Data-handling and Infrastructure

Ewan Pinnington
(many thanks to Cathal O’Brien and Florian Pinault)

ECMWF

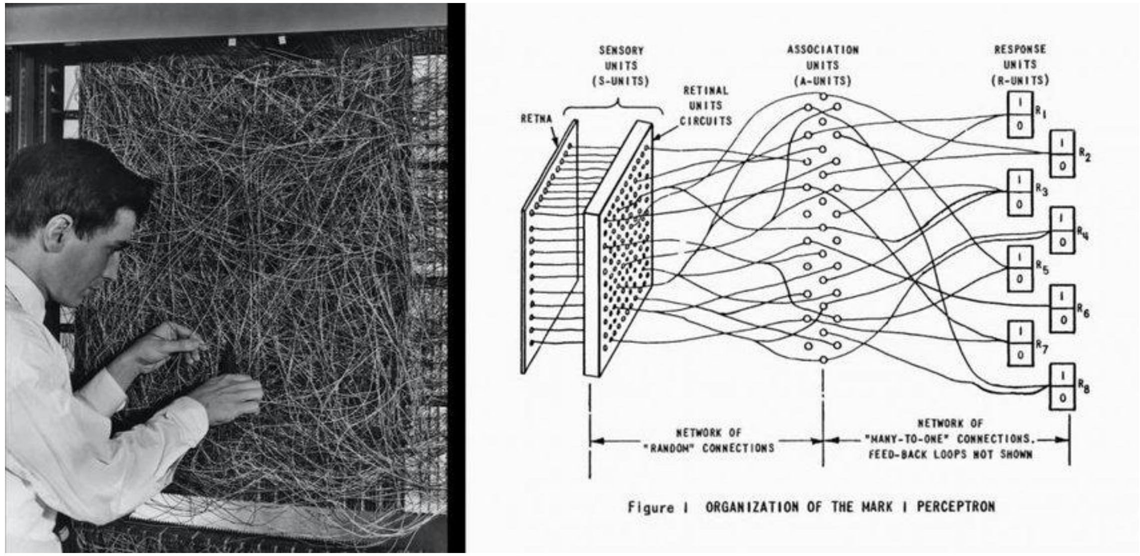
Overview

- Intro
- Where's the data live and where do we need it to get to?
- File Layout, Data Formats, Chunking
- High-level Summary

- Other Considerations with Data?
  - When can it go wrong?
  - Some real-life examples…

Background

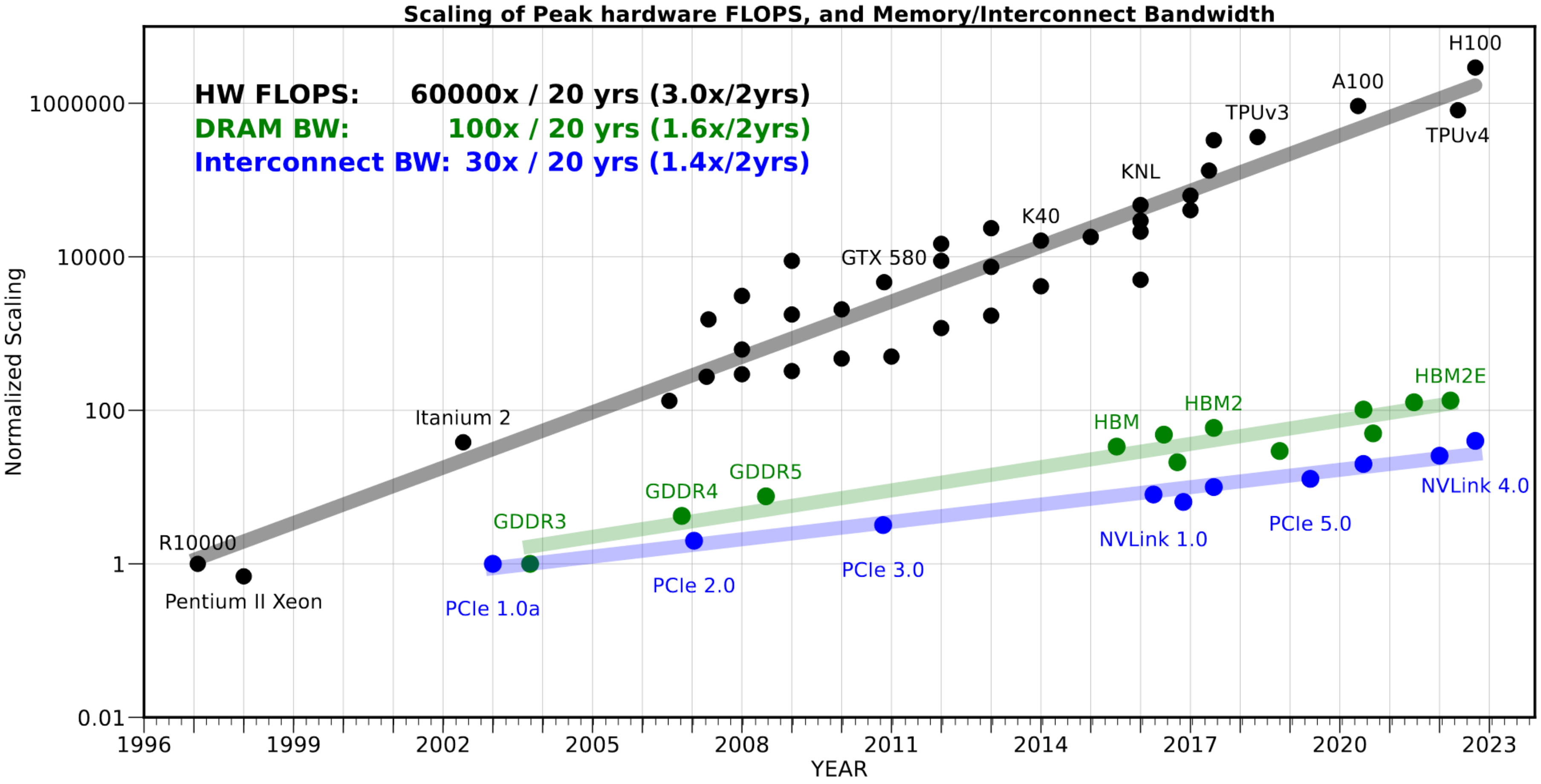
- Machine learning is an old idea



Frank Rosenblatt programming his Perceptron, 1970s

- Increased compute power (and available data) makes machine learning possible today
  - Moores law
  - ”Domain specific compute architectures” e.g. GPUs
- But there is a problem… as compute got faster, other components couldn’t keep up

The “Memory Wall” 😱



https://siliconmatter.substack.com/p/the-memory-wall-and-its-implications

1TB of data

64 GPUs

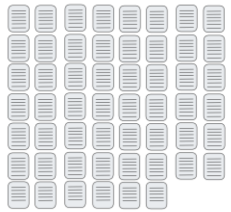
1 day

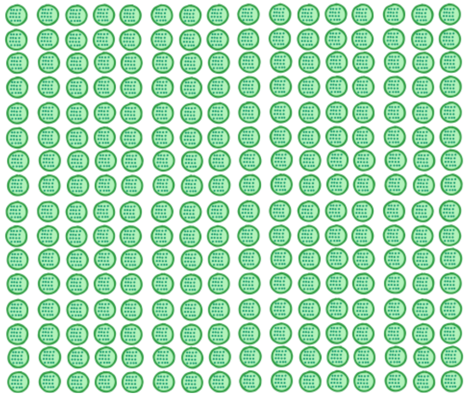
Input data

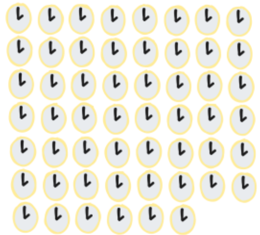
Compute

Walltime

Llama 3







1TB of data

64 GPUs

1 day

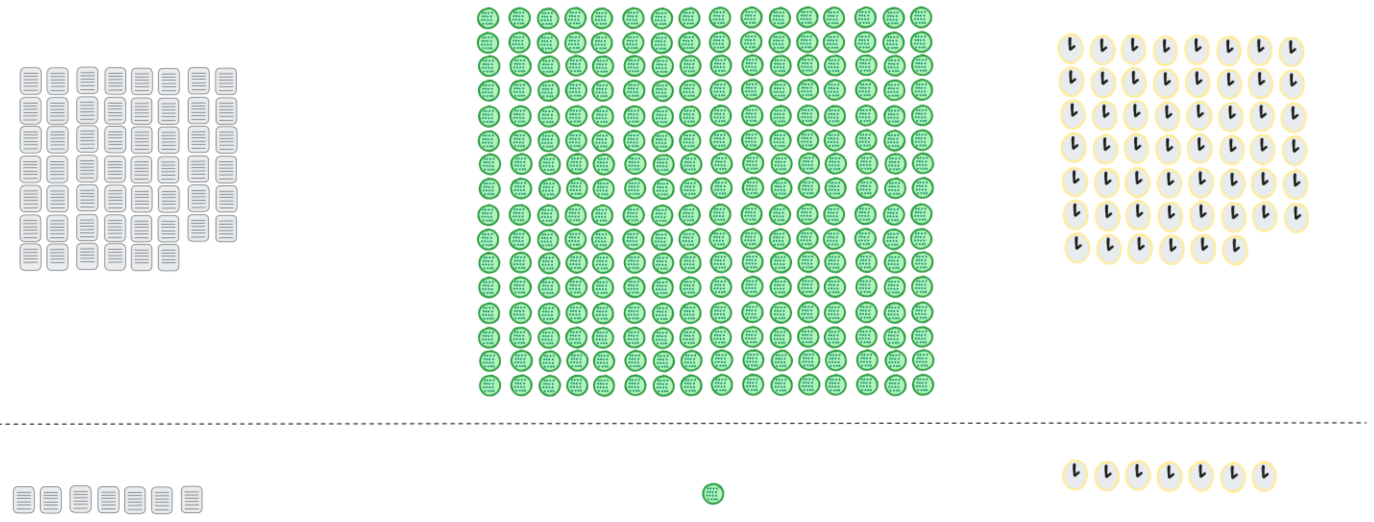
Input data

Compute

Walltime

Llama 3

AIFS single v1



Data Handling and infrastructure

Questions

How do I parallelise part … of my workflow?

How do I transfer my data to … ?

Should I use more memory/machines/nodes/GPUs?

Which data should I copy? Where?

Which format should I choose for my data?

How to make "it" fast?

Data Handling and infrastructure

Questions

How do I parallelise part … of my workflow?

How do I transfer my data to … ?

Should I use more memory/machines/nodes/GPUs?

Which data should I copy? Where?

Which format should I choose for my data?

How to make “it” fast?

Know the technology

HPC (High Performance Computing)? Cloud?

S3 buckets vs Lustre filesystems?

“Bring the code close to the data instead of data to the code.”

“Cloud-friendly” format?

Know your dataset

Total size on disk?

Total uncompressed size?

How many files?

Any missing data? Nans ?

Dimensions of the data? Full n-dimensional array? Several arrays?

For machine learning :

What is the size of one training sample? Of one batch?

Where is the data from? On which data will I run inference? How?

Know your read/write patterns

Random read (shuffling)

Transpose the data if needed

Universal answer

➔“It depends”

The best solutions will usually depend

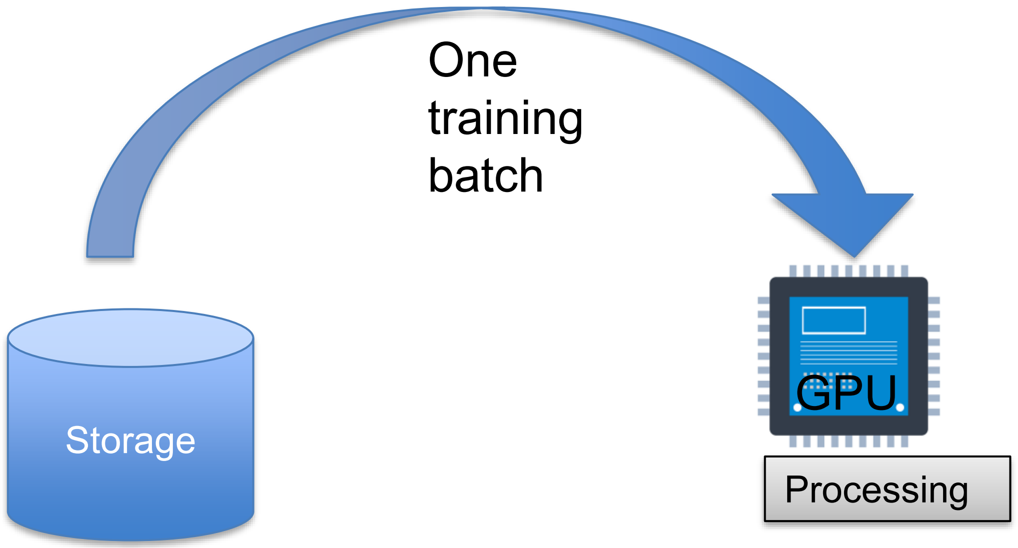
- on the size of the data
- on the project requirements
- on the available funding
- on previous experience
- on personal preferences
- and more…

“When you have a hammer, everything looks like a nail.”

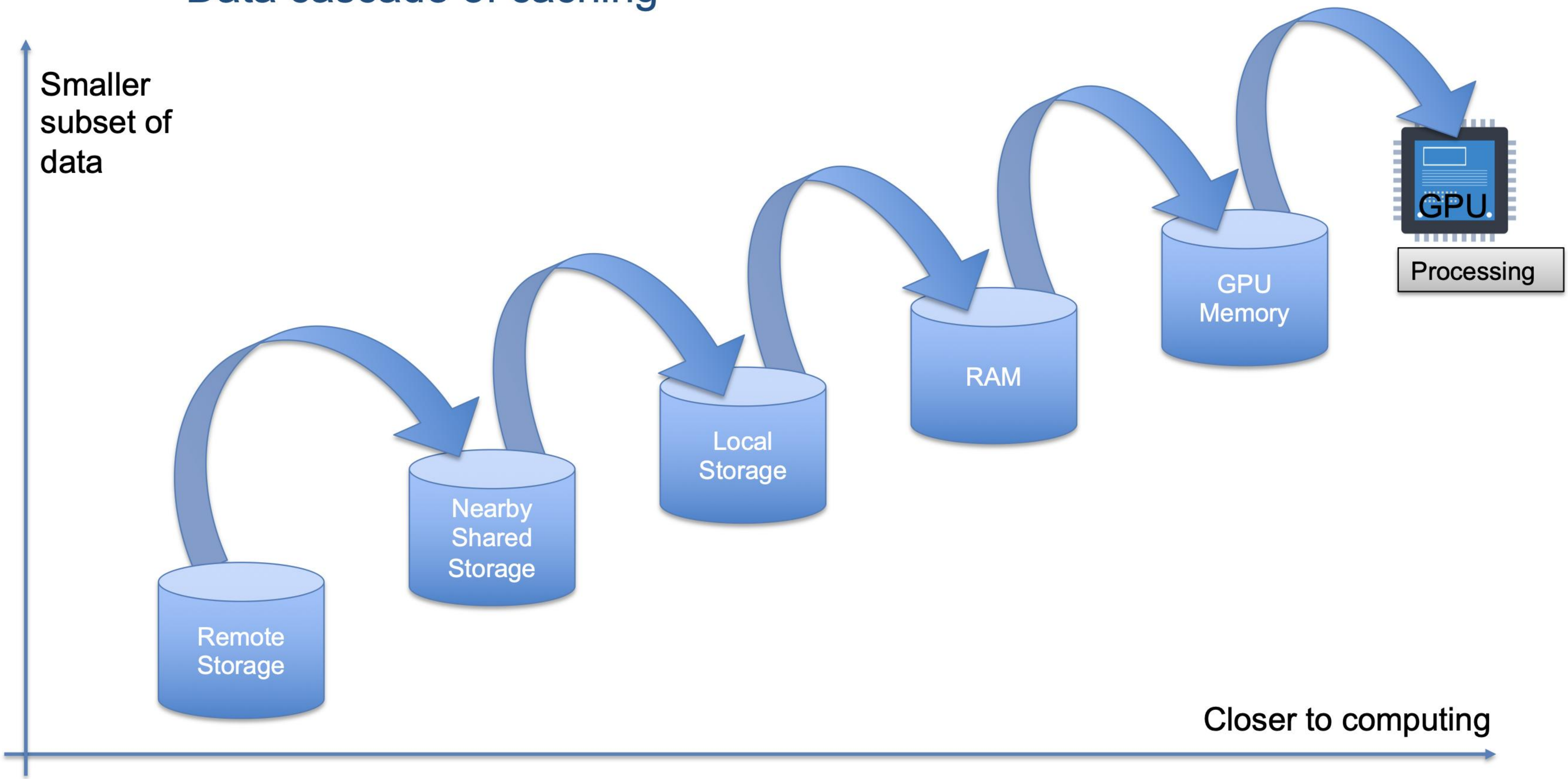
Data flow: Feeding the GPU with data

- Data is on disk
- We need to move it to the GPU
- One main requirement :

