- CMIP6 Data Request Web GUI -

Customized Variable Mapping Tables and Post Processing Script Fragments
Edition November 2018

Martin Schupfner

Deutsches Klimarechenzentrum (DKRZ)

Contents

1 Introduction

A web application has been developed within the context of the German national Coupled Model Intercomparison Project Phase 6 (CMIP6) support project DICAD, funded by the **BMBF**. Its aim is to facilitate not only to collect but also to grasp customized data requests, depending on the CMIP6 experiments to be performed as well as the CMIP6 endorsed Model Intercomparison Projects (MIPs) that are supported.

The total CMIP6 Data Request is published as two xml files (**see CMIP6 Data Request web site:** `https://earthsystemcog.org/projects/wip/CMIP6DataRequest`), containing a large number of cross-linked objects. The coarse structure will be outlined in the following (Fig. 1.1).

The CMIP6 endorsed MIPs cover various scientific objectives and define experiments from which a number of variables are requested. The experiments are classified into 'tiers' and each variable request has an assigned 'priority'. The variable, as defined by one of the MIPs, is called ”MIP variable”, whereas their programmatic implementation(s) [1] are termed ”CMOR variables”, referring to the Climate Model Output Rewriter 'CMOR' (Nadeau, 2016), a software used to format and adapt the model data to the CMIP standard. When a CMOR variable is requested, it may be linked with even more attributes, which include among others a time slice, the priority of the request and the requested grid. Thus it follows that the number of variable requests can be higher than the approximately more than 2000 CMOR variables defined for CMIP6.

Several web applications have been developed in order to:

1. Compute and download CMIP6 Data Request and Data Volume Estimate, both customized to the user's needs.

2. Map model variables to requested CMOR-Variables. This mapping will be used in the project conformal reformatting of the model output.

3. Automatically build post processing script fragments and a configuration file to perform and direct the diagnostic and CMOR rewrite of the mapped variables.

The web applications take advantage of Martin Juckes' (BADC, British Atmospheric Data Center) Data Request Python API (DreqPy, Juckes (2016)), which is the officially released software allowing to interactively browse the CMIP6 Data Request.

The software is written in Python and supported by Python in versions 2.6.6, 2.7 as well as 3.x. To overcome the need of a local installation of Martin Juckes' software, the DKRZ made the basic use cases of DreqPy accessible through this web GUI.

The Variable-Mapping Web Application uses information prepared by the DreqPy software. It builds on previous efforts of Karl–Hermann Wieners (MPI-M), to simply collect Variable-Mapping Information in a database via a web application. This database then can be used to automatically create post processing script fragments for the diagnostic and standard conform rewrite of the model output.

In the following sections the design and usage of both the web applications will be described in detail.

[1] Programmatic implementations are different aggregations, shapes and cell-methods.

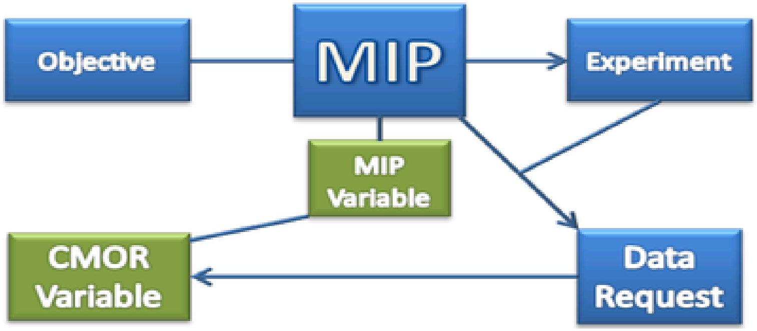


Figure 1.1: *A coarse structural illustration of the CMIP6 Data Request.*

2 Data Request WebGUI

The web GUI facilitates the use of DreqPy by providing its basic functionality via a web interface. Three kinds of activities are supported, each of which will be described in more details in the following subsections:

- Generate a customized data request in csv-format (2.2).
- Generate a customized data request in xlsx-format (2.3).
- Calculate a data volume estimate of a customized data request (2.4).

All of which have in common that the user has to choose the Model Intercomparison Projects (MIPs) he will support, as well as the experiments to be performed. The data request from all MIPs will be calculated for each experiment individually, and for the entire selection of experiments (joint/merged request). Additional info will be displayed on the website, containing a summary of the user selections and the generated data request.

2.1 Generate a customized CMIP6 Data Request - Basics

Selection of MIPs and Experiments

In order to generate a customized data request the tab **Requested Variables** has to be opened. This is the default entry point when accessing the web GUI.

Here, the MIP(s) and experiment(s) have to be selected as displayed in Fig. 2.1.

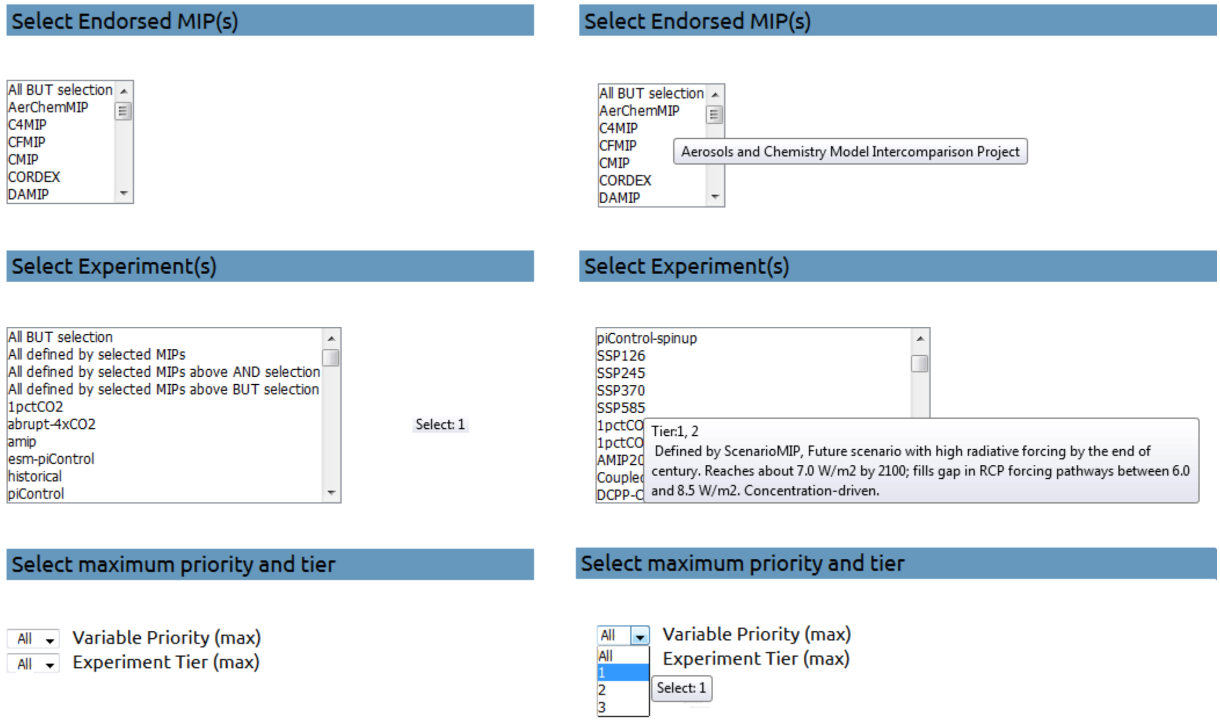
Hovering over the select box will cause additional information about the MIP or experiment to be displayed. When holding the ctrl-key, more than one option in each select box may be selected. Besides solely selecting MIPs and Experiments, a special option can be chosen:

- '**All BUT selection**'

For the selection of Experiments, furthermore the following options are available. Only one special option can be chosen.

- '**All defined by selected MIPs**': Each experiment is defined by a certain MIP. This option will select every experiment that is defined by any of the selected MIPs.
- '**All defined by selected MIPs AND/BUT selection**': This option will additionally to the previous one add/remove the selected experiments.

Note, that the automatic selection will happen within the program and will not be displayed in the web GUI. The list of selected experiments and MIPs will be shown as additional information next to the download button, after generating the data request either in csv- or xlsx-format (Section 2.2, Fig. 2.3). Choosing more than one special option in the 'Experiment(s)'select box will result in a warning message or in an error.



Aerosols and Chemistry Model Intercomparison Project

Tier:1, 2
Defined by ScenarioMIP, Future scenario with high radiative forcing by the end of century. Reaches about 7.0 W/m2 by 2100; fills gap in RCP forcing pathways between 6.0 and 8.5 W/m2. Concentration-driven.

*Figure 2.1: The selection dialogues of the tab '**Requested Variables**'. The right hand side shows information that appears when hovering over an entry.*

Selection of Priority and Tier

The lowermost selection box shown in Fig. 2.1 allows to choose the variable priority and experiment tier that shall be taken into account.

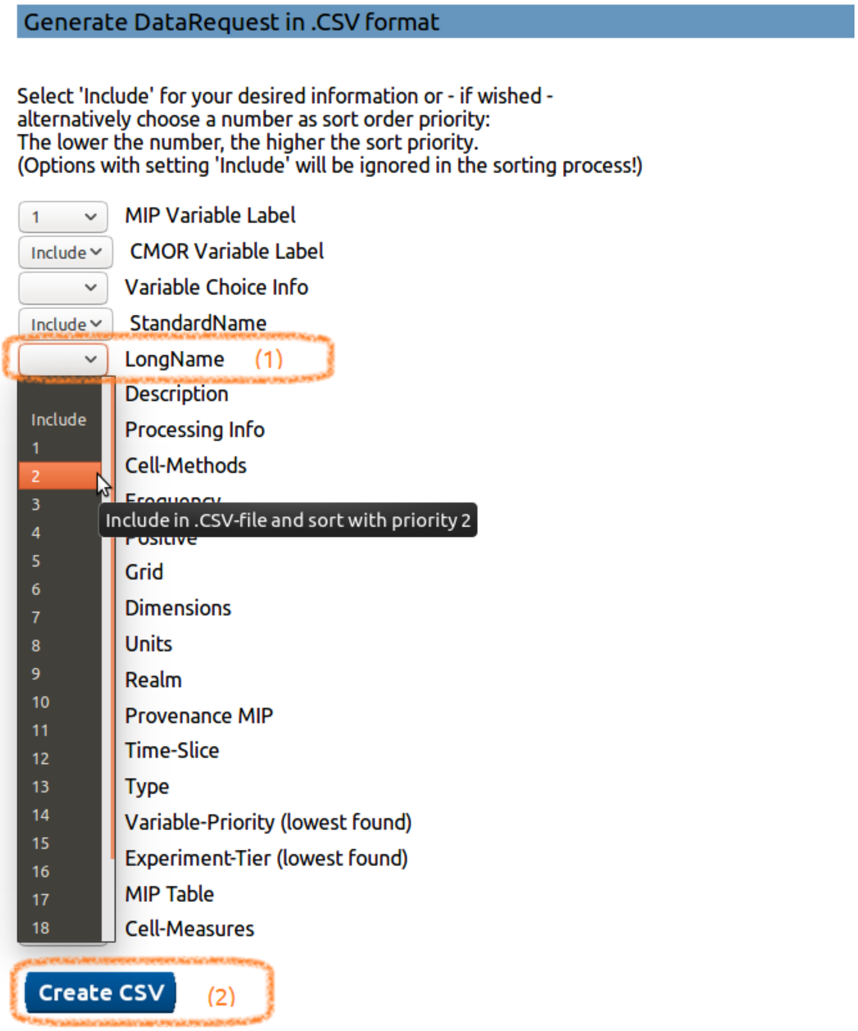
Every experiment has one or more assigned tiers. The tier may depend on the MIP requesting the data and for different tiers, the data request may be different as well. Note that different realisiations of an experiment may have assigned different tiers. The experiment tier can have values between 1 and 4, where 4 signifies the lowest tier (= lowest rank of the experiment) and 1 the highest tier.

The variable priority can have values between 1 and 3, where 3 signifies the lowest priority and 1 the highest priority. The priority depends solely on the MIP requesting this variable from a certain experiment.

The smaller the value for the maximum variable priority and experiment tier, the less variables will be included in the generated data request. For example, when choosing the maximum priority 2, all variables requested with priority 1 and 2 will be included in the data request.

2.2 Generate a customized CMIP6 Data Request in csv-format

When all settings as described in previous section (Sec. 2.1) are made, the data request can be generated in csv-format. It has to be chosen which information about the requested CMOR variables will be included. This can be done via a list of select boxes in the 'Generate DataRequest in .csv format ' section (see Fig. 2.2). Each variable attribute or bit of information can be included by setting its select box to 'Include' (see (1) in Fig. 2.2). Alternatively one can select a number, which then represents the sort order priority of this attribute. For example, selecting 'MIP Variable Label' as '1' and 'LongName'as '2', will cause the resulting list of variables to be sorted by their 'MIP Variable Label' in first order, and by 'LongName' in second.



Generate DataRequest in .CSV format

Select 'Include' for your desired information or - if wished -
alternatively choose a number as sort order priority:
The lower the number, the higher the sort priority.
(Options with setting 'Include' will be ignored in the sorting process!)

1 MIP Variable Label
Include CMOR Variable Label
Variable Choice Info
Include StandardName
LongName (1)
Description
Processing Info
Cell-Methods
Frequency
Positive
Grid
Dimensions
Units
Realm
Provenance MIP
Time-Slice
Type
Variable-Priority (lowest found)
Experiment-Tier (lowest found)
MIP Table
Cell-Measures

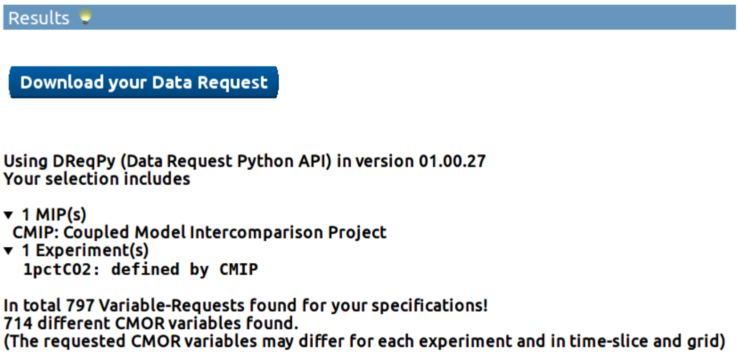
Include in .CSV-file and sort with priority 2

Figure 2.2: *When generating the data request in csv format, the information to be included can be chosen. Additionally, a sort order priority can be chosen. In the example above, the resulting list of variables will be sorted by their MIP variable label in first order, and by LongName in second order. Information that shall be included in the csv file, but can be ignored by the sorting algorithm, has to be selected as 'Include'.*

Note, that each row in the resulting csv file will be unique. This allows for example to create a data request containing only the 'MIP Table' as desired information. The resulting csv file(s) will then include a single column labeled 'MIP Table' having as much rows as there are MIP Tables despite having one row per variable request!

Finally, the 'Create CSV'-button has to be clicked to start generating the data request. It may take several seconds for the data request to be calculated.

For each selected experiment, one csv-file will be generated, with all variables requested by the selected MIPs. Additonally, one file with the joint data request for all selected experiments will be generated. A download button and additional information will appear as can be seen in Fig. 2.3. For the example shown in Fig. 2.3, CMIP makes 797 requests from the experiment '1pctCO2'. This 797 requests include 714 different CMOR variables which can be requested for different time slices or on different grids. Variable priority and experiment tier will be merged to their lowest value (= the highest priority and tier, respectively). The generated list of variables would have this 797 entries if all attributes or information were selected to be included in the csv-file. Less selected attributes or information will lead to a shorter list as double entries get removed.



