RaBIDS

**Rapid analysis pipeline using the Brain Imaging Data Structure**

Tutorial with example data

Christian Paret – 2020

RG Psychobiologie of Selfregulation

Department of Psychosomatic and Psychotherapeutic Medicine

Central Institute of Mental Health

[christian.paret@zi-mannheim.de](mailto:christian.paret@zi-mannheim.de)

# Introduction to this practice session

In this practice session, you will import a training fMRI-data set to BIDS-validated structure. This introduction explains the required structure of the raw data, i.e. the fMRI data and logfiles received from an MRI measurement at one of our scanners at the Central Institute. It is shown in the subsequent sections how you need to provide study and subject information to use the fully automated import and analysis routines.

Matlab, SPM12 and dicm2nii need to be installed on your computer. Consult the manual for detailed information.

## MRI images

### Naming of MRI series

An MRI measurement protocol can consist of several MRI series. For example, you may administer two functional tasks in one session, and the technical assistant (MTA) will start an echoplanar imaging (EPI) sequence for each task-run by selecting the corresponding tab at the MRI computer. It is recommended to assign unique names to each of these MRI series by changing the name of this tab. Ask the MTA to do this when you set up the sequence. You should also have information ready on the number of images per sequence.

In this training data set, two tasks were administered: a task called “scenes” and a task called “faces”. The names of the EPI-sequences were “ep2d\_TR2000\_64\_Scenes” and “ep2d\_TR2000\_64\_Faces”, respectively. Number of images for the scenes task is between 426 and 430, and number of images in the faces task is between 366 and 370. In addition, an anatomical MPRAGE scan was acquired (name: “t1\_mpr\_ns\_sag\_pat2\_iso\_asy”, 192 images).

For more information on the training task see section 6.

### Exporting MRI series

A training dataset of one participant was recorded (see section 6 for task information). The MTA who operated the MRI machine exported the images to the data exchange server (at Central Institute also known as “hobbes”) according to the structure below. Siemens machines export MRI images typically to the dicom-mosaic data format with the file ending “.IMA” (sometimes “.dcm”).

Generic structure for saving MRI images from the scanner site, suggested for your future projects (names in upper case are placeholders and need to be replaced according to your individual study). Each line corresponds to a directory. Indents indicate hierarchical structure: indented lines relate to directories that are nested within the superordinate directory :

* DATA EXCHANGE SERVER (i.e., hobbes)
  + STUDYNAME
    - dicomdir
      * sub-SUBJECTNUMBER
        + ses-SESSIONNUMBER

*here are the MRI images*

For this tutorial you will be provided with this architecture:

* data exchange server
  + RABIDS-example
    - dicomdir
      * sub-RABIDS01
        + ses-01

*RABIDS01.MR.PSM\_BI-STUDIE…..IMA*

*…*

To anticipate: If you adhere to the suggestions above in your own future projects, it will not be necessary to copy your MRI images from hobbes to your local project directory anymore. Instead, you will define the path beforehand to point to your study directory on hobbes. RaBIDS will recognize and import your data automatically from there. This saves space on the hard drive, it saves your valuable time of tedious copy pasting of MRI data and it protects for possible errors.

## Presentation logfiles

Presentation (provided by the company neurobehavioral systems) is used for stimulus presentation during MRI scanning. Future versions of RaBIDS may handle logfiles from other software, but Presentation is used in nearly all of our studies and therefore was used to develop this beta RaBIDS version.

Presentation logfiles contain task information that is required to receive stimulus functions from your expermiment, and the latter are necessary for modeling your fMRI data. The stimulus function lists the events of your experiment (e.g. onset of IAPS picture, duration of picture presentation, onset of fixation cross, …) and links them to the time when the fMRI volumes were acquired.

The data structure for saving logfiles from the scanner site is similar to the one used above, with the difference that data is saved to a directory called “sourcedata” instead of “dicomdir”.

