Project: **1437**
Project title: **GitLab CI/CD pipeline for MESSy development**
Principal investigator: **Patrick Jöckel**
Report period: **2024-05-01 to 2025-04-30**

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**

- The transition from the old “levante-fake runners”, based on virtual machines, to the “levante runners” utilizing levante nodes and slurm job scheduling for the CI/CD tasks, as suggested by DKRZ, could successfully be accomplished.
- 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:
    Since we are currently in the transition from autoconf/make based builds to cmake/make/ninja based builds, some builds are currently performed redundantly. Once the transition to cmake/make/ninja based builds is complete (with all possible features), it is planned to disable the autoconf/make based builds.
  - 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.
  - Very recently a test for the automatic code generation of the kinetic solver (submodel MECCA) has been added (NAG compiler suite).
- 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. A corresponding build (currently with the GNU compiler suite only) is already incorporated as a task in the MESSy GitLab CI/CD pipeline. 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*** space 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.
  The rest of the **/work** space I 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, the WLA granted 6300 node-h, from which we only utilized about 150 node-h (status on April 14, 2025). This low 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.

- In addition, we were for the first time able to use about 40 GPU-node-h for testing a simplified accelerated DWARF setup (see above). Testing accelerated code will be expanded in near future, in particular including run-time tests of accelerated MESSy in comparison to CPU-only MESSy.

**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.