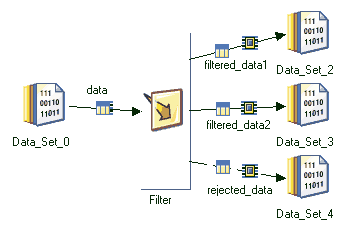
The Filter stage is a processing stage. This stage transfers, unmodified, the records of the input data set which satisfy the specified requirements and filters out all other records.

The Filter stage is a processing stage. It can have a single input link and a any number of output links and, optionally, a single reject link.

The Filter stage transfers, unmodified, the records of the input data set which satisfy the specified requirements and filters out all other records. You can specify different requirements to route rows down different output links. The filtered out records can be routed to a reject link, if required.



The stage editor has three pages:

* [**Stage Page**](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/c_deeref_Stage_Page_filter_stage.html?view=kc). This is always present and is used to specify general information about the stage.
* [**Input Page**](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/c_deeref_Input_Page_filter_stage.html?view=kc). This is where you specify details about the input link carrying the data to be filtered.
* [**Output Page**](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/c_deeref_Outputs_Page_filter_stage.html?view=kc). This is where you specify details about the filtered data being output from the stage down the various output links.

Lists the elements that you can use for specifying the expressions. The operation of the filter stage is governed by the expressions that you set.

## About this task

The operation of the filter stage is governed by the expressions you set in the **Where** property on the Properties tab. You can use the following elements to specify the expressions:

* Input columns.
* Requirements involving the contents of the input columns.
* Optional constants to be used in comparisons.
* The Boolean operators AND and OR to combine requirements.

When a record meets the requirements, it is written unchanged to the specified output link. The **Where** property supports standard SQL expressions, except when comparing strings.

When quoting in the filter, you should use single, not double, inverted commas.

* [**Input data columns**](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/c_deeref_Input_Data_Columns.html?view=kc)  
  The single column specified for evaluation can be of any data type. If you compare columns they must be of the same or compatible data types. Column data type conversion is based on the rules mentioned in this section.
* [**Supported Boolean expressions and operators**](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/c_deeref_Supported_Boolean_Expressions_and_Operators.html?view=kc)  
  Certain Boolean expressions and operators are supported.
* [**String comparison**](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/c_deeref_String_Comparison.html?view=kc)  
  InfoSphere® DataStage® sorts string values according to general rules.
* [**Examples**](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/g_deeref_Examples_filter_stage.html?view=kc)  
  The following give some example **Where** properties.

The single column specified for evaluation can be of any data type. If you compare columns they must be of the same or compatible data types. Column data type conversion is based on the rules mentioned in this section.

If you specify a single column for evaluation, that column can be of any data type. Note that InfoSphere® DataStage®’s treatment of strings differs slightly from that of standard SQL. If you compare columns they must be of the same or compatible data types. Otherwise, the operation terminates with an error. Compatible data types are those that InfoSphere DataStageconverts by default. Regardless of any conversions the whole row is transferred unchanged to the output. If the columns are not compatible upstream of the filter stage, you can convert the types by using a Modify stage prior to the Filter stage.

Column data type conversion is based on the following rules:

* Any integer, signed or unsigned, when compared to a floating-point type, is converted to floating-point.
* Comparisons within a general type convert the smaller to the larger size (sfloat to dfloat, uint8 to uint16, and so on.)
* When signed and unsigned integers are compared, unsigned are converted to signed.
* Decimal, raw, string, time, date, and timestamp do not figure in type conversions. When any of these is compared to another type, filter returns an error and terminates.

The input field can contain nulls. If it does, null values are less than all non-null values, unless you specify the operators's **nulls last** option.

**Note**The conversion of numeric data types might result in a loss of range and cause incorrect results. InfoSphere DataStagedisplays a warning message to that effect when range is lost.

Certain Boolean expressions and operators are supported.

The following list summarizes the Boolean expressions that are supported. In the list, BOOLEAN denotes any Boolean expression.

* true
* false
* six comparison operators: =, <>, <, >, <=, >=
* is null
* is not null
* like 'abc' (the second operand must be a regular expression)
* between (for example, A between B and C is equivalent to B <= A and A> = C)
* not BOOLEAN
* BOOLEAN is true
* BOOLEAN is false
* BOOLEAN is not true
* BOOLEAN is not false

Any of these can be combined using AND or OR.

