Developing next-generation weather models in Python

Christian Kühnlein

Team: Till Ehrengruber, Stefano Ubbiali, Nicolai Krieger, Lukas Papritz, Sara Faghih-Naini, Gabriel Vollenweider, Heini Wernli

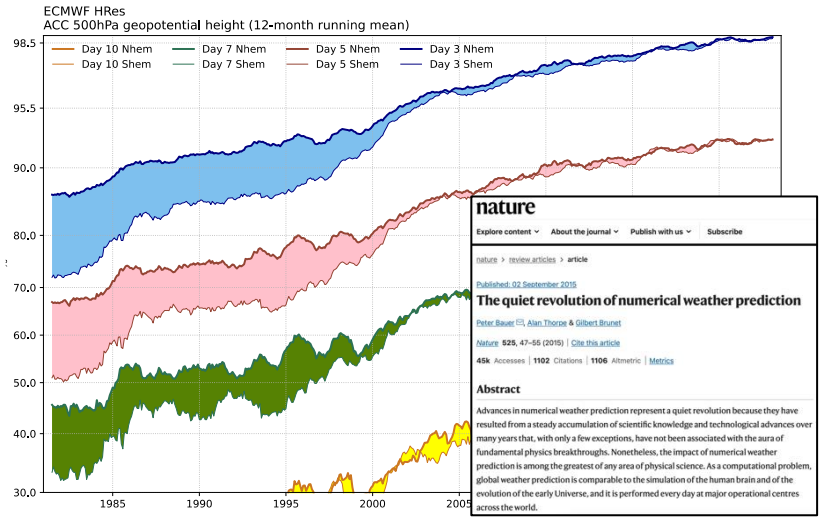
CSCS
Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

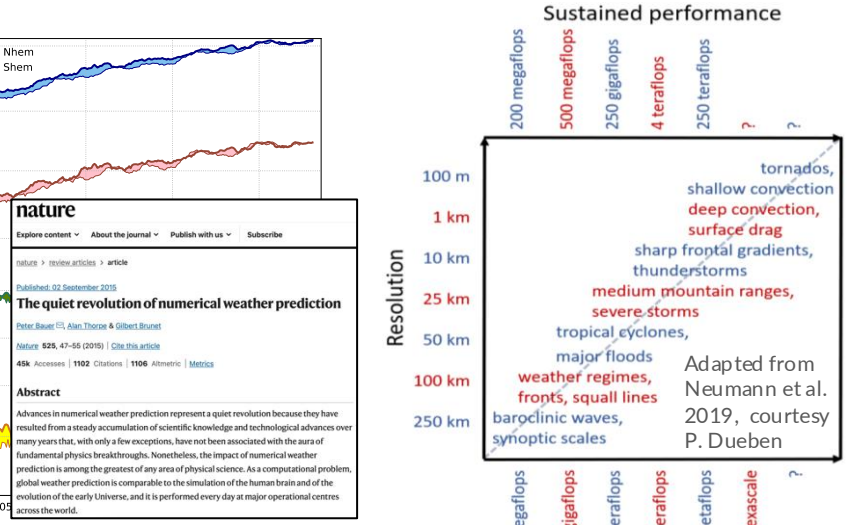
Funded by
the European Union

Destination Earth

implemented by

After decades of steady progress, we are seeing challenges and opportunities




nature

Explore content · About the journal · Publish with us · Subscribe

nature › review articles › article

Published: 02 September 2015

The quiet revolution of numerical weather prediction

Peter Bauer, Alan Thorpe & Gilbert Brunet

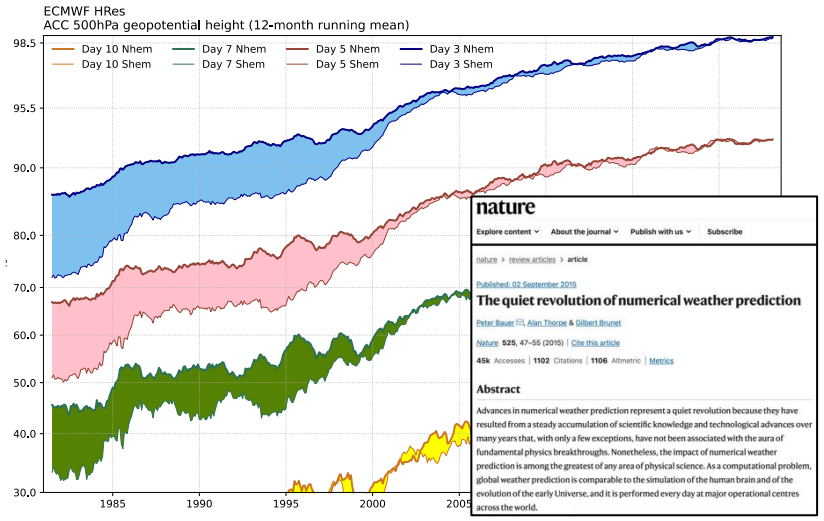
Nature 525, 47–55 (2015) | Cite this article

45k Accesses | 1102 Citations | 1106 Altmetric | Metrics

Abstract

Advances in numerical weather prediction represent a quiet revolution because they have resulted from a steady accumulation of scientific knowledge and technological advances over many years that, with only a few exceptions, have not been associated with the aura of fundamental physics breakthroughs. Nonetheless, the impact of numerical weather prediction is among the greatest of any area of physical science. As a computational problem, global weather prediction is comparable to the simulation of the human brain and of the evolution of the early Universe, and it is performed every day at major operational centres across the world.

After decades of steady progress, we are seeing challenges and opportunities



nature

Explore content · About the journal · Publish with us · Subscribe

nature > review articles > article

Published: 02 September 2015

The quiet revolution of numerical weather prediction

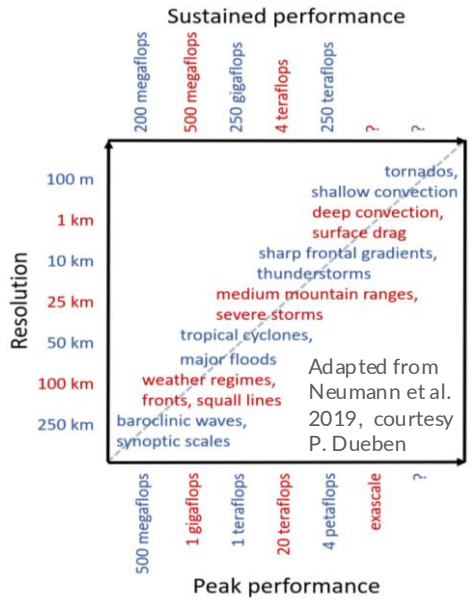
Peter Bauer, Alan Thorpe & Gilbert Brunet

Nature 525, 47–55 (2015) | Cite this article

45k Accesses | 1102 Citations | 1106 Altmetric | Metrics

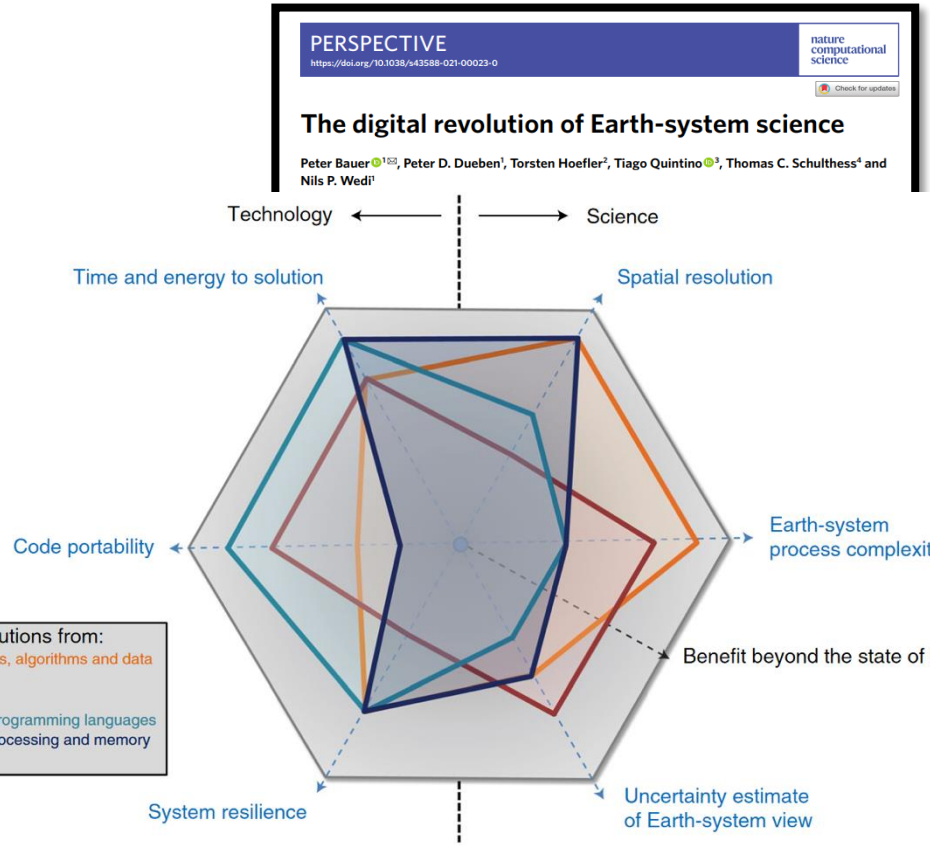
Abstract

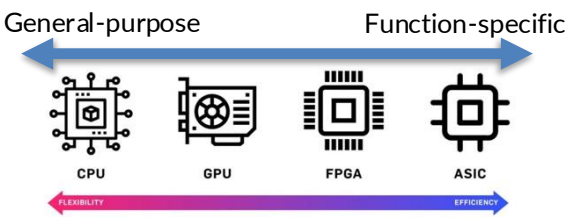
Advances in numerical weather prediction represent a quiet revolution because they have resulted from a steady accumulation of scientific knowledge and technological advances over many years that, with only a few exceptions, have not been associated with the aura of fundamental physics breakthroughs. Nonetheless, the impact of numerical weather prediction is among the greatest of any area of physical science. As a computational problem, global weather prediction is comparable to the simulation of the human brain and of the evolution of the early Universe, and it is performed every day at major operational centres across the world.



Individual contributions from:

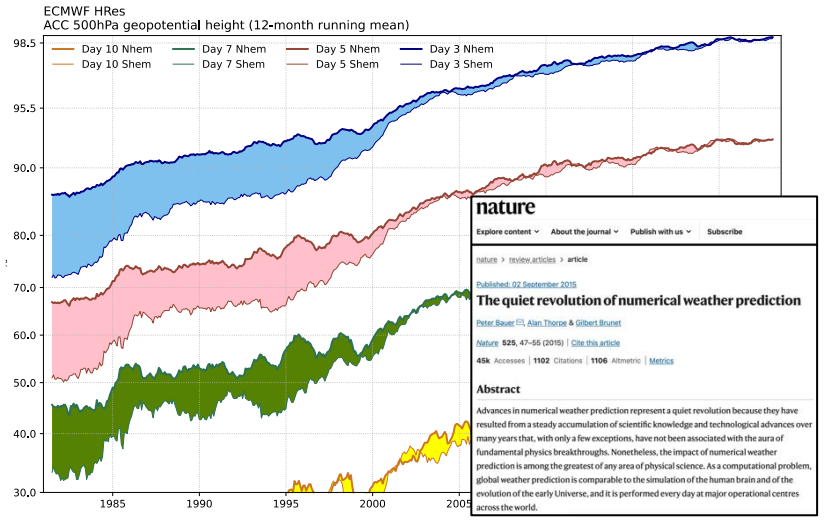
- Numerical methods, algorithms and data structures
- Machine learning
- Domain-specific programming languages
- Heterogeneous processing and memory architectures