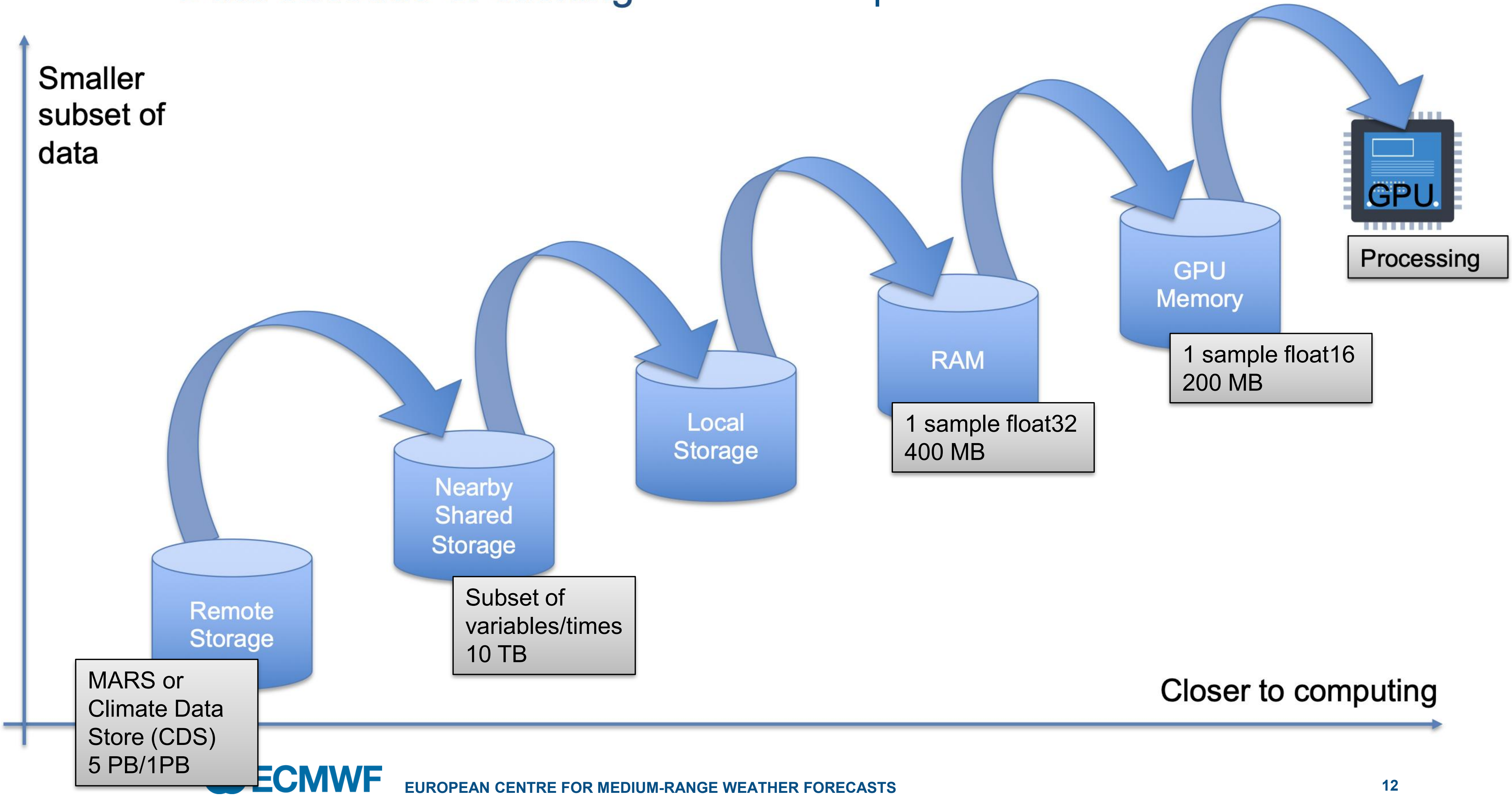
    Faster than GPU processing



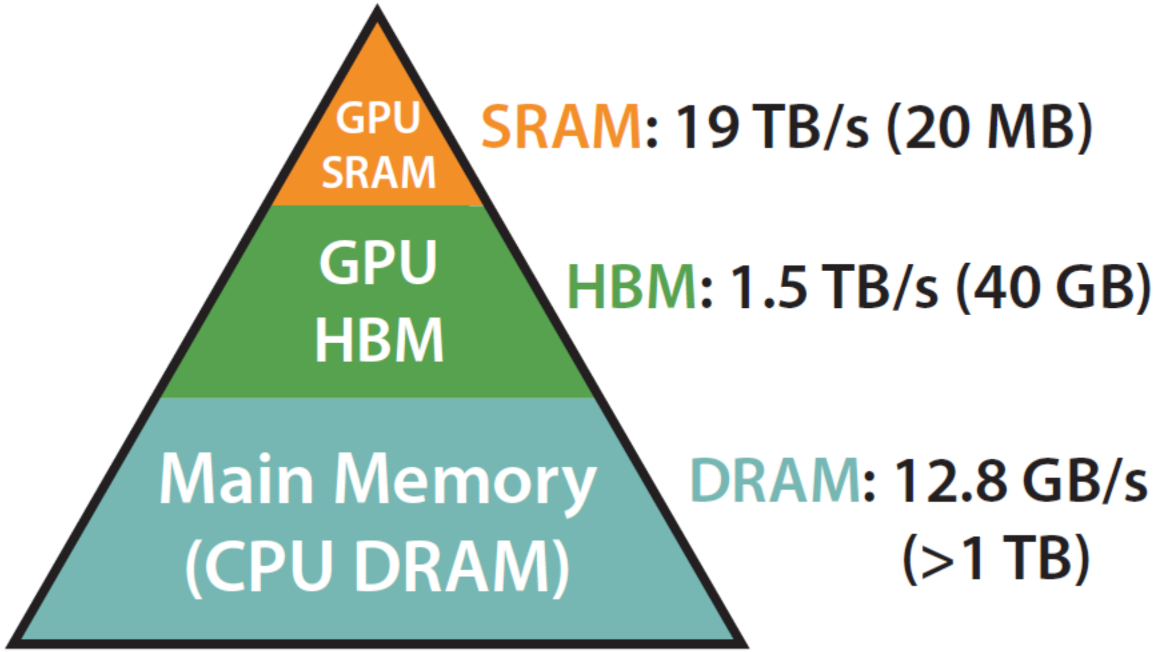
Data cascade of caching



Data cascade of caching: ERA5 example

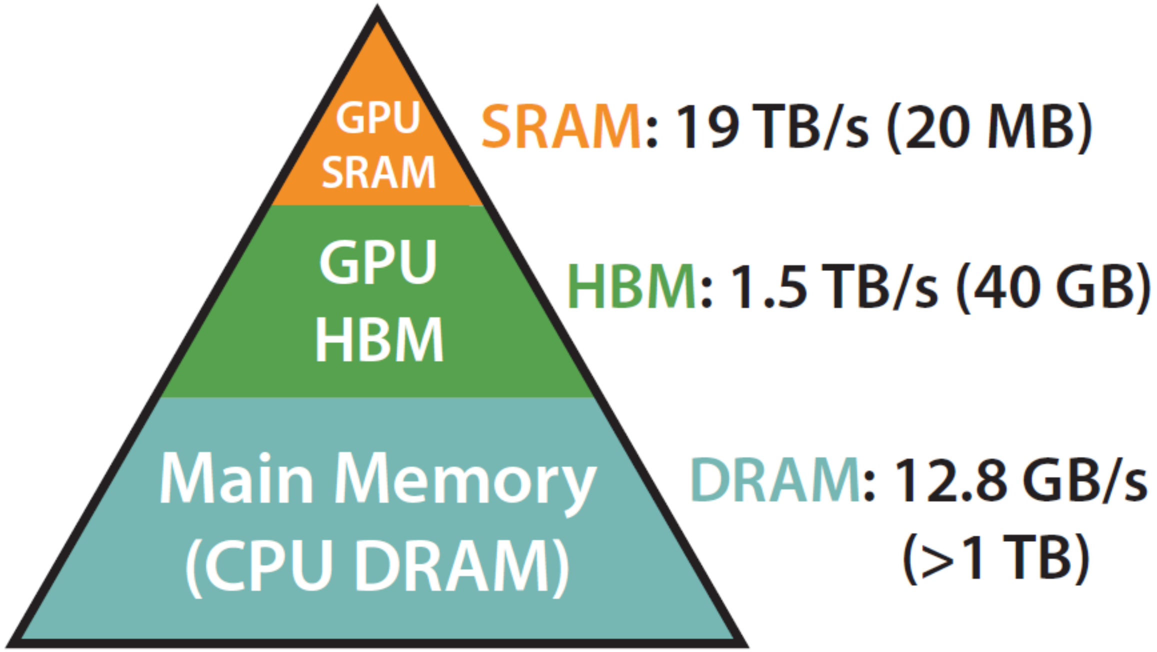


GPU Memory Hierarchy



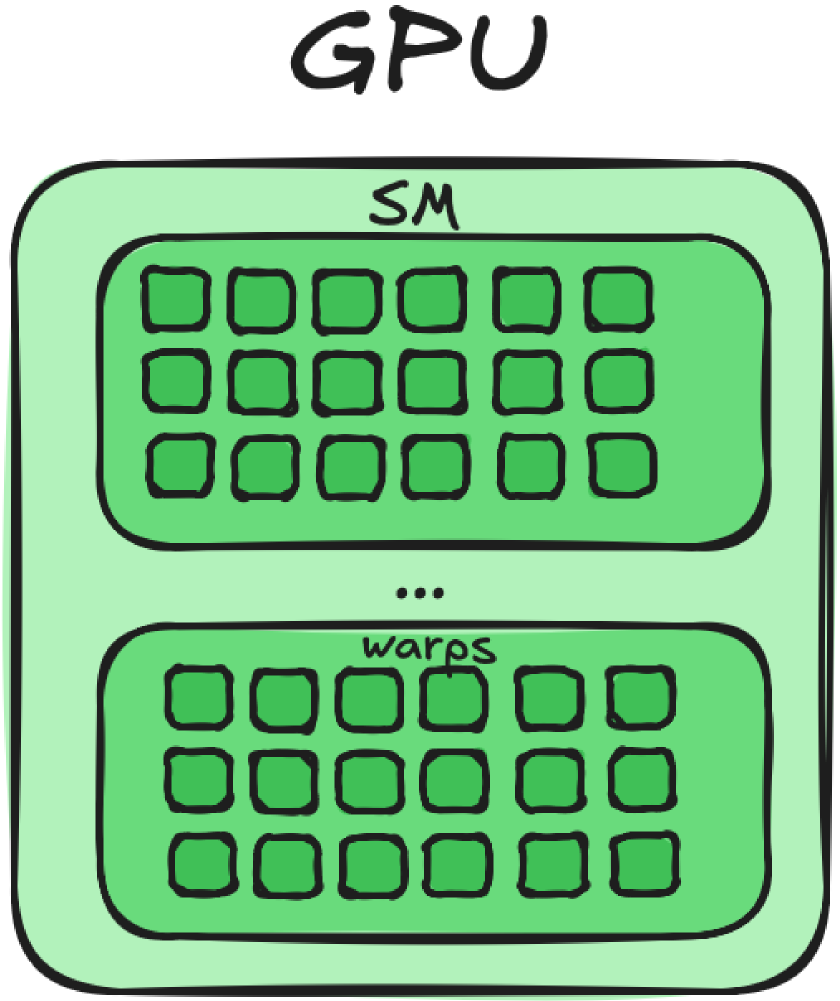
Memory Hierarchy with
Bandwidth & Memory Size

GPU Memory Hierarchy

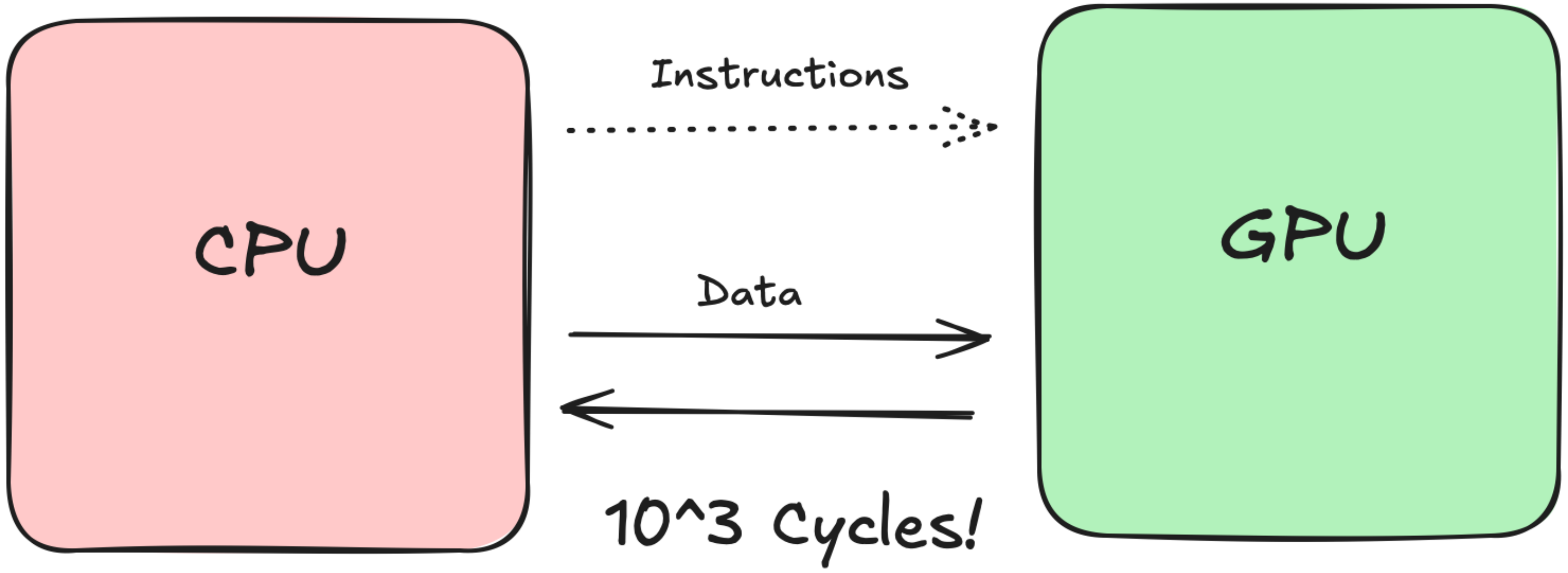


Memory Hierarchy with Bandwidth & Memory Size

What is a GPU?



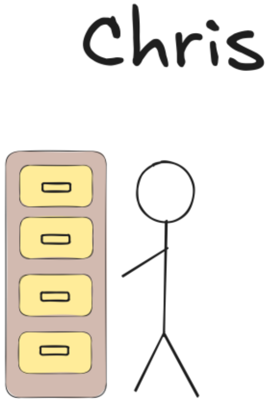
What is a CPU?

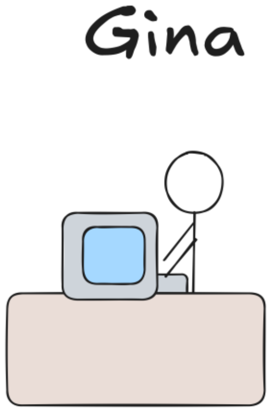


- Must be driven by a CPU
  - CPU launches all programs (called ‘kernels’) on GPU
  - CPU and GPU have separate memory spaces and must explicitly transfer data (slow)

CPU

- Primary use in machine learning is to prepare data for the GPU





CPU

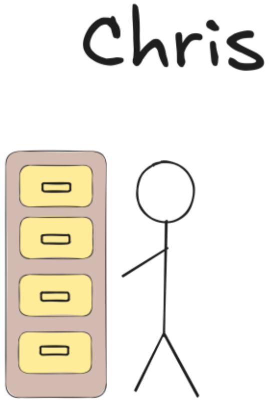
- Primary use in machine learning is to prepare data for the GPU

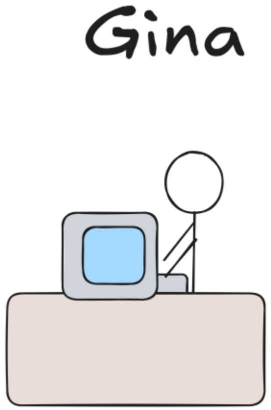
CPU

- Primary use in machine learning is to prepare data for the GPU

CPU

- Primary use in machine learning is to prepare data for the GPU





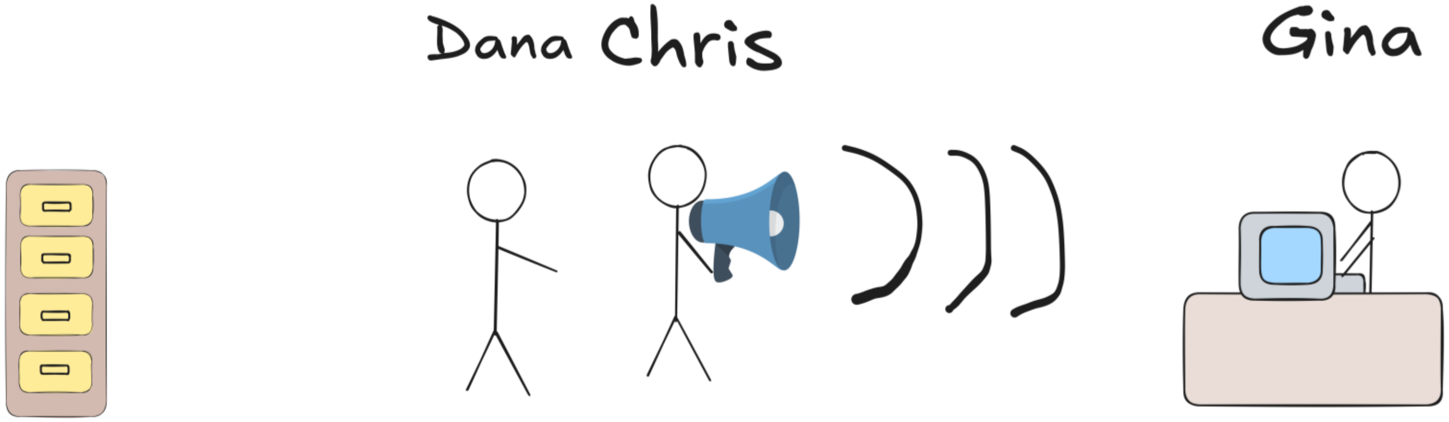

```
dataloader:
  num_workers:
    training: 0
```

