MDS Performance Analysis

Author: Parinay Kondekar <parinay.kondekar@sun.com>

AUTHOR	Version	DATE	DESCRIPTION OF DCOUMENT CHANGE
Parinay Kondekar	0.1	8th May 09	First draft of MDS performance benchmarking
Parinay Kondekar	0.2	11th May 09	Included review comments from Atul Vidwansa <Atul.Vidwansa@sun.com>

Cluster configuration

MDS	Sun Fire X4540 (Thor) with 2 Quad-Core AMD Opteron(tm) Processor 2356 ~64GM RAM
	48 Hitachi HUA721050KLA33 SATA disks of 500.1 GB each
MDT	RAID0 array of 20 RAID1 arrays
	Write Through NCQ/TCQ enabled disks Queue depth 64 Internal Journaling

OSS	Sun Fire X4540 (Thor) with 2 Quad-Core AMD Opteron(tm) Processor 2356 ~64GM RAM 4 OSS servers
OST	7 RAID6 arrays with external journaling Hitachi HUA721050KLA33 SATA disks

Clients	Total 70 clients. Pegasus+ blades 4 Quad-Core AMD Opteron(tm) Processor 8380 ~16GB RAM

Network	DDR Infiniband

Lustre versions	Lustre 1.6.7.1
	Lustre 1.8.0
	Lustre 2.0 (1.9.170)

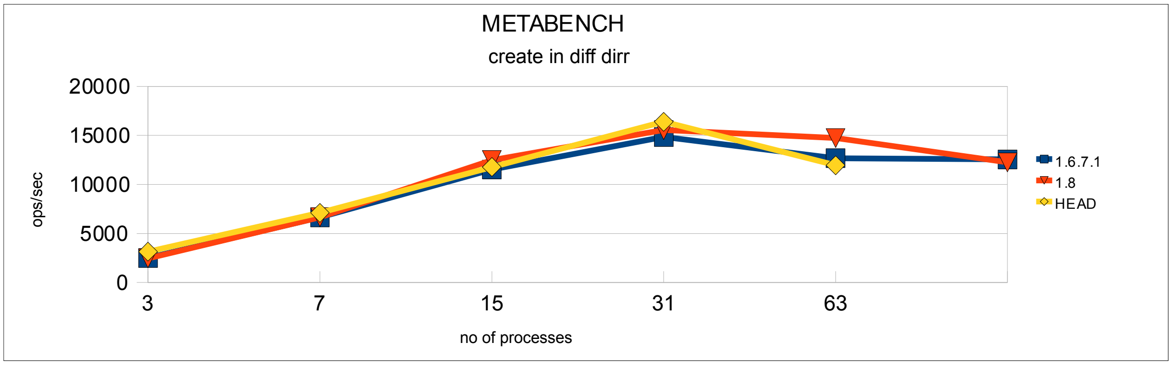
Kernel	RHEL5 with 2.6.18-128.1.1.el5 kernel on x86_64 architecture

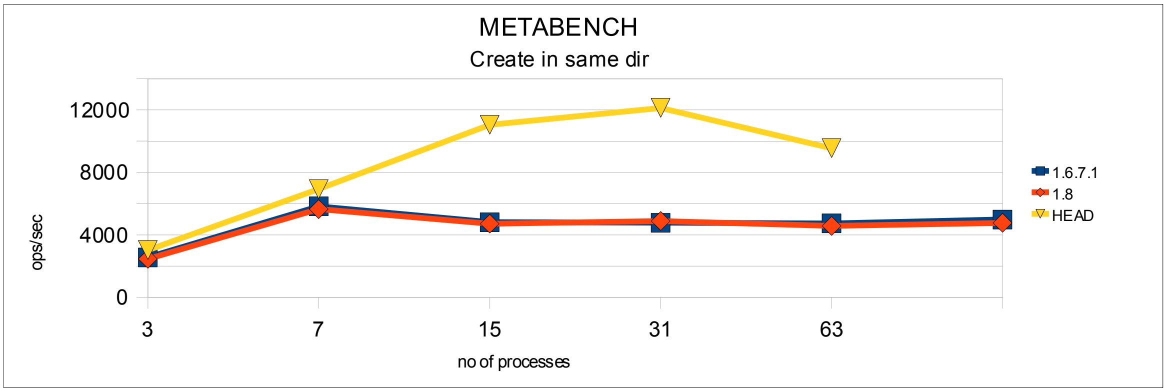
Tools	
metabench	-C(create file) -D (delete file) -S (stat file) -k(deleting all files after the tests)
mdsrate	--create --stat --unlink
mdtest	-N: stride # between neighbor tasks for file/dir stat (local=0)
	-p: pre-iteration delay (in seconds)
	-y' option to sync file after write

