Computational Challenges in Big Data Assimilation with Extreme-scale Simulations

Takemasa Miyoshi

RIKEN Advanced Institute for Computational Science

Takemasa.Miyoshi@riken.jp

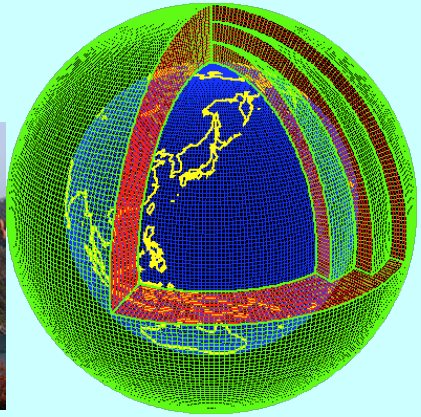
With many thanks to
Y. Sato (JMA),
UMD Weather-Chaos group,
Data Assimilation Research Team

Data Assimilation (DA)

Observations

Data Assimilation

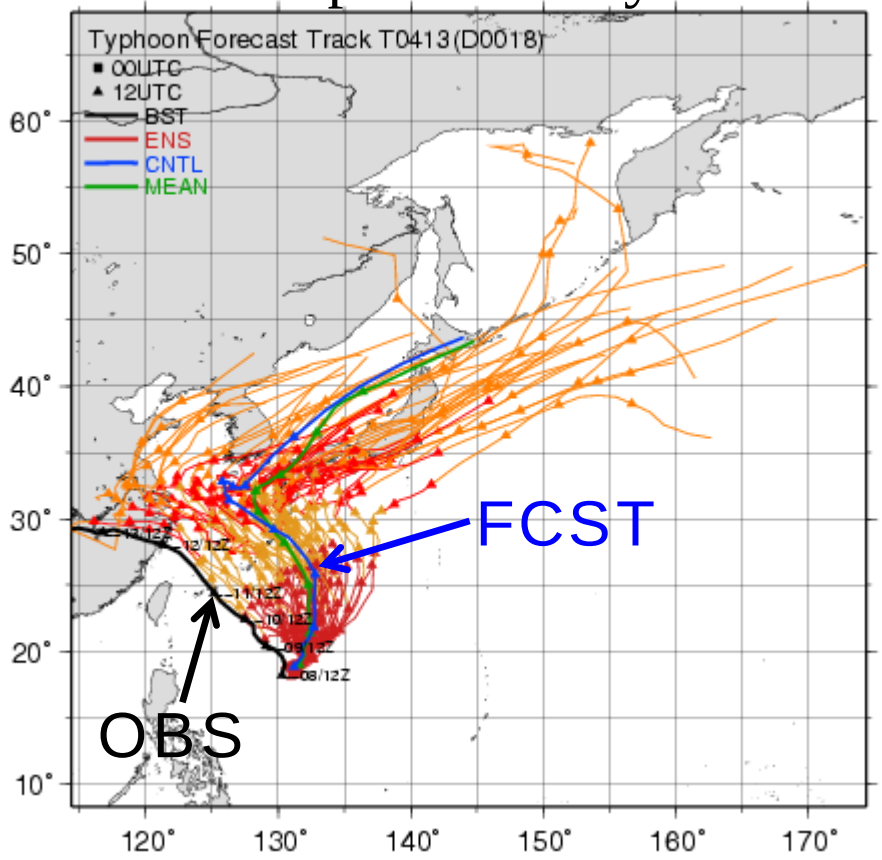
Numerical models



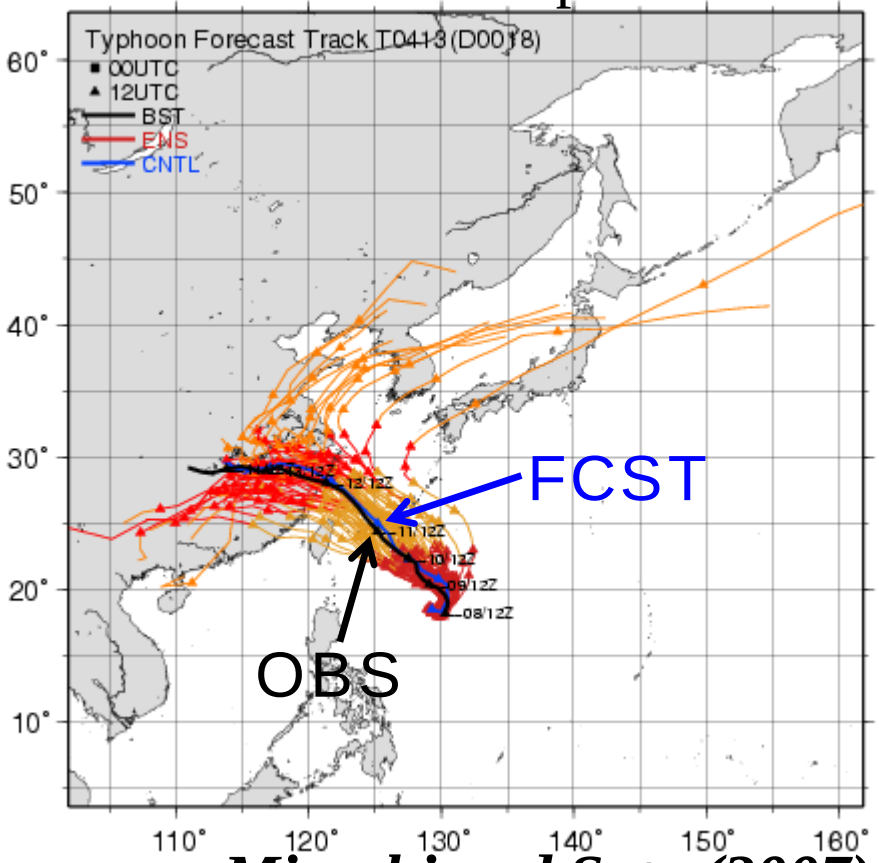
Data assimilation best combines observations and a model, and brings synergy.

DA has an impact.

