Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Eidgenössisches Departement des Innern EDI
Bundesamt für Meteorologie und Klimatologie MeteoSchweiz

Numerical Weather prediction at MeteoSwiss using ICON on GPUs

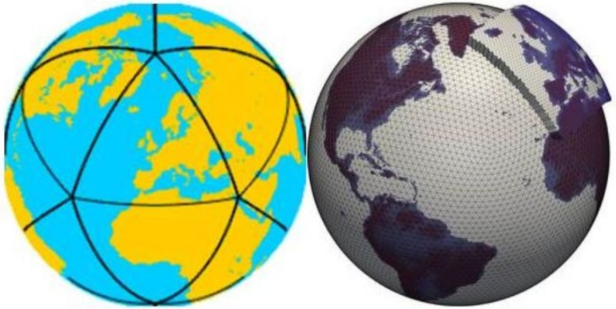
X. Lapillonne[1], N. Burgdorfer[1], V. Cherkas[1], R. Dietlicher[1], E. Germann[1], F. Gessler[1], D. Hupp[1], X. Lapillonne[1], C. Müller[1], M. Röthlin[1], M. Stellio[1], G. Van Parys[1], G. Vollenweider[1], C. Osuna[1], A. Walser[1], M. Bettiol[3], R. Meli[3], A. Gopal[3], M. Jacob[4], A. Jocksch[3], J. Jucker[2], R. Meli[3], W. Sawyer[3], U. Schättler[4], D. Alexeev[5]

[1]MeteoSwiss, [2]C2SM, [3]CSCS, [4]DWD, [5]Nvidia

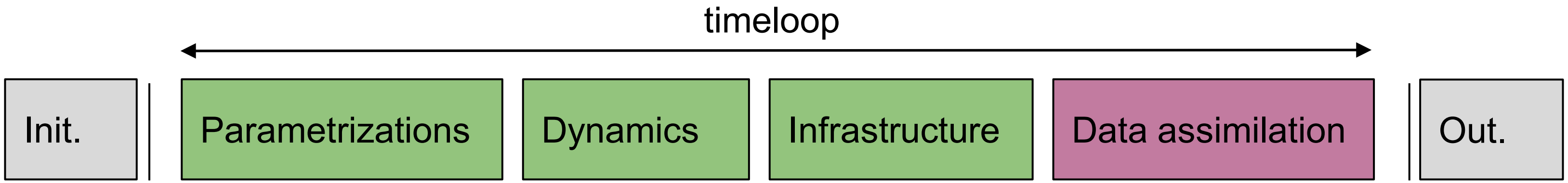
ICON port to GPU

- ICON: Non-hydrostatic global and regional unified climate and numerical weather prediction model.
- ICON partners: DWD, MPI-M, DKRZ, KIT – ICON dev. Partners: C2SM, COSMO …
- Initial GPU port: OpenACC compiler directives, other approach considered: DSL, …
- HPC diversity: running on Intel CPU, AMD CPU, NEC vector Aurora, Nvidia GPU, AMD GPU, …

ICON



ICON model on GPU



timeloop

Init.

Parametrizations

Dynamics

Infrastructure

Data assimilation

Out.

CPU - GPU - Mixed

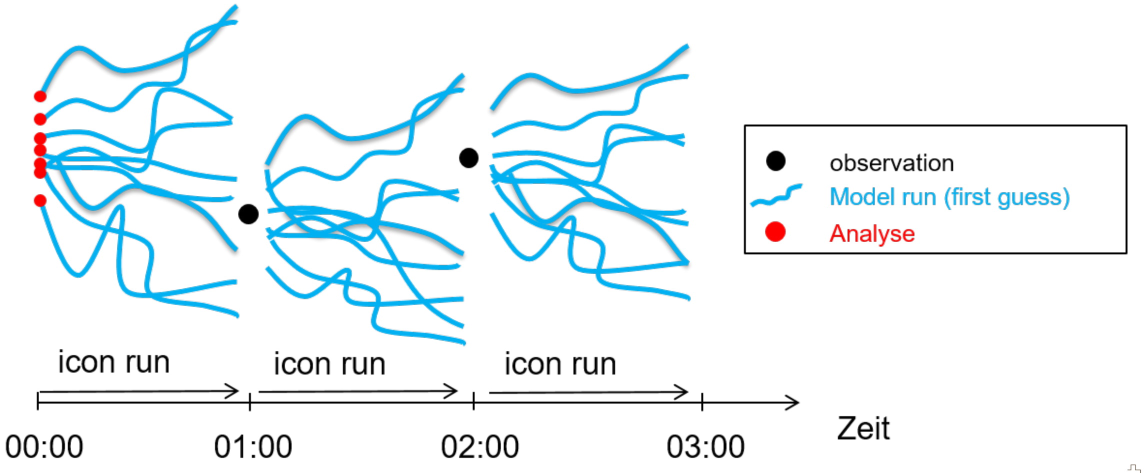
- Full port strategy : avoid GPU-CPU transfer: all components of the time loop need to be ported to GPU
  - Exception: Data assimilation runs on partly on CPU, some diagnostics