No of files/dirs	304000

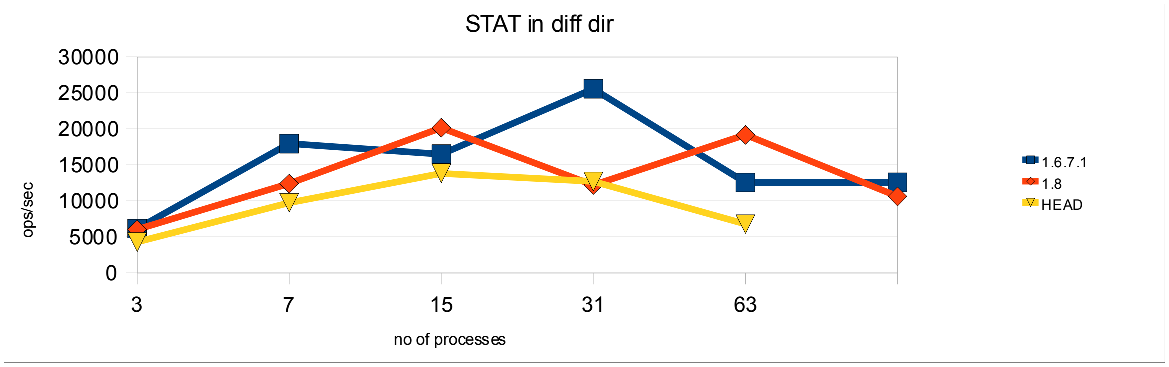
Summary	The no of clients in case of multi-clients run is 70 and no of files/dirs is 304000.
	This is observed that after $NP >32, the MD performance for 1.6,1.8,2.0 seems to either drop or doesn't scale. The client nodes are 16 CPU (4 Quad core AMD Opetron). It seems are cpu get saturated at $NP=16.
	1.8 and 1.6.7 performance is fairly close and the delta difference in them is small.
	Lustre 2.0 scales well with increase in $NP in case of multi as well single client runs
	2.0 performance in case of multi-client runs, in same dir is much better than the 1.6 or 1.8 Overall MD performance of 2.0 seems to be better compared 1.8 or 1.6.

