

# Creating and managing workflows

**i** Workflows in the [EPISODES Platform](#) is a mechanism that allows for additional organization of the data processing inside the workspace. They allow for a configuration of the applications execution in such a way, that one application is automatically started after another, being fed data that may either be specified beforehand or be produced by the preceding applications.

## Contents of this guide

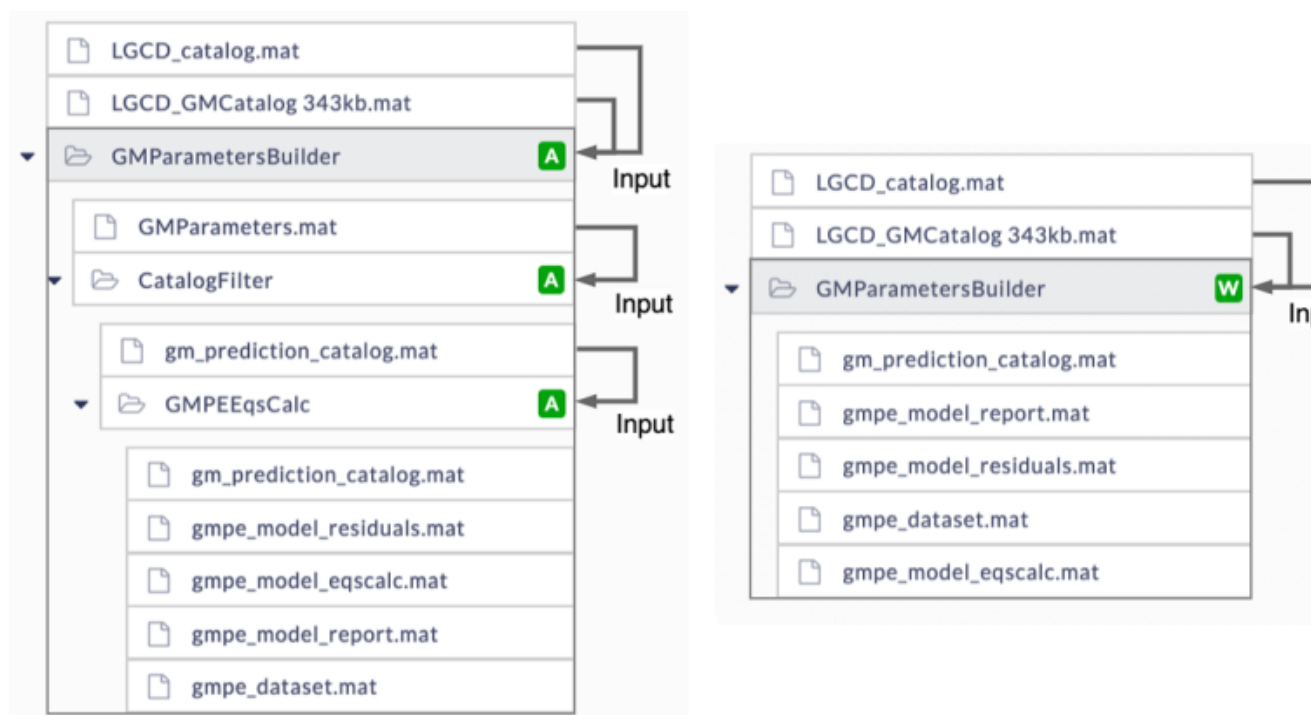
With this guide you will learn about workflows - a mechanism that will allow you for automating and managing the data processing inside your workspace. We will tell you how the workflows are constructed and how to use and edit them. Workflows are similar to regular [EPISODES Platform](#) applications, therefore, a basic knowledge of the what are these applications and how they are run is a requisite for using workflows, therefore, if you are not familiar with working with the applications yet, please check [Applications Quick Start Guide](#) first.

## What is a workflow?

A workflow is a sequence of tasks that processes a set of data. It can, in an automated way perform several actions that would, otherwise, have to be invoked one by one. Within the EPISODES Platform, the tasks composing a workflow are the applications - any set of applications present in your workspace can become a workflow. We will show the usage of the workflow mechanism on an example using the application [Ground Motion Prediction Equations: GMPE calculation](#) and two other applications that prepare data to be used in the former. The sequence of the applications used is:

1. [Ground Motion Parameters Catalog builder](#) (see the [application user guide](#))
2. [Catalog Filter](#) (see the [application user guide](#))
3. [Ground Motion Prediction Equations: GMPE calculation](#) (see the [application user guide](#))

Normally, when used in workspace, the three applications have to be managed separately, therefore, to run the last application (*Ground Motion Prediction Equations: GMPE calculation*), one has to first run the *Ground Motion Parameters Catalog builder*, then use its results for *Catalog Filter*, and finally, on the results of the latter, create and run the *Ground Motion Prediction Equations: GMPE calculation*. The amount of work required for running such a sequence of application grows with the number of the component applications, while when using a workflow, the whole sequence is reduced to one operation. Figure 1 shows a comparison between the aforementioned sequence of applications located in workspace and a workflow created from them.

