Performance Evaluation of Intel® SSD-Based Lustre* Cluster File Systems at the Intel® CRT-DC

Written by Michael Hebenstreit

Contact: michael.hebenstreit@intel.com

Contents

Executive Summary

This article compares the performance of the Intel® Customer Response Team Data Center's Lustre* systems in a multi-node test using up to 128 clients.

The Intel Customer Response Team Data Center (called CRT-DC), located in Albuquerque, New Mexico, runs a benchmarking data center with >500 compute nodes. The cluster, known as *Endeavor* is rebuilt on a regular basis with the latest hardware and has been listed in Top 500 SuperComputer Sites since 2006. To satisfy the storage needs, two commercial clustered file systems from Panasas* and DDN* are currently in use. The Panasas system is used as a long term data repository, the DDN system employing Lustre* serves as high speed scratch space. To address the increased need for volatile storage, a new Lustre system has been built in-house from commercial-off-the-shelf (COTS) hardware. Testing has been conducted to assess the performance of this new system in a multi-node test.

Performance Tests Conducted

We were limited in time and resource usage in conducting the performance tests, as both file system and compute nodes are in heavy use on a daily basis. Therefore, some tests could not be repeated as often as we would have liked, and other applicable tests could not be conducted at all.

The tests used were:

(a) *Iozone**: standard file system test (www.iozone.org) available on all Unix* platforms. This test runs a series of subtests and includes a cluster mode.

(b) *IOR**: standard file system test (http://sourceforge.net/projects/ior-sio/) available on all Unix* platforms. This test runs a cluster wide read/write test.

Reader comments and feedback regarding this article and the tests conducted are welcome and very much appreciated.

Background on Lustre

Lustre is a high performance Cluster File System. In contrast to more widely used SMB or NFS servers, Lustre differentiates between servers to store data and systems responsible for Metadata (like the file names). This separation allows it to scale independently the bandwidth and the storage capacity in a file system, as opposed to keeping all information on a single system. The basic layout of the Lustre systems used at the CRT-DC is summarized as follows:

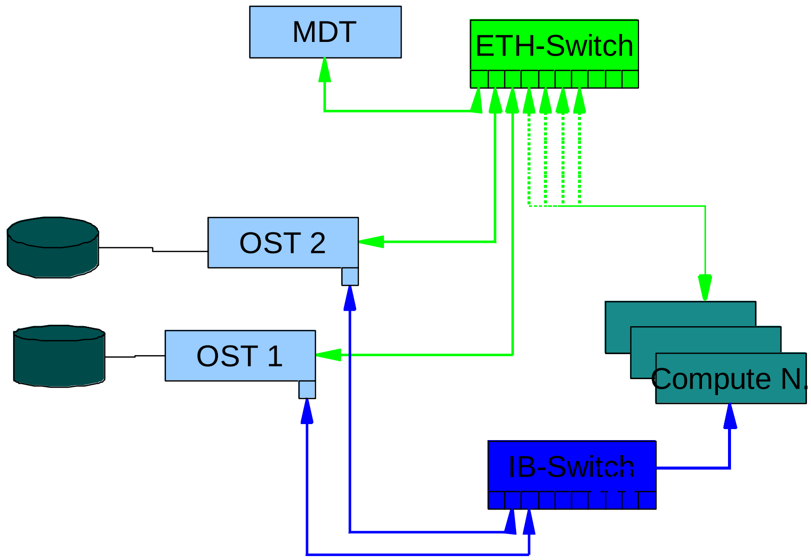

Figure 1: basic Lustre layout

All metadata information (for the Lustre experts—both MDT and MDS information) is stored on a single server called MDT. This MDT server was based on a simple two-socket server equipped with a couple of hard drives. One disk was used for the operating system; the others were coupled together into a software-based RAID configuration to hold the Lustre-specific MDS and MDT files. We found the load and memory consumption on this system to be very low.