- First GPU implementation using OpenACC compiler directives in the orignal Fortran code

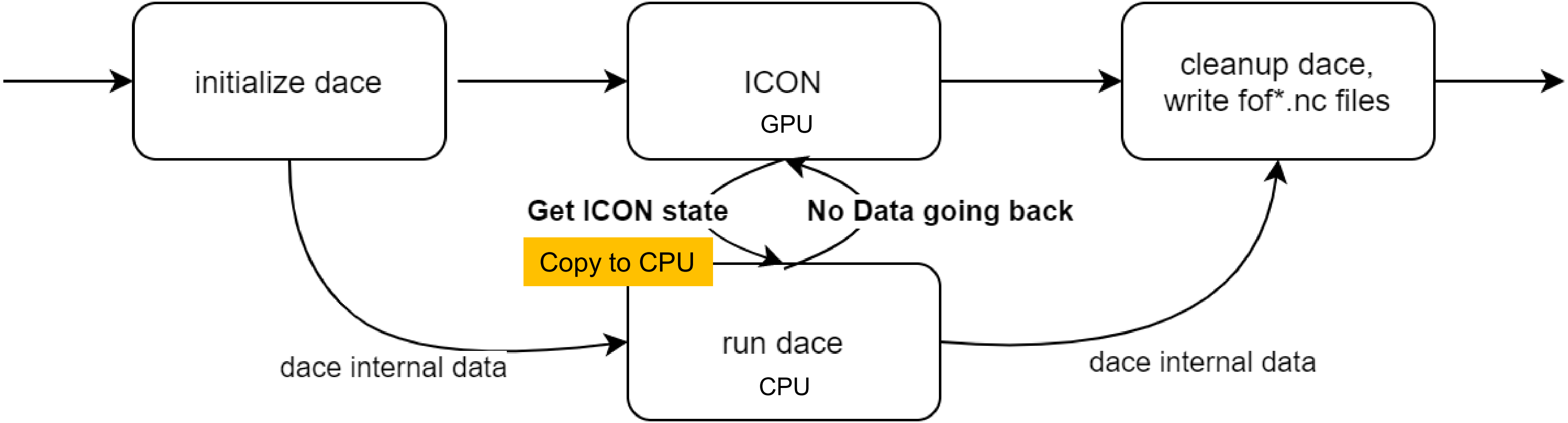
- All components for MCH NWP configuration ported to GPU. All changes in icon master.

ICON with Data Assimilation on GPU

- Kilometer-Scale Ensemble Data Assimilation (KENDA). Calculations in ensemble space spanned by the ensemble members, using observations to compute analysis
- Assimilation component takes the model and compares them with observations (DACE) – write feedback files (fof)



Data Assimilation (DACE) on GPU strategy

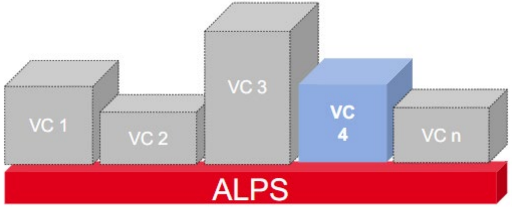


- The DACE code is kept on the CPU. Data is copied from GPU to CPU when needed.