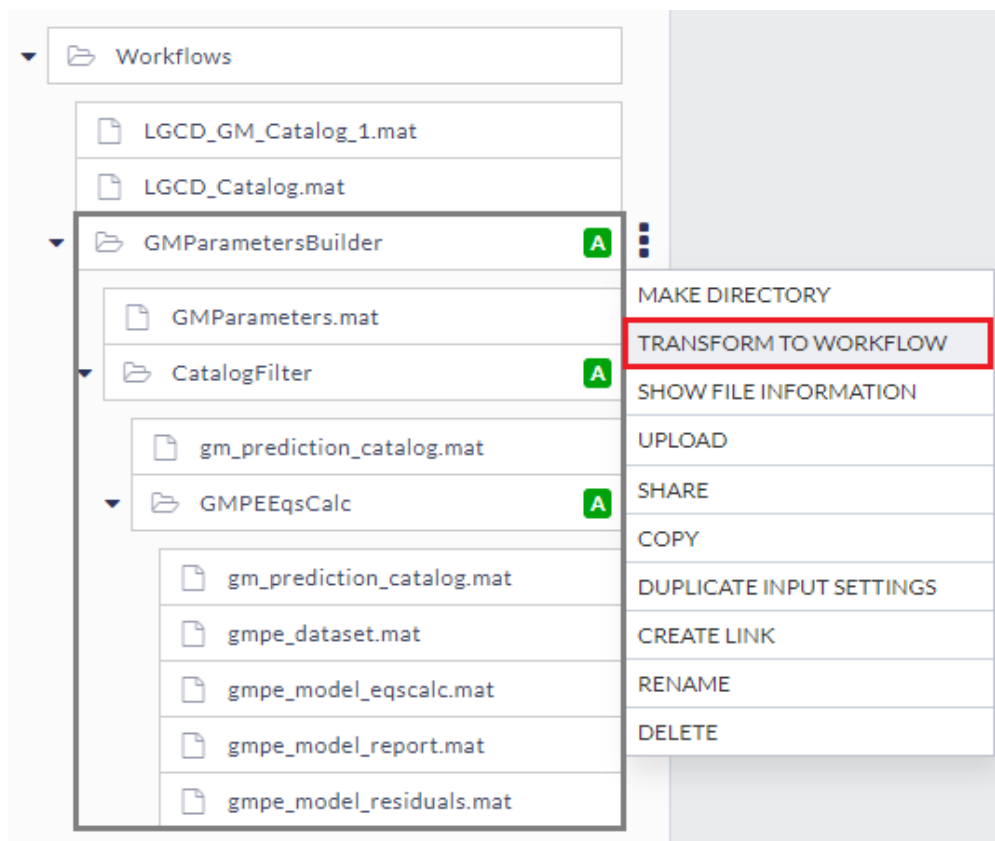


**Figure 1. Comparison of application sequence in workspace and a workflow created from this sequence**

A workflow constructed within the EPISODES Platform assumes that the user (a scientist) experiments with the data and its analyses performed by the applications in workspace, and when finding a combination of data and parameters that suits their research best saves that as a workflow. While a workflow itself is a known concept, the approach presented here is quite different from the most common workflow solutions (see [Related publications](#))

