

# Tutorial: Execute Commands

## Organizing Command Macros

Bob Zawalich June 7, 2021

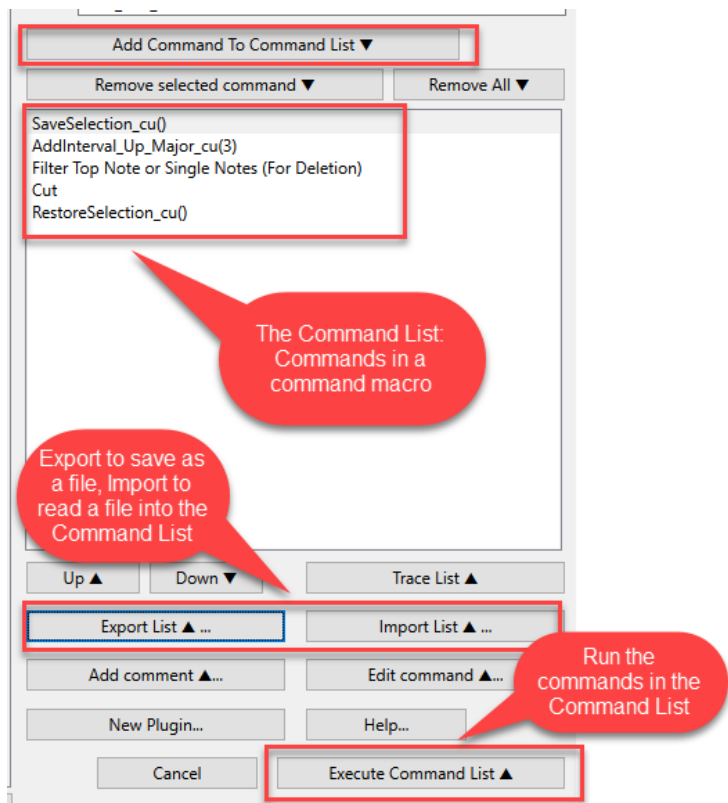
**Command Macros** are sequences of commands that can be built in the **Execute Commands** plugin, and saved as files using the **Export List** button.

These instructions refer to the most recent versions of the **Execute\_Commands** and **Run Command Macro** plugins. Please install the most recent versions of the plugins for best results.

### Creating and Exporting Command Macros in Execute Commands

**Execute Commands** saves these in a special **Execute\_Commands** subfolder of your default **Scores** folder, which was set up in Sibelius in **Files>Preferences>Saving and Exporting>Saving Scores**.

Here is the Command List of Export Commands:



You can use **Import List** to find your saved Command Macros and read them back into the Command list, and then use **Execute Command List** to run the commands, but that is not a very efficient way to use Command Macros.

