Evaluation of the Dorado 5100 Storage System

Hussein N. El-Harake and Colin McMurtrie
CSCS – Swiss National Supercomputing Centre
Lugano, Switzerland
Email: {hussein; colin}@cscs.ch

***Abstract*—In the first quarter of 2014 we evaluated the Oceanstor Dorado 5100 from Huawei. The system is a SAN solid-state storage connected to four IO servers over FC 8Gb/s. The purpose of this study is to evaluate the system, investigating reliability, manageability and integration of this solution in the CSCS environment. In this study, we benchmarked the throughput and IOPs during both read and write operations. Furthermore, we reconstructed multiple disks and RAID arrays and replaced a controller during high-load IO. The Dorado 5100 proved itself as a reliable solution, providing interesting IOPs in random read and random write but with limited throughput when compared to the figures announced by the manufacturer [1]. Finally we tested the system as a raw block device and when used with Lustre configured with multiple MDTs (aka DNE). The Dorado5100 storage solution met our expectations in terms of integration, performance and reliability.**

***Keywords*-**

Storage; Dorado5100; SSD; metadata; Lustre

I. Introduction

CSCS uses both Lustre and GPFS; these parallel file-systems are available as scratch space, project and home file-systems. Parallel file-systems typically separate data from metadata and hence we evaluated the Dorado5100 as an alternative solution to improve metadata performance on parallel file-systems. Huawei offer two methods to manage the system, namely via a GUI and via a command line interface (CLI). Both interfaces are complete and well documented. The GUI is based on Java and is easy to use while the CLI is one of the most completed interfaces we have tested, featuring full SLES 11 Linux with ssh access making it easy for scripting.

The Dorado5100 (shown in Figure 1) is designed specifically to hold SSDs only which made the solution an optimal choice from an architecture point of view. Alternative solutions must make allowance for spindle disks and in some cases that might create a performance penalty, particularly if SSDs and spindle disks are housed in the same device. The Dorado5100 solution offers the traditional RAID technologies including RAID5 and RAID10, both of which we used in our evaluation. Note that RAID6 is not supported. We used two scenarios in creating RAID arrays, based on 6 or 12 drives; in both cases we had an optimal number of RAID arrays mapped to our 4 IO servers.

Each controller has 48 GB of cache making a total of 96GB for the controller pair. It is possible to configure the memory cache as follows:

Figure 1. Photograph showing the Oceanstor Dorado 5100 from Huawei.

- Write-back with Mirroring: Caching with mirroring between the controllers;
- Write-back without Mirroring: Caching without mirroring between the controllers, not recommended, cached data will be lost in case of controller failure;
- Write-through: Bypass any caching and use the SSD disks directly.

We will discuss later in this document different scenarios in which we artificially created a critical situation by removing disks and even a controller during high-load tests, simulating disk or controller failures in production. The scenarios included pulling up to 8 disks at the same time (one disk per RAID array), pulled one controller and finally starting a parallel rebuild (with a maximum of 4 disks in parallel).

II. System Configuration

CSCS received a dual controller Dorado5100 with 8 dual SAS interfaces to connect the controllers to the enclosures and 4 enclosures, every enclosure containing 24 x 100GB SSD SLC disks. A total of 4.8 TB of usable space was available. In addition 4 IO servers were connected to the dual controller using eight FC 8Gb ports (see Figure 2 for an overview of the IO server specifications).

From the SPC benchmark report [2], the Dorado5100 delivered 600K IOPs which represents the maximum I/O request throughput at the 100% at load point. Based on the Dorado datasheet the peak performance is 1 million IOPs and 12GB/s. UltraPath is the IO server redundancy

Processor	Memory	Network
2 * Intel Xeon E5-2670	64GB	Dual port 8Gb FC Dual port 54Gb/s FDR

Figure 2. Hardware configuration of the IO servers.

