any data is processed on each node or once after any data is processed on each node. Then, the connector explicitly commits the current transaction.

When you specify the statement to run before or after processing, enter the SQL statements. Do not include input bind variables or output bind variables in the SQL statement. If the statement contains these types of variables, the connector logs an error message, and the operation stops.

If you specify a file name, the file must be on the computer where the InfoSphere<sup>®</sup> Information Server engine tier is installed, and you must set the corresponding **Read statement from file** property to **Yes**.

When the connector runs a set of statements that are specified in any of the **Before SQL statement**, **Before SQL (node) statement**, **After SQL (node) statement**, or **After SQL statement** properties, it reports an error and stops the job if any of the statements in the set fail. You can configure the connector to report a warning or informational message and continue processing the remaining statements in the set. To configure the connector to continue the job when a statement fails, set the **Fail on error** property for the **Before SQL statement**, **Before SQL (node) statement**, **After SQL statement**, or **After SQL (node) statement** property to **No**. When a statement fails and the **Fail on error** property is set to **No**, you can use the **Log statement error as** property to define the severity of the error message to log.

The order of the SQL statements that the connector runs before or after processing data differs from the normal operation when the write mode is insert and a staging table is not used. For these jobs, the connector writes directly to the target table, and the statements are run in the following order:

1. The statements that are specified for the **Before SQL statement** property
2. On each node, the statements that are specified for the **Before SQL (node) statement** property
3. The INSERT statements that write to the target table
4. On each node, the statements that are specified for the **After SQL (node) statement** property
5. The statements that are specified for the **After SQL statement** property

However, when the **Staging table mode** property is set to **Automatic** or **Existing**, the connector writes to the target table after the statements that are specified for the **After SQL (node) statement** property are run. The statements are run in the following order:

1. The statements that are specified for the **Before SQL statement** property
2. On each node, the statements that are specified for the **Before SQL (node) statement** property
3. The INSERT statements that write to the staging table
4. On each node, the statements that are specified for the **After SQL (node) statement** property
5. The statements that write from the staging table to the target table
6. The statements that are specified for the **After SQL statement** property

### Enable case-sensitive identifiers property

To maintain the case-sensitivity of Greenplum object names, you can manually enter double quotation marks around each name or set the **Enable case-sensitive identifiers** property to **Yes**.

### Usage

The Greenplum connector automatically generates and runs SQL statements. The SQL statements that are generated contain the names of the columns and the name of the table on which to complete the operation. The table name that is used in each SQL statement matches the table that is specified in the **Table name** property.

By default, the Greenplum database converts all object names to lowercase before it matches the names against the Greenplum object names in the database. If the names of the Greenplum objects are all lowercase, the way that you specify the names in the connector properties does not affect schema matching. However, if the Greenplum object names use all uppercase or mixed case, you must specify the names exactly as they are specified in the Greenplum schema. In addition, you must enclose each name in double quotation marks or set the **Enable case-sensitive identifiers** property to **Yes**.

### Table action property

Use the **Table action** property to configure the connector to complete create, replace, and truncate actions on a table at run time. These actions are completed before any data is written to the table.

### Usage

You can set the **Table action** property to the values that are listed in the following table.

*Table 1. Values of the **Table action** property*

Value	Description
Append	No action is completed on the table. This option is the default.

Table 1. Values of the **Table action** property (continued)

Value	Description
Create	<p>Create a table at run time.</p> <p>Use one of these methods to specify the CREATE TABLE statement:</p> <ul style="list-style-type: none"> <li>• Set <b>Generate create table statement at runtime</b> to <b>Yes</b> and enter the name of the table to create in the <b>Table name</b> property. In this case, the connector automatically generates the CREATE TABLE statement from the column definitions on the input link. The column names in the new table match the column names on the link. The data types of columns in the new table are mapped to the column definitions on the link.</li> <li>• Set <b>Generate create table statement at runtime</b> to <b>No</b>, and enter the CREATE TABLE statement in the <b>Create table statement</b> property.</li> </ul>
Replace	<p>Replace a table at run time.</p> <p>Use one of these methods to specify the DROP TABLE statement:</p> <ul style="list-style-type: none"> <li>• Set <b>Generate drop table statement at runtime</b> to <b>Yes</b>, and enter the name of the table to drop in the <b>Table name</b> property.</li> <li>• Set <b>Generate drop table statement at runtime</b> to <b>No</b>, and enter the DROP TABLE statement in the <b>Drop table statement</b> property.</li> </ul> <p>Use one of these methods to specify the CREATE TABLE statement:</p> <ul style="list-style-type: none"> <li>• Set <b>Generate create table statement at runtime</b> to <b>Yes</b>, and enter the name of the table to create in the <b>Table name</b> property.</li> <li>• Set <b>Generate create table statement at runtime</b> to <b>No</b>, and enter the CREATE TABLE statement in the <b>Create table statement</b> property.</li> </ul>
Truncate	<p>Truncate a table at run time.</p> <p>Use one of these methods to specify the TRUNCATE TABLE statement:</p> <ul style="list-style-type: none"> <li>• Set <b>Generate truncate table statement at runtime</b> to <b>Yes</b>, and enter the name of the table to truncate in the <b>Table name</b> property.</li> <li>• Set <b>Generate truncate table statement at runtime</b> to <b>No</b>, and enter the TRUNCATE TABLE statement in the <b>Truncate table statement</b> property.</li> </ul>

To configure the job to fail when the statement that is specified by the table action fails, you can set the appropriate **Fail on error** property for each table action to **Yes**. Otherwise, when the statement fails, the connector logs a message, and the job continues.

