

IBM InfoSphere DataStage and QualityStage
Version 11 Release 3

*Connectivity Guide for IBM InfoSphere
Classic Federation Server for z/OS*



IBM InfoSphere DataStage and QualityStage
Version 11 Release 3

*Connectivity Guide for IBM InfoSphere
Classic Federation Server for z/OS*



Note

Before using this information and the product that it supports, read the information in “Notices and trademarks” on page 55.

Contents

Chapter 1. Overview of the Classic Federation enterprise stage 1

Chapter 2. Classic Federation enterprise stage operations 3

Read operation	3
Write operation	3
Update and insert operations	3
Lookup operation.	3

Chapter 3. Configuring the Classic Federation enterprise stage 5

Stage page	5
General tab	5
Advanced tab	5
NLS Map tab	5
Input page for update and insert operations	6
Input page for write operations	7
Output page for classicfedread and classicfedlookup operations	11
Output page for classicfedread operations	11
Output page for classicfedlookup operations	15
Output page with a reject link for a classicfedupsert operation	17

Chapter 4. z/OS File stage 19

Configuring the z/OS File stage	20
Configuring the Classic federation environment	20
Installing the ODBC client	22
Configuring the ODBC driver	24
Setting Classic federation configuration parameters	25

Granting users access to the Classic federation data server	26
Using the z/OS File stage	27
Reading data from a z/OS file	27
Writing data to a z/OS file	27
Common tasks for reading and writing data	28
Options for reading and writing data.	32
Troubleshooting z/OS File stage	35
Error message reference (z/OS File stage)	35

Appendix A. Product accessibility 43

Appendix B. Reading command-line syntax 45

Appendix C. How to read syntax diagrams 47

Appendix D. Contacting IBM 49

Appendix E. Accessing the product documentation 51

Appendix F. Providing feedback on the product documentation 53

Notices and trademarks 55

Index 61

Chapter 1. Overview of the Classic Federation enterprise stage

The Classic Federation enterprise stage is a connectivity stage. With the Classic Federation enterprise stage, you read data from a classic federation server or write data to a classic federation server. You can also perform update, insert, and lookup operations with this stage.

Chapter 2. Classic Federation enterprise stage operations

You can perform database operations with the Classic Federation enterprise stage.

You can perform these database operations:

- Read operation
- Write operation
- Update and insert operations
- Lookup operation

Read operation

The stage reads data from an external data source and places the data in an IBM® InfoSphere® DataStage® output resource.

Write operation

The stage configures a connection to an external data source and insert records into a table. You can read records from a single input resource. The classicfedwrite operation determines how the records of a data set are inserted into the table.

Update and insert operations

The stage performs update and insert operations.

The stage performs these operations:

- Insert records into an external data source table
- Update records in an external data source table
- Delete records from an external data source table
- Insert and then update records in an external data source table
- Update and then insert records into an external data source table

You can match records based on the field names.

Lookup operation

The stage performs a classicfedlookup operation by using a join operation between one or more external data source tables and an IBM InfoSphere DataStage input resource. The data output is an InfoSphere DataStage data set.

Chapter 3. Configuring the Classic Federation enterprise stage

To use a Classic Federation enterprise stage in an IBM InfoSphere DataStage job, you must set the appropriate stage properties. You can set the stage properties by using options that the stage editor provides.

Procedure

To open the stage editor, double-click the stage icon.

Stage page

The Stage page has these tabs: General, Advanced, and NLS Map. With these tabs, you can specify the properties for the stage.

General tab

On the **General** tab, specify an optional description of the stage.

Advanced tab

The **Advanced** tab properties determine the behavior of the stage.

Some advanced properties are set by default, and you cannot change the values of such properties. The properties are described below:

- **Execution mode.** Specify whether the job runs in parallel or sequential mode. The value for the **Execution mode** property is set by default and cannot be changed. If the stage is operating on only one file and there is one reader, the execution mode is sequential. Otherwise, the execution mode is parallel.
- **Combinability mode.** In the default Auto mode, the operators underlying the parallel stages are combined so that the operators run in the same process to improve performance.
- **Preserve partitioning.** Specify Set or Clear modes. If you select Set, the read operation sends a request to the next stage to preserve partitioning.
- **Node pool and resource constraints.** Constrain parallel execution to the node pools and resource pools that are specified in the grid. Use the grid to select node pools and resource pools. The selections are populated from the configuration file.
- **Node map constraint.** Constrain parallel execution on the nodes in a defined node map. To define a node map type node numbers into the text box or click the browse button to select nodes from the Available Nodes window.

NLS Map tab

On the **NLS Map** tab, the character set map that you define overrides the default character set map for the project or the job. If a job requires the National Language Support (NLS) map as a parameter, specify that the map is supplied as a job parameter.

Input page for update and insert operations

On the Input page, specify how you want the data to be transferred from the stage to a remote host using the ODBC protocol.

You can use the **General**, **Properties**, **Partitioning**, **Columns**, and **Advanced** tabs and the **Columns** button to set appropriate properties.

The **General**, **Partitioning**, **Columns**, and **Advanced** tabs are similar for the classicfedwrite and classicfedupsert operations. You must specify different values for the **Properties** tab and **Columns** button for classicfedwrite and classicfedupsert operations. The following table lists the properties and their attributes.

Table 1. Target category properties of the Input page for classicfedupsert operations

Property	Values	Default Value	Required?	Repeats?	Property dependent on
Table	Table name	N/A	Yes	No	N/A
Write method	<ul style="list-style-type: none"> • Upsert • Write 	Write	Yes	No	N/A
Insert SQL	Insert statement	N/A	Yes	No	Write method
Update SQL	Update statement	N/A	Yes	No	Write method
Upsert method	<ul style="list-style-type: none"> • Auto-generated SQL • User-defined SQL 	Auto-generated SQL	Yes	No	Write method
Upsert mode	<ul style="list-style-type: none"> • Insert then update • Delete Only • Delete then Insert • Insert only • Update only • Update then Insert 	Insert then update	Yes	No	Write method

Table 2. Connection category properties of the Input page for classicfedupsert operations

Property	Values	Default Value	Required	Repeats?	Property dependent on
Data source	Data source name	N/A	Yes	No	N/A
User	User name	N/A	Yes	No	Data source
Password	Password	N/A	Yes	No	Data source

Table 3. Options category properties of the Input page for classicfedupsert operations

Property	Values	Default Value	Required	Repeats	Property dependent on
Insert array size	Integer	2000	No	No	N/A
Open command	SQL statement	No	No	No	N/A
Close command	SQL statement	No	No	No	N/A
Row commit interval	Integer	N/A	No	No	Insert array size
Output reject records	<ul style="list-style-type: none"> • True • False 	False	No	No	Write method

Input page for write operations

On the Input page, specify the properties to perform a classicfedwrite operation.

General tab

On the **General** tab, you can specify an optional description of the input link.

Properties tab

On the **Properties** tab, specify properties for the input link. The properties determine the data source, the operation to be performed on the data, the output resource, and so on. The following table lists the properties and their attributes. A more detailed description of each property follows.

Table 4. Target category properties of the Input page for classicfedwrite operations

Property	Values	Default value	Required?	Property dependent on
Table	Table name	N/A	Yes	N/A
Write method	<ul style="list-style-type: none"> • Upsert • Write 	Write	Yes	N/A
Write mode	<ul style="list-style-type: none"> • Append • Create • Replace • Truncate 	Append	Yes	N/A

Table 5. Connection category properties of the Input page for classicfedwrite operations

Property	Values	Default value	Required	Property dependent on
Data source	Data source name	N/A	Yes	N/A
User	User name	N/A	Yes	Data source
Password	Password	N/A	Yes	Data source

Table 6. Options category properties of the Input page for classicfedwrite operations

Property	Values	Default value	Required	Property dependent on
Insert array size	Integer	2000	No	N/A
Truncate column names	<ul style="list-style-type: none"> • False • True 	False	Yes	N/A
Open command	SQL statement	No	No	N/A
Close command	SQL statement	No	No	N/A
Isolation level	<ul style="list-style-type: none"> • Read uncommitted • Read committed • Repeatable read • Serializable 	Read uncommitted	No	N/A
Create statement	String	N/A	No	Write method and Write mode
Row commit interval	Integer	N/A	No	Insert array size
Drop unmatched column	<ul style="list-style-type: none"> • False • True 	False	No	N/A
Delimiter	ASCII character	@	No	N/A

Target category

Specify the **Table**, **Write Method**, and **Write Mode** values:

Table

Specify the appropriate value to connect the stage to a target file that is located on a remote host.

Write Method

Specify how data is written and exported.

Write Mode

Specify that the stage writes and exports data to a single table. Define how the records from the data source are inserted into the destination table. **Write Mode** can have one of the following values. Each of the modes requires specific user privileges:

- **Append** - This is the default mode. Append new row to the existing destination table. The destination table must exist, and the record schema of the data set must be compatible with the schema of the table. The schema of the existing table determines the input interface of the stage.
- **Create** - Creates a new destination table. If a table exists with the same name as the one being created, the operation terminates and an error message is displayed. The schema of the DataStage data set determines the schema of the new table. The table is created with simple default properties. To create a table with any properties other than the default properties, such as partitioned, indexed, or in a non-default table space, use the `-createtmt` option with the `- createtable` statement.

- **Replace** - Replace an existing destination table. ClassicFedWrite operation drops the table that you specify as the destination table and creates a new one in its place. If a table exists with the same name as the one being created, it is overwritten. The schema of the DataStage data set determines the schema of the new table.
- **Truncate** - Requires a destination table. The ClassicFedWrite operation retains the attributes of the destination table, but discards existing records and appends new records. The schema of the existing table determines the input interface of the stage.

Connection category

Specify **Data source**, **Password**, and **User** values:

Data source

This property is mandatory. Specify the database connection with one of these methods:

- Enter the data source name in the **Data source** field on the right side of the Properties page.
- Insert the required value as a job parameter. Click the pointer button on the extreme right side of the Properties page, and then click **Insert Job Parameters**.

