Further COSMO-Model Development

or:
Is it dangerous to buy a vector computer?

Ulrich Schättler (FE)
Elisabeth Krenzien, Henning Weber (TI)
Deutscher Wetterdienst

Contents

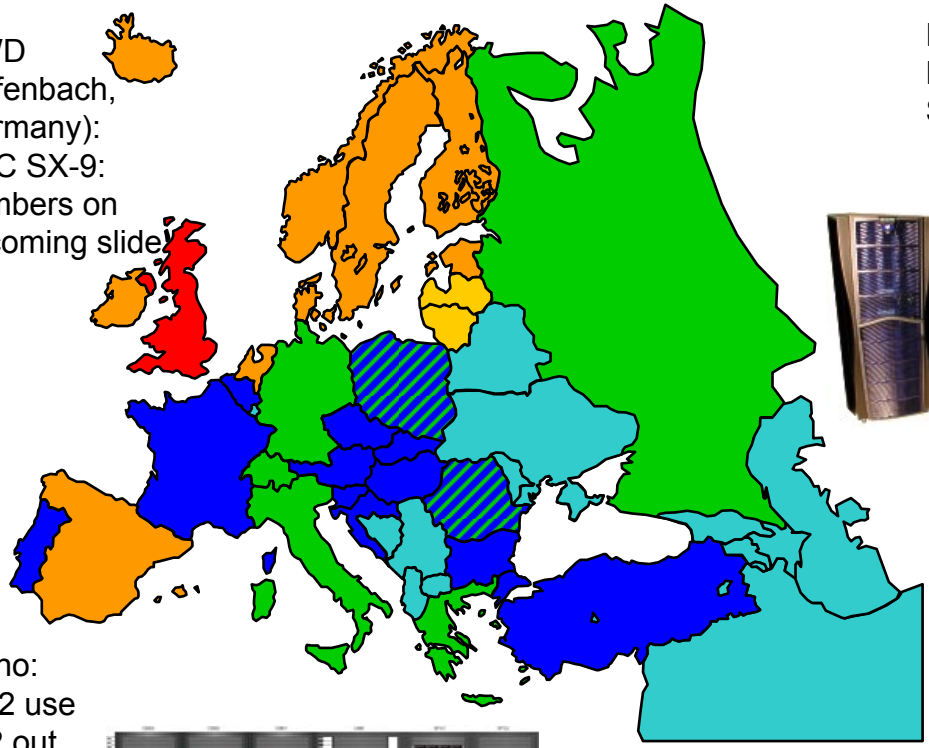
- The COSMO-Model in the last 2 years
- DWD‘s new supercomputer
- Is it dangerous to buy a vector computer
- Further COSMO-Model development

The COSMO-Model in the last 2 years

Acknowledgements to all our COSMO colleagues

COSMO-Model(s)

DWD (Offenbach, Germany): NEC SX-9: numbers on upcoming slide



Roshydromet (Moscow, Russia), NMA (Bucharest, Romania): Still in planning / procurement phase

IMGW (Warsawa, Poland): SGI Origin 3800: uses 88 of 100 nodes

MeteoSwiss: Cray XT4 at CSCS, Manno: COSMO-7 and COSMO-2 use 800+4 MPI-Tasks on 402 out of 448 dual core AMD nodes

ARPA-SIM (Bologna, Italy): IBM pwr5: up to 160 of 512 nodes at CINECA

COSMO-LEPS (at ECMWF): running on ECMWF pwr5 as member-state time-critical application

HNMS (Athens, Greece): IBM pwr4: 120 of 256 nodes

USAM (Rome, Italy): HP Linux Cluster Intel XEON biproc quadcore (1024 cores) System right now undergoing acceptance test

ARPA-SIM (Bologna, Italy): Linux-Intel x86-64 Cluster for testing (uses 56 of 120 cores)

The last 2 years (ADM)

- With deep regret we have to announce that the „Lokal-Modell“ (LM) has gone out of business soon after the last ECMWF HPC Workshop.
- But we are proud to inform you that all COSMO partners are now using one and the same model: The COSMO-Model
- Due to a highest management decision, the name of all former LM applications had to be replaced by COSMO-XX (where XX characterises the application, e.g. COSMO-EU or COSMO-DE)
- Russia joined COSMO as an applicant member

The last 2 years (SCI)