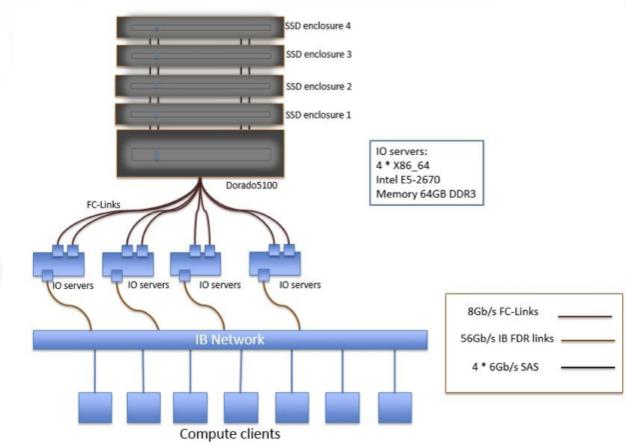

Figure 3. Schematic showing the test environment used for the FIO testing including storage, IO servers and compute clients.

software supported by Huawei, but we used the standard Linux multipath software which worked perfectly well.

III. Evaluation Method

We used the tool FIO [3] for the throughput and IOPs measurements. FIO simulates database load in sequential and random access and was used on raw devices; no filesystem was available at this time. Figure 3 shows a schematic of the test environment used for the FIO testing.

Later we deployed the Lustre filesystem [4] and used the Dorado5100 for metadata, mdtest was the benchmark tool to run on multiple MDSs with multiple MDTs which is referred to as Distributed Namespace (DNE). DNE is a new feature available in Lustre 2.4.x and above. We used 12 nodes, each having dual sockets, as a test HPC cluster. Hyper-threading and Turbo Boost were enabled on the 12 node cluster and this allowed 3.3 GHz maximum core frequency. We configured the system with RAID10 for the IOPs tests and RAID5 for the throughput tests. The management tool displays all relevant system information like capacity, RAID utilization and performance, etc. In the Lustre tests we used the three IO servers connected to the Huawei as MDSs and we added two servers as OSSs. Figure 5 shows a schematic of the test environment used for the Lustre testing.

IV. FIO Results

We decided to select 4 IO servers due to the number of disks available within every enclosure. Huawei informed us that 6 or 12 disks are the best performance choice for building RAID arrays on their system, a fact that we were able to later confirmed. We had a total of 96 disks, 8 RAID10 arrays each containing 12 disks. That changed to 24 RAIDS when we tested Lustre.

The results of the IOPs test are shown in Figure 4. We note that the measured 600K in sequential read is similar to what SPC measured [2]. Storage solutions based on SSDs have improved the gap between sequential and random access but we can see in our results that the problem is not completely solved. The random write results show interesting behavior in the three modes of operation (writeback with mirroring, writeback without mirroring and write through).

Our intention was to run only IOPs test, but we decided that might be interesting to share bandwidth numbers as well. The bandwidth results shown in Figure 6 appeared to be a bit low so we contacted Huawei for an explanation. It turned out that two of our disks enclosures are still using SAS 1, 3Gbit/s instead of SAS 2, 6Gbit/s interfaces. The net result was that the system had unbalanced RAID arrays and this directly impacted the throughput bandwidth. Huawei claimed to get better numbers running our code with the same flags on their site system, which had fully balanced RAID arrays, but we were unable to reproduce these results. We did however see benefit in bypassing caching in the case of Random Writes since our results showed 3.3GB/s in Write Through performance verses 2.1GB/s in the case of Writeback with Mirroring. Caching also boosted the sequential access by as much as 35% especially in the case of Writeback without Mirroring.

V. `mdtest` Testing of a Lustre Filesystem with the Dorado5100 as MetaData Storage

The widely used `mdtest` [5] measures the performance of multiple tasks undertaking create, stat and delete operations on files and directories. It is an MPI code which allows us to run multiple processes in parallel. Each node ran between 1 and 32 threads with each thread operating on 30K files or directories. Figures 7 and 8 shows the performance numbers for create, stat and remove operations on directories and files, respectively. We used a single metadata server (MDS) and scaled up to 8 metadata targets (MDTs). We observed scaling in creating files, 30K one MDT to 90K on 8 MDTs, which is not linear but nonetheless interesting.

Dorado RAW IOPS Results				
Tool	Block Size	Read		
FIO	4K	Random Read	480 K	
		Sequential Read	612 K	
		Write		
			Writeback with Mirror	Writeback no Mirror	Write Through
		Random Write	320K	430K	300K
		Sequential Write	400K	680K	315K

Figure 4. FIO IOPs results running on 4 servers - 128 threads (32 threads per server).