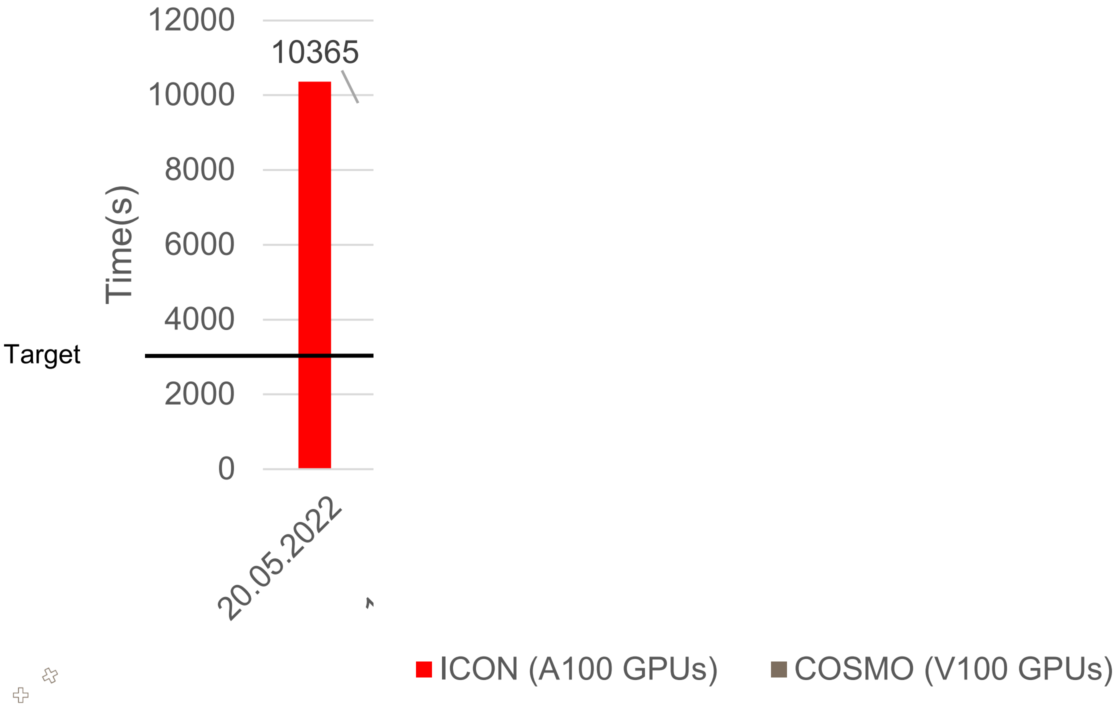
MeteoSwiss system at CSCS

MeteoSwiss system HPC Computing Services on Alps Plattform 2 Virtual Clusters (VC)

  - Production: 42 GPU / 15 CPU Nodes
  - R&D: 30-50 GPU / ~15 CPU Nodes (elastic)
- GPU nodes:
  - 4 x NVIDIA A100
  - 1 x AMD Epyc 64-cores CPU
- CPU nodes:
  - 2 x AMD Epyc 64-cores CPUs
- Not dedicated : part of the large system :
  - R&D partition can be extended
  - new challenges : maintenance, testing …

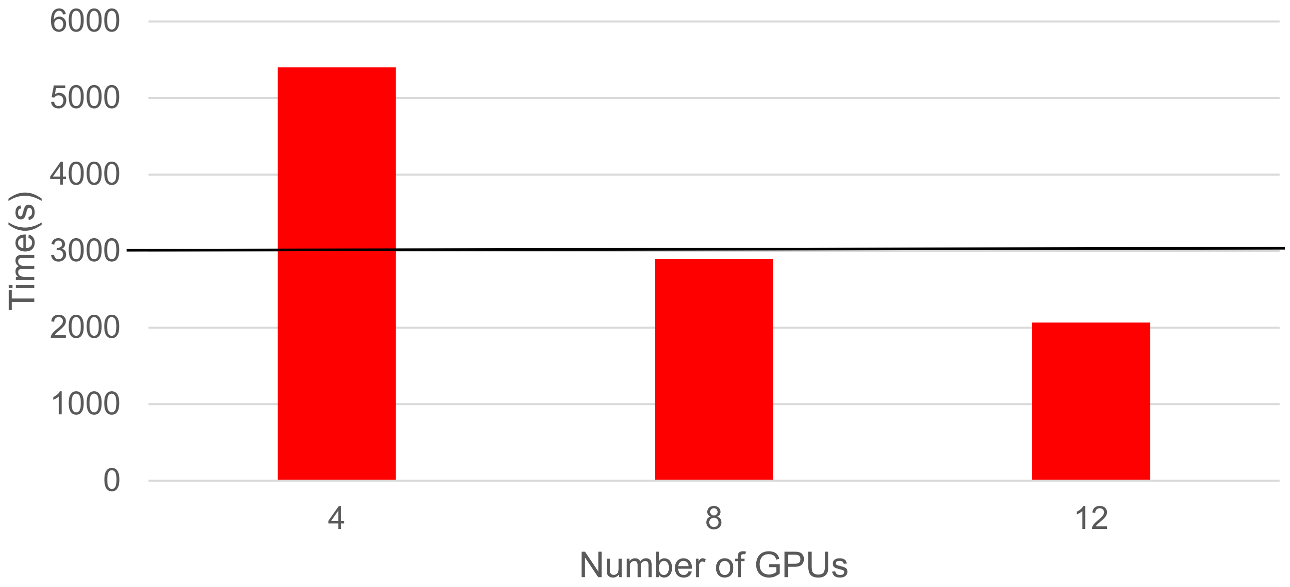


ICON-GPU MeteoSwiss : Performance optimization



- ICON-CH1-EPS on Alps, 33h on 8 A100 Nvidia GPUs, i.e. 2 nodes.
- Optimization reduced time to solution to target performance