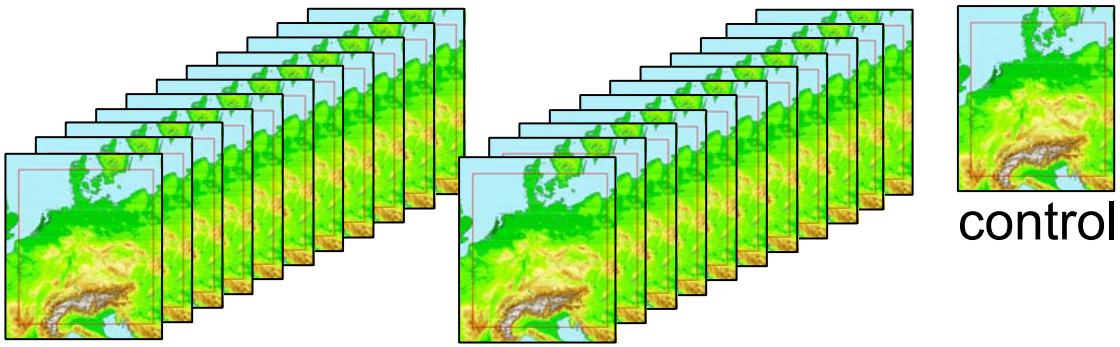
- High resolution applications
  - operational in Germany, Italy (2.8 km) and Switzerland (2.2 km)
  - including assimilation of radar data (Germany, Switzerland)
  - based on Runge-Kutta dynamical core
- Coarser resolution applications
  - have been adapted to Runge-Kutta dynamical core
  - implementation and testing of a sub-grid scale orography scheme

COSMO Ensembles

- COSMO LEPS
  - Limited Area EPS developed within COSMO to improve the short-to-medium range forecast of extreme weather events
  - 16 COSMO-Model members ($\Delta x \sim 10$ km) nested in selected members of ECMWF EPS
  - Running as member-state time-critical application at ECMWF
- COSMO SREPS
  - Short Range EPS to improve the support in case of high-impact weather
  - 16 COSMO-Model members ($\Delta x \sim 7$ km) driven by the COSMO members of the spanish Multi-Model EPS
  - Extensive testing during the MAP D-Phase DOP
  - Provides boundaries to the high-resolution COSMO-DE EPS

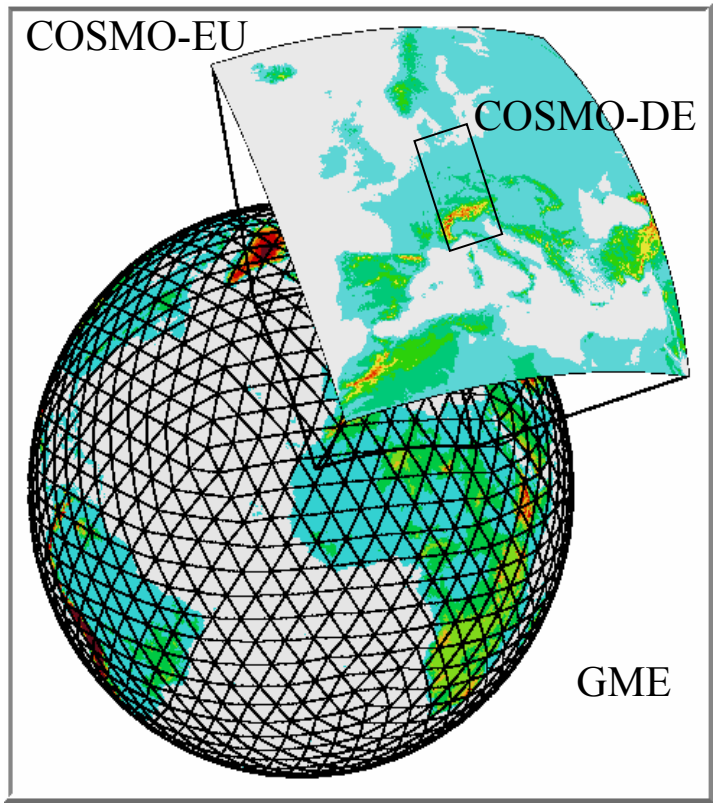
COSMO Ensembles

- COSMO-DE EPS
  - Convection resolving EPS based on COSMO-DE under development at DWD
  - Perturbations for initial and boundary conditions and model setup
  - Operational use with 20 members is aimed for 2009/10



