A Scalable and Efficient Workflow for Compressing High-resolution Earth System Model Data

Haiying Xu, Rich Loft, Kevin Paul, Anderson Banihirwe
National Center for Atmospheric Research

Problem: How to store and analyze the “big-data” output of high-resolution Earth System models

- Global storm resolving Earth system models (GSR-ESMs) at ~4 km resolution will enable scientists to capture important convective processes that lower-resolution models miss.
- A single snapshot, or history file, of the model state of a GSR-ESM is ~170 gigabytes (GBs).
- The model output, particularly for long experiments on exascale systems, can quickly reach unmanageable petabyte (PB) scales.

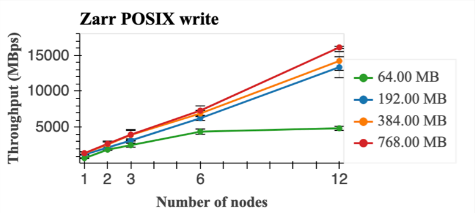
Idea 1: Use data compression to reduce dataset sizes

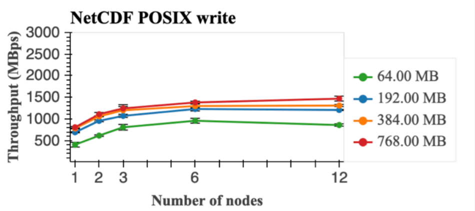
- ZFP is a lossy compressor that has error bounded compression and can best compress spatially correlated data and achieve larger compression ratios
- LZ4 is a faster lossless compressor which accounts for no entropy coding stage, but achieves less compression than ZFP
- We tried both ZFP and LZ4 in the workflow and compared performance

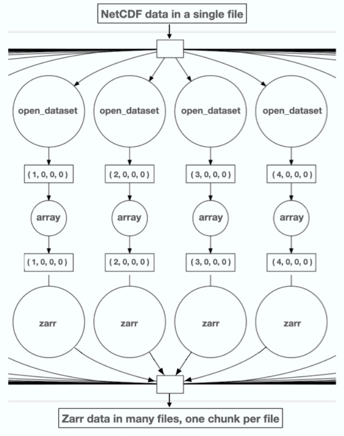
Idea 2: Use the parallelism of Dask, Xarray & Zarr in the workflow

- Dask and Xarray are part of **Pangeo**, which is increasingly used to build climate analysis tools.
- Writing out Zarr files has **much higher throughput** compared writing to traditional NetCDF files
- Zarr can write out data with **compression** encoding such as, Zlib, LZ4 or ZFP
- Zarr reads and writes also work on AWS S3 style **cloud** storage systems.
- Benchmarked this workflow on Intel Xeon processors and GLADE POSIX file system available on the Cheyenne supercomputer at NCAR.

Experiment: Zarr write throughput is 9 times faster than NetCDF



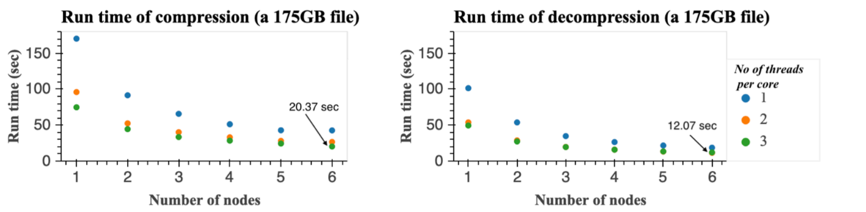




Workflow Chart: Parallel compression on chunk arrays

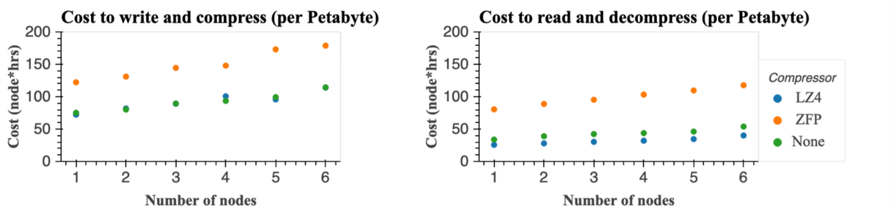
Result 1: Compressed a 175GB file to 43GB using ZFP in 20 sec

It takes 12.07 seconds to decompress the same file on 6 compute nodes.



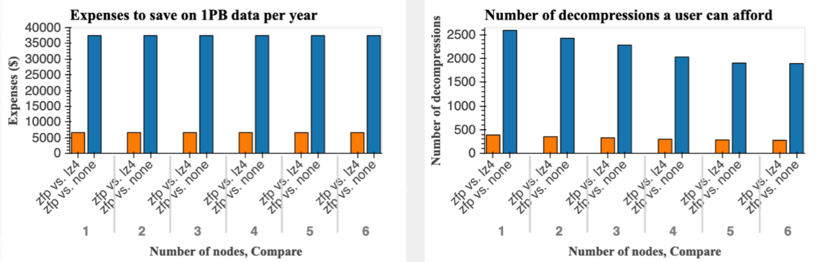
Result 2: Extra cost of compression is 64 node-hrs per Petabyte

Extra cost of decompression is 77 node*hrs on 6 compute nodes.



Result 3: Resource tradeoff: ZFP vs LZ4 and no compression

Lossy ZFP decompressions per year a user can afford until the alternative is better.



Conclusions

- Lossy compression of Zarr chunks using ZFP in a Dask + Xarray workflow is a performant, efficient and scalable strategy that will benefit scientists working with GSR-ESMs.
- We plan to adapt this workflow to the IO library of GSR-ESMs model in the near future

NCAR UCAR CISL Computational & Information Systems Lab
NCAR