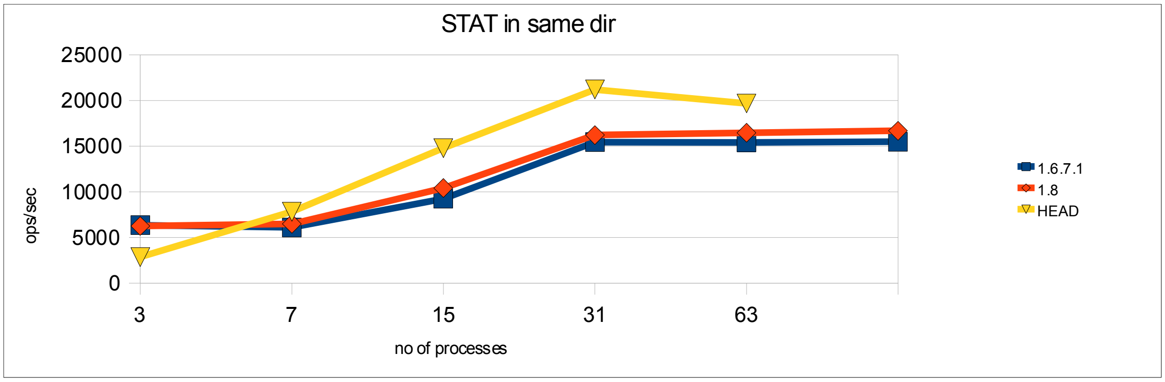
graphs- multi-clients



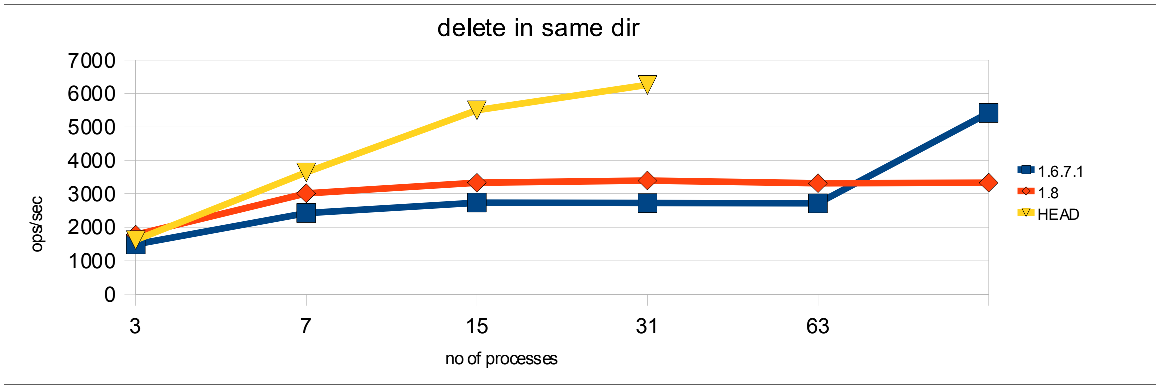


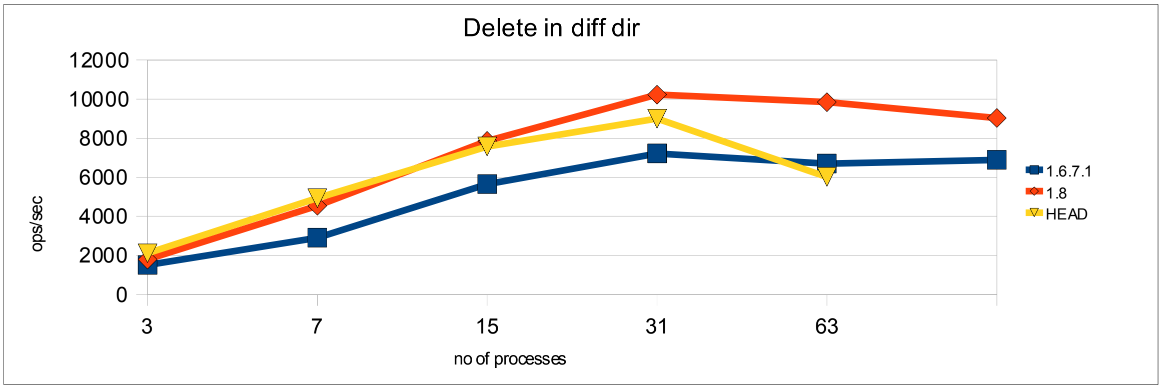
File create operation, in same dir performs better in 2.0 than 1.6 or 1.8. Create in diff dir scales linearly in all cases. The client is 4 Quad AMD(16 CPU), thus after 16 processes,
with the increase in $NP the performance drops.