We also used eight servers as an Object Storage Target (OST) to store the actual data. Each OST is equipped with three RAID controllers handling eight SSD drives each:

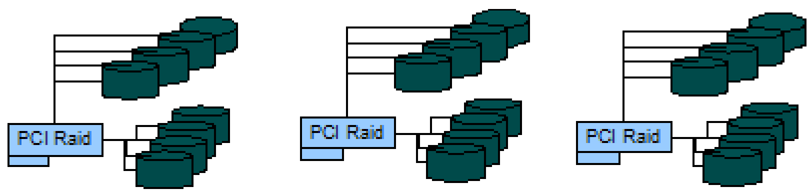

Figure 2: raid to disk setup

These OST servers were equipped with modern two-socket Intel® Xeon® E5-2680 @ 2.70GHz CPUs, offering both high integer performance and memory bandwidth. The 64GB memory onboard server will provide additional caching (in essence we have 512GB of caching available across the eight OSSs).

The systems are equipped with three RAID adapters (type *LSI Logic / Symbios Logic MegaRAID SAS 2208)*. Each adapter has eight ports and combines four SSD drives into each logical drive. This logical device appears as a singular SCSI device to the operating system and is handled as a separate OST by the Lustre file system. Therefore, we have available 24 SSDs per OSS, 6 OSTs per OSS, and 48 OSTs in total.

When an application tries to write a file, it informs the MTD. The user can decide how many OSTs should be used for each and every file. Based on system- and user-dependent configuration, the MDT decides where to place the data and informs the Lustre client software which OSTs to use. From that point on, communication is mostly between the compute node and the OSTs. Therefore, the MDT is NOT subject to heavy load.

Additionally, as communication between compute nodes and OST is conducted via InfiniBand, the data exchange has both low latency and high bandwidth.

Striping

Some applications, especially those with only one thread doing IO, will benefit from striping the information over several OSTs. Alternatively, application writing in parallel to multiple files will benefit if no operation collides with another, and therefore striping might not be beneficial at all.

There is no general rule as to which striping strategy works best, so for each case, users may have to find the solution that works for their needs. In our case, the stripe count was set to 1.

Multi-Node Test with Iozone

A big problem with file system tests is the local memory on the client nodes. In our case, the memory of 64 GB would essentially force us to set a test size of 128 GB "per node." Using a smaller size lets the system run much of the test out of the file system cache, but our time to perform those tests was limited, so we were forced to use a different strategy.

A small test program would start on all nodes, lock 95% of the available memory, and then go to sleep, consuming almost no CPU cycles at all. This situation is similar to a real life situation in which a HPC program consumes memory and does not leave the OS much memory for file system caching.

Then Iozone was started on all nodes in parallel:

```
iozone.x86_64 -s 16G -r 1m -t $COUNT -+m $NODEFILE
```

with COUNT going from 1 to 128 nodes, with a single Iozone process running on each node. Results from all nodes are automatically aggregated, with the following result (all values in MB/s):

	Test/Number of Nodes =>	1	2	4	8	16	32	64	96	128
SDD	initial_writers	232	445	473	1070	2523	4953	9644	15832	20649
HDD	initial_writers	105	223	404	932	2077	3839	3790	3805	3768
SDD	rewriters	334	475	418	1528	2923	7057	12442	18542	24313
HDD	rewriters	152	433	516	1139	2293	3674	3825	3824	3814
SDD	readers	663	1179	2255	4123	8518	16824	31129	43369	44087
HDD	readers	513	923	1733	3434	6580	8454	2406	2164	1977
SDD	re-readers	635	1179	2214	4211	8515	16085	30380	43405	44617
HDD	re-readers	469	827	1762	3425	6728	10849	2837	2127	1960
SDD	reverse_readers	534	934	1810	3411	6323	9793	15569	21841	28691
HDD	reverse_readers	408	766	1556	3151	5829	2564	1427	1605	1560
SDD	stride_readers	620	1013	1910	3780	7098	12541	20338	26707	32631
HDD	stride_readers	391	796	1604	3170	5898	5780	2385	1851	1654
SDD	pread_readers	692	1140	2272	4054	8490	15812	30113	42337	42199
HDD	pread_readers	438	871	1746	3432	5359	4533	2726	2305	2144
SDD	pwrite_writers	265	392	608	1105	2261	4886	9707	15882	20046
HDD	pwrite_writers	107	231	433	974	1990	3836	3788	3807	3784
SDD	random_readers	471	900	1696	3458	6552	10673	15826	21478	26120
HDD	random_readers	384	768	1552	3103	5636	4740	1927	1652	1459
SDD	random_writers	330	598	1100	1810	3618	7592	14337	21597	29007
HDD	random_writers	146	426	666	1359	2975	3719	3836	3825	3611
SDD	mixed_workload	484	702	1309	2462	5404	9497	15840	22289	27907
HDD	mixed_workload	392	462	1063	2454	4315	5047	4107	2674	2171
SDD	fwriters	363	516	961	1805	3533	7738	13401	21688	29276
HDD	fwriters	98	271	440	982	2746	4165	4018	3931	3894
SDD	freaders	713	1112	2276	4216	8616	16231	30582	44171	44847
HDD	freaders	442	828	1769	3514	6720	10225	2553	2445	2231

