



European Facility  
for Airborne Research

# **EGADS Lineage Graphical User Interface Documentation**

***Release 1.1.1***

**EUFAR, Olivier Henry**

**Feb 24, 2020**

# CONTENTS

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Installation</b>	<b>2</b>
2.1	Prerequisites . . . . .	2
2.2	Compatibility . . . . .	2
2.3	Installation . . . . .	2
2.4	Stand-alone . . . . .	2
2.5	Testing . . . . .	3
2.6	Options . . . . .	3
2.7	Log . . . . .	3
2.8	Update . . . . .	3
<b>3</b>	<b>Description</b>	<b>4</b>
3.1	Important information . . . . .	4
3.2	Exploring EGADS GUI . . . . .	4
3.2.1	Current GUI limitations . . . . .	4
3.2.2	The main window . . . . .	4
3.2.3	The menu bar . . . . .	5
3.2.4	The icon bar and the windows . . . . .	6
3.2.5	The central widget, example with a NetCDF file . . . . .	12
3.2.6	The toolbar . . . . .	14
3.2.7	EGADS and GUI options . . . . .	14
<b>4</b>	<b>Tutorial</b>	<b>15</b>
4.1	Important information . . . . .	15
4.2	How to open a file? . . . . .	15
4.3	How to save a file? . . . . .	16
4.4	How to close a file? . . . . .	17
4.5	How to modify the global attributes of a NetCDF file? . . . . .	18
4.5.1	From the global attributes tab . . . . .	18
4.5.2	From the global attributes window . . . . .	20
4.6	How to modify the attributes of a variable ? . . . . .	21
4.6.1	From the variable tab . . . . .	21
4.6.2	From the variable attributes window . . . . .	23
4.7	How to process a variable? . . . . .	25
4.8	How to migrate a variable from the new variable tab to the variable tab? . . . . .	28
4.9	How to delete a variable? . . . . .	28
4.10	How to create an algorithm through the GUI? . . . . .	29
4.11	How to display information about an algorithm? . . . . .	31
4.12	How to display variable data? . . . . .	32
4.13	How to plot a variable ? . . . . .	34
4.14	How to export data to an external format ? . . . . .	35
4.15	How to use the batch processing function ? . . . . .	38

---

**CHAPTER  
ONE**

---

## **INTRODUCTION**

The EGADS (EUFAR General Airborne Data-processing Software) core is a Python-based library of processing and file I/O routines designed to help analyze a wide range of airborne atmospheric science data. EGADS purpose is to provide a benchmark for airborne data-processing through its community-provided algorithms, and to act as a reference by providing guidance to researchers with an open-source design and well-documented processing routines.

The GUI (Graphical User Interface) developed for EGADS is a software based on Python and PyQt5 and designed to process/display/plot airborne atmospheric science data. The GUI purpose is to facilitate the use of EGADS for newcomers and to allow the display of data quickly with few clicks.

As the EUFAR FP7 project ended the 31st of January 2018, the development of EGADS GUI has been stopped for now. To continue to improve EGADS GUI outside the scope of EUFAR, a new branch has been created: EGADS Lineage GUI. EGADS Lineage GUI is still EGADS GUI, but compatible with Python 3 and maintained by Olivier Henry. All issues reported in EGADS GUI and EGADS Lineage GUI will be fixed in EGADS Lineage GUI only. A merging of both project will probably happen in the next EUFAR project.

---

**Note:** From now on, EGADS GUI will refer to EGADS Lineage GUI.

---

## INSTALLATION

The latest version of the EGADS GUI can be obtained from <https://github.com/EUFAR/egads-gui/tree/Lineage> and from <https://github.com/EUFAR/egads-gui/releases> for the Stand-alone version (look for the EGADS Lineage GUI STA System vX.X.X package, STA means stand-alone).

### 2.1 Prerequisites

The use of the EGADS GUI as a standard Python script requires the following packages:

- Python 3.5.4 or newer. Available at <https://www.python.org/>
- PyQt 5.11.3 or newer. Available at <https://www.riverbankcomputing.com/software/pyqt/download5>
- EGADS 1.0.0 or newer. Available at <https://pypi.python.org/pypi/egads>
- Matplotlib 3.0.3 or newer. Available at <https://pypi.python.org/pypi/matplotlib>
- Cartopy 0.17.0 or newer. Available at <https://pypi.org/project/Cartopy/>
- Simplekml 1.3.1 or newer. Available at <https://pypi.org/project/simplekml/>
- Requests 2.18.4 or newer. Optional, only for update checking. Available at <https://pypi.org/project/requests/>
- Markdown 3.1.1 or newer. Optional, only to have a nice changelog in the GUI. Available at <https://pypi.org/project/Markdown/>

### 2.2 Compatibility

The EGADS GUI has been tested on Linux and Windows. Even if it has not been tested on MacOS, it should be compatible with the OS made by Apple.

### 2.3 Installation

The EGADS GUI is actually available as a common Python script, thus it doesn't need any particular installation. To use it, the package must be downloaded and uncompressed somewhere on the hard drive, and the script executed with the usual command `python egads_gui.py` from a terminal launched in the EGADS GUI directory. This version can be executed on any version of Linux, Windows and MacOS as soon as prerequisites are installed and working.

To learn how to install EGADS, please read the EGADS documentation available at the following places: <https://github.com/EUFAR/egads/tree/Lineage/Documentation> & <https://egads.readthedocs.io/en/lineage/>

### 2.4 Stand-alone

Since version 1.0.0, a stand-alone package is available for those who wants to use the GUI without a Python installation. In that case, look for EGADS Lineage GUI STA in the release part of the repository. For Windows (from Windows 7 32), download the .msi package and launch the installation, it should be installed outside ProgramFiles to avoid issues with admin rights, then the GUI can be run by double clicking on `egads_gui.exe` or

from the shortcut in the Startup menu. A .zip package is also available for those who don't want to install it. For Linux (from Linux 4.15), download the tar.gz package somewhere on your hard drive (preferably in your home directory), extract it and run egads\_gui. The stand-alone versions for Linux and Windows have been created with PyInstaller, Windows 7 and Ubuntu 16.04.

## 2.5 Testing

No test system has been introduced at this time.

## 2.6 Options

Since version 0.10.0, an .ini file has been added to EGADS GUI to welcome few options: log level and path, automatic check for a new EGADS GUI version on GitHub, and few other options. If the file is not present in EGADS GUI directory, when importing, EGADS GUI will create it automatically with default options. The modification of the file is made through the option window of the GUI. Since version 0.13.0, the GUI can control the EGADS options. Information about each option in the Option window of the GUI is available through info buttons.

## 2.7 Log

A logging system has been integrated in the EGADS GUI since the version 0.7.0. By default, the output file is available in the GUI directory and the logging level has been set to DEBUG. Options for logging level and location are set in a config\_file `egads_gui.ini` and can be change directly through the Option window in the GUI.

Actual options to control the logging system are for now:

- `level`: the logging level ( DEBUG, INFO, WARNING, CRITICAL, ERROR ).
- `path`: the path of the file containing all logs.

The GUI has to be restarted to take into account the new path and new log level.

## 2.8 Update

Since version 0.11.0, EGADS GUI can check for an update on GitHub. The check system is launched in a separate thread and can run automatically at each startup if the option to do so has been set accordingly. To install the update, the user has to follow the install instructions found in actual document. The module Requests is optional for EGADS GUI but is mandatory to check for an update.

In the Stand-alone version, the update system can check for an update automatically at startup or manually any time, and if an update is available, it can be downloaded and installed automatically.

## DESCRIPTION

### 3.1 Important information

In the following tutorial, all pictures have been made on a Windows 10 system. On the contrary, all commands, paths and files are from a Linux Mageia 6 system.

### 3.2 Exploring EGADS GUI

The simplest way to start working with EGADS GUI is to run it from the terminal if using the Python package:

```
$ cd egads_gui/  
$ python egads_gui.py
```

Or by double clicking on the executable if using the stand-alone package (egads\_gui.exe for Windows and egads\_gui for Linux).

The EGADS GUI is easy to use and self explaining by the presence of information button (once a button is clicked, a popup will appear with text to explain the purpose of the area actually used) and tool tips.

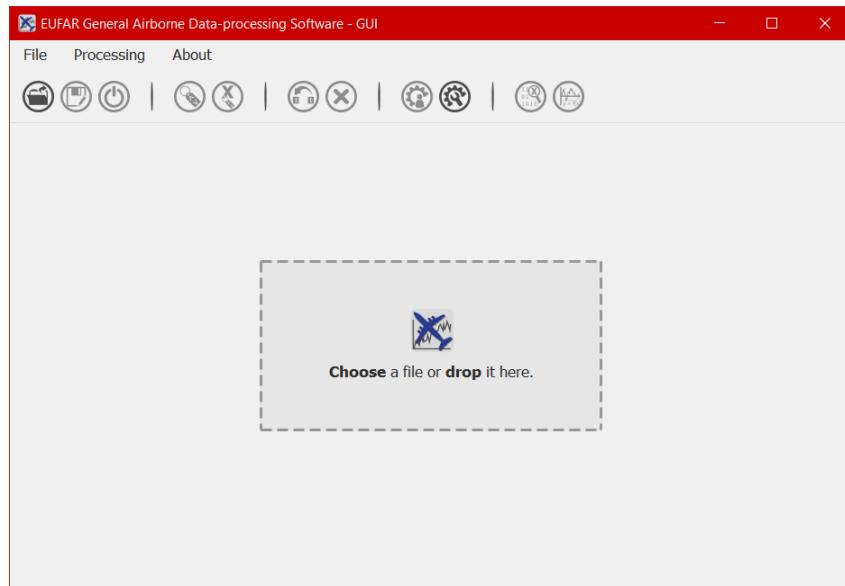
The EGADS GUI is composed of different windows, designed to display information to the user and accept interactions, and functions linking different parts of the GUI with EGADS. At the end of this section, information are also given about the option window, used to control EGADS and the GUI options.

#### 3.2.1 Current GUI limitations

Actually, the GUI has few limitations based on the development time and the number of person working on this project. First the file formats handled by the GUI. Even if EGADS can handle text files (raw, ASCII and csv), the GUI can handle only NetCDF and NASA Ames file format (FFI 1001) at this time.

#### 3.2.2 The main window

The main window of the GUI is what the user will see first once he has launched the software.



The main window is composed of four parts, from top to bottom:

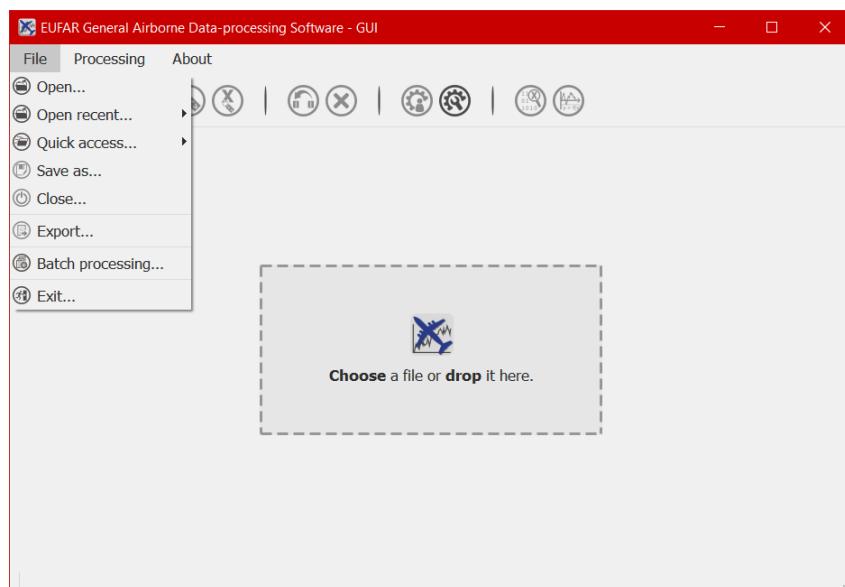
- the first one is a menu bar with three menus - File, Processing and About - allowing the user to open/save a file, launch a processing on more than one file, quit the software, display the different algorithms imbedded in EGADS, and display information about the software.
- the second one is a bar containing 12 icons and giving access to different kind of functions and windows as the plot window or the metadata window.
- the third one is a central widget whose purpose is to welcome the interface dedicated to NetCDF, NASA Ames and text files.
- the last one is a footer, used to display few information about the file actually loaded, like the name, the weight and the file conventions.

### **3.2.3 The menu bar**

The menu bar of EGADS GUI is composed of three submenus: File, Processing and About

#### **File**

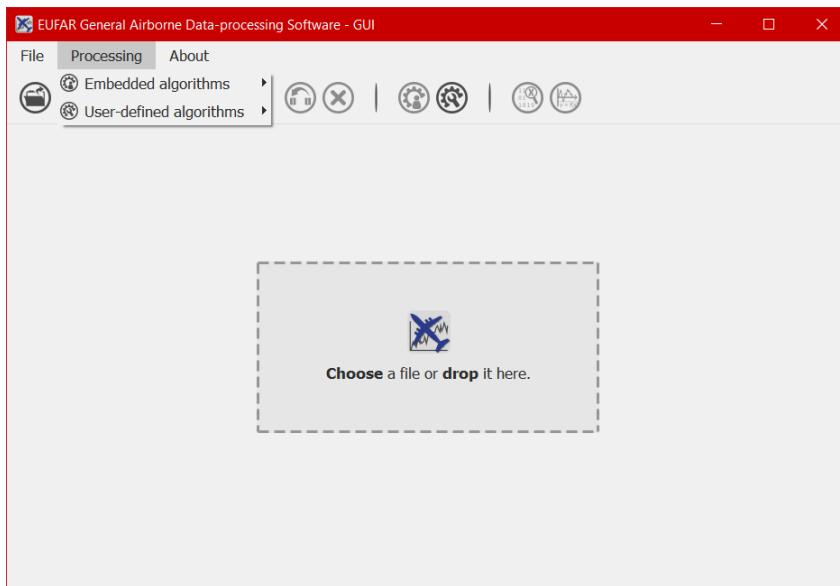
File is a submenu of the menu bar.



From here, it is possible to Open, Save As and Close a file. The Export allow the user to save data to KML/KMZ file format for Google Earth. If a user wants to launch the same processing on a certain amount of files, he has access to Batch processing. And to quit the software, the job is done by clicking on Quit.

### Processing

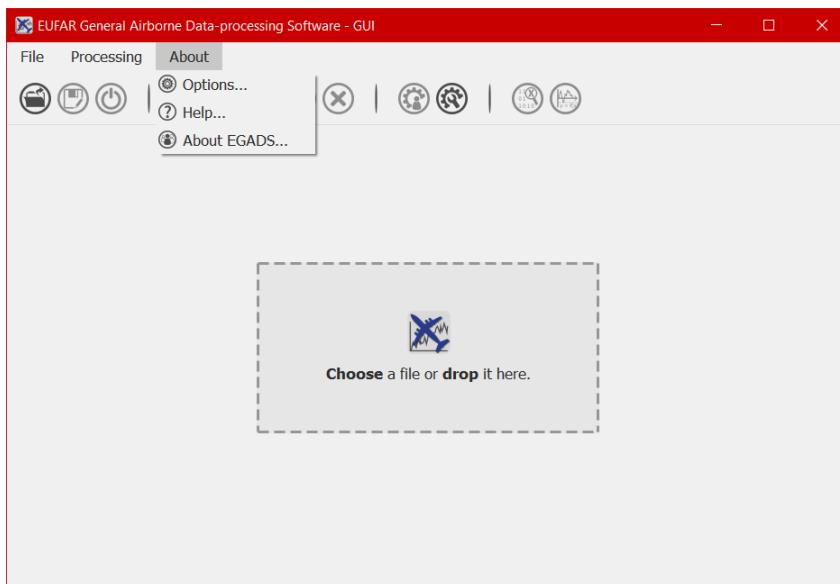
Processing is a submenu of the menu bar.



Its purpose is to give information about all algorithms embedded in EGADS. Navigate into the submenus and click on an algorithm to show a window displaying information about the selected algorithm. It is not possible to modify the algorithm from that window yet.

