Sprint Documentation 20

ICON-YAC-CLEO2 sprint

Lakshmi Aparna Devulapalli[1], Clara Bayley[2] and Wilton Jaciel Loch[1]

[1]German Climate Computing Center GmbH, Hamburg, Germany
[2]Max Planck Institute for Meteorology, Hamburg, Germany

Contact: info@nat-esm.de

Published on 23.02.2026 on https://www.nat-esm.de/services/accepted-sprints

1 Summary

This sprint is a follow-up of the previous ICON-YAC-CLEO sprint to finish the goals from the first sprint application which could not be completed. These goals are integral to enabling ICON simulations to use the new Super Droplet Model (SDM) microphysics. During the first sprint, the one-way coupling was completed and groundwork for two-way coupling was put in place. Additionally, MPI-based domain decomposition was implemented for distributed memory parallelization. These developments were crucial first steps to achieve the complete vision of using CLEO as a cloud-microphysics model in ICON.

The challenges which laid ahead for the second sprint were as follows 1) Make the MPI domain decomposition compatible with the CLEO-ICON coupling via YAC. 2) Complete the two-way coupling infrastructure. 3) Performance analysis of MPI parallelization, identifying bottlenecks and suggesting solutions. All these tasks used various configurations of the ICON bubble experiment to test.

The sprint is viewed as partially complete by both RSE and the scientist. We were able to achieve only half of the tasks in the time available. We generalized the initialization of communicators in CLEO and made CLEO's algorithms work with irregular grid sizes, towards the goal of making MPI domain decomposition compatible with YAC coupling. We were able to implement the infrastructure required for two-way coupling; however, testing with the bubble experiment showed that whilst the simulations (with one MPI process) finish successfully, the results are incorrect/poor for multiprocess simulations (in both one-way and two-way couplings). The goal of performance analysis of parallelization, identifying and improving bottlenecks could not be started.

2 General Information

Start and end date:	01.04.2025 – 30.09.2025
Intended period:	5 months
Responsible RSE:	Lakshmi Aparna Devulapalli (DKRZ)
Responsible scientist:	Clara Bayley (MPI-M)

The sprint involved three software packages - CLEO , YAC and ICON. YAC is the coupling library, acting as a bridge between CLEO and ICON to exchange their thermodynamic and wind fields. These three software packages together represent the coupled system of CLEO-YAC-ICON.

CLEO - Within the last decade, a new model for cloud microphysics called the Super-Droplet Model (SDM) has become increasingly relevant for climate research because in comparison with previous models of cloud microphysics, SDM has a much less ambiguous representation of cloud condensates and numerous computational advantages. CLEO is a C++ standalone implementation of SDM being developed to make superdroplet simulations with ICON in large domains (O(100km)) at lower resolution (circa. 100m) computationally feasible. It employs Kokkos as the main performance-portability layer, natively allowing multiple methods for shared memory parallelization as well as accelerator offloading. CLEO is divided into libraries dedicated to specific aspects of the simulation, like grid management and superdroplet movement, all tied together by example drivers. The C++ code has a size of around 10500 LOC.

YAC - The YAC Coupler Library (Yet Another Coupler) is a high-performance, modular software framework designed to couple components of Earth system models. It provides robust support for grid interpolation, parallel data exchange, and time coordination between heterogeneous model components. YAC is built for scalability and flexibility, enabling accurate and efficient coupling on massively parallel HPC systems.

ICON - The ICON (ICOsahedral Nonhydrostatic) model is a state-of-the-art global modeling system used for numerical weather prediction and climate simulations. It employs an unstructured icosahedral grid and nonhydrostatic dynamics, enabling high-resolution, scalable simulations from global to regional scales on modern HPC architectures. For this sprint, we used the ICON-MPIM repository.

3 Sprint Objectives

The goals listed in the sprint-application document are mentioned below.

1. Run the ICON 3-D cloud bubble test case: one-way coupled to CLEO on more than one node. (Independent of goals 2 and 4).
2. Run the ICON 3-D cloud bubble test case: two-way coupled to CLEO on one node. (independent of goals 1 and 4).
3. Run the ICON 3-D cloud bubble test case: two-way coupled to CLEO on more than one node. (Independent of goal 4).
4. Measure the computational performance of the MPI communication with an increasing number of superdroplets and gridboxes using a fixed thermodynamics test case of standalone CLEO. (Independent of goals 1, 2 and 3).
5. (stretch-goal) Measure the computational performance of the MPI communication as in goal 4 above but with the one-(or two-)way coupled 3-D cloud bubble test case.
6. (stretch-goal) Make CLEO's MPI domain decomposition compatible with its GPU parallelization. (Independent of goals 1, 2, 3, 4 and 5).

These goals were broken down into smaller sub-steps. The details of these sub-steps are explained in the next section in a chronological order.

4 Procedure and Insights

4.1 Technical Approach / Procedure

We started to tackle the task of making the MPI domain decomposition compatible with the YAC coupling first, before extending the two-way coupled infrastructure.

To finish this task, the following subtasks needed to be achieved. These contribute towards goal #1.

1. **Communicator class** - The goal was to allow the communicator generated by YAC as an alternative to `MPI_COMM_WORLD` when YAC coupling is enabled. A communicator class was introduced to handle the switch between these two communicators. This communicator class object is initialized in the `main.cpp` of the experiment being run. By analyzing the configuration setup for the experiment, the communicator class decides whether YAC is enabled or not. Based on this decision, an appropriate communicator for the simulation is initialized. Further, this communicator is used throughout CLEO to obtain process rank and the size of the communicator at various locations. The discussion and progress made during the implementation of the communicator class for CLEO led to having a different folder structure in CLEO. The scientist created a new folder "Configurations" and reorganized the files to handle interdependencies better.
2. **Irregular Grid boxes** - A small but crucial logic is required to determine the movements of superdroplets between gridboxes. During the first sprint, an algorithm was developed to implement this logic with the assumption of having same sized grid boxes throughout the domain. But to be compatible with coupling to ICON via YAC, CLEO's algorithm should be able to handle irregular gridbox sizes. The irregular nature of the gridbox sizes comes into picture because of the irregular vertical height levels of ICON's domain. The aforementioned algorithm was modified to be able to work with different gridbox sizes in all three directions in the CLEO domain.

At this stage, CLEO was tested with `fromfile` and `fromfile_irreg` test setups and it was successful.

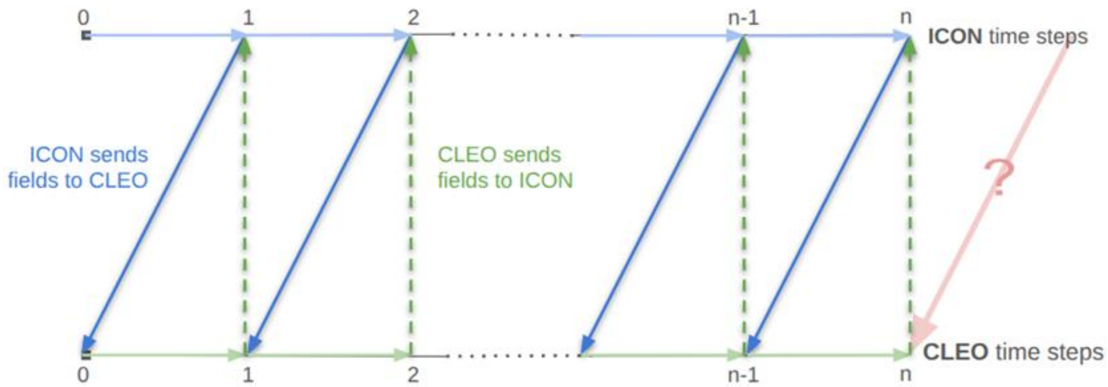
3. **Local sizes for data** - During the first sprint, the bubble test setup was made to work with only one process. But to make it work in a parallel environment, CLEO's code needed to be made compatible with multi-process. Global arrays holding gridbox indices were converted to hold local sections of the gridboxes. It was an interesting exercise to play with partition sizes, origins and bounds that helped achieve this task.

The next major task was to extend the coupling infrastructure to two-way coupling. The following subtasks would contribute towards achieving goal #2 and #3.

4. **Two-way coupling** - Extending the two-way coupling infrastructure turned out to be more complicated than assumed at the beginning of the sprint. The one-way coupling infrastructure failed to work with smaller `nproma` (arbitrary number to improve computational cache efficiency) size in ICON, which led to implementing the fix for the problem before moving forward with extending the two-way coupling infrastructure. Information about ICON's grid levels being stored in the reverse order was useful in developing the logic to send fields from ICON to CLEO. There were a couple of interesting challenges that are quite common with any two-way coupled setup and we tackled those in this sprint.

   The first challenge was the staggered nature of the coupling between CLEO and ICON. They are two-way coupled through YAC, which means that both components can send data to each other.

Additionally, they are coupled in a serial way in the sense that one waits for its counterpart to finish its timestep, receive data and then continue its simulation. The ICON side of coupling with YAC is designed in such a way that the coupling timestep or the exchange of data begins at the first time step. Whereas CLEO expects data from ICON from its zeroth time step. This creates a staggered nature of the coupling between ICON and CLEO (explained in the diagram below), and it also means CLEO runs behind ICON and waits for data from ICON at each coupling time step. The challenge happens at the last time step when ICON finishes its simulation, while CLEO is still waiting for data from ICON to execute its last time step. This indicates that CLEO has more "gets" than the number of "puts" that ICON can provide. This problem was solved by using the YAC builtin function `yac_cget_action()`.This function provides the information about the last "get" which will be issued by CLEO, and we explicitly stopped the final "get" to avoid a deadlock and to allow both components to finish their simulations successfully. With the caveat that the last time-step of CLEO will have stale data to use. This can be corrected by concluding CLEO's simulation at its second last time-step.



The second challenge was related to ICON not exchanging its fields from the zeroth time step. We used a concept called "Lag". It is a concept in YAC coupling which allows us to manipulate the date and time associated with the fields being exchanged as per our requirement. Since ICON exchanges data from its first time step, we have to introduce a lag in ICON's field definitions for the YAC to recognize ICON's first time step as the first valid "put/get" instead of the zeroth time step when simulation starts.

5. **Flag for one-way or two-way coupling**: Along the way, it became crucial to switch between one-way and two-way coupling for testing and debugging. A rudimentary flag, a named constant, indicating one-way or two-way coupling was implemented in ICON. To activate this flag, ICON code needs to be modified in two locations by choosing ***oneway_coupling*** or ***twoway_coupling*** at the following two locations.
   a. `Loc#1 :- icon-mpim/src/coupling/mo_atmo_coupling_frame.f90`
      `CALL construct_atmo_cleo_coupling_post_sync(.., [one/two]way_coupling)`
   b. `Loc#2 :- icon-mpim/src/atm_phy_aes/mo_aes_phy_main.f90`
      `if(is_coupled_to_cleo()) CALL interface_cleo(..., [one/two]way_coupling)`

This flag was set as component metadata for CLEO to let it know as well and issue "puts/gets" accordingly. This was useful to test various configurations towards the end of the sprint.

4.2 General Insights

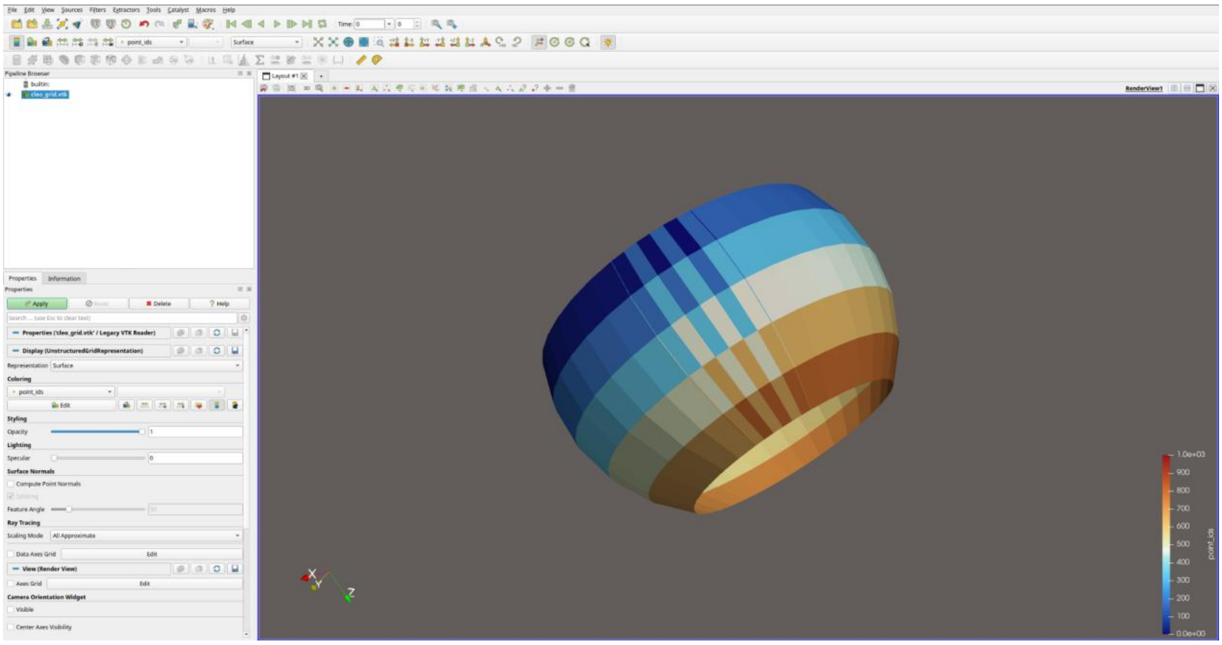
Unexpected hurdles are part and parcel of any project. During the sprint time, there were a few instances where run time or compile-time bugs took considerable time to resolve and move forward.

All the Sprint meetings and discussions were always aimed at understanding the fundamentals of the functioning of CLEO as a superdroplet model. Those discussions clarified many topics for the scientist and the RSE. Wilton Jaciel Loch, who was the responsible RSE during the first sprint, was instrumental in some deep-dive discussions that were needed to understand the algorithms implemented by him.

While extending the coupling infrastructure to two-way coupling, many discussions were aimed at making the parts of code which receive and send data in CLEO consistent and complementary for better readability of the code. Meetings were held with Moritz Hanke to understand the intricacies of coupling two components with YAC. He suggested a robust alternative for switching between one-way and two-way coupling in CLEO-YAC-ICON setup. This will improve the overall functioning of CLEO and ICON together.

Scripts developed by the scientist for running CLEO were extremely useful for testing and debugging during the development of the sprint tasks. We were able to exchange setups and reproduce results to be on the same page during various stages of the development.

The bubble test setup has a rectangular grid structure which is assumed to wrap around the sphere in the horizontal direction. Whereas, ICON has a Icosahedral grid structure which is made up of triangles. The data exchange between two different types of grids is prone to approximations which need to be scientifically looked at. The visualization below shows that the rectangular domain of the bubble test case is not perfectly wrapping around the sphere and has some overlap along the vertical edge. This is expected because the bubble test case is simply a dummy representation for latitudes and longitudes and not made for spherical domains. The results of the coupling between ICON and CLEO deteriorate over time and the wind fields are inverted. The scientist is currently investigating potential causes, e.g. due to ICON's boundary conditions and the choice of interpolation stack in YAC.



During the initial weeks of the sprint, we had recurring conversations about managing Github, e.g. how to organize branches and pull requests to suit the collaboration between the scientist and the RSE. However, towards the end of the sprint, we faced difficulties in testing due to conflicting git commits. A better git management strategy would be fruitful in the future. . Throughout the sprint, the communication between the scientist and RSE was frank and honest which led to a straightforward collaborative experience.

5 Results

We were able to generalize the initialization of communicators in CLEO and make CLEO's algorithms work with irregular grid sizes, towards the goal of making MPI domain decomposition compatible with YAC coupling. The implemented communicator class handles configurations by detecting whether YAC is enabled or not. This communicator-class object needs to be initialized in the `main.cpp` of each experiment and all the experiment files were adapted to have this development.

Further, CLEO's gridbox finding algorithm was successfully adapted to work with irregular gridbox sizes in all three dimensions. This made the code much more flexible, robust and more importantly compatible with ICON's vertical grid structure. To test the bubble experiment with more than one process, lengths of the grid definitions in CLEO were modified to represent the local partitions of the grid. All of these steps made it possible to test the bubble experiment in a one-way coupled setup on multiple processes. We had a successful testing, and this work was merged back into the main branch of CLEO.

Later, the two-way coupled infrastructure was implemented in both ICON and CLEO. The code structure in the ICON repository was inspired by the code structure of other modules which are coupled with ICON via YAC. The code structure in CLEO, as mentioned in the "Technical Approach" section, was made consistent to match the style of the already implemented one-way coupled setup. A flag was implemented to switch between one-way and two-way coupled setups. This was done by setting a constant in ICON and passing the data to CLEO through `yac_component_metadata`. But unfortunately, at the end of the sprint we could not test the two-way coupled setup on more than one process successfully. Following is the status of the testing of the CLEO-YAC-ICON coupled setup in various configurations:

Experiment	**Coupled to YAC**	**Number of processes**	**Test Result**
fromfile	No	1 and >1	Success
fromfile_irreg	No	1 and >1	Success
Bubble	Yes (one-way coupled)	1	Runs successfully, but scientific plots are not right
Bubble	Yes (two-way coupled)	1	Runs successfully, but scientific plots are not right
Bubble	Yes (one-way and two-way coupled)	>1	Breaks, runs forever

Significant time towards the end of the sprint was taken to understand the problems in the two-way coupled setup with >1 MPI process and making the bubble experiment run as expected. Many debugging methods were employed, without much success. Finally, at the end of the sprint, the cause of the bubble experiment breaking in a two-way coupled setup was better narrowed down. It is probably because of incorrect coupling for the wind fields, which is then indicated by an error raised in the algorithm for superdroplets movement. From the initial goals of the sprint, the goal of performance analysis of parallelization, identifying and improving bottlenecks could not be started.

It is agreed that the work done in the ICON repository is complete enough to be merged back into the ICON repository. Naturally, a Merge Request was opened, and it subsequently received some helpful suggestions by Moritz Hanke. These suggestions were deemed important and useful for CLEO and ICON, by both the scientist and the RSE and they need to be implemented in future. The suggestions include a more future-proof method of implementation of the flag to switch between one-way and two-way coupled setups via configuration files which can be read by both components - ICON and CLEO.

6 Conclusions and Outlook

The sprint tasks were challenging in a good way. As the tasks were tackled one by one, the discussions taking place parallel opened a lot of new points which can be addressed in the future for a robust and user-friendly coupled setup between CLEO and ICON. Since not all tests were successful in the end, the following points still remain open for development.

1. MR in ICON - The suggestions provided by Moritz Hanke need to be implemented and tested. Finally, a guard needs to be added in ICON to avoid users using the two-way coupled setup until the results of the simulations are corrected.
2. The bug(s) in the one-way and two-way coupling with > 1 MPI process need(s) to be identified and resolved.
3. The bug(s) in the quality of the one-way and two-way coupled simulation need(s) to be identified and resolved.

Since this sprint could not successfully produce a two-way coupled setup which runs on multiple processes, a lot of work needs to be put in towards making CLEO a fully functional cloud microphysics model for ICON. Scientifically, results need to be analyzed, and the code needs to be corrected in case of bugs.

Further down the road, CLEO can be completely parallelly coupled with ICON, which can be the scope for another natESM sprint. This will avoid the staggered nature of coupling and also potentially speed up the simulation time of the coupled setup.

7 References

1. Precursor sprint - [ICON-YAC-CLEO Sprint #14]
2. YAC Documentation - https://dkrz-sw.gitlab-pages.dkrz.de/yac/index.html
3. Publications under review:
   - https://doi.org/10.5194/egusphere-2025-4398
   - https://doi.org/10.5194/egusphere-2025-4399