Table 1: results from Iozone tests

Multi-Node Test with IOR on Up to 16 Nodes

Because IOR does not test as many subcases as Iozone does, it was not necessary to do anything other than maintain a standard file size of 128 GB per node. In a first test, up to 128 threads were used on 16 clients to assess the behavior when multiple threads were doing I/O on the same node. The command line executed was:

```
mpirun … ~/IOR/src/C/IOR -a MPIIO -r -w -F -i 3 -C -t 1m -b ${CS}g -o
./IOR
```

`$CS` was adopted to ensure 128GB were written per node. If multiple processes were running on a single node, `$CS` was modified accordingly. For example with two threads per node, each thread would work on 64 GB. This gave the following results (in MB/s):

		Read		Write	
Threads/Node	**Threads**	**lfs08 HDD**	**lfs09 SSD**	**lfs08 HDD**	**lfs09 SSD**
1	16	1760	5500	3552	5459
2	32	1938	9816	3584	10162
4	64	2026	17228	3602	19111
8	96	1860	21956	3722	24199
16	128	1803	24140	3733	24673

Table 2: IOR results on up to 16 nodes and up to 8 threads/node

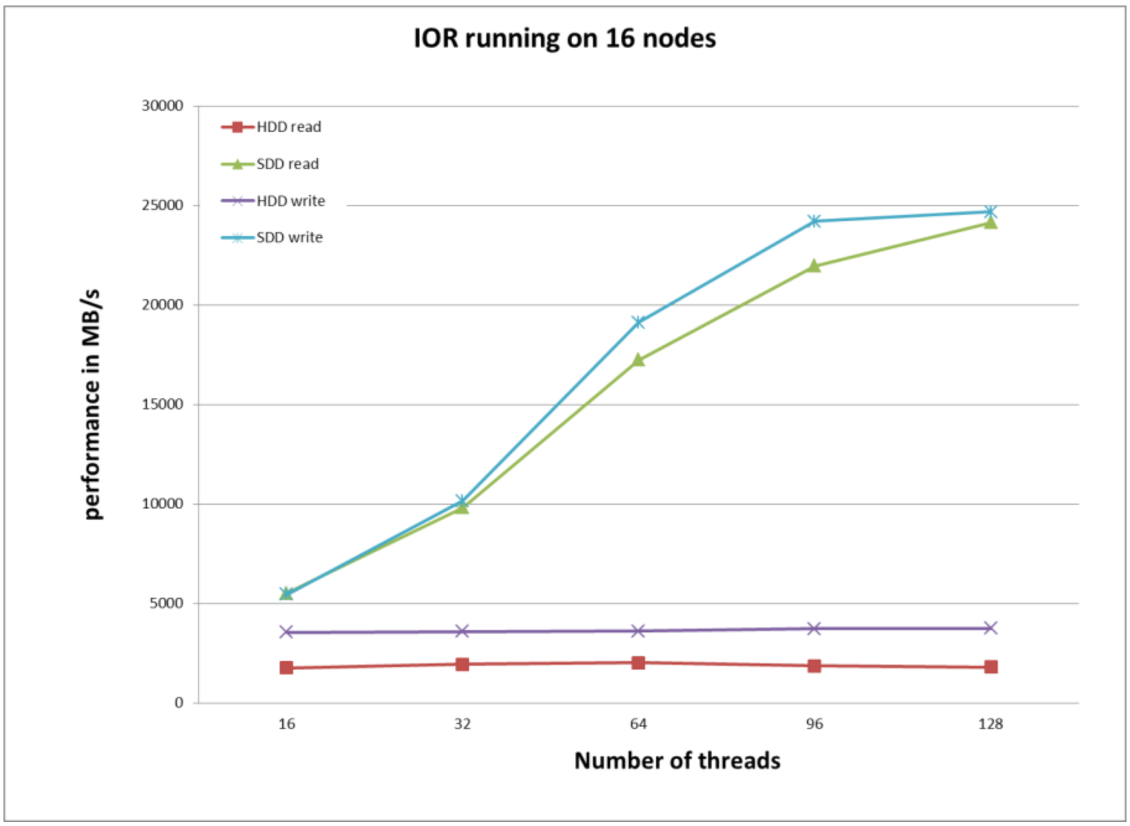