In this tutorial you can navigate through the data structure to the logfiles as shown below:

* data exchange directory
  + RaBIDS-example
    - sourcedata
      * sub-RABIDS01
        + ses-01

*sub-RABIDS01\_ses-01\_task-faces.log*

*sub-RABIDS01\_ses-01\_task-scenes.log*

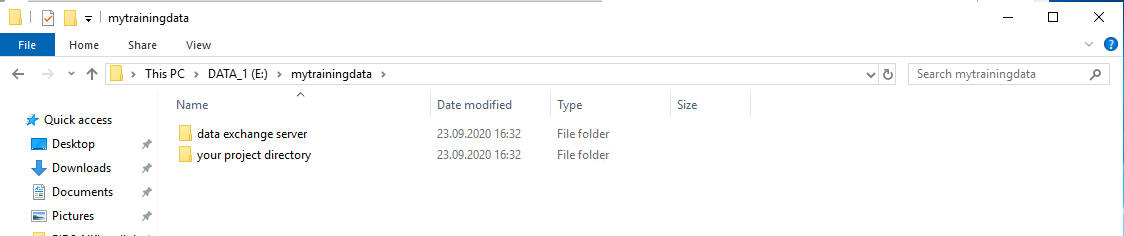
## Get training data

To get started with this example, unzip the directory called “example.zip” from [\\zi.local\flstorage\dep\_psm\share\RaBIDS](file:///\\zi.local\flstorage\dep_psm\share\RaBIDS)\tutorial to your local directory (you can call the target directory “mytrainingdata”, in this case it is named consistently with the directory used in this tutorial). To unzip the files you need to request the password from [christian.paret@zi-mannheim.de](mailto:christian.paret@zi-mannheim.de). Please do not run the practice session in the original directory – this will create a lot of useless data to crowd our hard drive.

The structure of the training data set should look like this:

* mytrainingdata
  + data exchange server (see 1.1.2 for information)
  + your project directory
    - dataset
    - sourcedata

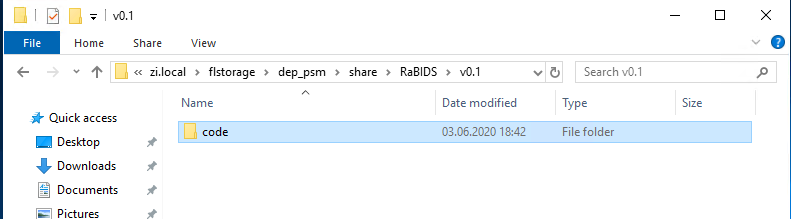
When you access the mytrainingdata directory on Windows it should look like this:



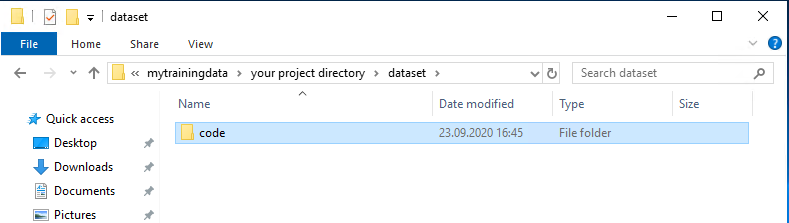
Follow the steps below to copy required data to “your project directory”:

* RaBIDS program: go to \\zi.local\flstorage\dep\_psm\share\RaBIDS, go to the directory including the latest RaBIDS release (e.g. v0.2) and copy the code-directory to the dataset directory (\mytrainingdata\your project directory\dataset\code).

