Tutorial BIDS Manager:

Brain Imaging Data Structure (BIDS) is a standardized way to organize and describe neuroimaging, electrophysiological and behavioral data. This organization has been adopted by a multitude of neuroscience labs around the world to facilitate sharing and analysis (https://www.nature.com/articles/sdata201644).

BIDS Manager software described in this manual is a tool that allows various users to easily import and explore databases in BIDS format.

This document will guide you to import your data in BIDS format and explore your BIDS Dataset.

Manually driven processes for data storing can lead to human errors, which cannot be tolerated in the context of a research/clinical datasets. BIDS manager offers a secure system to import and structure subject and patient datasets.

BIDS MANAGER aims to achieve the following objectives:

- Provide a software for clinicians and researchers with a user-friendly interface,
- Define the appropriate directory for the dataset corresponding to a study,
- Select required data,
- Select the data to be imported,
- Provide a monitoring and reporting system for data importation and storage.

Table of contents

Table of illustrations:

1. Before starting

For more information on BIDS Manager implementation and goals, please refer to Roehri et al., submitted. A tutorial video is also available at: https://www.youtube.com/watch?v=lZ6bW3uQUj4

A software demonstration to use BIDS Manager-Pipeline is also available at: https://www.youtube.com/watch?v=oFFJy5q6e3o.

1.1. BIDS Manager-Pipeline in few words

BIDS Manager (BM) is a tool to organize, convert, rename, and store neuroimaging data in a structured way. Figure 1 summarizes the functionality of BIDS Manager. BIDS Manager requires some specific files that will be explained in section 2. The first step is to design the structure that one wants to achieve (e.g. the type of data of interest, the label of the recording sessions, the task names). This informs BM how to link the input files and the BIDS structure. BM then converts the data in an open format, renames them according to the user input and finally store the data in a structured way.

BIDS Manager-Pipeline (BMP) is an extension to process automated analyses using open-source research software solutions on several subjects with common criteria. BMP requires some specific files as the description of software in json file. These files will be explained in section 2. The first is to create those files and to save it in a folder named *SoftwarePipeline*. This folder has to be stored at the same level of bids_manager.exe or *bids_manager* code. This informs BMP how to create the command line to run the software. Finally, BMP make sure the results are saved in the right directory in *derivatives*. (Figure 2)

1.2. Requirements

BIDS Manager runs on Windows and Linux systems, and requires python 3.7. Moreover, two software programs are mandatory to convert the data in the appropriate formats:

- AnyWave for the electrophysiological data, and
- dicm2nii for the neuroimaging data from Xiangrui Li, **OR**
- dcm2niix for the neuroimaging data.
- Matlab runtime >= 9.5 (Download here): Required for dicm2nii or for any software compiled by Matlab.

For now, only these three programs are supported as converters.

1.3. Installation

BIDS Manager does not need to be installed, the downloaded folder contains the software as an executable file.

How to launch the software: Once the folder [BIDSMANAGER] with its subfolders are stored onto the computer, double-click on BIDS_manager.exe. A terminal pops up at startup (black window), then the main interface is displayed.

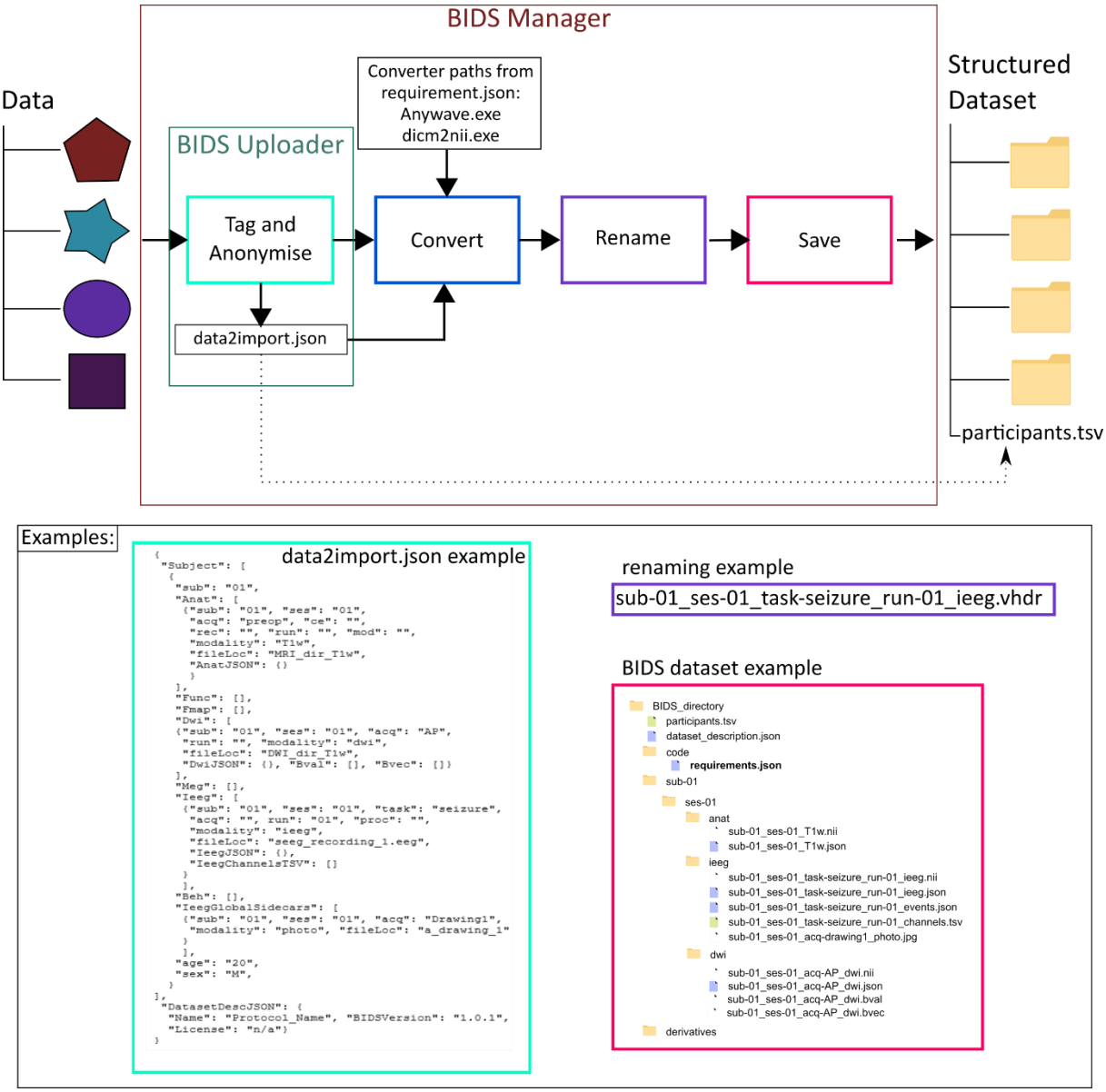


Figure 1: BIDS Manager schema

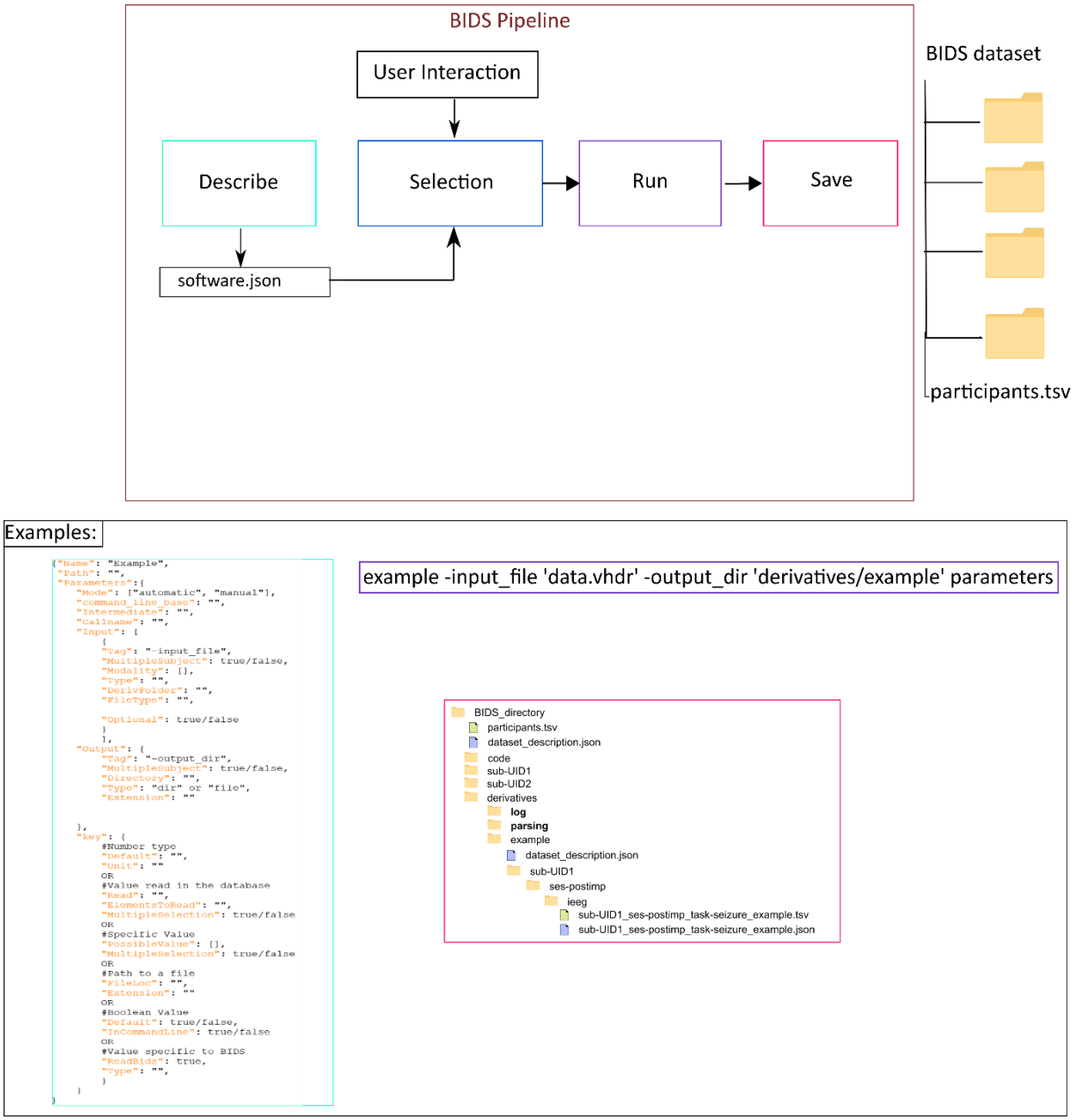


Figure 2: BIDS Manager-Pipeline schema

2. Dataset description

2.1. Supported File Formats:

Input	FORMATS
Subject Information	All relevant information for your dataset
Neuroimaging data	- Anatomical images in dicom data - Functional images in dicom format - Fieldmap data in dicom format - Dwi data in dicom format
Neurophysiology data	- Intracranial Electroencephalography (SEEG, ECoG) in different formats: Micromed, EDF+/BDF+, Brainvision Analyser - Electroencephalography (EEG), same format as SEEG
Electrodes implantation	- Pictures or file displaying the electrode position (pdf, jpg, png) - Implantation file with electrode coordinates (tsv file)
Supplementary files	- Pictures (jpg, png) - Process data (nifti, mat, tsv, vhdr)

2.2. Dataset preparation

Brain Imaging Data Structure (BIDS) is “a simple and intuitive way to organize and describe your neuroimaging and behavioral data” (https://bids.neuroimaging.io/). This organisation has been conceived by Krzysztof J. Gorgolewski and developed by the neuroscience community (Gorgolewski et al., 2016). The goal was to determine a consensus on how to organize and share data. At the beginning, it has been developed for neuroimaging datasets mainly, then it has been extended to all neuroscience data.

BIDS is inspired by the format used in OpenfMRI repository (https://www.openfmri.org/). The data are structured by subjects. In each subject folder can be found modality folders where the data is stored in an open format determined by the community. The name of each file is made of a series of key-value containing information on the subject and the acquisition.

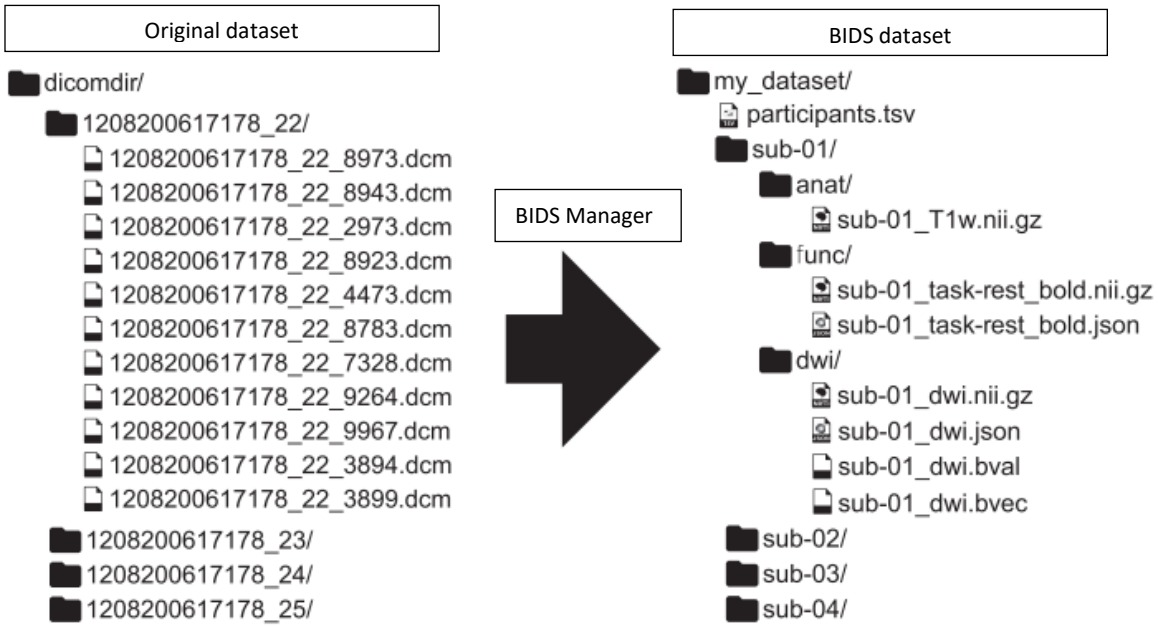


Figure 3: Bids structure dataset (Gorgolewski et al. 2016)

For more information on BIDS structure, please read the publication from Gorgolewski et al., 2016.

BIDS Manager helps to convert the original dataset into BIDS dataset.

2.3. Specific files and supplementary tool

To run properly, BIDS Manager needs information contained in several files that characterize each dataset. These files are required for the importation and the management of the dataset. This section describes the requested files and the procedure for generating them.

2.3.1 Requirements file

The first file is a specific file named “requirements.json”. This file can be created by BIDS Manager or manually by the user. This json file contains the information and data that should be present in the database. The requirements file is specific to a given protocol. It provides the participant characteristics (e.g. sex, age, study group) that should be present in participants.tsv, and the type of data that must be present in the database. This file allows BIDS Manager to verify if the different subjects in the database have the required data and are ready or not for further analysis.

Figure 4: Requirements file

2.3.2 Data2import file

Each time a new entry is made in the database, BM must be informed about the data to be imported. For this purpose, it is required to provide the data2import.json file. This file must be present in the folder containing the data to be imported.

Note: It is highly recommended to copy in a dedicated folder all the data that will be imported. This folder will be deleted after the importation.

This data2import.json file contains all information needed by BIDS Manager to organize the data.
Four sections are present in this file:

- *Subject* section: contains all the information about the subject: the participant characteristics and modalities of interest for the dataset. For each modality, different attributes are given to create the file name (tag) and the location of the original file is mentioned.
- *derivatives* section: indicates the files to import in the dedicated derivatives folder. This section is not yet fully developed, users are not encouraged to use it at this time.
- *DatasetDescJSON* section is similar to the "dataset_description.json" of the BIDS folder. This section is critical to ensure that BM imports the files into the correct directory. The most important is the name, which describes the protocol. BIDS Manager will compare this name with the one written in the dataset_description.json of the BIDS directory. If these names are different BIDS Manager will not import the data.
- *UploadDate* section informs about the date and time of the file creation.

Figure 5 represents an import directory and Figure 7 represents the associated data2import.json. Please note that there is a picture in this folder.

```
import_dir
  |-- data2import.json
  |-- seeg_recording_1.eeg
  |-- seeg_recording_2.trc
  |-- MRI_dir_T1w
       |-- MRI_dir_T1w_1.dcm
       |-- MRI_dir_T1w_2.dcm
       :
       |-- MRI_dir_T1w_N.dcm
  |-- DWI_dir_T1w
       |-- DWI_dir_T1w_1.dcm
       |-- DWI_dir_T1w_2.dcm
       :
       |-- DWI_dir_T1w_N.dcm
  |-- a_drawing_1.jpg
```

Figure 5: Import directory example

Pictures are considered as "sidecar" files in BIDS Manager because they can contain complementary information on a modality acquisition parameters. This picture is to be stored in the associated modality folder. In this illustration, the picture is associated with SEEG modality (i.e., IeegGlobalSidecars in the data2import.)

