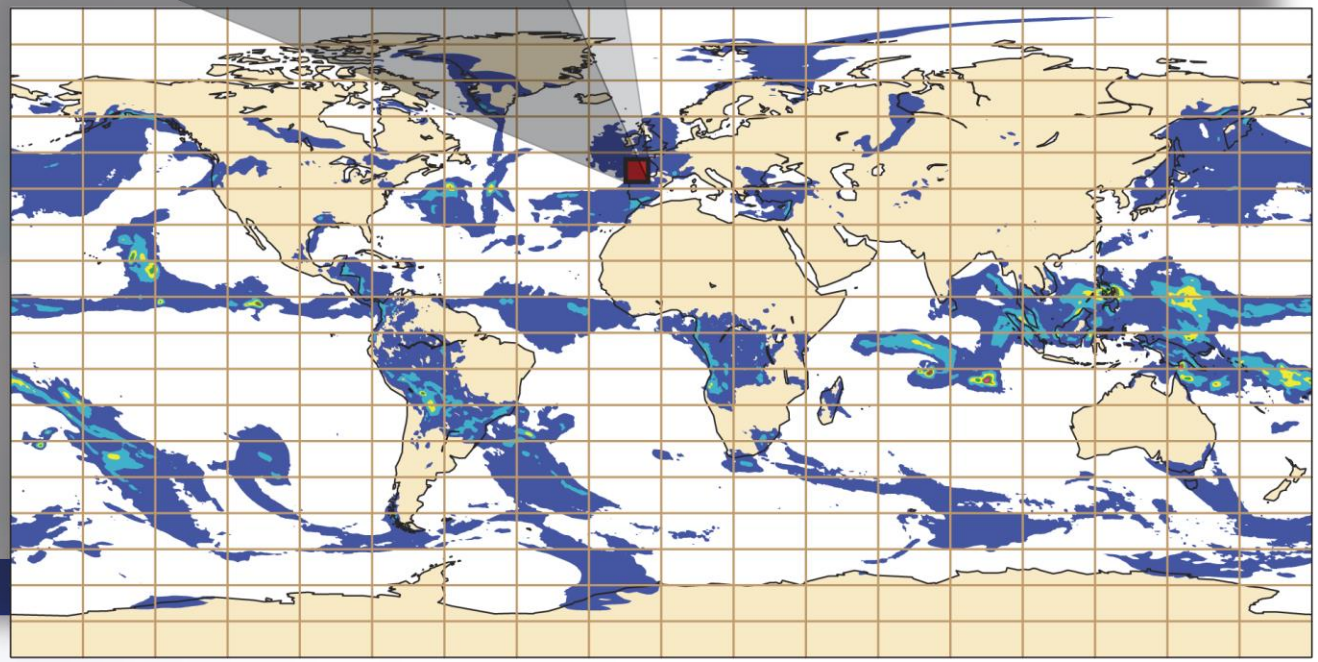
GRIB ⇔ NetCDF
Setting the scene

GRIB at ECMWF

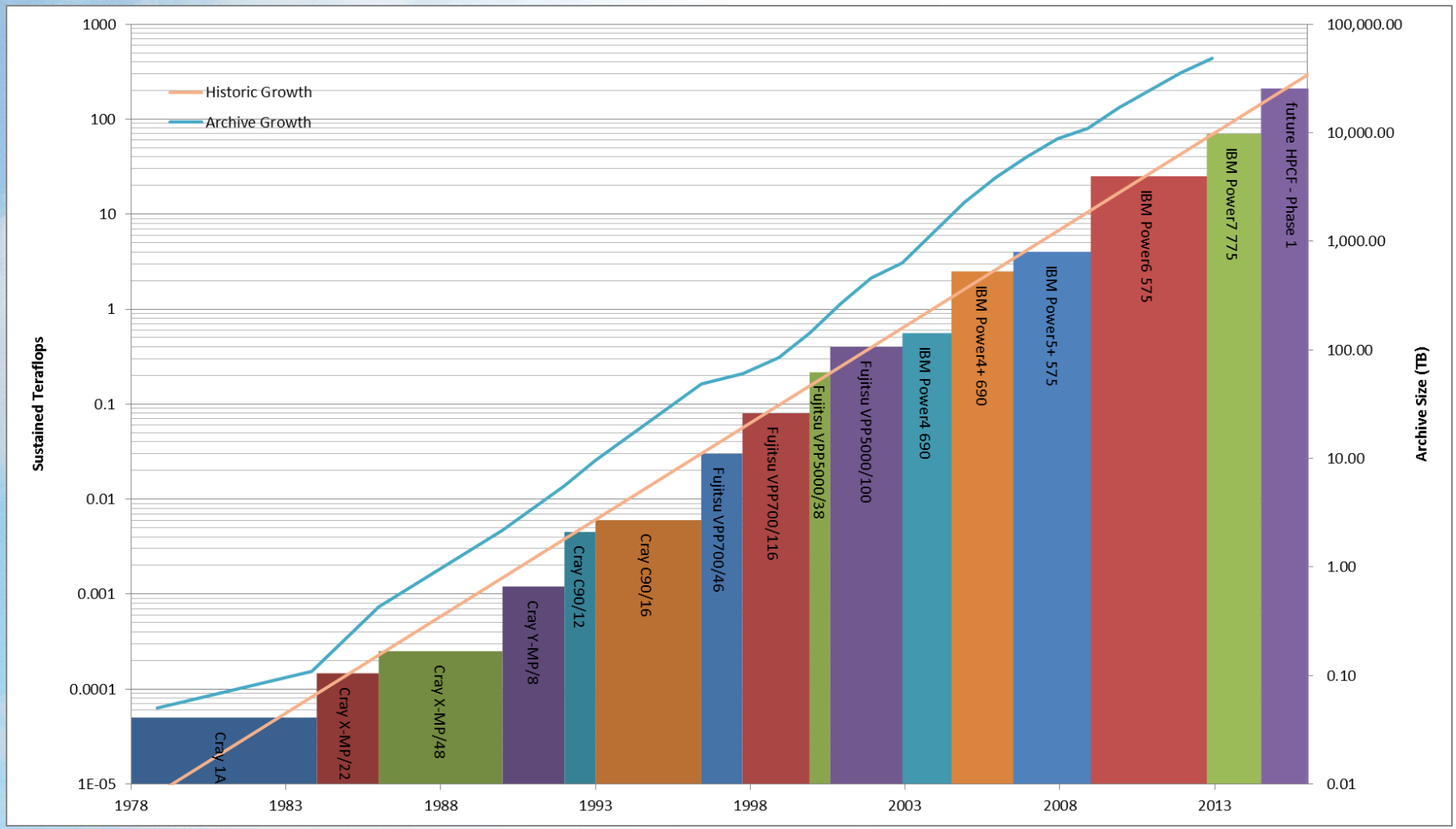
ECMWF model output are encoded in GRIB

- One parameter
- One date
- One time
- One step
- One level
- One forecasting system
- …

7.2	9.9	3.6	0.4	8.3	0.2	0.5	0.1	9.1	6.7
0.3	8.8	1.8	0.5	0.3	0.1	2.7	0.1	7.9	6.9
7.1	9.2	3.6	0.4	8.3	0.2	6.5	3.3	5.5	5.3
2.2	1.1	1.7	0.7	3.5	2.4	0.8	1.9	9.0	6.7
5.1	0.9	1.9	8.9	5.9	0.4	1.5	2.0	7.7	0.7
6.2	0.4	1.4	9.8	9.9	7.7	0.9	3.2	7.2	4.8
8.1	1.4	4.4	0.4	0.3	7.2	3.5	3.4	1.1	9.7
7.0	3.6	4.9	0.7	6.8	1.2	0.1	2.2	6.6	6.0
0.2	7.7	3.6	3.1	8.6	0.5	9.5	0.8	5.6	5.0
3.2	7.2	3.1	0.4	0.9	0.3	0.7	0.4	0.2	0.0



