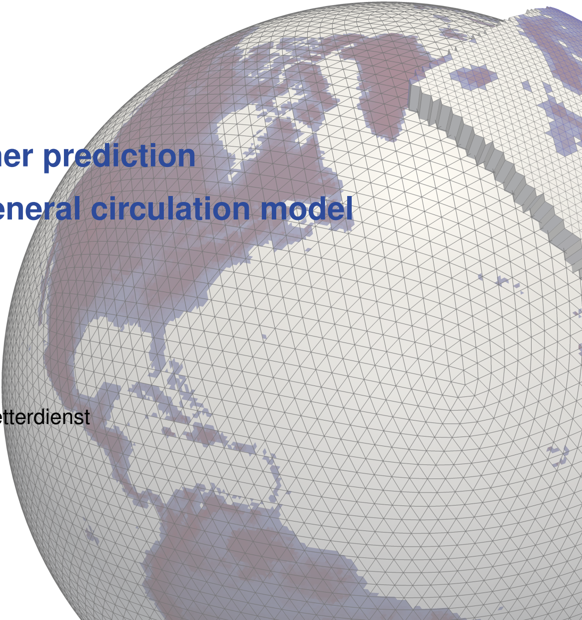
Deutscher Wetterdienst
Wetter und Klima aus einer Hand

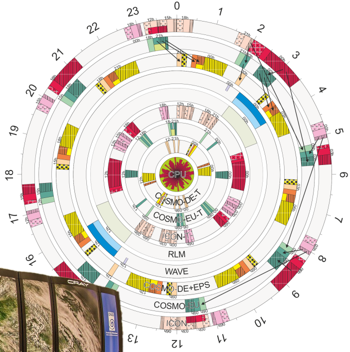


Numerical weather prediction with the ICON general circulation model

Florian Prill, Deutscher Wetterdienst

GridKa School 2015
September 7–11, 2015

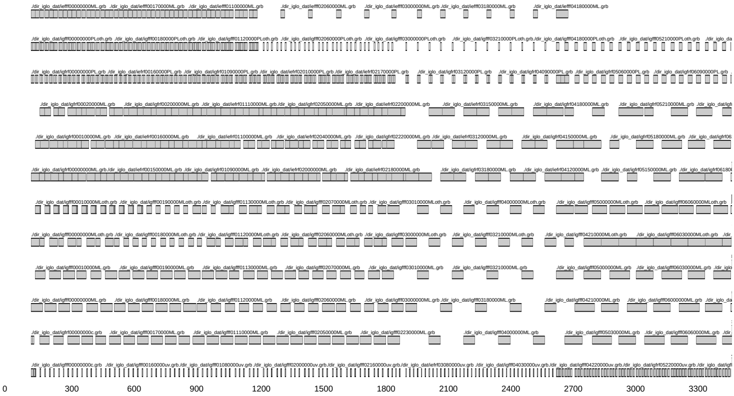
DWD – Deutscher Wetterdienst Offenbach



DWD headquarters
Offenbach

Model chain: T. Hanisch, DWD
Cray: M. Jonas, DWD

”What it’s all about”



Output schedule, global 13 km model

2015081200 run

Operational models produce 5.23 TB/day!

NWP – Numerical Weather Prediction

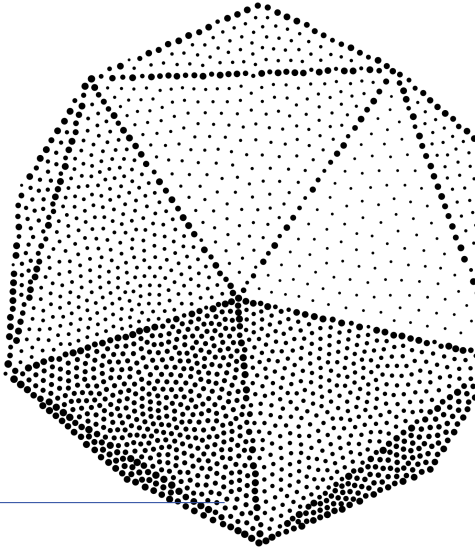
The problems:

- model complexity
- computational complexity
  size of solution vector: 300 million
  for a single unknown
- large amount of data: input/output
- chaotic dynamical system
  → ensemble solutions

... this in 24/7 operations!

