The Lookup stage is a processing stage that is used to perform lookup operations on a data set read into memory from any other Parallel job stage that can output data. The most common use for a lookup is to map short codes in the input data set onto expanded information from a lookup table which is then joined to the incoming data and output.

The Lookup stage is a processing stage. It is used to perform lookup operations on a data set read into memory from any other Parallel job stage that can output data. It can also perform lookups directly in a Db2® or Oracle database (see Connectivity Guides for these two databases) or in a lookup table contained in a Lookup File Set stage (see [Lookup file set stage](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/c_deeref_Lookup_File_Set_Stage.html?view=kc))

The most common use for a lookup is to map short codes in the input data set onto expanded information from a lookup table which is then joined to the incoming data and output. For example, you could have an input data set carrying names and addresses of your U.S. customers. The data as presented identifies state as a two letter U. S. state postal code, but you want the data to carry the full name of the state. You could define a lookup table that carries a list of codes matched to states, defining the code as the key column. As the Lookup stage reads each line, it uses the key to look up the state in the lookup table. It adds the state to a new column defined for the output link, and so the full state name is added to each address. If any state codes have been incorrectly entered in the data set, the code will not be found in the lookup table, and so that record will be rejected.

Lookups can also be used for validation of a row. If there is no corresponding entry in a lookup table to the key's values, the row is rejected.

The Lookup stage is one of three stages that join tables based on the values of key columns. The other two are:

* Join stage - [Join stage](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/c_deeref_Join_Stage.html?view=kc)
* Merge stage - [Merge Stage](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/c_deeref_Merge_Stage.html?view=kc)

The three stages differ mainly in the memory they use, the treatment of rows with unmatched keys, and their requirements for data being input (for example, whether it is sorted). See ["Lookup Versus Join"](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/c_deeref_Lookup_Versus_Join.html?view=kc) for help in deciding which stage to use.

The Lookup stage can have a reference link, a single input link, a single output link, and a single rejects link. Depending upon the type and setting of the stage(s) providing the look up information, it can have multiple reference links (where it is directly looking up a Db2 table or Oracle table, it can only have a single reference link). A lot of the setting up of a lookup operation takes place on the stage providing the lookup table.

The input link carries the data from the source data set and is known as the primary link. The following pictures show some example jobs performing lookups.

For each record of the source data set from the primary link, the Lookup stage performs a table lookup on each of the lookup tables attached by reference links. The table lookup is based on the values of a set of lookup key columns, one set for each table. The keys are defined on the Lookup stage. For lookups of data accessed through the Lookup File Set stage, the keys are specified when you create the look up file set.

You can specify a condition on each of the reference links, such that the stage will only perform a lookup on that reference link if the condition is satisfied.

Lookup stages do not require data on the input link or reference links to be sorted. Be aware, though, that large in-memory lookup tables will degrade performance because of their paging requirements.

Each record of the output data set contains columns from a source record plus columns from all the corresponding lookup records where corresponding source and lookup records have the same value for the lookup key columns. The lookup key columns do not have to have the same names in the primary and the reference links.

The optional reject link carries source records that do not have a corresponding entry in the input lookup tables.

You can also perform a range lookup, which compares the value of a source column to a range of values between two lookup table columns. If the source column value falls within the required range, a row is passed to the output link. Alternatively, you can compare the value of a lookup column to a range of values between two source columns. Range lookups must be based on column values, not constant values. Multiple ranges are supported.

There are some special partitioning considerations for Lookup stages. You need to ensure that the data being looked up in the lookup table is in the same partition as the input data referencing it. One way of doing this is to partition the lookup tables using the Entire method. Another way is to partition it in the same way as the input data (although this implies sorting of the data).

Unlike most of the other stages in a Parallel job, the Lookup stage has its own user interface. It does not use the generic interface as described in [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).

When you edit a Lookup stage, the Lookup Editor appears. The left pane represents input data and lookup data, and the right pane represents output data.

InfoSphere® DataStage® does not know how large your data is, so cannot make an informed choice whether to combine data using a Join stage or a Lookup stage. Here's how to decide which to use:

There are two data sets being combined. One is the primary or driving data set, sometimes called the left of the join. The other data set(s) are the reference data sets, or the right of the join.

In all cases the size of the reference data sets is a concern. If these take up a large amount of memory relative to the physical RAM memory size of the computer you are running on, then a Lookup stage might thrash because the reference data sets might not fit in RAM along with everything else that has to be in RAM. This results in very slow performance since each lookup operation can, and typically does, cause a page fault and an I/O operation.

So, if the reference data sets are big enough to cause trouble, use a join. A join does a high-speed sort on the driving and reference data sets. This can involve I/O if the data is big enough, but the I/O is all highly optimized and sequential. After the sort is over, the join processing is very fast and never involves paging or other I/O.