SV w/ 4D-Var
JMA operational system



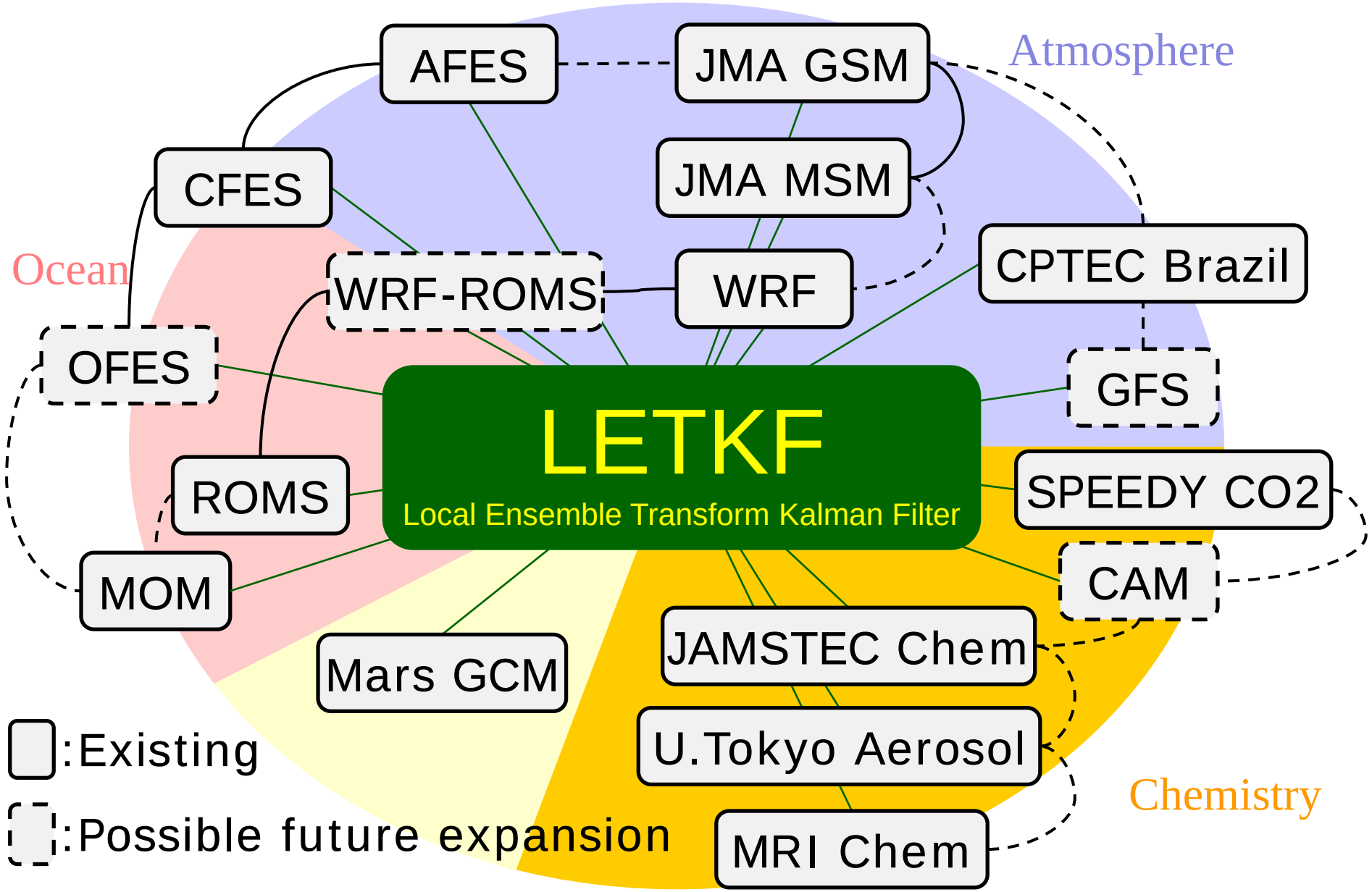
LETKF
under development



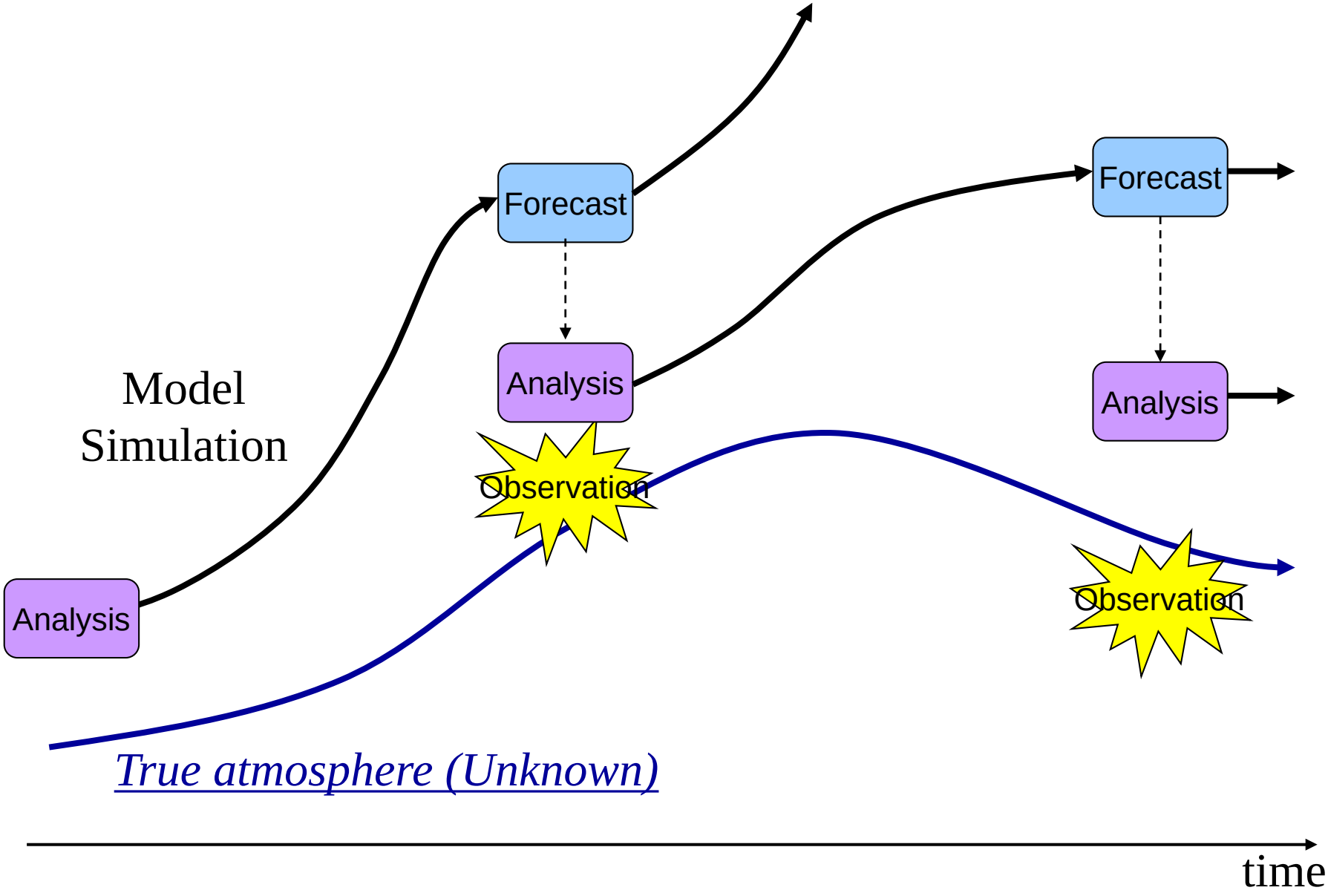
Miyoshi and Sato (2007)

Using the same NWP model and observations.
DA matters!