CPU

- Primary use in machine learning is to prepare data for the GPU

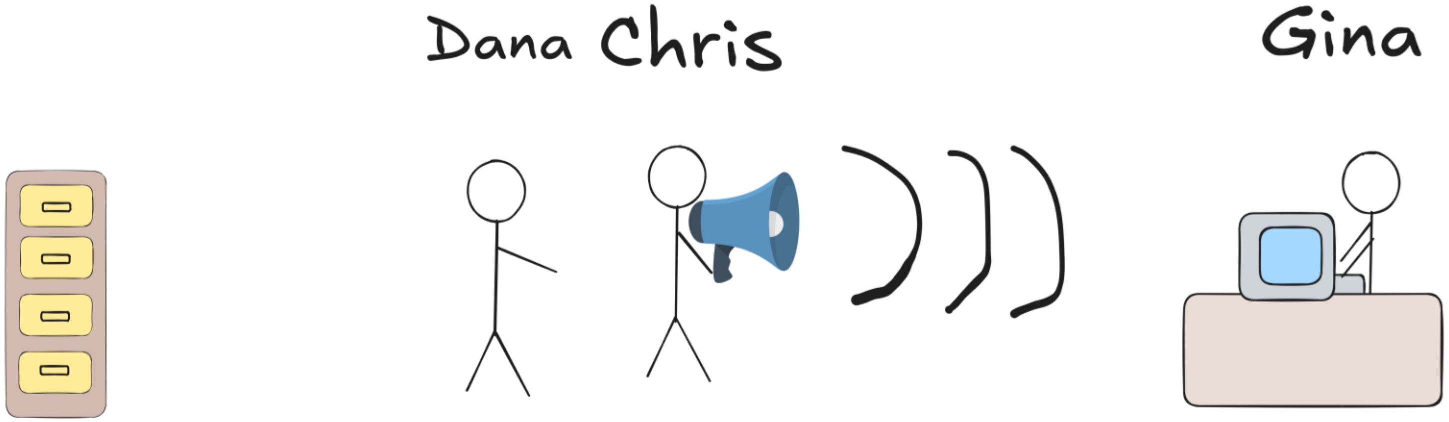
CPU

- Primary use in machine learning is to prepare data for the GPU



CPU

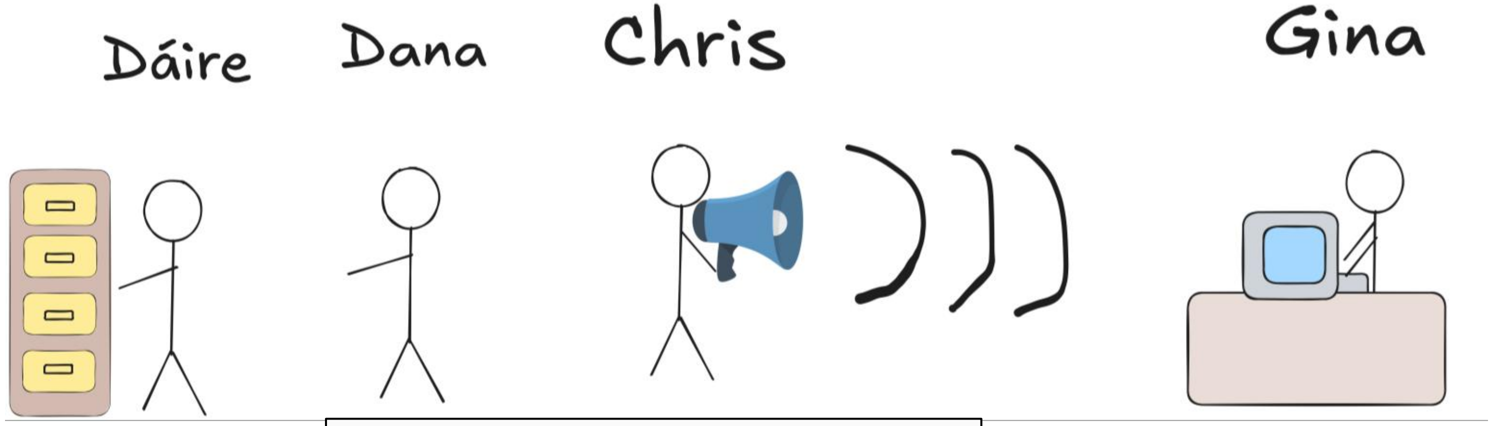
- Primary use in machine learning is to prepare data for the GPU




```
dataloader:
  num_workers:
    training: 1
```

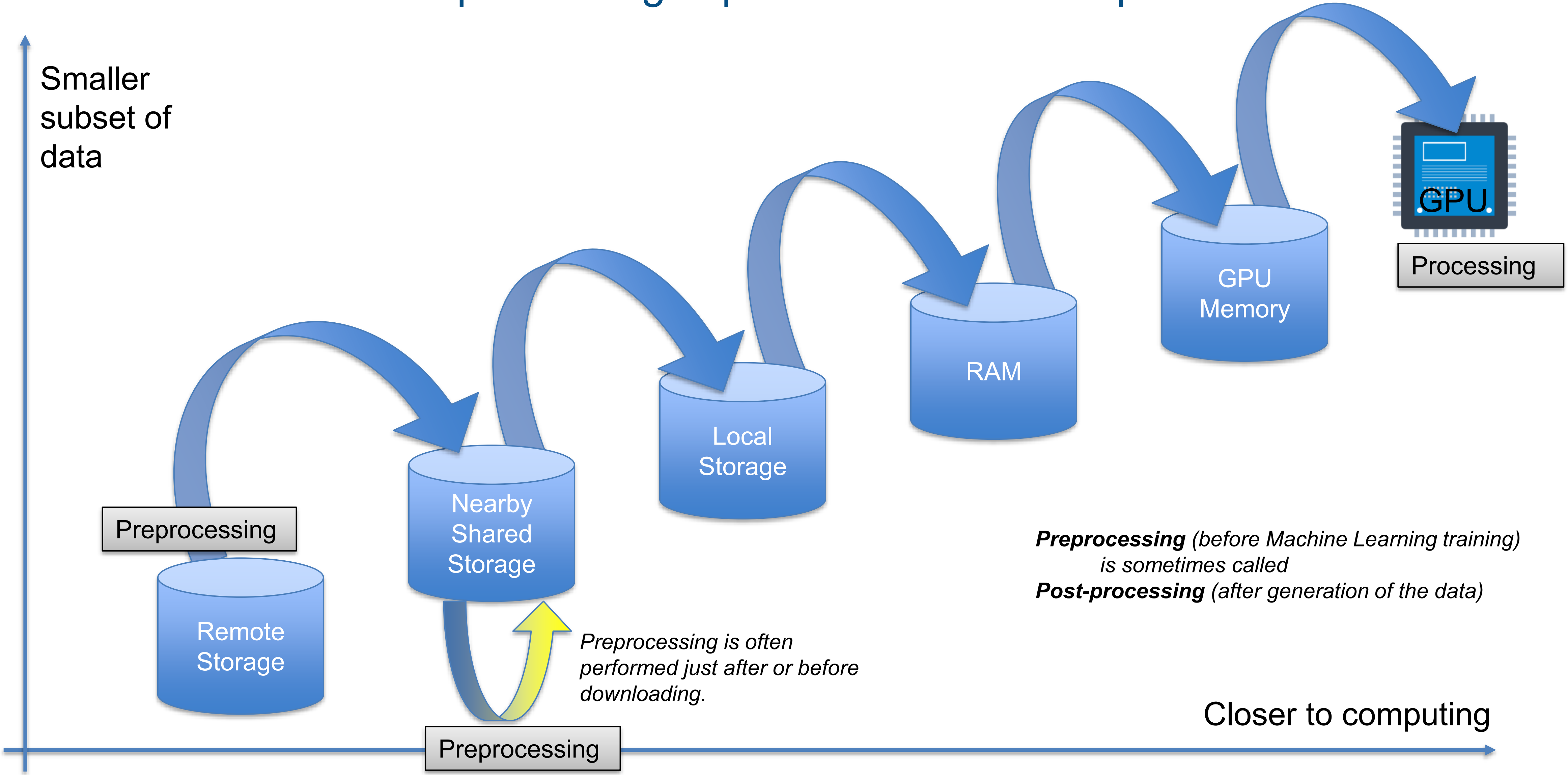
CPU

- Primary use in machine learning is to prepare data for the GPU

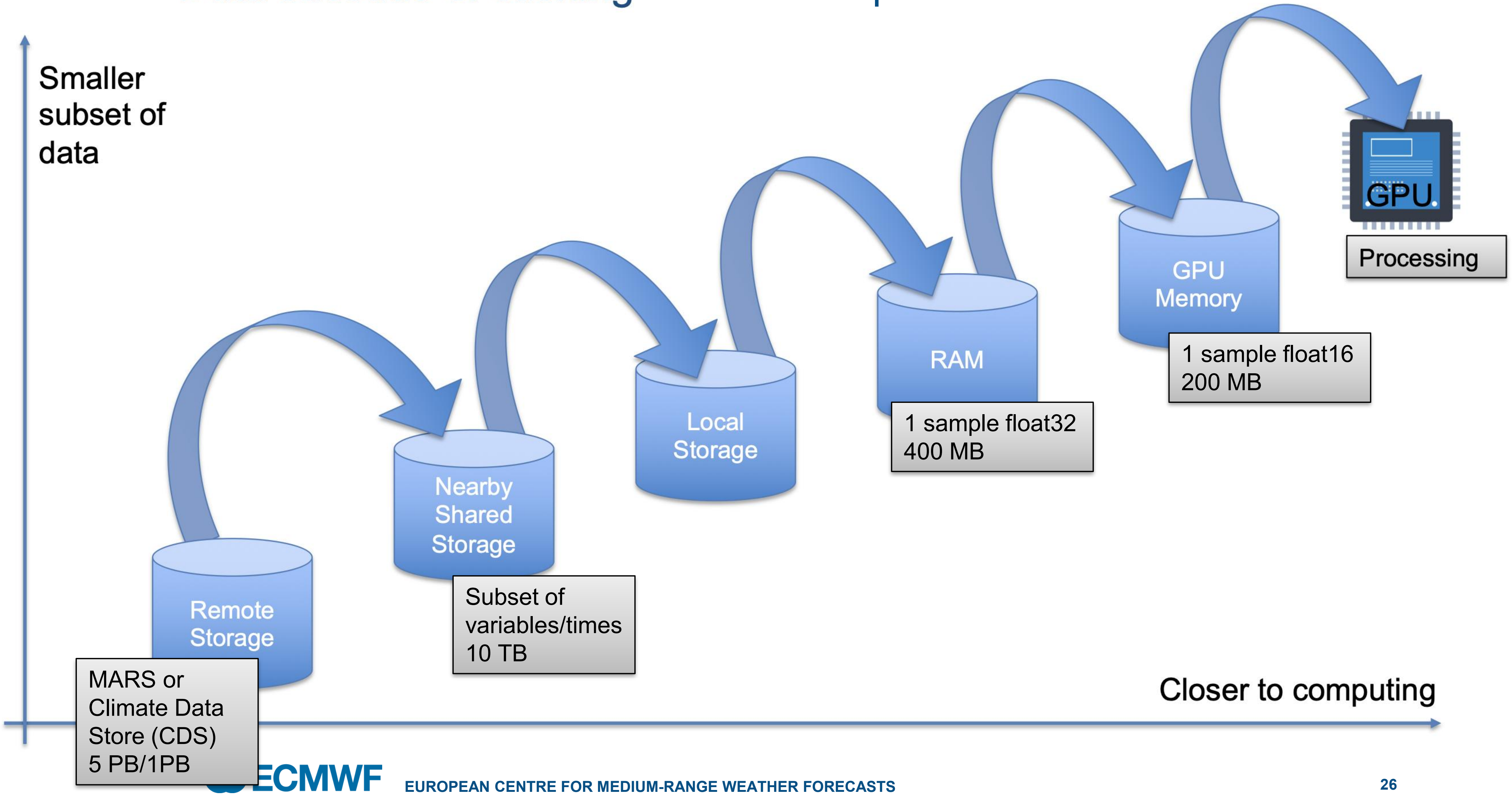



```
dataloader:
  num_workers:
    training: 1
```

Data flow: Preprocessing is possible on each step

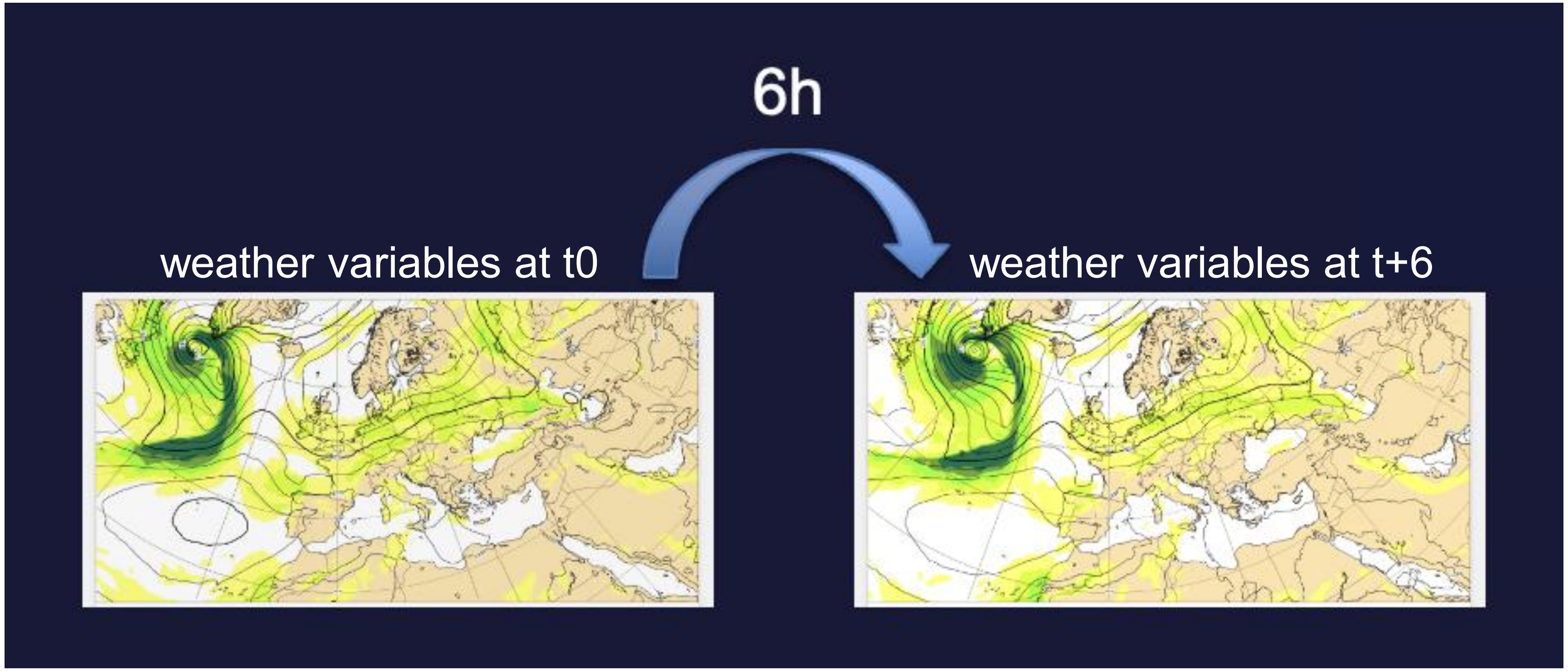


Data cascade of caching: ERA5 example



Data Formatting: Know your access patterns!

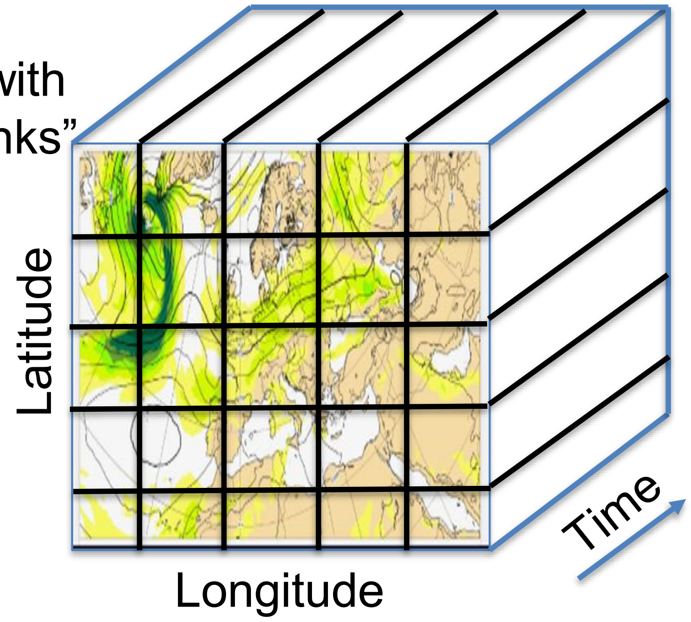
- The AIFS is trained on the reanalysis datasets **ERA5** produced by the Copernicus Climate Change Service (C3S) at ECMWF and fine-tuned on operational real-time analysis
- The forecast is produced by **autoregressively** stepping 6h into the future
- Dataset dimensions: [**time**, **latitude**, **longitude**, **variable**]



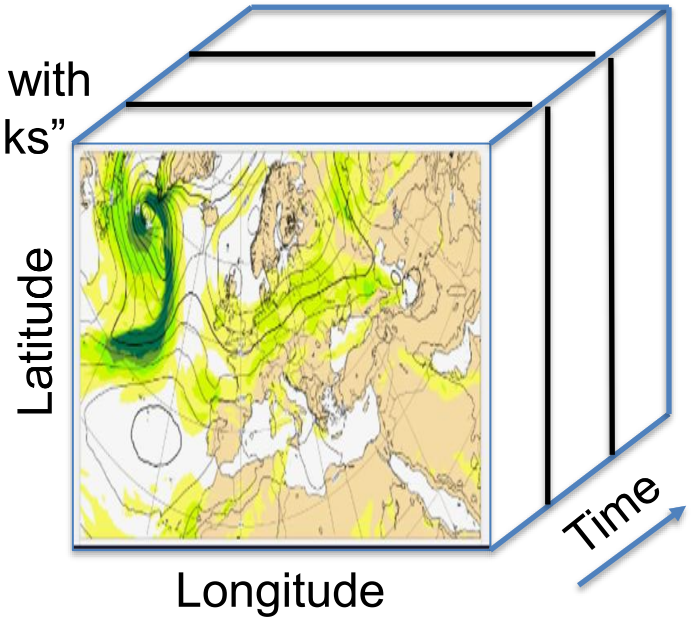
Data Formatting: Know your access patterns!

