

Release notes for version 3.9.1

What's new

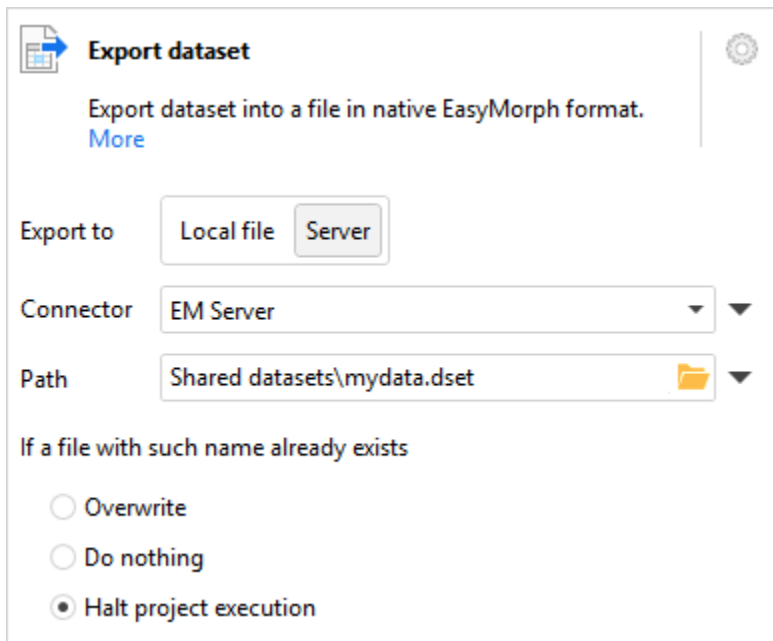
Overview

With this release we're starting a major long-term development effort which goal is to make EasyMorph Server a convenient hub for collaboration on datasets as well as easy-to-use dataset storage that will eventually become a sort of light-weight database for non-technical users.

We've also added the ability to specify native database data types for creating database tables – a long awaited feature for advanced users.

Exporting/importing datasets to/from EasyMorph Server

The “Export dataset” action now allows publishing datasets directly to EasyMorph Server. In a similar fashion, the “Import dataset” action can be used to retrieve a dataset from Server. The dataset format (.dset) is a compact, compressed representation of data in a form native to EasyMorph. Therefore publishing datasets to, and retrieving datasets from EasyMorph Server to a local machine is a very fast operation even when performed over a network. This makes it convenient to store datasets on Server, rather than on local machine. Publishing projects to Server also becomes more convenient, because there is no need to use calculated parameters for file locations. Also, it's easier to share projects with other users when datasets are stored on Server because there is no need to send data files with a project as they are always stored on Server.



Export dataset

Export dataset into a file in native EasyMorph format.
[More](#)

Export to:

Connector:

Path:

If a file with such name already exists

☐ Overwrite

☐ Do nothing

☒ Halt project execution

Screenshot 1: Exporting dataset to Server.

Import dataset

Import dataset from a file in the native EasyMorph format. [More](#)

Import from: One or more files Server

Connector: EM Server

Path: Shared datasets\mydata.dset

Screenshot 2: Importing dataset from Server.

Advanced data types for creating database tables

Until now, a database table could only be created with fields of three generic data types: Number, Text and Date. Under the hood, these data types would be replaced with pre-defined database field types depending on the target database system.

From now on, it's possible to specify a field type using data types native to the target database. Note that this feature is not available in the free edition.

Database command

Send a command to a database (e.g. create a table), or execute an arbitrary SQL statement (e.g. bulk load a file). [More](#)

Connector: sql server

Command: Create table Advanced

Table name: example

If table already exists: Fail

Data types: Simple Advanced

Select all Unselect all

Column name	Data type
<input checked="" type="checkbox"/> rank	float
<input checked="" type="checkbox"/> company	nvarchar(100)
<input checked="" type="checkbox"/> Revenue per worker	float
<input checked="" type="checkbox"/> city	nvarchar(50)
<input checked="" type="checkbox"/> workers	float
<input checked="" type="checkbox"/> revenue	money NOT NULL
<input type="checkbox"/> industry	

Column data type

revenue

☒ Standard data type ☐ Custom definition

Data type: money

- decimal
- float
- hierarchyid
- int
- money
- nchar
- ntext
- numeric
- nvarchar
- real
- smalldatetime
- smallint
- smallmoney
- text
- time
- tinyint
- uniqueidentifier
- varbinary

Screenshot 3: Advanced data types for table creation.