## Creating a workflow (transforming workspace content into a workflow)

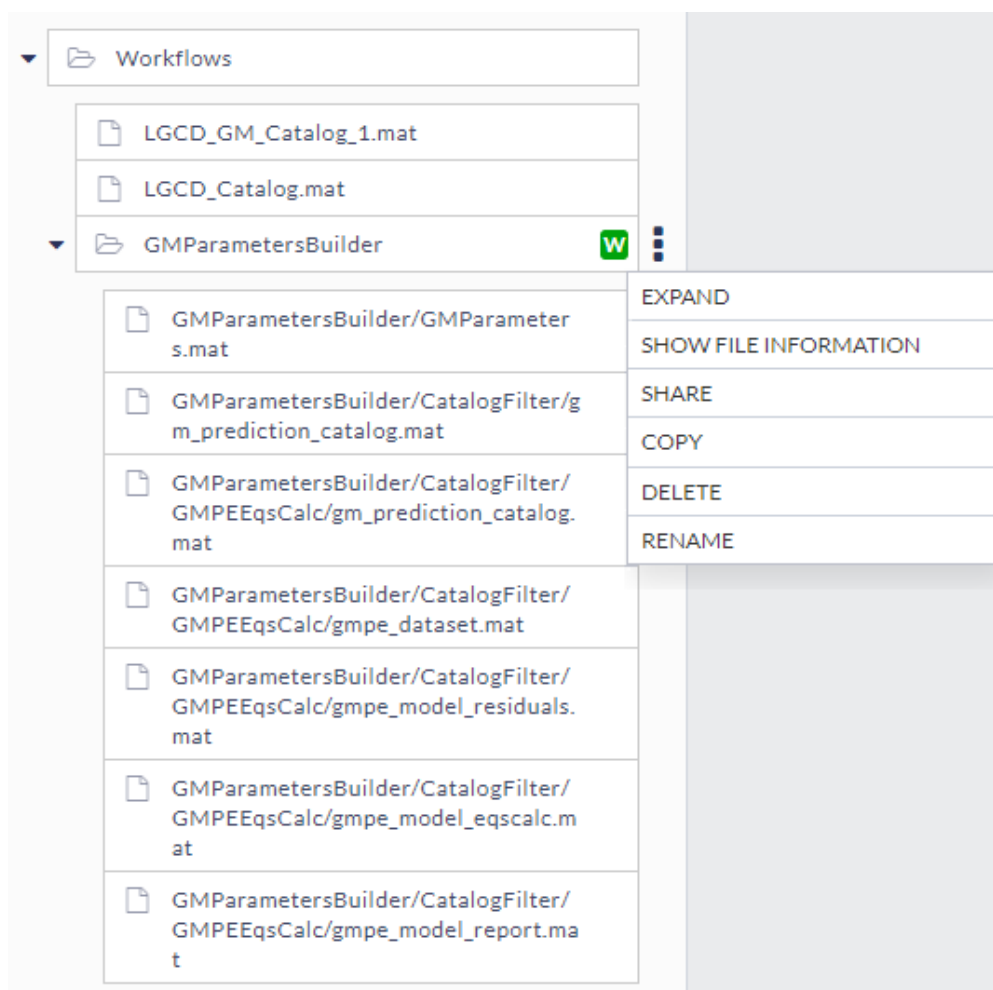
A workflow can be created by choosing the *Transform to workflow* action (see Figure 2) from the file menu of any directory (except from the root directory - /) in workspace. Note, that an application is also a directory, so you can also create a workflow for an application item in workspace.



**Figure 2.** File menu displayed for a directory in workspace, with *Transform to workflow* action marked in red. The marking in grey shows the content that will become internal to the created workflow

The *Transform to workflow* action will create a workflow from the whole sub-tree starting from the directory on which we initiated the action - in Figure 2, this will be application *GMParametersBuilder*, and the workflow will contain the part of the tree marked in grey. After this action, workspace will contain only single entry marked with "W" (meaning workflow) listing results from all the applications that were included in the workflow (you will learn how to configure which results should be shown and which not, further in this guide). In the example from Figure 2, creating the workflow, will produce a result as shown in Figure 3.

The workflow item in workspace (see *GMParametersBuilder* item in Figure 3) is similar to an application item - it is displayed as a directory with a colorful status icon on the right. As with other workspace items, it has an action menu available (see Figure 3). The action *Expand* is specific only to workflows and it is a reverse operation to creating workflow (to the *Transform to workflow* operation, visible in Figure 2) - it transforms the workflow again to a regular directory. Other actions are similar to the actions available for a directory or application (compare [My Workspace Quick Start Guide](#) and [Applications Quick Start Guide](#)), however, note, that the *Rename* action changes the only the name of the workflow, not the name of the underlying directory. Actions like *Upload* are not available for a workflow, as the directory structure inside is not visible when a workflow is created and the result of such action could be ambiguous (see also the [Editing workflow section](#)). Note also that, if there were any files in the base workflow directory (in our example, the *GMParametersBuilder* directory) that were not used by any of the applications, they will simply be hidden from you, as the workflow would only show the content related to the applications inside. Therefore, if you want to have access to these files when the workflow is created, move them outside of its directory. Creating a workflow is a fully reversible operation, as it does not change the logical structure of applications nor files inside. Therefore, in case of any amendments needed within the directory structure underneath, the workflow can be expanded (action *Expand*, described earlier) and created again after the necessary corrections.



**Figure 3. Directories from Figure 2 after creating the workflow, with file menu displayed**

Note that, in the example above, some inputs - `LGCD_Catalog.mat` and `LGCD_GM_Catalog_1.mat` in Figure 3 - are outside of the `GMParametersBuilder` directory. Files like these - inputs of the applications inside the workflow that are outside of the transformed directory will become workflow inputs - see also the description below and Figure 4.

As in case of any other application or file inside the workspace, a workflow can be displayed by clicking on its item in the workspace tree (`GMParametersBuilder` item in Figure 3). Once the workflow is shown, you can see the following contents:

- Inputs (marked with **(1)** in Figure 4) - files that are input to the workflow, grouped by the applications that take the files as inputs.
  - Note, that only inputs that were outside of the transformed directory are editable. In the example we used before, `LGCD_Catalog.mat` and `LGCD_GM_Catalog_1.mat` files were outside the `GMParametersBuilder` directory. Other inputs are hidden as, if they were inside the directory, it meant that they were passed from one application to another. You can modify them after expanding the workflow back to directory.
- Forms (marked with **(2)** in Figure 4) - forms of the applications inside the workflow, again, grouped by the application. By default, forms for all applications are displayed, but they can later be hidden by [editing the workflow](#).
- Controls for running the workflow (marked with **(3)** in Figure 4) - controls related to running and filling forms, similar to those that are displayed for a regular applications, but this time, they apply to the whole workflow - i.e. the *Run* button will execute all applications inside the workflow in the correct order, and the *Save* button will save all the forms, etc.
- Status (marked with **(4)** in Figure 4) showing the overall workflow status and individual application statuses.
- Outputs (marked with **(5)** in Figure 4) - outputs of the applications inside the workflow, grouped by the application. By default, outputs of all applications are visible in this section, but you can choose which outputs are visible in the workflow when [editing the workflow](#).