## Greenplum data types

When you use the table action property to create a table that uses Greenplum data types that are not directly supported by InfoSphere DataStage (for example, circle, polygon, XML), you must complete one of the following tasks:

- Set the **Generate create table statement at runtime** property to **No**, and then enter an SQL statement that contains the Greenplum data types.
- Set the **Generate create table statement at runtime** property to **Yes**. To enable the Native type attribute, on the column's page in the Greenplum Connector stage, right-click in the column area, and then click **Properties**. Then, select the check box for the Native type. Then, on the Columns page, specify a Greenplum data type as the value for the **Native type**.

Column name	Key	SQL type	Extended	Length	Scale	Nullable	Data element	Native type	Description
1 col1	<input type="checkbox"/>	VarChar		32		No			
2 col2	<input type="checkbox"/>	VarChar		32		No			
3 col3	<input checked="" type="checkbox"/>	VarChar		32		No		circle	

Figure 4. Example of using the connector to create a circle column in the table

If you have a VarChar column on your input link, the connector creates a VarChar (character varying) column of the Greenplum data type in the table. For example, if you set the **Native type** to **circle**, the connector creates a circle column in the table.

```
CREATE TABLE <<MyTable>>(col1 varchar(32), col2 varchar(32), col3 circle);
```

## Runtime column propagation property

You can configure a Greenplum Connector stage to automatically add missing columns to the link schema at run time.

### Usage

Before you can enable runtime column propagation in a stage, runtime column propagation must be enabled for parallel jobs at the project level from the IBM InfoSphere DataStage and QualityStage Administrator.

When the runtime column propagation property is enabled for the Greenplum Connector stage, the connector inspects the columns for the result set of the statement that it is configured to run. Then the connector compares this set of columns to the set of columns that are defined on the output link. If any columns in the result set are not defined on the output links, the connector adds them to the output link. Columns that are on the output link but not in the result set are removed from the link.

To display and enable the **Runtime column propagation** property, you need to first enable the **Enable Runtime Column Propagation for Parallel Jobs** option for the current DataStage project in the DataStage and QualityStage Administrator.

Then select the **Runtime column propagation** check box on the Columns page for the output link in the stage editor.

## Example

The following examples illustrate how runtime column propagation works.

Assume that the Greenplum Connector stage is configured to fetch data from the data source and provide records on the output link. The examples further assume that the **Runtime column propagation** check box is selected for the output link.

Suppose there are no columns defined for the output link, and the connector is configured to automatically generate SELECT statement to read from table TABLE1. TABLE1 contains the C1, C2 and C3 columns. The connector generates and runs the statement `SELECT * FROM TABLE1` and adds the C1, C2, and C3 columns to the output link.

## Character set property

Use the **Character set** property to define the encoding of the data that is read from or written to the Greenplum databases. You can use the Greenplum connector with national language characters in identifiers like database, schema, table, and column names and in the data that is read from or written to the Greenplum database.

## Usage

You can define the encoding of the data that is read from or written to the Greenplum databases by using one of the following methods:

- Click **Select** to choose an encoding from a list of Greenplum supported encoding.
- Manually enter the encoding name or you can also copy-paste a name. This name must be a Greenplum supported encoding or their alias.

If you do not specify an encoding or if you specify the value **Default**, the connector uses the default system encoding on the InfoSphere Information Server engine tier.

When the Greenplum connector reads data from a Greenplum database, the data is converted from the Greenplum database character set to the character set that is specified in the connector.

- If a link column is of a non-Unicode text type (Char, VarChar, or LongVarChar) and the Extended Unicode attribute of the column is not set, no further conversion occurs. All stages in the job must be configured to use the same character set encoding for these types of columns.
- If a link column is of a Unicode text type (NChar, NVarChar, or LongNVarChar, or a Char, VarChar, or LongVarChar with the Extended Unicode attribute set), the connector converts the data from the character set that is specified in the **Character set** property to UTF-16 before it is sent to the output link.

When Greenplum connector writes data to Greenplum database, the data that is received from the connector is converted from the character set that is specified in the connector to the Greenplum database character set.