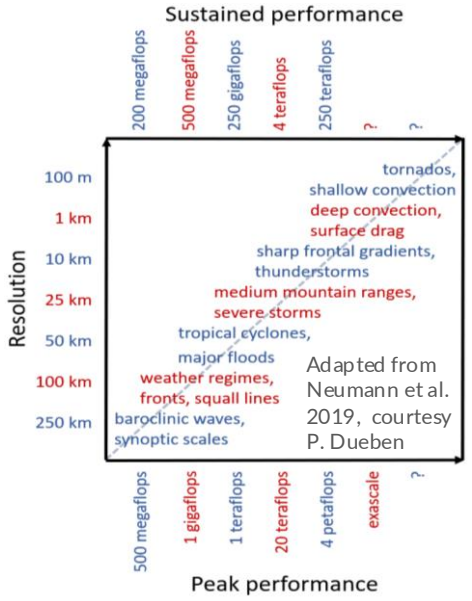


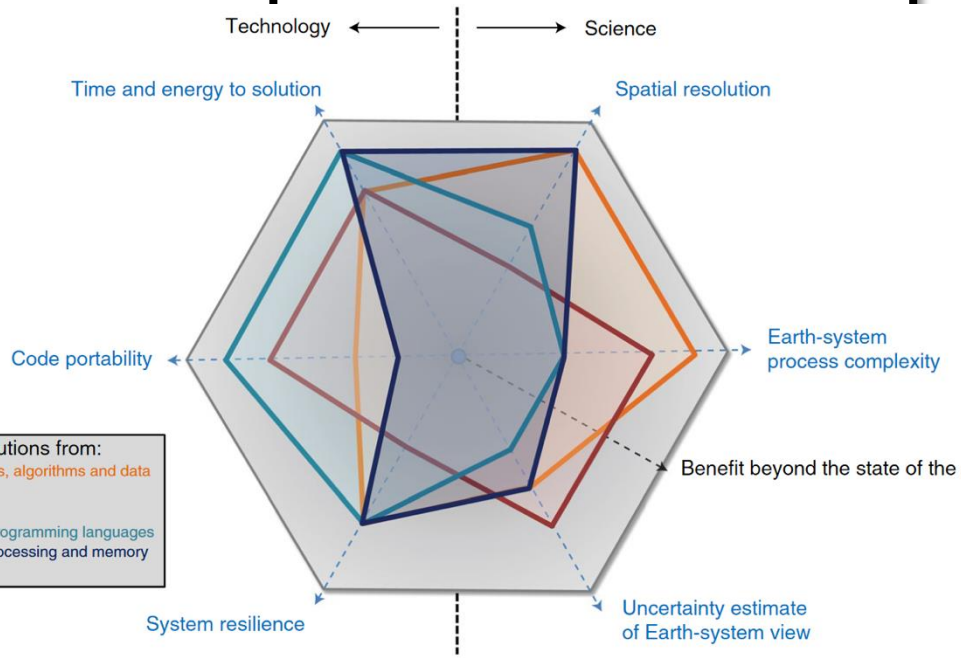


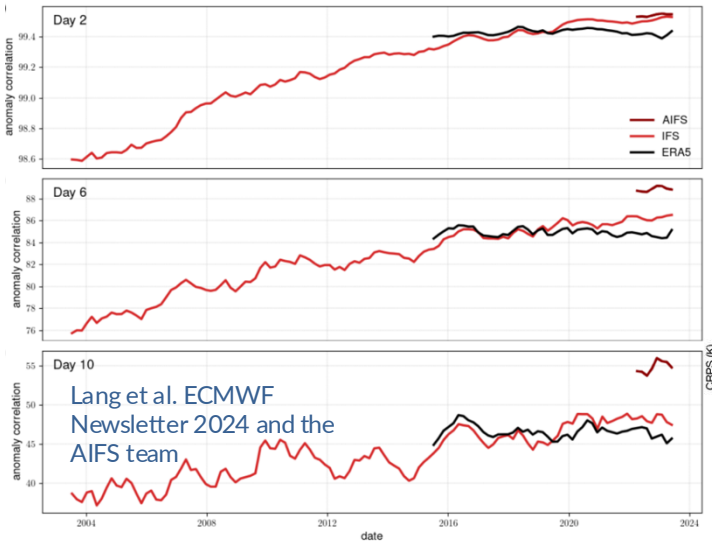
Source: venturebeat.com

After decades of steady progress, we are seeing challenges and opportunities

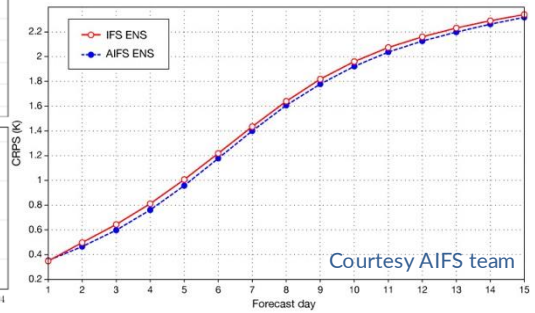






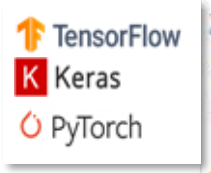


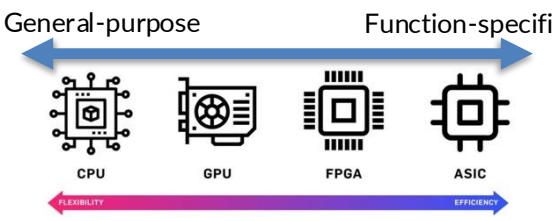
A transformation of weather forecasting is ongoing using various forms of data-driven approaches!



Individual contributions from:

- Numerical methods, algorithms and data structures
- Machine learning
- Domain-specific programming languages
- Heterogeneous processing and memory architectures





Source: venturebeat.com

Preparing the existing Integrated Forecasting System at ECMWF for GPUs

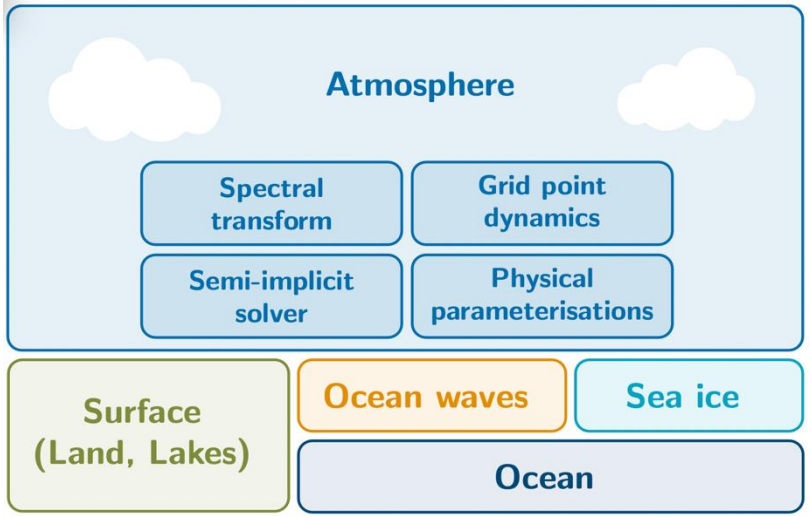
Standalone components

ecTrans
Spectral Transform

ecRad
Radiation model

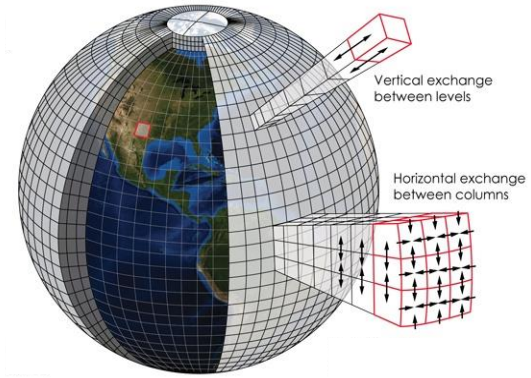
ecWAM
Wave model

ecLand
Land surface model



PSyclone NEMO

Source-to-source



Single-column recipes
Optimise compute kernels
Schedule data transfers

GPU-enabled data structures
Generalised data transfer API

Atlas | FIELD API | Loki S2S tool

The front-end remains in **Fortran** and hybrid GPU/CPU execution is enabled mostly by means of **directives/pragmas**.

Slide courtesy Michael Lange, ECMWF

EuroHPC supercomputers

LEONARDO CINECA

JÜLICH Forschungszentrum JUPITER

BSC Barcelona Supercomputing Center Centro Nacional de Supercomputación

CSC LUMI

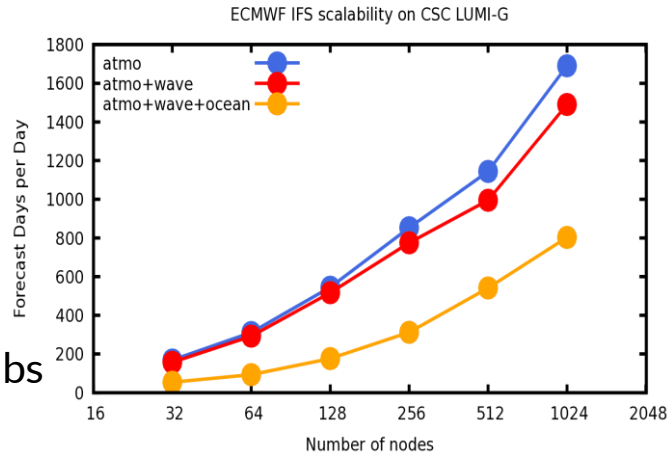
EuroHPC Joint Undertaking

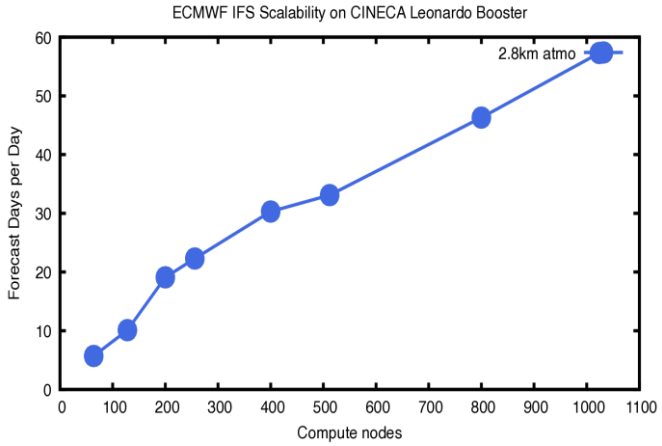
The Integrated Forecasting System in Destination Earth

Digital Twin Earth

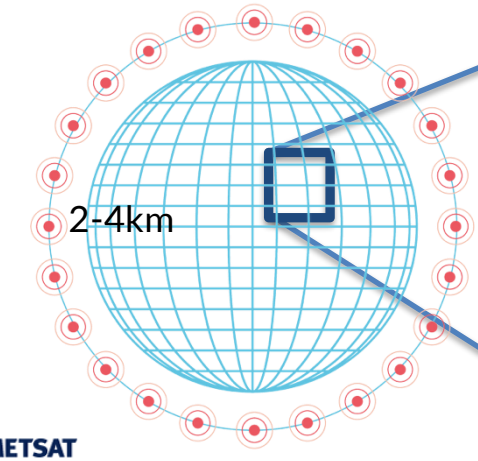
IFS 1.4 km

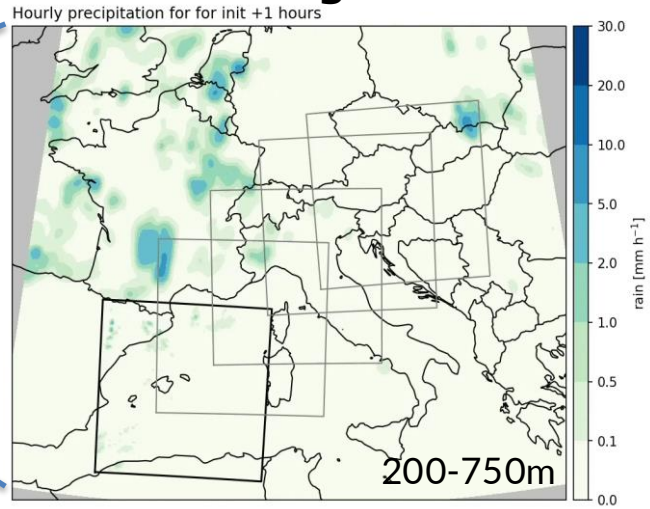
obs





Extremes Digital Twin





ECMWF member states using e.g. AROME

EuroHPC Joint Undertaking

Funded by the European Union | Destination Earth implemented by ECMWF esa EUMETSAT

ECMWF EUROPEAN CENTRE FOR MEDIUM-RANGE WEATHER FORECASTS

Slide courtesy Nils Wedi, ECMWF

The future of numerical models for weather and climate

Challenges with established operational models (specifically IFS and AROME, HARMONIE-AROME, ALARO):

- Fortran programming (limitations in compiler and library support, **modern coding paradigms**, narrow ecosystem)
- Resource-intensive to maintain, to develop, to use, and to adapt to **new hardware technologies**
- Lacks code interoperability and versatility particularly for new use cases such as **hybrid AI/ML**
- High barrier of entry for **next generation of scientists**, lower acceptance of legacy programming approaches among software engineers, smaller available talent pool
- Numerical limitations particularly for **high resolution**, e.g.:
  - Steep orography stability and representation
  - (Local) conservation
  - Flexibility of time stepping and physics coupling
  - Large-eddy simulation capabilities

Portable Model for multi-scale Atmospheric Prediction (PMAP)

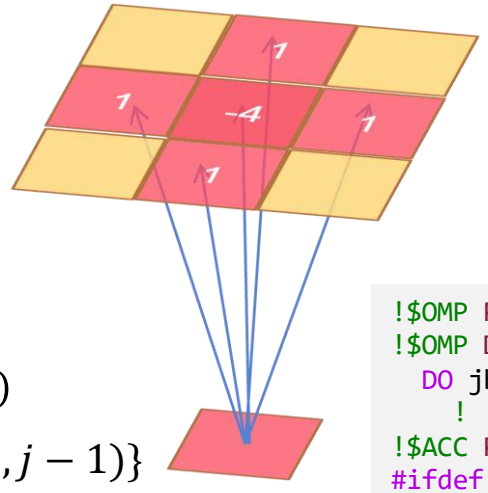
Building on numerical principles of ECMWF's **semi-implicit finite-volume** dynamical core **FVM** in Fortran (Smolarkiewicz et al. 2014, 2017; Kühnlein et al. 2019) and selected ESM components of the IFS, key goals for the **PMAP** are:

- **High-level programming** forecast model implementation completely in Python
- **Performance-portability and computing hardware adaptation** by means GridTools for Python (GT4Py)
- Numerical methods suitability, physical fidelity, and high efficiency from **kilometer to meter scales**
- Support both **regional and global domains** using quadrilateral grids, and (one- and two-way) nesting functionalities
- Serve as **operational weather prediction** model and as **flexible and user-friendly research tool**
- Deploy model across largest CPU/GPU supercomputers, cloud services and laptops
- Towards **auto-differentiation** of full model, for hybrid data-driven modelling, seamless ML, and data assimilation

Future model development and coding practices

Mathematical Formulation

$$\Psi = \Delta_{x,y}\phi$$



Discretization

$$\Psi(i,j) = \frac{1}{h^2}\{-4 * \phi(i,j) + \phi(i+1,j) + \phi(i-1,j) + \phi(i,j+1) + \phi(i,j-1)\}$$

Prototype Implementation

```python
import numpy as np

def laplacian(inp: np.ndarray):
    return (
        -4.0 * inp[1:-1, 1:-1]
        + inp[2:, 1:-1] + inp[:-2, 1:-1]
        + inp[1:-1, 2:] + inp[1:-1, :-2]
    )
```

HPC Implementation

```
!$OMP PARALLEL
!$OMP DO PRIVATE(jb,i_startidx,i_endidx,je,jk)
  DO jb = i_startblk, i_endblk
    ! ...
!$ACC PARALLEL IF( i_am_accel_node .AND. acc_on )
#ifdef __LOOP_EXCHANGE
   !$ACC LOOP GANG
    DO je = i_startidx, i_endidx
     !$ACC LOOP VECTOR
      DO jk = slev, elev
#else
!CDIR UNROLL=3
   !$ACC LOOP GANG
    DO jk = slev, elev
      !$ACC LOOP VECTOR
      DO je = i_startidx, i_endidx
#endif
         ! here goes the laplacian
      END DO
    END DO
!$ACC END PARALLEL
  END DO
!$OMP END DO NOWAIT
!$OMP END PARALLEL
```

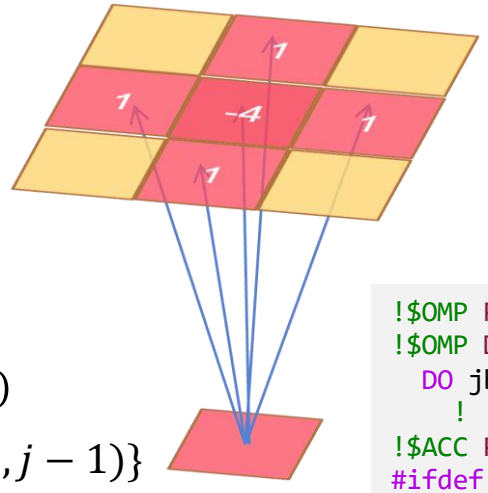
Inspired by slides from O. Fuhrer (MeteoSwiss)

