Interactive comment on "Towards improved and more routine Earth system model evaluation in CMIP" by Veronika Eyring et al.

Reply to Anonymous Referee #1

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

The main changes compared to the previous version are:

- We have clarified that this is a viewpoint paper (see comments by Reviewer 3)
- We made the distinction between what is planned for CMIP6 and what is a long-term vision clearer in the text
- We have expanded the paper with additional information on the tools that will be applied to CMIP6 model simulations as soon as the output is submitted to the ESGF. We have also included two more example figures from these tools. However, we note that this paper is not a detailed documentation of specific evaluation tools that are described elsewhere in the literature. To make it easier for the reader, we have included an additional table with references and links to these tools.

The authors advocate the very laudable goal of developing a set of diagnostic tools that could be applied during future phases of the Coupled Model Intercomparison Project (CMIP). Unfortunately, however, the authors remain rather vague regarding some elementary design features of the proposed framework in which these tools should be implemented.

For example, the question whether the set of tools should be easily portable to users' platforms or whether it will be more or less tied to the ESGF framework is only answered somewhat implicitly on page 9, especially since "open source" clearly does not imply portability.

The tools shall run both in the ESGF structure and locally. The use of "open source" was inadequate in this context; we have revised the text to clearly state that users can download the source codes of the tools and run them on their local systems. The tools are designed to be portable and have been tested on a number of systems already, which is explicitly stated now in the paper.

Also, it would be interesting to know whether the proposed code can also be used to interpolate and/or extract data or whether it will only provide ready made plots.

The evaluation tools are not targeted towards flexibility, rather they produce as we say standard evaluation plots. Some options will still be left to the user choice, concerning for example regridding methods, confidence level in statistical tests, etc. The focus of the evaluation tools is however currently not on simply extracting data, although they could in principle be used to do this and we have added this to the text.

I think it would be good to answer these questions already at an early stage in the design of the tool.

We have expanded the text in Sect. 2.2 with details on the evaluation tools and provided some additional examples and figures. However, we note again that we are not describing the design of specific tools in this paper, rather the framework for a more routine model

evaluation in CMIP. Specifics of the design of the tools are detailed in the corresponding documentations of the tools. A few examples of available evaluation tools are given throughout the manuscript with the corresponding references that we refer to and available evaluation tools for CMIP6 are now also listed in a new Table 2.

I also think an overview of the key requirements/specifications (or maybe it is better to say "goals" instead of "requirements" since after all the authors are planning to provide another valuable and voluntary service to the community) in a list that might help the reader to understand what might be coming and also to serve as a guide for the further development of the framework would be helpful.

We have framed the discussion in terms of goals to deploy initially two available software package (also including other tools) that we plan to apply to CMIP6, to then build an infrastructure capable of supporting additional codes in the future. More details, in terms of achievable goals and actual specifications will depend on what the ESGF consortium can actually deploy in 2017 and beyond – which to the most part is still unknown.

Specific comments and questions:

1) could the proposed code also be used to interpolate and extract data or will it only provide ready made plots? Interpolating and extracting seems to be one capability that is needed anyway in designing the evaluation tool, and that by itself would be of great benefit to the user community. Would it make sense to construct the tool in order to eventually do data extraction and/or interpolation on the server side and plotting on the user side?

The priority at the moment is to provide the specific diagnostics already identified by the scientific community. Additional functionality, such as generic sub setting and interpolation might be desirable, but this is up to the teams that develop the tools. The ESMValTool, for example, allows for a separation of the data processing and graphic part. The results of each diagnostic are stored in a NetCDF file before drawing the plots. The users can therefore ignore the graphical part of the tool and make use of the NetCDF output and plot the results offline with a different software. Since this is quite a specific comment that is treated differently in different tools, we have not included these details in the text, but rather only say that in principle some of the tools can also be used to extract data, noting that this is not their main purpose at this stage.

2) in case a standard set of plots will be provided during CMIP, will the plots be archived permanently, will they be citable, and should they be copied and included in publications?

The plots produced by the standardize evaluation process outlined here will eventually be archived and become part of model documentation. In the meantime they can also be included in publications on model evaluation: since the tools that produce them are open source, the resulting plots are also effectively freely available. However, we would expect users to cite both software versions and technical papers produced by the tool developers to provide the formal provenance for the plots. This has been added to the text.