This example shows what happens when data is looked up in a lookup table.

The stage in this case will look up the interest rate for each customer based on the account type. Here is the data that arrives on the primary link:

| **Customer** | **accountNo** | **accountType** | **balance** |
| --- | --- | --- | --- |
| Latimer | 7125678 | plat | 7890.76 |
| Ridley | 7238892 | flexi | 234.88 |
| Cranmer | 7611236 | gold | 1288.00 |
| Hooper | 7176672 | flexi | 3456.99 |
| Moore | 7146789 | gold | 424.76 |
| *Table 1. Input data* | | | |

Here is the data in the lookup table:

**accountType**

InterestRate

**bronze**

1.25

**silver**

1.50

**gold**

1.75

**plat**

2.00

**flexi**

1.88

**fixterm**

3.00

Here is what the lookup stage will output:

| **Customer** | **accountNo** | **accountType** | **balance** | **InterestRate** |
| --- | --- | --- | --- | --- |
| Latimer | 7125678 | plat | 7890.76 | 2.00 |
| Ridley | 7238892 | flexi | 234.88 | 1.88 |
| Cranmer | 7611236 | gold | 1288.00 | 1.75 |
| Hooper | 7176672 | flexi | 3456.99 | 1.88 |
| Moore | 7146789 | gold | 424.76 | 1.75 |
| *Table 2. Stage output* | | | | |

Here is a job that performs this simple lookup:

The accounts data set holds the details of customers and their account types, the interest rates are held in an Oracle table. The lookup stage is set as follows:

All the columns in the accounts data set are mapped over to the output link. The AccountType column in the accounts data set has been joined to the AccountType column of the interest\_rates table. For each row, the AccountType is looked up in the interest\_rates table and the corresponding interest rate is returned.

The reference link has a condition on it. This detects if the balance is null in any of the rows of the accounts data set. If the balance is null the row is sent to the rejects link (the rejects link does not appear in the lookup editor because there is nothing you can change).

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

## About this task

InfoSphere® DataStage® has many defaults which means that lookups can be simple to set up. 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.

The exact steps you need to take when setting up a Lookup stage depend on what type of lookup table you are looking up.

* [**Using In-Memory Lookup tables**](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/t_deeref_Using_InMemory_Lookup_tables.html?view=kc)  
  If you are using a lookup table read into memory from some other stage, you need to specify the details required in the Data Input source stage, the stage providing the lookup table, and the Lookup stage.
* [**Using Oracle or Db2 Databases Directly**](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/t_deeref_Using_Oracle_or_DB2_Databases_Directly.html?view=kc)  
  If you are using Oracle or Db2® Databases Directly, you need to specify the details required in the Data Input source stage, and the Oracle or Db2.UDB Enterprise Stage.
* [**Using Lookup File Set**](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/t_deeref_Using_Lookup_Fileset.html?view=kc)  
  If you are using a lookup table in a lookup file set, you need to specify the details required in the Data Input source stage, and in the Lookup File Set stage.

If you are using a lookup table read into memory from some other stage, you need to specify the details required in the Data Input source stage, the stage providing the lookup table, and the Lookup stage.

**About this task**

If you are accessing a lookup table read into memory from some other stage, you need to do the following:

In the Data Input source stage:

* Specify details about the data source (for example, if using a File Set stage, give the name of the File Set).
* Ensure required column meta data has been specified.
* Fulfil any "must do's" for that particular stage editor.

In the stage providing the lookup table:

* Ensure required column meta data has been specified.
* Fulfil any "must do's" for that particular stage editor.

In the Lookup stage:

* Map the required columns from your data input link to the output link (you can drag them or copy and paste them).
* Map the required columns from your lookup table or tables to the output link (again you can drag them or copy and paste them).
* Specify the key column or columns which are used for the lookup. Do this by dragging - or copying and pasting - key columns from the data link to the **Key Expression** field in the lookup table link. Note that key expressions can only be specified on key fields (that is, columns that have the key field selected in the column definitions). If you drag a column that is not currently defined as a key, you are asked if you want to make it one. If you want the comparison performed on this column to ignore case, then select the **Caseless** check box.

If you want to impose conditions on your lookup, or want to use a reject link, you need to double-click on the Condition header of a reference link, choose **Conditions** from the link shortcut menu, or click the **Condition** toolbar button. The **Lookup Stage Conditions** dialog box appears. This allows you to:

* Specify that one of the reference links is allowed to return multiple rows when performing a lookup without causing an error (choose the relevant reference link from the **Multiple rows returned from link** list).
* Specify a condition for the required references. Double-click the Condition box (or press CTRL-E) to open the expression editor. This expression can access all the columns of the primary link, plus columns in reference links that are processed before this link.
* Specify what happens if the condition is not met on each link.
* Specify what happens if a lookup fails on each link.