A dialog is displayed with a list of available job parameters. To create a new parameter for the job, click **[New...]** from the list and create an appropriate environment variable in the Job Properties dialog.

Note: With IBM InfoSphere DataStage and QualityStage[®] Administrator, you can create parameters for all jobs within the project.

User

This field is optional. Specify the user name for connections to the data source with one of these methods:

- Enter the user name in the **User** field on the right side of the Properties page.
- Insert the required value as a job parameter. Click the pointer button on the extreme right side of the Properties page, and then click **Insert Job Parameters**.

A dialog appears with a list of available job parameters. Create a new parameter for the job, click **[New...]** from the list, and create an appropriate environment variable with the Job Properties dialog.

Note: If you insert all or some of the **Connection** category values from the job parameter list, then this job takes the environment variables from the operating system. At runtime, you are prompted to modify the values for those environment variables.

Password:

This field is optional. Specify the password for connections to the data source with one of these methods:

- Enter the password in the **Password** field on the right side of the Properties page.
- Insert the required value as a job parameter. Click the pointer button on the extreme right side of the Properties page, and then click **Insert Job Parameters**.

A dialog appears with a list of available job parameters from which you can choose. If you want to create a new parameter for the job, click **[New...]** from the list and create an appropriate environment variable in the Job Properties dialog.

Options category

Specify the values for the **Insert Array Size**, **Truncate Column Names**, **Close Command**, **Length to Truncate**, **Drop unmatched column**, **Open Command**, and **Isolation level** properties. The **Truncate Column Names** property is the default. You can add the other properties from the **Available properties to add** list.

Insert Array Size

Specify the size of the insert host array. You can only enter an integer. The default value is 2000.

Truncate Column Names

Set one value depending upon your requirements:

- **True** - Indicate that column names are truncated to the size that is allowed by the Classic Federation driver.
- **False** - Disable truncation of the column names.

Close Command

Enter the SQL statement to be run after an insert array is processed. This statement is run only once on the conductor node.

Drop unmatched column

Set one value depending upon your requirements:

- **True** - Indicate that unmatched columns in the data set are dropped. An unmatched column is a column for which there is no similar column in the data source table.
- **False** - Indicate that unmatched fields of the data set are not dropped.

Open Command

Enter the SQL statement to be run before the insert array is processed. This statement is run only once on the conductor node.

Isolation level

Select the isolation level for accessing data.

Create statement

Specify the SQL statement to create the table. This property is displayed only if the Write mode is create or replace.

The database that is specified in the data source determines the default isolation level.

Partitioning tab

On the **Partitioning** tab, specify details to determine how the incoming data is partitioned or collected before it is written to the destination database. You can also specify that the data is sorted before it is written to the destination database.

By default, the stage partitions data in Auto mode. Auto mode provides the best partitioning method depending on the execution mode of the current and preceding stages and how many nodes are specified in the configuration file.

If the stage is operating in sequential mode, the stage collects data before writing the data to the destination database with the default **Auto** collection method. Use the **Partitioning** tab to override this default behavior. The properties on this tab behave differently according to whether the current and preceding stages are set to run in parallel or sequential mode.

Columns tab

On the **Columns** tab, view and modify column metadata for the input link. Use the **Save** button to save any modifications that you make in the column metadata. Use the **Load** button to load an existing source table.

From the Table Definitions window, select the appropriate table to load and click **OK**. The Select Column dialog is displayed. To ensure the appropriate conversion of data types, clear the **Ensure all Char columns use Unicode** check box.

Advanced tab

On the **Advanced** tab, you can specify how input data for the stage is buffered. By default, data is buffered so that no deadlocks can arise. A *deadlock* is a situation in which a number of stages are mutually dependent. An example is a situation in which one or more stages are waiting for input from another stage, and those stages cannot output data until they have received input.

Columns button

Use the **Columns** button to define column names for the destination table.

Output page for classicfedread and classicfedlookup operations

The Output page appears both in the classicfedread and classicfedlookup modes of the Classic Federation enterprise stage.

On the Output page, provide details about the output link to the stage from a remote host. With the output link, you can access data from a remote host by using IBM InfoSphere Classic Federation Server for z/OS®. In the classicfedread mode, the stage only has an output link. In the classicfedlookup mode, the stage has an output link and a reference link that connects the stage to a lookup stage.

You can use the **General**, **Properties**, **Partitioning**, **Columns**, and **Advanced** tabs and the **Columns** button to set appropriate properties.

The **General**, **Partitioning**, **Columns**, and **Advanced** tabs are similar for classicfedread and classicfedlookup operations. You must specify different values on the **Properties** tab and **Columns** button for classicfedread and classicfedlookup operations.

Output page for classicfedread operations

On this page, use the tabs to specify the properties to perform a classicfedread operation.

General tab

On the **General** tab, enter an optional description for the classicfedread operation.

Properties tab

On the **Properties** tab, specify properties for the output link. The properties determine the data source, the operation to be performed on the data, the input resource, and so on. The following table lists the properties and their attributes. A more detailed description of each property follows.

Table 7. Source category properties of the Output page for ClassicFedRead operations

Property	Values	Default value	Required?	Property dependent on
Table	Table name	N/A	Yes	N/A
Read method	<ul style="list-style-type: none"> • Auto-generated SQL • Table • User-defined SQL 	Table	Yes	N/A
Query	Select Statement	N/A	Yes	Read method
Table	<ul style="list-style-type: none"> • Select List • Where Clause 	N/A	Yes	Read method
Select list	List of columns	N/A	Yes	Table
Where clause	Rows of a table	N/A	Yes	Table

Table 8. Connection category properties of the Output page for ClassicFedRead operations

Property	Value	Default value	Required?	Property dependent on
Data source	Data source name	N/A	Yes	N/A
User	User name	N/A	Yes	Data source
Password	Password	N/A	Yes	Data source

Table 9. Options category properties of the Output page for ClassicFedRead operations

Property	Value	Default value	Required?	Property dependent on
Fetch array size	Integer	1	Yes	N/A
Open command	SQL statement	No	Yes	N/A
Close command	SQL statement	No	Yes	N/A
Isolation level	<ul style="list-style-type: none"> • Read Uncommitted • Read Committed • Repeatable Read • Serializable 	Read Uncommitted	Yes	N/A

Source category

Specify the values for **Read Method**, **Query**, and **Table**:

Read Method

Specify a table or a query for reading the database. The default value is **Table**. If you select **Table**, you must specify the data source table for the **Table** option. Alternatively, you can set up **Read Method** as an SQL query. In that case, you must specify whether you want the query to be generated automatically or you want to define the query yourself.

Note: The **Query** property is displayed on the list of properties only when you select **Auto-generated SQL** or **User-defined SQL** as the **Read** method. To select one of these types of query, click **Read Method**, and then select the appropriate option from the **Read Method** list on the right side of the properties list.

Query

Specify an SQL statement for reading a table. The statement specifies the table to read and the processes to perform on the table during the ClassicFedRead operation. This statement can contain join and view operations, database links, synonyms, and so on. Choose from the following values:

- **Auto-generated SQL:** Specify that an SQL query is generated automatically that is based on the table that you specify in the **Table** field and the columns that you define.
- **User-defined SQL:** Define a custom SQL query.

Note: An SQL query is read sequentially on one node.

Table

When you select **Table** as the read method, specify the name of the source table on the federated system. The specified table must exist, and you must have SELECT privileges for this table. If your federated user name does not correspond to the owner of the specified table, you can prefix it with a table owner. You must add a new job parameter to fix the table owner name.

To fix the table owner name:

1. Click **Table** and then the arrow on the right side of the dialog.
2. Use the WHERE clause in your SELECT statement to specify the rows of the table to be included or excluded from the read operation. If you do not supply a WHERE clause, all rows are read.
3. Click **Insert job parameter** and then [New...] from the list.
4. You can specify in your SELECT statement the columns that you want to read. You must specify the columns in this list in the same order as they are defined in the record schema of the input table.
5. In the Job Properties dialog enter the required table details in **Default Value** column for the **user** parameter in this format:

```
table_owner.table_name
```

Connection category

Specify the **Data source**, **Password**, and **User** values:

Data source

This is a required field. Specify the database connection in this field by using any one of the following methods:

- Enter the data source name in the **Data source** field on the right side of the Properties page.

- Insert the required value as a job parameter. Click the pointer button on the extreme right side of the Properties page, and then click **Insert Job Parameters**.

A list of available job parameters is displayed. To create a new parameter for the job, click [New...] from the list and create an appropriate environment variable in the Job Properties dialog.

User

This is an optional field. Specify the user name for connecting to the data source by using any one of the following methods:

- Enter the user name in the **User** field on the right side of the Properties page.
- Insert the desired value as a job parameter. Click the pointer button on the extreme right side of the Properties page, and then click **Insert Job Parameters**.

A dialog appears with a list of available job parameters. To create a new parameter for the job, click [New...] from the list and create an appropriate environment variable in the Job Properties dialog.

Note: If you insert all or some of the **Connection** category values from the job parameter list, then the job takes the environment variables from the operating system. At run time, you are prompted to modify the values for those environment variables.

Password

This is an optional field. Specify the password for connections to the data source with one of these methods:

- Enter the password in the **Password** field on the right side of the Properties page.
- Insert the required value as a job parameter. Click the pointer button on the extreme right side of the Properties page, and then click **Insert Job Parameters**.

A dialog appears with a list of available job parameters. To create a new parameter for the job, click [New...] from the list and create an appropriate environment variable with the Job Properties dialog.

Options category

Specify values for the **FetchArraySize**, **Isolation Level**, **Close Command**, and **Open Command** properties. All of these properties are optional. You see these properties in the **Available properties to add** list that appears in the lower right corner of the Output page. To add these subproperties, click **Options**, and then click the property that you want to add from the **Available properties to add** list.