The “Other SQL” dialect

With EasyMorph it’s still possible to connect to an ODBC data source and send custom SQL queries even when the data source’s SQL dialect is not supported by EasyMorph’s visual query builder. To simplify working with such sources, we’ve added a new SQL dialect type for ODBC connectors – “Other SQL”. Visual query builder is not available for ODBC connectors with this dialect. However it’s still possible to write custom SQL. Use the “Other SQL” dialect whenever you need to connect to data sources with ODBC drivers such as Google BigQuery or Teradata.

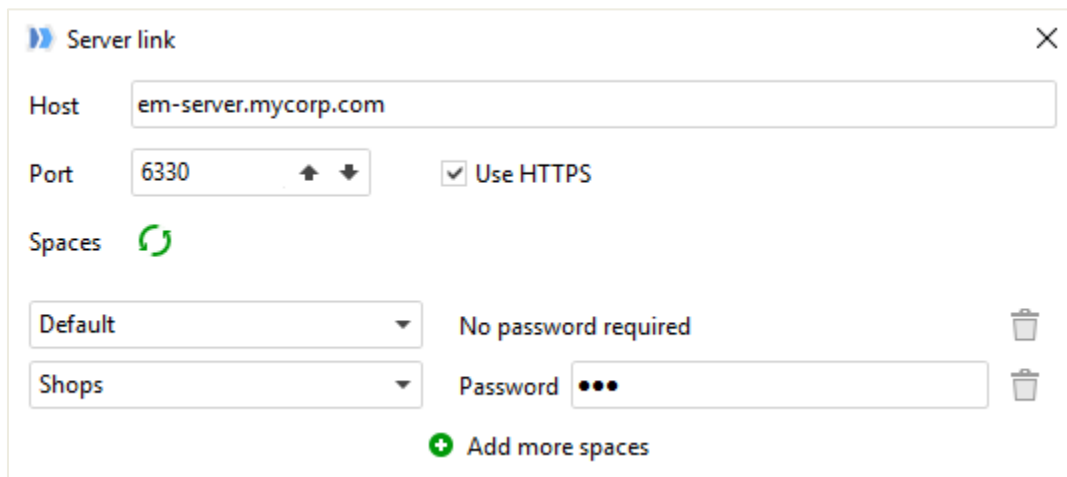
What’s new in EasyMorph Desktop

Server Link

Server Link is the default connection to Server from Desktop. It’s project-independent and can be set up on the Start page of the application. When configured, it can be used for:

- publishing projects to Server (dialog “Publish to Server”)
- opening projects from Server (button “Open from Server” on the Start screen)
- publishing datasets to Server (command “Send to Server” in the action right-click menu)
- loading datasets from Server (command “Receive from Server” in the workspace right-click menu)

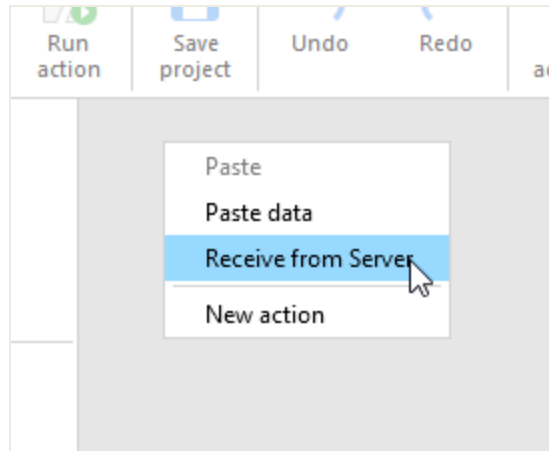
Multiple spaces of a Server can be specified in Server Link configuration.

The screenshot shows a dialog box titled "Server link" with a close button (X) in the top right corner. Inside the dialog, there are several configuration fields: "Host" with a text input containing "em-server.mycorp.com"; "Port" with a numeric input containing "6330" and up/down arrow buttons; a checked checkbox labeled "Use HTTPS"; a "Spaces" section with a green circular refresh icon; two dropdown menus for selecting spaces, with "Default" and "Shops" currently selected; a "No password required" option with a trash icon, and a "Password" field with three dots and a trash icon; and a green plus icon followed by the text "Add more spaces".

