MooseFS performance scores high on InfiniBand Network

We are excited to announce that tests were successfully conducted by Core Technology in cooperation with the Interdisciplinary Centre for Mathematical and Computational Modelling at University of Warsaw, to check the performance of MooseFS over IPoIB configuration, demonstrating throughput numbers in single client and distributed setup environments. These tests were performed with MooseFS 4.0 software version but the results are also achievable with MooseFS 3.0.92+ version.

The tests show that MooseFS distributed file system is able to achieve very good performance with IPoIB protocol. Also, we get to understand that the best performance can be achieved using at least 4 threads and block size of at least 64k. Block size is a very important aspect in TCP/IP network communication, especially for random operations.

The gathered data, shows that not in all cases, increasing the number of threads increases MooseFS client performance. When we use block sizes greater than 128k the performance of sequential and random read/write does not increase more. However, increasing the number of threads, very quickly leads to maximum throughput for sequential read and write. Also, random read performance increases up to 12 threads for 2048k blocks and is linear for 16k block in whole test range from 1 to 16 threads.

All the results were achieved with IPoIB configuration. Native IB throughput achieved in such a setup is unparalleled. All tests proved that storage based on MooseFS with InfiniBand network was able to provide exceptional performance. MooseFS network defined storage is a perfect solution for HPC environment. The optimal power of MooseFS is noticeable with parallel operations on many distributed MooseFS clients.This is indeed a very good news for all the MooseFS users!

In this blog, we will first give some information about MooseFS and ICM UW and then we will explain in detail about the two tests conducted on single client and distributed client set-up environments.This will be concluded with detailed test results as mentioned in the appendix.

About MooseFS

MooseFS is a fault tolerant, highly available, highly performing, scaling-out, network distributed file system. It spreads data over several physical commodity servers, which are visible to the user as one resource.

For standard file operations MooseFS acts like any other Unix-like system:

- A hierarchical structure (directory tree)
- Stores POSIX file attributes (permissions, last access and modification times)
- Supports special files (block and character devices, pipes and sockets)
- Symbolic links (file names pointing to target files, not necessarily on MooseFS) and hard links (different names of files that refer to the same data on MooseFS)
- Access to the file system can be limited basing on IP address and/or password

Distinctive features of MooseFS are:

- High availability (i.e. redundant meta-data servers)
- High reliability (several copies of the data can be stored on separate computers)
- Capacity is dynamically expandable by simply adding new servers or disks
- Deleted files are retained for a configurable period of time (a file system level "trash bin")
- Coherent snapshots of files, even during write/access operations

MooseFS is an Open Source software available on GitHub: https://github.com/moosefs/moosefs
For more information about MooseFS please visit: http://moosefs.com

About ICM UW (Interdisciplinary Centre for Mathematical and Computational Modelling at University of Warsaw)

ICM UW is a leading data science facility in Central Europe. High performance computers used for processing, analysis, visualization and advanced computing tasks are ICM speciality. ICM's goal is to understand data and provide innovative solutions to organizations and institutions, taking advantage of their data science expertise.
For more information please visit: http://icm.edu.pl

The tests were conducted in Single Client and Distributed Client setup Environments.The below two sections provide us with the detailed analysis in these two setups.

1. Single client test

The following section provides single client test description and configuration details. Single client test means that in the whole MooseFS cluster setup, only one server was dedicated as MooseFS client. Benchmark was executed inside MooseFS client mount point. Benchmark tool used in this test was IOzone software, version 3.465.

MooseFS client tests were performed to show the differences between different block sizes and number of threads. In data transmission and data storage, a block, sometimes called a physical record, is a sequence of bytes or bits, usually containing some whole number of records, having a maximum length. The number of threads in IOzone benchmark means the number of parallel processes executed during measurement. Each thread operates on one file. In single client test, maximum number of threads was set to 16. It means that 16 files were created in MooseFS cluster.

To properly measure performance differences between different block sizes and number of threads, the test was executed five times for each set of parameters. Maximum and minimum results were removed from average calculations.

IOzone command used in tests:

```
$ iozone -eI -r {blocksize} -s1g -i0 -i1 -i2 -t {threads}
```

IOzone benchmark options:

- e - Include flush (fsync, ush) in the timing calculations.
- I - DIRECT I/O for all file operations. Tells the file system that all operations are to bypass the buffer cache and go directly to disk.
- r - Record/block size
- s - File size 1GB.
- i - 0 = write operations, 1 = read operations, 2 = random read and random write operations.
- t - Allows the user to specify how many threads or processes to be active during the measurement.

