1st ESMValTool Workshop 2022

The first ESMValTool Workshop 2022 took place at DLR Oberpfaffenhofen from 13–15 June as a hybrid meeting. Participants were from BSC, DLR, Met Office, NLeSC, PML, SMHI, U Bremen and U Reading. The main goal of the workshop was to bring together the development community, discuss future strategies and provide updates on progress since the last workshop in November 2022. Specifically, the workshop included the following topics:

- Update and current status of the consortium agreement and contributing license agreement (CLA)
- Team reports
  - Technical Lead Development Team
  - Science Lead Development Team
  - User Engagement Team
- Scientific long-term strategy
- Observational datasets
- Strategy for handling and communicating implementation of non-backward compatible changes with the aim of minimizing unnoticed breaking of recipes new changes
- Testing strategy (including automatic comparison of results)
- Technical strategy for ESMValTool v3.0
- General discussion
  - New ESMValTool website
  - Handling of inactive developers
  - Optimization of the tool’s performance
  - Future funding and new road map

Some participants of the ESMValTool Workshop in June 2022.

In the following, thoughts and ideas discussed during the individual sessions are briefly summarized as bullet points. This summary is rather intended as a reminder of the sessions for the participants and therefore makes no claim of completeness.

1 Update and current status of the consortium agreement and contributing license agreement (CLA)

- transparent governance: developed principals in a formal document
- six partners/institutes are interested
- CLA
  - part of the agreement, code developers have to sign
  - purpose: ESMValTool has then the right to distribute the code, protect us as a consortium, strong recommendation from Met Office legal office
  - in practice as easy as possible, we use linked CLA assistant in GitHub, integrated in pull request workflow (one button when opening a pull request, needs to be signed only for the first pull request of a user)
- Data privacy: data stored in database on the cloud platform; administrators of ESMValTool repository have access; not generally visible; main branch: Apache License, agreement is signed
- People having no personal copy right but their institute → check with the institute/legal person who could authorize that, ensure in advance if there is the chance to sign

2 Team reports

2.1 Technical Lead Development Team

- check release notes
- Saskia is responsible for new release (2.6); Manuel supports
- meeting every two months, agenda is published on the community website on GitHub

2.2 Science Lead Development Team

- team responsibilities
- role of the team lead, new members, gallery on the website, slides and logo, different version of observational data
- finding reviewers for pull requests: approach: assign people directly
- workshop topics
  - maintainer not available anymore
  - testing strategy
  - scientific long-term strategy, backward compatibility, general diagnostics
- communication of dates regarding releases (feature freeze)
- installation instruction specific for target groups
- NCL diagnostics: priority list of diagnostics we would like to keep (GitHub issue)

2.3 User Engagement Team (UET)

- number of tutorials given has increased, different users
- documenting tutorial activities on GitHub: report and feedback from performed tutorials
- updating the tutorial regarding ESMValTool development: updating, adding new lessons
- different categories (stages, project specific, organisation specific, tool/component specific)
- challenges: maintaining tutorial, staying updated, feedback from users, standardized tutorial, documentation vs tutorials, funding
- way forward: collect feedback, ways to communicate: UET meeting, ESMValTool monthly meetings, UET email address

2.3.1 Action items

- **Ranjini (and others):** Meeting with some people from the Met Office to talk about ideas about getting feedback from tutorial participants

3 Scientific long-term strategy

3.1 Main aim of the tool

- Do we still want to keep recipes fully functional in the latest release? Yes, but…
  - Broken recipes reflect badly on ESMValTool
  - If we encounter a recipe broken, mark as broken and remove e.g. after three releases
  - Also remove code of broken recipes, but keep documentation with link to last working ESMValTool version
  - Offer option to publish diagnostics independent of the release
  - Divide recipes into general-purpose / frequently used recipes and special recipes from publications
  - Be clearer what we expect from maintainers, also add possibility to step down, but also add possibility to "revive" recipes that do not work anymore (user could ask for help)
- Future of NCL diagnostics
  - In future NCL might not usable anymore
  - A list of diagnostics which should be prioritized has been compiled: (https://github.com/ESMValGroup/ESMValTool/discussions/2617)
- More documentation for diagnostics, not only for recipes to make things easier to find. Maybe introduce tags; user friendly search functions for documentation
  - Maybe two groups regarding coding standards: high standards for basic recipes, lower entrance point for specialized recipes

