Technical challenges in km-scale resolution earth system models

Daniel Klocke
and many colleagues at MPI-M and partner institutions

MAX-PLANCK-INSTITUT
FÜR METEOROLOGIE

Why simulating the earth system at kilometer-scale?

- More physics: resolve the dominant mode of energy transport in the tropics (vertical), eddies in the ocean, ice-leads using laws of physics, realistic lower-boundary conditions in topography and land-cover

- Improved large scale circulation, process level air-sea interactions, (better) representation of extremes, information at scales relevant for impact on peoples life (eg. catchment scales) and on scales we observe the Earth -> linking to observations

- Scale interactions from local to global scale.. and back

- Less equations, less lines of code, less bugs, less assumptions and essentially simpler models, making them easier to understand

- Don’t underestimate: Inspiration , communication and beautiful pictures

What are kilometer-scale earth system models useful for?

- Studies of convergence start to make sense

- Do we get consistent responses to perturbations - at least in sign and on climate regime scale?

- Ensembles: do we really sample uncertainty right with a large sample of structurally similar models? Do we get out-of-sample trajectories?

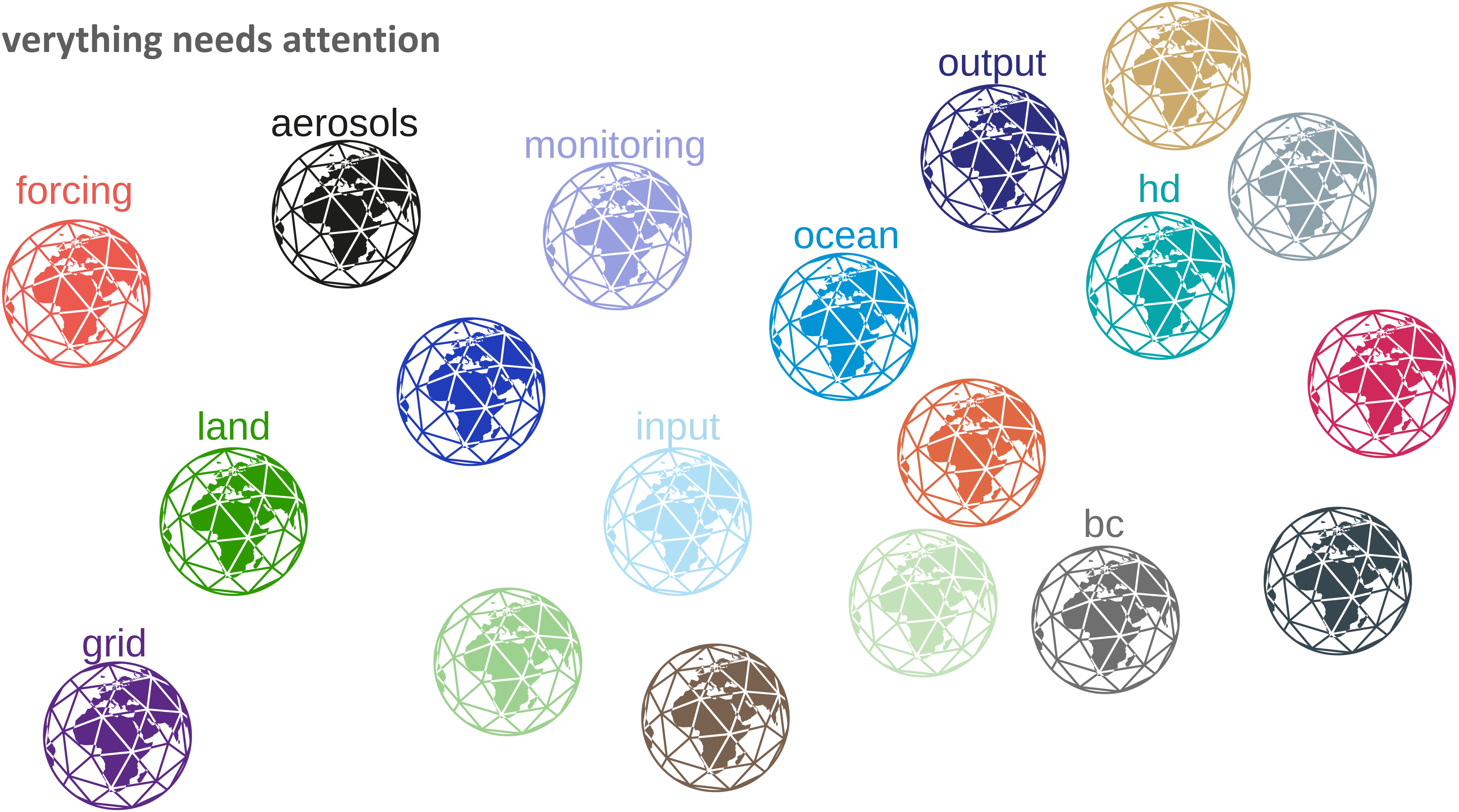
- Are current models overfitted and react too stable to perturbations?

- Fundamental questions: Will the rain forests collapse, will we see major circulation shifts, is the ITCZ structure stable, how will the monsoon margins change.. all hard to answer with current models, but very important questions.

- Simulating weather of past, present and future climates.

- Fascination of our planet. We can visualise phenomena, which we experience from our lives.