1.1 Topology

Single client test cluster consisted of two master servers (leader and follower), seven chunk servers and one client server (Figure 1). MooseFS client software was installed only on one physical server. All servers were connected through Mellanox FDR switch with 0.02 ms port to port latency declared by producer. InfiniBand adapter used in each server was ConnectX-3 Mellanox card with maximum throughput 56 Gbit/s. All connections were made with QSFP+ fiber optic cables.

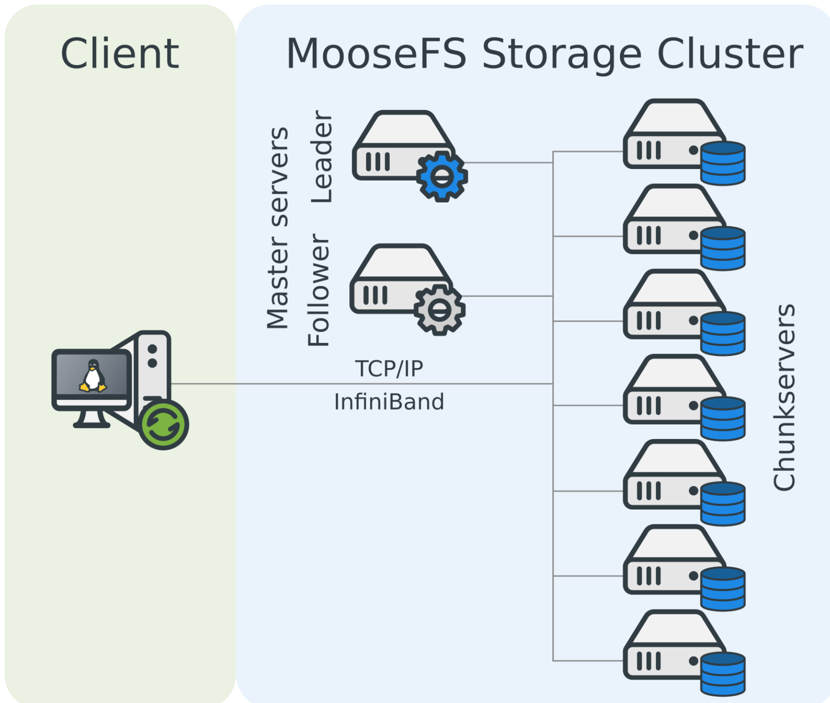

Figure 1: Single client test topology

1.2 Configuration

To eliminate hard disk bottleneck, 100GB RAM disks were created on each chunk server. Network transport used IPoIB protocol. No kernel modifications and no additional components were required. MooseFS replication was set to goal 1. Measured average ping between client server and other servers in cluster was 0.022 ms. Operating system was Centos 7.3 with kernel 3.10.0-514.6.1.el7.x86_64.

Hardware configuration of all machines:

- CPU - 2 x Intel Xeon CPU E5-2680 v3 2,5GHz (12 cores, 24 threads)
- RAM - 128GB DDR4 2133 MHz
- NIC - ConnectX-3 Mellanox MT27500 Family (56 Gbit/s)
- Mellanox FDR switch

1.3 Results

The following subsection shows plots with test results for sequential and random read/write operations. Figures 2, 3 show how performance changes with block size and number of processing threads. We choose 4 and 8 threads to prepare block size plot (Figure 2) and 16k and 2048 blocks for threads plot (Figure 3). Figures 4, 5 show performance during random access read/write operations, similar to the previous two. The last plot (Figure 6) shows sequential and random access for read/write IOPS with 16k blocks and threads in range from 1 to 16.