3.2 Non-maintained recipes

- delete code and keep documentation
- Keep information on last working version and add links to the documentation
- Possibility for users to ask to bring back / reactivate a recipe
- Making responsibilities more explicit in the documentation and allow for people to step down as maintainer

3.3 Prioritizing diagnostics for testing (releases)

- define which recipes are expected to run + e.g. remove recipes that do not work
- periodically test a set of selected recipes that cover as many features as possible
- automated comparison of recipe output would be very helpful for release managers
- progress has been made over the last year: automatic download feature greatly helped with testing recipes
- Testing using new general-purpose diagnostics

3.4 Addition of set of generic diagnostics

- Monitoring diagnostics close to such general-purpose diagnostics
- Currently available diagnostics are often difficult to find for new users (complex file names, etc.)
- use monitoring diagnostics as example recipes
- add monitoring diagnostics to tutorial
- monitoring diagnostics good addition to tutorial
- Example recipes for general-purpose diagnostics could be used for testing

3.5 Action items

1. **Birgit & Axel:** Compile description of what it means to be a maintainer (**all:** check the pull request and comment!)
2. **Lukas:** Add GitHub usernames to config-references.yml file
3. **Manuel:** Make “maintainer” field mandatory in recipes by introducing a unit test; compile list of unmaintained recipes; add information that field “maintainer” is necessary to “contributing” section
4. **Axel:** Open pull request about what to do with non-working recipes (**all:** check the pull request and comment!)

4 Observational datasets

4.1 Cmorizer Interface

- The Cmorizer may not use extract dates function in ESMValTool - if it did then it would be advantageous — to be checked/confirmed by Remi.
- All data sets present with cmorizers but downloads may have limitations if you have to enter credentials.
- Those who can should try out the new features and report back if something does not work.

4.2 Tier system for obs

- We may need to have legal department involved in this since we deal with licenses. We need to check if redistribution is allowed and connect Tier classification with that.
- Are we allowed as data poolers to allow access to others? What qualifies as redistribution?
- MOHC legal colleagues can be consulted on this - may take the next year for some clarity on this.

- For ESM2025 project, data duplication to another compute center in France is needed - so may mean we have to assign some urgency to it — one temporary solution for this is to register ESM2025 as an entity and apply for access to Tier 3 data sets.
- User perspectives - users want access to obs data without hassle (e.g. on JASMIN and MOHC machines) and also people who want to share their data through ESMValTool (either as a ready data set but also for raw data + cmorizer provided) - questions for the legal team.
- Can licensing information be made available with the downloading script? Possibly but the format and where this information may not be readily accessible so it will take effort to put that together.
- Data created specifically for projects especially if not available to the public, should probably not be shared and neither will the cmorizer scripts be but can be maintained in a private branch. We only keep cmorizer/data available if the data can be shared.
- List what we want to do with our data - download to a shared platform, distribute across shared platforms, analyze data and share analysis results. Then ask legal team what to do given the data sets and access and ask legal team which of the actions we can do on each data set? Also ask for what license will be acceptable for new data sets on which we can do the actions listed above. Perhaps when this is mature then we can document this as minimum requirements for data sets to be shared with ESMValTool.
- Should we update Tier classification status periodically? Yes, but this will require updating recipes with the new Tier status after moving the data.
- At this time, should we keep Tier classification? Maybe we can remove the mention of Tier in recipes and have it determined automatically form the directory structure. We can have a mapping in extra facets. For now, we will keep Tier classification in principle and in recipes till we hear back from legal team and revisit this.

4.3 Distribution of data between computing centers

- rsync is done manually. Data centers have reservations to set up cron jobs with ssh-keys due to security concerns as the transfers would be password free. It is a push model so JASMIN has to agree for Levante to write without password. If changed to a Pull model, JASMIN will need password free access but only to read.
- Three possibilities -(a) synchronize more or make cmorization easier on the duplicate servers. (b) A merge on the dataset PR-> web hook-> starts cmorization on connected sites (c) manage data on individual nodes.

4.4 Automatic updates and versioning of obs data

- Different data sets may have different dates for when new data is added so perhaps an automatic check monthly should be made to see when individual data sets get updated.
- Maybe no need to overengineer this - people working on data sets can trigger change with a PR. If datasets are not used much then we can wait and do it annually.
- Data locations may change as well.
- One solution proposed - explicit list of datasets for which we need updates. Maybe have the download script look it up.
- Provide clear guidelines in doc for when you need a new version or newer data for existing data sets.
- We should support different versions of same dataset.
- Can consider automating updates for some datasets e.g. ERA5.