Slide courtesy Hannes Vogt (CSCS)

Future model development and coding practices

Mathematical Formulation

$$\Psi = \Delta_{x,y}\phi$$



Discretization

$$\Psi(i,j) = \frac{1}{h^2}\{-4 * \phi(i,j) + \phi(i+1,j) + \phi(i-1,j) + \phi(i,j+1) + \phi(i,j-1)\}$$

Prototype Implementation

```
@gtx.field_operator
def lap(inp: gtx.Field[gtx.Dims[IDim, JDim, KDim], float]):
    return -4.0 * inp + inp(I-1) + inp(I+1) + inp(J-1) + inp(J+1)
```

HPC Implementation

```
!$OMP PARALLEL
!$OMP DO PRIVATE(jb,i_startidx,i_endidx,je,jk)
  DO jb = i_startblk, i_endblk
    ! ...
!$ACC PARALLEL IF( i_am_accel_node .AND. acc_on )
#ifdef __LOOP_EXCHANGE
   !$ACC LOOP GANG
    DO je = i_startidx, i_endidx
     !$ACC LOOP VECTOR
      DO jk = slev, elev
#else
!CDIR UNROLL=3
   !$ACC LOOP GANG
    DO jk = slev, elev
      !$ACC LOOP VECTOR
      DO je = i_startidx, i_endidx
#endif
         ! here goes the laplacian
      END DO
    END DO
!$ACC END PARALLEL
  END DO
!$OMP END DO NOWAIT
!$OMP END PARALLEL
```

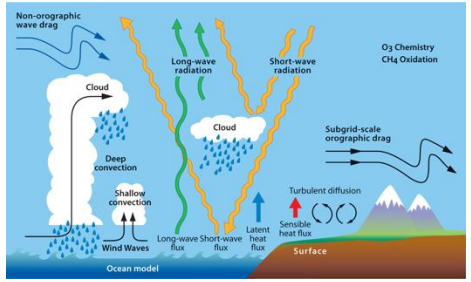
Inspired by slides from O. Fuhrer (MeteoSwiss)

Slide courtesy Hannes Vogt (CSCS)

High-level domain-specific programming with GridTools for Python

PMAP domain model

$$\frac{\partial \mathcal{G}\rho}{\partial t} + \nabla \cdot (\mathbf{v}\mathcal{G}\rho) = 0 \ ,$$
$$\frac{\partial \mathcal{G}\rho\mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{v}\mathcal{G}\rho\mathbf{u}) = \mathcal{G}\rho\left[-c_p\theta_\rho \underline{\mathbf{G}}\nabla\pi + \mathbf{g} - \mathbf{f}\times\mathbf{u} + \boldsymbol{M}^{\mathbf{u}} + \boldsymbol{B}^{\mathbf{u}} + \boldsymbol{D}^{\mathbf{u}} + \boldsymbol{P}^{\mathbf{u}}\right]$$
$$\frac{\partial \mathcal{G}\rho\theta'}{\partial t} + \nabla \cdot (\mathbf{v}\mathcal{G}\rho\,\theta') = \mathcal{G}\rho\left[-\underline{\mathbf{G}}^T\mathbf{u}\cdot\nabla\theta_a + B^\theta + D^\theta + P^\theta\right] \ ,$$
$$\frac{\partial \mathcal{G}\rho\psi}{\partial t} + \nabla \cdot (\mathbf{v}\mathcal{G}\rho\psi) = \mathcal{G}\rho\left[B^\psi + D^\psi + P^\psi\right] \ , \qquad \psi = r_v\,, r_l\,, r_r\,, r_i\,, r_s\,, r_g\,, \Lambda_f\,, \ldots$$
$$\pi = \left(\frac{R}{p_0}\rho\,\theta\,(1 + r_v/\varepsilon)\right)^{R/c_v}$$

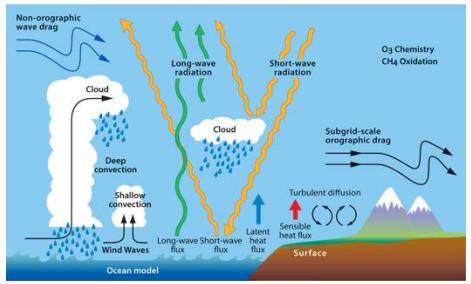


Performance engineering → HPC execution

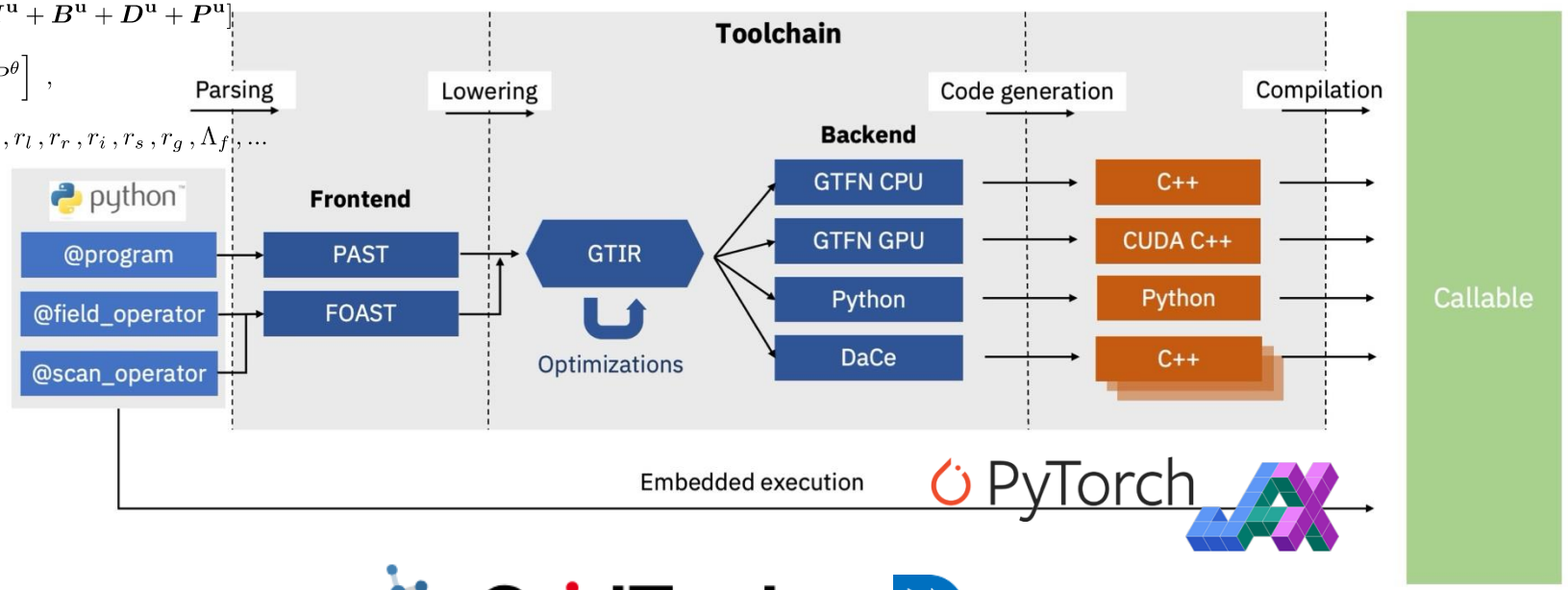
High-level domain-specific programming with GridTools for Python

PMAP domain model

$$\frac{\partial \mathcal{G}\rho}{\partial t} + \nabla \cdot (\mathbf{v}\mathcal{G}\rho) = 0 \ ,$$
$$\frac{\partial \mathcal{G}\rho\mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{v}\mathcal{G}\rho\mathbf{u}) = \mathcal{G}\rho\left[-c_p\theta_\rho \underline{\mathbf{G}}\nabla\pi + \mathbf{g} - \mathbf{f}\times\mathbf{u} + \boldsymbol{M}^{\mathbf{u}} + \boldsymbol{B}^{\mathbf{u}} + \boldsymbol{D}^{\mathbf{u}} + \boldsymbol{P}^{\mathbf{u}}\right]$$
$$\frac{\partial \mathcal{G}\rho\theta'}{\partial t} + \nabla \cdot (\mathbf{v}\mathcal{G}\rho\,\theta') = \mathcal{G}\rho\left[-\underline{\mathbf{G}}^T\mathbf{u}\cdot\nabla\theta_a + B^\theta + D^\theta + P^\theta\right] \ ,$$
$$\frac{\partial \mathcal{G}\rho\psi}{\partial t} + \nabla \cdot (\mathbf{v}\mathcal{G}\rho\psi) = \mathcal{G}\rho\left[B^\psi + D^\psi + P^\psi\right] \ , \qquad \psi = r_v\,, r_l\,, r_r\,, r_i\,, r_s\,, r_g\,, \Lambda_f, \ldots$$
$$\pi = \left(\frac{R}{p_0}\rho\,\theta\,(1 + r_v/\varepsilon)\right)^{R/c_v}$$



Performance engineering → HPC execution



GridTools 4python

DaCe

https://github.com/GridTools/gt4py

Domain-specific GT4Py framework in Python

GridTools 4python

- **GT4Py** is embedded in the **Python** eco-system
  - Versatile, portable, productive programming language
  - Broad selection of modules/libraries
  - Enables new user and developer workflows
  - Low barrier of entry for domain scientists and academia
  - Favourable choice with respect to ML/AI applications

GridTools/**atlas4py**

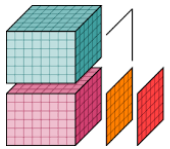
spcl/**dace**

DaCe - Data Centric Parallel Programming

69 Contributors · 25 Used by · 15 Discussions · 497 Stars · 129 Forks

5 Contributors · 1 Issue · 1 Star · 3 Forks

earthkit



xarray

pandas

ecmwf/**pyflow**

A high level Python interface to ecFlow allowing the creation of ecFlow suites in a modular and "pythonic" way

9 Contributors · 3 Used by · 8 Stars · 7 Forks

pyFFTW/**pyFFTW**

A pythonic python wrapper around FFTW

28 Contributors · 938 Used by · 377 Stars · 108 Forks

ECMWF ETH zürich CSCS Centro Svizzero di Calcolo Scientifico Swiss National Supercomputing Centre

GT4Py domain-specific library

GridTools 4python

- **Besides IFS/PMAP, two other major atmospheric models are rewritten/developed in Python with GT4Py**

  - **ICON4Py is** the GT4Py.next implementation of the ICON model in the EXCLAIM project at ETH Zurich

    https://github.com/C2SM/icon4py

    EXCLAIM

    Schweizerische Eidgenossenschaft
    Confédération suisse
    Confederazione Svizzera
    Confederaziun svizra

    Swiss Confederation

    Federal Department of Home Affairs FDHA
    Federal Office of Meteorology and Climatology
    MeteoSwiss

    MeteoSwiss

  - **Pace** (Ben-Nun et al. 2022; Dahm et al. 2023) is the GT4Py.cartesian implementation of the FV3GFS/SHiELD model of GFDL and NOAA