The contents are each displayed in the same order as the order in which the applications were added in the source directory.

The screenshot shows the 'GMPParametersBuilder' workflow interface. At the top, there's a header with the workflow name and an 'ACTIONS' button. Below this is a 'Description' field containing the text 'Workflow composed of 3 applications GMPParametersBuilder, CatalogFilter, ...'. The main area is divided into 'INPUTS' and 'OUTPUTS' sections. The 'INPUTS' section contains two input fields: 'Catalog' (with value 'workflow/LGCD\_Catalog.mat') and 'Ground Motion Catalog' (with value 'workflow/LGCD\_GM\_Catalog\_1.mat'). Below these are four dropdown menus for 'GM Parameters', 'Source characterization parameter', 'Path characterization', and 'Site characterization'. The 'OUTPUTS' section shows three output fields: 'GMPParametersBuilder', 'GMPParametersBuilder/CatalogFilter', and 'GMPParametersBuilder/CatalogFilter/GMPPEqsCalc'. At the bottom right, there is an 'EDIT / DEBUG WORKFLOW' button. Numbered annotations (1-6) point to specific elements: (1) points to the 'GMPParametersBuilder inputs' section header; (2) points to the 'GMPParametersBuilder/CatalogFilter/GMPPEqsCalc forms' section header; (3) points to the 'RUN' button; (4) points to the status messages for the three applications; (5) points to the 'OUTPUTS' section header; and (6) points to the 'EDIT / DEBUG WORKFLOW' button.

Figure 4. View of an open workflow with most important elements marked

## Editing the workflow

Editing workflow will allow you to customize which forms and outputs are visible within the workflow view (Figure 4). Not all the features of a workflow are editable, since some of them depend on the structure of the underlying directory. For this reason you cannot remove or add applications to workflow, nor change their order. Neither can you change which are the inputs to the workflow, as they are based on the structure of the inputs of the applications inside (see also previous section). To edit the workflow structure, you need to expand it (see previous section) and create again.

To edit the workflow, choose the *Edit / debug workflow* button marked with (6) in Figure 4. This will change the workflow view, into an editable form - a view similar to previous one, but with additional options to control the editable features - each of them is described below in the order in which it appears in the workflow edit view. We recommend to edit the workflow after its structure is well established, as operation of expanding it, will erase all the changes done while editing.

### Workflow description and application names

Figure 5 shows the part of the editing view that allows to edit the workflow description and application names. The workflow *Description* is a summary, which you can fill (by editing the field marked with (1) in Figure 5) with your description of what the workflow does and what it should be used for. By default it is filled with the list of the applications that constitute the workflow. The application names are your custom names displayed next to the application individual inputs, forms, statuses and outputs (see markings (1,2,4,5) in Figure 4) The application names will also be displayed next to the workflow outputs in the workspace tree (see Figure 6, compare with Figure 3 - view before changing application names). By default, the application names are filled with the original names of the application directories inside the workflow - see fields marked with (2) in Figure 5.

### WORKFLOW DESCRIPTION

Description
1

Workflow composed of 3 applications GMParametersBuilder, CatalogFilter, GMPEEqCalc

Application path
Visible name

/
GMParametersBuilder

/CatalogFilter
GMParametersBuilder/CatalogFilter

/CatalogFilter/GMPEEqCalc
GMParametersBuilder/CatalogFilter/GMPEEqCalc

### WORKFLOW DESCRIPTION

Description

Workflow for Ground Motion Prediction Equation (GMPE) estimation with input construction and filtering

Application path
Visible name

/
Input creation

/CatalogFilter
Catalog filtering

/CatalogFilter/GMPEEqCalc
GMPE

Figure 5. Part of workflow editing view containing workflow description and application names, with most important elements marked, filled with default values (upper figure) and after changing to custom names (lower figure)