### About

About is a submenu of the menu bar.



The user will find here information about the software, options and link to the documentation.

#### 3.2.4 The icon bar and the windows

12 icons are embedded in the icon bar, 6 of them give access to the GUI functions and 6 of them are directly linked to sub windows.

### **The function icons**

The three first icons of the icon bar give access to few basic functions to control a file:

-  : To open a file
-  : To save a file with a different name
-  : To close a file

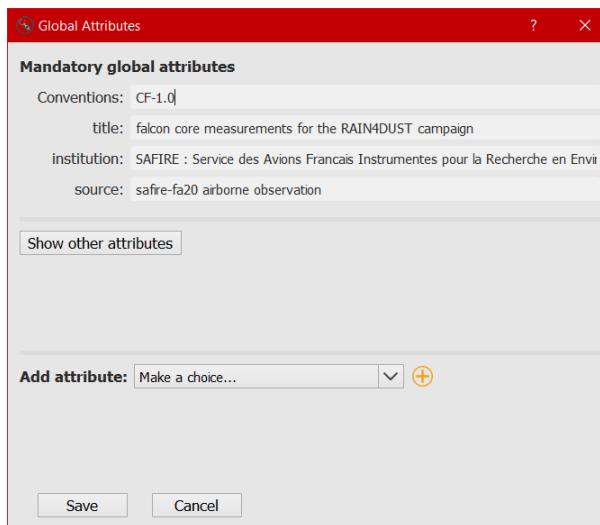
### **The data icons**

The sixth and seventh icons are here to manipulate data:

-  : In EGADS GUI, once a variable is processed through an algorithm, the result is a new variable displayed in the New variables tab of the central widget. In this tab, a variable can't be saved in a file, only variables in the Variable tab are saved. The user has to migrate the new variable to the Variables tab. And he can do that with this function.
-  : The purpose of this function is to delete a variable. If the file is saved after a deletion, obviously the deleted variable won't appear in the file anymore.

### **The global attributes window**

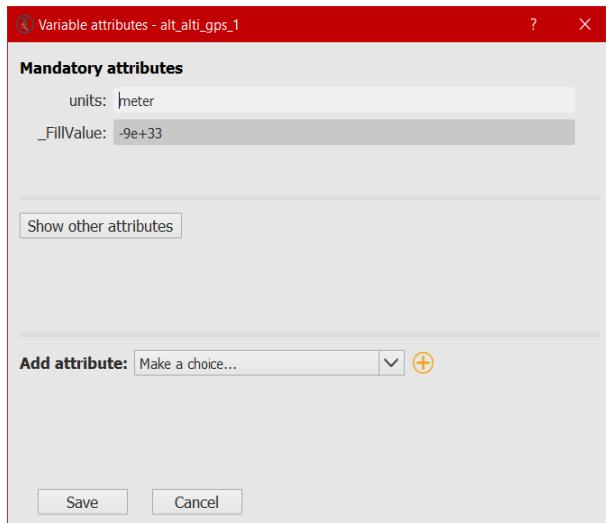
The user has to click on the icon  to open the global attributes window.



All global attributes embedded in a NASA Ames or NetCDF files are loaded in this window. The user has the possibility to modify, create and delete them. As EGADS GUI follows the EUFAR Standards & Protocols recommendations, few global attributes are thus mandatory and displayed above others. They can be modified, but they can't be deleted.

### **The variable attributes window**

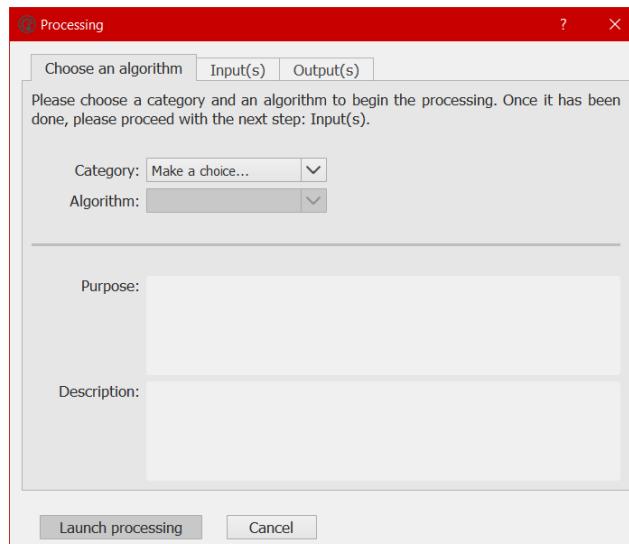
Accessible by clicking on the icon , the user has to select first a variable from the Variables tab of the main window.



All attributes linked to the selected variable, coming from a NetCDF or a NASA Ames file, are loaded in this window. As for the global attributes window, the user can create/modify/delete attributes. Following the EUFAR Standards & Protocols recommendations, two attributes can't be deleted and modified: `units` and `_FillValue`.

### The processing window

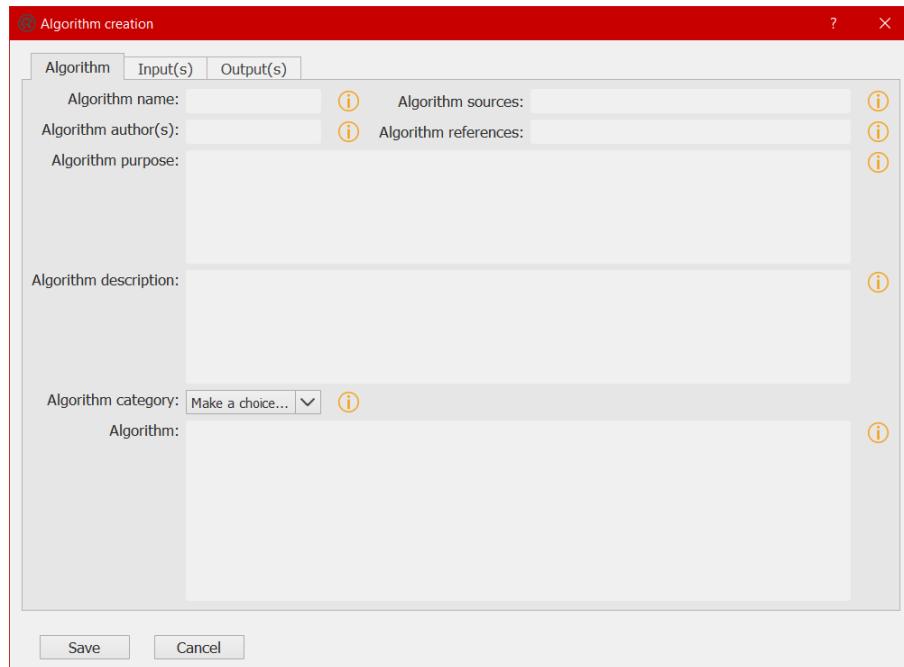
Accessible by clicking on the icon a variable can be processed with an algorithm already embedded in EGADS or created by a user.



Here the user has the possibility to select an algorithm from the EGADS algorithm list, select one or more variables from the opened file, add factors or numbers, name the output and launch the process. Information about the algorithm is displayed in the first tab. The second tab displays information coming from the input section of each algorithm, info buttons are here to give details about each input. The third tab is dedicated to the output(s).

### The algorithm creation window

One of the main goals of EGADS is to let the user creates his own algorithms. Accessible by clicking on the icon , this window is an automated system to create algorithm from the user inputs.



The window is composed of three tabs. The first tab is dedicated to the metadata of the algorithm and, the most important, to the mathematic formula. The second tab is dedicated to the input(s). The user will fill the different text boxes to prepare the input variable(s). The last tab is dedicated to the output(s). As the window follows the EGADS convention, few tips have to be used here and will be explained later. Once the user click on **Save**, an automated task will check every item in the window, in particular units, and will display a warning popup if something went wrong. The algorithm is then saved in the `user` sub-folder of the EGADS/Algorithms folder. For help purpose, few info buttons are present. The system doesn't test the algorithm created by the user, it is expected to work, and the user has to give his greatest attention when writing the algorithm using Python 3. If a complex algorithm needs to be written, the use of a text editor and the manual declaration of the algorithm in EGADS are strongly encouraged.

---

**Note:** English and normal characters must be used to write every part of the algorithm. Special characters can't be used and it will lead to a crash if the GUI tries to save a file with special characters.

---

**Note:** For more details on how to create algorithms and the purpose of each field, please take a look in the EGADS Lineage Documentation, chapter ALGORITHM DEVELOPMENT.

---

### The data display window

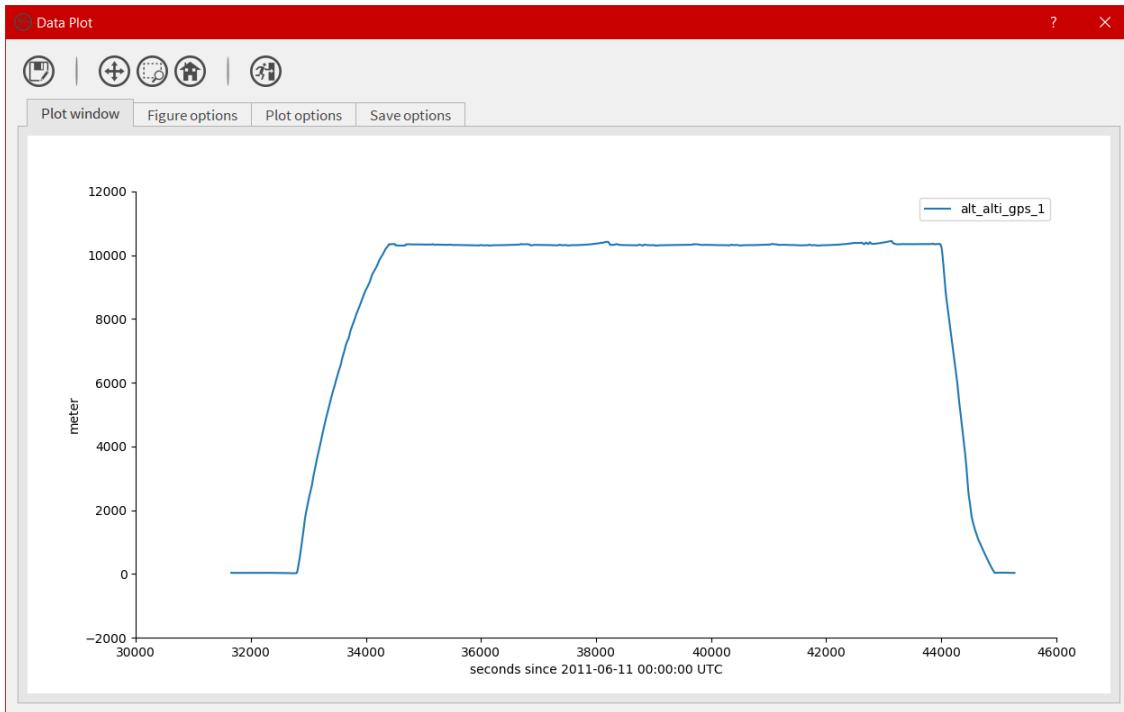
Accessible by clicking on the icon , the user has to select first a variable from the Variables tab of the main window.



Here the user can take a look directly in the variable.

### The plot window

Accessible by clicking on the icon  or by the right click menu, the user has to select a variable to enable the window. The purpose of that window is to give the user the possibility to plot all kind of data and to save the corresponding figure. It is possible to plot multiple variables by selecting them in the list using ctrl + left click.



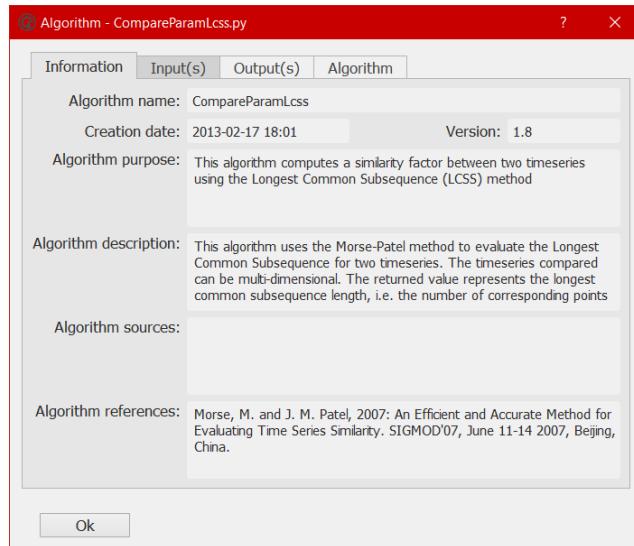
The plot window is composed of an icon bar including six icons and four different tabs:

-  : To save a figure
-  : To move the view in the figure
-  : To zoom on a selection
-  : To reset the view
-  : To quit the plot window
- The first tab is dedicated to the plot area, the software will create the figure automatically with default options.
- In the second and third tab, the software gives the possibility to the user to change few options for each curve or for the whole figure.
- The last tab is only dedicated to the options involved when a figure is saved.

The design of the plot window is based on the access to as many options as possible to let the user modify the figure to his heart content, quickly and easily. Obviously, if the user wants to create complex figures, it is strongly encouraged to use EGADS and Matplotlib from script.

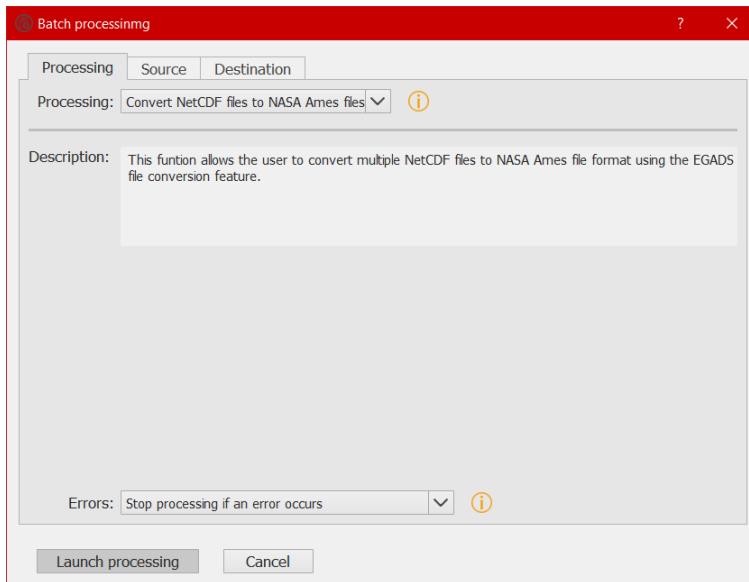
### **The algorithm information window**

From the Processing menu, the user has the possibility to display information about each algorithm embedded in EGADS or created by himself.



### **The batch processing window**

For those who want to process more than one file at a time, they can use the batch processing function from the File menu.



### **The export window**

For those who want to export their data into another external format, they can use the export function from the File menu. At this time, only KML/KMZ file format for Google Earth (data along a path, generally time series) is available. The export possibilities will expand in the future.

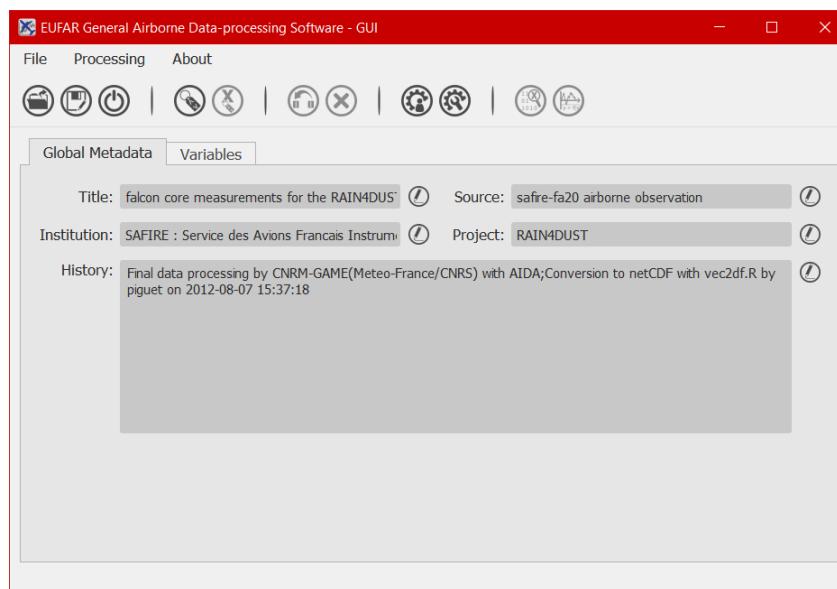


### 3.2.5 The central widget, example with a NetCDF file

Once a NetCDF or NASA Ames file has been opened in the GUI, a new interface object is displayed. It's a Tab Widget composed of three tabs.