If you want to specify a range lookup, select the **Range** check box on the stream link or select **Range** from the **Key Type** list on the reference link. Then double-click the **Key Expression** field to open the Range dialog box, where you can build the range expression.

Next you need to open the **Stage Properties** dialog box for the Lookup stage. Do this by choosing the Stage Properties icon from the stage editor toolbar, or by choosing **Stage Properties** or **Link Properties** from the stage editor shortcut menu (choosing **Link Properties** will open the dialog box with the link you are looking at selected, otherwise you might need to choose the correct link from the **Input name** or **Output name** list).

* 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_lookup_stage.html?view=kc), check that your links are correctly identified as "primary" and "lookup(s)", and reorder if required (the links will be shown in the new order on the Lookup canvas).
* Unless you have particular partitioning requirements, leave the default auto setting on the **Input Page** [**Partitioning Tab**](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/r_deeref_Partitioning_Tab_lookup_stage.html?view=kc).

If you are using Oracle or Db2® Databases Directly, you need to specify the details required in the Data Input source stage, and the Oracle or Db2.UDB Enterprise Stage.

**About this task**

If you are doing a direct lookup in an Oracle or Db2 database table (known as `sparse' mode), you need to do the following:

In the Data Input source stage:

* Specify details about the data source (for example, if using a File Set stage, give the name of the File Set).
* Ensure required column meta data has been specified (this can be done in another stage).
* Fulfil any "must do's" for that particular stage editor.

In the Oracle or Db2.UDB Enterprise Stage:

* Set the Lookup Type to sparse. If you don't do this the lookup will operate as an in-memory lookup.
* Specify required details for connecting to the database table.
* Ensure required column meta data has been specified (this can be omitted altogether if you are relying on Runtime Column Propagation).

See the Connectivity Guides for these two databases for more details.

In the Lookup stage:

* Map the required columns from your data input link to the output link (you can drag them or copy and paste them).
* Map the required columns from your lookup table or tables to the output link (again you can drag them or copy and paste them).

If you want to impose conditions on your lookup, or want to use a reject link, you need to double-click on the Condition header, choose **Conditions** from the link shortcut menu, or click the **Condition** toolbar icon. The **Lookup Stage Conditions** dialog box appears. This allows you to:

* Specify what happens if a lookup fails on this link.

If you want to specify a range lookup, select the **Range** check box on the stream link or select **Range** from the **Key Type** list on the reference link. Then double-click the **Key Expression** field to open the Range dialog box, where you can build the range expression.

Next you need to open the **Stage Properties** dialog box for the Lookup stage. Do this by choosing the Stage Properties icon from the stage editor toolbar, or by choosing **Stage Properties** or **Link Properties** from the stage editor shortcut menu (choosing **Link Properties** will open the dialog box with the link you are looking at selected, otherwise you might need to choose the correct link from the **Input name** or **Output name** list).

* 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_lookup_stage.html?view=kc), check that your links are correctly identified as "primary" and "lookup(s)", and reorder if required.
* Unless you have particular partitioning requirements, leave the default auto setting on the **Input Page** [**Partitioning Tab**](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/r_deeref_Partitioning_Tab_lookup_stage.html?view=kc).

If you are using a lookup table in a lookup file set, you need to specify the details required in the Data Input source stage, and in the Lookup File Set stage.

## About this task

If you are accessing a lookup table held in a lookup file set that you have previously created using InfoSphere® DataStage®, you need to do the following:

In the Data Input source stage:

* Specify details about the data source (for example, if using a File Set stage, give the name of the file set).
* Ensure required column meta data has been specified.
* Fulfil any "must do's" for that particular stage editor.

In the Lookup File Set stage:

* Specify the name of the file set holding the lookup table.
* If you are performing a range lookup, specify the upper and lower bound columns (or the bounded column if the lookup is reversed).
* Make sure that the key column or columns were specified when the file set holding the lookup table was created.
* Ensure required column meta data has been specified.

See [Lookup file set stage](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/c_deeref_Lookup_File_Set_Stage.html?view=kc) for details about the Lookup File Set stage.

In the Lookup stage:

* Map the required columns from your data input link to the output link (you can drag them or copy and paste them).
* Map the required columns from your lookup table or tables to the output link (again you can drag them or copy and paste them).

As you are using a lookup file set this is all the mapping you need to do, the key column or columns for the lookup is defined when you create the lookup file set.