* [**Order of association**](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/c_deeref_Order_of_Association.html?view=kc)

The order of association of expressions is from left to right.

The order of association of expressions is from left to right.

As in SQL, expressions are associated left to right. AND and OR have the same precedence. You might group fields and expressions in parentheses to affect the order of evaluation.

InfoSphere® DataStage® sorts string values according to general rules.

The string values are sorted according to the following rules:

* Characters are sorted in lexicographic order.
* Strings are evaluated by their ASCII value.
* Sorting is case sensitive, that is, uppercase letters appear before lowercase letter in sorted data.
* Null characters appear before non-null characters in a sorted data set, unless you specify the **nulls last** option.
* Byte-for-byte comparison is performed.

You want to compare columns number1 and number2.

If the data in column number1 is greater than the data in column number2, the corresponding records are to be written to output link 2.

You enter the following in the **Where** property:

name1 > name2[Copy](javascript:void(0);)

You then select output link 2 in the dependent **Output Link** property. (You use the Link Ordering tab to specify the number order of the output links).

Use this property to check if the column serial number contains a null.

You want to test column serialno to see if it contains a null. If it does, you want to write the corresponding records to the output link.

You enter the following in the **Where** property:

serialno is null[Copy](javascript:void(0);)

In this example the stage only has one output link. You do not need to specify the **Output Link** property because the stage will write to the output link by default.

You want to evaluate each input row to see if certain conditions prevail.

You want to evaluate each input row to see if these conditions prevail:

* EITHER all the following are true
  + Column number1 does not have the value 0
  + Column number2 does not have the value 3
  + Column number3 has the value 0
* OR column name equals the string ZAG

You enter the following in the **Where** property:

number1 <> 0 and number2 <> 3 and number3 = 0 or name = 'ZAG'[Copy](javascript:void(0);)

If these conditions are met, the stage writes the row to the output link.

This section specifies the minimum steps to take to get a Filter stage functioning.

**About this task**

InfoSphere® DataStage® has many defaults which means that it can be very easy to include Filter stages in a job. InfoSphere DataStage provides a versatile user interface, and there are many shortcuts to achieving a particular end, this section describes the basic method, you will learn where the shortcuts are when you get familiar with the product.

To use a Filter stage:

* In the Stage page [**Properties Tab**](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/r_deeref_Properties_Tab_filter_stage.html?view=kc):
  + Supply the specifications that determine which records are accepted and which are filtered out. This is given in the form of a Where clause. You can multiple statements each applying to different links.
  + Specify which Where clause correspond to which output links.
  + Specify whether rows that fail to satisfy any of the Where clauses will be routed to a reject link.
  + Specify whether rows are output only for the first Where clause they satisfy, or for any clauses they satisfy.
* In the Stage page [**Link Ordering Tab**](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/r_deeref_Link_Ordering_Tab_filter_stage.html?view=kc), specify which order the output links are processed in. This is important where you specify that rows are only output for the first Where clause that they satisfy.
* Ensure that meta data has been defined for input link and output links, and reject link, if applicable.
* In the Output Page [**Mapping Tab**](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/r_deeref_Mapping_Tab_filter_stage.html?view=kc), specify how the input columns of the data set being filtered map onto the columns of the various output links.

The General tab allows you to specify an optional description of the stage. The Properties tab lets you specify what the stage does. The Advanced tab allows you to specify how the stage executes. The Link Ordering tab allows you to specify what order the output links are processed in. The NLS Locale tab appears if your have NLS enabled on your system. It allows you to select a locale other than the project default to determine collating rules.

Use the Properties tab to specify how the Filter stage operates.

The Properties tab allows you to specify properties which determine what the stage actually does. Some of the properties are mandatory, although many have default settings. Properties without default settings appear in the warning color (red by default) and turn black when you supply a value for them.

The following table gives a quick reference list of the properties and their attributes. A more detailed description of each property follows.

| **Category/Property** | **Values** | **Default** | **Mandatory?** | **Repeats?** | **Dependent of** |
| --- | --- | --- | --- | --- | --- |
| Predicates/[Where clause](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/r_deeref_Predicates_Category.html?view=kc) | string | N/A | Y | Y | N/A |
| Predicates/[Output link](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/r_deeref_Predicates_Category.html?view=kc) | Output link | N/A | Y | N | Where clause |
| Options/[Output rejects](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/r_deeref_Options_Category_filter_stage.html?view=kc) | True/False | False | Y | N | N/A |
| Options/[Output rows only once](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/r_deeref_Options_Category_filter_stage.html?view=kc) | True/False | False | Y | N | N/A |
| Options/[Nulls value](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/r_deeref_Options_Category_filter_stage.html?view=kc) | Less Than/Greater Than | Less Than | N | N | N/A |
| *Table 1. Properties* | | | | | |

Use the Properties tab to specify how the Filter stage operates on the Predicates category.

