

Geosci. Model Dev. Discuss., referee comment RC1  
<https://doi.org/10.5194/gmd-2022-38-RC1>, 2022  
© Author(s) 2022. This work is distributed under  
the Creative Commons Attribution 4.0 License.

## **RC Comment on gmd-2022-38**

Valeriu Predoi (Referee)

---

Referee comment on "The E3SM Diagnostics Package (E3SM Diags v2.7): a Python-based diagnostics package for Earth system model evaluation" by Chengzhu Zhang et al., Geosci. Model Dev. Discuss., <https://doi.org/10.5194/gmd-2022-38-RC1>, 2022

---

Review also available at [https://github.com/valeripredoi/review\\_gmd-2022-38/blob/master/review\\_gmd-2022-38.md](https://github.com/valeripredoi/review_gmd-2022-38/blob/master/review_gmd-2022-38.md)

**Paper title: The E3SM Diagnostics Package (E3SM Diags v2.6): A Python-based Diagnostics Package for Earth System Models Evaluation**

**Preprint:**

**<https://gmd.copernicus.org/preprints/gmd-2022-38/gmd-2022-38.pdf>**

### **Review data**

- Review by: Valeriu Predoi
- Date: 10 May 2022

### **General remarks**

The paper reads well, and contains about the right balance between technical vs scientific aspects, however, I have a couple suggestions:

- is this a technical overview work or a more scientifically-oriented description of the diags/metrics in E3SM\_Diags? It'd be worth stating this clearly at the very top;
- the technical aspect opens the narrative, with a general description of the package's workflow, only to be interrupted by a very detailed scientific description of a number of standard metrics, and, after these metrics have been explored, the narrative comes back to technical aspects related to installation, I/O etc. It would be good to have the technical

- aspects bunched together, to allow the reader to follow through, and understand how the thing works; same for the metrics/diagnostics - probably better understood after the reader has insight into the compute workflow aspects of E3SM\_Diags;
- versions matter: what is the purpose of describing E3SM\_Diags v2 (actually, a very specific v2.6) - are there any major enhancements/refactorings/performance improvements from v1? Is there a paper describing v1, and if so, please link it? If it was me, I'd really drop the very specific v2.6 and instead refer to the tool with a more general v2 only;
- probably avoid including very specific script filenames (e.g. ``postprocessing_E3SM_data_for_TC_analysis.sh``), these are documented in the code documentation already; the reader should get a bird's eye of the workflow without getting them into the nitty-gritty details;
- *some* information about the code license is needed (GPL, Apache etc)

## General suggestions

- title: E3SM Diags v2.6 is mentioned, but 2.6.1 on conda-forge  
[https://anaconda.org/conda-forge/e3sm\\_diags](https://anaconda.org/conda-forge/e3sm_diags)  
 suggestion: either change to "2.6+" or don't specify the version in the title, or specify why 2.6 is a major release even though the number doesn't suggest that way;
- Abstract: please provide a reference/link for AMWG diagnostics package;
- Introduction: line 27: "Most tools listed here are focused on the atmosphere" - not entirely true, ESMValTool has a solid component for ocean model data evaluation; suggest changing to "Most tools listed here have started with a main focus on the atmosphere, but evolved to analyze other ESM realms too"
- Introduction: line 32 - please provide a reference (methods paper, documentation and/or software repository) for AMWG
- Introduction: line 57 - "...project takes a distinct but related approach" - contradiction in terms
- Introduction: line 63 "Distribution of these Python packages is now mostly accomplished through Conda" - please provide a reference to the Anaconda documentation/main page; also note that "conda" is the executable and the package manager tool, but Anaconda is the organization and the actual package manager, so it's probably best to use Anaconda in this context, maybe even "Anaconda/Miniconda"
- Introduction: line 72 - "This paper is a comprehensive description of E3SM Diags (as of version 2.6)" - anything special about v2.6? Please briefly describe why the choice of version;
- Table 1: I am assuming the references will be available as active links in the final version of the paper?  
 (idem): ESMValTool: AR5 and AR6, please, also - ESMValTool is a PyPi package too  
<https://pypi.org/project/ESMValTool/>
- Optional: ESMValTool handles OBS datasets too, that are not necessarily CMIP-like file formats, via a process of CMOR-standardization (CMORization), maybe with mentioning?
- Tech overview: line 76 - maybe provide a link to the installation section of your documentation?
- Tech overview: line 83: multi-threading, really, up to you if you want to change from

multiprocessing to multi-threading

- Tech overview: general: it would be nice to provide links to various main sections of the package documentation

  - e.g. when talking about installation, configuration, running instructions

- S3.1: line 128: how is the reference grid chosen?

- S3.2: line 157: a very detailed explanation on how the (calibration) run works, but a rather frugal explanation of how CMIP data is retrieved from an ESGF node: "The built-in derived variable module takes in CMIP variables and handles variable name and unit conversions" - it'd be good to have this explained a bit better, and in a user-friendly way e.g. data specifications include: var name, mip, experiment name, frequency etc - with these parameters gathered from user input, the code goes and grabs data from the ESGF via standard-path DRS structure conventions etc

- S4.1.1: line 340 - are the two yaml environment files (stable and dev) distributed within the conda package? If so, this is bad practice, since it may lead to confusion on the part of the user; the conda package should contain files that are only from the stable released version, and if the user wants to use the development distribution, they should have the files associated with it when downloading (cloning) the gitball;

- S4.1.3: line 375: `e3sm\_diags` executable - mention the word executable;

- S4.2: Zippy - is that a dependency of E3SM\_Diags? Is it included with it as a sub-module? If it's a dependency, please include a link to its GH repo, or wherever the codebase lives, and to its documentation;

- same as above for S4.3: IICE

- Appendix A: line 460 - please don't use the copy-past call into a Python script - just say "execute/run the following Python script" (and it would be good if you could use e.g. Markdown colouring to simulate Python code colors); ideally try and make that an iPython Notebook - much more modern these days :)