Using DReqPy (Data Request Python API) in version 01.00.27
Your selection includes

▾ 1 MIP(s)
CMIP: Coupled Model Intercomparison Project
▾ 1 Experiment(s)
1pctCO2: defined by CMIP

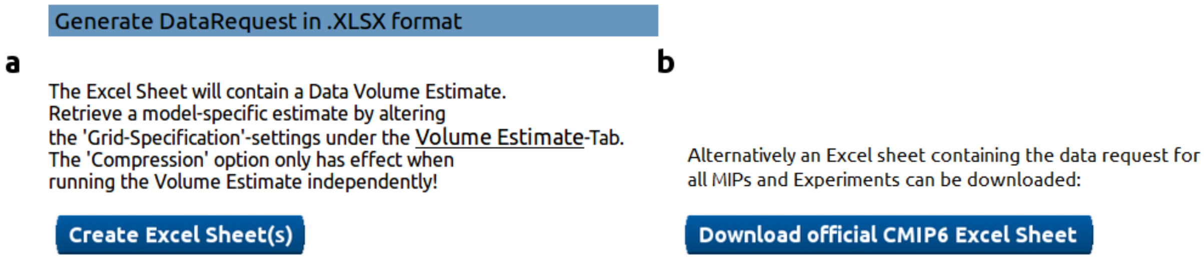
In total 797 Variable-Requests found for your specifications!
714 different CMOR variables found.
(The requested CMOR variables may differ for each experiment and in time-slice and grid)

Figure 2.3: *Resulting customized data request: Download button and additional information. In this example the data request for CMIP and all experiments defined by CMIP has been generated.*

2.3 Generate a customized Data Request in xlsx-format

When all settings as described in Sec. 2.1 are made, the data request can be generated in xlsx-format. To initiate the calculation the 'Create Excel Sheet(s)' button has to be clicked (see Fig. 2.4). This process may take up to a minute.

As the produced Excel sheets will contain a volume estimate, the shape of the model grid can be entered under the tab '**Volume Estimate**'. To change to this tab, one can either follow the link in the description above the two buttons (Fig. 2.4) or select the tab on the top of the web page.


Generate DataRequest in .XLSX format
a
b
Create Excel Sheet(s)
Download official CMIP6 Excel Sheet

The Excel Sheet will contain a Data Volume Estimate.
Retrieve a model-specific estimate by altering
the 'Grid-Specification'-settings under the Volume Estimate-Tab.
The 'Compression' option only has effect when
running the Volume Estimate independently!

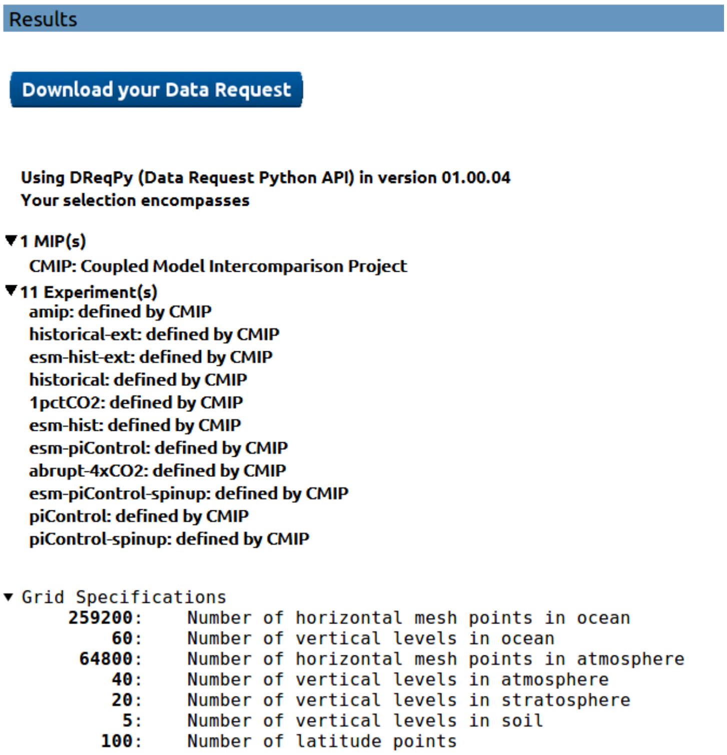
Alternatively an Excel sheet containing the data request for
all MIPs and Experiments can be downloaded:

Figure 2.4: *Section in the web GUI to generate the data request in xlsx format (a). The xlsx-file will contain a data volume estimate. Alternatively to customized data requests, the total CMIP6 data request, containing all CMOR variables, can be downloaded by clicking the button **Download official CMIP6 Excel Sheet** (b) that can be found at the top of the page. Here, however, a volume estimate is not included.*

Note that other but the model grid specifications settings to be found under the tab '**Volume Estimate**' only have effect when calculating the volume estimate separately. More information about calculating the data request volume estimate will be given in the next section 2.4. After adjusting the 'Volume Estimate' settings generation of the data request in xlsx format under the tab '**Requested Variables**'can be initiated, as described above.

The button shown in Fig. 2.4b will start the download of a pre-processed data request containing all variables in the CMIP6 project. This file has been originally delivered with the DreqPy software.

After successfully finishing the calculations a download button and additional info will appear, as can be seen in Fig. 2.5. The shown additional information comprises the chosen MIPs and experiments as well as the model grid specifications used to generate the volume estimate included in the xlsx files.



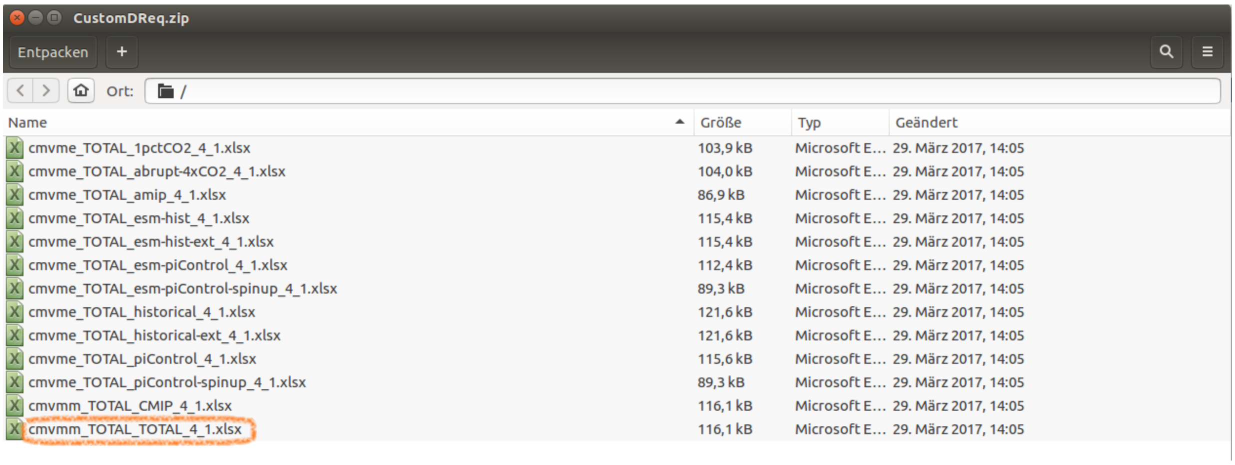
Using DReqPy (Data Request Python API) in version 01.00.04
Your selection encompasses

▼1 MIP(s)
CMIP: Coupled Model Intercomparison Project
▼11 Experiment(s)
amip: defined by CMIP
historical-ext: defined by CMIP
esm-hist-ext: defined by CMIP
historical: defined by CMIP
1pctCO2: defined by CMIP
esm-hist: defined by CMIP
esm-piControl: defined by CMIP
abrupt-4xCO2: defined by CMIP
esm-piControl-spinup: defined by CMIP
piControl: defined by CMIP
piControl-spinup: defined by CMIP

```
▾ Grid Specifications
     259200:    Number of horizontal mesh points in ocean
         60:    Number of vertical levels in ocean
      64800:    Number of horizontal mesh points in atmosphere
         40:    Number of vertical levels in atmosphere
         20:    Number of vertical levels in stratosphere
          5:    Number of vertical levels in soil
        100:    Number of latitude points
```

Figure 2.5: *Resulting customized data request in xlsx format: Download button and additional information. In this example the data request for CMIP and all its defined experiments has been generated.*

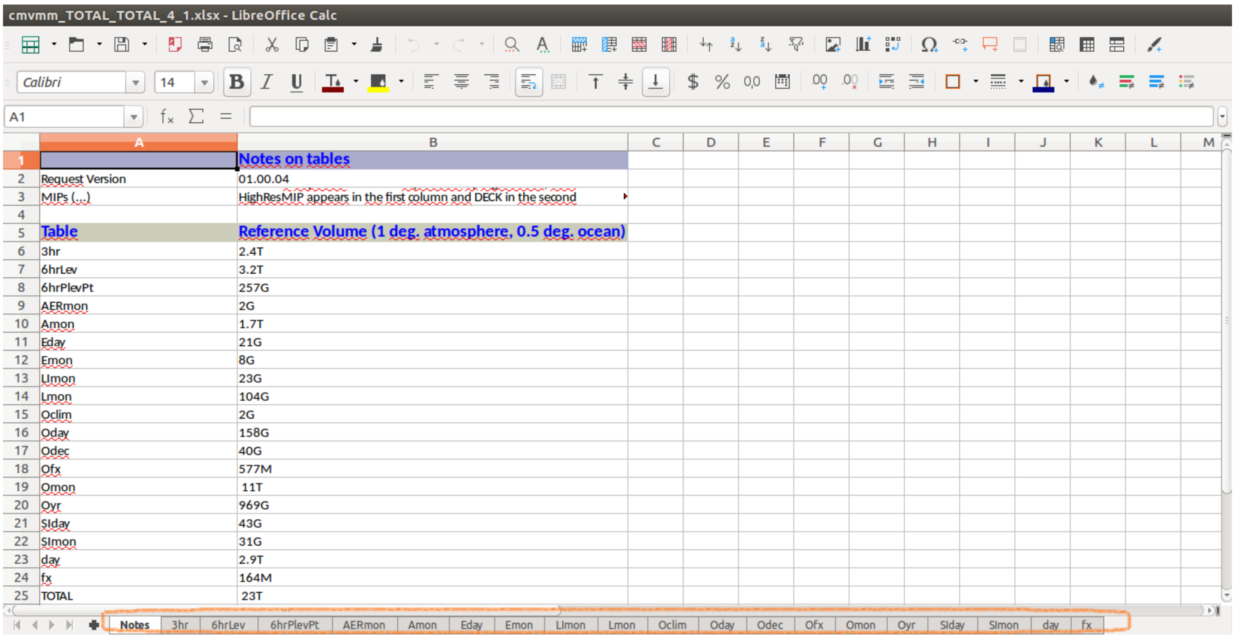
A click on the 'Download your Data Request' button will initiate the download of the zip file, containing the data request in several xlsx files (see Fig. 2.6). The file with the substring 'TOTAL_TOTAL' is the one containing the joint data request from all selected MIPs for all selected experiments. The other files contain the data request from combinations of each one of the chosen experiments and MIPs. For example 'cmvme_TOTAL_1pctCO2_4_1.xlsx' would be the total request from all selected MIPs for the experiment '1pctCO2' with the experiment tier being equal to 4 and the variable priority being equal to 1. And 'cmvme_TOTAL_CMIP_4_1.xlsx' would be the total request from all selected MIPs for all experiments that 'CMIP' defines for tier equal to 4 and priority equal to 1. As in this case CMIP is the only selected MIP, this request is equal to the one labeled 'TOTAL_TOTAL'. More information on the spreadsheet formats can be found in the DreqPy User's guide (Juckes, 2016).



Name	Größe	Typ	Geändert
cmvme_TOTAL_1pctCO2_4_1.xlsx	103,9 kB	Microsoft E…	29. März 2017, 14:05
cmvme_TOTAL_abrupt-4xCO2_4_1.xlsx	104,0 kB	Microsoft E…	29. März 2017, 14:05
cmvme_TOTAL_amip_4_1.xlsx	86,9 kB	Microsoft E…	29. März 2017, 14:05
cmvme_TOTAL_esm-hist_4_1.xlsx	115,4 kB	Microsoft E…	29. März 2017, 14:05
cmvme_TOTAL_esm-hist-ext_4_1.xlsx	115,4 kB	Microsoft E…	29. März 2017, 14:05
cmvme_TOTAL_esm-piControl_4_1.xlsx	112,4 kB	Microsoft E…	29. März 2017, 14:05
cmvme_TOTAL_esm-piControl-spinup_4_1.xlsx	89,3 kB	Microsoft E…	29. März 2017, 14:05
cmvme_TOTAL_historical_4_1.xlsx	121,6 kB	Microsoft E…	29. März 2017, 14:05
cmvme_TOTAL_historical-ext_4_1.xlsx	121,6 kB	Microsoft E…	29. März 2017, 14:05
cmvme_TOTAL_piControl_4_1.xlsx	115,6 kB	Microsoft E…	29. März 2017, 14:05
cmvme_TOTAL_piControl-spinup_4_1.xlsx	89,3 kB	Microsoft E…	29. März 2017, 14:05
cmvmm_TOTAL_CMIP_4_1.xlsx	116,1 kB	Microsoft E…	29. März 2017, 14:05
cmvmm_TOTAL_TOTAL_4_1.xlsx	116,1 kB	Microsoft E…	29. März 2017, 14:05

Figure 2.6: *Zip file containing the data request in xlsx format. The different files contain the data request per experiment or MIP. The complete/joint data request from all MIPs and experiments is contained in the file whose name includes the substring 'TOTAL_TOTAL'.*

The xlsx files each contain several sheets (see Fig. 2.7). The first sheet includes a volume estimate and the remaining sheets the data request per MIP table. The web GUI just captures the basic functionality of DreqPy. With DreqPy it is also possible to calculate the intersection of the data requests of several MIPs and experiments, as well as the variables requested uniquely by a certain MIP. Moreover, the volume estimate and excel sheets can be given per frequency.

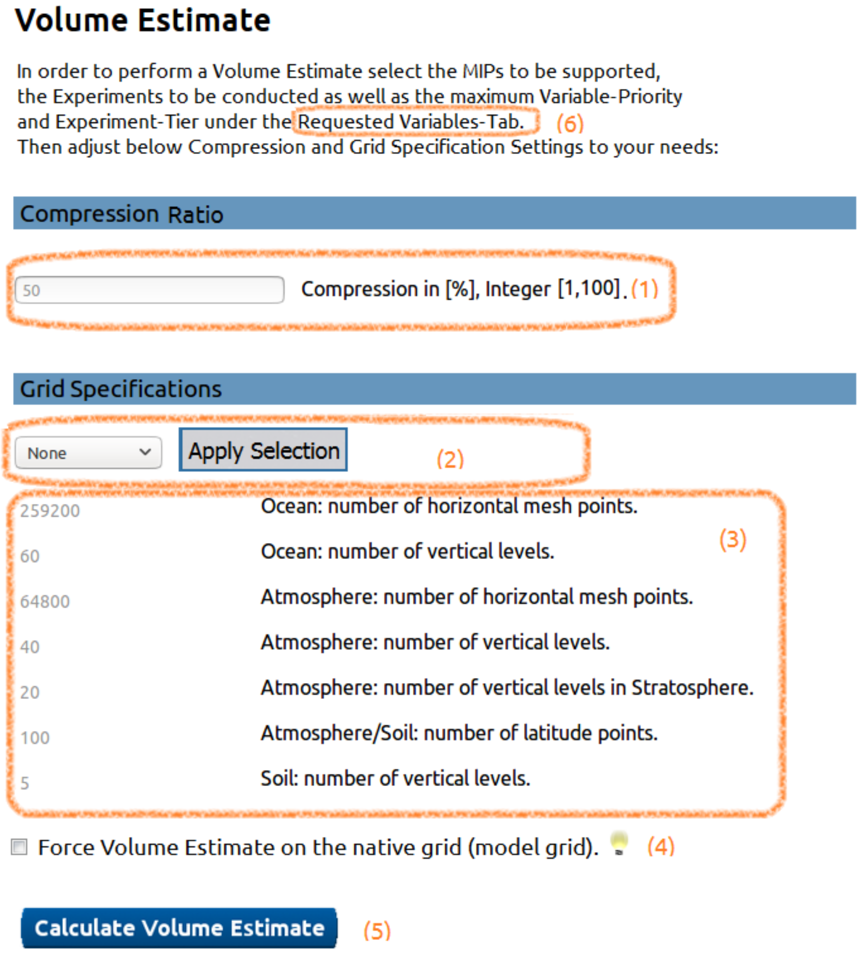


	A	B
1		Notes on tables
2	Request Version	01.00.04
3	MIPs (...)	HighResMIP appears in the first column and DECK in the second
4		
5	Table	Reference Volume (1 deg. atmosphere, 0.5 deg. ocean)
6	3hr	2.4T
7	6hrLev	3.2T
8	6hrPlevPt	257G
9	AERmon	2G
10	Amon	1.7T
11	Eday	21G
12	Emon	8G
13	LImon	23G
14	Lmon	104G
15	Oclim	2G
16	Oday	158G
17	Odec	40G
18	Ofx	577M
19	Omon	11T
20	Oyr	969G
21	SIday	43G
22	SImon	31G
23	day	2.9T
24	fx	164M
25	TOTAL	23T

Figure 2.7: *Data request in xlsx format. The first sheet contains a volume estimate. The other sheets contain the data request per MIP table.*

2.4 Generate Volume Estimate for a customized Data Request

To perform a volume estimate for a custom data request, at first all settings discussed in Sec. 2.1 have to be made. Thereafter the tab '**Volume Estimate**' has to be opened to alter the compression and model grid specifications (see Fig. 2.8).