Fetch Array Size

Specify the number of rows to retrieve during each fetch operation. The default value is 1.

Isolation Level

Enter the isolation level for accessing data.

The database for the **Data source** property determines the default isolation level.

Close Command

Enter an SQL statement to be run after the insert array is processed. You cannot commit work with this property. The statements are run only once on the conductor node.

Open Command

Enter an SQL statement to be run before the insert array is processed. The statements are run only once on the conductor node.

Note: With IBM InfoSphere DataStage and QualityStage Administrator, you can create parameters at the project level for all jobs within the project.

Advanced tab

Use the **Advanced** tab to specify how input and output data for the stage is buffered. By default, data is buffered so that no deadlocks can arise.

Columns button

Use the **Columns** button to define a list of column names for the output table.

View Data button

To view the number of rows in the table that you specified for the **Table** property under **Source**, click the **View Data** button. You can specify the number of rows that you want to view at a time.

Output page for classicfedlookup operations

For a lookup operation, you need to specify appropriate values on the **Properties** tab.

The **General**, **Partitioning**, **Columns**, and **Advanced** tabs are similar for classicfedread and classicfedlookup operations. You must specify different values on the **Properties** tab and **Columns** button for classicfedread and classicfedlookup operations.

Properties tab

Use the **Properties** tab to set appropriate values for the **Source**, **Connection**, and **Options** properties.

Source category

Specify values for **Lookup Type**, **Query**, **Read Method**, and **Table**:

Lookup Type

Specify **Normal** or **Sparse** as the lookup method :

- **Normal:** This is the default. A normal lookup is an in-memory lookup on federation database table. The lookup stage can have multiple reference links.
- **Sparse:** Accesses the source database directly. The lookup stage has one reference link.

Query

Specify an SQL statement for reading a table. The statement specifies the table to read and the processes to perform on the table during the read operation.

This statement can contain **join** and **view** operations, database links, synonyms, and so on. Choose from the following available values:

- **Auto-generated SQL:** Specify that SQL query is generated automatically that is based on the table in the **Table** field and the columns that you define.
- **User-defined SQL:** Define a custom SQL query.

Note: An SQL query is read sequentially on one node.

Table

When you select **Table** as the read method, you must specify the name of the source table on the federated system. The specified table must exist, and you must have SELECT privileges for this table. If your federated user name does not correspond to the owner of the specified table, you can prefix it with a table owner. You must add a new job parameter to fix the table owner name.

To fix the table owner name:

1. Click **Table** and then the arrow on the right side of the dialog.
2. Use the WHERE clause in your SELECT statement to specify the rows of the table to be included or excluded from the read operation. If you do not supply a WHERE clause, all rows are read.
3. Click **Insert job parameter** and then [New...] from the list.
4. Specify in your SELECT statement the columns that you want to read. You must specify the columns in this list in the same order as they are defined in the record schema of the input table.
5. In the Job Properties dialog, enter the required table details in **Default Value** column for the **user** parameter in this format:

`table_owner.table_name`

Connection category

Specify the **Data source**, **Password**, and **User** values:

Data source

This is a required field. Specify the database connection with one of these methods:

- Enter the data source name in the **Data source** field on the right side of the Properties page.
- Insert the required value as a job parameter. Click the pointer button on the extreme right side of the Properties page, and then click **Insert Job Parameters**.

A list of available job parameters is displayed. To create a new parameter for the job, click [New...] and create an appropriate environment variable in the Job Properties dialog.

With IBM InfoSphere DataStage and QualityStage Administrator, you can create parameters for all jobs within the project.

User

This is a required field. Specify the user name for connecting to the data source using one of these methods:

- Enter the user name in the **User** field on the right side of the Properties page.
- Insert the required value as a job parameter. Click the pointer button on the extreme right side of the Properties page, and then click **Insert Job Parameters**.

A list of available job parameters is displayed. To create a new parameter for the job, click **[New...]** list and create an appropriate environment variable in the Job Properties dialog.

Note: If you insert all or some of the **Connection** category values from the job parameter list, then the job takes the environment variables from the operating system. At run time, you are prompted to modify the values for those environment variables.

Password

This is a required field. Specify the password for connecting to the data source using one of these methods below:

- Enter the password in the **Password** field on the right side of the Properties page.
- Insert the required value as a job parameter. Click the pointer button on the extreme right side of the Properties page, and then click **Insert Job Parameters**.

A list of available job parameters is displayed. If you want to create a new parameter for the job, click **[New...]** from the list and create an appropriate environment variable in the Job Properties dialog.

Options category

Specify the **Fetch Array Size**, **Isolation Level**, **Close Command** and **Open Commands** properties. All of these properties are optional. You see these properties in the **Available properties to add** list that appears in the bottom right corner of the Output page. To add these properties under **Options**, click **Options**, and then the property from the **Available properties to add** list. **Isolation Level** is in the list only if you select **Lookup Type** as **Normal**.

Fetch Array Size

Specify the number of rows to retrieve during each fetch operation. The default value is 1.

Isolation Level

Enter the isolation level for accessing data.

The database for the Data source property determines the default isolation level.

Close Command

Enter an SQL statement to be executed after the insert array is processed. You cannot commit work using this option. The statements are executed only once on the conductor node.

Open Command

Enter an SQL statement to be executed before the insert array is processed. The statements are executed only once on the conductor node.

Output page with a reject link for a classicfedupsert operation

A reject link contains raw data for columns rejected due to schema mismatch after your SELECT statement is run.

If the stage editor shows both input and output links for a classicfedlookup or classicfedwrite operation, you can choose any of the links as the reject link.

An Output page with a reject link for a classicfedupsert operation contains the same tabs and buttons as the Output page for a classicfedread or classicfedlookup operation.

Chapter 4. z/OS File stage

Use the z/OS File stage to identify and define file-level information about a z/OS file.

The z/OS File stage, which is available only on the parallel canvas, is always used with a Complex Flat File stage. The z/OS File stage identifies the z/OS file; the Complex Flat File stage converts the records into individual fields. The IBM InfoSphere Classic Federation Server for z/OS reads records from and writes records to the z/OS file. TCP/IP is used to pass blocks of records back and forth between IBM InfoSphere DataStage and Classic federation.

The following steps describe a job that reads data from a z/OS file. A job that writes data to a z/OS file follows similar steps.

1. The z/OS File stage provides the information that is required to make a connection to the data server and to identify the z/OS file.
2. The ODBC driver uses one of the following methods to obtain the data source information (the host name or IP address and the port number) of the data server:
 - On Microsoft Windows, the ODBC driver uses the information that is specified in the ODBC Data Source Administrator to obtain the location of the data server.
 - On Linux and UNIX, the ODBC driver uses the file that is specified in the CAC_CONFIG environment variable to obtain the location of the data server. The file is typically named cac.ini.
3. The job communicates with the data server to determine if the z/OS file exists and to obtain information about the file. Classic federation uses the file information to dynamically create a table definition that describes the records.
4. The job issues a SELECT statement on the table that was dynamically created on the data server.
5. The data server sends a block of records from the z/OS file to the ODBC driver. The **FETCH BUFFER SIZE** parameter specifies the number of records in each block. On Microsoft Windows, the ODBC Data Source Administrator program specifies this parameter; on Linux and UNIX, the file that is specified in the CAC_CONFIG environment variable specifies this parameter.
6. Each time that the job issues a fetch request and the block still contains records, the ODBC driver returns a record to the job. There is no network transaction between InfoSphere DataStage and the data server. If the block doesn't contain any more records, the ODBC driver issues a fetch request to the data server, which reads and returns the next block of records to the ODBC driver.
7. The z/OS File stage passes the records to the Complex Flat File stage, which processes the records into fields and columns.
8. The z/OS File stage notifies Classic federation to drop the table that was defined for the file and to deallocate the file from the data server.

Configuring the z/OS File stage

You perform configuration tasks on the InfoSphere Information Server and on the Classic federation data server.

To configure the z/OS File stage, complete these tasks:

Configuring the Classic federation environment

If you do not already have a Classic federation data server installed, you must install and configure it. If you already have a Classic federation data server, you must update the existing metadata catalog.

Use one of the following topics to configure the Classic federation environment:

Installing and configuring a new Classic federation environment

To use the z/OS File stage in a new Classic federation environment, you must install and configure the data server.

About this task

The z/OS File stage enables direct access from IBM InfoSphere Information Server to files that are stored on the z/OS machine. For complete information about configuring the data server, see the topic "Customizing installations" in the IBM InfoSphere Classic information center at <http://publib.boulder.ibm.com/infocenter/iisclzos/v9r5/topic/com.ibm.swg.im.iis.prod.installcust.classic.doc/topics/iippiicintr.html>.

To install and configure the data server:

Procedure

1. Install Classic Federation Version 9.5 Rollup 2 by completing the SMP/E installation procedures. See the Program Directory for IBM InfoSphere Classic Federation Server for z/OS for detailed installation information.
2. To customize and validate the Classic federation environment, edit the user samples allocation utility JCL in the distribution samples member SCACSAMP(CACCUSJ1) to specify input parameters. SCACSAMP(CACCUSJ1) is the only member in the distribution data sets that you modify. You specify the following input parameters:

CACINHLQ=CAC.V9R5M00

The value specified for the CACINHLQ keyword must be the high-level qualifier for Classic distribution data sets produced by the SMP/E installation.

CACUSHLQ=USER.V9R5M00

The value specified for the CACUSHLQ keyword must be the high-level qualifier for the *USERHLQ.SCACSAMP* and *USERHLQ.SCACCONF* data sets that are created or updated by the samples allocation utility.

CACDUNIT=UNIT

The value specified for the CACDUNIT keyword identifies the disk unit that is used when allocating the *USERHLQ.SCACSAMP* and *USERHLQ.SCACSAMP* data sets.