Size of the archive vs. Sustained HPC performance



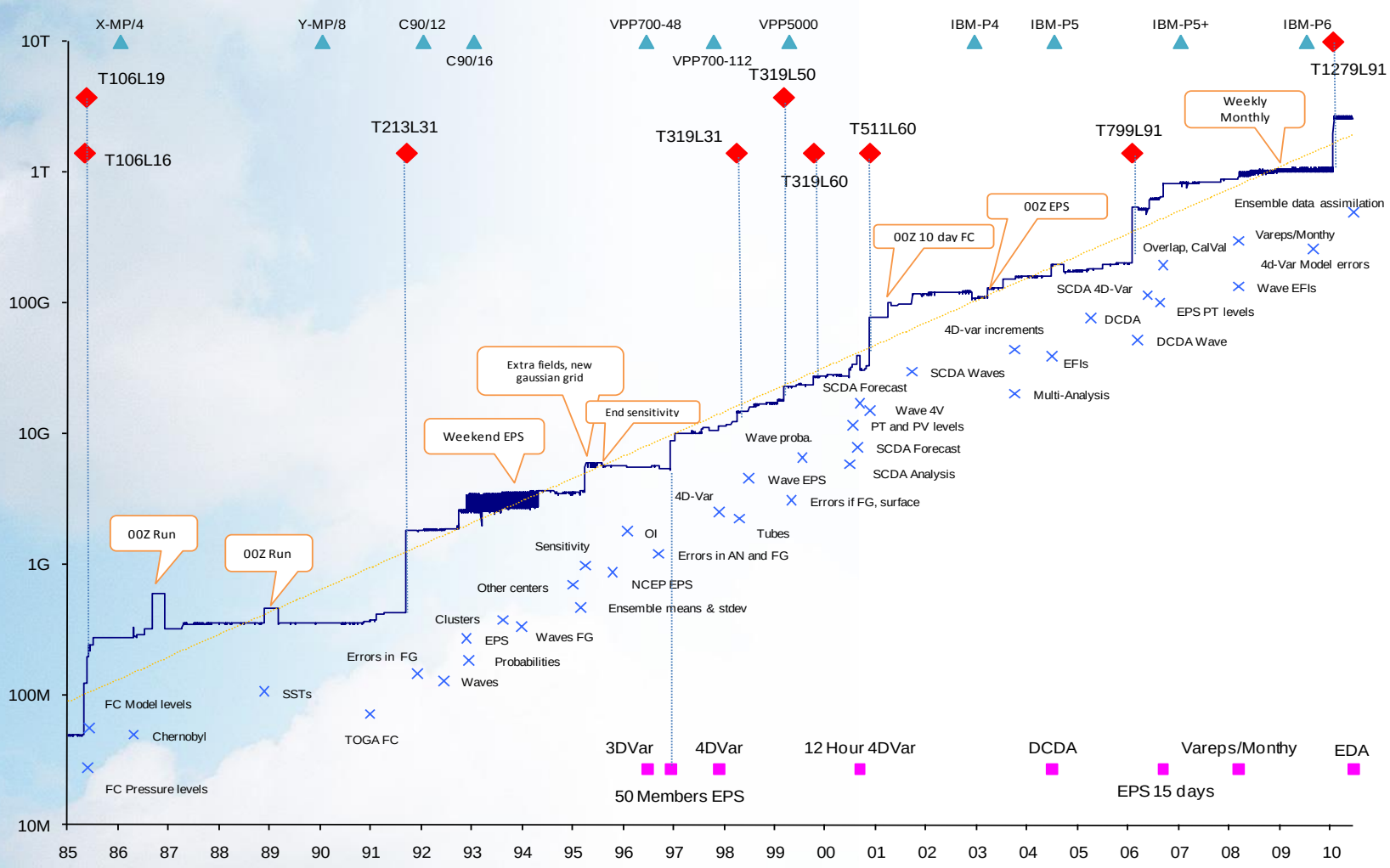
GRIB are stored in MARS

- 28 years in existence
- A managed archive
- MARS is not a file system
  - Users are not aware of the location of the data
  - Retrievals are expressed in meteorological terms
- An archive, not a database
  - Metadata online
  - Data offline (automated tape library)

MARS in numbers

- 53 Petabytes of primary data in ~ 11 million files, for more than 170 billion ($1.7 \cdot 10^{11}$) meteorological fields
- ~ 800 Gigabytes of metadata
- 200 million fields added daily (peaks at 100 Terabytes)
- 650 active users/day executing 1.5 million requests/day
- ~ 100 Terabytes retrieved daily

Addition of data type in MARS



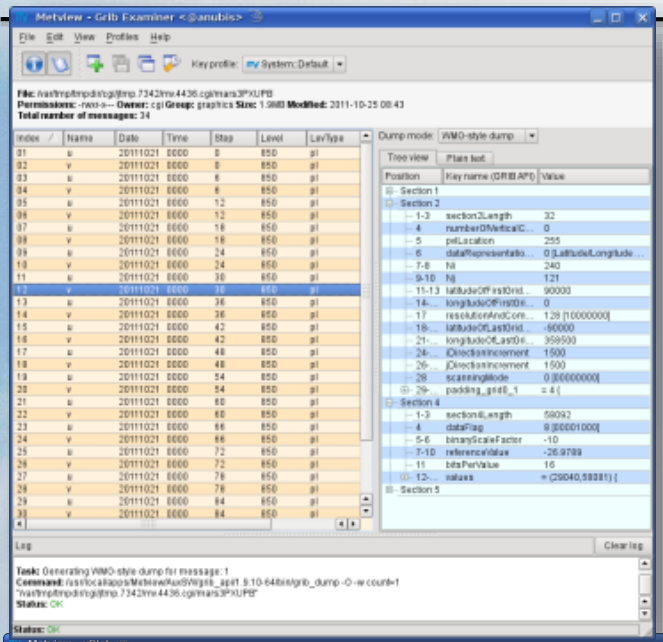
What data?

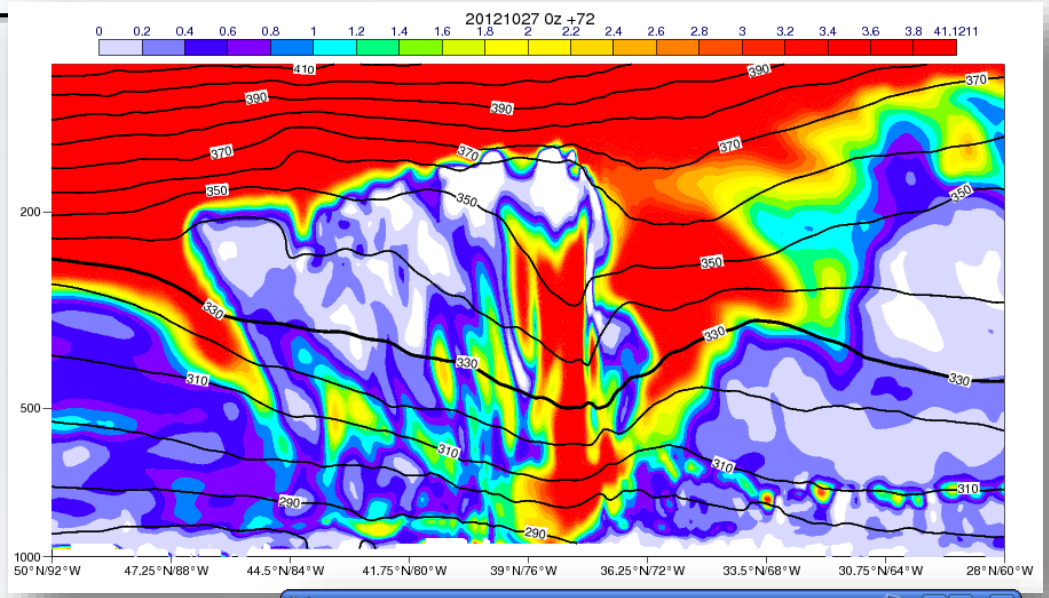
- Operational runs
  - Medium-range (15 days, twice a day, including ensemble)
  - Extended-range (a month), Long-range (a year)
  - Re-forecasts , Ocean waves
- Projects
  - Reanalyses (15 years, 45 years, 100 years)
  - WMO: TIGGE, TIGGE-LAM, S2S
  - EU projects: DEMETER, ENSEMBLES, EURO4M, MACC, PROVOST, ECSN…
- Research experiments
  - ECMWF, Member States
- Member States' Projects
  - COSMO-LEPS , Aladin-LEAF

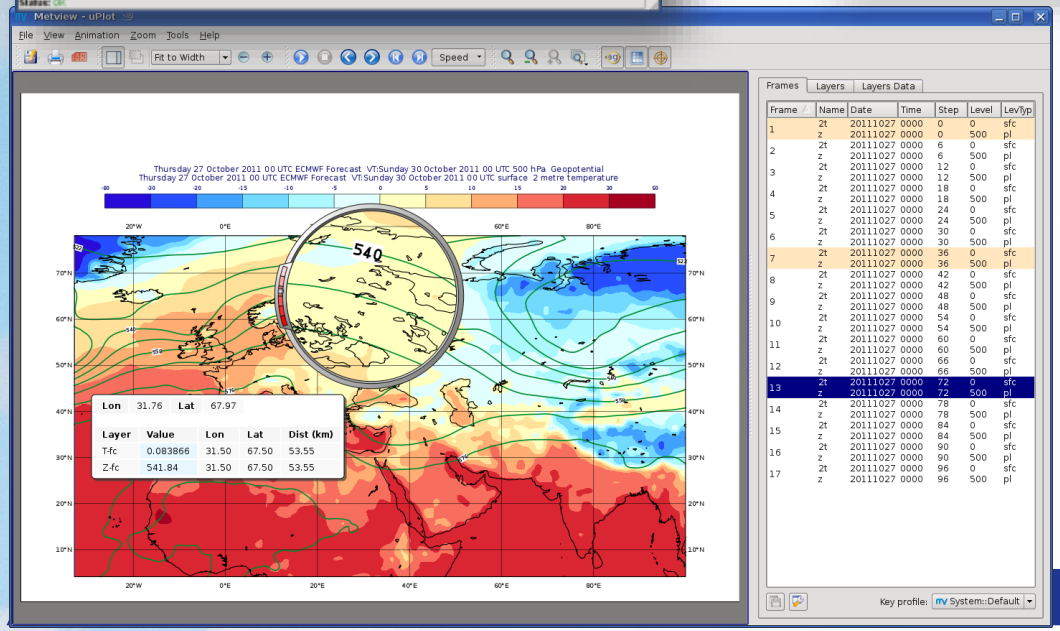
MARS language – Retrieve request

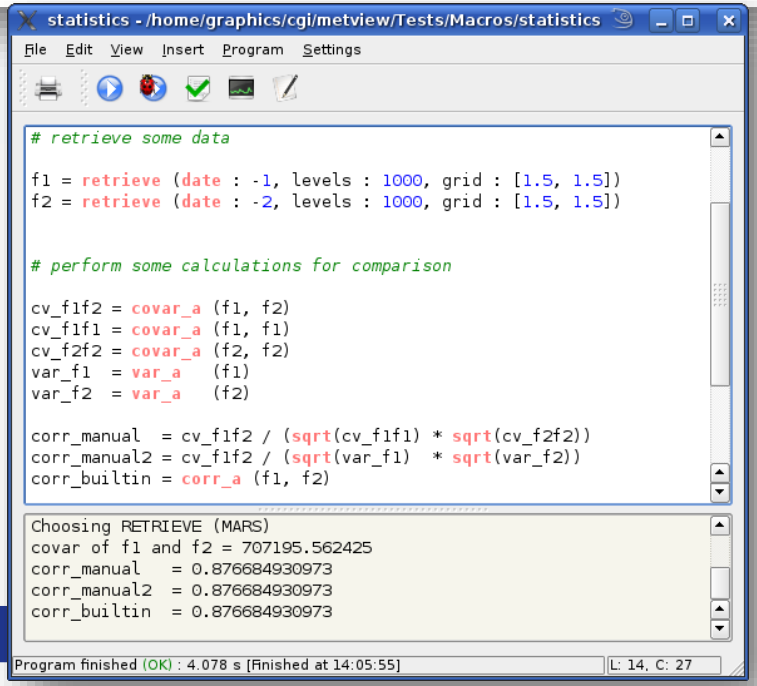
```
retrieve,                                   action
    class     = od,                             identification
    stream    = oper,
    expver    = 1,

    date      = -3,                             date & time related
    time      = 12,

    type      = analysis,                       data related
    levtype   = model levels,
    levelist  = 1/to/137,
    param     = temperature,

    grid      = 2.5/2.5,                        post-processing

    target    = "analysis"                      storage
```

