# Benchmarking for weather/climate

**Hisashi YASHIRO**
Research Scientist
RIKEN Advanced Institute for Computational Science
Kobe, Japan

# Weather/Climate simulations in HPC

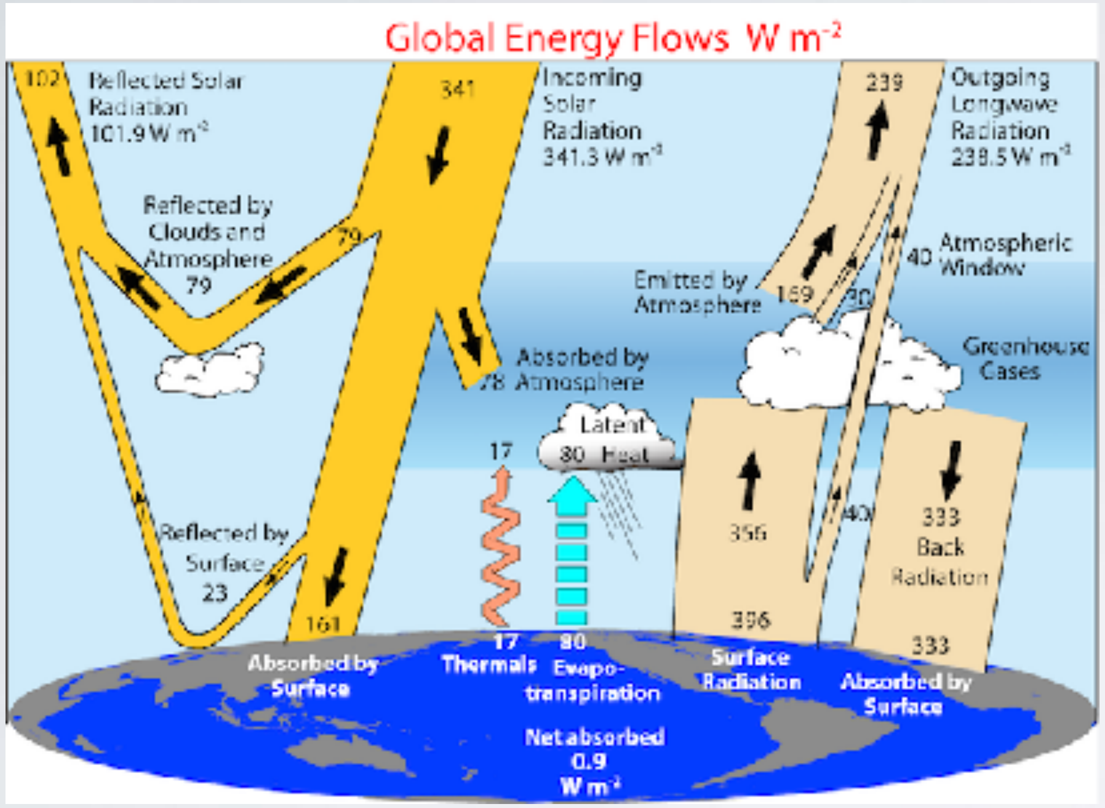
**Weather/Climate simulations produce 'really' big data**

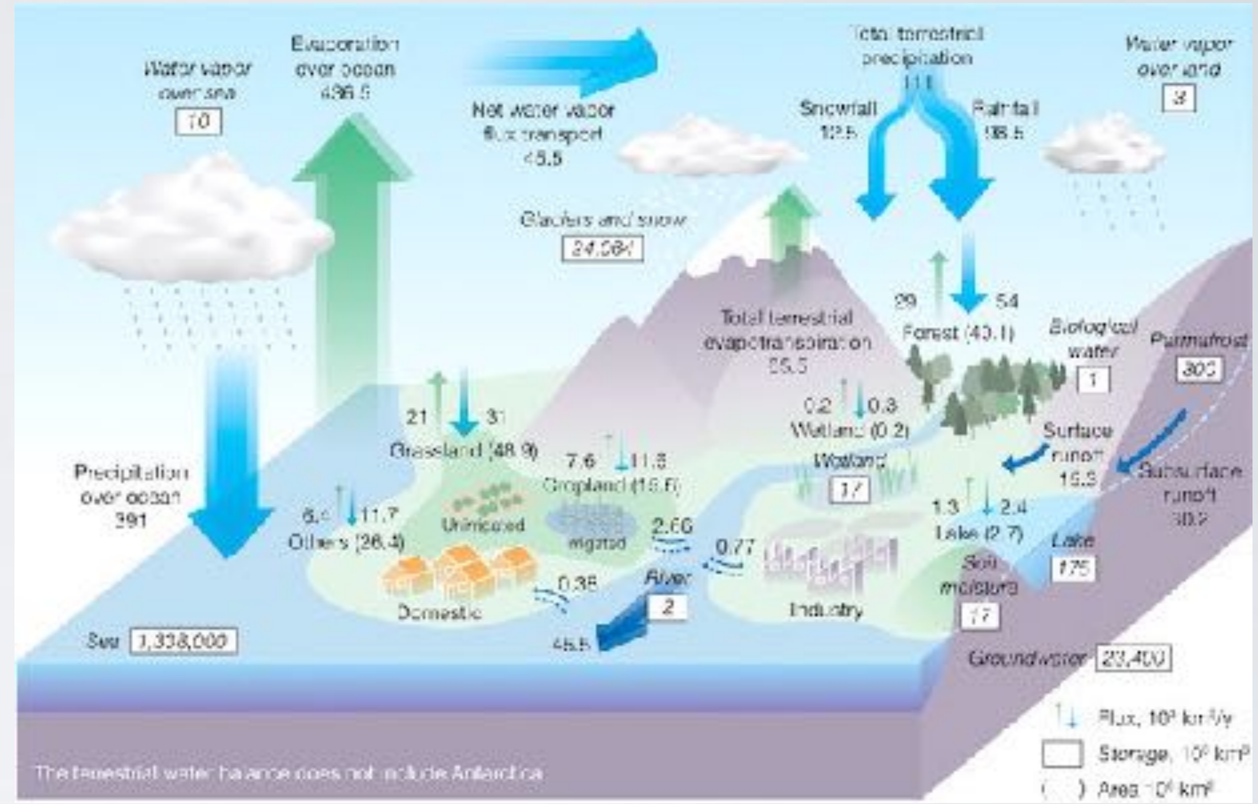
- Big size
- Huge number of the files

**Weather/Climate applications is data-intensive**

- Not only file I/O, but also memory and cache
- Low computational intensity, large memory footprint

# Variables, variables, variables!




**Variables for time integration: 10~100**

- Wind, temperature, pressure
- Tracers: water (gas,liquid,solid), aerosols, other gas species