Volume Estimate

In order to perform a Volume Estimate select the MIPs to be supported, the Experiments to be conducted as well as the maximum Variable-Priority and Experiment-Tier under the Requested Variables-Tab. (6)
Then adjust below Compression and Grid Specification Settings to your needs:

Compression Ratio

50 Compression in [%], Integer [1,100]. (1)

Grid Specifications

259200	Ocean: number of horizontal mesh points.
60	Ocean: number of vertical levels.
64800	Atmosphere: number of horizontal mesh points.
40	Atmosphere: number of vertical levels.
20	Atmosphere: number of vertical levels in Stratosphere.
100	Atmosphere/Soil: number of latitude points.
5	Soil: number of vertical levels.

□ Force Volume Estimate on the native grid (model grid). (4)

Figure 2.8: Compression and model grid specifications may be altered when generating data request volume estimates. For several models pre-defined grid specifications exist. Missing pre-definitions can be forwarded via the feedback functionality of the web GUI.

The default settings that are initially entered as a placeholder are as follows:

- 50% netCDF file compression
- Ocean: 0.5 degree grid
- Atmosphere/Soil: 1 degree grid

- In deviation to this, 100 latitude points in the Atmosphere/Soil.

With the help of Fig. 2.8 hereafter the different settings will be explained.

(1) **Compression**: Desired compression factor (ratio of compressed to original filesize).

(2) **Grid Specifications Pre-Definitions**: Select box to choose a climate model. The selected model's grid specifcation will be applied when clicking the button '**Apply Pre-Definitions**'. When a model cannot be found in the pre-definitions list, its grid specifications may be forwarded via the web site's feedback option.

(3) Besides using Pre-Definitions the model grid specifications can be entered manually.

(4) The data request volume estimate will be provided separately for every selected MIP and the experiment, that this MIP requests data from. Combinations of MIPs and experiments that do not contribute to the volume estimate will be omitted. Below the individual data requests, the volume estimate for the joint data request is given. A MIP may request variables to be on a certain grid, eg. a 1 degree grid. Per default, the volume estimate will be calculated on the entered model grid, if the data request does not state otherwise. Activating the option 'Force Volume Estimate on the native grid', will perform all calculations for the volume estimate using the model grid specifications (see also Fig. 2.9).

(5) Button '**Calculate Volume Estimate**': Starts the calculation.

(6) Link to change to the tab '**Requested Variables**'. Alternatively the respective tab at the top of the web site can be opened directly.

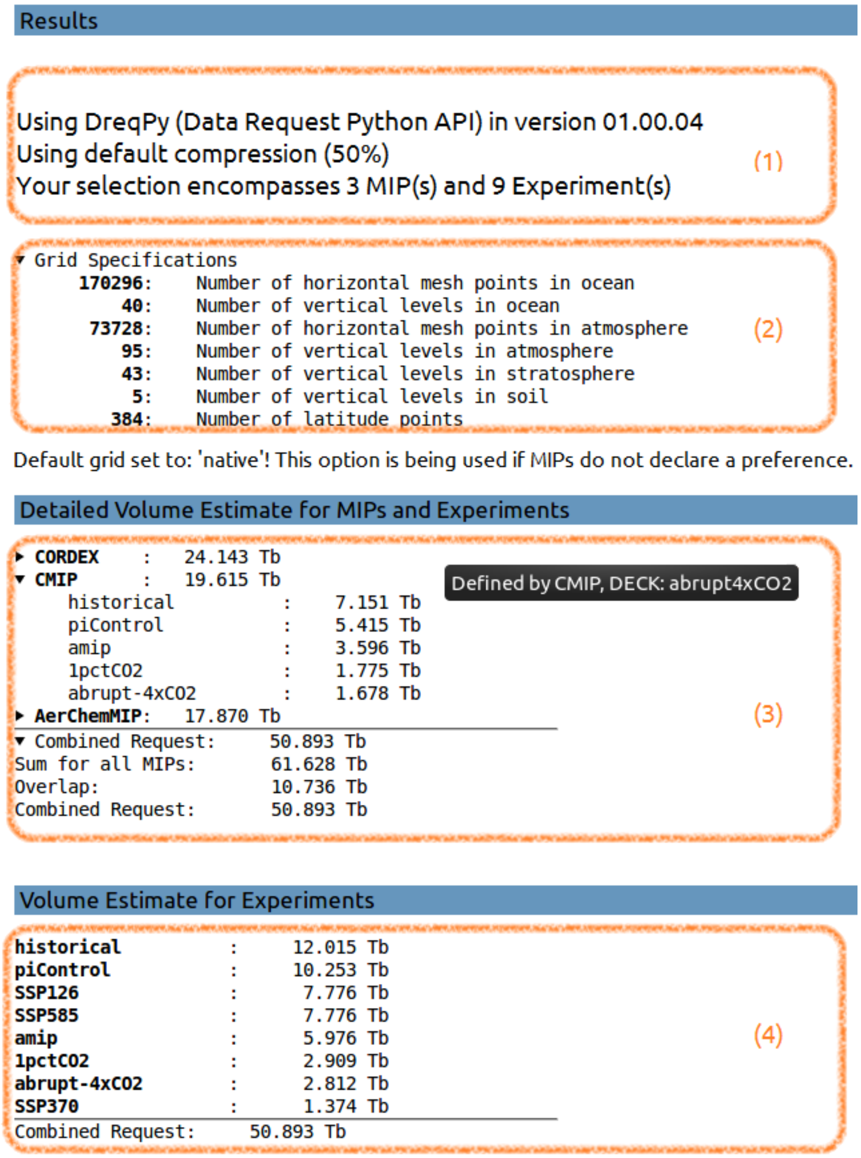
Depending on the number of chosen MIPs and experiments, the calculation may take up to a minute. An example result is shown in Fig. 2.9 and is explained in the following:

(1) Basic information about the selected MIPs, experiments and the DreqPy-version as well as the compression ratio. When hovering over 'MIP(s)' or 'Experiment(s)' additional information about the MIPs and experiments will pop up.

(2) The grid specifications are repeated here.

(3) For every MIP a volume estimate will be displayed. The volume estimate is subdivided into volume per experiment. MIPs or experiments with no requested data are not shown. The combined request is the sum of volume estimates for the different MIPs minus a possible overlap, as some of the MIPs request the same variables.

(4) The data request volume estimate for each experiment is displayed here. When hovering over the experiment label, additional information will pop up.



Results

Using DreqPy (Data Request Python API) in version 01.00.04
Using default compression (50%)
Your selection encompasses 3 MIP(s) and 9 Experiment(s)

```
▾ Grid Specifications
      170296:    Number of horizontal mesh points in ocean
          40:    Number of vertical levels in ocean
       73728:    Number of horizontal mesh points in atmosphere
          95:    Number of vertical levels in atmosphere
          43:    Number of vertical levels in stratosphere
           5:    Number of vertical levels in soil
         384:    Number of latitude points
```

Default grid set to: 'native'! This option is being used if MIPs do not declare a preference.

Detailed Volume Estimate for MIPs and Experiments

```
▸ CORDEX     :   24.143 Tb
▾ CMIP       :   19.615 Tb
     historical          :     7.151 Tb
     piControl           :     5.415 Tb
     amip                :     3.596 Tb
     1pctCO2             :     1.775 Tb
     abrupt-4xCO2        :     1.678 Tb
▸ AerChemMIP:    17.870 Tb
▾ Combined Request:      50.893 Tb
Sum for all MIPs:        61.628 Tb
Overlap:                 10.736 Tb
Combined Request:        50.893 Tb
```

Volume Estimate for Experiments

```
historical          :     12.015 Tb
piControl           :     10.253 Tb
SSP126              :      7.776 Tb
SSP585              :      7.776 Tb
amip                :      5.976 Tb
1pctCO2             :      2.909 Tb
abrupt-4xCO2        :      2.812 Tb
SSP370              :      1.374 Tb
Combined Request:      50.893 Tb
```

Figure 2.9: *Example result of a data request volume estimate.*

3 Variable Mapping Web GUI

Variables are usually named differently in the raw output of climate models and the CMIP variable definitions. Therefore, a mapping, connecting CMOR-variables to their model counterparts, is needed when calling the CMOR library to re-write the model output in CMIP and CF compliant format.

In order to simplify the task of creating such a mapping, the Variable Mapping Web GUI has been developed, allowing the modelers to produce this variable mapping online. When CMOR variables cannot be mapped directly to a model variable, also diagnostics can be entered, with more than one model variable as input.

The mapping will be stored in a database and can be downloaded in form of a mapping table, which is used by the 'cdo cmor' operator (Wachsmann (2017)). At the same time, a recipe table will be produced, which serves for an automated creation of post processing scripts for diagnostic and the CMIP standard conform rewrite of data using cdo cmor.

3.1 Variable Mapping - Basics

The tab '**VariableMapping**' gives access to the **Variable Mapping Web GUI**. Here, a selection menu is provided: The models and options are organized in a table with three columns (Fig. 3.1).

Two possible activities can be chosen for a model and submodel combination:

1. Continue to enter variable mapping information for a model-submodel-combination (eg. MPI-ESM1.2 and ECHAM6).
2. Generate Mapping Table files for a model-submodel-combination.

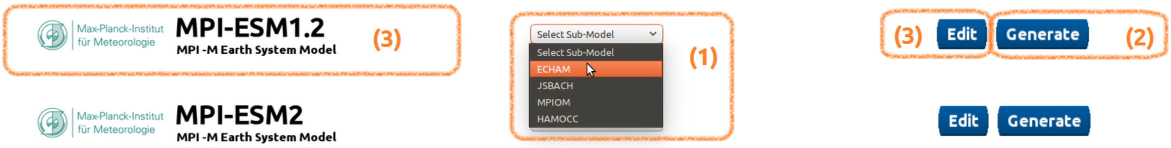
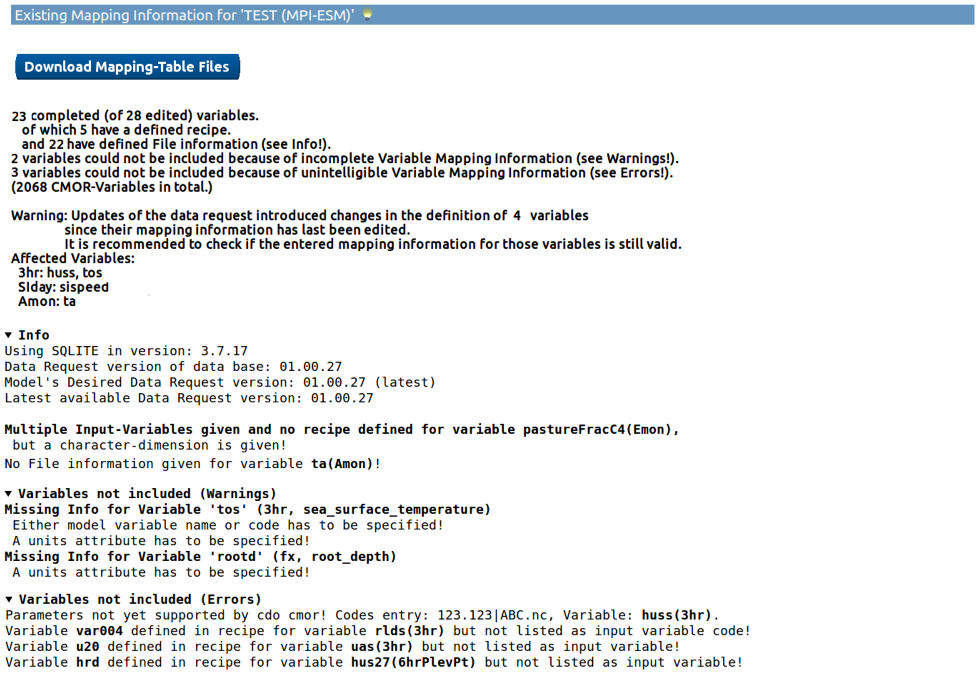

Max-Planck-Institut für Meteorologie
MPI-ESM1.2
MPI -M Earth System Model
(3)
Max-Planck-Institut für Meteorologie
MPI-ESM2
MPI -M Earth System Model
Select Sub-Model
Select Sub-Model
ECHAM
JSBACH
MPIOM
HAMOCC
(1)
(3)
Edit
Generate
(2)
Edit
Generate

Figure 3.1: *Still of the 'VariableMapping'-Tab. Available models are organized in a table view. At first, a submodel has to be chosen in the middle column of the appropriate row (1). Then the Mapping-Table files can be generated using button (2) of this row. Alternatively the mapping table can be accessed to fill in mapping-information for the selected model/submodel-combination by clicking on one of the buttons labeled (3).*

At first, a submodel has to be selected in the model's row (see Fig. 3.1, (1)). To generate Mapping-Table files, the button 'Generate' (2) has to be clicked thereafter. These files are crucial for the CMIP-conform formatted rewrite of the climate data. Their content is listed further below. In order to access the mapping database where the (sub)model's variable mapping information can be entered, the button 'Edit' or the one labeled with the model name (3) has to be clicked.

When generating Mapping-Table files

If the variable mapping information for at least one variable is present in the database, a file will be created and a download button will appear (Fig. 3.2).


Existing Mapping Information for 'TEST (MPI-ESM)'
Download Mapping-Table Files

```
23 completed (of 28 edited) variables.
  of which 5 have a defined recipe.
  and 22 have defined File information (see Info!).
2 variables could not be included because of incomplete Variable Mapping Information (see Warnings!).
3 variables could not be included because of unintelligible Variable Mapping Information (see Errors!).
(2068 CMOR-Variables in total.)

Warning: Updates of the data request introduced changes in the definition of 4  variables
           since their mapping information has last been edited.
           It is recommended to check if the entered mapping information for those variables is still valid.
Affected Variables:
 3hr: huss, tos
 SIday: sispeed
 Amon: ta

▾ Info
Using SQLITE in version: 3.7.17
Data Request version of data base: 01.00.27
Model's Desired Data Request version: 01.00.27 (latest)
Latest available Data Request version: 01.00.27

Multiple Input-Variables given and no recipe defined for variable pastureFracC4(Emon),
 but a character-dimension is given!
No File information given for variable ta(Amon)!

▾ Variables not included (Warnings)
Missing Info for Variable 'tos' (3hr, sea_surface_temperature)
 Either model variable name or code has to be specified!
 A units attribute has to be specified!
Missing Info for Variable 'rootd' (fx, root_depth)
 A units attribute has to be specified!

▾ Variables not included (Errors)
Parameters not yet supported by cdo cmor! Codes entry: 123.123|ABC.nc, Variable: huss(3hr).
Variable var004 defined in recipe for variable rlds(3hr) but not listed as input variable code!
Variable u20 defined in recipe for variable uas(3hr) but not listed as input variable!
Variable hrd defined in recipe for variable hus27(6hrPlevPt) but not listed as input variable!
```

Figure 3.2: *When generating Mapping-Table files for one of the model-submodel combinations a download button and further information will be displayed.*

The created file is a .zip-file containing:

- Mapping table in .txt-format.
- Mapping table in .json-format.
- Recipe table in .txt-format.
- Recipe table in .json-format.

