natESM

DKRZ DEUTSCHES KLIMARECHENZENTRUM

JÜLICH Forschungszentrum | JÜLICH SUPERCOMPUTING CENTRE

DKK Deutsches Klima Konsortium

Federal Ministry of Education and Research

~12 participants

## Working group 2

# Interfaces and model composition

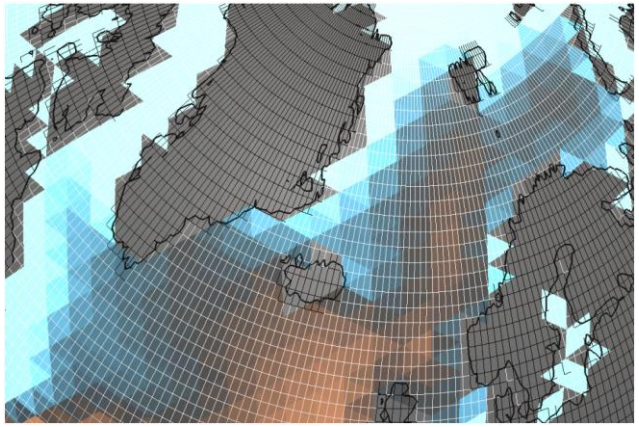
Sabine Attinger (UFZ), Ina Tegen (TROPOS), Hendryk Bockelmann (DKRZ)

# An ICONic example

The ICON development distinguishes two main interfaces

## I. Classical coupling using YAC

- External models can use different grids
- Library takes care of the fields interpolation
- Independent of ICON because YAC is standalone library