Using the data2import.json, the importation process can be run by BIDS Manager. In order to avoid incorrect importation, BIDS Manager follows the pipeline described in Figure 6.

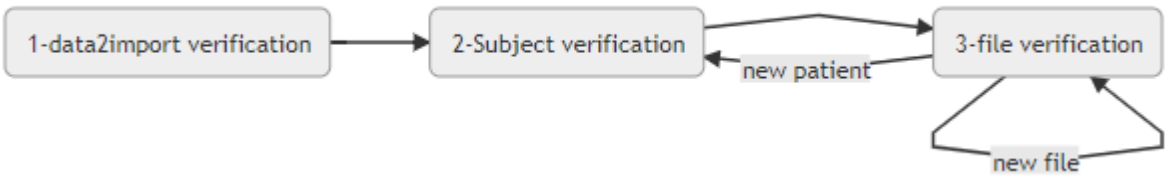


Figure 6: Diagram explaining the verification steps for importation

This pipeline includes several verifications on the subject and files. For more information on the verification procedure, please go to the annex.

```
{
 "Subject": [
  {
   "sub": "01",
   "Anat": [
    {"sub": "01", "ses": "01", "acq": "preop", "ce": "",
     "rec": "", "run": "", "mod": "", "modality": "T1w",
     "fileLoc": "MRI_dir_T1w",
     "AnatJSON": {}
     }
   ],
   "Func": [],
   "Fmap": [],
   "Dwi": [
   {"sub": "01", "ses": "01", "acq": "AP", "run": "", "modality": "dwi",
   "fileLoc": "DWI_dir_T1w", "DwiJSON": {}, "Bval": [], "Bvec": []}
   ],
   "Meg": [],
   "Ieeg": [
    {"sub": "01", "ses": "01", "task": "seizure", "acq": "",
    "run": "01", "proc": "", "modality": "ieeg",
     "fileLoc": "seeg_recording_1.eeg",
     "IeegJSON": {},
     "IeegChannelsTSV": []
    },
    {"sub": "01", "ses": "01", "task": "seizure", "acq": "",
    "run": "02", "proc": "", "modality": "ieeg",
     "fileLoc": "seeg_recording_2.trc",
     "IeegJSON": {},
     "IeegChannelsTSV": []
    }
    ],
   "Beh": [],
   "IeegGlobalSidecars": [
    {"sub": "01", "ses": "01", "acq": "Drawing1", "modality": "photo",
     "fileLoc": "a_drawing_1"
    }
    ],
   "age": "20",
   "sex": "M",
  }
],
 "DatasetDescJSON": {
  "Name": "Protocol_Name", "BIDSVersion": "1.0.1", "License": "n/a"}
}
```

Subject ID

Attributes to create the filename, and the sub-folders in the subject directory

Location of the imagery in the import folder

characteristics of this subject. Values that will be present in the participants.tsv

Figure 7: data2import.json example

2.3.3 Participants file

Participants file is required by BIDS specifications. This document contains the "id" of all participants in the dataset. Moreover, for each participant, their characteristics are provided. The values displayed in this file will depend on the values specified in the requirements file. BM also incorporates various indications as to whether participants' dataset have all the required data.

According to the modalities required in the dataset, the participants.tsv table will have column dedicated to those modalities and display 'ready' or not for each corresponding cell. There is a general column to notify if the participant is ready: "Subject_ready" and there are several other columns corresponding to the modalities in the requirements.json file.

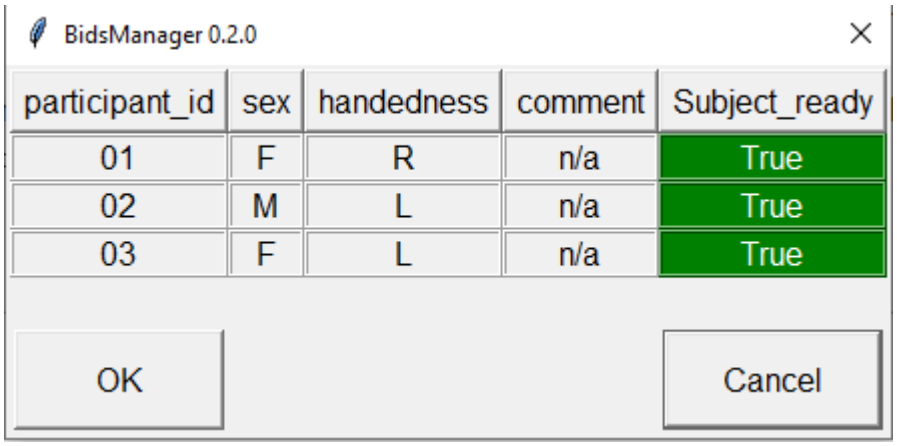


Figure 8:Participants tsv file showing by BIDS Manager

In this example (figure 4), the modalities required for each participant are one anatomical image, one iEEG recording, one iEEG implantation file. The participants.tsv (figure 9) have thus the three corresponding columns (i.e. Anat_ready, Ieeg_read, IeegGlobalSidecars_ready). It also contains

Anat_ready	Ieeg_ready	Ieeg_integrity	IeegGlobalSidecars_ready
True	True	False	True
True	True	False	True
True	True	False	True
False	True	False	True
False	True	False	True

Figure 9: Column describing if the participant's data are ready

another one (Ieeg_Integrity) which states whether in all iEEG files the electrodes have the same name as the ones found in the IeegGlobalSidecars electrodes.tsv.

2.3.4 Integrated tool for importation: BIDS Uploader

BIDS Manager integrates complementary tool named “BIDS Uploader”, which prevents the user from generating the data2import.json manually, which can be time-consuming and source of error. This tool creates an import directory, copies inside the original data (possibly anonymizes them) and generates the corresponding data2import.json. To call this tool, the requirements file must be present in the BIDS directory. Moreover, it must be completed with all sections (e.g. subject information, modalities required, and possible modalities). BIDS Uploader reads the requirements file to offer the complete specific information of the current BIDS dataset.

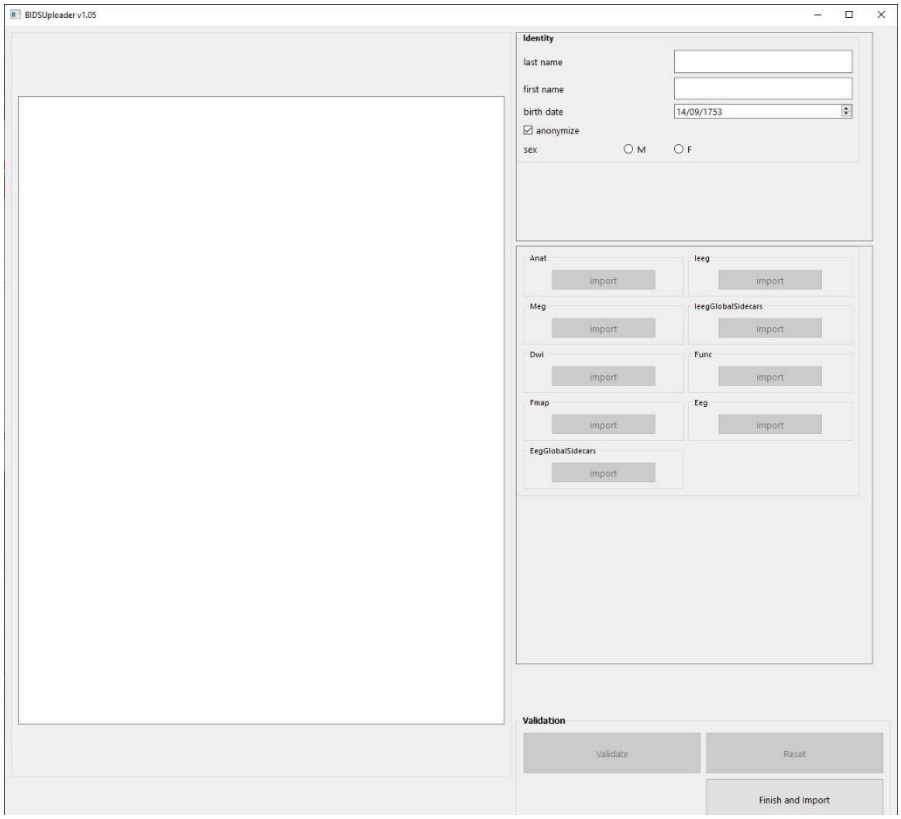


Figure 10: BIDS Uploader interface

For each subject, their characteristics (e.g. sex, age), modalities to be imported and file locations have to be specified and validated. The data2import is generated and data are copied in a temporary folder.

More explanation will be given on the GUI in the section functionality (4.2.1).

2.3.5 Software json file

To call a process with BIDS Pipeline, the process/software must be described in json file saved in *SoftwarePipeline* folder. This file should follow a template (fig. 11) with specific keys/values.

```
{"Name": "",
 "Path": "",
 "Parameters": {
        "Mode": ["automatic", "manual"],
        "command_line_base": "",
        "Intermediate": "",
        "Callname": "",
        "Input": [{
                "Tag": "",
                "MultipleSubject": true/false,
                "Modality": [],
                "Type": "file", “dir”, “4D“ or “montage”,
                "DerivFolder": "",
                "FileType": "",
                “CombinationMode”: true/false,
                "Optional": true/false
                } ],
        "Output": {
                "Tag": "",
                "MultipleSubject": true/false,
                "Directory": true/false,
                "Type": "file",
                "Extension": ""
        },
        "key": {
                #Number type
                "Default": "",
                "Unit": ""
                OR
                #Value read in the database
                "Read": "",
                "ElementsToRead": "",
                "MultipleSelection": true/false
                OR
                #Specific Value
                "PossibleValue": [],
                "MultipleSelection": true/false
                OR
                #Path to a file
                "FileLoc": "",
                "Extension": ""
                OR
                #Boolean Value
                "Default": true/false,
                "InCommandLine": true/false
                OR
                #Value specific to BIDS
                "ReadBids": true,
                "Type": "",
                }
        }
}
```

Figure 11: Template to describe module in JSON

Figure 11 presents a template on how to create the JSON file that is necessary for a given piece of software to be readable by BIDS Pipeline. Three main parts are present in the file under the keys: “Name”, “Path” and “Parameters”. The values associated with the above keys are, respectively, the software name, the location of the software on the computer and a dictionary with the software parameters. In the “Parameters” dictionary, some key/value pairs are required. The four first elements are fixed keys, i.e., the name of these keys cannot be changed.

The key “Mode” specifies whether the software can be launched in automatic or in manual mode. In automatic mode, the software performs the process without any user input and runs in the background. In manual mode, the software needs user interaction and opens at least an interface at some point.

The key “command_line_base” is optional and is specific to BIDS App software. It can be used to specify the way to call docker arguments. For example, “docker run -ti –rm”.

The key “Intermediate” specifies which software will run the module, for example Matlab, Docker or AnyWave.

The key “Callname” is the name of the software used in the command line.

The key “Input” determines the inputs required by the module. It is a list of dictionary because the module can have multiple inputs. The meaning of the keys is:

- Tag: Name of the input used in command line
- MultipleSubject: Boolean value to say if the module can handle multiple subject at the same time. (Used when the input is a directory)
- Type: To determine which kind of file BMP has to give to the module. There is 4 options, “file” is a file i.e *.vhdr, “dir” requires a directory, “4D” requires the meg folder of the subject not the C,rfDC i.e. sub-xx_ses-xx_task-x_run-0x_meg (for the C,rfDC put “file”) , “montage” is the montage file in AnyWave format i.e *_montage.mtg.
- DerivFolder: To specify if the inputs must be taken in derivatives folder and not in raw data. The value should stay empty, only having the key in the json means take the data in derivatives folder, the selection of the specific folder will be in the GUI.
- FileType (only with Derivfolder): Must specify the type of the file (e.g. *_rate.mat)
- CombinationMode: This key is to indicate, if multiple inputs, BMP has to use the same file with different files of the other input
- Optionnal: If the inputs is not required and the module can be run without this input say yes

The key “Output” determines the type of output required by the module. The meaning of the key is:

- Tag: Name of the input used in command line
- MultipleSubject: Boolean value to say if the module can handle multiple subject at the same time. (Used when the input is a directory)
- Type: To determine which kind of file BMP has to give to the module. There is one option for now is “file”.
- Directory: To indicate if the module requires a folder to save the results
- Extension: A list of the extension of the files to write

The type of parameters that are accepted are:

- Number type: Two values must be specified, the “Default” value and its “Unit”. For the duration of interest of a signal, one could write “Default”: 10, “Unit”: “s”.

- Value read in the database: Three values must be specified, the “Read” key is to inform BIDS pipeline which file (in the BIDS Dataset) should be read to extract the value and propose it to the user. The “ElementsToRead” key is to inform which part of the file should be read. The “multipleselection” key is to know if the user can select multiple value for this specific parameter.
- Value specific to BIDS: Two values must be specified, the “ReadBids” key is a Boolean to say if BIDS Pipeline should take the value in BIDS key. The “Type” key is to specify which attributes from BIDS data (e.g., sub, ses, task).
- Specific value: Two values must be specified, the “PossibleValue” key should be a list of the value accepted by the software for this parameter. The “MultipleSelection” key is to know if the user can select multiple value for this specific parameter.
- Boolean value: Two values must be specified, the “Default” key is the default Boolean value of the software for this parameter. The “InCommandLine” key is to mention if the key to this parameter should appear (in command line) with its value or if only the key should appear its value is true.
- Path to a file: Two values must be specified, the “FileLoc” key will be fill in by the user with the GUI. The “Extension” key should mention the extension of the file.

Thanks to this file BMP will be able to create the GUI for the subject and parameter selection and to create the command line to call the software with chosen parameters.

3. Software interface

3.1. Interface Overview

The main interface of BIDS Manager offers four drop-down menus.

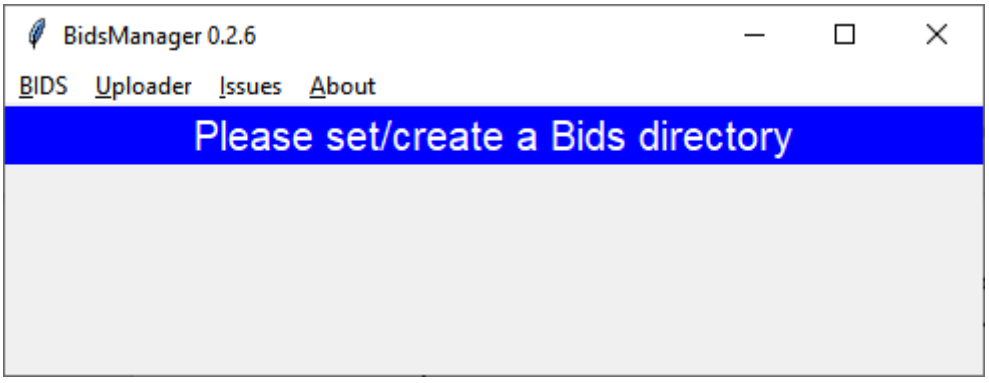


Figure 12: BIDS Manager main interface

BIDS menu will be used to set and explore the BIDS directory.

The menu *Uploader* allows to import data in BIDS format.

If there are issues during the importation, menu *Issues* is here to help the user.

The *About* menu gives information about BIDS Manager Licence and how to cite the software.

3.2. Windows description

According to the menu selected in BIDS Manager, different windows can be displayed. Most of the time, the main window will not change, and other windows will be opened, but in some cases the main window changes its configuration (e.g. when *Issues* menu is selected, see Section 5).

3.2.1 Window to show JSON files

This window displays the content of json files. The keys are displayed in the left column whereas the values are displayed in the right columns. If a key is mandatory according to BIDS specifications, it will be displayed in red. Figure 15 shows an example with the dataset description file.

3.2.2 Window to show TSV files

This window displays the content of tsv files. It appears as a table. The header of the tsv files are displayed as buttons whereas the content is displayed as boxes. In some cases, the background color can be red or green. The green color means good or ready while red means bad or not ready. Figure 8 shows an example with the participants.tsv file.

3.2.3 Window to explore the dataset

This window displays the elements in the selected BIDS Dataset. On the left column can be found the keys and on the right the values. If the value is a directory or a file and thus openable, it will be displayed in a blank box. A double-click will pop-up a menu with different actions if it is a file, or a new window will be opened to display the content if it is a directory. Figure 32 shows an example with one BIDS dataset.

3.2.4 Window to display issue

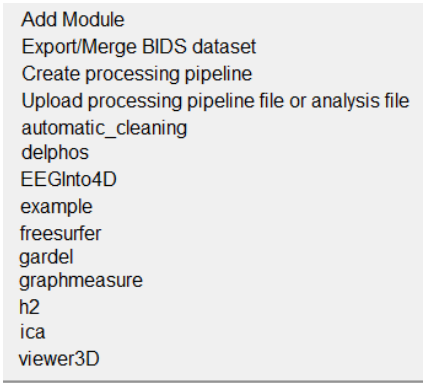
In that case, the main window is displayed but its configuration changes. The main window is divided in two list panels: the right one displays the issues, the left the chosen actions. Two buttons appear on the right, one to apply, the second to delete the actions. Figure 43 illustrates an example with importation issues.