#### The global attributes tab

The purpose of the first tab is to display the main global attributes of a NetCDF or NASA Ames. Concerning NetCDF, the tab is following the EUFAR Standards & Protocols NetCDF convention and few attributes are automatically displayed. Important information about the dataset is usually recorded in the global attributes.



The edit icon (a pencil inside a circle) for each field gives the user the possibility to modify each global attribute from the tab view. Once an attribute has been modified, the file has to be saved to keep the new attribute. Finally an object in the lower part of the central widget is here to give information about the compatibility of the file, if it is a NetCDF one, with the official EUFAR Standards and Protocols NetCDF convention.

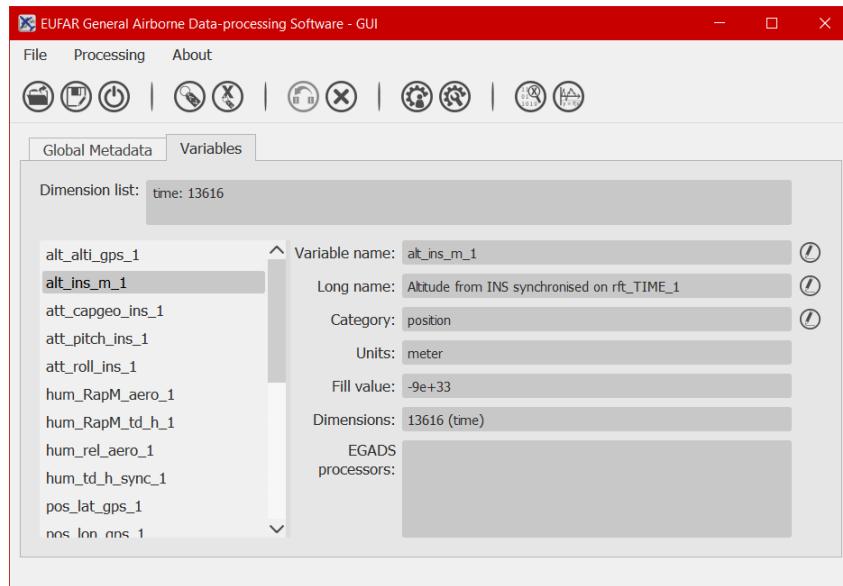
---

**Note:** Once the user has clicked on the (edit icon), and before confirming its modification by clicking on the (checkmark icon), it is possible to cancel the modification by doing a right click on the same button. It will close the edit mode and replace the new text by the old one.

---

#### The variables tab

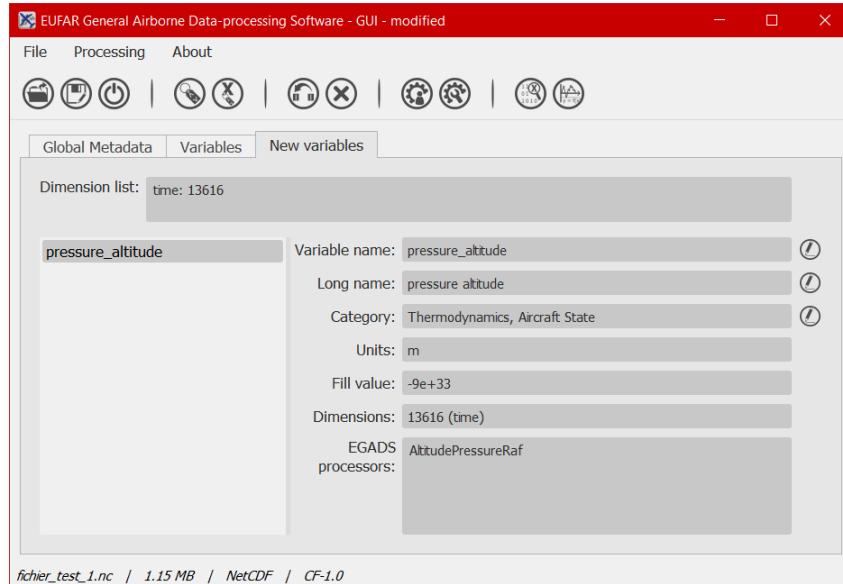
The second tab is dedicated to variables and their attributes.



A list of all variables included in the NetCDF/NASA Ames file is displayed in the left side, ordered alphabetically. If a user clicks on a variable, attributes will be displayed on the right side of the tab. As for the global attributes tab, an edit icon is here to let the user modify directly the variable attributes from the tab. Right clicks are also registered to cancel a modification in edit mode. Few of them are not intended to be modify, like the units or the non value, consequently a grey colour is superimposed on them to inform the user.

### The new variables tab

The purpose of the third tab, not visible by default, is to welcome newly-created variables, to avoid wrong manipulation and for the sake of convenience. Once a variable is created, it will appear in this new tab. If the user wants to save the new variable(s), he must migrate it/them in the Variables tab.



This tab behaves completely as the Variables tab.

---

**Note:** Variables created from algorithms or by the user have to be migrated from the new variables tab to the variables tab if the user wants to save them. Variables still in the new variables tab won't be saved.

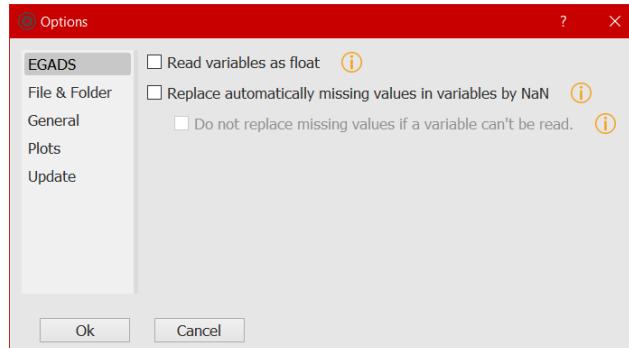
---

### 3.2.6 The toolbar

The purpose of the toolbar is to display information about the file actually loaded, like the name, the size and the conventions, and messages to confirm the actions of the user, like the modification of metadata or the creation of variables. Since version 1.0.0, messages are displayed for few seconds about the actions performed and confirmed by the user: modification of metadata, deletion of variables, ...

### 3.2.7 EGADS and GUI options

The options of EGADS and its GUI are controloed through the option window, in the About menu.



In the Log options section, the user will find options to control the logging system: the log level and the path where to save the log file. In the EGADS options section, the user can control options usually set in EGADS. Please refer to the documentation of EGADS for more details about those options.

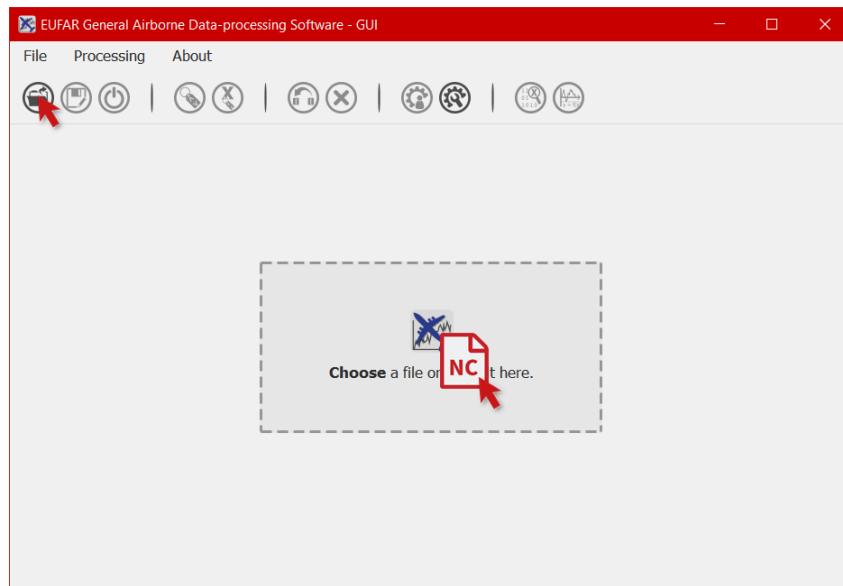
## TUTORIAL

### 4.1 Important information

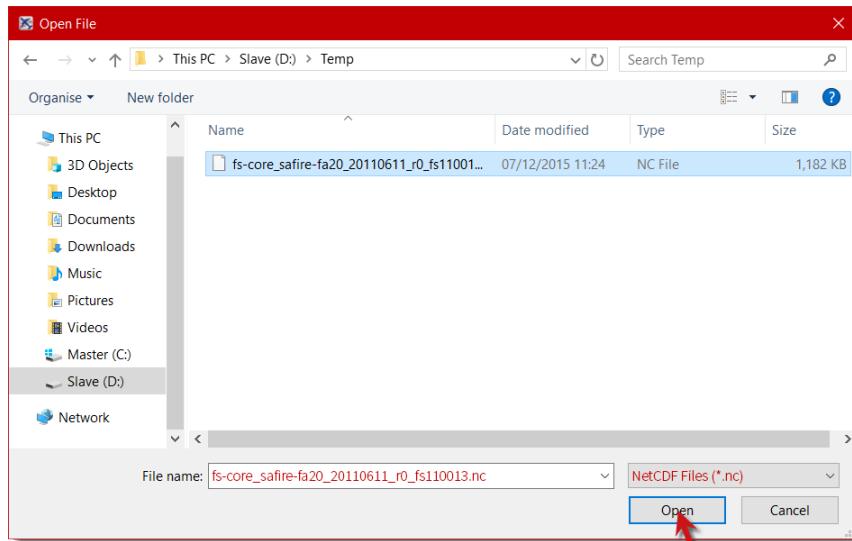
All tutorials proposed here are produced using a NetCDF file as a reference. Apart from the fact that metadata are different, there is almost no difference concerning the interface if using a NASA Ames file. Once the GUI is ready to handle raw, csv and text files, the corresponding tutorials will be added here if certain differences exist compared to NetCDF handling.

### 4.2 How to open a file?

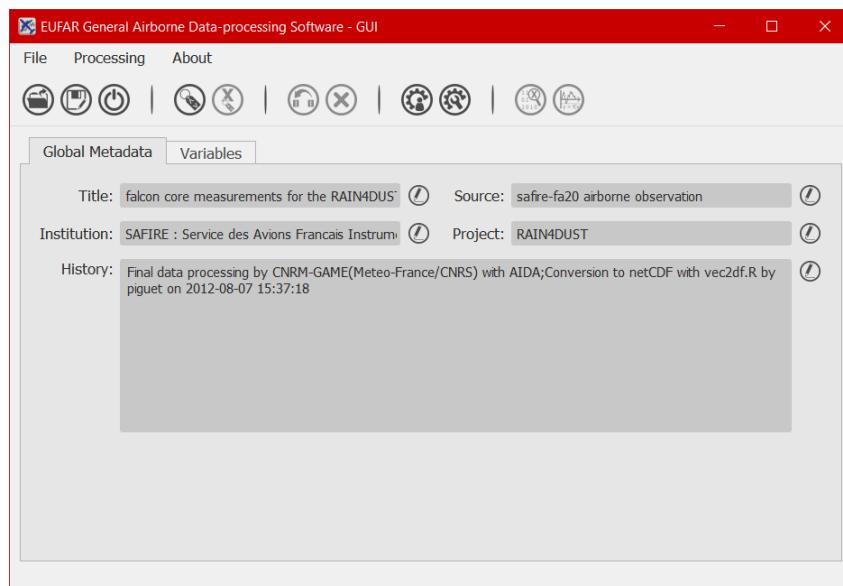
- To open a file, a NetCDF file for example, the easiest way is to drag and drop the file in the dedicated area of the GUI. Or there is still the possibility to use the open  button, from the tool bar or from the File menu.



- Then apply the filter **NetCDF Files (\*.nc)**, select the NetCDF file you want to open in the list, and click on Open.

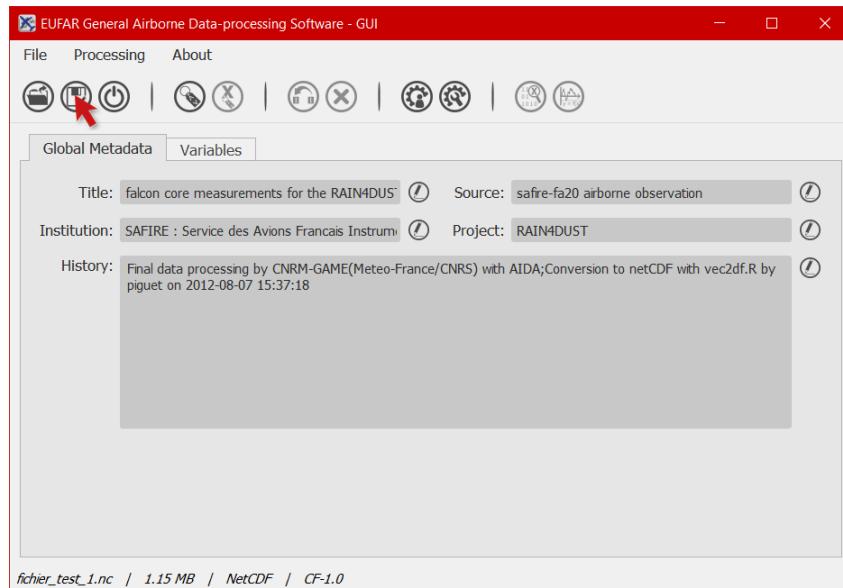


- In any way, if the file is loaded as it should, the interface specific to NASA Ames/NetCDF files should be displayed without any warning.



## 4.3 How to save a file?

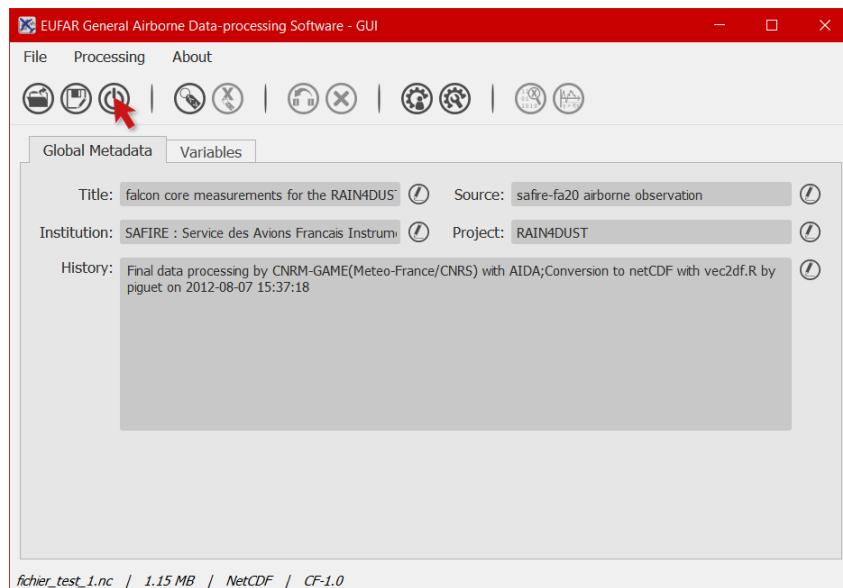
- To save a file, the user has to click on . It's the usual way to save a file with a new name, or to convert a file from a format to another format.