If you are performing a range lookup, select the **Range** check box on the stream link or select **Range** from the **Key Type** list on the reference link. Then double-click the **Key Expression** field to open the Range dialog box, where you can build the range expression.

Next you need to open the **Stage Properties** dialog box for the Lookup stage. Do this by choosing the Stage Properties icon from the stage editor toolbar, or by choosing **Stage Properties** or **Link Properties** from the stage editor shortcut menu (choosing **Link Properties** will open the dialog with the link you are looking at selected, otherwise you might need to choose the correct link from the **Input name** or **Output name** list).

* 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_lookup_stage.html?view=kc), check that your links are correctly identified as "primary" and "lookup(s)", and reorder if required.
* Unless you have particular partitioning requirements, leave the default auto setting on the **Input Page** [**Partitioning Tab**](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/r_deeref_Partitioning_Tab_lookup_stage.html?view=kc).

The Lookup stage toolbar contains the listed buttons. Some of them include Stage properties, Constraints, Show all, Cut, Copy, and Load column definition.

The Lookup toolbar contains the following buttons (from left to right):

* Stage properties
* Constraints
* Show all
* Show/hide stage variables
* Cut
* Copy
* Paste
* Find/replace
* Load column definition
* Save column definition
* Column auto-match
* Input link execution order
* Output link execution order

A link area is a top area that displays links to and from the Lookup stage, showing their columns and the relationships between them.

The link area is divided into two panes; you can drag the splitter bar between them to resize the panes relative to one another. There is also a horizontal scroll bar, allowing you to scroll the view left or right.

The left pane shows the input link, the right pane shows output links. Output columns that have an invalid derivation defined are shown in red. Reference link input key columns with invalid key expressions are also shown in red.

Within the Lookup Editor, a single link can be selected at any one time. When selected, the link's title bar is highlighted, and arrowheads indicate any selected columns within that link.

For the Lookup stage, Metadata area is a bottom area that is divided into two panes: left and right. The left pane shows the input link metadata and the right pane shows the output link metadata.

The meta data for each link is shown in a grid contained within a tabbed page. Click the tab to bring the required link to the front. That link is also selected in the link area.

If you select a link in the link area, its meta data tab is brought to the front automatically.

You can edit the grids to change the column meta data on any of the links. You can also add and delete meta data.

As with column meta data grids on other stage editors, edit row in the context menu allows editing of the full meta data definitions (see ["Columns Tab"](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/r_deeref_Columns_Tab_stage_editors_inputs.html?view=kc)).

The Lookup Editor shortcut menus are displayed by right-clicking the links in the links area.

There are slightly different menus, depending on whether you right-click an input link, or an output link. The input link menu offers you operations on input columns, the output link menu offers you operations on output columns and their derivations.

The shortcut menu enables you to:

* Open the Stage Properties dialog box in order to specify stage or link properties.
* Open the Lookup Stage Conditions dialog box to specify a conditional lookup.
* Open the **Column** **Auto Match** dialog box.
* Display the **Find/Replace** dialog box.
* Display the Select dialog box.
* Validate, or clear a derivation.
* Append a new column to the selected link.
* Select all columns on a link.
* Insert or delete columns.
* Cut, copy, and paste a column or a key expression or a derivation.

If you display the menu from the links area background, you can:

* Open the Stage Properties dialog box in order to specify stage or link properties.
* Open the Lookup Stage Conditions dialog box to specify a conditional lookup.
* Open the Link Execution Order dialog box in order to specify the order in which links should be processed.
* Toggle between viewing link relations for all links, or for the selected link only.

Right-clicking in the meta data area of the Lookup Editor opens the standard grid editing shortcut menus.

Using the Lookup Editor, you can perform various operations on a Lookup stage.

The Lookup Editor enables you to perform the following operations on a Lookup stage:

* Create new columns on a link
* Delete columns from within a link
* Move columns within a link
* Edit column meta data
* Specify key expressions
* Map input columns to output columns

Many of the Lookup stage edits can be made simpler by using the Lookup Editor's drag-and-drop functionality.

**About this task**

You can drag columns from any link to any other link. Common uses are:

* Copying input columns to output links
* Moving columns within a link
* Setting derivation or key expressions

**Procedure**

1. Click the source cell to select it.
2. Click the selected cell again and, without releasing the mouse button, drag the mouse pointer to the desired location within the target link. An insert point appears on the target link to indicate where the new cell will go. This can be to create a new column, or set a derivation. The exact action depends on where you drop.
3. Release the mouse button to drop the selected cell.

**Results**

You can drag multiple columns, key expressions, or derivations. Use the standard Explorer keys when selecting the source column cells, then proceed as for a single cell.

You can drag and drop the full column set by dragging the link title.