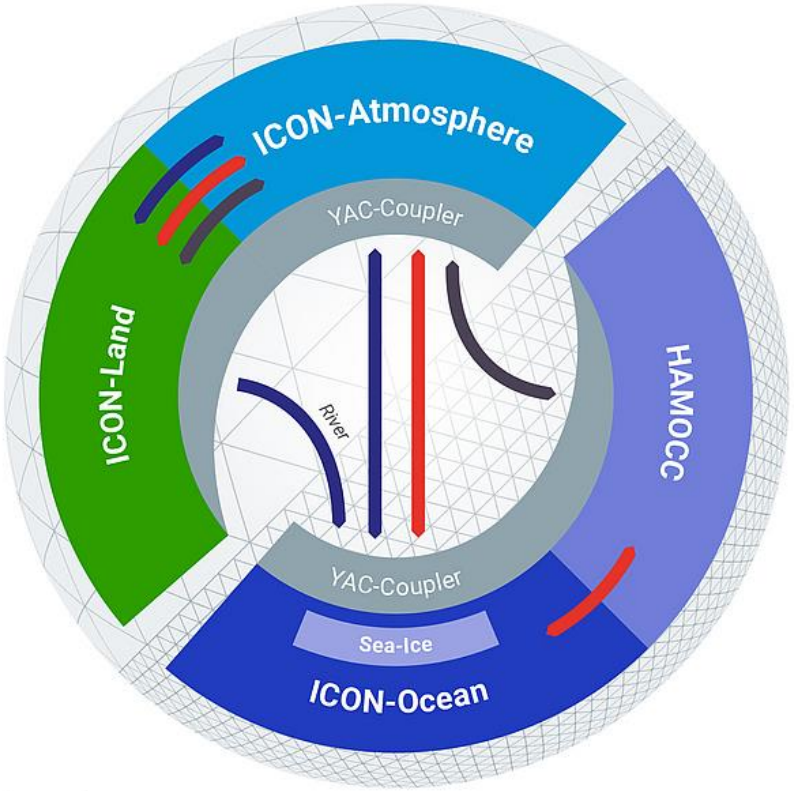


## II. Direct hook into ICON using ComIn

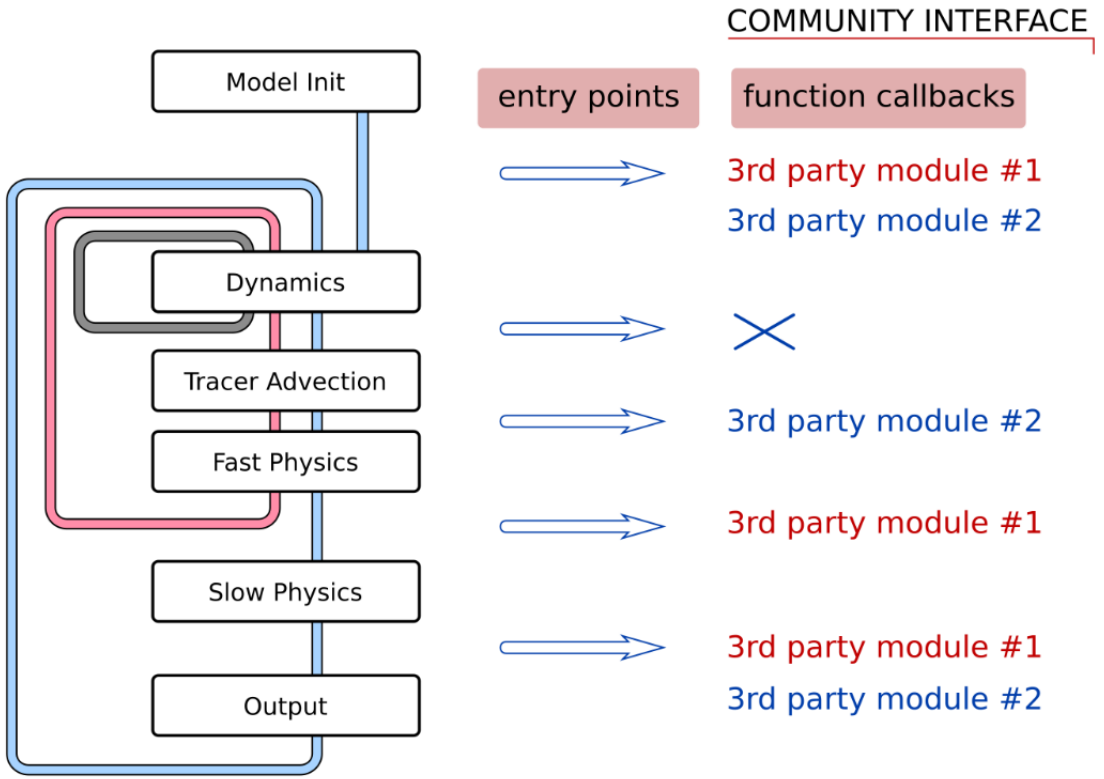
- External components register callback functions at predefined entry points
- Fields are defined on the ICON grid and data are accessed through the library
- Tight integration into ICON (aka direct access to data)

# An ICONic example

## Coupling using YAC




## Direct hook using ComIn



# An ICONic example

## I. Classical coupling using YAC

ICON-O coupled to ICON-A

FESOM2 coupled to ICON-A (just feasibility study; technically done; real experiments a different issue)

mHM coupled to ICON-A (at least one-way coupling)

HD coupled to ICON-A and ICON-O two-way

## II. Direct hook into ICON using ComIn

PoC for future ICON design by DWD and DLR

Kind of “MESSy-way” into ICON

# Guiding questions

1. What kind of interfaces do you use so far (e.g., couplers and/or direct hooks in the code), and which are missing?
2. Which models (or components) are already successfully coupled, and which should be newly enabled?
3. How to deal with varying requirements for interfaces (e.g., spatial-temporal resolution and/or same variables in different models)?

~12 people in working group with different experiences and usage of couplers/interfaces so far

# What kind of interfaces?

- Classical couplers YAC, OASIS, FINAM (https://finam.pages.ufz.de/)
  - SW/Library should support different grids, interpolation schemes
  - Choice is based on existing community; „full“ feature set & documentation; convenience in use; existing support and commitment by developer; release management
- Offline (i.e. file system based, input/output) coupling
  - Threshold for (non-)use: size of data to be exchanged, time stepping requirements, runtime
  - Benefit: you do not need to know how to run GCM if you are just interested in output data; more flexibility (e.g. time stepping scheme)
- Directly inside code (MESSy, ICON-ART)

$\Rightarrow$ Consensus that these cover nearly all needs

# What is missing?

- Efficient implementation/coupling of ensemble runs (e.g. ensemble Kalman filter; training AI)
- Energy conservation when coupling (really solved?)
- User interface to data products needed for offline (aka input/output) coupling

Follow-up discussion:

- How to ensure that coupled models still produce meaningful results? At least within a natESM setting
- How to tune model configuration? Who will do it? Needs release management and QC to build trust in model configurations and interfacing to it
- Guideline missing: when to add stuff directly into ICON (or other model) and when to use the ComIn for external usage?

# How to deal with varying requirements?

What if models use different e.g. topography? Using scaling or correction factor

- Discussion if that should/could be done in coupler/model/workflow-manager
- Issue: definition of interfaces might turn into long wish-list (cmp CMOR)
- There has to be meta-data to the field that can be exchanged via couplers
- Temporal scaling must be considered

$\Rightarrow$ Well-defined model composition still unclear, but cannot be solved by technical means alone

# Results

- ComIn highly appreciated
  - Should allow for well-defined and sustainable usage/cooperation with ICON as a natESM base model
  - Nevertheless, classical coupler will remain if external app needs e.g. own grid
- 3-4 new ideas how to use it in upcoming sprints
  - will test how ICON as core component can be used for community
- natESM should establish documentation and training for interfaces