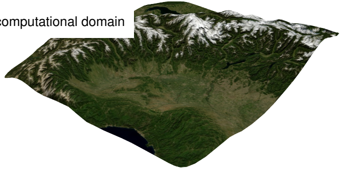
Talk outline

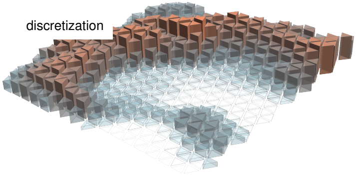
1. **Modelling**
   . . . complex multiscale nature
2. **Computational complexity**
   . . . innovation in hardware brings new challenges
3. **Data handling**
   . . . ever-growing demand for bandwidth and storage
4. Wrap-up



Modelling

Components of NWP models





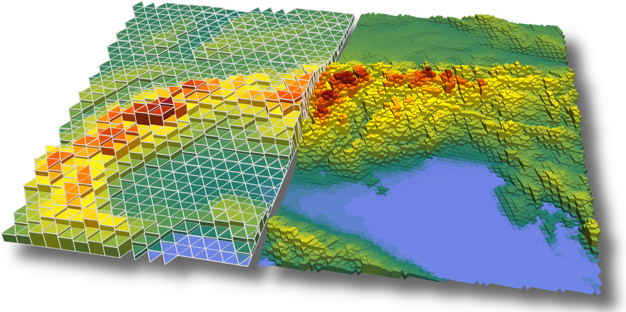
computer implementation of numerical scheme

assimilation	dynamics	physics

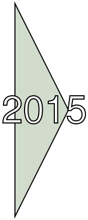
observations

NWP products

Model resolution



NWP model chain



- **GME** computes **global weather forecasts** $\leq$ 7 days
- **COSMO-EU** and **COSMO-DE**: regional components

- **ICON**:
  next-generation NWP and climate modelling system
- grid nesting in order to replace both GME and COSMO in the operational suite of DWD

Nonhydrostatic equation system (dry adiabatic)

$$
\begin{aligned}
\partial_t v_n \quad + (\zeta + f)\, v_t \quad + \partial_n K + w\, \partial_z v_n &= -c_{pd} \theta_v \partial_n \pi \\
\partial_t w \quad + \mathbf{v}_n \cdot \nabla w \quad + w\, \partial_z w &= -c_{pd} \theta_v \partial_z \pi - g \\
\partial_t \rho \quad + \nabla \cdot (\mathbf{v} \rho) &= 0 \\
\partial_t (\rho \theta_v) + \nabla \cdot (\mathbf{v} \rho \theta_v) &= 0
\end{aligned}
$$

(v_n, w, ρ, θ_v: prognostic variables)

- v_n, w: velocity components
- ρ: density
- θ_v: virtual potential temperature
- K: horizontal kinetic energy
- ζ: vertical vorticity component
- π: Exner function

Simplifying assumptions

- spherical earth
- shallow atmosphere

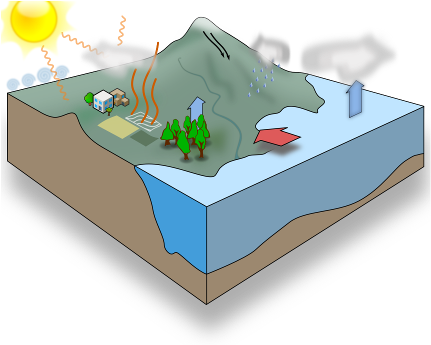
Physics parameterization

Predict (approximately) the state of the atmosphere.

Systems of differential equations
based on the laws of
physics, fluid dynamics and chemistry.

Partly represented by parameterizations

- Radiation
- Land
- Convection
- Turbulent transfer
- Microphysics
- Cloud cover
- . . .