DWD‘s new Supercomputer

Compute Requirements

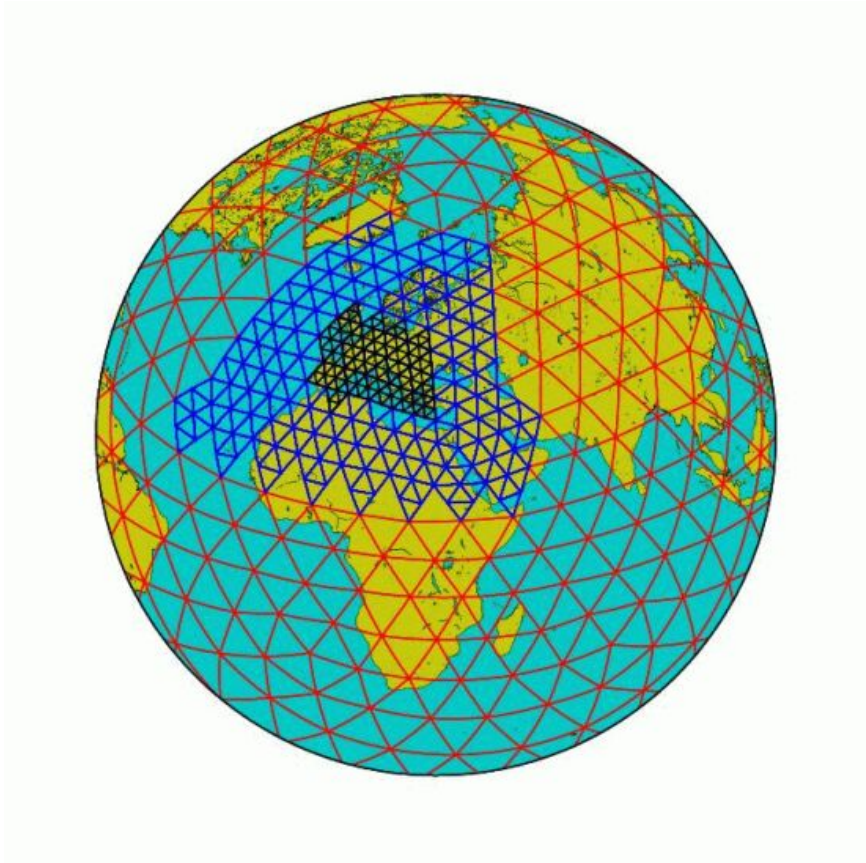


Available Budget:
nearly 40 Mio € for ~5 years
including maintenance

- Today:
  - GME (40 km;168 h)
  - COSMO-EU (7 km; 78 h)
  - COSMO-DE (2.8 km; 21 h)
- ≥ 2008 (Phase I):
  - GME (20km; 168 h)
  - COSMO-EU (7 km; 78 h)
  - COSMO-DE EPS (20 members; 2.8 km; 21 h)
- ≥ 2010 (Phase II):
  - ICON with local zooming option replaces GME and COSMO-EU
  - COSMO-DE EPS with more members and / or higher resolution
- Additional 25 % for military service

ICON

- A new unified global and regional forecast model ICON is developed in collaboration with the *Max-Planck-Institut für Meteorologie*
  - ICOsahedral Nonhydrostatic
  - for global and limited area modeling
  - Triangular grid
  - Compressible; conservation properties (mass, energy, ...)
  - for weather forecasting and climate modeling



Technical Constraints

Data volume today:

- GME about 375 GB/day
- COSMO-EU about 200 GB/day
- COSMO-DE about 286 GB/day
- COSMO-RM about 50 GB/day
- COSMO-RMK about 75 GB/day

- Makes 7 TB/day
- Need transfer rates of 2 GB/s

Problem:

Amount of data increases also by a factor of 15!

We also need a new data base server!

General:
Need a twin-system for operational and experimental jobs

Benchmark Tests

- Performance Test: capacity
  - 30 COSMO-DE EPS members must run in 1400 seconds
- Scalability Test: capability
  - Run a very large model (1500×1500×50) on few and on many processors (domain as COSMO-EU with resolution as COSMO-DE: 2.8 km)
- Operational Test: switch-over
  - Run the performance test on a full machine
- Granularity Test:
  - what happens, if the performance requirements are increased: Run the Performance Test with dt=25s instead of dt=30s in 1400 seconds.

NEC SX-9: Original Plan

- Phase I:
  - Because no prototype was available, all benchmark tests have been performed on SX-8 and SX-8R
  - Projection to the SX-9: 30 COSMO-DE EPS members can run on 8 nodes with 16 CPUs each.
  - In Phase I there will be 8+8+1 nodes.
  - These machines are just built up in Offenbach.
- Phase II:
  - The increase of performance was a matter of competition
  - DWD expected a doubling of performance
  - NEC offered a factor of 3!

Ongoing Actions

- Acceptance test for Phase I will start in November.
- This is later than planned due to a delay in manufacturing and delivering.
- In the meantime, the operational suite is being migrated to an interim system (SX-8R)

Is it dangerous to buy a vector computer?