CACDVOLM=VOLUME

The value specified for the CACDVOLM keyword identifies the disk volume that is used when allocating the *USERHLQ.SCACSAMP* and *USERHLQ.SCACSAMP* data sets.

ISPFHLQ=ISP

The value specified for the ISPFHLQ keyword identifies the high-level qualifier for ISPF installation. The samples allocation utility runs a TSO batch application and uses TSO functions.

ISPFLANG=ENU

The value specified for the ISPFLANG keyword identifies the language prefix for the ISPF installation.

COPY=ALL/PARMS

The value specified for the COPY keyword controls whether all SCACSAMP members are copied to the *USERHLQ.SCACSAMP* data set or only the customization parameters file and customization utility JCL. COPY=ALL is the default.

The first time that you run the utility, specify COPY=ALL to ensure that all required members are copied.

3. Submit *SCACSAMP(CACCUSJ1)* to allocate the *USERHLQ.SCACSAMP* and *USERHLQ.SCACCONF* data sets. This JCL allocates the SCACMENU data set for the message catalog. Verify that all job steps result in a return code <= 4.
4. Edit the customization parameters file *USERHLQ.SCACSAMP(CACCUSP2)* to provide customization parameters. For more information about the customization parameters, see the topic "Working with the customization parameters file" in the IBM InfoSphere Classic information center at <http://publib.boulder.ibm.com/infocenter/iisclzos/v9r5/topic/com.ibm.swg.im.iis.prod.installcust.classic.doc/topics/iipyiccpgl.html>.
5. Submit the generated *USERHLQ.SCACSAMP(CACCUSJ2)* customization utility JCL. Verify that all job steps result in a return code <= 4.
6. Submit the generated *USERHLQ.SCACSAMP(CACCATFG)* JCL to allocate and initialize the metadata catalog and allocate the configuration data sets.
7. Submit the generated *USERHLQ.SCACSAMP(CACDS)* JCL to start the data server. The data server starts with the default services:
 - Connection handler service
 - Logger service
 - Operator service
 - Query processor service
 - Region controller service
8. Submit the generated *USERHLQ.SCACSAMP(CACCUSVF)* validation job. Verify that all job steps result in a return code <= 4.

Updating an existing Classic federation environment

In an existing Classic federation environment, you must use Version 9.5 Rollup 2 to update the existing metadata catalog.

About this task

To update an existing Classic federation environment:

Procedure

1. Install Classic federation Version 9.5 Rollup 2 by completing the SMP/E installation procedures. See the Program Directory for IBM InfoSphere Classic Federation Server for z/OS for detailed installation information.
2. Use the METALOAD command to run the catalog initialization and maintenance utility. For complete information about the data server, see the IBM InfoSphere Classic Information center at <http://publib.boulder.ibm.com/infocenter/iisclzos/v9r5/index.jsp>.

Installing the ODBC client

Install the ODBC client to connect InfoSphere Information Server to the InfoSphere Classic Federation Server for z/OS.

About this task

The following list contains the ODBC client installation programs:

- cac95ax – IBM AIX®
- cac95zlx – SuSE Linux on IBM System z®
- cac95win.exe – Microsoft Windows

To install the ODBC client:

Procedure

1. On IBM InfoSphere Information Server, log in as a user with administrator authority (user root on UNIX and Linux).
A non-root user can also install the client programs, but can update and uninstall only that user's instance.
2. Run the installation wizard.

Restriction:

If you do not use the default installation path, ensure that the directory you choose does not contain prior versions of the ODBC client.

The default installation paths are as follows:

Linux: /opt/ibm/wsclassic95

UNIX: /opt/IBM/wsclassic95

Microsoft Windows: C:\Program Files\IBM\WScClassic95

If you install the client programs as a non-root user on UNIX or Linux, the default installation path is \$HOME/wsclassic95.

Method	Steps
From CD	<p>Microsoft Windows:</p> <ol style="list-style-type: none"> 1. Insert the Classic ODBC/CLI and JDBC client CD. 2. If autorun is enabled, the wizard will start automatically. If the wizard does not start, you can launch the wizard from a command prompt. <i>CD_drive:\cac95wn.exe</i> <p>UNIX and Linux:</p> <ol style="list-style-type: none"> 1. Insert and mount the Classic ODBC/CLI and JDBC client CD. 2. Start the wizard by using the installation program that corresponds to the operating system.. <i>/mounted_volume/installation_program</i>
From ibm.com [®]	<ol style="list-style-type: none"> 1. On your system, create a temporary directory with at least 150 MB of free space. 2. Download the installation program that corresponds to the operating system into the temporary directory. 3. On Linux and UNIX systems, add execute permission to the installation program. 4. Start the wizard using the installation program that corresponds to the operating system. <ul style="list-style-type: none"> • Microsoft Windows: <i>Drive:\path_to_tempdir\cac95wn.exe</i> • UNIX and Linux: <i>cd</i> <i>/path_to_tempdir/installation_program</i> <i>chmod 755 ./cac95xx ./cac95xx</i>

Results

If you need to troubleshoot an install session, log information is available:

Microsoft Windows:

```
install_path\_log\log.txt
install_path\_log\install.log
```

For a more detailed debugging log, you can rerun the installation program with the following options:

```
cac95wn.exe -is:log_full_pathname_of_logfile -debug
```

Linux:

```
install_path/_log/log.txt
install_path/_log/install.log
```

For a more detailed debugging log, you can rerun the installation program with the following options (AIX korn shell redirection syntax is used as an example):

```
cac95ax -is:javaconsole -debug >full_pathname_of_logfile 2>&1
```

Configuring the ODBC driver

After you install the ODBC driver, you must configure the driver and create the data source.

Use one of the following topics to configure the ODBC driver:

Configuring the ODBC driver and creating the data source (Linux and UNIX)

To configure the ODBC driver on Linux or UNIX, you set environment variables and specify parameters for the `cac.ini` file.

About this task

To configure the ODBC driver and create the data source:

Procedure

1. On the IBM InfoSphere Information Server, set the following environment variables:

```
LIBPATH=$LIBPATH:/install_path/cli/lib
PATH=$PATH:/install_path/cli/bin
CAC_CONFIG=/directory/cac.ini
```

where `install_path` is `/opt/ibm/wsclassic95` for Linux and `/opt/IBM/wsclassic95` for UNIX and `directory` is any directory.

2. To create the data source, open the `cac.ini` file, as shown in the following example:

```
*****
* Sample CLI Client Configuration File          *
*****
* messages and codes catalog
NL CAT = /opt/IBM/wsclassic95/cli/lib
NL = US English
* user id/pwd needed for catalog security
USERID = cacuser
USERPASSWORD = cacpwd
* default datasource location
DEFLOC = CACSAMP
DATASOURCE = CACSAMP tcp/machine.url.com/5001
* performance and memory parameters
FETCH BUFFER SIZE = 32000
MESSAGE POOL SIZE = 16777216
* codepage parameters
SERVER CODEPAGE = IBM-037
CLIENT CODEPAGE = IBM-850
```

The configuration file resides in the installation directory `/opt/IBM/wsclassic95/cli/lib/cac.ini` for UNIX or `/opt/ibm/wsclassic95/cli/lib/cac.ini` for Linux. For the 64-bit version, the configuration file resides in the 64-bit lib subdirectories.

3. Modify the `DATASOURCE` parameter. `DATASOURCE` is a required parameter that specifies the hostname, IP address, port, and name of the data source that the client attempts to connect to. The data source name that you enter ("CACSAMP" in the `cac.ini` example above) must match the name of the query processor service on the data server.
4. To configure the data source for the DataDirect ODBC driver manager, complete the following steps:
 - a. Edit the `odbc.ini` file to add the new data source for the client:

```
[ODBC data sources]
CACXSAMP=WebSphere Classic Federation client
...
[CACSAMP]
Driver=/opt/IBM/wscClassic95/cli/lib/lib64/client
```

where *client* is *cacsqlcli* (UNIX) or *libcacsqlcli.so* (Linux).

- b. Make sure that the ODBCINI environment variable, which specifies the location of the *odbc.ini* file, is defined to the IBM InfoSphere DataStageenvironment.
5. Use the DataDirect "example" application that comes with InfoSphere Information Server to test the connection to the data server.

Configuring the ODBC driver and creating the data source (Microsoft Windows)

If InfoSphere Information Server runs on the Microsoft Windows operating system, after you install the ODBC driver, you must configure the driver and create the data source.

Procedure

1. To create the data source, complete these steps: .
 - a. From Microsoft Windows, choose **Start > Control panel > Administrative Tools > Data sources (ODBC)**
 - b. On the **System DSN** tab, click **Add** to add a new data source.
 - c. Select **IBM InfoSphere Classic ODBC Driver V9.5** and then click **Finish**.
 - d. Select **TCP/IP** as the communications interface, and press **OK**.
 - e. On the **General** tab of the InfoSphere Classic ODBC Driver Setup page, enter the data source name, host IP address, and port number of the data server. The data source name that you enter must match the name of the query processor service on the data server.
2. Use the Microsoft ODBC Test application, which is available from Microsoft, to test the connection to the data server.

Setting Classic federation configuration parameters

Setting these Classic federation configuration parameters might improve performance.

About this task

To set Classic federation configuration parameters on the Classic federation client, use one of these methods:

- On Microsoft Windows, use the ODBC Administrator by selecting **Start > Programs > IBM WebSphere V9.5 > WebSphere Classic ODBC Administrator**.
- On Linux or UNIX, edit the file that is named in the *CAC_CONFIG* environment variable on the Classic federation client. This file is typically named *cac.ini*.

Procedure

1. Set the following parameters in the Classic federation configuration file on the client:

FETCH BUFFER SIZE

The default value is 32000. For z/OS read operations, set this parameter to a value between 32000 and 1500000. Use an iterative approach to find the optimal value for your environment. *FETCH BUFFER SIZE* can

be as large as MESSAGE POOL SIZE. Be aware that if you set this value too high, performance might degrade. This parameter applies only to z/OS read operations.