Resolution-dependent modelling

Physics parameterization (cont'd)

Process	Authors	Scheme	Origin
Radiation	Mlawer et al. (1997) Barker et al. (2002)	RRTM (later with McICA McSI)	ECHAM6/IFS
	Ritter and Geleyn (1992)	δ two-stream	GME/COSMO
Non-orographic gravity wave drag	Scinocca (2003) Orr, Bechtold et al. (2010)	wave dissipation at critical level	IFS
Sub-grid scale orographic drag	Lott and Miller (1997)	blocking, GWD	IFS
Cloud cover	Doms and Schättler (2004)	sub-grid diagnostic	GME/COSMO
	Köhler et al. (new development)	diagnostic (later prognostic) PDF	ICON
Microphysics	Doms and Schättler (2004) Seifert (2010)	prognostic: water vapor, cloud water,cloud ice, rain and snow	GME/COSMO
Convection	Tiedtke (1989) Bechtold et al. (2008)	mass-flux shallow and deep	IFS
Turbulent transfer	Raschendorfer (2001)	prognostic TKE	COSMO
	Louis (1979)	1^{st} order closure	GME
	Neggers, Köhler, Beljaars (2010)	EDMF-DUALM	IFS
Land	Heise and Schrodin (2002), Machulskaya, Helmert, Mironov (2008, lake)	tiled TERRA + FLAKE + multi-layer snow	GME/COSMO
	Raddatz, Knorr	JSBACH	ECHAM6

Model assessment: Verification

Modifications of NWP model are verified in a sandbox (“Parallelroutine”) before being introduced in operational suite.

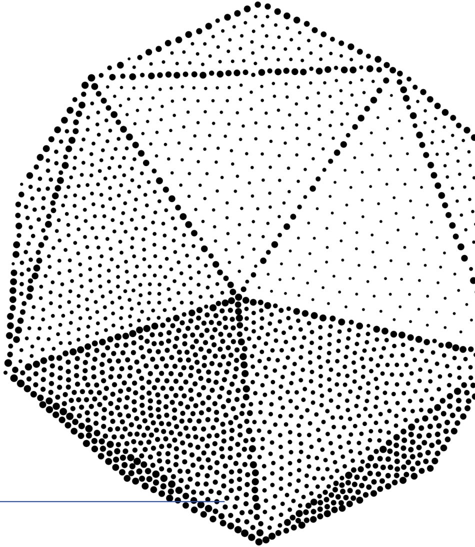
World Meteorological Organization (WMO) standard verification for assessment of NWP suite

$$\mathrm{BIAS} = \overline{\mathbf{F} - \mathbf{A}} \qquad \mathrm{STDV} = \overline{\left[\mathbf{F} - \mathbf{A} - \overline{\mathbf{F} - \mathbf{A}}\right]^2}$$

$$\mathrm{ABSE} = \overline{|\mathbf{F} - \mathbf{A}|} \qquad \mathrm{ANOC} = \frac{\overline{\left[\mathbf{F} - \mathbf{R} - \overline{\mathbf{F} - \mathbf{R}}\right]\left[\mathbf{A} - \mathbf{R} - \overline{\mathbf{A} - \mathbf{R}}\right]}}{\sqrt{\overline{\left[\mathbf{F} - \mathbf{R} - \overline{\mathbf{F} - \mathbf{R}}\right]^2}\;\overline{\left[\mathbf{A} - \mathbf{R} - \overline{\mathbf{A} - \mathbf{R}}\right]^2}}}$$

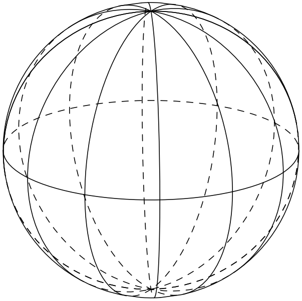
$$\mathrm{RMSE} = \sqrt{\overline{(\mathbf{F} - \mathbf{A})^2}} \qquad \mathrm{SKS1} = 100\,\frac{\sum |\mathbf{G}_F - \mathbf{G}_A|}{\sum \max\left(|\mathbf{G}_F|, |\mathbf{G}_A|\right)}$$

Computational Complexity



Model discretization

Regular latitude-longitude grids lead to clustering of grid lines and reduced grid spacing at the poles.