15 years of experience tell me

It is dangerous to buy any new supercomputer!

Possible Risks

1) Your codes may not run efficiently on the new computer:

The COSMO-Model is running on vector processors since more than 10 years. Back in the 90ies, the first prototype was developed on a Cray C90. Therefore we expected no problems. This was confirmed by the benchmark tests and our first experiences on the SX-8R:

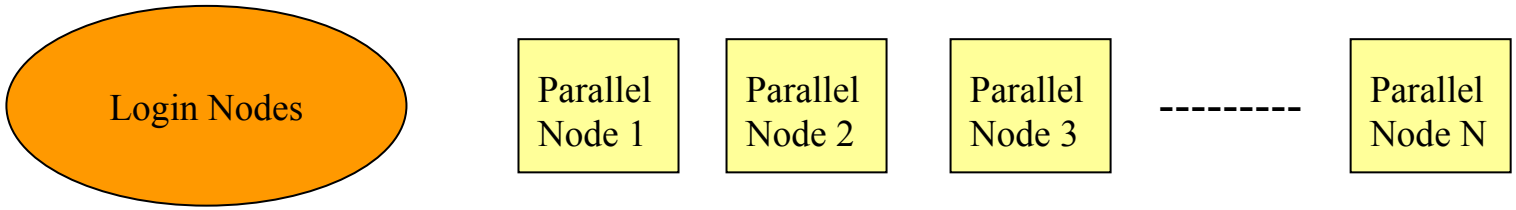
full COSMO-DE	NEC SX-8R	IBM pwr5
# Processors	4 nodes, 32 PEs	32 nodes, 256 PEs, 512 MPI tasks (SMT)
Time for 21 h	1308 s	1359 s
Operations	$182 * 10^{12}$	-
GFlop / s	143	-
MFlop / s / PE	5076 (15.6 % of peak)	-

The same code was running on both machines!

Possible Risks

2) With a new system you usually have to handle increased complexity

Principal architecture of „useable“ high-performance computers:



You can choose, in which part of the system you want to have the highest complexity:

- SMP nodes with single / dual / … / multiple-core nodes
- Login Nodes identical to parallel nodes or not
- File System (Global / Parallel)

This time DWD decided on less complex parallel nodes and has to handle increased complexity between login nodes and parallel nodes.

Possible Risks

3) The systems are changing, but some things always remain the same:

New systems mean new compilers and new software:

- You have to learn about the compilers: good features, awkward features, bugs, …
- Example: some routines of our models have to be compiled with less compiler optimizations, in order to get reproducible results. It took us more than 2 days to find out, which routines.
- You have to adapt your scripts and suites to new batch systems (work in progress)

Possible Risks

4) You never know what you are buying.

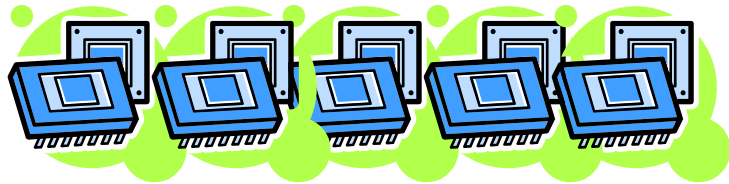
Because if such a system already exists, it would be out-of-date by the time you can use it in your computer room.

But this means you have to live with surprises.

Possible Risks

5) Once the system is built up and running smoothly, you can run your application for years and think you need not care about the rest of the world.

Multi-Core Chips



For the COSMO-Model we face the problem of scalability:
A flat MPI implementation might not be efficient on multi-core architectures.
The time to evaluate and test this is now!
And not in 3 years, when the next procurement is started.

Further COSMO-Model Development

COSMO: Priority Projects

- UTCS:
  - Towards a Unified Turbulence and Convection Scheme
  - To parameterise boundary-layer turbulence and shallow non-precipitating convection in a unified framework
  - To achieve a better coupling between turbulence, convection and radiation
- KENDA:
  - To develop a Km-scale ENsemble-based Data Assimilation for the convective scale

