Experiences with a Burst Buffer at NERSC

Debbie Bard, Wahid Bhimji, David Paul, Glenn K. Lockwood, Nicholas J Wright, Katie Antypas, Prabhat
Science Apps: Steve Farrell, Andrey Ovsyannikov, Melissa Romanus, Brian Van Straalen, David Trebotich, Guenter Weber
Lawrence Berkeley National Laboratory, Berkeley, CA 94720 USA, Email: wbhimji@lbl.gov

***Index Terms*—Nonvolatile memory, Data storage systems, Burst Buffer, Parallel I/O, High Performance Computing**

I. Burst Buffer Architecture and Software

HPC faces a growing I/O challenge. One path forward is a fast storage layer, close to the compute, termed a Burst Buffer. Such a layer was deployed with the first phase of the Cori Cray XC40 System at the National Energy Research Scientific Computing Center (NERSC) at Lawrence Berkeley National Laboratory in the later half of 2015, providing around 900 TB of NVRAM-based storage on 144 nodes. With the Phase 2 Cori system, an additional 900 TB (144 nodes) of storage has been added to the Burst Buffer pool.

A. Burst Buffer Architecture

Each Cori Burst Buffer node contains two Intel P3608 3.2 TB NAND flash SSD modules attached over two PCIe gen3 interfaces. These are attached directly to the Cray Aries network interconnect of the Cori system.

B. Software Environment

In addition to the hardware, NERSC has invested in software projects with Cray and SchedMD, supporting the Cray DataWarp software and integration with the SLURM workload manager (WLM). Users can allocate Burst Buffer resources via the SLURM WLM. Resources can be striped across different Burst Buffer nodes, or used in a 'private' mode whereby each compute node gets its own namespace which potentially offers improved metadata handling. Details of the entire DataWarp software stack, including the services that create the mount points, can be found in the DataWarp admin guide [1].

C. Benchmark Performance

The performance of the Cori Phase 1 Burst Buffer was measured on installation using the IOR benchmark. Bandwidth tests were performed with 8 GB block size and 1 MB transfer size. IOPS benchmark tests were performed with random 4KB-sized transfers. 1120 compute nodes were used with 4 processes per node. At the time 140 Burst Buffer nodes were in service. Results are given in the table below. The MPI-IO shared file bandwidth has since been improved in later versions of the DataWarp software. All benchmark bandwidth numbers outperform the Lustre scratch filesystem.

Posix File-Per-Process		MPI-IO Shared File		IOPS	
Read	Write	Read	Write	Read	Write
905 GB/s	873 GB/s	803 GB/s	351 GB/s	12.6 M	12.5 M

II. Science Use-Cases

In August 2015, NERSC put out a call for proposals that could benefit from the Burst Buffer.

We focus on detailed results from two representative projects, from both the simulation and experimental science community. The latter provides interesting use-cases for the Burst Buffer in having both challenging I/O requirements and workflows that extend beyond the compute facility. For many more use-cases see [2].

A. Chombo-Crunch + VisIt

1) Project Workflow: Chombo-Crunch [3] is an MPI-based simulation software for modeling subsurface flow and reactive transport processes associated with carbon sequestration. The simulation generates data in the form of a single data file per time step, i.e., all MPI ranks contribute the data they computed to a single shared data file.

The other component application for this workflow is VisIt [4], a visualization and analysis tool for scientific data, which reads and analyses each data file as it is generated by Chombo-Crunch. The result is an image file.

This workflow is too IO-intensive for traditional disk-based storage, where Chombo-Crunch and Visit run sequentially and each datafile is written to the Lustre PFS. The end-result images are of scientific interest, and the intermediate files need not be retained. In the modified Burst Buffer workflow, VisIt runs simultaneously with Chombo-Crunch and can read data files directly from the Burst Buffer as soon as they are generated. In addition, VisIt can write the resultant image files to the Burst Buffer. These image files are staged out to the PFS at the end of the job.

2) Results: Figure 1 shows the bandwidth and scaling achieved by Chombo-Crunch in writing data to the Burst Buffer and to the Cori Lustre PFS. The Burst Buffer significantly out performs Lustre for this application at all resolution levels and the bandwidth scales exceptionally well.

B. ATLAS: data analysis for the ATLAS detector at the LHC

The ATLAS detector at the Large Hadron Collider (LHC) has been restricted only to running the least I/O intensive simulation workloads on HPC. The Burst Buffer allows the most I/O intensive 'analysis' applications to run on Cori. We analyse a 475G dataset of Atlas data per node fully occupied with 32 processes.

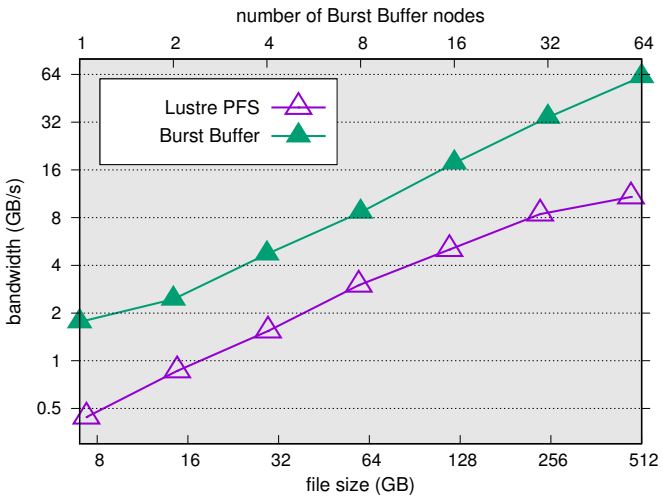

Fig. 1. Chombo-Crunch I/O bandwidth scaling.The compute node to Burst Buffer node ratio is fixed at 16:1

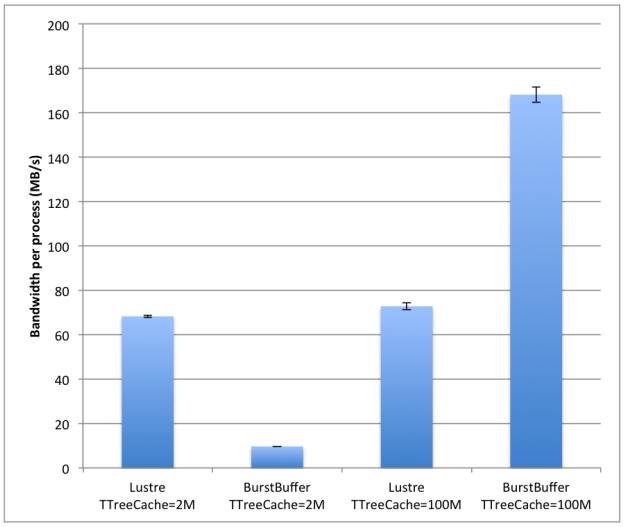

Fig. 2. Bandwidth per process for the ATLAS analysis application with the default 'TTreeCache' size of 2M and an increased cache of 100M.

1) Results: In Figure 2 we show the average bandwidth per process for reading data in this ATLAS data analysis application. The initial run showed significantly worse performance for the Burst Buffer than the Lustre filesystem – the application was making over 2 million read calls. Increasing the application-side memory cache size from 2.5 MB to 100 MB significantly improves the Burst Buffer performance which then out-performs Lustre. Results from scaling to 50 TB of data and 4096 cores is show in the poster.

III. Usage Monitoring

NERSC is collecting system-level monitoring information related to the Burst Buffer, which complements application-level metrics shown in the use-case section. For example:

- Intel SSD Data Centre tool for NVMe allows device-level I/O monitoring showed that the Burst Buffer workload was write heavy (see Figure 3).
- Collectd on Burst Buffer nodes allows node-level I/O monitoring which can be correlated with data from the Lustre LMT monitoring to track how an applications I/O translates to the system.

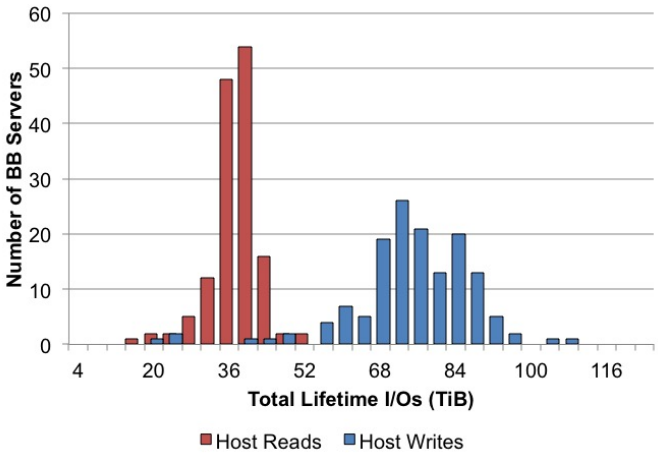

Fig. 3. Data read and written during early user period from low-level Intel SSD device monitoring.

IV. Conclusions

NERSC has successfully brought a Burst Buffer into production with its new Cori system. This offers a novel approach to creating flexibly-sized, on-demand filesystems backed by high-performance NVRAM hardware. The Phase 1 system is capable of around 900 GB/s bandwidth and 12.5M IOPS. We ran an Early User Program which was crucial to our debugging of this complex new technology. It exposed issues that would be impossible to identify with synthetic tests, and led to performance improvements as well as fixes to operational and usability issues. We have also put in place system level monitoring to further diagnose and tune performance. Through these efforts and thanks also to development by Cray and SchedMD, the NERSC Burst Buffer now functions well in production. We highlight here how the NERSC Burst Buffer provides a high-performance solution for scientific I/O and is now starting to enable new science workflows

References

[1] Cray. (2016) DataWarp Administration Guide. [Online]. Available: http://docs.cray.com/books/S-2557-5204/S-2557-5204.pdf

[2] W. Bhimji, D. Bard, M. Romanus, D. Paul, A. Ovsyannikov, B. Friesen, M. Bryson, J. Correa, G. K. Lockwood, V. Tsulaia *et al.*, "Accelerating science with the nersc burst buffer early user program," in *Proceedings of Cray Users Group*. [Online]. Available: https://cug.org/proceedings/cug2016_proceedings/includes/files/pap162.pdf

[3] D. Trebotich, M. F. Adams, S. Molins, C. I. Steefel, and C. Shen, "High-resolution simulation of pore-scale reactive transport processes associated with carbon sequestration," *Computing in Science & Engineering*, vol. 16, no. 6, pp. 22–31, 2014.

[4] H. Childs, E. Brugger, B. Whitlock, J. Meredith, S. Ahern, D. Pugmire, K. Biagas, M. Miller, C. Harrison, G. H. Weber, H. Krishnan, T. Fogal, A. Sanderson, C. Garth, E. W. Bethel, D. Camp, O. Rübel, M. Durant, J. M. Favre, and P. Navrátil, "VisIt: An End-User Tool For Visualizing and Analyzing Very Large Data," in *High Performance Visualization–Enabling Extreme-Scale Scientific Insight*, Oct 2012, pp. 357–372.