MESSAGE POOL SIZE

The default value is 16777216. For z/OS read and write operations, set this parameter to a value between 33554432 and 67108864. Try using a value that is 10 to 20 times the size of the **FETCH BUFFER SIZE** parameter. If the client logs out-of-memory errors, increase the value of the **MESSAGE POOL SIZE** parameter. The maximum value for this parameter is 1 GB or more, depending on the operating system.

2. Set the following configuration parameters on the data server:

For complete information about configuring the data server, see the topic "Configuring data servers" at <http://publib.boulder.ibm.com/infocenter/iisclzos/v9r5/topic/com.ibm.svg.im.iis.prod.config.classic.doc/topics/iifcsvgpcfgov.html>.

MESSAGE POOL SIZE

The default value is 16777216. For z/OS File stage, set this parameter to a value between 33554432 and 13417728. For z/OS write operations that involve large files, set the parameter to 67108864 and then use an iterative approach to find the optimal value for your environment.

USERSUBPOOLMAX

The default value is 8192. This parameter optimizes the use of system resources by limiting the amount of memory that an individual task can use. The subpool is taken from the message pool. For z/OS write operations that involve a large batch size, use the maximum value of 214730.

INTERLEAVINT

This parameter controls how many rows are blocked to process a specific query before returning control to process other pending queries. For z/OS read and z/OS write operations, set this parameter to 10000.

STMTRETENTION

The default value is 0. The STMTRETENTION parameter defines the behavior of a prepared statement when a commit or rollback operation occurs. For z/OS read and z/OS write operations, set this parameter to 2 (2=DISCONNECT). A STMTRETENTION of 2 releases the statement only when the user disconnects from the data server. All prepared statements are retained across both COMMIT and ROLLBACK calls.

Granting users access to the Classic federation data server

The user that is specified in the z/OS File stage properties must have DBADM authority on the data server.

About this task

To grant access:

Procedure

1. Start the Classic federation data server. You can use the Classic Federation Metadata Utility on z/OS or the Classic Data Architect tool to grant users access. For an example of using the Classic Federation Metadata Utility on z/OS, see the SCACSAMP member CACGRANT.

2. Grant sequential file access to users.
GRANT DBADM ON DATABASE \$SEQUENT TO [user|group|PUBLIC]
3. Grant VSAM file access to users.
GRANT DBADM ON DATABASE \$VSAM TO [user|group|PUBLIC]

Using the z/OS File stage

Follow these step-by-step directions for using the z/OS File stage to read data from or write data to a z/OS file.

Reading data from a z/OS file

Configure the z/OS File stage to connect to a z/OS file on the Classic federation data server and read data from it.

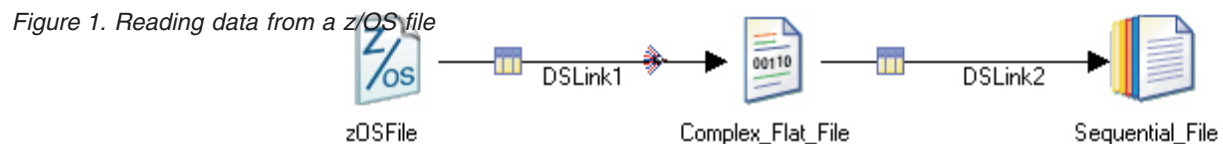
Before you begin

Complete these prerequisite tasks:

- Verify that the z/OS File stage is configured properly.
- Verify that the user name that connects to the Classic federation data server has DBADM authority.

About this task

The following figure shows an example of using the z/OS File stage to read data. The z/OS File stage identifies the z/OS file; the Complex Flat File stage converts the records into individual fields, which are passed to the Sequential File stage for further processing.



Procedure

To read data from a z/OS file, complete these tasks:

1. "Creating a job that includes the z/OS File stage and required links" on page 28.
2. "Configuring the Complex Flat File stage for use in a z/OS File stage job" on page 29.
3. "Defining a connection to the Classic federation server" on page 29.
4. "Compiling and running a job" on page 32.

Writing data to a z/OS file

Configure the z/OS File stage to connect to a z/OS file on the Classic federation data server and write data to it.

Before you begin

Complete these prerequisite tasks:

- Verify that the z/OS File stage is configured properly.
- Verify that the user name that connects to the Classic federation data server has DBADM authority.

About this task

The following figure shows an example of using the z/OS File stage to write data. Here the Complex Flat File stage reassembles the data into a record so that it can be passed on to the z/OS File stage.

Figure 2. Writing data to a z/OS file



Procedure

To write data to an IBM z/OS file, complete these tasks:

1. "Creating a job that includes the z/OS File stage and required links."
2. "Configuring the Complex Flat File stage for use in a z/OS File stage job" on page 29
3. "Defining a connection to the Classic federation server" on page 29.
4. "Specifying the write mode" on page 30.
5. "Compiling and running a job" on page 32.

Common tasks for reading and writing data

All jobs require that you complete some of these common tasks.

Creating a job that includes the z/OS File stage and required links

Before you can read data from or write data to a z/OS file, you must create a job that includes the z/OS File stage and the Complex Flat File stage, add any required additional stages, and create the necessary links.

Procedure

To create a job that includes the z/OS File stage and the required links:

1. From the IBM InfoSphere DataStage and QualityStage Designer, select **File** > **New** from the menu.
2. In the **New** window, select the Parallel Job icon, and click **OK**.
3. Follow these steps to add the z/OS File stage to the job:
 - a. In the Designer client palette, select the **File** category.
 - b. Locate zOSFile in the list of available stages.
 - c. Drag the zOSFile stage to the canvas.
4. Create the necessary links and add additional stages for the job:

- For a job that reads a z/OS file, add the Complex Flat File stage to the job. Then create an output link from the z/OS File stage to any stage that accepts an input link, for example to a Sequential file stage or a Peek stage.
- For a job that reads from a z/OS file and writes to a z/OS file, add two Complex Flat File stages after the z/OS file stage in the to the job. Then add another z/OS File stage after the two Complex Flat File stages. As an alternative, you can add a Sequential file stage and a Complex Flat File stage before the z/OS file stage in the job.

Configuring the Complex Flat File stage for use in a z/OS File stage job

For a job that uses the z/OS File stage, you specify the column mapping for the job in the Complex Flat File stage.

About this task

When the z/OS File stage is used with a Complex Flat File stage, the **Stage > File** page for the Complex Flat File stage is disabled because the file information is provided by the z/OS File stage.

Procedure

On the **Columns** tab of the Complex Flat File stage, specify how to break the incoming data into columns.

Defining a connection to the Classic federation server

To access data in a z/OS file, you must define a connection that specifies the data source name for the Classic federation server, as well as the data set, user name, and user password.

Before you begin

Complete these prerequisite tasks:

- Verify that the z/OS File stage is configured properly.
- Create a job that reads or writes data by adding the z/OS File stage and Complex Flat File stage to it and creating any required links.
- Verify that the user name that connects to the Classic federation data server has DBADM authority.

Procedure

1. Double-click the z/OS File stage icon to open the stage properties.
2. In the **Data source name** field, enter the ODBC data source name for the Classic federation driver. On UNIX and Linux, use the name of the data source that you specified in the **DATASOURCE** parameter in the file that is named in the CAC_CONFIG environment variable. On Microsoft Windows, use the name of the data source that you created by using the Microsoft ODBC Data Source Administrator.
3. In the **User name** field, enter the name of the user who has authorization to access the z/OS files that you want to read from or write to. On UNIX and Linux, if you do not enter a user name, the value of the **USERID** parameter in the file that is named in the CAC_CONFIG environment variable is used.
4. In the **User password** field, enter the password that is used to log into the Classic federation data server. On UNIX and Linux, if you do not enter a

password, the value of the **USERPASSWORD** parameter in the file that is named in the CAC_CONFIG environment variable is used.

5. In the **Dataset name** field, enter a fully qualified valid data set name.
6. If you are defining a connection for a job that reads a z/OS file, you might need to complete these additional optional fields:
 - **Volume(s)** – Enter a list of one or more volume names, separated by commas. Specify a volume name if the z/OS data set is not cataloged.
 - **Allocation type** – Choose **Shared** (default) to specify that the file can be shared with other jobs, or choose **Exclusive** to specify that the file is for use only by this job.
 - **Missing file** – Choose what to do if the file specified in the **Dataset name** field cannot be found. Choose **Error** (default), and the job terminates with an error. Choose **OK**, and the job runs but treats the missing file as an empty file.

Specifying the write mode

To configure the z/OS File stage to write records, you must specify whether to write to a new file, overwrite an existing file, or append data to an existing file.

Before you begin

Complete these prerequisite tasks:

- Create a job that includes the z/OS File stage, Complex Flat File stage, and required links.
- Define a connection to the Classic federation server.
- Configure the Complex Flat File stage.

About this task

To use the z/OS File stage to write a file that requires more than 65,635 tracks, you must use one of the following methods to preallocate space for the file and then choose **Overwrite** as the **Write mode**:

- You can pre-allocate the data set by using JCL (for example, by running a job containing an IEFBR14 step). When this is done, the DD statement that defines the data set (in addition to the other DD parameters that would normally be necessary) must also contain a DSNTYPE=LARGE parameter.
- If you use the ISPF Dataset Utility Allocation option to pre-allocate the file, set the Data set name type to LARGE (in addition to the other parameters on the ISPF panel that would normally be necessary).

Procedure

1. From the parallel canvas, double-click the z/OS File stage icon and then select the input link to edit.
2. To write to a new file, complete the following steps:
 - a. For the **Write mode** property, select **New** (default). The **DISP** parameter of the z/OS file is set to NEW.
 - b. For the **Primary amount** property, enter a positive integer to use as the value for the **SPACE** parameter for the z/OS file. The default value is 10. You can specify just the **Primary amount** or you can specify both the **Primary amount** and the **Secondary amount** . z/OS uses the primary amount for the first allocation of space. Then if the file fills up and more

space is needed, z/OS allocates the secondary amount. z/OS allocates the secondary amount 15 times, and if there is still more data to write, issues an out-of-space message.