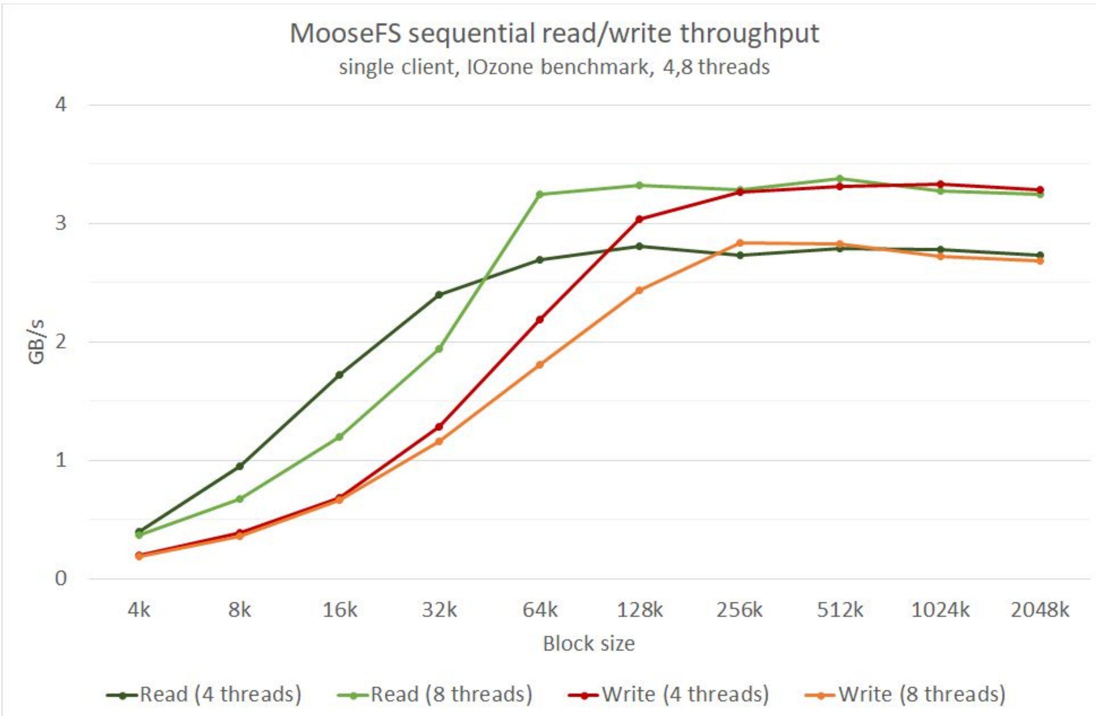

Figure 2: Read/write test results using 4 and 8 threads for block sizes starting from 4k to 2048k

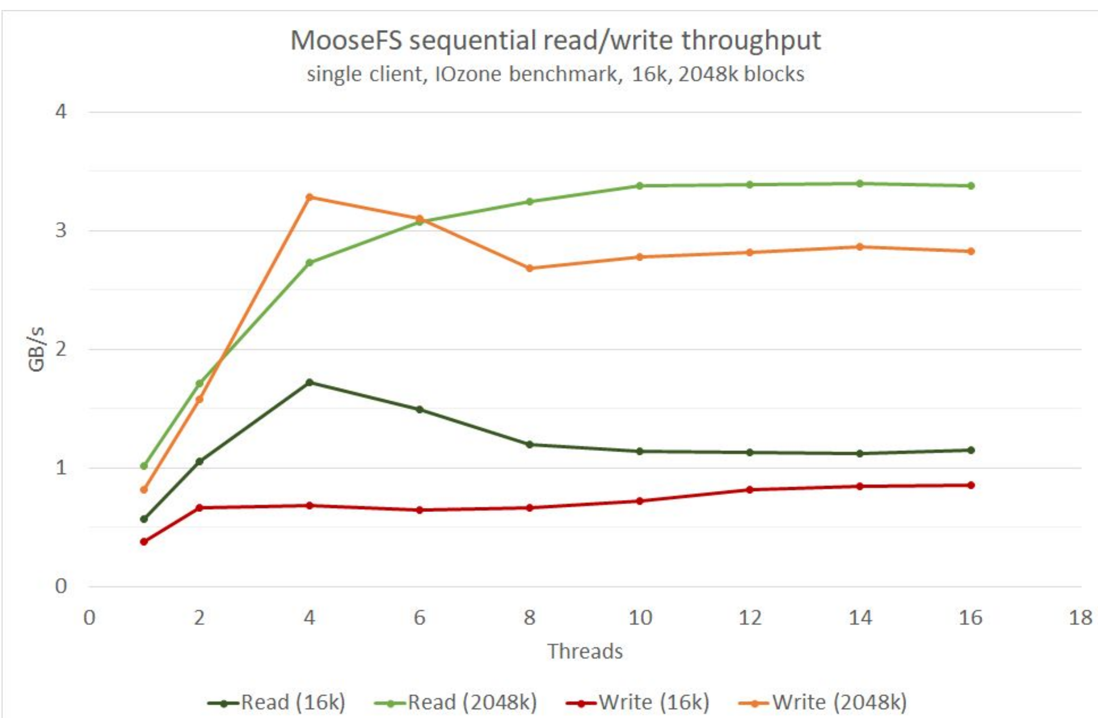

Figure 3: Read/write test results using 16k and 2048 blocks for number of threads from 1 to 16