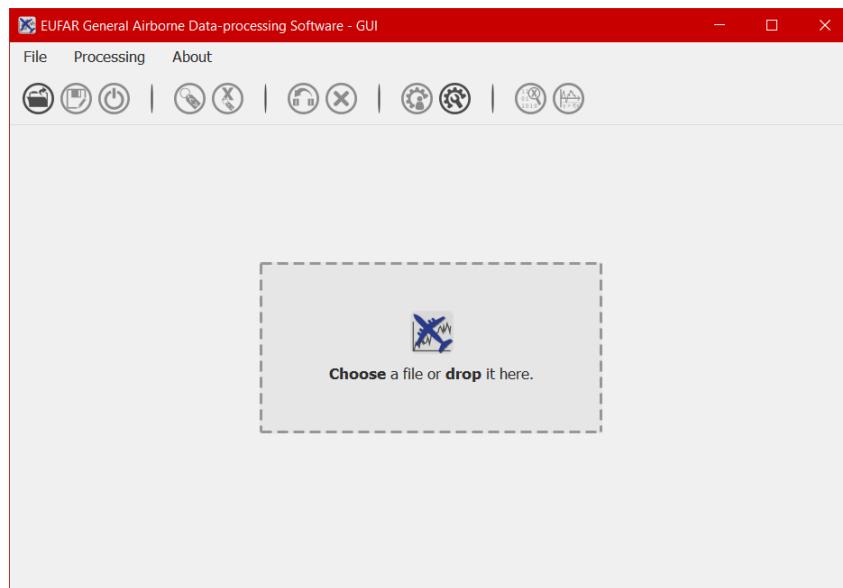
- When Saving As is selected, just enter the name of the new file and the format, then click Save to confirm the action.

## 4.4 How to close a file?

- Once a file has been opened, it is possible to close it by clicking on .



- As a consequence, the graphical interface should be cleaned.

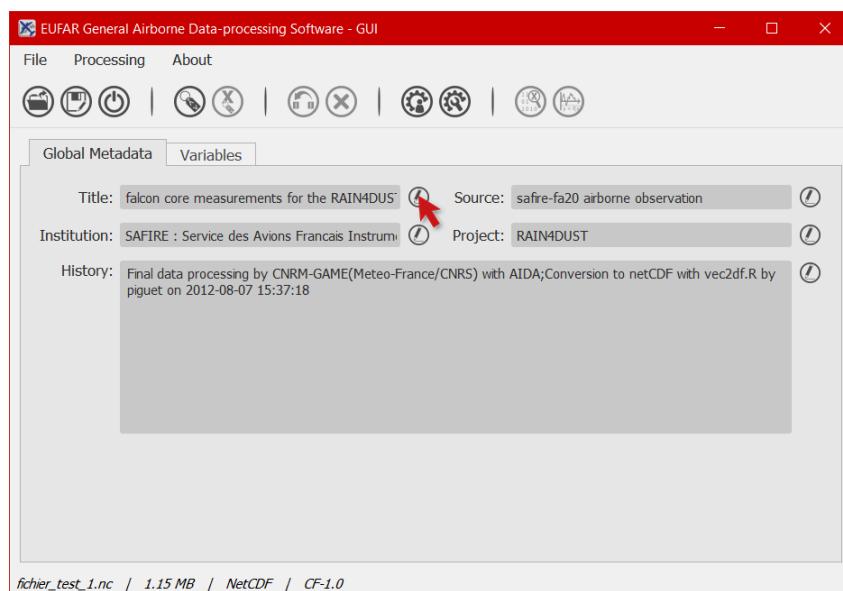


## 4.5 How to modify the global attributes of a NetCDF file?

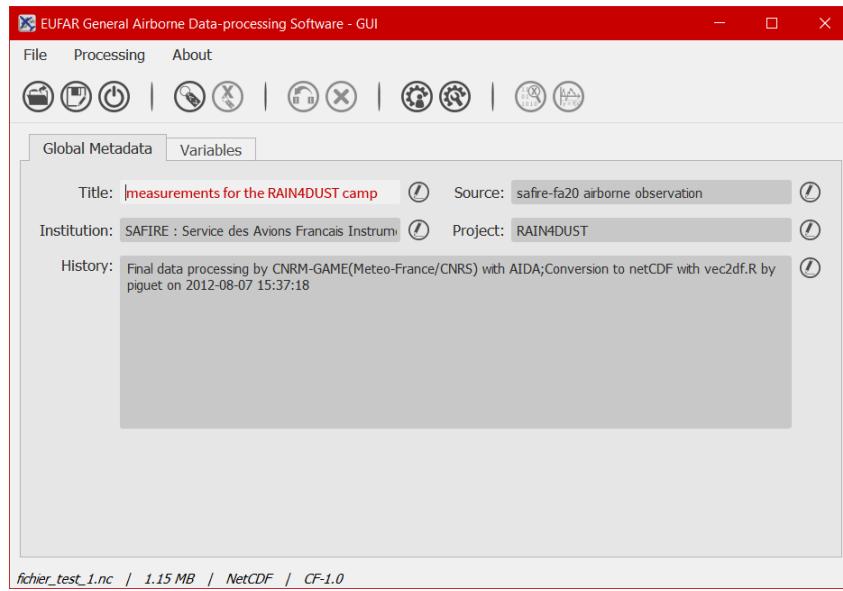
EGADS GUI proposes two ways to modify a global attribute: from the global attributes tab and from the global attributes window. The global attributes tab only shows the most important attributes, in agreement with the EUFAR Standards & Protocols NetCDF convention. The second solution is the only one if the user wants to display other attributes and to create and/or delete global attributes.

### 4.5.1 From the global attributes tab

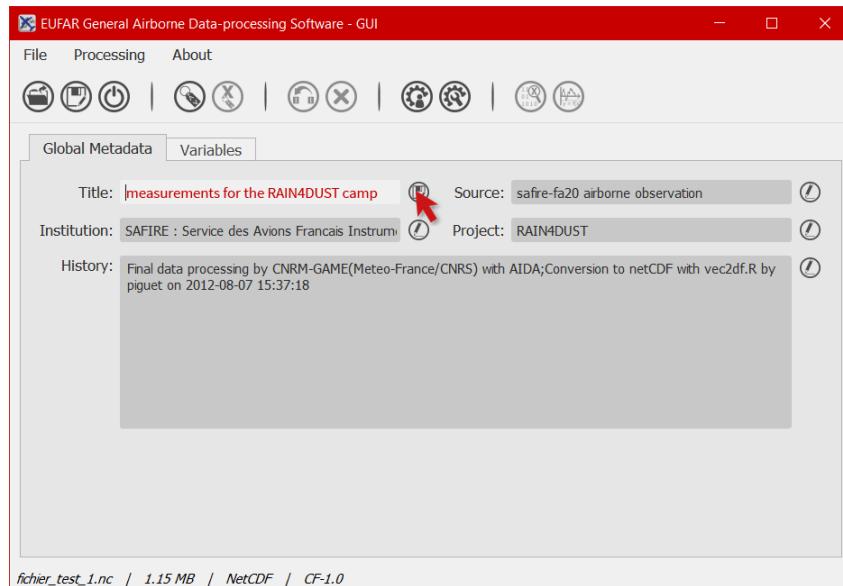
- To unlock a field and edit the associated attribute, click on . The icon is replaced by a icon to confirm that you are in edit mode.



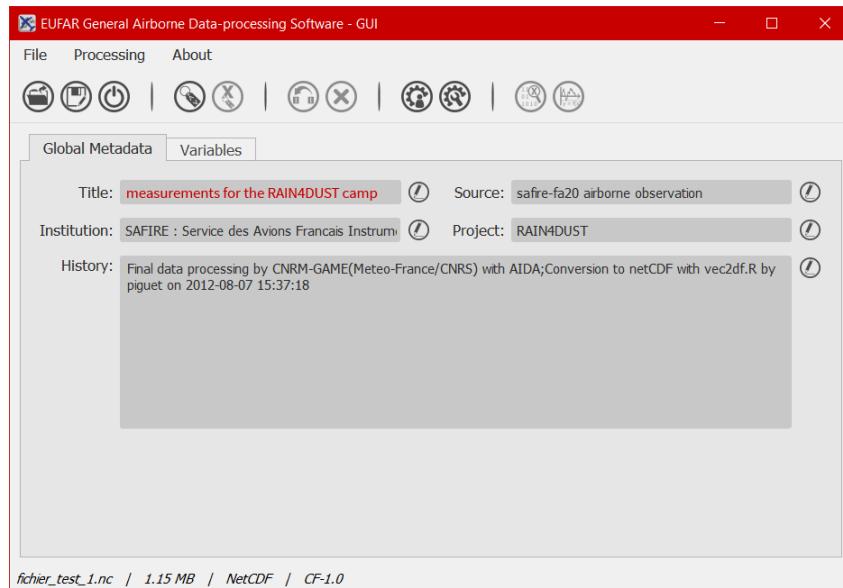
- Then, modify the attribute as you wish.



- And click on to confirm the modification.

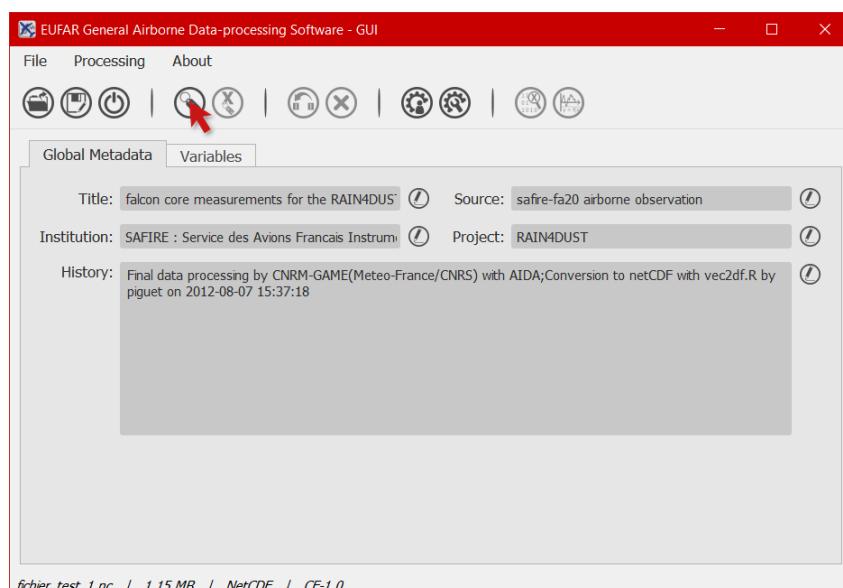


- The software should display the modified attribute and the word **modified** in the window title.

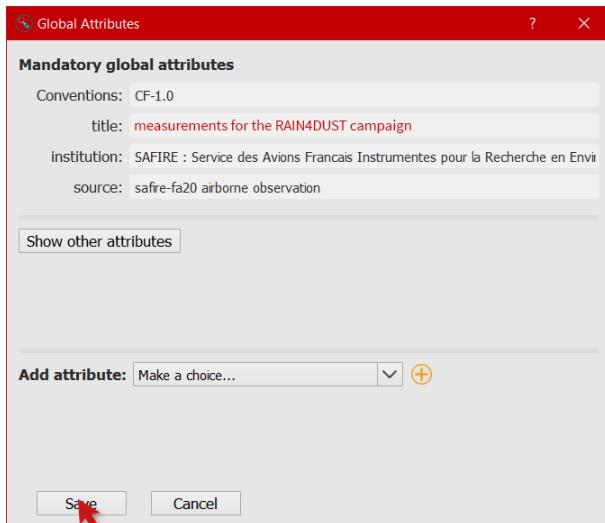


#### 4.5.2 From the global attributes window

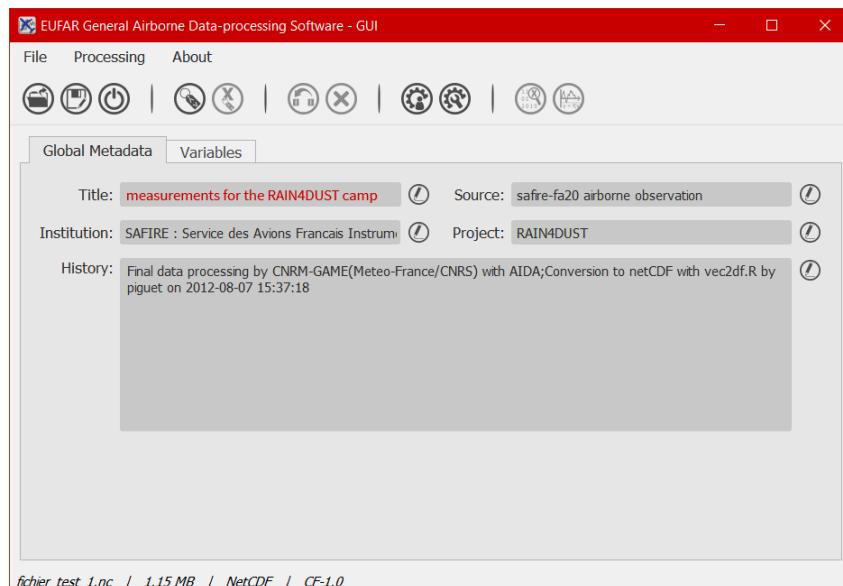
- Click on to open the global attributes window.



- Here you have the possibility to create, modify and delete global attributes. Click on Show other attributes to display other attributes not showed in the window. To modify a global attribute, just click in the associated field and enter your text. Then click on Save to confirm your modification.



- The software should display the modified attribute and the word **modified** in the window title.

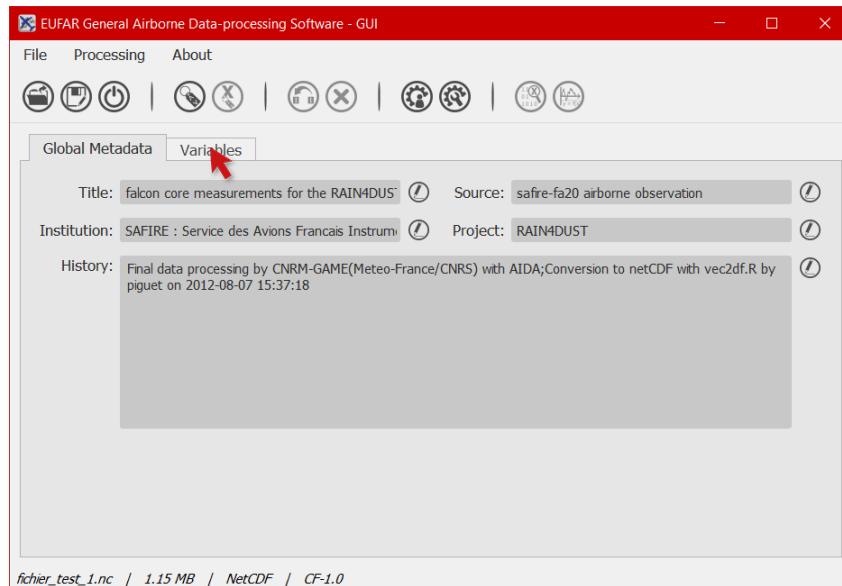


## 4.6 How to modify the attributes of a variable ?

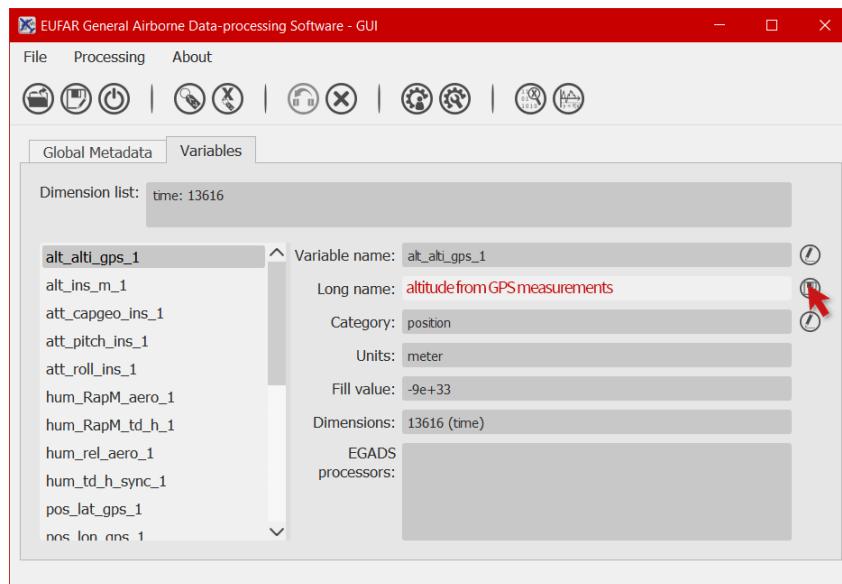
EGADS GUI propose two ways to modify a variable attribute: from the variable attributes tab and from the variable attributes window. The variable attributes tab only shows the most important attributes, in agreement with the EUFAR Standards & Protocols NetCDF convention. The second solution is the only one if the user wants to display other attributes and to create and/or delete variable attributes.