3) p. 3, line 26f: I find that there has been absolutely no lack of studies that have pointed out model deficits, and modelers do know where the problems are. One thing that has been missing is sufficient investments in climate model developers. My impression is that having a standard suite of evaluation tools will not do much to actually ameliorate the problem described in line 26f.

We highlight that model development is important. The evaluation activity proposed here can however provide guidance for model development which is mentioned. It can help focus a group of developers which span institutions for instance.

4) p. 11, line 7: could you either give a reference and/or describe what these "well defined standard interfaces" might look like? (Please see also my "additional comment" below).

This paragraph has been removed, but the fundamental point is that tools will need to declare inputs and outputs, via an interface – and an appropriate one doesn't yet exist, but is a named task in one of the many projects aiming at delivering this functionality.

5) would it be better to use just a single language (python) that can in principle could replace the potpourri of all the other languages in the ESMValTool? Please discuss.

What languages are used is a decision of the corresponding development teams of the tools. While there are certainly advantages of having tools all written in python or another language, experience in the project EMBRACE has shown that new diagnostics can be more readily added if different languages are used. Several existing diagnostic packages (such as the CVDP developed at NCAR and then included as part of the ESMValTool) are written in languages other than python. There are no mechanisms to fund the rewriting and supporting of the existing tools.

6) who should users contact if they are interested in contributing to the tool? Who will decide which diagnostics are to be included and which diagnostics are not to be included in the framework? Will there be "standard" and "user supplied" diagnostics? I understand that some strategy still needs to be developed, but it would also be nice to know what the possible outcomes of these developments might be.

The users should contact the respective PIs of the evaluation tools. What diagnostics are included in the tools is decided by the tool development teams. As mentioned in the paper, they correspond to " well-established parts of ESM evaluation that have demonstrated their value in the peer-reviewed literature". The distinction between standard and user-supplied is misleading in this context, since user-supplied diagnostic can also become part of a standard set after they have been included and tested in the respective tools.

7) p. 10, line 2: "users can however make substantial use of the tools by downloading the open source versions and by running them locally on their machines" -> this seems to me a major design requirement and it should be mentioned already at an early on in the manuscript. Is the code meant to be portable? Or will it tied to the ESGF servers?

As mentioned above, the tools are designed to be portable, i.e. not tied to ESGF. We changed the text and mention the portability issue early on in Sect. 2.2.

8) is it thought that individual users will eventually be able to adapt the code that they run on the ESGF machines? In other words, will users eventually operate their own version of the code on the servers in which they can adapt not only namelist settings, but also add diagnostics? Will it be possible to use additional data that might not be stored on the servers in these diagnostics? If yes, how could this be achieved? Would it make sense to do data extraction and/or interpolation on the server side and plotting on the user side as suggested in point 1 above?

Servers side extraction is already possible via OPENDAP, but as yet we have no specification of (or plans for) how to define a required interpolation operation on the server from a client. At the moment the easiest route is to bundle interpolation into the diagnostic itself – and yes, we expect it would be possible for users running their own versions of the code to add inputs and make configuration changes. Ideally where such changes extend beyond configuration, they would contribute their changes and extensions back to the community. The details of how that should be done will be tool dependent (and is therefore in the tool documentation). We think details of this sort go beyond what should appear in the paper itself.

9) p. 7, line 25: how are the groups supposed to use the tool during model development if it is run on the ESGF nodes? Will there be a stand-alone and an online version or will it just be one tool that can do both jobs? And also, how are you planning to deal with the dual requirements that the code should facilitate automatic processing while at the same time be user friendly, highly portable, and easily adaptable and expandable?

As now said in the manuscript, these tools should operate in both modes: integrated into the ESGF structure, and locally during model development. Concerning the dual requirement of facilitating automatic processing while being user-friendly, portable, adaptable and expandable, we do not see a contradiction here. The automatic processing of model data is based on the standardization of model output (CF/CMOR) which has been already successfully achieved within CMIP5 and other MIPs, such as ACCMIP and AEROCOM. Standardization of observations is a more complicated issue, but progress has been made within obs4mip and ana4mips, and some of the existing tools such as the ESMValTool or the CIS Tools (http://www.cistools.net/) proposed methods for dealing with the large variety of formats in the observational community. The portability aspect has been clarified in the text.