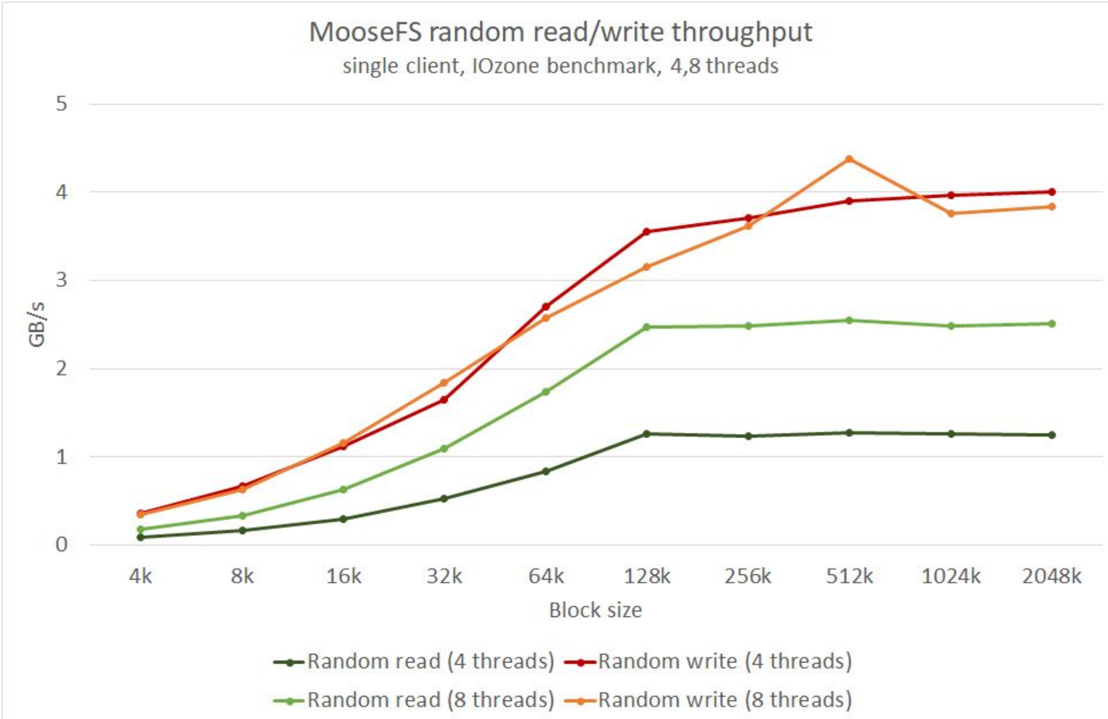

Figure 4: Random read/write test results with 4, 8 threads for block size from 4k to 2048k

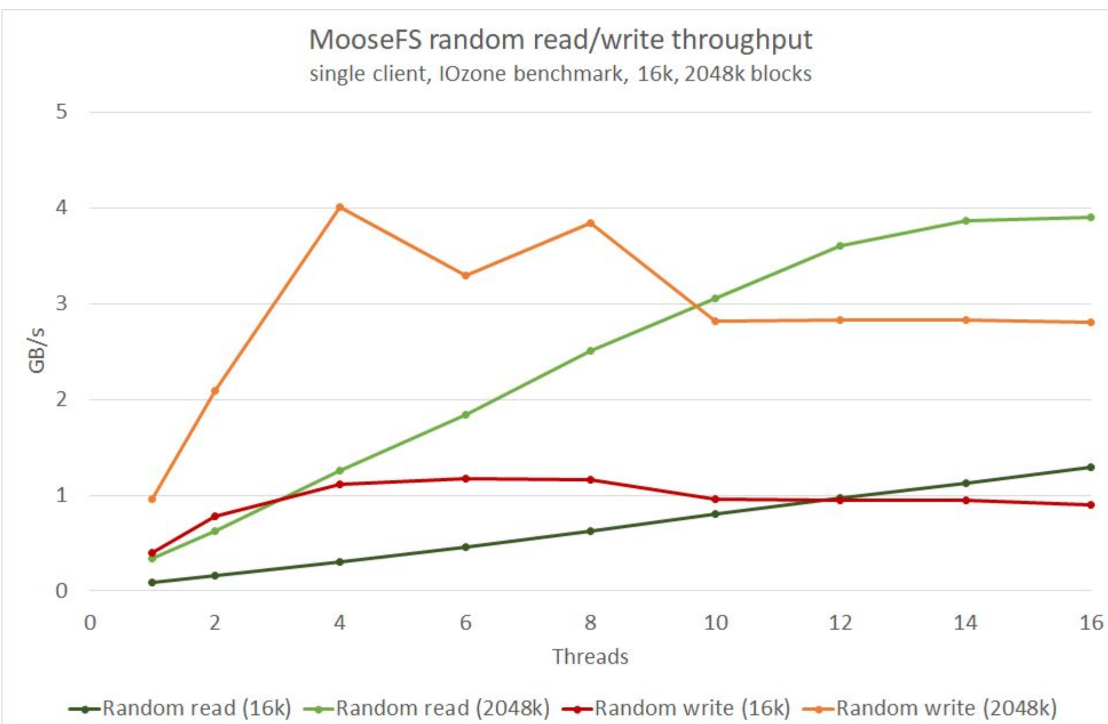

Figure 5: Random read/write test results with 16k, 2048k blocks for threads from 1 to 16

