

Release notes for version 3.7

Important! Create a backup copy of your projects before updating to the new version. Projects saved in the new version can't be opened in earlier versions.

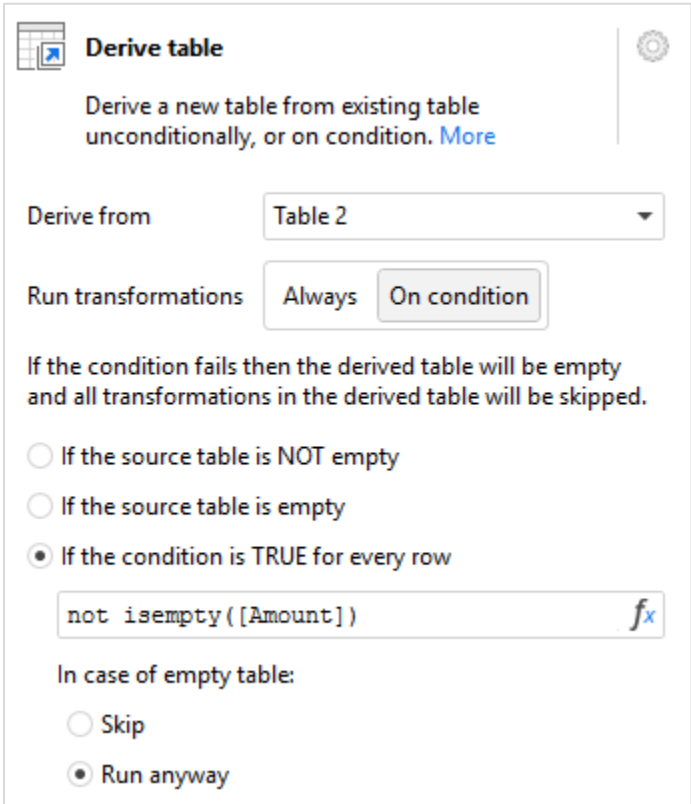
What's new

Conditionally derived tables

Previously, arranging IF...THEN...ELSE workflows required using various, frequently obscure, workarounds. Starting from this release, arranging conditional workflows has become simple and straight forward because derived tables in EasyMorph can now work in two modes:

Unconditional – this is the usual, regular mode known from the very first version of EasyMorph. In this mode a derived table obtains its dataset from the source table, and all transformations in the derived table are executed as usually.

Conditional – this is a new mode in which transformations in a derived table are executed **only** when a certain condition is satisfied. If the condition is not met, then all transformations are quietly skipped and the resulting dataset is always empty.



Screenshot 1: Conditionally derived table.

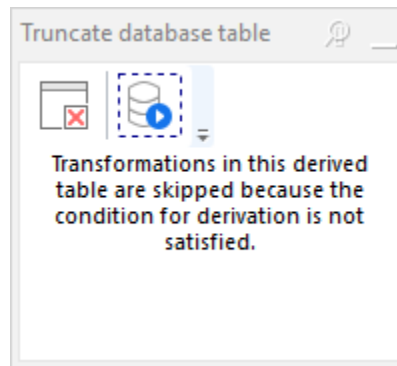
Possible conditions are similar to the ones used in the “Halt on condition” transformation, namely:

- When table is empty
- When table is NOT empty
- When an expression is TRUE for each row in table

To arrange an IF ... THEN workflow create a derived table with condition.

To arrange an IF ... THEN ... ELSE workflow create two derived tables with conditions that are inverse to each other.

The transformation icon indicates when derivation condition is not satisfied and all transformations are skipped.



Screenshot 2: Skipped derived table.

Disable/enable transformations

Now it is possible to temporarily disable a transformation. A disabled transformation becomes “transparent” – it performs no action and simply passes through its input dataset to the next transformation. Disabled transformations are marked with a special overlay icon.



Screenshot 3: Disabled transformation.

To disable (enable) a transformation right-click it and choose Disable (Enable).

New transformations



Replace transformation allows replacing particular values in a column with new ones. It can be used for instance to replace mistyped values with correct ones.



File Command transformation is a convenient way for simple file operations. Currently it has 5 commands:

- Copy/Move File
- Rename/Clone File
- Delete File
- Delete All Files in Folder
- Unzip File



PowerShell transformation executes one or more PowerShell commands. It requires PowerShell version 3 or above installed. A column can be used to provide an input collection of values accessible through `$input` from within the PowerShell script. The output collection of values can be captured back into EasyMorph, as well as errors. This transformation is not available in the free edition.



XSLT transformation modifies XML files using the [Extensible Stylesheet Language Transformations](#) (XSLT) language. It's a powerful language that allows extracting data from XML files, inserting new nodes, amending XML elements and attributes. For instance, file paths in Tableau workbooks and EasyMorph projects can be automatically modified using this transformation.