Figure 3: results from IOR tests on 16 nodes

Multi-Node Test with IOR on Up to 128 Nodes

Because IOR does not test as many subcases as Iozone does, it was not necessary to do anything other than maintain a standard file size of 128 GB per node. In a second step, up to 128 nodes were used with only a single process per node. The command line executed was:

```
mpirun … ~/IOR/src/C/IOR -a MPIIO -r -w -F -i 3 -C -t 1m -b 128g -o
./IOR
```

This produced the following results (in MB/s):

		Read		Write	
Threads/Node	**Threads**	**lfs08 HDD**	**lfs09 SSD**	**lfs08 HDD**	**lfs09 SSD**
1	1	421	369	435	368
1	2	763	704	792	712
1	4	1136	1456	1623	1432
1	8	1662	2795	3128	2815
1	16	1760	5500	3552	5459
1	32	1730	10112	3685	10267
1	64	1416	18513	3695	19178
1	96		24904		24707
1	128		27036		26152

Table 3: IOR results on up to 12 nodes and 1 thread/node

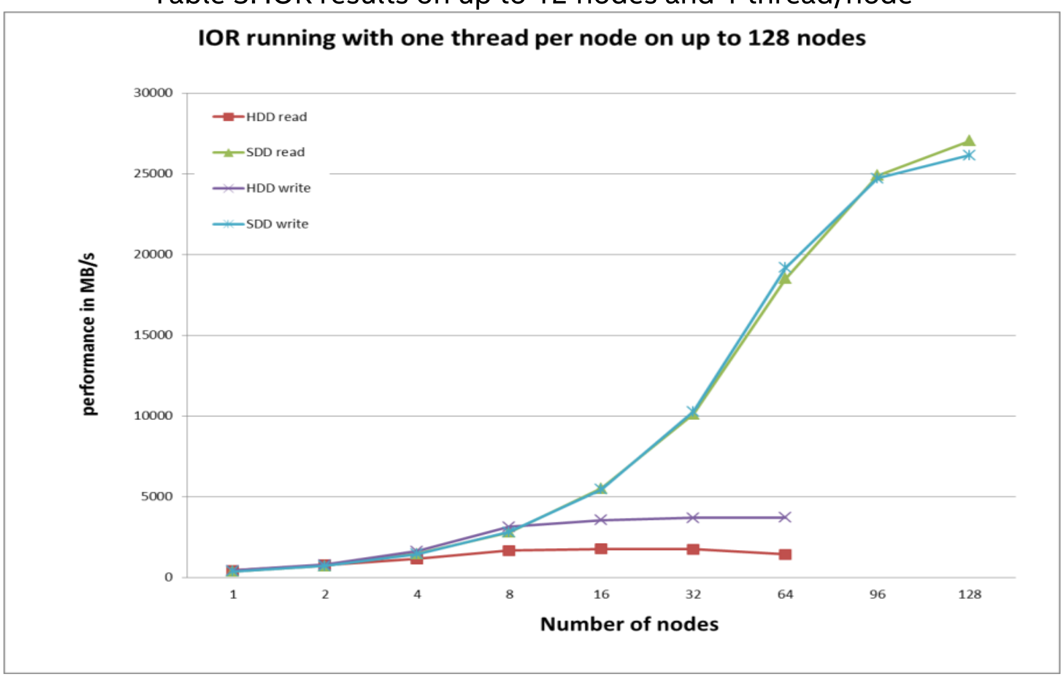

Figure 4: results from IOR tests on up to 128 nodes (old Lustre driver)

Note: For a limited time, no data could be gathered on lfs08 for 96 and 128 nodes.