- The AIFS is trained on the reanalysis datasets **ERA5** produced by the Copernicus Climate Change Service (C3S) at ECMWF and fine-tuned on operational real-time analysis
- The forecast is produced by **autoregressively** stepping 6h into the future
- Dataset dimensions: [**time**, **longitude**, **latitude**, **variable**]
- Approximate sizes for AIFS: [**67000**, **1440**, **720**, **100**]

Save data with space “chunks”

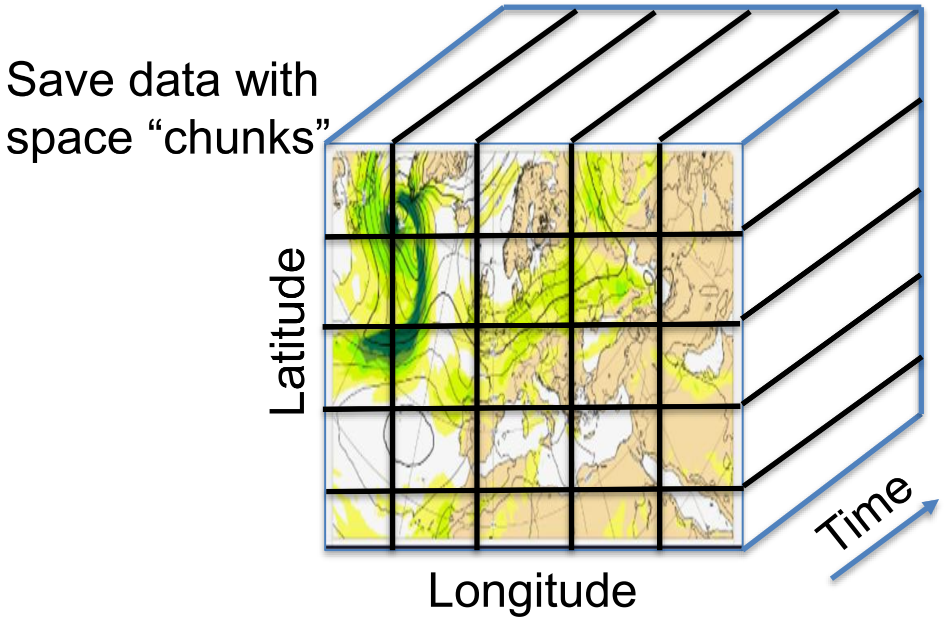


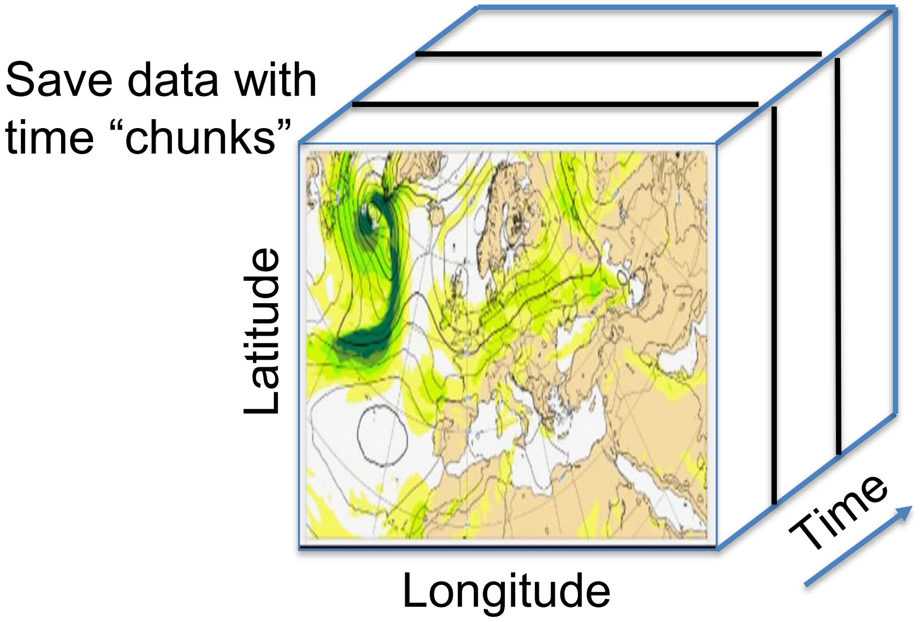
Save data with time “chunks”



Data Formatting: Know your access patterns!

- The AIFS is trained on the reanalysis datasets **ERA5** produced by the Copernicus Climate Change Service (C3S) at ECMWF and fine-tuned on operational real-time analysis
- The forecast is produced by **autoregressively** stepping 6h into the future
- Dataset dimensions: [**time**, **space**, **variable**]
- Approximate sizes for AIFS: [**67000**, **542080**, **100**]





Know your access patterns: transposition

Example data

9 fields (f)

14 dates (t)

Real data

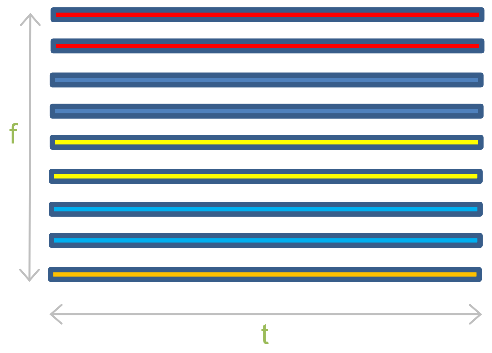
100 fields

1M dates

+ additional

dimensions

Data saved by field

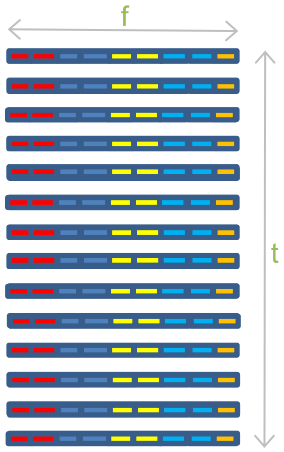


One file contains the timeseries of one given **field**

9 fields

≠

Data saved by date



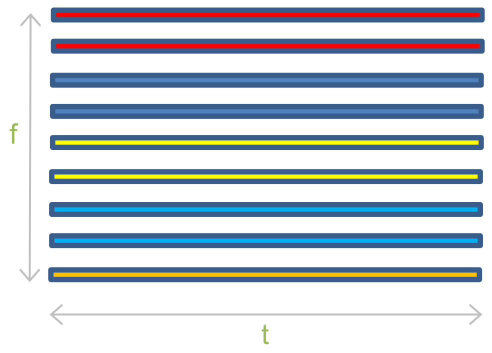
One file contains the set of values for one given **date**

9 dates

More generally:
Each "file" here could be an S3 object, or a part of a file, a record in a database, etc.

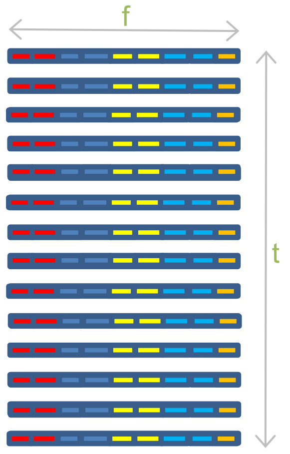
Know your access patterns: transposition

Data saved by field



≠

Data saved by date

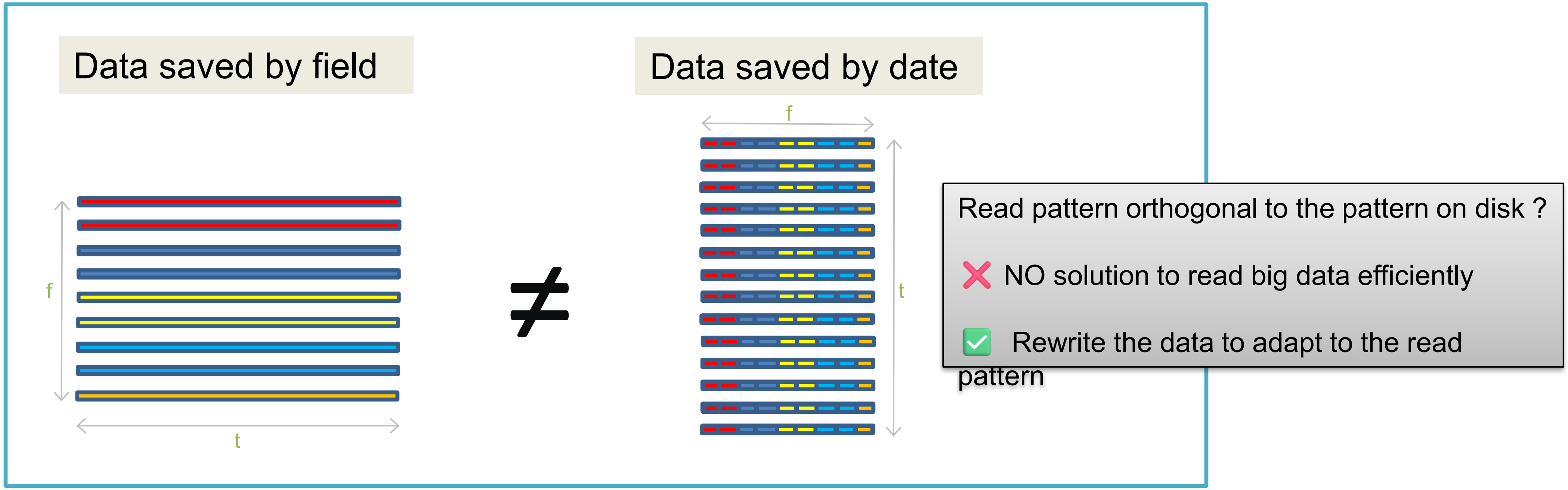


Read pattern orthogonal to the pattern on disk ?

❌ NO solution to read big data efficiently

✅ Rewrite the data to adapt to the read pattern

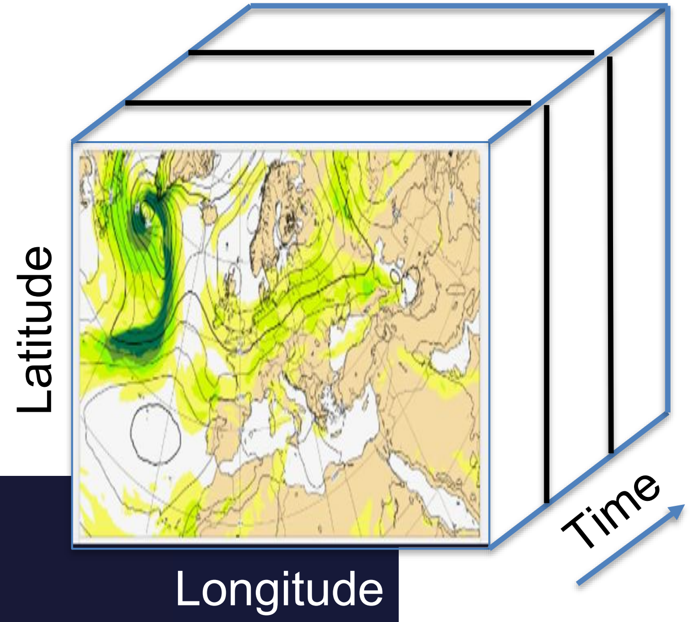
Know your access patterns: transposition

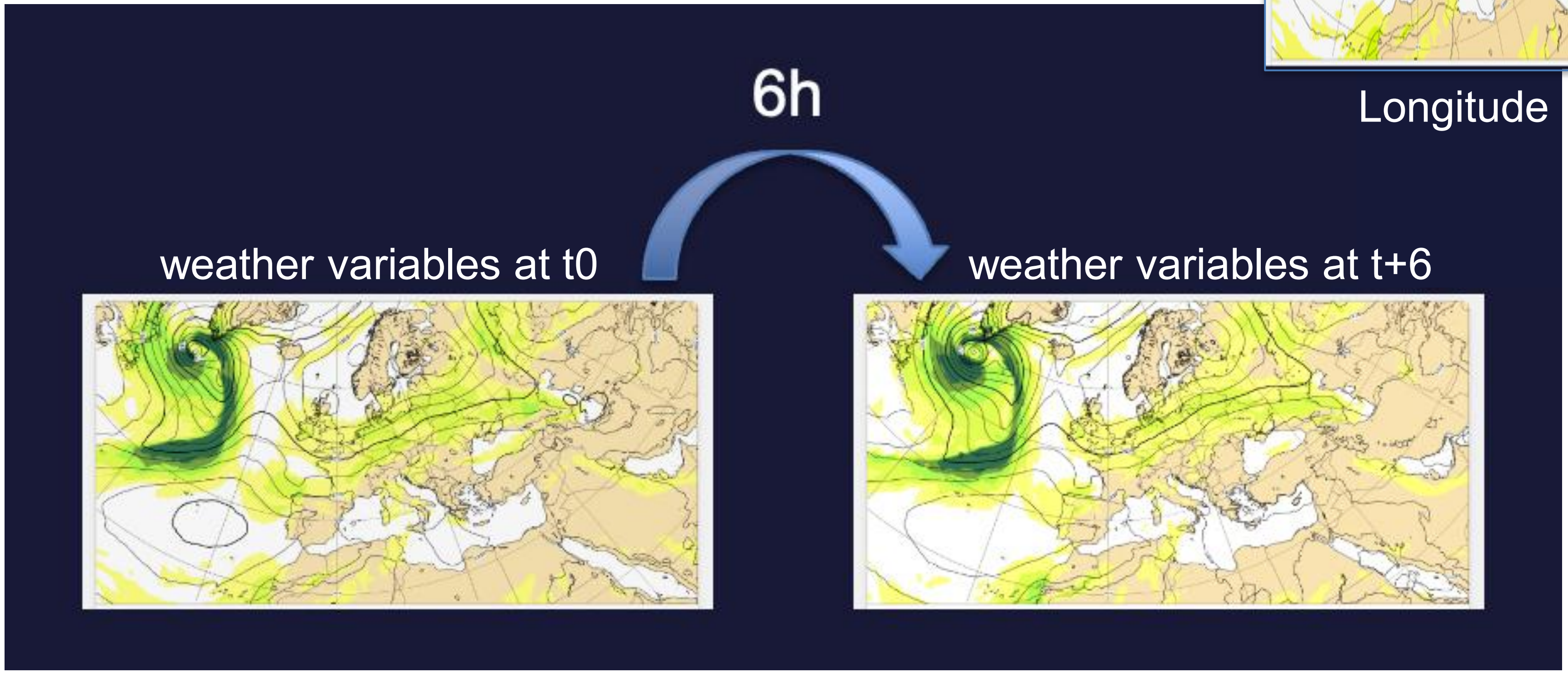


- If you need to read data by timeseries. Store the data by timeseries.
- If you read training samples with 100 parameters for 1 date, do save files containing 100 parameter for 1 dates (but ”it depends”)
- General solution: create a dataset dedicated to your training task (perform the transposition **offline** if required).

Other examples at ECMWF

- ECMWF AIFS – Forecasting Weather in next 6-hour time slice
- Needs access to **previous 6-hour time slice** to predict next step
- Dataset dimensions: [**time**, space, variable] (**bold** = chunked dimension)

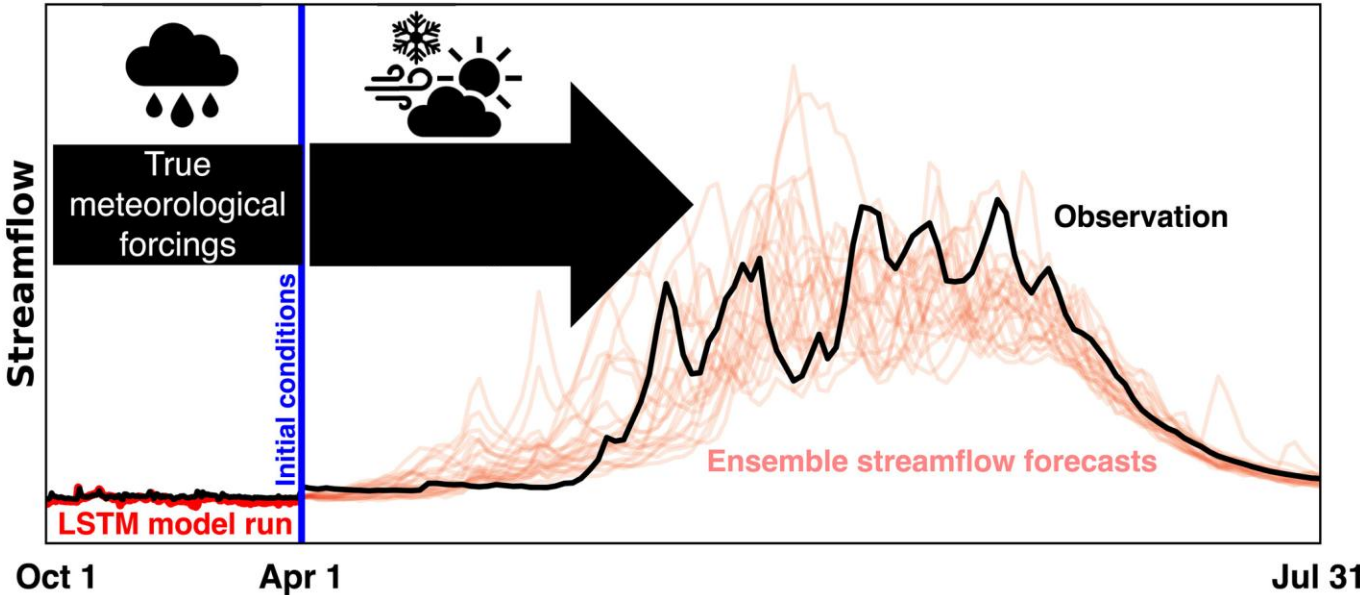


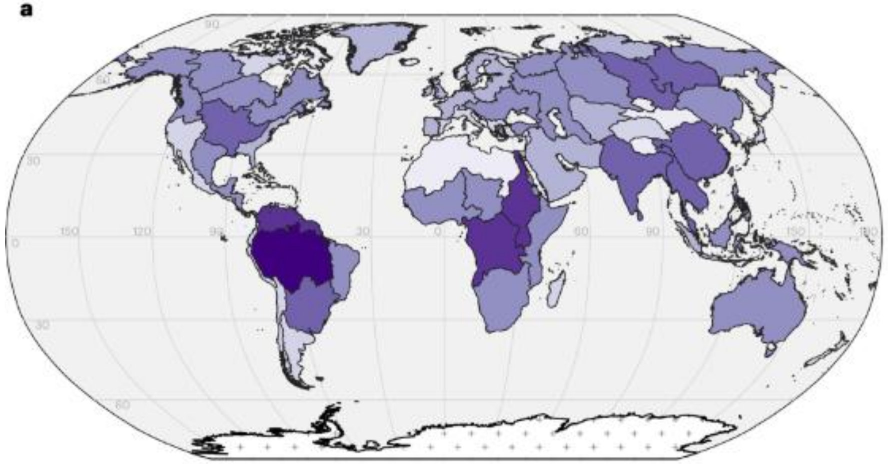


Other examples at ECMWF

Credit:
Maria Luisa Taccari
Kenza Tazi

- ECMWF AIFL – Streamflow forecasting using a Long Short-Term Memory Neural Network (LSTM)
- Needs access to **~180 days** of previously data to predict next step
- Dataset dimensions: [*time*, **space**, variable] (**bold** = chunked dimension)