Everything needs attention



forcing

aerosols

land

grid

monitoring

input

ocean

output

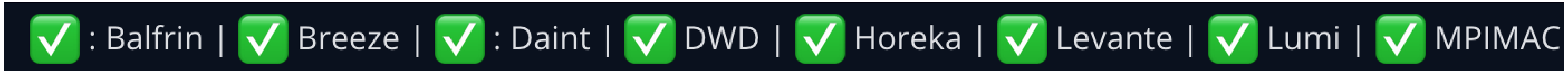
hd

bc

Diverse ecosystem of machines

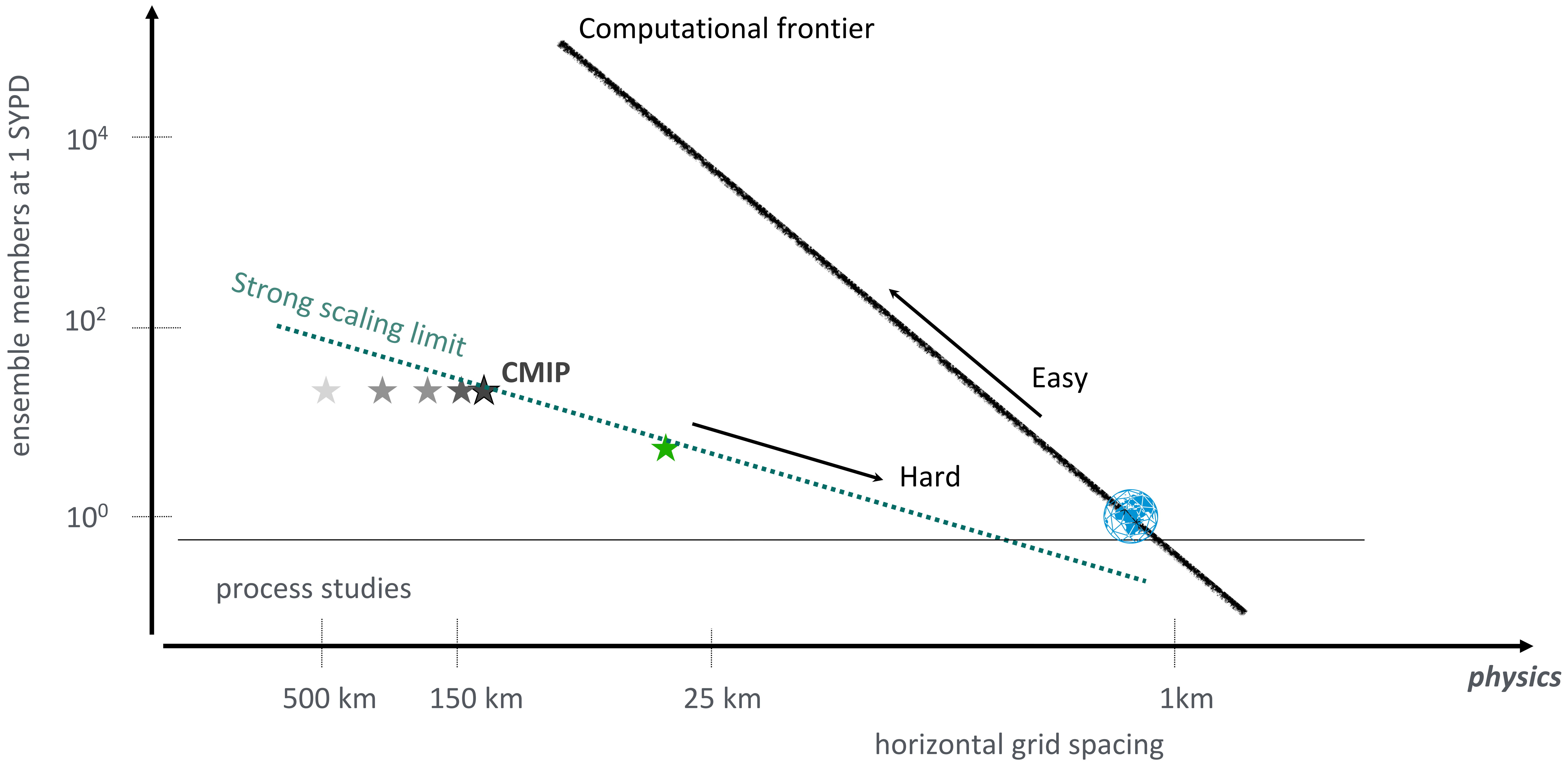
Systems ICON is currently running and tested on:

- levante@DKRZ: AMD x86 CPU and Nvidia A100 GPU
- PizDaint@CSCS: Intel x86 CPU and Nvidia P100 GPU
- juwels@JSC: AMD x86 CPU and Nvidia A100 GPU
- nec@DWD: NEC Aurora vector engine
- lumi@CSC: AMD x86 CPU and AMD MI250X GPU
- horeka@KIT: Intel x86 CPU and Nvidia A100 GPU



✅ : Balfrin | ✅ Breeze | ✅ : Daint | ✅ DWD | ✅ Horeka | ✅ Levante | ✅ Lumi | ✅ MPIMAC