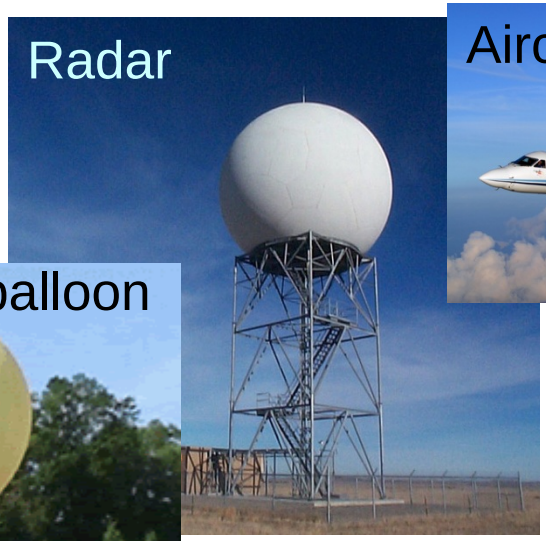
Expanding collaborations



Numerical Weather Prediction (NWP)

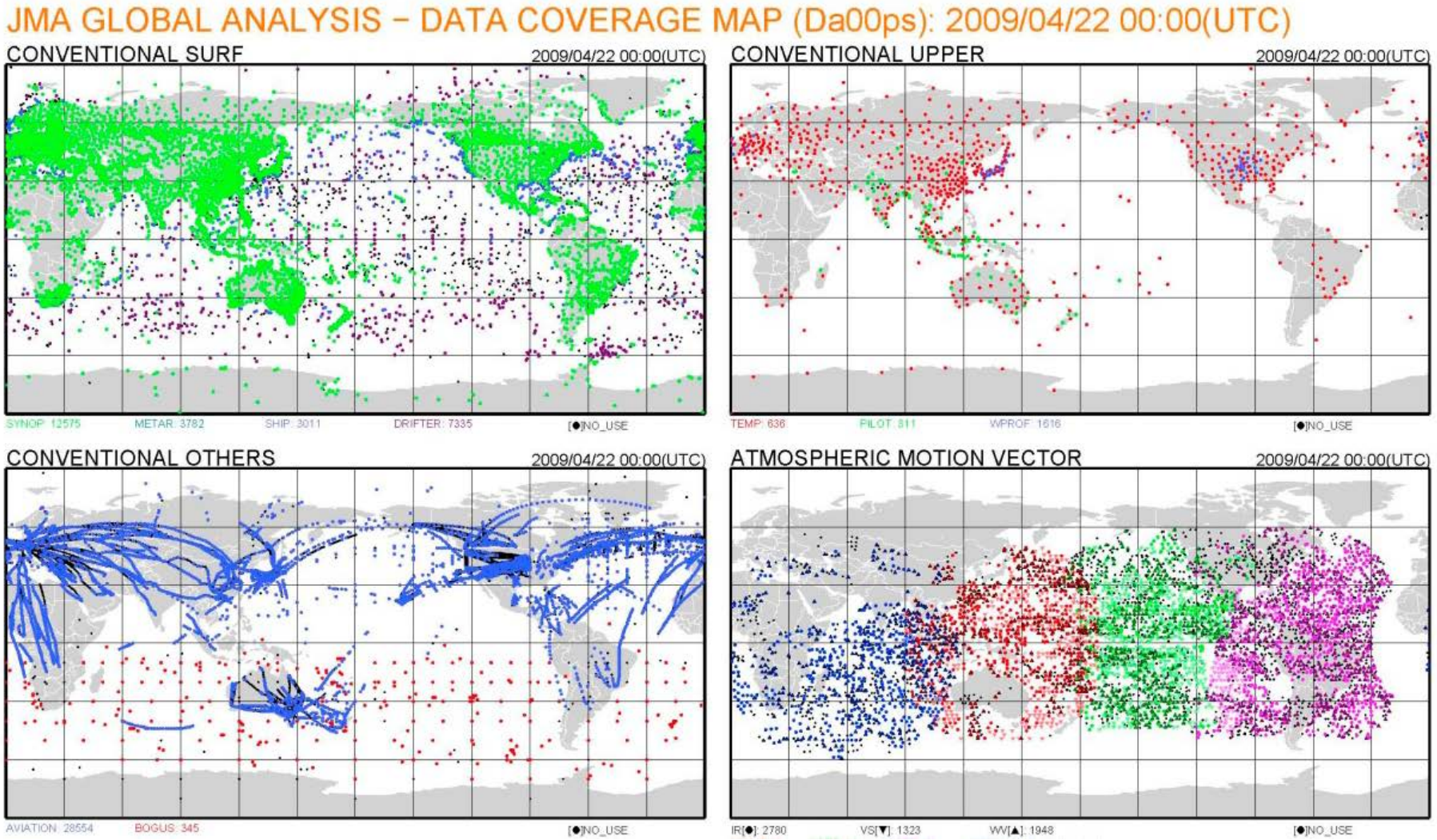


Global Observing System



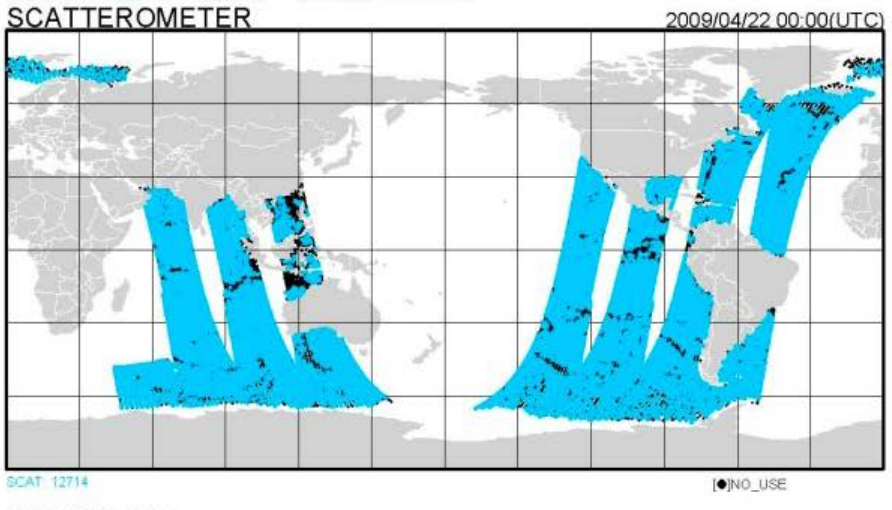
Satellite

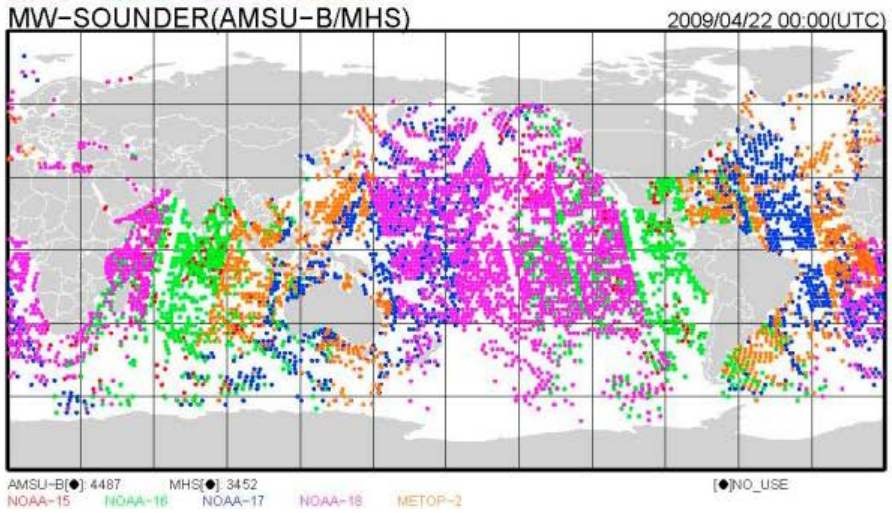
Collecting the data

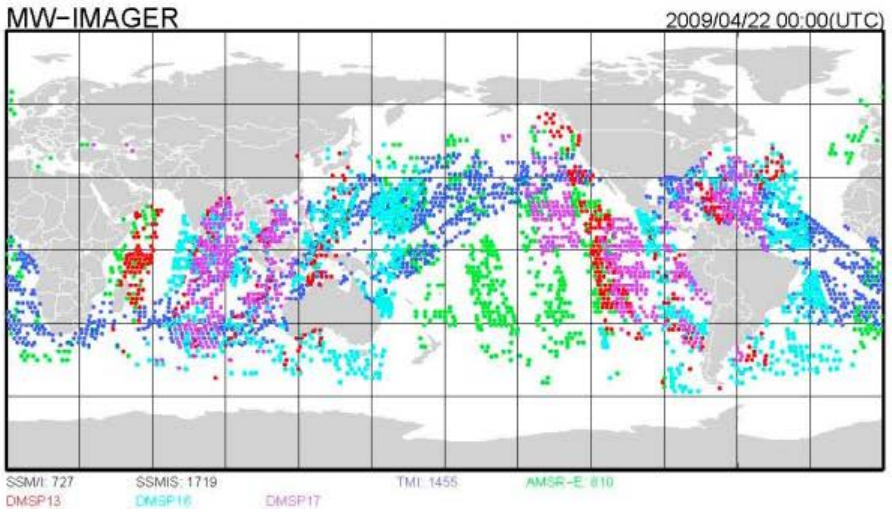


World's effort! (no border in the atmosphere)