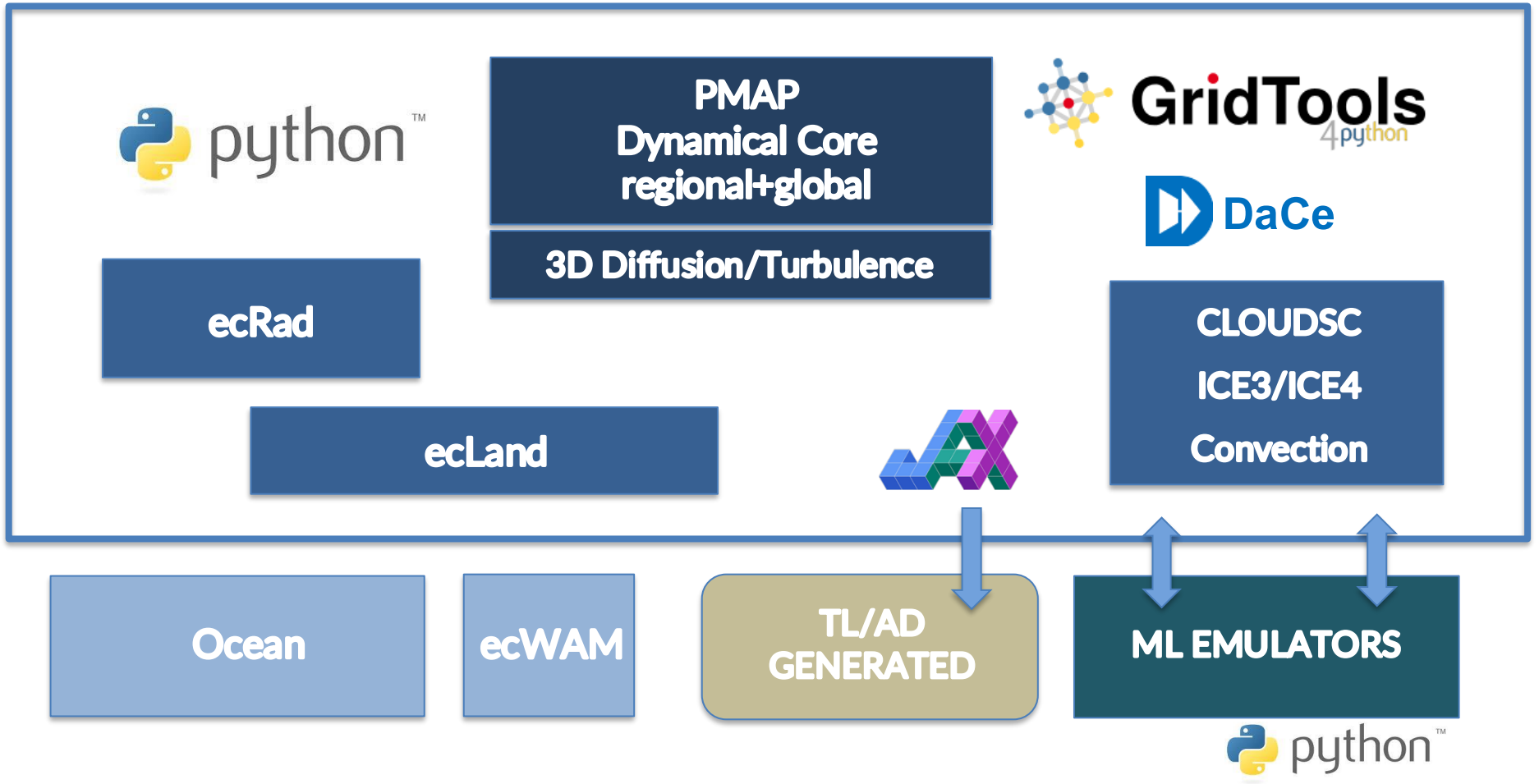
    https://github.com/NOAA-GFDL/pace

    Ai2

    NOAA

    NASA

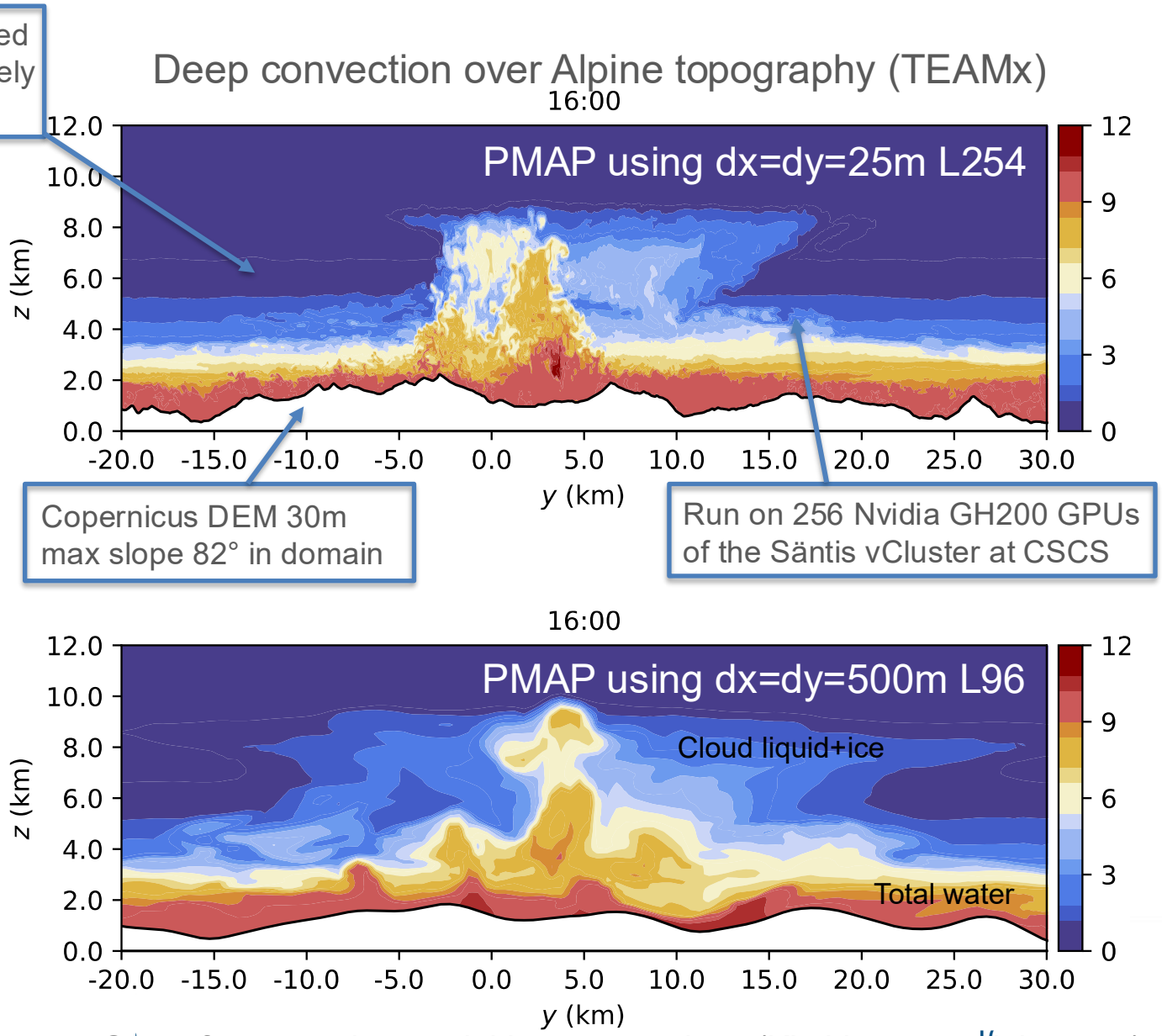
Envisaged PMAP forecast system with its components and codes



ECMWF ETH zürich CSCS Centro Svizzero di Calcolo Scientifico Swiss National Supercomputing Centre

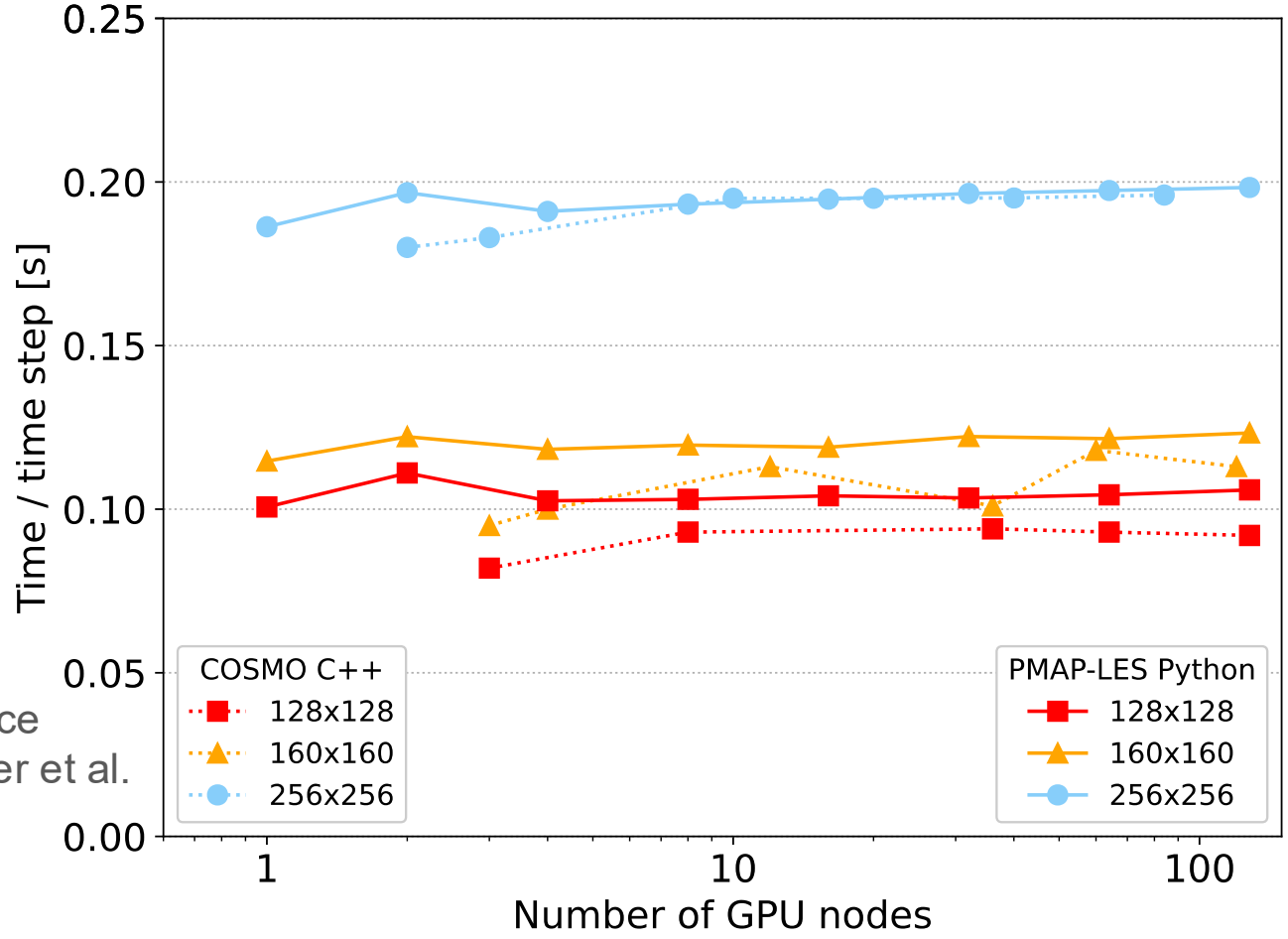
PMAP offers advanced high-resolution simulation capabilities

- Fully compressible equations in a strong conservation form accounting for comprehensive moist processes formulated in terms of dry density, dry entropy perturbation, Exner pressure perturbation, dry density and moist/hydrometeor mixing ratios
- Generalized terrain-following curvilinear coordinate based on height
- Numerical integration based on **3D implicit time integration** for core dynamics combined with **non-oscillatory flux-form** (i.e., locally conservative) semi-Lagrangian advective transport with adaptive time stepping
- Large-eddy simulation capabilities using 3D flux-form diffusion and scale-aware prognostic subgrid-scale turbulence kinetic energy (TKE) scheme



Setup from ongoing model intercomparison (Kirshbaum et al. in prep.)

Is the PMAP Python approach viable in terms of computational performance? Yes.



COSMO performance numbers from Fuhrer et al. GMD 2018

PMAP (Python DSL) vs COSMO (C++ DSL) showed same performance

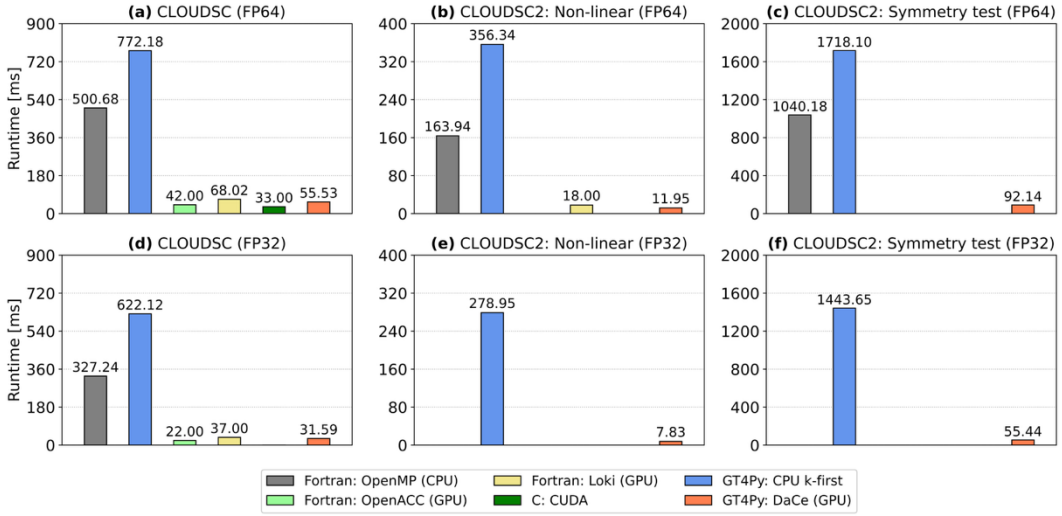
Notes:
COSMO was
- operational (production-level),
- GPU-optimized for Piz Daint
- faster than its Fortran version

This comparison was done in 2024.

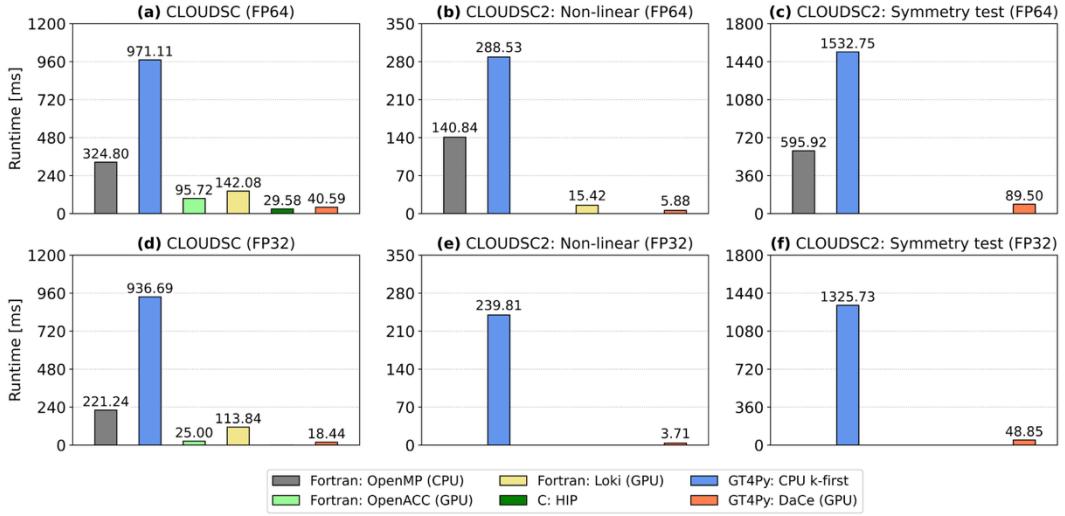
PMAP (solid lines) with the production-level COSMO model of MeteoSwiss based on the STELLA DSL in C++ (dashed lines), on the identical Nvidia GPU hardware of Piz Daint (CSCS). → Details discussed in PMAP paper in preparation for GMD

