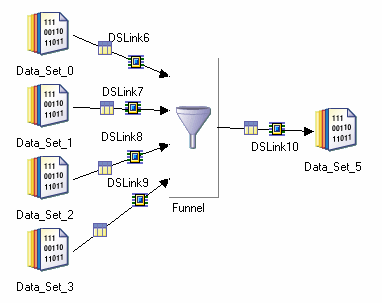
The Funnel stage is a processing stage that copies multiple input data sets to a single output data set. You can use this operation for combining separate data sets into a single large data set.

The Funnel stage is a processing stage. It copies multiple input data sets to a single output data set. This operation is useful for combining separate data sets into a single large data set. The stage can have any number of input links and a single output link.



The Funnel stage can operate in one of three modes:

* **Continuous Funnel** combines the records of the input data in no guaranteed order. It takes one record from each input link in turn. If data is not available on an input link, the stage skips to the next link rather than waiting.
* **Sort Funnel** combines the input records in the order defined by the value(s) of one or more key columns and the order of the output records is determined by these sorting keys.
* **Sequence** copies all records from the first input data set to the output data set, then all the records from the second input data set, and so on.

For all methods the meta data of all input data sets must be identical.

The sort funnel method has some particular requirements about its input data. All input data sets must be sorted by the same key columns as to be used by the Funnel operation.

Typically all input data sets for a sort funnel operation are hash-partitioned before they're sorted (choosing the auto partitioning method will ensure that this is done). Hash partitioning guarantees that all records with the same key column values are located in the same partition and so are processed on the same node. If sorting and partitioning are carried out on separate stages before the Funnel stage, this partitioning must be preserved.

The **sortfunnel** operation allows you to set one *primary* key and multiple *secondary* keys. The Funnel stage first examines the primary key in each input record. For multiple records with the same primary key value, it then examines secondary keys to determine the order of records it will output.

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_funnel_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_Inputs_Page_funnel_stage.html?view=kc). This is where you specify details about the data sets being joined.
* [**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_funnel_stage.html?view=kc). This is where you specify details about the joined data being output from the stage.

The example data comprises four separate data sets. Each data set contains a list of GlobalCo customers from a particular country.

Here is a sample of the data for the US customers:

"JON SMITH","789 LEDBURY ROAD","GC13849","GlobalCoUS"

"MARY GARDENER","127 BORDER ST","GC13933","GlobalCoUS"

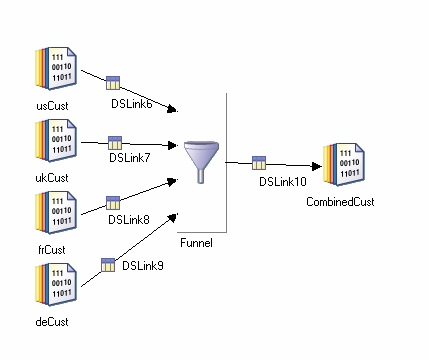
"CHRIS TRAIN","1400 NEW ST","GC14036","GlobalCoUS"

"HUW WILLIAMS","579 DIGBETH AVENUE","GC14127","GlobalCoUS"

"SARA PEARS","45 ALCESTER WAY","GC14263","GlobalCoUS"

"LUC TEACHER","3 BIRMINGHAM ROAD","GC14346","GlobalCoUS"[Copy](javascript:void(0);)

The Funnel stage, when set to continuous funnel, will combine these into a single data set. The job to perform the funnel is as follows:



The continuous funnel method is selected on the Stage page **Properties** tab of the Funnel stage:

The continuous funnel method does not attempt to impose any order on the data it is processing. It simply writes rows as they become available on the input links. In the example the stage has written a row from each input link in turn. A sample of the final, funneled, data is as follows:

"JON SMITH","789 LEDBURY ROAD","GC13849","GlobalCoUS"

"ELIZABETH PARKER","35 YORK ROAD","GC21745","GlobalCoUK"

"JEAN DUPONT","576 RUE DE PARIS","GC20002","GlobalCoFR"

"ADELE BECKER","AM HUNGERSBERG 123","GC22145","GlobalCoDE"

"MARY GARDENER","127 BORDER ST","GC13933","GlobalCoUS"