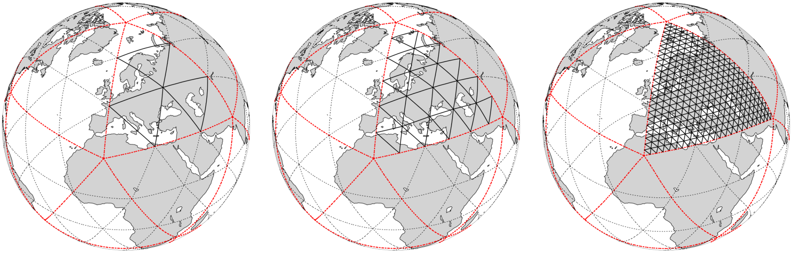
- This creates a severe Courant-Friedrichs-Lewy (CFL) restriction on the time step

  CFL number $C = u \frac{\Delta t}{\Delta x}$

- Non-scalable inter-process communication

Quasi-uniform grids

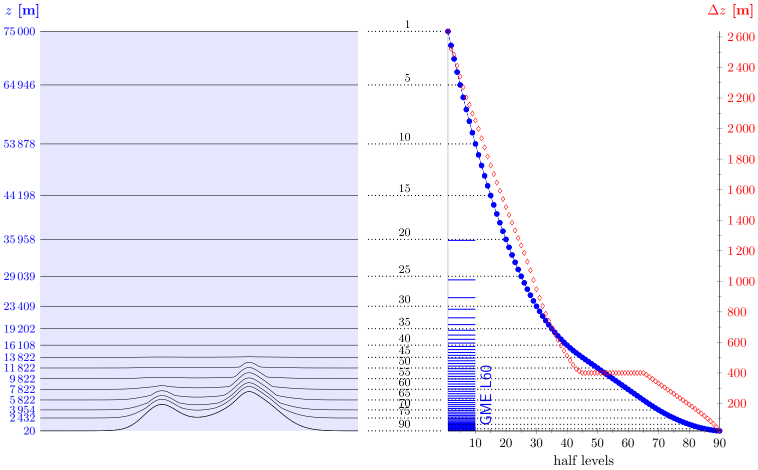
Tiling the surface with triangles avoids the pole problem.
DWD ICON model: Spherical geodesic grids derived from icosahedron



R3B00 **R3B01** **R3B03 optimized**

Example: R3B7 grid, 13 km global resolution
$\approx 2.95 \cdot 10^6$ spherical triangles

Smooth Level Vertical (SLEVE) Coordinate



Grid structure with nested domains

Resolution-dependent modelling with (two-way-) nesting capability for multiple non-overlapping nests per nesting level.

Example:

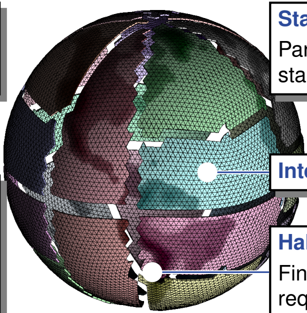
13 km global × 90 levels
+ 6.5 km Europe nest
× 60 levels
≈ 305 million grid points.

Distributed-memory partitioning

Model performance is determined by balance of workload, communication and memory consumption.

Domain decomposition
Each MPI task operates on a separate partition.

Static load balancing
Partitioning fixed at start-up.



Interior of partition

Geometric subdivision
Recursive lat/lon bisection
For radiation grids each task comprises shadowed and sunlit parts.

Halo region
Finite difference stencils require communication.

Cost to update the boundary is $\mathcal{O}(1)$,
i.e. communication with only a small number of neighbours.