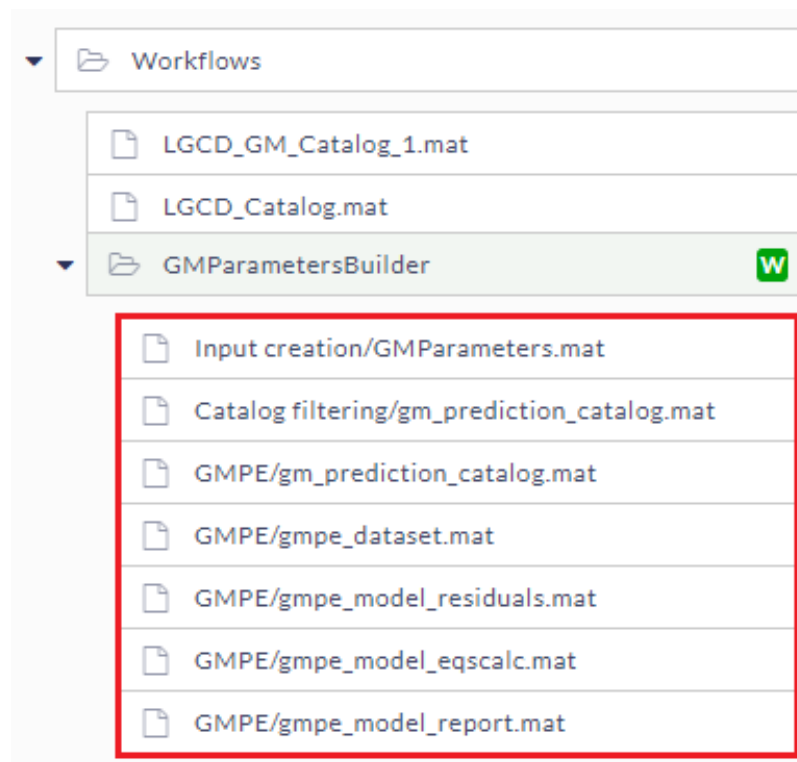


Figure 6. Workflow from Figure 3 after changing the application names as in Figure 5

### Forms and outputs visibility

Figure 7 shows the part of the editing view that allows to edit the workflow forms visibility. The form for each application can be shown or hidden by using the '+' and '-' buttons, respectively. After using the '-' button (marked with **(1)** in Figure 7), the form is added to *Hidden forms* (marked with **(2)** in Figure 7). It can be made visible again, by using the '+' button (marked with **(3)** in Figure 7). The hidden forms will not be displayed within the main workflow view, which implies also that the parameters of this form will not be able to be edited when running the workflow. By default, all forms are visible.

**▼ GMPE forms** 1 -

GM Parameters Peak\_Accelerations ▼

PGA\_E ▼

Source characterization parameter None ▼

Path characterization None ▼

Site characterization None ▼

**HIDDEN FORMS** 2

**i** These forms will not be visible for the end user. You can add forms to workflow view by clicking pulsing + buttons.

► Input creation forms +

► Catalog filtering forms +

3

Figure 7. Part of workflow editing view containing controls of visibility of forms, with most important elements marked

A similar mechanism applies to outputs of the application - their visibility can be changed with similar '+' and '-' buttons as in case of forms. If any application output is set as hidden, it is additionally removed from the workflow view in the workspace tree (see Figure 8).

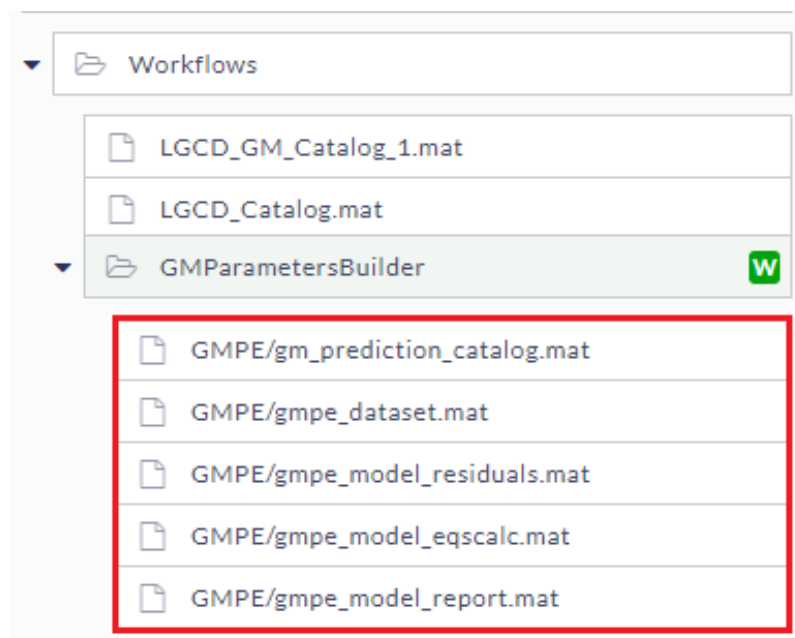


Figure 8. Workflow from Figure 6 after hiding the outputs of two first applications

### Debugging the workflow

The last part of the workflow edit view is dedicated to the workflow debugging (see Figure 9). The controls in this part give you more insight to the workflow structure and ability to change inputs that otherwise is not available. However, be careful, as manipulating this part can unexpectedly change the workflow structure, causing the workflow view to be desynchronized with the actual structure.