In the Lookup stage, you can use the Find and Replace column facilities to help locate a particular column or expression and change it.

**About this task**

If you are working on a complex job where several links, each containing several columns, go in and out of the Lookup stage, you can use the find/replace column facility to help locate a particular column or expression and change it.

The find/replace facility enables you to:

* Find and replace a column name
* Find and replace expression text
* Find the next empty expression
* Find the next expression that contains an error

To use the find/replace facilities, do one of the following:

* Click the **Find/Replace** button on the toolbar
* Choose **Find/Replace** from the link shortcut menu
* Press **Ctrl-F**

The Find and Replace dialog box appears. It has three tabs:

* **Expression Text**. Allows you to locate the occurrence of a particular string within an expression, and replace it if required. You can search up or down, and choose to match case, match whole words, or neither. You can also choose to replace all occurrences of the string within an expression.
* **Column Names**. Allows you to find a particular column and rename it if required. You can search up or down, and choose to match case, match the whole word, or neither.
* **Expression Types**. Allows you to find the next empty expression or the next expression that contains an error. You can also press **Ctrl-M** to find the next empty expression or **Ctrl-N** to find the next erroneous expression.

**Note**The find and replace results are shown in the color specified in **Tools** > **Options**.

Press **F3** to repeat the last search you made without opening the **Find and Replace** dialog box.

If you are working on a complex job where several links, each containing several columns, go in and out of the Lookup stage, you can use the select column facility to select multiple columns.

**About this task**

The select facility enables you to:

* Select all columns whose expressions contains text that matches the text specified.
* Select all columns whose name contains the text specified (and, optionally, matches a specified type).
* Select all columns with a certain data type.
* Select all columns with missing or invalid expressions.

To use the select facilities, choose **Select** from the link shortcut menu. The Select dialog box appears. It has three tabs:

* **Expression Text**. The **Expression Text** tab allows you to select all columns/stage variables whose expressions contain text that matches the text specified. The text specified is a simple text match, taking into account the **Match case** setting.
* **Column Names**. The **Column Names** tab allows you to select all column/stage variables whose Name contains the text specified. There is an additional **Data Type** list, that will limit the columns selected to those with that data type. You can use the **Data Type** list on its own to select all columns of a certain data type. For example, all string columns can be selected by leaving the text field blank, and selecting String as the data type. The data types in the list are generic data types, where each of the column SQL data types belong to one of these generic types.
* **Expression Types**. The **Expression Types** tab allows you to select all columns with either empty expressions or invalid expressions.

States how you can create and delete columns using various methods.

**About this task**

You can create columns on links to the Lookup stage using any of the following methods:

* Select the link, then click the **Load Column Definition** button in the toolbar to open the standard load columns dialog box.
* Use drag-and-drop or copy and paste functionality to create a new column by copying from an existing column on another link.
* Use the shortcut menus to create a new column definition.
* Edit the grids in the link's meta data tab to insert a new column.

When copying columns, a new column is created with the same meta data as the column it was copied from.

To delete a column from within the Lookup Editor, select the column you want to delete and click the **cut** button or choose **Delete** **Column** from the shortcut menu.

In the Lookup stage, you can move the columns within a link using drag-and-drop or cut and paste functions.

## About this task

Use drag-and drop or cut and paste functions to move the columns within a link. Select the required column, then drag it to its new location, or cut it and paste it in its new location.

You can edit column meta data from within the grid in the bottom of the Lookup Editor.

## About this task

In the Lookup Editor, edit the column meta data from within the grid in the bottom. Select the tab for the link meta data that you want to edit, then use the standard InfoSphere® DataStage® edit grid controls.

The meta data shown does not include column derivations since these are edited in the links area.

You can define the derivation of output columns from within the Lookup Editor in a number of ways.

## About this task

Choose one of the following ways to define the derivation of output columns from within the Lookup Editor:

* To map an input column (from data input or reference input) onto an output column you can use drag-and-drop or copy and paste to copy an input column to an output link. The output columns will have the same names as the input columns from which they were derived.
* If the output column already exists, you can drag or copy an input column to the output column's **Derivation** field. This specifies that the column is directly derived from an input column, with no transformations performed.
* You can use the column auto-match facility to automatically set that output columns are derived from their matching input columns.

If a derivation is displayed in red (or the color defined in **Tools** > **Options**), it means that the Lookup Editor considers it incorrect. To see why it is invalid, choose **Validate Derivation** from the shortcut menu.

After an output link column has a derivation defined that contains any input link columns, then a relationship line is drawn between the input column and the output column. There can be one or more relationship lines between columns. You can choose whether to view the relationships for all links, or just the relationships for the selected links, using the button in the toolbar.