- If a link column is of a non-Unicode text type (Char, VarChar, or LongVarChar) and the Extended Unicode attribute of the column is not set, no further conversion occurs. All stages in the job must be configured to use the same character set encoding for these types of columns.

- If a link column that is providing the data to the connector is of a Unicode text type (NChar, NVarChar, or LongNVarChar, or a Char, VarChar, or LongVarChar with the Extended Unicode attribute set), the connector converts the data from UTF-16 to the character set that is specified in the **Character set** property before it is sent to the Greenplum database.

## Write modes

When you configure the Greenplum connector as a target, you can use the **Write mode** property to specify the mode to use to write rows to the Greenplum database.

The following table lists the write modes and describes the operations that the connector completes on the target table for each write mode.

*Table 2. Write modes and descriptions*

Write mode	Description
Insert	The connector attempts to insert records from the input link as rows into the target table.
Update	<p>The connector attempts to update rows in the target table that correspond to records that arrive on the input link and match a set of key columns.</p> <p>Columns to use in the WHERE clause of the UPDATE statement are specified in the <b>Key columns</b> property. Similarly, columns to use in the SET clause of the UPDATE statement are specified in the <b>Update columns</b> property.</p> <p>If the input data contain multiple rows that match the same key combination, you can use the <b>Unique-key column</b> property to specify a column that can be used with the key columns to identify each input row. For every key combination, the connector uses the row with the highest value in the unique-key column to perform the update.</p>
Delete	<p>The connector attempts to delete rows in the target table that correspond to records that arrive on the input link and match a set of key columns.</p> <p>Columns to use in the WHERE clause of the DELETE statement are specified in the <b>Key columns</b> property.</p>

Table 2. Write modes and descriptions (continued)

Write mode	Description
Update then Insert	<p>The connector attempts to update rows in the target table that correspond to records that arrive on the input link and match a set of key columns. If the update operation results indicate that no rows were updated, the connector inserts the record as a new row in the target table.</p> <p>Columns to use in the WHERE clause of the UPDATE statement are specified in the <b>Key columns</b> property. Similarly, columns to use in the SET clause of the UPDATE statement are specified in the <b>Update columns</b> property.</p> <p>If the input data contain multiple rows that match the same key combination, you can use the <b>Unique-key column</b> property to specify a column that can be used with the key columns to identify each input row. For every key combination, the connector uses the row with the highest value in the unique-key column to perform the update.</p>
Delete then Insert	<p>The connector attempts to delete rows in the target table that correspond to records that arrive on the input link and match a set of key columns. If the delete operation results indicate that no rows were deleted, the connector inserts the record as a new row in the target table.</p> <p>Columns to use in the WHERE clause of the DELETE statement are specified in the <b>Key columns</b> property.</p>

Table 2. Write modes and descriptions (continued)

Write mode	Description
User-defined SQL	<p>The connector runs the SQL statements that you specify. You can separate SQL statements with a semicolon.</p> <p>The following variables can be used in the user-defined SQL:</p> <ul style="list-style-type: none"> <li>• <code>[[staging]]</code> – Before the statement is run, the connector replaces this variable with the name of the staging table before the statement is executed. If <b>Staging table mode</b> is set to <b>Existing</b>, this variable is replaced with the name that is specified in the <b>Staging table mode-&gt;Table name</b> property. If the <b>Staging table mode</b> property is set to <b>Automatic</b>, this variable is replaced with the staging table name that the connector generates when the job is run, for example, <code>GPCC_TT_20140211135938443mytable</code>.</li> <li>• <code>[[table]]</code> – Before the SQL statement is run, the connector replaces this variable with the name of the table that is specified in the <b>Table name</b> property.</li> </ul> <pre>UPDATE TABLE [[table]] AS t SET t.col2 = s.col2 FROM [[staging]] AS s WHERE t.col1 = s.col1</pre> <p>The statement that is run might look like this statement:</p> <pre>UPDATE TABLE mytable AS t SET t.col2 = s.col2 FROM GPCC_TT_20140211135938443mytable AS s WHERE t.col1 = s.col1</pre>

**Related tasks:**

“Configuring the Greenplum connector as a target” on page 6

To configure a Greenplum Connector stage to write rows to a Greenplum table, you must specify the target table or view or define the SQL statements.

## Parallel execution

At run time, the Greenplum connector invokes the Greenplum parallel file server program `gpfdist` on each of the InfoSphere DataStage processing nodes.

One `gpfdist` program is used for each processing node, and the number of processing nodes is determined by the InfoSphere DataStage parallel engine configuration file (`$APT_CONFIG_FILE`). For example, if four nodes are defined in your configuration file, the connector invokes four instances of `gpfdist`, one on each processing node.

You can use node map constraints to limit the number of processing nodes on which the Greenplum connector can run.

## Data type mappings from InfoSphere DataStage to Greenplum

When the Greenplum Connector stage reads data or writes data, it converts InfoSphere DataStage data types to Greenplum data types.