Changes to existing transformations

The **Export to Tableau** transformation now uses a connector to Tableau Server from a repository (or embedded). Also Tableau projects and data source names can be picked from a list, or defined using a parameter.

The **Iterate** transformation now has a new mode that allows continuing execution even if one or more iterations failed. In this mode, a new column "Iteration status" is created. The column contains errors of iterations, if any. This mode together with conditionally derived tables can be used for arranging failover logic.

The **Export to Excel** transformation now allows using a parameter to specify the target sheet name.

If no working directory defined in the **Run Program** transformation, it's now implicitly assumed that the working directory is the directory of the project.

It is now possible to change the order of rules in the **Rule** transformation.

New functions

ispathrooted(path_text) – detects if a path to file or folder contains a root. Example

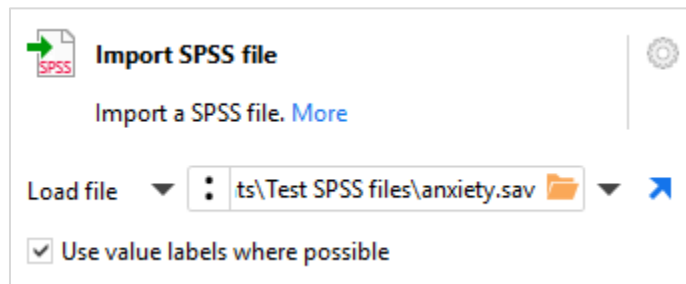
```
ispathrooted(' C:\Documents\myproject.morph') returns TRUE.
```

Non-database connectors

From now on, non-database connectors can be created and stored in a repository, or embedded in a project. In this release new connector types are Tableau Server, EasyMorph Server, and 1010data. Future releases will introduce new connector types for various applications and cloud services.

Support for SPSS/PSPP files

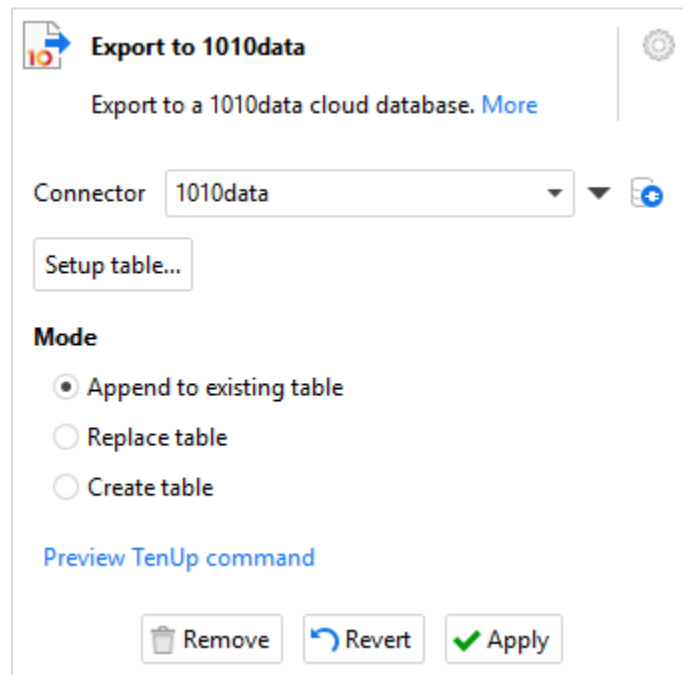
The “Import SPSS file” transformation imports one or multiple .sav files created in SPSS or PSPP. Column values can be automatically replaced with value labels, if necessary.



Screenshot 4: Import SPSS file with value labels.

Export to 1010data (experimental)

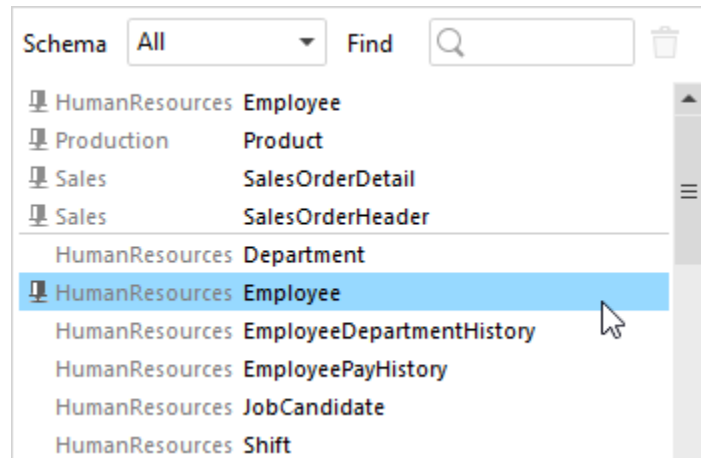
We’re adding the ability to export from EasyMorph to 1010data cloud analytical database using the “Export to 1010data” transformation. The export is done using the TenUp utility which is included into the EasyMorph installation package. The generated command line for TenUp can be previewed as well as the contents of tree file.