The mapping table contains the basic mapping information consisting of:

- CMOR variable label
- Model variable label / code
- MIP-table label
- Cell-method identifier
- Character axis label

- Variable comment
- Positive attribute (flux direction)
- Variable units

The recipe table additionally contains information about:

- CMOR variable unique ID (in DreqPy)
- CMOR variable standard-name
- CMOR variable frequency
- Diagnostic recipe
- Input file
- Remark / note

If variables cannot be included in the mapping info files, warning and error messages are displayed below the download button (Fig. 3.2). Warnings signify missing information. Errors signify that entered values cannot be interpreted.

The mapping table serves as input for the 'cdo cmor' operator (Wachsmann (2017)) and the recipe table serves as input for the CMIP6 post processing workflow consisting of diagnostic as well as the CF- and CMIP-conform rewrite of the aggregated model output using CMOR (through the cdo-cmor operator).

When accessing the Variable Mapping Table

As stated above, the variable mapping table can be accessed by clicking on the 'Edit' button or the button displaying the desired model label, respectively, after selecting one of its submodels.

To access the variable mapping table, an authorized user account is required.

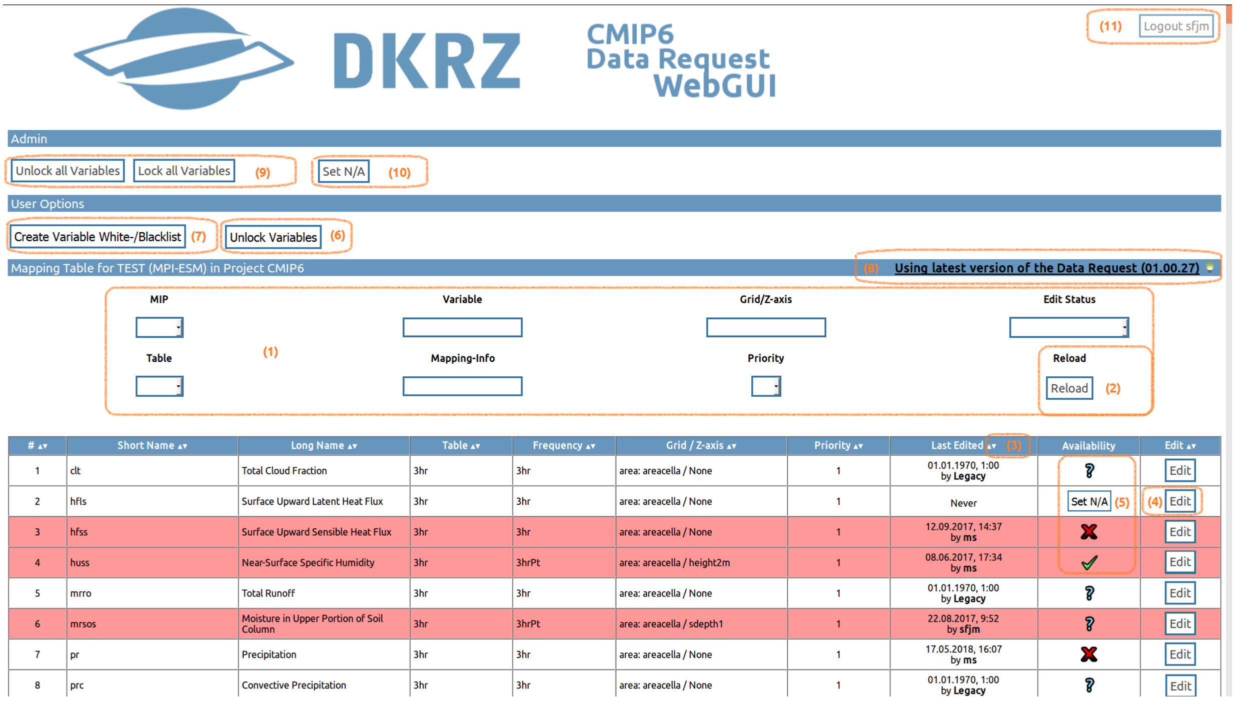
Access can be requested by using the Register functionality. Registrations have to be approved by a site admin. Feedback can be given via the feedback form of the **Data Request Web GUI**.

A prototype of the web-based interface for the Variable Mapping database has been developed by Karl-Hermann Wieners (MPI-M). This prototype has been considerably extended, allowing for example to track changes in variable definitions and data requests which are subject to continuous updates.

3.2 Variable Mapping - The Table View

The main view of the variable mapping database is the table view (Fig. 3.3). The different table columns hold the following information:

- Short Name: CMOR-/MIP-table entry name
- Long Name: variable title
- Frequency
- Table: CMOR-/MIP-table name.
- Grid/Z-axis: cell methods and z-axis are displayed where applicable.
- Priority: The highest priority (= lowest number) with which the variable is requested by any MIP for any experiment.
- Last Edited: Date and user account of the last editing action. When hovering over this entry an editor's note appears if available (see Fig. 3.5).
- Availability: CMOR-variables can be marked as available or unavailable, this information is displayed here. When hovering over the symbol all variable mapping information already entered will be shown (see Fig. 3.4).
- Edit: Opens a new tab with the edit view of the chosen variable.



# ▴▾	Short Name ▴▾	Long Name ▴▾	Table ▴▾	Frequency ▴▾	Grid / Z-axis ▴▾	Priority ▴▾	Last Edited ▴▾	Availability	Edit ▴▾
1	clt	Total Cloud Fraction	3hr	3hr	area: areacella / None	1	01.01.1970, 1:00 by **Legacy**	?	Edit
2	hfls	Surface Upward Latent Heat Flux	3hr	3hr	area: areacella / None	1	Never	Set N/A	Edit
3	hfss	Surface Upward Sensible Heat Flux	3hr	3hr	area: areacella / None	1	12.09.2017, 14:37 by **ms**	✗	Edit
4	huss	Near-Surface Specific Humidity	3hr	3hrPt	area: areacella / height2m	1	08.06.2017, 17:34 by **ms**	✓	Edit
5	mrro	Total Runoff	3hr	3hr	area: areacella / None	1	01.01.1970, 1:00 by **Legacy**	?	Edit
6	mrsos	Moisture in Upper Portion of Soil Column	3hr	3hrPt	area: areacella / sdepth1	1	22.08.2017, 9:52 by **sfjm**	?	Edit
7	pr	Precipitation	3hr	3hr	area: areacella / None	1	17.05.2018, 16:07 by **ms**	✗	Edit
8	prc	Convective Precipitation	3hr	3hr	area: areacella / None	1	01.01.1970, 1:00 by **Legacy**	?	Edit

Figure 3.3: *'VariableMapping' table view: At first, all 2000 and more CMOR variables of the CMIP6 Data Request are displayed in a table view. Filters (1) and the sort function (3) help to identify the variables of interest. A click on the button '**Edit**' will open the edit view of this row's variable.*

Hereafter, the highlighted areas in Fig. 3.3 will be explained:

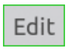
(1) When accessing the table view, all 2000 and more CMOR variables are displayed. The *filters* help to narrow this list down to the variables of interest. The filter '**Variable**' sifts through the CMOR variable's

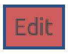
short name and long name for the entered string. The '**Mapping Info**' filter helps to identify variables by the already entered model mapping information (variable name, code, inputfile, recipe, comment, note). For the text search filters, wildcard characters may be used. These are '_' for single characters and '%' for an arbitrary number of characters. The filter listed in the select box '**Edit Status**' helps to identify variables with missing mapping information. As usual, when hovering over the filters and their options, more information is being displayed and will help to become familiar with their use.

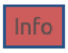
(2) When applying a filter or after editing a variable, the button '**Reload**' has to be clicked to refresh the table view.

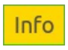
(3) The columns can be sorted by clicking on the upward and downward arrows in the column header.

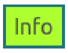
(4) When clicking on the 'Edit' button of a row, the edit view (see next section 3.3) of this variable will open up in a new tab. The variable will be locked, so no other user can make changes to it, until the edit view has been closed properly. Also similar variables that have not been edited will be locked (more information in the next section 3.3). Depending on the 'lock status' of a variable the button '**Edit**' will change its layout:

- 

This variable is currently not locked and can be edited by any user.
- 

This variable is currently being edited by oneself.
- 

This variable is currently being edited by oneself as similar variable (more information in the next section 3.3).
- 

This variable is currently being edited by another user.
- 

This variable has been locked by an admin as its variable mapping information is considered as being complete.

When clicking on buttons labeled '**Info**' the edit view will still open in a new tab, but the variable mapping information cannot be modified. Hovering over the buttons will display information about who locked the variable and when it has been locked.

(5) There are three symbols and one button possible in the column '**Availability**'. They indicate whether the variable is available in the model, unavailable in the model or its availability in the model has not been declared yet though any mapping information has been already entered. When a variable has no entered model mapping information at all, a button is displayed which sets this variable to unavailable in the model when clicked. Variables that are not set as 'available' will not be included in the mapping and recipe tables, even if their mapping information is complete.

(6) This button serves to unlock all variables locked by the user. Usually this happens automatically when the user closes the edit view properly.

(7) This button opens a new tab where a Black-/Whitelist can be created that allows to exclude variables from the post processing (see chapter 4) that are set as 'available in the model' (more information in the next section 3.5).

(8) Here the currently used version of the CMIP6 Data Request is displayed. The CMIP6 Data Request and the underlying DreqPy software are possibly subject to updates. Also the definitions of CMOR variables may change with a new version of the data request. Information about variable definition changes will be given in the edit view of a variable. However, when a new variable is introduced, the respective row in the table view will be highlighted in green. When the definition of an already edited variable gets updated, the respective row will be highlighted in red, until someone confirms the entered variable mapping information for this variable (in the edit view). Variables that have been locked by an admin and get a definition update, will be unlocked automatically. There is also a filter in the '**Edit Status**' select box (1), displaying only variables that are highlighted in green and red. The displayed information about the data request version is also a link, leading to an overview of the variable definition changes introduced with data request updates (see section 3.4).

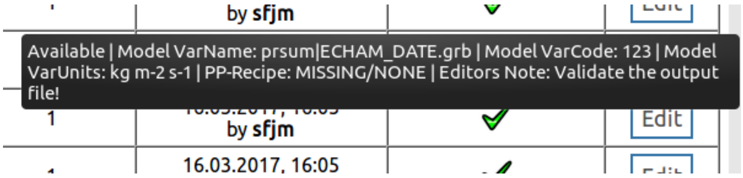
(9) Admins have two more buttons to unlock and lock variables. The button '**Unlock all Variables**' unlocks all variables currently locked for editing. The button '**Lock all Variables**' locks all variables that have so far been edited.

(10) Additionally, admins have the option to set a number of variables at once to unavailable in the model. The button 'Set N/A' affects all variables that match every of the following conditions:

- The variable matches the set filters 'Variable' and 'Table'.
- The variable is currently not locked by any user, including oneself.
- The variable has never been edited by any user.

Therefore, this option is helpful to set an entire MIP table to unavailable in the model, for example 'E1hr' when the model does not produce hourly output.

[11) The logout button will unlock all variables locked by the user and then log of the user's account.



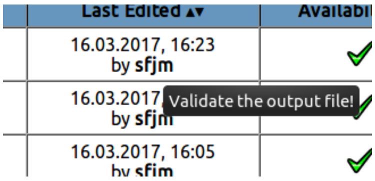
by sfjm

Available | Model VarName: prsum|ECHAM_DATE.grb | Model VarCode: 123 | Model VarUnits: kg m-2 s-1 | PP-Recipe: MISSING/NONE | Editors Note: Validate the output file!

1 | by sfjm | ✓ | Edit

16.03.2017, 16:05

Figure 3.4: *'VariableMapping' table view: Hovering over the Availability symbol (5 in Fig. 3.3) will cause the so far entered mapping information to be displayed.*



Last Edited ▴▾	Availabi
16.03.2017, 16:23 by sfjm	✓
16.03.2017, Validate the output file! by sfjm	✓
16.03.2017, 16:05 by sfjm	✓

Figure 3.5: *'VariableMapping' table view: Hovering over an entry in the column 'Last Edited' will display an entered editor's note.*

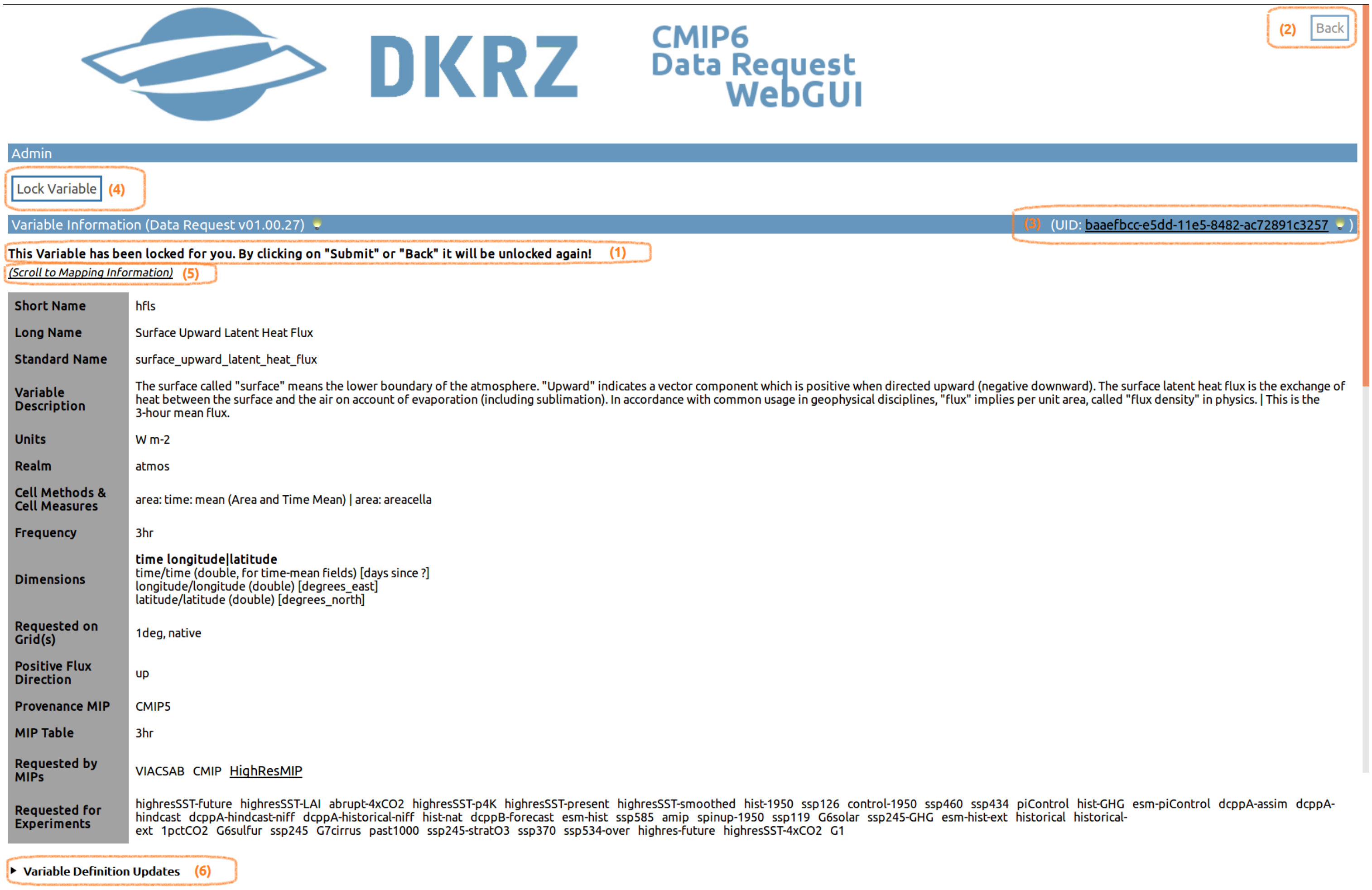
3.3 Variable Mapping - The Edit View

The edit view of a CMOR variable consists of two parts. The upper half of the web page is the 'Variable Information' section, displaying all information about the CMOR variable, as it is defined in the CMIP6 data request.