Screenshot 4: Server Link configuration.

Server Link makes it easier to use Server for collaboration. Datasets can be shared and consumed by users in a few clicks. To publish a dataset on Server, right-click the desired action which result you would like to publish, and choose “Send to Server”.

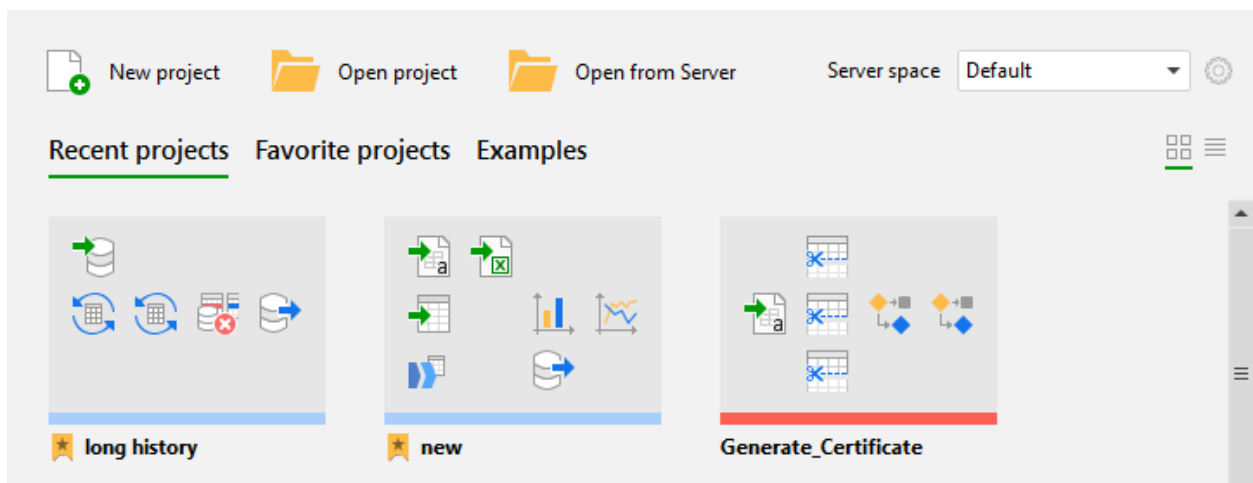
Right-click the workspace and choose “Receive from Server” to obtain a dataset from Server.



Screenshot 5: Receiving dataset from Server.