Global hydrological catchments

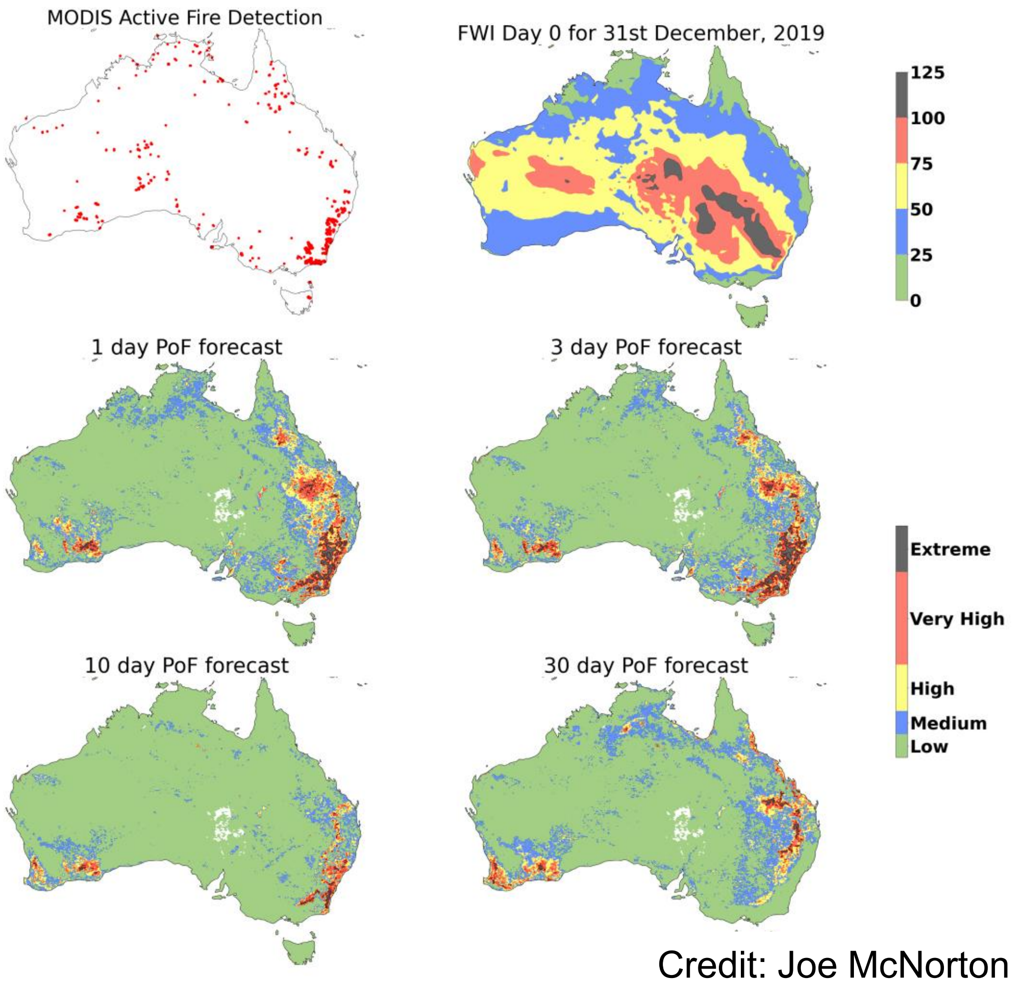
Other examples at ECMWF

- ECMWF Probability of Fire (PoF) – Predicting fire probabilities with XGBoost
  - (training on a subset of the data = 10%)

- Needs access to **instantaneous** data for set of variables for single grid points to predict probability of fire

- Dataset dimensions: [**time**, **space**, variable] (**bold** = chunked dimension)

Time	Lat	Lon	Weather	Fuel load	Static feats	MODIS active fires



Credit: Joe McNorton

Geophysical Research Letters

Research Letter | Open Access

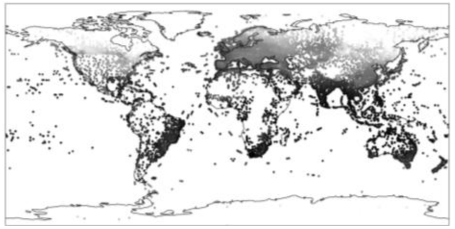
A Global Probability-Of-Fire (PoF) Forecast

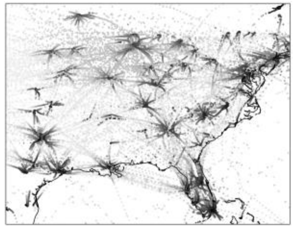
J. R. McNorton, F. Di Giuseppe, E. Pinnington, M. Chantry, C. Barnard

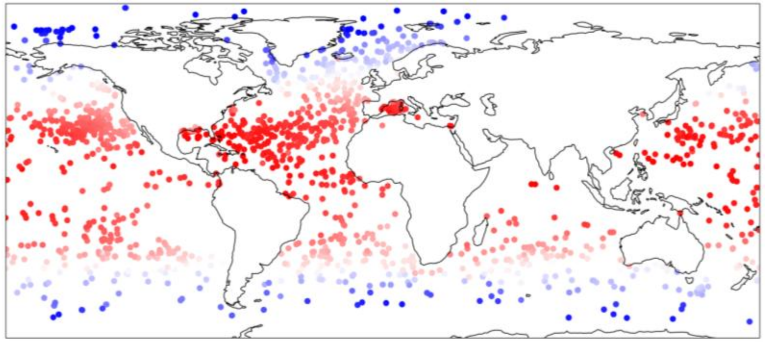
First published: 14 June 2024 | https://doi.org/10.1029/2023GL107929 |

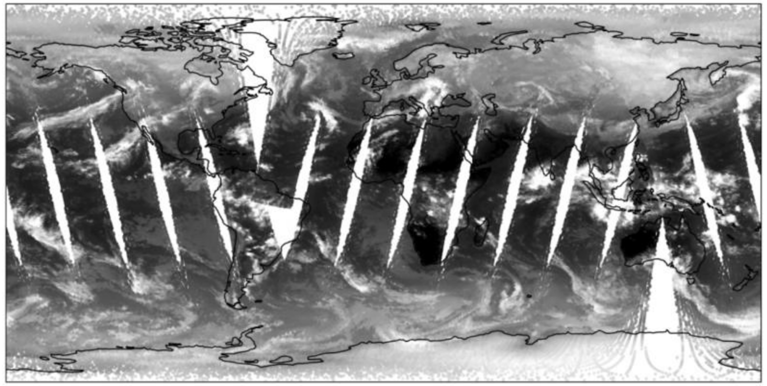
Other examples at ECMWF

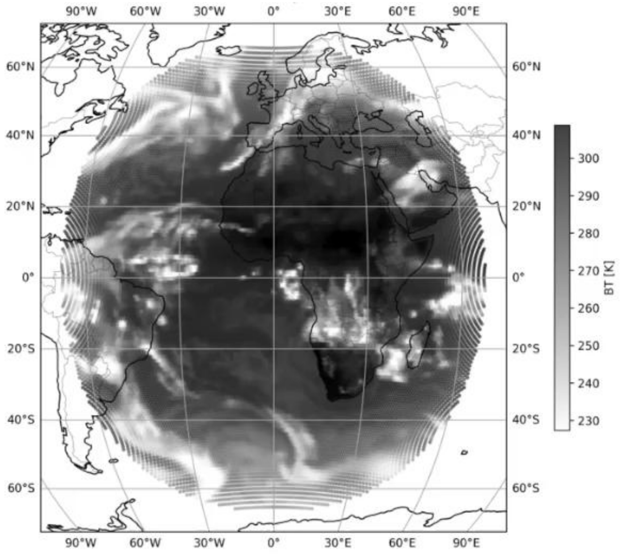
- ECMWF Direct Observation Prediction – Predicting observations into the future
- Gets observations in a time window (e.g. 12hrs) and predicts next time window of observations
- Dataset dimensions: [**time**, **space**, **observation-type**, variable] (**bold** = chunked dimension)



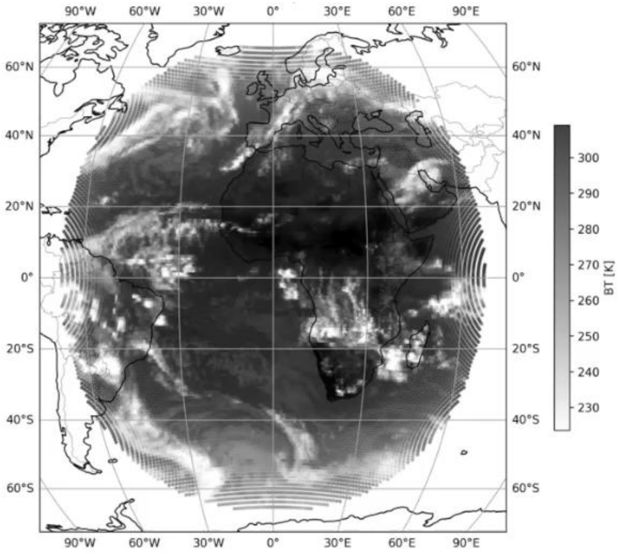








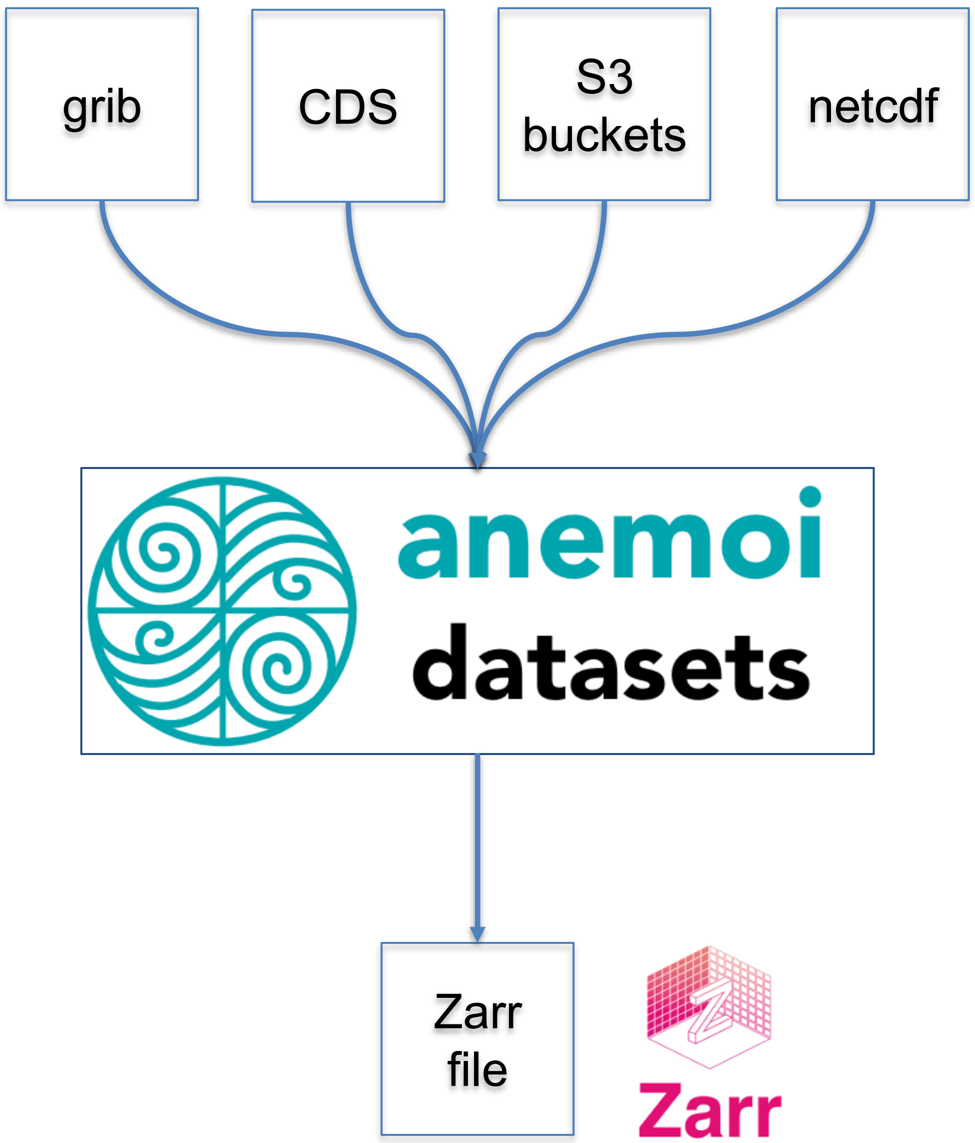
Model prediction



Target Observations

Creating datasets optimized for training

- To make training efficient we often have to rewrite or “rechunk” our data to match the read patterns
- Lots of efficient data formats to do this: HDF5, NetCDF4, Grib, Zarr, Parquet (tabular data), Cloud Optimized Geotiffs, …
- For examples at ECMWF we are using **Anemoi-Datasets** to rewrite data into optimized Zarr datasets for Machine Learning
- https://github.com/ecmwf/anemoi-datasets
- More info @ Thursday 9am Harrison + Mat



Further reading

DATA MOVEMENT IS ALL YOU NEED: A CASE STUDY ON OPTIMIZING TRANSFORMERS

Andrei Ivanov [*1] Nikoli Dryden [*1] Tal Ben-Nun [1] Shigang Li [1] Torsten Hoefler [1]

GLENN K. LOCKWOOD

SEARC

Writing to make sense of HPC, storage, and system design

HPC · STORAGE · PERSONAL · ABOUT THIS BLOG · MY WEBSITE

Labels

ai

hpc

February 01, 2025

LLM TRAINING WITHOUT A PARALLEL FILE SYSTEM

FLASHATTENTION-2:
Faster Attention with Better Parallelism and Work Partitioning

Tri Dao[1,2]

Summary: But what does this all mean for you?

- Most of the complexity of managing memory is handled by PyTorch, Anemoi, etc.
  - Don’t over engineer if working with smaller datasets!
- Try to fill your GPU memory as much as possible
  - Increase batch size, #channels, parameter count
- Try to fill your CPU memory as much as possible
  - Play with the num_workers parameter
- Think about where your data is coming from
  - Downloading from the internet vs loading from an SSD vs loading from a high performance file system
  - How is the stored and what are your access patterns?

- Answers will depend on your problem!

More info on optimization and profiling
Friday 10:45am
Cathal and Jan

Other Data Issues 👀

- Data is a fundamental quantity in Machine Learning

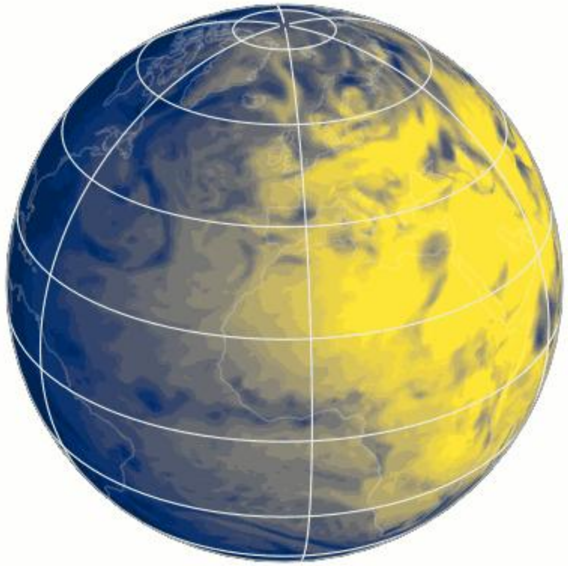
- Need to make sure what we are presenting to the model “makes sense” (e.g. times aligned, variables consistent, re-gridding reasonable)

- Models can inherit all the positive and negative aspects of their training data

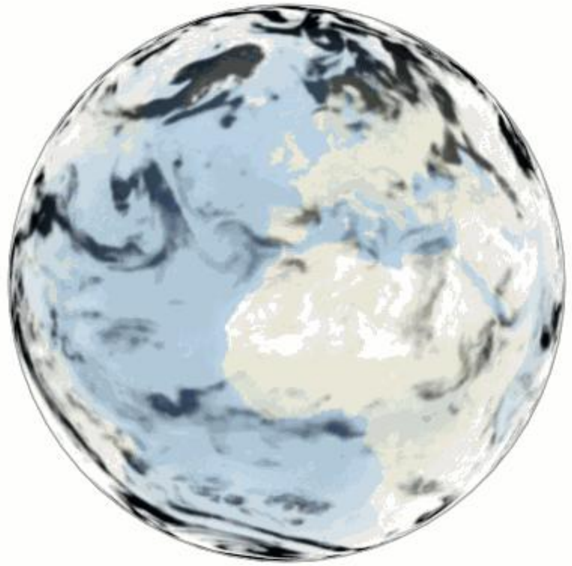
- Can present problems if there are data distribution shifts when fine-tuning or running inference with models!