4.5 Action items

1. **All:** Those who can should try out the new cmorizer framework and report back if something does not work
2. **Alistair:** check with the legal team to see how we can proceed with the data licenses and how to distribute the data
3. **Klaus & Rémi:** maybe switch to a pull-from-the-other-host model rather than push-from-DKRZ model

5 Strategy for handling and communicating implementation of non-backward compatible changes with the aim of minimizing unnoticed breaking of recipes new changes

- "Proposal team" (Alistair, Axel, Bouwe, Emma) will develop the contributed ideas (from Nov. 2021 workshop) and put a proposal to the community
- Iteration with community → v1 strategy
- Implement – reiterate – review
- Key Points
    - Balance between effort for users (upgrading existing recipes/diagnostics) and effort for developers (flexibility)
    - How often should we have a breaking release? ~1 per year
    - How should we signal breaking releases to users?
    - Helping users + developers to upgrade: advice on upgrading in tutorial + release-specific guidance in the release notes
- Most breaking changes will affect only a few recipes → scale on "how breaking" a breaking change is. It's also very difficult to plan ahead since additions are not really planned → improve planning within the core developer team.
- Add version number to recipe to support multiple different recipe formats – In principle good idea, but probably too much effort (development + maintenance), also confusing for users
- Keep features optional in the beginning (e.g., instead of making the "title" mandatory at the beginning), deprecate later
- Deprecations can make things more complicated → People need to commit to remove it later, code tends to grow more complicated
- Need to improve communication (one warning in possibly huge log file is not enough)
- Technical changes need to be addressed by technical personnel, maintainers, communicating that to users is really difficult
- Maybe the technical team is not the right team to communicate these changes. Get support from the user engagement team or science lead team regarding these changes? E.g., "popular" summary of the release in a "not so techy" language
- Communication channels
    - Newsletter that summarizes changes, e.g., send around changelog in newsletter?
    - We need attention, also from people not fully active as developers on GitHub → e.g., website, Twitter, mailing list.

  - Move deprecations to the end of the log file, not the beginning (visibility!), only deprecations that are relevant for the corresponding recipe

6 Testing strategy (including automatic comparison of results)

- A tool is now available to compare ESMValTool output from one (or multiple) runs to each other (or multiple other) runs and prints a summary of the changes
- Script is available in esmvaltool/utils directory
- Simple recipes to periodically test recipes to avoid getting surprised before new releases
- Example branch with 2 recipes is here: https://github.com/ESMValGroup/ESMValTool/tree/workshop_test_recipes_al
- Bouwe will start creating the framework to test these recipes
- Consider different realms (atmosphere, land, ocean), start with monitoring diagnostics

6.1 Action items

1. **All:** If you have recipes, put them in a branch and open a pull request yourself (to ESMValTool/esmvaltool/recipes/testing).
2. **Bouwe:** will start creating the framework to use these recipes for regular testing

7 Technical strategy for ESMValTool v3.0

- HPC job templates e.g. for slurm
- Iris support for dask.distributed schedulers
- Move task-oriented multiprocessing towards dask approaches (fair amount of thinking and discussion required)
- Chunking strategy?
- Cloud computing, what's the status, how to improve.
- CORDEX
  - Would be nice to have an implemented standard grid that is compatible to something like CORDEX and fix/unify small differences in coordinates and meta data on the fly.
  - Focus on support for data that we are claiming to be able to work with. Not reaching the goals, can also affect funding.
  - Replace claim that we can process CORDEX data with “we strive to support CORDEX data” in the documentation
  - Meet with Iris people to discuss technical issues that affect both.

7.1 Action items

1. **Saskia & Pep:** Working on CORDEX pull requests

8 General discussion

8.1 New ESMValTool website

- current status about the website hosted on DLR system → quick and easy contributions are very hard
- moving to GitHub pages now, website is hosted by GitHub
    - all ESMValTool developers can open pull request
    - merging pull requests by admins (Veronika, Axel, Bouwe, Klaus, Remi, V); alternative: UET
    - no requirement for pull request reviews
- Website mobile phone friendly?