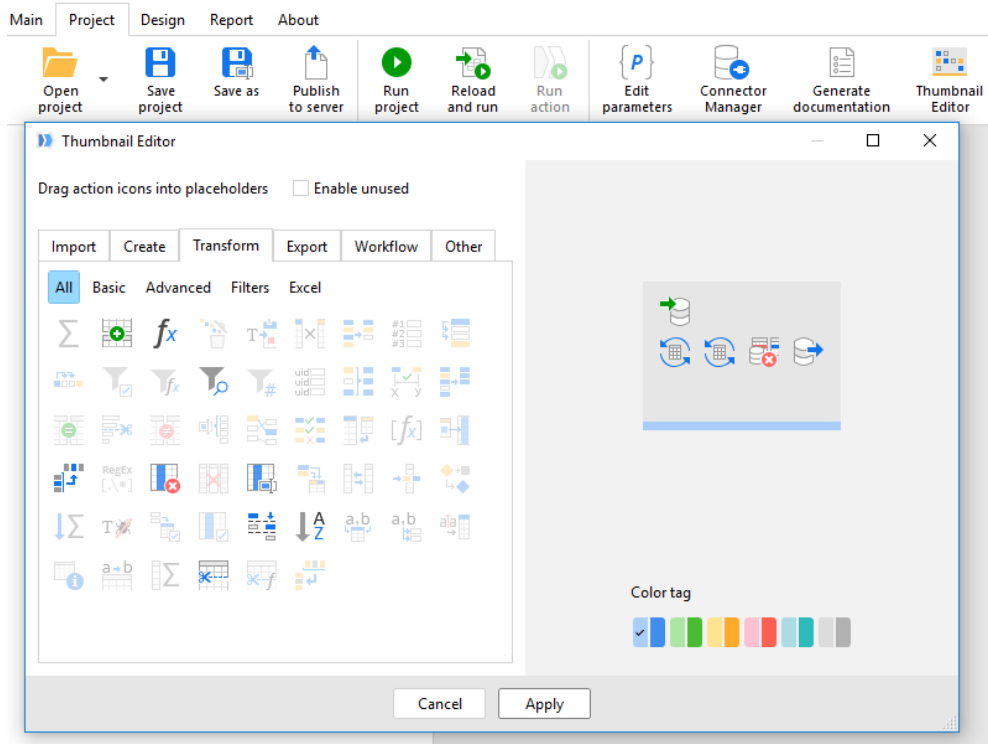
Project thumbnails and Thumbnail Editor

Thumbnail view has been added in addition to the list view of projects on the Start screen. It shows projects as thumbnails.



Screenshot 6: Thumbnail view.

Project thumbnails can be created and edited using the Thumbnail Editor available in the “Project” menu. Future versions of EasyMorph will add the ability to automatically generate thumbnails.



Screenshot 7: Thumbnail editor.

Miscellaneous

- PDF reports now automatically skip a page with report objects that don't contain any data.
- Action "Calculate" in the action right-click menu has been renamed to "Run action" and is now always enabled.
- Now the Start screen by default shows recently used database connectors instead of recently saved queries.
- The "Send to file" command in the action right-click menu now contains 3 new options: Tableau (.tde), Qlik (.qvd), and Dataset (.dset).
- The column profiler now shows the total sum of all numbers in the selected column.

What's new in EasyMorph Server

Low memory warning

Now, when a task starts but less than 10% of RAM is available to EasyMorph Server it writes a warning into the server log.

Update to .NET 4.6.2

The Server now requires .NET framework version 4.6.2 or above. If you keep your Windows updated no action is required as .NET 4.6.2 (or later) most likely has been already installed on your computer.

Release notes for version 3.9

Important! Create a backup copy of your projects and connector repositories before updating to the new version. Projects saved in version 3.9.x can't be opened in earlier versions.

What's new

Fetching emails with attachments

With the help of the “Fetch email” action it is now possible to retrieve email messages and their attachments from Microsoft Exchange and email servers that support the IMAP protocol. The action produces a table with a list of attributes such as sender, received date/time, subject, etc. for each email message. If the action is instructed to fetch attachments then for each attachment the table will have a file name and path to the folder where the attachment has been saved. Later, the attachments can be loaded using either the new “Load file list” mode in import actions (described below), or using iterations.

The screenshot displays two side-by-side panels of the "Fetch email" action configuration. Both panels have a title bar with a green envelope icon and the text "Fetch email". Below the title bar is a description: "Retrieve emails and generate a table with email attributes (sender, subject, etc.). Save attachments. [More](#)".

Both panels share the same "Connector" dropdown menu, which is set to "Exchange".

Under "Fetch from", both panels have radio buttons for "Inbox" (selected) and "Other folder".

The left panel has three tabs: "Pre-filter", "Attachments", and "Options". The "Pre-filter" tab is active, showing the following options:

- ☒ Pre-filter by sender: A dropdown menu with "@easymorph.com" selected.
- ☐ Pre-filter by subject
- ☒ Fetch only unread messages

Below these is the "Messages to fetch" section with two radio buttons:

- ☐ Last 100 messages
- ☒ Last 24 hours

The right panel also has three tabs: "Pre-filter", "Attachments", and "Options". The "Attachments" tab is active, showing the following options:

- ☒ Fetch attachments: A note below says "This will create a new line for each attachment."
- Folder path: A dropdown menu with "D:\attachments" selected.
- ☒ For each message create a new temporary subfolder with attachments
- ☐ Save all attachments from all messages into one folder
- ☐ Enable images in attachments