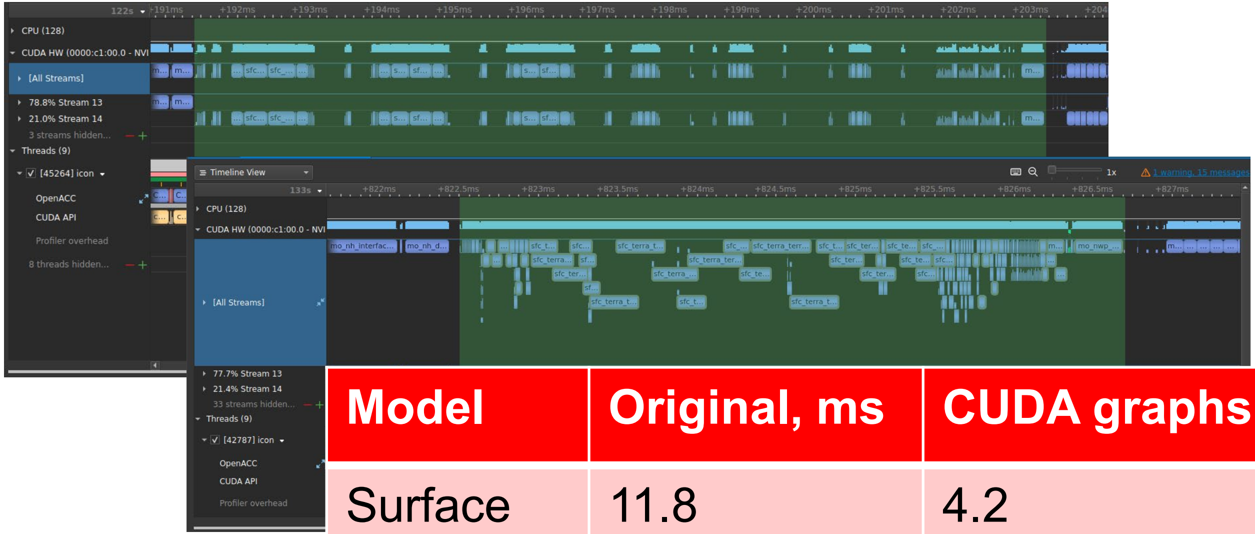
Scaling of ICON-CH1-EPS



- Fit within time to solution on 8 GPUs (=2 nodes), still good scaling when using 12 GPUs (= 3 nodes)