- c. Optional: Complete any of these properties:
 - **Unit** – Enter the string to use as the value of the **UNIT** parameter for the z/OS file.
 - **Data class** – Enter the string to use as the value of the **DATACLAS** parameter for the z/OS file.
 - **Mgmt class** – Enter the string to use as the value of the **MGMTCLAS** parameter for the z/OS file.
 - **Storage class** – Enter the string to use as the value of the **STORCLAS** parameter for the z/OS file.
 - **Space type** – Choose **Tracks** (default) or **Cylinders**. This value is used for the **SPACE** parameter for the z/OS file.
 - **Secondary amount** – Enter a positive integer to use as the value for the **SPACE** parameter for the z/OS file. The default value is 10.
 - d. For the **Record format** property, choose the format for the new file.
 - e. Optional: If you set the **Record format** property to **Fixed block** or **Variable block**, you can specify a positive integer for the **Max block size** property. The default is 0, meaning that the z/OS system determines an efficient block size, depending on the type of device. For **Fixed block** format, the block size that is actually used as the **BLKSIZE** parameter for the z/OS file is the largest multiple of the **LRECL** parameter that is less than or equal to **Max block size**. Therefore, **Max block size** must be equal to or greater than the **LRECL** parameter. For **Variable block** format, the block size that is actually used as **BLKSIZE** for the z/OS file is the value specified for **Max block size** plus 4. Because z/OS adds 4 bytes to the **LRECL** parameter, the **Max block size** must be equal to or greater than the **LRECL** parameter plus 8. The **LRECL** comes from the Complex Flat File stage that is connected to the z/OS File stage. The Complex Flat File stage calculates the **LRECL** based on the lengths of the columns entered into the Complex Flat File stage.
3. To overwrite an existing file, complete the following steps:
- a. For the **Write mode** property, select **Overwrite**. If the file already exists, the **DISP** parameter of the z/OS file is set to OLD. If the file does not yet exist, the **DISP** parameter of the z/OS file is set to NEW, and a new file is created, based on the values that you specify in the following properties:
 - **Primary amount** – Enter a positive integer to use as the value for the **SPACE** parameter for the z/OS file. The default value is 10.
 - **Unit** – Enter the string to use as the value of the **UNIT** parameter for the z/OS file.
 - **Data class** – Enter the string to use as the value of the **DATACLAS** parameter for the z/OS file.
 - **Mgmt class** – Enter the string to use as the value of the **MGMTCLAS** parameter for the z/OS file.
 - **Storage class** – Enter the string to use as the value of the **STORCLAS** parameter for the z/OS file.
 - **Space type** – Choose **Tracks** (default) or **Cylinders**. This value is used for the **SPACE** parameter for the z/OS file.
 - **Secondary amount** – Enter a positive integer to use as the value for the **SPACE** parameter for the z/OS file. The default value is 10.

- b. Optional: For the **Record format** property, choose the format that matches the record format that is already specified for the existing file.
 - c. Optional: If you set the **Record format** property to **Fixed block** or **Variable block**, you can specify a positive integer for the **Max block size** property. The default is 0, meaning that the z/OS system determines an efficient block size, depending on the type of device. For **Fixed block** format, the block size that is actually used as the **BLKSIZE** parameter for the z/OS file is the largest multiple of the **LRECL** parameter that is less than or equal to **Max block size**. Therefore, **Max block size** must be equal to or greater than the **LRECL** parameter. For **Variable block** format, the block size that is actually used as **BLKSIZE** for the z/OS file is the value specified for **Max block size** plus 4. Because z/OS adds 4 bytes to the **LRECL** parameter, the **Max block size** must be equal to or greater than the **LRECL** parameter plus 8.
4. To append data to an existing file, choose **Append** as the **Write mode**. The attributes of the existing file are used, and the **DISP** parameter of the z/OS file is set to MOD.

Compiling and running a job

When you finish designing a job, compile and run it. Then use the log file to gather information that might help you adjust the job configuration or correct error conditions.

About this task

After you compile a job, you can use either the IBM InfoSphere DataStage and QualityStage Director or IBM InfoSphere DataStage and QualityStage Designer to run the job.

Procedure

1. In the Designer client, open the job that you want to compile.
2. Click **Compile**.
3. If the Compilation Status area displays errors, edit the job to resolve the errors. After resolving the errors, click **Compile** again.
4. When the job compiles successfully, click **Run**, and specify the job run options:
 - a. Enter the job parameters, as required.
 - b. Click the **Validate** button to check the job configuration without actually reading or writing any data.
 - c. Click the **Run** button to read, write, or look up data.
5. View the status of the job:
 - a. Open the Director client.
 - b. In the Status column, verify that the job was validated and completed successfully. If the job or the validation failed, choose **View > Log** to view messages that describe runtime problems.
6. If the job has runtime problems, fix the problems, recompile, validate, and run the job until it completes successfully.

Options for reading and writing data

Use these options to modify how the z/OS File stage reads and writes data.

Specifying end-of-job handling

When a job ends successfully or unexpectedly, you can specify what to do with the file.

About this task

You might want to catalog the file or keep the file when the job completes successfully or when the job ends unexpectedly and you want to have the diagnostic information to use to determine the cause of the failure. You might want to delete the file or uncatalog the file when the file is not needed after the job completes successfully or when there is no need for the file after the job ends unexpectedly.

Procedure

1. From the parallel canvas, double-click the z/OS File stage icon and then select the link to edit.
2. To choose how to handle a file when the job ends successfully, select one of the following for the **Normal EOJ handling** property. To choose how to handle a file when the job ends unexpectedly, select one of the following for the **Abnormal EOJ handling**.
 - **Catalog** (default) – Keep the file, and add it to the z/OS system catalog.
 - **Keep** – Keep the file. The file might also be cataloged if your site uses SMS.
 - **Delete** – Remove the file from the z/OS system catalog, and delete the file.
 - **Uncatalog** – Remove the file from the z/OS system catalog, but do not delete the file.

Cleaning up temporary tables

A job that includes the z/OS File stage creates a temporary table in the Classic federation repository.

About this task

The data server uses the temporary table to read from and write to z/OS data sets. When the job ends, the temporary table is deleted from the repository, and the z/OS data set is deallocated from the data server. However, if the job terminates unexpectedly, the temporary table is not deleted, and the data set is not deallocated.

Use the **Temporary table clean up** property to specify the number of days that a temporary table can remain on the Classic federation data server. To calculate the optimal value for the property, evaluate all of the z/OS File stage jobs to find the one that runs the most number of days, and then add 1 to that number. For example, if the longest z/OS File stage job runs 3 days, then enter 4.

Procedure

1. From the parallel canvas, double-click the z/OS File stage icon and then select the link to edit.
2. In the **Temporary table clean up** property, enter the maximum number of days that a temporary table can remain on the Classic federation data server. Enter 0 to delete all temporary tables, regardless of their age.

Reporting the progress of a z/OS File stage job

You can specify how many records to read or write before logging a progress message in the log file for InfoSphere DataStage.

About this task

For example, if you specify 30, after each set of 30 records are read or written, a progress message is logged in the log file. The message specifies the data set name and the total number of records that were processed. A message that specifies the total number of records that were read from or written to the file is always logged at the end of the job.

Procedure

1. From the parallel canvas, double-click the z/OS File stage icon and then select the link to edit.
2. In the **Report progress** property, enter the number of records to read or write before logging a message.

Reading only a subset of the records in a z/OS file

By default, a job reads all of the records in a z/OS file; but you can specify that the job read only a subset of the records.

About this task

Reading a subset of the records is useful while you develop a job and when you test it. By reading a small set of files, you can determine if the job is working correctly before you run the job on the entire file.

Procedure

1. From the parallel canvas, double-click the z/OS File stage icon and then select the link to edit.
2. In the **Read first *n* rows** property, enter the number of records to read. The job reads only the first *n* records in the job.

Sending records in a batch

In a z/OS File stage job that writes data, you can specify the number of records to send in a batch to the Classic federation data server.

About this task

Sending records in a batch affects job performance. The larger the batch size, the more records will be queued in IBM InfoSphere DataStage before being sent to the Classic federation server. However, keep in mind that the larger the batch size, the greater the demand for system resources.

Procedure

1. From the parallel canvas, double-click the z/OS File stage icon and then select the link to edit.
2. In the **Batch size** property, enter the number of records to send as a batch to the Classic federation data server. The number that you enter is related to the value that you specify for the FETCH BUFFER SIZE environment variable that you configure in the Classic federation configuration file on the client. The value that you specify for the **Batch size** property must be less than $(\text{FETCH BUFFER SIZE} - 108) / (\text{value of the LRECL parameter} + 6)$. The LRECL comes from the Complex Flat File stage that is connected to the z/OS File stage. The Complex Flat File stage calculates the LRECL based on the lengths of the columns entered into the Complex Flat File stage.

Releasing disk space back to the z/OS operating system

In a z/OS File stage job that writes data, you can release disk space back to the z/OS operating system.

Procedure

1. From the parallel canvas, double-click the z/OS File stage icon and then select the link to edit.
2. In the **Release** property, choose **Yes** (default) to release disk space back to the z/OS operating system. Choosing **Yes** causes the **RLSE** sub-parameter of the **SPACE** parameter to be used.

Troubleshooting z/OS File stage

To troubleshoot the z/OS File stage, look in the log files for messages that help you solve problems that occur.

Checking the log files

In most cases, all of the messages that describe how a job runs are in the log file for IBM InfoSphere DataStage. However, in some cases, additional messages about the job might be in the log for IBM InfoSphere Classic Federation Server for z/OS and in the z/OS system log. When you check the z/OS logs, look for messages that correspond to the date and time when the job ran.