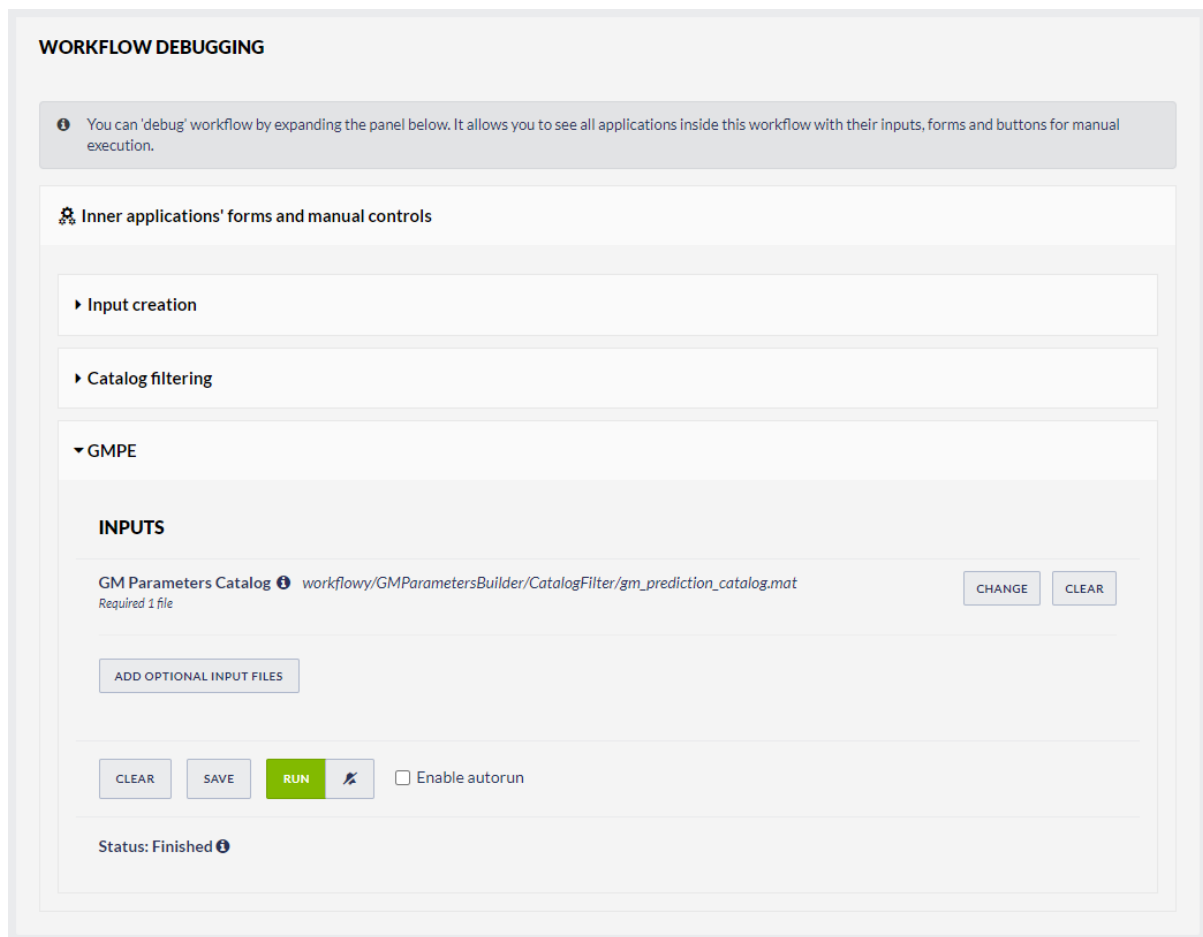


Figure 9. Part of workflow editing view containing workflow debugging options

### Saving changes

The changes done while editing the workflow can be saved using *Save workflow template* button in lower right corner (see Figure 10). After saving, the view is displayed again (as in Figure 4) with the changes applied.

