Loop

The component will be used for the loop execution of the node selected by a user. As a rule, the Supernode can be used as such node. The actions performed with data in each iteration are set in it. There are the following methods used to set the loop iterations:

1. **Set number of times** is analogue of "FOR ... TO ...".
2. **Iteration count is set by the loop exit condition**. It is an analogue of "DO ... WHILE ...". The value of the looped output node variable is analysed to ensure its compliance with this condition.
3. **Iteration count is set by the number of groups to which the input data set rows are divided** . It is an analogue of "FOR EACH". In the limiting case, iteration count is equal to row count of the input data set.

It is not obligatory to use the input data set in the first and second cases. But if it is used, it is required to supply all rows of this data set (data sets) to the looped node input for each iteration.

In the third case, the input data set rows are divided according to the given parameters to the groups of rows, and the rows of the next group are supplied to the looped node input for each iteration. If unique identifiers of the input data set rows are specified as the group parameter, such group will contain only one row. In this case, the loop will iterate over the input data rows, and then it will transfer the next row to the looped node input.

Ports

Being originally created, the node does not have ports. Upon configuration of parameters in the wizard, the loop node has a set of ports of the looped node, apart from the output ports for variables.

Wizard

Step 1. Selection of the Loop Node

A list of nodes of the current module is provided at this stage. To loop the node, it is required to mark it with the radio button and go to the next step.

The wizard interface enables to search for the nodes in the general list by means of filters, namely, by name and node comment.

It is not allowed to run the handlers in the loop that have been created on the basis of the following components: Node execution, Reference node, Condition and *Loop*.

Step 2. Configuration of the Loop Type

Source node: the informational field that displays the looped node.

Loop type: determination of the loop operation logics. It is set using the radio button:

- **Given iterations**: this loop type will be executed as many times as it will be set in the "Iteration count" parameter.
- **Postcondition loop**: the iteration count of such loop is controlled according to the condition for consistency with which a variable is checked for each iteration. The loop ends at the iteration that makes the variable value meet such condition. The following settings are set:
  - **Variable**: selection of a variable for the loop exit condition. The selection list contains the variables sent to the output ports of the looped node.
  - **Termination condition**: relation of the variable comparability with the Value.
  - **Value**: the field for entering of the value with which the variable will be compared. It should be noted that the looped node must presuppose the operation with this variable that will enable to meet the loop exit condition, otherwise, the loop will be infinite.
- **Group processing**: this loop divides the source data to the groups according to the selected criterion. Thus, the looped node processes each data group separately. The division criterion is governed by the "Type of group processing" parameter:
  - **Fixed group size**: the source data set will be successively divided to the groups the size of which is controlled according to the "Row count in the group" parameter. If the row count in the group is not in multiples of the row count in the source data set, the latter group will have smaller size.
  - **Fixed group count**: the source data set will be divided to the given count of groups. This count is set by the "Group count" parameter. If the count of groups is not in multiples of the row count in the source data set, the row count will be different in some groups, and they will be uniformly distributed among other groups.
  - **Grouping by the unique fields values**: it is required to select the listed fields of the source data set that enable to set the unique identifier of the row group. The row count will be equal to the count of the unique identifiers.

Parallel processing is used to speed up calculations during the loop operation. Thread count is defined by the "Maximum thread count" parameter. Parallel processing is not supported by the postcondition loop.

> **Important**: In some cases parallel processing is not allowed. For example, if access to the data source that does not support parallel requests is required for each iteration.

Add iteration identifiers: the checkbox enables to add the "Iteration identifier" field to the output table where the number of the iteration at which the row has been created is specified for each row.

Ignore errors: the checkbox disables interruption of the loop execution in the case of errors detection.

Loop variable: the variable to which the current iteration number is assigned during the loop execution. Iterations are numbered from 0. The variable is selected from a list of variables of the input ports of the looped node.

Step 3. Mapping Variables

This step can be available only when looping the *Supernodes* that have input and output ports for variables. During this stage, it is required to configure transfer of values of the output variables to the corresponding input variables at the next loop iteration.

To provide mapping of the input and output variables, it is required to drag the input variable symbol to the output variable symbol. This mapping will be graphically displayed as a link.

These links can be deleted using  button. It is not required to link variables at this stage.