IOR Tests on New Lustre Client

As single node/single thread performance was rather unsatisfying, the author replaced the client side Lustre packages with a development version 2.5.55 and repeated the IOR tests. This time, the performance was measured on both MPIIO and POSIX interfaces with a varied record size from 64kB to 4GB. The command line was as follows:

```
IOR -a POSIX|MPIIO -r -w -F -i 3 -C -t RECORD -b SIZE -o PATH
```

`SIZE` was adjusted so that each node wrote 128 GB in each test. Thus, if a node was running four threads, each one would read/write 128/4 or 32GB.

Result: Although single thread performance increased up to 800 MB/s, the overall picture did not change.

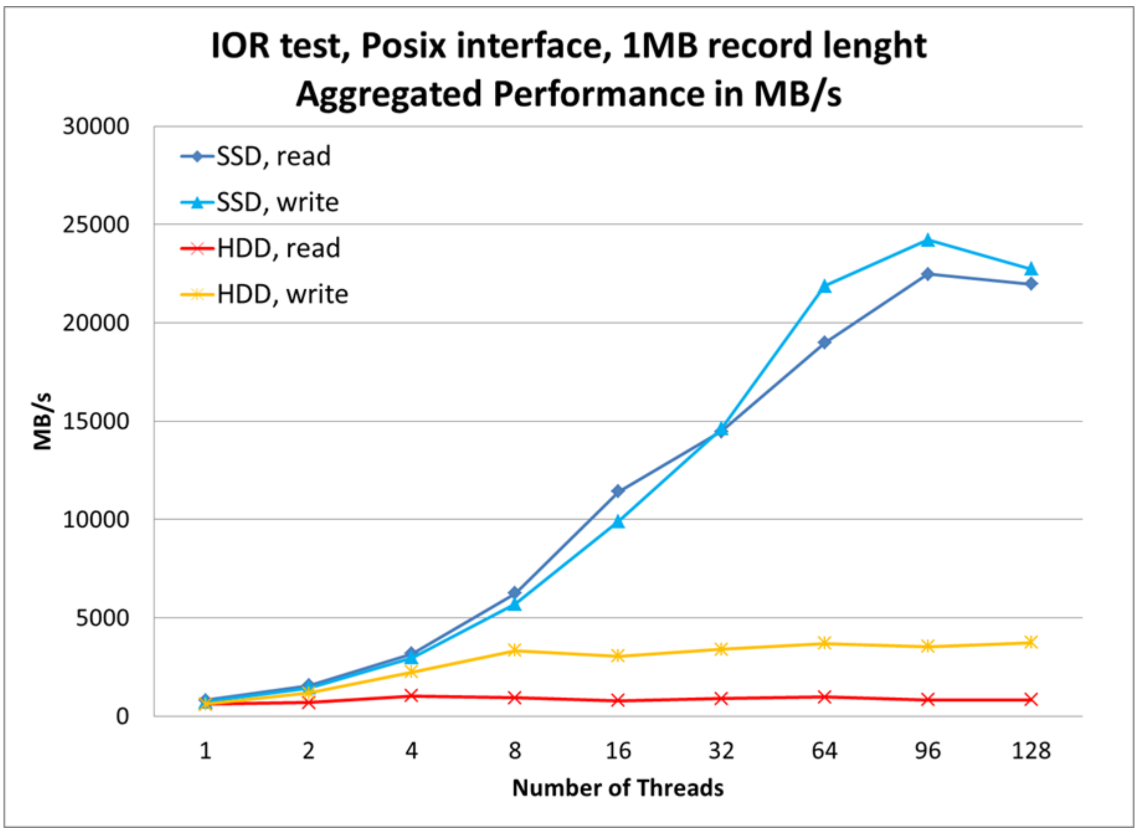

Figure 5: results from IOR tests on up to 128 nodes (new Lustre driver)

The HDD-based system leveled out at around 3.5B/s write and 1GB/s read performance, while for the SSD-based solution, both reads and writes could reach well over 22GB/s. Detailed results are found below.