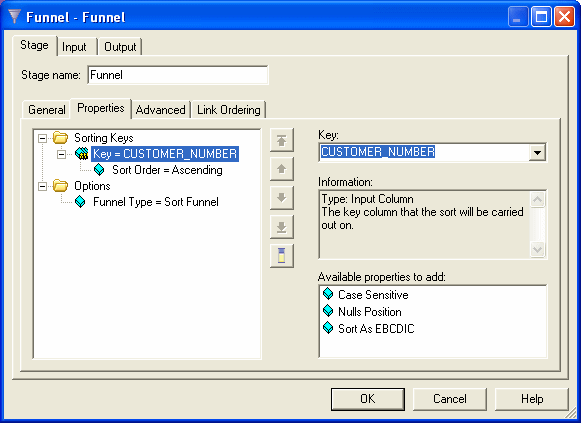
"WINSTON HILL","87 MULBERRY CLOSE","GC21874","GlobalCoUK"

"MARIE TISON","14 AVENUE DE CALAIS","GC20012","GlobalCoFR"

"KLAUS SCHMIDT","WOLBURGSWEG 7645","GC22478","GlobalCoDE"[Copy](javascript:void(0);)

This example shows the use of the funnel stage to sort the GlobalCo data by customer number.

In this example we are going to use the funnel stage to sort the GlobalCo data by customer number as it combines the data into a single data set. The data and the basic job are the same as for the Continuous Funnel example, but now we set the Funnel stage properties as follows:



The following is a sample of the output data set:

"JON SMITH","789 LEDBURY ROAD","GC13849","GlobalCoUS"

"MARY GARDENER","127 BORDER ST","GC13933","GlobalCoUS"

"CHRIS TRAIN","1400 NEW ST","GC14036","GlobalCoUS"

"HUW WILLIAMS","579 DIGBETH AVENUE","GC14127","GlobalCoUS"

"SARA PEARS","45 ALCESTER WAY","GC14263","GlobalCoUS"

"LUC TEACHER","3 BIRMINGHAM ROAD","GC14346","GlobalCoUS"

"JEAN DUPONT","576 RUE DE PARIS","GC20002","GlobalCoFR"

"MARIE TISON","14 AVENUE DE CALAIS","GC20012","GlobalCoFR"

"PIERRE FOURNIER","321 RUE VOLTAIRE","GC20021","GlobalCoFR"

"LOUIS LEROY","3 RUE DES TREUILS","GC20032","GlobalCoFR"

"NICOLE GIRARD","1234 QUAI DE LA TOURNELLE","GC20040","GlobalCoFR"

"DANIELLE BLANC","987 BOULEVARD AUXERRE","GC20049","GlobalCoFR"/sd/[Copy](javascript:void(0);)

**Note**If you are running your sort funnel stage in parallel, you should be aware of the various considerations about sorting data and partitions. These are described in ["Sort Stage."](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/c_deeref_Sort_Stage.html?view=kc)

In this example you funnel the GlobalCo data on input one data set at a time. The Funnel type property is set to Sequence on the Properties tab.

You get a data set that contains all the US customers, then all the UK ones, then all the French ones and so on. Again the basic job and the source data are the same as for the continuous funnel example.

When using the sequence method, you need to specify the order in which the Funnel stage processes its input links, as this affects the order of the sequencing. This is done on the Stage page Link Ordering tab.

If you run the sequence funnel stage in parallel, you need to be mindful of the effects of data partitioning. If, for example, you ran the example job on a four-node system, you would get four partitions each containing a section of US data, a section of UKdata, a section of FR data and so on.

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

**About this task**

InfoSphere® DataStage® has many defaults which means that it can be very easy to include Funnel 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 Funnel 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_funnel_stage.html?view=kc), specify the Funnel Type. Continuous Funnel is the default, but you can also choose Sequence or Sort Funnel.

If you choose to use the Sort Funnel method, you also need to specify the key on which data will be sorted. You can repeat the key property to specify a composite key.

* If you are using the Sequence method, 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_funnel_stage.html?view=kc) specify the order in which your data sets will be combined.
* 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_funnel_stage.html?view=kc), specify how the output columns are derived.

