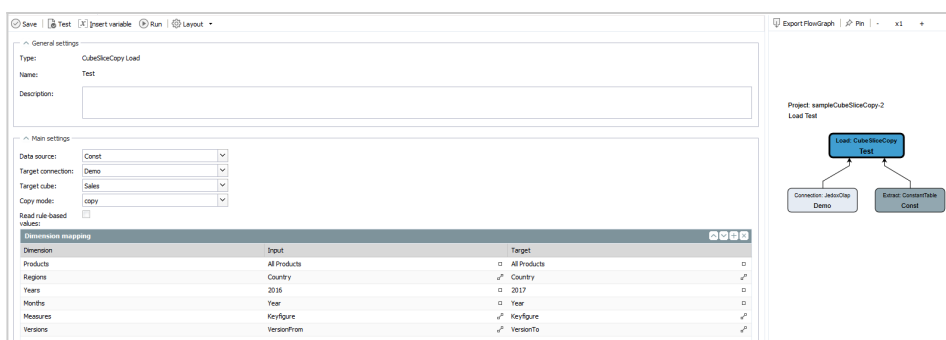


# CubeSliceCopy Load

With this load type, you can copy the values from one cube slice to another in an Integrator job. It can also perform “like” operations. The new load will execute the copy operation in In-Memory DB itself, without first reading values.



## Main Settings

<b>Data Source:</b>	The source is an extract or transform that defines in its rows the data to be written to the cube. The first columns of the source must contain the names of the dimension elements of the cube. The last column contains the value that is written in the cube cell.
<b>Target Connection:</b>	Connection to jedox In-Memory DB
<b>Target Cube:</b>	Cube in the jedox In-Memory DB
<b>Copy mode:</b>	<p><b>copy:</b> copies values from one cube slice to another</p> <p><b>copyLike:</b> copies values from one cube slice, including their distribution on the base level, to another cube slice</p>
<b>Read rule-based values</b>	If set, all rule-based cell values are extracted. If the rule value has the result 0, it is extracted if "empty cells" is set to "excludeEmpty" but not if it's set to "excludeEmptyAndZero".

## Dimension Mapping

In the Dimension Mapping table, any dimension of the cube can be assigned for each source column. For columns that should not be loaded to the cube, the entry "" is used. Thus it is possible to rename and skip columns. The dimension order cannot be changed, which is only relevant if the cube does not yet exist. The last source column is not displayed in the Dimension Mapping, as it always contains the cube cell value.

By default, if no Dimension Mapping is given, the names of the cube dimensions are entirely defined by the names of the source column names. In this case, the number of dimensions must be equal to the number of columns in the source minus 1.

---