Metview and GRIB

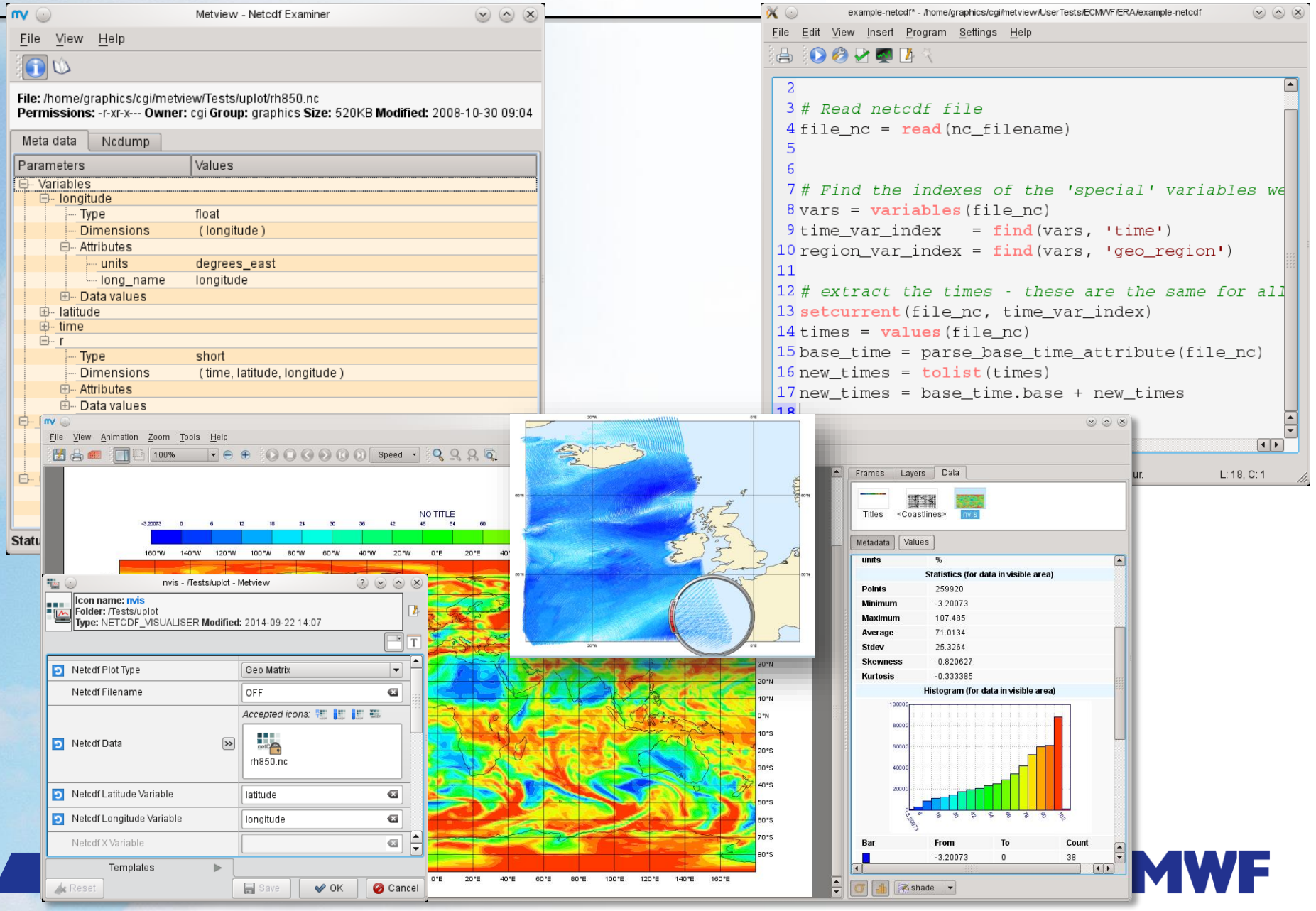








Metview and NetCDF



But we want more…

We want to archive NetCDF in MARS

…and provide a service on par with what we do for GRIBs.

GRIB ⇔ NetCDF
Setting the scene (part 2)

Disclaimer

GRIB vs. NetCDF: I have no preferences

I know GRIB better, that’s all.

GRIB

- Designed for telecoms
- As small as possible, table driven (must to read the doc :-)
- Record/message format, in memory
- No separation between format and data model
- Used in operational NWP, exchanged on the GTS
- Designed by committee

- No such thing as a “GRIB file”, just a file with GRIB messages

```
% cat file1.grib file2.grib > file3.grib
```

- `file3.grib` is a valid “GRIB file”

NetCDF

- Primarily an API/library
- Self describing
- Needs a convention (CF)
- Clean separation between format and data model (CF)
- Loads of tools
- Used in academia, oceanography and climate modeling
- CF: community driven

- NetCDF is a file format:

```
% cat file1.nc file2.nc > file3.nc
```

- `file3.nc` is NOT a valid NetCDF file.

NetCDF vs. GRIB/BUFR

NetCDF is an envelop

CF: data stored using a convention

GRIB/BUFR: data and envelop are mixed

Converting from GRIB to NetCDF

- Metadata
  - Date and time
  - Parameters
- Data
  - Units
  - Grids
- Compression
  - Internal (Packing)
  - External (zlib)
- File structures

Metadata

Why do we need metadata?

- Type 1: we cannot use the data otherwise
  - E.g. description of the grid (latitudes, longitudes)
  - E.g. units
- Type 2: identification (used for indexing, use for querying)
  - E.g. date/time
- Type 3: Nice to have
  - E.g. contact details of principal investigator
- GRIB => NetCDF
  - How to map this metadata?
  - How to map the data?

Convention? What convention?

- Parameter names are well covered by CF.
- What about other attributes:
  - lat/lon?
  - lat/long?
  - latitude/longitude?
  - x/y?
  - y/x?
- And:
  - lev?
  - level?
  - height?
  - z?
- I have seen them all. Are users supposed to inspect new files before using them?

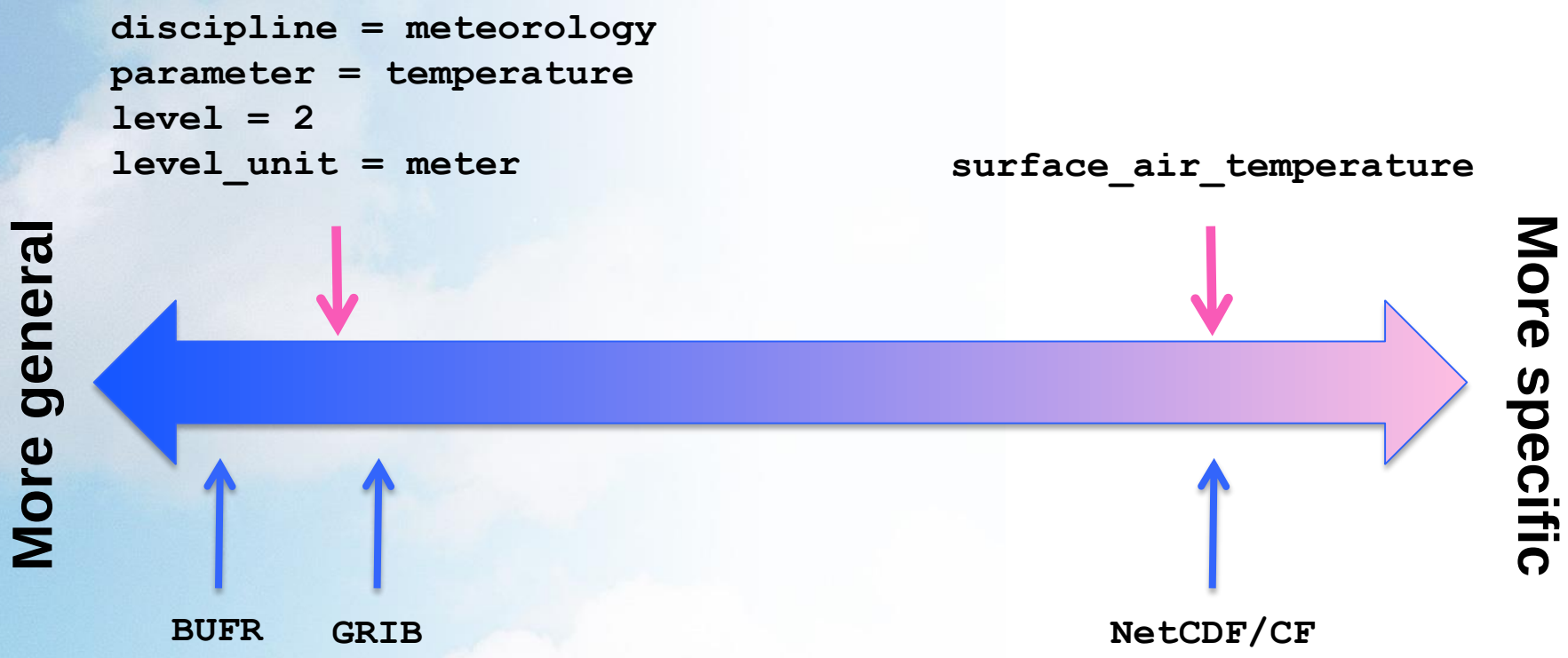
Units

Should covertion modify the data?

- Example: Total Precipitations
  - NetCDF: kg m^{-2} (e.g. mm assuming 1l of water = 1km)?
  - GRIB: m m^{-2} ?
  - Mapping implies multiplication by 1000
- Is that acceptable?

Parameter names

“Semantic Spectrum”



Computer “friendly”

Human friendly

Two interesting examples:

```
tendency_of_atmosphere_mass_content_of_
particulate_organic_matter_dry_aerosol_
expressed_as_carbon_due_to_emission_from_
residential_and_commercial_combustion
```

```
surface_upward_mass_flux_of_carbon_dioxide_
expressed_as_carbon_due_to_emission_from_
fires_excluding_anthropogenic_land_use_change
```

File structure

File structures

- How to structure NetCDF files?