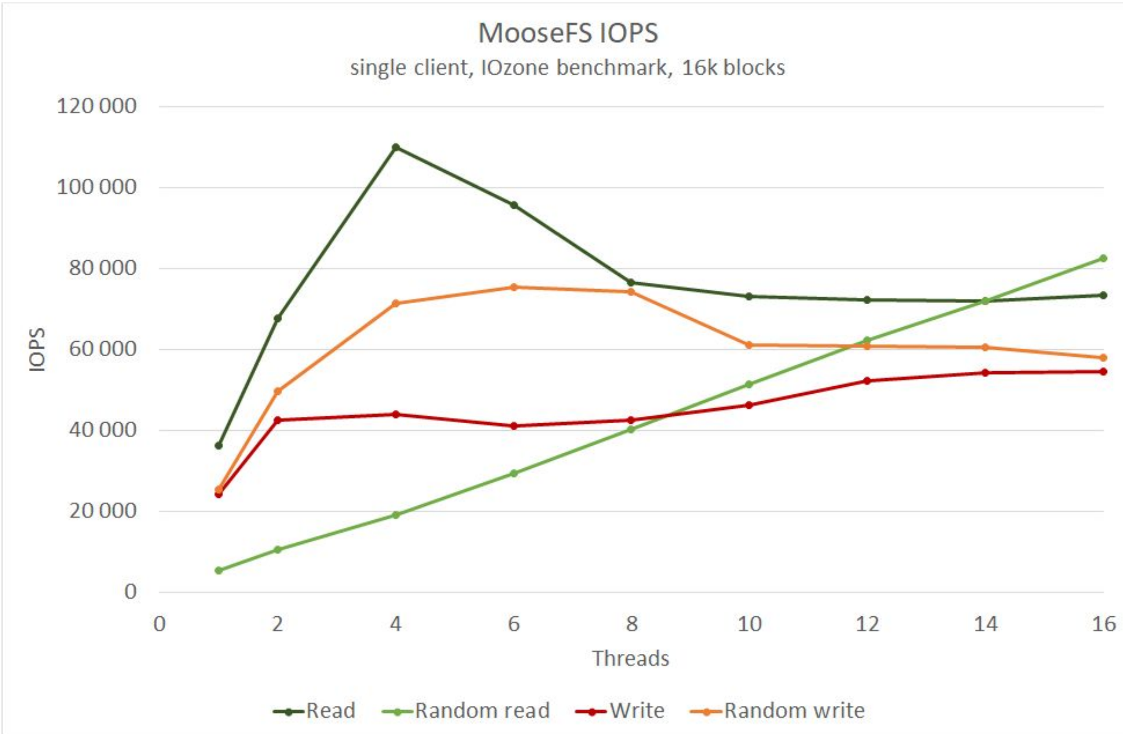

Figure 6: Sequential and random read/write IOPS with 16k blocks

2. Distributed client test

This section provides description and configuration details for the distributed test. In this test all eight MooseFS servers worked as chunkserver and client simultaneously. IOzone benchmark software was executed in cluster testing mode. Each MooseFS client handled 4 separate IOzone processes, each IOzone process operated on four files. In total, the test had 32 threads distributed over eight servers. To properly present performance differences between different block sizes, the test was executed five times. Maximum and minimum results were removed from average calculations.

IOzone command line:

```
$ iozone -ceIT -i0 -i1 -i2 -+n -r {blocksize} -s1g -+H moosefs -m1 -+m hosts.cfh -t32
```

IOzone benchmark options:

- **c** - Include close() in the timing calculations
- **e** - Include flush (fsync, ush) in the timing calculations
- **I** - Direct I/O for all file operations. Tells the file system that all operations are to bypass the buffer cache and go directly to disk
- **T** - Use POSIX pthreads for throughput tests. Available on platforms that have POSIX threads.

- **i** - **0** = write, **1** = read, **2** = random read and random write operations
- **-+n** - No retests selected.
- **r** - Record/block size
- **s1g** - File size 1GB.
- **-+H** - Hostname of the PIT server
- **-+m** - hosts.cfg file contains the configuration information of the clients for cluster testing
- **t** - Allows the user to specify how many threads or processes to have active during the measurement.

2.1 Distributed client test topology

Distributed client test cluster consists of two master servers and eight chunk servers and clients. All hardware components were the same as in the single client test. One additional chunkserver was prepared on client machine from previous test. All of eight chunk servers used MooseFS client to run IOzone tests.