Performance in MB/s over Record Lengths for the MPIIO Interface

		MPIIO									
		read					**write**				
	#threads/record length =>	1g	4m	1m	256k	64k	1g	4m	1m	256k	64k
lfs09	1	747	797	610	336	115	717	761	611	334	114
	2	1555	1509	1170	670	241	1467	1389	1198	677	238
	4	2895	3110	2256	1314	441	2908	2894	2289	1313	440
	8	5941	5774	4724	2454	819	5595	5562	4627	2462	824
	16	11364	11724	8920	4712	1499	10832	10093	9031	4720	1496
	32	15255	12868	13619	7259	1851	18809	16471	14223	7417	1834
	64	18743	17841	16652	7437	1751	21490	20973	20555	6849	1727
	96	19787	18165	17898	6912	1842	23732	22282	22448	6971	1827
	128	20067	20747	19162	6780	1838	28040	26288	22950	7081	1827
lfs08	1	817	765	746	791	680	805	737	774	755	627
	2	810	1322	848	699	1044	1597	1547	1467	1474	1267
	4	1090	999	1042	891	1106	2818	2887	2714	2701	2410
	8	927	971	924	1036	1063	2597	3303	3086	3757	3207
	16	1312	945	956	1107	919	3560	3688	3359	3670	2930
	32	962	1009	906	1066	1080	3295	3786	3436	3824	3622
	64	1093	935	1019	941	944	3867	3741	3803	3609	3629
	96	902	966	934	1017	943	3699	3757	3571	3889	3780
	128	890	945	819	938	845	3708	3869	3716	3966	3539

Table 4: complete result IOR test with MPIIO interface

Performance in Operations/s over Record Lengths with MPIIO

	MPIIO									
	read					**write**				
Row Labels	1g	4m	1m	256k	64k	1g	4m	1m	256k	64k
1	1	199	610	1344	1844	1	190	611	1336	1820
2	2	377	1170	2679	3859	1	347	1198	2707	3812
4	3	777	2256	5255	7056	3	723	2289	5252	7041
8	6	1444	4724	9816	13109	5	1391	4627	9848	13181
16	11	2931	8920	18847	23976	11	2523	9031	18881	23942
32	15	3217	13619	29034	29619	18	4118	14223	29669	29342
64	18	4460	16652	29750	28008	21	5243	20555	27396	27632
96	19	4541	17898	27650	29478	23	5571	22448	27883	29231
128	20	5187	19162	27119	29410	27	6572	22950	28324	29239
1	1	191	746	3163	10887	1	184	774	3019	10025
2	1	330	848	2798	16706	2	387	1467	5896	20266
4	1	250	1042	3565	17694	3	722	2714	10805	38566
8	1	243	924	4145	17005	3	826	3086	15030	51304
16	1	236	956	4428	14701	3	922	3359	14679	46884
32	1	252	906	4264	17283	3	947	3436	15296	57949
64	1	234	1019	3763	15098	4	935	3803	14436	58069
96	1	242	934	4069	15081	4	939	3571	15557	60487
128	1	236	819	3751	13518	4	967	3716	15866	56618

Table 5: complete IPOs result IOR test with MPIIO interface

Performance in MB/s over Record Lengths for POSIX Interface

	POSIX									
	read					write				
#threads/record length =>	1g	4m	1m	256k	64k	1g	4m	1m	256k	64k
1	751	810	811	794	663	693	719	736	710	577
2	1551	1608	1560	1529	1379	1467	1464	1431	1396	1212
4	2922	3141	3169	3034	2598	2801	2846	2971	2730	2368
8	5677	6341	6251	5935	5304	5214	5794	5699	5483	4742
16	10934	11306	11422	11472	10185	9935	10151	9909	10894	8736
32	14625	14848	14480	14640	18440	17144	15614	14617	15542	16576
64	19823	19988	18973	17432	17557	20107	22605	21870	18795	20779
96	19924	23598	22475	22779	21674	20416	26049	24220	24455	23502
128	23664	23754	21979	22679	24436	25701	24623	22737	24391	27079
1	667	673	620	437	178	749	650	608	406	174
2	1020	698	696	817	345	1492	1331	1167	827	345
4	891	970	1041	1061	642	2459	2543	2252	1566	645
8	1077	961	933	992	991	3828	3444	3345	2792	1211
16	947	993	792	952	949	3406	3859	3054	3494	2214
32	1002	1043	899	1031	960	3562	3581	3401	3546	2410
64	964	911	981	879	939	3768	3549	3693	3586	2398
96	846	840	842	853	970	3741	3615	3532	3684	2276
128	868	887	831	838	973	3581	3660	3736	3509	2225