The following table shows the mapping rules between InfoSphere DataStage data types and Greenplum data types.

*Table 3. InfoSphere DataStage data types and their corresponding Greenplum data types*

InfoSphere DataStage data types	Greenplum data types
BigInt	bigint
Binary	bit (not supported with sparse lookup)
Bit	boolean
Char	char
Date	date
Decimal	decimal
Double	double
Float	real
Integer	int
LongNVarChar	text
LongVarBinary	bytea
LongVarChar	text
NChar	char
Numeric	decimal
NVarChar	varchar
Real	real
SmallInt	smallint
Time	time (does not include microseconds)
TimeStamp	timestamp (always includes microseconds)
TinyInt	(No direct mapping)
VarBinary	bit varying (not supported with sparse lookup)
VarChar	varchar

The following Greenplum data types do not map directly to InfoSphere DataStage data types. When you configure a job to these data types, use the InfoSphere DataStage data type VarChar.

- box
- cidr
- circle
- inet
- interval
- lseg
- macaddr
- path
- point

- polygon
- timetz
- timestamptz
- xml

---

## Troubleshooting the Greenplum Connector stage

Several common errors are specific to the Greenplum Connector stage.

### The connector was not able to start the gpfdist program

The Greenplum parallel file distribution program, `gpfdist`, runs on each InfoSphere DataStage node, but an error indicates that the program could not be started.

#### Symptoms

You get an error that indicates that the program could not be started. You might also get an operating system error message that indicates that the `gpfdist` executable file is not in the system path.

#### Causes

The `gpfdist` program is not in the system path on the InfoSphere Information Server engine tier.

#### Resolving the problem

Complete the following tasks:

- Update the following:
  - **Windows** The system environment if you are using Windows, or
  - **Linux** The `dsenv` file if you are using Linux.
- Verify if you have followed all the instructions included in the Greenplum parallel file distribution program (`gpfdist`) topic in the IBM InfoSphere Information Server Connectivity Guide for Accessing Greenplum Databases.
- Ensure that the operating system that the user associated with the DataStage has access to `gpfdist` and the permission to execute `gpfdist`. You can allow access and permissions to execute `gpfdist` on the directory `$GPHOME_LOADERS/bin`.

### Connection with the gpfdist program fails

The Greenplum parallel file distribution program, `gpfdist`, runs on each InfoSphere DataStage node, but an error indicates that the program cannot be accessed.

#### Symptoms

You get an error with the 110 error code, which indicates that network access cannot be reached between the `gpfdist` program and the Greenplum server.

#### Resolving the problem

At run time, the `gpfdist` program is accessed by the Greenplum server. To ensure that the Greenplum server has network access to the InfoSphere DataStage node where the `gpfdist` runs, complete one or more of the following tasks:

- Test connectivity from the Greenplum server to the InfoSphere DataStage node by using the `ping` or `wget` commands.

- Because the gpfdist program uses the http protocol, ensure that an http connection is open and accessible on the Greenplum server and the InfoSphere DataStage node. To test http connectivity, use a web browser or the **lynx** command.
- Ensure that the host name for the InfoSphere DataStage node as it appears in the InfoSphere DataStage parallel configuration file fastname entry can be resolved by the Greenplum server. You might need to add the IP address and host name of the InfoSphere Information Server engine tier to the network hosts file.

## Data truncation with Greenplum geometric or multiple format data types

When the Greenplum connector uses the InfoSphere DataStage Varchar data type to read from Greenplum geometric or multiple format data types, some data might get truncated.

### Symptoms

When using the InfoSphere DataStage Varchar data type to read from Greenplum geometric or multiple formats data types, the data that is sent to the output link is truncated. The connector does not report a warning or error.

### Causes

Some Greenplum data types do not map directly to an InfoSphere DataStage type and use the Varchar data type on the column grid. If the length that is defined for the Varchar column is too short to contain all of the data for one of these Greenplum data types, data is truncated.

### Resolving the problem

Increase the column length of the Varchar column on the link so that the column can contain all of the data in the record.

## External table has more gpfdist instances than available primary segments

The Greenplum connector reports an error when the connector uses more processing nodes than the number of primary segments that are available in the Greenplum server.

### Symptoms

An error from the driver reports that the external table has more gpfdist instances (or URLs) than the number of primary segments that are available.

### Resolving the problem

The number of gpfdist instances invoked by the Greenplum connector is determined by the number of processing nodes used by the DataStage job. Each processing node invokes and connects to an instance of the gpfdist program that serves a location that is defined in the external table.

To resolve the error, identify the number of processing nodes for the connector and complete one of the following tasks:

- Decrease the number of nodes defined in the parallel configuration file (\$APT\_CONFIG\_FILE), to a value less than or equal the number of the primary segments that are available.
- Create a node pool to define a subset of processing nodes for the Greenplum connector with a lesser number of processing nodes than available segments.