This picture shows the code directory from the v0.1 release:

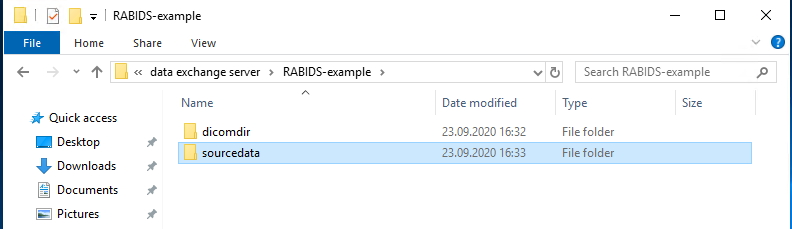


… And this picture shows the code-directory where should copy-paste it:

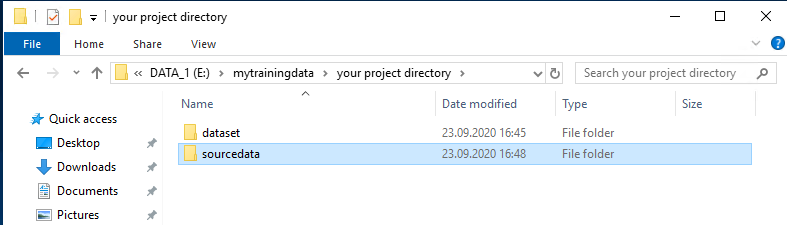


* Sourcedata: copy the sourcedata folder from \\data exchange server\RABIDS-example to your project directory.

Picture showing where to find the sourcedata directory:



… And you need to copy-paste it here:



This latter step of copy-pasting sourcedata seems a bit odd. Nonetheless we incorporate it here because you will be required to copy-paste sourcedata from your own future projects (i.e. the subject’s logfiles) from hobbes to your project directory by hand, following the described procedure (will be changed in future).

# Import data

Go to the templates directory (\\your project directory\dataset\code\RaBIDS templates), select the file “datasheet.xlsx” and copy it to the code directory (your project directory\dataset\code\).

Image showing where to find the datasheet.xlsx:

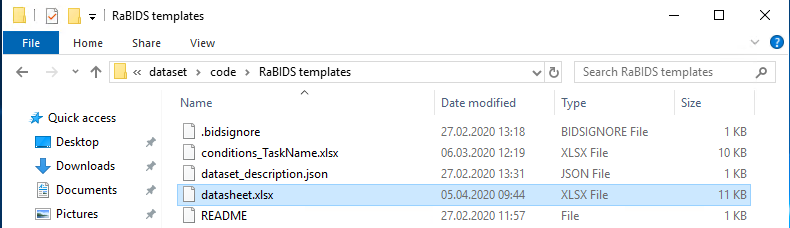
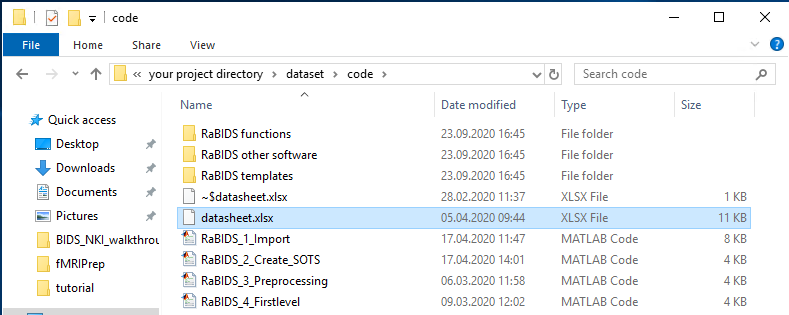


Image showing where to copy-paste it:

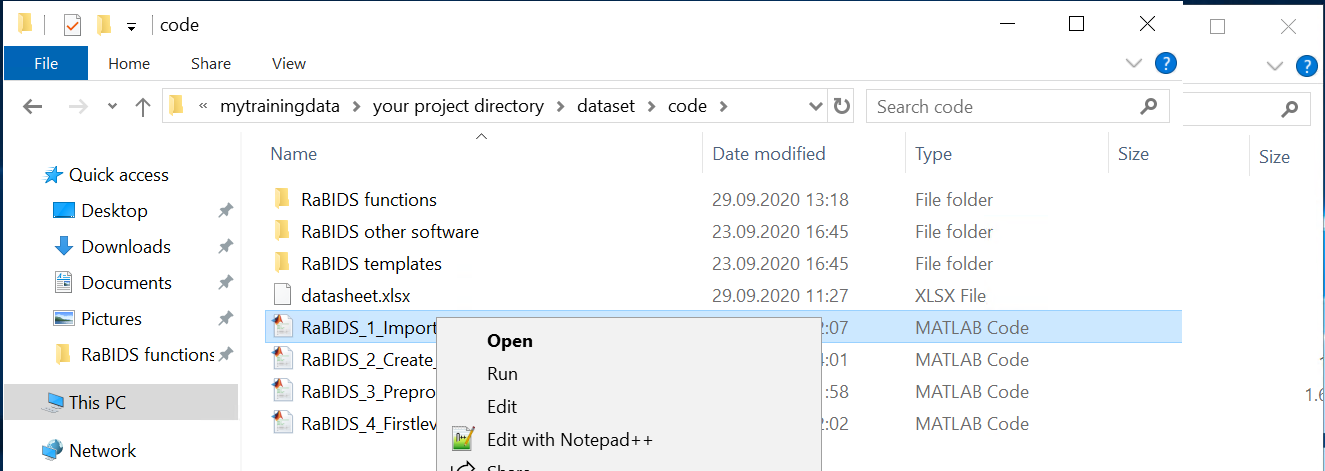


To use the automated import program we first need to populate the “datasheet.xlsx” with information about the training data. **Table 1** (see Appendix) displays the required user input to import the data with the program “RaBIDS\_1\_Import.m”. Fill in the user input as shown in **Table 1**. Alternatively, you can skip this step and copy-paste the table datasheet.xlsx from the directory “tutorial supportive data”. This table is ready to run for this tutorial.

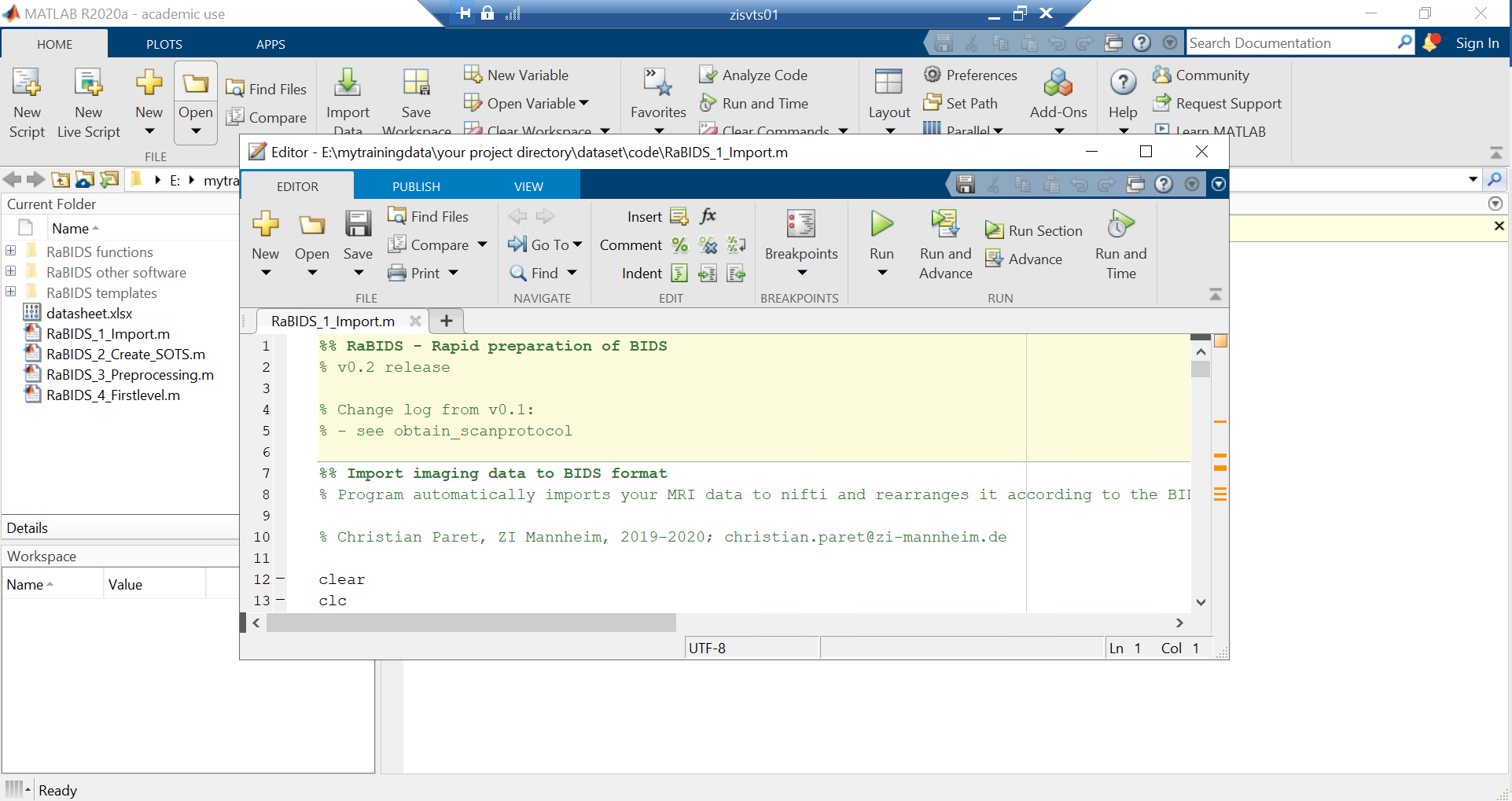
**Important when filling in the datasheet:** Go to user-input field in object-type “data analysis path” (field C6) and write path name to your project directory. Go to user-input field in object-type “data exchange path” (field C7) and write path name to the directory containing the MR images (i.e., the directory \data exchange server\RABIDS-example\dicomdir). In your future projects, you need to fill in here the path to the dicomdir on hobbes.

