Project: **1437**
Project title: **GitLab CI/CD pipeline for MESSy development**
Principal investigator: **Patrick Jöckel**
Report period: **2025-07-01 to 2026-06-30**

The project 1437 is supposed to provide the resources to execute the CI/CD pipeline tasks using the “levante-runner”, with which jobs are sent via slurm to levante and hence need (parallel!) computing resources, for each member of the MESSy GitLab project. As such, the project supports all other MESSy related projects at DKRZ.

Activities during allocation period

- In the MESSy GitLab the CI/CD pipeline currently covers these tasks, triggered by merge requests into the main developer branch:
    - build (configuration and compilation) of the MESSy software with the GNU, NAG and NVHPC compiler suites: For our upcoming open source release of MESSy, we intend to provide both, an autoconf/make and a cmake/make/ninja based build system. Therefore, some builds are currently performed redundantly.
    - After successful merge into the main developer branch, a web page is automatically generated, which lists the remaining compiler warnings (NAG and GNU) as a guideline for further code improvements.
    - A web-page with the list of publications is automatically created, whenever publications are added to the documentation.
    - A specific openACC-“accelerated” DWARF setup is compiled with the NVHPC compiler suite and a (very) short run-time test is performed, utilizing GPU-node-h.
    - A test for the automatic code generation of the kinetic solver (submodel MECCA) with the NAG compiler suite has been proven extremely beneficial, since expansion and modifications of the chemical kinetics, due to its complexity, can easily break features, which are not necessarily applied in standard setups, such as for instance extended diagnostics.
- The major on-going development within MESSy aims at a complete revision of the ICON/MESSy model towards a MESSy plugin for the ICON-ComIn Community Interface (***Hartung et al., 2024***): With this approach, the basemodel ICON will no longer be part of the MESSy repository, but the standard ICON model is compiled with the ComIn extension separately. MESSy is build separately as a shared library and only at run-time loaded by the ICON-ComIn call-back library. The corresponding build (currently with the GNU compiler suite only) incorporated during the previous allocation period as a task in the MESSy GitLab CI/CD pipeline turned out to be extremely useful for this overall development. The corresponding pre-compiled ICON model (incl. all its submodules) is stored (to be used by the GitLab CI/CD pipeline and by all users of ICON-ComIn/MESSy!) under the ***/work*** *s*pace of the project. A repeated re-compilation of ICON within the MESSy GitLab CI/CD pipeline is therefore becoming obsolete as soon as the development of ICON-ComIn/MESSy (i.e. MESSy as a ComIn plugin) has been completed.

  For the future storage of the pre-compiled ICON, additional **/work** space is required, since several builds for debugging & testing and production simulations need to be stored for all available compiler suites and for different versions of ICON.
- In the **/work** space of this project, we furthermore store additional software that is required for the ICON-ComIn/MESSy model system and which is shared among all MESSy users. In the recent allocation period, a specific python environment (py34m, including the pre-compiled **yac** coupler with python-bindings and **yaxt** library) was added, which python internally binds xarray etc. to the correct HPC MPI and netCDF libraries. This

environment needs to be installed for each compiler suite (NAG, GNU, INTEL, NVHPC) and various versions (due to updates of yac and/or yaxt) are required concurrently. The py34m is required for our MPMD setup with externally coupled, python-based data input and output servers, which are currently under development for MESSy.

- The rest of the **/work** space is utilized to store test and benchmarking data (model output for inter-comparison with output of further developed model versions).
- For the allocation period we requested 10000 node-h on levante plus 500 GPU-node-h, the WLA granted 4000 node-h and 120 GPU-node-h, from which we utilized about 2137 node-h and 95 GPU-node-h (status on March 23, 2026). This under-utilization is a consequence of using only single-core configuration and compilation tasks so far, but parallel run-time tests were not yet performed. The latter are planned for the near future, thus more resources will be required.

References:

Hartung, K., Kern, B., Dreier, N.-A., Geisbüsch, J., Haghighatnasab, M., Jöckel, P., Kerkweg, A., Loch, W. J., Prill, F., and Rieger, D.: ICON ComIn – the ICON Community Interface (ComIn version 0.1.0, with ICON version 2024.01-01), Geosci. Model Dev., 18, 1001–1015, https://doi.org/10.5194/gmd-18-1001-2025, 2025.