- ECMWF golden rule:
  - File must be self describing
  - File name MUST NOT carry any semantic

- Consider:

```
% mv ECMWF-ERA20C-geopotential-20010101.nc foo.nc
```

- What is in foo.nc ?
  - `ncdump` (or `grib_dump`) should tell us

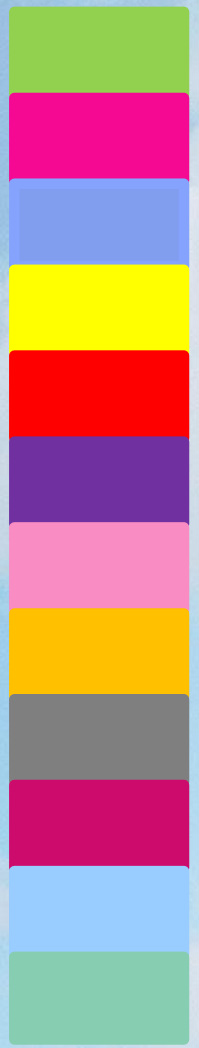
So how do we structure a NetCDF file?

- One 2D field per file?
- One 3D field per file?
- Many 3D fields per file?
- Many 4D files per files?

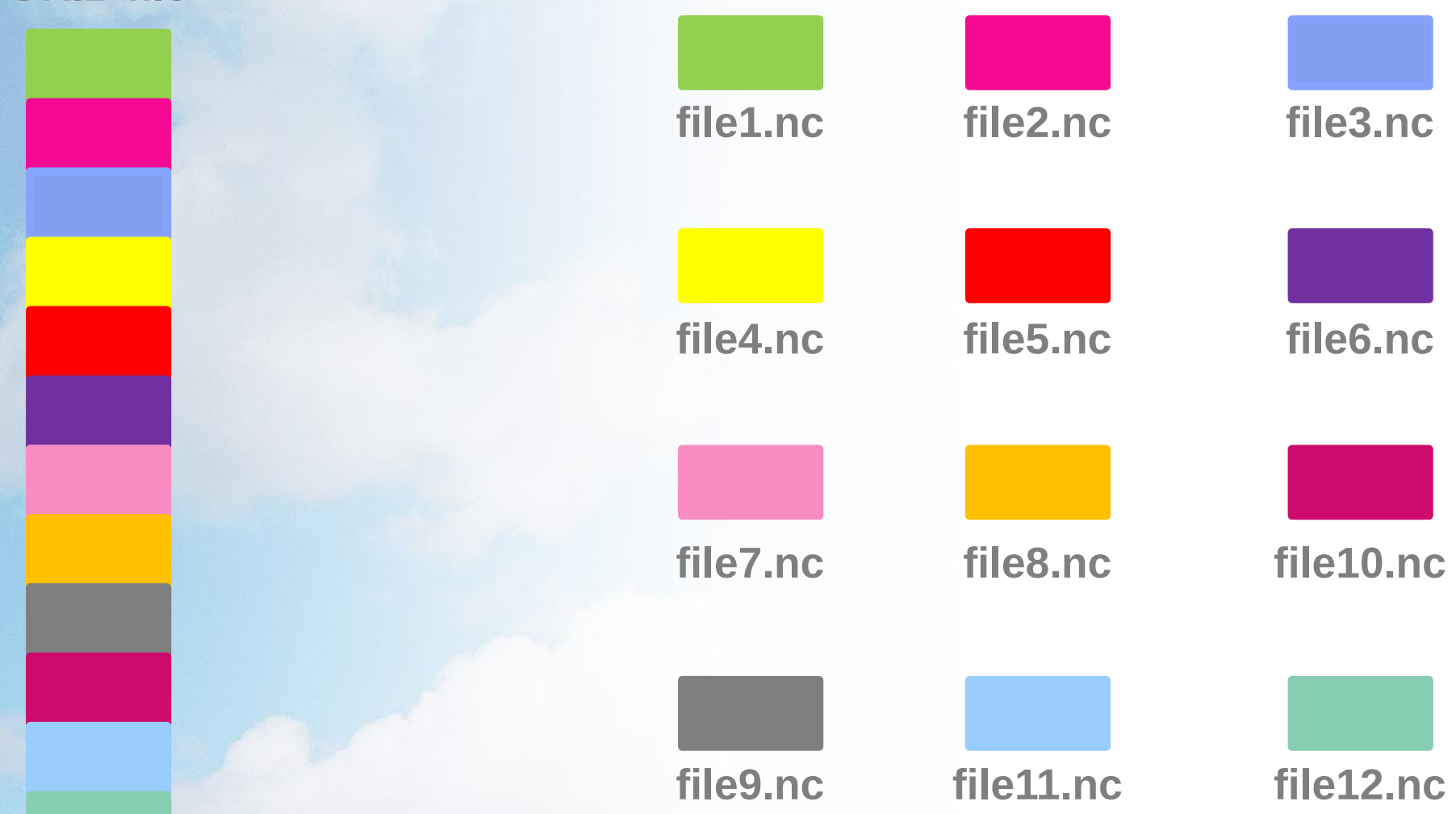
=> See effect on packing

File structure: the problem: how to map to NetCDF files?

GRIB file



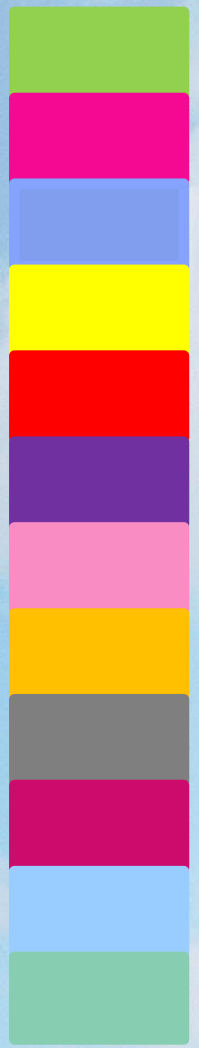
File structure: one file per field



File structure: group by time? By level? Both?

File structure: group by time? By level? Both?