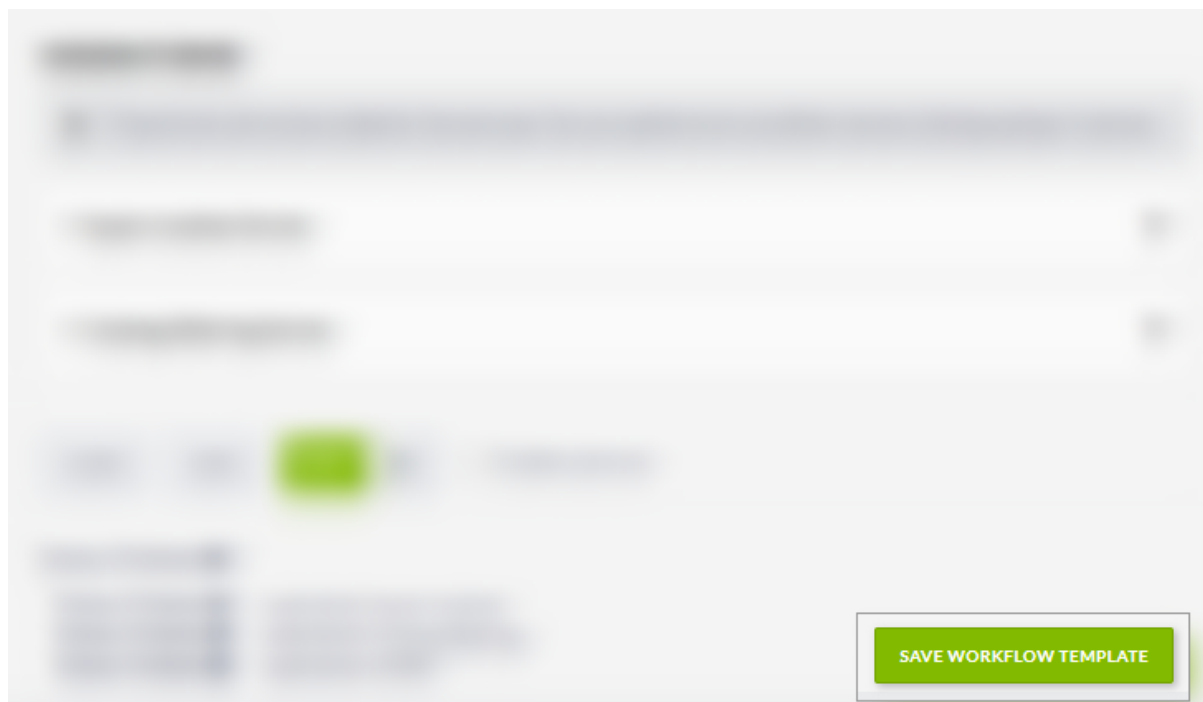


Figure 10. Workflow editing view with *Save workflow template* button visible



## Deleting / Expanding the workflow

Once a workflow is created, you have only limited options of reorganization of the applications inside and data flow between them (options described in section [Editing the workflow](#)). If, after creating it, you see that the workflow should have different inputs, or the flow of data between the applications inside should be organized differently, you can always expand it with the *Expand workflow* action accessible from the workspace item menu (see Figure 11). This operation will return the workspace to the state before creating the workflow (in our example, the workspace tree will come back to the situation from Figure 2). You can then rearrange the individual applications, add or delete them, and, when ready, create the workflow anew. Creating and expanding workflows is an operation only on the files metadata - the files are not moved nor copied, therefore, it does not affect the space in your workspace, nor takes much computing resources, therefore, you can create and expand workflows as you choose.

Note that, when expanding workflow, all the configuration done when [Editing the workflow](#) will be lost, therefore, we advise to make the workflow edition only after the workflow structure is settled.

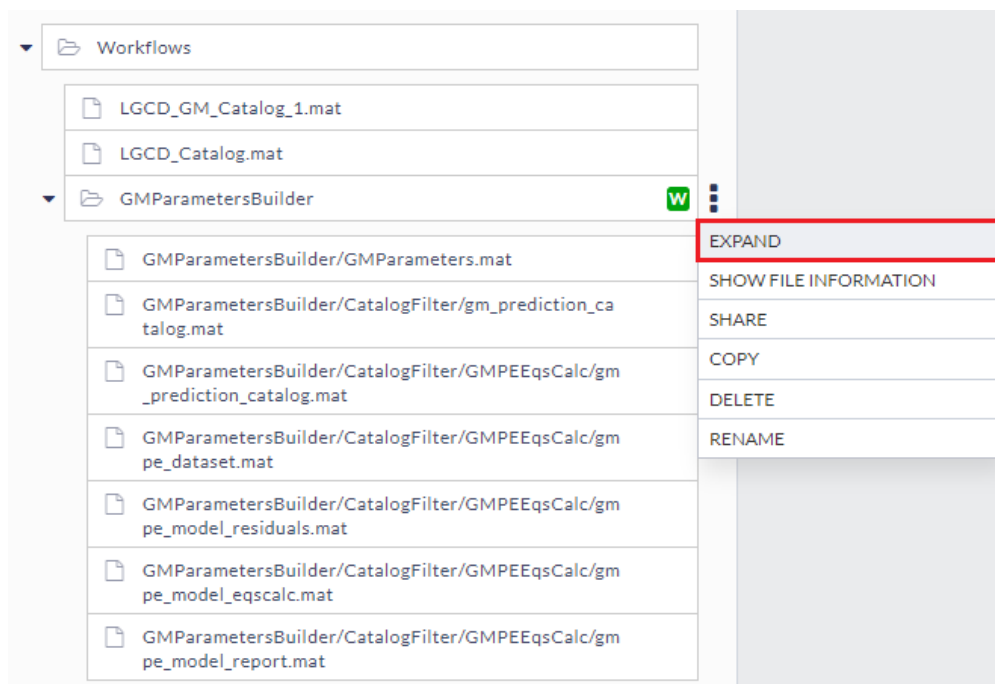


Figure 11. File menu displayed for a directory in workspace, with *Expand* action marked in red.

