Accessing CEPH’s Performance
Comparing NVMe and RAM

Gabryel Mason-Williams
Diamond Light Source
gabryel.mason-williams@diamond.ac.uk

Abstract

Accessing CEPH’s Object storage performance on RAM and how the different sizes of RAM block effect it, then seeing how that compares to NVMes to see the viability of using CEPH at Diamond Light Source.

Introduction

Diamond Light Source is currently exploring new ways of handling data, as current detectors on Diamond’s beamlines can generate between 1Gb/s and 40Gb/s and in the future 800Gb/s. Objects stores may prove to be faster than file systems and be more able to deal with a broader range of file (object) sizes with less tuning whilst being easier to integrate into web applications such as remote data analysis for the cloud.

Main Objectives

The main objects of this project so far have been to test and see what kind of performance we can expect from CEPH when it is running on RAM instead of a standard OSD. Since we are using RAM to remove bottlenecks, this will show what the potential performance would be when needed if used as a caching tier. With the end goal of creating a cluster deployable on RAM CEPH instance for data processing.

Testing and Benchmarking Methodology

The systems network read and write speeds were recorded using iperf[1] for the network and dd[7] and hdparm[5] for read and write respectively to create a theoretical maximum that the cluster could achieve, each operational test was run five times then averaged.

OSD Node Setup

1. Intel(R) Xeon(R) CPU E5-2670 0 @ 2.60GHz

2. RAM speed 1333Hz at 94GB total

3. NVMe Model: INTEL SSDPED1K375GA 375GB

Mon Setup

1. Intel(R) Xeon(R) Bronze 3104 CPU @ 1.70GHz

To test the performance of the cluster, we used a program called cosbench[4] with all CEPH pools replication size set to 1, and the size of the object being written and read set to 1MB. For clusters with OSDs less than six the pg_num and pgp_num was to 64, OSDs with 18 or 12 it was 512, and for 90 OSDs it was 4096. Each test operation was run five times and then averaged to produce the result.

Using RAM Blocks instead of RAM disks

Since CEPH uses LVM[6] to place its underlying system bluestore onto the block devices, this meant that the RAM had to be represented as a generic block device, as ramdisks are a specialised block device that are incompatible with LVM and thus bluestore. Creating a new module that is specifically designed to make a generic block device store on RAM allowed CEPH to run on RAM; as it a block than LVM can use.

RAM Blocks used

Two RAM blocks were used: the BRD module[2] and the one based on the compression Linux Kernel module ZRAM[3]. This RAM block could be used to create a compressed block device on RAM acting as either a swap partition or general block. However, the block being used GRAM (General RAM) is a mirror copy of ZRAM but has had the compression section removed and the corresponding checks that were required to make sure the compression had happened correctly to help improve performance, as in this project the compression was not required.

Results

Table 1 shows the benchmark and average result expected from one of these drives by default, with this data we would expect the best performance when running as OSDs to be a size of 10GB for the GRAM Module and BRD module.

Size GB	Drive	Write GB/s	Buffered Reads MB/s
2	/dev/gRAM0	1.52	1369.95
10	/dev/gRAM0	1.42	1956.188
15	/dev/gRAM0	1.34	1992.458
30	/dev/gRAM0	1.46	1960.6
60	/dev/gRAM0	1.3	2046.948
2	/dev/RAM0	1.0714	1452.018
10	/dev/RAM0	1.4764	2057.402
15	/dev/RAM0	1.164	1893.93
30	/dev/RAM0	1.0628	2041.676
60	/dev/RAM0	1.043	1877.55
375	/dev/NVMe0n1	1.08	1773.9812