**Variables for output: 100~1000**

- Prognostics and diagnostics
- States, fluxes, and tendencies

# Coupled Model Intercomparison Project (CMIP)

| | CMIP5 | CMIP6 | CMIP7 |
|---|---|---|---|
| | **2 012,00** | **2 017,00** | **2 022,00** |
| Number of simulated years | 10 000,00 | 10 000,00 | 10 000,00 |
| | | | |
| Data produced PB | 4,35 | 120,56 | 949,34 |
| Number of days to complete / scenario 1 | 50,00 | 50,00 | 50,00 |
| Data produced PB / day / scenario 1 | 0,09 | 2,41 | 18,99 |
| Number of days to complete / scenario 2 | 100,00 | 100,00 | 100,00 |
| Data produced PB / day / scenario 2 | 0,04 | 1,21 | 9,49 |
| Number of days to complete / scenario 3 | 200,00 | 200,00 | 200,00 |
| Data produced PB / day / scenario 3 | 0,02 | 0,60 | 4,75 |

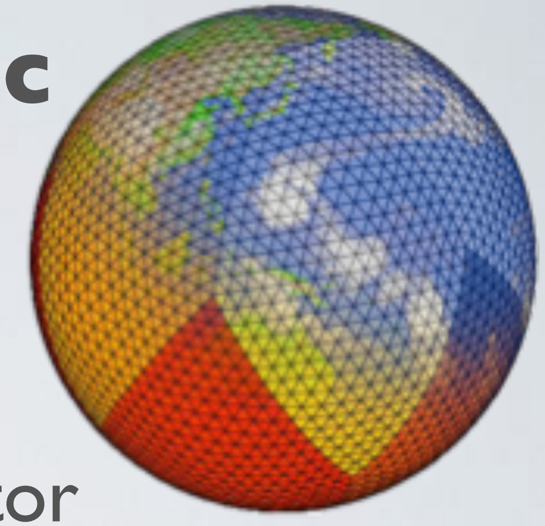
From the presentation of Sébastien Denvil (IPSL) in IS-ENES/PRACE meeting

# Practical cases by super-high resolution model

- **Global sub-km simulation**
- **Ensemble data assimilation system**
- **Simulation on the GPU-based system**

# Global sub-km simulation

# NICAM



## **N**on-hydrostatic **I**cosahedral **A**tmospheric **M**odel (NICAM)

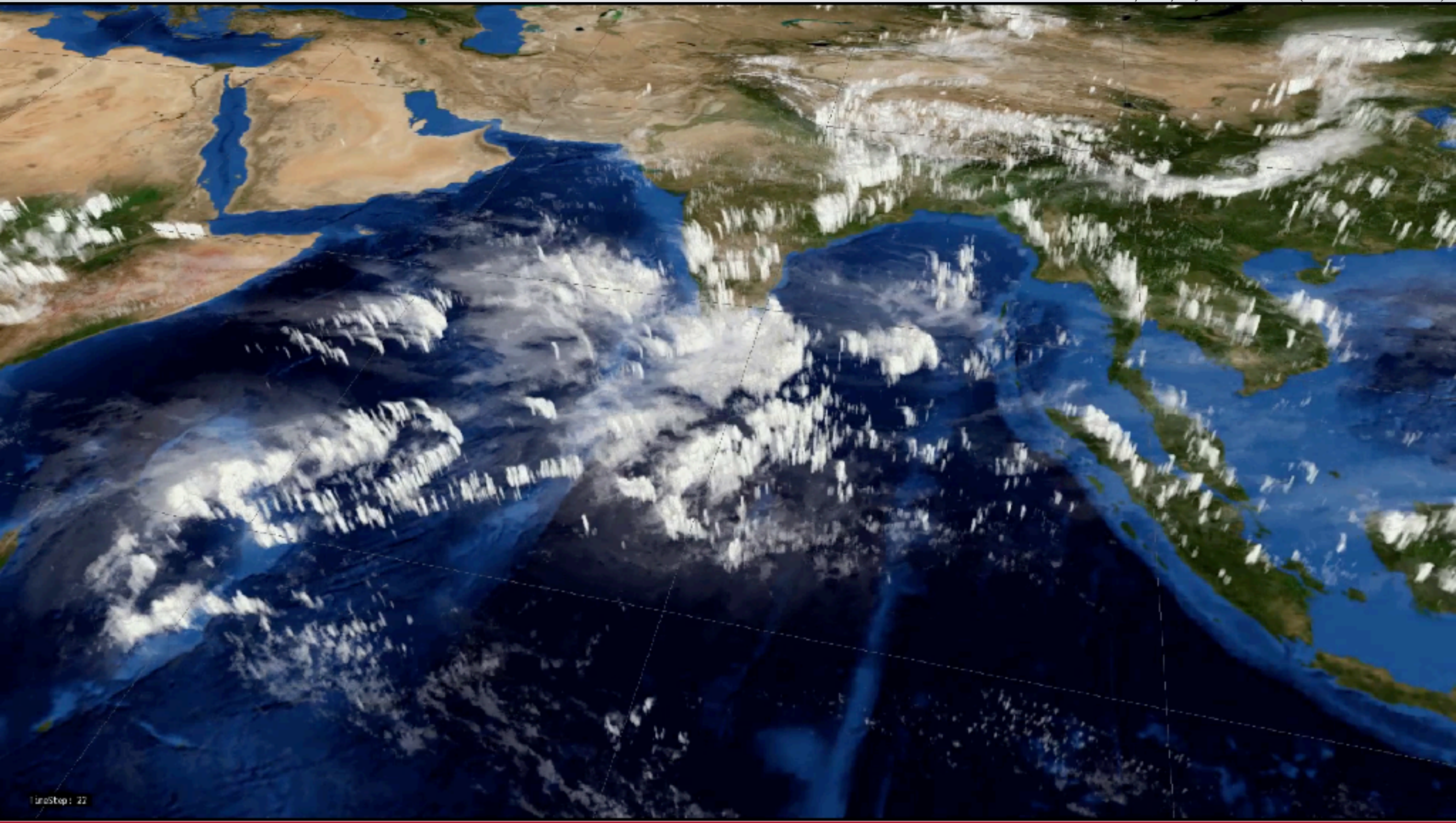
- Development was started since 2000
  Tomita and Satoh (2005), Satoh et al. (2008, 2014)
- First global dx=3.5km run in 2004 using the Earth Simulator
  Tomita et al. (2005), Miura et al. (2007, Science)
- First global dx=0.87km run in 2012 using the K computer
  Miyamoto et al. (2013, 2015), Kajikawa et al. (2016)

- Main target : high-resolution simulation without convection parameterization, without lateral boundary

- Compressive, non-hydrostatic equations are solved using finite volume method on the icosahedral grid
  - Most part is written by Fortran90
  - ~50 users, ~10 active developers

# The first global sub-km atmospheric simulation (Miyamoto et al., 2013)

Movie by Ryuji Yoshida(RIKEN AICS)