- Verify that the Greenplum segments are not occupied by other processes and that the Greenplum server is configured to allow sufficient number of segments to scan external tables. You can control the number of segments that will scan external tables using the Greenplum server configuration parameter `gp_external_max_segs`.

## Jobs using key columns fail with ERROR: operator does not exist

The Greenplum server does not support using the equals operator (=) for at least one of the columns in the SQL statement.

### Symptoms

An error is reported by the driver from the Greenplum server indicating that an operator does not exist for the type *<type>*. The error message might appear as follows:

```
ERROR: operator does not exist:<type>=<type> (Hint No operator matches the given name and argument type(s). You may need to add explicit type casts.
```

### Causes

When the connector generates the SELECT, DELETE, or UPDATE statements the columns specified in the **Key columns** property are used in the WHERE clause of the statements. The columns used in the WHERE clause must support the equals operator. This error occurs because one or all of the data types of the key columns (for example, xml, polygon) is not supported with the equal operator.

### Resolving the problem

Perform one of the following steps:

- Use a User defined statement and cast the offending column of type *<type>* to a type that supports the equal operator, like text.
- Remove the offending column from the Key columns list.

## Jobs fail with permission error

The Greenplum connector reports an error when it reads or writes data by using the gpfdist program and external tables.

### Symptoms

An error from the driver indicates that you do not have the privilege that is required to create external tables.

```
ERROR: permission denied: no privilege to create a type gpfdist(s) external table
```

where, *type* is the type of external table that the connector is creating.

### Causes

The job fails because the user or role that is provided to the connector does not have the privileges that are required to create external tables.

### Resolving the problem

To resolve the error, you must grant the CREATEEXTTABLE privileges to the user or role that the connector uses to connect to the Greenplum database.

For example, ALTER ROLE dsuser WITH CREATEEXTTABLE;

---

## Appendix A. Product accessibility

You can get information about the accessibility status of IBM products.

The IBM InfoSphere Information Server product modules and user interfaces are not fully accessible.

For information about the accessibility status of IBM products, see the IBM product accessibility information at [http://www.ibm.com/able/product\\_accessibility/index.html](http://www.ibm.com/able/product_accessibility/index.html).

### Accessible documentation

Accessible documentation for InfoSphere Information Server products is provided in an information center. The information center presents the documentation in XHTML 1.0 format, which is viewable in most web browsers. Because the information center uses XHTML, you can set display preferences in your browser. This also allows you to use screen readers and other assistive technologies to access the documentation.

The documentation that is in the information center is also provided in PDF files, which are not fully accessible.

### IBM and accessibility

See the IBM Human Ability and Accessibility Center for more information about the commitment that IBM has to accessibility.



---

## Appendix B. Reading command-line syntax

This documentation uses special characters to define the command-line syntax.

The following special characters define the command-line syntax:

- [ ] Identifies an optional argument. Arguments that are not enclosed in brackets are required.
- ... Indicates that you can specify multiple values for the previous argument.
- | Indicates mutually exclusive information. You can use the argument to the left of the separator or the argument to the right of the separator. You cannot use both arguments in a single use of the command.
- { } Delimits a set of mutually exclusive arguments when one of the arguments is required. If the arguments are optional, they are enclosed in brackets ([ ]).

### Note:

- The maximum number of characters in an argument is 256.
- Enclose argument values that have embedded spaces with either single or double quotation marks.

For example:

```
wsetsrc[-S server] [-l label] [-n name] source
```

The *source* argument is the only required argument for the **wsetsrc** command. The brackets around the other arguments indicate that these arguments are optional.

```
wlsac [-l | -f format] [key... ] profile
```

In this example, the -l and -f format arguments are mutually exclusive and optional. The *profile* argument is required. The *key* argument is optional. The ellipsis (...) that follows the *key* argument indicates that you can specify multiple key names.

```
wrb -import {rule_pack | rule_set}...
```

In this example, the *rule\_pack* and *rule\_set* arguments are mutually exclusive, but one of the arguments must be specified. Also, the ellipsis marks (...) indicate that you can specify multiple rule packs or rule sets.



---

## Appendix C. How to read syntax diagrams

The following rules apply to the syntax diagrams that are used in this information:

- Read the syntax diagrams from left to right, from top to bottom, following the path of the line. The following conventions are used:
  - The >>--- symbol indicates the beginning of a syntax diagram.
  - The ---> symbol indicates that the syntax diagram is continued on the next line.
  - The >--- symbol indicates that a syntax diagram is continued from the previous line.
  - The --->< symbol indicates the end of a syntax diagram.
- Required items appear on the horizontal line (the main path).



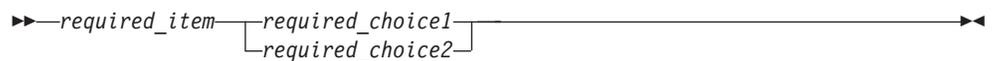
- Optional items appear below the main path.