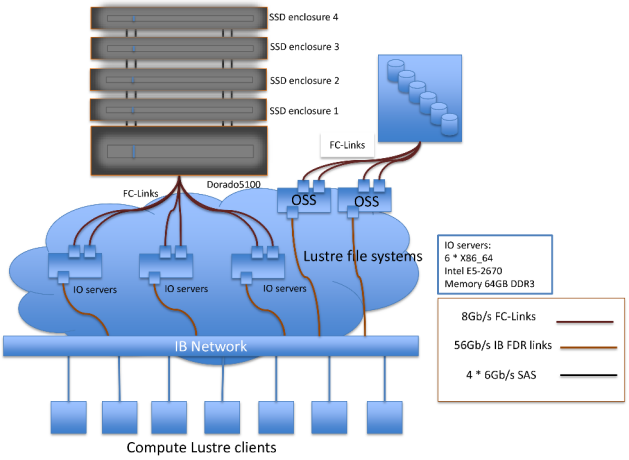

Figure 5. Schematic showing the test environment used for the Lustre testing including IO servers, Dorado5100 as metadata storage, some additional data storage and compute clients.

Dorado RAW Throughput Results				
Tool	Block Size	Read		
FIO	1M	Random Read	4300 MB/s	
		Sequential Read	4700 MB/s	
		write		
			Writeback with Mirror	Writeback no Mirror	Write Through
		Random Write	2100 MB/s	2100 MB/s	3375 MB/s
		Sequential Write	3500 MB/s	4700MB/s	3425 MB/s

Figure 6. FIO Bandwidth results running on 4 servers - 128 threads (32 threads per server).

Figures 9 and 10 show the performance figures for create, remove and stat operations on directories and files, respectively, using 2 MDSs instead of one, and from 2 to 16 MDTs. Scalability is better, and numbers were boosted by between 60% and 80%. We observed a linear speedup in create operations for directories and files up to 16 MDTs, with notably a maximum of 150K for file removal. In the case of directory removal and file creation there

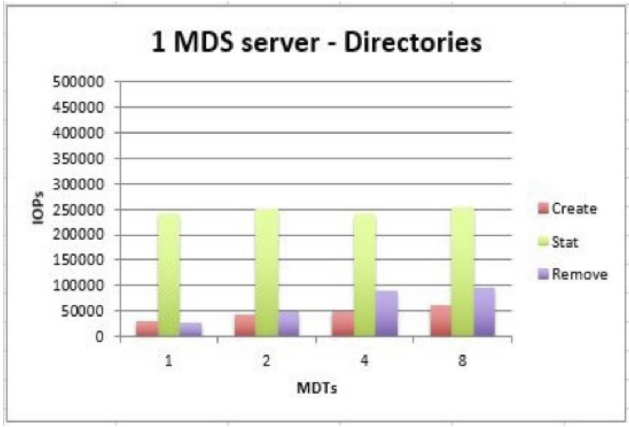

Figure 7. `mdtest` results for directory operations using a single metadata server (MDS) with 1 to 8 metadata targets (MDTs).

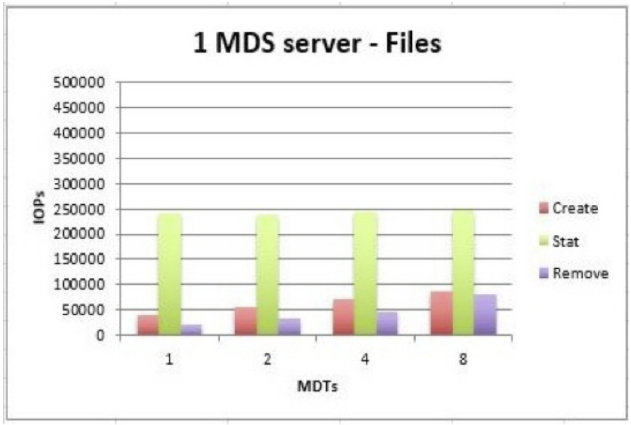

Figure 8. `mdtest` results for file operations using a single metadata server (MDS) with 1 to 8 metadata targets (MDTs).