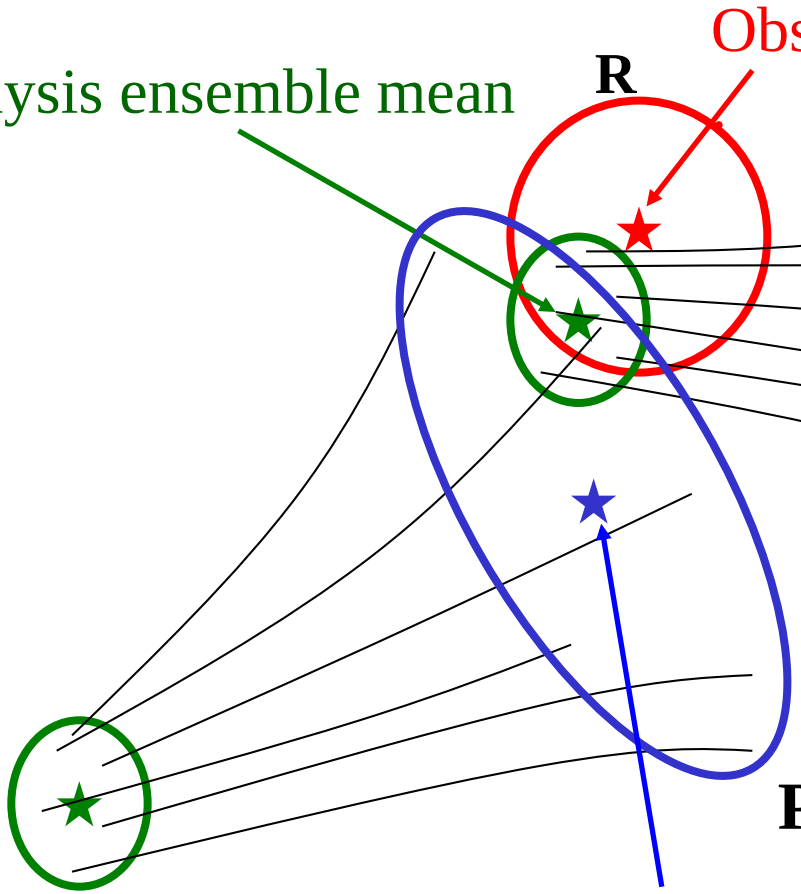
Collecting the data







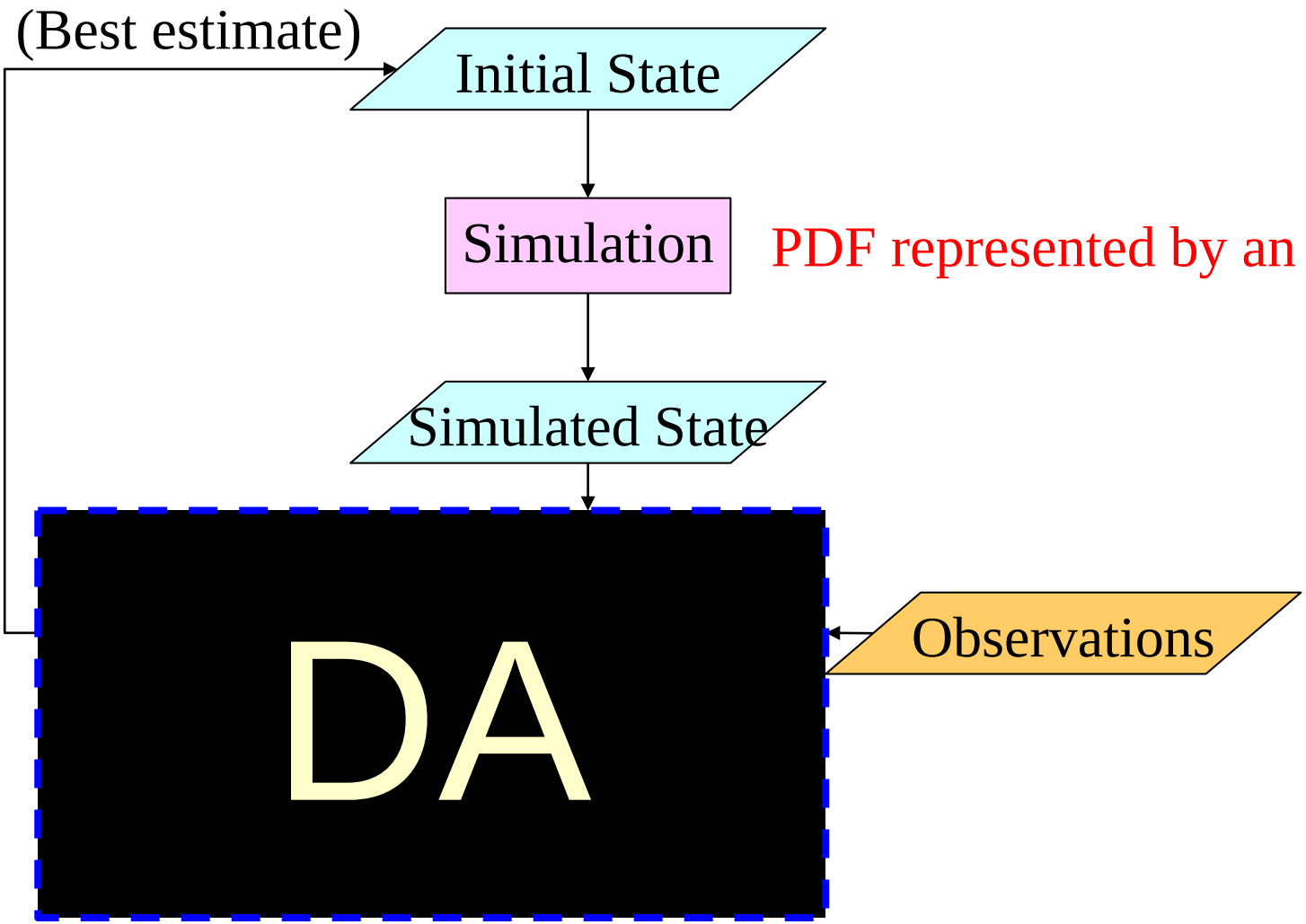
We consider the evolution of PDF



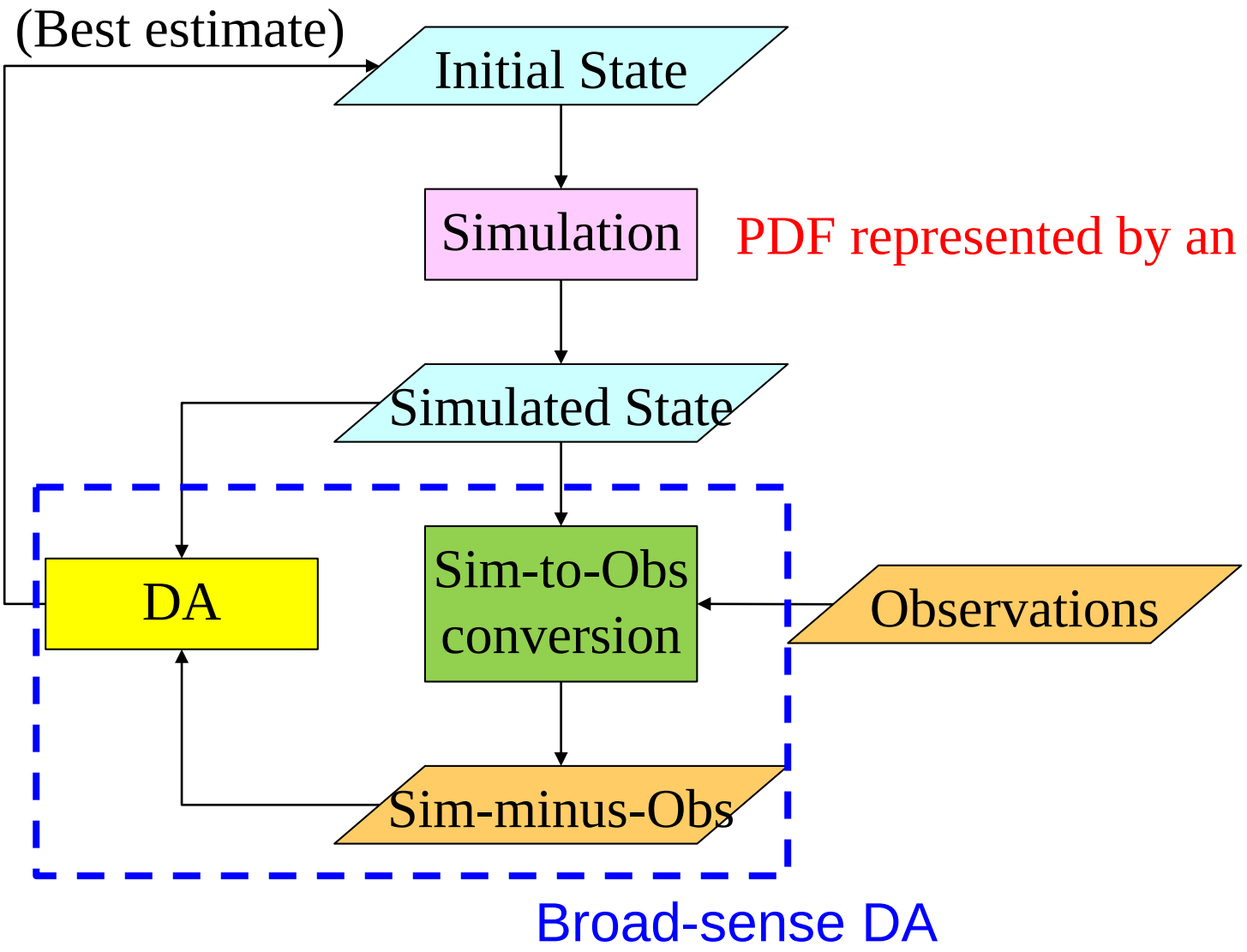
An approximation to KF with ensemble representations

$$\mathbf{P}_{t1}^{f} \approx \frac{\delta\mathbf{X}_{t1}^{f}(\delta\mathbf{X}_{t1}^{f})^{T}}{m-1}$$