Cuda-graph Optimization

Beneficial for components with many small kernels

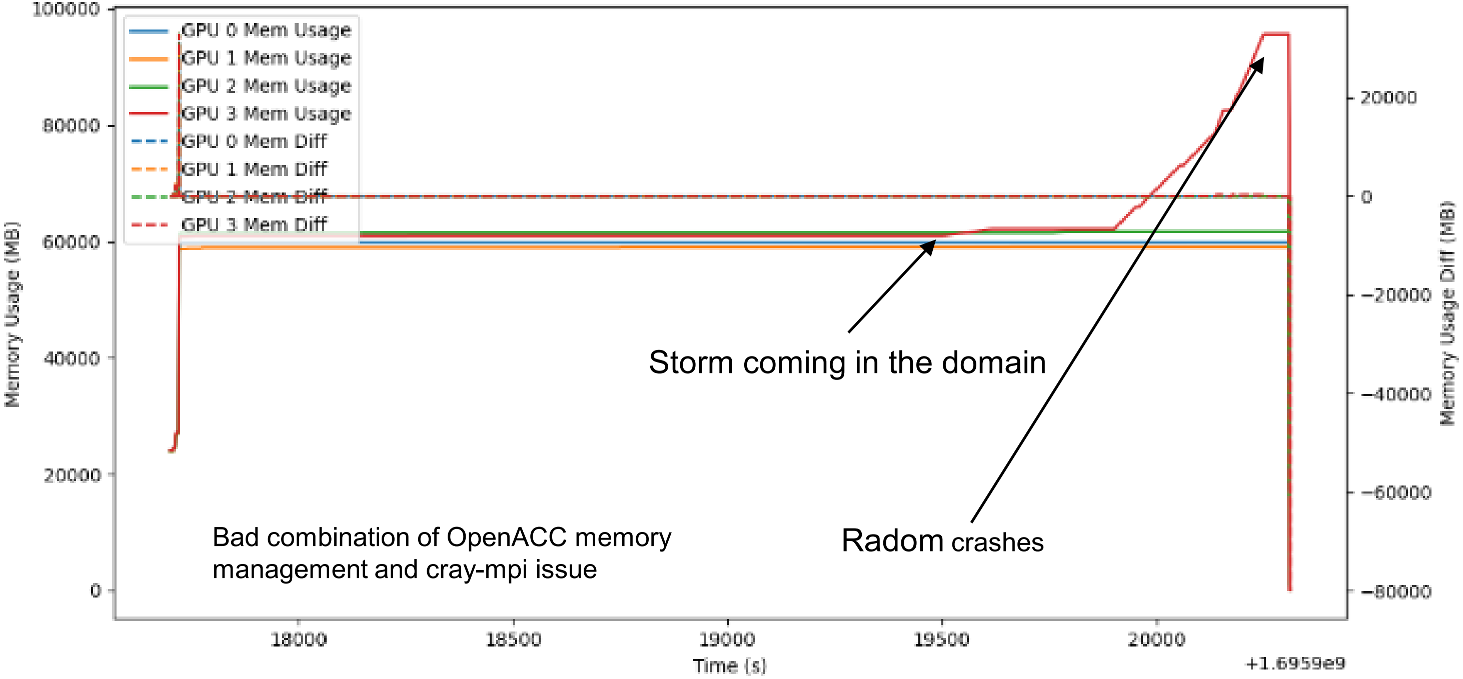


Model	Original, ms	CUDA graphs, ms	Speedup
Surface	11.8	4.2	2.8x
Turbtrans	10.4	2.0	5.1x

Source: Dmitry Alexeev, NVIDIA

GPU development in ICON

- Many issues on our way OpenACC, mpi, competing changes in other MR …
- 9 days before pre-operation: random crash for some weather situation