# The first global sub-km atmospheric simulation

**10000x larger number of grids than the current climate model !**

## Problem settings

- Δx=870m, 94layers : 63billion grids
- 48hours integration with Δt=2sec : 86,400steps

## Simulation

- 4.5hours for 1hour simulation with 20,480nodes (163,840cores)
- 8TB of checkpoint file for every 3600 steps (2 hours)
- Output variables as "history" for every 900 steps (30 minutes)
  : 320TB in total
- We met job failure only once (of 2hour integration x 24)
  : hardware failure of the storage system

# Experiments on the K computer

## Our simulation didn't have any problems in I/O

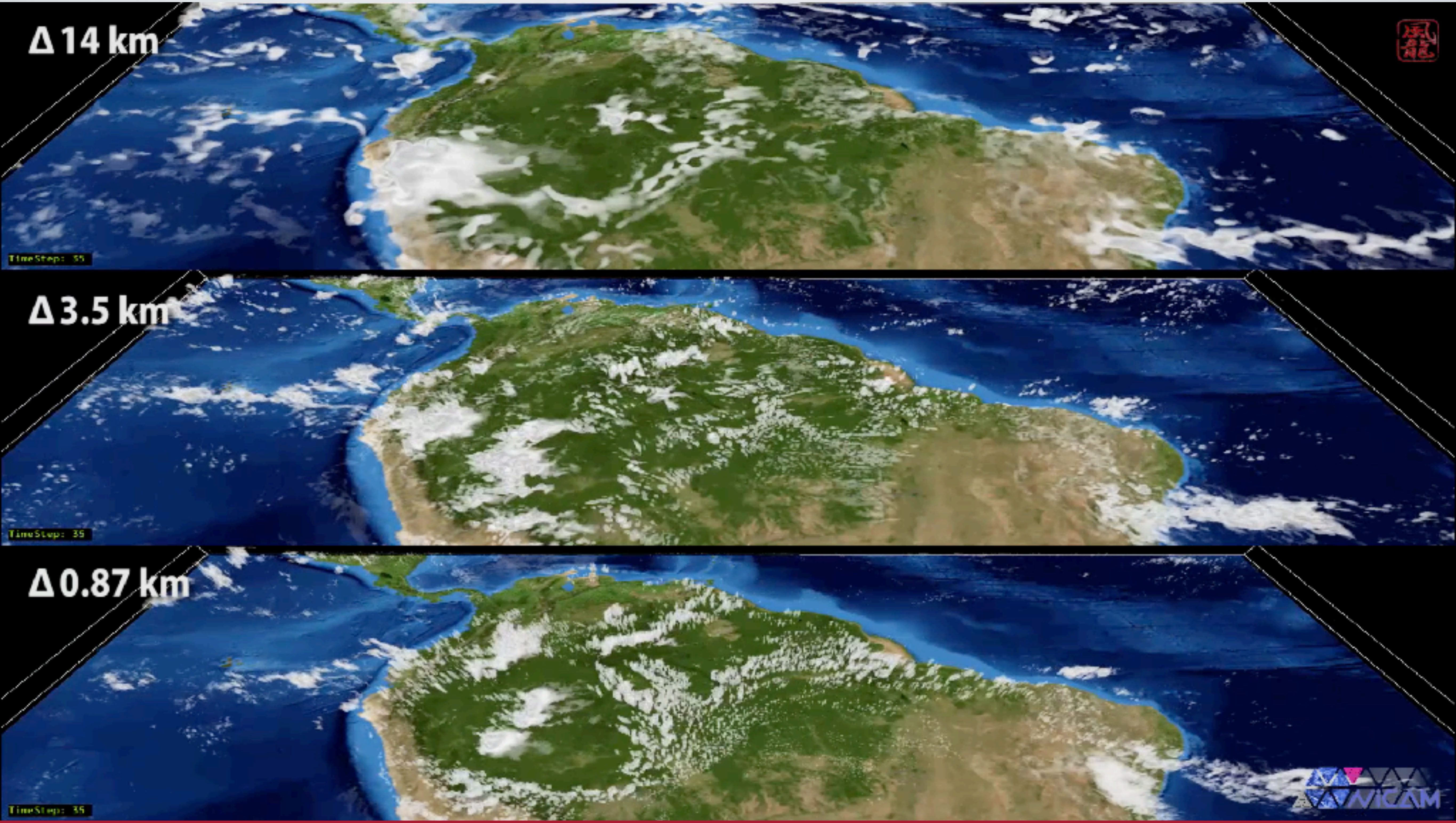
- Excluding one hardware failure during the stage out process
- I/O times were negligibly small

## Why?

- **System side**
  - File staging : isolated from the crowd global FS
    - A different storage disk is assigned to each MPI rank
  - I/O node : we don't have to wait writing due to the large buffer
- **Application side**
  - Distributed file I/O : each MPI rank writes the files
  - Reducing the number of the files per MPI rank
    - file for each variable → one checkpoint file and one history file

# Analysis of precipitation diurnal cycles

by Ryuji Yoshida(AICS,Kobe U.)



# ‘Big’ data analysis in the weather/climate study

Every 30min Snapshot for 0.87km run: ~3TB
x 48 steps (for last 1day output) = 160TB

Grid remapping
from icosahedral
to latitude-longitude

2 months on the post-process cluster

File I/O issue!