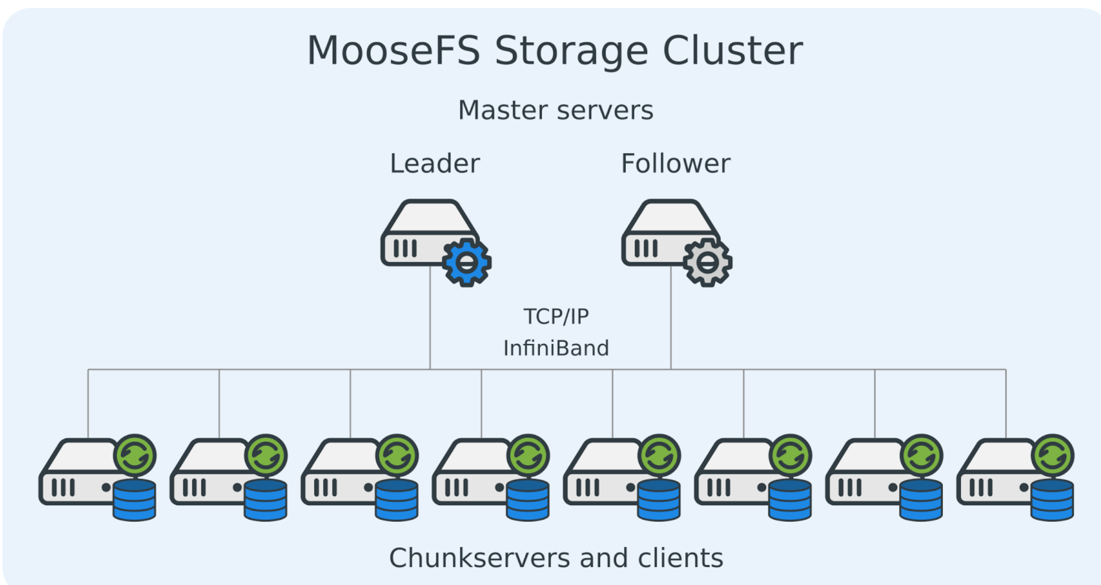

Figure 7: MooseFS distributed test infrastructure

2.2 Distributed test results

The following graph shows read, write, random read and random write operations throughput with different block size for 32 threads distributed test. On X axis is the block size and on Y axis is the throughput in gigabytes per second.

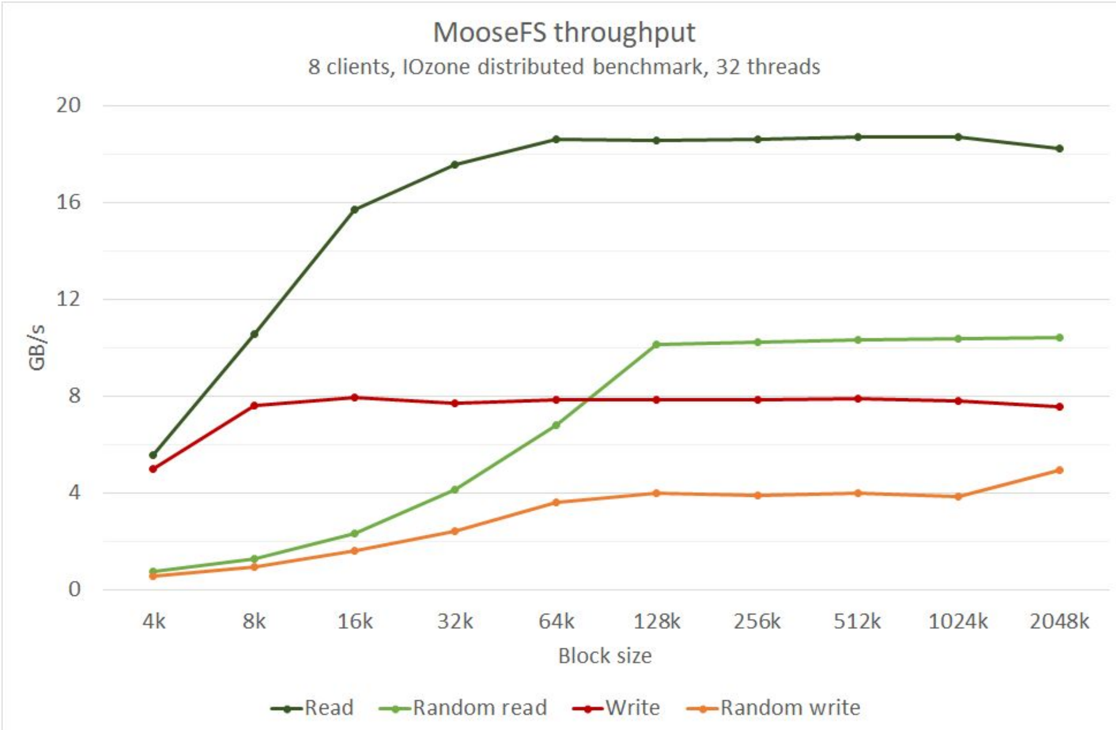

Figure 8: Sequential and random read/write distributed test results with 32 threads and block size in range from 4k to 2048k