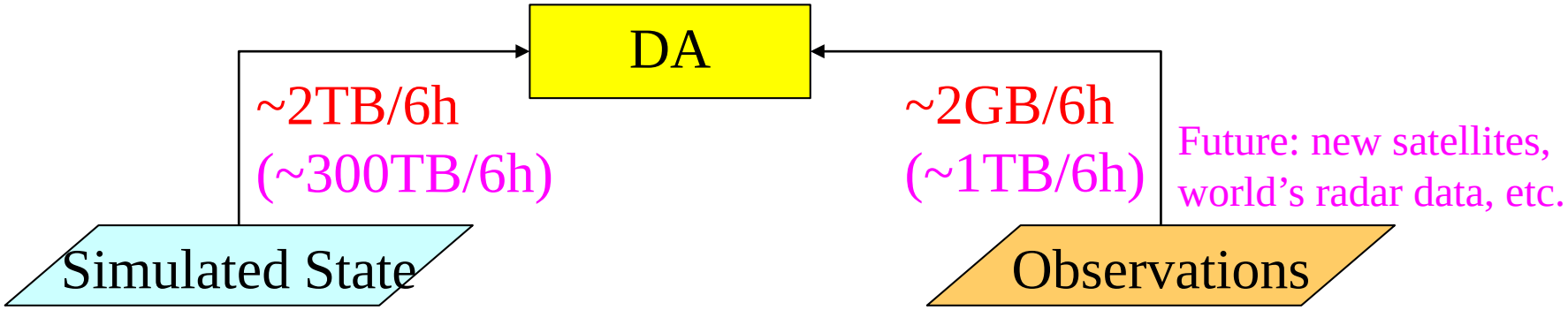
Flow chart of DA



Flow chart of DA

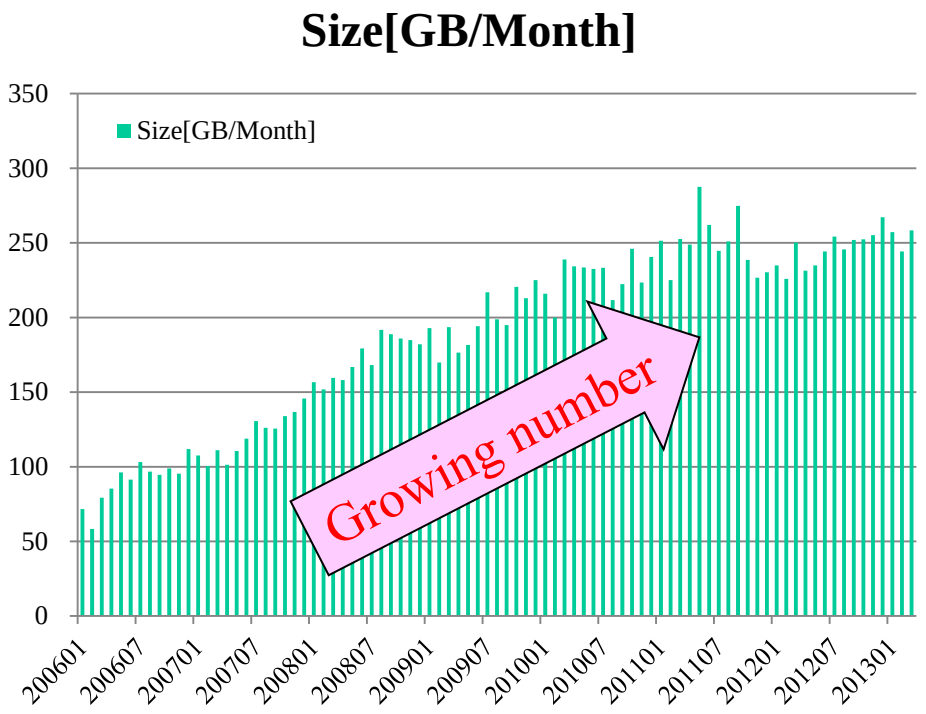


Data size in NWP



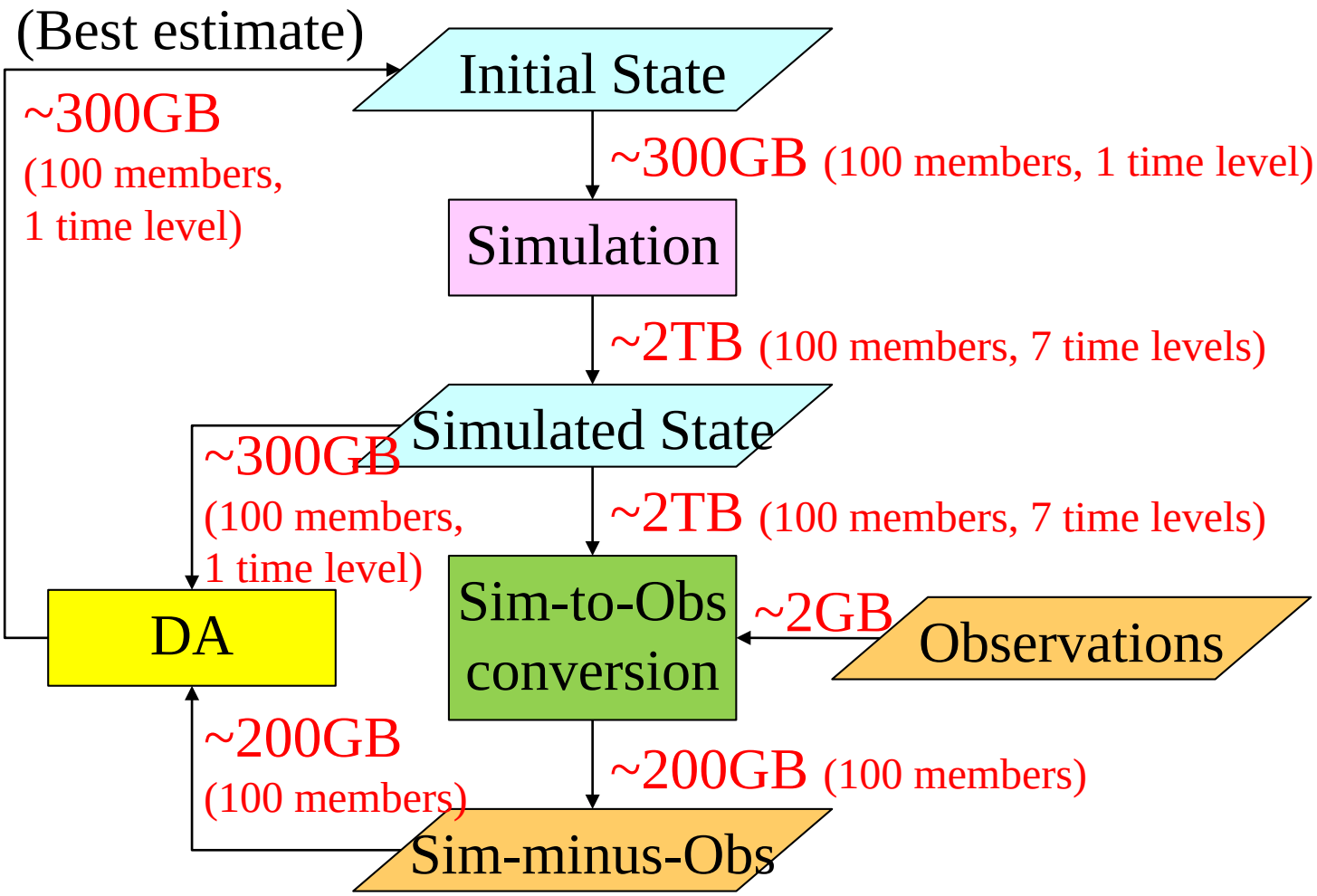
28-km global mesh ~3GB
100 members for PDF ~300GB
7 time slots ~2TB

Extreme-scale Simulation:
3.5-km global mesh ~400GB
100 members ~40TB
7 time slots ~300TB