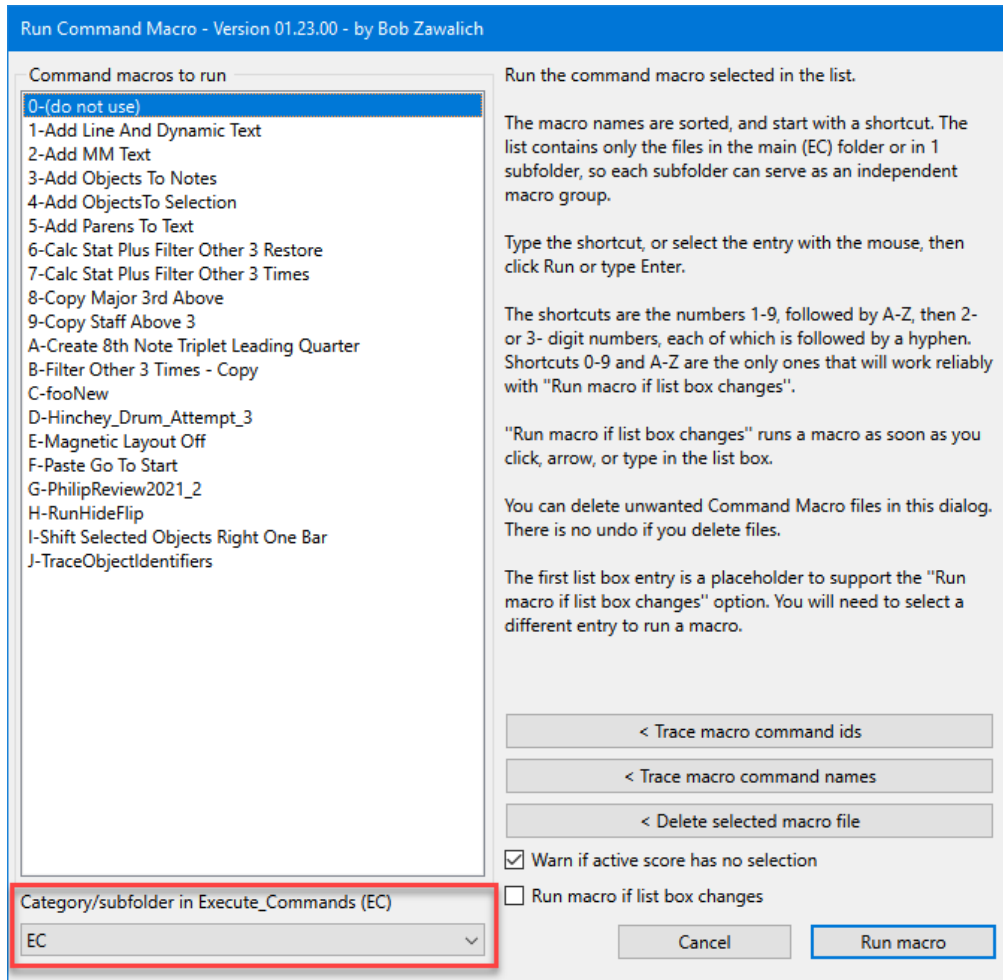
Instead, use **Execute Command** to create and test your Command Macros, and use the plugin **Run Command Macro** to run them.

### Run Command Macro

At the basic level, **Run Command Macro** puts all the Command Macros you have exported into a list box, sorted by name, and gives them a numeric or alphabetic index (1 – 9, then a – z, then multi-digit numeric indexes).

To run a Command Macro, select its name in the list box and press **Enter** to run it. If you check the checkbox **Run Macro if list box changes**, the plug-in will run as soon as a new selection is made in the list box.

It can be very useful to assign a keyboard shortcut to **Run Command Macro**. You can type the shortcut, then type the index to the Command macro you want to run, then Enter, to have a fairly quick mechanism to run the macro. (If you use **Run Macro if list box changes**, you will save one more keystroke each time).



If you keep writing and exporting macros, the list becomes longer and harder to deal with. After you use up the slots for 1 – 9 and a – z, the indexes become multi-digit numbers, and do not work well with **Run Macro if list box changes**. The macro names are sorted alphabetically, but if you have 100 or more macros they get hard to find.

**Run Command Macro** tries to help by letting you display Command Macros that are in your Execute\_Commands (EC) folder, and those in subfolders of the EC folder, as separate lists.

Use the drop down list box labeled **Category/subfolder in Execute Commands** in the lower left corner to select the commands in a different subfolder.

Each sub-list has its own set of automatically-generated indexes, and if you keep the number of Command Macros in a subfolder to 35 or fewer, you will have easy-to-use single-digit indexes in each list.