Analysis on
latitude-longitude grid

2 months on the post-process cluster

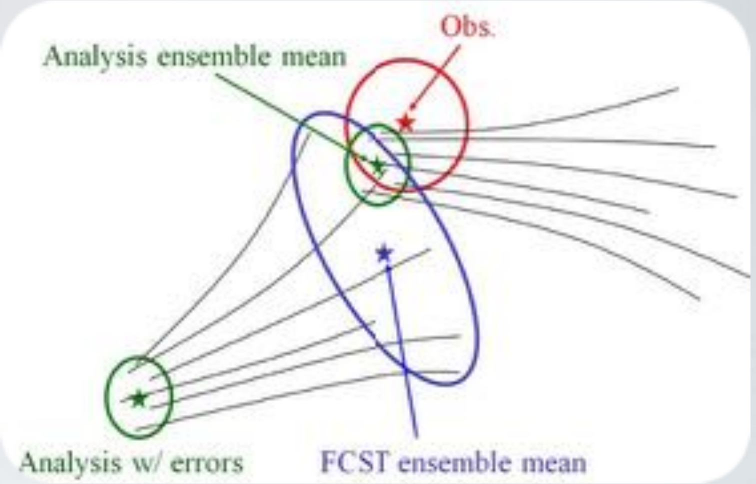
Analysis on
icosahedral grid

1 hour on the K computer

# Ensemble data assimilation

# LETKF

## **L**ocal **E**nsemble **T**ransform **K**alman **F**ilter (LETKF, Hunt et al. 2007)



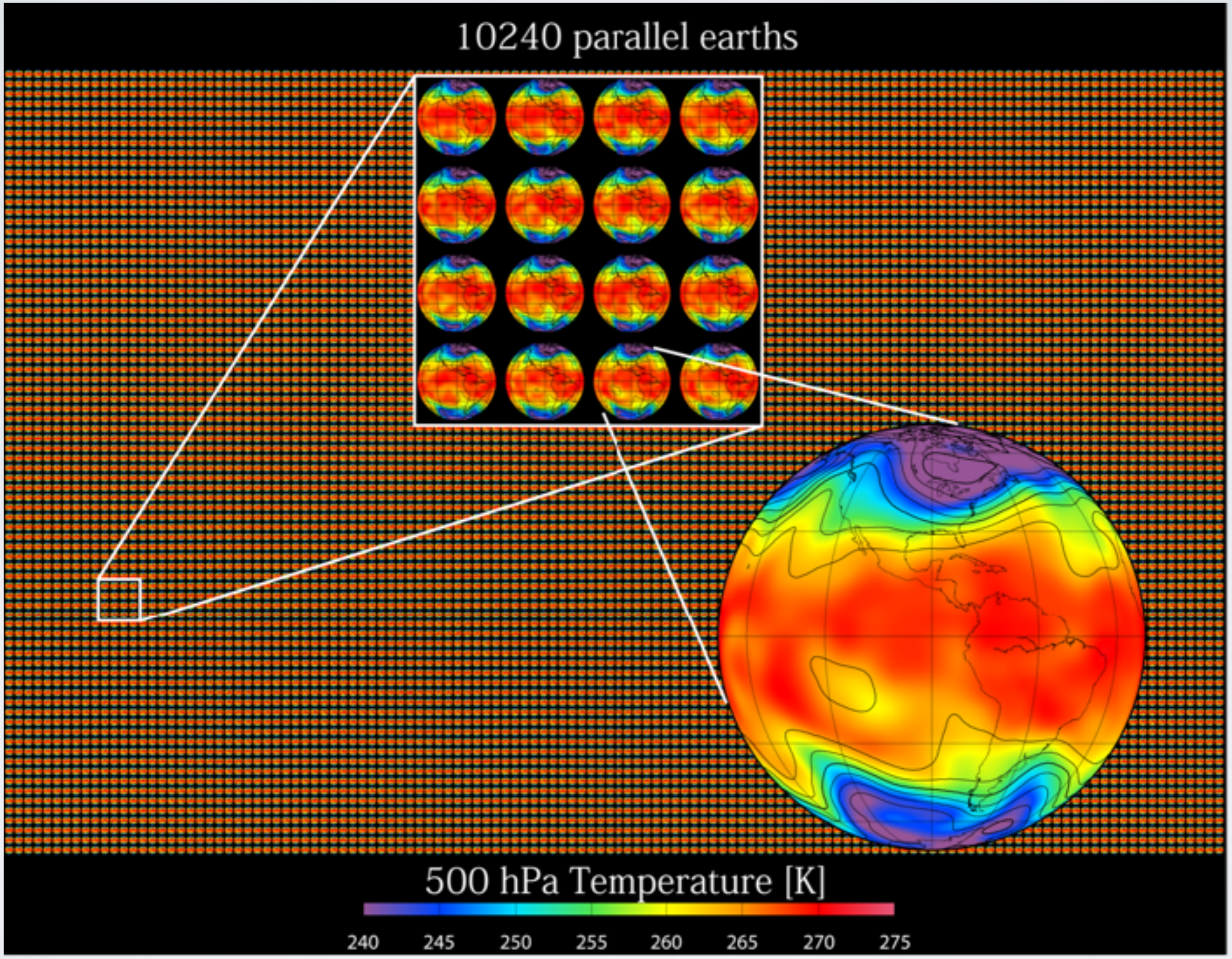

- Ensemble-based data assimilation method
- Applied to NWP models
  (JMA/GSM, WRF-ARW, NCAR/GFS, SPEEDY AGCM, JMA/NHM, AFES , MIROC, etc..)
  Miyoshi and Yamane (2007), Miyoshi and Kunii (2012), etc…

- Library requirement: eigenvalue solver, DGEMM
- Written by Fortran90

- Filtering operation is individually applied to each grid point
  : easy to parallelization
- Adjoint code is not required: easy to apply other model

# Ensemble data assimilation with 10240 members (Kondo et al., 2014)

4608 nodes (36864 cores), 263TFLOPS(46%)



