



Sprint 12

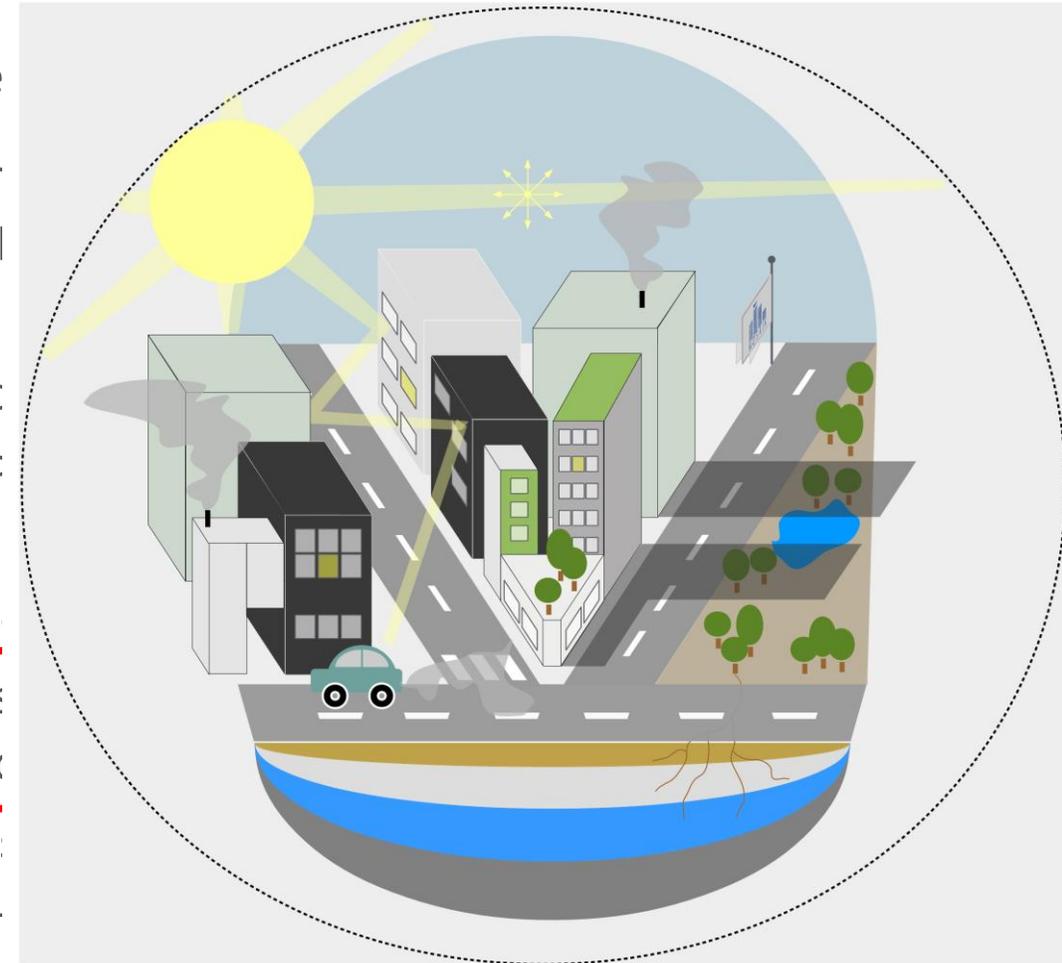
Challenges and results experienced during the PALM sprint

Siegfried Raasch (Leibniz Universität Hannover), Wilton Jaciel Loch (DKRZ), Enrico Degregori (DKRZ)

General information



- PALM model system (**PAR**allelized **LES** **M**odel)
 - turbulence resolving community model for micro-scale
 - Fortran 2008 code, developed since 1997 originally at I
 - more than 1000 users worldwide, international develop
 - solves anelastic NS-equations (direct (FFT) or multigrid
 - 2d domain decomposition, cartesian grid allowing for c
 - modular structure (full atmospheric chemistry using KF model, ICON cloud physics, wind turbine model, ...)
 - self-nesting using MPI one-sided communication, mes
 - detailed modelling of surface processes (soil model bas including radiation modelling via ray-tracing, indoor m
 - highly optimized for cache-based and vector processor
 - maintained via gitlab (strict rules for developers, CI tes



General information



- Purpose of PALM sprint: *to advance and extend the existing GPU porting*
- Status of GPU porting **before** the sprint:
 - first porting via OpenACC directives started in 2011
 - complete code running on GPU (except initialization), to minimize host-device data transfer
 - porting restricted to PALM core (dynamics, thermodynamics), based on vector-optimized code, standalone (non-MPI), no netCDF data output
 - ported code entered the SPEC ACCEL suite in 2014
 - after 2014 multi-GPU capabilities have been added, using CUDA-aware MPI
 - very poor performance when using MPI-derived datatypes for halo data exchange
 - failed to enter the SPEC HPC 2021 benchmark (in final stage)
 - low motivation for spending more time on GPU porting
 - strongly increasing availability of GPUs has changed our motivation
 - natESM sprint offered us a chance to continue with the porting

PALM sprint original working program



- GPU porting of all surface relevant processes
 - land surface model
 - urban surface model, including the ray-tracing algorithm
 - chemistry module
- Performance improvement of CUDA-aware MPI
- Porting of the multigrid Poisson-solver