Exploring GT4Py for the NWP domain using ECMWF microphysics schemes

Piz Daint, Nvidia P100, CSCS, Switzerland



LUMI-G, AMD MI 250X CSC, Finland



FP64

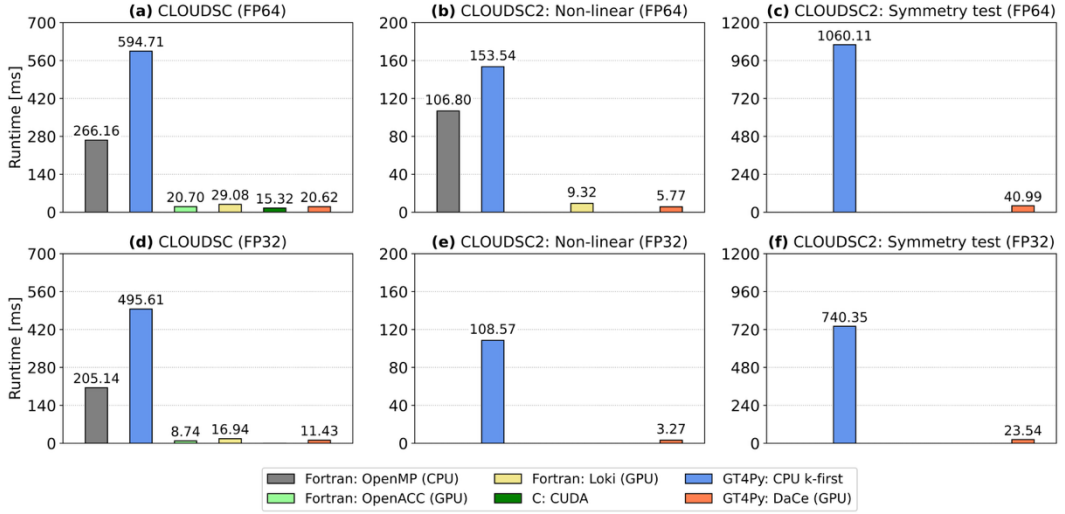
FP32

CLOUDSC

simplified nonlinear

TL/AD symmetry test

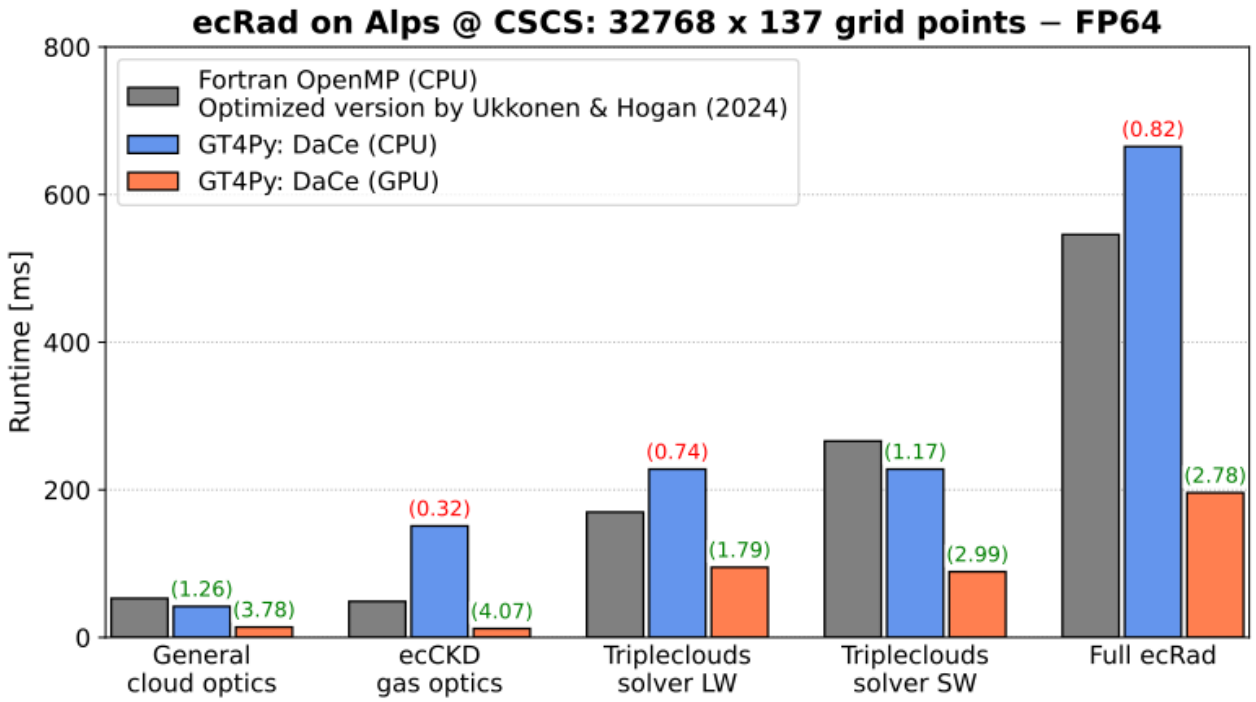
MeluXina, Nvidia A100, LuxConnect, Luxemburg



- Performance testing Python implementations of CLOUDSC, simplified nonlinear CLOUDSC2, tangent-linear CLOUDSC2, and adjoint CLOUDSC2 with GT4Py
- GPU vs CPUs, 64-bit vs 32-bit precision, various GT4Py backends
- Ubbiali et al. 2025 https://gmd.copernicus.org/articles/18/529/2025/)

Complete Python-GT4Py version of the ecRad radiation scheme

- **PMAP 3D flux-form prognostic turbulence kinetic energy** (TKE) scheme
- **CLOUDSC** various versions including CY49R1 ported and coupled. NL, TL, AD of CLOUDSC2 in GT4Py.cartesian (Ubbiali et al. 2025) and recent GT4Py.next implementation by H. Vogt (CSCS) with embedded JAX execution to auto-generate TL, AD
- **ecRad** ported and coupling almost complete (G. Vollenweider and S. Ubbiali)
- **ecLand** Python version to be completed in 2026 including coupling to PMAP
- Coupling of PMAP to **ai-land** (E. Pinnington, N. Raoult) in Python will also be done soon
- **Physics-dynamics coupling in PMAP** will enable flexibility and various coupling strategies supported by the high-level Python programming



ecRad in Python with GT4Py executes on Nvidia (CUDA C++) GPUs and AMD (HIP C++, not shown) as well as on CPUs (fig. S. Ubbiali)

EXTREMES DT : A MAGNIFYING GLASS on EXTREME WEATHER EVENTS

Global and **daily** simulations of extreme weather
4 days ahead at **4.4km**

Regional simulations
2 days ahead at **750m** to **500m**

Impact-sector models:
user-relevant information for societal impacts

IFS-NEMO

ECMWF

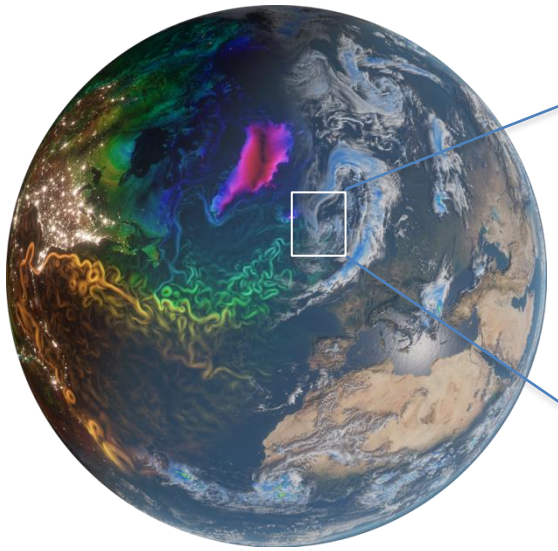
Arome
Harmonie-Arome
Alaro

METEO FRANCE
À VOS CÔTÉS, DANS UN CLIMAT QUI CHANGE

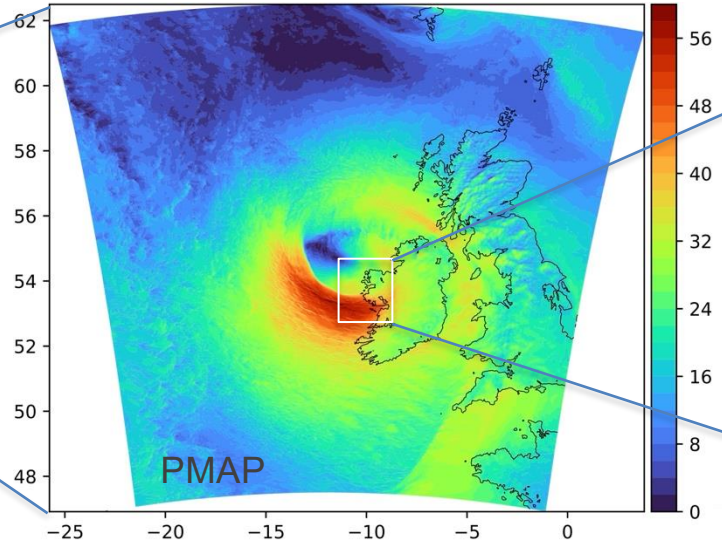
Norwegian Meteorological Institute · METEO FRANCE · Danmarks Meteorologiske Institut · AEMet · FMI · SMHI · LVGMC · IPMA · RMI · DHMZ · CHMI · TAL TECH · Republic of Estonia Environment Agency · ZAMG · KNMI · IMGW · Icelandic Met Office · ARSO Meteo Slovenia · INRAE · CSC · Lietuvos hidrometeorologijos tarnyba · RIVM · BSC Barcelona Supercomputing Center · SHMU · Országos Meteorológiai Szolgálat · GeoSphere Austria

Weather and climate across scales: flexible configurations driven by Destination Earth Digital Twins

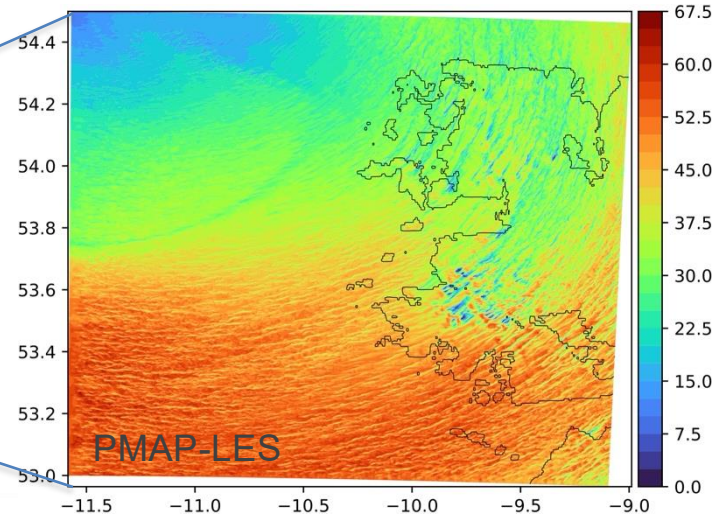
DestinE Digital Twins



Km- and Sub-km-scale PMAP driven by global DTs



Hecto- and decametre-scale PMAP-LES



Horizontal wind (m/s) in ABL

Setup and validation of **PMAP at kilometre, subkilometre and decametre resolutions** on EuroHPC GPU systems

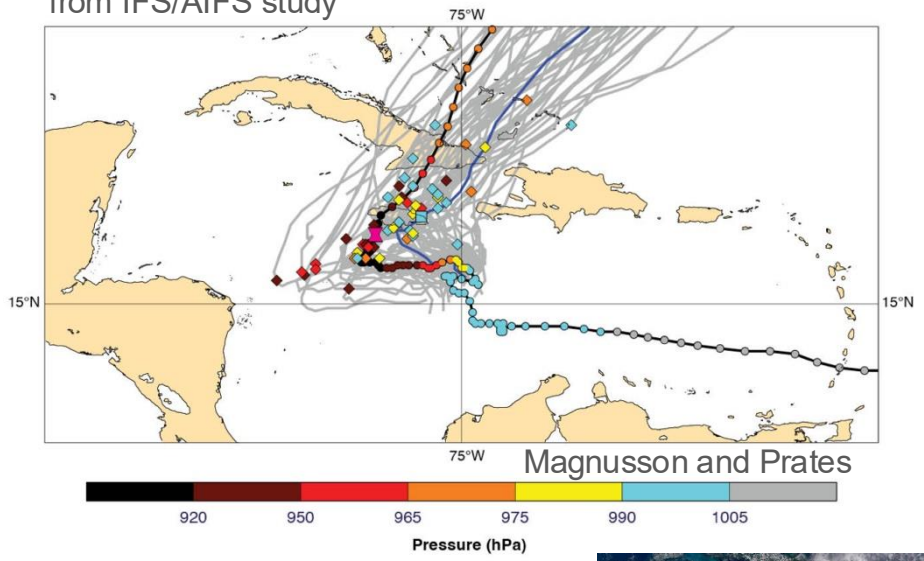
Funded by the European Union

Destination Earth implemented by ECMWF esa EUMETSAT

ECMWF ETH zürich CSCS Centro Svizzero di Calcolo Scientifico Swiss National Supercomputing Centre

Developing PMAP real weather prediction capabilities across scales

Forecast uncertainty in track and intensity close to landfall, from IFS/AIFS study

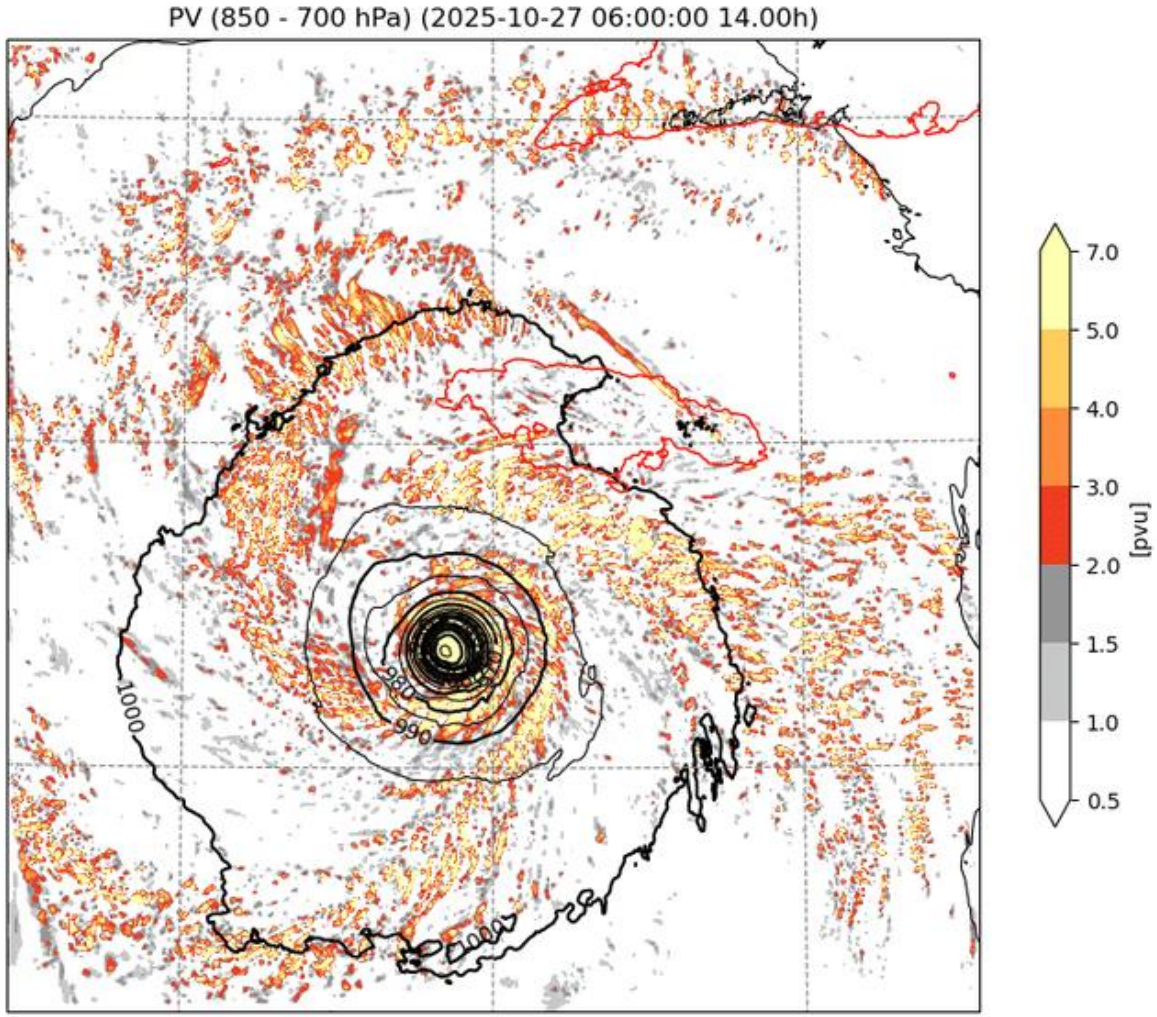


Hurricane Melissa

Category 5, October 2025

PMAP experiments at 500m grid spacing with comprehensive physical parametrizations