Compute power increased by a factor 1 Mio, we want to use it

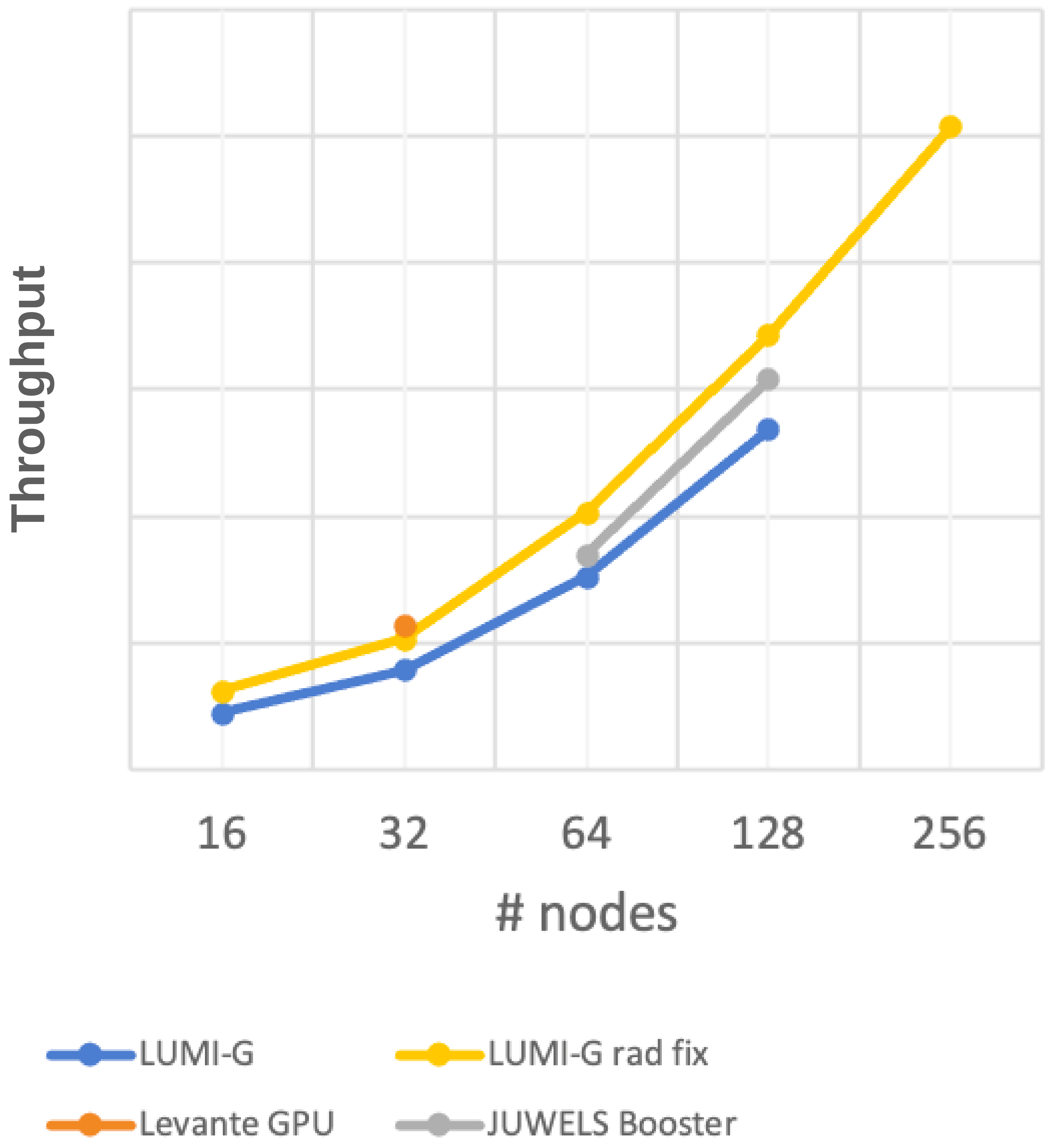


Intensity increases as Δx^{-3}, throughput as Δx^{-1} .

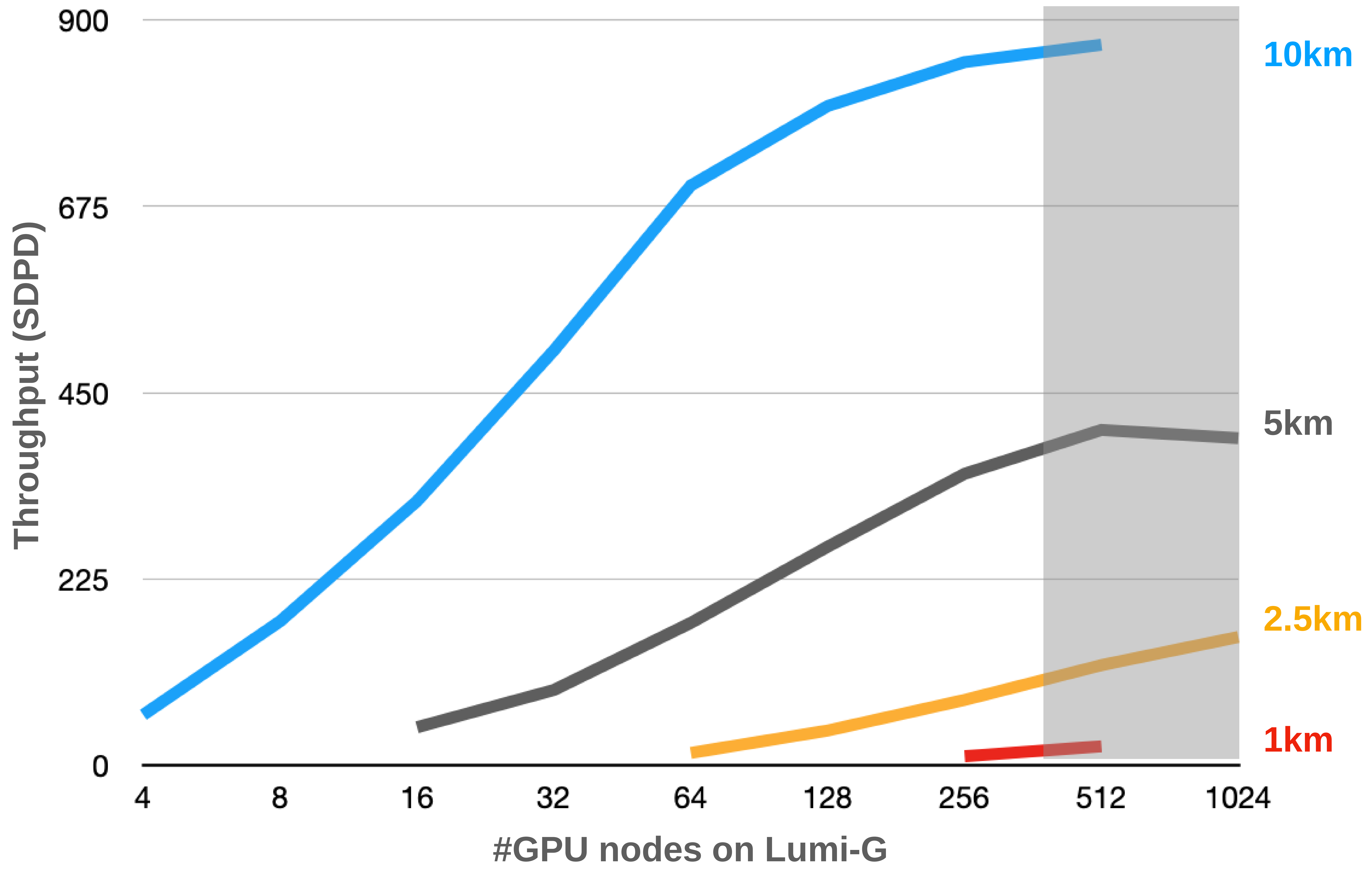
New technologies are more energy efficient (5-10x for ICON)

See also Hoefler et al. 2023

ICON Aquaplanet 5 km Performance



Atmosphere (including land)