### 4.6.1 From the variable tab

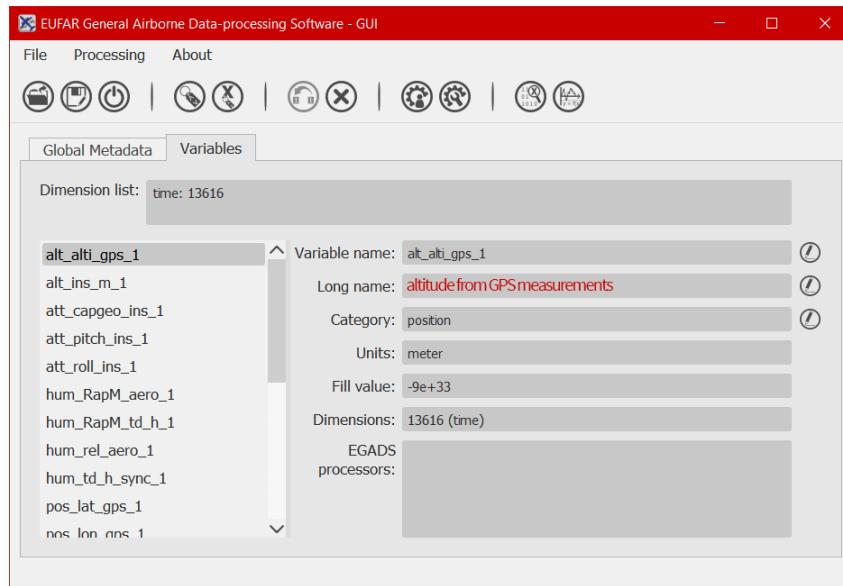
- Select the Variables tab.



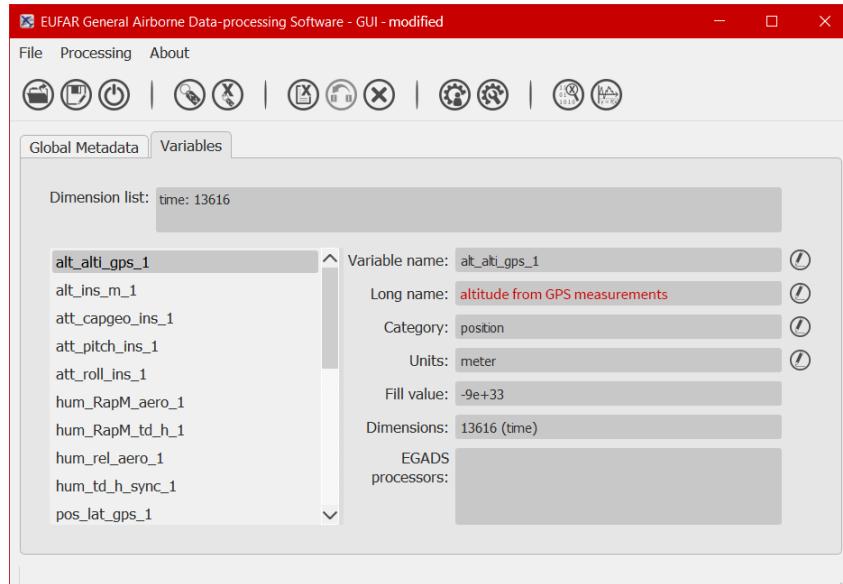
- Select a variable in the left list and click on to unlock the associated field of the attribute you want to modify. The icon is replaced by a icon to confirm that you are in edit mode.



- Enter your text and click on to confirm your modification.

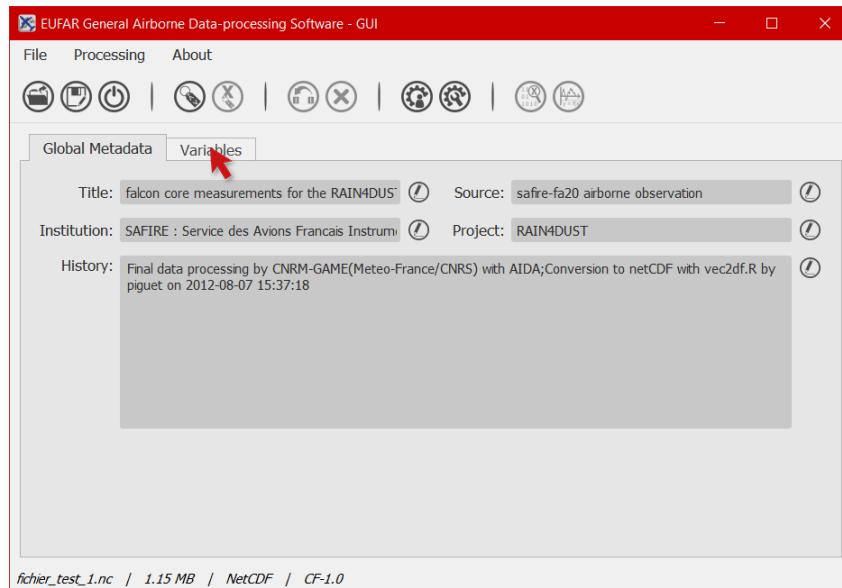


- The software should display the modified attribute and the word **modified** in the window title.

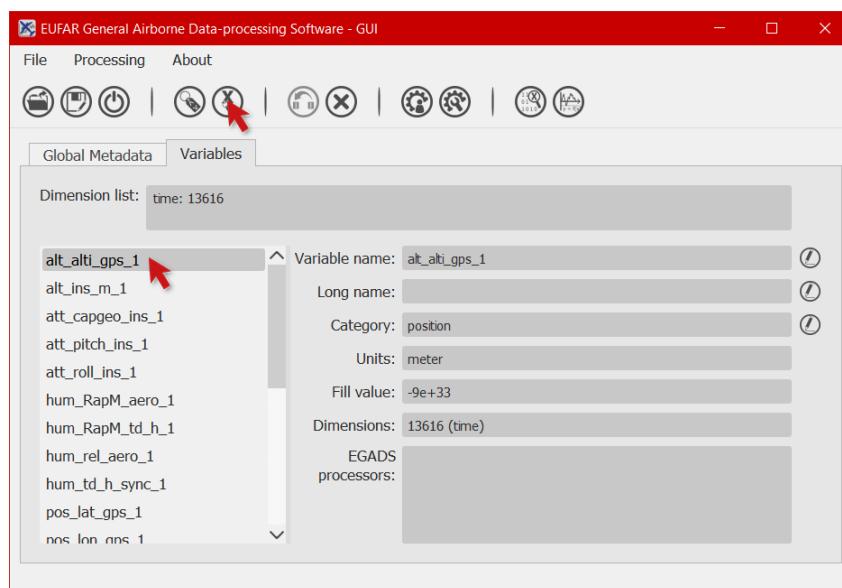


#### **4.6.2 From the variable attributes window**

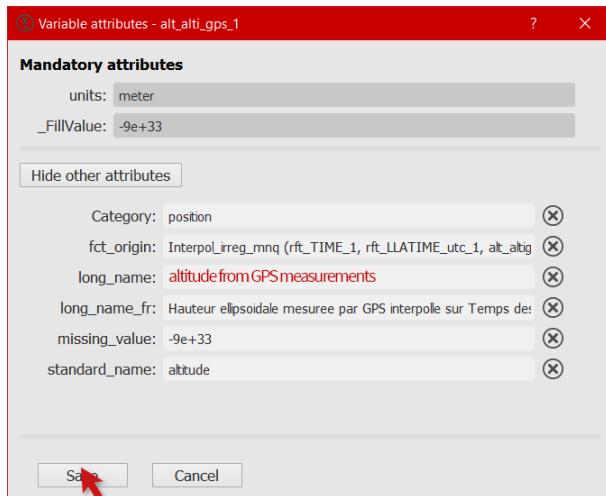
- Select the Variables tab.



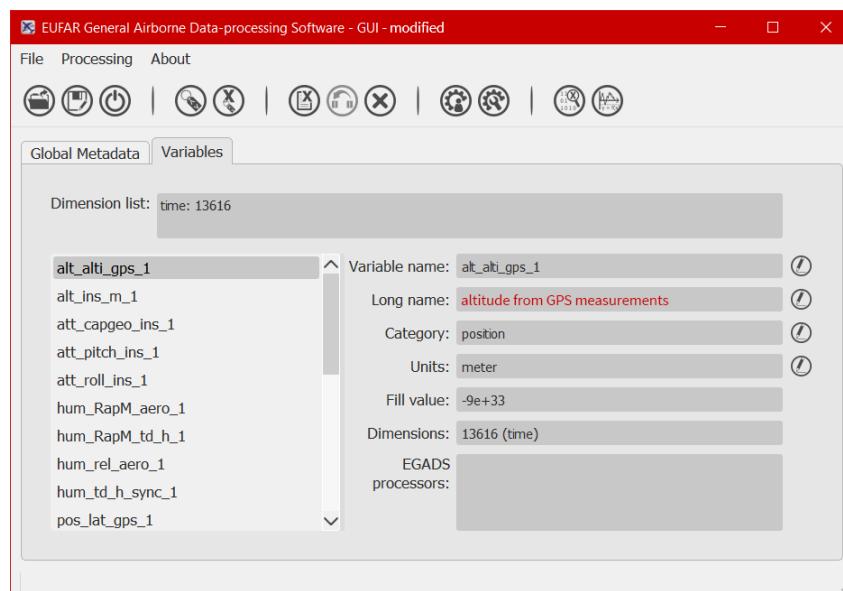
- Select a variable and click on to open the global attributes window.



- Here you have the possibility to create, modify and delete variable attributes. Click on Show other attributes to display other attributes not showed in the window. To modify a variable attribute, just click in the associated field and enter your text. Then click on Save to confirm your modification.



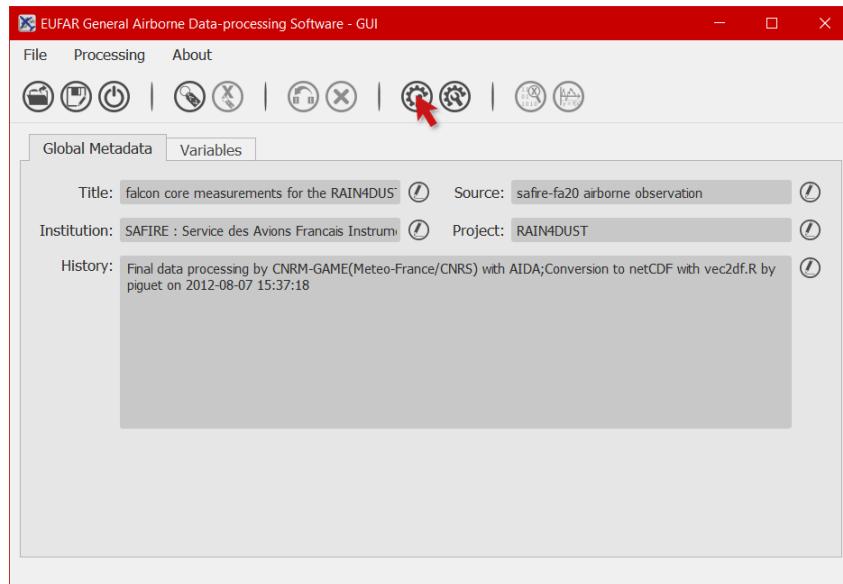
- The software should display the modified attribute and the word **modified** in the window title.



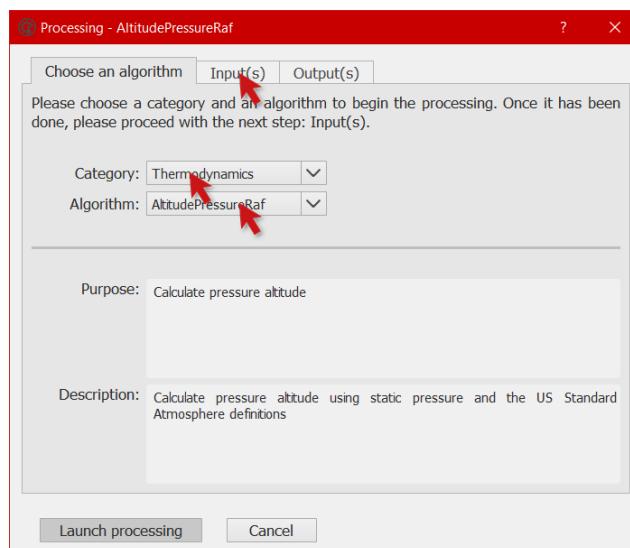
## **4.7 How to process a variable?**

EGADS GUI gives the user the possibility to use and apply algorithm embedded in EGADS. Because of the limitation of the GUI, actually, it is only possible to execute one algorithm at a time. All processings are done through the processing window.

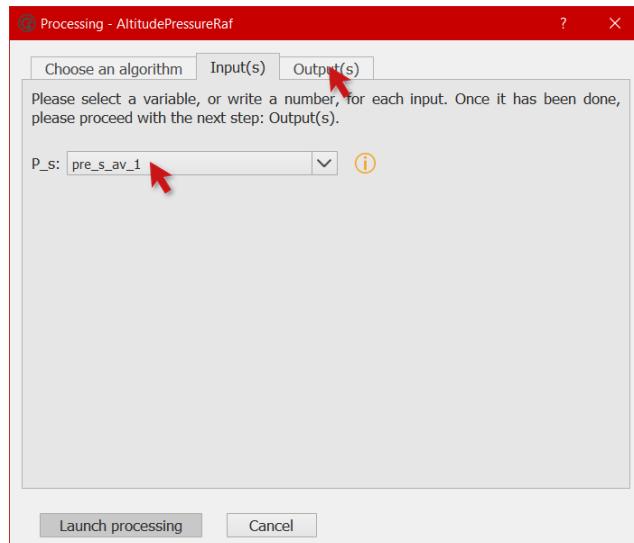
- To launch the processing window, click on . The button should be enabled if variables are detected in the GUI.



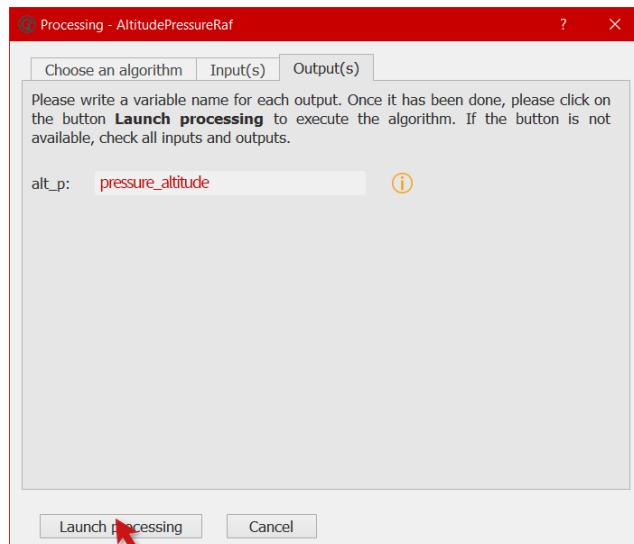
- The processing window is composed of three tabs: the first one to choose the algorithm, the second one to choose the variable(s) processed by the algorithm, and the last one to set the output(s). First, the user has to choose an algorithm by selecting a Category and an Algorithm. Information are displayed in the lower part of the tab. Then click on the Input (s) tab.