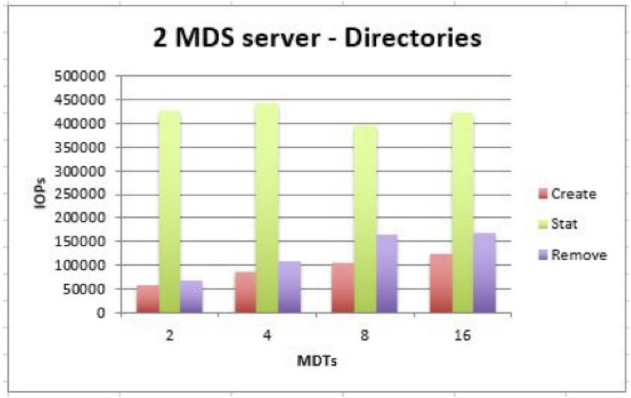

Figure 9. `mdtest` results for directory operations using a 2 metadata servers (MDSs) having from 2 to 16 metadata targets (MDTs).

is a linear speedup up to 8 MDTs at which point the performance flattens off. File stat operations show little change and remain at almost a constant 400K, essentially twice the performance when compared to the single MDS case.

We added one more server with 8 more MDTs for a total of 24 MDTs and ran the tests again. As can be

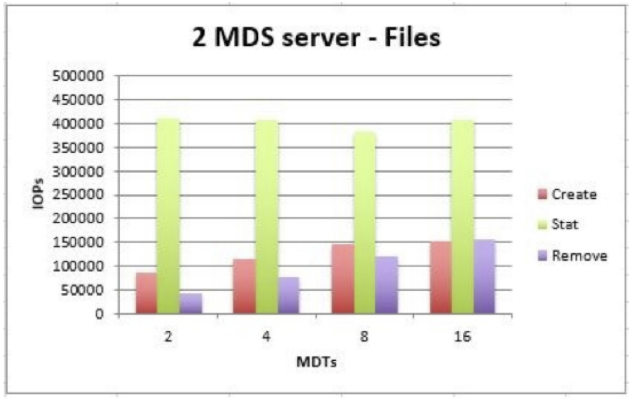

Figure 10. `mdtest` results for files operations using a 2 metadata servers (MDSs) having from 2 to 16 metadata targets (MDTs).

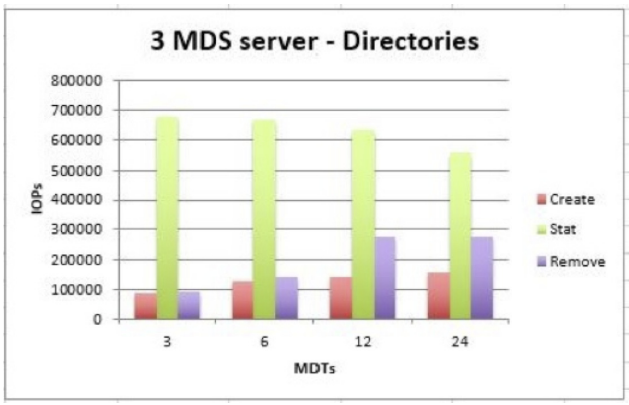

Figure 11. `mdtest` results for directory operations using a 3 metadata servers (MDSs) having from 3 to 24 metadata targets (MDTs), i.e. 1 to 6 director blades.

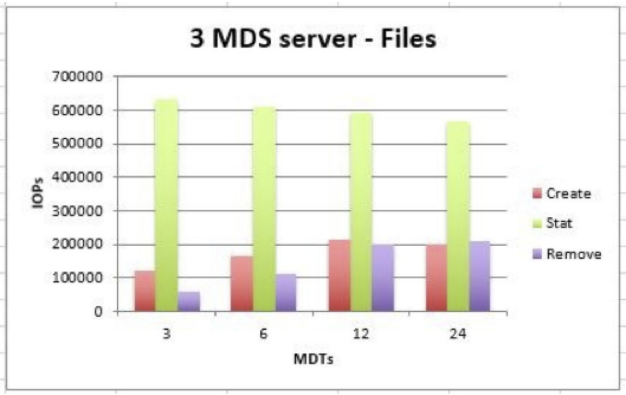

Figure 12. `mdtest` results for file operations using a 3 metadata servers (MDSs) having from 3 to 24 metadata targets (MDTs), i.e. 1 to 6 director blades.

seen in Figures 11 and 12 results continue to scale. We exceeded 200K in create and remove file operations. Create directories hit almost 150K while remove directories reached ~280K. Files and directories stat showed very linear scaling hitting ~600K IOPs. We stopped at 3 MDSs since we had no more MDTs to add; Dorado5100 had only 24 RAID arrays.

If we compare the results of the Lustre filesystem and raw tests we see lower performance in Lustre, which is to be expected. Multiple MDSs with multiple MDTs (aka DNE) is a relatively new feature in Lustre and we were in fact testing the first GA release. We are expecting DNE 2 in the future and the will be the subject of further testing and evaluation.