# Data management in the DA system

- **File-based delivery between NICAM and LETKF**
  : Data transpose is required

Model simulation

( **grid group** x PEs ) x **ensemble members**
**Divide horizontally**

Each ensemble member must have all grids on the globe

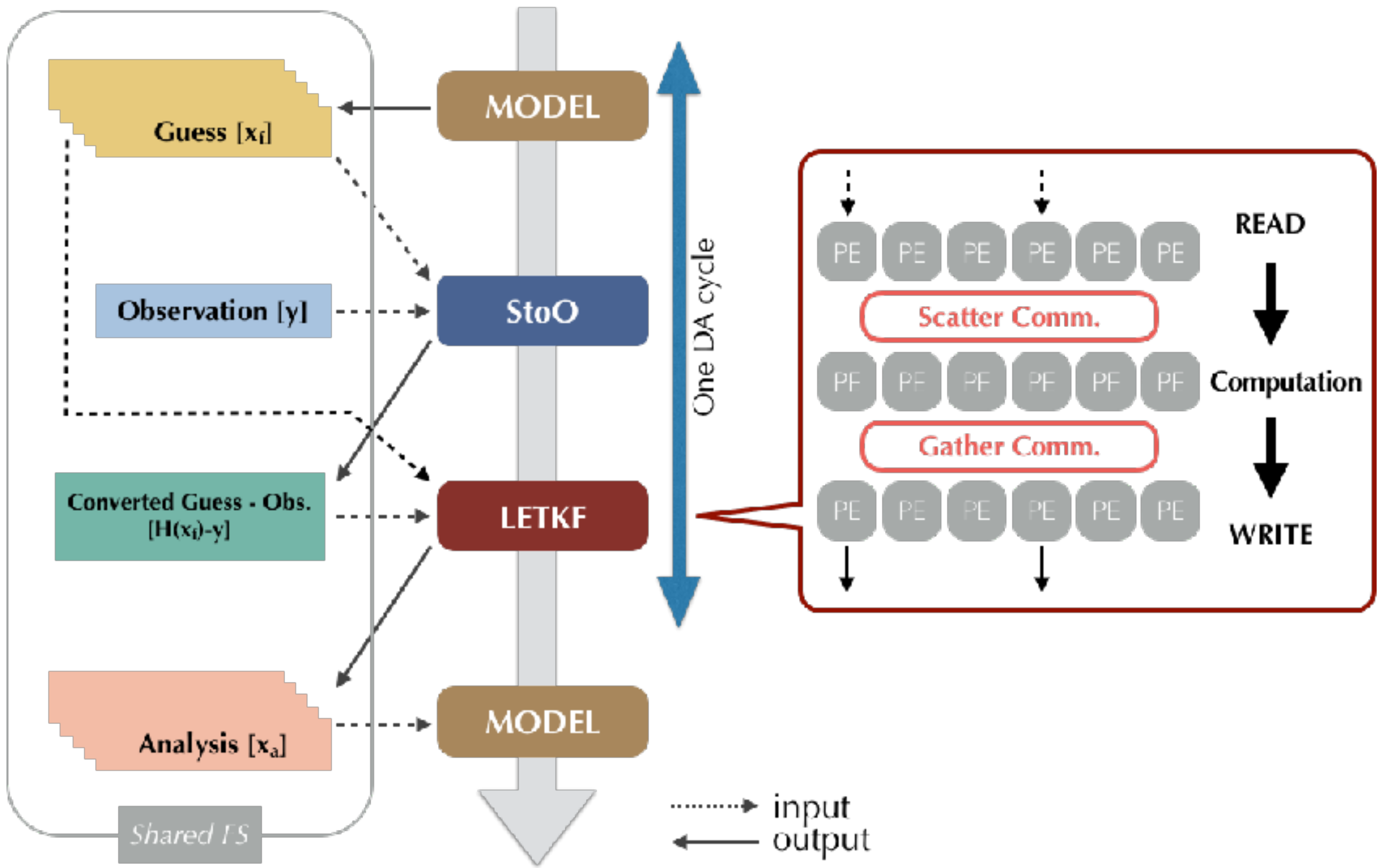
exchange

Data assimilation

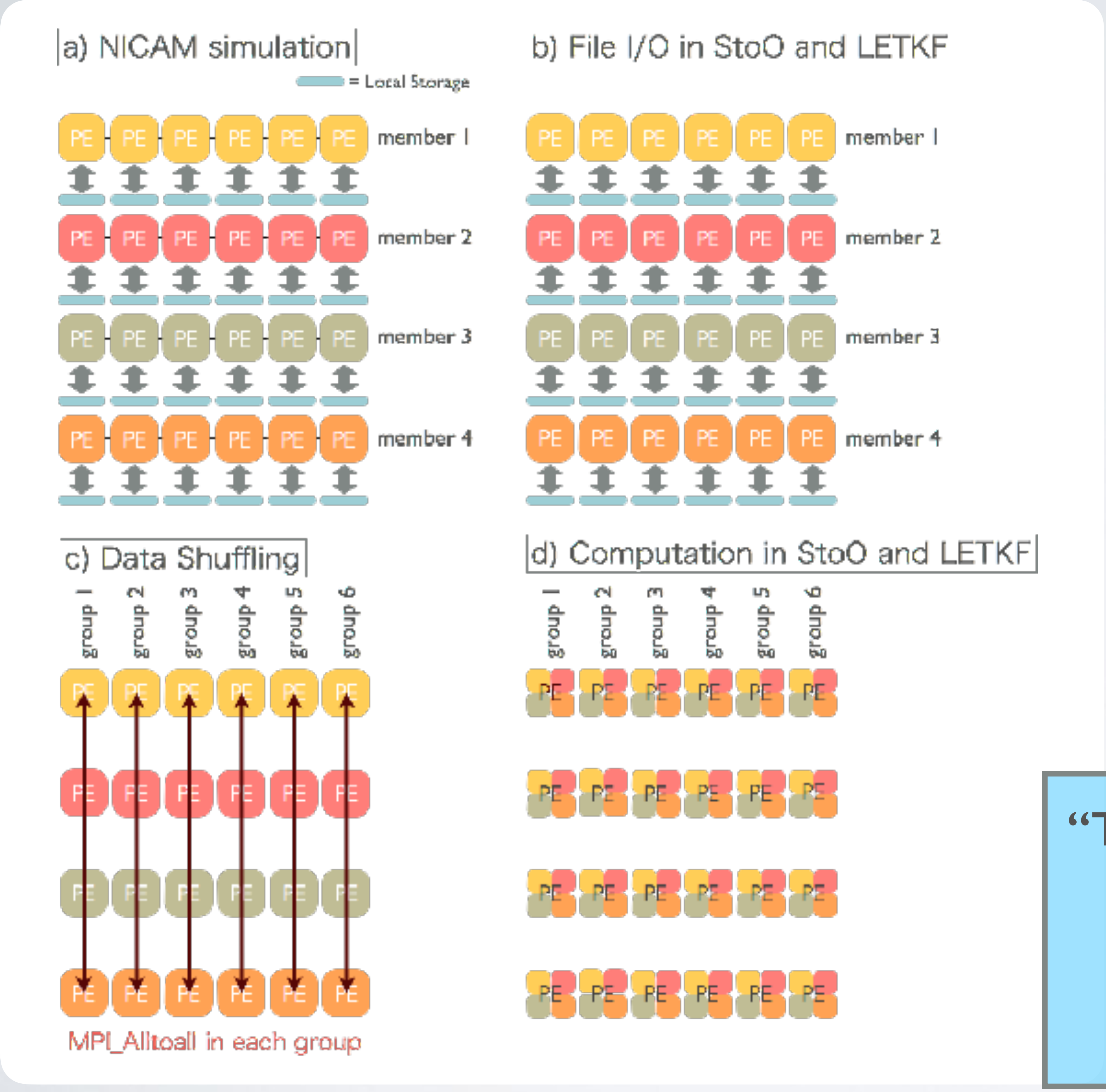
**all ensemble members** x **grid group** x PEs
**Divide horizontally**

