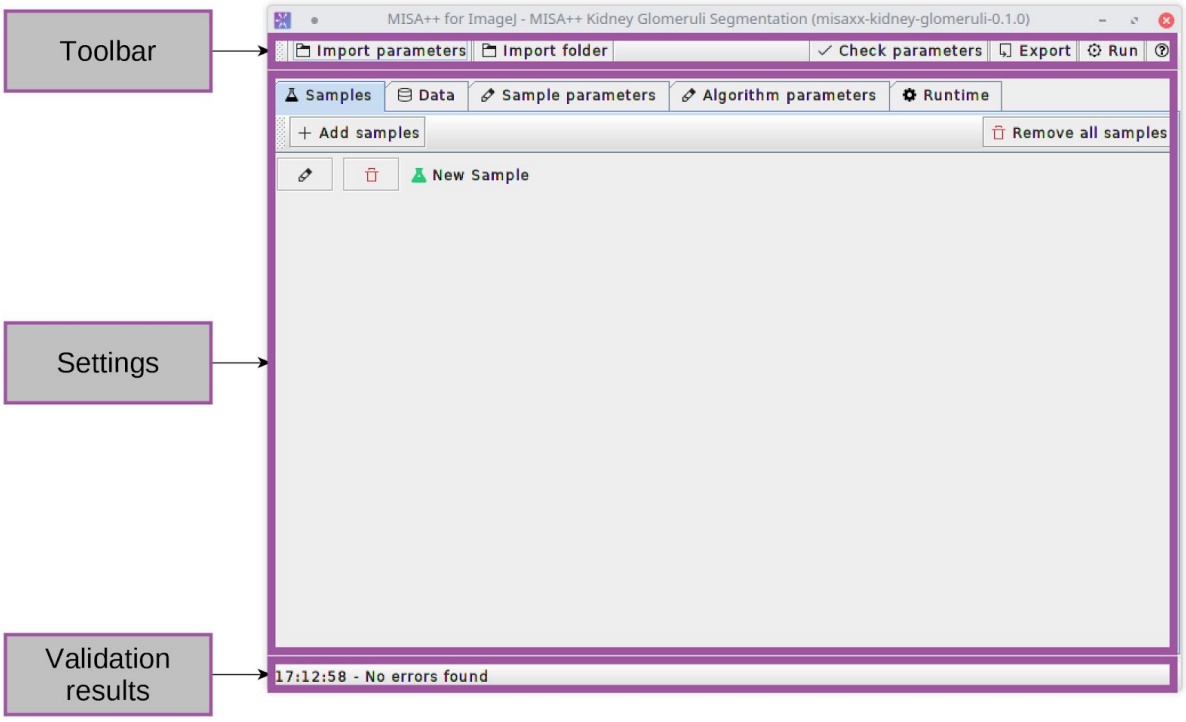
Analyzing data

MISA++ applications require a specific set of parameters and that the input data is organized in a certain way. You can either setup those conditions manually (see the MISA++ application's manual) or use the parameter editor tool that is provided by the MISA++ ImageJ plugin.
See Current application to see how to open the tool.

Overview



The user interface is divided into three sections:

1. A toolbar with global actions
2. Available settings, categorized into different tabs
3. Parameter validation results

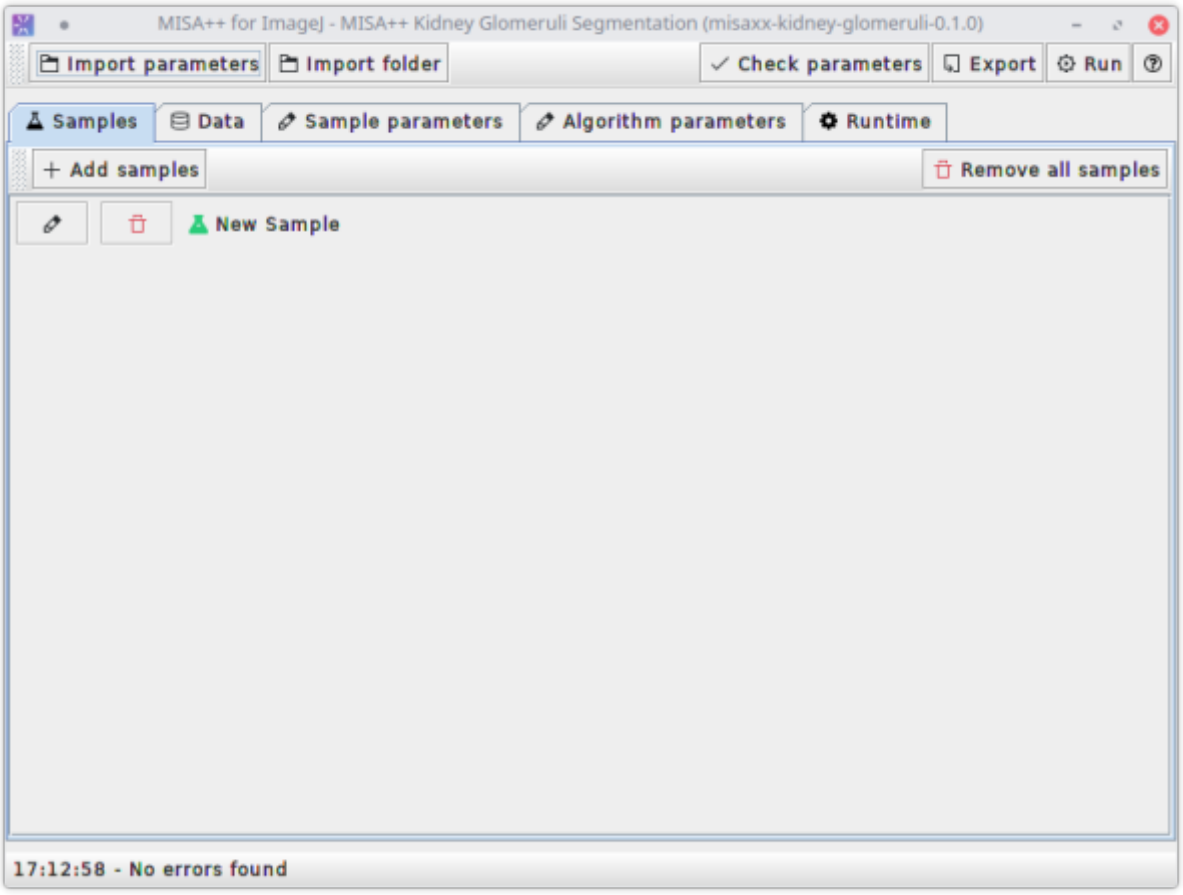
Toolbar actions

The toolbar contains actions that affect multiple setting categories. Following actions are available:

Action	Description
Import parameters	Imports a compatible parameter file. It will import: ● Samples (but not the data) ● Algorithm parameters ● Sample parameters

	• Runtime parameters
Import folder	Imports a folder that contains input data and creates new samples if necessary. The selected folder must follow a specific structure: • It must contains sub-folders that represent the samples • The sample-folders are structured as seen in the "*Data*" category (see Adding data) • If the folder contains a *parameters.json* file, it will be loaded via the "*Import parameters*" action
Check parameters	Triggers a manual check of all parameters (see Validating the current settings)
Export	Exports the current data and parameters into a ready-to-use package for other computers if the current settings are valid (see Validating the current settings). This will copy all the data into the selected folder. Please make sure that the file system has enough space. This will not copy the the MISA++ application itself.
Run	Executes the analysis on the current computer. When the data is successfully analyzed, you can directly analyze the results (see Analyzing results).
Help	Provides access to the parameter editor documentation and the documentation of the current MISA++ application (*Module documentation*)