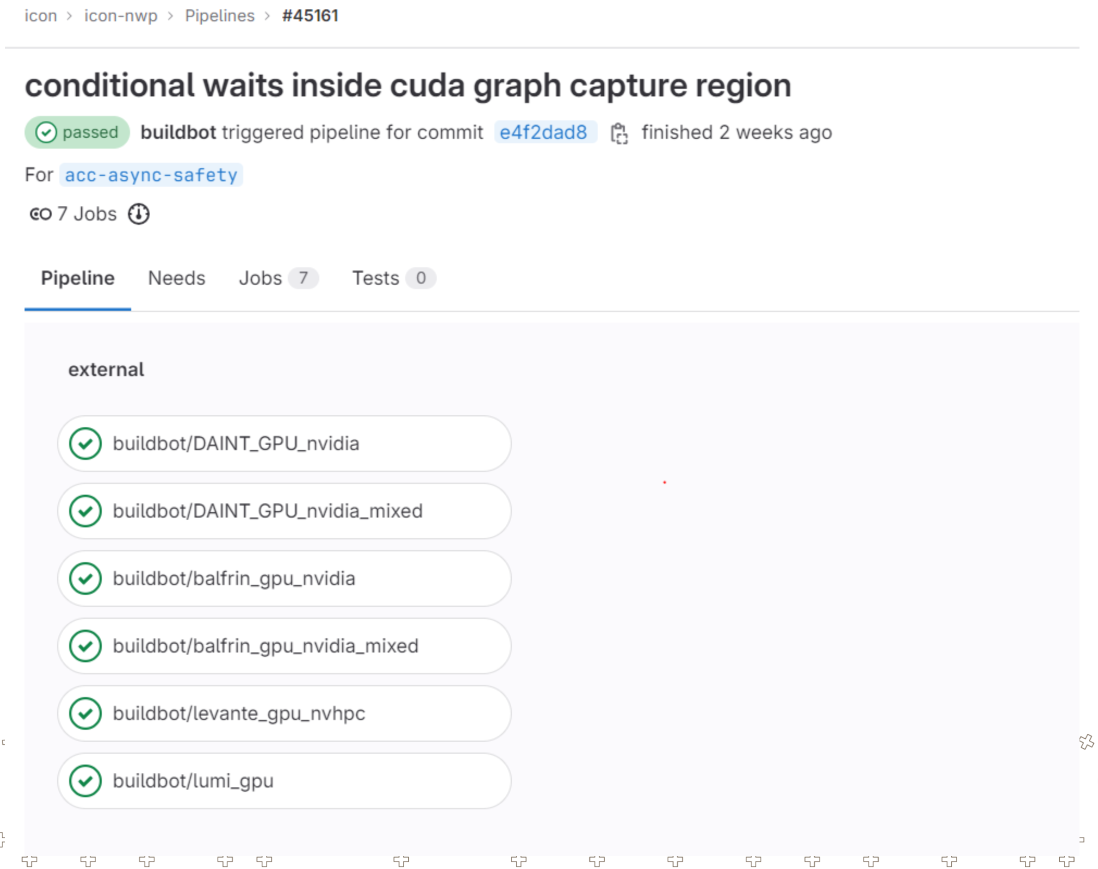
Storm coming in the domain

Radom crashes

Bad combination of OpenACC memory management and cray-mpi issue

Testing and validation

- Short regression tests validating that GPU is within a perturbed CPU ensemble, mpi test …
- Running on many systems of the ICON community, integrated with gitlab
- Long validation (multiple seasons) against observations and CPU executable



icon › icon-nwp › Pipelines › #45161

conditional waits inside cuda graph capture region

passed **buildbot** triggered pipeline for commit e4f2dad8 finished 2 weeks ago

For acc-async-safety

7 Jobs

Pipeline | Needs | Jobs 7 | Tests 0

external

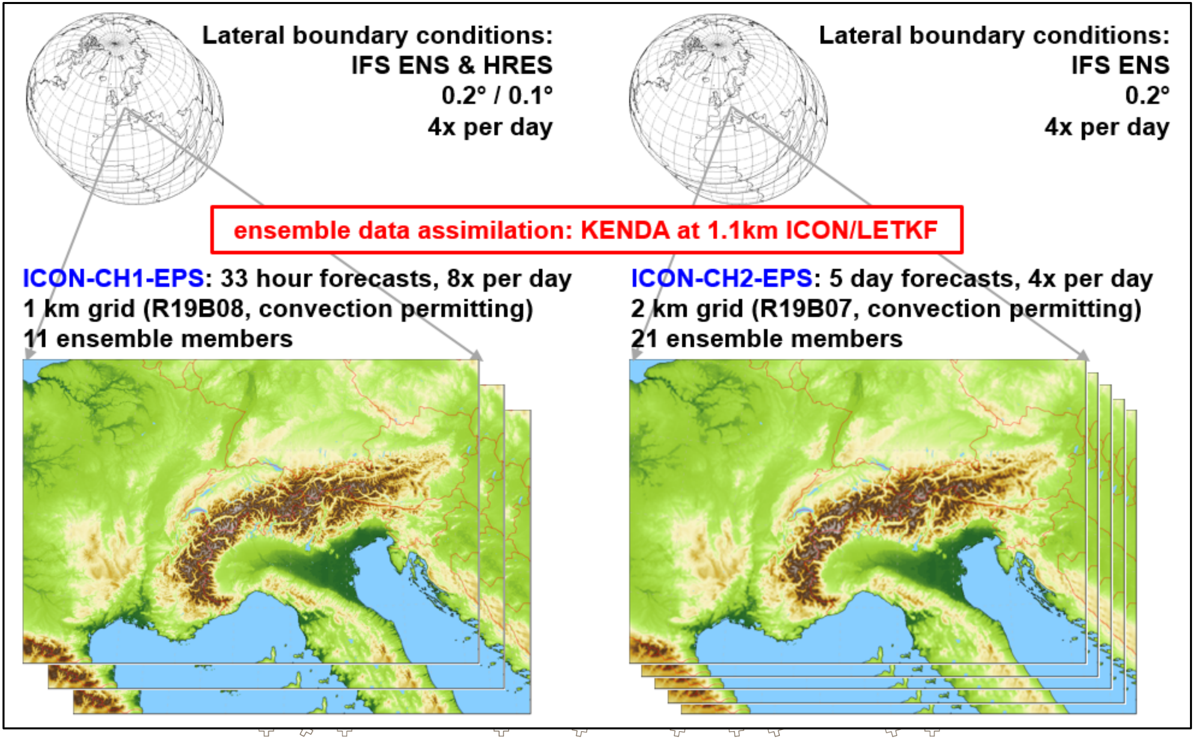
- ✓ buildbot/DAINT_GPU_nvidia
- ✓ buildbot/DAINT_GPU_nvidia_mixed
- ✓ buildbot/balfrin_gpu_nvidia
- ✓ buildbot/balfrin_gpu_nvidia_mixed
- ✓ buildbot/levante_gpu_nvhpc
- ✓ buildbot/lumi_gpu

MeteoSchweiz

MeteoSwiss ensemble system

- Pre-operational phase started October 2nd 2023
- Not using all optimization yet (no cuda-graph)