When you are finished, you need to save the excel file (press Ctrl+S on your keyboard). Now you sould be able to run the program RaBIDS\_1\_Import.m from Matlab:

Click on RaBIDS\_1\_Import with the right mouse button:



Alternatively, open Matlab and press Run:

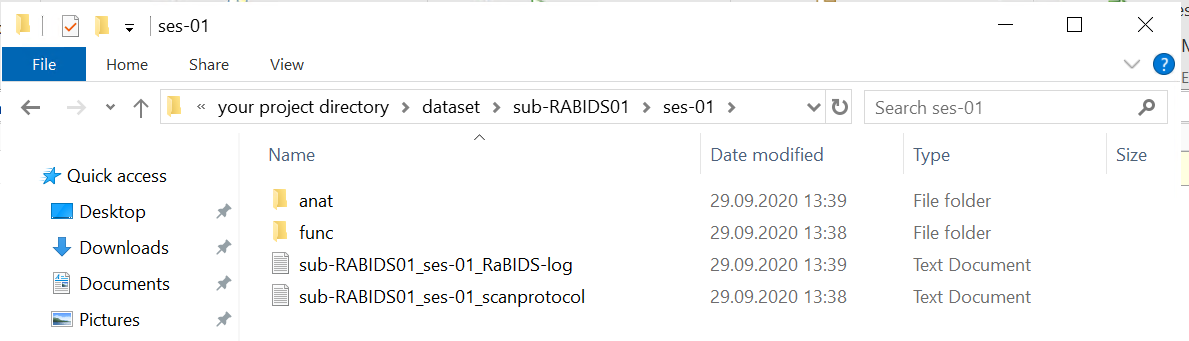


Matlab will write a warning to the command window that variable names were modified (Error #1)[[1]](#footnote-1). We can ignore this message.

