LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-CONF-667209

# Towards Cloud-based Burst Buffers for I/O Intensive Computing in Cloud

T. . Xu, K. . Sato, S. . Matsuoka

February 12, 2015

HPC in Asia in conjunction with ISC High Performance conference
Frankfurt, Germany
July 12, 2015 through July 16, 2015

## Disclaimer

# Towards Cloud-based Burst Buffers for I/O Intensive Computing in Cloud


Tianqi Xu
*Dept. of Mathematical and Computing Sciences*
*Tokyo Institute of Technology*
*2-12-1-W8-33, Ohokayama, Meguro-ku, Tokyo 152-8552 Japan*
*Email: jyo.t.aa@m.titech.ac.jp*

Kento Sato
*Center for Applied Scientific Computing*
*Lawrence Livermore National Laboratory*
*Livermore, CA 94551 USA*
*Email: kento@llnl.gov*

Satoshi Matsuoka
*Global Scientific Information and Computing Center*
*Tokyo Institute of Technology*
*2-12-1-W8-33, Ohokayama, Meguro-ku, Tokyo 152-8552 Japan*
*Email: matsu@is.titech.ac.jp*



***Abstract*—A deluge of data has been generated from distinctive domains, and analyzing such big data becomes a pressing need. Cloud environments provide virtually unlimited computation resources to enable users to run applications at larger scale than their local systems. However current cloud environments are suffering from low I/O bandwidth between compute nodes and shared storage. Here we introduce cloud-based burst buffers as a new tier in cloud shared storage. We implement a prototype, and evaluate it on Amazon EC2/S3. The results show that our proposal system can accelerate sequential I/O performance for a single client as well as linear scale-out when the number of burst buffer nodes increases.**



***Keywords*-cloud, burst buffers, data intensive computing**


## I. INTRODUCTION

There are growing interests on cloud computing for its high scalability, high computational resources, and fast setup as well as on-demand usage available. Such cloud environments are suitable for large-scale distributed computation. However, shared cloud storage, for example, Amazon Simple Storage Service (Amazon S3), can only provide several hundreds MB/s throughput [1], and can not meet I/O bandwidth requirements for large-scale data intensive applications. The low I/O bandwidth means substantial increases in execution time, as well as monetary cost in cloud.

Burst buffers have been proposed as a new tier to absorb bursty I/O request in HPC systems. [2]. In this paper, we apply burst buffers to cloud environments, and propose cloud-based burst buffers in cloud storage systems to accelerate I/O performance. Because the LAN connection inside a cloud usually can achieve high bandwidth, furthermore, HPC applications, especially *data intensive workflow* applications, show high data locality in I/O [3], we can improve applications data access performance by buffering I/O data in compute nodes located in the same LAN environment. Hence in our design, we use several dedicated nodes in the same LAN environment as burst buffer nodes to buffer I/O data in main memory. We implement a prototype of our proposal system, and evaluate it in a real cloud environment, Amazon EC2/S3. The results show that our proposal system can greatly improve I/O performance for a single client, as well as show linear scale-out when the number of burst buffer nodes increases.

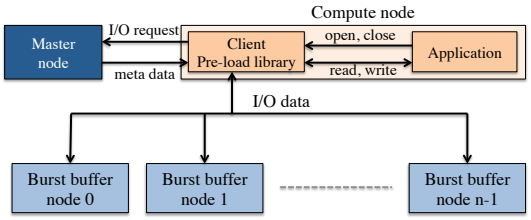


Figure 1. Overview of prototype

## II. CLOUD BASED BURST BUFFERS

We define our cloud-based burst buffers as a new tier between compute nodes and shared storage in cloud. Figure. 1 shows the design of the cloud-based burst buffers. There are three kinds of nodes in our prototype:

- *Master node*: A node managing file metadata and *Burst buffer nodes* information such as *Burst buffer nodes* list.
- *Burst buffer nodes*: Nodes storing actual data in main memory, and being responsible for sending to and receiving data from *Compute nodes*.
- *Compute nodes*: Nodes running user applications, and interacting with *Master node* and *Burst buffer nodes*.

In read cases, I/O performance can be improved when reading data buffered in the burst buffer nodes. As well as in write case, buffering output data in the burst buffer nodes with high I/O bandwidth can also save time.

To reduce the modification of application source codes, our proposal system supports standard POSIX APIs. We use a preload library to override these POSIX API calls and determine whether we should use cloud-based burst buffers according to a specified file path.