GRIB file



NetCDF file

A =

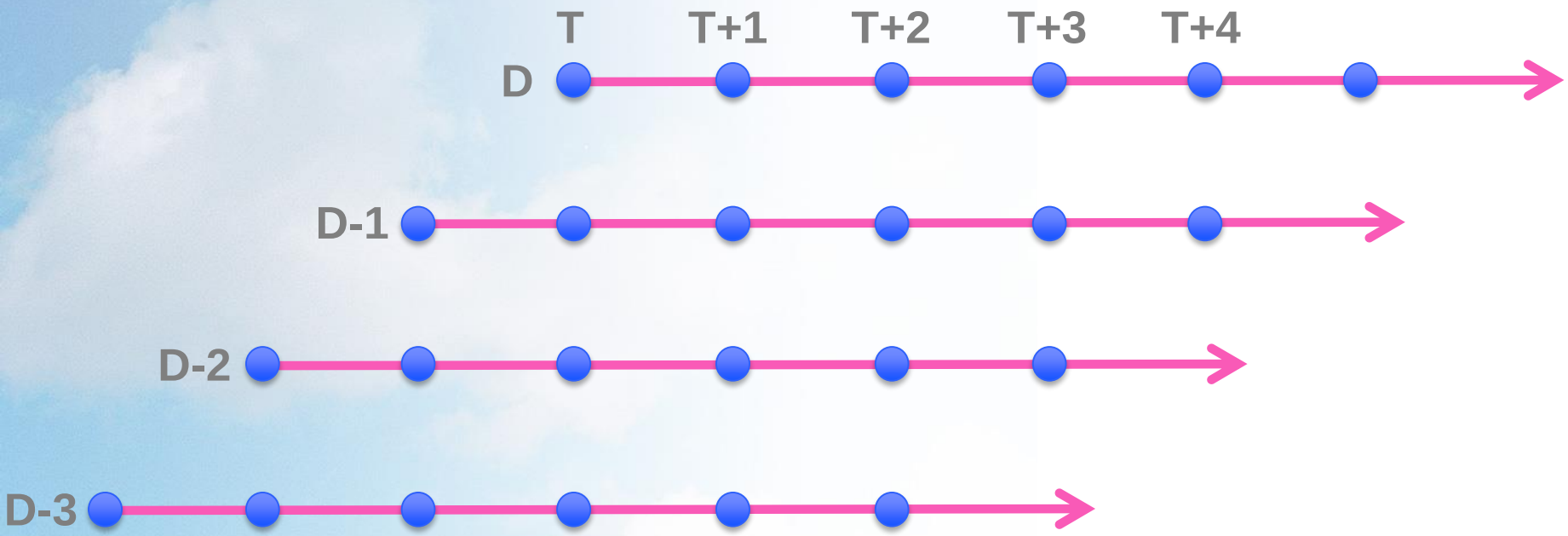
B =

C =

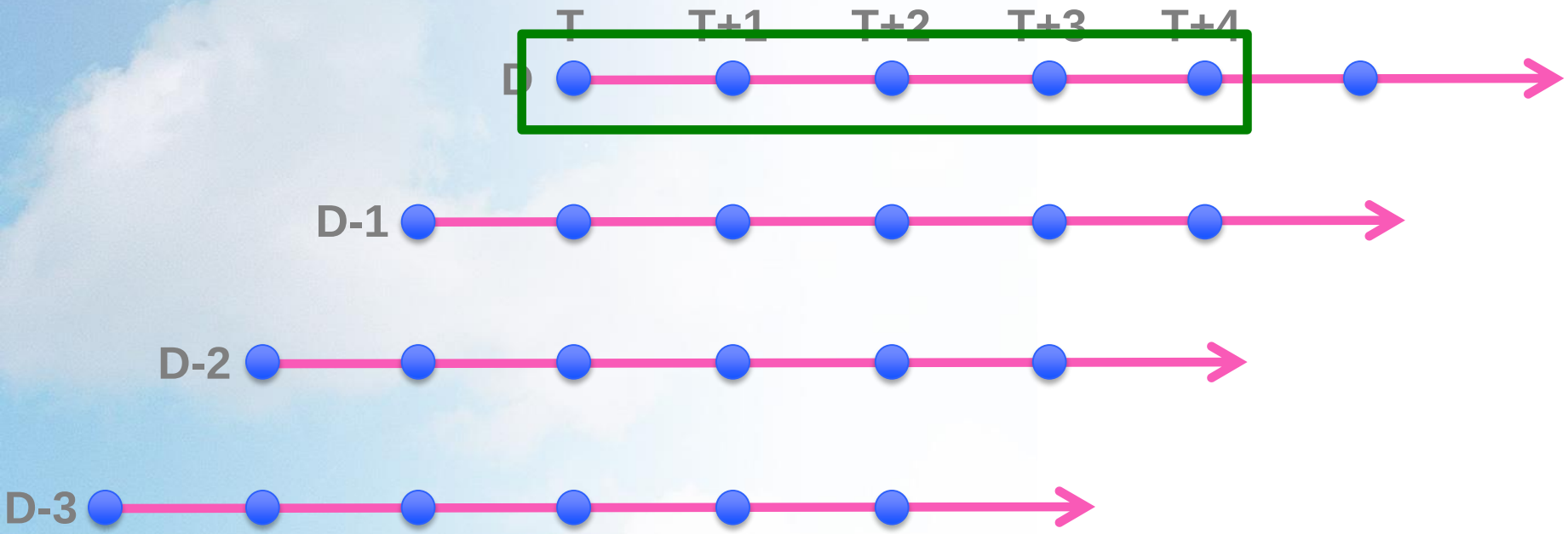
D =

Date and time

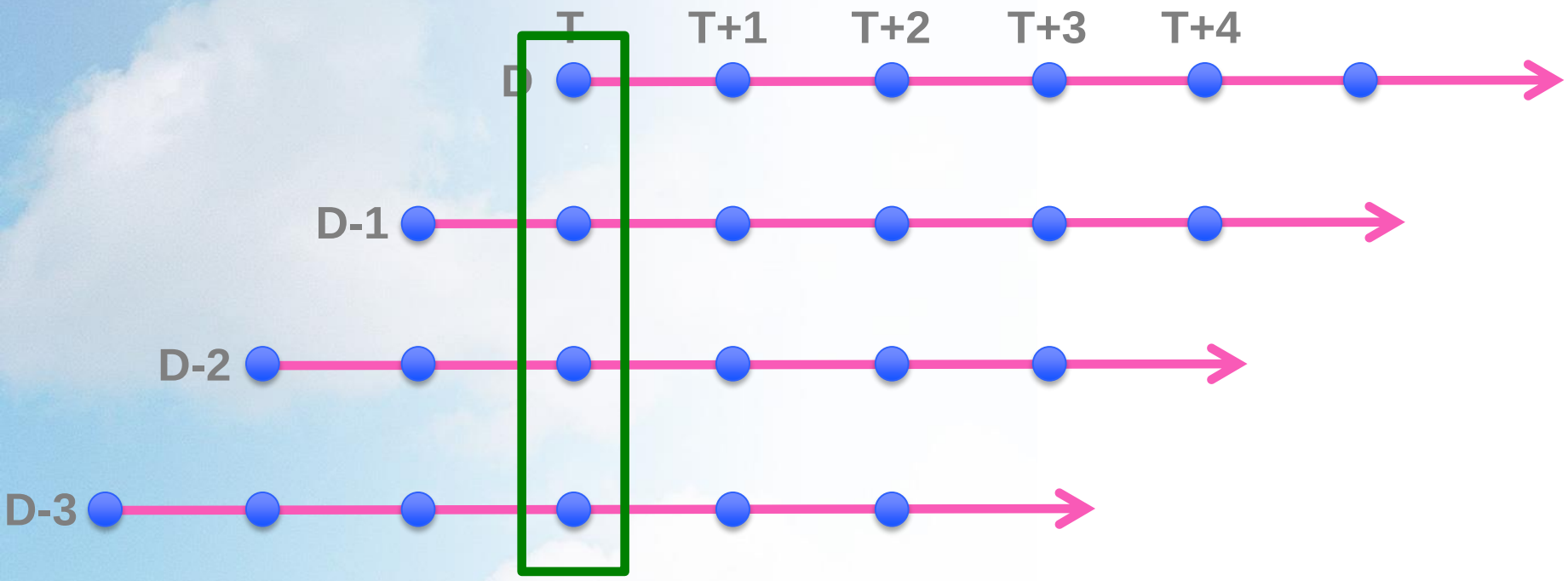
Date & time



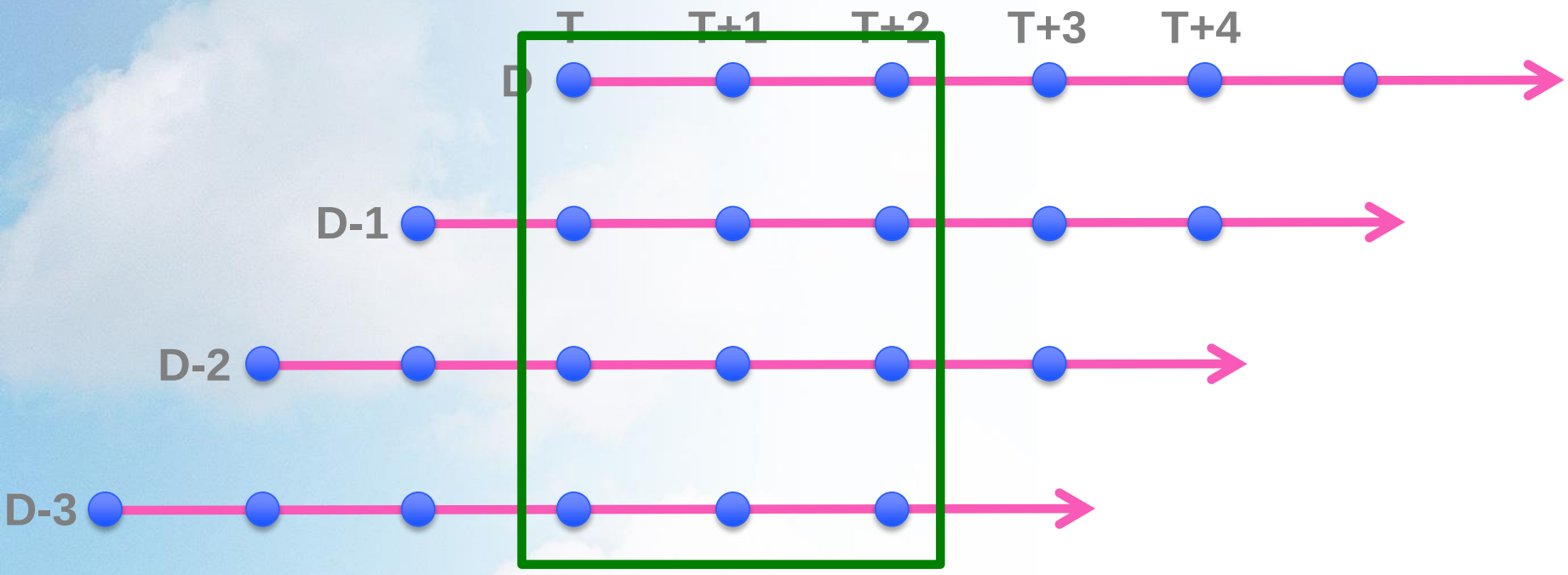