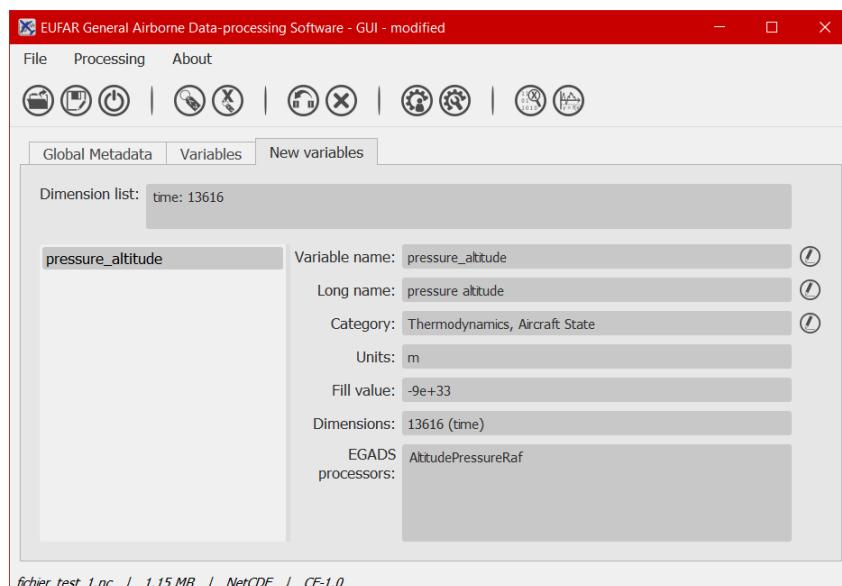
- Select the variable(s) to be injected in the algorithm. Info buttons are here for the description of the variables involved in the algorithm. Then click on Output (s).



- Choose a name for the output(s). The name(s) will be displayed in the main window after the processing.

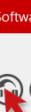


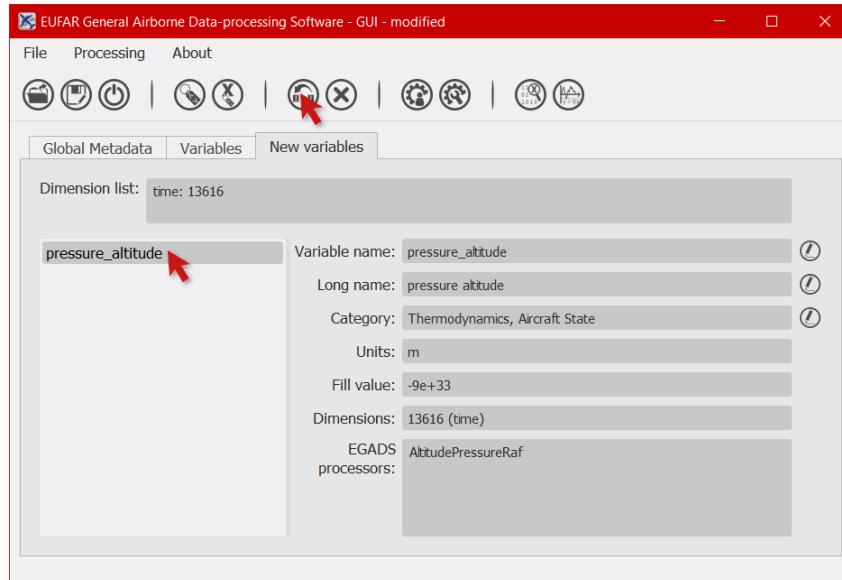
- Depending on the size of the input variables and on the complexity of the algorithm, the processing can take time. Once the output(s) is(are) ready, a new tab appears and new variables are stored here.



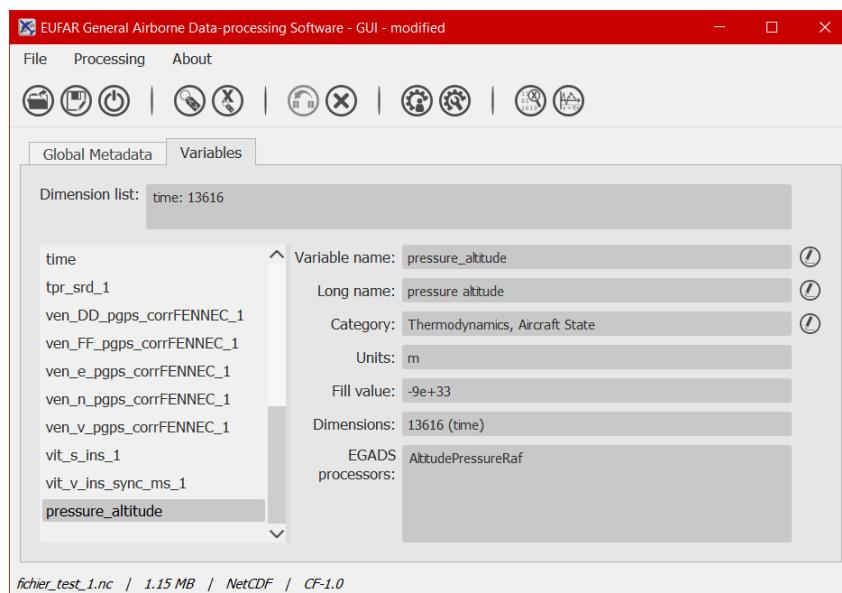
## 4.8 How to migrate a variable from the new variable tab to the variable tab?

To avoid wrong manipulation and for the sake of convenience, a tab dedicated to new variables and called New variables has been introduced in EGADS GUI. Once a variable is created, it will appear in this new tab. If the user wants to save the new variable(s), he must migrate it/them in the Variables tab.

- To migrate a newly-created variable, select the variable to be migrated in the New variables tab and click on .

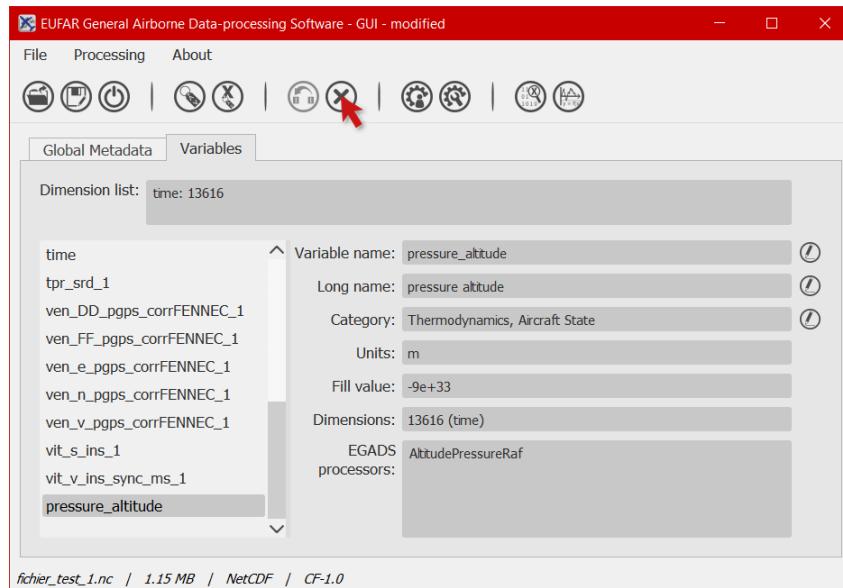


- The newly-created variable should disappear from the New variables tab (the tab is removed if the list of variable is empty) and appear in the Variables tab.



## 4.9 How to delete a variable?

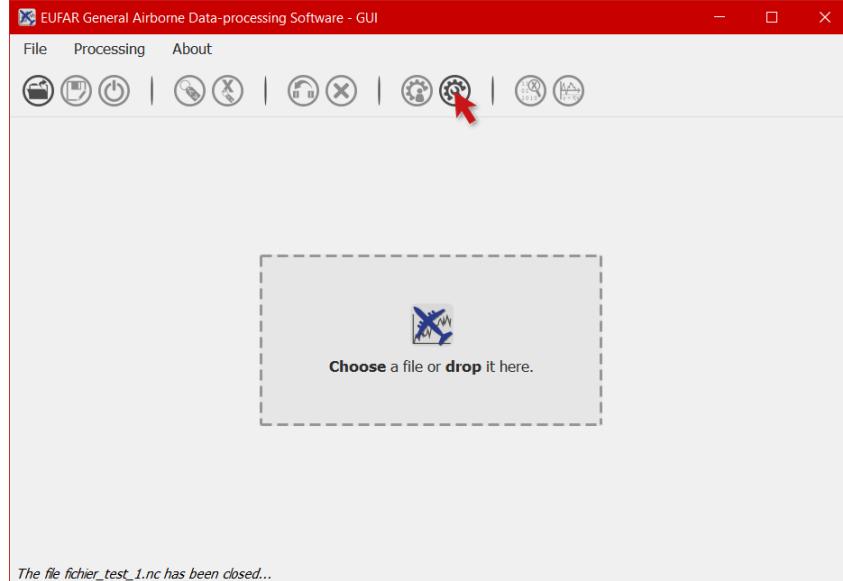
- In the Variables or New variables tab, select a variable in the list and click on .



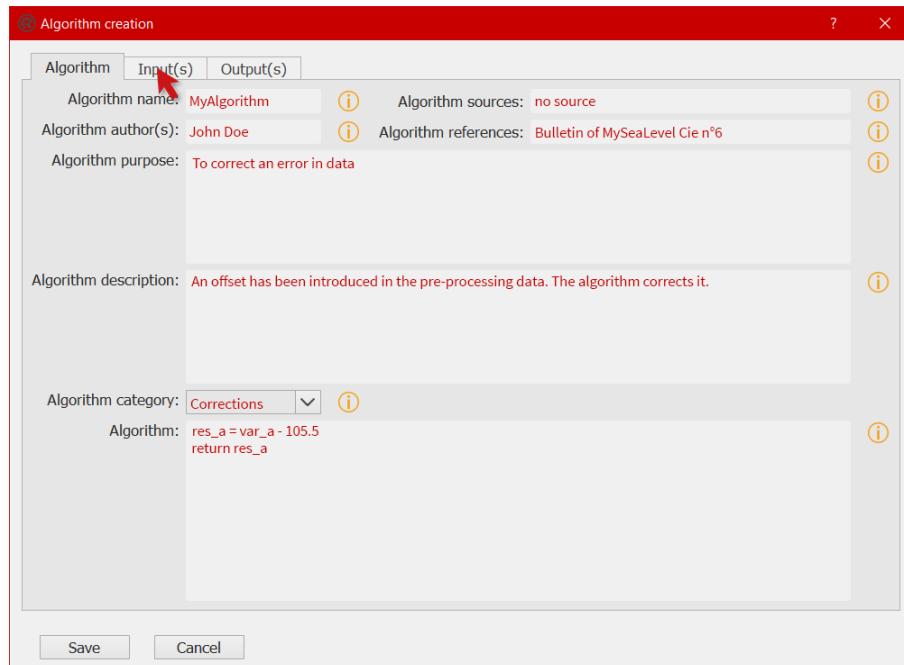
## 4.10 How to create an algorithm through the GUI?

EGADS GUI offers the possibility to create algorithm from a window, by filling in different type of fields. If a complex algorithm has to be written, the most suitable way is to use a text editor like Notepad++ (Windows) or Kate (Linux) and modify the template provided in the EGADS algorithm directory. For more details on how to create algorithms and the purpose of each field, please take a look in the EGADS Lineage Documentation, chapter ALGORITHM DEVELOPMENT.

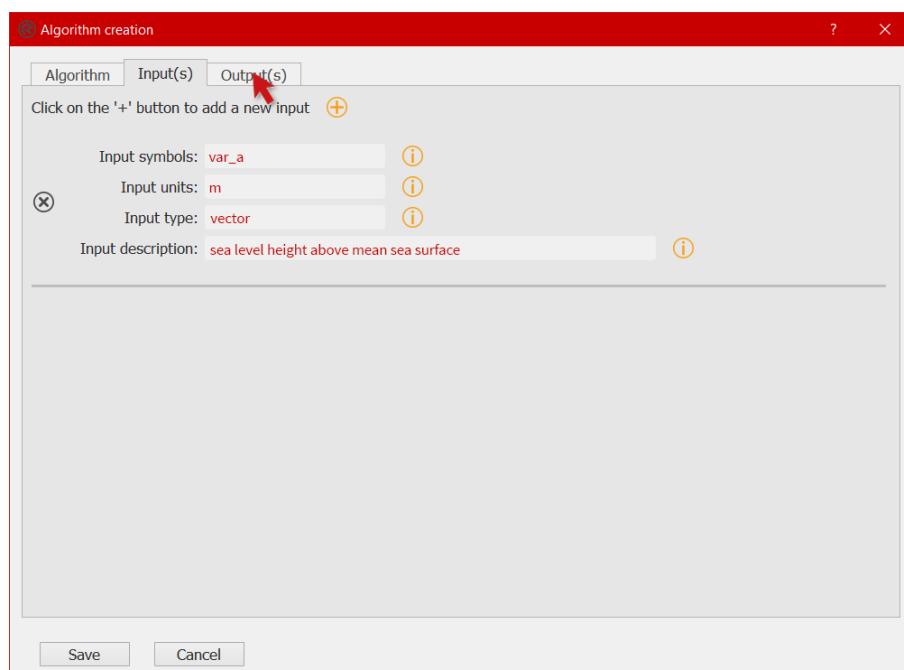
- To launch the algorithm creation window, just click on .



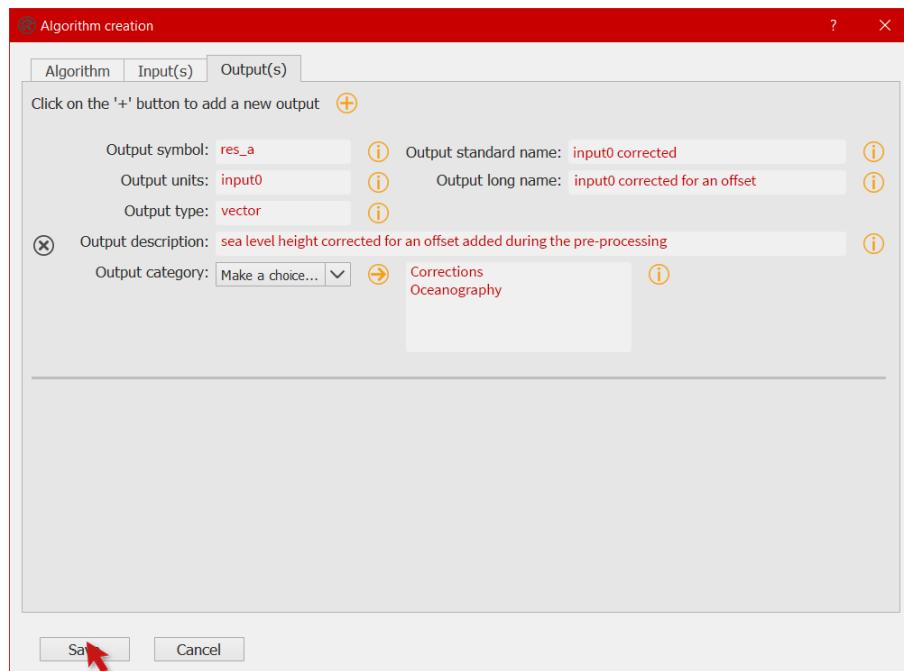
- The algorithm creation window is composed of three tabs. The first one to prepare all metadata (category, sources, ...) and the algorithm formula, the second one to create all inputs needed by the algorithm, and the last one to create all outputs returned by the algorithm. First let's prepare all metadata and then the algorithm itself. A special attention should be paid to the Algorithm field. The formula has to be written in Python 3 (importing modules like numpy is possible). The input variables injected in the algorithm and the output variables returned by the algorithm have to be prepared in the next tabs. Once you are done with the first tab, click on the tab Input(s).



- To create an input variable, click on the + button and fill in the different fields. Then click on the tab Output(s).



- To create an output variable, click on the + button. The output variable(s) is(are) the result of the algorithm. As EGADS and EGADS GUI are coded in Python, the output variable(s) should be returned by the algorithm: `return var`. If the Units of a result has to be the same as one of the input, the user can enter `input` and the input number (starting at 0): `input0` if the output unit has to be the same than the first input variable. It's the same for the Output standard name and the Output long name. Once all fields are filled, click on the Algorithm tab.