Other Data Issues Example – AIFS 1.0

- Build new datasets through `anemoi-datasets` github.com/ecmwf/anemoi-datasets
  - New variables, new years of data, etc.
  - Now training on
    - ERA5: 1979-2022 (1-step, 6-hour forecasts)
    - IFS-Operations: 2016-2022 (fine-tuning/rollout, 6 to 72-hour forecasts)
- Decide which features to include in release from `anemoi-core` github.com/ecmwf/anemoi-core
- Train example models at **o96** (~1 degree) resolution with varying configurations
- For most promising models train versions at **n320** (~30 km) resolution
- Perform more rigorous validation and decide on final candidate



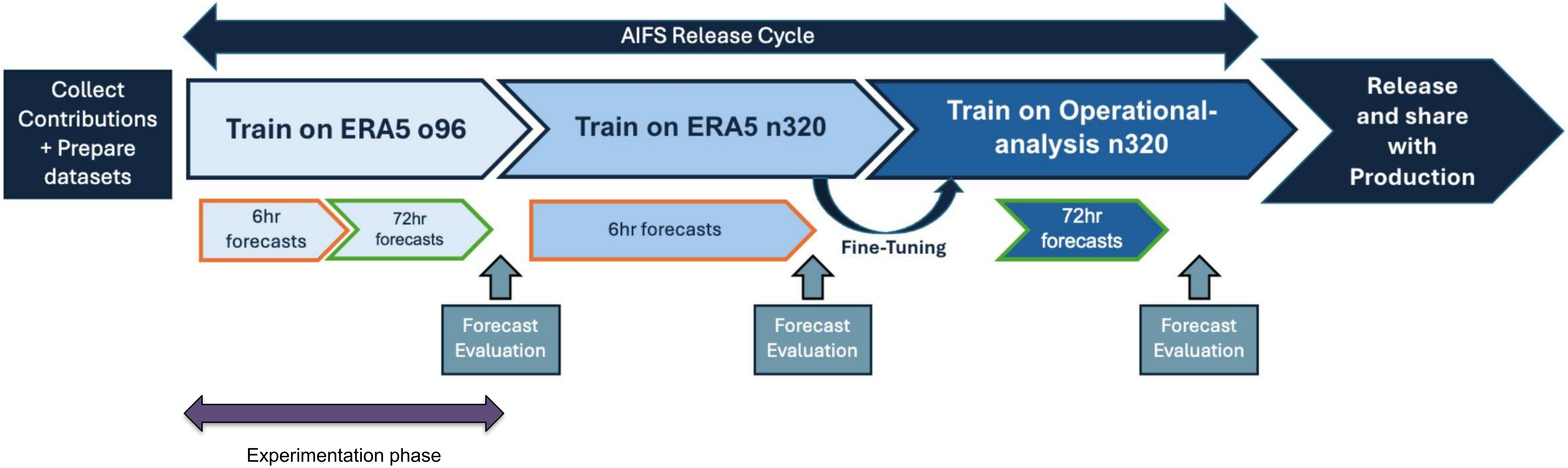
Surface Solar Radiation



Cloud Cover

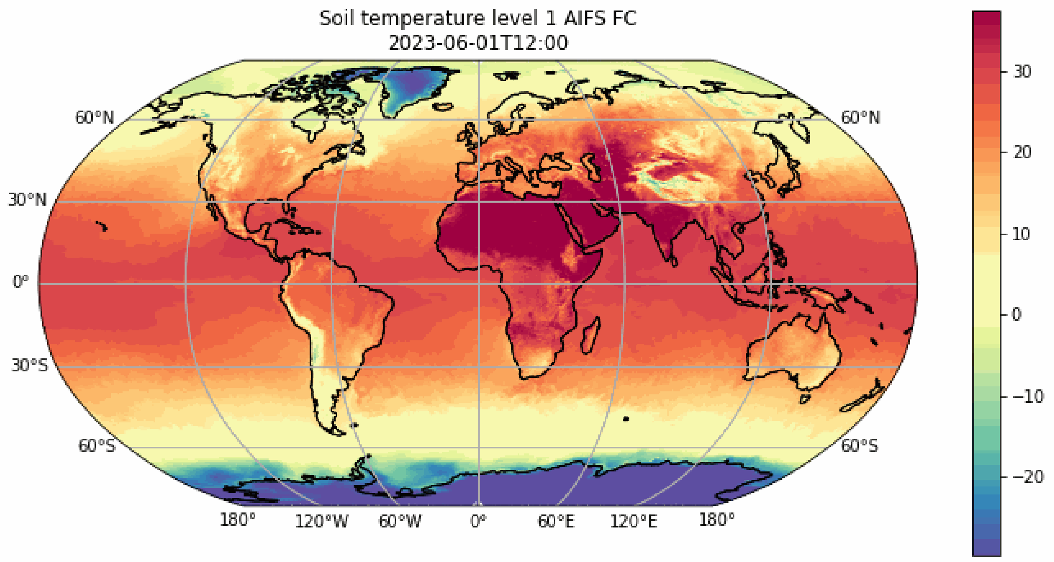
ECMWF

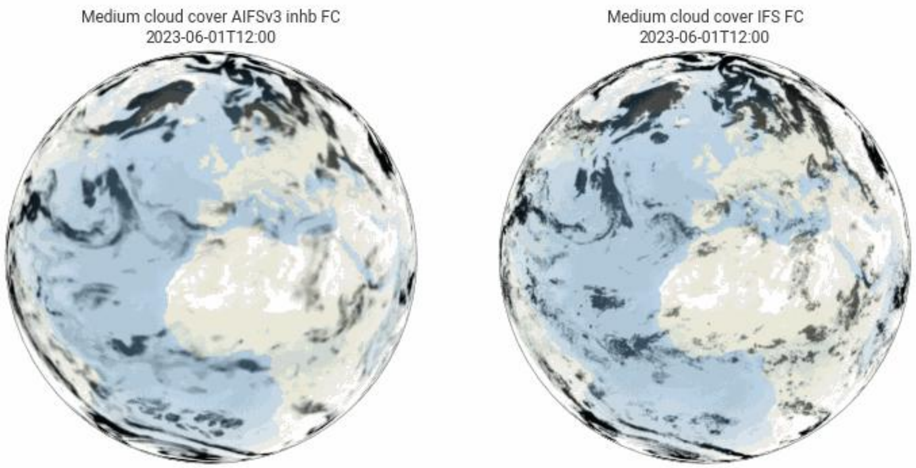
AIFS Single v1



Other Data Issues Example – AIFS 1.0

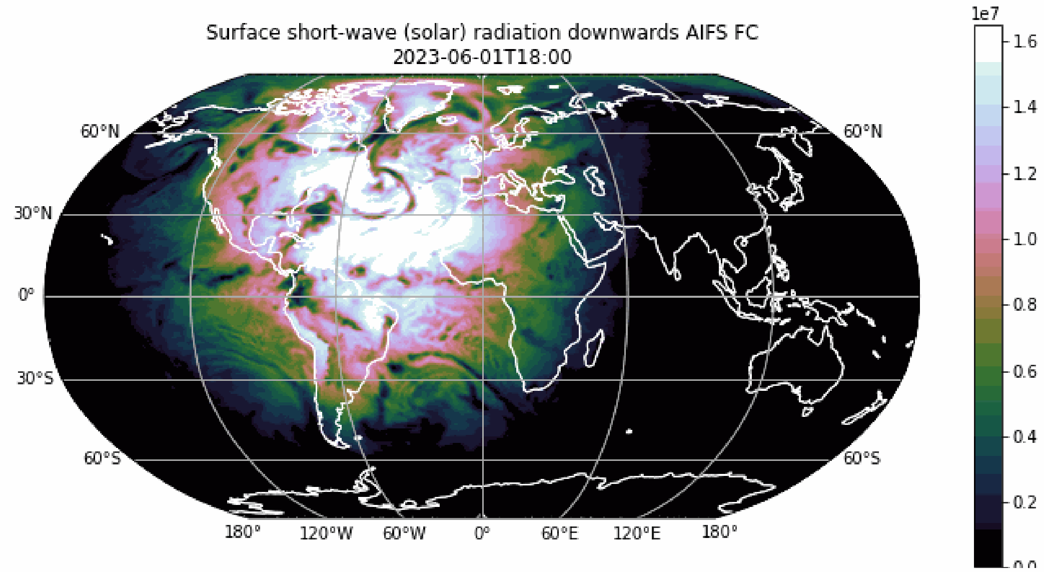
- New variables added to AIFSv1
  - Prognostics:
    - Soil moisture (layer 1 + 2) 💧
    - Soil temperature (layer 1 + 2) 🌡️
  - Diagnostics:
    - Cloud covers (tcc, lcc, mcc, hcc) ☁️
    - Surface radiations (strd, ssrd) 🌞
    - 100m winds (100u, 100v) 🌬️
    - Snow fall (sf) ❄️
    - Runoff (ro) 🏃
- AIFS copes well with increased number of target variables

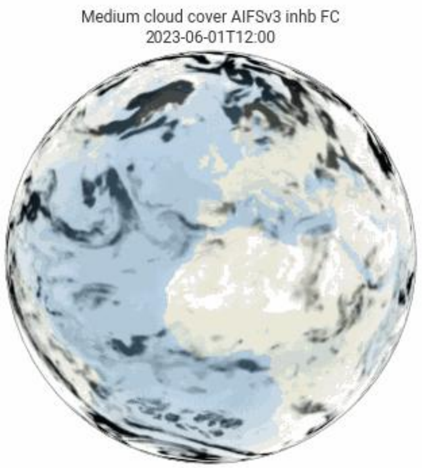


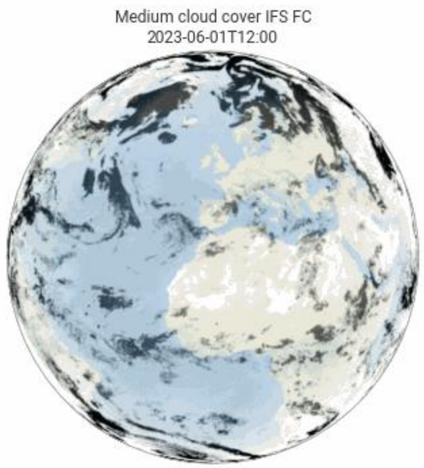


Other Data Issues Example – AIFS 1.0

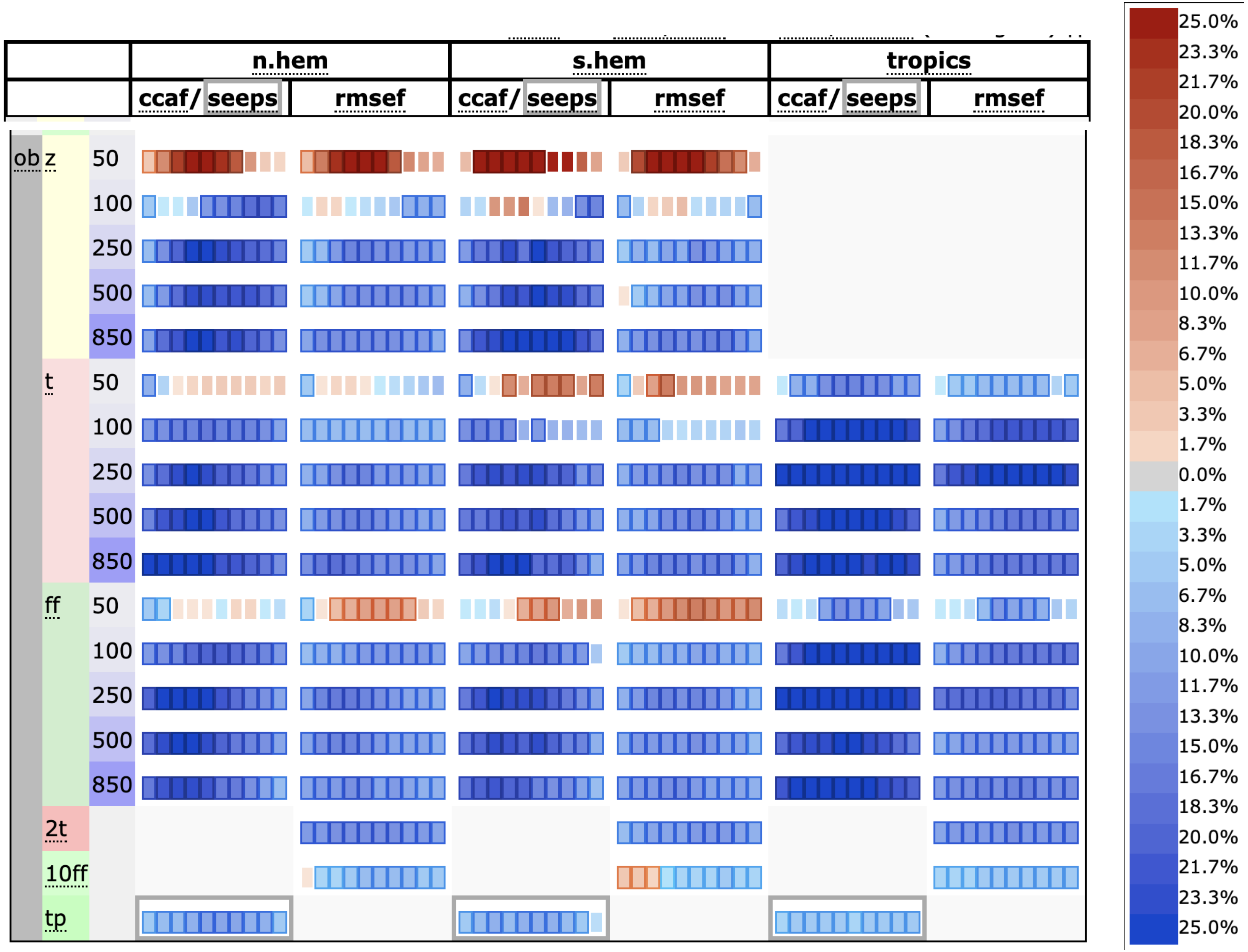
- New variables added to AIFSv1
  - Prognostics:
    - Soil moisture (layer 1 + 2) 💧
    - Soil temperature (layer 1 + 2) 🌡️
  - Diagnostics:
    - Cloud covers (tcc, lcc, mcc, hcc) ☁️
    - Surface radiations (strd, ssrd) 🌞
    - 100m winds (100u, 100v) 🌬️
    - Snow fall (sf) ❄️
    - Runoff (ro) 🏃
- AIFS copes well with increased number of target variables



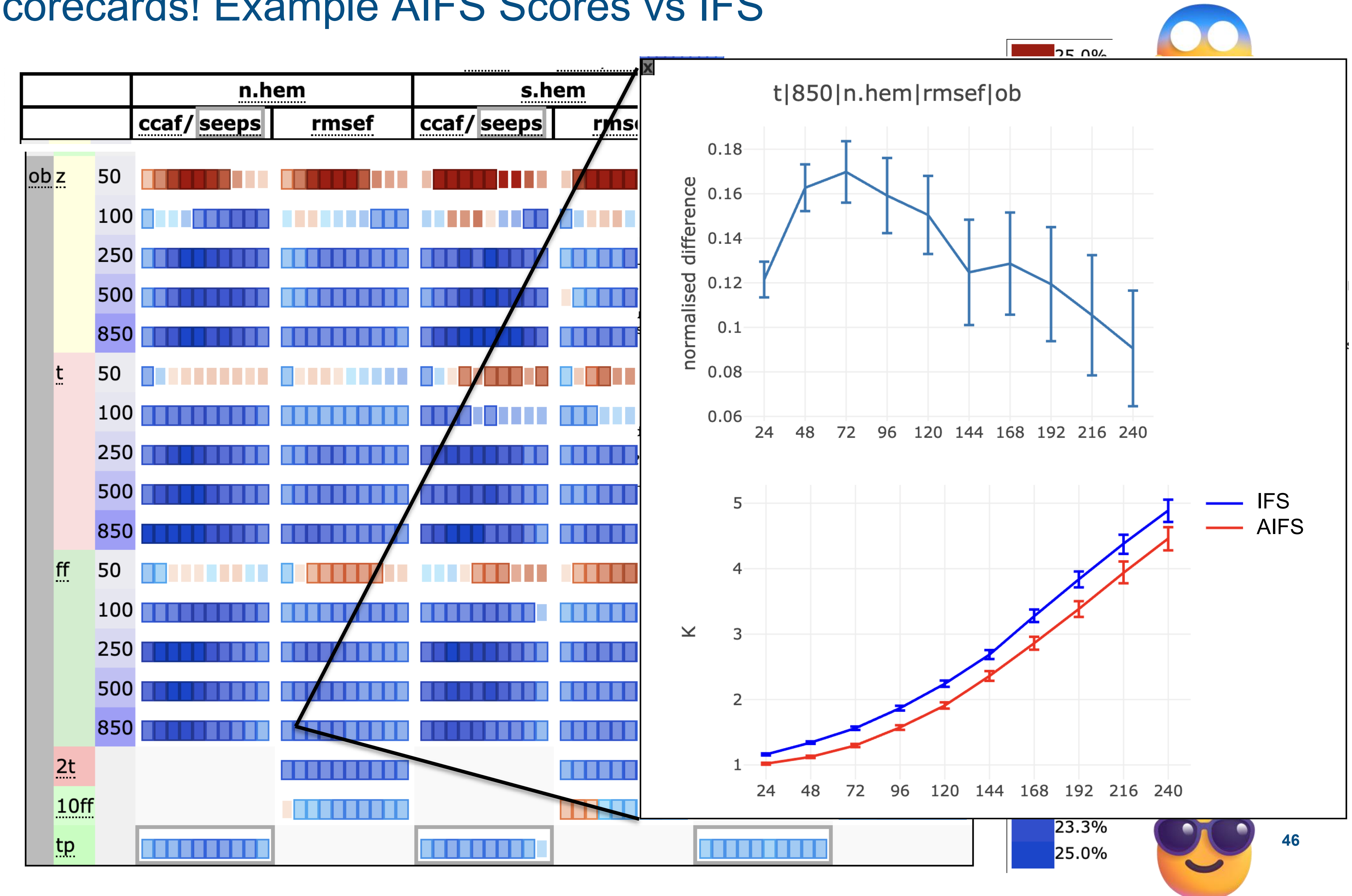




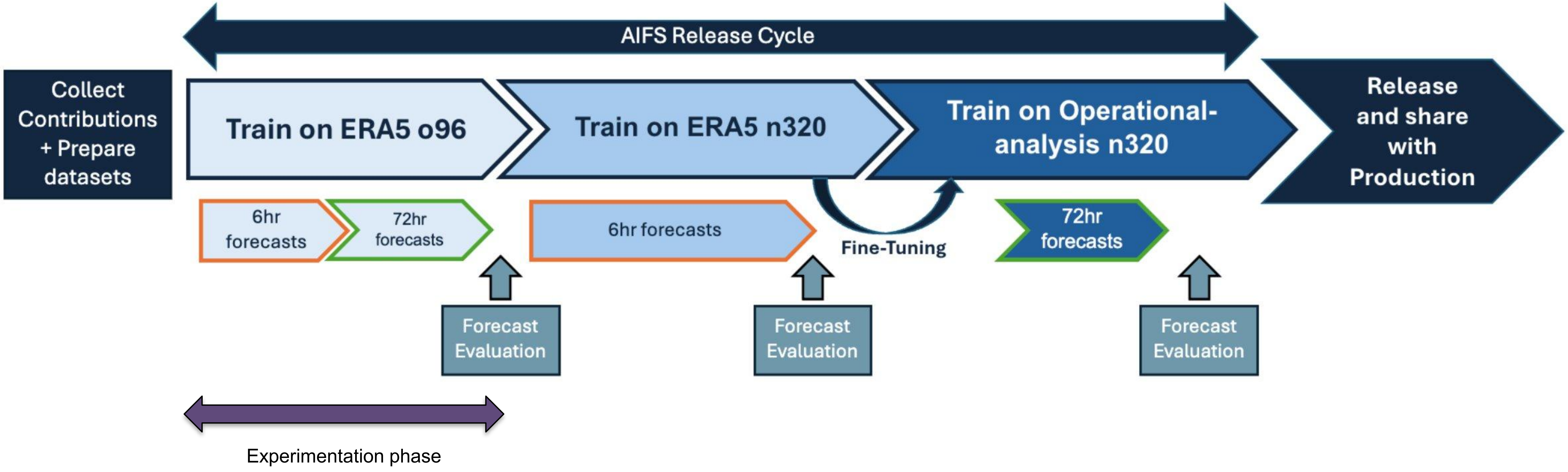
Intro to Scorecards! Example AIFS Scores vs IFS