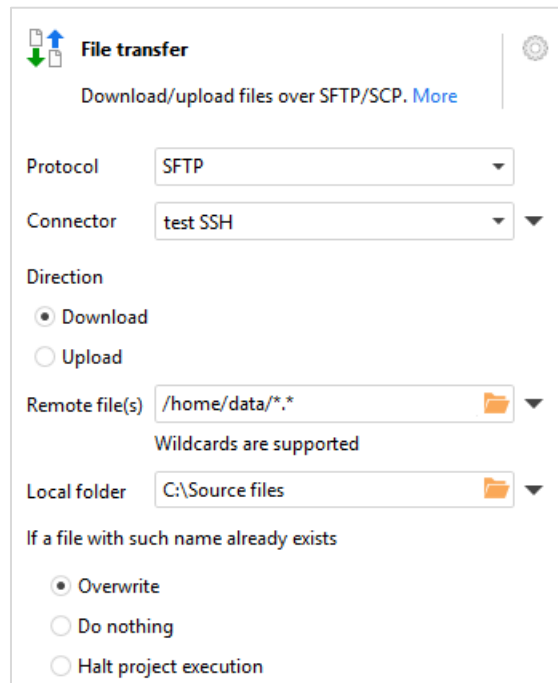
Screenshot 8: The "Fetch email" action.

A typical workflow that fetches email attachments and loads them in EasyMorph would be as follows:

1. Use the “Fetch email” action to retrieve email messages and save their attachments into a designated folder. The action produces a table with message attributes and paths to folder(s) with attachments.
2. Filter messages/attachments based on their attributes, if necessary.
3. Use an import action in the “Load list of files” mode (described below) to load retrieved attachments into EasyMorph.
4. Use the “Clean folder” command of the “File command” action in order to purge attachments. In some scenarios, this step may come before fetching emails.

SFTP/SCP file transfer

EasyMorph now has a built-in SFTP/SCP client that allows downloading and uploading files from/to a remote computer over an SSH connection. A typical use case would be collecting source data from a remote computer, or publishing transformation results to a remote server.



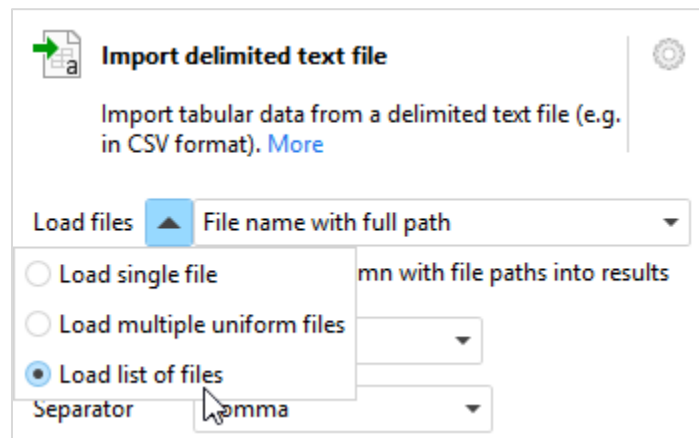
The screenshot shows the 'File transfer' configuration window. At the top, there's a title bar with a file transfer icon and the text 'File transfer'. Below it, a subtitle reads 'Download/upload files over SFTP/SCP. [More](#)'. The main configuration area includes several fields: 'Protocol' is set to 'SFTP'; 'Connector' is set to 'test SSH'; 'Direction' has two radio buttons, 'Download' (selected) and 'Upload'; 'Remote file(s)' is set to '/home/data/*.*' with a folder icon and a dropdown arrow, and a note below it says 'Wildcards are supported'; 'Local folder' is set to 'C:\Source files' with a folder icon and a dropdown arrow. At the bottom, there's a section titled 'If a file with such name already exists' with three radio buttons: 'Overwrite' (selected), 'Do nothing', and 'Halt project execution'.

Screenshot 9: The "File transfer" action.

Loading a list of files without iterations

All import actions now have a new mode – “Load list of files”. In this mode the list of file paths to load is obtained from a column in the input dataset (i.e. the output of previous action). This mode allows loading multiple files at once without using iterations. It can be helpful in cases when uniform files are

located in different folders, or need to be filtered using a complex condition based, for instance, on file size or file creation date. In most typical cases, an initial list of files would be generated using the “List of files” action or the “Fetch email” action.