Screenshot 5: Export to 1010data.

New database table picker

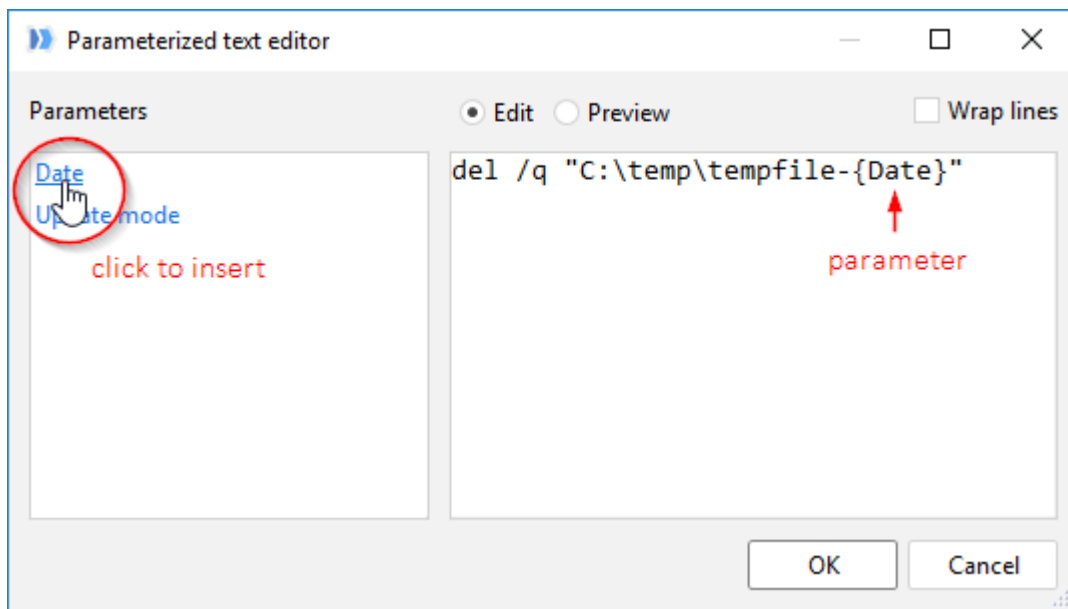
This release introduces a new database table picker. It's used in queries and transformations where a database table should be picked from list. The picker is capable of working smoothly with tens of thousands table names and hundreds of schemas. Frequently used tables can be "pinned" to the top of list for convenience.



Screenshot 6: New database table picker.

Multi-line editor for "Run Program" and "PowerShell"

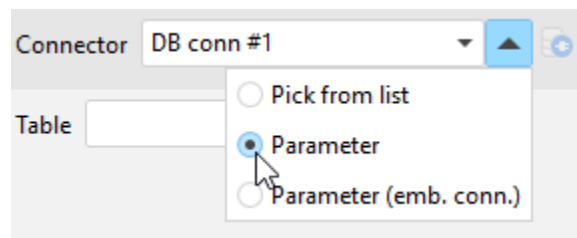
The transformations got a "Pop-up" button that opens a multi-line editor for more convenient command editing. The editor has convenient means for inserting project parameters into text – just click a parameter name on the left and it will be inserted at the cursor position in the text. You can switch to the preview mode to see what the text would look like after parameter substitution.



Screenshot 7: Editor with parameter insertion.

Connectors specified by parameters

Starting from this release it is possible to specify connector using a parameter in all transformations that use a connector, as well as in the Query Editor. A parameter can be used to specify an embedded connector as well.



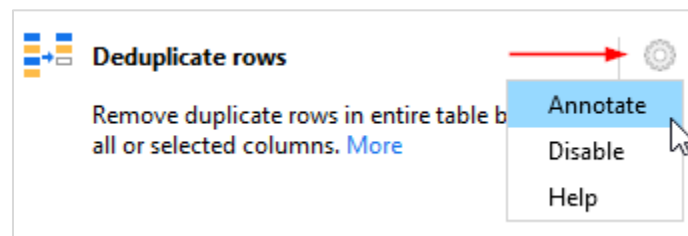
Screenshot 8: Connector specifies by a parameter.

Block indentation in text editor

EasyMorph now uses a new, improved text editor for editing expressions, custom SQL queries and commands, and other multi-line texts in the application. The editor allows performing block indentation (select multiple lines and press Tab or Shift+Tab).

Transformation settings menu

Creating annotations is moved into the transformation settings menu (the gear icon next to transformation name and description). The menu also contains commands for enabling/disabling transformations and a link to the web-help for the transformation.



Screenshot 9: Transformation settings menu.

Miscellaneous

- Drag a folder with multiple files into EasyMorph to load all these files at once.
- If a called project contains only 1 table it's implicitly considered the default result table for Iterate/Call transformations.
- Documentation can now be generated even for projects with transformations with invalid properties. Such transformations will be marked red.