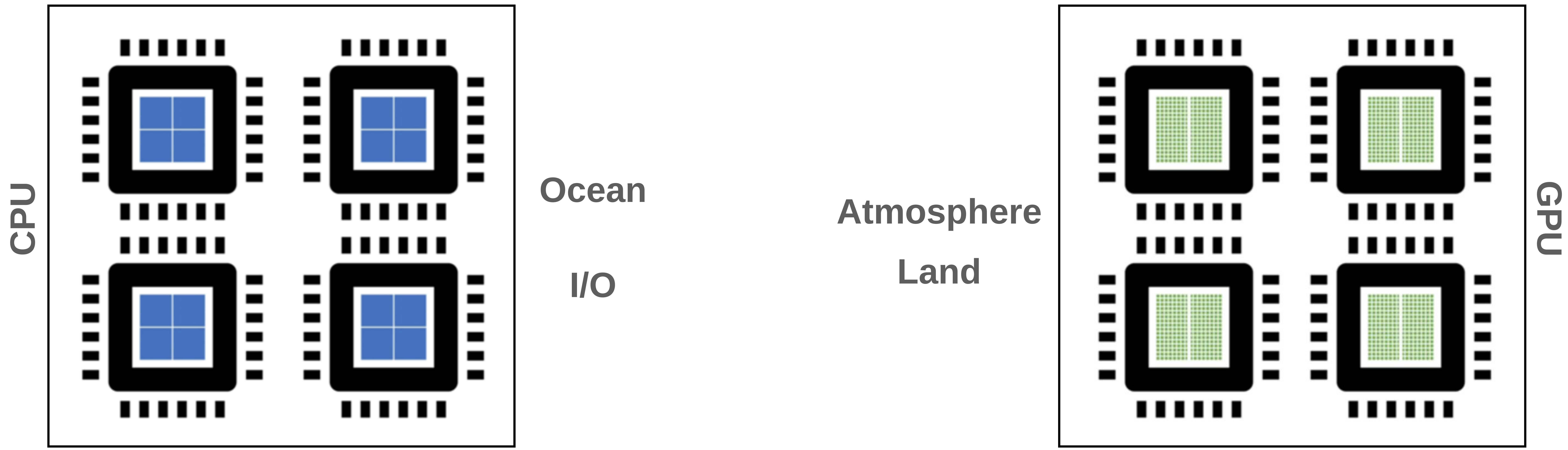
New technologies are more energy efficient (5-10x for ICON)

GPUs to rule them all?

- Porting large codes is non-trivial
- Getting GPU enabled code run performant is non-trivial
- GPU is not GPU
- GPU porting only makes sense for very compute-intensive codes/simulations
- All-or-nothing: data transfer between CPUs and GPUs can counteract all performance gains
- Loosely coupled components can still benefit

Heterogeneous configurations

- Porting large codes is non-trivial
- Getting GPU enabled code run performant is non-trivial
- GPU is not GPU
- GPU porting only makes sense for very compute-intensive codes/simulations
- All-or-nothing: data transfer between CPUs and GPUs can counteract all performance gains
- Loosely coupled components can still benefit through different compute architectures



Heterogeneous configurations - example

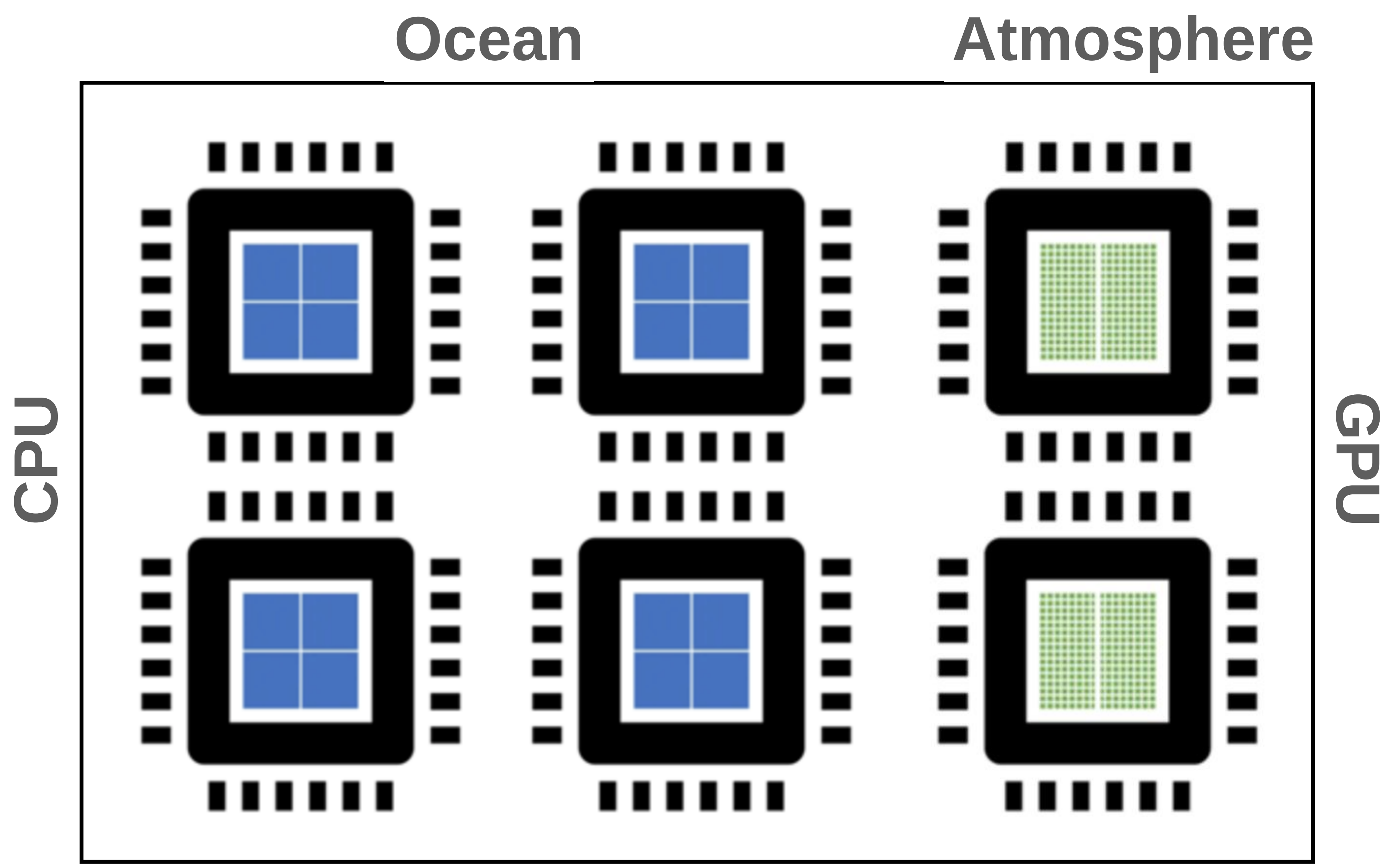
```
klockeda@uan03:~> squeue --job 6232305
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
         6232305+0 ju-strate a0uq_202 lrb_4650  R    1:23:36    128 nid[005124-005251]
         6232305+1 ju-strate a0uq_202 lrb_4650  R    1:23:36     28 nid[005252-005279]
         6232305+2 ju-strate a0uq_202 lrb_4650  R    1:23:36      2 nid[006238-006239]
```