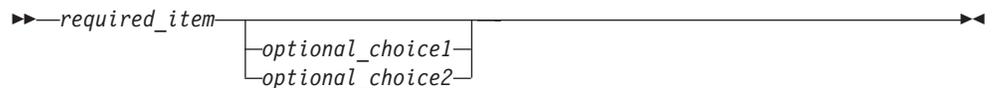
If an optional item appears above the main path, that item has no effect on the execution of the syntax element and is used only for readability.



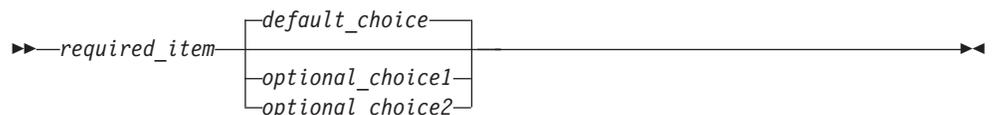
- If you can choose from two or more items, they appear vertically, in a stack. If you must choose one of the items, one item of the stack appears on the main path.



If choosing one of the items is optional, the entire stack appears below the main path.



If one of the items is the default, it appears above the main path, and the remaining choices are shown below.



- An arrow returning to the left, above the main line, indicates an item that can be repeated.



If the repeat arrow contains a comma, you must separate repeated items with a comma.



A repeat arrow above a stack indicates that you can repeat the items in the stack.

- Sometimes a diagram must be split into fragments. The syntax fragment is shown separately from the main syntax diagram, but the contents of the fragment should be read as if they are on the main path of the diagram.



**Fragment-name:**



- Keywords, and their minimum abbreviations if applicable, appear in uppercase. They must be spelled exactly as shown.
- Variables appear in all lowercase italic letters (for example, *column-name*). They represent user-supplied names or values.
- Separate keywords and parameters by at least one space if no intervening punctuation is shown in the diagram.
- Enter punctuation marks, parentheses, arithmetic operators, and other symbols, exactly as shown in the diagram.
- Footnotes are shown by a number in parentheses, for example (1).

---

## Appendix D. Contacting IBM

You can contact IBM for customer support, software services, product information, and general information. You also can provide feedback to IBM about products and documentation.

The following table lists resources for customer support, software services, training, and product and solutions information.

*Table 4. IBM resources*

<b>Resource</b>	<b>Description and location</b>
IBM Support Portal	You can customize support information by choosing the products and the topics that interest you at <a href="http://www.ibm.com/support/entry/portal/Software/Information_Management/InfoSphere_Information_Server">www.ibm.com/support/entry/portal/Software/Information_Management/InfoSphere_Information_Server</a>
Software services	You can find information about software, IT, and business consulting services, on the solutions site at <a href="http://www.ibm.com/businesssolutions/">www.ibm.com/businesssolutions/</a>
My IBM	You can manage links to IBM Web sites and information that meet your specific technical support needs by creating an account on the My IBM site at <a href="http://www.ibm.com/account/">www.ibm.com/account/</a>
Training and certification	You can learn about technical training and education services designed for individuals, companies, and public organizations to acquire, maintain, and optimize their IT skills at <a href="http://www.ibm.com/training">http://www.ibm.com/training</a>
IBM representatives	You can contact an IBM representative to learn about solutions at <a href="http://www.ibm.com/connect/ibm/us/en/">www.ibm.com/connect/ibm/us/en/</a>



---

## Appendix E. Accessing the product documentation

Documentation is provided in a variety of formats: in the online IBM Knowledge Center, in an optional locally installed information center, and as PDF books. You can access the online or locally installed help directly from the product client interfaces.

IBM Knowledge Center is the best place to find the most up-to-date information for InfoSphere Information Server. IBM Knowledge Center contains help for most of the product interfaces, as well as complete documentation for all the product modules in the suite. You can open IBM Knowledge Center from the installed product or from a web browser.

### Accessing IBM Knowledge Center

There are various ways to access the online documentation:

- Click the **Help** link in the upper right of the client interface.
- Press the F1 key. The F1 key typically opens the topic that describes the current context of the client interface.

**Note:** The F1 key does not work in web clients.

- Type the address in a web browser, for example, when you are not logged in to the product.