MPI – Message Passing Interface

- standard for data-parallel applications on CPUs
- current version 3.0 (Sept. ’12)
- basic concept: communicating processes,
  data exchange via sending/receiving message blocks

```
CALL MPI_INIT(ierr)
if (rank == from) then
   CALL MPI_SEND(data_buf, msg_size, MPI_REAL, to, tag, &
        &         comm, ierr)
   if (ierr /= 0) write (*,*) "MPI Error!"
else
   CALL MPI_RECV(data_buf, msg_size, MPI_REAL, from, tag, &
        &         comm, status, ierr)
   if (ierr /= 0) write (*,*) "MPI Error!"
end if
CALL MPI_FINALIZE(ierr)
```

OpenMP – Shared memory programming

- standard for shared memory parallelization
- current version 4.0 (Juli ’13)
- parallelization on thread/loop level
- compilers may ignore OpenMP directives, applications still run

```
!$OMP PARALLEL
!$OMP DO PRIVATE(i,t)
    DO i=1,N
      t = intp_data(i)
      intp_data(i)%wgt  = t%wgt/area(t%didx,t%dblk)
    END DO
!$OMP END DO
!$OMP END PARALLEL
```

Hybrid MPI-OpenMP parallelization

Scalability: capability of an algorithm to handle a growing amount of work and a growing number of participating compute nodes.

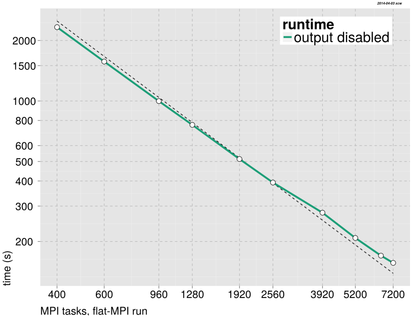
- Underlying numerical method is fundamentally scalable
- Scalability is limited by
  - quality of the implementation of the application
  - scalability of the computing platform
- **Hybrid parallelization:** The implementation of the ICON model uses the MPI library and OpenMP for parallelization.

Strong scaling characteristics

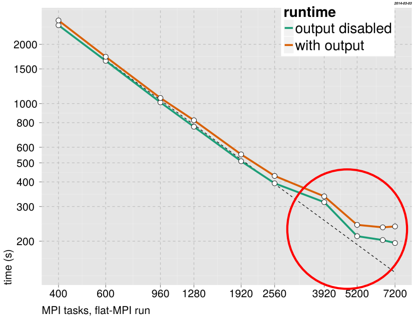
Fixed size problem is executed on variety of core counts.
For a code with ideal strong scaling, use of twice the number of cores will reduce the execution time in half.

Strong scaling characteristics

Cray XC West



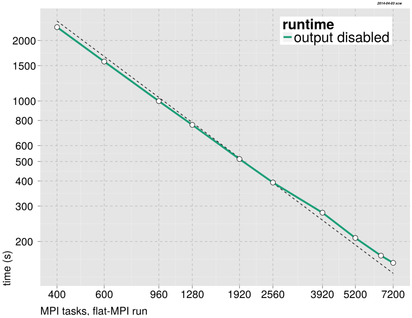
Cray XC East



Test setup: **ICON real data setup**
13 km global resolution, 24 h forecast,
with reduced radiation grid

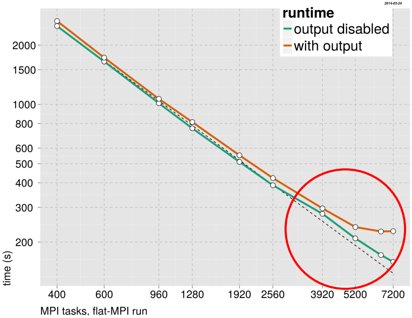
Strong scaling characteristics

Cray XC West



Test setup: **ICON real data setup**
13 km global resolution, 24 h forecast,
with reduced radiation grid

Cray XC East . . . after hardware fix



Hardware-oriented programming

code balance

ratio of memory data traffic to arithmetic work

usually much larger than

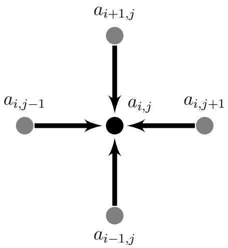
machine balance

ratio of memory bandwidth to peak arithmetic performance

Important class of loops in NWP codes:
iterative stencil loops

Each matrix element is updated based on the values of its neighbouring elements.