## Organizing Command Macros into Subfolders

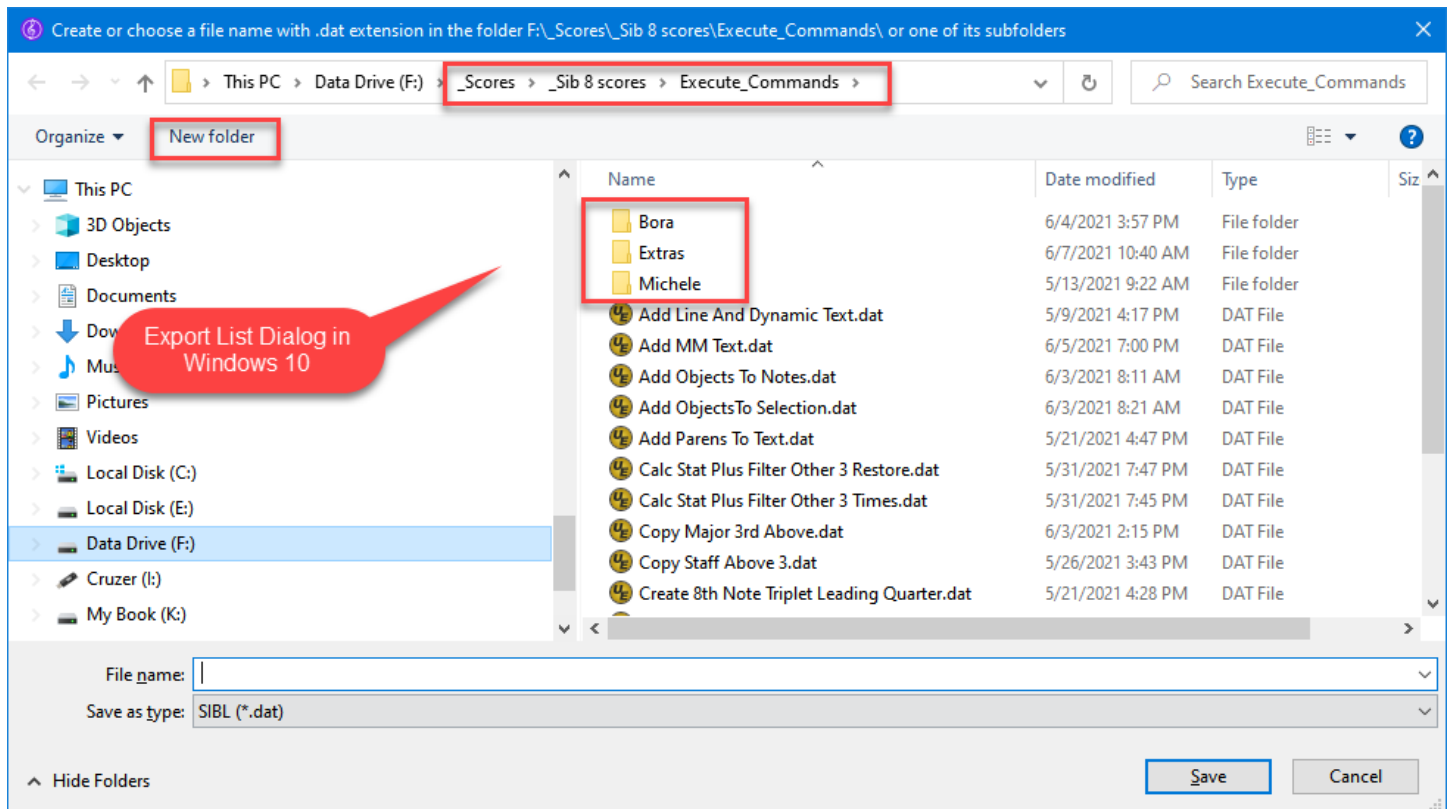
To make this work effectively, set up your **Execute\_Commands** subfolders to help with a given workflow. You can have sets of macros you use for specific projects, or you can set up categories of commands that are all for specific purposes.

I am tending to keep my most all-purpose macros in the main **Execute\_Commands** folder (which **Run Command Macro** refers to as **EC**), and set up subfolders for specific tasks. If you get too many macros in a folder, you can create another subfolder, and move the files around.

It is probably best to do this sort of thing in the file system (File Explorer or Finder), rather than in one of the plugins. You will need to find your default **Scores** folder, and then go into the **Execute\_Commands** folder, which Execute Commands will create for you the first time you use **Export List**.

You can change folder and move the files around at will, except that you always need to start at the **Execute\_Commands** subfolder of the **Scores** folder, since **Execute\_Commands** and **Run Command Macro** always start there.

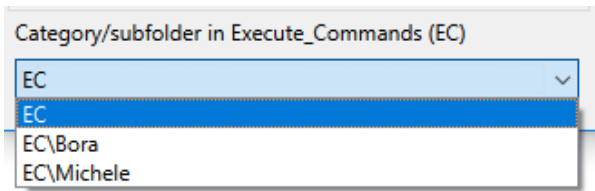
If you use **Export List** in **Execute Commands**, the dialog will start at the **Execute\_Commands** folder, but you will see the subfolders available, and you can create new subfolders. You can actually do a fair bit of file manipulation in this dialog if you want, including moving files from one subfolder to another.