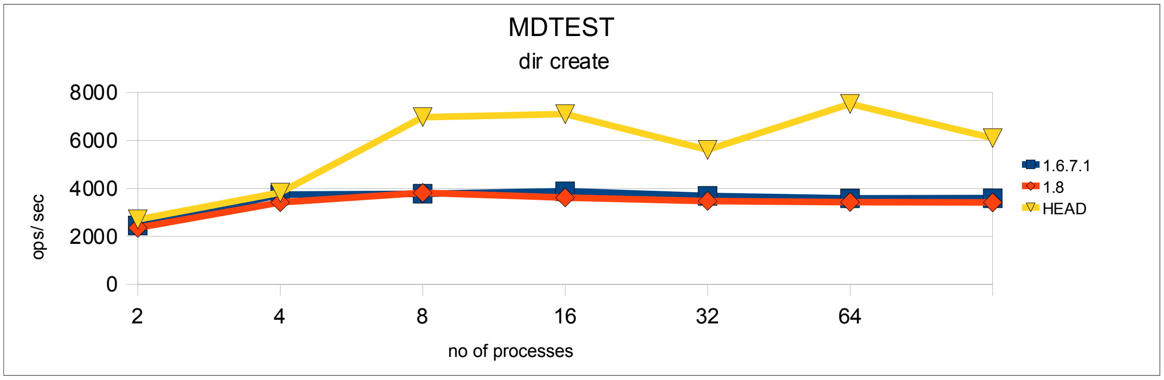
File STAT in same dir in case of 2.0 is performing well compared to 1.8 and 2.0. With increase in $NP it scales well. STAT in diff dir is not consistent at all.