## Filter stage: Where clause

Specify a Where statement that a row must satisfy in order to be routed down this link. This is like an SQL Where clause, see ["Specifying the Filter"](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/t_deeref_Specifying_the_Filter.html?view=kc) for details.

## Output link

Specify the output link corresponding to the Where clause.

Use the Options category to specify how the Filter stage operates.

## Output rejects

Set this to true to output rows that satisfy no Where clauses down the reject link (remember to specify which link is the reject link on the parallel job canvas).

## Output rows only once

Set this to true to specify that rows are only output down the link of the first Where clause they satisfy. Set to false to have rows output down the links of all Where clauses that they satisfy.

## Nulls value

Specify whether null values are treated as greater than or less than other values.

This tab allows you to specify options.

This tab allows you to specify the following:

* **Execution Mode**. The stage can execute in parallel mode or sequential mode. In parallel mode the input data is processed by the available nodes as specified in the Configuration file, and by any node constraints specified on the Advanced tab. In Sequential mode the entire data set is processed by the conductor node.
* **Combinability mode**. This is Auto by default, which allows InfoSphere® DataStage® to combine the operators that underlie parallel stages so that they run in the same process if it is sensible for this type of stage.
* **Preserve partitioning**. This is **Propagate** by default. It adopts the setting of the previous stage.You can explicitly select **Set**or **Clear**. Select **Set** to request the stage should attempt to maintain the partitioning.
* **Node pool and resource constraints**. Select this option to constrain parallel execution to the node pool or pools or resource pool or pools specified in the grid. The grid allows you to make choices from drop down lists populated from the Configuration file.
* **Node map constraint**. Select this option to constrain parallel execution to the nodes in a defined node map. You can define a node map by typing node numbers into the text box or by clicking the browse button to open the Available Nodes dialog box and selecting nodes from there. You are effectively defining a new node pool for this stage (in addition to any node pools defined in the Configuration file).

**Note**In the **Node map constraint** text box, you can enter jobs parameters as well as numbers. You can enter a single parameter, for example #testnode#, or you can enter a comma separated lists of parameters, for example #testnode#, #testnode2#. The browse button next to the text box will display a list of the node names from the last configuration file that was referenced by the job, but the browse button will not display the node names that were specified by the job parameters.

This tab allows you to specify the order in which output links are processed. This is important where you have set the Output rows only once property to True.

For the Filter stage, the NLS Locale tab appears if you have NLS enabled on your system. It lets you view the current default collate convention, and select a different one for this stage if required.

You can also use a job parameter to specify the locale, or browse for a file that defines custom collate rules. The collate convention defines the order in which characters are collated. The Filter stage uses this when evaluating Where clauses. Select a locale from the list, or click the arrow button next to the list to use a job parameter or browse for a collate file.

The Input page allows you to specify details about the data set being filtered. There is only one input link.

The Input page allows you to specify details about the data set being filtered. There is only one input link.

The General tab allows you to specify an optional description of the link. The Partitioning tab allows you to specify how incoming data on the source data set link is partitioned. The Columns tab specifies the column definitions of incoming data. The Advanced tab allows you to change the default buffering settings for the input link.

Details about Filter stage partitioning are given in the following section. See ["Stage Editors,"](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/c_deeref_Stage_Editors.html?view=kc) for a general description of the other tabs.

* [**Filter stage: Partitioning on input links**](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/r_deeref_Partitioning_on_Input_Links_filter_stage.html?view=kc)

The Partitioning tab allows you to specify details about how the incoming data is partitioned or collected before the filter is performed.

The Partitioning tab allows you to specify details about how the incoming data is partitioned or collected before the filter is performed.

By default the stage uses the auto partitioning method.

If the Filter stage is operating in sequential mode, it will first collect the data before writing it to the file using the default auto collection method.