3.3 Interface with BIDS Manager-Pipeline

A new menu appears on the main window named *Pipelines.* This menu is created according to the json file presents in *SoftwarePipeline* folder. A list of the software that can be launch is offered to the user as well as two others an option to create a batch and to upload an analysis that already been realized.

Another menu is presents to create the statistical table.

Figure 13: Main window with BMP



Once the user clicks on , the list appears. The two last option are permanent, but the rest depends on *SoftwarePipeline* folder. Clicking on one process/software/module will make appear the GUI to select the subject to analyse and the parameters. (More information on section 4).

3.4 Interface with AnyWave

On the figure 13, you can see a new menu named “AnyWave”. This menu is here to handle what we call AnyWave files. When an electrophysiological signal is opened with AnyWave, some files are created :

- .mrk: the markers

- .mtg: the montage

- .bad: channels written as bad

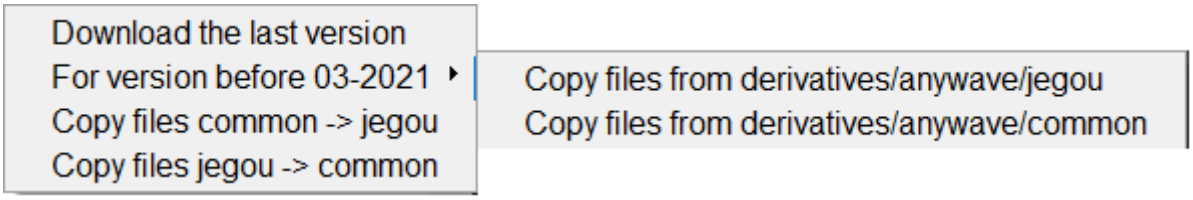
- .levels: amplitude level

- .flt: if a filter is applied

- .display: information about the display

- .sel: channels selected

Those files are normally at the same level of the raw data. However, it is not BIDS so we decided to put them in derivatives folder (e.g. *derivatives/anywave/username/sub-xx/ses-xx/ieeg*). In derivatives/anywave, the files can be saved in the folder of the user (named like him) or in common folder which represents the files with general use. The new version of AnyWave does it naturally but with the oldest, BM will dot the changes.



In the Menu AnyWave, the first option is to download AnyWave version (opening AnyWave website). The second option is to copy the anywave file from derivatives folder (username or common) to the raw folder. The third and the fourth option are to make a copy of the anywave files from username to common or inverse.

Before to copy the files from your folder to the one in common, make sure that your configuration (markers, montage, etc.) are really interesting for all users.

4. Procedure

This section describes step by step how to create the BIDS Directory, import data, explore a dataset, run a process, and create the statistical table.

4.1 Create the BIDS dataset

BIDS Manager allows the user to create the BIDS directory that will be needed to store the data with associated files. To start, user should click on “create new BIDS Directory” in the BIDS menu (see figure 14).

It will ask to select an empty folder, then, a window will appear to create the dataset description file (Figure 15).

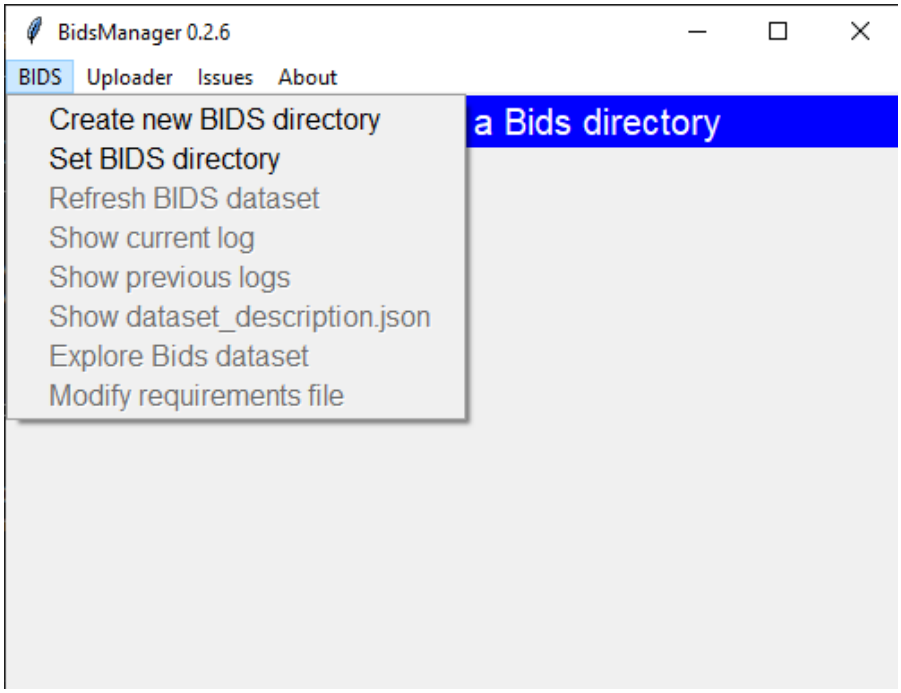


Figure 14: BIDS Menu – BIDS Manager

4.1.1 Dataset Description

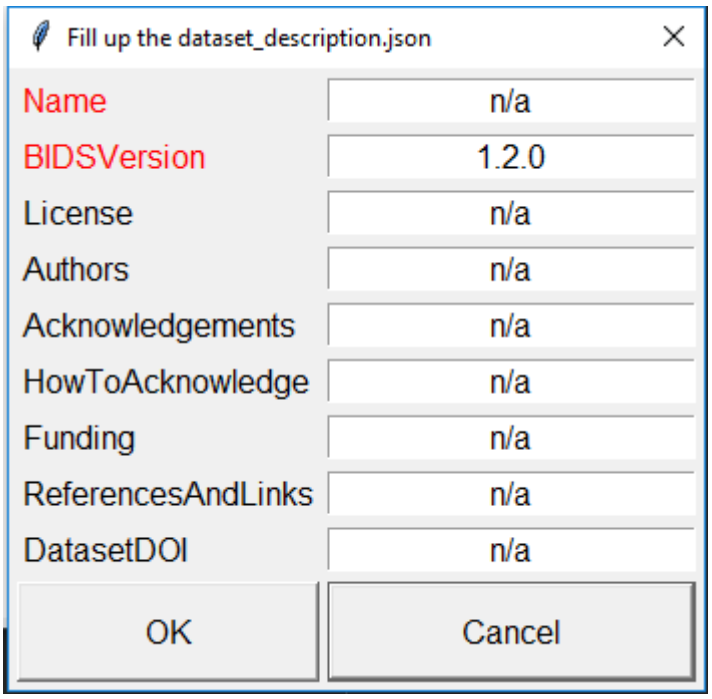


Figure 15: Dataset description

The value in red are very important, if the "Name" is not informed, an error will occur. The "Name" corresponds to the protocol name. It allows to verify that subjects are imported in the right BIDS directory.

The second step is to inform the value for the requirements file.

4.1.2 Requirements

The interface will guide the user to create the requirements file or to load an existing requirements file.

The first three buttons offer the following option:

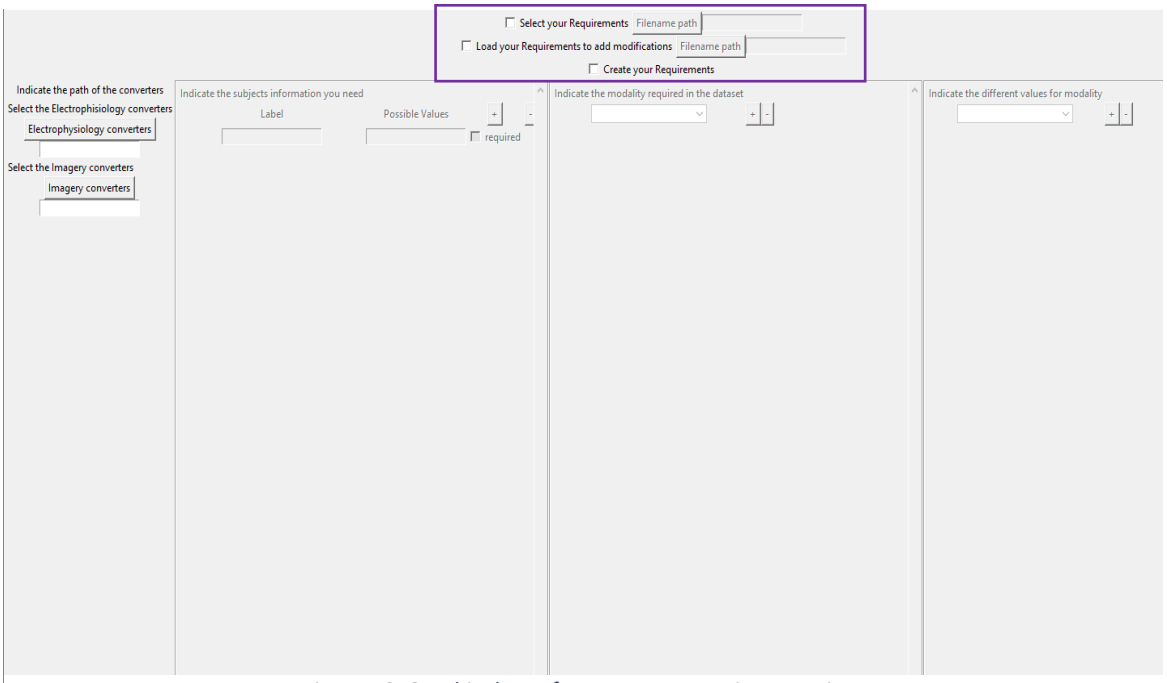


Figure 16: Graphical Interface to create requirements.json

- select an existing requirements.json,
- load a requirements.json and modify it by adding elements using the GUI,
- and create a new requirements.json thanks to the GUI.

By selecting “Create your Requirements” or “Load … to add...”, three frames will be displayed.

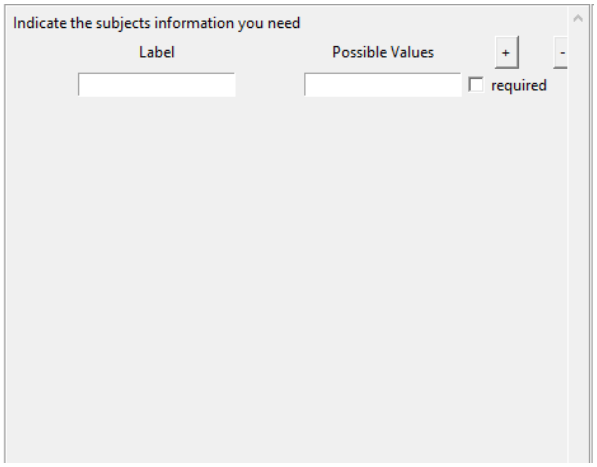


Figure 17: Interface to inform participant information

The first frame is designed to fill the type of participant characteristics required in the participants.tsv.

Label corresponds to the name of the participant information (ex: age, sex, handedness) and cannot be empty. **Possible values** represent the list of the values that the label can take (e.g. F, M), it can stay empty. The values in “possible Values” must be separated by a comma.

The checkbox **required** is used to indicate whether the information is mandatory.

There are two buttons to the right of the frame to add (+) and remove (-) lines. It is possible to add as much information as necessary.

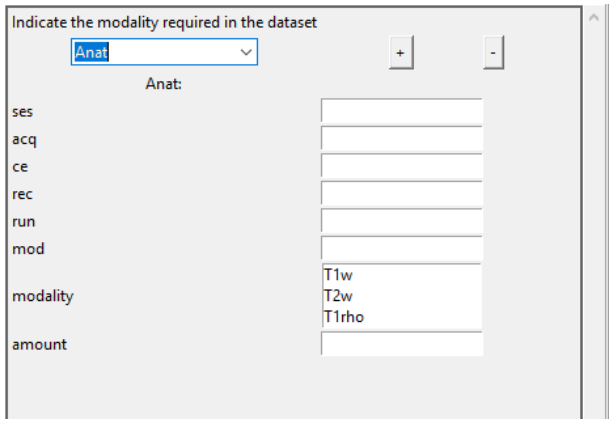


Figure 18: Interface to indicate the data required in the dataset

This second frame: “Indicate the modality required in the dataset” permits to inform which type of data are mandatory in the dataset.

The drop-down list is used to select a modality type (e.g. Anat or iEEG) and the (+) button add it. Then, the user can inform the specificity of the data (e.g. session, modality). For instance, the user could require an T1w MRI from session 1. The (-) button can be used to delete the last modality added.

For the modality part (in Figure 18: T1w, T2w, T1rho, …), several modalities can be selected. However, if the user selects multiple ones, it means that all the selected modalities are required in the dataset. If none are selected, it means that one modality is required but the type does not matter. For the other keys, if the user does not add a specific value but want to check that this key is in the name of the required file, the user should write “_”. However, if the user writes “_”, he must write a list of elements at this key in the third column.

The user can add as much information as needed. However, be careful by selecting the type of data required because all subject folders must contain them.

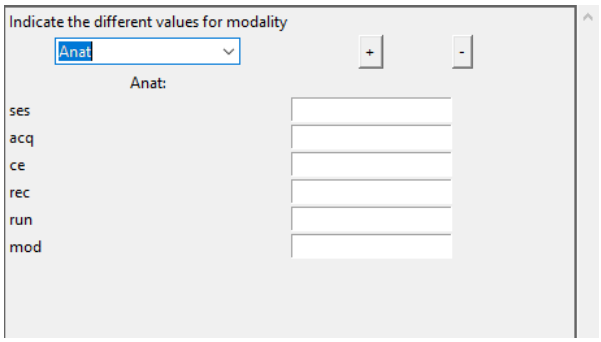


Figure 19: Interface to inform possible attributes for each modality

The third frame: "Indicate the different values for modality" is set to indicate the specificity of each modality type. By modality type, the user indicates the possible values of the different keywords used to name the data (e.g. ses = preimp, postimp). The values must be separated by a comma. The difference with the second frame is that these modalities are not mandatory in the database but the possibility to have them is offered. The user should add all the modalities that he/she can have in a given database with all possible keys.

The last part is used to indicate the path of the software programs used to convert the electrophysiological and imaging data. For now, only AnyWave and dicm2nii are validated converters.

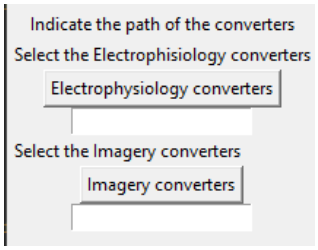


Figure 20: Interface to inform converter paths

Once the user clicked "Ok", the 'requirements.json' is created/saved in the folder Code, and the 'dataset_description.json' and the 'participants.tsv' are created in your BIDS directory.

Once the "requirements.json" is created, the user can still modify it by selecting the option "Modify requirements file" in *BIDS* menu (option available since version 0.2.6 of Bids Manager). It will open the GUI that we just describe.

4.2 Import data in the BIDS dataset

Once a BIDS directory has been specified to BIDS Manager either by using the option "Create new BIDS directory" or by using the option "Set BIDS directory", data can be imported in the dataset. There are two manners to import data with BIDS Manager. Both options are available in the menu *Uploader*.

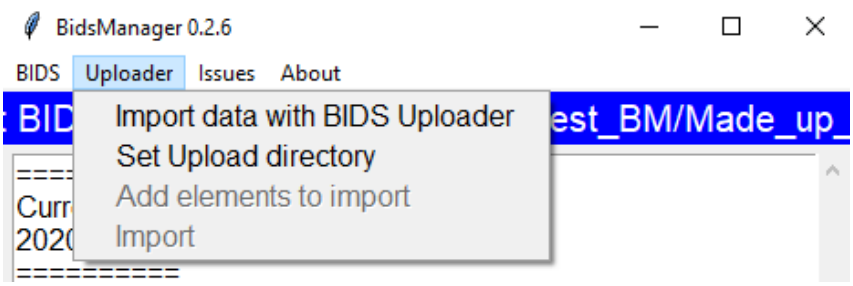


Figure 21: Uploader Menu - Bids Manager

The first option ‘Import data with BIDS Uploader” launches BIDS Uploader to create the data2import file. The second option “Set upload directory” is used when a folder with the data to be imported already contains a data2import file. The following sections describes these two options.

4.2.1 Import data with BIDS Uploader

By clicking on this option, BIDS Manager opens “BIDS Uploader”. BIDS Uploader permits the user to indicate, for a given subject, the data to be imported, to tag the data, copy them in a specific directory and anonymize them, and create the data2import.