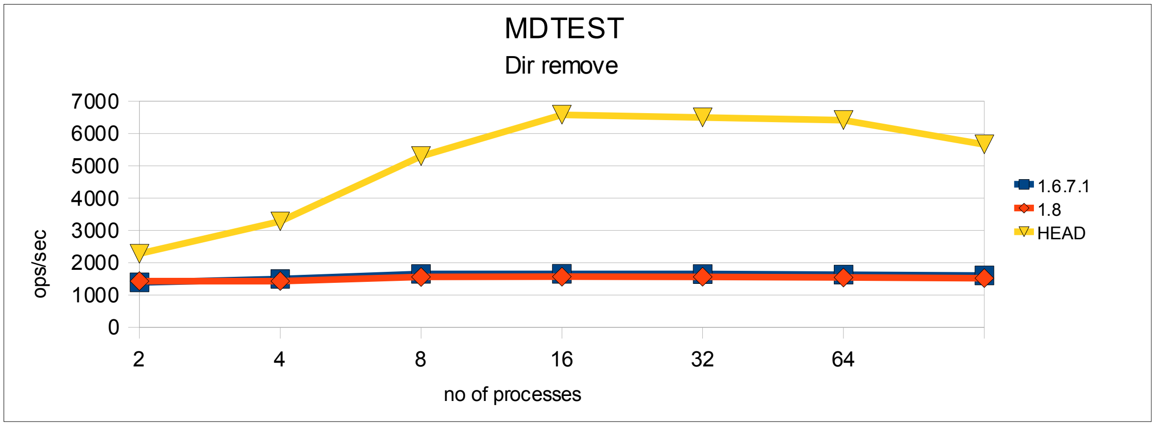


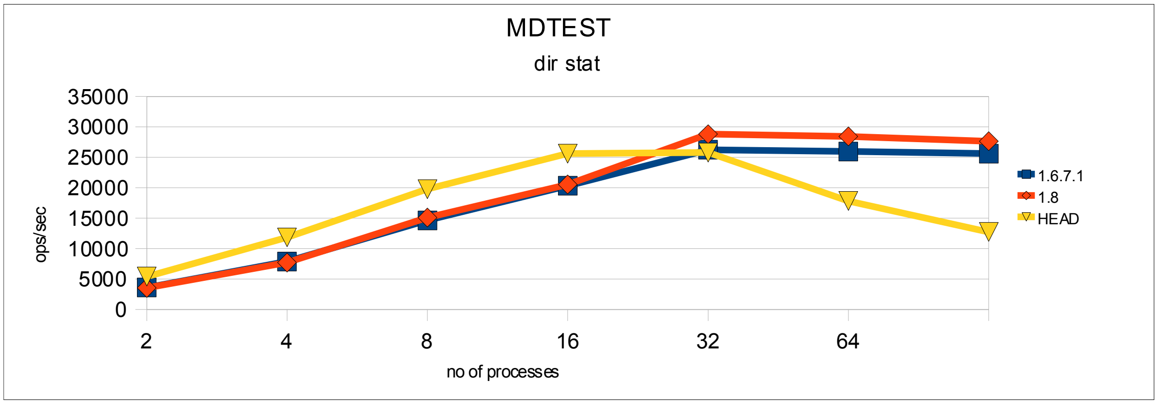


Delete in diff dir, with increase of $NP the performance is linearly scaling. With $NP > 32 there is a dip in performance.

