I/O Performance with Compression Enabled

Haiying Xu, John Dennis
NCAR/ASAP/IOWA
June 19, 2019
CESM Data Volume Increasing

CMIP5 175TB
CMIP6 230TB (compressed)
Introduction

• CESM need compression with **good IO performance**
  – Enabled parallel compression
  – Enabled asynchronous read/write in PIO
  – Evaluate cloud-friendly data format
  – Compare IO alternatives with IOR benchmark results
  – Conclusion
Impact of Compression on IO Performance

• Compression performance analysis
  – Configuration:
    • Pecount = 288, ompthreads = 3 => 8 io processors
  – IO format:
    • Pnetcdf: parallel NetCDF in 64bit. default in CESM
    • NetCDF4c: NetCDF4 with serial compression
    • NetCDF4p: NetCDF4 with parallel
    • NetCDF4pc: NetCDF4 with parallel compression
    • Async NetCDF4pc: NetCDF4 with parallel compression in asynchronous mode
Enabled Parallel Compression in CESM/PIO

<table>
<thead>
<tr>
<th>Grid</th>
<th>Pnetcdf</th>
<th>Netcdf4c</th>
<th>Netcdf4p</th>
<th>Netcdf4pc</th>
<th>Netcdf4pc</th>
<th>Async</th>
</tr>
</thead>
<tbody>
<tr>
<td>f09_f09 (total/history io)(sec)</td>
<td>560/2.78</td>
<td>577/10.36</td>
<td>600/12.68</td>
<td>609/23.98</td>
<td>590.7/13.09</td>
<td>584.7/14.78</td>
</tr>
<tr>
<td>f09_f09 (io percent)</td>
<td>0.5%</td>
<td>1.8%</td>
<td>2.1%</td>
<td>3.9%</td>
<td>2.2%</td>
<td>2.5%</td>
</tr>
<tr>
<td>Total increased</td>
<td>3.0%</td>
<td>7.1%</td>
<td>8.8%</td>
<td></td>
<td>5.4%</td>
<td>4.3%</td>
</tr>
</tbody>
</table>
### Enabled Async Mode in CESM/PIO

<table>
<thead>
<tr>
<th>Grid</th>
<th>Pnetcdf</th>
<th>Netcdf4c</th>
<th>Netcdf4p</th>
<th>Netcdf4pc</th>
<th>Netcdf4pc</th>
<th>Async</th>
</tr>
</thead>
<tbody>
<tr>
<td>f09_f09 (total/history io)(sec)</td>
<td>560/2.78</td>
<td>577/10.36</td>
<td>600/12.68</td>
<td>609/23.98</td>
<td>590.7/13.09</td>
<td>584.7/14.78</td>
</tr>
<tr>
<td>f09_f09 (io percent)</td>
<td>0.5%</td>
<td>1.8%</td>
<td>2.1%</td>
<td>3.9%</td>
<td>2.2%</td>
<td>2.5%</td>
</tr>
<tr>
<td>Total increased</td>
<td>3.0%</td>
<td>7.1%</td>
<td>8.8%</td>
<td>5.4%</td>
<td></td>
<td>4.3%</td>
</tr>
</tbody>
</table>
Evaluate Cloud -Friendly Data Format

• Zarr/Z5
  – Provides Python/C++ classes and functions for N-dimensional array
  – Writes out array data in chunks and each chunk is compressed
  – Convert to/from NetCDF files easily
  – Uses distributed storage systems: S3Map, HDFSMap, GCSMap

• Working on a SIParCS project with Weile to enable Z5 backend in PIO
IOR Benchmark

- IOR: Interleaved or Random
  - File system benchmarking tool
  - Well-suited for evaluating the performance of parallel system
  - Have several IO backend: POSIX, MPIIO, NCMPI, HDF5, S3

- Purpose
  - IOR has so many IO backend
  - Can compare performance among different IO backends

- Prepare for IOR benchmarking
  - Added c wrapper of Z5
  - Integrated in IOR benchmark tool
IOR Benchmark
Write 16 Segments of 256KB Blocks

- One processor per node
MPIIO Collective Write On Multiple Nodes

- Write throughput decreased to one tenth
- One processor per node

![Graph showing throughput comparison between different methods for varying number of nodes.]
Conclusion on IO with Compression

- Write out from one node instead of multiple nodes
- Evaluate Z5 backend
  - Flexible chunking, compression and metadata management
  - Both pthread and MPI
  - Better with cloud storage
- Enable async mode with large buffers

- Potentially get a compression IO with good performance
Question?

- Contact me at haiyingx@ucar.edu
IOR Benchmark Scaling Study

Scaling study by multiple I/O Libraries

- Z5
- HDF5
- NetCDF

Throughput (MB/s)

- 64MB
- 128MB
- 256MB
- 512MB
- 1024MB
IOR Benchmark Aggregate Study on 1 Node

Aggregate study on one node by multiple I/O Libraries

Throughput (MB/s)

- Z5
- HDF5
- NetCDF

Throughput measurements at different data points:
- 1 point: Low throughput
- 2 points: Moderate throughput
- 4 points: Significant increase
- 8 points: Further significant increase
- 16 points: Highest throughput
XIOS Server

• XML-IO-SERVER
  – Two ways: attached mode, and server mode
  – Accept one-to-one file mode and multiple-to-one file mode
  – Use double buffers on client side, and circular buffer on server side
  – Very good performance
    • 1.5% IO for daily mean output (4322x2882x31, 8160 cores, 32 XIOS)
    • 5% IO for 6 hours mean output
    • 15%-20% for hourly mean output (128 XIOS)
  – No API for inquiring dimension, variable id, etc.
  – Need to make sure all communications overlapped by computation
  – All writing overlapped by computation
  – Otherwise, blocking time will take longer
IOR Benchmark Conclusion

- Z5, NetCDF, HDF5 with compression
  - same performance when write once from multiple processors
  - Z5 and NetCDF need more memories
  - Z5 and HDF5 are better when write aggregately in one node
  - Z5 is five times better when write aggregately from multiple nodes
- IO processors of CESM/PIO now is distributed on processors on multiple nodes
- Z5 has the advantages on chunking, compression and metadata