* [**Column Auto-Match Facility**](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/t_deeref_Column_AutoMatch_Facility_lookup_stage.html?view=kc)  
  You can specify to use the column auto-match facility, a feature that automatically sets columns on an output link to be derived from matching columns on an input link.

You can specify to use the column auto-match facility, a feature that automatically sets columns on an output link to be derived from matching columns on an input link.

**About this task**

This time-saving feature allows you to automatically set columns on an output link to be derived from matching columns on an input link. Using this feature you can fill in all the output link derivations to route data from corresponding input columns, then go back and edit individual output link columns where you want a different derivation.

**Procedure**

1. Do one of the following:
   * Click the **Auto-match** button in the Lookup Editor toolbar.
   * Choose **Auto-match** from the input link header or output link header shortcut menu.

TheColumn Auto-Match dialog box appears.

Choose the output link that you want to match columns with the input link from the list.

1. Click **Location match** or **Name match** from the **Match type** area.

If you choose **Location match**, this will set output column derivations to the input link columns in the equivalent positions. It starts with the first input link column going to the first output link column, and works its way down until there are no more input columns left.

1. Click **OK** to proceed with the auto-matching.

**Note**Auto-matching does not take into account any data type incompatibility between matched columns; the derivations are set regardless.

You can define key expressions for key fields of reference inputs.

## About this task

Define key expressions for key fields of reference inputs. This is similar to defining derivations for output columns.

The key expression is an equijoin from a primary input link column. You can specify it in two ways:

* Use drag-and-drop to drag a primary input link column to the appropriate key expression cell.
* Use copy and paste to copy a primary input link column and paste it on the appropriate key expression cell.

A relationship link is drawn between the primary input link column and the key expression.

You can also use drag-and-drop or copy and paste to copy an existing key expression to another input column, and you can drag or copy multiple selections.

If a key expression is displayed in red (or the color defined in **Tools** > **Options**), it means that the Transformer Editor considers it incorrect. To see why it is invalid, choose **Validate Derivation** from the shortcut menu.

In the Properties dialog of the Lookup stage, you can specify details about how the stage operates.

The Lookup stage has a Properties dialog box which allows you to specify details about how the stage operates.

The Lookup Stage Properties dialog box 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_lookup_stage.html?view=kc). This 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_lookup_stage.html?view=kc). This is where you specify details about the data input to the Lookup stage.
* [**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_lookup_stage.html?view=kc). This is where you specify details about the output links from the Lookup stage.

You can specify aspects of the Lookup stage from the Lookup stage: Stage Page.

The **General** tab allows you to specify an optional description of the stage. 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. The **Build** tab allows you to override the default compiler and linker flags for this particular stage.

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 on the stream link. You can explicitly select **Set** or **Clear**. Select **Set** to request the next stage in the job 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 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 which input link is the primary link and the order in which the reference links are processed.

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.

You can also access this tab by clicking the input link order button in the toolbar, or by choosing **Reorder input links** from the shortcut menu.

For the Lookup 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 Lookup stage uses this when it is determining the order of the key fields. 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.

In some cases the Lookup stage might use C++ code to implement your lookup.

In this case, you can use the **Build** tab to override the compiler and linker flags that have been set for the job or project. The flags you specify here will take effect for this stage and this stage alone. The flags available are platform and compiler-dependent.

The **Input page** allows you to specify details about the incoming data set and the reference links. Choose a link from the **Input name** list to specify which link you want to work on.

The **Input page** allows you to specify details about the incoming data set and the reference links. Choose a link from the **Input name** list to specify which link you want to work on.

The **General** tab allows you to specify an optional description of the link. When you are performing an in-memory lookup, the **General** tab has two additional fields:

* **Save to lookup fileset**. Allows you to specify a lookup file set to save the look up data.
* **Diskpool**. Specify the name of the disk pool into which to write the file set. You can also specify a job parameter.

The **Partitioning** tab allows you to specify how incoming data on the source data set link is partitioned. The **Advanced** tab allows you to change the default buffering settings for the input link.

Details about Lookup 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 incoming data is partitioned or collected before the lookup is performed.

It also allows you to specify that the data should be sorted before the lookup. Note that you cannot specify partitioning or sorting on the reference links, this is specified in their source stage.

By default the stage uses the auto partitioning method. If the Preserve Partitioning option has been set on the previous stage in the job the stage will warn you when the job runs if it cannot preserve the partitioning of the incoming data.

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

You might need to ensure that your lookup tables have been partitioned using the Entire method, so that the lookup tables will always contain the full set of data that might need to be looked up. For lookup files and lookup tables being looked up in databases, the partitioning is performed on those stages.

If the Lookup 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** 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 Lookup 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 Lookup 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 lookup 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 Lookup stage.