Table 1: Write and read of drives

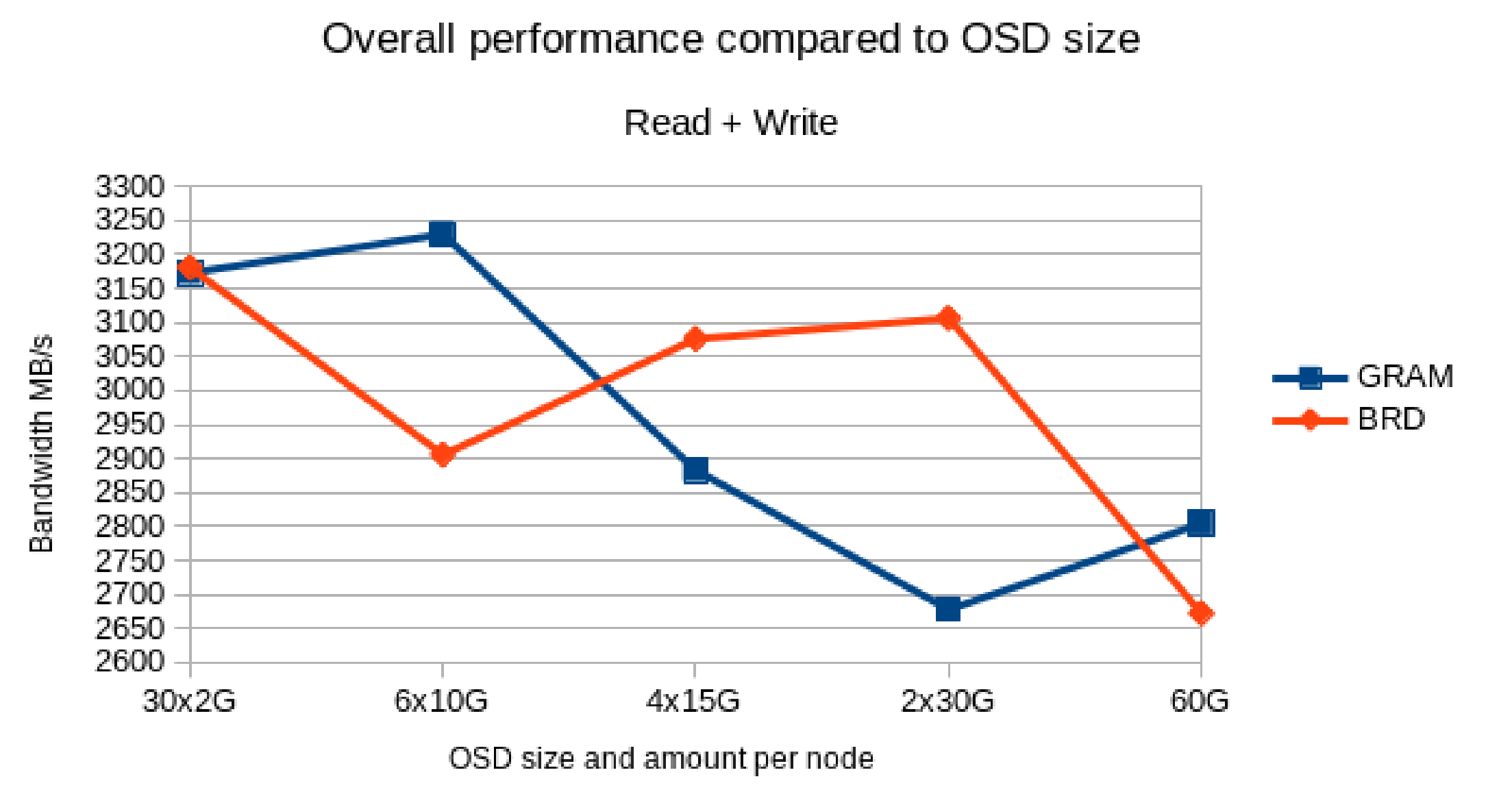

Figure 1: Overall Performance compared to OSD size in RAM modules

From figure 2, it is clear that changing the size of the RAM blocks can significantly impact the performance of the CEPH cluster, with the GRAM Module following a linear increase with each decrease in size; with 10GB providing the best for overall performance, whereas the BRD module does appear to follow the same trend, prefering larger sizes of block size. However, BRD best overall performance came from a block size of 2GB, which is surprising as this was the least performant hardware size, suggesting that CEPH is performing some kind of optimisation here with the amount of OSDs.