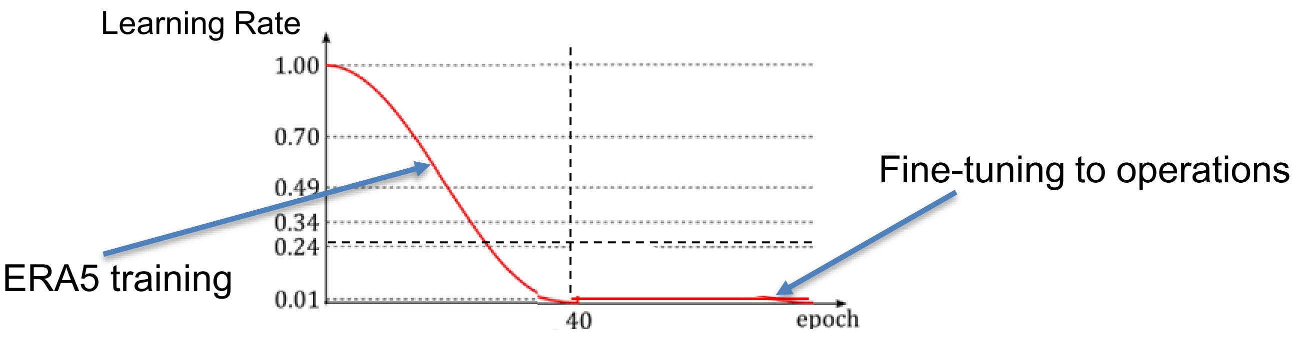


Intro to Scorecards! Example AIFS Scores vs IFS

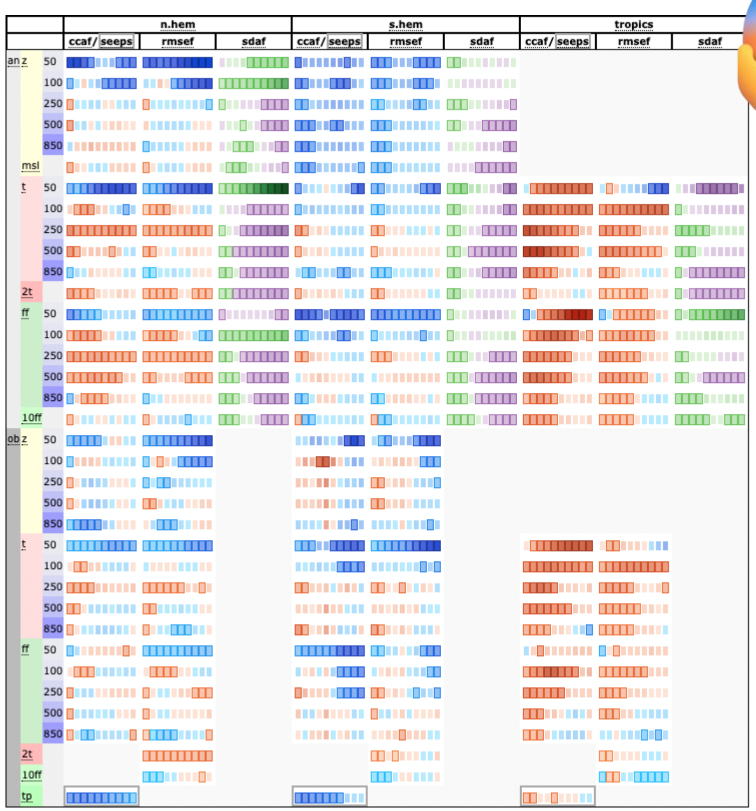


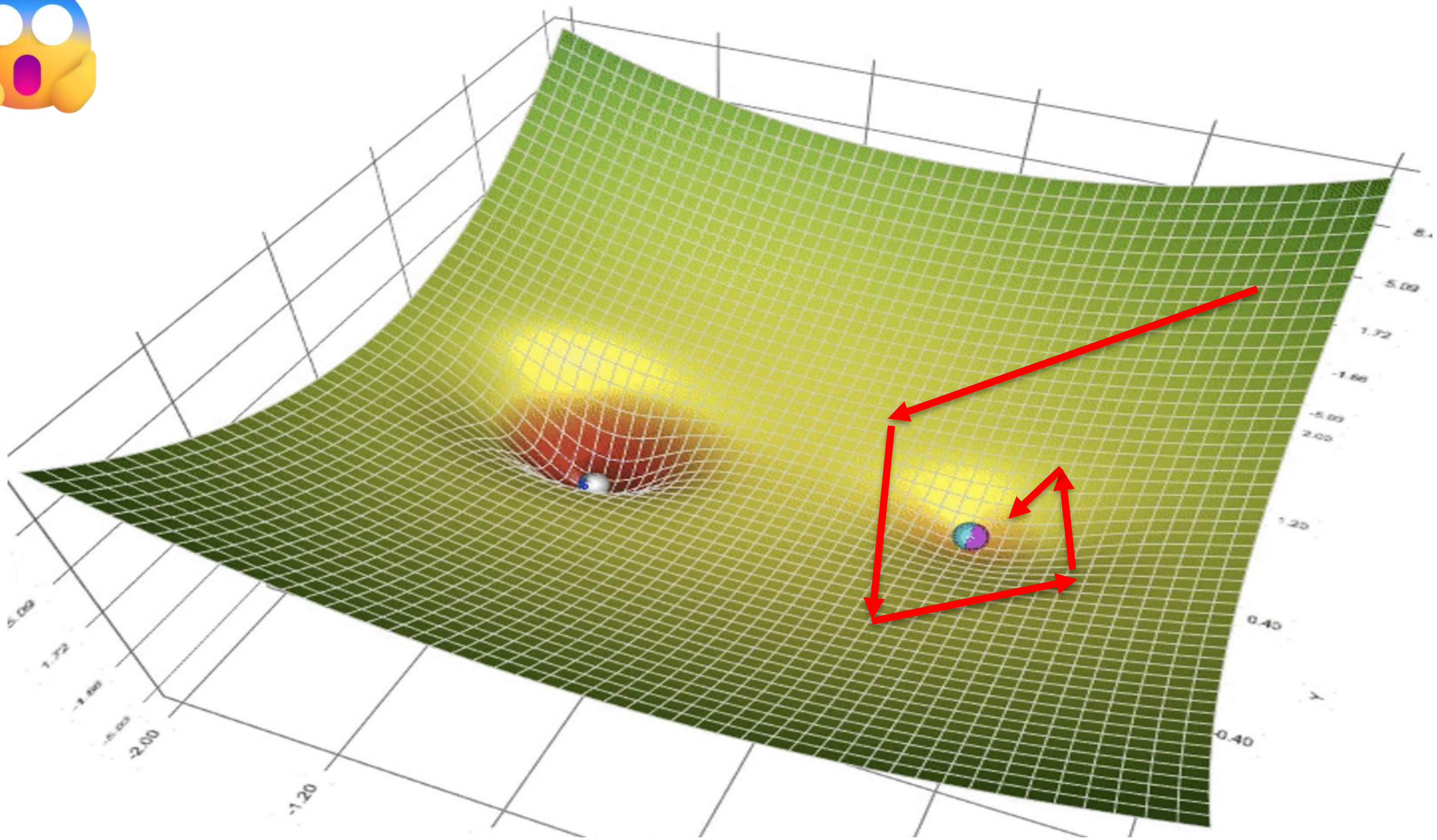
AIFS Single v1

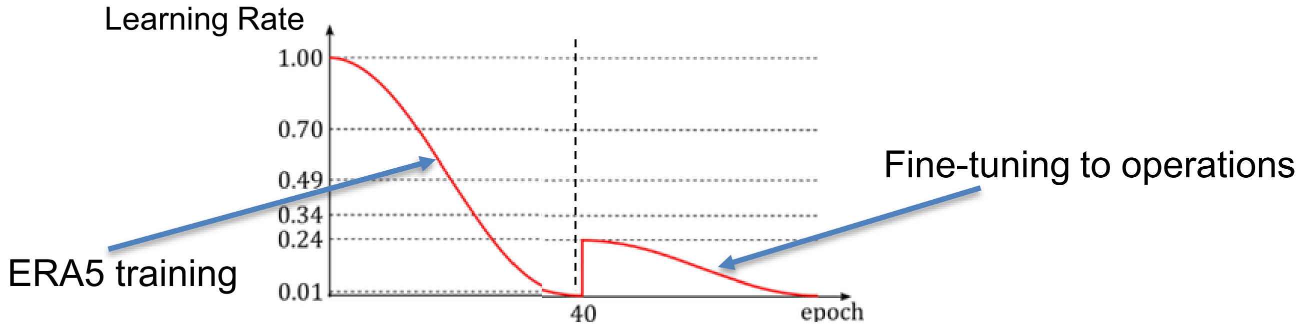




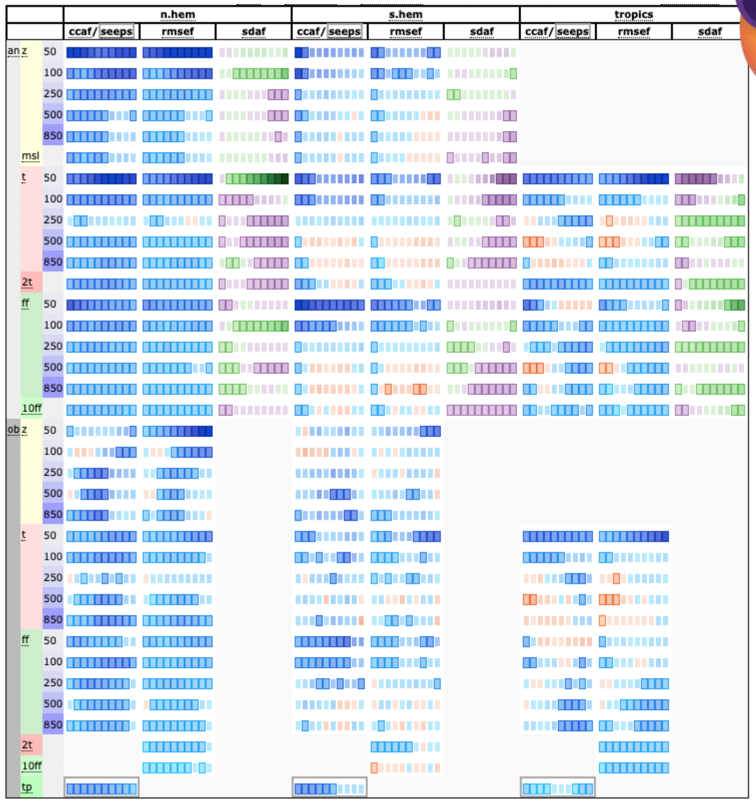
AIFSv1 candidate1 vs AIFSv0.2.1

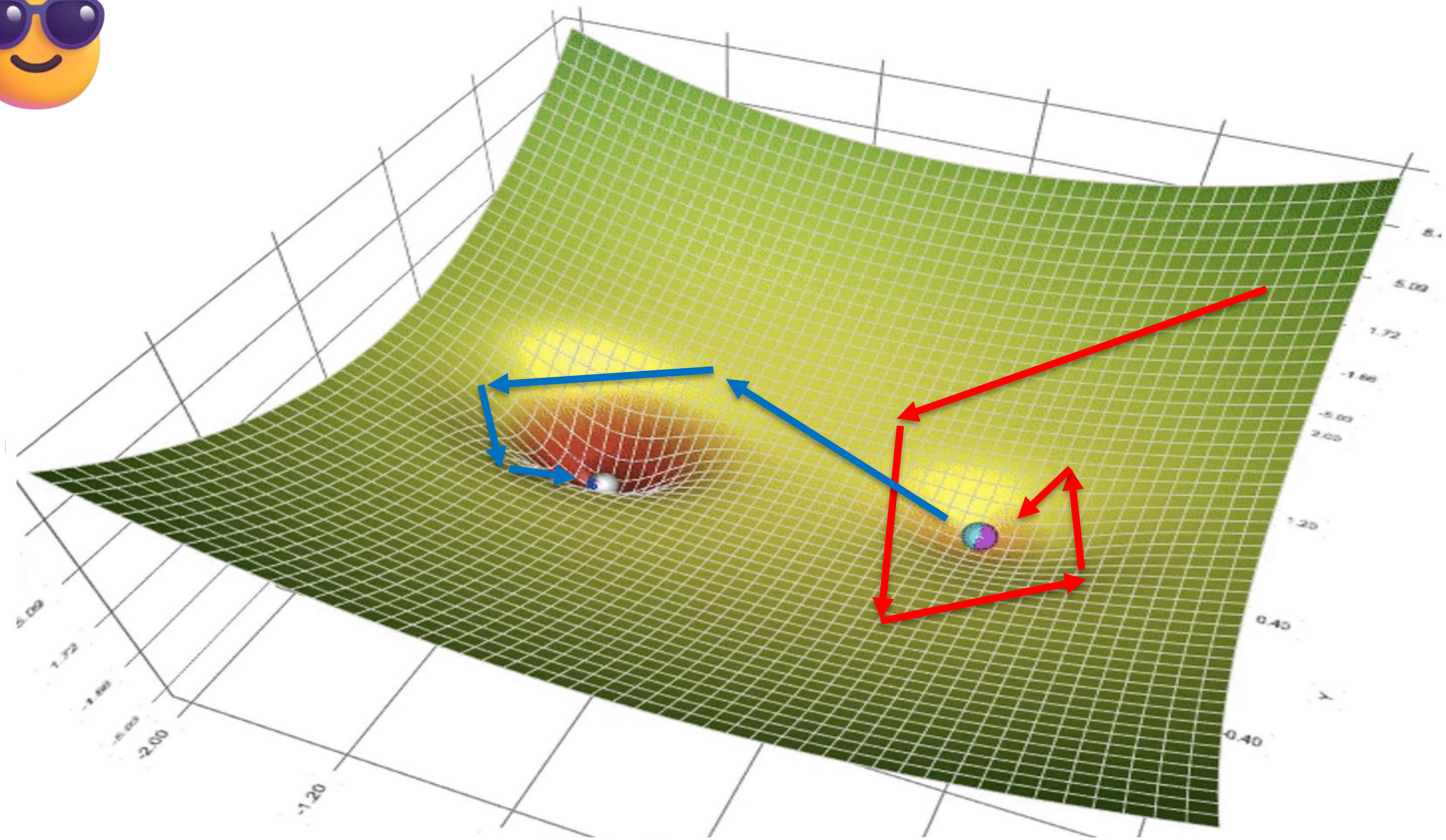


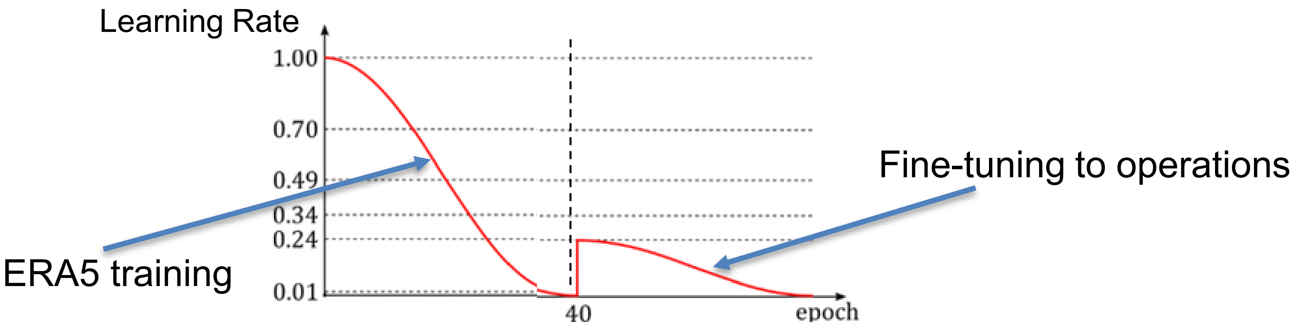




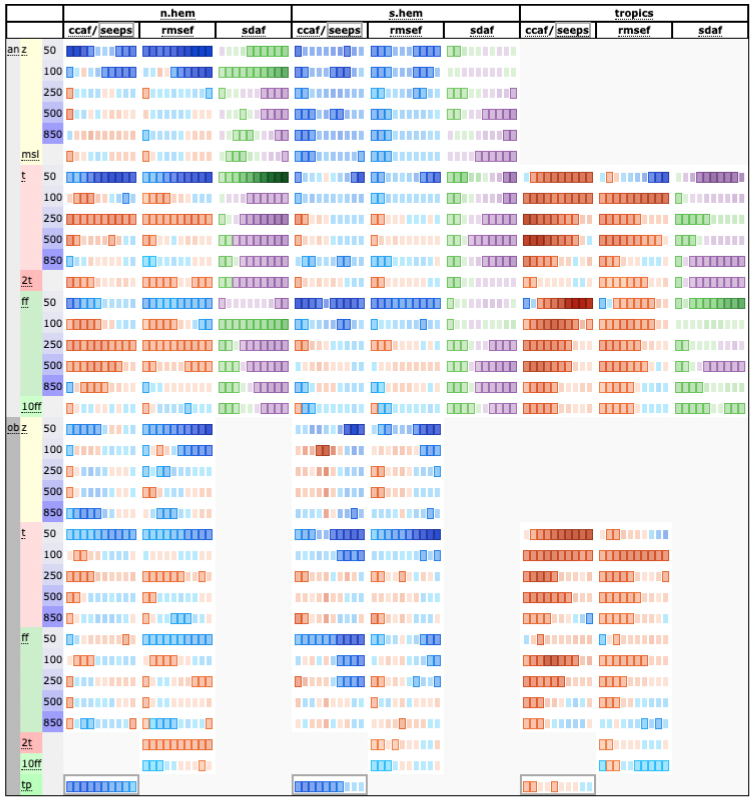
AIFSv1 candidate2 vs AIFSv0.2.1



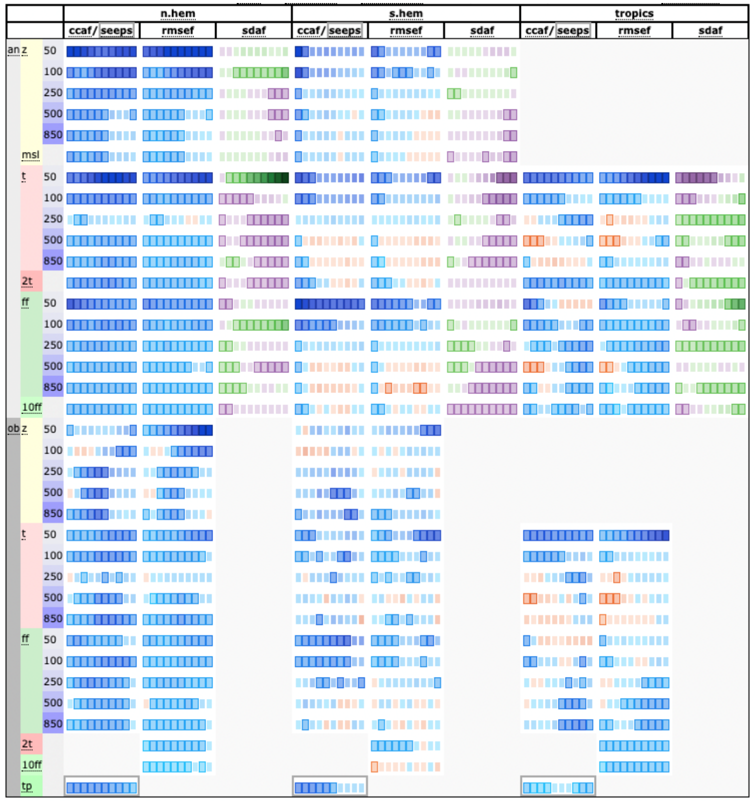




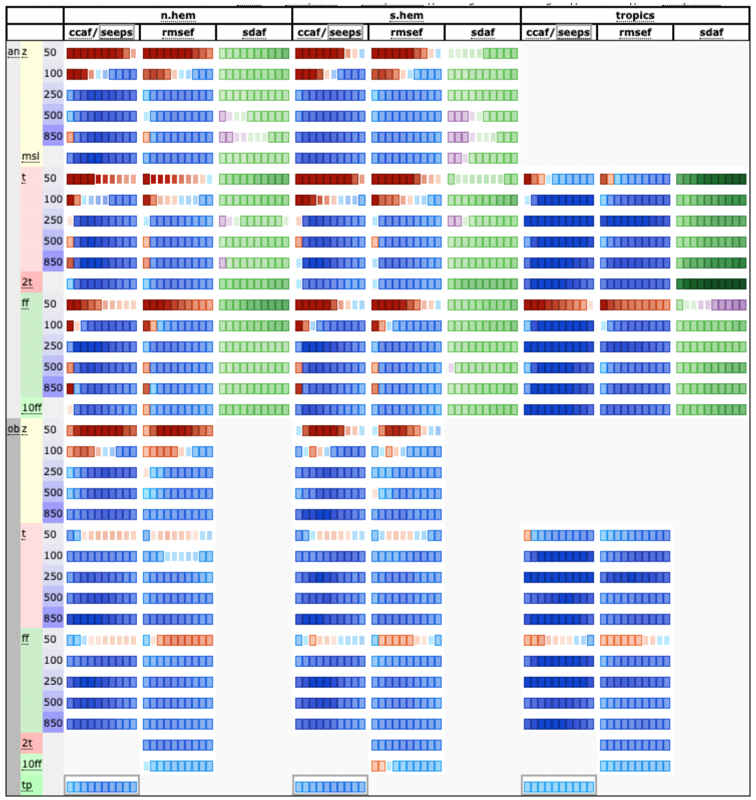
AIFSv1 candidate1 vs AIFSv0.2.1



AIFSv1 candidate2 vs AIFSv0.2.1



AIFSv1 candidate2 vs IFS



ECMWF

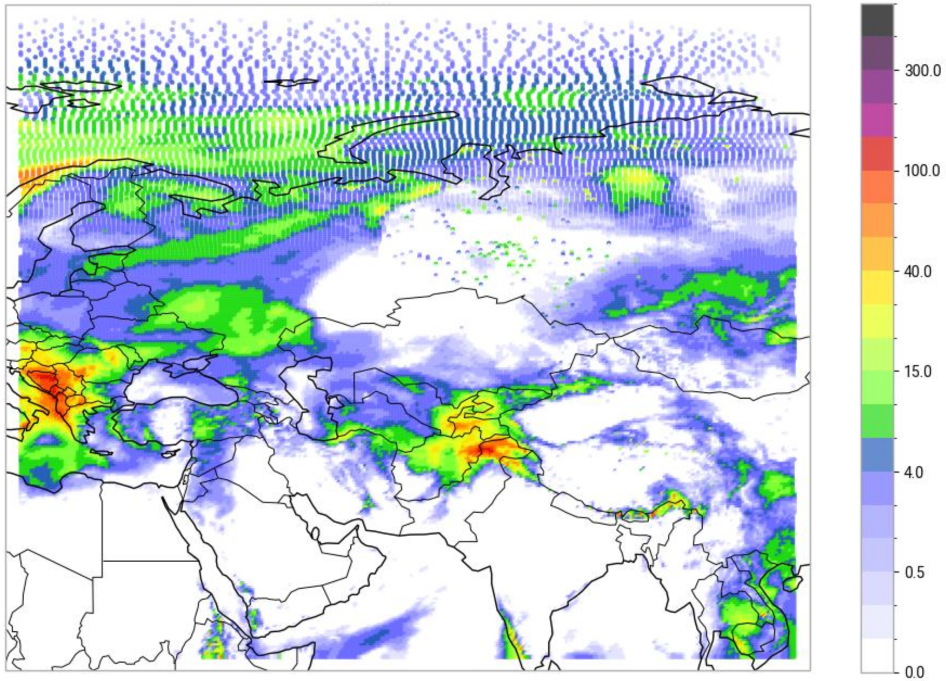
Other Data Issues Example – Rain Pox! 🤒

- After going into operations in Feb 2025 the AIFS started to exhibit spurious points of rainfall

- Linked to change in initial conditions with new Cycle 49R1 release at ECMWF

- Issues with soil moisture that the model hadn't seen in training triggering isolated instances of rainfall

AIFS Single



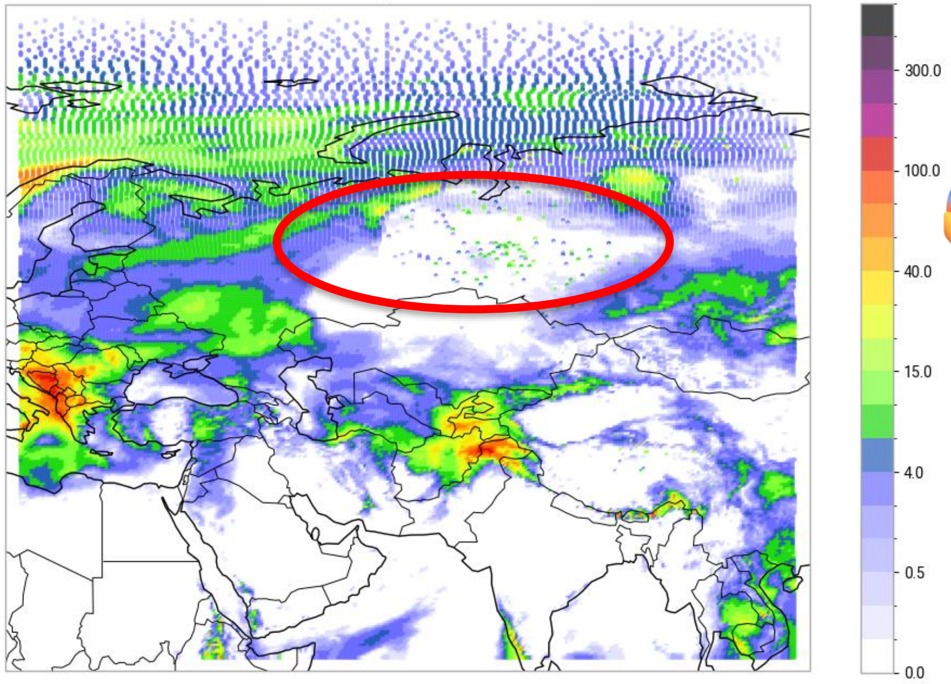
Other Data Issues Example – Rain Pox! 🥵

- After going into operations in Feb 2025 the AIFS started to exhibit spurious points of rainfall

- Linked to change in initial conditions with new Cycle 49R1 release at ECMWF

- Issues with soil moisture that the model hadn't seen in training triggering isolated instances of rainfall

AIFS Single

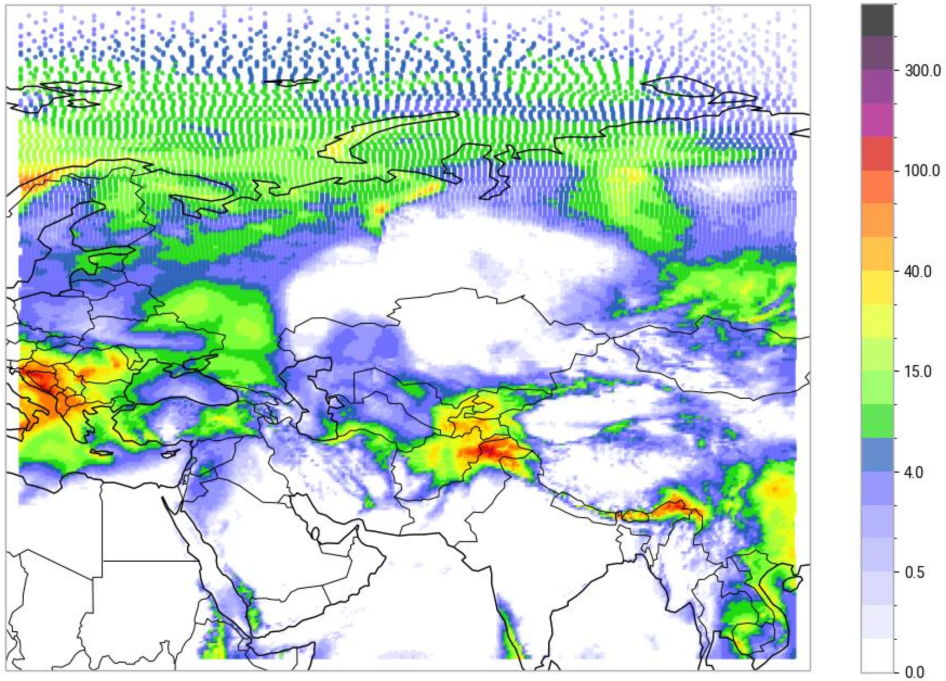


ECMWF

Other Data Issues Example – Rain Pox! 🤒

- After going into operations in Feb 2025 the AIFS started to exhibit spurious points of rainfall
- Linked to change in initial conditions with new Cycle 49R1 release at ECMWF