The Partitioning tab allows you to override this default behavior. The exact operation of this tab depends on:

* Whether the Filter stage is set to execute in parallel or sequential mode.
* Whether the preceding stage in the job is set to execute in parallel or sequential mode.

If the Filter stage is set to execute in parallel, then you can set a partitioning method by selecting from the **Partition type** drop-down list. This will override any current partitioning.

If the Filter stage is set to execute in sequential mode, but the preceding stage is executing in parallel, then you can set a collection method from the **Collector type** drop-down list. This will override the default auto collection method.

The following partitioning methods are available:

* **(Auto)**. InfoSphere® DataStage® attempts to work out the best partitioning method depending on execution modes of current and preceding stages and how many nodes are specified in the Configuration file. This is the default method for the Filter stage.
* **Entire**. Each file written to receives the entire data set.
* **Hash**. The records are hashed into partitions based on the value of a key column or columns selected from the **Available**list.
* **Modulus**. The records are partitioned using a modulus function on the key column selected from the **Available** list. This is commonly used to partition on tag fields.
* **Random**. The records are partitioned randomly, based on the output of a random number generator.
* **Round Robin**. The records are partitioned on a round robin basis as they enter the stage.
* **Same**. Preserves the partitioning already in place.
* **Db2®**. Replicates the Db2 partitioning method of a specific Db2 table. Requires extra properties to be set. Access these properties by clicking the properties button.
* **Range**. Divides a data set into approximately equal size partitions based on one or more partitioning keys. Range partitioning is often a preprocessing step to performing a total sort on a data set. Requires extra properties to be set. Access these properties by clicking the properties button.

The following Collection methods are available:

* **(Auto)**. This is the default collection method for the Filter stage. Normally, when you are using Auto mode, InfoSphere DataStage will eagerly read any row from any input partition as it becomes available.
* **Ordered**. Reads all records from the first partition, then all records from the second partition, and so on.
* **Round Robin**. Reads a record from the first input partition, then from the second partition, and so on. After reaching the last partition, the operator starts over.
* **Sort Merge**. Reads records in an order based on one or more columns of the record. This requires you to select a collecting key column from the **Available** list.

The Partitioning tab also allows you to specify that data arriving on the input link should be sorted before the remove duplicates operation is performed. The sort is always carried out within data partitions. If the stage is partitioning incoming data the sort occurs after the partitioning. If the stage is collecting data, the sort occurs before the collection. The availability of sorting depends on the partitioning or collecting method chosen (it is not available for the default auto methods).

Select the check boxes as follows:

* **Perform Sort**. Select this to specify that data coming in on the link should be sorted. Select the column or columns to sort on from the **Available** list.
* **Stable**. Select this if you want to preserve previously sorted data sets. This is the default.
* **Unique**. Select this to specify that, if multiple records have identical sorting key values, only one record is retained. If stable sort is also set, the first record is retained.

If NLS is enabled an additional button opens a dialog box allowing you to select a locale specifying the collate convention for the sort.

You can also specify sort direction, case sensitivity, whether sorted as ASCII or EBCDIC, and whether null columns will appear first or last for each column. Where you are using a keyed partitioning method, you can also specify whether the column is used as a key for sorting, for partitioning, or for both. Select the column in the Selected list and right-click to invoke the shortcut menu.

The Output page allows you to specify details about data output from the Filter stage.

The Filter stage can have any number of output links, plus one reject link. Choose the one you want to work on from the **Output name** drop down list.

The General tab allows you to specify an optional description of the output link. The Columns tab specifies the column definitions of the data. The Mapping tab allows you to specify the relationship between the columns being input to the Filter stage and the output columns. The Advanced tab allows you to change the default buffering settings for the output links.

Details about Filter stage mapping is given in the following section. See ["Stage Editors,"](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/c_deeref_Stage_Editors.html?view=kc) for a general description of the other tabs.

* [**Filter stage: Mapping tab**](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/r_deeref_Mapping_Tab_filter_stage.html?view=kc)

For the Filter stage, the Mapping tab allows you to specify how the output columns are derived, that is, what filtered columns map onto them.

For the Filter stage, the Mapping tab allows you to specify how the output columns are derived, that is, what filtered columns map onto them.

The left pane shows the filtered columns. These are read only and cannot be modified on this tab.

The right pane shows the output columns for the output link. This has a **Derivations** field where you can specify how the column is derived. You can fill it in by dragging copied columns over, or by using the Auto-match facility.