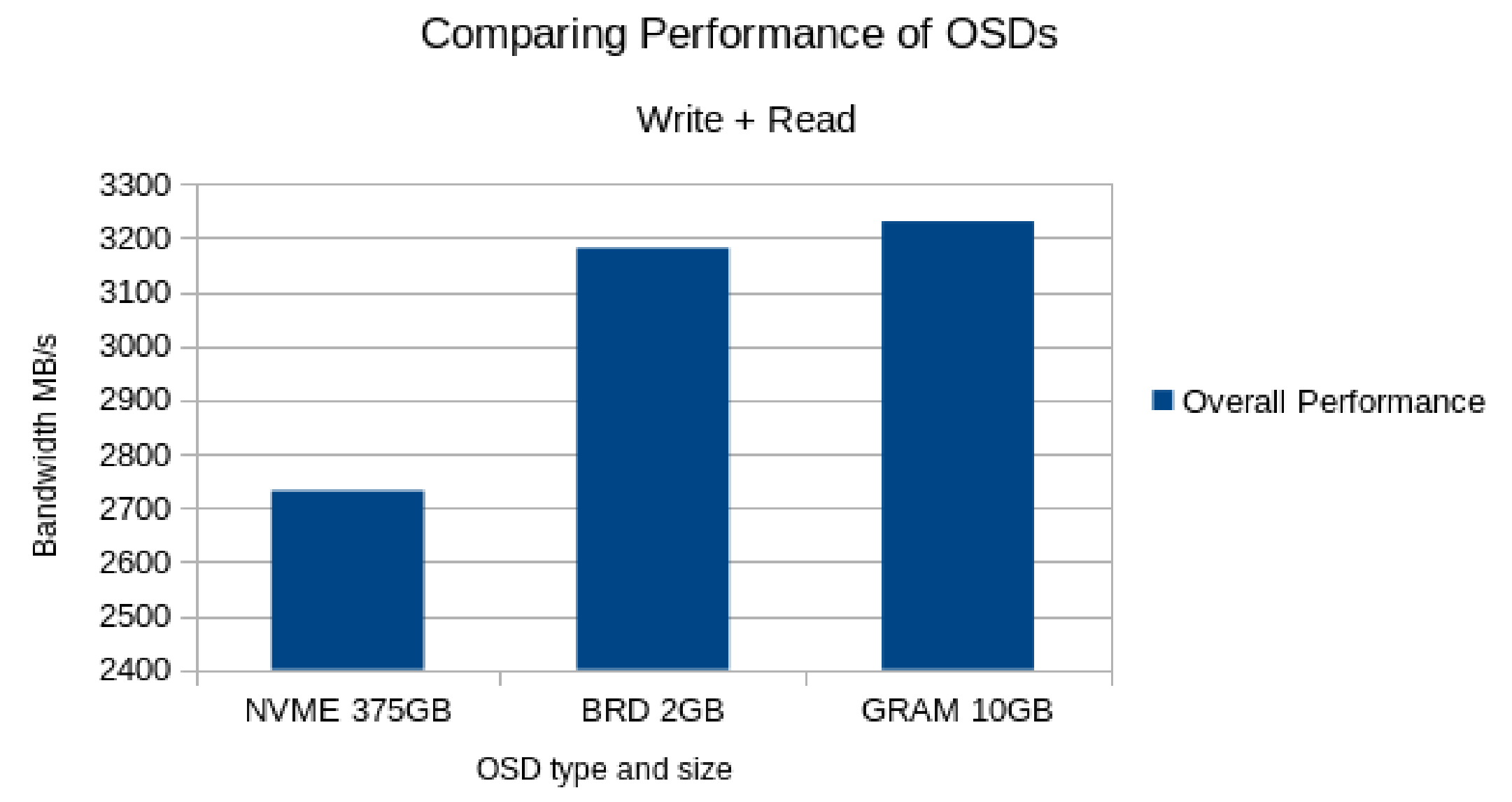

Figure 2: Comparing performance of OSDs

When comparing RAM OSDs to the NVMe OSDs it is clear to see that RAM is more performant than the NVMe’s by more than 450MB/s which was an expected result.

Conclusions

These results show that CEPH using RAM as an OSD is a viable option instead of/or in conjunction with NVMes and that CEPH has good native single-threaded performance without tuning, and we can reasonably expect a 20% increase in performance after tuning. This result is useful for Diamond as a lot of the data from the detectors comes out in single streams due to the way it is recorded and processed, technologies such as KAFKA (apache project) may help this issue as it splits a queue into many threads, removing one of our major technological hurdles. However, this single-threaded performance is still promising to see.

There is however a lot more research that needs to be done to find the optimal size of a RAM block when using CEPH as it is still not clear as to why the performance is so good on 2GB block sizes since the hardware benchmarking would suggest that this should be the least performant OSD size, initial thoughts would suggest that CEPH may have some vertical scaling aspect to it. However, this will need more exploration as it could be to do with the way the CPU is handling the RAM instead of how CEPH is handling the OSDs.

Forthcoming Research

Other areas of CEPH that will be looked into in the coming months are:

- Creating a cluster deployable CEPH instance, allowing testing at scale over Diamond’s compute cluster and provided further insight into how CEPH copes at scale.
- Different ways to optimise CEPH for single-threaded performance.
- CEPH’s performance when using CEPHFS on top of CEPH object stores as a lot of Diamond’s software is built to be run on a file system so is important to see what the performance degradation is when using CEPHFS as that will affect the viability of Diamond using CEPH.

References

[1] iperf3. https://iperf.fr/, 2017. Version, 3.1.7.

[2] Brd. Linux Kernel, 2019.

[3] Zram. Linux Kernel, 2019.

[4] intel. cosbench. https://github.com/intel-cloud/cosbench, 2016. Version 0.4.2.

[5] Lord Mark. hdparm. Version 9.43.

[6] Heinz Mauelshagen. Lvm. http://www.sourceware.org/lvm2/, 2018. Version 2.02.180.

[7] Rubin Paul, MacKenzie David, and Kemp Stuart. dd (coreutils). Version, 8.22.

Acknowledgements

I would like to thank my supervisor Dave Bond (dave.bond@diamond.ac.uk) for help and useful guidance on this project.