Enter the following address to access all versions of InfoSphere Information Server documentation:

```
http://www.ibm.com/support/knowledgecenter/SSZJPZ/
```

If you want to access a particular topic, specify the version number with the product identifier, the documentation plug-in name, and the topic path in the URL. For example, the URL for the 11.3 version of this topic is as follows. (The ⇒ symbol indicates a line continuation):

```
http://www.ibm.com/support/knowledgecenter/SSZJPZ_11.3.0/⇒  
com.ibm.swg.im.iis.common.doc/common/accessingiidoc.html
```

**Tip:**

The knowledge center has a short URL as well:

```
http://ibm.biz/knowctr
```

To specify a short URL to a specific product page, version, or topic, use a hash character (#) between the short URL and the product identifier. For example, the short URL to all the InfoSphere Information Server documentation is the following URL:

```
http://ibm.biz/knowctr#SSZJPZ/
```

And, the short URL to the topic above to create a slightly shorter URL is the following URL (The ⇒ symbol indicates a line continuation):

```
http://ibm.biz/knowctr#SSZJPZ_11.3.0/com.ibm.swg.im.iis.common.doc/⇒  
common/accessingiidoc.html
```

## Changing help links to refer to locally installed documentation

IBM Knowledge Center contains the most up-to-date version of the documentation. However, you can install a local version of the documentation as an information center and configure your help links to point to it. A local information center is useful if your enterprise does not provide access to the internet.

Use the installation instructions that come with the information center installation package to install it on the computer of your choice. After you install and start the information center, you can use the **iisAdmin** command on the services tier computer to change the documentation location that the product F1 and help links refer to. (The `⇒` symbol indicates a line continuation):

### Windows

```
IS_install_path\ASBServer\bin\iisAdmin.bat -set -key ⇒  
com.ibm.iis.infocenter.url -value http://<host>:<port>/help/topic/
```

### AIX® Linux

```
IS_install_path/ASBServer/bin/iisAdmin.sh -set -key ⇒  
com.ibm.iis.infocenter.url -value http://<host>:<port>/help/topic/
```

Where `<host>` is the name of the computer where the information center is installed and `<port>` is the port number for the information center. The default port number is 8888. For example, on a computer named `server1.example.com` that uses the default port, the URL value would be `http://server1.example.com:8888/help/topic/`.

## Obtaining PDF and hardcopy documentation

- The PDF file books are available online and can be accessed from this support document: <https://www.ibm.com/support/docview.wss?uid=swg27008803&wv=1>.
- You can also order IBM publications in hardcopy format online or through your local IBM representative. To order publications online, go to the IBM Publications Center at <http://www.ibm.com/e-business/linkweb/publications/servlet/pbi.wss>.

---

## Appendix F. Providing feedback on the product documentation

You can provide helpful feedback regarding IBM documentation.

Your feedback helps IBM to provide quality information. You can use any of the following methods to provide comments:

- To provide a comment about a topic in IBM Knowledge Center that is hosted on the IBM website, sign in and add a comment by clicking **Add Comment** button at the bottom of the topic. Comments submitted this way are viewable by the public.
- To send a comment about the topic in IBM Knowledge Center to IBM that is not viewable by anyone else, sign in and click the **Feedback** link at the bottom of IBM Knowledge Center.
- Send your comments by using the online readers' comment form at [www.ibm.com/software/awdtools/rcf/](http://www.ibm.com/software/awdtools/rcf/).
- Send your comments by e-mail to [comments@us.ibm.com](mailto:comments@us.ibm.com). Include the name of the product, the version number of the product, and the name and part number of the information (if applicable). If you are commenting on specific text, include the location of the text (for example, a title, a table number, or a page number).



---

## Notices and trademarks

This information was developed for products and services offered in the U.S.A. This material may be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

### Notices

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785 U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing  
Legal and Intellectual Property Law  
IBM Japan Ltd.  
19-21, Nihonbashi-Hakozakicho, Chuo-ku  
Tokyo 103-8510, Japan

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:**

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation  
J46A/G4  
555 Bailey Avenue  
San Jose, CA 95141-1003 U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. \_enter the year or years\_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

## Privacy policy considerations

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

Depending upon the configurations deployed, this Software Offering may use session or persistent cookies. If a product or component is not listed, that product or component does not use cookies.

*Table 5. Use of cookies by InfoSphere Information Server products and components*

Product module	Component or feature	Type of cookie that is used	Collect this data	Purpose of data	Disabling the cookies
Any (part of InfoSphere Information Server installation)	InfoSphere Information Server web console	<ul style="list-style-type: none"> <li>• Session</li> <li>• Persistent</li> </ul>	User name	<ul style="list-style-type: none"> <li>• Session management</li> <li>• Authentication</li> </ul>	Cannot be disabled
Any (part of InfoSphere Information Server installation)	InfoSphere Metadata Asset Manager	<ul style="list-style-type: none"> <li>• Session</li> <li>• Persistent</li> </ul>	No personally identifiable information	<ul style="list-style-type: none"> <li>• Session management</li> <li>• Authentication</li> <li>• Enhanced user usability</li> <li>• Single sign-on configuration</li> </ul>	Cannot be disabled

Table 5. Use of cookies by InfoSphere Information Server products and components (continued)