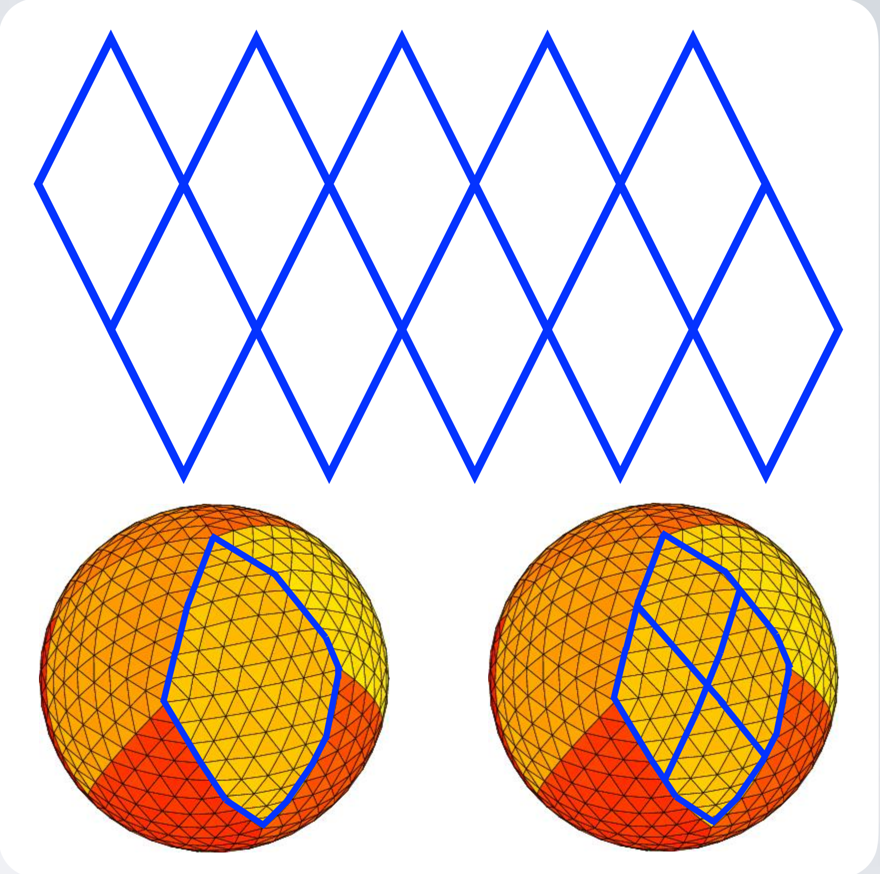
Each process must have same grid of all ensemble members