---

**Note:** It is important to remember that special characters and punctuation, apart from the following list . , ; : – \_, can't be used in the different fields. The system won't remove them and it can lead to an exception when the GUI tries to load the algorithm.

---

**Note:** There is no system to check the algorithm result. Once the algorithm is saved, EGADS and its GUI expect the algorithm to be true and stable.

---

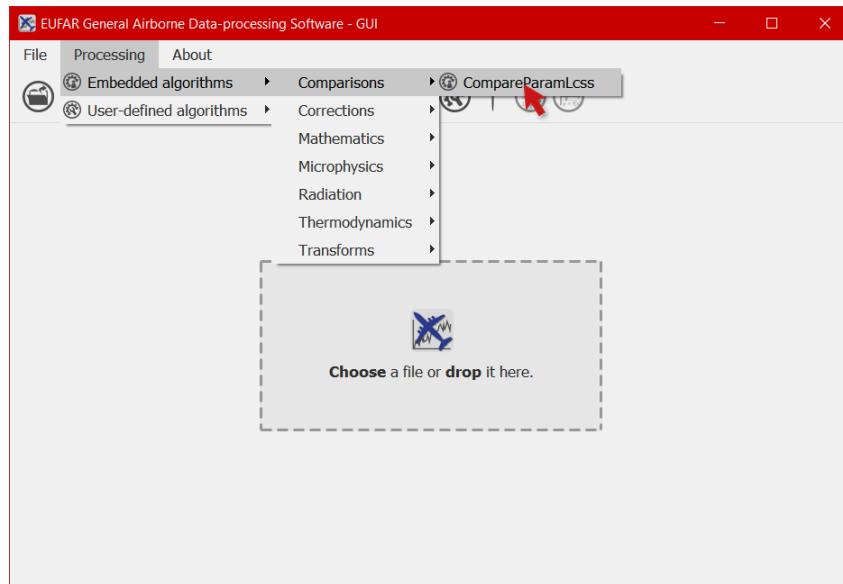
**Note:** If using the stand-alone version of the GUI, new algorithms are saved in the `user_algorithms` folder in the main directory. The user always has the possibility to create an algorithm manually with a text editor. In that case, follow the EGADS documentation about new algorithms and save the algorithm in the `user_algorithms` folder. Do not forget to edit or create `__init__.py` files.

---

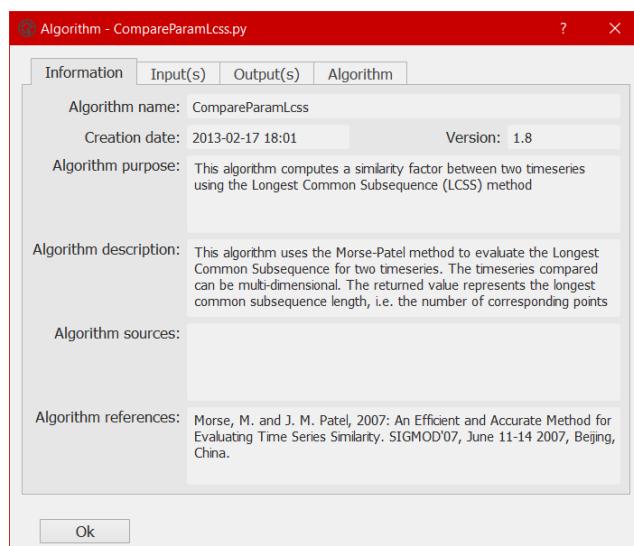
## 4.11 How to display information about an algorithm?

When processing data, it is always a good idea to understand how an algorithm works before making use of it. EGADS GUI offers the possibility to display an algorithm in a window.

- To launch the algorithm display window, just click on the **Processing** item in the menu bar, then select a category and click on an algorithm.



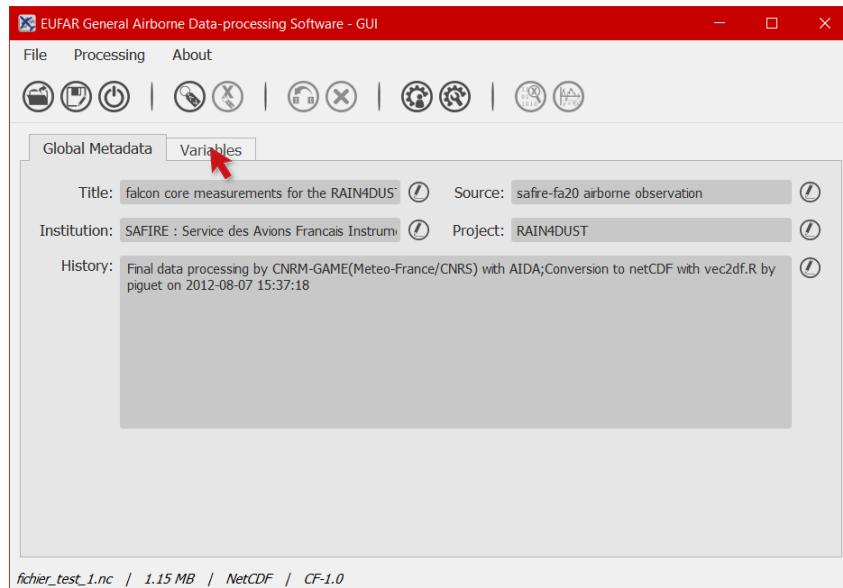
- Information about an algorithm can be found in several tabs.



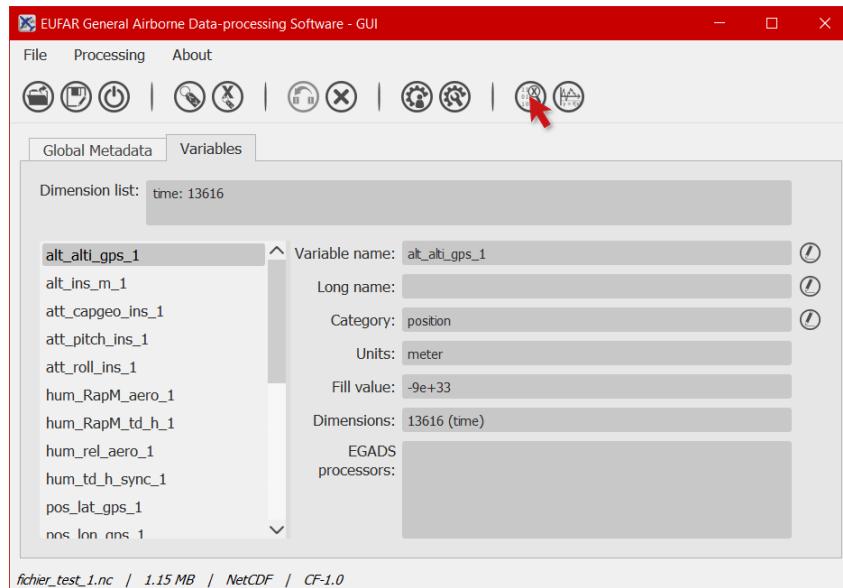
## 4.12 How to display variable data?

The EGADS GUI offers the possibility to display the values of a variable.

- First select the Variables tab.



- Then select a variable and click on . You can also right click on a variable to display the variable menu and then click on Display data.



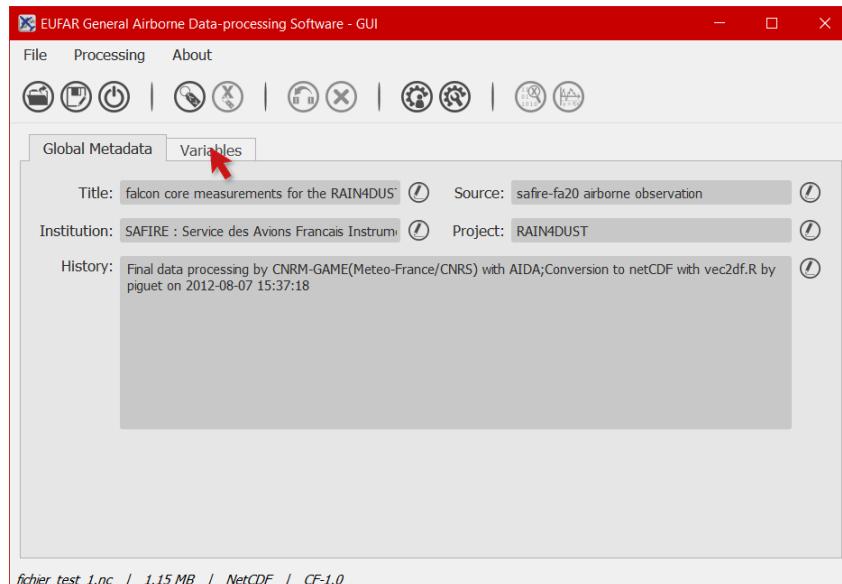
- The values are displayed in a table, and few information are given like the units and the name of the variable.

Time				
11659.2401100000	31660.240009	31661.240009	31662.240009	31663.240009
1 35.945126	35.945126	35.60266	35.418068	35.341366

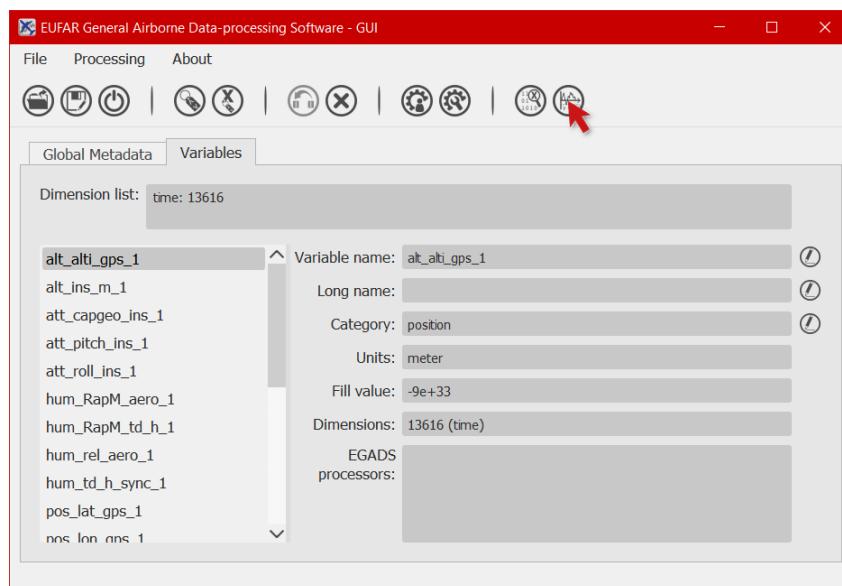
## 4.13 How to plot a variable ?

By integrating the Python modules Matplotlib and Cartopy, the EGADS GUI gives the user a great tool to plot data and save the result in a graphic file. Times series and gridded data, georeferenced or not, are handled by the plot window. Actually, there is a limitation with gridded data. Due to an early version of Cartopy, the modification of ticks and labels is only possible for PlateCarree and Mercator projection, until a new version of Cartopy is released.

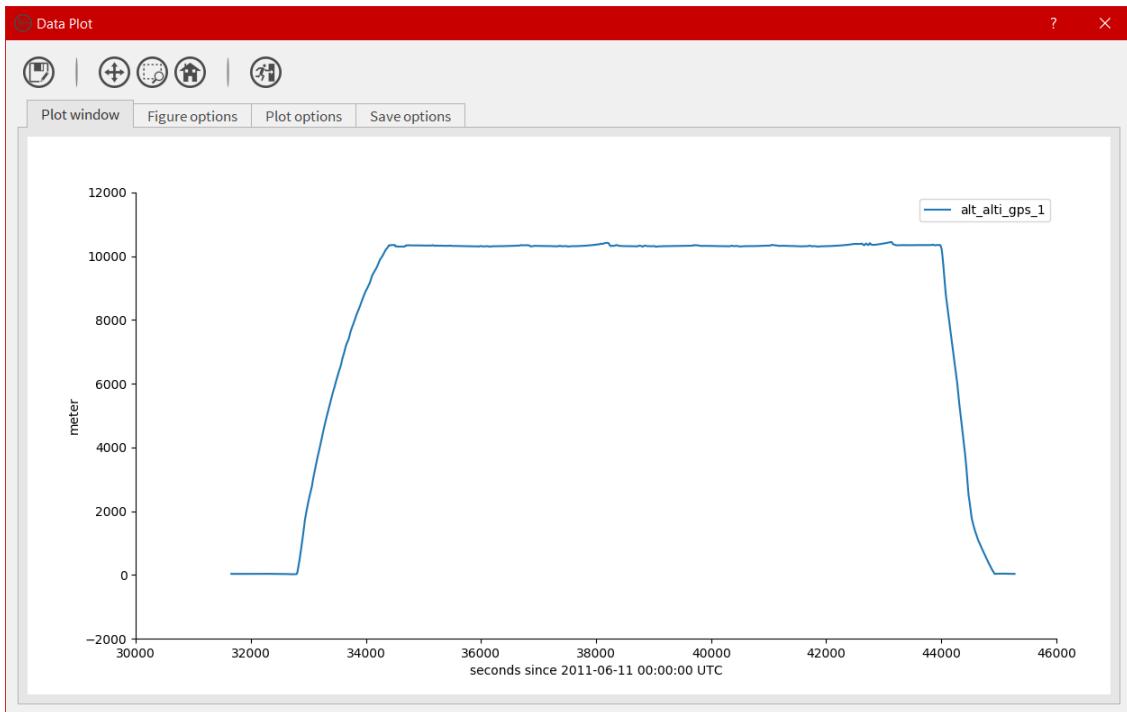
- To access the plot module, select first the Variables tab.



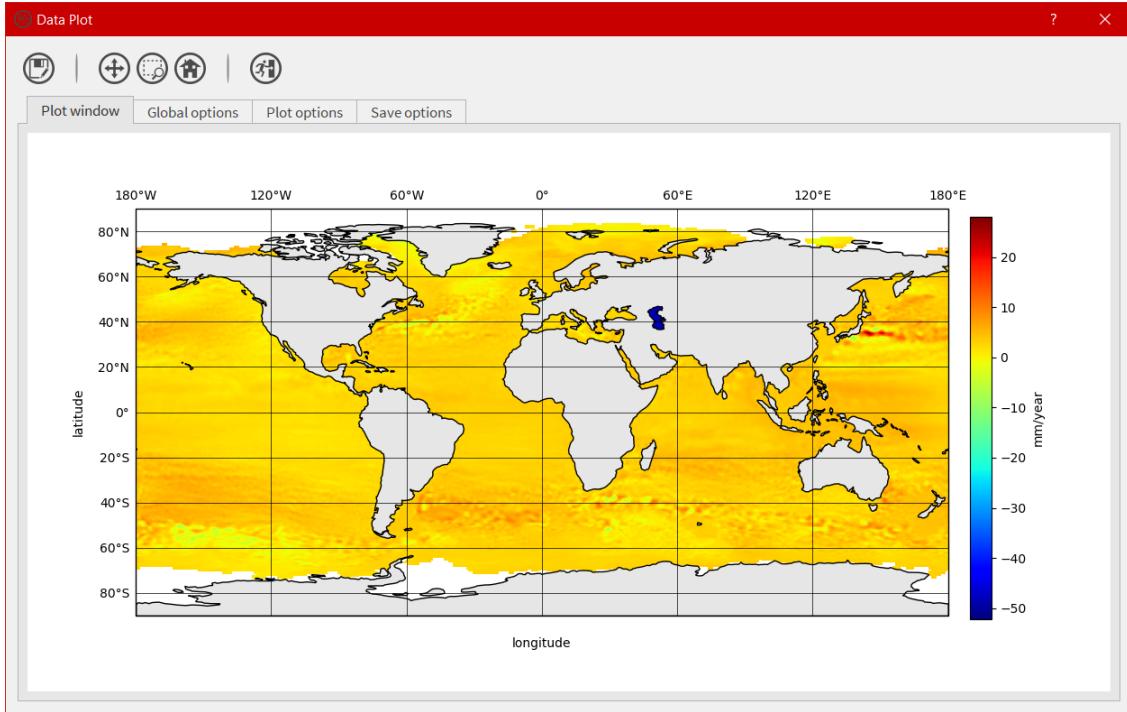
- Then select a variable (or more) and click on . You can also right click on a variable to display the variable menu and then click on Plot data.