## III. EVALUATION

### A. Experiment Setting

We introduce the evaluation of our prototype implementation. We evaluate sequential read and write performance

| Region | Tokyo |
|---|---|
| Instance Type | m3.xlarge |
| vCPUs | 4 |
| ECUs | 13 |
| Memory | 15GiB |
| Instance Storage | 2*40GB(SSD) |
| Network Performance | High |
| Mount Method | s3fs |

Table I
EXPERIMENT ENVIRONMENT

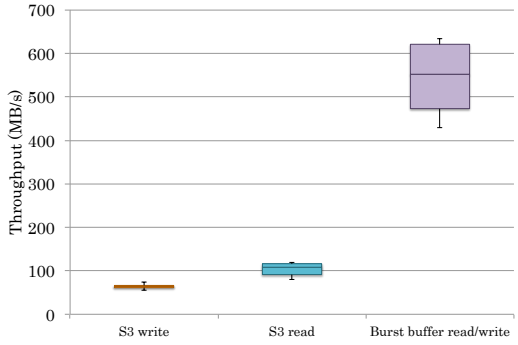


Figure 2. Accelerate single compute node I/O performance (data size=100 MB)

on Amazon EC2/S3. The environment is shown in Table. I. In order to simplify the evaluations, we make following assumptions:

- One compute node access to only one burst buffer node.
- All I/O data can be buffered in the burst buffer nodes.

### B. Sequential Read & Write Performance

Firstly, we evaluate I/O performance of a single compute node to figure out how our proposal system can accelerate I/O performance on Amazon cloud. Figure. 2 shows the performance comparison of sequential I/O with and without our proposal system, the data size is 100 MB. As shown in Figure. 2, our proposal system can achieve 550 MB/s in both read and write. In contrast, S3 can only achieve 108 MB/s in read, and 63 MB/s in write, thus we can improve 5.1x performance in read and 8.6x in write.

Then, in order to show how our proposal system scales out when the number of burst buffer nodes increases, we evaluate the entire system I/O throughput using multiple burst buffer nodes and multiple clients. Here we let client number equal to burst buffer node number, and each client accesses to one burst buffer node independently. As shown in Figure. 3, our proposal system shows linear scale-out as the number of burst buffer node increases, and can improve I/O performance up to 4.8x in read and 9.3x in write using 8 burst buffer nodes.

Since data intensive HPC applications have heavy I/Os and high I/O data locality, our proposal system can improve I/O throughput for reusable data and output data, and hence we expect our system can accelerate these data intensive HPC applications.

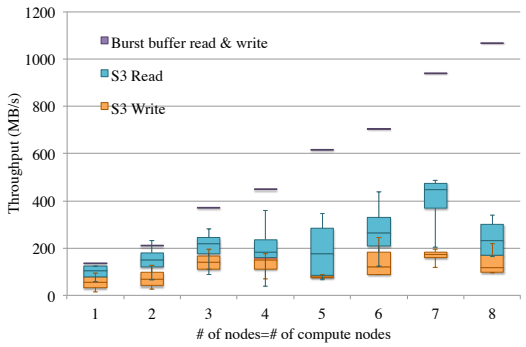


Figure 3. Scale-out as the number of burst buffer nodes increases (data size=1 GB)

## IV. CONCLUSION AND FUTURE WORK

We have introduced cloud-based burst buffers as a new tier of cloud storage to improve I/O throughput in cloud. We also have implemented a prototype of our proposal system and evaluated it on Amazon EC2/S3. According to the results, our proposal system can improve I/O performance for a single client greatly, and show linear scale-out as the number of burst buffer nodes increases.

For future work, we will evaluate real applications on our proposal system. Also, on-demand burst buffer node allocation is critical for cost reduction in cloud. Furthermore, fault tolerance is always the key part of a file system, so we will introduce fault tolerance to our system.

## V. ACKNOWLEDGEMENT

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. (LLNL-CONF-667209). This research was supported by JST, CREST (Research Area: Advanced Core Technologies for Big Data Integration).

## REFERENCES

[1] T. Chiba, M. den Burger, T. Kielmann, and S. Matsuoka, "Dynamic load-balanced multicast for data-intensive applications on clouds," in *Cluster, Cloud and Grid Computing (CCGrid), 2010 10th IEEE/ACM International Conference on*, May 2010, pp. 5–14.

[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*, April 2012, pp. 1–11.

[3] E. Deelman, G. Singh, M. Livny, B. Berriman, and J. Good, "The cost of doing science on the cloud: The montage example," in *High Performance Computing, Networking, Storage and Analysis, 2008. SC 2008. International Conference for*, Nov 2008, pp. 1–12.