Challenges



- estimated time for working packages much too optimistic because of
 - complexity of modules
 - unforeseen additional work (missing parts of PALM core that have not been ported)
 - change in personnel at DKRZ (Enrico Degregori, Wilton Loch)
- porting restricted to **land surface model** and **multigrid Poisson-solver**

Results



- Performance improvement of CUDA-aware MPI
 - copy non-contiguous halo data to contiguous buffers instead of using MPI-derived data types

```
CPU measures for      4 PEs (      2(x) *      2(y) tasks *      1 threads):
gridpoints (x/y/z):      256 *      256 *      512
```

place:	mean		counts	min	max	rms
	sec.	%		sec.	sec.	sec.
MPI derived datatypes						
timesteps	82.756	95.91	115	82.76	82.76	0.00
exchange_horiz	49.445	57.31	4162	47.98	50.94	1.46
poisfft	21.069	24.42	346	19.66	22.48	1.4
contiguous buffers						
timesteps	35.679	91.03	115	35.68	35.68	0.00
poisfft	21.118	53.88	346	19.67	22.58	1.43
exchange_horiz	2.438	6.22	4162	1.01	3.90	1.43

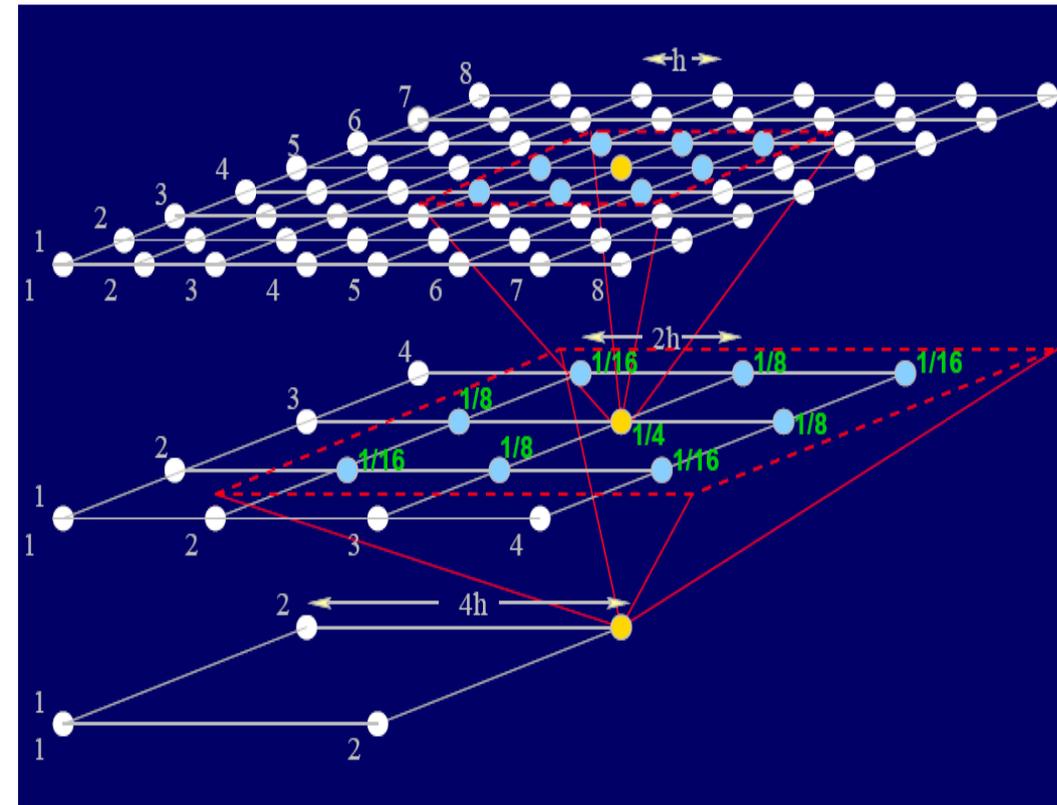
Results



- Porting of the multigrid (MG) Poisson-solver

- MG solves Poisson equation via Gauss-Seidel iterations on different grid levels
- recursive routine
- requires very frequent halo exchanges

- porting finished successfully
- results are bit compatible
- no time for loop optimization → poor performance
e.g. for Gauss-Seidel iterations
14.4 s (64 core CPU) ↔ 5.6 s (4 GPU)



Results



- Porting of the land surface model (LSM)
 - includes soil model, energy balance equation, calculation of surface fluxes via Monin-Obukhov
 - all surface grid point data is stored in 1d-arrays to get long vectors

- requires humidity, but prognostic equation for mixing ratio had not been ported to GPU so far
- no bit compatibility, traced back to ATAN / LOG used in calculating the MO stability functions
- no performance optimizations (only very small test cases used)
- recently performed tests for larger grids show that LSM performance behaves like the PALM core

Outlook & open questions



- PALM GPU mode now able to perform netCDF I/O (i.e. suitable for production runs)
- GPU porting part of PALM release 25.10
 - see https://docs.palm-model.org/latest/Guide/LES_Model/GPU_mode_usage/
- probably GPU performance can be further improved, **but:**
- new BMBF project HiFiOffWind
 - industrial application of PALM for big offshore wind farms
 - Fraunhofer IWES, Univ. Oldenburg, pecanode GmbH
 - starting in April
 - includes PALM porting via openMP5, to be carried out by pecanode
- Is PALM a candidate for natESM?