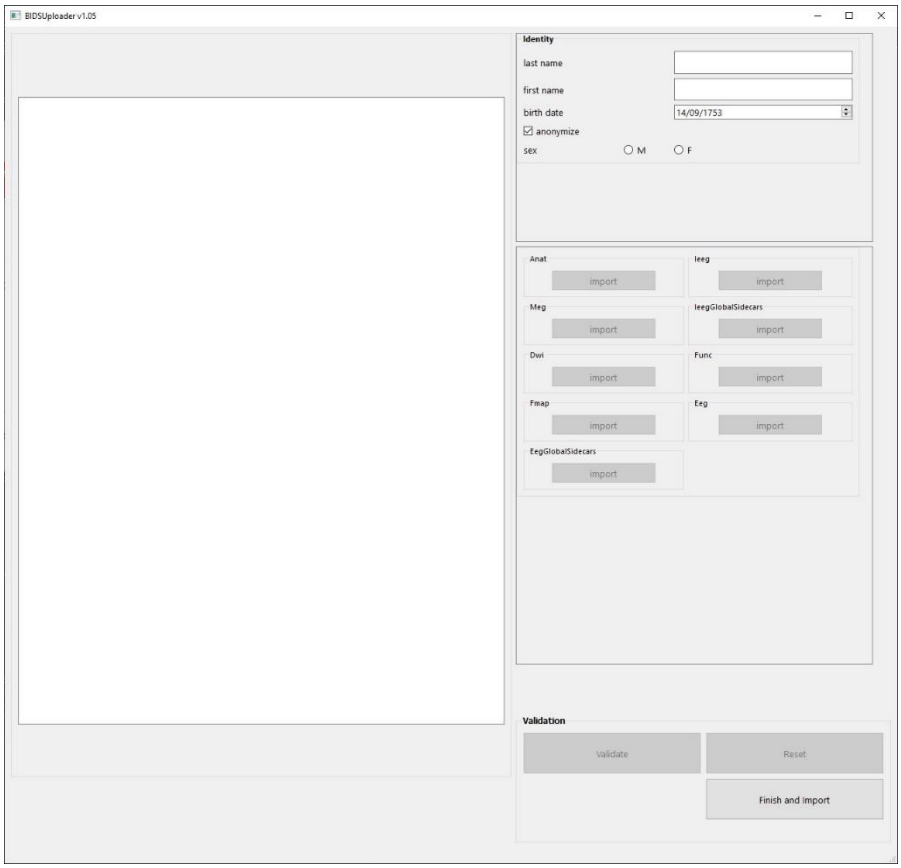


Figure 22: BIDS Uploader interface

This GUI is made according to the requirements file. The requirements file informs the uploader on which type of data can be imported and the subject information to fill in. If the requirements file has not been created, we suggest doing it now before going further.

The main frame of the uploader (left-hand side of Figure 22) is the log.

The *identity* frame is to fill in the subject information. Once the name, sex and date of birth are informed, a new window pop-up with the subject characteristics requested for the BIDS database.

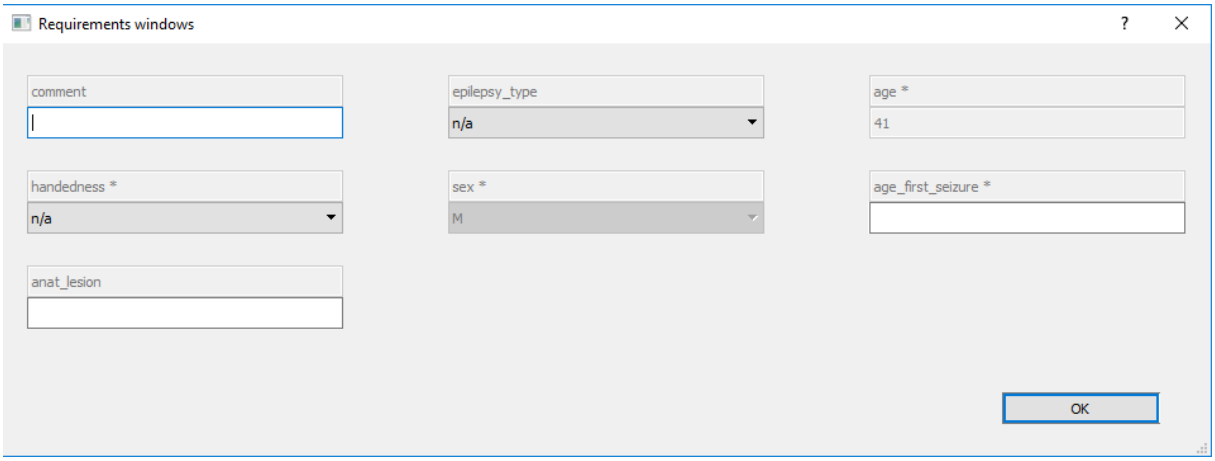


Figure 23: Participant characteristic interface - BIDS Uploader

The anonymize checkbox allows to anonymize the subject data by creating a hexadecimal ID key.

The * indicates the required information.

The other frames are used to import the modalities. Clicking on “import_<modality>” opens a new

Figure 24: Modality interface - BIDS uploader

window with the attributes to be informed. To import a file, select the attributes of the file and click on “Import”. The file explorer will pop-up for the user to select the file. Several files/folders can be selected at once if they have the same attributes, and only the run number changes.

At each step, the log is updated, and the user can see what has been done. He can also verify or modify these steps.

By double clicking on the first line (the one for the subject information), the user will be able to review or modify these characteristics. If the user clicks on “modify”, the frame “identity” will be available again.

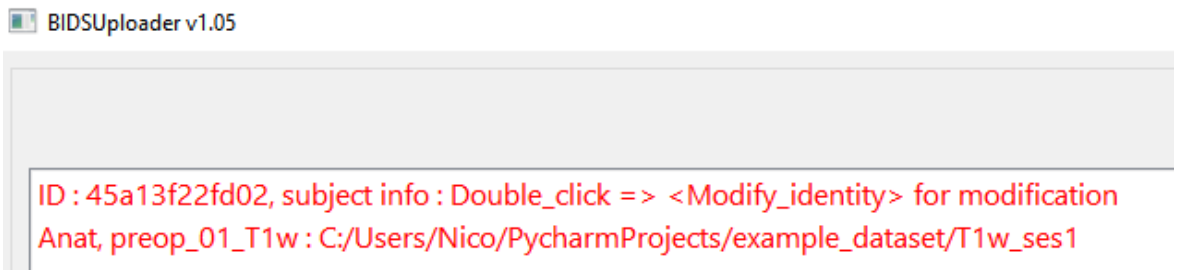


Figure 25: Main frame - BIDS Uploader

The last step is to click on *Validate and format*. If some lines in the log are still in red, it means BIDS Uploader found a mismatch between the identity entered by the user and the identity found in the header of the file. The user has to "Check" the file by right-clicking on it and forcing the verification in case of a pre-anonymized data or delete the file otherwise. When all the lines in the log are green or yellow, the user can validate again, and it will create the data2import for this subject and reset BIDS Uploader. New subjects can be added. Once, all subjects have been done, the window of BIDS Uploader can be closed, and BIDS Manager will import these data in the BIDS directory.

4.2.2 Import data by setting an upload directory

For this option, an upload directory with one data2import.json must exist. By clicking on "Set Upload Directory", the path of the upload directory can be informed.

Once the folder is selected, BIDS Manager reads the data2import and asks the user to verify the files that need to be imported.

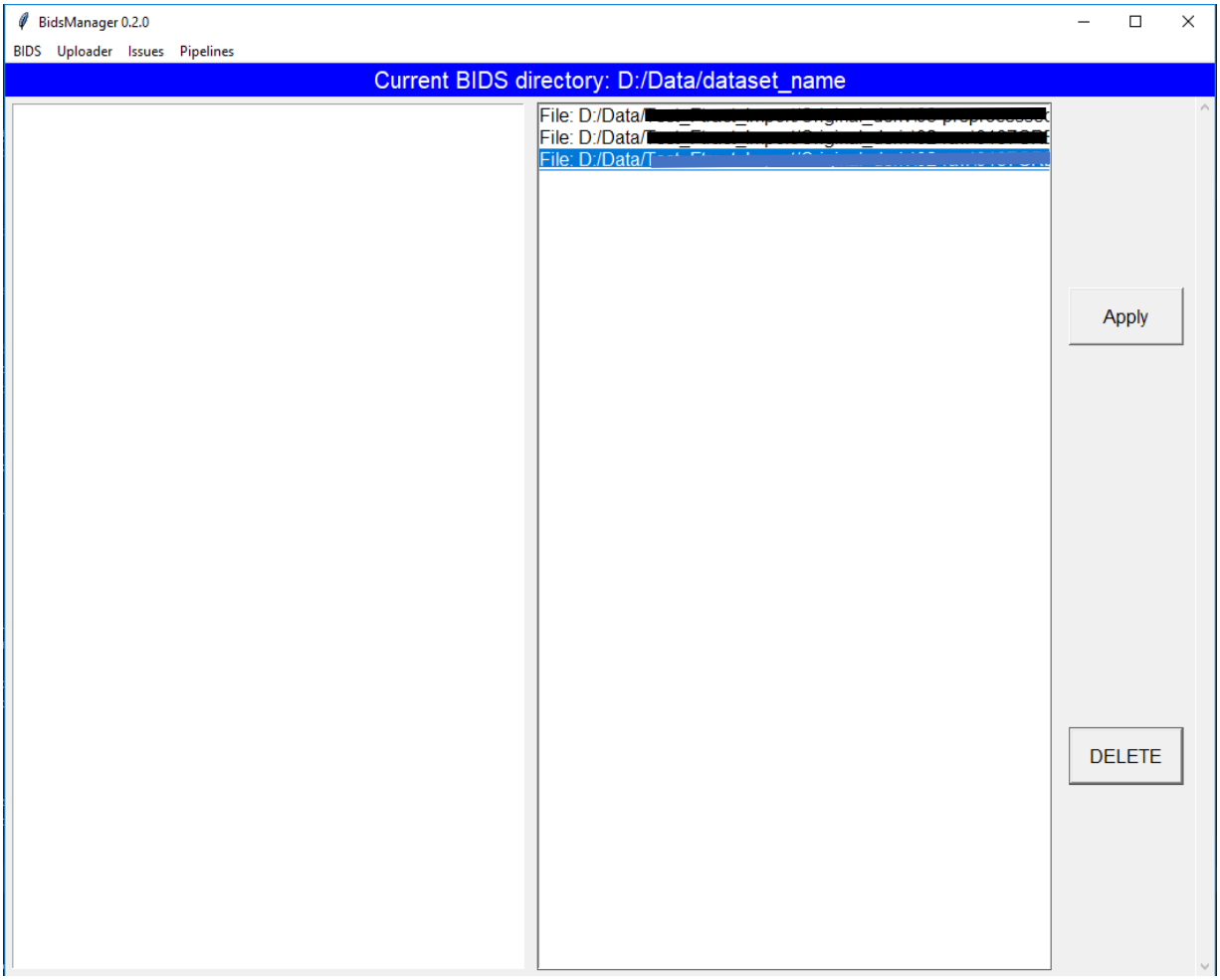


Figure 26: Verification interface - BIDS Manager

The reconfiguration of the main window (Figure 26) allows the user to run some verifications and apply some actions. The main window displays: "/import directory/fileLoc is not verify". By double clicking on this line, a drop-down menu appears to select the adequate actions.

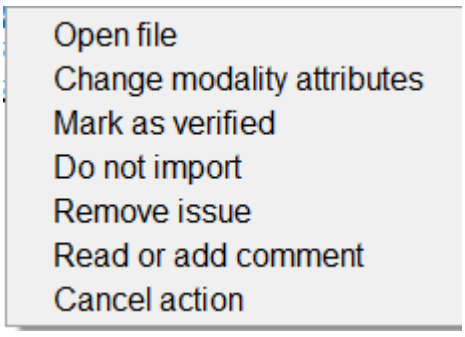


Figure 27: Action Menu

To run the importation, files must be "mark(ed) as verified". This step is set to ensure that the importation of the file is made with the corresponding attributes and that the file is not corrupted.

However, before applying this action, the user can do other actions: add comment or open the file. The action "Change modality attributes" allows the user to modify some attributes before import. Thus, a new window is displayed with the attributes corresponding to the modality (figure 28).

The attributes in red are required.

For some attributes, a drop-down list is present with the possible values. Those values are the ones already present in the database. The one present in the data2import can be modified for each data file and BIDS Manager will modify the data2import.json file.

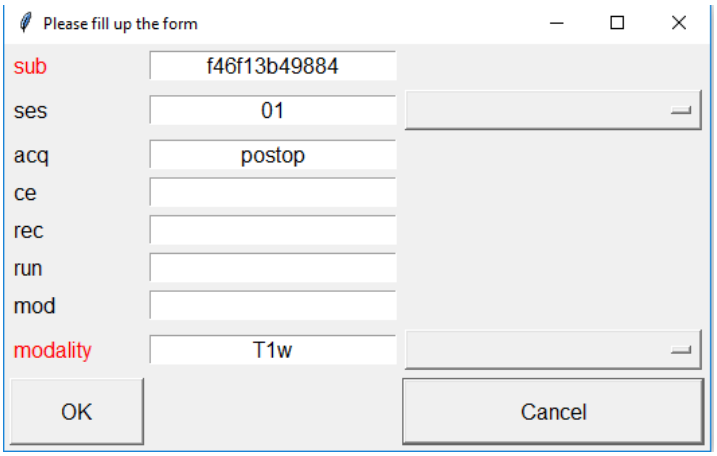


Figure 28: Interface to modify attributes

Once the user has specified the actions to be performed on the data, he/she can either apply these actions or delete those actions. Once all data have been verified, they can be imported.

However, if the user has forgotten some data, he/she can add it by selecting the corresponding directory and the data2import. To do this, the option "add element to import" must be used. It can be found in the *Uploader* menu which becomes enabled when a download directory is selected.

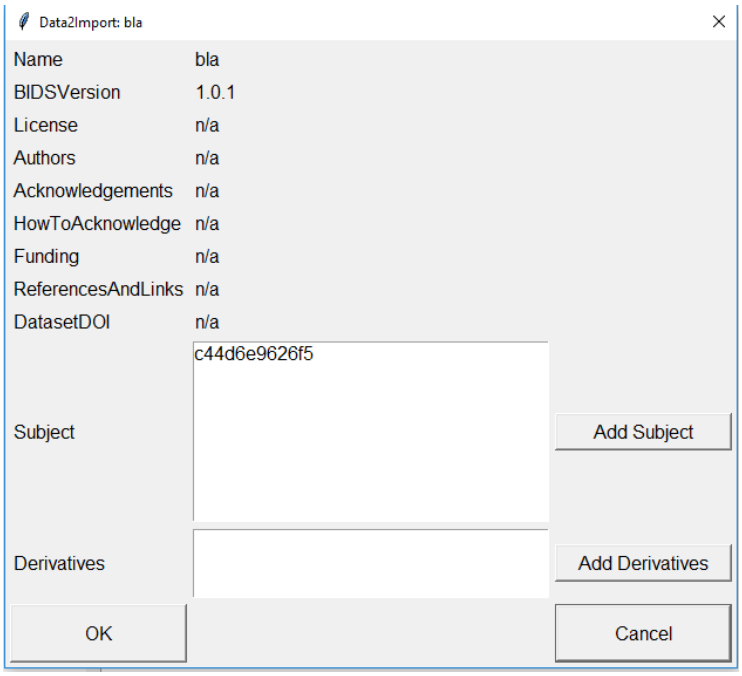


Figure 29: Interface to add elements in data2import

This new window permits to select the subject for the new elements (figure 29). By double clicking on the subject's code, a new window appears that allows indicating the characteristics of the subject and selecting the new modality to add (Figure 30).

Multiple windows pop-up to fill the attributes of the file with its location (Figure 31).

For some attributes, a list is available in order to select a value that already exist in the dataset.

Once the user validates the upload directory and data2import, he/she can click on "Import" in *Uploader* menu and BIDS Manager will import the data in the appropriate folders.

The main window will display the progress of the importation and whether an error occurs. The progress can be retrieved in the log file located in derivatives/log of the BIDS directory.