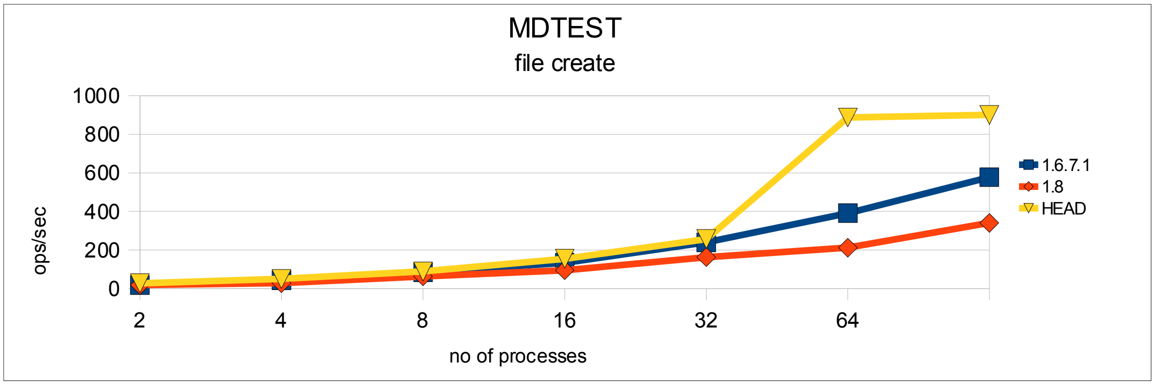
graphs- multi-clients

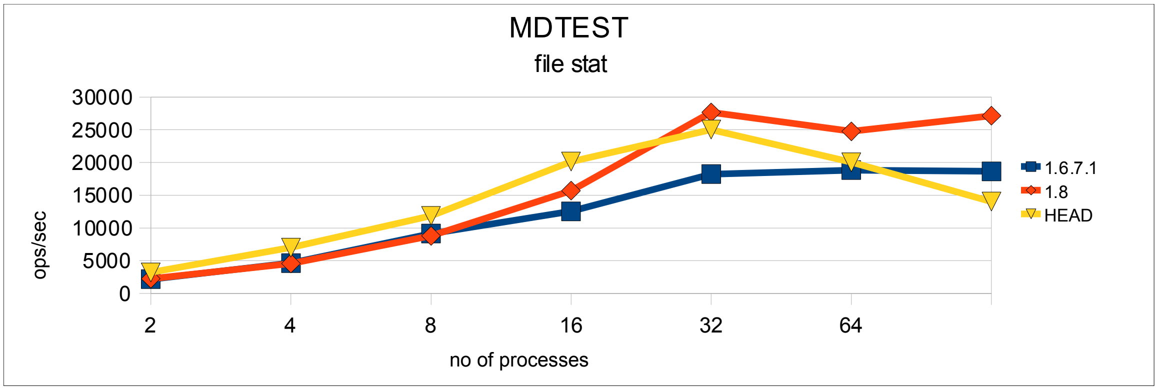


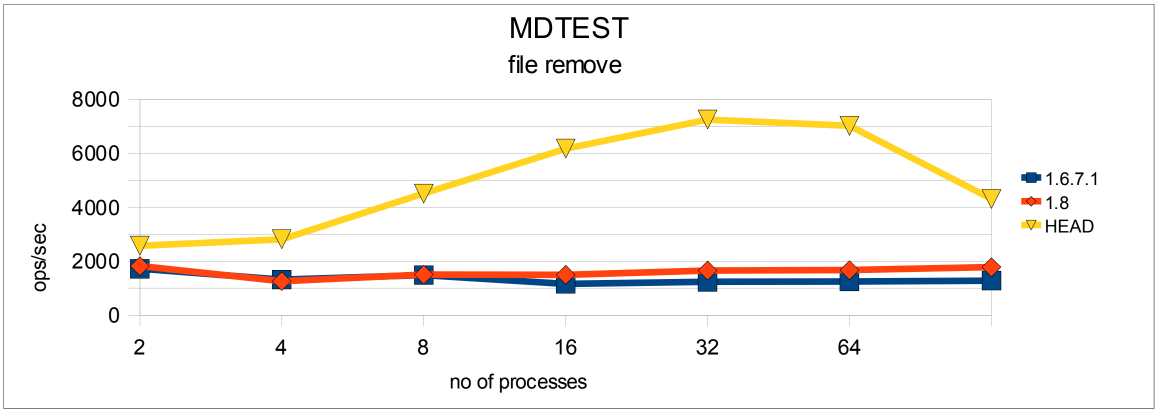




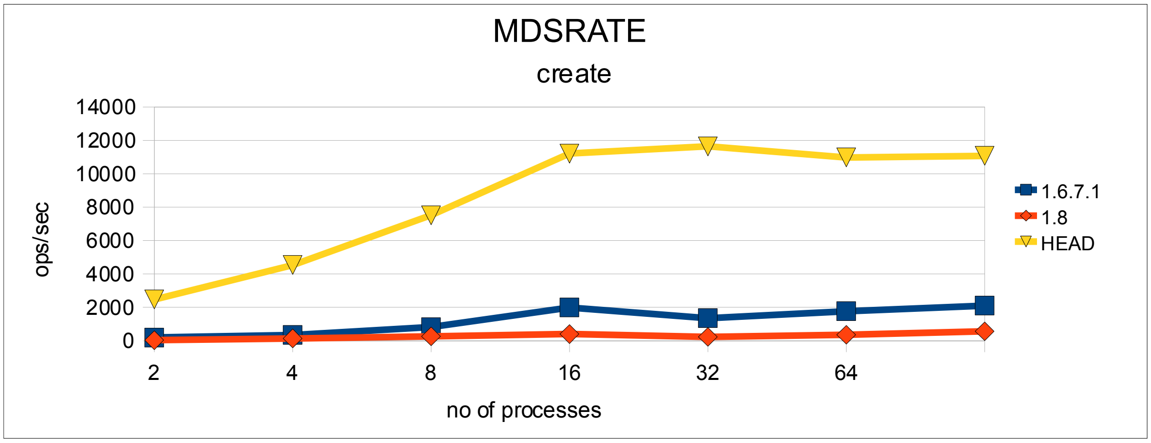
MDTEST runs till a little longer than expected due to test parameter “-y – sync after write”. 2.0 Outperforms both 1.6 and 1.8 for all, dir create, stat, remove. 1.8 performs slightly better than 1.6.