The lower half of the website contains the 'Variable Mapping Information', where the variable in the climate model output corresponding to the above described CMOR variable has to be entered.

Variable Information

An example for three-hourly precipitation is shown in Fig. 3.6. The main part of the Variable Information is a two column table holding all necessary attributes of the CMOR variable which shall be mapped to a model variable.


DKRZ
CMIP6
Data Request
WebGUI
(2) Back
Admin
Lock Variable (4)
Variable Information (Data Request v01.00.27)
(3) (UID: baaefbcc-e5dd-11e5-8482-ac72891c3257)

This Variable has been locked for you. By clicking on "Submit" or "Back" it will be unlocked again! (1)

(Scroll to Mapping Information) (5)

Short Name	hfls
Long Name	Surface Upward Latent Heat Flux
Standard Name	surface_upward_latent_heat_flux
Variable Description	The surface called "surface" means the lower boundary of the atmosphere. "Upward" indicates a vector component which is positive when directed upward (negative downward). The surface latent heat flux is the exchange of heat between the surface and the air on account of evaporation (including sublimation). In accordance with common usage in geophysical disciplines, "flux" implies per unit area, called "flux density" in physics. | This is the 3-hour mean flux.
Units	W m-2
Realm	atmos
Cell Methods & Cell Measures	area: time: mean (Area and Time Mean) | area: areacella
Frequency	3hr
Dimensions	**time longitude|latitude** time/time (double, for time-mean fields) [days since ?] longitude/longitude (double) [degrees_east] latitude/latitude (double) [degrees_north]
Requested on Grid(s)	1deg, native
Positive Flux Direction	up
Provenance MIP	CMIP5
MIP Table	3hr
Requested by MIPs	VIACSAB CMIP HighResMIP
Requested for Experiments	highresSST-future highresSST-LAI abrupt-4xCO2 highresSST-p4K highresSST-present highresSST-smoothed hist-1950 ssp126 control-1950 ssp460 ssp434 piControl hist-GHG esm-piControl dcppA-assim dcppA-hindcast dcppA-hindcast-niff dcppA-historical-niff hist-nat dcppB-forecast esm-hist ssp585 amip spinup-1950 ssp119 G6solar ssp245-GHG esm-hist-ext historical historical-ext 1pctCO2 G6sulfur ssp245 G7cirrus past1000 ssp245-stratO3 ssp370 ssp534-over highres-future highresSST-4xCO2 G1

▸ Variable Definition Updates (6)

Figure 3.6: *'VariableMapping' edit view: Example of the upper half of the edit view, taken by the 'Variable Information' of the CMOR variable.*

The highlighted points in Fig. 3.6 are explained in the following:

(1) A message reminds the user that this variable has been locked (possibly along with several similar variables). No other user can edit this variable while it is locked. In order to close the edit view and

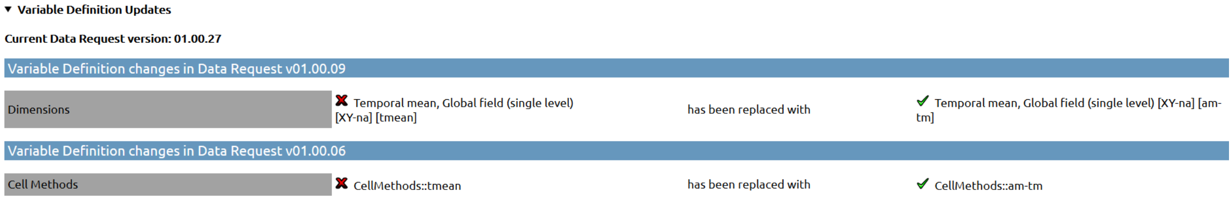
unlock the variable (without performing any changes on the variable mapping information) one has to use the 'Back' button (2). There are two buttons of this kind on the web page, one at the top and one at the bottom. If a variable is locked by another user, an admin, or as a similar variable, there will be another message notifying that this variable cannot be edited at the moment.

(3) The displayed UID is a link to the DreqPy variable website, which basically contains the same amount of information and additionally provides links to other sections of the CMIP6 data request (experiments, MIPs, request items, ...).

(4) Admin button to lock or unlock the variable. Variables that are locked in this way will have a green info button instead of an edit button (see Sec. 3.2).

(5) is a link to jump to the 'Variable Mapping Information'.

(6) signifies that this variables definition was subject to updates in the past. When expanding the 'Variable Definition Updates' all updates in the CMIP6 Data Request up to the latest version will be displayed (Fig. 3.7). Additionally a warning will be displayed if the definition has changed since the last edit of the variable's mapping information (Fig. 3.8).

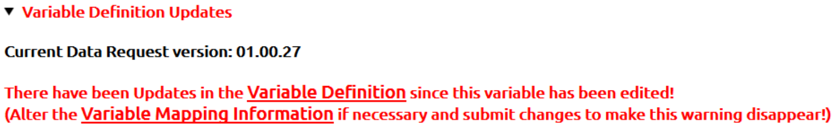


▼ Variable Definition Updates

Current Data Request version: 01.00.27

Variable Definition changes in Data Request v01.00.09			
Dimensions	✗ Temporal mean, Global field (single level) [XY-na] [tmean]	has been replaced with	✓ Temporal mean, Global field (single level) [XY-na] [am-tm]
Variable Definition changes in Data Request v01.00.06			
Cell Methods	✗ CellMethods::tmean	has been replaced with	✓ CellMethods::am-tm

Figure 3.7: *'Variable Definition Updates': When expanding the 'Variable Definition Updates' all updates in the CMIP6 Data Request up to the latest version will be displayed.*



▼ Variable Definition Updates

Current Data Request version: 01.00.27

There have been Updates in the Variable Definition since this variable has been edited!
(Alter the Variable Mapping Information if necessary and submit changes to make this warning disappear!)

Figure 3.8: *'Variable Definition Updates': If the definition changed since the mapping information for this variable has been altered, a warning message is shown. In this case the variable's row in the table view would be coloured red (Fig. 3.3).*

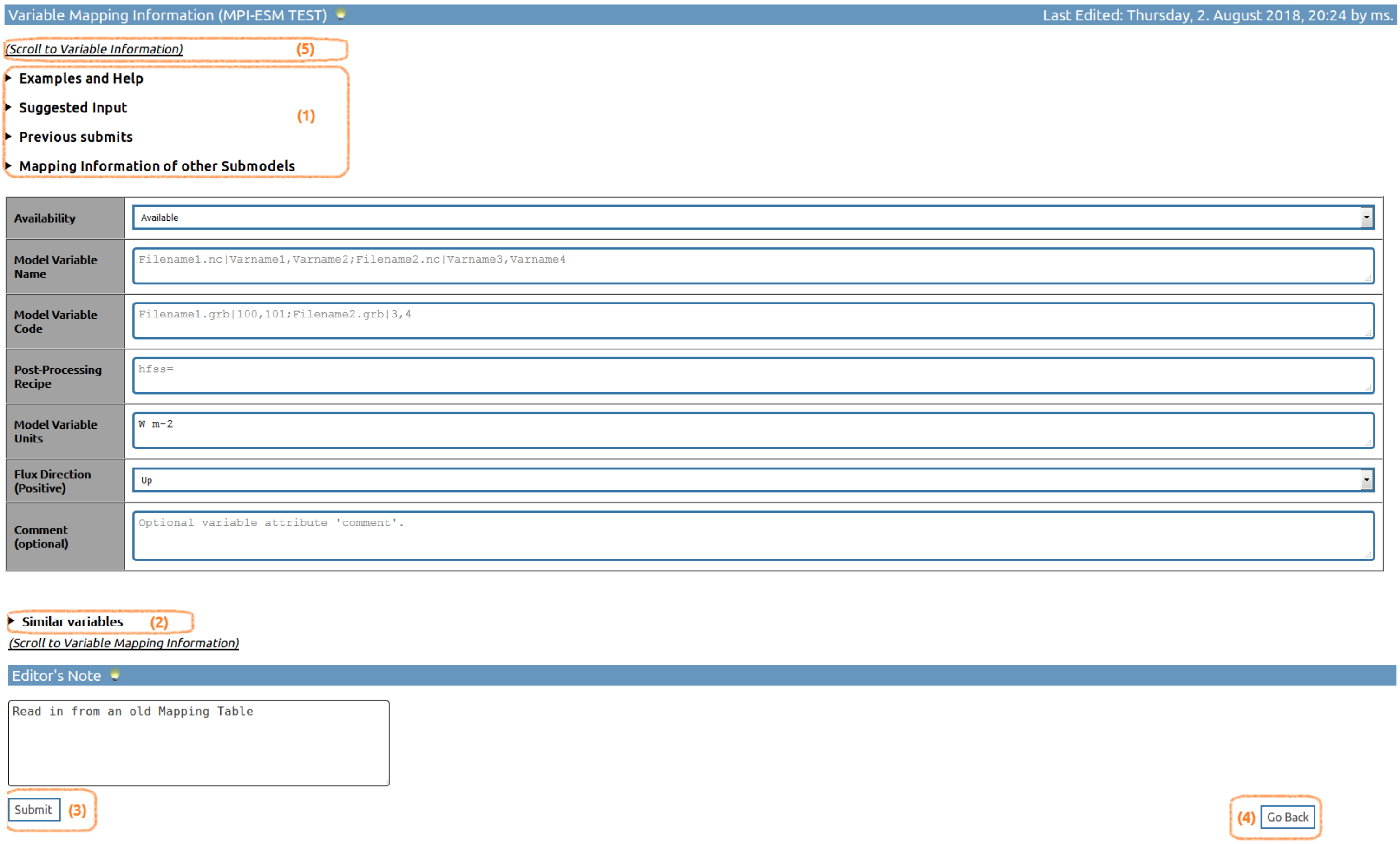
Variable Mapping Information

The 'Variable Mapping Information' section at the bottom comes up with several select and text boxes to enter information about the **model counterpart** of above defined CMOR variable:

- **Availability**: Select, whether the CMOR variable has a model counterpart or not. When nothing or 'unavailable' is selected, the variable will appear neither in the mapping nor in the recipe table, even when the remaining information has been filled in.
- **Model Variable Name**: Here, the name (or names) of the model counterpart of the actual CMOR variable has to be entered. Also a unique identifier for the model output file containing this variable has to be given.

- **Model Variable Code**: If the model output is in GRIB format, the variable's code and file information has to be entered here. Both, '**Model Variable Name**' and '**Model Variable Code**' can be given if a recipe is defined (see below or the examples at the end of this section). When a '**Model Variable Code**' is defined, it is assumed that the output file is in GRIB format.
- **Post-Processing Recipe**: Here, a short diagnostic recipe can be entered using the format of the cdo operator 'expr'. Variables from different input files with different file formats are supported.
- **Model Variable Units**: Here, the model variable's units have to be entered. On its 'journey' along the post processing chain the units of a variable may change. The units to be entered correspond to the variable state after aggregation and diagnostic have been performed (and before the variable will be passed on to the CMOR rewrite process).
- **Positive / Flux Direction**: For some variables a flux direction (or: positive attribute) has to be entered. This is the direction with respect to the earth's surface in which the flux direction is considered as positive.
- **Comment** (CMOR Variable Attribute): Here, a comment can be entered which will be written as variable attribute into the output file, during the CMOR rewrite process.
- **Editor's Note**: The editor's note has no purpose for the variable mapping itself and serves solely as additional information for other editors of this variable or as a reminder for oneself. If additionally a recipe is defined, the editor's note will be written as comment into the diagnostic script fragment.

The user input in the text boxes has to be in a certain format to be able to capture and process it programmatically. More information about the input format will be given below. Hereafter, the highlighted points from Fig. 3.9 shall be explained:



Variable Mapping Information (MPI-ESM TEST)

Last Edited: Thursday, 2. August 2018, 20:24 by ms.

(Scroll to Variable Information)

Availability	Available
Model Variable Name	Filename1.nc|Varname1,Varname2;Filename2.nc|Varname3,Varname4
Model Variable Code	Filename1.grb|100,101;Filename2.grb|3,4
Post-Processing Recipe	hfss=
Model Variable Units	W m-2
Flux Direction (Positive)	Up
Comment (optional)	Optional variable attribute 'comment'.

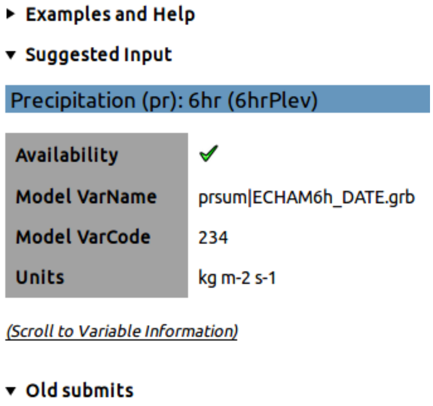
(Scroll to Variable Mapping Information)

Editor's Note

Read in from an old Mapping Table

Figure 3.9: 'Variable Mapping Information' in the lower half of the edit view.

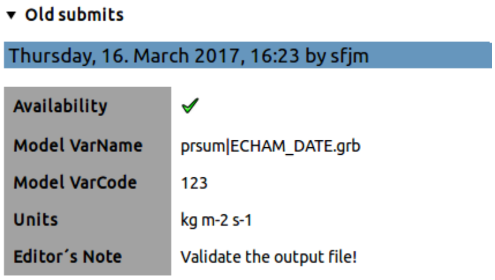
Under (1) several options are listed that may help to enter the variable's mapping information. '**Examples and Help**' provide, as the title suggests, detailed examples on how to fill in the variable's mapping information. '**Suggested Input**' lists all similar variables which already have been edited and displays their mapping information (see Fig. 3.10). Similar variables differ only in terms of frequency or requested grid (dimensions, axes, cell methods) from the variable one is currently editing. Thus it is very likely that also the model counterpart is similar and one can copy and paste some of the suggested input in the textboxes below. '**Previous submits**' allow the editors to follow the history of changes of the variable's mapping information (see Fig. 3.11). '**Mapping Information of other Submodels**' lists entered mapping information that has been entered for other submodels that are registered for the current model. The user has the possibility to open the Edit View of the same variable by clicking on the Edit button. A click on the model name will open the model's table view (see Fig. 3.12).



Precipitation (pr): 6hr (6hrPlev)

Availability	✓
Model VarName	prsum|ECHAM6h_DATE.grb
Model VarCode	234
Units	kg m-2 s-1

Figure 3.10: Example of an expanded list of 'Suggested Input', displaying already entered mapping information from similar variables.



Thursday, 16. March 2017, 16:23 by sfjm

Availability	✓
Model VarName	prsum|ECHAM_DATE.grb
Model VarCode	123
Units	kg m-2 s-1
Editor´s Note	Validate the output file!

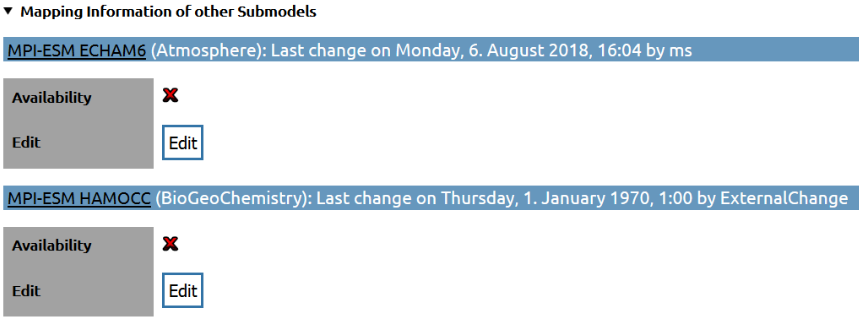
Figure 3.11: Example of an expanded list of 'Previous submits'.