- Time series or gridded data are plot using default options. Figure and plot options are available to modify entirely the plot. There is also a tab for options dedicated to the saving process.



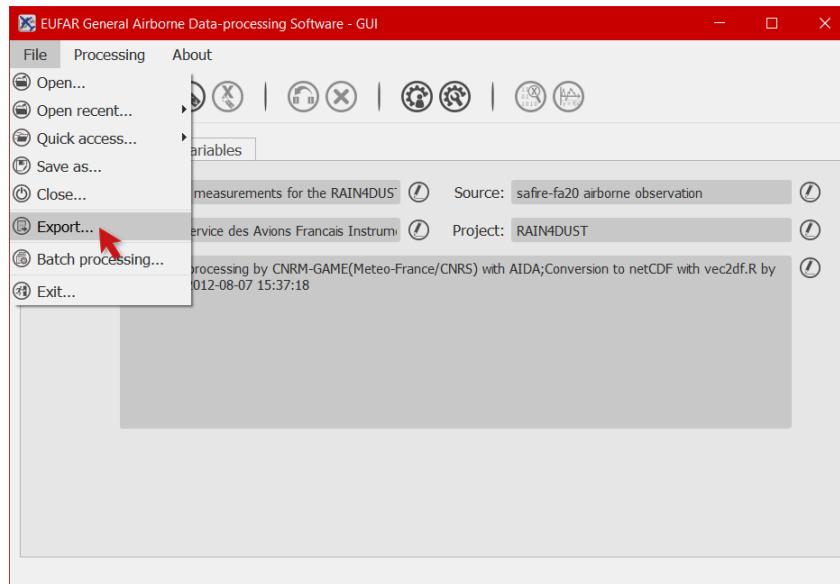
- An example of what can be achieved with gridded data.



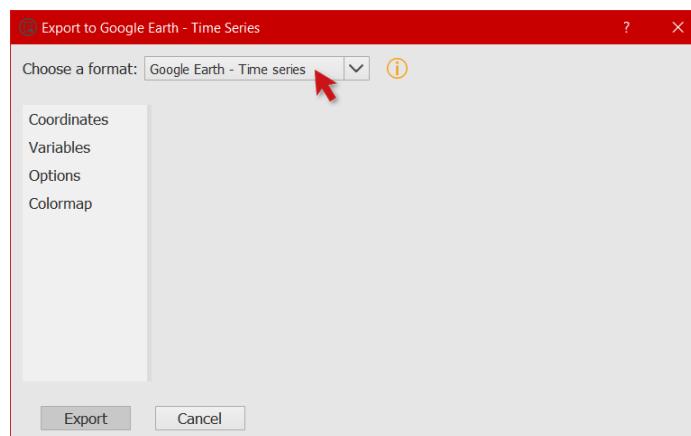
## 4.14 How to export data to an external format ?

If the user wants to display data into Google Earth software, there is a possibility to export data to KML/KMZ file format. At this time, only data along a georeferenced path (generally time series) can be exported and displayed. Later, it will be possible to generate map and embed it in a KML/KMZ file for gridded data.

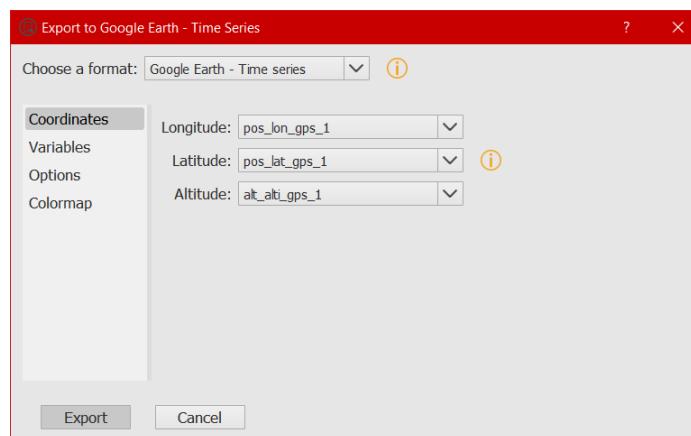
- To launch the export window, click on the **File** menu and then on . The button should be enabled if a netCDF or NasaAmes file has been opened.



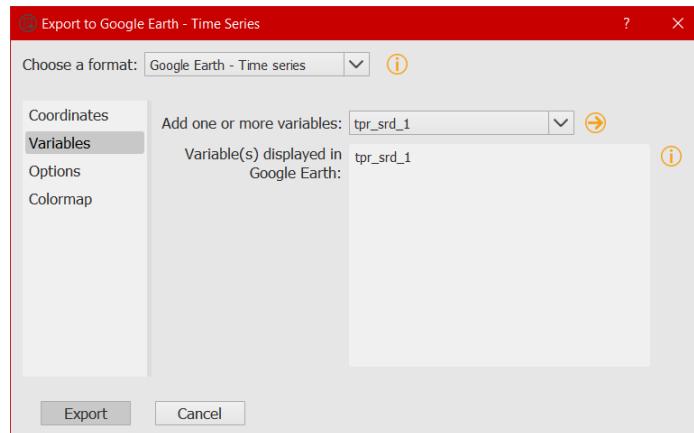
- The export window is composed of a combobox to select the new format, a menu on the left for the different sections linked to the selected format and a central widget displaying the different options. First, select the format Google Earth - Time series.



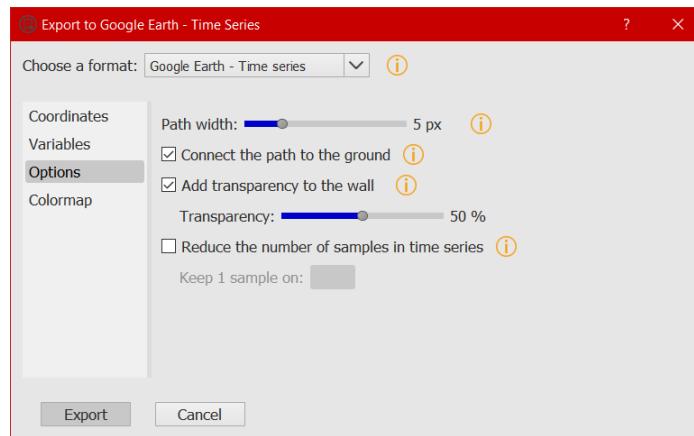
- Click on Coordinates. This part is mandatory, it is used to build the path in Google Earth. Choose a variable for the longitude and latitude, then a variable for the altitude, or select Stick to the ground if altitude is not necessary.



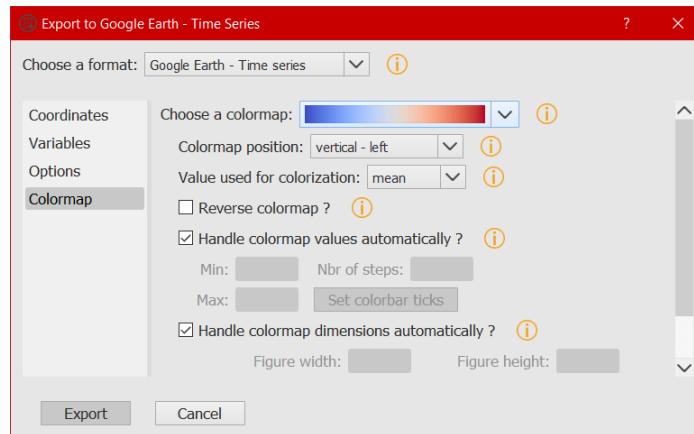
- Then click on Variables. Select the variable you want to “display” in Google Earth and click on the arrow button to register if. At this time, only one variable can be registered, but this number should increase once the function is stable.



- Click on Options. Here are referenced few options to tweak the look of the path. If the path and the variable to be displayed in Google Earth contain a great number of points, it is possible to reduce the amount of points with the last option.



- And finally click on Colormap to tweak the colormap. Eight colormaps are available at this time and more tweaking options will be available in the future. To save the file, click on Export once the button has been activated : only Coordinates, Variables and Colormap are mandatory to activate it.



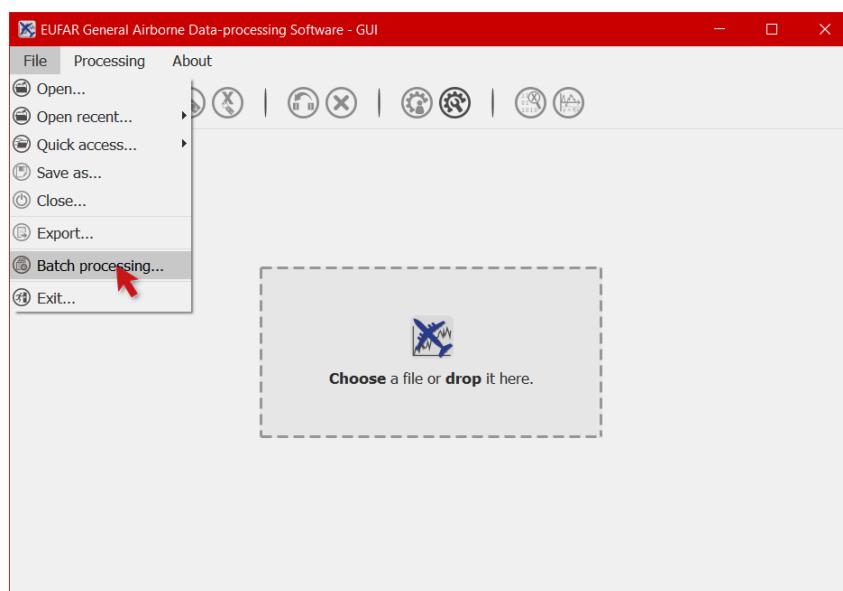
- Here is the result in Google Earth:



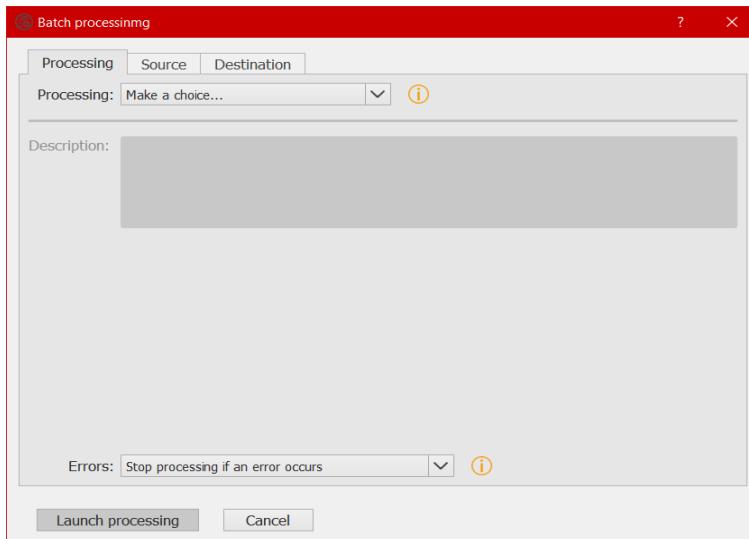
## 4.15 How to use the batch processing function ?

EGADS GUI gives the user the possibility to use a batch processing function. This way it is possible to use an algorithm, concatenate, delete one or more variables, one or more metadata, or just convert between file formats, with multiple files.

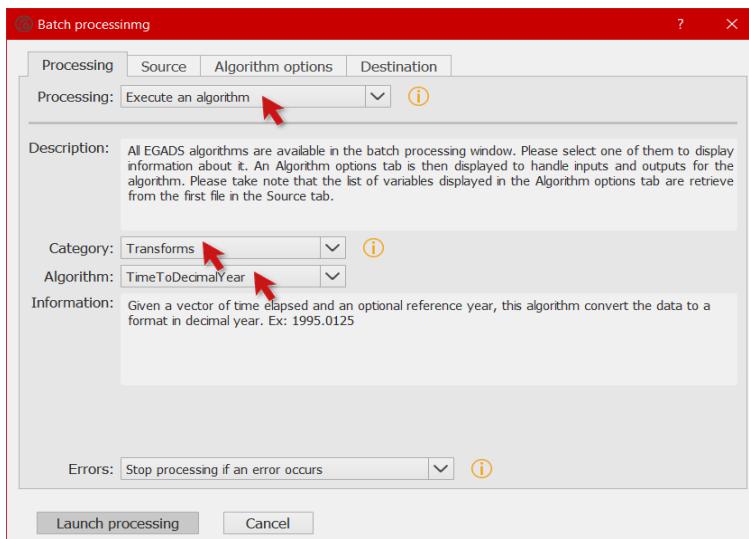
- To launch the batch processing window, click on the File menu and then on . The button should be enabled if variables are detected in the GUI.



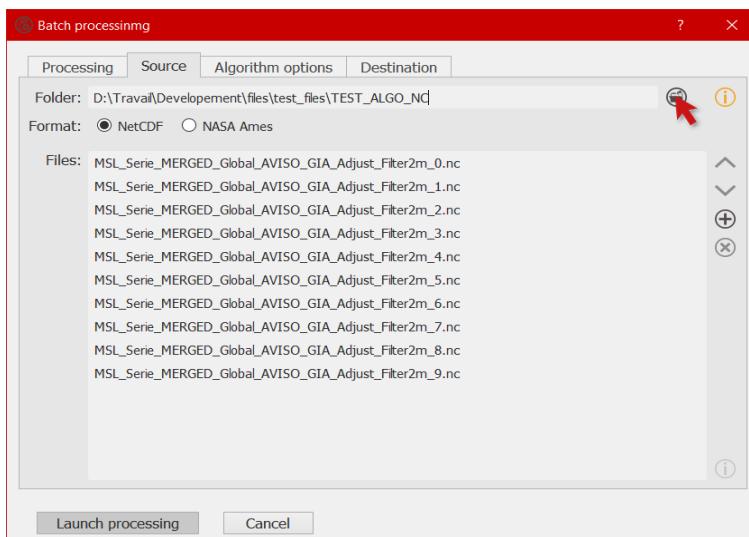
- The processing window is composed of three to four tabs: the first one to choose the processing (concatenation, variable or metadata deletion, conversion between file format, algorithm processing), the second one to choose the files to be processed, the third one, if needed by the processing, can list options, and the last one to set the output folder and the filename options of all new files.



- First, select a processing in the processing combobox, for example Execute an algorithm, then select a Category and an Algorithm in the following comboboxes.

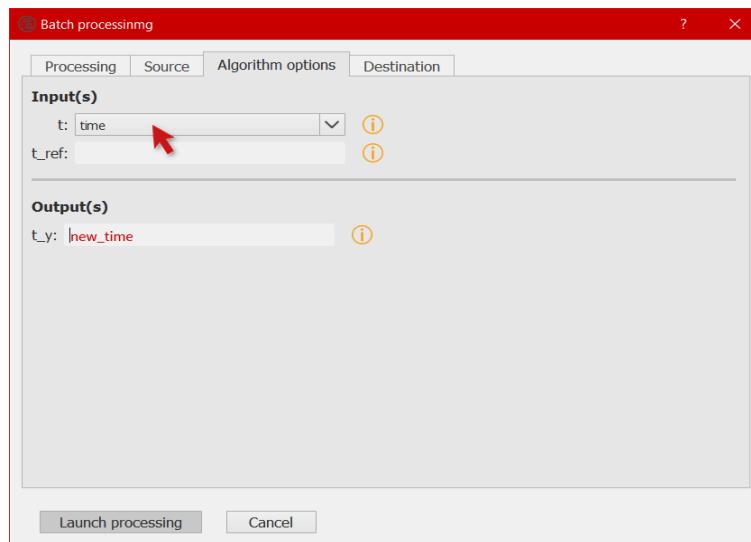


- Select the files to be processed by the algorithm. It will populate automatically the options and a list of variable in the next tab.



- As in the processing window, select the variables to be injected in the algorithm, and choose a name for the

output(s).



- Finally select an output folder, and set options to define a new filename for all output files. Click on Launch processing to start the batch processing.