8.2 Handling of inactive developers

- Finding reviewers is usually difficult because many developers seem inactive
- 158 developers now listed in the ESMValTool team on GitHub
- Inactive GitHub users could be easy entry points for hackers
- Some of the developers develop in the private repository, don't appear as active in the public branch
- List of developers helps for telling funders how many people use ESMValTool, more visibility through developers, developers should open issues when start working, so everyone could see what others are doing
- GitHub discussion on how to deal with inactive developers will be opened for everyone to comment on
- 2 years of no activity on GitHub could be a threshold for moving someone to e.g. group "former developer"
- Highlight contributors (increase visibility)

8.3 Optimization of the tool's performance

- see section 7

8.4 Future funding and new road map

- A list of all projects with involvement of the ESMValTool would be helpful
- Getting funding to ESMValTool is getting more difficult
    - work together as consortium, bringing other institutions together in a project, share contacts/ideas/… in the consortium
    - keep basic of expertise, difficulties about funding are the uncertainties about e.g. follow-up projects
- strategy/idea: making more preprocessor functions lazy (about 50% of the preprocessor functions are not lazy yet)
- eScience center support scientists with software development, looking for consortium (need to be Dutch scientists): ideas and contacts would be helpful
- SMHI was contacted by an Italian group searching a postdoc to evaluate high resolution models with ESMValTool
- Would it be a good idea to have a wish list of missing features for people who are writing proposals

(hybrid meeting, all times are given in CEST)

13 June 2022 - Day 1 (Monday)

10.15 am – 10.20 am Introduction, welcome and workshop agenda (Birgit)

10.20 am – 10.30 am Update and current status of the consortium agreement and contributing license agreement (CLA) (Alistair)

10.30 am – 10.45 am Report from Technical Lead Development Team (V)

10.45 am – 11.00 am Report from Science Lead Development Team (Axel)

11.00 am – 11.15 am Break

11.15 am – 12.15 pm User Engagement Team

a. Introduction – Tasks and responsibilities of the User Engagement Team (Lisa)

b. Strategy and types of tutorials (Ranjini)

12.15 pm – 01.30 pm Break

01.30 pm – 03.30 pm Scientific long-term strategy

a. Introduction and main aims of diagnostics (Axel)

b. Porting of diagnostics from NCL to Python (Axel)

c. Dealing with recipe maintainers no longer active (Lee Bouwe)

d. Prioritizing diagnostics for testing (releases) (Manuel)

e. Addition of set of generic diagnostics (Manuel, Saskia)

03.30 pm – 03.45 pm Break

03.45 pm – 04.00 pm Summary and wrap up of first day (Birgit)

14 June 2022 - Day 2 (Tuesday)

10.15 am – 10.20 am Introduction day 2 and workshop agenda (Birgit)

10.20 am – 12.00 pm Observations

a. Short demo on the new cmorizer interface (Manuel)

b. Revision of current tier classification (Axel, Birgit)

c. Distribution of data between computing centers (Klaus)

d. Automatic updates and versioning of observational datasets (Klaus)

12.15 pm – 01.30 pm Break

01.30 pm – 02.30 pm Strategy for handling and communicating implementation of non-backward compatible changes with the aim of minimizing unnoticed breaking of recipes new changes (Alistair)

02.30 pm – 02.45 pm Break

02.45 pm – 03.45 pm Testing strategy (including automatic comparison of results) (Birgit, Bouwe)

03.45 pm – 04.00 pm Break

04.00 pm – 05.00 pm Scientific long-term strategy (continued from Monday) (Axel, Manuel, Klaus, Saskia)

05.00 pm – 05.15 pm Summary and wrap up of second day (Birgit)

06.30 pm Workshop dinner

15 June 2022 - Day 3 (Wednesday)

10.15 am – 10.20 am Introduction day 3 and workshop agenda (Birgit)

10.20 am – 12.00 pm Technical strategy for ESMValTool v3.0 including e.g., (V, Bouwe, Klaus)

a. Higher resolution models
b. Handling of complex grids including CORDEX datasets
c. Parallelization, memory management, cloud computing

12.15 pm – 01.30 pm Break

01.30 pm – 03.00 pm General discussion

a. New ESMValTool website (Axel)
b. Handling of inactive developers (Alistair)
c. Optimization of the tool's performance (Bouwe)
d. Future funding and new road map (Birgit)

03.00 pm – 03.15 pm Break

03.15 pm – 04.00 pm Wrap up and future workshops (Alistair, Birgit)

The workshop was funded by the EU Horizon 2020 research and innovation programme under the grant agreement No 824084 (IS-ENES3 project).