

Geosci. Model Dev. Discuss., author comment AC2
<https://doi.org/10.5194/gmd-2022-205-AC2>, 2022
© Author(s) 2022. This work is distributed under
the Creative Commons Attribution 4.0 License.

Reply on RC2

Manuel Schlund et al.

Author comment on "Evaluation of native Earth system model output with ESMValTool v2.6.0" by Manuel Schlund et al., Geosci. Model Dev. Discuss.,
<https://doi.org/10.5194/gmd-2022-205-AC2>, 2022

Reviewer comments are given in **bold**, our answers in regular font.

The submitted article by Schlund et al provides details and examples on how ESMValTool have been further developed in order to read raw output from a number of Earth System Models. The structure provided also give a framework on how to extemnd these capabilities to other models. Even though the article is very technical adding and describing these capabilities are an important step in order to use the tool efficiently, so I think it may be accepted in GMD.

We thank the reviewer for the helpful and constructive comments. We have now revised our manuscript in light of these and the other reviewer's comments we have received. A pointwise reply is given below. Line numbers in the answers correspond to line numbers in the revised manuscript.

The article are however a bit unclear on the first step of the process or whether there are some initial steps prior to the process not covered by the article, e.g time averaging, preprocessing, e.g. for EC-Earth. I think it might be useful if the authors provide a even more specific overview of the steps needed, i.e. in a model table. I presume 4-5 cells /categories would be enough for the purpose. E.g: Preprocessing; Create facets; Run preprocessors; ...

We agree that the manuscript needs to be clearer on the specific steps necessary to read and process native model output. For this reason, we clarified the definition of "native" in the context of our paper: we use the term "native" to refer to "operational" model output, i.e., model output that is the result of the standard workflow used by the corresponding modeling centers to run their model:

L.11–12: "*[...] native climate model output, i.e., operational output produced by running the climate model through the standard workflow of the corresponding modeling institute.*"

L.65–67: "*In the context of this paper, the term "native" refers to operational output produced by running the climate model through the standard workflow of the corresponding modeling institute including potential postprocessing steps commonly used in practice.*"

For some models, this workflow may include postprocessing steps like applying the software package `ece2cmor3` for EC-Earth. This has been clarified in the revised manuscript:

L.188–192: " *As part of the standard workflow used to run the model, this data is then postprocessed to CMOR- and CF-compliant netCDF format. For this, the Python package `ece2cmor3` (van den Oord, 2017, <https://github.com/EC-Earth/ece2cmor3>, last access: 1 November 2022) is used, which contains modules to format output from each of the model components. Thus, a CMOR-like reformatting of the native (i.e., operational) EC-Earth3 output within ESMValTool during runtime is not necessary. "*

We also provided links to the documentation of ESMValTool that describes how to use the new feature, and how to add support for new climate models. Moreover, we added references to a publicly available ESMValTool recipe that showcases the usage of these new features:

L.151–153: " *Detailed user instructions on this can be found in ESMValTool's documentation (<https://docs.esmvaltool.org/en/latest/input.html#datasets-in-native-format>, last access 1 November 2022). The documentation provides links with further details on all the available models and instructions on how to add support for new climate models. "*

L.333–336: " *The following paragraphs illustrate five example plots (one for each currently supported climate model) created with these new diagnostics. A recipe to reproduce these figures is publicly available on Zenodo (Schlund, 2022). This recipe showcases the usage of the monitoring diagnostics on native model output and serves as a convenient starting point for users who want to process native model output with ESMValTool. "*

L.443–448: " *An example recipe to get started with processing native model output with ESMValTool is publicly available on Zenodo at <https://doi.org/10.5281/zenodo.7254312> (Schlund, 2022). This recipe reproduces Figures 2–7 of this paper. Detailed user instructions on the CMOR-like reformatting of native model output can be found in ESMValTool's documentation at <https://docs.esmvaltool.org/en/latest/input.html#datasets-in-native-format> (last access 1 November 2022). The documentation is recommended as a starting point for new users and provides links with further details on all currently supported models and instructions on how to add support for new climate models. "*

We also updated Figure 1 to show the new features (and compare them to the "old" approach) in more detail.

On a related note, while it is good that the authors show the capabilities by adding examples for all the models I also think it is a bit unclear whether this means that the process need to be different for all the models all the way until the plotting scripts. Or are there a point e.g. after the `config_developer` where the formats of the different model are equivalent and that later procedures/scripts can be used for all models producing the same type of plots as for the model specific example.

This is in particular important for section 4 and 5. While the models are clearly separated in section 2 this is not the case for section 4 and 5. Given that any plots may have been produced with any of the models this is fine. If not I think there should be a more clear separation between the models.

After the initial preprocessing, all model output is fully CMOR-compliant. Thus, native models do not need any special treatment for further preprocessing steps such as horizontal or vertical regridding, time averaging, area selection or masking, or the

subsequent diagnostic scripts. All diagnostics (except those that require very specific input) can in principle be applied to suitable arbitrary input, as is the case for the monitoring diagnostics showed in Section 4. To make this clearer in the manuscript, we updated Figure 1 (which now shows that all native input data is fully CMOR-compliant after the preprocessing) and also expanded the text:

Caption of Figure 1: " *Here, we describe additions that allow reading and processing native model output (bottom left blue ellipse) with ESMValTool through a CMOR-like reformatting (yellow round rectangles) within the ESMValCore preprocessing pipeline. As a result, the data is fully CMOR-compliant after this initial preprocessing step and can be processed by the diagnostic scripts (orange box) just like any other input data set. The diagnostic scripts do not need to treat native model output in any special way.* "

L.380–382: " *Since preprocessed output by ESMValTool is fully CMOR-compliant for all input data sets (see Figure 1), no specific changes to these diagnostics scripts are required when dealing with native model output.* "

Minor issues

Line 62. Typo: "rawoutput" -> raw output

This part has been removed from the manuscript.

Line 90 I presume at least some of the intermediate products be stored. Preprocessing is often heavy for long timeseries so you want to reduce the runtime. Also mentioned at line 413, but can be included already here.

We added the following sentence to Line 90:

L.95–96: " *If desired by the user, these files can also be saved to disk, which allows ESMValTool to be used as a CMORization tool.* "

line 238 Oceanic model -> ocean model

Thank you, fixed.

line 329. It is often very useful to compare the experiment with a control version. Is this easily done e.g using earlier stored fields?

Yes, this is possible! We expanded the first sentence in the corresponding paragraph to clarify this:

L.337–338: " *For a direct comparison with one or multiple reference data sets (e.g., observations, reanalyses, output from other model versions, etc.), Figure 3 shows [...].* "

Page 16, figure 6. May be useful also to show if you can provide simple statistics, e.g. global averages ? bias rmse?

Yes! We added simple statistics (global averages, bias, RMSE, and R^2) to Figures 5 and 6. This feature is already available in the latest version (v2.7.0) of ESMValTool (see <https://github.com/ESMValGroup/ESMValTool/pull/2790>).

line 403: Any plans for developing more advanced regridding?

Currently, there are no plans to develop regridding algorithms within the ESMValTool repository. However, as described in the manuscript, ESMValTool now allows the usage of

external regridding packages (in addition to native Iris schemes) with arbitrary options. Thus, we can take advantage of other software packages that specialize on the development of regridding algorithms. An example for this is the actively developed iris-esmf-regrid package (<https://github.com/SciTools-incubator/iris-esmf-regrid>), which allows the usage of ESMF regridding algorithms within Iris, and thus within ESMValTool.