The Lookup stage can have only one output link. It can also have a single reject link, where records can be sent if the lookup fails. The **Output Link** list allows you to choose whether you are looking at details of the main output link (the stream link) or the reject link.

The **General** tab allows you to specify an optional description of the output link. The **Advanced** tab allows you to change the default buffering settings for the output links.

Use the Reject links to specify how the Lookup stage operates.

You cannot set the mapping or edit the column definitions for a reject link. The link uses the column definitions for the primary input link.

You can specify conditions for lookup and actions to be taken depending on the lookup results from the **Lookup Stage Conditions** dialog box.

The Lookup stage has a **Lookup Stage Conditions** dialog box that allows you to specify:

* Which reference link (if any) can return multiple rows from a lookup.
* A condition that should be fulfilled before a lookup is performed on a reference link.
* What action should be taken if a condition on a reference link is not met.
* What action should be taken if a lookup on a link fails.

You can open the **Lookup Stage Conditions** dialog box by:

* Double-clicking the Condition: bar on a reference link.
* Selecting **Conditions** from the background shortcut menu.
* Clicking the **Conditions** toolbar button.
* Selecting **Conditions** from the link shortcut menu.

To specify that a link can legitimately return multiple rows:

* Select the link name from the **Multiple rows returned from link** list (note that only one reference link in a Lookup stage is allowed to return multiple rows, and that this feature is only available for in-memory lookups).

To specify a condition for a reference link:

* Double-click the **Condition** field for the link you want to specify a condition for. The field expands to let you type in a condition, or click the browse button to open the expression editor to get help in specifying an expression. The condition should return a TRUE/FALSE result (for example DSLINK1.COL1 > 0).

To specify the action taken if the specified condition is not met:

* Choose an action from the **Condition Not Met** list. Possible actions are:
  + **Continue**. The fields from that link are set to NULL if the field is nullable, or to a default value if not. Continues processing any further lookups before sending the row to the output link.
  + **Drop**. Drops the row and continues with the next lookup.
  + **Fail**. Causes the job to issue a fatal error and stop.
  + **Reject**. Sends the row to the reject link.

To specify the action taken if a lookup on a link fails:

* Choose an action from the **Lookup Failure** list. Possible actions are:
  + **Continue**. The fields from that link are set to NULL if the field is nullable, or to a default value if not. Continues processing any further lookups before sending the row to the output link.
  + **Drop**. Drops the row and continues with the next lookup.
  + **Fail**. Causes the job to issue a fatal error and stop.
  + **Reject**. Sends the row to the reject link.

You can define a range lookup.

## About this task

You can define a range lookup on the stream link or a reference link of a Lookup stage. On the stream link, the lookup compares the value of a source column to a range of values between two lookup columns. On the reference link, the lookup compares the value of a lookup column to a range of values between two source columns. Multiple ranges are supported.

## Procedure

1. Select the column for the lookup:
   * To define the lookup on the stream link, select the **Range** check box next to the source column in the links area.
   * To define the lookup on a reference link, select the **Key** check box next to the reference column in the meta data area. In the links area, select **Range** from the **Key Type** list. The data on the reference link must be sorted. (**Tip:** See if the APT\_NO\_SORT\_INSERTION environment variable is used and set to true for the reference link. If the APT\_NO\_SORT\_INSERTION environment variable is set to true, then you must specify a Sort stage on the reference link to ensure that the data is sorted.)
2. Double-click the **Key Expression** field next to the selected column to open the Range dialog box.
3. Select a link from the **Lookup Link** list. (If you are defining the lookup on a reference link, the stream link appears by default.)
4. Define the range expression by selecting the upper bound and lower bound range columns and the required operators. For example:

Account\_Detail.Trans\_Date >= Customer\_Detail.Start\_Date AND[Copy](javascript:void(0);)

Account\_Detail.Trans\_Date <= Customer\_Detail.End\_Date[Copy](javascript:void(0);)

As you build the expression, it appears in the **Expression** box.

1. Select **Caseless** if you want the lookup to ignore case.
2. Click **OK**.

The InfoSphere® DataStage® Expression Editor helps you to enter correct expressions when you edit Lookup stages.

The Expression Editor can:

* Facilitate the entry of expression elements
* Complete the names of frequently used variables
* Validate the expression

The Expression Editor can be opened from:

* **Lookup Stage Conditions** dialog box

The format of an expression is given.

