

Release notes for version 3.1

Important! Create a backup copy of your projects before updating to the new version.

What's new

New transformations



Running total transformation calculates rank, cumulative count, sum or % from total sum for a given column inside groups, or table-wide.



Iterate another table transformation is similar to “Iterate” transformation, but allows passing a table to an iterated project. The iterated project should use “Input” transformation in order to receive the table. This transformation allows significantly increasing performance in cases when iterated project had to read the same file in each iteration (in previous versions of EasyMorph). Now such file can be read once and then passed to an iterated project in memory.



Generate UIDs transformation assigns a unique identifier for each row in table. It can be used for generating primary keys.



Rename with lookup transformation renames table columns according to a lookup table. It can be used, for instance, to make all column names uppercase, or correct column names if they are not consistent.

Changes to existing transformations

The changes below don't affect calculations in projects created in older versions, although they are not backward compatible.

SQL Command transformations now has 4 commands:

- *Custom SQL command* – sends a custom SQL command to database. This is the same functionality as in previous versions.
- *Create table* – creates a database table with column names as in current EasyMorph table. Data types are specified as Text, Number or Date and are universal for all database types. Under the hood, a type mapping exists that maps each data type to a native data type of a particular database. The type mapping can be adjusted, if necessary.
- *Delete table* – removes a database table.
- *Delete all rows* – purges all data from a database table. Under the hood this command uses TRUNCATE where possible, or emulates its behavior with DELETE FROM.

Export to Tableau transformation now supports Date type. It can be used for columns with dates as numbers in EasyMorph. Text dates should be converted to number dates before exporting to Tableau using *date()* function.

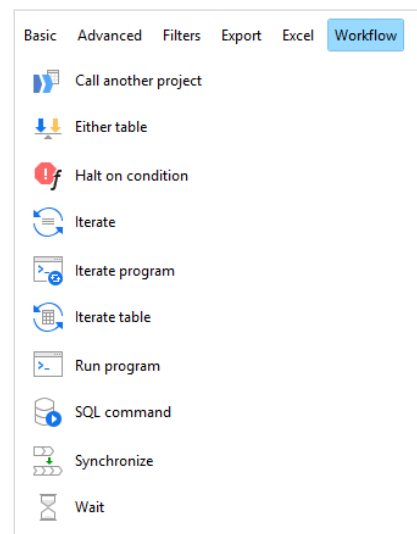
Call another project transformation now allows assigning parameters of the called project with text or number constants. Previously it was possible only using parameters of the calling project.

Run program transformation now can capture STDOUT and STDERR of the executed external application or Windows shell command. For instance it is possible to run *dir* and capture its output as a table in EasyMorph, or forward output of an application into EasyMorph without creating a temporary intermediate CSV file. STDERR is captured in a separate column, which can later be used for error-handling routines. Since EasyMorph also outputs errors into STDERR this feature can be used for error handling of EasyMorph projects executed in the command line mode.

New category for transformations: Workflow

Several transformations have been moved to new category Workflow:

- Call another project
- Either table
- Halt on condition
- Iterate
- Iterate program
- Iterate table
- Run program
- SQL command
- Synchronize
- Wait



New functions

mirror(text) – reverses the order of characters in a text. This function allows using text functions such as *keepbefore()* or *keepafter()* to operate with the last occurrence of a substring in a text. Example:

```
mirror(keepafter(mirror('one-two-three'), '-')) returns 'one-two'.
```

roundxl(number) – rounds a number using Excel's rounding logic. Example:

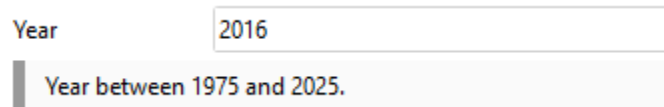
```
roundxl( 5.65, 1 ) returns 5.7.
```

now() – current date and time represented as a number. Example:

```
now() returns 42614.5357184491.
```

Annotations for parameters

Starting from this release project parameters can have annotations.



The image shows a user interface for editing a parameter. On the left, the label 'Year' is displayed. To its right is a text input field containing the value '2016'. Below the input field, there is a light gray box containing the text 'Year between 1975 and 2025.', which represents an annotation for the parameter.

Annotations are created in the Parameter Editor.

Parameter substitution in Run Program

Run Program transformation allows defining its command line as a text or as a calculated expression. Starting from this release it is possible to insert parameters into text, in the same manner as it can already be done for SQL queries and commands. For instance:

```
copy /Y {SourceFileName} {DestinationFileName}
```

The command line above is equivalent to the calculated expression below, but is easier to use.

```
“copy /Y “ & {SourceFileName} & “ “ & {DestinationFileName}
```

SourceFileName and DestinationFileName are parameter names.

Tab copying

Now it is possible to copy/paste entire tab with all tables and transformations. It also works for copying a tab from one project to another project.

Updates to data drivers

- Added support for Amazon Redshift