- Issues with soil moisture that the model hadn't seen in training triggering isolated instances of rainfall
- Fix: Down-weight soil moisture in loss, avoid learning unphysical relationships

AIFS Pox Fix



Other Data Issues Example – Rain Pox! 🤒

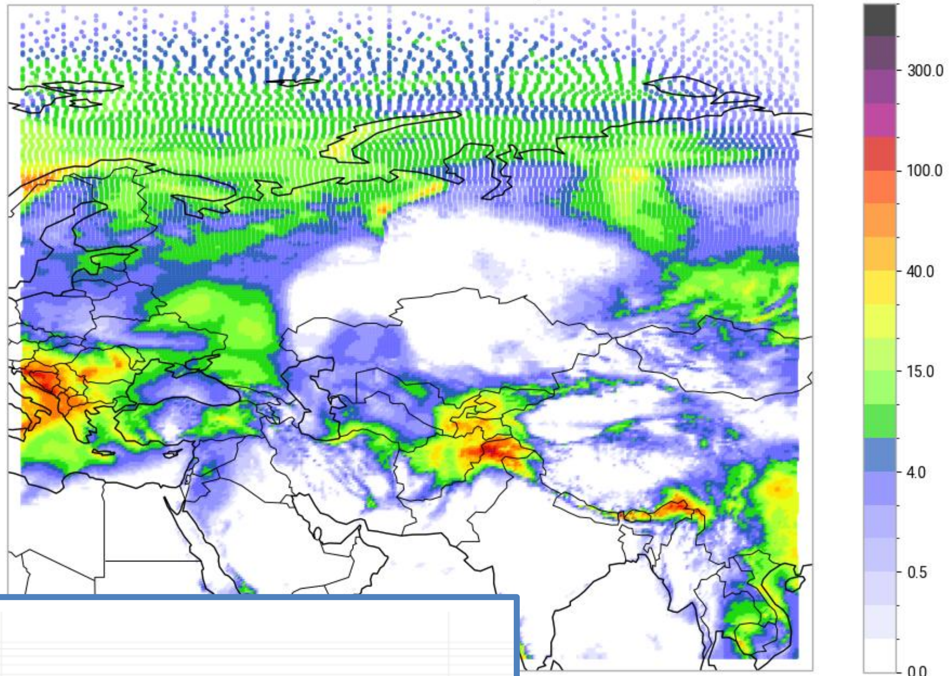
- After going into operations in Feb 2025 the AIFS started to exhibit spurious points of rainfall

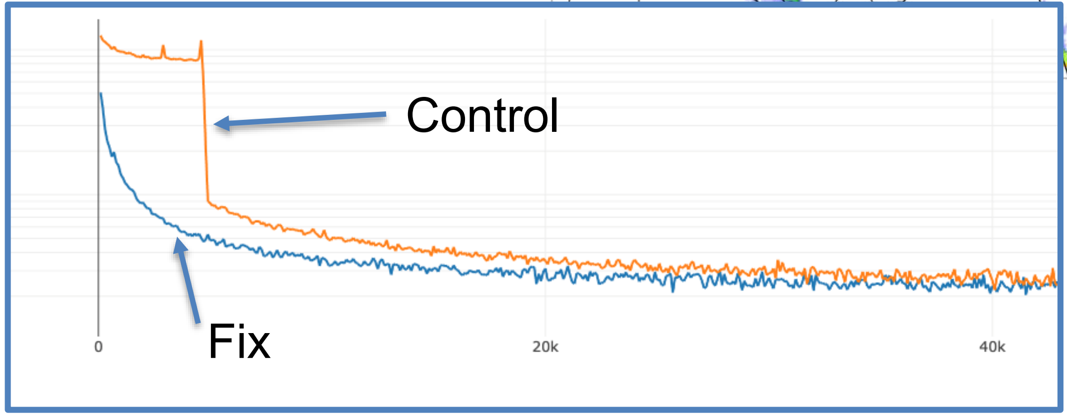
- Linked to change in initial conditions with new Cycle 49R1 release at ECMWF

- Issues with soil moisture that the model hadn't seen in training triggering isolated instances of rainfall

- Fix: Down-weight soil moisture in loss, avoid learning unphysical relationships

AIFS Pox Fix





ECMWF

Summary: Other Data Issues

- Data is a fundamental quantity in Machine Learning
- Be careful to check data quality and watch for signs things aren't right in the loss
- Data distribution shifts can cause issues!
  - If fine-tuning may have to tweak learning-rates
  - For inference if data shifts happen can impact forecasts!

What we learned & Questions

- Training ML models can be limited by how fast we can load the data
- Need to consider data access patterns and pre-process data accordingly! (can use e.g. Zarr, Anemoi, etc. for this)
- Tweak batch size and num_workers to try and optimize training times once data formatted correctly
- Think about where data is coming from
  - Bring code to the data/compute

- Other Data issues
  - Careful to check data quality (misaligned, inconsistent samples, etc.)
  - If using multiple data sources distribution shifts can cause issues!

- Questions