DXT: Darshan Extended Tracing

Cong Xu, Omkar Kulkarni, Vishwanath Venkatesan, Kalyana Chadalavada

Intel Corporation

Shane Snyder, Philip Carns

Argonne National Laboratory

Suren Byna

Lawrence Berkeley National Laboratory

Robert Sisneros

National Center for Supercomputing Applications

Outline

- Motivation
- Darshan eXtended Tracing (DXT)
- Overhead Measurement
- Case Studies
- Future Work

Motivation

- Optimizing I/O is difficult
  - Supercomputers evolve to exascale
  - Increasingly complex I/O subsystems
- The Challenge: Profiling Tools
  - Facilitate characterization of I/O activities
  - Existing tools: Darshan, ScalaIOTrace, Breeze HPC, LIOProf, LMT, etc.
- Need for a comprehensive solution.
  - More control over the resolution.
  - Minimal runtime performance impact.
  - Correlate data from multiple sources for complete picture

Darshan eXtended Tracing (DXT)

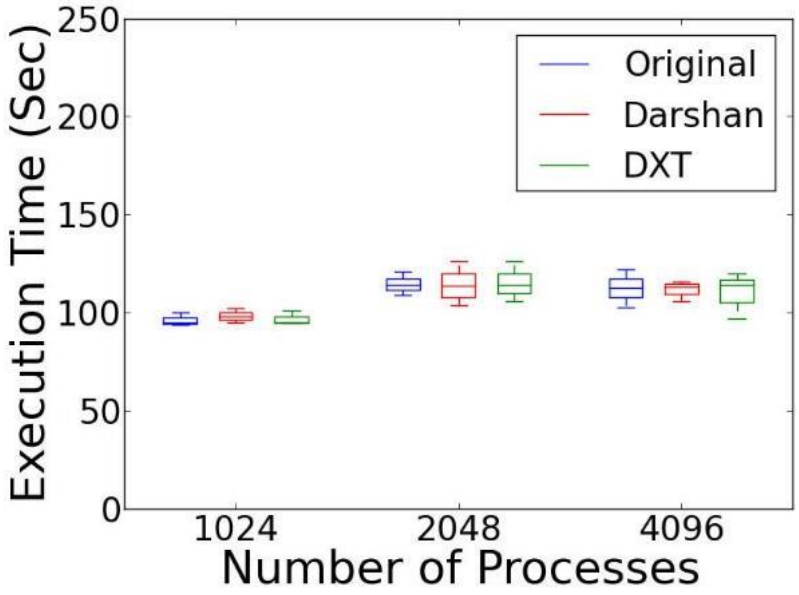
- What is Darshan?
  - I/O profiling tool from ANL deployed on many large systems.
  - Intercepts Application I/O and reports aggregate statistics
- What is “extended tracing”?
  - Enhance Darshan to (optionally) report every intercepted call.
  - Traces appear as a time series and can be post-processed offline.
  - Provide tools for applying different types of analyses to the logs.
  - Aggregate statistics and/or drill down to any level of granularity.

DXT Components

- Logging
  - Records each intercepted I/O call.
  - Request offset, length, start time, end time, MPI rank and the hostname.
  - Can be switched on or off at runtime using an environment variable.
  - Log buffer starts small and expands gradually as needed.
  - Uses compression to limit the size of the output log file.
- Analysis
  - Python script; basic analysis and visualization, but can be enhanced.
  - Correlates traces with Lustre striping information.
  - Group/filter requests by rank, host or Lustre OST.
  - Detects outliers.

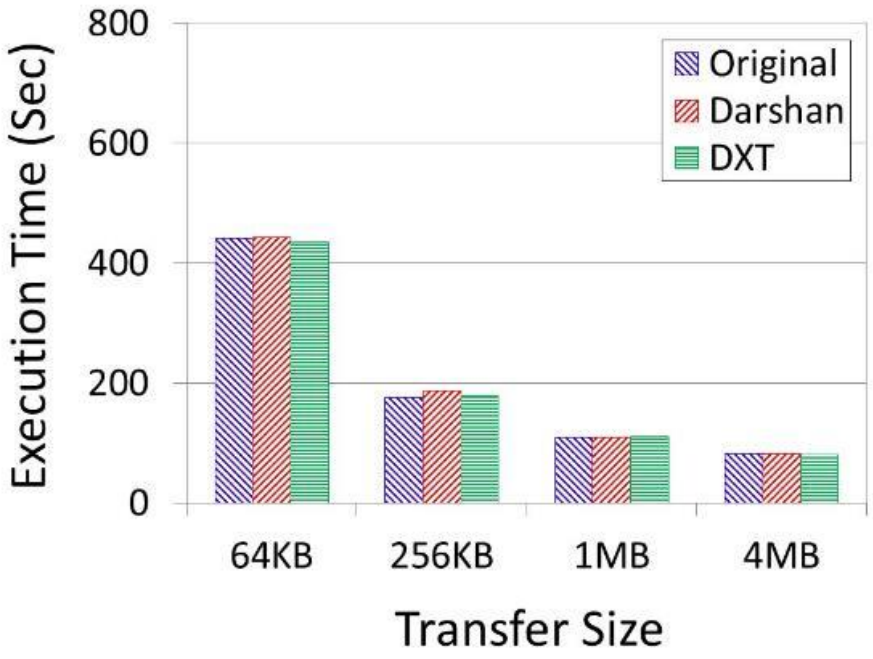
DXT Overhead: IOR on Cori

- Evaluation Environment
  - Range from 1024 to 4096 processes
  - Interleaved I/O on a single shared file
  - FileSize: 4TB, BlockSize: 4MB, TransferSize: 4MB, Aggregators: 128
  - Lustre – OSTs: 128, Clients: 128, Stripe Size: 4MB, Stripe Count: 128



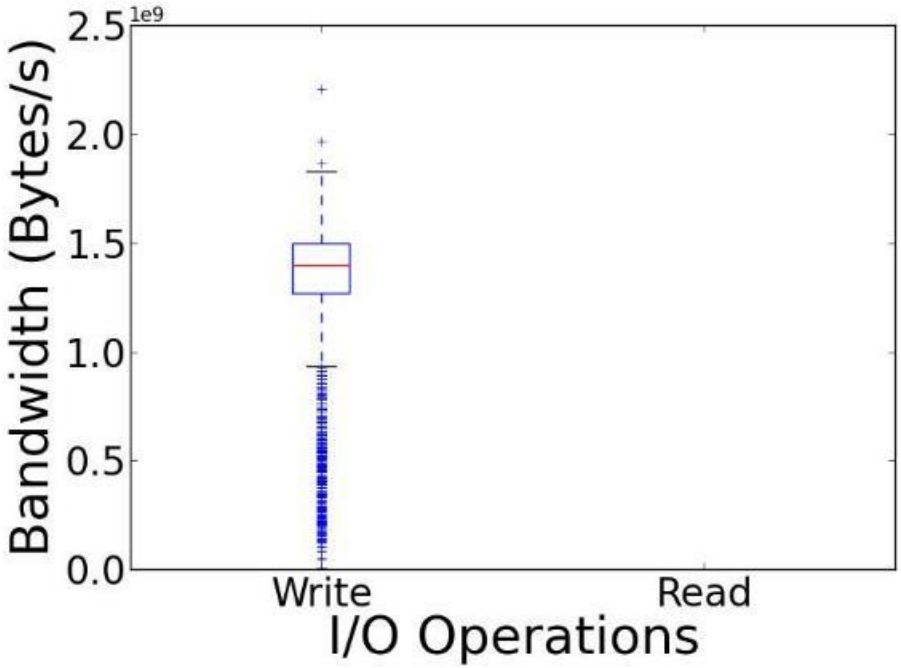
DXT Overhead: Varying Transfer Size

- IOR Transfer Size
  - Tunes the data transferred per I/O operation.
  - For constant file size, smaller transfers result in more I/O operations.
  - Larger DXT log file size due to more log entries.



Case Study: GCRM-IO

- I/O kernel of a climate code that models global atmospheric circulation.
- 256 processes write the pressure variable
  - Grid: 10, Subdomain: 4, Timesteps: 64



Case Study: GCRM-IO

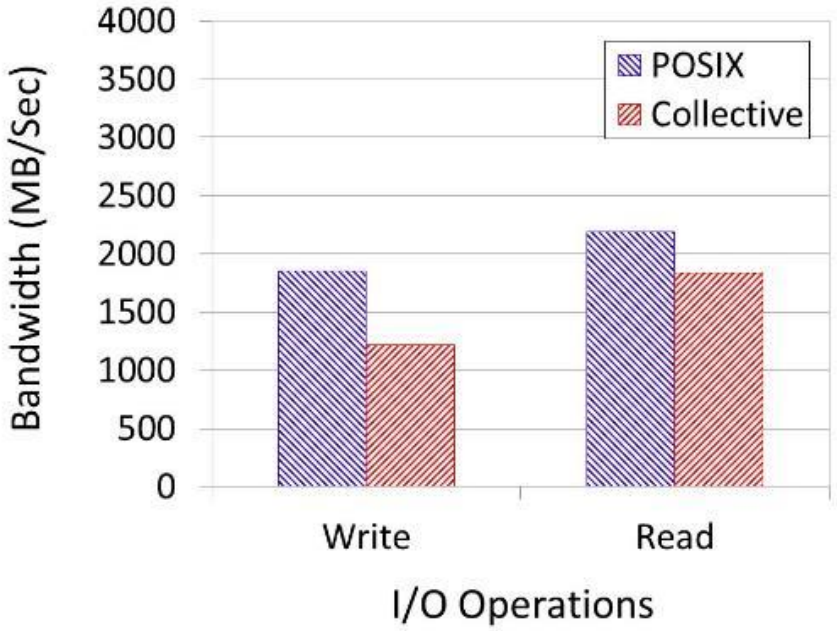
- Lock contention due to false sharing.
- Due to optimistic extent-based locks granted by LDLM.

Outliers in POSIX Write Operations
[Mean(GB/s): 1.28, Median(GB/s): 1.40]

Rank	Offset	Length	sTime	eTime	BW(Bytes/s)	Stripe	OST
0	96	40	24.87	28.41	11.30	0	0
1	800	40	24.88	28.38	11.41	0	0
2	1384	120	24.85	28.36	34.28	0	0
4	3568	328	24.88	28.38	93.63	0	0
3	2536	328	24.85	28.35	93.77	0	0
5	1090523536	40	24.85	28.35	11.41	260	0
7	1090525976	328	24.88	28.39	93.48	260	0
6	1090524944	328	24.87	28.38	93.65	260	0
9	2181047352	328	24.87	28.38	93.55	520	0
10	2181048384	328	24.85	28.36	93.66	520	0
12	3271568936	120	24.88	28.38	34.26	780	0
14	3271570792	328	24.88	28.39	93.47	780	0
13	3271569760	328	24.85	28.35	93.58	780	0
11	3271568024	328	24.85	28.35	93.78	780	0

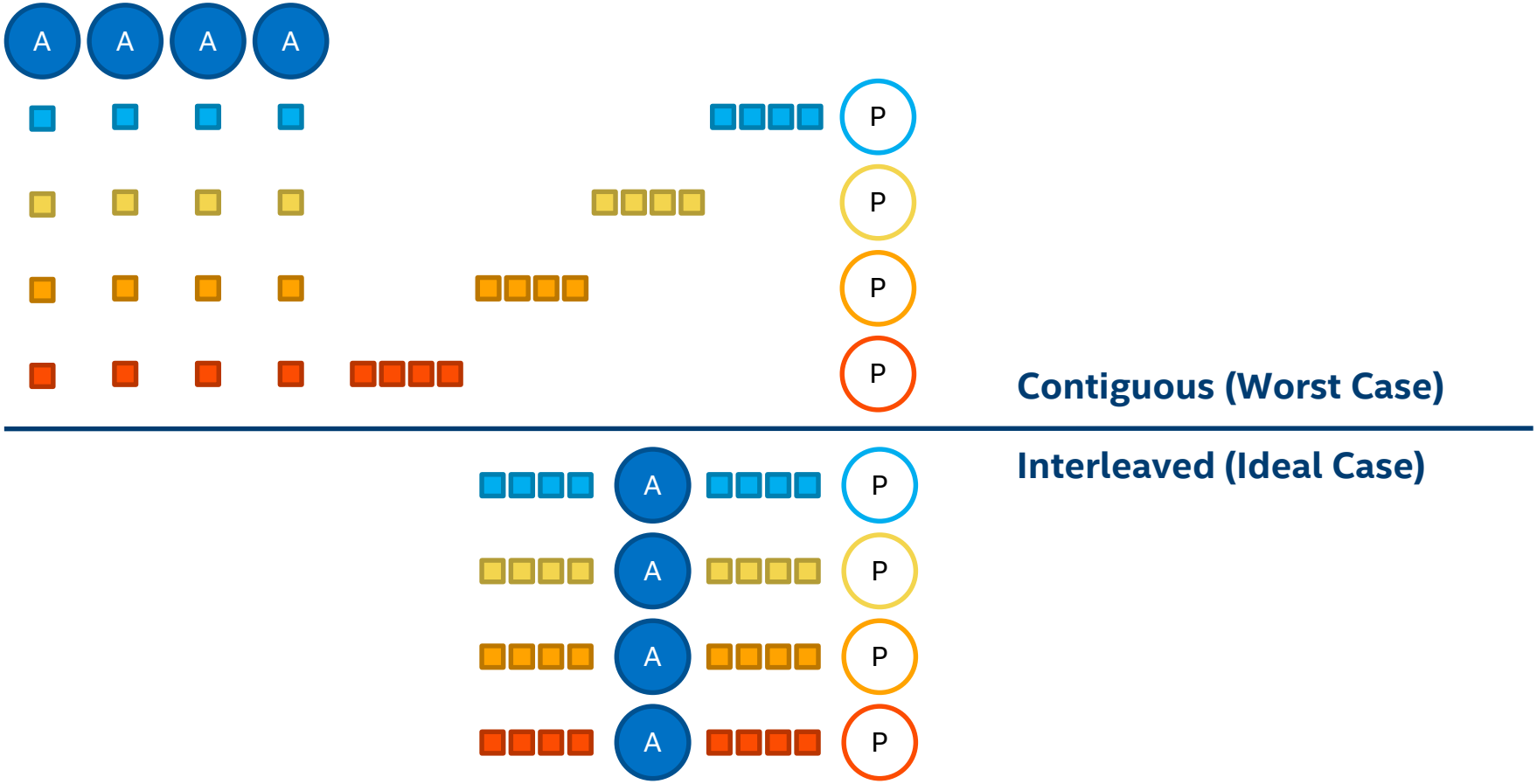
Case Study: HACC-IO

- I/O kernel of Hardware Accelerated Cosmology Code.
- 256 Processes write 8 billion particles to single shared file
  - ~300GB
- MPI Collective I/O performs much worse than POSIX



Aliasing

- Serialization of requests during communication phase.



Future Work

- DXT has landed and is available at: http://www.mcs.anl.gov/research/projects/darshan/
- First step to a comprehensive solution.
- Add more features to the analysis tool.
- Correlate data from multiple sources more effectively.
- Analysis over multiple jobs / system-wide.
- ML based adaptive caching/pre-fetching for reads.