Check the command window output after import. For each of the three Tasks (yes, RaBIDS knows the anatomical image as a “task”!) the program should return that dicom import was successful.

## Scan protocol

Go to the subject directory of session 1 (\mytrainingdata\your project directory\dataset\sub-RABIDS01\ses-01) and open the text file called “sub-RABIDS01\_ses-01\_scanprotocol.txt” with Excel. It lists the series number, name and number of scans of each MRI series. This protocol is generated automatically before the import step.



## Logfile

RaBIDS rights the Matlab command window output to a logfile. You find it in the subject directory of session 1 (\mytrainingdata\your project directory\dataset\sub-RABIDS01\ses-01), called “sub-RABIDS01\_ses-01\_RaBIDS-log.txt”. The file is an important source for debugging or locating errors.

## Define task conditions and SOT

In this section, you define the conditions and stimulus onset times (SOT). The program will use this information to define the stimulation protocol of the subject, based on the information from the Presentation logfile. Output will be saved to your project directory\dataset\sub-RaBIDS01\ses-01\func. Two files will be generated per task:

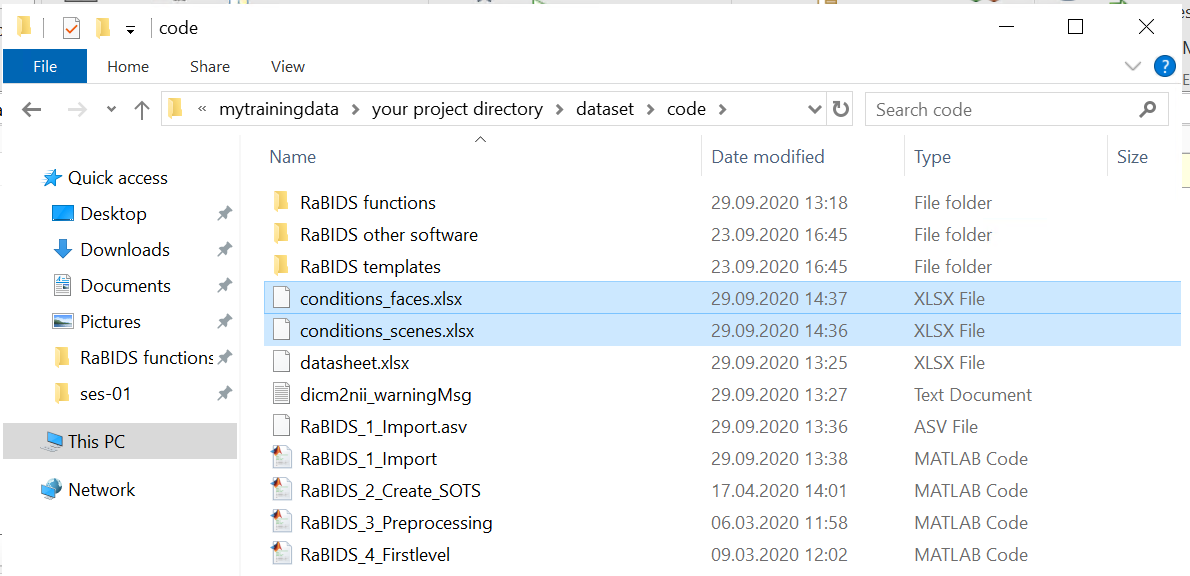
1. sub-RaBIDS\_ses-01\_func\_task-TaskName\_multicond.mat: SOTS-file for SPM12 analysis
2. sub-RaBIDS01\_ses-01\_func\_task-TaskName\_events.tsv: SOTS-file in BIDS-format.

Go to the templates directory (your project directory\dataset\code\RaBIDS templates), select the file “conditions\_TaskName.xlsx” and copy it to the code directory (your project directory\dataset\code\). Rename the file to “conditions\_scenes.xlsx” and open it.[[2]](#footnote-2)

Enter the information as shown in Table 2 to the conditions\_scenes.xlsx file.

