IO Performance of a Climate Modeling Application

Mark Howison
Lawrence Berkeley National Lab

March 12, 2009
Acknowledgements

• NERSC / LBNL
  ◦ Prabhat
  ◦ John Shalf
  ◦ Katie Antypas
  ◦ Noel Keen
  ◦ Tony Drummand
  ◦ Andrew Uselton
  ◦ Shane Canon
  ◦ Hongzhang Shan
  ◦ Michael Wehner
  ◦ Wes Bethel
  ◦ Janet Jacobsen

• Colorado State University
  ◦ Dave Randall
  ◦ Ross Heikes

• Pacific Northwest National Lab
  ◦ Karen Schuchardt
  ◦ Bruce Palmer
  ◦ Annette Koontz

• Cray
  ◦ David Knaak
Projects

- Design and Testing of a Global Cloud Resolving Model (GCRM) (Scidac / INCITE19 / Randall)
- Community Access to Global Cloud Resolving Model Data and Analyses (Scidac / Schuchardt)
Cloud resolving models

- Finer resolution (< 4km) can resolve cirrus clouds, which strongly influence weather patterns
- Cloud-resolving models have been shown to agree with radar observations
- Could replace the cumulus and stratiform cloud parameterizations used in global models

“Cirrus Cloud Properties from a Cloud-Resolving Model Simulation…”
Images from Wikipedia
Global Cloud Resolving Models

- Questionable “parameterizations” are used to represent cloud effects in lower-resolution global models
- Computationally expensive to extend a cloud-resolving model to a global model
  - Now possible on high-end systems like Franklin and Jaguar
- GCRM model will be verified using satellite, radar, and in-situ observations

“Counting the clouds.” David Randall (2005)

Figure from Celal Konor, Joon Hee Jung, Ross Heikes, David Randall, Akio Arakawa
Geodesic grid

- Grid is constructed similar to a “subdivision surface”
- Cells can be ordered linearly using a space-filling curve

Figures from Bruce Palmer & Karen Schuchardt (PNNL) / Charlotte DeMott (CSU)
Visualization in VisIt

- Custom Visit plug-in written by Prabhat
- Loads geodesic grid data
- Parallel version forthcoming

- Temperature
- Vorticity
- Velocity
GCRM implementation

- 3.9 km resolution model
- 24 hour run on 30K nodes
- Generates 10TB of data
- Sustained 2GB/s write performance required for IO to take <5% runtime
GCRM IO pattern

Reproduced in IOR
2560 core test run
41943042 cells
@ 100 levels
= 6.1MB writes (shared)

File-per-proc

7MB writes (shared)
Tuning the IO pattern

Baseline from PNNL
Katie Antypas reconfigures MPI calls
Proof-of-concept IO pattern using MPI-IO in independent mode and 7MB transfers
Performance issues

- < 1GB/s write bandwidth when IO patterns do not align to lustre stripes
- Shared file performance is worse than file-per-proc, except in special cases
- MPI-IO collective mode (2-phase) is effectively broken in vendor library
Performance issues (MPI-IO)

Synchronous vs. Asynchronous Write Calls for Same IO Pattern

Cray’s MPI-IO Implementation (1294 MB/s) ~ MPI-IO VFD collective mode

IOR POSIX Shared File (6535 MB/s) ~ MPI-POSIX VFD

Test Parameters
- Nodes/stripes: 80
- Aggregate data: 40GB
- Stripe width: 8MB
- Write size: 8MB
- Writes per node: 64

Key
- Open
- Read
- Write
- Seek
- Close

Data collected and graphed using Noel Keen’s (LBNL) ipmMEGA library + tools.
Performance issues (MPI-IO)

- MPI-IO synchronous issue affects pNetCDF and GCRM API
- Introduced PNNL group to IPM profiling of IO performance

PNNL’s API / pNetCDF (collective mode)

H5Block alternative (independent mode)
Performance issues (HDF5)

- H5Block and HDF5 (MPI-POSIX VFD) performance is close to POSIX Shared File

![Graph showing write bandwidth comparison between POSIX, HDF5, and H5Block]
Performance issues (HDF5)

- Can only use chunking + padding in one dimension
- Splitting arrays into contiguous 1MB pieces without chunking is difficult
- Hongzhang Shan has created an unofficial HDF5 patch for multi-dimensional chunking/padding
  - Working with HDF5 group to integrate into official release
Performance issues (HDF5)

- 2-phase IO offers another solution:
  - Aggregate array on writer nodes
  - Writer node treats data as flat 1D array, which is split into 1MB segments
Upcoming IO improvements

- NERSC/HDF5 collaboration (recent workshop in January)
  - Add lustre hooks to HDF5 tunable parameters
  - Pad/align chunks to stripe boundaries
- New Cray MPI-IO implementation with improved 2-phase mode
  - Fewer writer nodes reduces burden on OSTs
  - Data shipping leverages SeaStar bandwidth
  - User space solutions are complicated: want solution at the MPI-IO level
- Hardware upgrades (just announced 3/11)