- Full schedule on Tasna (Alps)
- ICON-CH1-EPS : 3 nodes/member
- ICON-CH2-EPS : 1 node/member
- KENDA-CH1 : 1 node/member



Lateral boundary conditions:
IFS ENS & HRES
0.2° / 0.1°
4x per day

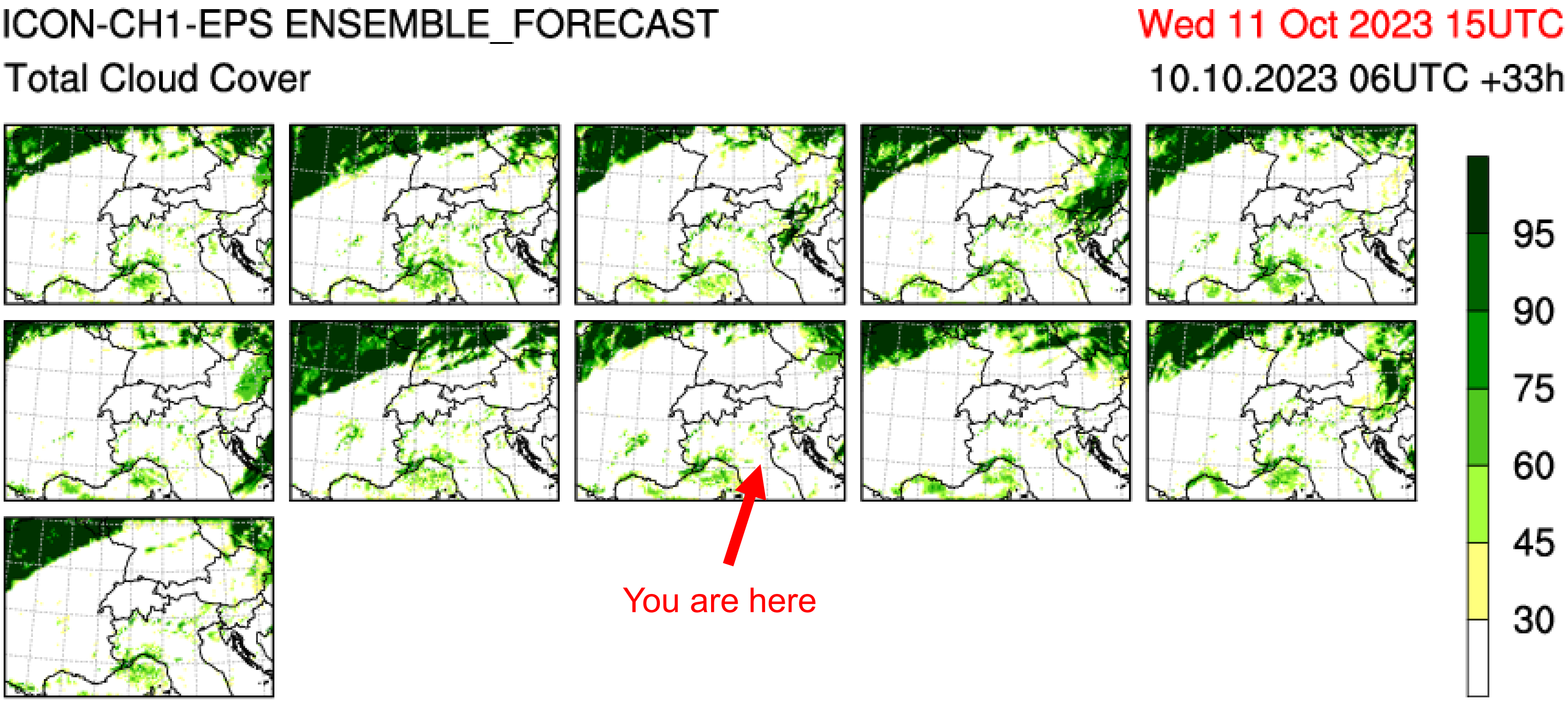
Lateral boundary conditions:
IFS ENS
0.2°
4x per day

ensemble data assimilation: KENDA at 1.1km ICON/LETKF

ICON-CH1-EPS: 33 hour forecasts, 8x per day
1 km grid (R19B08, convection permitting)
11 ensemble members

ICON-CH2-EPS: 5 day forecasts, 4x per day
2 km grid (R19B07, convection permitting)
21 ensemble members