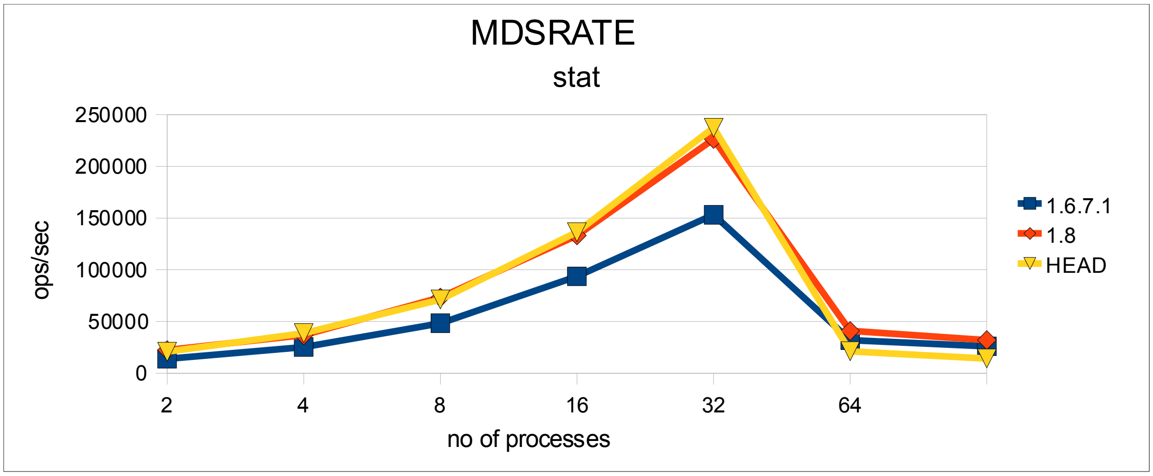


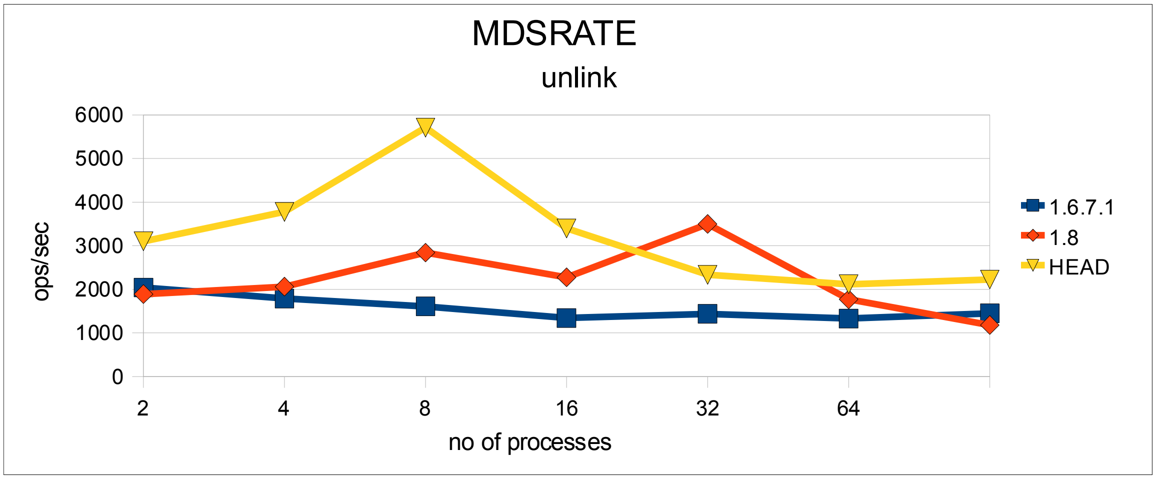




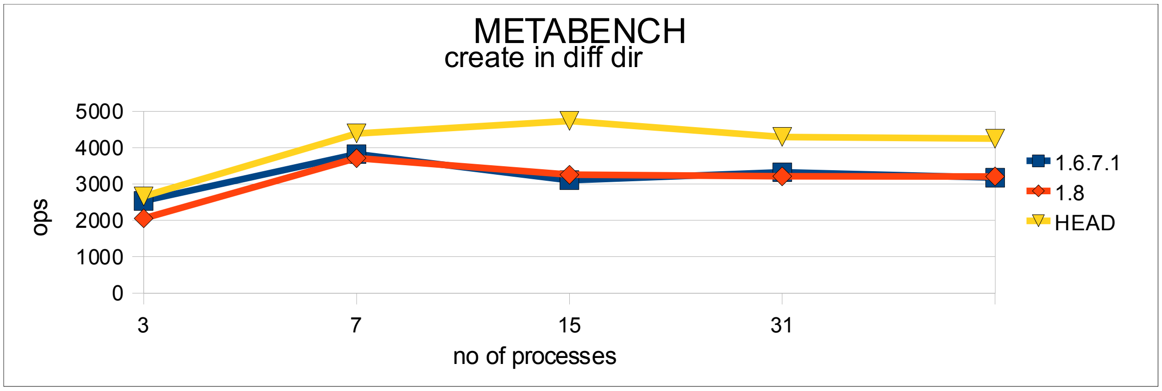