Table 6: complete result IOR test with Posix interface

Performance in Operations/s over Record Lengths for POSIX

	POSIX									
	read					write				
Row Labels	1g	4m	1m	256k	64k	1g	4m	1m	256k	64k
1	1	202	811	3175	10607	1	180	736	2839	9232
2	2	402	1560	6116	22058	1	366	1431	5582	19388
4	3	785	3169	12135	41570	3	711	2971	10919	37893
8	6	1585	6251	23740	84857	5	1448	5699	21934	75865
16	11	2827	11422	45888	162953	10	2538	9909	43578	139772
32	14	3712	14480	58558	295034	17	3903	14617	62170	265222
64	19	4997	18973	69730	280912	20	5651	21870	75180	332464
96	19	5899	22475	91115	346792	20	6512	24220	97820	376034
128	23	5939	21979	90716	390981	25	6156	22737	97565	433270
1	1	168	620	1749	2850	1	162	608	1622	2790
2	1	174	696	3269	5521	1	333	1167	3306	5519
4	1	243	1041	4242	10278	2	636	2252	6264	10317
8	1	240	933	3967	15854	4	861	3345	11168	19383
16	1	248	792	3809	15191	3	965	3054	13975	35431
32	1	261	899	4124	15355	3	895	3401	14182	38565
64	1	228	981	3516	15024	4	887	3693	14346	38365
96	1	210	842	3412	15525	4	904	3532	14735	36421
128	1	222	831	3353	15560	4	915	3736	14037	35607

Table 7: complete IOPs result IOR test with Posix interface

Discussion and Summary

The CRT-DC in Albuquerque compared high performance cluster file systems based on off-the-shelf servers with commercial solutions.

- With current technology, it is very easy to create a Cluster File System that can provide more bandwidth over FDR InfiniBand than a single two-processor client typically found in modern clusters can use.
- From the data, the performance of single clients is mostly defined by the client itself and not so much by the differences of the file system. Both lfs08 and lfs09 can easily handle a few clients in parallel.
- The differences appear in scaling. While the SSD-based solution can deliver over 40 GB/s aggregated bandwidth, the HDD-based solution in its best case showed about 10GB/s. More problematic, though, is that with the HDD-based solution, the performance does not stay level at peak with an increase of clients, but significantly decreases as more and more clients are added.
- Performance-wise, home grown solutions based on “off the shelf” Intel servers can easily compete with commercial solutions. In the overall TCO calculations soft factors like support and uptime therefore become important parameters.
- A solution based on Intel SSDs can outperform far more expensive solutions based on standard HDDs. The challenge shifts here to find software able to use this system to full advantage.
- With the new SSD based system the CRT-DC team is certain to handle the upcoming challenges in the next year.

Description of Hardware and Software

Server Hardware LFS08

Back end:

DDN SFA10000 Couplet full speed
5 x 60-slot drive enclosures
240 x 300 GB 15K 3.5” SAS drive
16 LUNs each 10 x Hitachi 15K300 SAS drives per LUN x 16 LUNS.
2 LUNS per Storage server connected via SRP/IB

Meta Data Server (MDS) – 1:

Intel® SR1560SF server
Intel® R2208GL4GS G29051-352 Grizzly Pass board
2 x Intel® Xeon® CPU E5-2680 @ 2.70GHz
64 GB total/node (8*8GB 1600MHz* Reg ECC DDR3)
Bios SE5C600.86B.01.03.0002.062020121504
1 OS hard drive (Intel® 320 Series SSDs Model:SSDSA2CW600G3)
MDT 2 x Seagate Constellation (SATA) ST9500530NS RAID0
InfiniBand HCA Mellanox MCX353A-FCAT ConnectX-3 VPI, FDR IB (56Gb/s)
Firmware: 2.30.3200

8 OSS (Storage Server):

Intel® SR1560SF server
Intel® R2208GL4GS G29051-352 Grizzly Pass board
2 x Intel®Xeon® CPU E5-2680 @ 2.70GHz
64 GB total/node (8*8GB 1600MHz* Reg ECC DDR3)
Bios SE5C600.86B.01.03.0002.062020121504
1 OS hard drive (Intel® 320 Series SSD Model:SSDSA2CW600G3)
2 InfiniBand HCA Mellanox MCX353A-FCAT ConnectX-3 VPI, FDR IB (56Gb/s), Firmware: 2.30.3200 (one connected to FDR backbone, one connected to DDN backend)