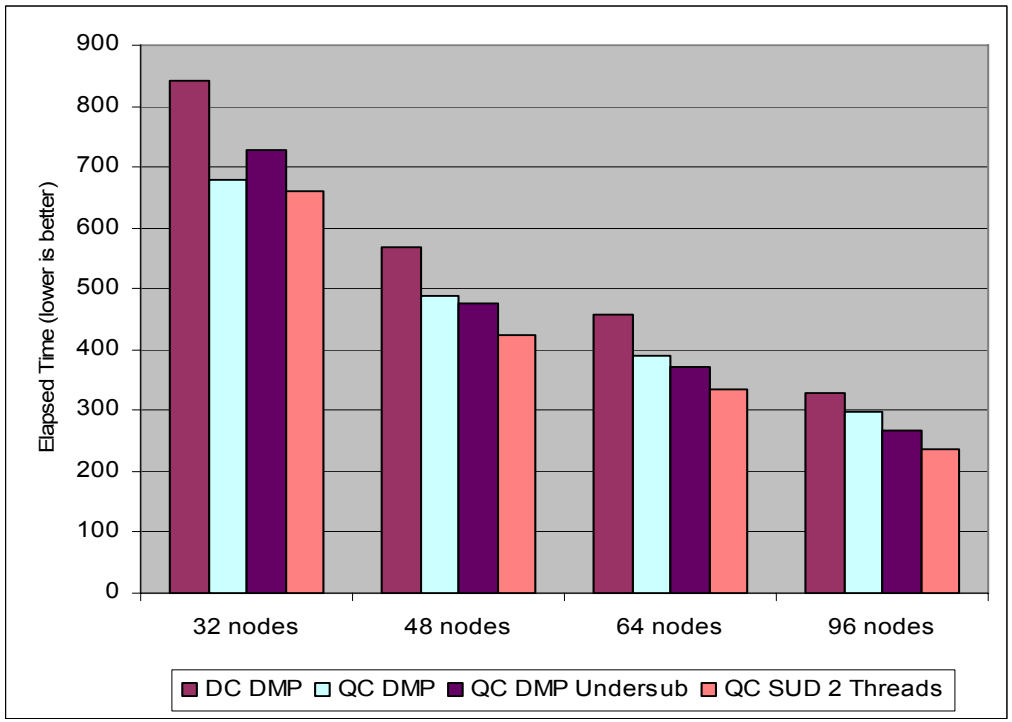
COSMO: Outside World

- COSMO-CLM
  - CLimate Mode of COSMO-Model
  - One source code is used for the weather forecast and the climate mode of the COSMO-Model
  - There are ongoing efforts to maintain this
  - Development of an Earth System Model (Coupling)
- COSMO-ART:
  - Online coupled chemistry module: Aerosols and Reactive Tracers
  - Developed at KIT: Karlsruhe Institute of Technology (former FZK)
  - Prototype exists; Code is taken over to official COSMO-Model in the near future

COSMO: Parallelization

- Some groups have worked on a hybrid parallelization of LM_RAPS using OpenMP
  - Years ago: PALLAS
  - Intel benchmarkers developed a prototype of COSMO-CLM for the DKRZ benchmark
  - Michael Riedmann from HP started a similar work together with an intern: They multi-tasked the Runge-Kutta dynamics for COSMO-DE and the turbulence part of the physical parameterizations with „inserting and debugging ~400 directives“

LM_RAPS for COSMO-DE



Courtesy of
Michael Riedmann,
Hewlett-Packard

DMP: Distributed Memory Parallel
Undersub: only use every other core
SUD: „shared under distributed“ parallelization

	32 nodes	48 nodes	64 nodes	96 nodes
Additional Cores and Cache	1,24	1,17	1,17	1,10
Undersubscription	0,94	1,02	1,05	1,12
SUD Parallism	1,10	1,13	1,11	1,13
Overall Gain	**1,27**	**1,34**	**1,36**	**1,39**

COSMO: Parallelization

- This work inspired us to make some first tests
  - Focused on the physical parameterizations
  - But inserted only 1 (in words: one) OpenMP directive
  - Used other code modifications: put together big chunks of work in the physical parameterizations and call the parameterizations for these chunks (keywords: blocking, NPROMA)
  - Up to now: Radiation, Turbulence
  - No further optimizations and tuning
  - Times (in seconds) for a small test case:

8 MPI	4 MPI; 1 Thread			4 MPI; 2 Threads		
2×4_1	1×4_1	2×2_1	4×1_1	1×4_2	2×2_2	4×1_2
22.76	45.37	45.79	46.78	23.00	23.44	24.19

COSMO: Parallelization

- The approach seems to work, but:
- Much work has to be done to build a full and efficient OpenMP implementation
- Ideas, problems, questions:
  - How to put together the chunks?
  - Can we save communication time (halo exchange, I/O)?
  - Do we need changes in memory layout for better cache (re)use without destroying vectorization?

To learn more about this, visit the 14th HPC Workshop in 2010

Or the next
RAPS Workshop

Thank you

very much

for your

attention