A click on '**Similar variables**' (2) lists all yet unedited similar variables (see Fig. 3.13). For each variable one may choose whether to apply above entered variable mapping information as well (✓) or if this variable is not available in the model (✗).

A click on the button '**Submit changes**' (3) will store the entered mapping information in the database, unlock the previously locked variables and close the tab. There is a basic format check performed on the user input. If it fails, an error message will appear at the top of the website and the user has to change its input accordingly and submit his changes again.

The '**Back**' button (4) will unlock all previously locked variables and close the tab, without writing mapping information changes to the database. When just closing the tab by using the browser's own button ('x') the variables will stay locked.

Finally, the link 'Scroll to Variable Information' (5) will cause the browser to jump to the '**Variable Information**' section that occupies the upper half of the edit view.



▾ **Mapping Information of other Submodels**

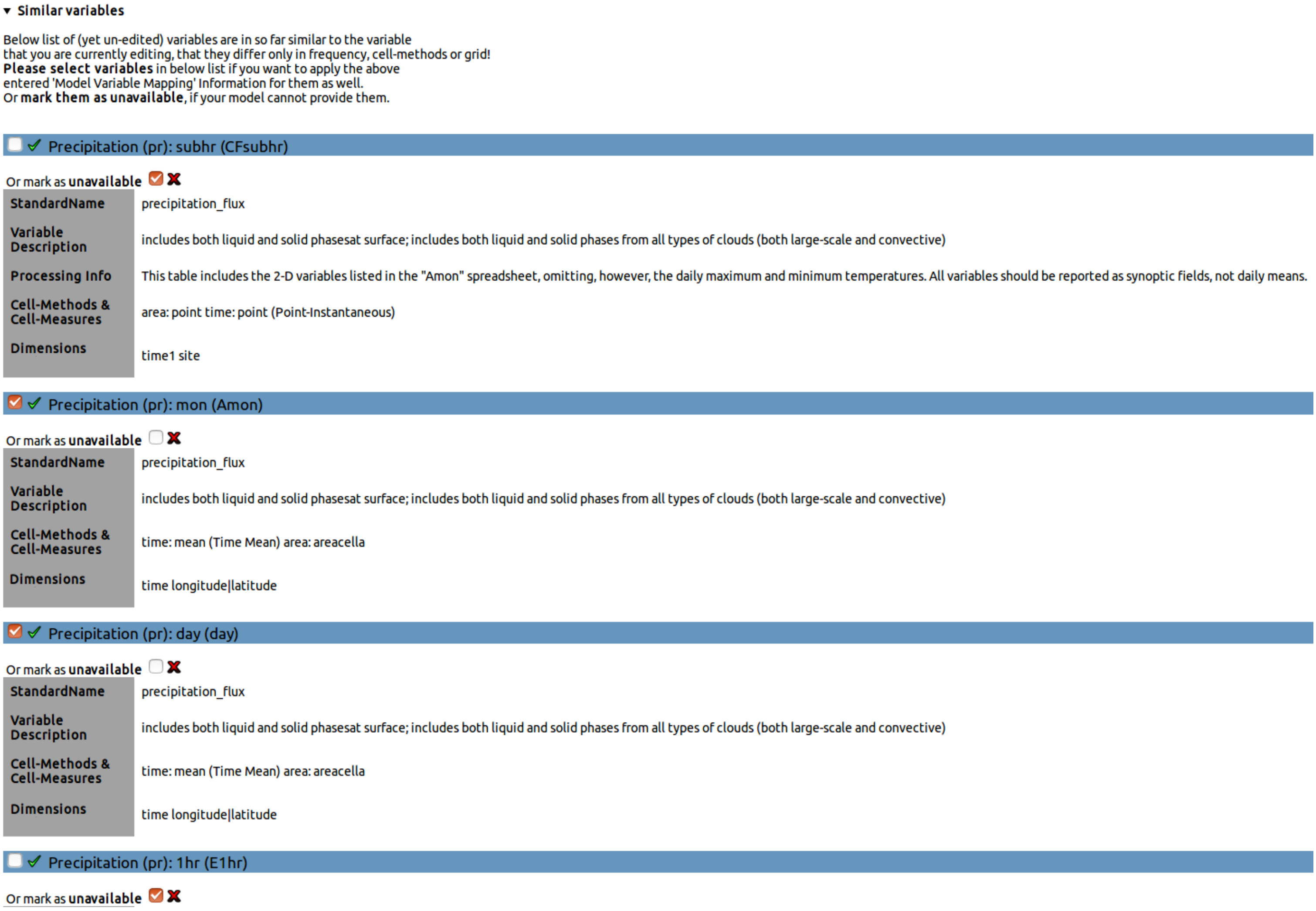
MPI-ESM ECHAM6 (Atmosphere): Last change on Monday, 6. August 2018, 16:04 by ms

Availability	✗
Edit	Edit

MPI-ESM HAMOCC (BioGeoChemistry): Last change on Thursday, 1. January 1970, 1:00 by ExternalChange

Availability	✗
Edit	Edit

Figure 3.12: *Example of an expanded list of 'Mapping Information of other Submodels'.*



▾ **Similar variables**

Below list of (yet un-edited) variables are in so far similar to the variable that you are currently editing, that they differ only in frequency, cell-methods or grid!
Please select variables in below list if you want to apply the above entered 'Model Variable Mapping' Information for them as well.
Or **mark them as unavailable**, if your model cannot provide them.

□ ✓ Precipitation (pr): subhr (CFsubhr)

Or mark as **unavailable** ☑ ✗

StandardName	precipitation_flux
Variable Description	includes both liquid and solid phasesat surface; includes both liquid and solid phases from all types of clouds (both large-scale and convective)
Processing Info	This table includes the 2-D variables listed in the "Amon" spreadsheet, omitting, however, the daily maximum and minimum temperatures. All variables should be reported as synoptic fields, not daily means.
Cell-Methods & Cell-Measures	area: point time: point (Point-Instantaneous)
Dimensions	time1 site

☑ ✓ Precipitation (pr): mon (Amon)

Or mark as **unavailable** □ ✗

StandardName	precipitation_flux
Variable Description	includes both liquid and solid phasesat surface; includes both liquid and solid phases from all types of clouds (both large-scale and convective)
Cell-Methods & Cell-Measures	time: mean (Time Mean) area: areacella
Dimensions	time longitude|latitude

☑ ✓ Precipitation (pr): day (day)

Or mark as **unavailable** □ ✗

StandardName	precipitation_flux
Variable Description	includes both liquid and solid phasesat surface; includes both liquid and solid phases from all types of clouds (both large-scale and convective)
Cell-Methods & Cell-Measures	time: mean (Time Mean) area: areacella
Dimensions	time longitude|latitude

□ ✓ Precipitation (pr): 1hr (E1hr)

Or mark as **unavailable** ☑ ✗

Figure 3.13: *Example of an expanded list of 'Similar variables'. The mapping information for the variable as is entered above may be marked as inherited (✓) or the variable may be marked as unavailable (✗).*

Examples
Below you find several examples that will help you to enter the information in the correct format:

- **Example 1 (General)**: Adding your counterpart for monthly mean air_temperature. Starting point: Variable 'mtemp' stored in 'piControl_base01_ECHAM6_198001_mm.nc'.
  - You select 'Available' in the 'Availability'-column.
  - You write 'EXPT_SIMU_ECHAM6_DATE_mm.nc|mtemp' in the 'Model Variable Name'-column.
  - '|' serves as separator between variable name and filename identifier, which is a string to identify the file that contains this variable. This string needs to be unambiguous as it will be used in the post processing workflow to identify the correct file.
  - More variables could be added by using commas as separators (see Example 3 or 5).
  - The following placeholders can be used for the filenames entered in the 'Model Variable Name' field:
    - 'DATE' is a placeholder for the date string.
    - 'SIMU' is a placeholder for the simulation (eg. base01 or r1i1f1p1).
    - 'EXPT' is a placeholder for the experiment (eg. piControl, historical).
    - 'MODX' is a placeholder for the model name (eg. echam, jsbach).
    - Besides you can use the common wildcard characters '*' and '?'.
  - 'Model Variable Code' field remains blank, as the output is in netCDF4 format.
  - You leave the 'Post-Processing Recipe'-column blank as no further diagnostic is required.
  - The variable's units are 'K', which you fill into the units column.
  - The model's air temperature has been systematically 2K too high, which has been corrected.
  - Therefore you add the comment 'Adjusted systematically too high values (-2.0 K globally)', which will be added as variable comment by CMOR.
  - You may add an 'Editor's Note', which serves as a reminder for you and other editors but has no influence on the mapping or post processing.
  - 'Similar Variables' are variables differing only in terms of aggregation and have no entered mapping information yet. In this example list of other 'air-temperature' instances can be expanded.
  - 'Apply above Variable Mapping Information' or 'Not available in the model' can be selected for each variable individually if appropriate.
  - In this way you might avoid entering the same information twice.
  - After a check for typos you click the 'Submit' Button, which closes the tab and updates the data base.
  - Do not forget to edit the other instances of 'air-temperature' you selected in the 'Similar Variables' list where necessary.
- **Example 2 (Code)**: Adding a code for grib files as input: Assume your variable has the code '23' and is known within the modeling group as 'no2'. Furthermore it is stored in files named similar to 'aprupt4xCO2_r1i1f1p1_AERCHEM_198001.grb':
  - As 'Model Variable Code'you enter ' EXPT_SIMU_AERCHEM_DATE.grb|23'.
  - The field for 'Model Variable Name' will be left blank, as the input file is of the GRIB and not netCDF4 format.

- **Example 3 (Recipe)**: Adding multiple variables as input and using a recipe: Assume the complete cloud fraction is diagnosed by adding fractions of large-scale ('clsf') and convective clouds ('cconf'), and that both variables are stored in a file named 'aprupt4xCO2_r1i1p1f1_CLOUD_198001.nc':
  - As 'Model Variable Name' you enter therefore: 'EXPT_SIMU_CLOUD_DATE.nc|clsf,cconf', knowing that you do not have to specify the input file twice when it is the same for all variables.
  - If one of the variables was stored in a .grb file, this variable would have to be filled in the 'Model Variable Code' field along with the respective filename.
  - As 'Post-Processing Recipe' you enter 'clsf+conf' which will be interpreted as 'cdo expr,"clt=clsf+cconf" Inputfile Outputfile'.
  - If you did specify two different input files for the variables, a 'cdo merge' for both files would be added automatically to the diagnostic script.
  - The recipes support only the cdo operator 'expr'. More complicated diagnostic has to be made outside of the CMIP6 support workflow or added directly in the automatically generated diagnostic script. For the latter case you may set 'CUSTOM' as 'Post-Processing Recipe', so a placeholder will be generated for this variable in the diagnostic script where you can fill in your custom diagnostic later. The editor's note will be added as comment in the diagnostic script.
- **Example 4 (Character Axis)**: Adding multiple variables as input for variables with character axis.
  - If the selected CMOR variable features a 'character axis', more than one variable may serve as input.
  - Examples are vegetation types and landuse types. For each vegetation type one input variable may be given. (Alternatively provide just one variable as input if it already has a proper character axis.)
  - Further settings have to be adjusted in the cdo cmor configuration files (Model configuration file in this case). For example the corresponding axes have to be defined in the cdo cmor configuration files before executing the post processing scripts or running cdo cmor!
  - Multiple variables have to be entered in the same format as shown above.
  - 'Post-Processing Recipes' for each of the input variables may be defined using a semi-colon as separator. Make sure to give a recipe for each of the variables then! eg. '$var001 + var002; var003; var004 \times 0.5$'.
- **Example 5 (Multiple Inputfiles): Adding variables from multiple files.**
  - This is only allowed when also entering a post-processing recipe or when a character axis is requested.
  - The format would be: 'FileA|var1,var2;FileB|var3,var4,var5' and so on.
  - It is also possible to enter variables in the Model Variable Name AND the Model Variable Code fields.
  - The file information may not be entered in the recipe. Only variable names and/or codes may be used. Eg. '$var23 + clf * 0.001$', when specifying 'Filexy.nc|clf' in the Model Variable Name field and 'FileC.grb|23' in the Model Variable Code field.
  - If you want to specify the entries of a requested character axis the recipe can be set up as in Example 4. Eg. '$treefrac3; var23 + var24; var3; treefrac2 + treefrac1 * 0.2$', when specifying 'Filexy.nc|treefrac1,treefrac2;Fileyz.nc|treefrac3' in the Model Variable Name field and 'FileC.grb|23,24' in the Model Variable Code field.

3.4 Variable Mapping - Variable Definition Changes

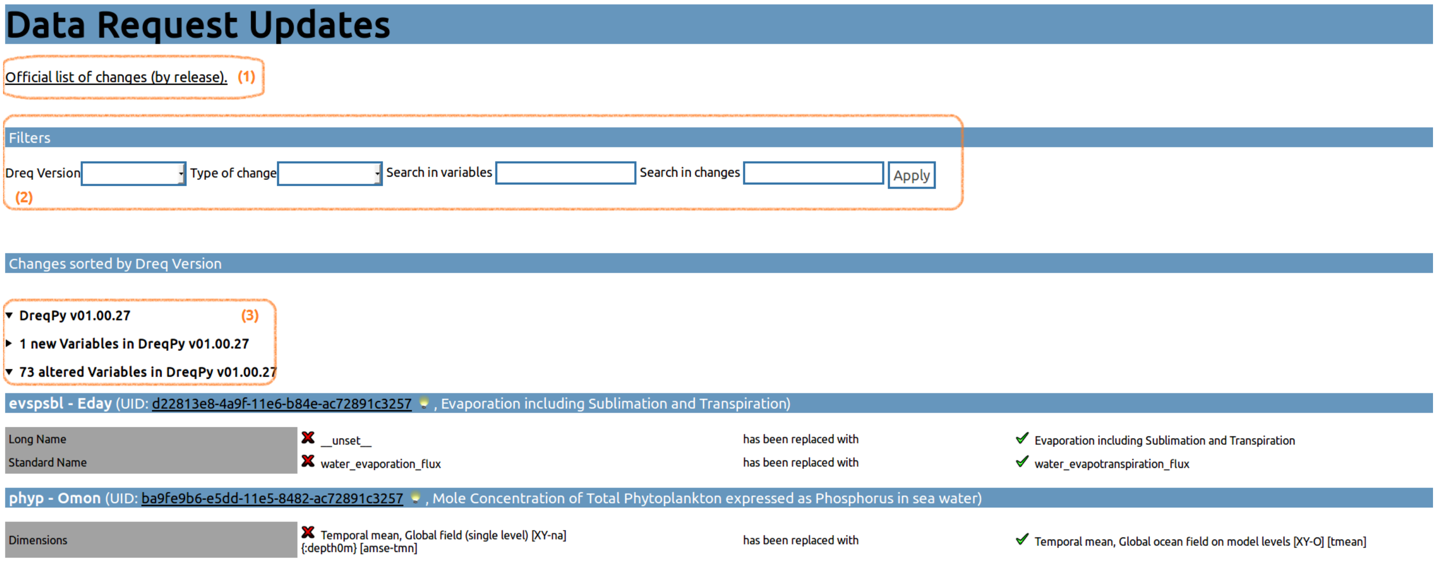
The 'Data Request Updates' list all changes to the variable definitions from one version to the next. On the basis of Fig. 3.14 the different buttons and filters will be explained:

(1) The official list of changes is linked at the top of the website.

(2) Filters narrow down the list of changed variables:

- Dreq Version: only the changes between the selected and the previous version of the data request will be shown.
- Type of change: it can be chosen between deleted, added and altered variable.
- Search in variables: the entered string will be compared with the latest variable name of each variable, and all changes for the matching variables will be listed. Wildcard characters can be used.
- Search in changes: one can enter any value that has been changed or the name of the attribute. For example, entering 'Units' would list all variable changes with altered Units attribute. Entering textquotesingle%water%' would list all variables where any attribute changed to or from a string containing 'water'.

