Design and Modelling of Cloud-based Burst Buffers

[Extended Abstract]

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

Kento Sato
Center for Applied Scientific Computing
Lawrence Livermore National Laboratory
Livermore, CA 94551 USA
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
matsu@is.titech.ac.jp

1. INTRODUCTION

There are growing interests on public cloud computing for its high scalability, high computational resources, and fast setup as well as on-demand usage available. With the growing data size, as known as *Big Data*, public cloud computing offers high computational resources to fulfill the requirement of *Big Data* processing. However, low I/O performance and loose consistency model in shared cloud storage systems degrade applications running on cloud.

In order to accelerate I/O performance and resolve the consistency issue in cloud storage systems, we have proposed a cloud-based burst buffer system (CloudBB) as a new tier in cloud storages hierarchy. However, the system configurations have great impact on the performance of our system and the approach to achieve the optimal performance remains unsolved in our previous work. Hence in this paper, we introduce the performance model of our system to predict he performance and help to determine the optimal configuration while using our system. According to the experiment results of a real HPC application on real public cloud Amazon EC2/S3, our model can predict the optimal configuration.

2. CLOUD-BASED BURST BUFFERS

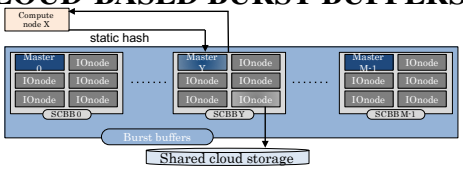

Figure 1: Architecture of cloud-based burst buffers

We have proposed CloudBB as a new tier in cloud storage hierarchy [3]. We use several dedicated instances as *burst buffers* to provide remote data cache with the same LAN environment to accelerate I/O performance and resolve the consistency issue. Data buffered in CloudBB can be written back to remote shared storage for fault tolerance. We implemented our system on the top of FUSE framework [1], and supported standard POSIX APIs. Figure 1 shows the architecture of our system, the whole system consists of several Sub CloudBBs (SCBBs) and in each SCBB there are three kinds of nodes in our system:

- *Masters* manage file metadata and *IOnodes* informations;
- *IOnodes* store actual data in main memory, and send/receive data from user's applications;
- *Compute Nodes* run users' applications and interact with *Masters* and *IOnodes*.

As shown in Figure 2 and 3, our system can greatly

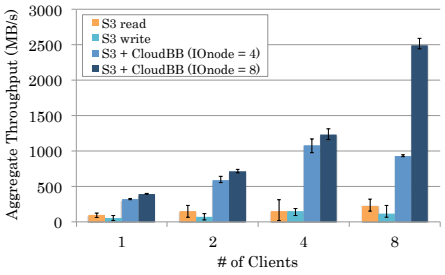

Figure 2: Sequential Performance Comparison

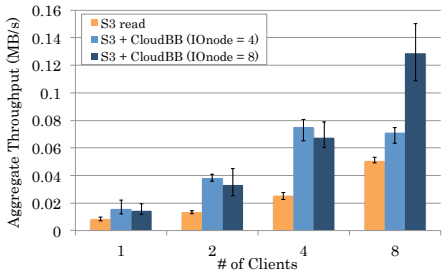

Figure 3: Random Performance Comparison

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

Table 1: Experiment Environment

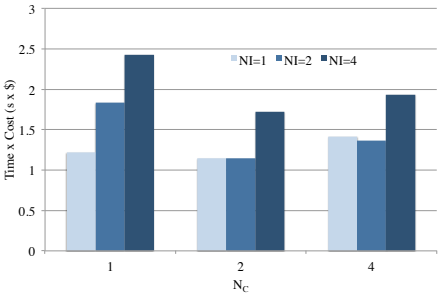

Figure 4: Experiment Results of Montage (Time * Cost)

accelerate the I/O performance. Our evaluations on several HPC applications like Montage [2] show that the improvement we achieved in I/O performance can also greatly accelerate the execution of these HPC applications.

3. PERFORMANCE MODEL

As shown in Section 2, the system configuration such as the number of IOnodes can have great impact on the performance. According to Figure 2 and 3, I/O throughput can be improved greatly by increasing the number of IOnodes, however, according to cloud pay-as-you-go pricing policy, using more nodes means more cost. Hence it is critical for users to decide the optimal configuration to achieve high performance as well as save cost. Here we introduce the performance model to predict the performance under a given configuration and help users to determine the optimal configuration according to the characteristics of their applications and execution environment. First, we make two assumptions to simplify the model: (1) The Master evenly distributes data of applications across IOnodes so that I/O workloads are balanced across IOnodes; (2) If multiple Compute Nodes access to a single IOnode, the bandwidth is divided by the number of Compute Nodes accessing to that IOnode.

Our model focuses on two aspects of the execution: execution time (*Time*) and the overall cost (*Cost*). The model optimize the number of Masters (N_M), IOnode(N_I) from total I/O size (D_{input} and D_{buff}), cost of the instances (P_M, P_I, P_C), I/O throughput ($Thr_{CloudBB}$ and Thr_{Cloud}) and number of Compute Nodes used (N_C). In order to achieve both the optimal of execution time and overall cost, our model optimize N_M and N_I by minimizing the value of $Time \times Cost$.

4. EVALUATION RESULTS AND MODEL PREDICTION

Variable	Meaning	Value
D_{input}	The total input size	25 MB
D_{buff}	The total data size can be buffered in burst buffer	215 MB
T_C	The total time in computation	2.638 sec
r	The radio of tasks must be executed serially in total tasks	0.45
$P_C = P_I = P_M$	Unit price of node	0.405 $/sec
N_C	The number of Compute Nodes	1, 2, 4
N_I	The number of IOnodes	1, 2, 4
N_M	The number of Master Node	1
Thr_{Cloud}	The average throughput of cloud storage	18 MB/s
Thr_m	The throughput of CloudBB under the given configuration	135 MB/s

Table 2: Dataset and Experiment Setting Details

# of Compute Nodes	Optimal # of IOnodes (Prediction)	Optimal # of IOnodes (Experiment)
1	1	1
2	1	1
4	2	2

Table 3: Prediction Results

In order to validate the effectiveness of our model, we introduce the evaluation of our system on a real public cloud Amazon EC2/S3. The environment is shown in Table 1. We evaluate a real HPC application, Montage [2]. The data set we used is shown in Table 2.

According to the results of Montage [2] (Figure 4) and the prediction shown in Table 3, our model can predict the performance and optimal configuration while using our system.

5. CONCLUSION

We proposed performance model for our cloud-based burst buffers to predict the performance and help users to determine the optimal performance. We validate our model using the experiment result of HPC application on Amazon EC2/S3. According to the results, our performance model can predict the performance and determine the optimal performance.

6. 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-676071). This research was supported by JST, CREST (Research Area: Advanced Core Technologies for Big Data Integration).

7. REFERENCES

[1] FUSE. http://fuse.sourceforge.net/.
[2] Montage. http://montage.ipac.caltech.edu/docs/grid.html.
[3] T. Xu, K. Sato, and S. Matsuoka. "Cloud-based Burst Buffers for I/O Acceleration". In *Summer United Workshops on Parallel, Distributed and Cooperative Processing (SWoPP), 2015*, July 2015.