10) p. 2, line 26: I can not find any useful documentation of CMIP5 models under the web site given for ES-DOC. This reminds me of another project that has received funding for collecting meta-information on CMIP5 models, but that provided a poorly designed questionnaire and website to the model developers and as far as I can see has ultimately failed to be useful to users as well.

The http://es-doc.org provides access to model documentation through search and compare functions in the documentation folder. The information gathered through the CMIP5 questionnaire has been displayed with easy menu tables. Information can be displayed per model or compared between models. The website is already given in our manuscript that points the reader to further information on ES-DOC.

11) on p.8 in line 26 you are suggesting that the software will be able to acquire cache data from other servers. Will this cache data be kept for when one of the other servers is down? My experience has been that due to the distributed nature of ESGF it is sometimes very difficult to have access to all the data sets one wants to analyze at a given time. Would it be useful to cache processed (interpolated/extracted) data on the user side once the tool is opened up to users?

That's why we're suggesting that the data will be replicated to the supernodes, so the supernode tooling will have local copies of the data. Users running their own local copies will of course be able to manually cache their data as necessary.

12) p. 9 line 7f: "these supernodes have the necessary storage and computing resources". In line 17 it says: "requires the extension of current hardware" and in line 30, it says that the computing resources might not suffice for users to base their own analysis on this tool.

Thanks for spotting this. We rephrased these sentences for consistency. Our point here is that the existing supernodes are already providing large storage and computing resources, but may need to be extended to handle the larger CMIP6 data amount compared to CMIP5.

13) p. 10 line 7: "whereby new diagnostics developed by individual scientists can quickly and routinely" -> how will porting diagnostic tools be handled? Especially, what do the scientists have to do in order to port their diagnostic tools to the framework or to have them considered?

There will be two routes: (1) adding diagnostics to existing tools, and (2) adding new tools. With respect to (1): The ESMValTool for example offers a development environment that is open for the community to join. Both PMP and ESMValTool are also on github and are thus open to anyone contributing and sending a pull request. This has been clarified by extending the description of both tools. With respect to (2), the mechanism for new codes to be added into the system is another area where work is underway, but details are not yet available. It's likely this will be rudimentary at best for CMIP6.

14) p. 11 line 15: to me it seems important that the data version should be somehow documented. Yet, this is not mentioned here. As far as I can see, with CMIP5, finding out the version of a data set can only be achieved via sending a query with a checksum to an ESGF server. Maybe the users' interest in version numbers for the data sets has been underestimated? Also, will old versions of the data be stored so that one can reproduce results later without having to keep a local copy of the data? With CMIP5 this is not clear to me.

There are several questions here which go beyond the scope of our paper. However, it will never be practical to keep all the CMIP data, the volume would be prohibitive. That said, some of the supernodes are committed to keeping as much as is possible, and all are committed to keeping metadata about what data did exist, and how it differed. There will be a new errata system in CMIP6 to help this. The existing DOI system has been enhanced, and new work is in place to address workflow provenance to identify which data was used (even if it is not longer available) – however much of this is still under development. Please follow the CMIP6 Special Issue for further details that the WIP will provide, and more generally, the CMIP Panel website for up-to-date information on CMIP6.

15) p. 11, line 3: I think for all practical purposes, this would require either a new electronic data base format for citing the data or else summary doi's, which I don't think would work. I don't think that having 500 references to data sets each with its own doi would make much sense in something that might be printed on a printer, even if it would certainly be possible to automatically generate the corresponding list.

There has been considerable discussion of this in the community (e.g. http://home.badc.rl.ac.uk/lawrence/blog/2013/08/23/gavin%27s_proposal). The bottom line is that you're right, it won't be 500 references per paper.

16) p. 12, line 2f: "Model evaluations must take into account the details of any model tuning" -> how? Are you planning to archive output from all the untuned model versions? I don't understand what this sentence and also the following sentences might mean in practical terms. I also don't quite understand why this might be useful at all.

We simply mean that clear and concise information about what tuning went into setting up the model needs to be made available, so evaluations can be cognizant of any consequences. ES-DOC will be collecting some information to aid this process.