In this example we use the Duration feature. You can use the OffsetID feature to define trial duration, too (refer to manual for more information)[[3]](#footnote-3).

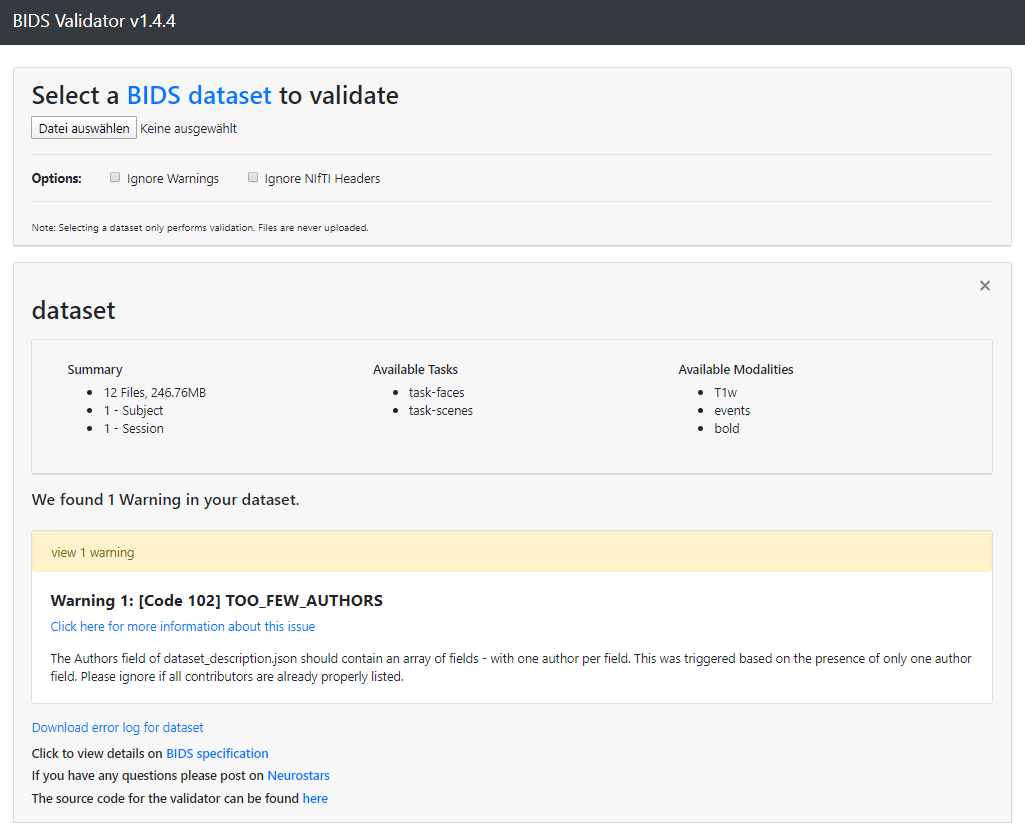
Now you should find the conditions\_scenes.xlsx file in your code directory as shown below:



Run RaBIDS\_2\_Create\_SOTS.m in Matlab. Matlab requests to input the TaskName via the Command Window. Use inverted commas around TaskName that it is recognized as character string (e.g. for TaskName=scenes, type ‘scenes’).

You need to repeat these steps for the faces task (see Table 3).

# BIDS validation

For fun, go to website <https://bids-standard.github.io/bids-validator/> and validate your dataset. The ouput shows that the dataset was recognized in good order: 

Congratulations, you produced your first BIDS-compatible data set!

# Data description

## Tasks

### Faces

Participants were shown faces with emotional expression from the Warsaw Set of Emotional Facial Expression Pictures (WSEFEP, http://www.emotional-face.org/). A block design of 12 blocks with 6 negative emotional faces and 12 blocks with scrambled faces (control condition) was used. In sum, 72 negative faces of 24 actors (12 female, 12 male) were shown for 3 seconds each. The inter-trial interval was jittered between 10±1 seconds. To ensure attention, participants were asked to press a button to indicate for every picture whether the person was male or female, or whether color of the bounding box around scrambled faces was blue or green.