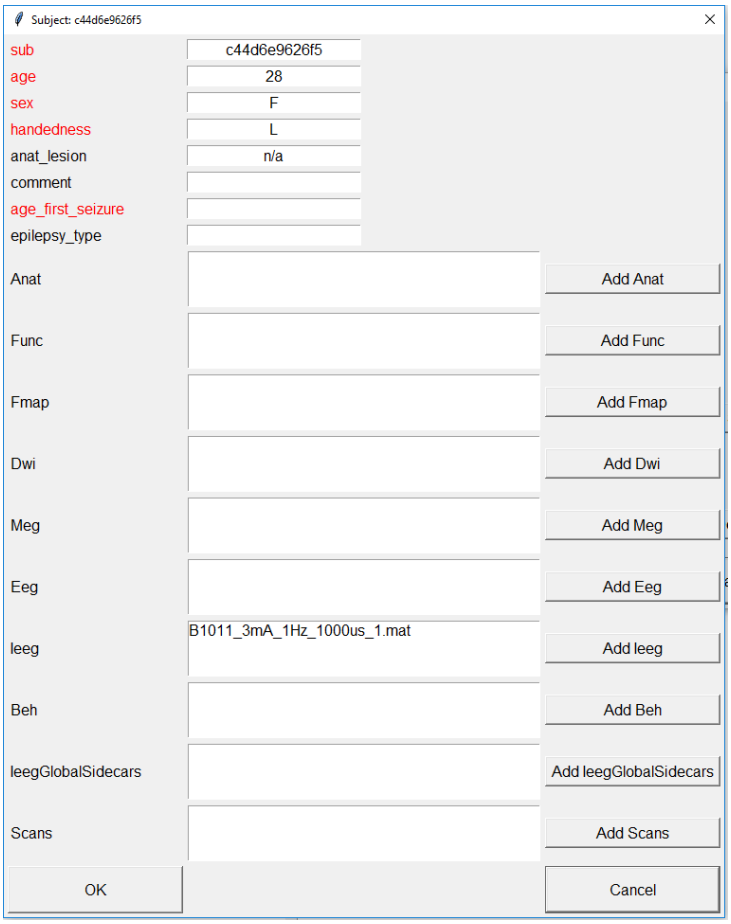


Figure 30: Interface to add elements in data2import

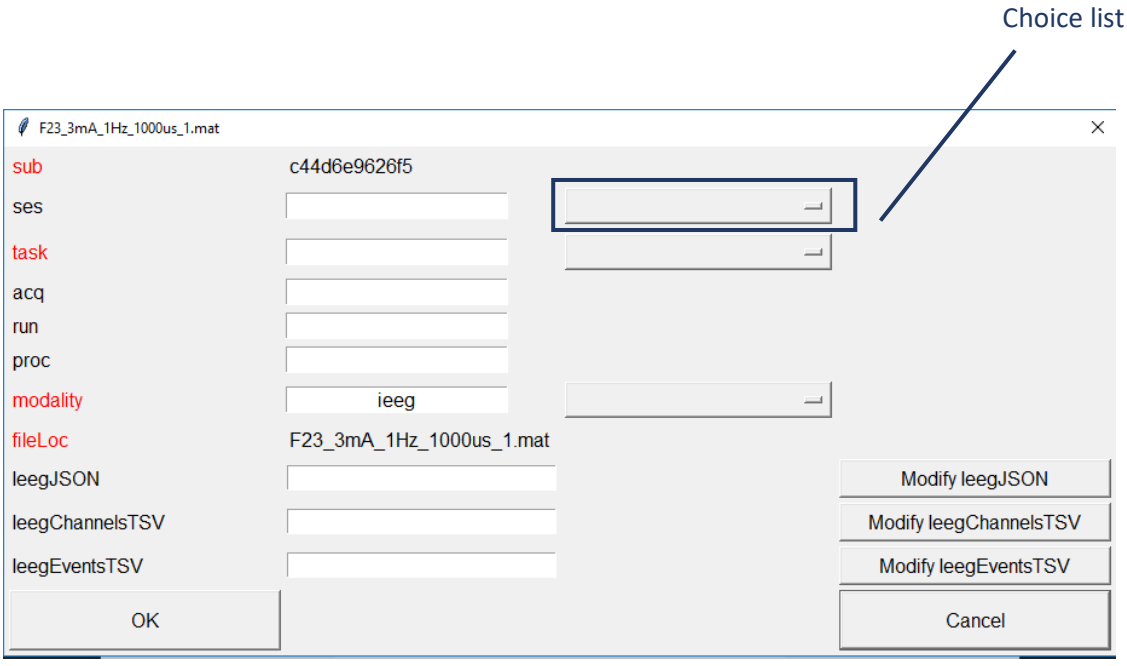


Figure 31: Interface to add elements in data2import

4.2.3 Create data2import template

This option exists in the version 0.2.6 and higher. It permits to the user to create a data2import template in a specific directory with the key to fill in to import specific modality. It is an alternative, if user can't or don't want to use BIDS Uploader. However, once the template is created, the user has to open it in an editor to fill in the attribute's values, and the location of the file.

Select the option "Create Data2import" template in *Uploader* menu. The user will have to select the path of the folder containing his data that he wants to import. An interface will open to add your subject, modalities, and derivatives.

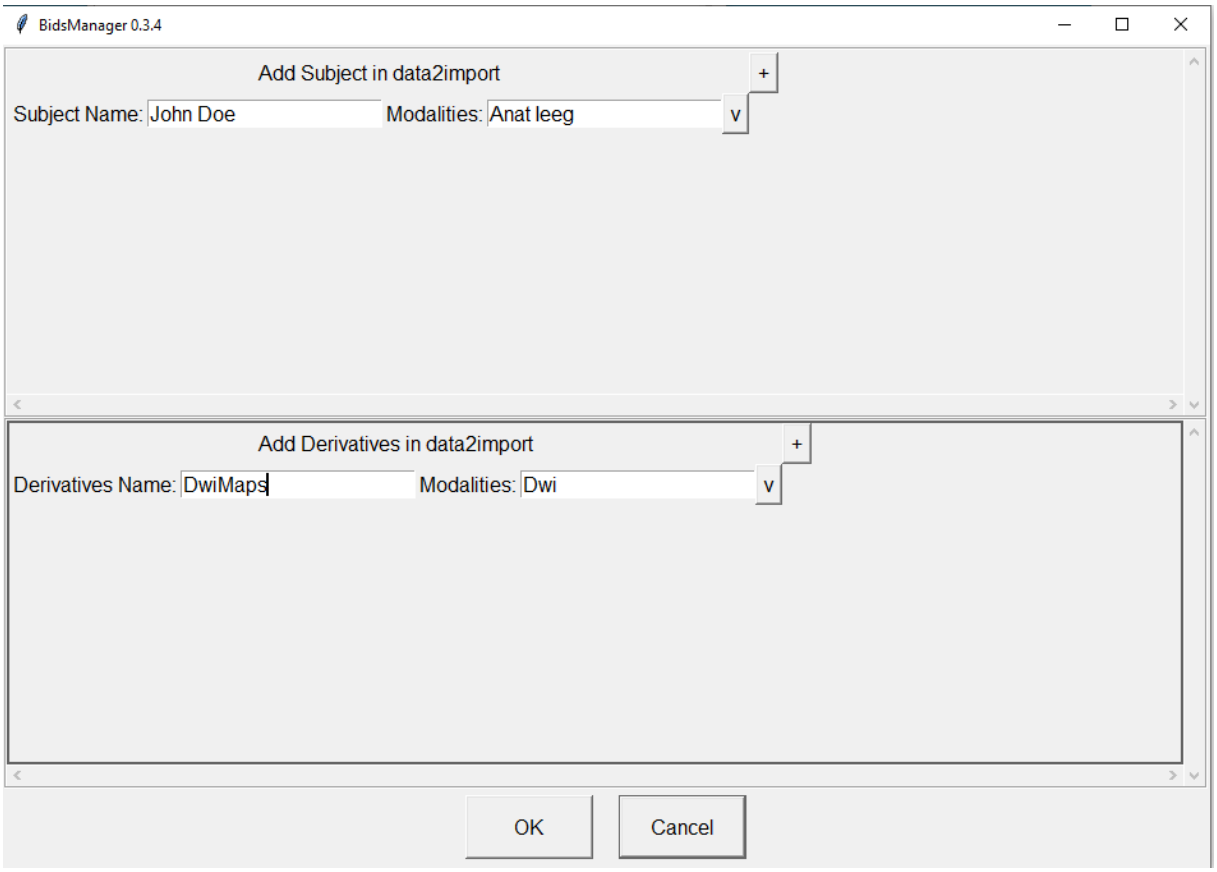


Figure 32: Interface to create data2import template

Click on '+' button to add subjects or derivatives according to the panel (see figure32). Then indicate, the names of your subject (or derivatives) and the modality that you want to import. By clicking on 'OK', the data2import.json will be written in the folder that you selected. Then, you have to edit with the attributes in an editor before to submit this folder in the option "Set Upload directory" (see section 4.2.2).

The data2import.json and the files to import must be saved in the same folder.

4.3 Explore BIDS Dataset

Once the BIDS directory has been filled, it is time to explore what is present in this directory. In BIDS menu, there is the option: “Explore BIDS dataset”.

The figure 32 shows a new window pop-up displaying the information present in the

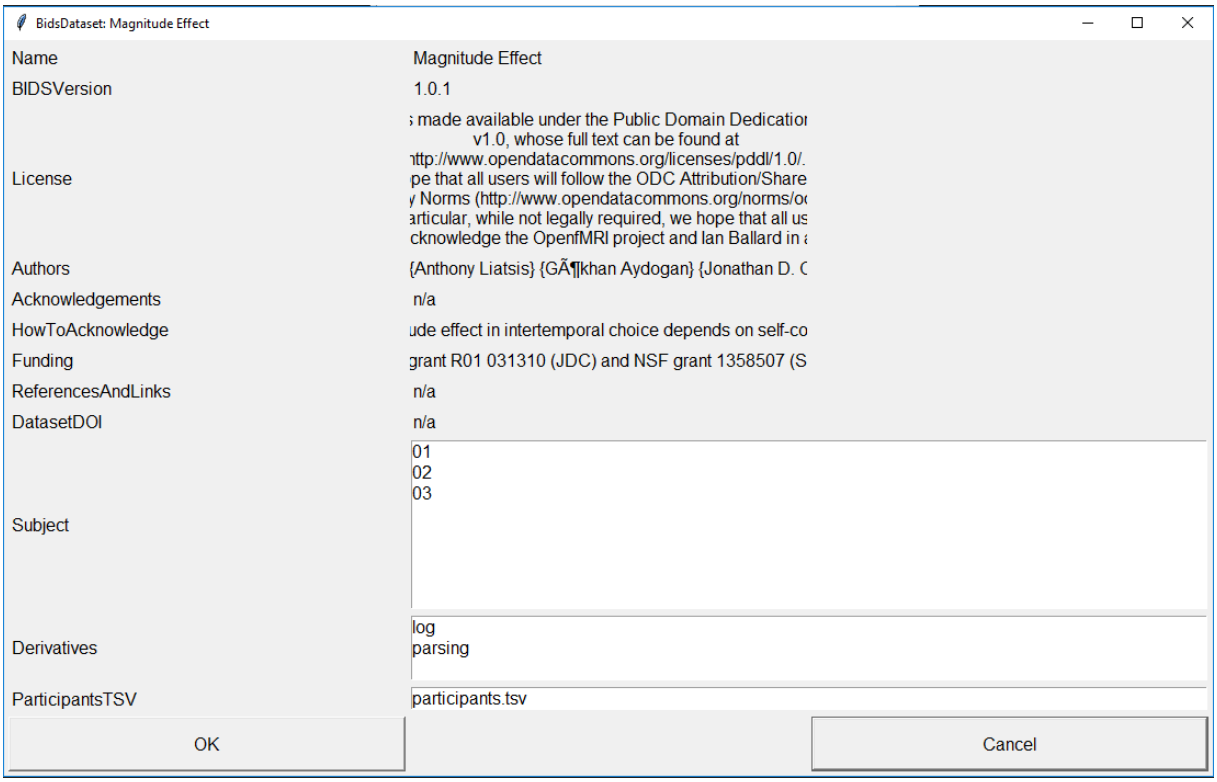


Figure 33: Explore Bids Dataset

dataset_description.json, the subject folders, derivatives folders, and the participants.tsv file.

By double clicking on one subject or one derivative, a new window appears to show the data present and the subject information.

To see what data are present for one subject, double-click on the chosen subject ID and a window will appear with the files present for this subject, classified by modality. (Figure 34)

By double clicking on one file, one menu will appear offering different actions:

- Open file
- Show attributes
- Remove file

Open file will open the file with the appropriate software. Show attributes open a new window to display the specificity of the selected file and its sidecar files (e.g. events, channels). (Figure 33)

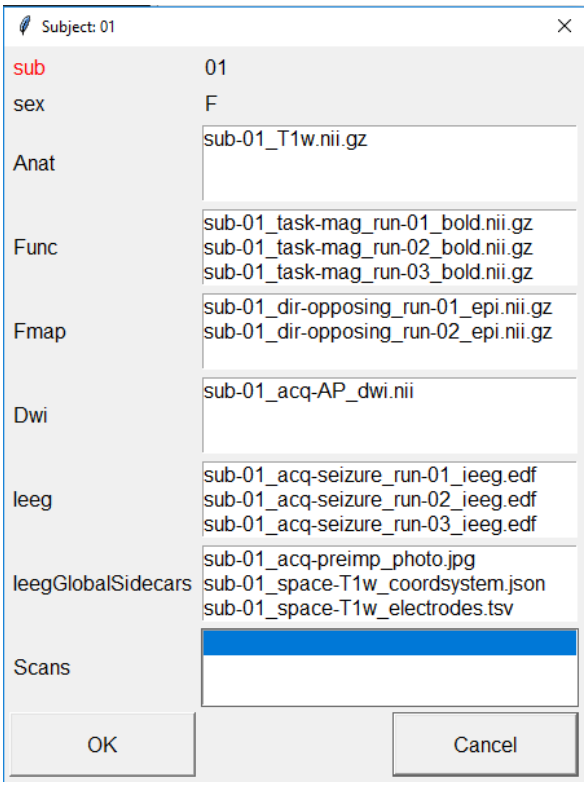


Figure 34: Subject's dataset

The window shows the attributes of the file, its location, and the different sidecar files. For example, for an iEEG file, three files are written: the channels, the events and the json file. It is possible to open these files in BIDS Manager.

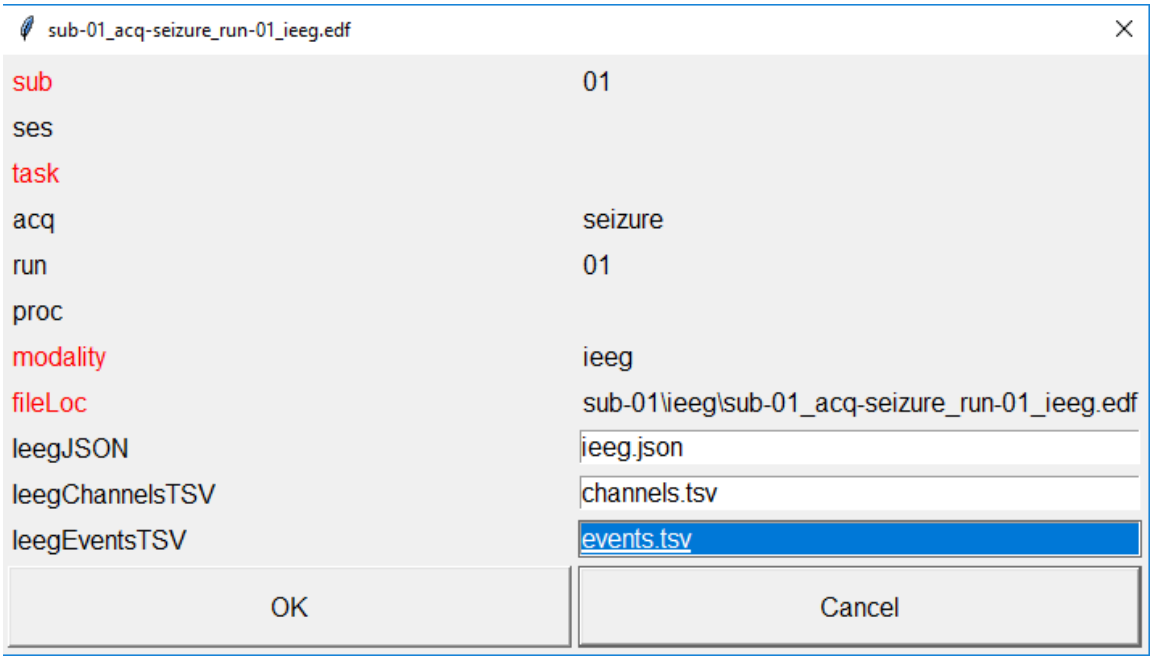


Figure 35: Attributes of one iEEG file

Double clicking on the sidecar file opens the file. Some tsv files have the particularity to be modified from BIDS Manager GUI. The files concerned are participants.tsv and channels.tsv (file for the electrophysiological modalities).

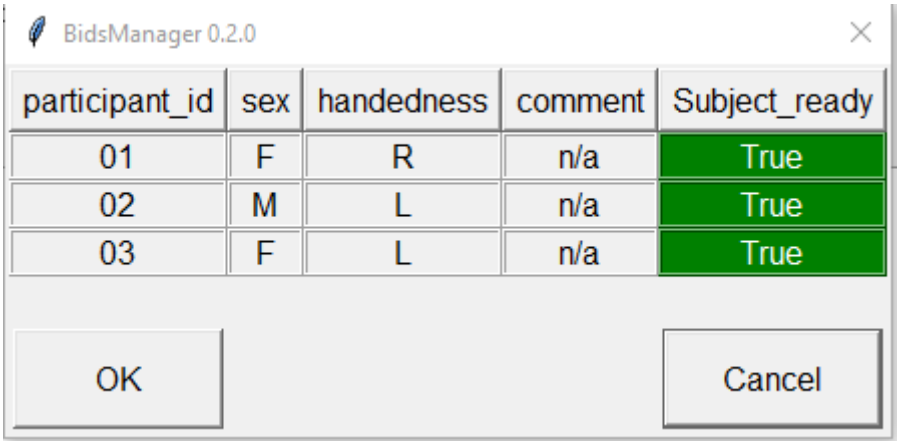


Figure 36: Participants.tsv file opened with Bids Manager

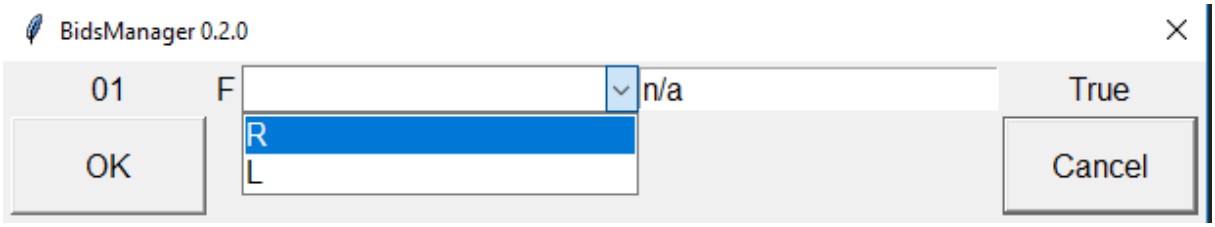


Figure 37: Interface to modify a line of the participants.tsv

For the participants.tsv, one line can be modified by double clicking on the participant_id of the subject and select “Modify the participant’s line”. A new window opens with the line selected and the elements that are possible to modify. If values are fixed in the requirements, the user must choose from the list.