Managing samples

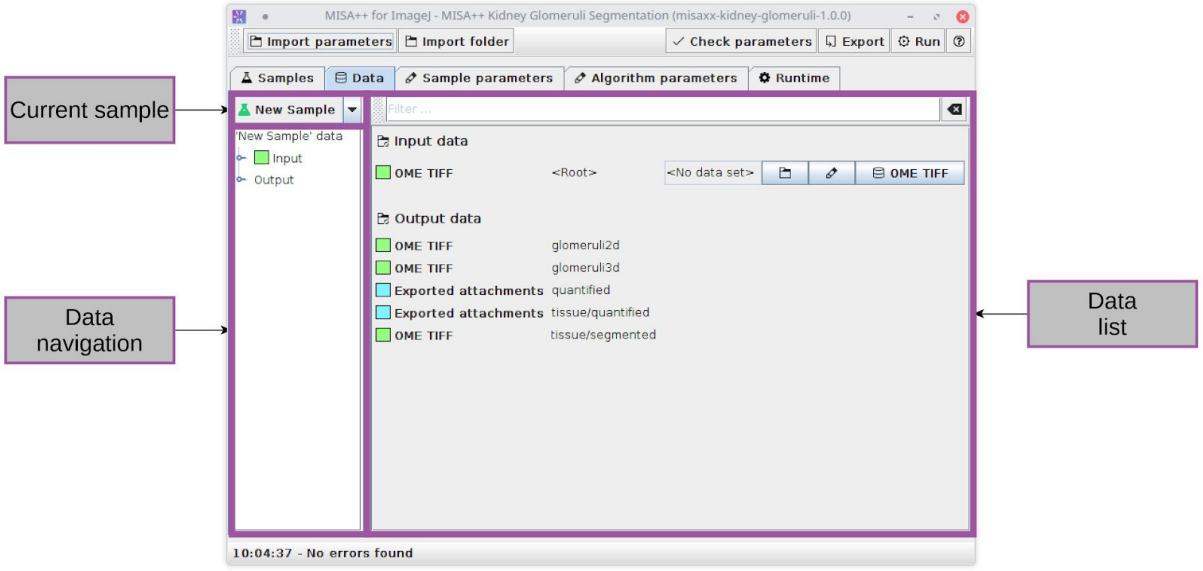


To add samples, select the “Samples” category and click the “Add samples” button. You can enter the name of one samples or add multiple lines where each line correspond to one sample.
Next to each sample, there are two available actions:

Action	**Description**
Rename sample	Allows you to rename the sample. If the name already exists or is empty, nothing will happen.
Remove sample	Removes the sample

Importing data

MISA++ applications define which input data is required and which data is generated as output. Input data must be structured in a way expected by the application. The MISA++ ImageJ plugin provides a graphical user interface for importing data.

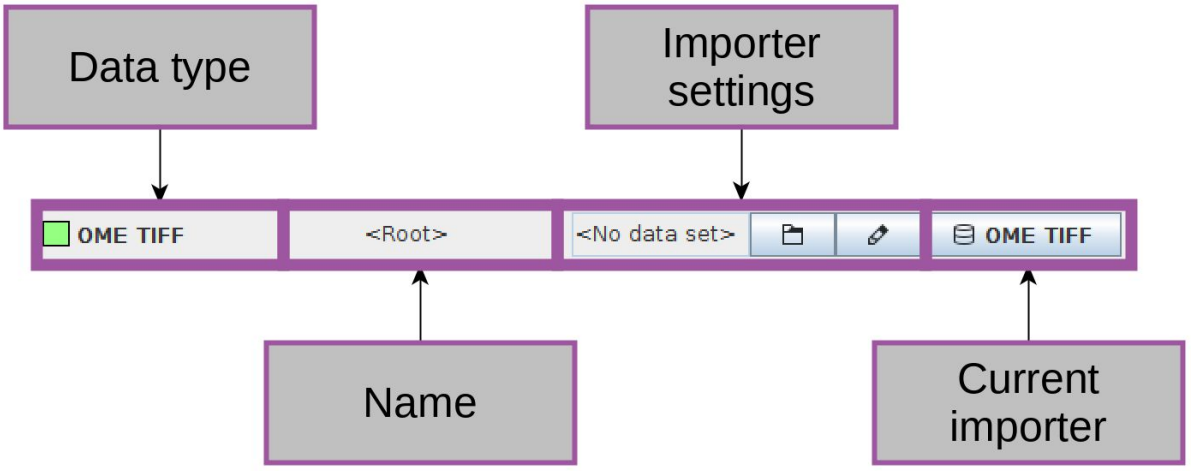


The user interface is divided into three sections:

1. The currently selected sample
2. A tree to navigate the input and output data
3. A list of input and output data

To import data into a sample, select it in the list of available samples. To refine the data list if there are a lot of entries, use the tree and the "*Filter …*" bar.

By default, the list will display input and output data. While the output data is only shown for reference, you are required to set input data via the user interface elements.
Each data entry always has the following structure:



Input data interface element	**Description**
Data type	The type of the data. **Tip**: Hover your mouse over this element to show information about the data type

Name	Name of the data. Also acts as inbternal location within the MISA++ application.
Importer settings	User interface specific to the currently selected importer.
Current Importer	The importer that provides the data to the MISA++ application.