Setting the logging level

To obtain additional debugging information about a job that does not run correctly, change the logging level.

- Set the environment variable `APT_ZOSFILE_DEBUG_VALUE` to 1 (trace), and detailed messages about how the job runs will be written to the job log. Other valid values for this environment variable are 2 (debug) and 3 (normal).
- On Microsoft Windows, use the InfoSphere Classic ODBC Administrator (the `cacadmin.exe` program, which is usually found in `C:\Program Files\IBM\WSClassic95\ODBC\bin`) to set the **Trace level** to 1.
- On Linux or UNIX, add the following line to the `cac.ini` file:

```
TRACE LEVEL = 1
```

If the job log for InfoSphere DataStage displays ODBC-related error and warning messages, go to <http://publib.boulder.ibm.com/infocenter/iisclzos/v9r5/topic/com.ibm.svg.im.iis.classic.msgs.doc/topics/iycsallmessages.html> for additional information.

Using containers

If you create a job that uses a container, you must include both the z/OS File stage and the Complex Flat File stage in the same container. If you do not include both stages in the container, the job does not compile.

Error message reference (z/OS File stage)

These error messages might appear in the job log for InfoSphere DataStage.

The following messages are generated by the IBM InfoSphere Classic Federation Server for z/OS, but they appear in the job log for IBM InfoSphere DataStage. For

the complete text of these messages, go to <http://publib.boulder.ibm.com/infocenter/iisclzos/v9r5/topic/com.ibm.swg.im.iis.classic.msgs.doc/topics/iycsallmessages.html>

Table 10. z/OS File stage messages

Message number	Severity	Description
0x00DA0001	Informational	No defined text; this message is used to write messages that appear on the console to the job log for Classic federation. The action to take depends on the message text. This message is not returned to the InfoSphere DataStage job log; the message is written only to the job log for Classic federation.
0x00DA0002	Informational	No defined text; this message is used to write general diagnostic trace messages to the job log for Classic federation. This message number is used when the Classic federation TRACE LEVEL parameter is set to 1, 2, or 3. This message is not be returned to the job log for InfoSphere DataStage; the message is written only to the job log for Classic federation.
0x00DA0003	Error	Internal error, initialization failed. An internal error occurred when Classic federation initialized the InfoSphere DataStage job. Call IBM support.
0x00DA0004	Error	Unrecognized command, <i>cmd</i> The InfoSphere DataStage job passed an invalid command to the Classic federation server. This error might be caused when the version of InfoSphere DataStage and the version of the Classic federation server are incompatible.
0x00DA0005	Informational	Dataset not found on request for information for <i>dsname</i> The InfoSphere DataStage job attempted to get information about a data set that does not exist. This message does not necessarily indicate that an error has occurred.
0x00DA0018	Error	Dataset, <i>dsname</i> , exclusively held by another job/user for command <i>cmd</i> A z/OS File stage job requested the exclusive use of a data set that is in use by another job or by another z/OS user. To correct this error, do one of the following: <ul style="list-style-type: none"> • Change the value of the Allocation type property in the z/OS Stage properties to Shared. • Run the z/OS File stage job when the data set is available for exclusive use. You can use a z/OS system command similar to /D GRS,RES=(hlq,*) to determine if other jobs are holding a data set.
0x00DA001c	Error	DDName, <i>ddname</i> , is already in use The Classic federation server is already using the specified DD name. The DD name is only used internally.
0x00DA0027	Informational	JCL parameters ignored for command <i>cmd</i> . JCL- <i>parms</i> ... One or more of the parameters that InfoSphere DataStage sent to the data server were ignored. The parameters that were ignored are displayed in a list.

Table 10. z/OS File stage messages (continued)

Message number	Severity	Description
0x00DA0028	Error	Internal error on command, <i>cmd</i> . Required parameter DD name passed as null. A DD name is a required parameter on many of the calls that InfoSphere DataStage makes to the data server. On the command that is specified in the message, the DD name was missing.
0x00DA0029	Error	Internal error on command, <i>cmd</i> . Required parameter ReplyMsg passed as null. An area in which to place reply message text is a required parameter on the commands that InfoSphere DataStage makes to the data server. On the command listed, the reply area was missing.
0x00DA002a	Error	Internal error on command, GETINFO. Dataset, <i>dsname</i> , found in allocation table. But value of DD name is null in allocation object. The data server encountered an unexpected problem.
0x00DA0101	Error	Incorrect disposition, <i>disposition</i> . An invalid value was used in the DISP JCL parameter. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA0102	Error	LRECL already set Two LRECL JCL parameters were passed to the data server. One parameter was a stand-alone parameter, and one was a sub-parameter of DCB. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA0103	Error	BLKSIZE already set Two BLKSIZE JCL parameters were passed to the data server. One parameter was a stand-alone parameter, and one was a sub-parameter of DCB. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA0104	Error	RECFM already set Two RECFM JCL parameters were passed to the data server. One parameter was a stand-alone parameter, and one was a sub-parameter of DCB. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA0105	Error	setDISP internal error This message indicates an internal error in the data server.
0x00DA0106	Error	SPACE unit parm invalid, <i>unit</i> The SPACE JCL parameter specified a value other than TRK or CYL. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA0107	Error	Primary space parm missing A SPACE JCL parameter is missing the required primary space amount subparameter. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.

Table 10. z/OS File stage messages (continued)

Message number	Severity	Description
0x00DA0108	Error	Non numeric in primary space parm A SPACE JCL parameter specified a non-numeric value for the primary space amount subparameter. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA0109	Error	Non numeric in secondary space parm A SPACE JCL parameter specified a non-numeric value for the secondary space amount subparameter. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA010b	Error	Comma missing between SPACE unit and primary space parms A SPACE JCL parameter has an invalid format. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA010c	Error	Comma missing in SPACE parm before RLSE A SPACE JCL parameter has an invalid format. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA010d	Error	Too many parms for SPACE primary, secondary space allocation amounts. A SPACE JCL parameter has an invalid format. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA010e	Error	Too many parms on SPACE keyword A SPACE JCL parameter has an invalid format. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA010f	Error	Embedded parens not allowed in DCB parmlist A DCB JCL parameter has an invalid format. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer. Contact IBM support.
0x00DA0110	Error	Unexpected end of DCB parmlist A DCB JCL parameter has an invalid format. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA0111	Error	Equal sign missing after DCB keyword <i>keyword</i> A DCB JCL parameter has an invalid format. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA0112	Error	Non numeric in LRECL parameter An LRECL JCL parameter has a non-numeric value. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.

Table 10. z/OS File stage messages (continued)

Message number	Severity	Description
0x00DA0113	Error	Non numeric in BLKSIZE parameter A BLKSIZE JCL parameter has a non-numeric value. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA0114	Error	Invalid RECFM value, <i>recfm-value</i> A RECFM JCL parameter has an invalid value. Valid values are F, FB, FBS, V, VB, and VBS. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA0115	Error	Invalid DSORG value, <i>dsorg-value</i> A DSORG JCL parameter has an invalid value. The only valid value is PS. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA0116	Error	Unrecognized DCB keyword, <i>keyword</i> A DCB JCL parameter has an invalid subparameter keyword value. Valid keyword values are LRECL, BLKSIZE, DSORG, and RECFM. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA0117	Error	Non numeric in LRECL parm An LRECL JCL parameter has a non-numeric value. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA0118	Error	SPACE specification missing for create of new file A SPACE JCL parameter is required to create a new data set. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA0119	Error	SPACE specification missing for create of new file A SPACE JCL parameter is required to create a new data set. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA011a	Error	DCB specification missing for create of new file A DCB JCL parameter is required to create a new data set. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA011b	Error	Invalid delimiter, <i>delim</i> , in DSN after data set name, <i>dsname</i> The message might be caused by an invalid data set name. Verify that Dataset name property of the z/OS File stage specifies a valid data set name.
0x00DA011c	Error	Invalid delimiter, <i>delim</i> , in DISP An invalid DISP JCL parameter was found. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.

Table 10. z/OS File stage messages (continued)

Message number	Severity	Description
0x00DA0131	Error	Name/value terminator, ' <i>terminator-char</i> ', missing after <i>keyword</i> A required character such as a comma or a close parenthesis was not found. This string is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA0132	Error	Equal sign missing after keyword, <i>keyword</i> An required equal sign is missing. This string is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA0133	Error	Comma missing after keyword, <i>keyword</i> A required comma is missing. This string is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA0134	Error	Unrecognized JCL parameter An unrecognized JCL parameter was found. JCL is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA013e	Error	The specified member name, <i>member-name</i> , on data set, <i>dsname</i> , is invalid The PDS or PDSE member name has an invalid format or the member does not exist. Specify the correct value for the Dataset name property of the of the z/OS File stage.
0x00DA0200	Error	Call to AllocSQLMsg failed for error code <i>error-code</i> . <i>Descriptive-error-information</i> An error occurred in the data server. This message is not returned to the job log forInfoSphere DataStage. The message is written to the job log for Classic federation and to the z/OS system log.
0x00DA0201	Error	<i>Cmd error Descriptive-error-text</i> The dynamic z/OS data set allocation routine that the data server called failed. The descriptive error text might contain z/OS codes and messages. The job log for Classic federation and the z/OS system log might contain other messages. Use these messages to diagnose and correct the problem. For more information about these codes and messages, see the z/OS V1R10.0 MVS Programming Authorized Assembler Services Guide, which is available at http://publib.boulder.ibm.com/infocenter/zos/v1r10/index.jsp?topic=/com.ibm.zos.r10.iea/iea.htm
0x00DA0202	Error	Name/token, <i>name/token</i> , not found for, <i>name-token-cmd</i> An invalid request, parameter, or parameter value was sent to the data server. The request string is generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA0203	Error	Name/token error <i>rc</i> returned by request, <i>name-token-cmd</i> , for token name, <i>name-token</i> . An invalid request, parameter, or parameter value was sent to the data server. The request string is generated internally and cannot be controlled by the InfoSphere DataStage job developer.