It is also possible to open the subject dataset by double clicking on the participant_id and select “Open subject dataset”.

The way to open and modify channels.tsv is pretty similar to participants.tsv except that only the status of the electrodes can be turned in bad or good by double clicking on it.

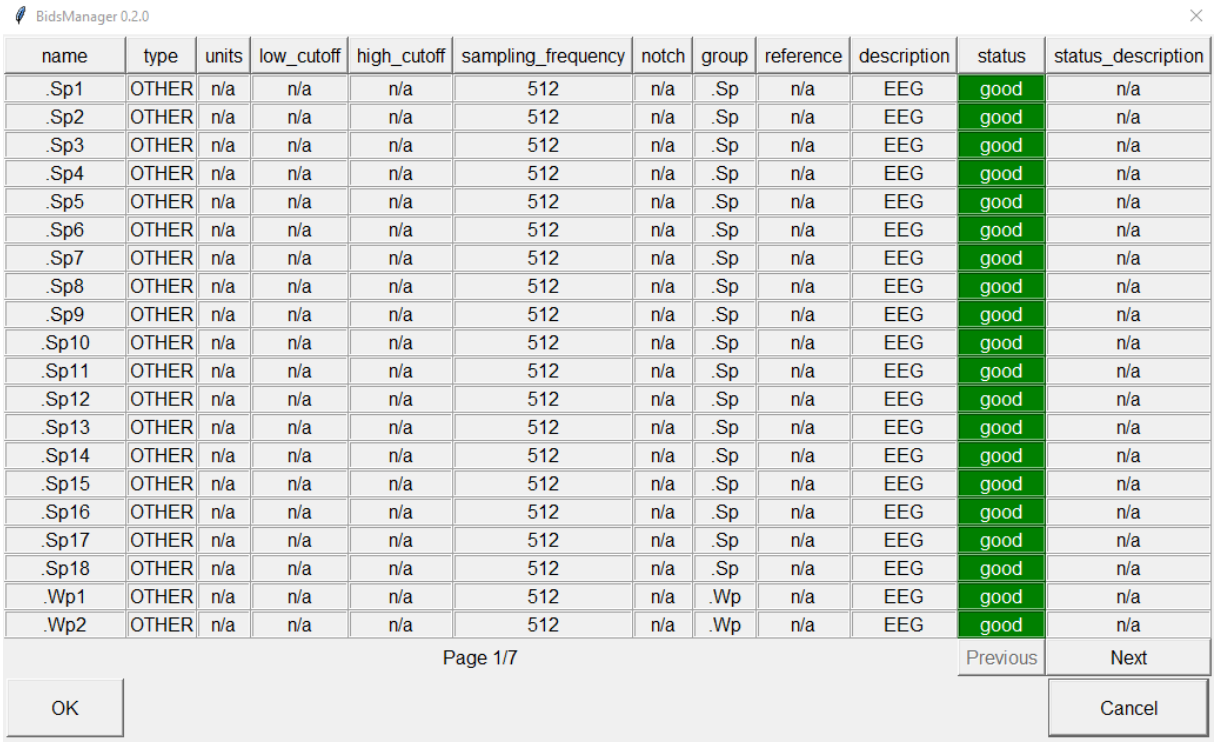


Figure 38: iEEG channels file opened in Bids Manager

Explore BIDS dataset is a good way to visualize what is inside the dataset and if necessary, do some modifications.

4.4 Launch a Process on the dataset

On this section, user will be able to run a process on specific subjects of the dataset. First step is to select on *Pipelines* list the module that user wants to apply on the dataset. Then, a graphical interface will appear to let the user choose the subjects to analyse and the parameters (Figure 40).

The first frame represents the subject selection. There is four modes, the first is all subject, the second is to let the user select the subjects by ID, the third is to select the subject by criteria. For example, user can choose to have the subject aged between 10 and 40. BMP will select the subject corresponding to this criterion for the analysis. The criteria offered are made according to the *participants.tsv* of the dataset. The last is to select subject that are not present in a previous analysis made by BMP. The user selects the folder corresponding to the analysis in derivatives folder and BMP will select the subjects of the dataset that are not present in this folder and fit the criteria of the analysis. Moreover, the parameters will be automatically updated to correspond to this previous analysis.

 The GUI refresh every 3 second until you finish to select the subjects. So, once you have finished your subject selection, wait 3second before to select the parameters

The second frame is to select the type of input and the parameters. This frame is made according to the software json file. On the left of the frame, user select what type of input he wants, either Ieeg, Eeg, Anat, etc. and he can specify the run, ses, task, etc to analyse. On the right, user can specify the value for each parameter, if he doesn't give value, the default one will be taken for the analysis. There is also an option to import the parameters value used in a precedent analysis, in that case user have to select "Select your script with parameters values" and choose the file path.

Last option, just before to say OK, it is to select a specific directory to write the results instead of writing the results in the folder *derivatives* of the BIDS dataset. The chosen directory must be empty and the results will be saved inside in the same hierarchy of

Once everything has been set, the analysis can be launch by clicking on OK. BMP will do some verification to make sure the subjects selected have the required inputs for this analysis, then BMP will run the process for each subject.

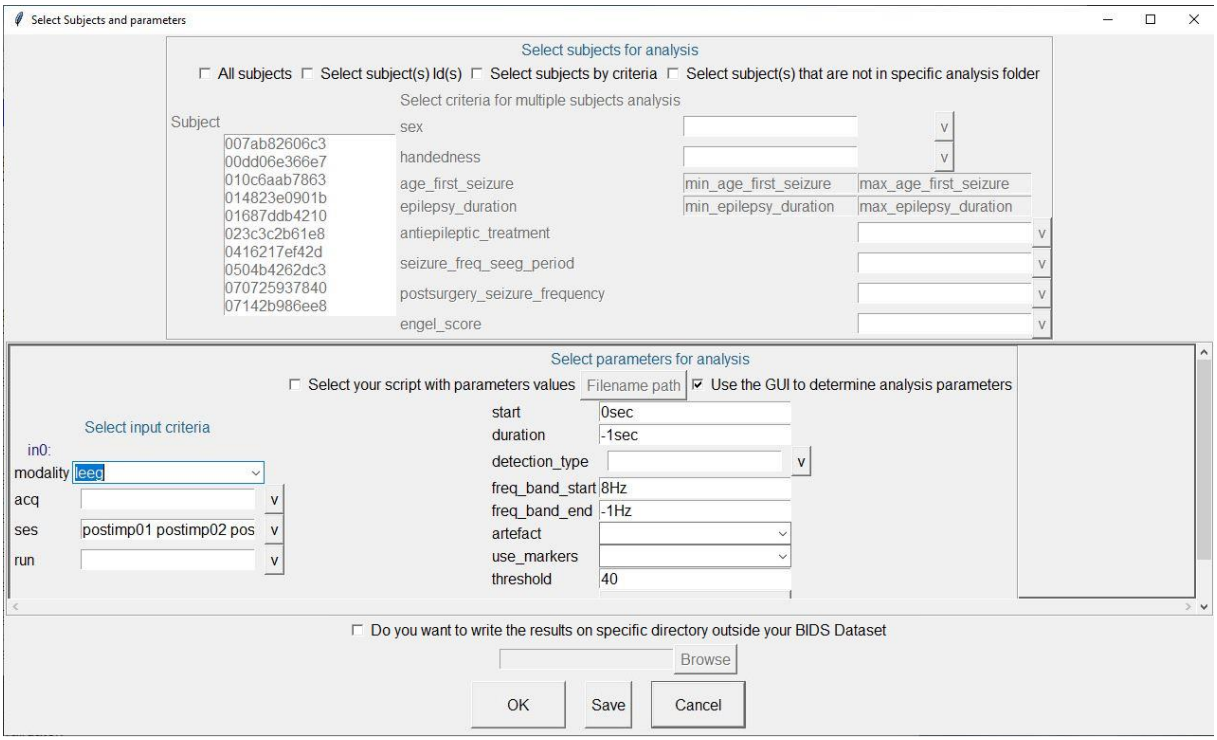


Figure 39: Interface to select subject and parameter for the analysis

4.5 Launch processing pipeline

To do a processing pipeline, user have to select "Create processing pipeline" in the Pipelines menu. Then, it opens Graphical Interface, where the user can add all the module required one after the other. To do that, user have to select he module in the list and click on "+" to add it. The others selections are similar to the one see previously (4.4)

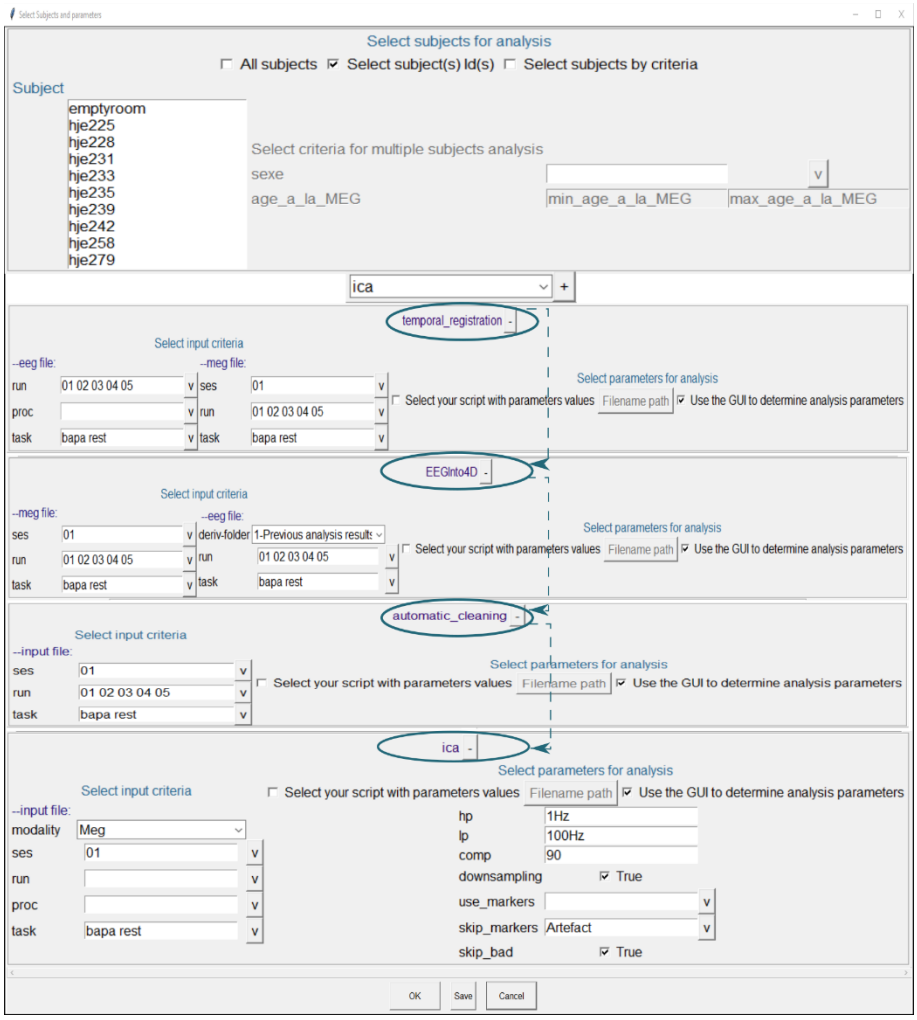


Figure 40: Interface for the preocessing pipeline

4.6 Create statistical table

The last step, of the procedure, is to create the statistical table of the dataset. The statistical table is a table regrouping the results of all Ieeg analysis by channels and by patient. For that, user has to select *Statistics*/*Statistical table* menu. An interface will appear to select the analysis that should appear in the table. (figure 39)

The frame on the right displays the *dataset_description.json* of the analysis where the parameters and the subject analysed are summarized. This tool is to help the user to do his selection. Once the selection is over, clicking on OK create the table. This table is saved in the *derivatives* folder of the dataset.

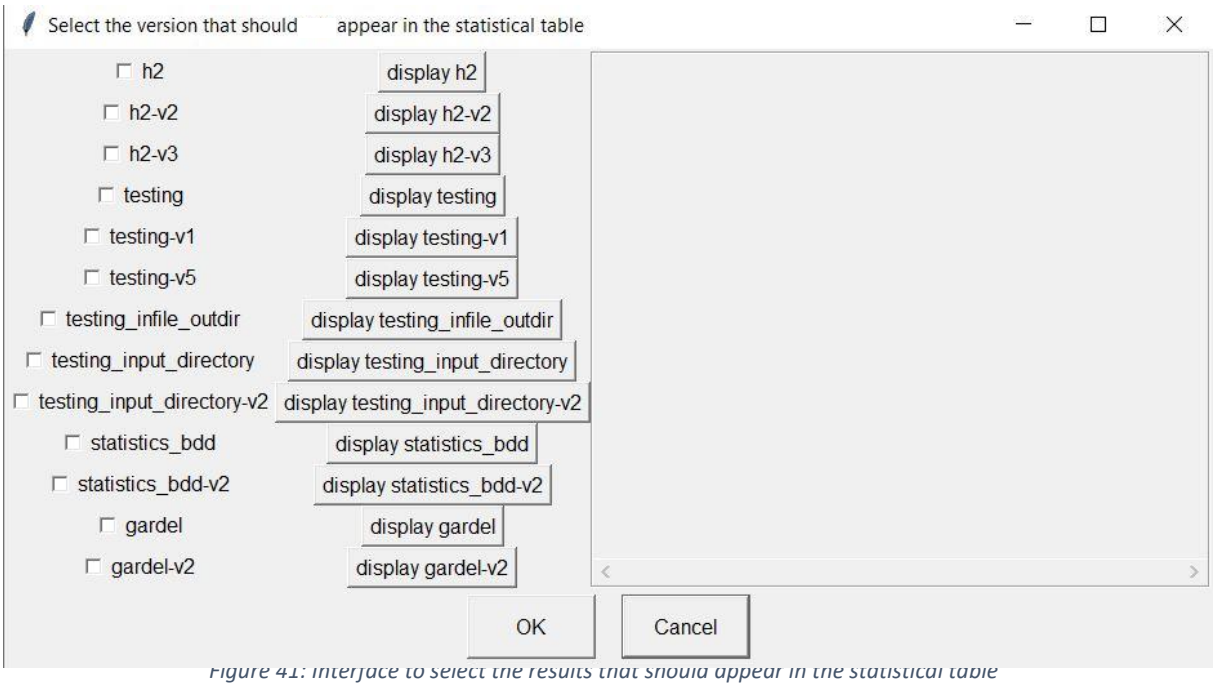


Figure 41: Interface to select the results that should appear in the statistical table

5. Issues

If an issue is prompted during the importation, it can be fixed by selecting the *Issues* section. This menu can fix three main issues. The first one is the verification of the content of the folder to be uploaded, the second is an importation issue, and the third is derived from issues with the channels or the electrophysiological data.

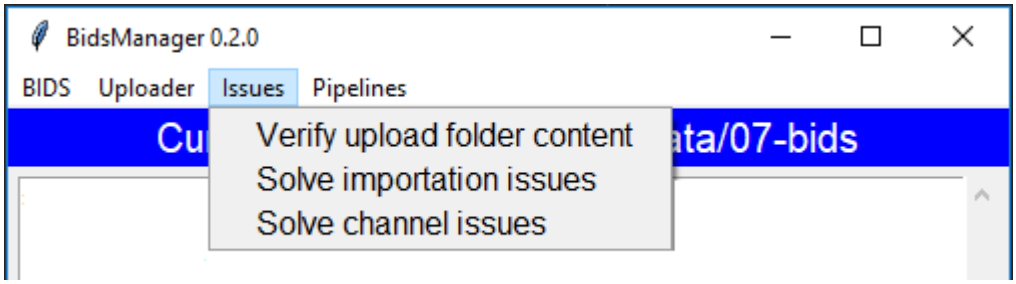


Figure 42: Issues Menu - BIDS Manager

5.1 Verify Upload folder content

This menu is useful when the option “Set Upload directory” is selected instead of “Import data with BIDS Uploader”. It allows verifying the content of the upload folder to avoid importing incorrect files and modify the data2import if necessary (see Section 4.2.2). Each file present in the folder must be ‘marked as verified’ to be able to import the folder.