DestinE scenario run on Lumi-G (30 years at 5km)

- 128 nodes: atmosphere and land running on GPUs
- 28 nodes: ocean running on the CPUs of the GPU nodes
- 2 nodes: I/O processes running on CPUs of GPU nodes, coupled via YAC

Heterogeneous configurations

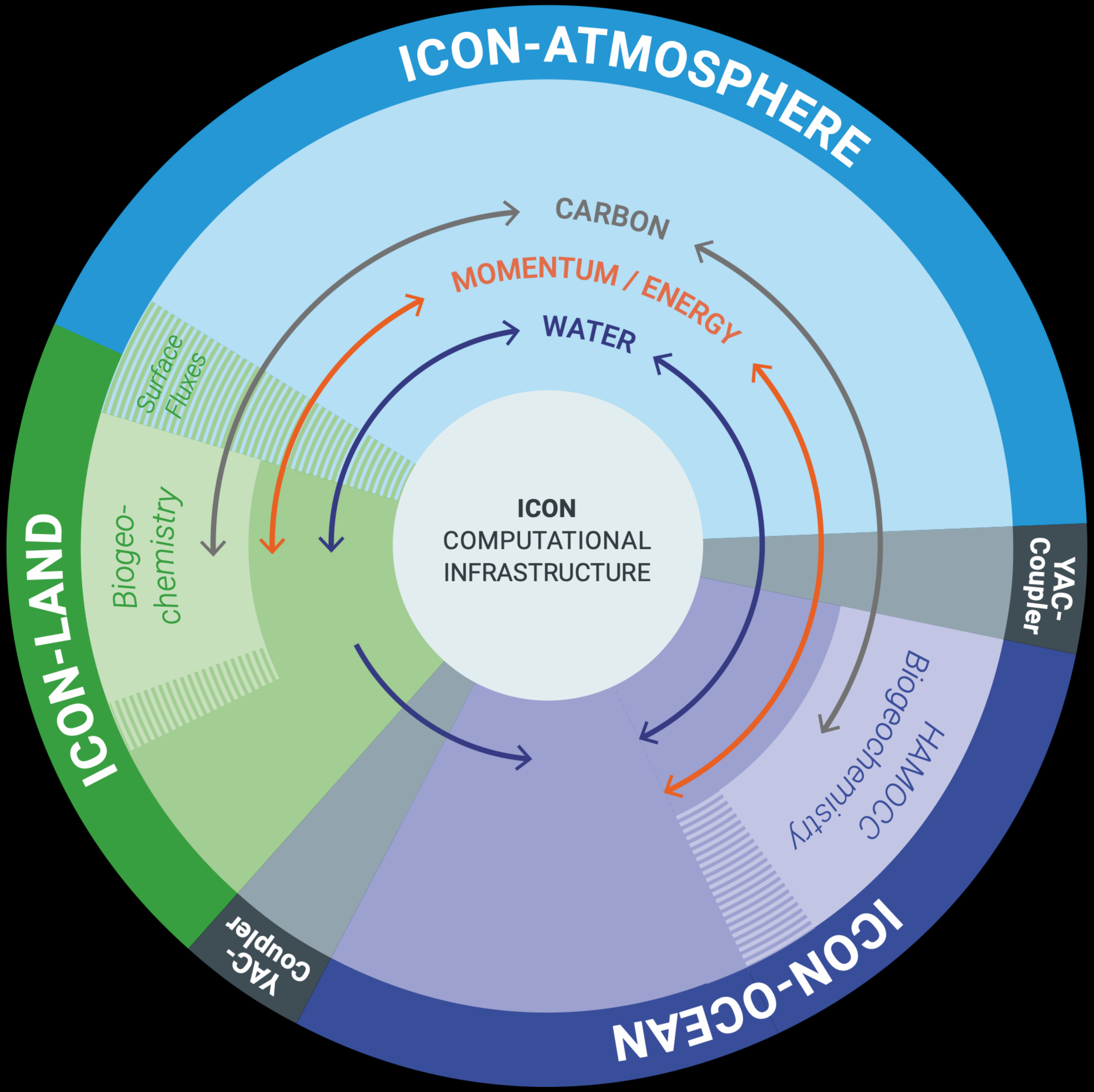
- Porting large codes is non-trivial
- Getting GPU enabled code run performant is non-trivial
- GPU is not GPU
- GPU porting only makes sense for very compute-intensive codes/simulations
- All-or-nothing: data transfer between CPUs and GPUs can counteract all performance gains
- Loosely coupled components can still benefit