### Scenes

During this task, pictures from the OASIS picture set (Kurdi et al., 2017) are presented to induce negative affect. We used pictures with negative affective valence and high arousal (aversive condition, AC) in a block-design. During each of 14 blocks, lasting 18 seconds, three picture stimuli were presented consecutively for 6 seconds each, resulting in a set of 42 negative pictures in total. Scrambled pictures were used in a non-affective control condition (neutral condition, NC) with same number of trials. During the intertrial interval (10±1 seconds), participants viewed a white fixation cross on a black background. To ensure attention, participants were asked to press a button to indicate for every picture whether a person can be seen or not, or whether color of the bounding box around scrambled pictures is blue or green.

### MRI data acquisition

FMRI data were acquired on a 3 Tesla MRI Scanner (Magnetom Trio with TIM technology, Siemens Medical Service, Erlangen, Germany) equipped with a 32 channel head coil. Functional images of the BOLD contrast were acquired with gradient echo T2\* weighted echo-planar-imaging sequence (TE=30 ms, TR=2 s, FOV=220 mm x 220 mm, Matrix size=64 x 64, flip angle=80°). A volume comprised 36 slices in AC-PC orientation with a thickness of 3 mm and slice gap of 1 mm. Participants’ heads were lightly restrained using soft pads to prevent head movement. A T1-weighted anatomical image was also recorded (TE=3.03 ms, TR=2.3 s, 192 slices and FOV=256 mm x 256 mm, Matrix size 256 x 256, Slice thickness = 1 mm).

Scans were acquired in this order: localizer – anatomical – fieldmap – task-faces – task-scenes

Table 1. datasheet.xlsx

Note: paths to spm12, dicm2nii may be different on your computer. Data analysis path and data exchange path need to be costumized, too. Consult the manual for more information on ObjectType. Note: Matlab 2020a requires fields MinImages and MaxImages to be filled in for the first add path object.



Table 2. conditions\_scenes.xlsx



Log ID format can either be ‘BIDS’ or ‘free’ for this data set.

Table 3. conditions\_faces.xlsx



The OffsetID feature is used to define trial durations for illustrative reasons. The program will assume that each trial with OnsetID “onset\_face” and “onset\_scrambled” ends at time when the next “rest” event was written to the logfile.

Log ID format can either be ‘BIDS’ or ‘free’ for this data set.

1. You can look up more information about errors in the document “RaBIDS Error Reference” in the manuals directory ([\\zi.local\flstorage\dep\_psm\share\RaBIDS\manuals](file:///\\zi.local\flstorage\dep_psm\share\RaBIDS\manuals)). [↑](#footnote-ref-1)
2. **Excursus:** To understand what information is needed in the table, go to the directory containing the logfiles (\your project directory\sourcedata\sub-RaBIDS01\ses-01). You can use Excel to open the file “sub-RaBIDS01\_ses-01\_task-faces.log”.

   Colomn C “Code” shows the relevant event information. For us, the relevant entries are:

   * “onset\_negative\_ XXX”: onset of an 18-sec picture trial of condition “negative”
   * “onset\_scrambled\_XXX”: onset of an 18-sec scrambled-picture trial of condition “control”
   * “rest”: onset of fixation cross

   You can see that the events are listed with corresponding onset time. When running the next program, this information is looked up automatically for each trial and is written to the ouput files. Now you can close the logfile again. [↑](#footnote-ref-2)
3. In order to add a new Condition to the conditions\_scenes.xlsx file, you need to add a new line to the table. To do this, you can mark the third line by clicking on the number ‘3’ on the very left of the excel-table and press Ctrl and + on the keyboard. [↑](#footnote-ref-3)