(3) The changes are sorted by data request version and type of change. The list of changes gets expanded by clicking on the black triangle.



Data Request Updates

Official list of changes (by release). (1)

Filters

Dreq Version Type of change Search in variables Search in changes Apply

(2)

Changes sorted by Dreq Version

▼ DreqPy v01.00.27 (3)

▶ 1 new Variables in DreqPy v01.00.27

▼ 73 altered Variables in DreqPy v01.00.27

evspsbl - Eday (UID: d22813e8-4a9f-11e6-b84e-ac72891c3257 , Evaporation including Sublimation and Transpiration)

Long Name	✗ __unset__	has been replaced with	✓ Evaporation including Sublimation and Transpiration
Standard Name	✗ water_evaporation_flux	has been replaced with	✓ water_evapotranspiration_flux

phyp - Omon (UID: ba9fe9b6-e5dd-11e5-8482-ac72891c3257 , Mole Concentration of Total Phytoplankton expressed as Phosphorus in sea water)

Dimensions	✗ Temporal mean, Global field (single level) [XY-na] {:depth0m} [amse-tmn]	has been replaced with	✓ Temporal mean, Global ocean field on model levels [XY-O] [tmean]

Figure 3.14: Example of an expanded list of 'Similar variables'. The mapping information for the variable as is entered above may be marked as inherited

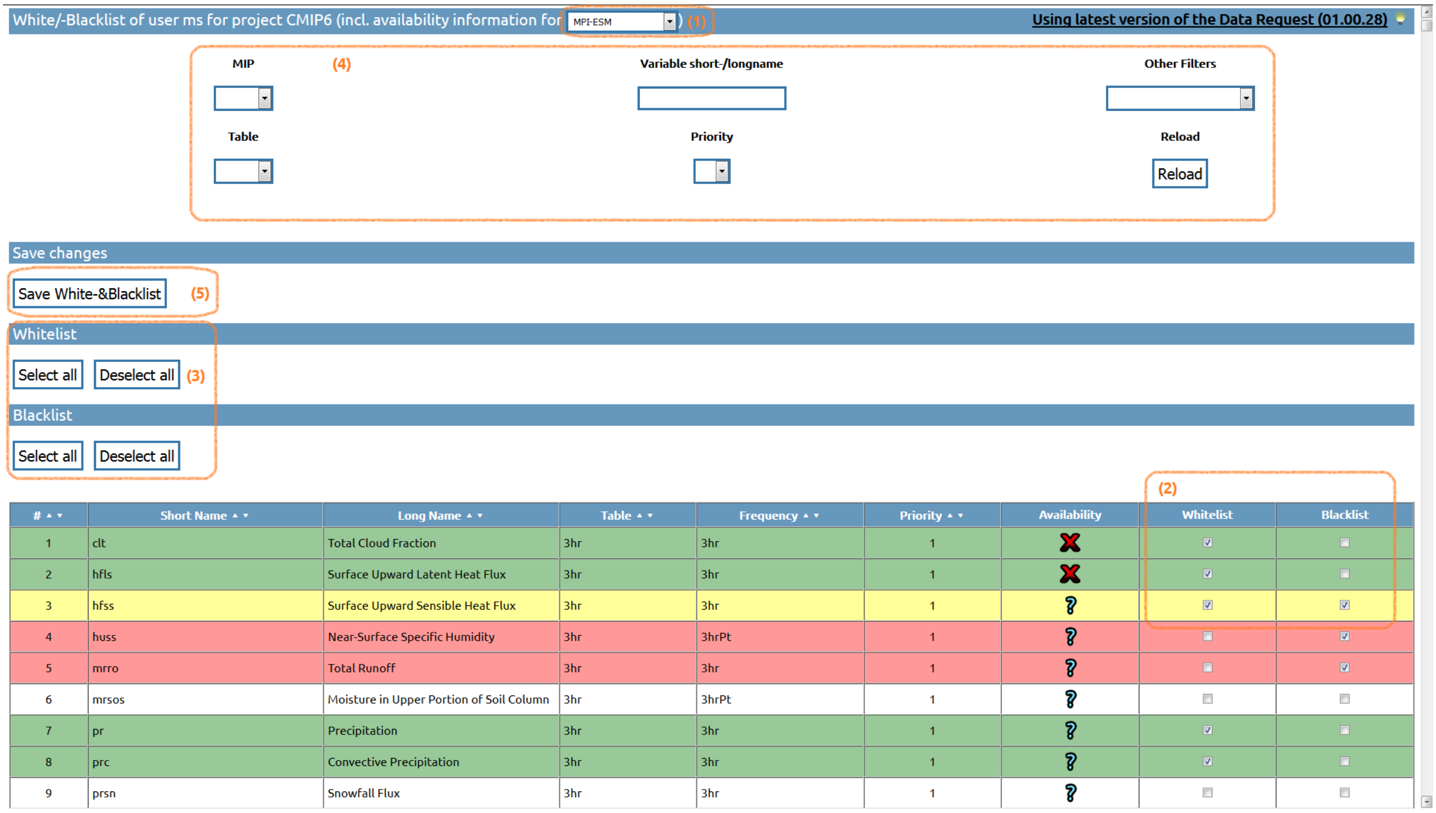
Example: I want to find out what happened to the variable 'prlq' that I cannot find any longer in the latest data request version.

- I enter 'prlq' in the text field of the filter 'Search for latest variable names'. The search returns no matches. This variable label does not exist anymore or this variable's definition was never subject to any changes.
- I enter 'prlq' in the text field of the filter 'Search for specific changes'. As result I get an entry for data request version 01.00.05: Label 'prlq' has been replaced with 'prra'.
- I enter 'prra' in the text field of the filter 'Search for specific changes'. I see that the label did not change anymore since version 01.00.05.
- I enter 'prra' in the text field of the filter 'Search for latest variable names'. As result I get all changes of this variable's definition between version 01.00.00 and the latest version.

3.5 Variable Mapping - Variable Black- and Whitelist

The 'Black- and Whitelist' allows to exclude variables from the post processing that are set to 'available in the model'. To apply the Black/-Whitelist, the respective option has to be enabled when creating the post processing script fragments (see chapter 4).

- Blacklist: Variables in the blacklist will be excluded from the post processing.
- Whitelist: Variables in the whitelist are the only ones included in the post processing.



White/-Blacklist of user ms for project CMIP6 (incl. availability information for MPI-ESM) (1)

Using latest version of the Data Request (01.00.28)

#	Short Name	Long Name	Table	Frequency	Priority	Availability	Whitelist	Blacklist
1	clt	Total Cloud Fraction	3hr	3hr	1	✗	☑	□
2	hfls	Surface Upward Latent Heat Flux	3hr	3hr	1	✗	☑	□
3	hfss	Surface Upward Sensible Heat Flux	3hr	3hr	1	?	☑	☑
4	huss	Near-Surface Specific Humidity	3hr	3hrPt	1	?	□	☑
5	mrro	Total Runoff	3hr	3hr	1	?	□	☑
6	mrsos	Moisture in Upper Portion of Soil Column	3hr	3hrPt	1	?	□	□
7	pr	Precipitation	3hr	3hr	1	?	☑	□
8	prc	Convective Precipitation	3hr	3hr	1	?	☑	□
9	prsn	Snowfall Flux	3hr	3hr	1	?	□	□

Figure 3.15: *Example of the Black- and Whitelist creation page. Filters can be used to reduce the number of the displayed variables. After selecting the desired variables as part of the Black- and/or Whitelist (2) the changes have to be saved by pressing the button (5).*

The view in Fig. 3.15 is similar to the variable mapping table view (see section 3.2). A list of all CMOR variables in the data request is displayed with several filters and buttons to edit the Black-/Whitelist:

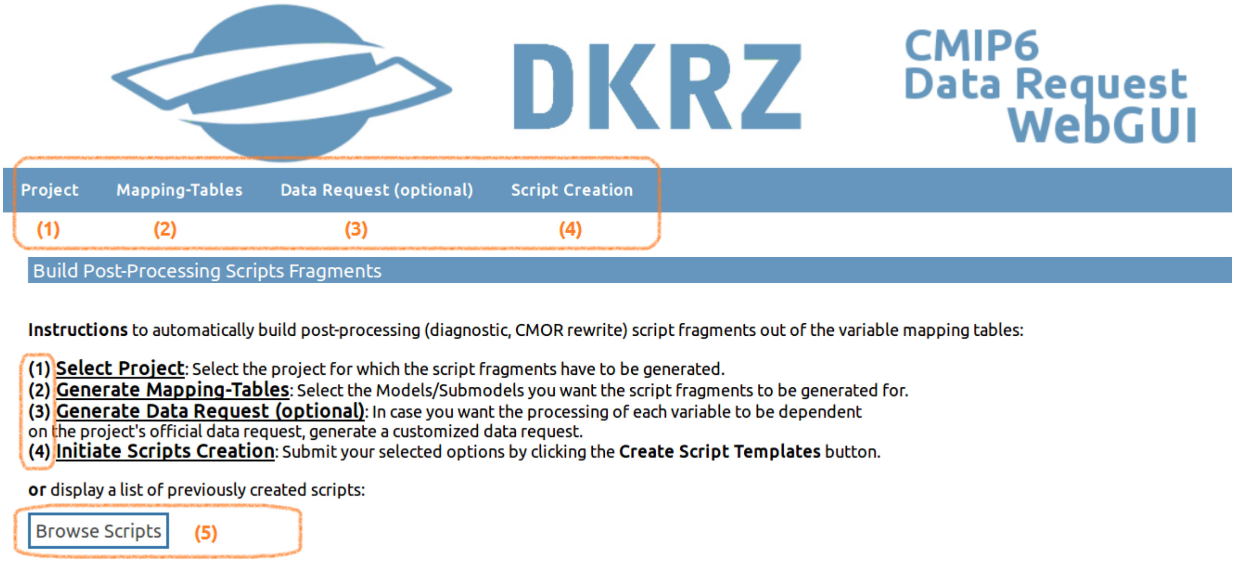
- (1) Here the model can be selected for which availability information is shown in the table below. A variable is considered available if it is available in any of the model's submodels and as unavailable if it is unavailable in all of the model's submodels. Else a questionmark is shown.
- (2) To add a variable to the Black-/Whitelist the respective checkbox has to be selected.
- (3) The buttons Select/Deselect all will select or deselect all displayed variables as black- or whitelisted.
- (4) The filters help to reduce the number of displayed variables.
- (5) To save the created Black-/Whitelist the button 'Save White-&Blacklist' has to be clicked.

Example: My simulation does not produce sub-3hourly output. However, the model in general is able to produce sub-3hourly output and therefore several variables of that frequencies are available and have entered mapping information. Thus, I want to exclude all variables from being post-processed that are requested with a frequency that is higher than 3 hours.

- I select 'AERhr' as Table filter and click on 'Reload'.
- I click on the button 'Select all' listed under Blacklist.
- I repeat both steps with the Table filters 'E1hr', 'E1hrClimMon', 'CFsubhr' and 'Esubhr'.
- Now all variables from MIPtables with a frequency that is higher than 3 hours are selected as blacklisted.
- Eventually I click on the 'Save White-&Blacklist' to save the changes.

4 Post Processing Web GUI - Automatic Script Fragment Creation

The information from the mapping and recipe tables can be used to automatically create post-processing script fragments for both, diagnostic and standard conform rewrite ('CMORising'). The mapping table contains the basic mapping information consisting of (Fig. 4.1):



Instructions to automatically build post-processing (diagnostic, CMOR rewrite) script fragments out of the variable mapping tables:

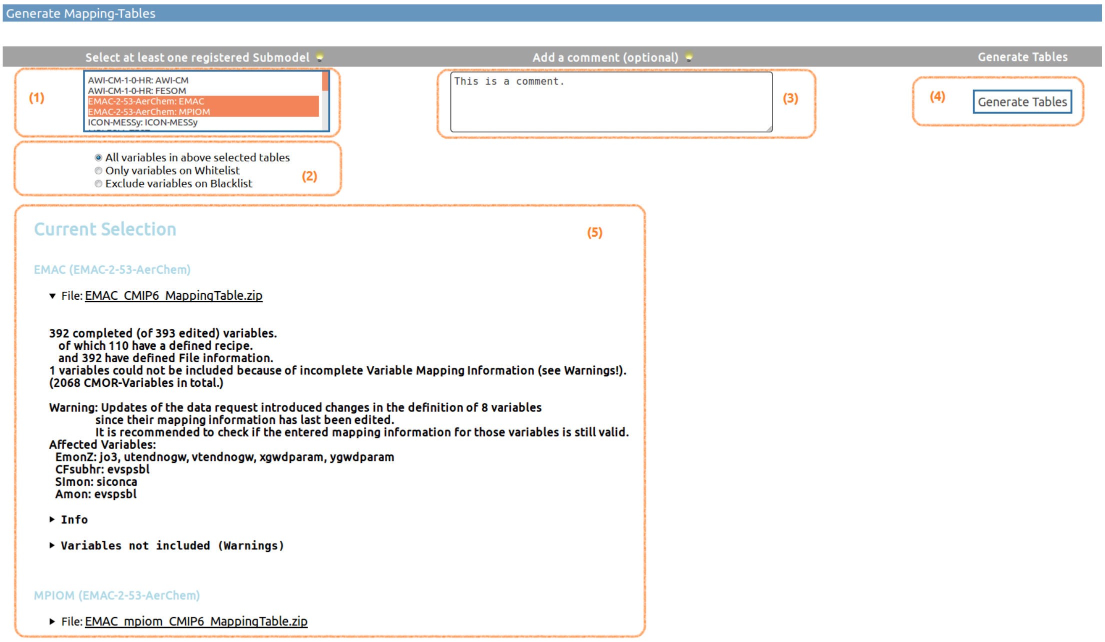
(1) Select Project: Select the project for which the script fragments have to be generated.
(2) Generate Mapping-Tables: Select the Models/Submodels you want the script fragments to be generated for.
(3) Generate Data Request (optional): In case you want the processing of each variable to be dependent on the project's official data request, generate a customized data request.
(4) Initiate Scripts Creation: Submit your selected options by clicking the **Create Script Templates** button.

or display a list of previously created scripts:

Figure 4.1: The PostProcessing WebGUI allows to automatically create diagnostic and CMOR rewrite script fragments including several configuration files. The user is guided through the process, which consists of 4 steps.

1. **Select Project**: Select the project for which the script fragments have to be generated.
2. **Generate Mapping-Tables**: Select the Models/Submodels you want the script fragments to be generated for. Apply a Black- or Whitelist if desired (see also Fig. 4.2).
3. **Generate Data Request** (optional): In case you want the processing of each variable to be dependent on the project's official data request, generate a customized data request (see also Fig. 4.3).
4. **Initiate Scripts Creation**: Submit your selected options by clicking the Create Script Templates button (see also Fig. 4.4).
5. **Browse Scripts**: Opens a table containing all previously created scripts (see also Fig. 4.5). The list can be filtered to identify the desired script which then can be downloaded. Alternatively all

details of this script's creation process can be accessed (see Fig. 4.6). The creator of the script has the option to delete this script.



