

# MEGneto

A guide on commonly encountered errors and how to tackle them

<b>Improper JSON config file set up</b>	<b>2</b>
<b>Forgetting to populate the subject .CSV files</b>	<b>3</b>
<b>Subjects missing matching MRI and MEG data</b>	<b>4</b>
<b>Incorrect file extension for MRI data (mri data is not .mri extension)</b>	<b>4</b>

## Improper JSON config file set up

Error occurrence #1: During fcp1 a warning pops up that says “not enough markers found”.

```
=====
...DS_FILE: ST03_MEG072_20121107_Both.ds
SUBJECT: ST03
14:33:28      Finding t0 markers...
undoing the G3BR balancing for the gradiometer definition
Warning: Not enough markers found - check marker file!
Reference to non-existent field 'bad_chann'.

Error in fcp 1 TaskEpoching (line 223)
bad_subj = subj_match.pid(cellfun('length',fcp1_output.bad_chann) > 15);
```

Why did this occur?

In the JSON config file set up the user must specify marker names that are specific to their data/analysis. Without this, when the data is read, the pipeline has no information as to which events to epoch around (they are not found unless their names are specified by the user).

How to debug:

1. Navigate to the JSON config file.
2. Find the `task.trialdef.markers` field and ensure these markers match the markers specific to your analysis. You should have a marker file that is specific to your data with which you can compare these fields.

Note: For more information on the JSON config file please navigate to the `ConfigParams.md` file for a description of each field.

Error occurrence #2: Error in load config, parsing config, or recursive json struct.

```
14:37:40      Now running **fcp_1_TaskEpoching**.  
Error using str2func  
Invalid function name ''  
  
Error in recursive_json_struct_string_to_func (line 5)  
    y = str2func(y.function);  
  
Error in recursive_json_struct_string_to_func (line 11)  
    y.(z{ii}) = recursive_json_struct_string_to_func( y.(z{ii}) );  
  
Error in recursive_json_struct_string_to_func (line 11)  
    y.(z{ii}) = recursive_json_struct_string_to_func( y.(z{ii}) );  
  
Error in load_config (line 12)  
conf = recursive_json_struct_string_to_func(conf);
```

Why did this occur?

The user must populate a JSON config file which specifies various parameters, specific to the user's analysis/data, that the pipeline relies on for extracting information.

How to debug:

1. Navigate to the JSON config file and fill in the fields manually OR in the main template file, run the `interactive_JSON_config` function which guides the user through populating the config file.

Note: For more information on the JSON config file please navigate to the ConfigParams.md file for a description of each field.

## Forgetting to populate the subject .CSV files

Error occurrence: Error using load participants

```
14:40:04      Now running **fcp_1_TaskEpoching**.  
Error using load_participants (line 14)  
No participants specified - please add ds files to  
/home/dmatic/MEGProjects/analysis/overwrite_megne2setup/config/subj_fcp1.csv  
  
Error in ds_pid_match (line 9)  
ds = load_participants(paths, step);  
  
Error in fcp_1_TaskEpoching (line 66)  
subj_match = ds_pid_match(paths,step);
```

Why did this occur?

The user must populate the specific step's comma separated value (csv) file with participants such that the pipeline can perform the analysis on those participants. If that csv file is empty, the pipeline has nothing to perform an analysis on/no data to work with.

How to debug:

1. Navigate to the step's csv file (`subj_fcpX.csv`, where X is the number of the step you are on in the pipeline) and populate the file with participants OR uncomment the lines above the call to `fcp1` to auto populate the csv file. If you wish to auto populate subsequent steps, copy the code above the call to `fcp1` and change the `subj_fcp1` is `fid = fopen(paths.subj_fcp1, 'w')` to match the step number that you are on.
  - a. Note that auto population is fine to do for every step, however the user must ensure they manually delete any participants they wish to exclude (due to an excess of bad channels, bad trials, etc.).

## Subjects missing matching MRI and MEG data

Error occurrence: Error using `ds_pid_match` which finds matching MEG and MRI data.

```
14:58:13      Now running **fcp_1_TaskEpoching**.
```

---

```
Subjects missing MRI files:
  'ST03_MEG072_20121107_Both.ds'
```

Error using **ds\_pid\_match** (line 25)  
The participants above are missing MRI data. Please investigate, edit `subj_fcp1.csv`, and re-run `fcp1`.

Error in **fcp\_1\_TaskEpoching** (line 66)  
`subj_match = ds_pid_match(paths,step);`

Why did this occur?

Some subjects listed in the `subj_fcpX.csv` file (where X corresponds to the current step), may be missing matching MEG/MRI data.

How to debug:

1. First, check if the matching MEG/MRI data exists somewhere. If so, add it to the MRI data folder so that the participant(s) now have the matching MRI data.
2. If there is no existing MEG/MRI data, navigate the `subj_fcpX.csv` and remove participants that are missing matching MEG/MRI data. The error should specify which participants are missing this data.

## Incorrect file extension for MRI data

Error occurrence:

Why did this occur?

MEGneto requires MRI data to have the .mri extension. If a participant's MRI data that does not have this extension, the pipeline will error out.

How to debug:

1. Ensure all MRI files end in .mri. If they do not, check if there is existing .mri data for these participants elsewhere and add it to the MRI folder. Else, delete the participants that do not have .mri data.

## Incorrect MRI file naming convention

Error occurrence: Error occurs in megne2setup's path generation.

```
15:0:39      Now running **megne2setup**.  
Warning: Name is nonexistent or not a directory: /home/dmantic/fieldtrip  
Warning: incorrect PID format: more than one underscore or period was found  
Error using cell2struct  
Duplicate field name "ST01"  
  
Error in path_generation (line 85)  
    all_participants = cell2struct(all_participants,pids);  
  
Error in megne2setup (line 115)  
[paths, subj] = path_generation(project_path, analysis_name, rawdata_path, mri_path, overwrite);
```

Why did this occur?

Participant MRI data follows a naming convention. If the naming convention is violated, the pipeline errors out as it gets confused.

How to debug:

1. Navigate to the folder which contains the MRI data and alter the file names to match the following convention
  - a. MRI files should only contain one underscore after the last slash that designates the file path (e.g., /xxx/xxx/xxx/ST01\_V1.mri is acceptable but /xxx/xxx/xxx/ST01\_V1\_V2.mri is not). Note that underscores are not required (e.g., /xxx/xxx/xxx/ST01.mri is acceptable).