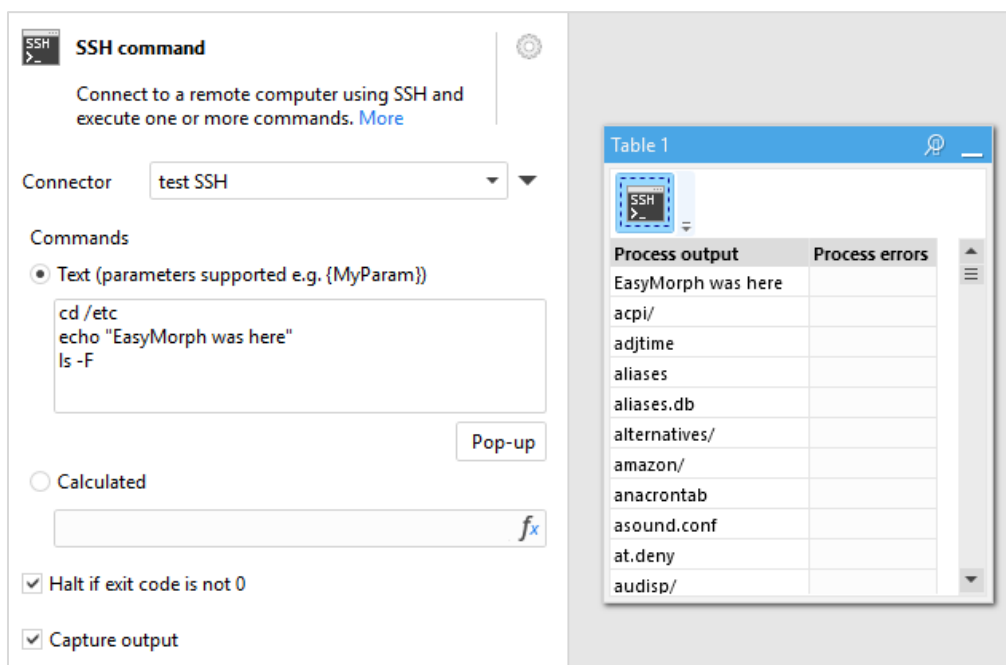


Screenshot 10: The “Load list of files” mode.

SSH Command

The **SSH Command** action allows executing one or multiple commands on a remote computer over an SSH connection. The commands can be composed using project parameters. The output of commands can be captured in EasyMorph.

A typical use case for “SSH Command” would be triggering a process on remote server and later fetching files (using the “File transfer” action) that were produced by that process.



Screenshot 11: SSH Command.

Native data format

The native data file format stores data in a form that is very close to internal data representation in the EasyMorph's in-memory engine. Therefore, saving a dataset into a file and reading a dataset from a file doesn't have performance overhead such as data conversion or compression/decompression, and therefore is very fast.



The **Export dataset** action saves an EasyMorph dataset into a file in the native format.



The **Import dataset** action loads an EasyMorph dataset from a file.

Changes to existing actions

The **File Command** action now has a few new commands to deal with folders and zip archives:

- Create folder
- Delete folder
- Zip file
- Zip folder

Also the “Clean folder” command now allows removing subfolders or cleaning them recursively.

The **Send email** action now doesn't fail if the message body contains a missing parameter. This is done in order to enable embedded CSS stylesheets in HTML messages.

New functions

weekdayname(date_as_number) – returns the abbreviated name of day. Example:

`weekdayname(#2018-08-22)` returns “Wed”.

New limits of the free edition

The limits for actions and iterations in the free edition have been reduced to 24 actions per project (including iterated projects) and 24 iterations. The free edition now allows using the “Input” action.

Connectors

This version introduces a new connector type – “SSH connection”. The connector can be used in two actions: **File Transfer** and **SSH Command**. It can use authentication with a key pair. The private key should be in the Open SSH format (for keys generated with PuttyGen see instructions [here](#)).

The email connector has been extended with options for configuring IMAP.

Miscellaneous

- The minimal version of .NET Framework required for EasyMorph is now 4.6.2.
- Database drivers and Tableau SDK have been updated to the latest versions.