Date & time



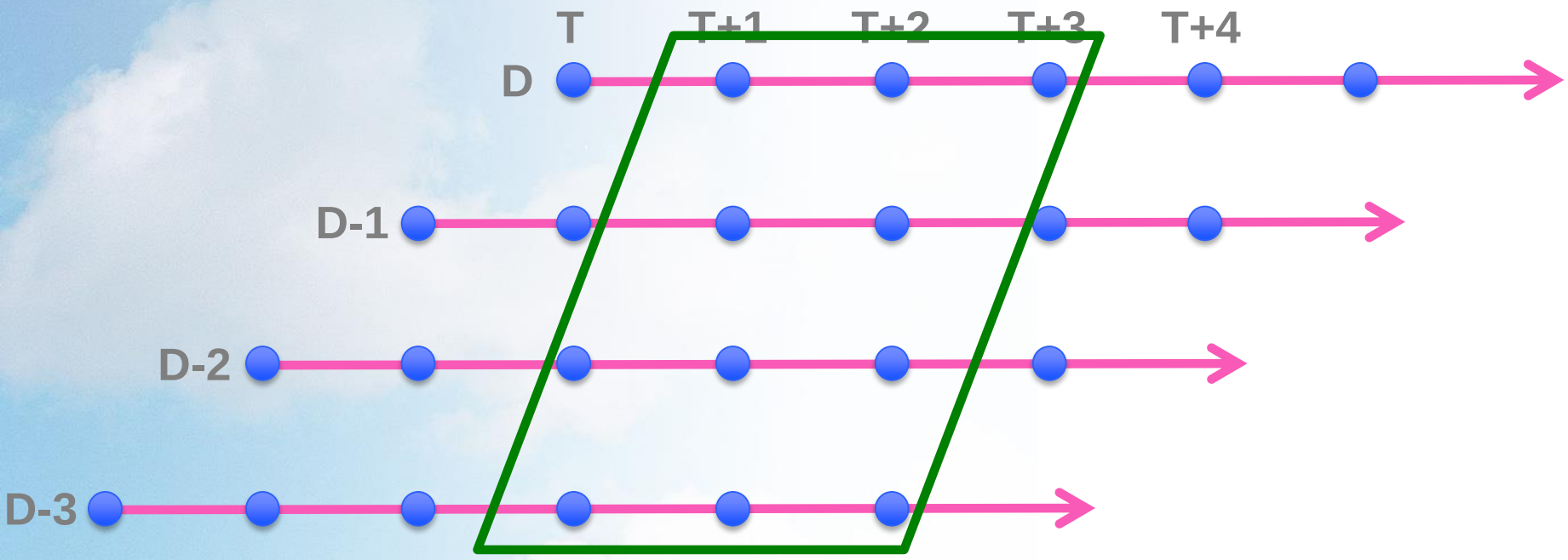
Date & time



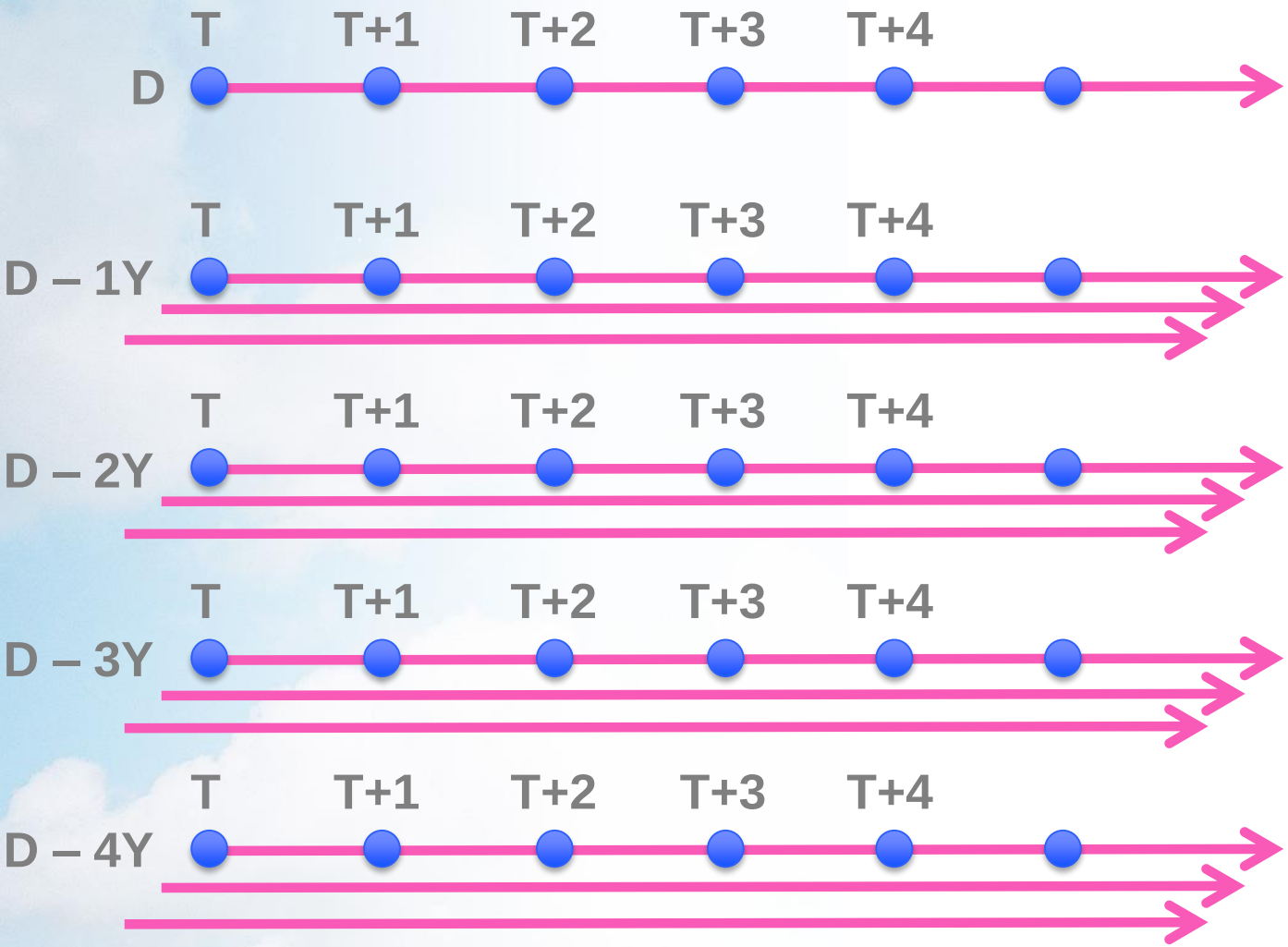
Date & time



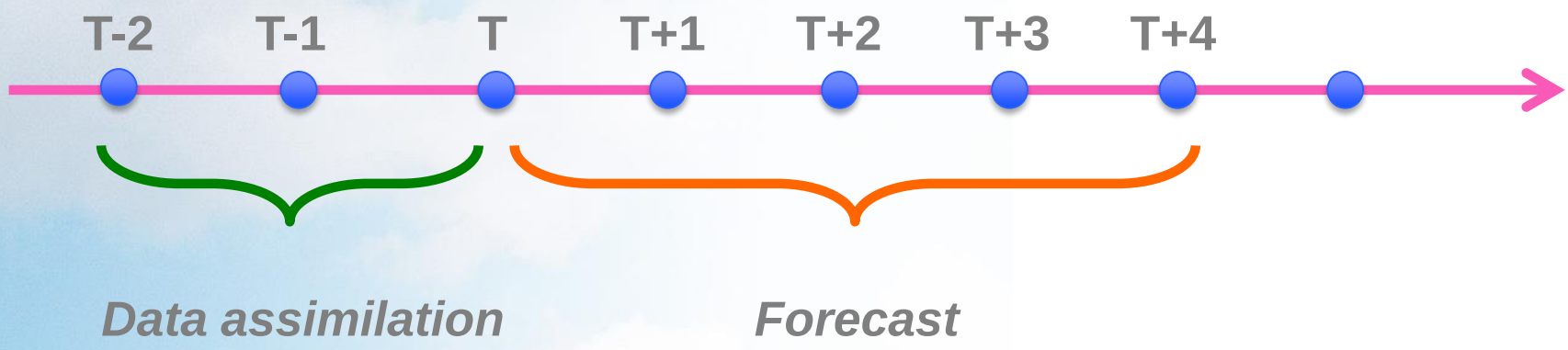
Date & time



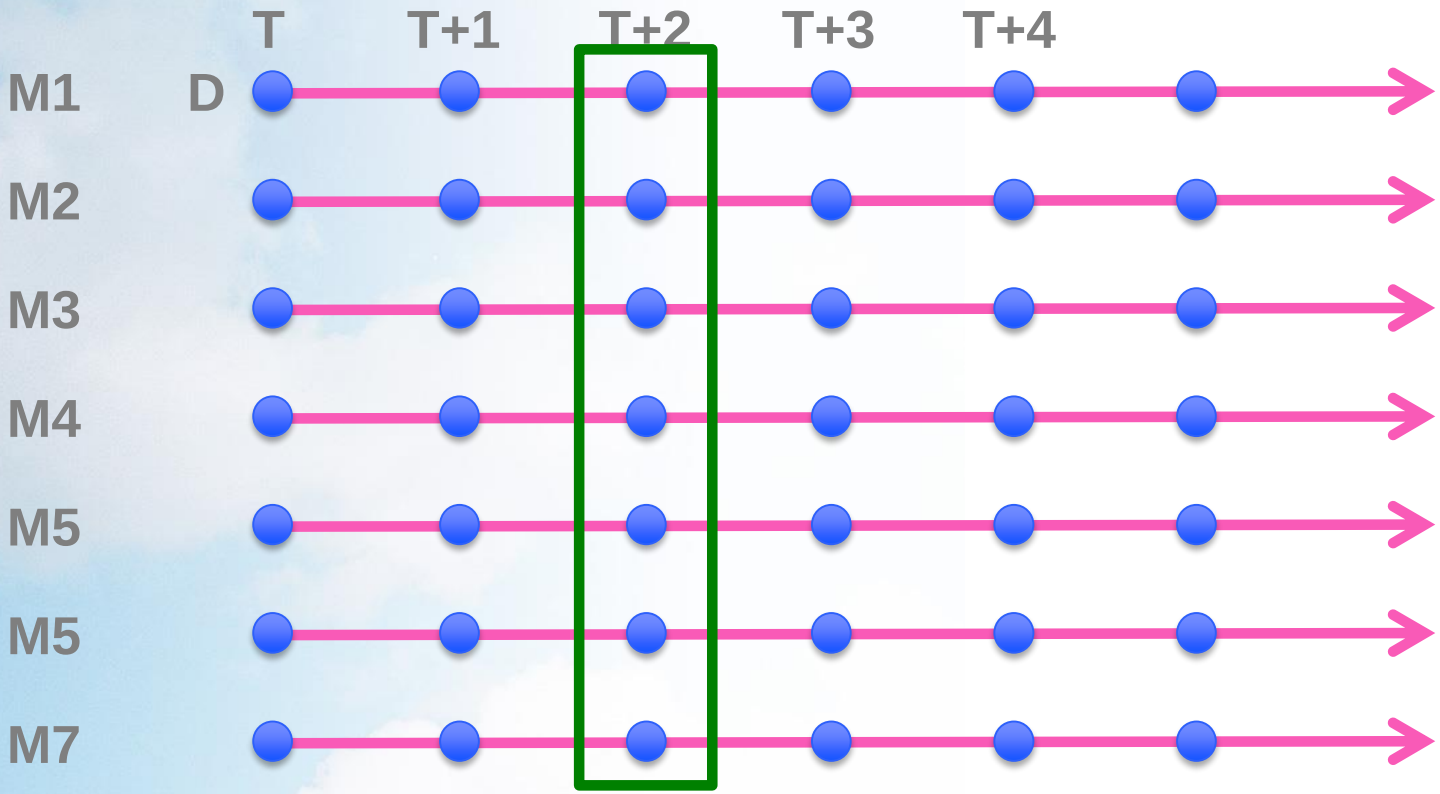
Date & time (Hindcasts)