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 which order the input 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 Funnel 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** |
| --- | --- | --- | --- | --- | --- |
| Options/[Funnel Type](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/r_deeref_Options_Category_funnel_stage.html?view=kc) | Continuous Funnel/ Sequence/ Sort funnel | Continuous Funnel | Y | N | N/A |
| Sorting Keys/[Key](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/r_deeref_Sorting_Keys_Category_funnel_stage.html?view=kc) | Input Column | N/A | Y (if Funnel Type = Sort Funnel) | Y | N/A |
| Sorting Keys/[Sort Order](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/r_deeref_Sorting_Keys_Category_funnel_stage.html?view=kc) | Ascending/ Descending | Ascending | Y (if Funnel Type = Sort Funnel) | N | Key |
| Sorting Keys/[Nulls position](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/r_deeref_Sorting_Keys_Category_funnel_stage.html?view=kc) | First/Last | First | Y (if Funnel Type = Sort Funnel) | N | Key |
| Sorting Keys/[Case Sensitive](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/r_deeref_Sorting_Keys_Category_funnel_stage.html?view=kc) | True/False | True | N | N | Key |
| Sorting Keys/[Sort as EBCDIC](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/r_deeref_Sorting_Keys_Category_funnel_stage.html?view=kc) | True/False | False | N | N | Key |
| *Table 1. Properties* | | | | | |

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

**Funnel type**

Specifies the type of Funnel operation. Choose from:

* Continuous Funnel
* Sequence
* Sort Funnel

The default is Continuous Funnel.

Use the Sorting keys category to specify how the Funnel stage operates.

**Key**

This property is only required for Sort Funnel operations. Specify the key column that the sort will be carried out on. The first column you specify is the primary key, you can add multiple secondary keys by repeating the key property. You can use the Column Selection dialog box to select several keys at once if required.

Key has the following dependent properties:

* **Sort Order**

Choose Ascending or Descending. The default is Ascending.

* **Nulls position**

By default columns containing null values appear first in the funneled data set. To override this default so that columns containing null values appear last in the funneled data set, select Last.

* **Sort as EBCDIC**

To sort as in the EBCDIC character set, choose True.

* **Case Sensitive**

Use this to specify whether each key is case sensitive or not, this is set to True by default, that is, the values "CASE" and "case" would not be judged equivalent.

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 which results from ORing the settings of the input stages, that is, if any of the input stages uses **Set** then this stage will use **Set**. You can explicitly select **Set** or **Clear**. Select **Set** to request that the next stage in the job attempts 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 links input to the Funnel stage are processed. This is only relevant if you have chosen the Sequence Funnel Type.

By default the input links will be processed in the order they were added. To rearrange them, choose an input link and click the up arrow button or the down arrow button.

For the Funnel stage, the NLS Locale tab appears if you have NLS enabled on your system.

If you are using the Sort Funnel funnel type, it lets you view the current default collate convention, and select a different one for this stage if required (for other funnel types, it is blank). 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 Funnel stage uses this when it is determining the sort order for sort funnel. 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 incoming data sets. Choose an input link from the **Input name** drop down list to specify which link you want to work on.

The Input page allows you to specify details about the incoming data sets. Choose an input link from the **Input name** drop down list to specify which link you want to work on.

The General tab allows you to specify an optional description of the input link. The Partitioning tab allows you to specify how incoming data is partitioned before being funneled. 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 Funnel 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.

The Partitioning tab allows you to specify details about how the data on each of the incoming links is partitioned or collected before it is funneled.

It also allows you to specify that the data should be sorted before being operated on.

By default the stage partitions in Auto mode. This 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.

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

If you are using the Sort Funnel method, and haven't partitioned the data in a previous stage, you should key partition it by choosing the Hash or modulus partition method on this tab.

If the Funnel 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 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 partitioning method for the Funnel 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 Funnel stages. 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 being funneled. 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.

If you are using the Sort Funnel method, and haven't sorted the data in a previous stage, you should sort it here using the same keys that the data is hash partitioned on and funneled on. 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.

In the Output page, you can specify details about data output from the Filter stage. You can choose to work on one of the output links or the reject link. The tabs in this stage allow you specify various options for the link.

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

For the Funnel stage, the Mapping tab allows you to specify how the output columns are derived, that is, what input columns map onto them or how they are generated.

The left pane shows the input columns. These are read only and cannot be modified on this tab. It is a requirement of the Funnel stage that all input links have identical meta data, so only one set of column definitions is shown.

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