The plugins will adapt to changes in file location on the fly. **Run Command Macro** analyzes the subfolder structure and creates separate Command Macro sets each time it starts. Changing the file order will change the indexes used, so be aware that there are not always going to be the same index if you change the contents of the subfolders.

## Changing subfolders in Run Command Macro

This is easy. Drop the listbox at the bottom left of the dialog and choose the subfolder you want to use.

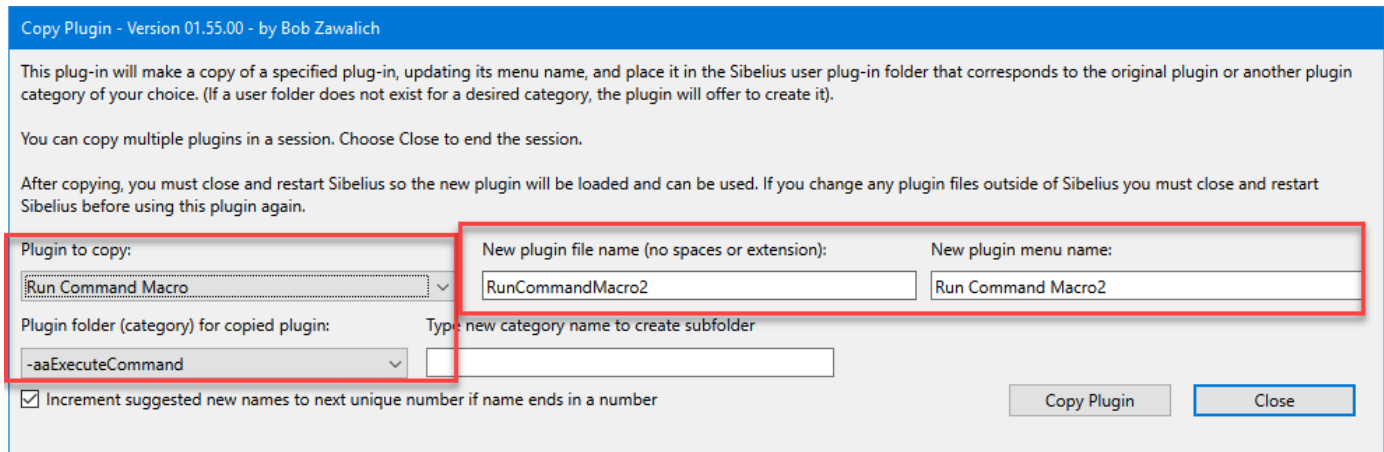


## Cloning Run Command Macro

If you have a number of subfolders, you might find it inconvenient to have to bring up **Run Command Macro** and change the subfolder all the time.

One workaround for this is to make multiple copies (clones) of **Run Command Macro**, and set each of them to start in a different subfolder (the subfolder setting is remembered across Sibelius sessions). Then you can set up a different shortcut for each clone.

This technique is used in the Focus Sets/ Focus Families plugins, which provide a set of cloned plugins, but it is possible to make your own clones using the downloadable plugin **Copy Plugin**.



In **Plugin to copy** you will find all your installed plugins in alphabetical order. When you choose **Run Command Macro**, the **Plugin folder (category)** will be set to the home folder for **Run Command Macros**, but you can change that.

You need to specify both a file name and a menu name for the clone plugin, and I recommend that the 2 names have the same text, with the menu name having spaces between the words, and the file name having no spaces, but these are only guidelines.

**Copy Plugin** automatically suggests the clone name as the original name with a number added to the end of the names, but in this case I would suggest naming the plugins to suggest the purpose of the clone, such as **Run Command Macro Text** or **Run Command Macro Orchestration**.

When you click on **Copy Plugin**, a new plugin file is created, which is the same as the original but with a new file name and a new menu name. The menu name change is actually a change to the text of the plugin file.

After making as many copies as you want, you will need to close and restart Sibelius. When you restart it, run each of the close and set up the subfolder you want each of them to run in, and assign shortcuts to the plugins, or set them up to run in a plugin like **My Plugins** or **Run Plugin By Name**.

And that is about all there is to it.