Software Stack:

Redhat* Enterprise Linux* 6.4
Kernel 2.6.32-279.19.1.el6.x86_64.crt1
OFED 3.5-mic-alpha1
Lustre 2.1.5

Server Hardware LFS09

Back end: None, OSTs contain 24 SSDs

Meta Data Server (MDS) – 1:

- Intel® Server System R2224GZ4GC4
- Intel® R2208GZ4GC G11481-352 Grizzly Pass board
- 2 x Intel®Xeon®CPU E5-2680 @ 2.70GHz
- 64 GB total/node (8*8GB 1600MHz* Reg ECC DDR3)
- Bios SE5C600.86B.02.01.0002.082220131453
- 1 OS hard drive (Intel® 320 Series SSDs Model:SSDSA2CW600G3)
- 3 RAID Controller 8 SAS/SATA targets:
  - LSI Logic / Symbios Logic MegaRAID SAS 2208 [Thunderbolt] (rev 05)
  - MDT 2 x 7 SSDs (Intel 320 Series, Model: SSDSA2CW600G3) RAID0
- InfiniBand HCA Mellanox MCX353A-FCAT ConnectX-3 VPI, FDR IB (56Gb/s)
- Firmware: 2.30.3200

8 OSS (Storage Server):

- Intel® Server System R2224GZ4GC4
- Intel® R2208GZ4GC G11481-352 Grizzly Pass board
- 2 x Intel® Xeon®CPU E5-2680 @ 2.70GHz
- 64 GB total/node (8*8GB 1600MHz* Reg ECC DDR3)
- Bios SE5C600.86B.02.01.0002.082220131453
- 1 OS hard drive (Intel® 320 Series SSD Model:SSDSA2CW600G3)
- 3 RAID controllers with 8 SAS/SATA targets:
  - LSI Logic / Symbios Logic MegaRAID SAS 2208 [Thunderbolt] (rev 05)
  - 6 OST (Targets) targets per server
  - Each target consists of 4 SSDs "Intel DC S3500, 600GB "
  - SSDs are configured to fill only to 75% to conserve performance
  - 1.3 TB available storage per target (instead of theoretical maximum 2.4 TB)
  - Total capacity per server: 7.5 TB
- InfiniBand HCA Mellanox MCX353A-FCAT ConnectX-3 VPI, FDR IB (56Gb/s)
- Firmware: 2.30.3200

Software Stack:

- Redhat* Enterprise Linux* 6.4
- Kernel 2.6.32-279.19.1.el6.x86_64.crt1
- OFED 3.5-mic-alpha1
- Lustre 2.1.5

Clients

Intel® R2208GZ4GC platform
2 Intel® Xeon® E5-2697 2.7 GHz, 12 core,8GT/s dual QPI links, 130 W, 3.5GHz Max Turbo Frequency, 768kB instr L1 / 3072kB L2 / 30MB L3 cache
64 GB Memory, 8*8GB 1600MHz Reg ECC DDR3
BIOS Rev 4.6 SE5C600.86B.02.01.0002.082220131453 08/22/2013
1 OS disk (SEAGATE ST9600205SS)
InfiniBand HCA Mellanox MCX353A-FCAT ConnectX-3 VPI, FDR IB (56Gb/s)
Firmware: 2.30.3200
Redhat* Enterprise Linux* 6.4
OFED 3.5-mic-alpha1
Lustre 2.3

Acknowledgments

The author thanks Christian Black and the SSD Solutions Architecture and Engineering team (Enterprise Solutions Marketing) for their assistance in providing hardware and consulting for this article.

About the Author

Michael Hebenstreit (michael.hebenstreit@intel.com) is a senior cluster architect and tech lead for the Endeavor HPC benchmarking datacenter with over 25 years' experience in HPC. In his 9 years at Intel he helped to make Endeavor a prime HPC benchmarking datacenter in the world and was essential in integrating the Intel® Xeon Phi™ coprocessors into the cluster.

Notices

Optimization Notice
Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Notice revision #20110804