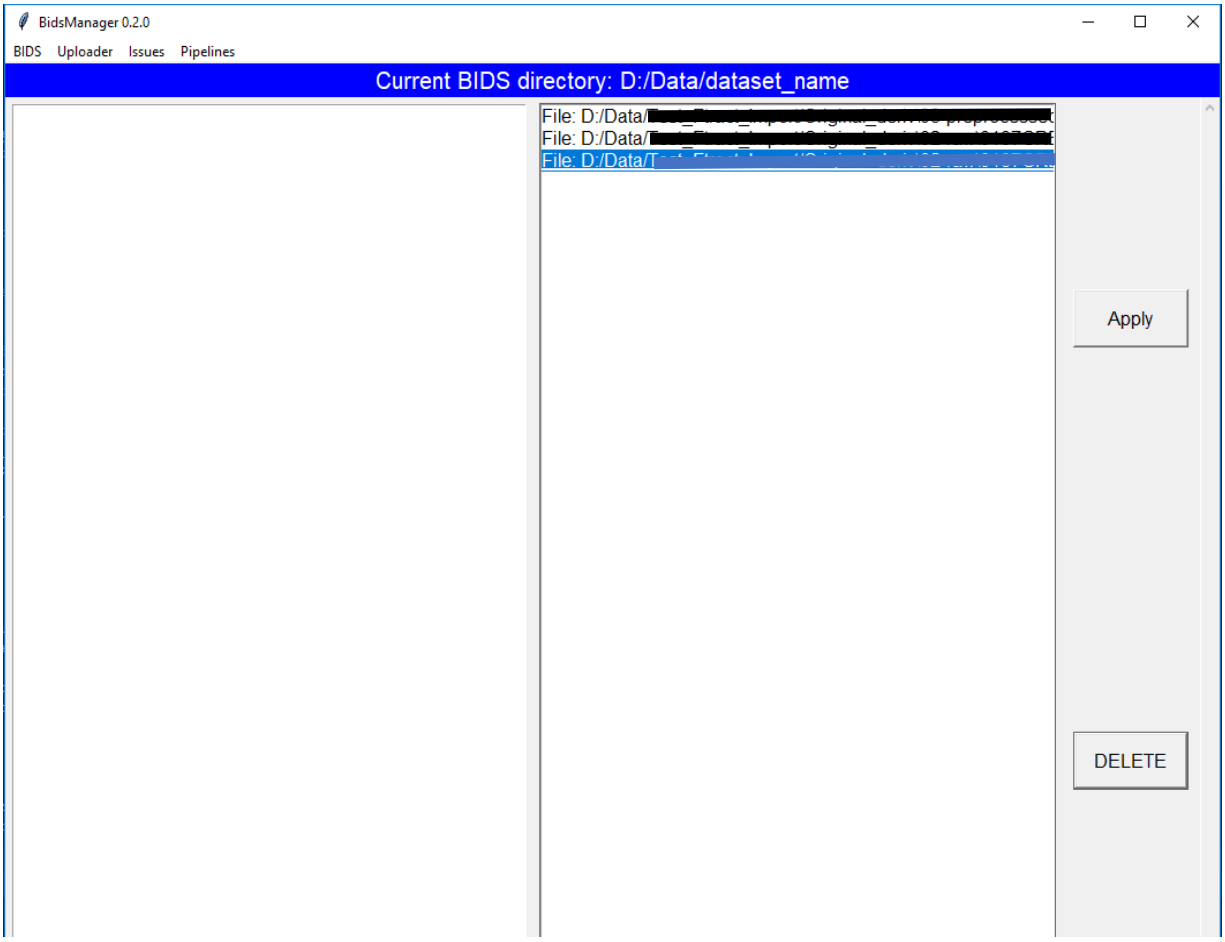


Figure 43: Verify upload folder content interface

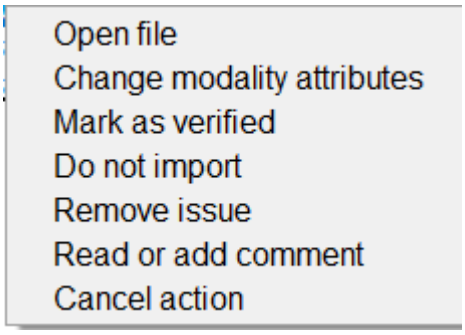


Figure 44: Menu to change issue in folder content

By double clicking on the line of the file, there are several possibilities (see below, Figure 42).

5.1.1 Open file

This option allows to open the file with the application marked as default on the computer.

5.1.2 Change modality attributes

This option is used to modify the attributes of the file. For instance, change the session from "01" to "02" or the task "rest" to "sleep". BIDS Manager will modify the data2import.json file with the values entered by the user for this specific file. However, be careful because if BIDS Manager notices that this specific file is already present in the dataset, an issue will occur during the importation.

5.1.3 Mark as verified

This option allows the user to do a double verification on the files to be imported and is mandatory before the importation when "Set Upload directory" option is chosen as mentioned in section 4.2.2.

5.1.4 Do not import

This option allows removing a file of the data2import so BIDS Manager will not import. It can be used after an importation error, either the "already present" or the "source file with the same name". This option does not remove the file from the upload folder, only from the data2import file.

5.1.5 Remove Issue

Sometimes, while an error occurred during the importation, some issues that have been solved stay in BIDS Manager. This option has been created in this purpose, the issue can be removed from the dataset issues and the importation can be done.

5.1.6 Read or add comment

This option is useful when different users manipulate the BIDS dataset. Users can write a comment to the issue and wait for a discussion before applying an action.

5.1.7 Cancel action

This option is used to cancel a previous action. For example, if the action of "Mark as verified" is applied, by using "cancel action", the issue will be back to its original state.

5.2 Importation Issue

After the importation is over, it is possible that the program finds mismatches between the dataset and the data to import. The encountered issue can be:

- Wrong protocol: Not the same dataset name
- Wrong subject characteristics: The attributes of the patient are different between the one in the dataset and the ones in the importation file
- Same source data: the subject's folder already contains a source data with the same name
- Files with same attributes: the user is trying to import a file with the same attributes as a file already in the dataset

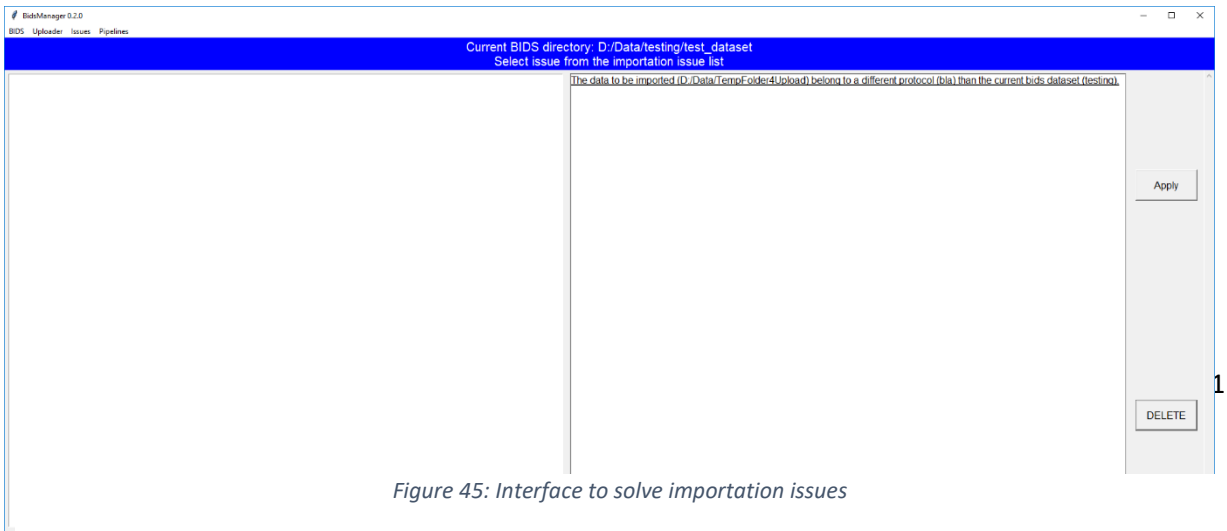


Figure 45: Interface to solve importation issues

Clicking on “Solve importation issues” will display the different issue in the main windows as it can be seen in figure 44. Different actions can be applied depending on the issue.

5.2.1 Wrong protocol

This issue is due to an incoherence between the protocol name of the BIDS directory (dataset_description.json) and the name given in the data2import (DatasetDescJSON).

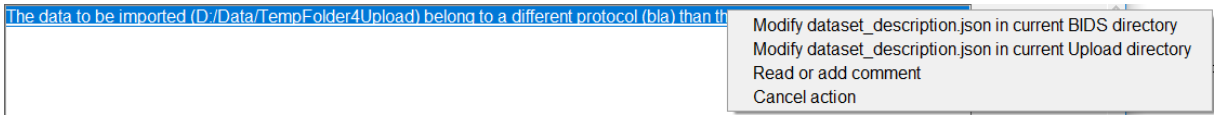


Figure 46: Possible action to apply to solve importation

By double clicking on the issue, BIDS Manager offers to do some modifications. The protocol name of either the BIDS dataset or the data2import.json file can be changed. Comment can also be added which will not solve the issue but can be useful to let another user know about the problem.

5.2.2 Wrong subject characteristics

This issue is due to a mismatch between the participant characteristics in the dataset and the one entered in the data2import file. The method to apply the action is similar to the precedent one. There are two options: modifying the data2import with the value in the dataset or modifying the value in the dataset.

5.2.3 Same source data

This issue states that the user is trying to import data that are already in the dataset. User can decide not to import this file by double clicking on the issue.

5.2.4 File with same attributes

This issue shows that a file with the same attributes as the one the user is trying to import is already in the dataset. The attributes of the file can be changed in the data2import to solve the issue by double clicking on ‘Change modality attributes’ (section 5.1.2).

Once none of the above issues remain, you can launch the importation procedure again via the ‘Set Upload directory’ (section 4.2.2).

5.3 Channels Issues

The importation went well but some warnings appeared concerning the channels of electrophysiological files.

BIDS Manager verifies inside the subject's dataset (for a given session) that all IEEG/EEG files have the same electrode names as in electrode.tsv tables, it also verifies the type of the electrode. The possible issues are:

- Channel type is not conformed
- Channel name is not conformed

Clicking on the function "Solve channel issues" will display the channels issues in the main window. If the name or type of the electrode is wrong, the user will have the following message: "In file X of subject X, {electrode name} does not match electrodes.tsv reference."

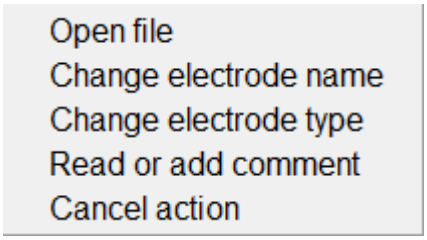


Figure 47: Menu to apply action on channels issue

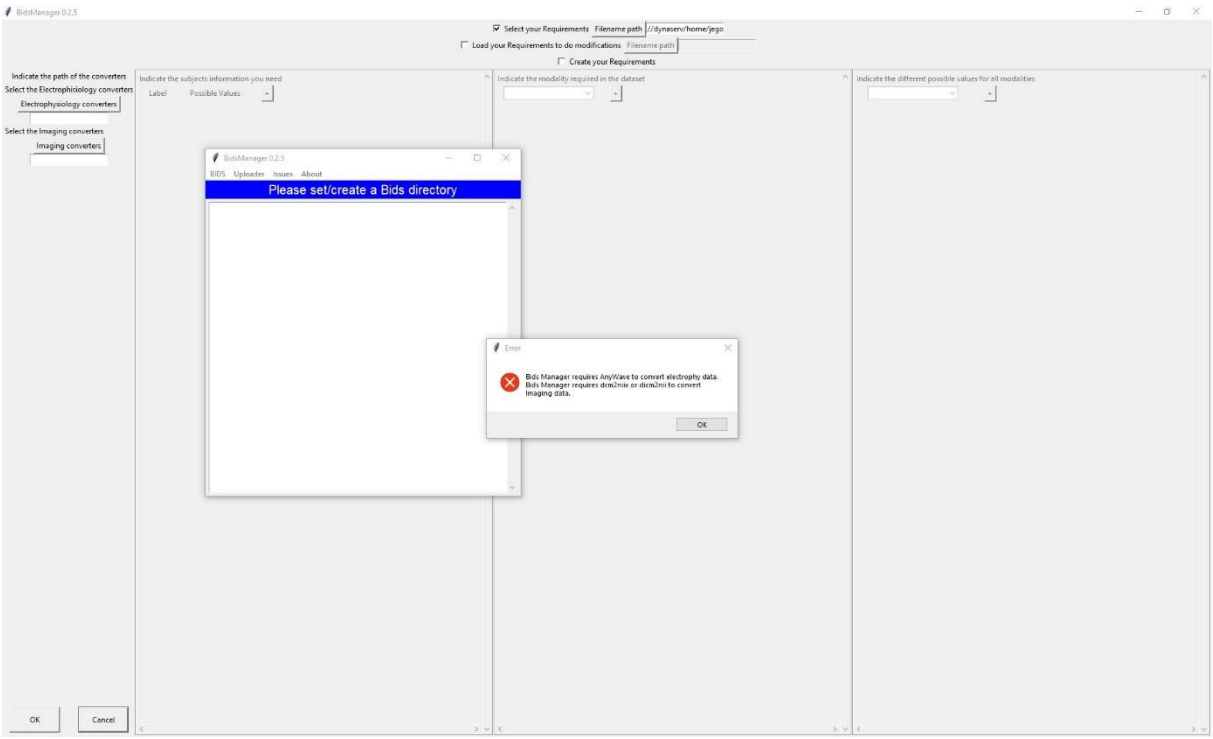
Double clicking on the issue will offer two options, changing the name or the type of the electrode. Then, it will open a new window and suggest the channel name or type.

6. Error handling

This section will describe the possible errors that users could encounter by using BIDS Manager and give the solution to solve them.

6.1 Errors with requirements

Once requirements have been loaded or created, the user has to indicate the paths of the electrophysiological and imaging converters. Otherwise, the error in figure 46 will appear. To solve it the user is required to add the path in the left part of the GUI.



6.2 Error with importation

During the importation of the data, some errors, which are not expected as the issues

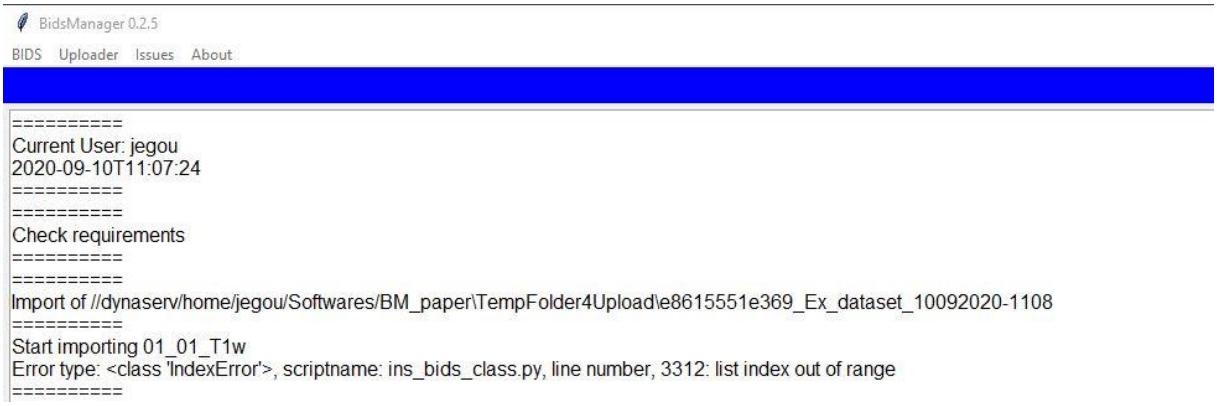


Figure 49: Error during importation

described above, can occur. The following sections will detail the procedure to overcome these errors.

6.2.1 Error with the converters

An error can come from the converters used. If BIDS Manager gives the error illustrated in figure 46, it means that the converters didn't convert the data properly and BM cannot find the converted file. To solve it, the user should change his converter. For the imaging converter, make sur to use the *dicm2nii.exe* provided on the GitHub (https://github.com/Dynamap/BIDS_Manager). For AnyWave, the problem can also come from the version so do not hesitate to use a previous version (if you have one) or contact us at: christian.benar@univ-amu.fr

The antimalware or firewall can also block the conversion, the user must give the permission to the converters, in the corresponding software, to execute it on the computer.

6.2.2 Errors import anatomical data

If the imaging converter cannot convert the DICOM folder, it is possible that some NIFTI files are present in the DICOM folder. The NIFTI files must be removed for successful conversion.

6.2.3 Errors with 'Importation Issue'

Sometimes after an importation that failed or stopped, the Importation Issues keep some information about this importation. This can be problematic. Removing those issues by using BIDS Manager menu Issues/Solve Importation issue (see section 5.1.5) will fix it.

6.3 Manual Errors

In some cases, if the dataset has been modified manually, an error will occur in BIDS Manager. For example, a file has been erased or modified manually. To solve the error, a "Refresh BIDS dataset" must be done. If the BIDS Dataset cannot be opened at all due to a bad parsing of the data, the parsing files must be erased in the folder *derivatives/parsing*, then another parsing will be done on the dataset once the BIDS directory is set.

