The Search for Missing Parallel I/O Performance on the Cori Supercomputer

Matt Bryson
Advisors: Suren Byna, Alex Sim, John Wu
Lawrence Berkeley National Lab

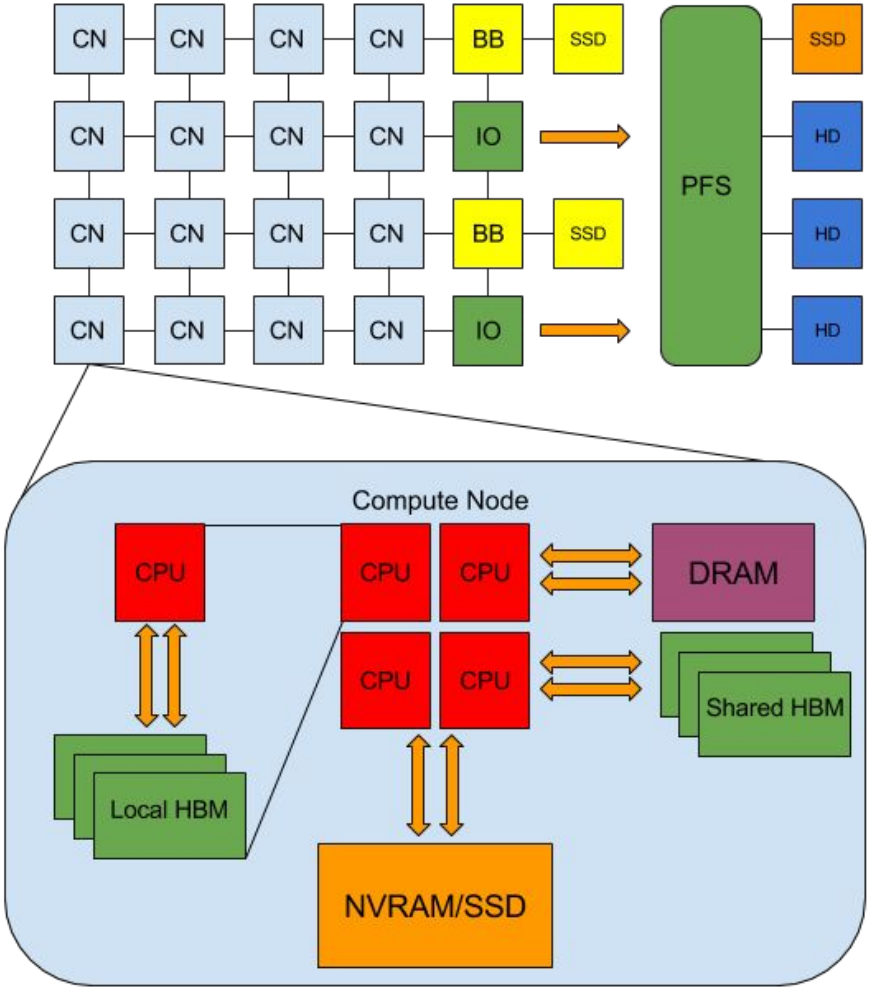
BERKELEY LAB

California Lutheran University

Baskin Engineering
UC SANTA CRUZ

Non-Volatile Storage & Its Place in HPC

- More data is being generated by HPC applications, making I/O systems the bottleneck for future HPCs
- To solve this, non-volatile storage is being integrated in the HPC memory/storage hierarchy
- HPC applications currently are not optimized for non-volatile storage

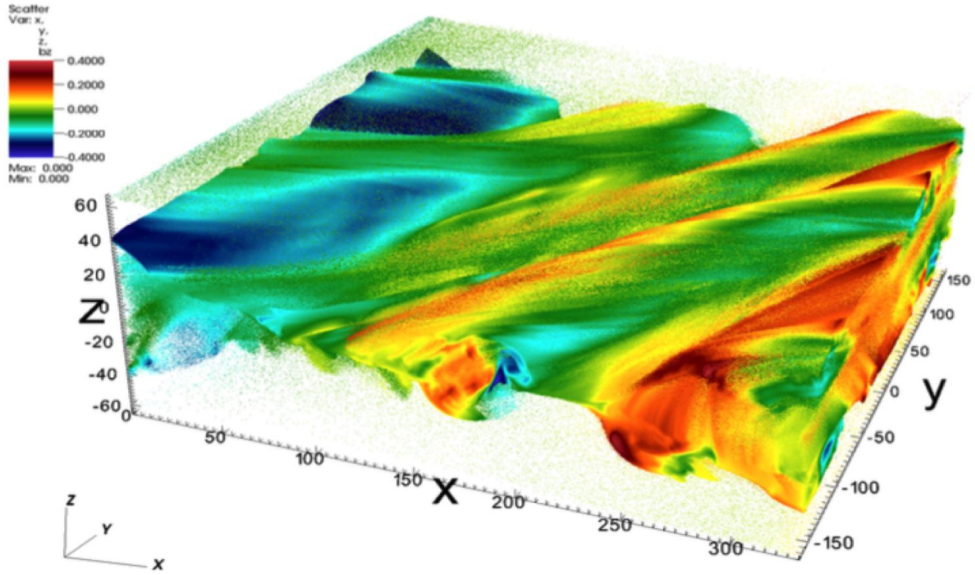


The Case of the Missing I/O Performance

- The IOR Storage Benchmarking Tool with sequential I/O showed that the Cori Burst Buffer could achieve 700 GB/S
- When tested with the Cori Burst Buffer, Vector Particle in Cell (VPIC), which uses a parallel I/O pattern, only performed at 15% of optimal (41 GB/S)
- There is extra time being spent on parallel I/O

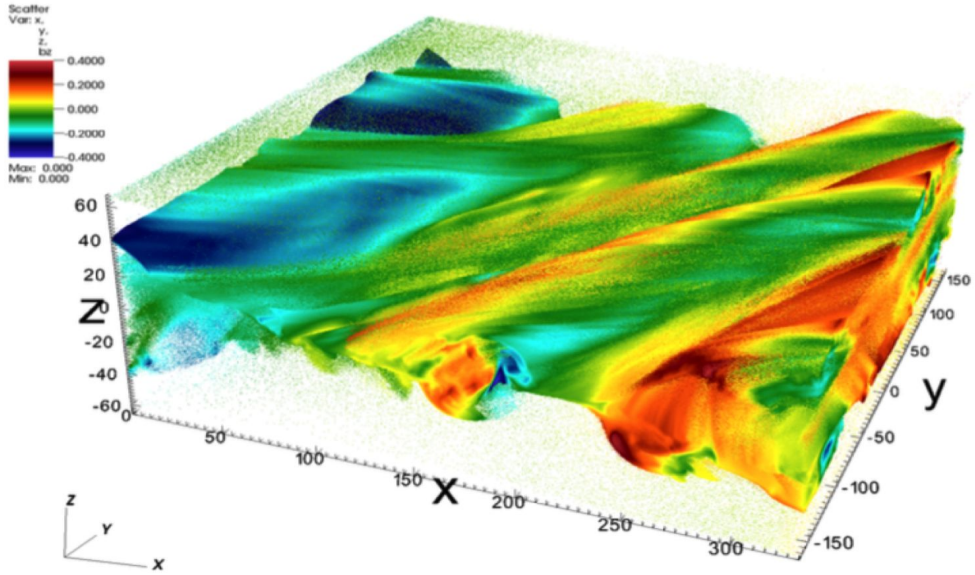
Methods of Analysis

- Vector Particle in Cell (VPIC) I/O Kernel
  - I/O Bound Application
  - Modifiable I/O Kernel
  - Affected by Parallel I/O
  - Instrumented to show performance difference with different HDF5 optimizations
  - Many I/O Steps
- Darshan Logs
  - Lightweight I/O Profiling Tool
  - Individual Process I/O Time
  - Used to show I/O Time Variance



Methods of Analysis

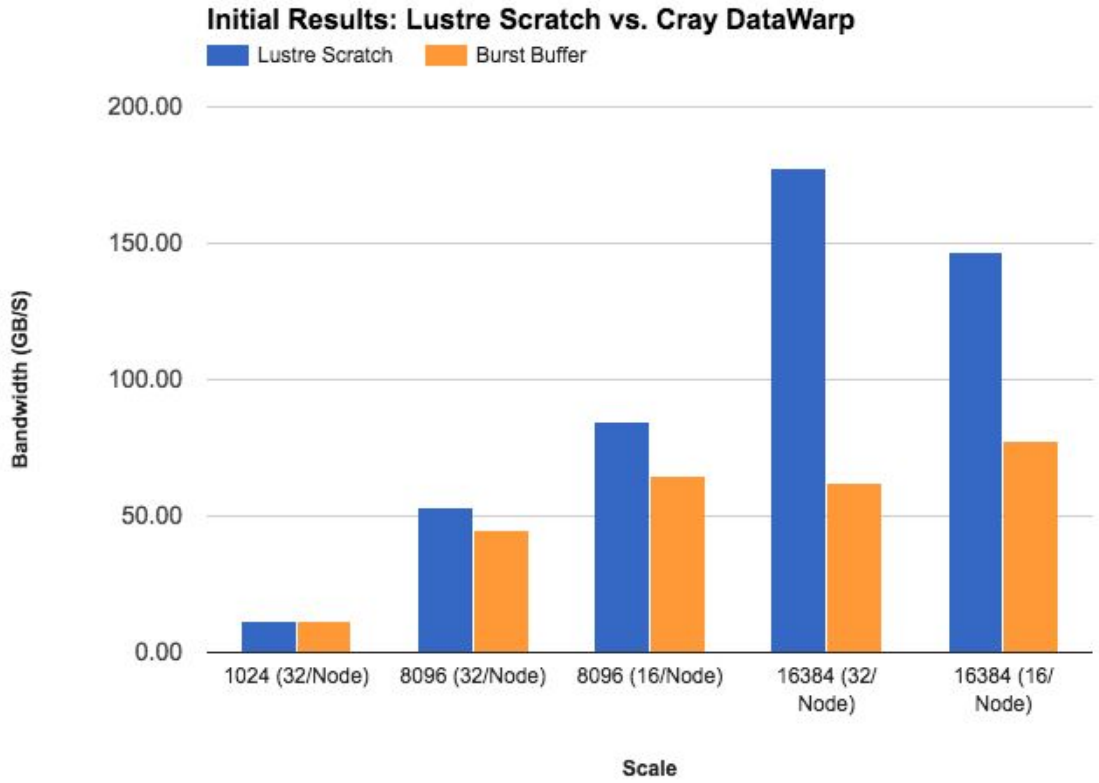
- Cray Performance Counters
  - Amount of data written per thread
  - Critical for determining data write imbalance
- IOR Benchmark Tool
  - Generation of sequential benchmarks



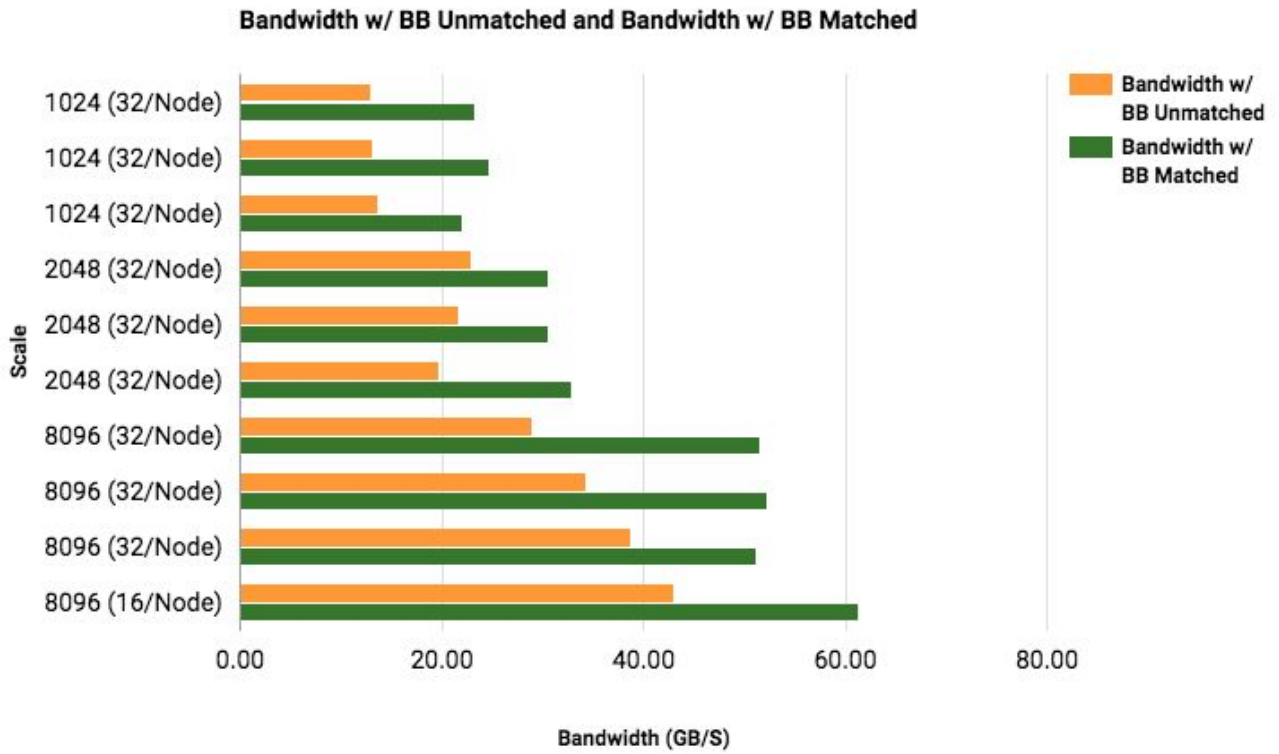
System Configuration

- NERSC Cori Phase 1 High Performance Computer
- Cray DataWarp Burst Buffer System
  - 64 or 65 Nodes
  - Storage granularity of 200GB per node

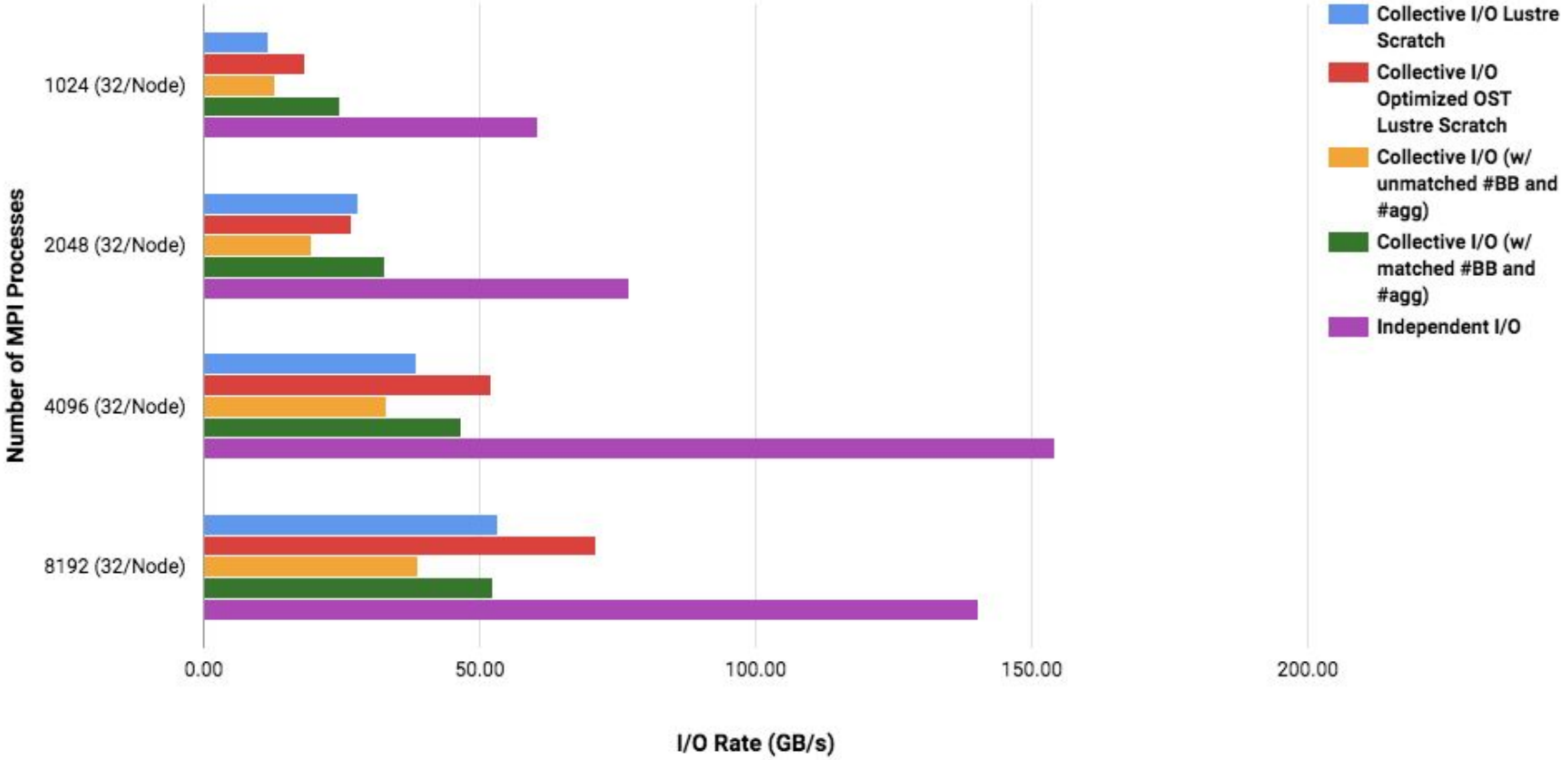
Results & the Path to Them



Results & the Path to Them



Results & the Path to Them



Conclusions

- Collective buffering on the Burst Buffer when using MPI-IO is a bottleneck.
  - Independent I/O mode bypasses collective buffering and results in better performance.
  - Independent I/O saturates the BB more effectively than collective I/O.
  - Using Independent I/O can result in up to a 4.6x performance increase.
- If Collective I/O must be used
  - Ensuring divisibility can result in an approximately 50% performance increase.

Acknowledgements

- Mentors
  - Alex Sim
  - Suren Byna
- PI
  - K. John Wu
- Support
  - Glenn Lockwood
  - Debbie Bard
  - Wahid Bhimji
  - Elizabeth Bautista
- Funding
  - Science Undergraduate Laboratory Internship (SULI), Lawrence Berkeley National Lab
  - Department of Energy Contract No. DE-AC02-05CH11231
- Travel Support
  - Baskin School of Engineering, UC Santa Cruz
  - Association for Computing Machinery

U.S. DEPARTMENT OF ENERGY

Office of Science

Association for Computing Machinery

Baskin Engineering
UC SANTA CRUZ

Sources Cited

[1] N. Liu, J. Cope, P. Carns, C. Carothers, R. Ross, G. Grider, A. Crume, and C. Maltzahn, “On the Role of Burst Buffers in Leadership-Class Storage Systems,” in Mass Storage Systems and Technologies (MSST), 2012 IEEE 28th Symposium on. IEEE, 2012, pp. 1–11. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs all.jsp?arnumber=6232369

[2] M. Funk, “The What And Why Of Burst Buffers.” [Online]. Available: http://www.theplatform.net/2015/05/19/ the- what- and- why- of- burst- buffers/

[3] “Cori.” [Online]. Available: http://www.nersc.gov/users/ computational- systems/cori/

[4] S. Byna, J. Chou, O. Rbel, H. Karimabadi, W. S. Daughton, V. Roytershteyn, E. Bethel, M. Howison, K.-J. Hsu, K.-W. Lin, and others, “Parallel I/O, analysis, and visualization of a trillion particle simulation,” in Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis. IEEE Computer Society Press, 2012, p. 59. [Online]. Available: http://dl.acm.org/citation.cfm?id=2389077