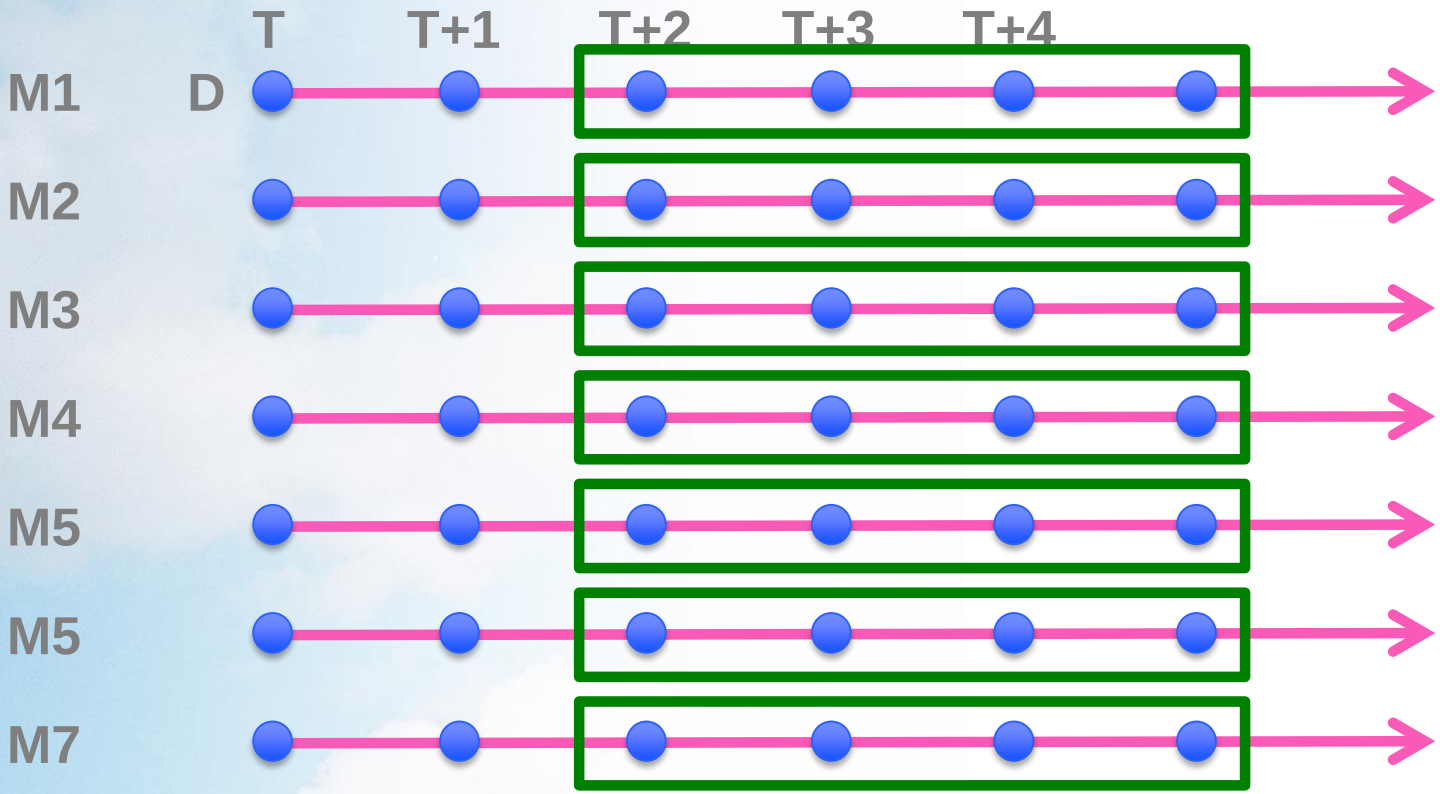
Date & time (long window 4D-var)



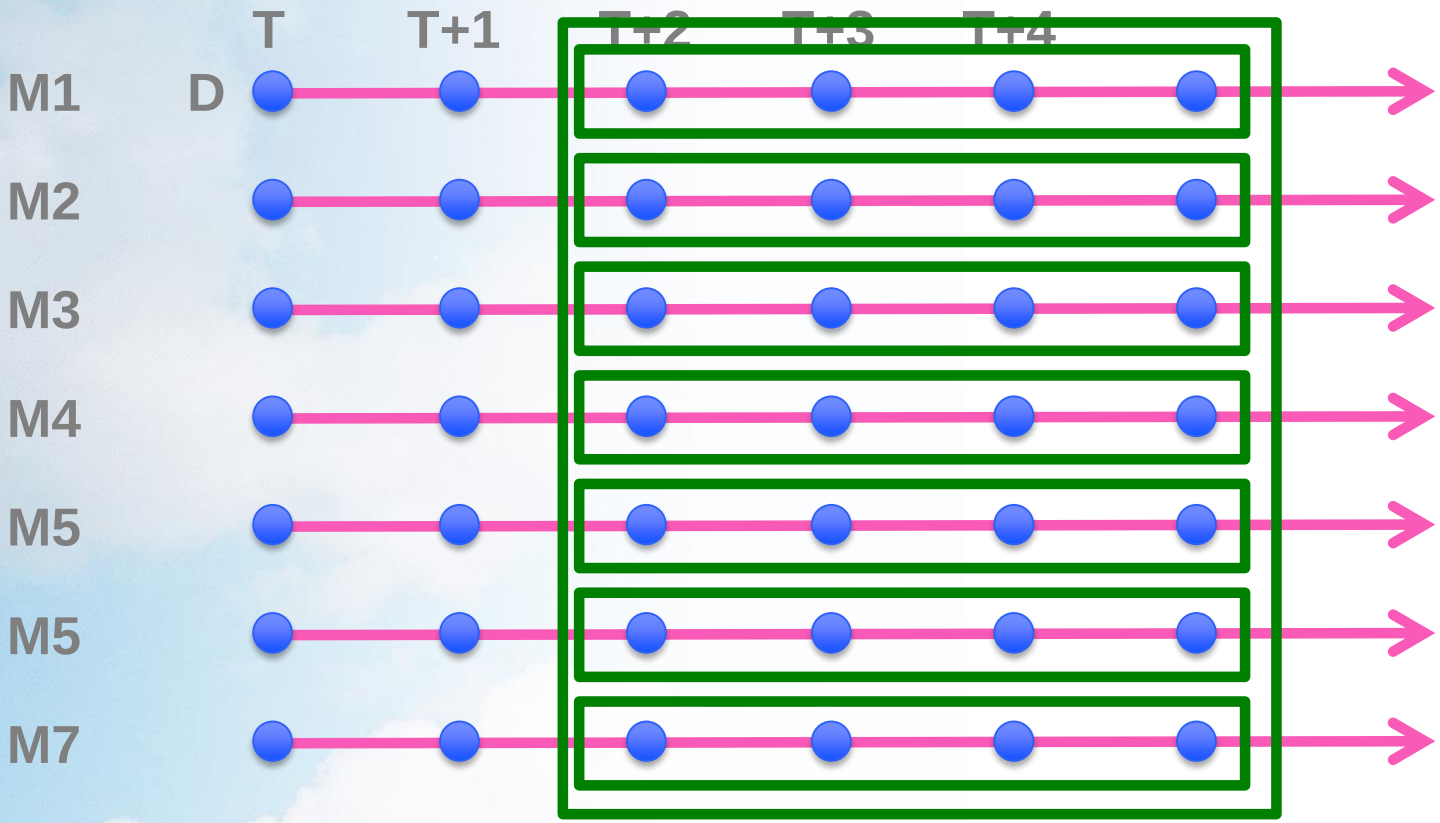
Averaging: ensemble means



Averaging: monthly means



Averaging: monthly means of ensemble means



Compression

GRIB simple packing

- Maps floating point range $[Field_{min}, Field_{max}]$ to integer range $[0, 2^n-1]$
- It's equivalent to sampling the field into 2^n buckets
  - Packing is lossy
- n can be anything between 1 and 32 (standard does not prevent n to be 255!)

- Most of the fields are packed with $n = 16$.
  - GRIB fields are half the size of the equivalent single precision float (or a quarter of double precision)

- Blind conversion from GRIB to NetCDF will create files twice as large (NC_FLOAT) or four time bigger (NC_DOUBLE)

NetCDF supports “simple packing”

- Using scale_factor and add_offset, and packing to NC_BYTE, NC_SHORT, NC_INT
  - Please note that these are signed (unsigned version comes with NetCDF4)
- Only multiple of 8 bits are supported
  - NC_BYTE = 8, NC_SHORT = 16, NC_INT = 32
- Missing values:
  - GRIB uses a “bitmap” to mark the missing values
  - NetCDF uses _FillValue to mark missing values
  - Consequence:
    - When packing NetCDF to NC_BYTE or NC_SHORT , we have 1 less value than GRIB
    - We cannot have encode the same range