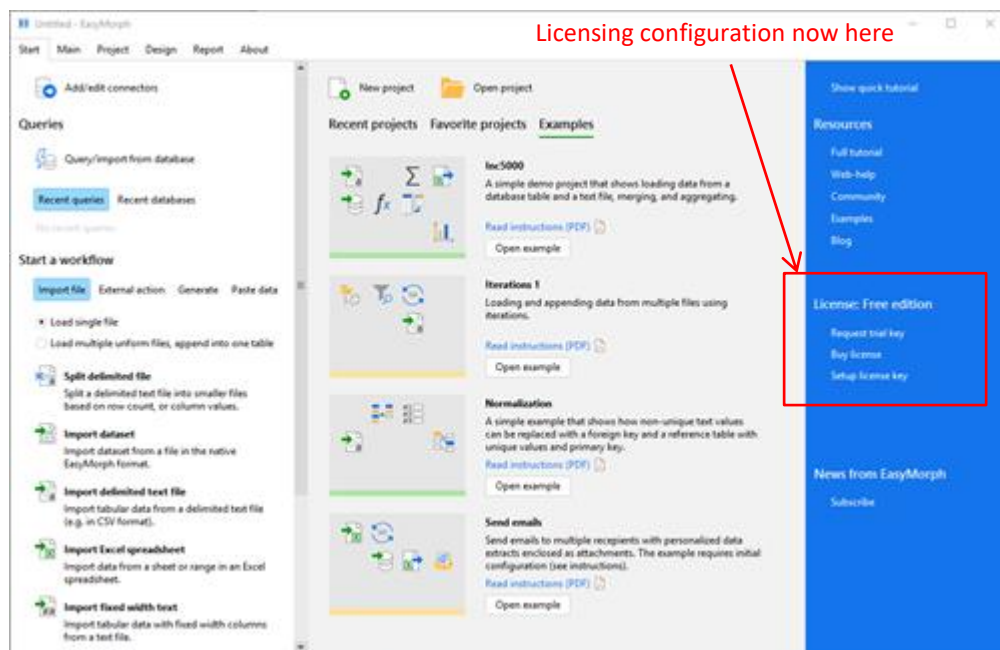
What's new in EasyMorph Desktop

Editing operations with multiple actions

Now it is possible to copy, paste, move to another/new table, or delete multiple actions at once. To select multiple actions, *click* the first action then *Shift+click* the last action. After selecting multiple actions, right-click for context menu, or drag them to another location.

New Start screen

The Start screen has been significantly redesigned to simplify onboarding experience for new users and dealing with many projects for existing users. A trial license key can be requested right from the Start screen. Also license key configuration has been moved from the About menu to the Start screen.



Screenshot 12: New Start screen.

Miscellaneous

- On the first launch EasyMorph now shows a wizard to help configure licensing (e.g. request a trial key) and subscribe to news and updates.

What's new in EasyMorph Server

Monthly Schedule

EasyMorph Server now allows scheduling tasks using a monthly schedule. Days of execution can be defined in one of the two ways:

- As one or more month days (1, 2, ..., Last day, 1 day before last day, 2 days before last day).
- As one or more week days (Sun, Mon, ...) for one or more weeks (First, Second, ..., Last).

The screenshot shows the configuration interface for a task in EasyMorph Server. At the top, there are two buttons: "Save changes" (in blue) and "Revert". To the right, there is a checkbox labeled "Task disabled" which is currently unchecked. Below these, the "Project" field contains "train.morph" with a file icon and a "Browse..." button. The "Note" field contains "Some description of the project". Under "Project parameters", there is a button "Assign more parameters". The "Schedule" dropdown is set to "Monthly". Below this, there are two radio buttons: "Days" (selected) and "On". The "Days" field shows "1, 14, Day before last day". The "On" field shows "1st, 3rd" and "Mon, Thu". The "Time" field shows "11:45 am" with a clock icon. At the bottom, there is a checkbox labeled "Enable Get" which is unchecked.

Save changes Revert		<input type="checkbox"/> Task disabled
Project	train.morph Browse...	
Note	Some description of the project	
Project parameters	Assign more parameters	
Schedule	Monthly ▲	
<input type="radio"/> Days	1, 14, Day before last day ▲	
<input checked="" type="radio"/> On	1st, 3rd ▲	Mon, Thu ▲
Time	11:45 am	
Enable Get	<input type="checkbox"/>	

Screenshot 13: Monthly schedule