Working in progress

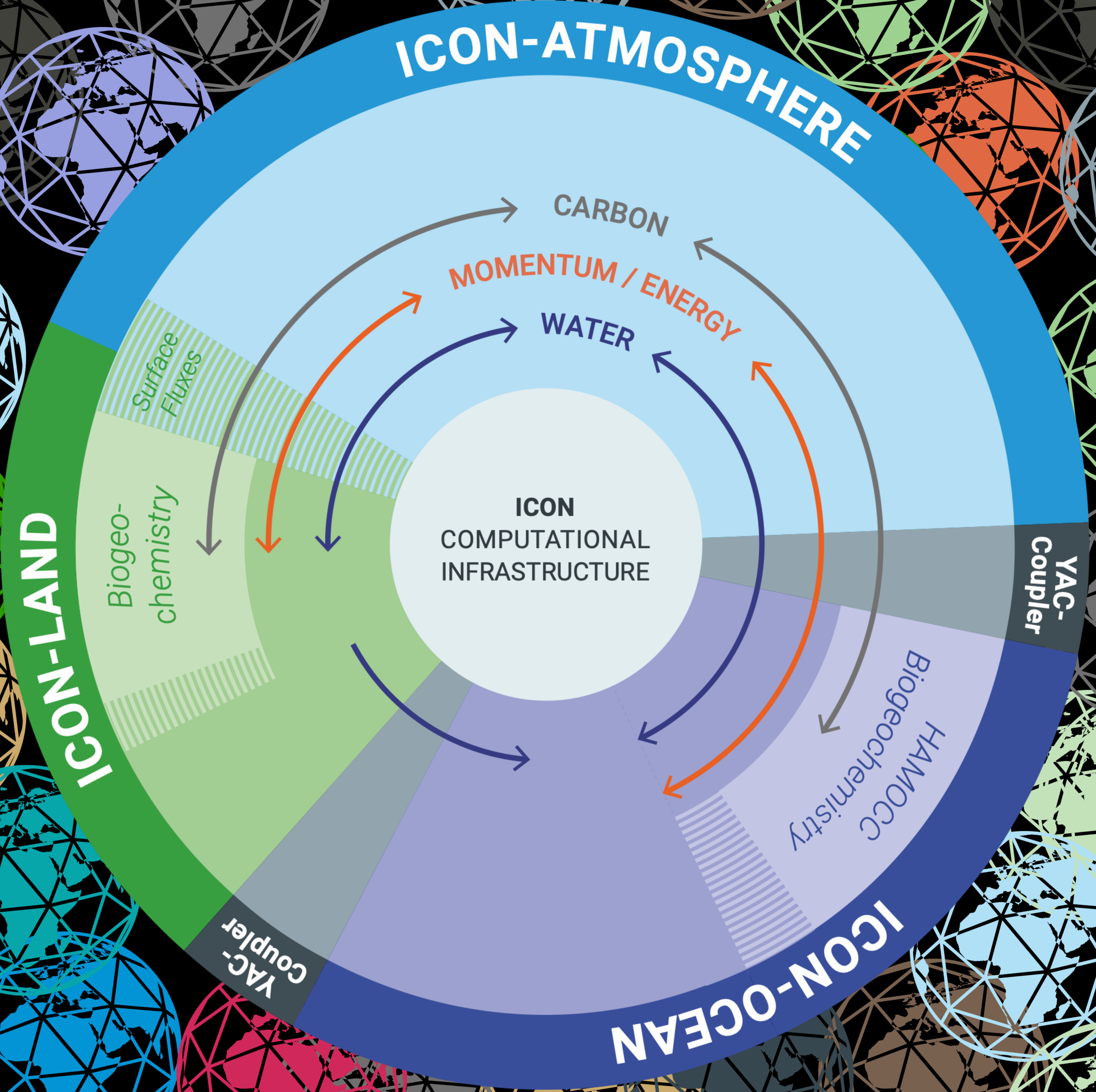
Potential speed-up and increased energy efficiency, could work across several components