VI. Resiliency tests

We simulated different failures starting by removing one disk and rebuilding it; we noticed a drop in FIO performance for a couple of seconds, but there was appreciable impact to the overall performance. We repeated the same tests by removing 2, 4 and 8 disks (we removed one disk per RAID array; see Figure 13). In all cases the behaviour was the same in the sense that FIO performance dropped for a couple of seconds and but continued to run

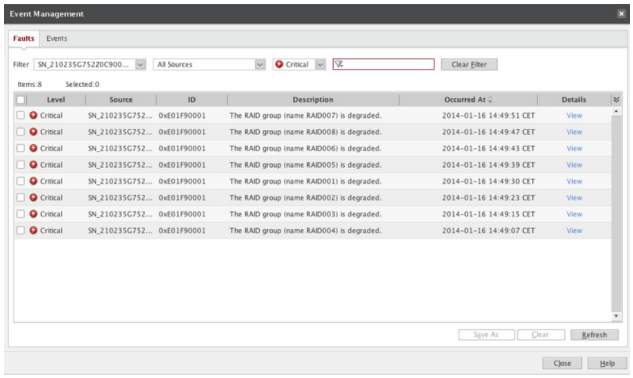

Figure 13. Simulating the failure of 8 disks in 8 RAID arrays.

without any apparent performance degradation. We started the rebuild of the 8 disks with only 4 being rebuilt at a time (see Figure 14). It took 40 minutes to complete the rebuild process with no noticeable performance degradation during this time. Furthermore it took the same amount of time to rebuild 1 to 4 drives. Finally, we removed one controller while running the FIO test; the performance dropped by 50%. Reinserting the controller caused the system to recover and the figures were back to 100%.

VII. Conclusion

The Dorado5100 SSD-based storage is a complete solution for the metadata of parallel file systems in High Performance Computing (HPC) environments and would also be very effective in an environment where small random or sequential access is required, as for example in the case of a database. Our test bed system achieved an aggregate of 600K IOPs in read and exceeded 600K in sequential write. Write throughput of 4.7GB/s was achieved in sequential read and write.

Moreover the Dorado5100 system has high resiliency features; the hardware is hot-swappable and is very stable. It is very easy to manage and has very powerful scripting capability (via the CLI) which will aid in environments that must manage multiple systems. It was interesting to see how the system handled the simulated disk and

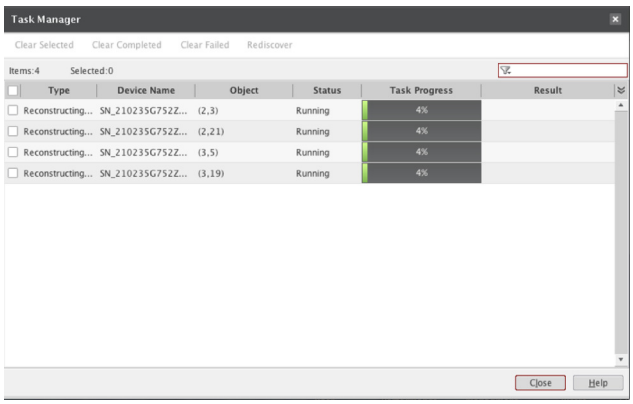

Figure 14. Rebuilding 4 disks at a time.

controller failures but it would be good to verify the behaviour when rebuilding larger disks. Furthermore the Huawei support was very helpful and ready at any moment to handle any open issue. The GUI could be improved in different places such as monitoring the performance, where headers could be added to enable better understanding of the graphs.

REFERENCES

[1] "Huawei Dorado 5100 Datasheet." [Online]. Available: http://enterprise.huawei.com/ilink/enenterprise/download/HW_143886

[2] "Storage Performance Council Report on the Huawei Oceanstor Dorado5100." [Online]. Available: http://www.storageperformance.org/benchmark_results_files/SPC-1/Huawei/A00119_Huawei_Dorado5100a00119_Huawei_Dorado5100_SPC-1_executive-summary.pdf

[3] "fio Benchmark." [Online]. Available: http://freecode.com/projects/fio

[4] "The Lustre File System." [Online]. Available: http://opensfs.org/lustre/

[5] "The mdtest MPI Metadata Benchmark." [Online]. Available: http://sourceforge.net/projects/mdtest/