Product module	Component or feature	Type of cookie that is used	Collect this data	Purpose of data	Disabling the cookies
InfoSphere DataStage	Big Data File stage	<ul style="list-style-type: none"> <li>• Session</li> <li>• Persistent</li> </ul>	<ul style="list-style-type: none"> <li>• User name</li> <li>• Digital signature</li> <li>• Session ID</li> </ul>	<ul style="list-style-type: none"> <li>• Session management</li> <li>• Authentication</li> <li>• Single sign-on configuration</li> </ul>	Cannot be disabled
InfoSphere DataStage	XML stage	Session	Internal identifiers	<ul style="list-style-type: none"> <li>• Session management</li> <li>• Authentication</li> </ul>	Cannot be disabled
InfoSphere DataStage	IBM InfoSphere DataStage and QualityStage Operations Console	Session	No personally identifiable information	<ul style="list-style-type: none"> <li>• Session management</li> <li>• Authentication</li> </ul>	Cannot be disabled
InfoSphere Data Click	InfoSphere Information Server web console	<ul style="list-style-type: none"> <li>• Session</li> <li>• Persistent</li> </ul>	User name	<ul style="list-style-type: none"> <li>• Session management</li> <li>• Authentication</li> </ul>	Cannot be disabled
InfoSphere Data Quality Console		Session	No personally identifiable information	<ul style="list-style-type: none"> <li>• Session management</li> <li>• Authentication</li> <li>• Single sign-on configuration</li> </ul>	Cannot be disabled
InfoSphere QualityStage Standardization Rules Designer	InfoSphere Information Server web console	<ul style="list-style-type: none"> <li>• Session</li> <li>• Persistent</li> </ul>	User name	<ul style="list-style-type: none"> <li>• Session management</li> <li>• Authentication</li> </ul>	Cannot be disabled
InfoSphere Information Governance Catalog		<ul style="list-style-type: none"> <li>• Session</li> <li>• Persistent</li> </ul>	<ul style="list-style-type: none"> <li>• User name</li> <li>• Internal identifiers</li> <li>• State of the tree</li> </ul>	<ul style="list-style-type: none"> <li>• Session management</li> <li>• Authentication</li> <li>• Single sign-on configuration</li> </ul>	Cannot be disabled
InfoSphere Information Analyzer	Data Rules stage in the InfoSphere DataStage and QualityStage Designer client	Session	Session ID	Session management	Cannot be disabled

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM's Privacy Policy at <http://www.ibm.com/privacy> and IBM's Online Privacy Statement at <http://www.ibm.com/privacy/details> the section entitled "Cookies, Web Beacons and Other Technologies" and the "IBM Software Products and Software-as-a-Service Privacy Statement" at <http://www.ibm.com/software/info/product-privacy>.

## Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com)<sup>®</sup> are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

The following terms are trademarks or registered trademarks of other companies:

Adobe is a registered trademark of Adobe Systems Incorporated in the United States, and/or other countries.

Intel and Itanium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows and Windows NT are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java<sup>™</sup> and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

The United States Postal Service owns the following trademarks: CASS, CASS Certified, DPV, LACS<sup>Link</sup>, ZIP, ZIP + 4, ZIP Code, Post Office, Postal Service, USPS and United States Postal Service. IBM Corporation is a non-exclusive DPV and LACS<sup>Link</sup> licensee of the United States Postal Service.

Other company, product or service names may be trademarks or service marks of others.



---

# Index

## C

- command-line syntax
  - conventions 25
- commands
  - syntax 25
- configuring 8
- connector
  - column definitions 5, 7
- customer support
  - contacting 29

## D

- data types
  - DataStage 19
  - Greenplum 19
  - writing data 19
- DELETE Statement
  - job fails 22

## G

- Greenplum
  - job fails
    - permission denied 22
    - no privilege to create an external table 22
- Greenplum connector 1
  - configuration
    - configuring the Greenplum connector as a source 5
    - configuring the Greenplum connector as a target 6
  - data truncation 21
  - designing jobs 2
  - external table 21
  - job definition 3
  - lookups
    - configuration 5
  - normal lookup 8
  - Parallel execution
    - gpfdist 18
  - properties
    - NLS 15
    - Run before and after SQL statements 11
    - runtime column propagation 14
    - table action 12
  - reads
    - configuration 5
  - runtime column propagation 14
  - sparse lookup
    - configuring 9
  - writes
    - actions to complete before writing 12
    - configuration 6
    - supported methods 16
- Greenplum databases
  - configuring 1

- Greenplum databases (*continued*)
  - Configuring ODBC access 1
- Greenplum metadata
  - importing 3
- Greenplum Parallel File Server
  - gpfdist 2

## L

- legal notices 35
- lookup operations 7

## M

- mapping
  - data types 19
- metadata
  - importing 3

## P

- product accessibility
  - accessibility 23
- product documentation
  - accessing 31

## R

- reference links 7

## S

- software services
  - contacting 29
- special characters
  - in command-line syntax 25
- support
  - customer 29
- syntax
  - command-line 25

## T

- trademarks
  - list of 35
- troubleshooting
  - Greenplum connector 20

## W

- web sites
  - non-IBM 27







Printed in USA

SC19-4204-00