Table 10. z/OS File stage messages (continued)

Message number	Severity	Description
0x00DA0204	Error	Error accessing internal allocation table, <i>func</i> . Error description is <i>alloc-table-error-desc</i> . An error occurred when the data server was accessing an internal allocation table.
0x00DA0205	Error	Memory allocation error An error occurred when the data server was allocating memory.
0x00DA0206	Error	Invalid parms on GETINFO request When the z/OS job requested information about a file, the data server encountered an invalid parameter on the request. This message might occur because the Dataset name property in the z/OS File stage properties specifies an invalid data set name. Specify a valid data set name. If data set name is valid and this message occurs, contact IBM support.
0x00DA0207	Error	Catalog interface error <i>Descriptive-information</i> The data server attempted to get information about a data set and encountered an error. This message might occur because the Dataset name property in the z/OS File stage properties specifies an invalid data set. Specify a valid data set name and verify that the file exists on z/OS. Correct the error and then run the job again.
0x00DA0209	Error	<i>Cmd error Descriptive-error-text</i> The data server attempted to delete a member of a PDS or PDSE and encountered an error. This message might occur because the member does not exist. If the member exists and this message occurs, contact IBM support.
0x00DA0210	Error	DD name <i>ddname</i> not defined A GETINFO request was made to process a file allocated to <i>ddname</i> , and the DD name was not allocated to the data server address space. DD names are generated internally and cannot be controlled by the InfoSphere DataStage job developer.
0x00DA0211	Error	DD name <i>ddname</i> refers to an HFS file A GETINFO request was made to process a file allocated to <i>ddname</i> , and the file was an HFS file. Classic federation does not support HFS files. Enter an MVS data set name in the Dataset name property in the z/OS File stage properties and then run the job again.

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.

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.

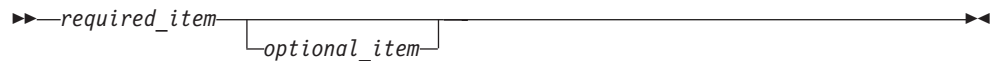
Appendix C. How to read syntax diagrams

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).



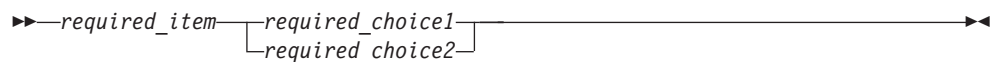
- Optional items appear below the main path.



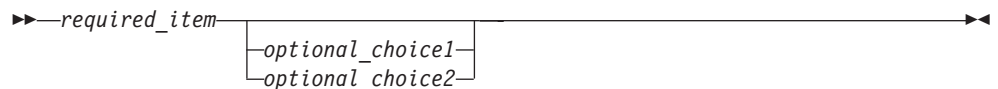
If an optional item appears above the main path, that item has no effect on the execution of the syntax element and is used only for readability.



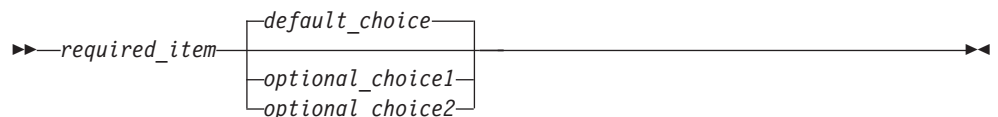
- If you can choose from two or more items, they appear vertically, in a stack. If you must choose one of the items, one item of the stack appears on the main path.



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



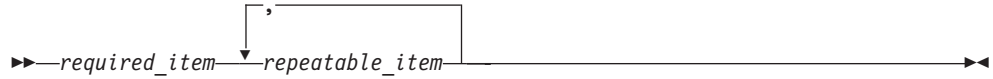
If one of the items is the default, it appears above the main path, and the remaining choices are shown below.



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

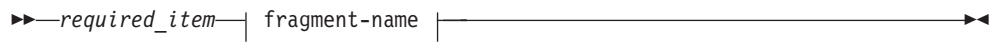


If the repeat arrow contains a comma, you must separate repeated items with a comma.

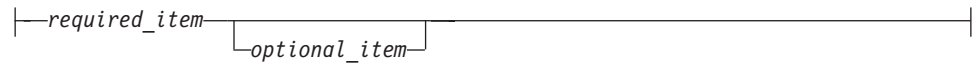


A repeat arrow above a stack indicates that you can repeat the items in the stack.

- 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 11. IBM resources

Resource	Description and location
IBM Support Portal	You can customize support information by choosing the products and the topics that interest you at www.ibm.com/support/entry/portal/Software/Information_Management/InfoSphere_Information_Server
Software services	You can find information about software, IT, and business consulting services, on the solutions site at www.ibm.com/businesssolutions/
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 www.ibm.com/account/
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 http://www.ibm.com/training
IBM representatives	You can contact an IBM representative to learn about solutions at www.ibm.com/connect/ibm/us/en/

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/.
- Send your comments by e-mail to 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 12. 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"> • Session • Persistent 	User name	<ul style="list-style-type: none"> • Session management • Authentication 	Cannot be disabled
Any (part of InfoSphere Information Server installation)	InfoSphere Metadata Asset Manager	<ul style="list-style-type: none"> • Session • Persistent 	No personally identifiable information	<ul style="list-style-type: none"> • Session management • Authentication • Enhanced user usability • Single sign-on configuration 	Cannot be disabled

Table 12. 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"> • Session • Persistent 	<ul style="list-style-type: none"> • User name • Digital signature • Session ID 	<ul style="list-style-type: none"> • Session management • Authentication • Single sign-on configuration 	Cannot be disabled
InfoSphere DataStage	XML stage	Session	Internal identifiers	<ul style="list-style-type: none"> • Session management • Authentication 	Cannot be disabled
InfoSphere DataStage	IBM InfoSphere DataStage and QualityStage Operations Console	Session	No personally identifiable information	<ul style="list-style-type: none"> • Session management • Authentication 	Cannot be disabled
InfoSphere Data Click	InfoSphere Information Server web console	<ul style="list-style-type: none"> • Session • Persistent 	User name	<ul style="list-style-type: none"> • Session management • Authentication 	Cannot be disabled
InfoSphere Data Quality Console		Session	No personally identifiable information	<ul style="list-style-type: none"> • Session management • Authentication • Single sign-on configuration 	Cannot be disabled
InfoSphere QualityStage Standardization Rules Designer	InfoSphere Information Server web console	<ul style="list-style-type: none"> • Session • Persistent 	User name	<ul style="list-style-type: none"> • Session management • Authentication 	Cannot be disabled
InfoSphere Information Governance Catalog		<ul style="list-style-type: none"> • Session • Persistent 	<ul style="list-style-type: none"> • User name • Internal identifiers • State of the tree 	<ul style="list-style-type: none"> • Session management • Authentication • Single sign-on configuration 	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 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.

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™ 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^{Link}, ZIP, ZIP + 4, ZIP Code, Post Office, Postal Service, USPS and United States Postal Service. IBM Corporation is a non-exclusive DPV and LACS^{Link} licensee of the United States Postal Service.

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

Index

C

- cac.ini file 19
- Classic federation
 - configuring 20
 - configuring for z/OS File stage 20
- Classic Federation enterprise stage 3
 - Combinability mode 5
 - Execution mode 5
 - NLS Map 6
 - Partitioning 7
 - Preserver partitioning
 - Node map constraint 5
 - Node pool and resource constraints 5
 - Properties 5
 - read 3
 - update 3
 - write 3
- Classic federation server
 - connecting to 29
- clients
 - installing 22
- command-line syntax
 - conventions 45
- commands
 - syntax 45
- configuring
 - Classic federation 20
- customer support
 - contacting 49

D

- data source
 - creating 24, 25

E

- environment variables
 - CAC_CONFIG 19
- error messages
 - z/OS File stage 35

F

- FETCH BUFFER SIZE parameter
 - configuring 25

I

- Input page
 - Operations
 - classicfedupsert 6
 - classicfedwrite 7
 - Connection 6
 - Options 6
 - Target 6
- installing
 - Classic federation 20

- installing (*continued*)
 - ODBC 22
- INTERLEAVINT parameter
 - configuring 25

J

- jobs
 - compiling 32

L

- legal notices 55

M

- MESSAGE POOL SIZE parameter
 - configuring 25
- metadata catalog
 - updating 21
- METALOAD command
 - using 21

O

- ODBC
 - installing 22
- ODBC driver
 - configuring 24, 25
- Output page
 - classicfedlookup
 - Connection 15
 - Options 15
 - Source 15
 - classicfedupsert for reject links 17
 - Operations
 - classicfedread 11
 - Property
 - Query 11
 - Read Method 11
 - Table 11

P

- parameters
 - Classic federation 20
- product accessibility
 - accessibility 43
- product documentation
 - accessing 51

S

- software services
 - contacting 49
- special characters
 - in command-line syntax 45

- STMTRETENTION parameter
 - configuring 25
- support
 - customer 49
- syntax
 - command-line 45

T

- trademarks
 - list of 55
- troubleshooting
 - ODBC 22
 - z/OS File stage 35

U

- user access
 - granting to Classic Federation Server 26
- USERSUBPOOLMAX parameter
 - configuring 25

W

- web sites
 - non-IBM 47

Z

- z/OS File stage
 - configuring 20, 27
 - end-of-job handling 33
 - logging progress 34
 - optional properties 32
 - overview 19
 - reading 34
 - reading data 27, 28
 - releasing disk space 35
 - sending batches of records 34
 - temporary table cleanup 33
 - troubleshooting 35
 - writing data 28, 30
- z/OS files
 - reading 27, 28
 - writing 28
- z.OS File stage
 - error messages 35
 - troubleshooting 35



Printed in USA

SC19-4258-00