Appendix

This section provides detailed results gathered during single and distributed IOzone benchmark tests. The following tables present more detailed information about IOzone tests. Table 1 presents IOzone test results with threads in range from 1 to 16 and block size in range from 4k to 2048k. Table 2 presents IOzone distributed test results with 32 threads using eight machines.

Table 1: MooseFS single client IOzone test results

Block size	**Threads**	**Read**		**Write**		**Random read**		**Random write**	
		MB/s	**IOPS**	**MB/s**	**IOPS**	**MB/s**	**IOPS**	**MB/s**	**IOPS**
4k	1	213	54 654	113	28 803	25	6 296	137	35 151
	2	403	103 114	205	52 590	46	11 879	242	61 839
	4	396	101 352	200	51 247	90	23 146	360	92 234
	6	355	90 860	176	44 933	132	33 697	353	90 347
	8	366	93 679	190	48 594	178	45 668	343	87 759
	10	379	97 018	207	52 893	229	58 661	294	75 150
	12	408	104 362	236	60 301	278	71 150	314	80 390

	14	433	110 837	256	65 526	330	84 432	328	83 970
	16	429	109 837	260	66 547	378	96 716	300	76 921
8k	1	379	48 528	224	28 657	47	5 960	224	28 619
	2	695	88 976	376	48 101	89	11 395	448	57 324
	4	953	121 935	386	49 373	167	21 408	663	84 917
	6	688	88 101	344	44 030	246	31 479	685	87 694
	8	678	86 799	361	46 152	330	42 183	627	80 299
	10	683	87 478	392	50 125	426	54 494	557	71 307
	12	693	88 661	449	57 461	512	65 597	567	72 601
	14	727	93 104	478	61 248	601	76 927	569	72 781
	16	761	97 395	489	62 582	679	86 893	547	70 006
16k	1	565	36 159	376	24 085	85	5 430	395	25 302
	2	1 059	67 756	662	42 382	162	10 354	774	49 535
	4	1 718	109 982	685	43 850	299	19 128	1 116	71 438
	6	1 493	95 564	643	41 177	457	29 273	1 178	75 389
	8	1 196	76 520	663	42 414	626	40 063	1 158	74 084
	10	1 142	73 102	723	46 271	800	51 219	955	61 111
	12	1 130	72 338	815	52 133	970	62 097	949	60 734
	14	1 125	72 021	845	54 087	1 122	71 829	945	60 450
	16	1 147	73 416	853	54 591	1 288	82 424	904	57 849
32k	1	806	25 781	578	18 499	148	4 727	599	19 166
	2	1 384	44 303	1 107	35 414	281	8 998	1 204	38 540
	4	2 400	76 799	1 279	40 927	531	17 000	1 650	52 794
	6	2 594	82 992	1 127	36 068	800	25 588	1 606	51 403
	8	1 936	61 944	1 163	37 230	1 095	35 055	1 839	58 860
	10	1 797	57 509	1 235	39 517	1 410	45 129	1 382	44 226
	12	1 713	54 822	1 352	43 270	1 706	54 596	1 452	46 452
	14	1 688	54 031	1 367	43 747	1 986	63 548	1 432	45 812
	16	1 707	54 627	1 380	44 166	2 252	72 064	1 423	45 540
64k	1	943	15 084	715	11 446	229	3 659	926	14 821
	2	1 563	25 004	1 412	22 594	428	6 848	1 666	26 658
	4	2 691	43 059	2 184	34 944	838	13 408	2 709	43 345
	6	3 009	48 147	1 771	28 339	1 267	20 266	2 694	43 102
	8	3 244	51 909	1 803	28 846	1 737	27 798	2 579	41 265