The format of an expression is as follows:

KEY:

something\_like\_this is a token

something\_in\_italics is a terminal, that is, does not break down any

further

| is a choice between tokens

[ is an optional part of the construction

"XXX" is a literal token (that is, use XXX not

including the quotes) [Copy](javascript:void(0);)

=================================================

expression ::= function\_call |

variable\_name |

other\_name |

constant |

unary\_expression |

binary\_expression |

if\_then\_else\_expression |

substring\_expression |

"(" expression ")"

function\_call ::= function\_name "(" [argument\_list] ")"

argument\_list ::= expression | expression "," argument\_list

function\_name ::= name of a built-in function |

name of a user-defined\_function

variable\_name ::= job\_parameter name

other\_name ::= name of a built-in macro, system variable, and so on.

constant ::= numeric\_constant | string\_constant

numeric\_constant ::= ["+" | "-"] digits ["." [digits]] ["E" | "e" ["+" | "-"] digits]

string\_constant ::= "'" [characters] "'" |

""" [characters] """ |

"\" [characters] "\"

unary\_expression ::= unary\_operator expression

unary\_operator ::= "+" | "-"

binary\_expression ::= expression binary\_operator expression

binary\_operator ::= arithmetic\_operator |

concatenation\_operator |

matches\_operator |

relational\_operator |

logical\_operator

arithmetic\_operator ::= "+" | "-" | "\*" | "/" | "^"

concatenation\_operator ::= ":"

relational\_operator ::= " =" |"EQ" |

"<>" | "#" | "NE" |

">" | "GT" |

">=" | "=>" | "GE" |

"<" | "LT" |

"<=" | "=<" | "LE"

logical\_operator ::= "AND" | "OR"

if\_then\_else\_expression ::= "IF" expression "THEN" expression "ELSE" expression

substring\_expression ::= expression "[" [expression ["," expression] "]"

field\_expression ::= expression "[" expression ","

expression ","

expression "]"

/\* That is, always 3 args

Note: keywords like "AND" or "IF" or "EQ" might be in any case[Copy](javascript:void(0);)

In the Lookup stage, you can use the Expression Editor to suggest the next element in your expression whenever the insertion point is in an expression box.

## About this task

Whenever the insertion point is in an expression box, you can use the Expression Editor to suggest the next element in your expression. Do this by right-clicking the box, or by clicking the **Suggest** button to the right of the box. This opens the **Suggest Operand** or **Suggest Operator** menu. Which menu appears depends on context, that is, whether you should be entering an operand or an operator as the next expression element. The Functions available from this menu are described in [Parallel transform functions](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/c_deeref_Functions_functions.html?view=kc). The DS macros are described in [Supported macros and system variables](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/limitationsmacros.html?view=kc). You can also specify custom routines for use in the expression editor (see in [Server routines](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.design.doc/topics/c_ddesref_Server_Routines.html?view=kc)).

In the Lookup stage, the Expression Editor stores variable names. You enter a variable name that was used before and type the first few characters. Press F5, the Expression Editor completes the variable name for you.

## About this task

The Expression Editor stores variable names. When you enter a variable name you have used before, you can type the first few characters, then press **F5**. The Expression Editor completes the variable name for you.

If you enter the name of the input link followed by a period, for example, **DailySales.**, the Expression Editor displays a list of the column names of the link. If you continue typing, the list selection changes to match what you type. You can also select a column name using the mouse. Enter a selected column name into the expression by pressing **Tab** or **Enter**. Press **Esc** to dismiss the list without selecting a column name.

Use this function to validate the expression in the Lookup stage. The Expression Editor checks if the syntax of an expression entered in the Transformer Editor is correct and any variable names used are acceptable to the compiler.

## About this task

When you have entered an expression in the Lookup Editor, press **Enter** to validate it. The Expression Editor checks that the syntax is correct and that any variable names used are acceptable to the compiler.

If there is an error, a message appears and the element causing the error is highlighted in the expression box. You can either correct the expression or close the Lookup Editor or Lookup dialog box.

For any expression, selecting Validate from its shortcut menu will also validate it and show any errors in a message box.

States the different ways by which you can exit the Expression Editor in the Lookup stage.

## About this task

You can exit the Expression Editor in the following ways:

* Press **Esc** (which discards changes).
* Press **Return** (which accepts changes).
* Click outside the Expression Editor box (which accepts changes).

In the Lookup stage, the Expression Editor is configured by editing the Designer options.

## About this task

You can resize the Expression Editor window by dragging the corner of the window. The next time you open the Expression Editor in the same context (for example, to edit output columns) on the same client, it will have the same size.

The Expression Editor is configured by editing the Designer options. To specify how `helpful' the expression editor is, see [The InfoSphere DataStage expression editor](https://www.ibm.com/support/knowledgecenter/SSZJPZ_11.7.0/com.ibm.swg.im.iis.ds.parjob.dev.doc/topics/c_deeref_The_DataStage_Expression_Editor_basic_transformer_stage.html?view=kc).