Validation and discussion in Papritz et al. EGU 2026

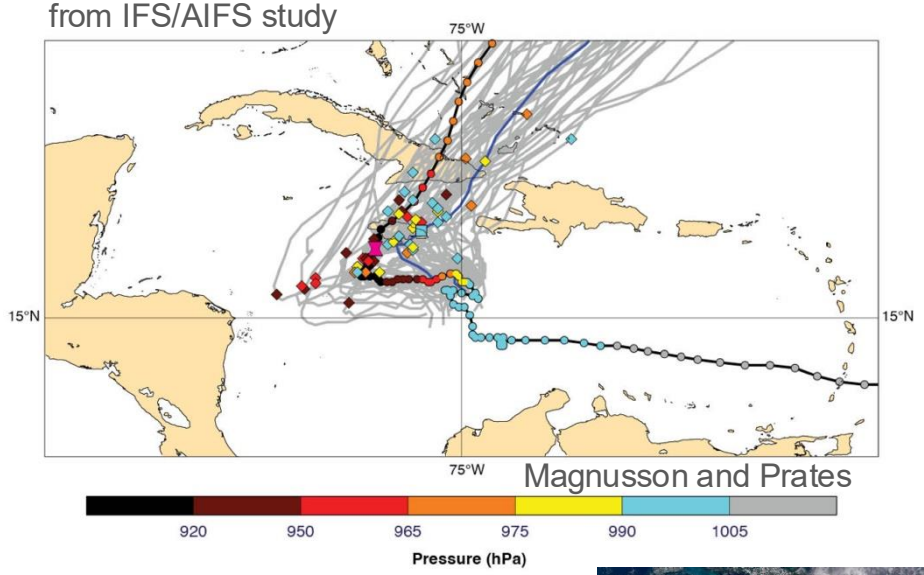


PMAP with full physics
(land-surface and ocean still non-interactive)

ECMWF ETH zürich CSCS Centro Svizzero di Calcolo Scientifico Swiss National Supercomputing Centre

Developing PMAP real weather prediction capabilities across scales

Forecast uncertainty in track and intensity close to landfall, from IFS/AIFS study

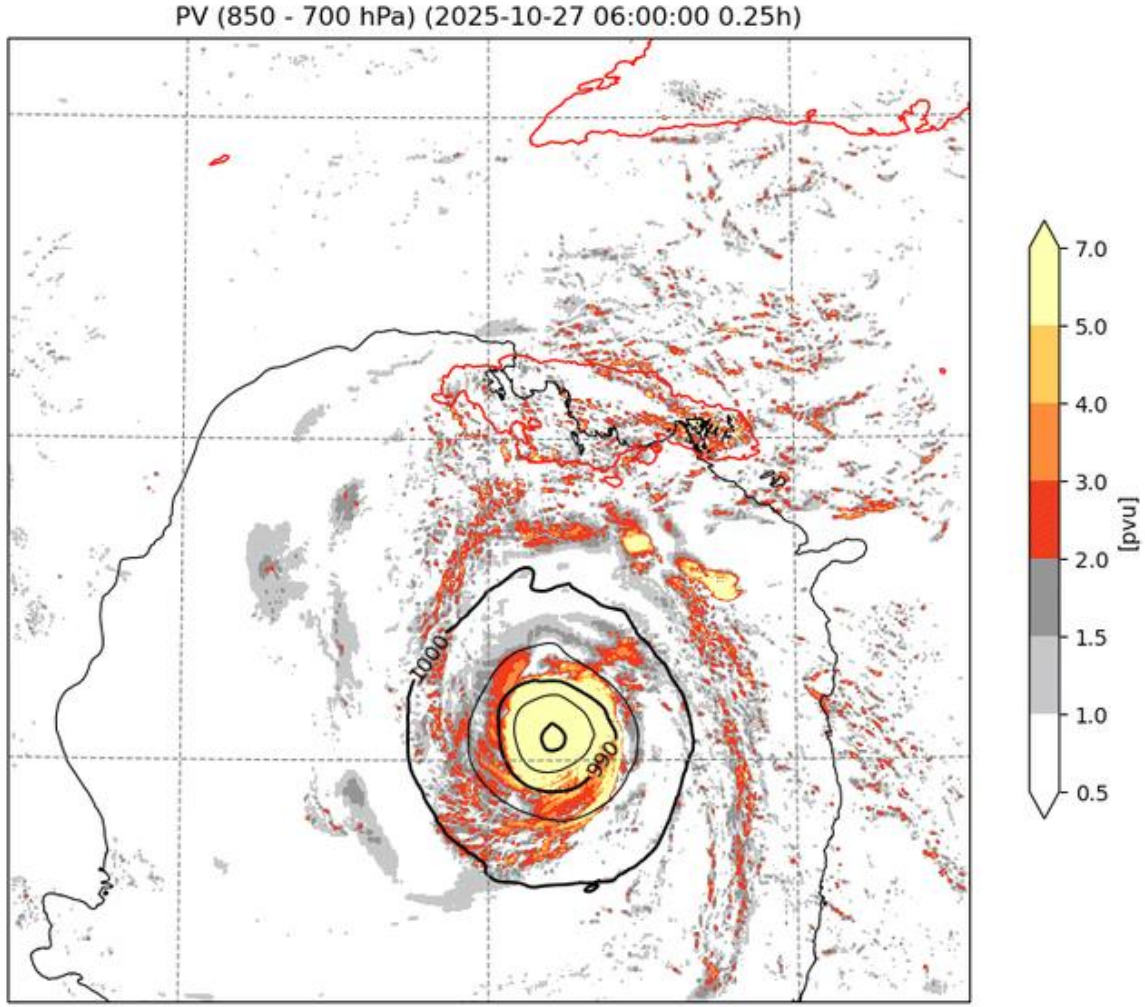


Hurricane Melissa

Category 5, October 2025

PMAP experiments at 500m grid spacing with comprehensive physical parametrizations

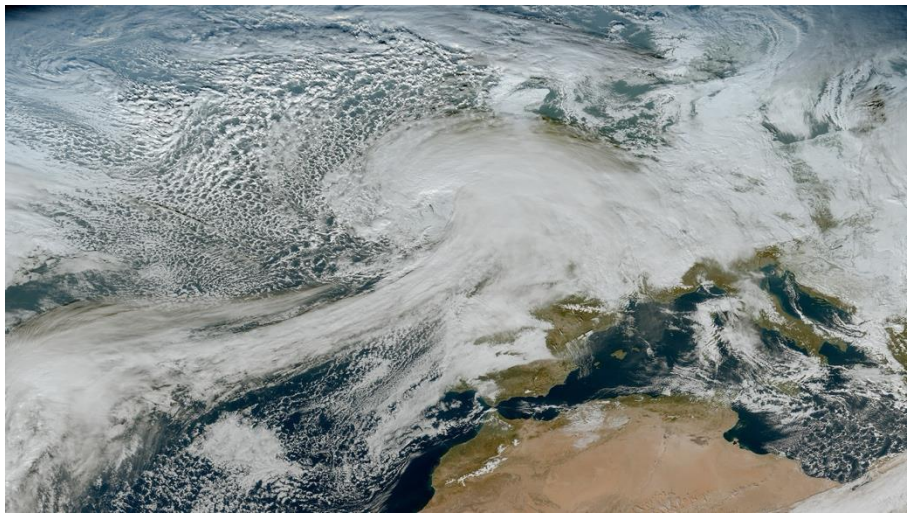
Validation and discussion in Papritz et al. EGU 2026

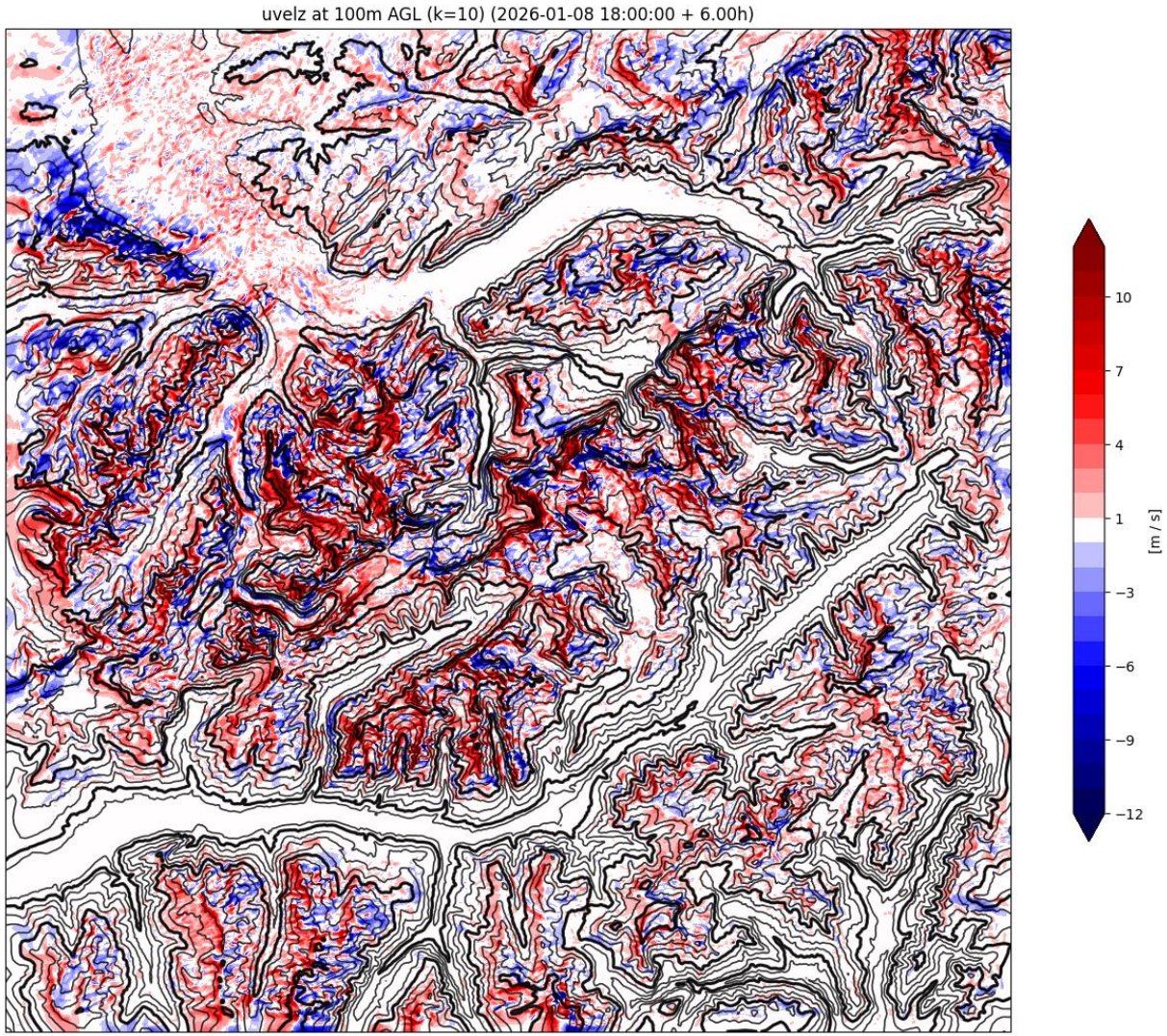


PMAP with full physics
(land-surface and ocean still non-interactive)

ECMWF ETH zürich CSCS Centro Svizzero di Calcolo Scientifico Swiss National Supercomputing Centre

Developing PMAP real weather prediction capabilities across scales





Storm Elli passing Swiss Alps

January 2026

PMAP experiments at <=50m grid spacing with comprehensive physical parametrizations

Max topography slope 82 deg

Validation and discussion in Papritz et al. EGU 2026

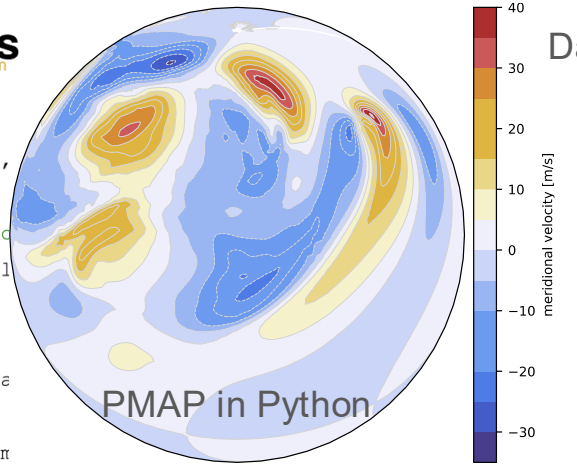
ECMWF ETH zürich CSCS Centro Svizzero di Calcolo Scientifico Swiss National Supercomputing Centre