I think that it might be nice to have output for the same model tuned in different ways (maybe as "physics options" p1, p2,..."). But the sentences in the manuscript sounds like you are advocating the archiving of data for untuned models? If yes, please explain what you expect to learn from this. I do not think that archiving the data of untuned models within the framework of a model intercomparison projects makes much sense. Untuned models do not generally simulate a realistic radiation balance at the top of the atmosphere, and I think it is save to discard them in for the sake of model intercomparisons, especially since you are talking about comparisons with observations.

Sentence revised for clarity. See above.

17) p. 14, line 15: "requires ongoing maintenance" -> very good point. How can this be achieved?

A close collaboration between the ESGF system manager and tool developers is essential. To that end, the WIP has established the CDNOT: the climate data node operations team to directly serve such requirements for WCRP.

18) Fig 1: given this centralized approach, how can sufficient reliability and redundancy be achieved? Just recently, the ESGF nodes have been completely unavailable for several months.

The unavailability of ESGF was due to a hacker attack and not to infrastructural or technical issues. As pointed out in a recent letter to the community, “while ESGF cannot guarantee immunity from future dedicated hackers, the project has taken several steps to minimize the likelihood of a future security incident, and to recover much faster in case such an event should happen. At the same time, we have upgraded our infrastructure in many respects, to make it faster, more resilient, and more reliable.”
https://verc.enes.org/community/announcements/ESGFOperationalLettertotheCommunity.pdf

19) Notwithstanding my criticisms above, I do think that the ESGF people have on the whole done a great job and that their efforts have been extremely useful to the community. I also very much appreciate the initiative for the standard model evaluation tool, and I am confident that it will ultimately be very useful as well. I was also glad to have find other sources of the CMIP5 data while the ESGF servers were down.

Thanks!

Minor Points:

1) p. 12, line 27f: for an "emerging constraint", one needs a relationship between climate sensitivity and a model diagnostic that varies between models but can be constrained by observations. I think the formulation in the manuscript is not entirely clear.

Sentence revised for clarity.

2) p. 12., line 31: "might" -> could be considered more likely to

Changed as suggested.

3) p. 12, line 32: "A question raised ... " -> I don't understand what is meant here. Please re-formulate.

Sentence revised for clarity.

4) p. 12, line 33: "Moreover, ..." -> I think that this is a very good point.

Noted.

5) p. 13, line 3: "studies need not lead to contradictory results" -> I don't understand this sentence. Please re-formulate.

Sentence revised for clarity.

6) p. 13, line 21: in my opinion, one key question might be how easily adaptable this platform is by individual users

We are not sure what the reviewer means with "adaptation". The evaluation tools shall be designed in such a way that the user can easily extend them with additional diagnostic and metrics. In terms of customization of the available software, that will depend on the individual tools. As mentioned above, however the goal is not to provide fully-flexible analysis tools (such as CDO, NCO, CISTools etc.,) which already exist, but rather to share diagnostic codes reproducing well-established analyses for climate model evaluation.

7) p. 24, line 24: could you please specify what you mean by "revolutionary"?

The evaluation task will be transferred from the local modelling groups to the whole community, which will share agreed-upon methodologies and approaches and apply well-established and well-tested analysis, diagnostics and metrics to evaluate models. The observational data used for the evaluation will also be shared and, thanks for the obs4mips and ana4mips efforts, well documented. In our opinion, this will represent a revolutionary step forward to the approaches which has been followed so far.

Additional comment:

I am using an analysis framework in which placeholders such as "###(obs_data_path)###" are used for variables in analysis scripts (which are e.g. in ncl, R, python, etc) which are then inserted e.g. based on values specified in .xml files. In other words, the xml file and the analysis scripts are parsed by a preprocessor that then inserts whatever values are provided by the .xml file into the scripts (e.g. paths to data, etc.) before the scripts are automatically executed. I liked this more than the interface approach in which various interfaces are used for the various languages. In my diagnostic package, one can combine diagnostics into packages by specifying the package name in the .xml file and then run a package of scripts. I do, however, sometimes ask myself whether I should convert to a language such as python that would make the whole construct more uniform.

See our response above.

Technical comments:

p.1 line 4: Scientifically more research -> nice pleonasm

Changed.

p.9 line 7: was the list intended to be in alphabetical order?

Changed to alphabetical order.

p. 10, line 2: can -> could

Changed as suggested.

p. 13, line 18 this is -> this would be

Changed as suggested.

p. 13, line 19f in shared -> a shared

We think this sentence is correct and did not change it.