## Nested workflows

As we said in one of the first sections, a workflow is similar to an application. This means also that it can be used in another workflow - in similar way as an application does. Expanding on the example used in the previous sections (workflow created as in Figure 1), we can now add another application before the *GMPParametersBuilder* workflow - namely the *CatalogMerger* application, that will merge two Ground Motion Catalogs into one - the result will be used in the *GMPParametersBuilder* workflow instead of the original *LGCD\_GM\_Catalog\_1.mat* file - see Figure 12. The workflows can be nested like this any number of times.

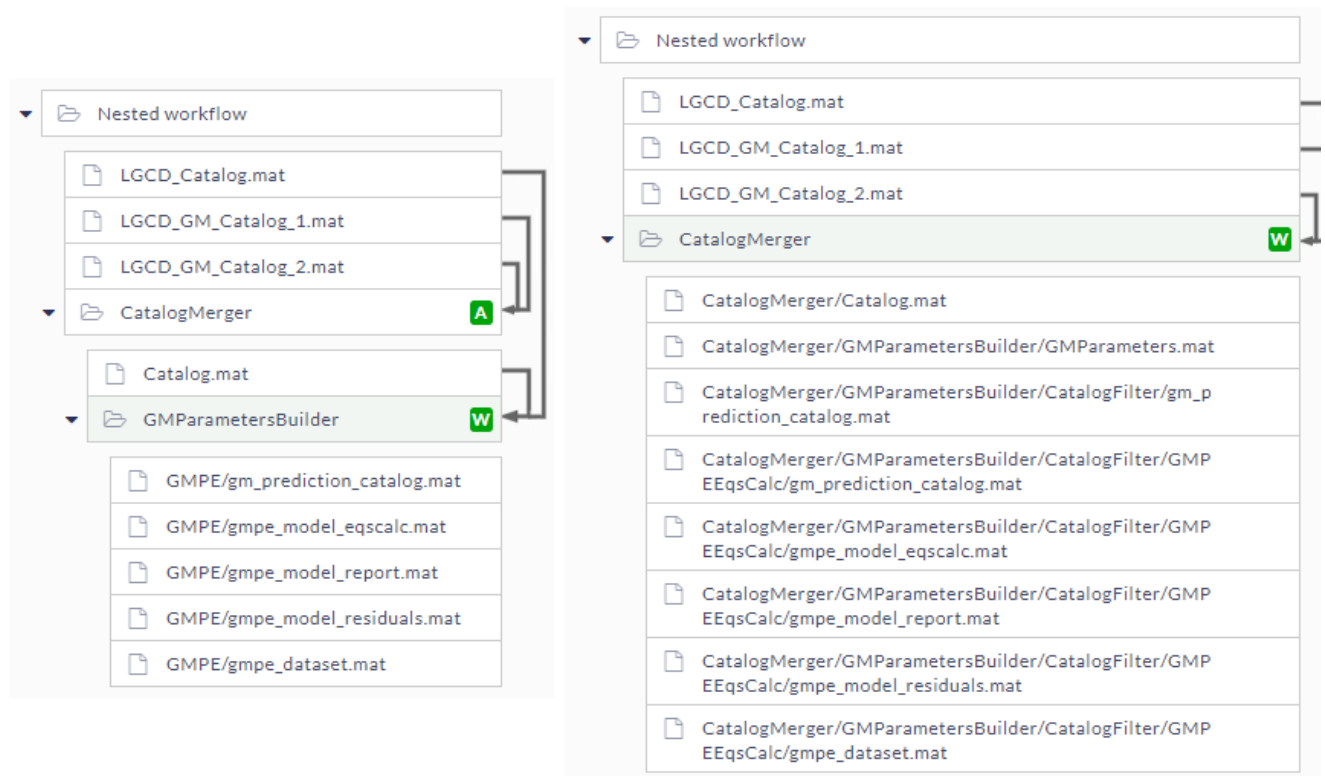


Figure 12. Workflow created on the top of another workflow and application

## Related publications

Makuch, M., M. Malawski, J. Kocot, and T. Szeplieniec (2020) *Applying workflows to scientific projects represented in file system directory tree*. In: 2020 IEEE/ACM Workflows in Support of Large-Scale Science (WORKS), pp. 25-32, <https://doi.org/10.1109/WORKS51914.2020.00009>

Makuch, M., M. Malawski, J. Kocot, and T. Szeplieniec (2022) *Model and system for scientific workflows represented in file system directory tree*. In: 2022 Future Generation Computer Systems, ISSN 0167-739X, <https://doi.org/10.1016/j.future.2022.03.023>

## Related Documents

- [Managing properties of workspace files](#)
- [Advanced Workspace Functionality](#)
- [Connecting application inputs to other applications](#)
- [Creating links in workspace](#)
- [Handling large files in workspace](#)
- [My Workspace](#)
- [Configuring application to run automatically - "Autorun"](#)