PMAP global dynamical core on the IFS octahedral grid

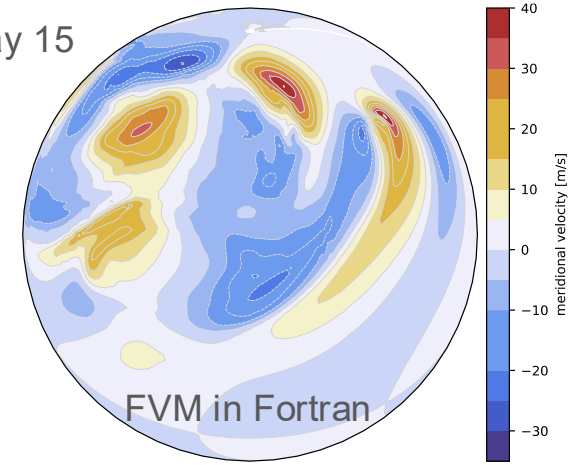
```python
@field_operator
def advection_scheme_upwind(
    rho: Field[[Vertex], float],
    dt: float,
    vel: tuple[Field[[Vertex], float], Field[[Vertex],
    vol: Field[[Vertex], float],
    dual_face_orientation: Field[[Vertex, V2EDim], flo
    dual_face_normal: tuple[Field[[Edge], float], Fiel
    dual_face_length: Field[[Edge], float]
) -> Field[[Vertex], float]:
    flux = upwind_flux(rho, vel, dual_face_normal, dua
    return rho - (dt / vol) * neighbor_sum(
        flux(V2E) * dual_face_orientation, axis=V2EDim
```

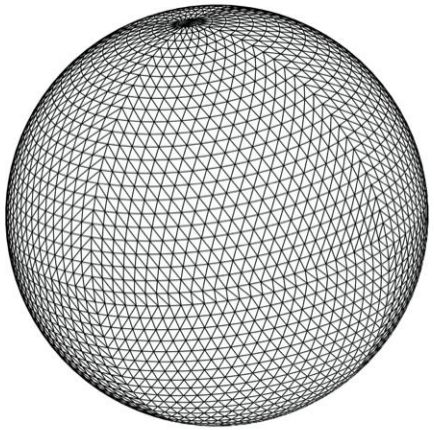
GridTools 4python

Baroclinic wave benchmark

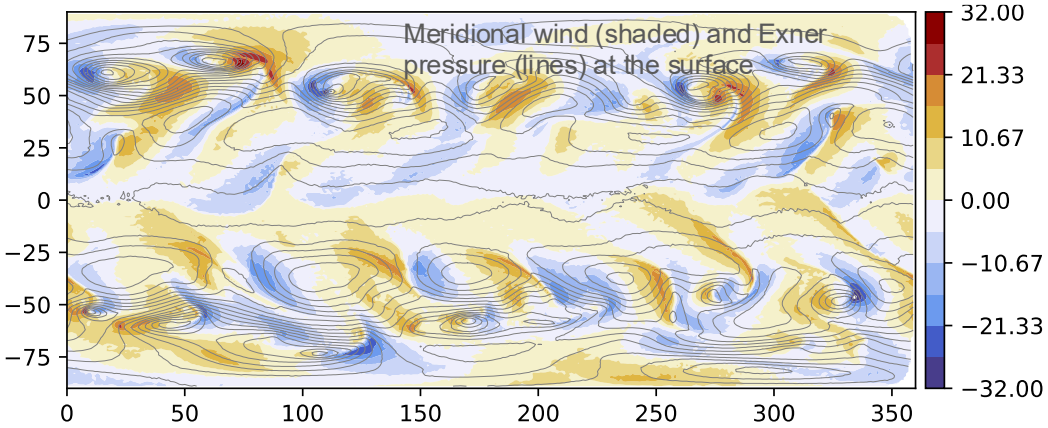


Day 15

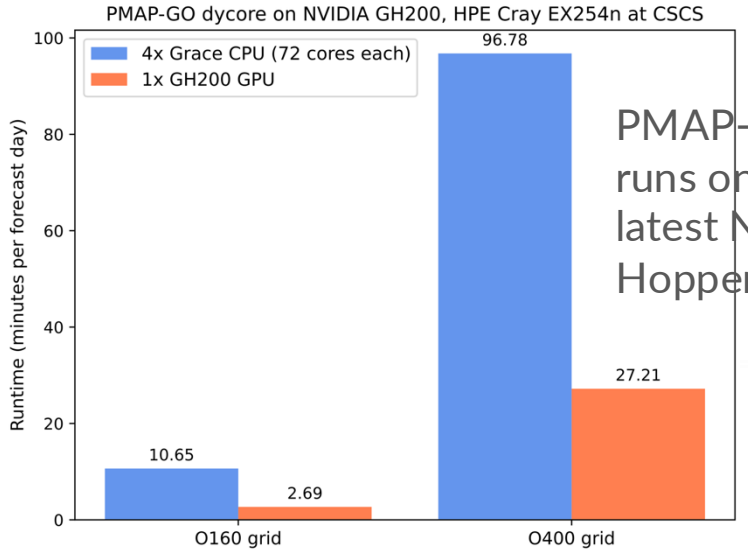




```python
from atlas4py import StructuredGrid
grid = StructuredGrid("O24")
mesh = AtlasMesh.generate(grid)
```



Held-Suarez climate benchmark with PMAP
(O360/L91 in FP32 fits on a single Nvidia GH200 GPU)



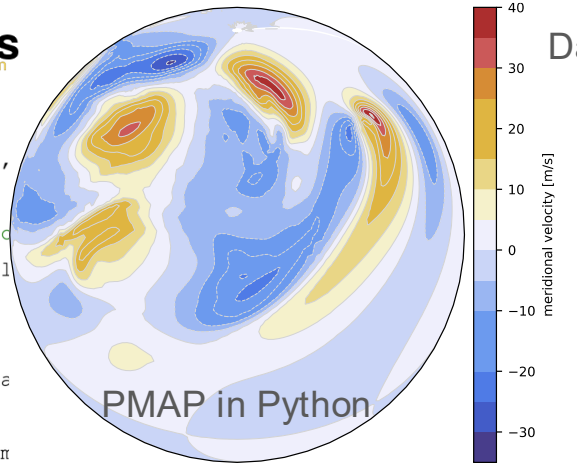
PMAP-GO completely runs on GPU including the latest Nvidia GH200 Grace Hopper Superchip at CSCS

ECMWF ETH zürich CSCS Centro Svizzero di Calcolo Scientifico Swiss National Supercomputing Centre

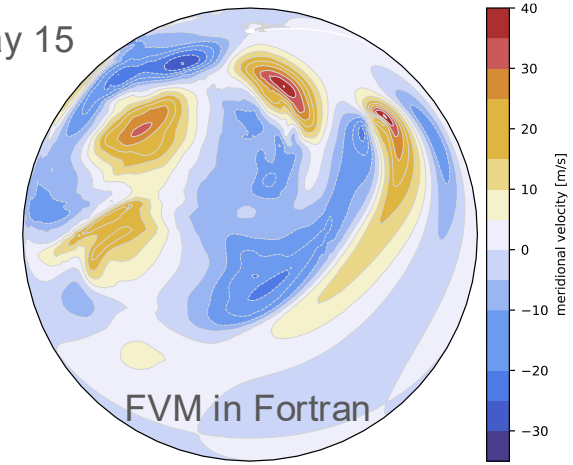
PMAP global dynamical core on the IFS octahedral grid

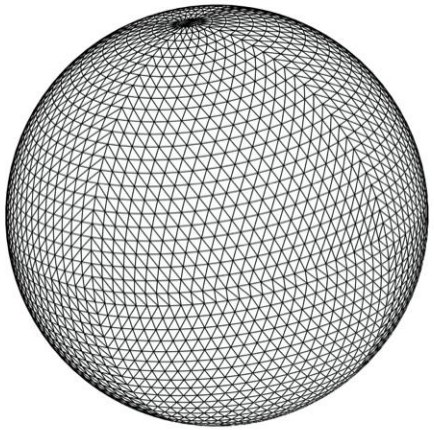
```
@field_operator
def advection_scheme_upwind(
    rho: Field[[Vertex], float],
    dt: float,
    vel: tuple[Field[[Vertex], float], Field[[Vertex],
    vol: Field[[Vertex], float],
    dual_face_orientation: Field[[Vertex, V2EDim], flo
    dual_face_normal: tuple[Field[[Edge], float], Fiel
    dual_face_length: Field[[Edge], float]
) -> Field[[Vertex], float]:
    flux = upwind_flux(rho, vel, dual_face_normal, dua
    return rho - (dt / vol) * neighbor_sum(
        flux(V2E) * dual_face_orientation, axis=V2EDim
```

Baroclinic wave benchmark

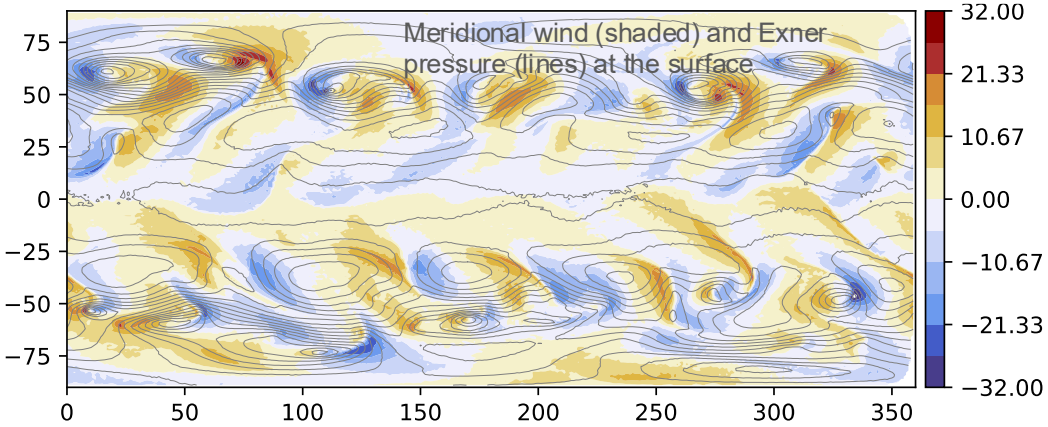


Day 15

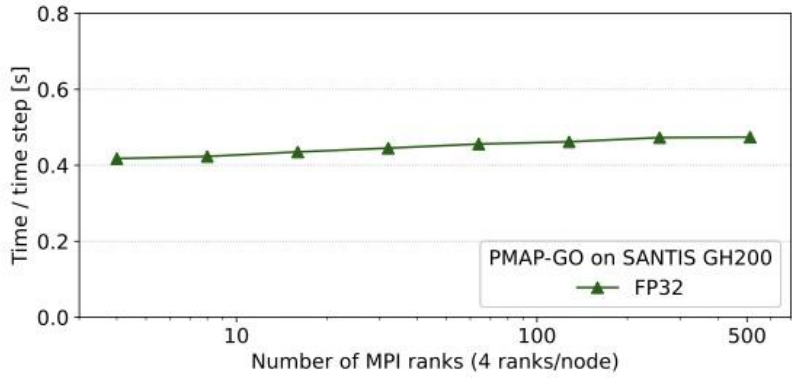




```
from atlas4py import StructuredGrid
grid = StructuredGrid("O24")
mesh = AtlasMesh.generate(grid)
```



Held-Suarez climate benchmark with PMAP
(O360/L91 in FP32 fits on a single Nvidia GH200 GPU)



Weak scaling on the CSCS Alps-Santis cluster (Grace Hopper GH200 GPUs). Computational grids: O640 (for 4 ranks) to O7240 (for 512 ranks)

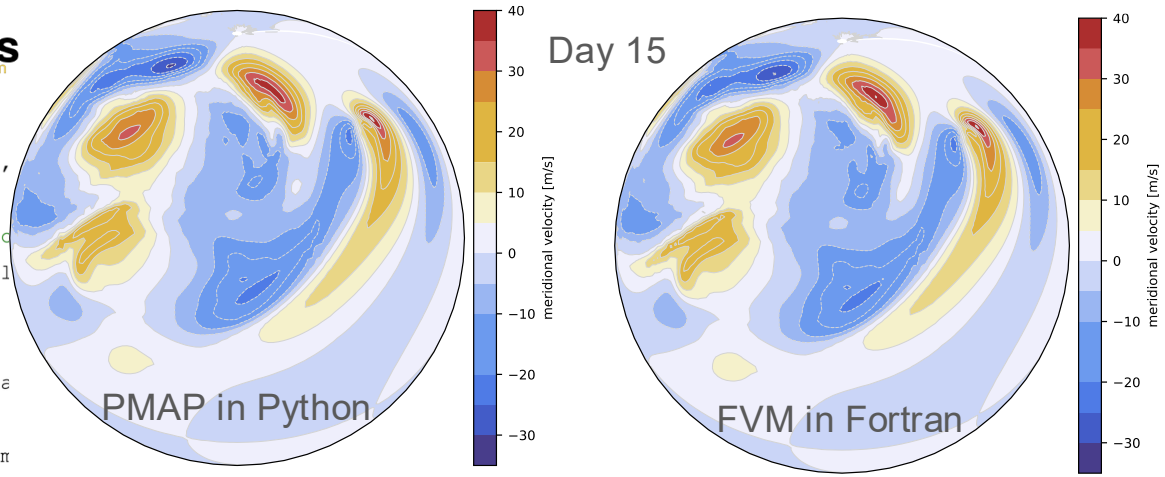
ECMWF ETH zürich CSCS Centro Svizzero di Calcolo Scientifico Swiss National Supercomputing Centre

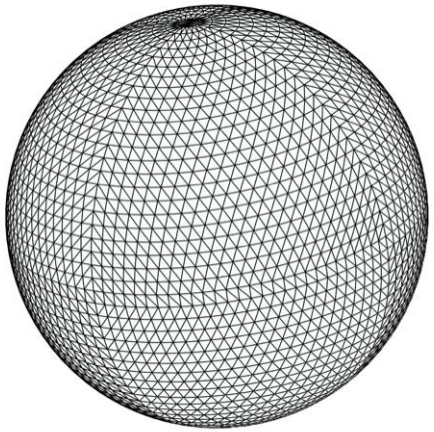
PMAP global dynamical core

```python
@field_operator
def advection_scheme_upwind(
    rho: Field[[Vertex], float],
    dt: float,
    vel: tuple[Field[[Vertex], float], Field[[Vertex],
    vol: Field[[Vertex], float],
    dual_face_orientation: Field[[Vertex, V2EDim], flo
    dual_face_normal: tuple[Field[[Edge], float], Fiel
    dual_face_length: Field[[Edge], float]
) -> Field[[Vertex], float]:
    flux = upwind_flux(rho, vel, dual_face_normal, dua
    return rho - (dt / vol) * neighbor_sum(
        flux(V2E) * dual_face_orientation, axis=V2EDim
```