Importers

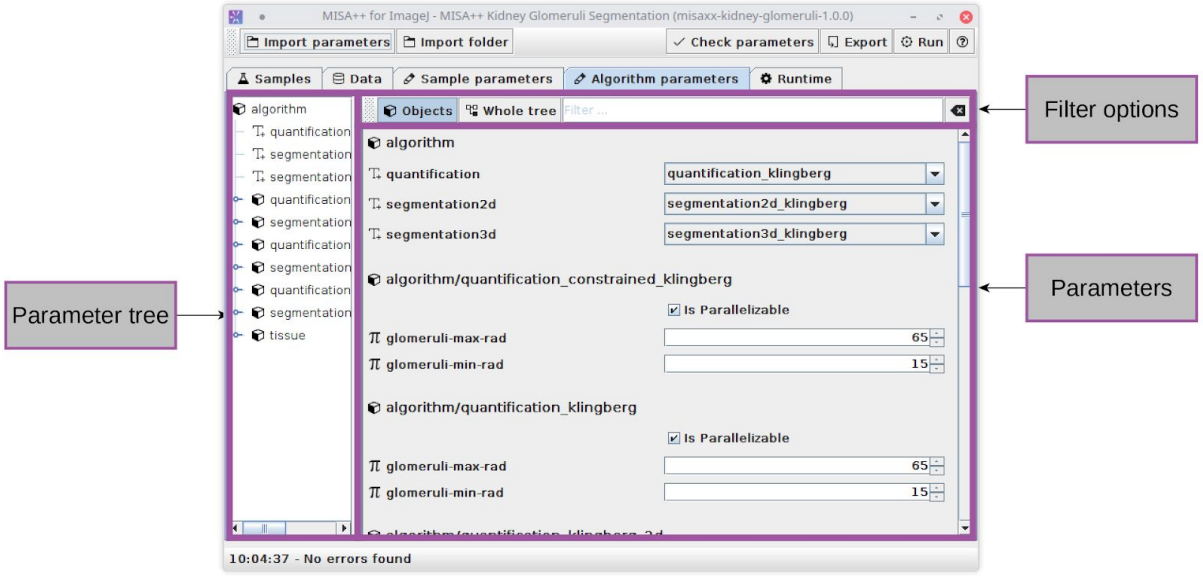
The ImageJ plugin is designed to be extendable via other plugins. This is for example needed if the MISA++ application defines data types outside of images and developers want to include support for this data type. It is also possible to integrate new types of image sources such as a link to an online database.
By design, users can select from various importers that are responsible for providing all necessary files to the MISA++ application. To change the current importer, click button Ⓓ and select it.

We included following importers by default:

Importer	**Description**
Folder link	Supports any data type (is always available) and allows you to select a folder that contains the data. Hover your mouse over the ⓘ button to see what kind of files are expected.
OME TIFF	Only available if the data type is “OME TIFF”. Click the 🗀 button to show possible image sources. If there are images loaded in ImageJ, you can import an image directly from ImageJ.

Sample and algorithm parameters

Depending on the MISA++ application there are a wide variety of sample and algorithm parameters that can be changed in the *Sample parameters and* *Algorithm parameters* categories.



The user interface is divided into three sections:

1. Parameter tree
2. View and filter options
3. Parameters

Depending on the MISA++ application, there can be many nested parameters. To navigate them, use the parameter tree. If you select an entry in the tree, the list of parameters will only show the selected value and its children.

To make navigation easier, you can hide sub-trees by deselecting *Objects* and filtering the list via the "*Filter …*" bar.
By default, the parameter list will limit the list of shown parameters. To disable this feature, select *Whole tree*.

Use the parameter list to change the parameters. Hover your mouse over the interface elements (such as the parameter name) to show documentation if available.

The *Sample parameters* category has an additional user interface element at the top of the parameter tree to select the current sample.

Runtime parameters

Runtime-parameters provide application-wide settings on the number of threads, how quantification data is written and additional settings specific to some data types.
The user interface is the same as for sample and algorithm parameters (see Sample and algorithm parameters).

Validating the current settings

Some actions such as running the MISA++ application require that all parameters are valid and all data is imported. The ImageJ plugin notifies about issues in the current settings via the status bar.

It will show “No errors found” if the parameters are valid and the error message if an issue was found. If there are multiple issues, click the error message to show a window that lists all errors.