Earth system complexity



Hohenegger et al. 2023

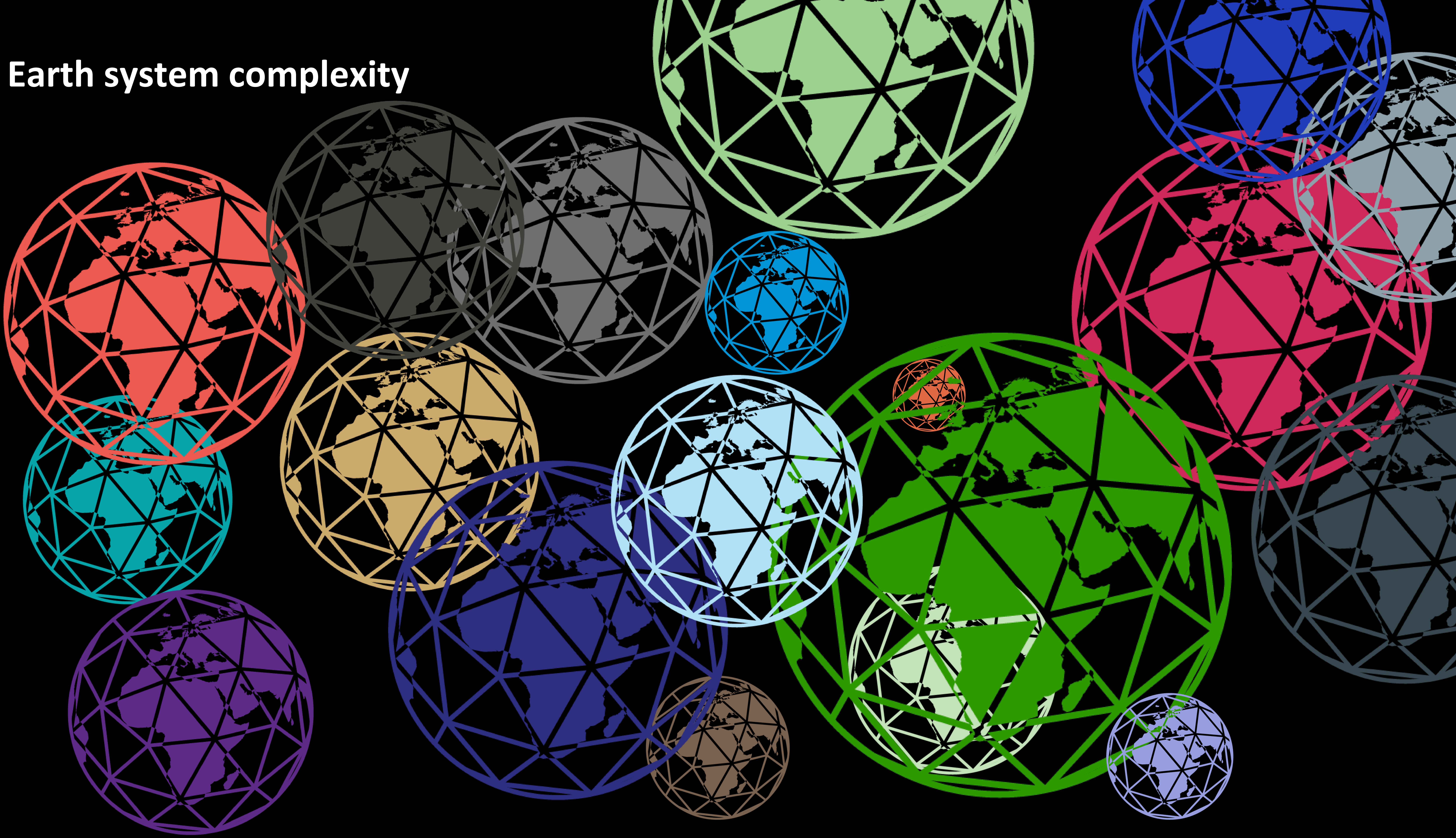
Earth system complexity



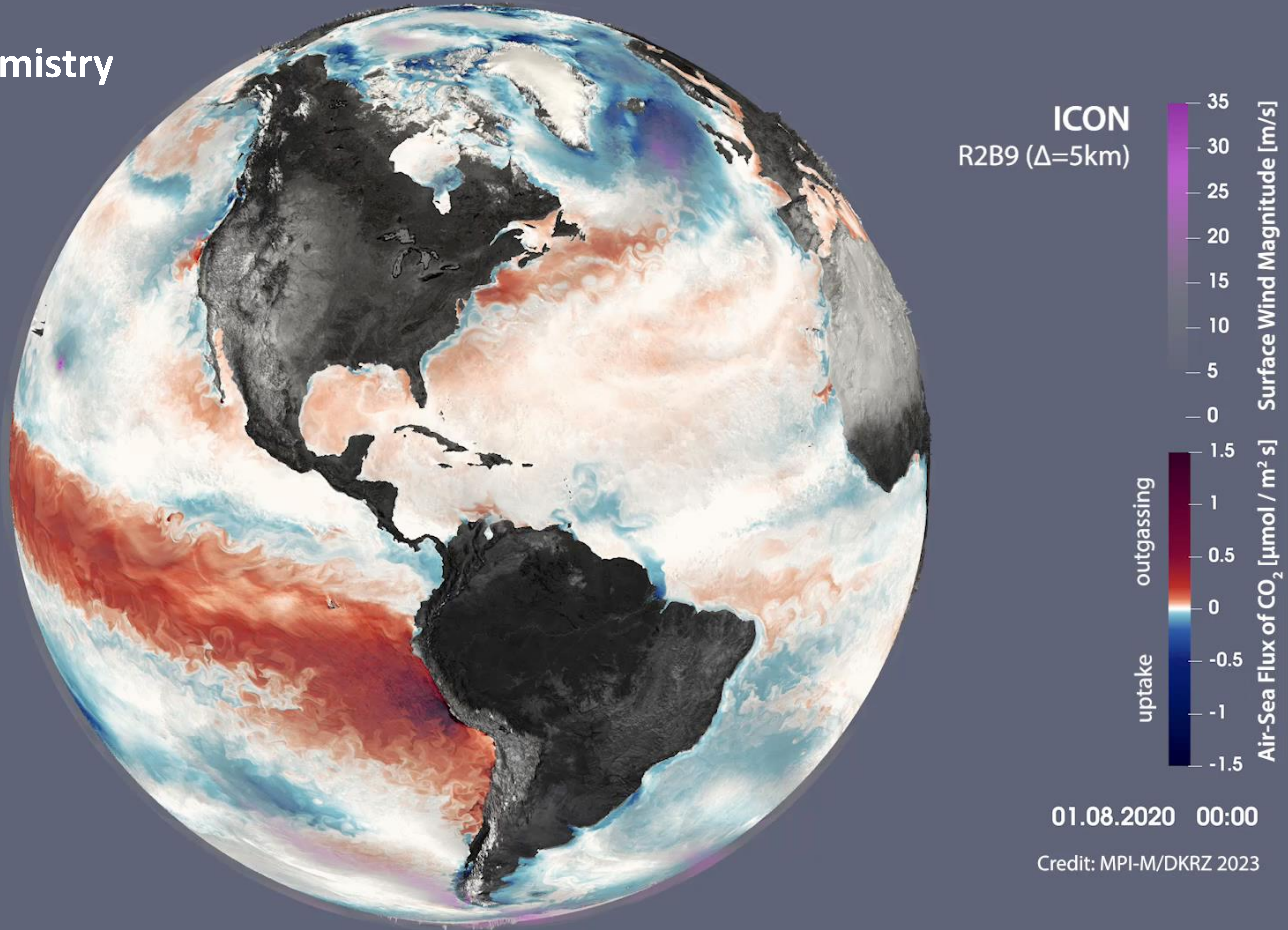
Moving ICON to Lumi

- About 1 Mio lines of (mostly) Fortran code
- 1000 Fortran source files (300 with OpenACC directives)
- A atmosphere only configuration in ‘production’ on (NVIDIA) GPUs since 2020
- All (almost) computations on GPUs -> speed up about 6
- Part Lumi procurement benchmarks
- Good collaboration with vendors (HPE and AMD) and compute centre (CSC)

Earth system complexity



Ocean Bio-geochemistry


35
30
25
20
15
10
5
0
Surface Wind Magnitude [m/s]
1.5
1
0.5
0
-0.5
-1
-1.5
outgassing
uptake
Air-Sea Flux of CO$_2$ [µmol / m^2 s]

ICON

R2B9 (Δ=5km)