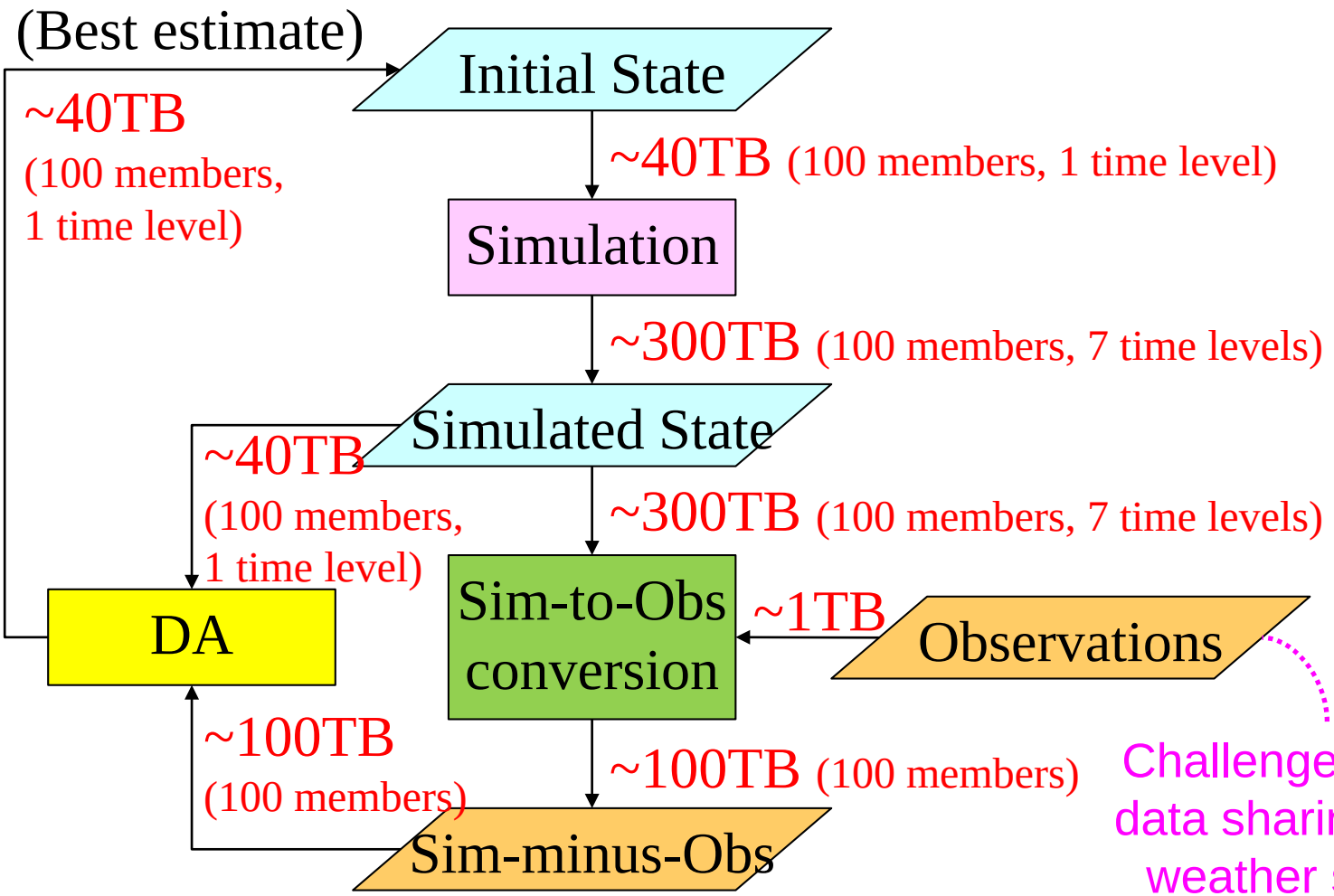


Courtesy of JMA

Flow chart with current data size



Flow chart with exa-scale data size



I/O intensive!

Repetitions of I/O between separate programs

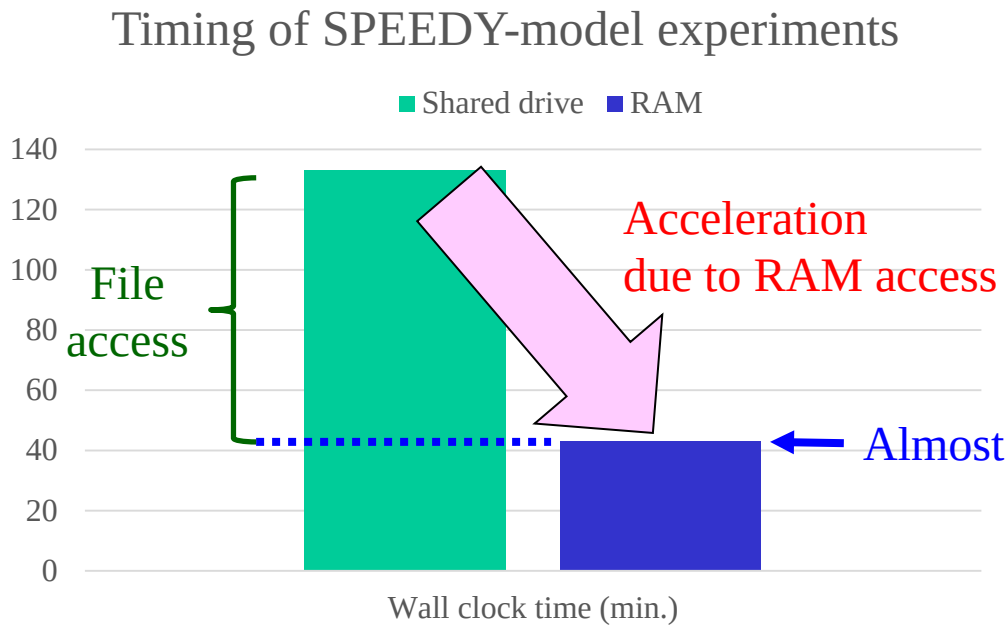
Strategy for fast I/O

Computational challenge:

I/O intensive!
Repetitions of I/O between separate programs

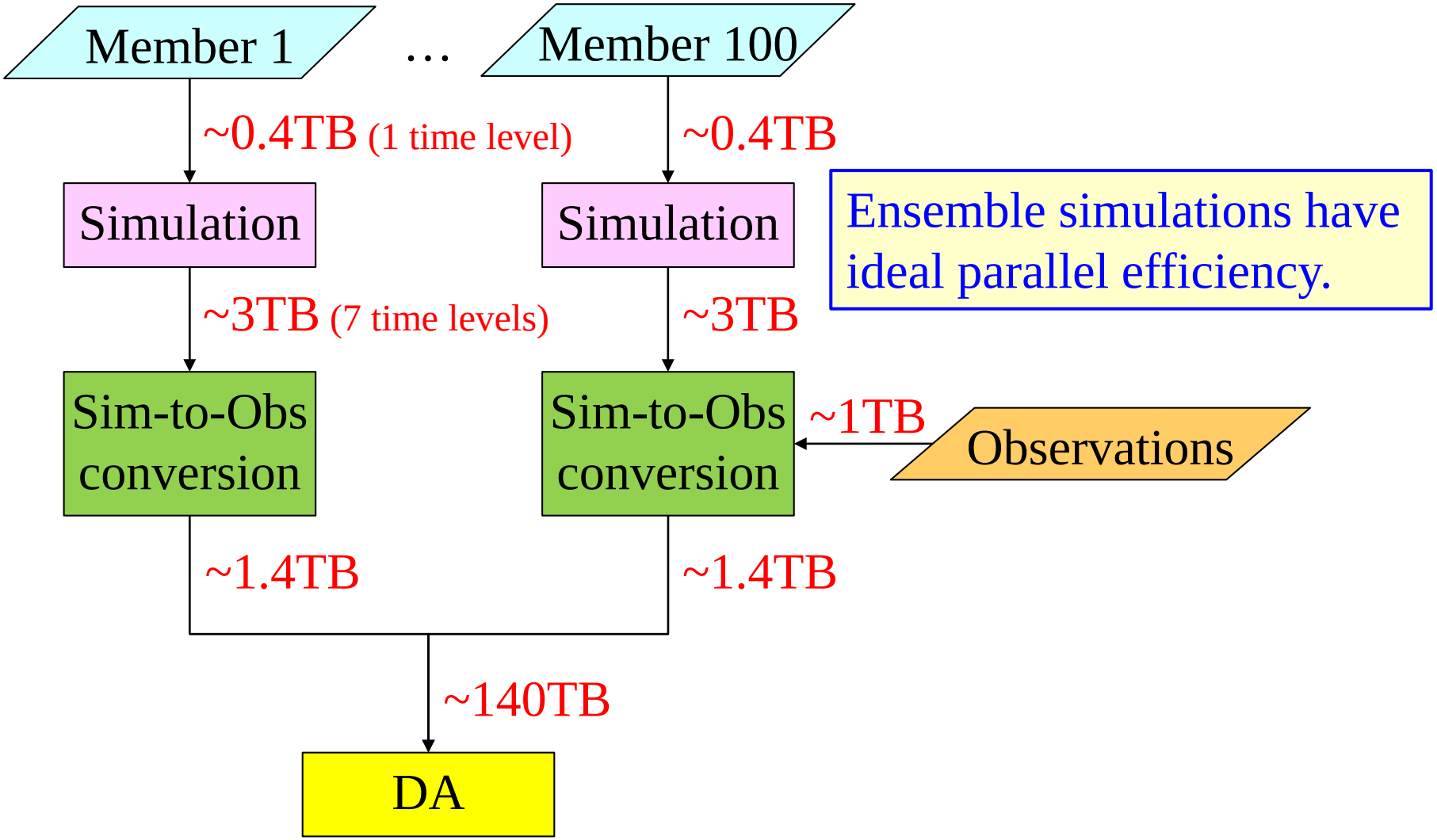
A strategy: It would be ideal to write files to RAM or fast-access memory device (~1PB required)

An experiment:



- Using a Linux cluster (4 nodes, 32 cores)
- 2-month DA cycles
- Experiments with an intermediate atmospheric model (SPEEDY model)

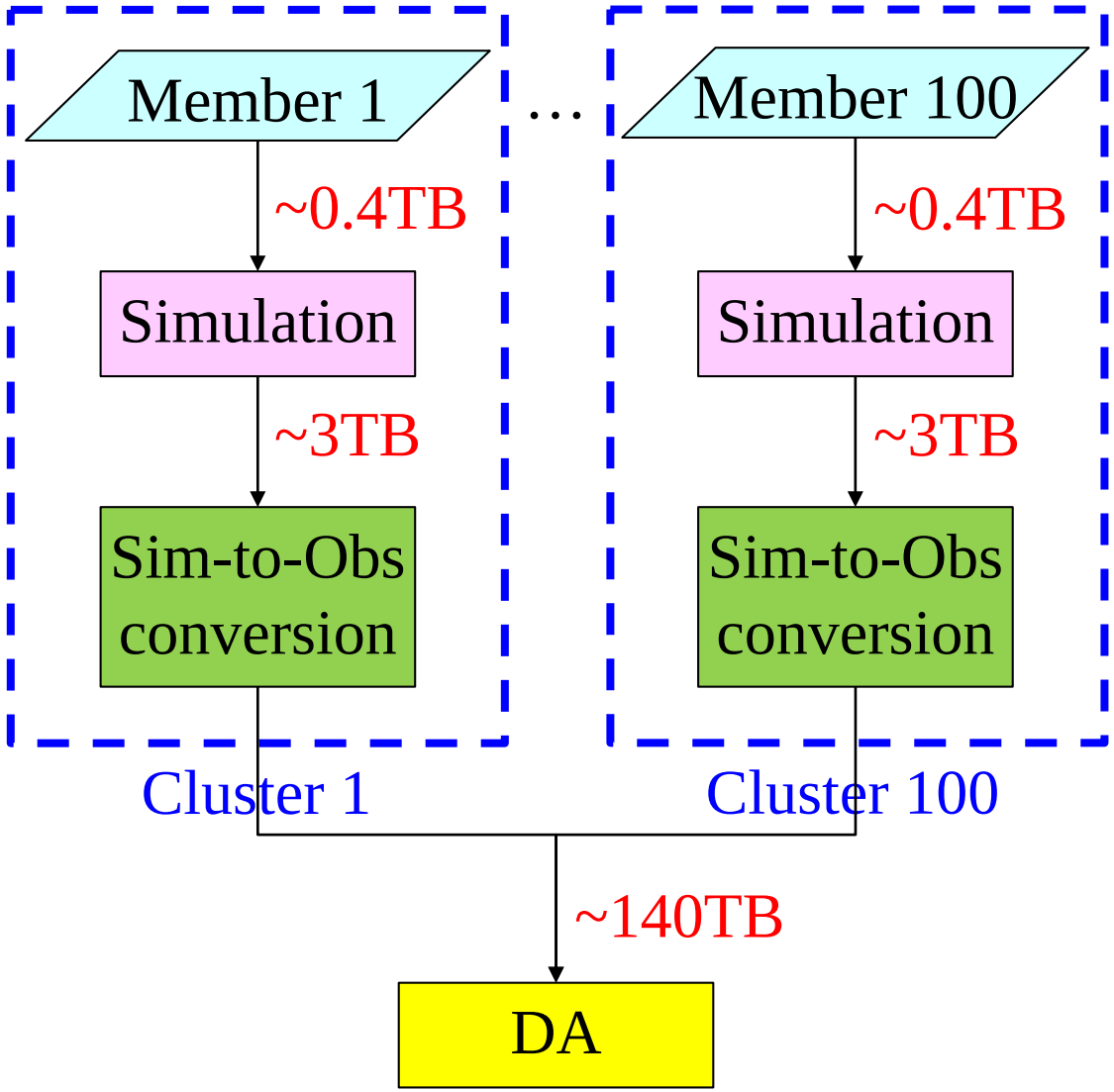
How about parallel processing?



Ensemble simulations have ideal parallel efficiency.

LETKF is parallel efficient, requiring all-to-all comm. only twice.

An efficient architectural design



Fast communication is important within each cluster; slower inter-cluster communication is acceptable.

LETKF requires inter-cluster communications only TWICE.

Other challenges of Big DA

- Transferring Big Data
  - To assimilate “Big Data” into extreme-scale simulations, we need to collect them in an HPC.
  - Can we apply a “cloud” approach?
- Exploring useful data
  - e.g., live camera images may be useful for weather forecasting, but it is hard to collect, qc, and use them…
- Archiving
  - Extreme-scale DA produces at least ~1PB per day.