# Flow of NICAM-LETKF DA system



# New design of the DA system (Yashiro et al., 2016, GMD)

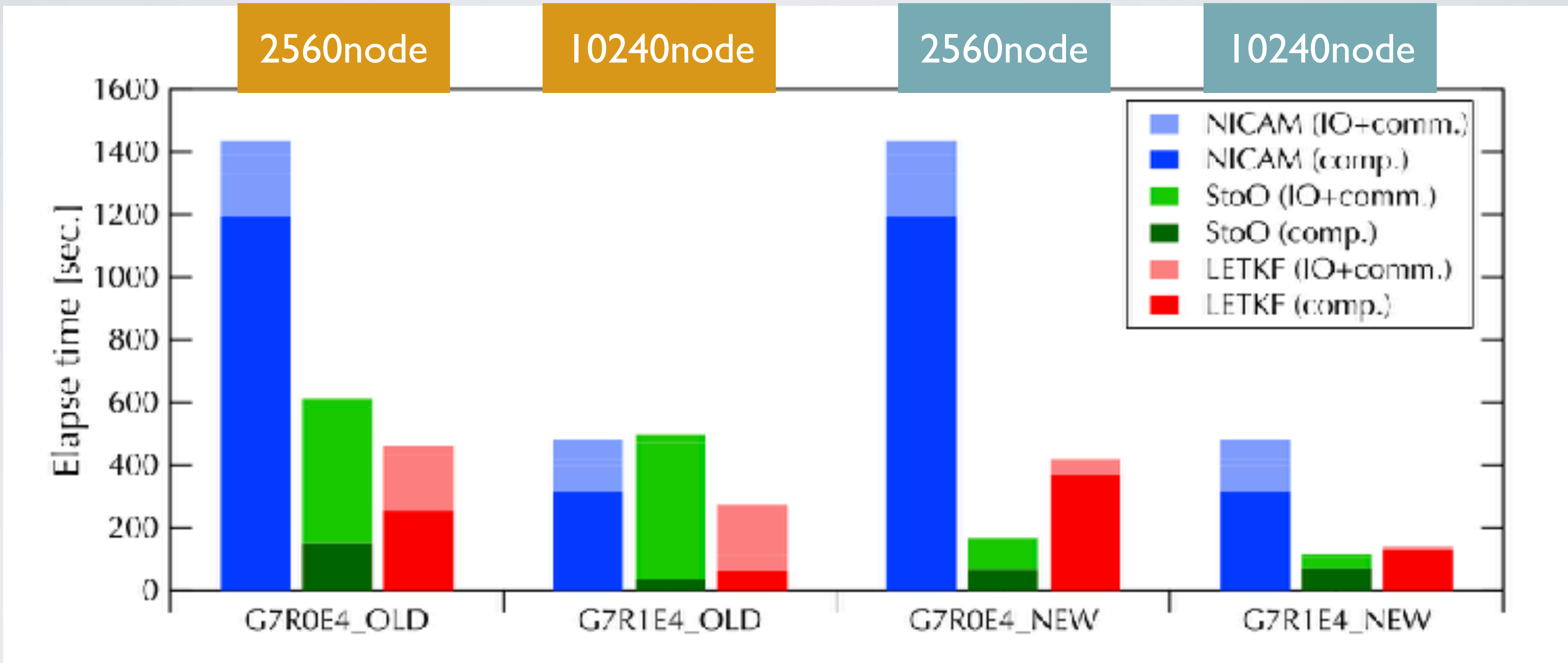


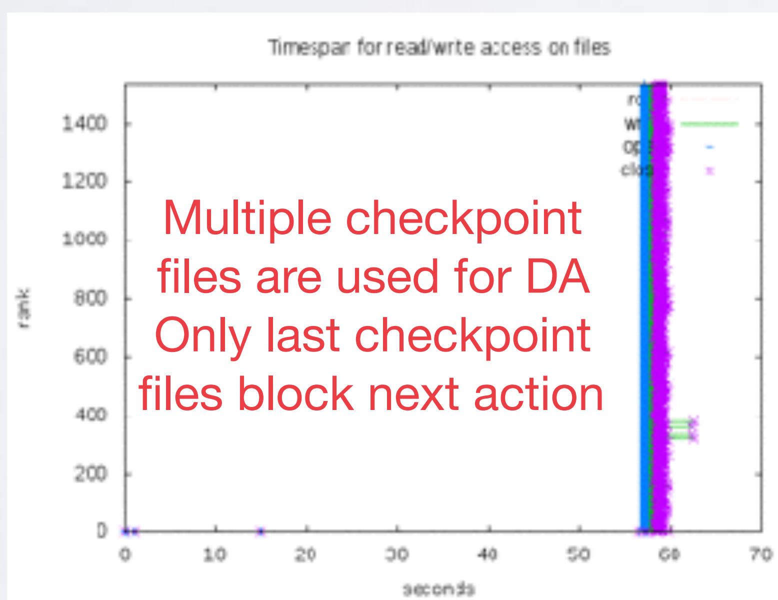
**"Throughput-aware" design**

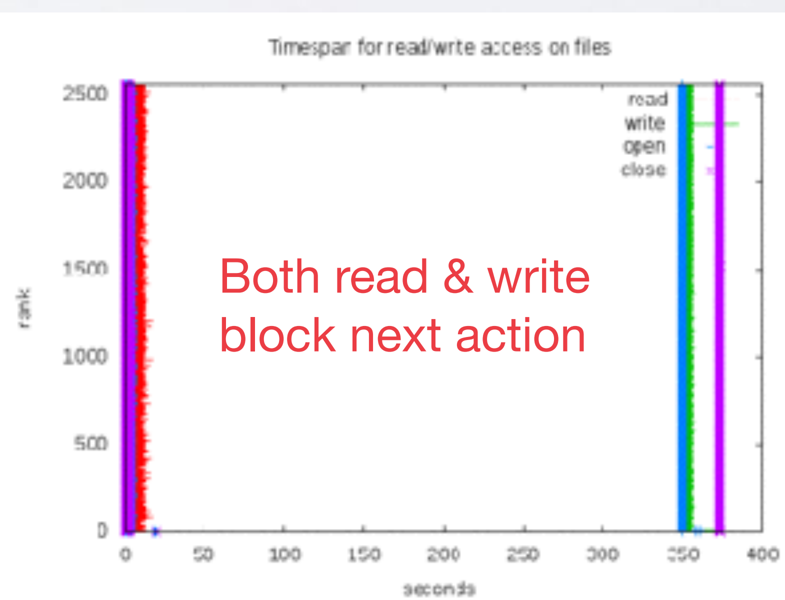
- reduce data movement
- use local storage
- avoid global communication