→ potential to reuse cached data
→ efficient use of caches is important



Hardware-oriented programming

code balance

ratio of memory data traffic to arithmetic work

usually much larger than

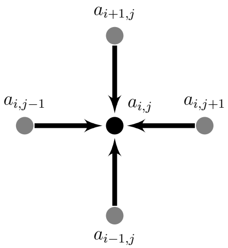
machine balance

ratio of memory bandwidth to peak arithmetic performance

Important class of loops in NWP codes:
iterative stencil loops

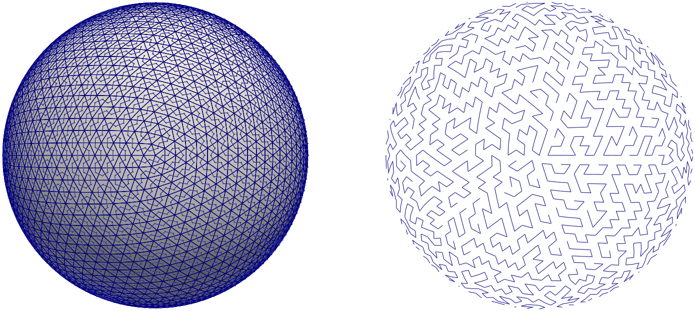
Each matrix element is updated based on the values of its neighbouring elements.

→ potential to reuse cached data
→ efficient use of caches is important



Stencil evaluation idea #1: Stripified stencils

- Hamiltonian path heuristics from 3D computer graphics rendering: Reorder cells, edges and vertices in stripes.
- Beneficial effects on cache misses, reduction of data-read operations.



Idea #1: Stripified stencils (cont'd)

```
DO j=1,N
  div(j) =  a(j)              * c(j,1)
          + a(neighbor(j,1)) * c(j,2)
          + a(neighbor(j,2)) * c(j,3)
          + a(neighbor(j,3)) * c(j,4)
END DO
```

- stencil computation with stripes:
  80.99% of standard runtime (Intel i7-4790, `gfortran O2`)

Idea #1: Stripified stencils (cont'd)

```
DO j=0,N,blocklength
  v1 = a(neighbor(j,1))
  v2 = a(j)
  DO k=j+1,j+blocklength
    v3 = a(neighbor(k,3))
    div(j) =  v2               * c(k,1)
            + v1               * c(k,2)
            + a(neighbor(j,2)) * c(k,3)
            + v3               * c(k,4)
    v1 = v2
    v2 = v3
  END DO
END DO
```

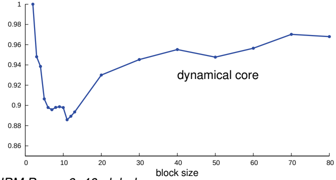
- Where it does **not** work:
  stencils with horizontal + vertical loops !!!

Stencil evaluation idea #2: Loop tiling (loop blocking)

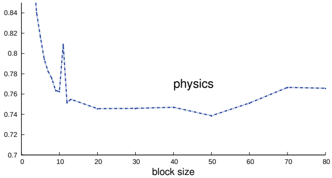
Well-known loop transformation to improve cache utilization.

- total size of the data exceeds the cache
- therefore: partition iteration space into loop tiles s.t. accessed array chunks fit into cache line
- tiling size = "automatic" optimization for a range of platforms

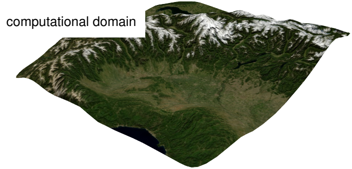
```
DO i=0,N,nproma
  DO j=i+1,(i+nproma)
    ...
  END DO
END DO
```

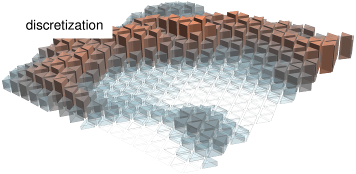


IBM Power 6, 40 global



Components of NWP models (revisited)