01.08.2020 00:00

Credit: MPI-M/DKRZ 2023

Interactive aerosols

1e−1
4
2
0
clivi clivi (kg/m^2)
1e−4
1.0
0.5
0.0
su (kg/m^2)
1e−6
5
0
bc (kg/m^2)
1e−5
1
0
ss (kg/m^2)
1e−3
5.0
2.5
0.0
du (kg/m^2)

nextGEMS

Philipp Weiss, University of Oxford

Interactive aerosols

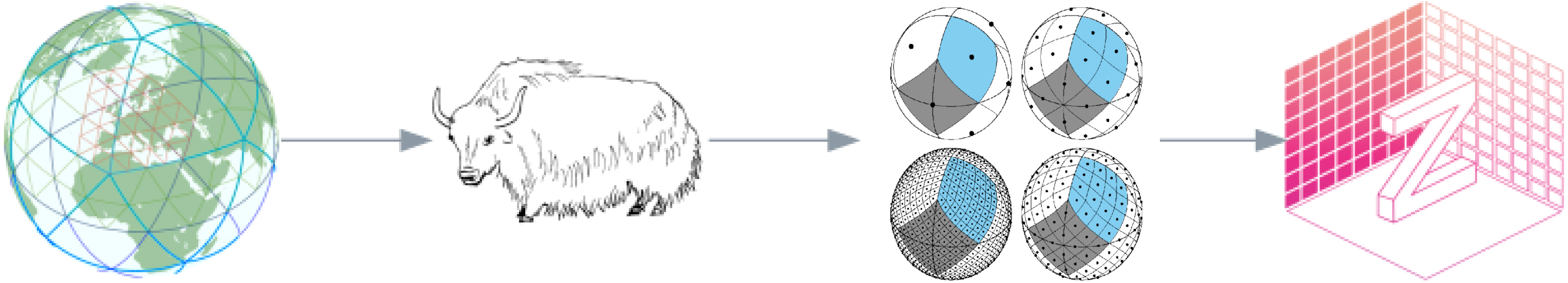
2020-09-25T16:15:00Z
1e−1
4
2
0
clivi (kg/m^2)
1e−4
1.0
0.5
0.0
su (kg/m^2)
1e−6
5
0
bc (kg/m^2)
1e−5
1
0
ss (kg/m^2)
1e−3
5.0
2.5
0.0
du (kg/m^2)

next GEMS

Philipp Weiss, University of Oxford

Handling the waterfall of data

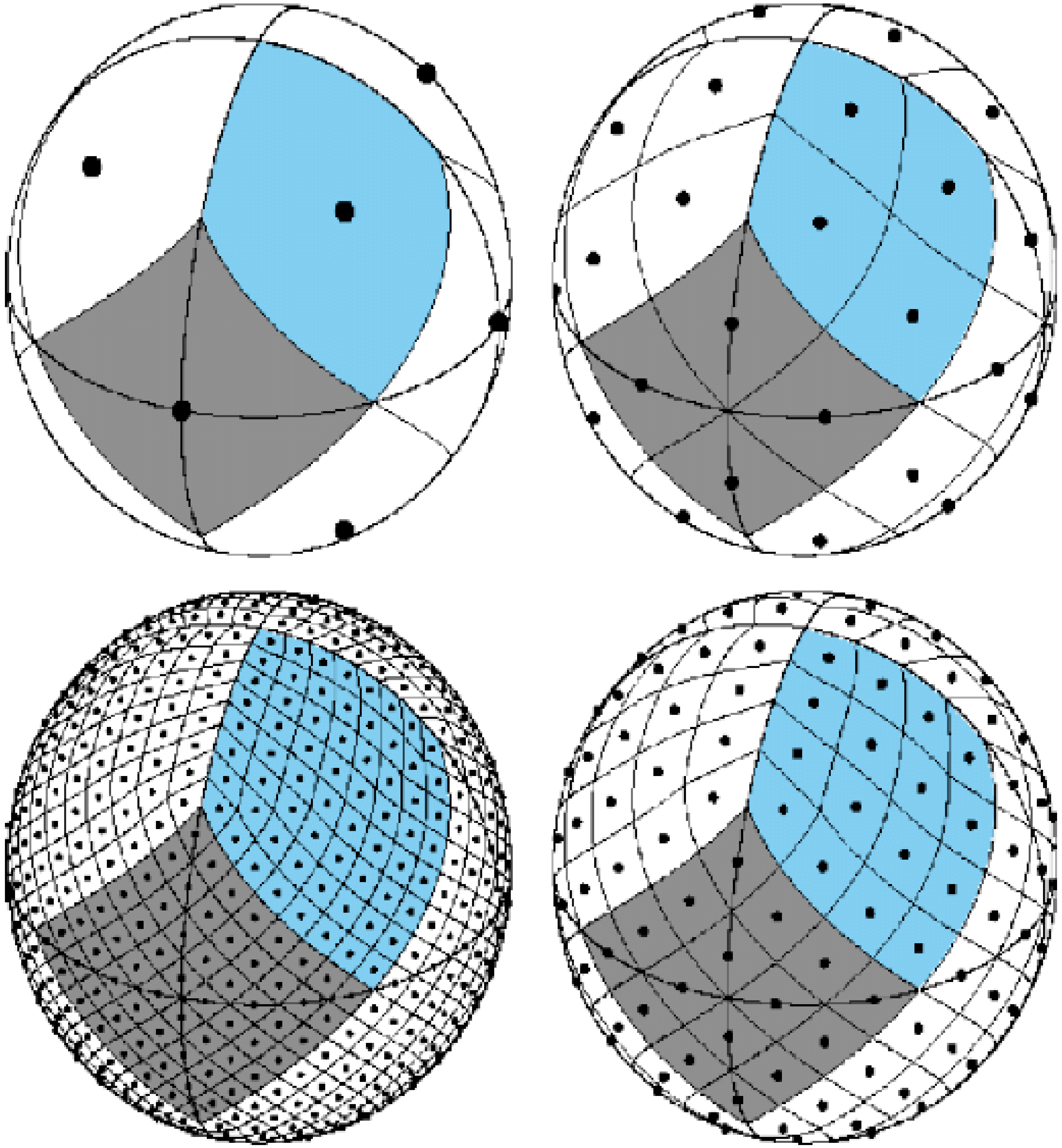
Good data is written once and read often



ICON → YAC coupled → hiopy → Zarr

Think about data sets: new approaches (for our community) enable the agile analysis of massive and diverse data.

HEALPix (**H**ierarchical **E**qual **A**rea iso**L**atitude **Pix**elation of a sphere)



Orders of magnitude in speed up in typical analysis… never a slow down

Technical challenges for km-scale earth system models - summary

- Everything needs attention around the model (grids, boundary conditions, input data, output, processing, etc.)
- Everything needs attention in the model, performance degrades very quickly, especially on GPUs
- Using GPUs efficiently is complicated and not helpful
- Components (weakly coupled) can be run on different parts of the machine

→ Tricky to set it up (software stack, slurm)

- Data sets need thought and need to be created in a usable way

→ Solutions exist and should be adopted by our community