# New design of the DA system (Yashiro et al., 2016, GMD)




Darshan results for one checkpoint output by NICAM

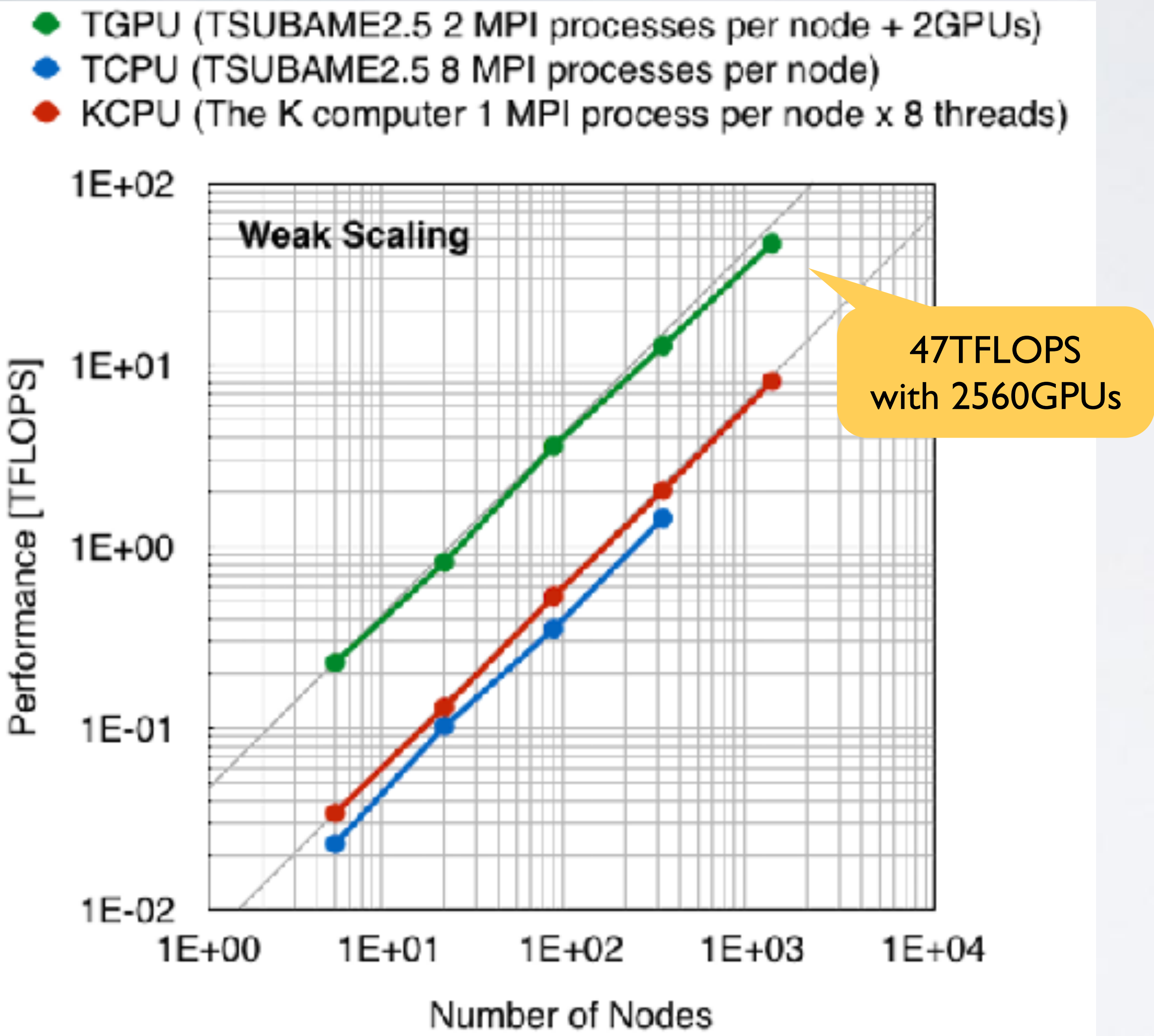



Darshan results for one assimilation by LETKF

# File I/O on the heterogeneous architecture

# Simulation on GPU-based supercomputer (Yashiro et al., 2016, PASC)



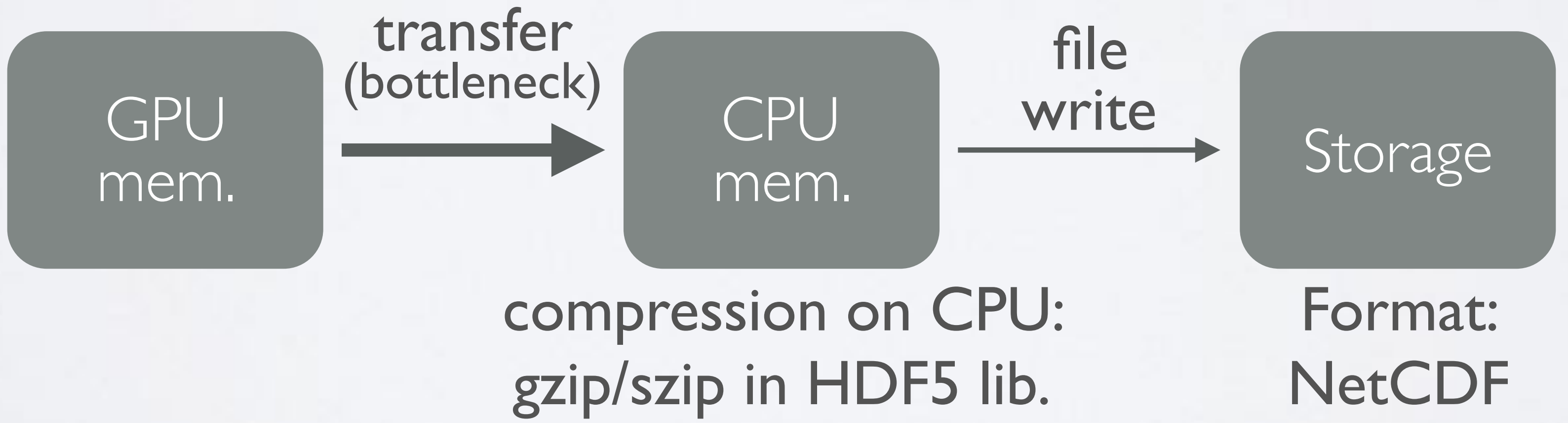

- Dynamical core package of NICAM was used
- Typical test case of dry atmosphere was conducted
- We adopt OpenACC for GPU implementation
- Performance evaluation on TSUBAME 2.5 in 2013
  - Largest GPU supercomputer in Japan
  - We used 2560GPUs (1280nodes x 2 K20x) at maximum

# Effect of file I/O in the GPU-based simulation

- **47TFLOPS in largest problem size**
  - In this case, diagnostic variables were written in every 15 min. of simulation time
  - By selecting the typical output interval (every 3 hours = 720 steps), we achieved **60TFLOPS**

- **File I/O is critical in production run**
  - We can compress output data on GPU

➡ We really need GPU-optimized, popular compression library: **cuHDF?**

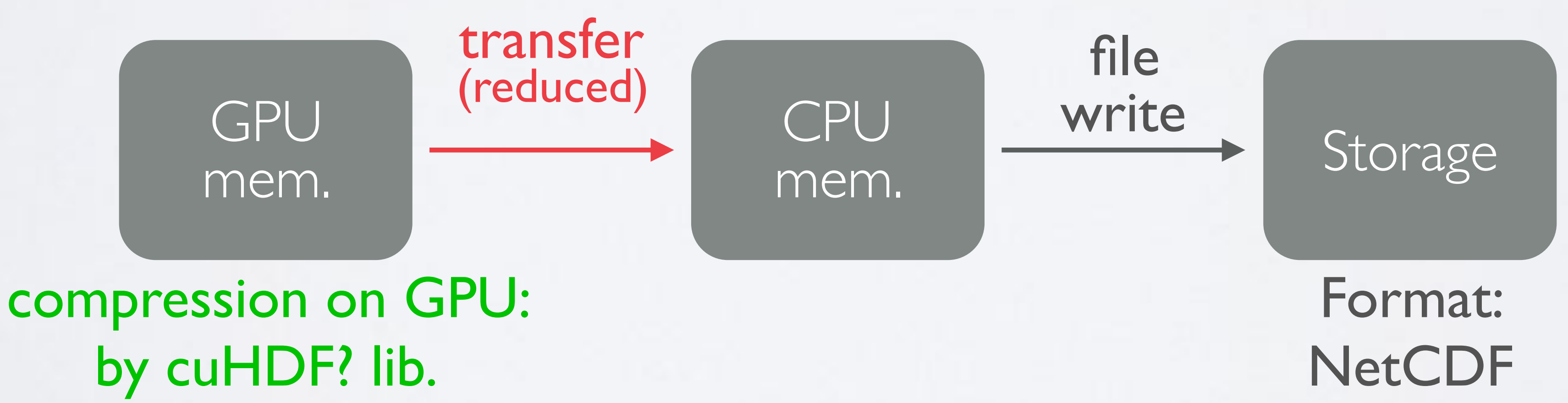


# Effect of file I/O in the GPU-based simulation

- **47TFLOPS in largest problem size**
  - In this case, diagnostic variables were written in every 15 min. of simulation time
  - By selecting the typical output interval (every 3 hours = 720 steps), we achieved **60TFLOPS**

- **File I/O is critical in production run**
  - We can compress output data on GPU

➡ We really need GPU-optimized, popular compression library: **cuHDF?**



# Summary

**Global super-high resolution simulation succeeded to simulate without any I/O interruptions**

- Asynchronous, distributed I/O is important
- Post-process and analysis should be conducted on the supercomputer
- In the future, we may use fast storages as a memory
  : Increase rate of total DDR memory capacity is slow

**I/O optimizations for ensemble data assimilation are challenging**

- Huge, but temporal data transfer is required
  : NVRAM & BurstBuffer will be useful

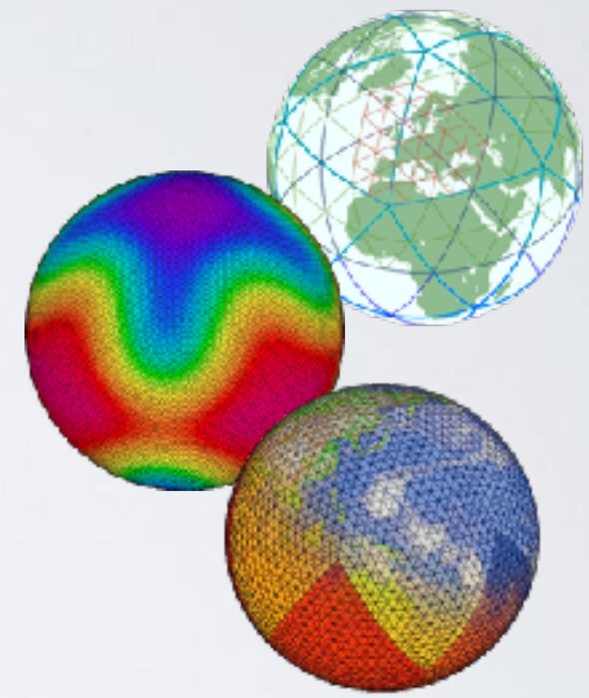
**Data compression is important not only for the storage, but also for the deep hierarchical memory system**

- We want to compress near the processor

# A benchmark suite from weather/climate domain

## AIMES (Advanced Computation and I/O Methods for Earth-System Simulations)

- Tri-lateral collaborative project funding
- Collaboration of icosahedral atmosphere model
  - DKRZ, DWD, U. Humburg (German) : ICON
  - IPSL, LSCE (France) : DYNAMICO
  - RIKEN, Tokyo Tech., U. Tokyo (Japan) : NICAM



## Targets

- DSL benefit for icosahedral atmospheric models
- Massive I/O **Precision-aware data compression library**
- Kernel suites and **mini-apps** from three state-of-art climate models

RIKEN/AICS, the building of the K computer, Kobe, Japan

**Thank you for the attention!**