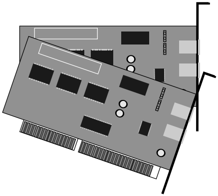
computer implementation of numerical scheme

assimilation	dynamics	physics

observations

NWP products

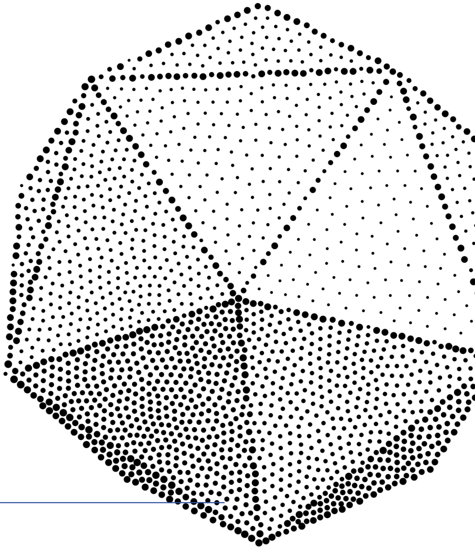
New architectures: GPGPUs



General Purpose Computation on Graphics Processing Unit (GPGPU):
Using graphics processors for parallel simulations

- host program on the CPU controls data flow to device
- high memory bandwidth
- very high degree of parallelization

- Pro: relatively cheap
- Con: no unified programming API established
- loop tiling is not beneficial here, new strategies are required!

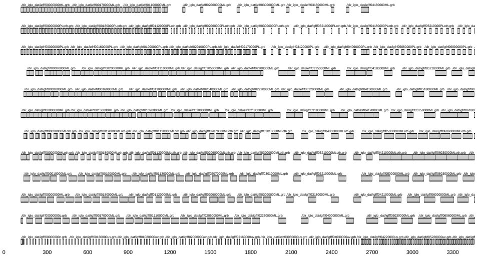


Data handling

Data handling: Critical issues

Before being post-processed into meteorological products, NWP results must be stored!

- bandwidth, performance
  convenient data formats and projections
- archiving of high-resolution data



Output schedule, global 13 km model

2015081200 run

Convenient data formats

GRIB2 data format ("GRIdded Binary")

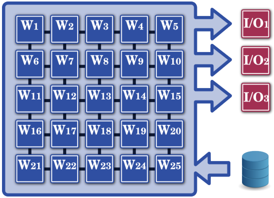
- Defined by the World Meteorological Organization (WMO), in use operationally worldwide
- GRIB version 2 supports both structured and unstructured model data, however standardization of unstructured GRIB records is relatively new.
- **Pros:** automatic data compression, rigid enforcement of proper metadata
- **Cons:** sequential records, complex read/write of metadata section

NetCDF storage format

- storage format for data arrays and attributes, structured and unstructured
- self-describing, machine independent
- **Pros:** may contain essential fields of the ICON grid topology
- **Cons:** GRIB2 storage $\approx$ 50% file size compared to NetCDF

Output products

- prognostic variables
- diagnostic data fields
- meteograms
- total integrals, e.g. computation of tracer mass error



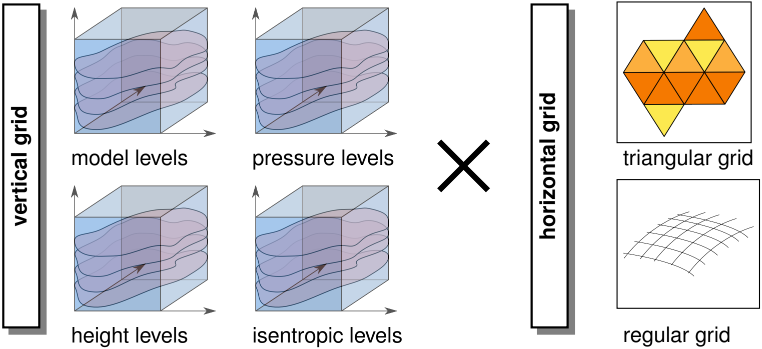
Model output specified for domains, subregions and time ranges.

Processors are divided into

- **Worker PEs** majority of MPI tasks, doing the actual work
- **Output PEs** dedicated I/O server tasks
- **Restart PEs** for asynchronous restart writing

Internal post-processing

Different regridding procedures for meteorological quantities.

