DICAD Workshop on CMIP6 Post-processing

Aim:

Learn how to produce CMIP6 conform data with the help of test model output for the piControl experiment and by using the **c6dreq WebGUI** and **cdo cmor**.

WebGUI: https://c6dreq.dkrz.de

E. 1) - 4):	Map model output variables to CMOR variables *(Tab: Variable Mapping)*
E. 5):	Configure cdo cmor *(Tab: Config Tables)*
E. 6) - 7):	Create Post-processing script fragments *(Tab: Post-Processing)*
E. 6 d) :	Customize data request *(Tab: Requested Variables / Volume Estimate)*

cdo cmor: https://code.mpimet.mpg.de/projects/cdo/wiki/CDO_CMOR_Operator

- Rewrite data in CMIP6 compliant format

WebGUI Exercises:

1) Get information about the test model output. If you work on mistral, search in the directory **/work/bm0021/workshopcmip6pp/testmodeloutput**
    a) Fill in the empty cells marked with blue color in the table below with the information you find in the meta data of the files
        i) use **ncdump -h** for netCDF files
        ii) use **cdo showcode** for GRIB files
2) Gain access to the Variable Mapping Table and get familiar with the table and edit view:
    a) Access https://c6dreq.dkrz.de
    b) Open the tab **Variable Mapping**
    c) Select the project **CMIP6 - Workshop**
    d) Select the submodel **Atmosphere** of the model assigned to you (**Group1-15**) and click on the button **Edit**
    e) Enter the login information assigned to you
    f) Use the filters to find the variable **tas - Amon**
    g) Click on the button **Edit** and fill the **Variable Mapping Information** form for this variable with the information from the table below
    h) Click on the button **Submit** to apply the changes
3) Map the remaining variables shown in the table below
    a) Enter the **Variable Mapping Information** from the completed table for the remaining variables in the same fashion as shown in **E. 2**, but:
        i) Choose the correct submodel for each variable (as in **E. 2c**)

        ii) Find out the correct CMOR Variable (MIP-table + CMOR name) that can be generated with the model output variable

4) Download a Mapping Table (MT)
    a) Open the tab **Variable Mapping** and select any submodel the model assigned to you (as in **E. 2d**)
    b) Click on the button **Generate**
    c) Note the information about the content of the mapping table (MT)
    d) Download the file via the “Download Mapping-Table Files” button
        i) On mistral, use 1. “curl $link --output $outputname”
        ii) 2. “unzip $outputname”
5) Create a configuration dataset for CMOR.
    a) Access https://c6dreq.dkrz.de and open the tab **Config Tables**
    b) Enter the specifications for your model/institution for a **piControl** simulation
    c) Download the resulting config table and open it.
        i) If you work on mistral, open a document “.cdocmorinfo” in the scripts directory and copy the content of the downloaded file
    d) You may have to add attributes and specifications
6) Create post processing script fragments for the entered Variable Mapping Information using a data request of all MIPs for piControl.
    a) Access https://c6dreq.dkrz.de and open the tab **Post-Processing**
    b) Enter the login information if necessary (as in **E. 2e**)
    c) Select the project **CMIP6 - Workshop** and then switch to the tab **Mapping-Tables** to generate the MTs for the submodels that you have edited before
    d) When you are in the **Data Request** tab:
        i) Click on the button **Proceed to Form**. A new tab opens up displaying the main WebGUI with some preconfigured settings. Here you have to create a customized DReq for the piControl experiment
        ii) Select **All BUT selection** under **Select MIPs**
        iii) Select **piControl** under **Select Experiments**
        iv) Do not change the **priority and tier** settings below
        v) Under **Generate DataRequest in .CSV format** you may add additional information to the customized DReq but not erase the preconfigured settings
        vi) Click on the **Create CSV** button and wait for the DReq to be generated. Then close the tab
        vii) You are now back to the still opened **Post-Processing** tab
    e) Click on **Confirm generation**, to load the DReq you just created
    f) Proceed to the **Script Creation** tab and click on **Create Script Templates**

- g) Use curl to download the scripts to your mistral work directory **/work/bk0988/workshopcmip6pp/<user>/** or any other suitable directory you want to perform the post processing in
    - i) Login to mistral and navigate to your WORK directory (**/work/bk0988/workshopcmip6pp/<user>/**) or any other suitable directory
    - ii) The curl and unzip commands to download and extract the created script fragment archive can be copied from the output shown in the GUI

7) After downloading and extracting the script fragments archive (**E. 6g**) use the included example wrapper script to perform the diagnostic and CMIP6 compliant rewrite of the data.
    - a) Copy the CMOR dataset configuration file from E. 5 to the same folder that you extracted the script fragments in. Rename it to **.cdocmorinfo**
    - b) Copy the custom diagnostic (NCL script and variable diagnostic script fragment) for the total ozone column (toz-AERmon) to your **incl_mod_atmosphere** folder (path see **Solution** sheets)
    - c) Apply all necessary changes to the dreq requested vars configuration file:
        - **i)** Find it in: **./conf/CMIP6-Workshop_requested_vars_piControl.conf**
        - ii) Add a custom timeslice for the climatological variable (co2massClim - Amon) for the years 1850-1851
            - (1) Format YYYYMMDDHH-YYYYMMDDHH
            - (2) See **Solution** sheets if you have difficulties
    - d) Make the necessary changes to the Wrapper_example.ksh script:
        - i) Adjust paths if necessary
        - ii) Depending on your PATH settings you might have to bind in the script fragments with **“. ./c6_fragment.h”** instead **“. c6_fragment.h”**
    - e) Load the NCL module on mistral if necessary to be able to perform the custom diagnostic: **module load ncl**
    - f) Run Wrapper_example.ksh and see the logs afterwards (./errors/)
    - g) Make changes to the **requested vars configuration in conf/** and/or to your **.cdocmorinfo** and run the script again. For more verbose output you can add “**-v** ”to the **if_requested** calls (in the script fragments) or the **Read_request_config** calls (in the Wrapper example script) or omit the “**-s**” in the calls.

Sub-model	**Frequency**	**Aggregation**	**CMOR name**	**Name/Code in Model**	**Diagnostic (formula)**	**Unique filename pattern of the input files**	**Model output units**
Ocean	Monthly	Mean	thetao			ocean_depths_mean_mon*	K
Ocean			deptho			ocean_fx*	m
Ocean	Monthly	Mean	siv	sivsic, sicv	sivsic/sicv		"m s-1"
	Monthly	Mean	tas			*nolev_mean_mon*	K
Atmos	Monthly	Mean	cli			*hyb_mean_mon*	
Atmos	Monthly	Mean	pr	prls,prc	prls+prc		“kg m-2 s-1”
Atmos	Monthly	Mean	toz	-	CUSTOM	-	-
Atmos	6hourly		va			*plev_pt_6hr*	“m s-1”
Atmos	Climatology	Climatology	co2massClim			*nolev_co2m*	kg
Land	Monthly	Mean	landCoverFrac	*see all x tiles in input file*	*All tiles separated with “;”*	*tiles_mean_mon*	%
Land	Monthly	Mean	ts	67			K
Land	Monthly	Mean	c3PftFrac	var12, C3C4_crop_mask	vertsum(sellevidxrange(var12,1,7))+sellevel(var12,9)+sellevidx(var12,11)*(1-C3C4_crop_mask)		1