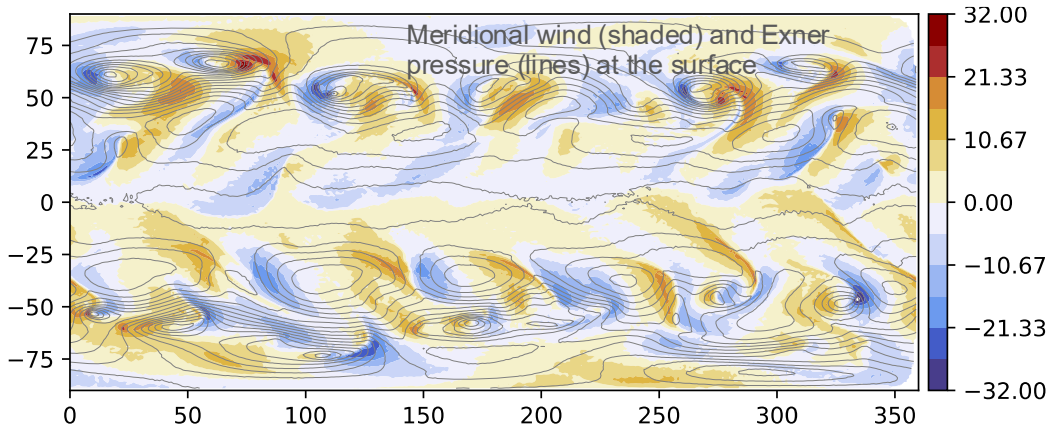
GridTools 4python

Baroclinic wave benchmark



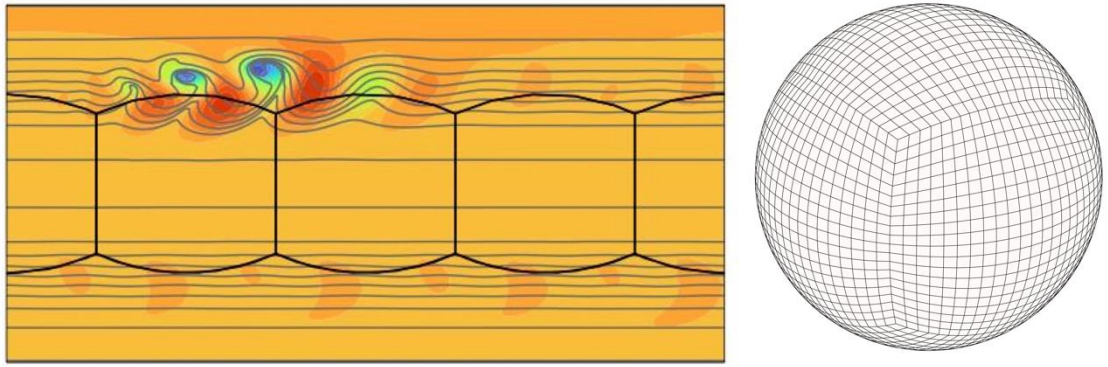


```python
from atlas4py import StructuredGrid
grid = StructuredGrid("O24")
mesh = AtlasMesh.generate(grid)
```



Held-Suarez climate benchmark with PMAP
(O360/L91 in FP32 fits on a single Nvidia GH200 GPU)

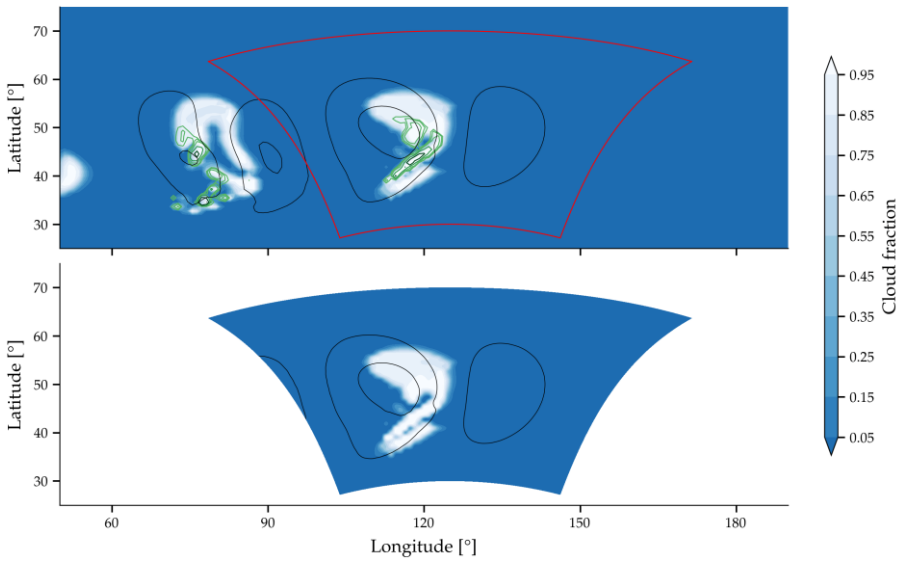
In preparation (L. Papritz): Global PMAP on cubed-sphere

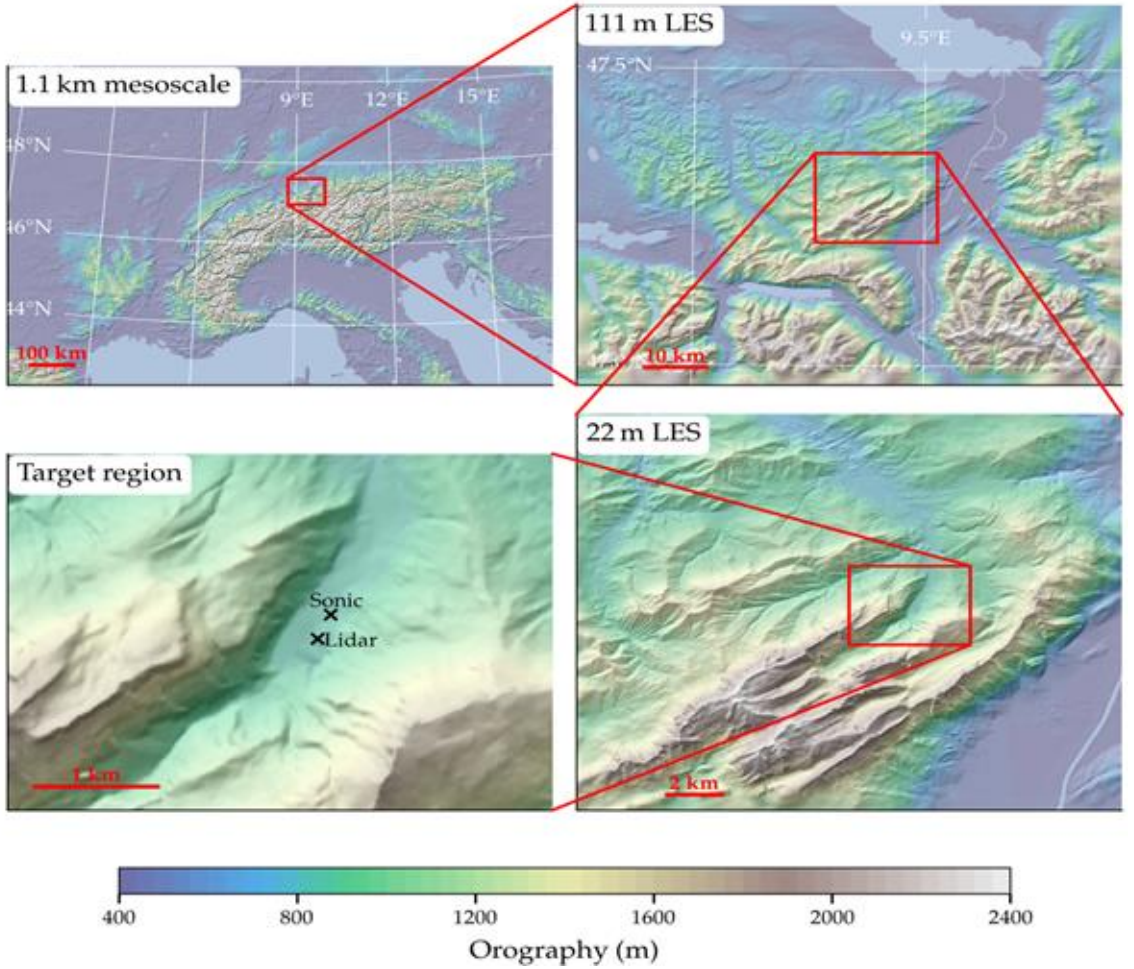


The cubed-sphere grid offers attractive properties and methods we can reuse directly from the regional PMAP (next slides)
-> This is not a new model but a better grid option for local numerics!

ECMWF ETH zürich CSCS Centro Svizzero di Calcolo Scientifico Swiss National Supercomputing Centre

Developing PMAP-LES LAM functionalities for member states and research

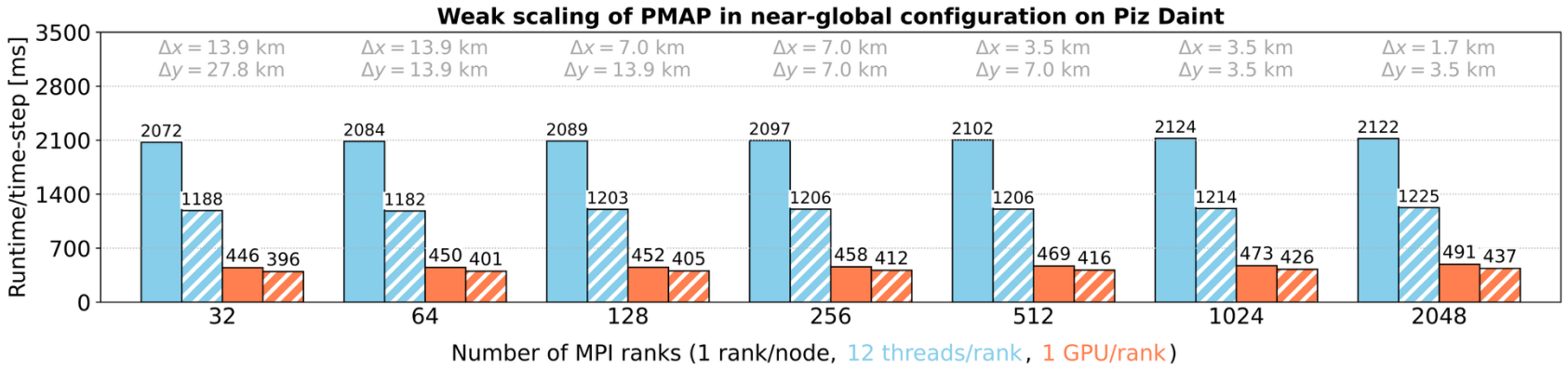




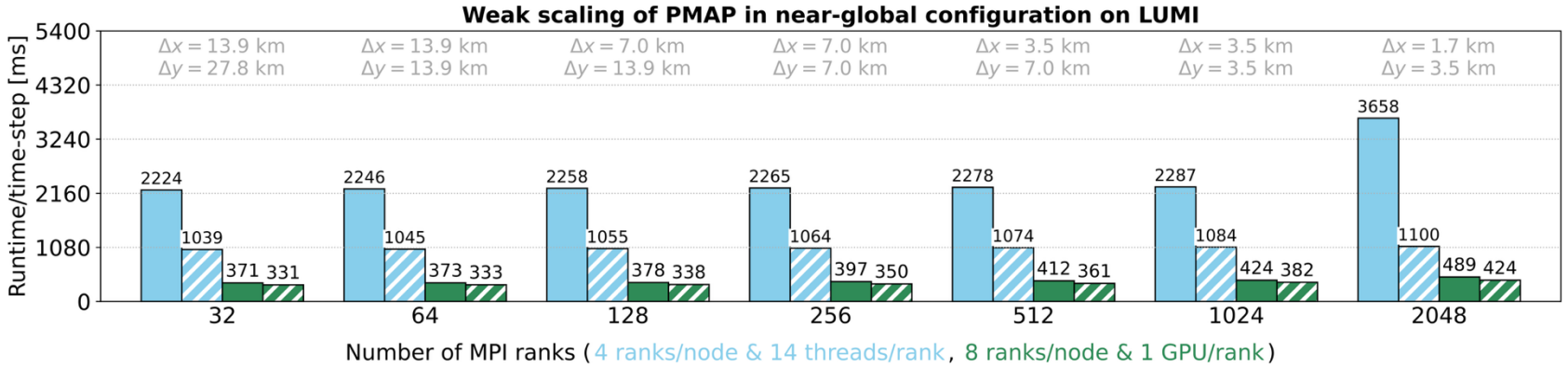
- LAM (Limited-Area Model) funtionalities relevant for some of ECMWF's member states and ETH Zurich research
- Structured quadrilateral grid, rotated spherical coordinates
- Large-eddy simulation (LES) schemes and capabilities (Krieger et al. in prep.; Kühnlein et al. in prep)

Portability and scalability of PMAP on diverse architectures and supercomputers

CSCS Piz Daint



CSC LUMI-G



- Tested Nvidia vs AMD GPUs, vs CPUs, GridTools vs DaCe GPU backends, 32 vs 64 bit
- Optimization of PMAP, GT4Py and the distributed model using GHEX (Generic exascale-ready library for halo-exchange operations) is an ongoing process!

Some references

Kühnlein C., W. Deconinck, R. Klein, S. Malardel, Z.P. Piotrowski, P.K. Smolarkiewicz, J. Szmelter, N.P. Wedi, FVM 1.0: a nonhydrostatic finite-volume dynamical core for the IFS, Geosci. Model Dev., 12, 651-676, 2019.

Krieger N., H. Wernli, M. Sprenger, C. Kühnlein, Revealing the dynamics of a local Alpine windstorm using large-eddy simulations, Weather and Climate Dynamics, 2025.

Smolarkiewicz P. K., C. Kühnlein, N. P. Wedi, A consistent framework for discrete integrations of soundproof and compressible PDEs of atmospheric dynamics, J. Comput. Phys., 2014.

Ubbiali S., C. Kühnlein, C. Schär, L. Schlemmer, T. C. Schulthess, M. Staneker, H. Wernli, Exploring a high-level programming model for the NWP domain using ECMWF microphysics schemes, Geosci. Model. Dev., 2025.

For details of the formulation and public code release watch out for:

Kühnlein C., T. Ehrengruber, S. Ubbiali, N. Krieger, L. Papritz, S. Faghih-Naini, G. Vollenweider, H. Wernli, et al.: Portable Model for multi-scale Atmospheric Prediction (PMAP). *In preparation for GMD 2026.*