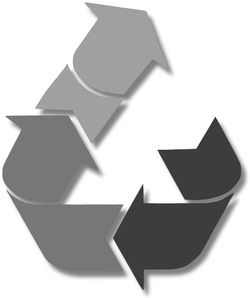


Levels of output accuracy

Need for different levels of accuracy:

- **Lossy compression**
  of prognostic and diagnostic data fields

- **"Defensive I/O"**
  to handle model crashes.
  The checkpoint/restart option allows to restart the execution from a pre-defined point using the data stored in a checkpoint file.



Data provenance

In the computational geosciences data provenance is crucial for the **reproducibility and the analysis of defects**.

One important building block:
Documentation of the external parameter sets and the computational meshes that have been used for production.

Object identifiers provided by GRIB2 can be used for example for fingerprinting the Cartesian coordinates of the mesh.
This way it is possible to identify the underlying grid for

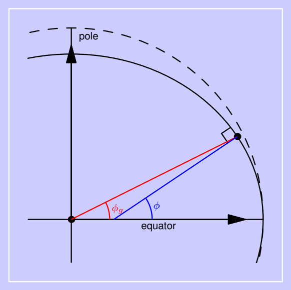
- external parameter files
- analysis data for forecast input
- data files containing the diagnostic output
- checkpointing files

One more thing ...

In most numerical models the earth is represented as a sphere of constant radius. But the earth is flattened slightly at the poles!

geographic latitude angle which a line perpendicular to the surface makes with the plane of the equator

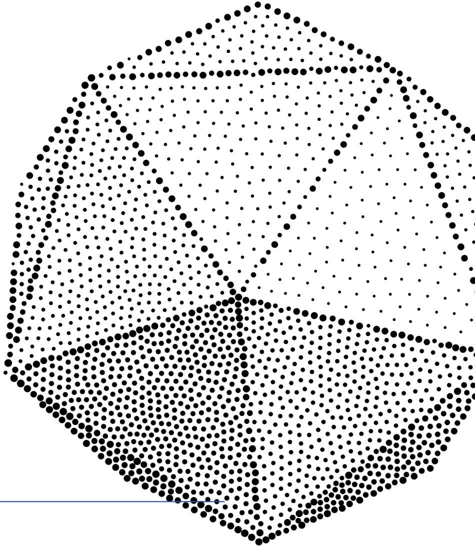
geocentric latitude angle made by a line to the earth's center



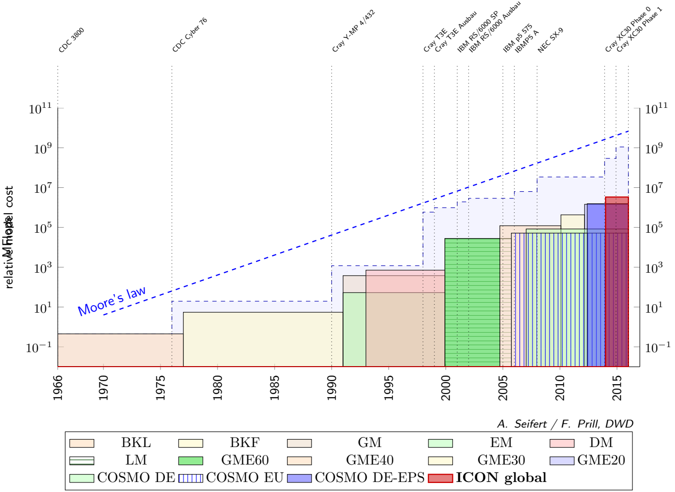
Input data and post-processing are often based on ellipsoids (e.g. WGS84). Differences up to several kilometres!

Careful conversion between geographic and geocentric coordinates is necessary.

Conclusion



Growth of performance and model cost at DWD



Wrap-up

- *(Evolution of the)*
  DWD model chain
- **Challenges:**
  modelling, computational complexity, data
- **Need to bring together**
  scientists from different disciplines

Thank you!

Florian Prill

Met. Analyse und Modellierung
Deutscher Wetterdienst

e-mail: Florian.Prill@dwd.de