AWI-CM-1-0-HR: AWI-CM
AWI-CM-1-0-HR: FESOM
EMAC-2-53-AerChem: EMAC
EMAC-2-53-AerChem: MPIOM
ICON-MESSy: ICON-MESSy

```
This is a comment.
```

● All variables in above selected tables
○ Only variables on Whitelist
○ Exclude variables on Blacklist

Current Selection

EMAC (EMAC-2-53-AerChem)

▼ File: EMAC_CMIP6_MappingTable.zip

```
392 completed (of 393 edited) variables.
  of which 110 have a defined recipe.
  and 392 have defined File information.
1 variables could not be included because of incomplete Variable Mapping Information (see Warnings!).
(2068 CMOR-Variables in total.)

Warning: Updates of the data request introduced changes in the definition of 8 variables
          since their mapping information has last been edited.
          It is recommended to check if the entered mapping information for those variables is still valid.
Affected Variables:
 EmonZ: jo3, utendnogw, vtendnogw, xgwdparam, ygwdparam
 CFsubhr: evspsbl
 SImon: siconca
 Amon: evspsbl
```

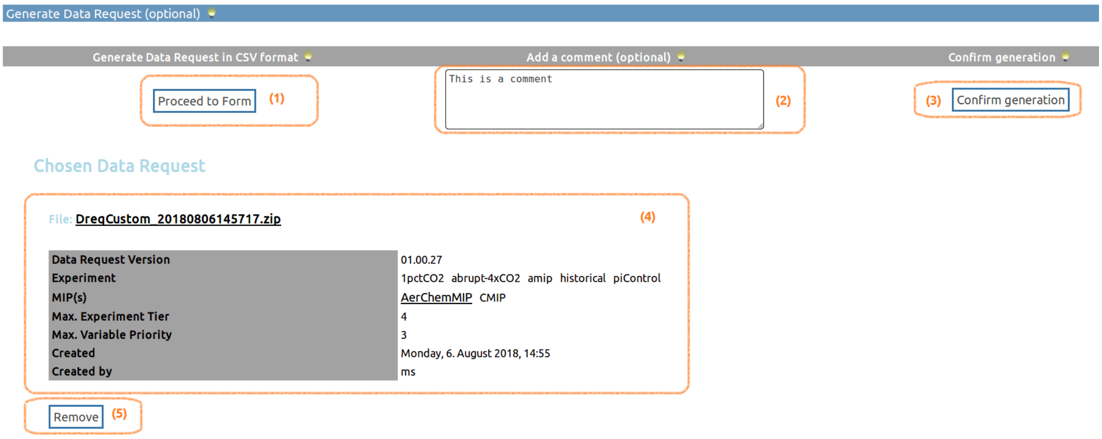
▸ Info

▸ Variables not included (Warnings)

MPIOM (EMAC-2-53-AerChem)

▸ File: EMAC_mpiom_CMIP6_MappingTable.zip

Figure 4.2: *Step 2 - Create mapping tables (as in Fig. 3.2). (1) Select the submodels. Hold down CTRL to select more than one model. (2) If you created a Black- or Whitelist in the Variable Mapping WebGUI (see section 3.5), you can apply it here to exclude the blacklisted variables from the post processing or exclusively include the whitelisted variables in the post processing. (3) Optionally enter a comment. (4) Click the Create button. (5) A list of the created tables will be displayed here, containing all necessary information as described in Sec. 3.1.*



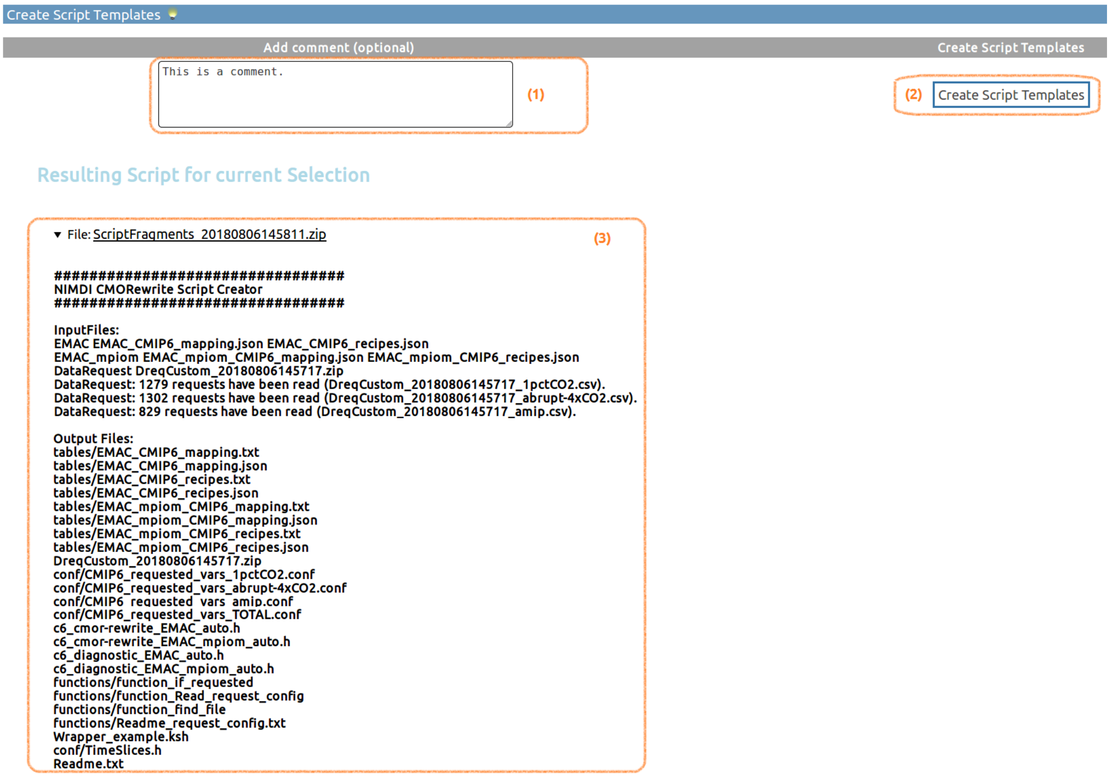
```
This is a comment
```

Chosen Data Request

File: DreqCustom_20180806145717.zip

Data Request Version	01.00.27
Experiment	1pctCO2 abrupt-4xCO2 amip historical piControl
MIP(s)	AerChemMIP CMIP
Max. Experiment Tier	4
Max. Variable Priority	3
Created	Monday, 6. August 2018, 14:55
Created by	ms

Figure 4.3: *Step 3 - Generate Data Request (optional): (1) A click on this button opens the Data Request WebGUI where one has to create a data request in csv format as specified in Sec. 2.1. The required attributes are already selected so it is sufficient to select experiments, mips, priority and tier. Once the request is created one has to press button (3) (after leaving an optional comment(2)). The result will be displayed (4). The selected data request can be overwritten by starting again at (1) or removed by clicking the Remove button (5).*

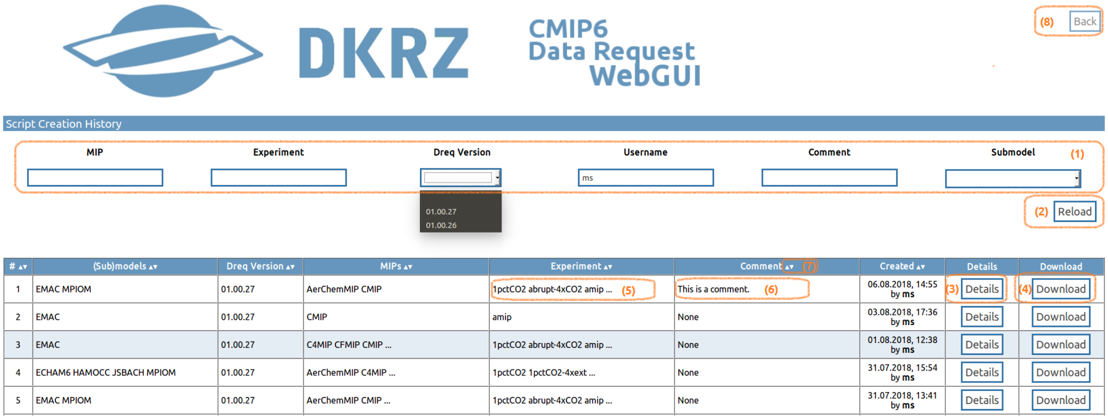


This is a comment.

▾ File: ScriptFragments_20180806145811.zip

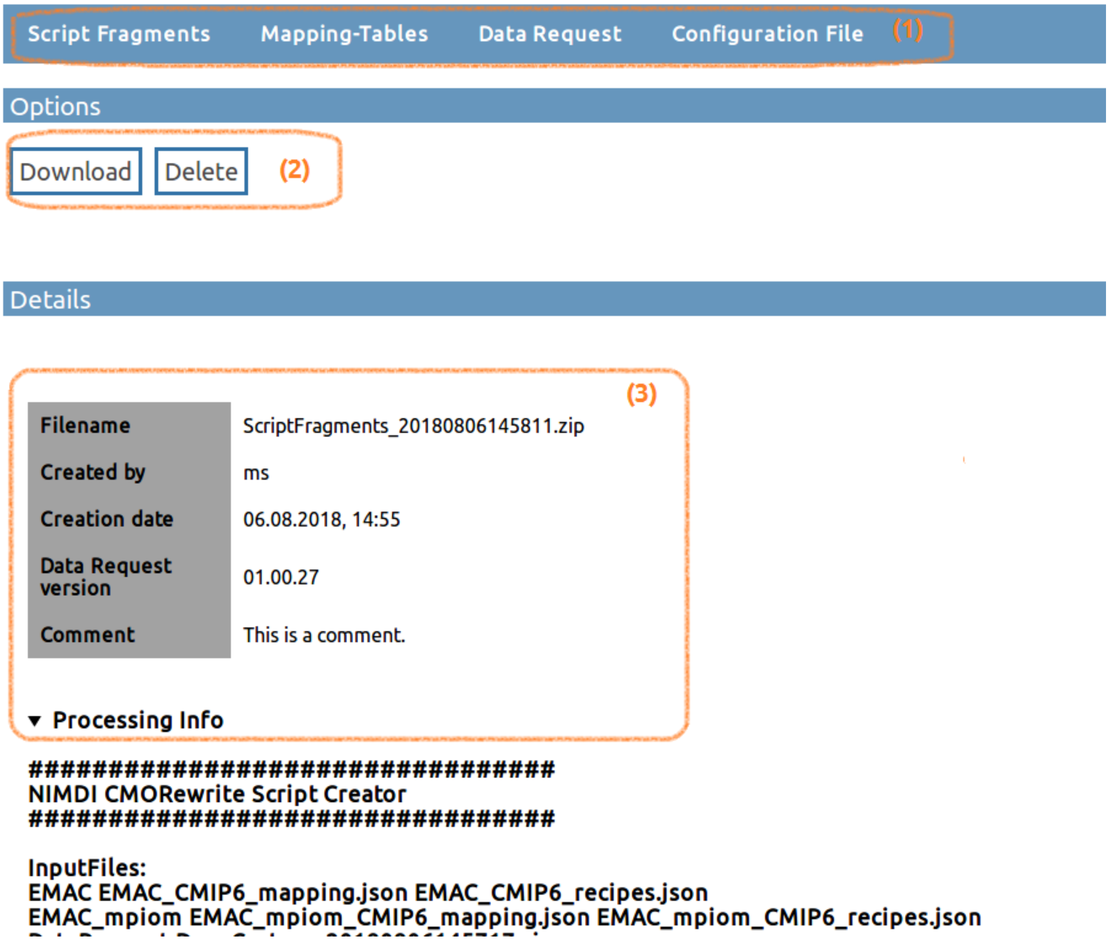
```
##################################
NIMDI CMORewrite Script Creator
##################################

InputFiles:
EMAC EMAC_CMIP6_mapping.json EMAC_CMIP6_recipes.json
EMAC_mpiom EMAC_mpiom_CMIP6_mapping.json EMAC_mpiom_CMIP6_recipes.json
DataRequest DreqCustom_20180806145717.zip
DataRequest: 1279 requests have been read (DreqCustom_20180806145717_1pctCO2.csv).
DataRequest: 1302 requests have been read (DreqCustom_20180806145717_abrupt-4xCO2.csv).
DataRequest: 829 requests have been read (DreqCustom_20180806145717_amip.csv).

Output Files:
tables/EMAC_CMIP6_mapping.txt
tables/EMAC_CMIP6_mapping.json
tables/EMAC_CMIP6_recipes.txt
tables/EMAC_CMIP6_recipes.json
tables/EMAC_mpiom_CMIP6_mapping.txt
tables/EMAC_mpiom_CMIP6_mapping.json
tables/EMAC_mpiom_CMIP6_recipes.txt
tables/EMAC_mpiom_CMIP6_recipes.json
DreqCustom_20180806145717.zip
conf/CMIP6_requested_vars_1pctCO2.conf
conf/CMIP6_requested_vars_abrupt-4xCO2.conf
conf/CMIP6_requested_vars_amip.conf
conf/CMIP6_requested_vars_TOTAL.conf
c6_cmor-rewrite_EMAC_auto.h
c6_cmor-rewrite_EMAC_mpiom_auto.h
c6_diagnostic_EMAC_auto.h
c6_diagnostic_EMAC_mpiom_auto.h
functions/function_if_requested
functions/function_Read_request_config
functions/function_find_file
functions/Readme_request_config.txt
Wrapper_example.ksh
conf/TimeSlices.h
Readme.txt
```

Figure 4.4: *Step 4 - Script Creation: (1) Optionally tag the script with a comment. (2) Then click the Create button. (3) After the successful creation the script can be downloaded by clicking on the filename. Additionaly some information is displayed.*



# ▴▾	(Sub)models ▴▾	Dreq Version ▴▾	MIPs ▴▾	Experiment ▴▾	Comment ▴▾	Created ▴▾	Details	Download
1	EMAC MPIOM	01.00.27	AerChemMIP CMIP	1pctCO2 abrupt-4xCO2 amip ...	This is a comment.	06.08.2018, 14:55 by **ms**	Details	Download
2	EMAC	01.00.27	CMIP	amip	None	03.08.2018, 17:36 by **ms**	Details	Download
3	EMAC	01.00.27	C4MIP CFMIP CMIP ...	1pctCO2 abrupt-4xCO2 amip ...	None	01.08.2018, 12:38 by **ms**	Details	Download
4	ECHAM6 HAMOCC JSBACH MPIOM	01.00.27	AerChemMIP C4MIP ...	1pctCO2 1pctCO2-4xext ...	None	31.07.2018, 15:54 by **ms**	Details	Download
5	EMAC MPIOM	01.00.27	AerChemMIP CMIP ...	1pctCO2 abrupt-4xCO2 amip ...	None	31.07.2018, 13:41 by **ms**	Details	Download

Figure 4.5: *Table view of the previously created scripts. (1) Several filters help to find the script of interest. Also scripts of other users can be listed. (2) The Reload button applies filters. (3) The Details button opens a website with all information about the script creation process. (4) The script can be downloaded. (5-6) If not all information is visible here, hover over the table entry and the full information will pop up. (7) The table can be sorted alphabetically in both directions.*



Filename	ScriptFragments_20180806145811.zip
Created by	ms
Creation date	06.08.2018, 14:55
Data Request version	01.00.27
Comment	This is a comment.

```
###################################
NIMDI CMORewrite Script Creator
###################################

InputFiles:
EMAC EMAC_CMIP6_mapping.json EMAC_CMIP6_recipes.json
EMAC_mpiom EMAC_mpiom_CMIP6_mapping.json EMAC_mpiom_CMIP6_recipes.json
```

Figure 4.6: *Detail page of a script. Several tabs with all information are available (1). The script can be downloaded and the creator has the option to delete the script (2). Information will be displayed below and can be further expanded (3).*

5 CMOR Global Attributes - CDO CMOR Config Tables

To use the 'cdo cmor' operator to rewrite data according to the CMIP standard, the values of several (global) attributes have to be provided by the user.

For that purpose Fabian Wachsmann devoloped a web gui that guides the user quite self explanatory through the process of collecting all the necessary attributes.

In the end the collection can be downloaded and may serve as input for the CMOR rewrite process using 'cdo cmor'.

More information on the required collection of global attributes and the use of 'cdo cmor' is given in the 'CDO CMOR User Guide' (Wachsmann (2017)).

Bibliography

Juckes, M.: Data Request Python API 01.00.00, Tech. rep., British Atmospheric Data Center, BADC, 2016.

Nadeau, D.: CMOR version 3.2, Tech. rep., Program for Climate Model Diagnosis and Intercomparison, PCMDI, 2016.

Wachsmann, F.: CDO CMOR User Guide, Tech. rep., Deutsches Klimarechenzentrum, DKRZ, Hamburg, Deutschland, 2017.