Simulating the Burst Buffer Storage Architecture on an IBM BlueGene/Q Supercomputer

Jian Peng[1], Sughosh Divanji[1], Ioan Raicu[1,2], Michael Lang[3]

[1] Department of Computer Science, Illinois Institute of Technology, Chicago, IL, USA
[2] Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL, USA
[3] Los Alamos Scientific Laboratory, Los Alamos, NM, USA

{jpeng10, sdivanji}@hawk.iit.edu, iraicu@cs.iit.edu, mlang@lanl.gov

***Abstract*- As the computing power of supercomputers keeps increasing, the I/O subsystem of these machines has become one of the major bottlenecks of the overall performance. Towards solving this problem, the burst buffer storage architecture has been adopted in the next generation supercomputers. In this paper, we simulated the burst buffer storage architecture on an IBM BlueGene/Q supercomputer using the CODES/ROSS simulation framework to study the potential I/O performance improvement with burst buffers. These results are a stepping stone towards studying this new storage architecture on future dragon-fly network based supercomputers, including resource management covering both storage and job scheduling.**

Keywords: supercomputer, I/O, burst buffers, simulation

I. Introduction

Recent trends in HPC applications are that they are becoming data-intensive which consume and produce large volumes of data, and have complex data dependencies. Accessing and retrieving large amounts of data is currently just a side effect on scheduling computations. One more trend observed in application I/O workloads in HPC systems is the burstiness. Ning Liu et al [1] show from the analysis of one month of production I/O activity that data is written in several bursts throughout job execution followed by significant period of idle time for I/O system.

Burst buffer architecture has been proposed to handle bursty I/O in HPC systems [2]. Burst buffers are high-throughput, low-capacity storage devices that act as a staging area or a write-behind cache for HPC storage systems. The approach we follow to incorporate burst buffers is to place these buffers on I/O nodes that connect to the external storage system and to manage these buffers as part of the I/O forwarding services. If burst buffers are sufficiently large and fast, they can absorb I/O bursts [1] effectively. With I/O requests aggregated and absorbed into the burst buffer layer, applications can overlap computations that follow I/O bursts while asynchronously pushing data to the storage for persistence. Meanwhile, burst buffer layer can provide some distributed file systems [8] more opportunities to improve data locality which is crucial for application performance [9].

We implemented our simulation using CODES (Co-Design of Multilayer Exascale Storage Architectures) [3] and ROSS (Rensselaer Optimistic Simulation System) [4] simulation framework. ROSS is a parallel discrete event simulator which uses time warp protocols [5] to simulate massive discrete events in parallel. CODES is built on top of ROSS, and it provides various network models for torus and dragonfly networks, and supports MPI collective communication. We measured the performance of our simulation for I/O operations of Mira (a 48K-node IBM BG/Q supercomputer at Argonne's ALCF) and compared the performance of the burst buffer architecture to the architecture without burst buffers and to the previous related work.

II. Implementation

Figure 1 is our modeling details of the BG/Q. Compute nodes (CN) are located in a 5D torus network. There are 16 computing cores on each CN. One midplane contains 512 CNs in 4 × 4 × 4 × 4 × 2 structure. One rack contains 2 midplanes and the entire system has 48 racks which sums up to 49,152 CNs, 786,432 cores. The bandwidth of the torus link is 2GB/s in both directions. There are 384 IO nodes(ION) connected to the torus network through specialized compute nodes called "bridge nodes". The CN/ION ratio in our simulation is 128:1. The bandwidth of each interconnection link between ION and DDN file system server is 4GB/s, which sums up to 1536 GB/s. There are 128 DDN file servers connected to disk arrays with total bandwidth of 250GB/s [6]. These simulation parameters closely match the real deployed BG/Q system Mira at Argonne.

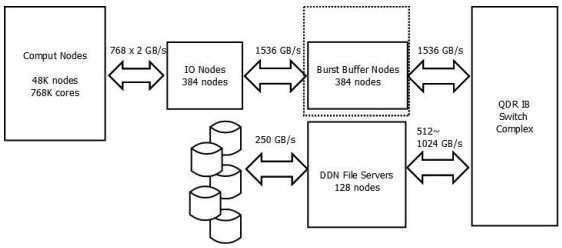

Figure 1. Modeling details

We have implemented a write protocol simulating the file write procedure on Mira as Figure 2.

III. Results

We conducted several sets of simulation with configurations of Intrepid (BG/P) and Mira (BG/Q). Due to memory limitation on the test system, the largest scale experiment tops

out at 131,072 compute processes. Figure 3 shows the aggregated write throughput with burst buffer enabled or not. Figure 4 shows the per IO node throughput with burst buffer enabled or not. We have validated our data on Intrepid and Mira respectively with Ning liu's et al [2] and Morozov's et al work[7]. From the results, we can see before 65K compute process scale, the aggregated throughput of Intrepid is higher than Mira. This is due to the bigger CN/ION ratio on Mira. Burst buffers on Mira gain a higher thoughput enhancement than Intrepid. This is because of the higher network bandwith between IONs and burst buffers on Mira.

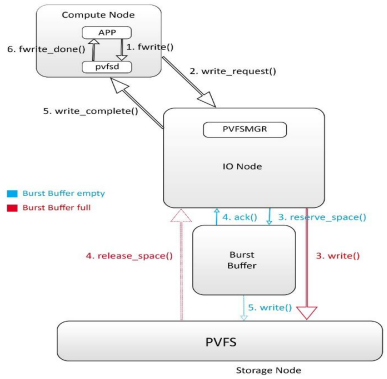

Figure 2. Burst buffer write protocol

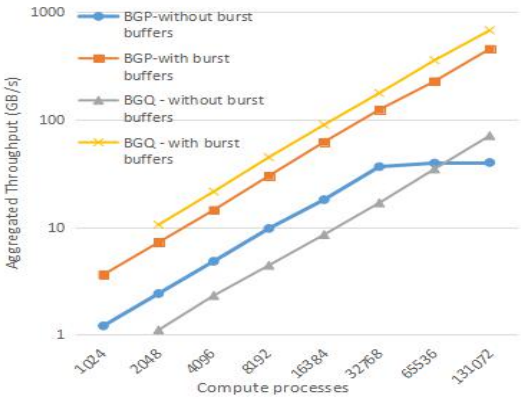

Figure 3. Aggregated write Throughput

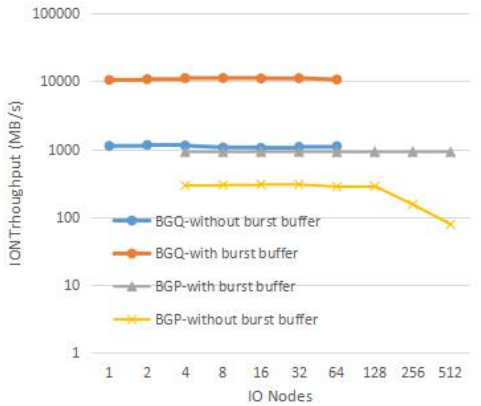

Figure 4. IO Node Throughput

IV. Related work

Four main areas of researches are related to our work: First, Ning Liu's et al burst buffer simulation work on Blue Gene/P. This work provides a stepping-stone and baseline for our work on Blue Gene/Q. Second, parallel file system simulations. As the foundation of HPC storage hierarchy, accurate modeling of PFS is crucial. There are many file system simulators like IMPIOUS, HECIOS and PFSsim. Third, CODES network modeling framework from Argonne National Laboratory. CODES provides various models for networks commonly used in supercomputers. And at last, ROSS discrete-event simulator. ROSS has been proven to be scalable well to support simulation at large scale [4] which makes it a powerful tool to help simulate ex-scale supercomputers.

V. Conclusion and Future Work

From our simulation results, we can conclude that: At the scale of our experiments (up to 131,072 cores), burst buffers can scale very well in Blue Gene/P and Blue Gene/Q storage architecture. The interconnection network between IO nodes and burst buffer nodes will be the bottleneck of full utilization of buffer buffers in Blue Gene series machines, and that bringing storage closer to the compute nodes can have a significant positive impact on performance.

Our future work are as follows: Simulating supercomputers with dragon-fly network topologies, to study the effectiveness of the burst buffer architecture. We also plan to explore real application IO traces. Furthermore, we will also study burst buffer aware job scheduling, where data movement between burst buffers and persistent storage, and among the burst buffer resources, can be minimized.

VI. References

[1] Liu, N., Carothers, C., Cope, J., Carns, P., Ross, R., Crume, A., & Maltzahn, C. (2011, September). Modeling a leadership-scale storage system. In*Proceedings of the 9th international conference on Parallel Processing and Applied Mathematics-Volume Part I* (pp. 10-19).

[2] Liu, Ning, et al. "On the role of burst buffers in leadership-class storage systems." *Mass Storage Systems and Technologies (MSST), 2012 IEEE 28th Symposium on*. IEEE, 2012.

[3] Cope, Jason, et al. "Codes: Enabling co-design of multilayer exascale storage architectures." Proceedings of the Workshop on Emerging Supercomputing Technologies. Vol. 2011. 2011.

[4] Carothers, Christopher D., David Bauer, and Shawn Pearce. "ROSS: A high-performance, low-memory, modular Time Warp system." Journal of Parallel and Distributed Computing 62.11 (2002): 1648-1669.

[5] Jefferson, David, et al. Time warp operating system. Vol. 21. No. 5. ACM, 1987.

[6] Chen, Dong, et al. "The IBM Blue Gene/Q interconnection network and message unit." 2011 International Conference for High Performance Computing, Networking, Storage and Analysis (SC). IEEE, 2011.

[7] Morozov, Vitali, et al. "Early experience on the Blue Gene/Q supercomputing system." Parallel & Distributed Processing (IPDPS), 2013 IEEE 27th International Symposium on. IEEE, 2013.

[8] Alex Szalay, Julian Bunn, Jim Gray, Ian Foster, Ioan Raicu. "The Importance of Data Locality in Distributed Computing Applications", NSF Workflow Workshop 2006

[9] Dongfang Zhao, Ioan Raicu. "Distributed File Systems for Exascale Computing, Doctoral Showcase, IEEE/ACM Supercomputing/SC 2012