BBAlloc: Towards Allocation based Management of Burst Buffer Systems

Sagar Thapaliya and Purushotham Bangalore
University of Alabama at Birmingham
Birmingham, AL, USA
Email: {sagar, puri}@uab.edu

Jay Lofstead
Sandia National Laboratories
Albuquerque, NM, USA
Email: gflofst@sandia.gov

Kathryn Mohror and Adam Moody
Lawrence Livermore National Laboratory
Livermore, CA, USA
Email: {kathryn, moody20}@llnl.gov

I. Introduction

In this work, we consider the problem of managing a Burst Buffer (BB) system in a shared environment. A BB system is a new storage technology for HPC architectures, that acts as an intermediate layer between performance-hungry HPC applications and the slow parallel file system [2]. It uses a tier of storage system, typically architected using non-volatile memory (NVM) such as flash-based SSDs. HPC systems equipped with BB will be available in the near future [1]. So, it is important to provide software infrastructure to manage resources and I/O traffic in BB, in order to support use cases such as checkpoint/restart and data staging.

In some recent works, researchers have presented and evaluated software systems to provide I/O access to BB systems and to conduct data management [3], [4]. However, they do not look at issues in shared BB systems for support of multiple jobs. Sharing can result in problems such as contention and load imbalance across BB nodes. Another issue that arises is: whether we should treat BB as a space resource such as memory or as a bandwidth resource such as parallel file system, or as something completely different.

We explore BB management as a resource allocation problem, with the goal of providing BB resources to applications, balancing the trade-offs between meeting application needs and whole system optimization. We first investigate implications of various allocation techniques including allocation based on space and efficiency requirements of applications and sharing with space- and time-sharing. We use analytic models, simulation, and empirical measurement to conduct this investigative study. Then we identify management requirements for BB resource allocation and present our BB allocation framework called BBAlloc, to capture those requirements.

II. BB Resource Allocation Problem

In this work, we focus on a particular architecture of a BB system based on the Trinity cluster [1], where the BB acts as a secondary storage system located on a set of dedicated nodes, and uses the same network as the compute nodes. When using such a BB system, jobs may have different requirement in terms of BB space and bandwidth. They need enough space to match with their output data size and bandwidth to meet their performance goals.

During our evaluation, we found that under our target BB system, it is important to consider both space and bandwidth requirement while allocating the number of BB nodes to a given job. In addition, we also found that it is not practical to allocate a subset of BB nodes to individual jobs for dedicated access, because of the small ratio of BB nodes to compute nodes. Instead, there will be need for sharing BB nodes across multiple jobs. In such shared system, applications can face I/O interference during concurrent access. We verified this behavior with empirical measurements on a test bed machine. We observed that the I/O bandwidth of a BB node gets divided across concurrent I/O processes. Another issue is wear levels of SSDs: SSDs can support only limited write cycles and it wears out after that. Therefore it is also important to balance SSD wear levels across BB nodes.

We argue that it is important to proactively manage allocation of BB resources to applications, and control these issues during the allocation.

III. BBAlloc Framework

Based on the above observations, we have begun to develop a framework called BBAlloc to manage allocation of space and bandwidth of a shared BB system. Under BBAlloc, we plan to manage BB resource allocation using multiple steps, including allocation of storage space, dividing the space into multiple partitions, and placement of these partitions on physical BB nodes. During placement, our goal is to balance multiple performance trade-offs across BB nodes, such as concurrent I/O traffic, SSD wear level and free space availability. In this talk, we will discuss the motivating issues for BBAlloc and also outline its design.

References

[1] Trinity-Overview. http://www.lanl.gov/projects/trinity/_assets/docs/trinity-overview-for-web.pdf. [Online; accessed 7-Sept-2015].

[2] 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*, pages 1–11, 2012.

[3] K. Sato, K. Mohror, A. Moody, T. Gamblin, B. R. Supinski, S. Maruyama, and S. Matsuoka. A User-Level InfiniBand-Based File System and Checkpoint Strategy for Burst Buffers. In *2014 14th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, Chicago, IL, USA, May 26-29, 2014*, pages 21–30, 2014.

[4] T. Wang, S. Oral, Yandong Wang, B. Settlemyer, S. Atchley, and Weikuan Yu. Burstmem: A high-performance burst buffer system for scientific applications. In *Big Data (Big Data), 2014 IEEE International Conference on*, pages 71–79, Oct 2014.

This work is performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344 (LLNL-CONF-677247). It is also supported in part by the National Science Foundation under Grant No. 12292820. Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.