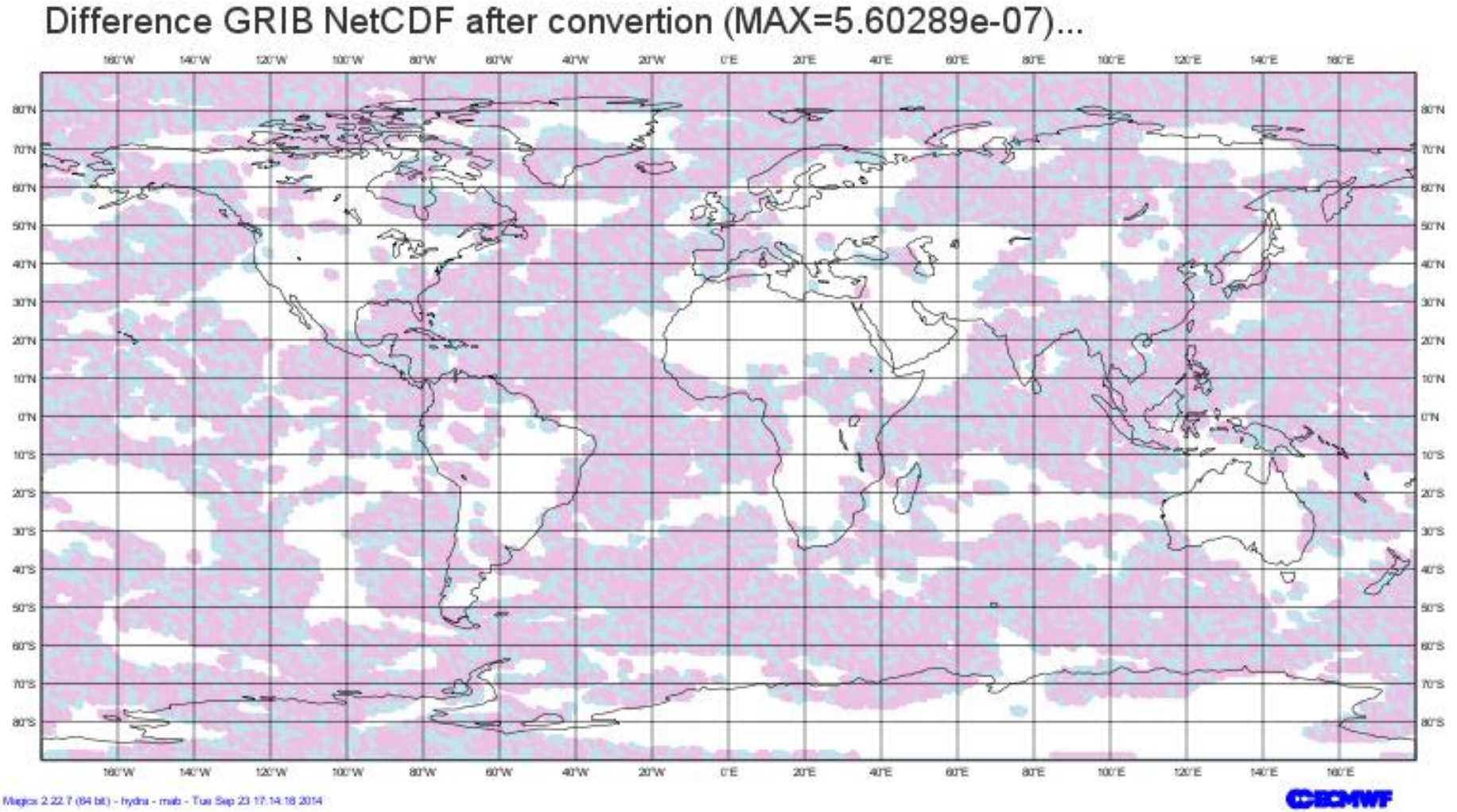
Conversion and “simple packing”: major issue

- GRIB applies simple packing per 2D field
- NetCDF may apply packing per 3D (space and level, space and time) and even 4D fields (space, level and time)

- Consequence: mapping floating point range $[Field_{min}, Field_{max}]$ to integer range $[0, 2^n-1]$ is done on more values in the case of NetCDF
  - Higher loss of accuracy

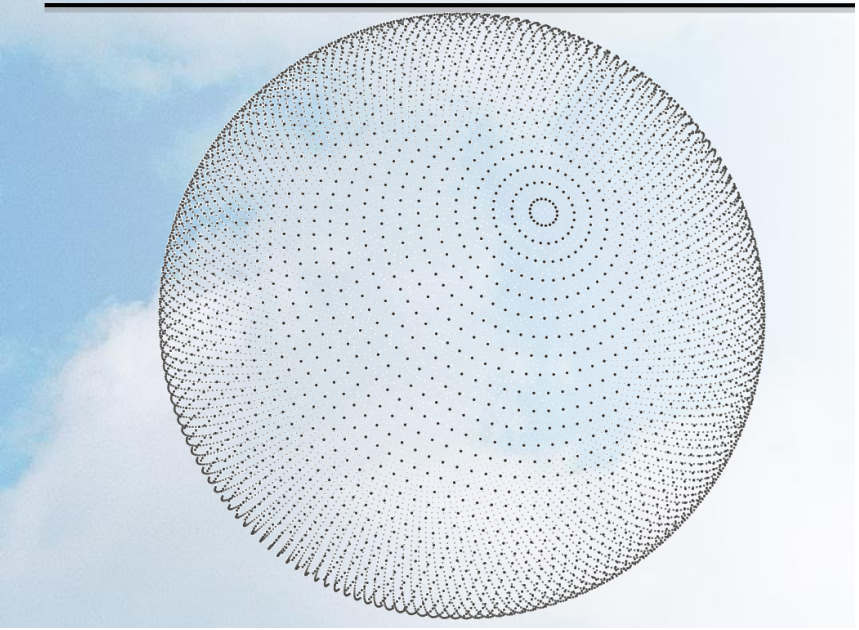
- Conversion leads to loss of information !!!!!
  - That’s not good™

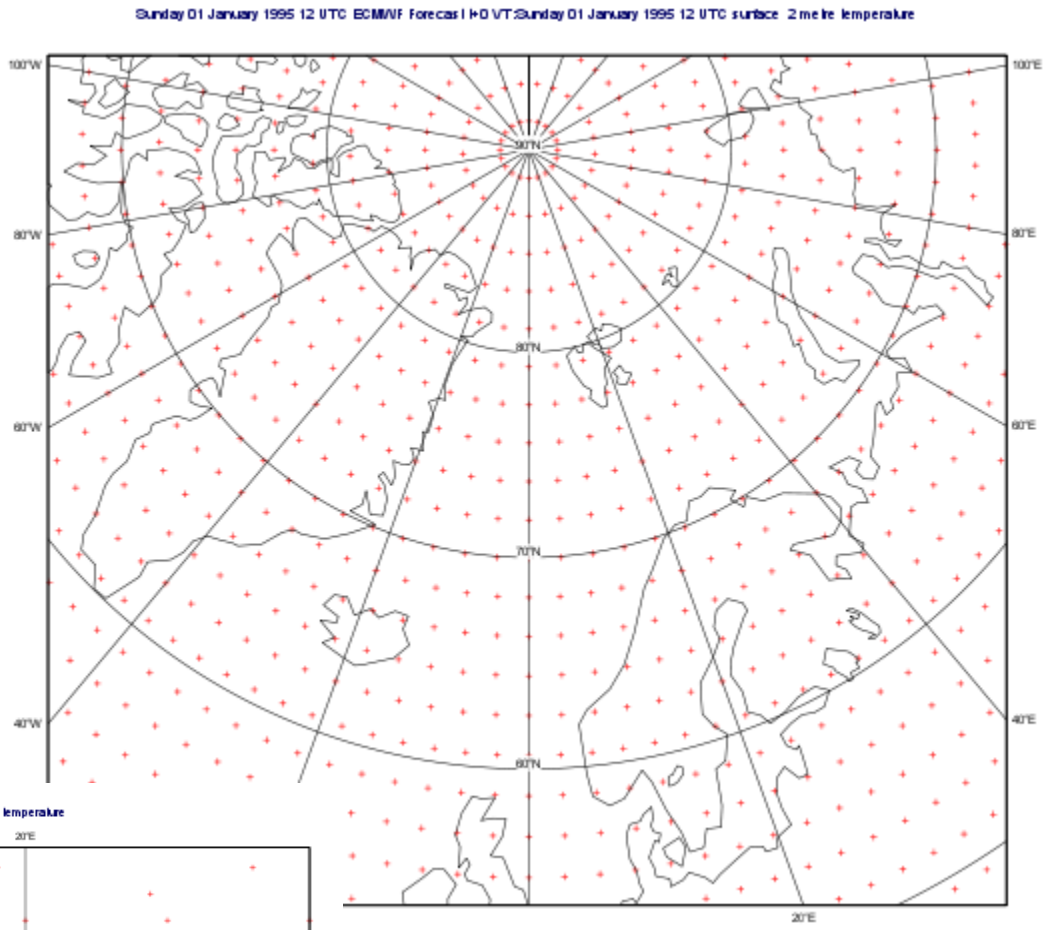
Difference is small, but non-zero (showing precipitations)

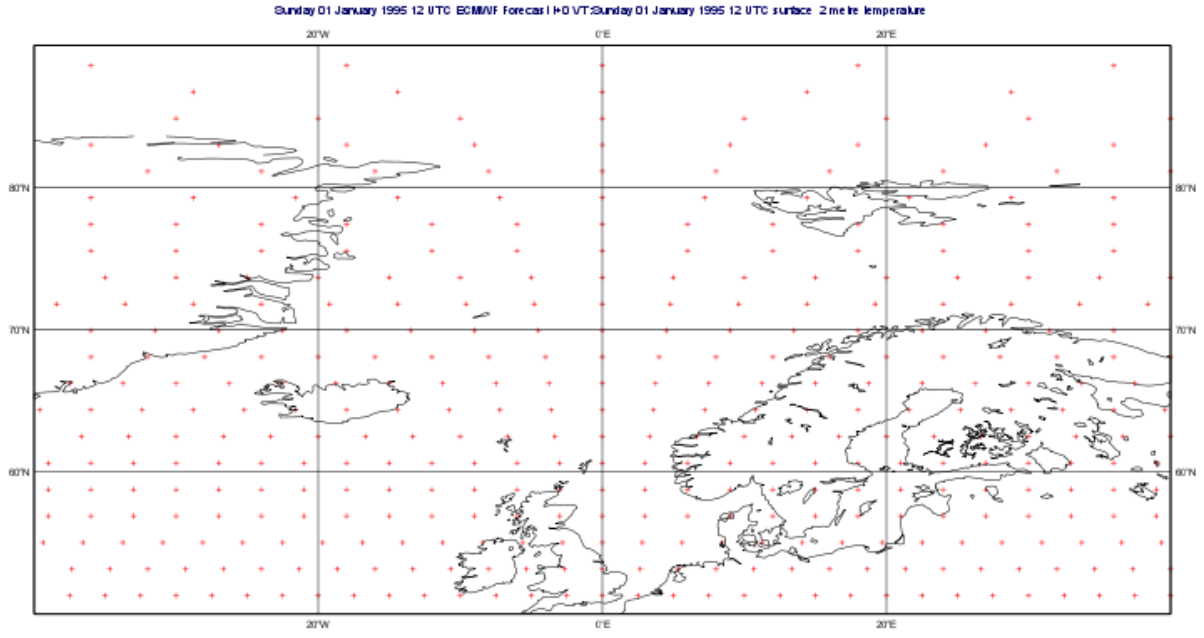


Grids

Reduced Gaussian Grid







ECMWF

What do I want from this workshop?

- A general agreement on how to map GRIB to NetCDF (parameters, units, metadata, file structures,…)
  - So no one complains that we “are not doing it right”
- A general agreement on how we deal with future requirements (new grids, new parameters, …)
  - Maybe a tighter collaboration between WMO and the CF community, like we did for OGC?