⚠ It is important to avoid any manual changes, it is better to do it through BIDS Manager.

6.4 Error "already in use"

BIDS Manager has been created to minimize errors in manipulating the dataset. One of the implementations is to ensure that only one user at a time has access to the BIDS dataset. If another user tries to access to one specific BIDS dataset and it is already used by someone else, a pop-up window will appear a signal which user is working on the BIDS dataset (figure 49).

Figure 50: Error access

This error can also occur if BIDS Manager has been unexpectedly closed. To solve the problem, and if the user is certain that no one is using the dataset, the file *access.json* in */derivatives/log* folder has to be removed.

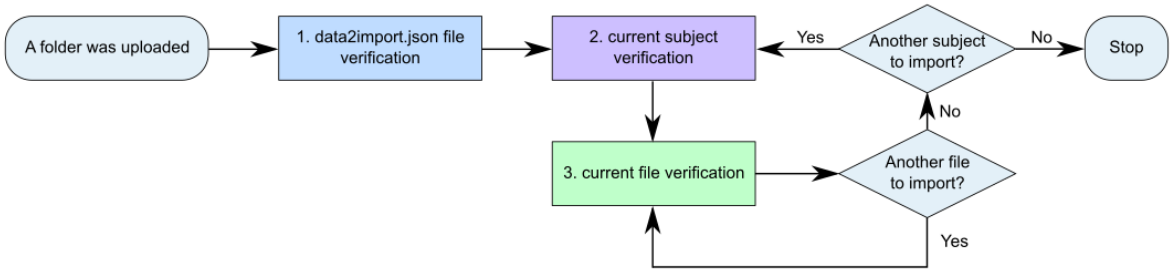
References:

Gorgolewski, K. J., Ghosh, S. S., Das, S., Calhoun, V. D., Hanke, M., Poldrack, R. A., … Schaefer, G. (2016). The brain imaging data structure, a format for organizing and describing outputs of neuroimaging experiments. *Scientific Data*, *3*, 160044. https://doi.org/10.1038/sdata.2016.44

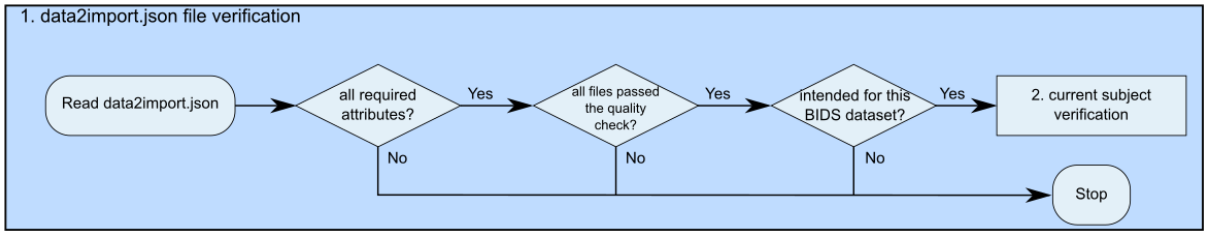
Roehri, N., Medina-Villalon, S., Jegou, A., Colombet, B., Giusiano, B., Ponz, A., & Bénar, C. G. (2021). Transfer , collection and organisation of electrophysiological and imaging data for multicenter studies. *Neuroinformatics*. https://doi.org/10.1007/s12021-020-09503-6

Annexes:

1. BIDS Manager workflow

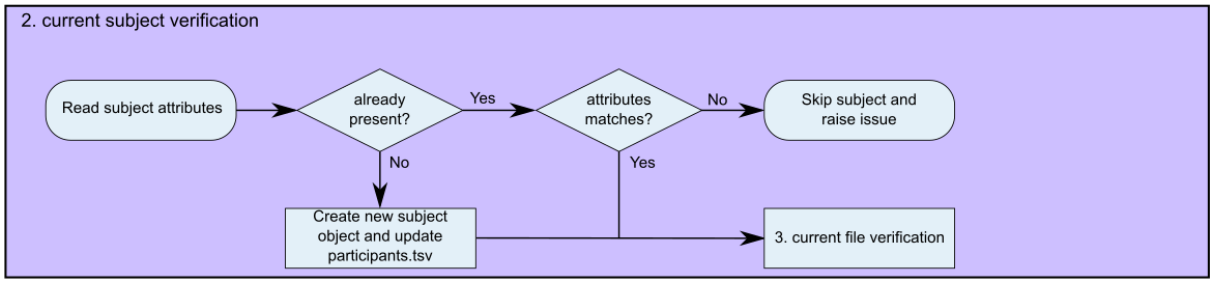


This part illustrates how the file importation works with BIDS Manager. The first diagram schematizes the main steps of the process while the following diagrams detail each step. The first step is the verification of the *data2import* file, the second the verification of the subject to be imported, and the third the verification of the files.

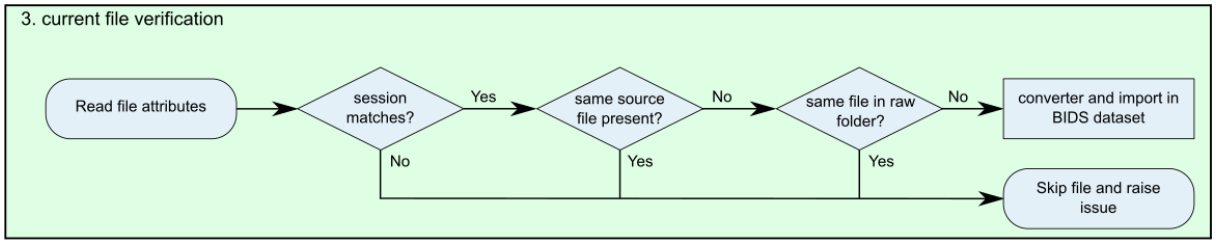


As described by the diagram, many verifications are done to check the data2import file. First, the attributes of the file must be consistent with the ones in BIDS Manager. Secondly, the files must be set to 'verified' by the user. Thirdly, the protocol name in data2import must be the same as the one in the BIDS dataset. Finally, if the verification is completed without raising an issue, it goes to the second step.

Once the data2import verifications are completed, the subjects to be imported are verified.

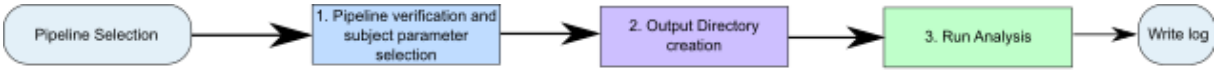


The subject present in data2import are compared to the subjects already present in the BIDS dataset. If the subject is already present, BIDS Manager checks the attributes (age, sex, etc. depending on the requirements file). If the attributes are not the same, the subject is not imported. If there are the same, BM go to the next step, "file verification".

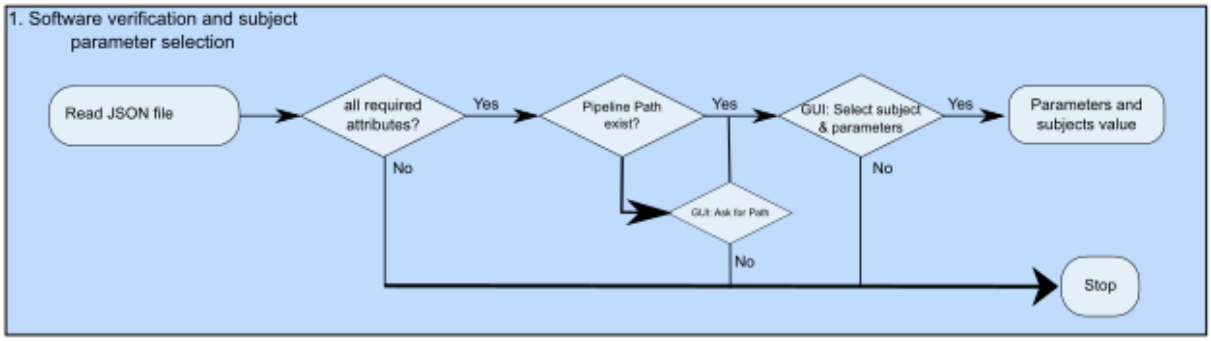


In this part, the modalities that should be imported are verified. As the precedent steps, their attributes are verified then the file location is compared to the source data present in the BIDS dataset. If the original name of the file is similar to an already present file, the import file is skipped. Otherwise, it continues and compares the attributes to create the filename. If a file with the same name is already present in the BIDS dataset, the file is not imported. Finally, if everything is correct, the file is converted in the appropriate format and imported in the expected folder.

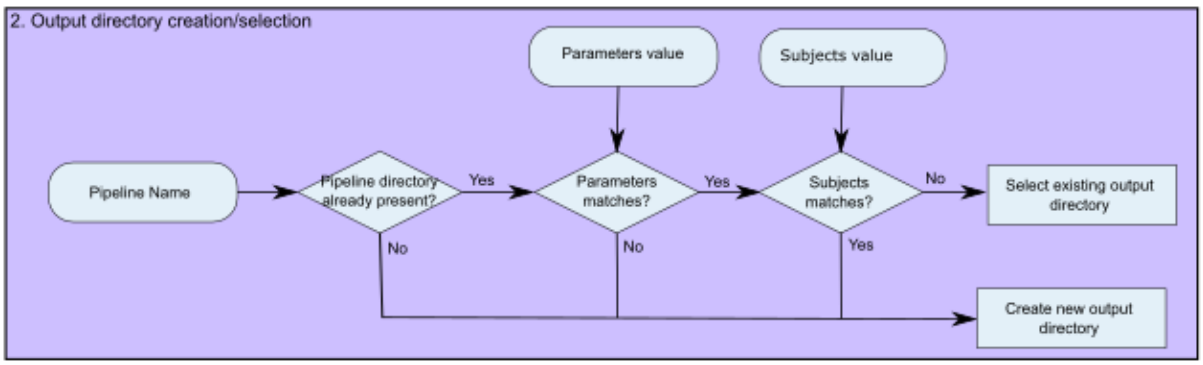
2. BIDS Manager-Pipeline workflow



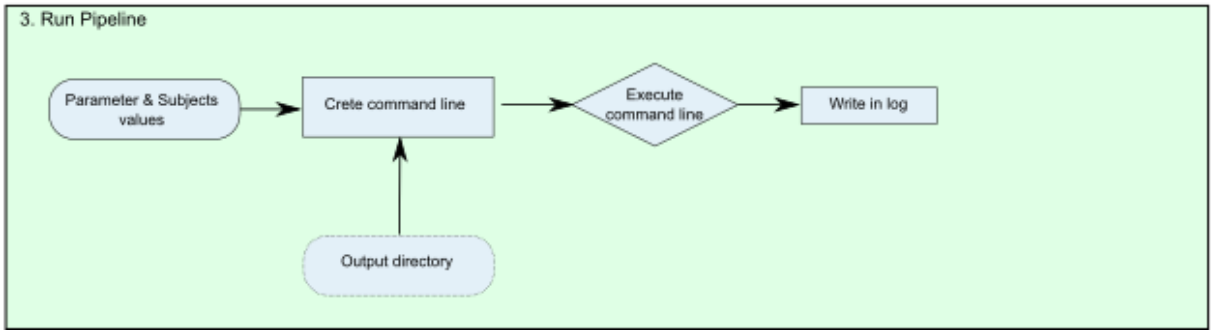
This part illustrates how BMP does the processing. The main diagram summarizes the main steps, then each step is detailed in a new diagram.



The first step is to verify the json describing the software and make sure it has the right keys. Then, the GUI is creating according to this file and the user do his selection. BMP verify the selection and that the subjects have all the required to be analysed with the parameters selected.



The second step is to select/create the output directory to save the results. BMP will verify if there is already a folder named like the module in derivatives folder. If yes, he will checks the parameters and the subject analysed to know if it can select this one for this analysis. The folder is selected if it has the same parameters and if the subjects selected are not in the folder. Otherwise, another output folder is created with a variant (-vx).

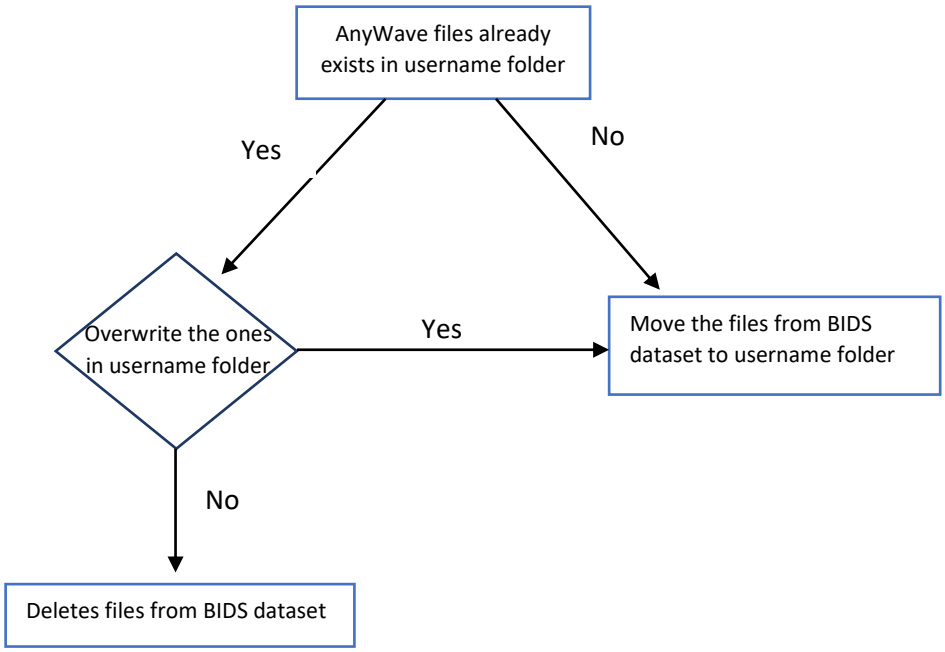


The last step is to create the command line to run the analysis according to the information gave by the json file and the user. Finally, a log is writing to give the information about the analysis to the user.

3. Use cases AnyWave

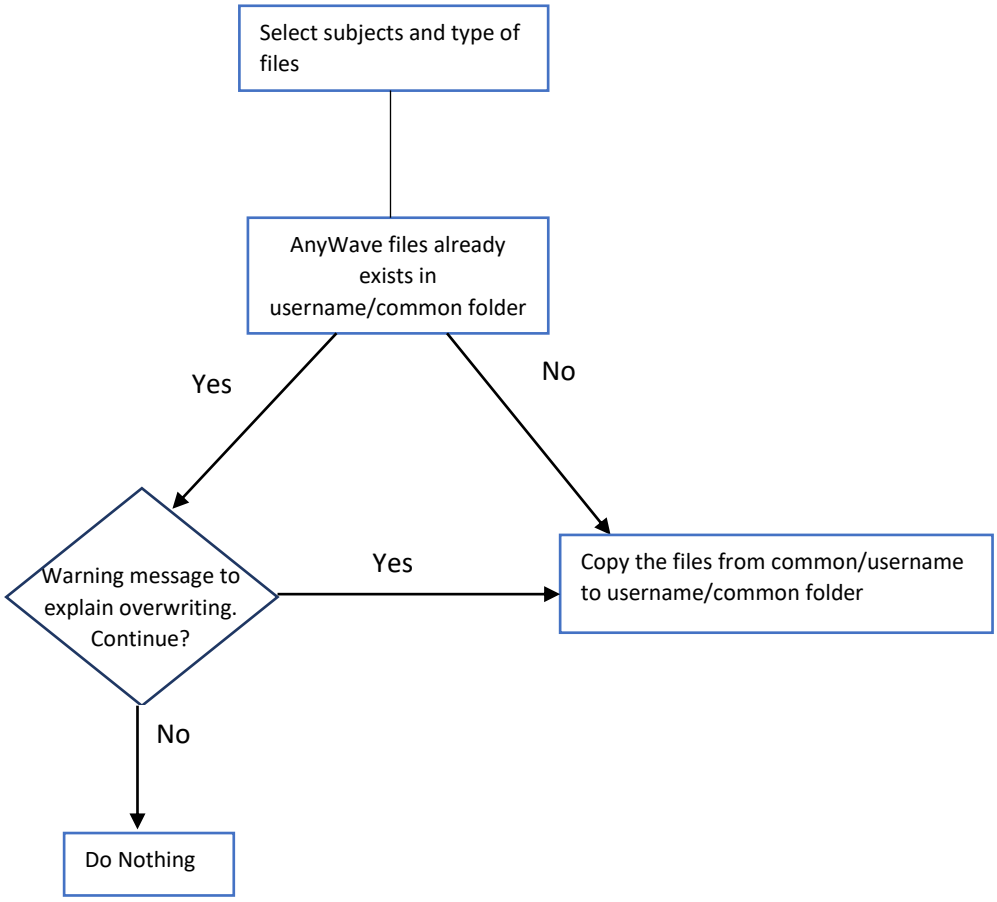
Use case: BIDS dataset must be cleaned while opening it so AnyWave files will be moved/deleted.

Solution – decision tree:



Use case: User wants to get the files from common OR user wants to copy files in common

Solution – decision tree:



Use case: User wants to do analysis with AnyWave plugin, BP has to copy the AnyWave files from derivatives/anywave

Solution – decision tree: