www.messy-interface.org

Coupling MESSy via ComIn to ICON

Astrid Kerkweg[1], K. Hartung[2] , B. Kern[2] , L. A. Devulapalli[3], W. J. Loch[3], A. Mitic[3], P. Jöckel[2]

[1] Institute of Climate and Energy Systems – Troposphere (ICE-3), Forschungszentrum Jülich, Germany
[2] Deutsches Zentrum für Luft- und Raumfahrt, Oberpfaffenhofen, Germany [3] Deutsches Klima-Rechenzentrum, Germany

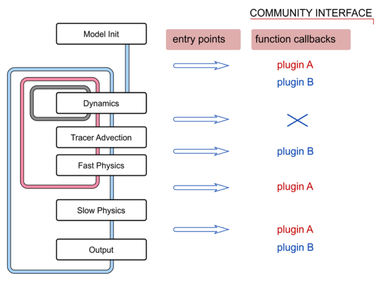
Abstract:

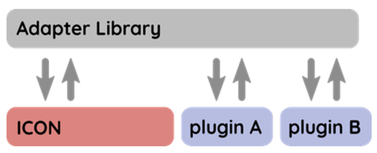
The ICON Community Interface (ComIn) was developed to provide a stable interface for integrating a variety of third-party models (plugins) into the ICON model. In the meantime, a number of ComIn plugins has been developed implementing diagnostics or specific physical processes. However, a fully functional coupling of a complex modelling framework using ComIn is still pending.
The Modular Earth Submodel System (MESSy) is a rather complex integrated framework and it is mainly used for simulations of atmospheric chemistry. For applications with complex atmospheric chemistry, the computational effort is more than 10 times higher than for a purely dynamic ICON simulation. Efficient coupling through ComIn is crucial here. Furthermore, MESSy features among others, its own timer, an own tracer infrastructure and its own memory management including input and output as well as checkpointing facilities, drastically increasing the challenges for coupling MESSy via ComIn to ICON.
The general integration of MESSy into ICON via ComIn has progressed far, also with the support of two natESM sprints, but several specific details remain, mainly related to limitations of the flexibility of ICON. We will provide an overview about the challenges met during the implementation and discuss open issues, which are not only relevant for MESSy but also for other plugins, which can only be solved from the ICON model side and not by ComIn.

ICON **Com**munity **In**terface
Contact: comin@icon-model.org

What are the aims of ComIn?

- ➡ Providing a standardized **public interface** for third party codes (‘**plugins**’) coupled to ICON
- ➡ Significantly **reduced maintenance** for ICON as well as for third party code developers
- ➡ Plugins **easier to migrate** to new ICON releases
- ➡ Establishing ICON as the core model for applications ranging from **NWP** to **ESM**
- ➡ Enables **multi-language support** (Fortran, C/C++, Python)

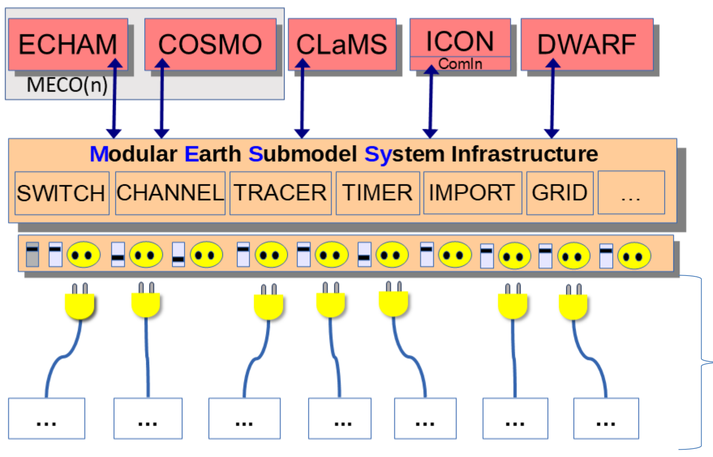




How does ComIn work in a nutshell?

- ➡ ComIn organizes the **data exchange** and **simulation events** between the ICON model and multiple plugins.
- ➡ **ComIn Callback Register**: Subroutines of the plugins are called at pre-defined events during the ICON simulation.
- ➡ The **ComIn Adapter Library** is included by ICON and the plugins. It contains descriptive data structures and regulates the access and the creation of model variables.

The Modular Earth Submodel System (MESSy)



(legacy) **base models**

infrastructure
connecting different types of base models (grids, decomposition, …) with base model independent MESSy submodels

submodel interface
connect submodel core (= scientific content) to infrastructure

sub-model = process descrip-tion

submodel core
base model independent: diagnostics, physical or chemical process in basic entity (e.g. box for a kinetic solver, column for convection)

MESSy was mainly developed to provide a flexible framework for all kinds of chemistry climate and air quality applications.

MESSy provides a middleware for coupling atmospheric legacy models (e.g., ICON, ECHAM, COSMO) and specialized ESM components, the so-called submodels (e.g., physical parameterizations, chemistry packages, diagnostics) via generalized interfaces for standardized control and coupling. Currently more than 100 submodels are included in MESSy.

Literature:
- 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. Discuss., https://doi.org/10.5194/gmd-2024-135, 2024.
- Jöckel, P., Kerkweg, A., Pozzer, A., Sander, R., Tost, H., Riede, H., Baumgaertner, A., Gromov, S., and Kern, B.: Development cycle 2 of the Modular Earth Submodel System (MESSy2), Geosci. Model Dev., 3, 717–752, https://doi.org/10.5194/gmd-3-717-2010, 2010

Encountered challenges for coupling plugins to ICON via ComIn

Plugin changes the wind:

- The ICON prognostic variable is the egde-normal velocity component v_n .
- MESSy submodels usually provide tendencies for the horizontal wind velocity components u and v.
- Challenge: when u,v are required by a MESSy submodel u,v need to be calculated from v_n. If u,v tendencies are provided, v_n needs to be updated.
- Solution: ComIn 0.5.0 provides the required descriptive data (rbf interpolation coefficients) for the transformation. However, routines required from `iconmath` must either be copied or `iconmath` needs to be coupled to the plugin as well.
- Status: implementation in MESSy in progress.

Plugin changes temperature:

- The ICON prognostic variable are Exner pressure and Θ_v .
- MESSy submodels usually provide tendencies for air temperature t .
- Challenge: each change in t and/or the hydrometeors changes the Exner pressure and Θ_v .
- Solution: each time a MESSy submodel changes the temperature or the humidity / hydrometeors, the Exner pressure and Θ_v need to be adjusted.
- Status: implementation in MESSy in progress.

ICON code changes required for fully enabling the MESSy plugin

Access to locally defined variable:

- ICON calculates its processes and includes only those variables in the global variable streams, which are required inside the ICON code.
- ComIn only exposes variables which are part of the global variable lists.
- MESSy and other plugins require access to variables, which are only locally defined within an ICON parametrisation, e.g., the process specific tendencies of each individual conversion process in the cloud microphysics scheme:

```
zqvt = sev + sidep - ssdep - sgdep - snuc - sconr
zqct = simelt - scau - scfrz  - scac  - sshed - srim - srim2
zqit = snuc + scfrz  - simelt - sicri + sidep - sdau - sagg - sagg2 -siau
zqrt = …
```

- Challenge: the local variables need to become part of global variable lists to be exposed via ComIn to the plugins.
- Solution: implementation of so-called “optional global variables”, which are ordered in optional global variable lists, which can be
  - requested via namelist (process-wise, i.e., microphysics, land, turbulence etc.)
  - deactivated via preprocessor directives.
- Status: a testing version for MESSy is implemented in an ICON branch

Dimension Semantics:

- Variables in the ICON variable namelist can have an arbitrary order of ranks. There is no fixed rule about the order and no information regarding which rank refers to what dimension (nproma, block, vertical dimension, number of tracers, number of tiles, … asf.).
- MESSy organises its internal variable accesses via its own memory management (submodel CHANNEL). Each of the ICON variables exposed by ComIn needs to be mapped into a so-called “channel object”. This requires also the information about the length and the meaning of the dimensions.
- Challenge: so far ICON does not ask for this information when variables are added, thus it can not provide the information necessary for an automated remapping, access of variables.
- Solution: implementation of the so-called “Dimension Semantics”, i.e., the information about which rank of an array serves which purpose. The following meanings are distinguished:
  - COMIN_DIM_SEMANTICS_UNDEF
  - COMIN_DIM_SEMANTICS_NPROMA
  - COMIN_DIM_SEMANTICS_BLOCK
  - COMIN_DIM_SEMANTICS_LEVEL
  - COMIN_DIM_SEMANTICS_CONTAINER
  - COMIN_DIM_SEMANTICS_UNUSED
- Status:
  - ICON: under development; a draft merge request was opened to icon GitLab repository by L. A. Devulapalli, however, is not yet ready to be merged.
  - ComIn: in mo_comin_adapter a heuristic is used to deduce the actual meaning of the ranks. However, this does only work for a subset of variables.

Diagnostic Variable:

- ICON provides a lot of useful diagnostic variables in the output.
- Some of these diagnostic variables are required by MESSy submodels.
- Challenge: ICON calculates these variables **only if requested** for the output and only in time step when they are written.
- Solution: The calculation (and creation of variables) for each diagnostic variable in ICON should be triggerable from namelist in addition to the output request.
- Status: desirable.

… and more:

The other boxes show the biggest and most important development steps required to run ICON with MESSy as a ComIn plugin without the necessity to patch the ICON model. We believe, that most of these points are relevant for other plugins as well. Additionally,

- further ComIn callback points will be required and
- for replacing a parametrisation (e.g. the microphysics) an additional option / switch is required.
  - This option would skip the actual call to the parametrisation itself.
  - However, this option would allow all fields that are normally filled by the parametrisation in ICON to be passed / fed to the rest of the ICON code for further processing, as if the parametrisations were active.