ICON forecast on GPU running on Alps



ICON-CH1-EPS ENSEMBLE_FORECAST

Total Cloud Cover

Wed 11 Oct 2023 15UTC

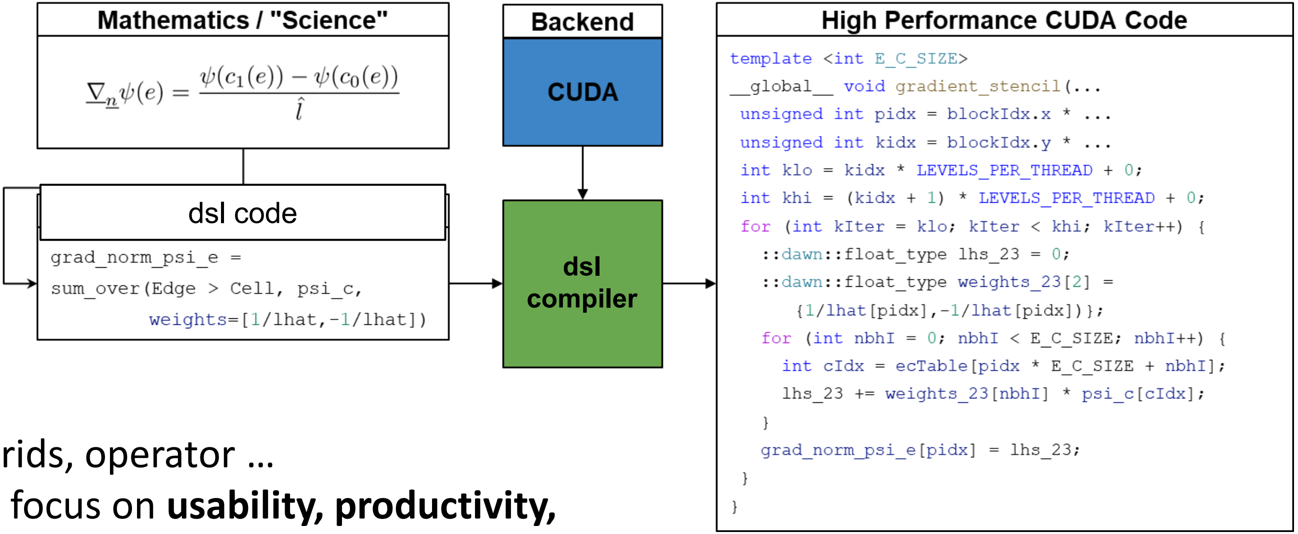
10.10.2023 06UTC +33h

You are here

High level DSL for weather and climate

Separation of concern:

numerical formulation
(domain scientist)
vs **hardware implementation**
(computer engineer)



Mathematics / "Science"

$$\underline{\nabla}_n \psi(e) = \frac{\psi(c_1(e)) - \psi(c_0(e))}{\hat{l}}$$

dsl code

```
grad_norm_psi_e =
sum_over(Edge > Cell, psi_c,
        weights=[1/lhat,-1/lhat])
```

High Performance CUDA Code

```
template <int E_C_SIZE>
__global__ void gradient_stencil(...
 unsigned int pidx = blockIdx.x * ...
 unsigned int kidx = blockIdx.y * ...
 int klo = kidx * LEVELS_PER_THREAD + 0;
 int khi = (kidx + 1) * LEVELS_PER_THREAD + 0;
 for (int kIter = klo; kIter < khi; kIter++) {
   ::dawn::float_type lhs_23 = 0;
   ::dawn::float_type weights_23[2] =
      {1/lhat[pidx],-1/lhat[pidx]}};
   for (int nbhI = 0; nbhI < E_C_SIZE; nbhI++) {
     int cIdx = ecTable[pidx * E_C_SIZE + nbhI];
     lhs_23 += weights_23[nbhI] * psi_c[cIdx];
   }
   grad_norm_psi_e[pidx] = lhs_23;
 }
}
```

- Domain specific concepts, grids, operator …
- High level Python based DSL focus on **usability, productivity, performance**
- Architecture agnostic user code, allow optimizations across components : exascale, HPC diverstiy
- Development of a compiler tool chain for ICON in the EXCLAIM project:
  - Talk : GT4Py: A Python framework for weather and climate applications, Till Ehrengruber
  - Poster: A python implementation of the ICON dynamical core for operational NWP, Daniel Hupp

Outlook

- The ICON model was ported to GPU using OpenACC including the required components for regional NWP forecast
- ICON-CH1-EPS, ICON-CH2-EPS are in pre-operation phase at MeteoSwiss since October 2nd 2023 on GPUs on the Alps infrastucture. The planned date for production is april 2024.
- Further work : improve stability, solve remainding issues, maintenance procedure …
- Key to success:
  - dedicated team, great support from vendor, e.g. Nvidia, and CSCS : thanks !
  - don’t wait for fix in compiler and libraries: find workaround
  - Continuous integration and testing at all steps