	10	3 347	53 550	1 810	28 967	2 187	34 986	1 956	31 299
	12	2 511	40 173	1 949	31 185	2 603	41 654	1 998	31 973
	14	2 557	40 918	1 970	31 518	3 017	48 264	1 981	31 695
	16	2 658	42 525	1 965	31 444	3 311	52 970	1 971	31 543
128k	1	1 026	8 206	804	6 432	331	2 644	903	7 221
	2	1 779	14 231	1 613	12 901	621	4 969	1 921	15 365
	4	2 810	22 484	3 036	24 287	1 262	10 093	3 549	28 394
	6	3 223	25 784	2 461	19 691	1 835	14 682	3 303	26 423
	8	3 324	26 590	2 439	19 512	2 469	19 749	3 155	25 241
	10	3 404	27 230	2 350	18 798	3 017	24 137	2 438	19 505
	12	3 386	27 091	2 472	19 773	3 556	28 451	2 515	20 119
	14	3 484	27 874	2 478	19 822	3 915	31 317	2 491	19 925
	16	3 303	26 425	2 492	19 938	3 843	30 742	2 487	19 899
256k	1	1 036	4 143	823	3 291	334	1 338	906	3 625
	2	1 713	6 852	1 678	6 710	629	2 514	1 959	7 837
	4	2 727	10 909	3 264	13 057	1 242	4 968	3 715	14 860
	6	3 032	12 129	2 691	10 763	1 861	7 444	3 091	12 362
	8	3 287	13 150	2 840	11 361	2 488	9 953	3 618	14 473
	10	3 378	13 512	2 567	10 268	3 059	12 235	2 629	10 518
	12	3 411	13 645	2 640	10 560	3 640	14 560	2 659	10 637
	14	3 354	13 417	2 655	10 621	3 942	15 768	2 635	10 542
	16	3 267	13 069	2 646	10 584	3 886	15 544	2 646	10 585
512k	1	1 058	2 116	829	1 659	334	669	960	1 919
	2	1 689	3 377	1 636	3 272	622	1 245	2 054	4 108
	4	2 785	5 571	3 313	6 626	1 269	2 539	3 897	7 794
	6	3 177	6 355	2 844	5 689	1 841	3 682	3 478	6 956
	8	3 380	6 760	2 826	5 652	2 546	5 091	4 384	8 769
	10	3 406	6 813	2 661	5 323	3 078	6 156	2 733	5 465
	12	3 437	6 874	2 742	5 483	3 623	7 245	2 738	5 477
	14	3 424	6 849	2 729	5 459	3 969	7 939	2 733	5 465
	16	3 277	6 554	2 742	5 484	3 844	7 688	2 730	5 461
1024k	1	1 031	1 031	841	841	335	335	969	969
	2	1 648	1 648	1 607	1 607	628	628	2 080	2 080
	4	2 774	2 774	3 330	3 330	1 258	1 258	3 966	3 966

	6	3 176	3 176	3 087	3 087	1 792	1 792	3 103	3 103
	8	3 274	3 274	2 721	2 721	2 480	2 480	3 767	3 767
	10	3 442	3 442	2 698	2 698	3 118	3 118	2 777	2 777
	12	3 373	3 373	2 777	2 777	3 602	3 602	2 767	2 767
	14	3 389	3 389	2 795	2 795	3 917	3 917	2 768	2 768
	16	3 353	3 353	2 805	2 805	3 838	3 838	2 797	2 797
2048k	1	1 020	510	815	407	337	169	958	479
	2	1 714	857	1 581	790	629	315	2 090	1 045
	4	2 734	1 367	3 283	1 642	1 255	627	4 009	2 005
	6	3 075	1 538	3 103	1 551	1 838	919	3 298	1 649
	8	3 248	1 624	2 686	1 343	2 507	1 253	3 837	1 919
	10	3 381	1 691	2 777	1 388	3 054	1 527	2 822	1 411
	12	3 388	1 694	2 819	1 409	3 602	1 801	2 826	1 413
	14	3 397	1 699	2 864	1 432	3 869	1 934	2 823	1 411
	16	3 380	1 690	2 824	1 412	3 899	1 950	2 803	1 401

Table 2: MooseFS distributed IOzone test with 32 threads

Block size	Threads	Read		Write		Random read		Random write	
		MB/s	IOPS	MB/s	IOPS	MB/s	IOPS	MB/s	IOPS
4k	32	5 575	1 427 207	5 008	1 281 999	734	187 937	537	137 516
8k		10 570	1 352 946	7 602	973 014	1 262	161 583	939	120 175
16k		15 724	1 006 352	7 947	508 592	2 309	147 771	1 586	101 510
32k		17 581	562 595	7 711	246 761	4 143	132 588	2 408	77 062
64k		18 623	297 962	7 853	125 656	6 805	108 881	3 585	57 363
128k		18 552	148 417	7 839	62 712	10 144	81 151	4 000	32 001
256k		18 590	74 362	7 833	31 332	10 218	40 871	3 872	15 489
512k		18 704	37 409	7 878	15 757	10 323	20 646	3 964	7 928
1024k		18 700	18 700	7 802	7 802	10 371	10 371	3 828	3 828
2048k		18 247	9 123	7 565	3 783	10 424	5 212	4 950	2 475