2.0 performance is better compared to 1.6 and 1.8. 2.0 is also scaling well.

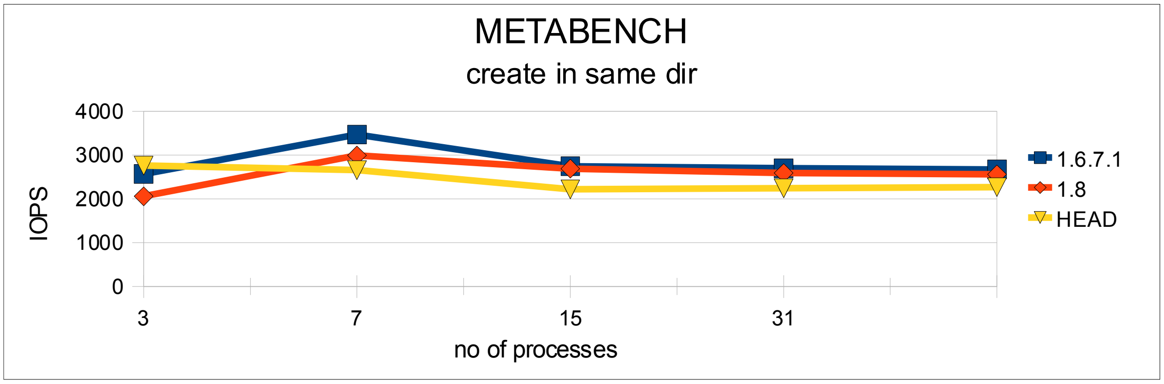


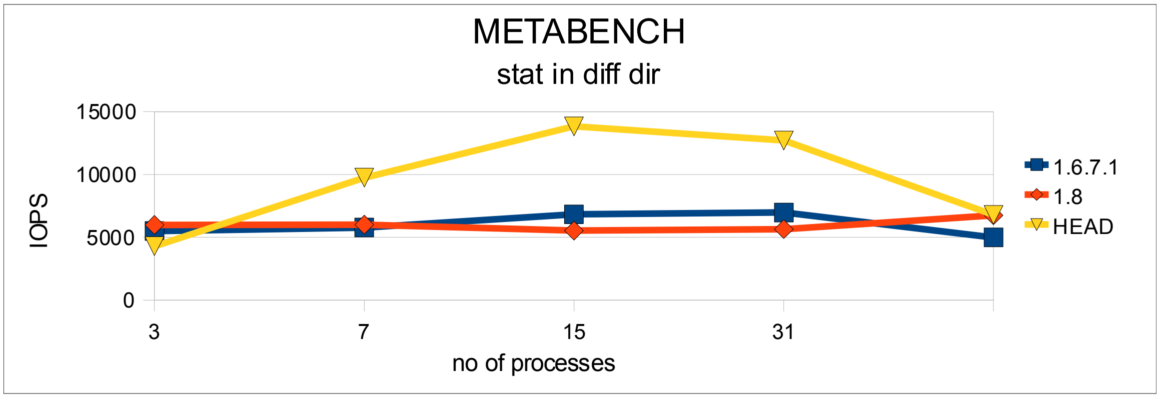


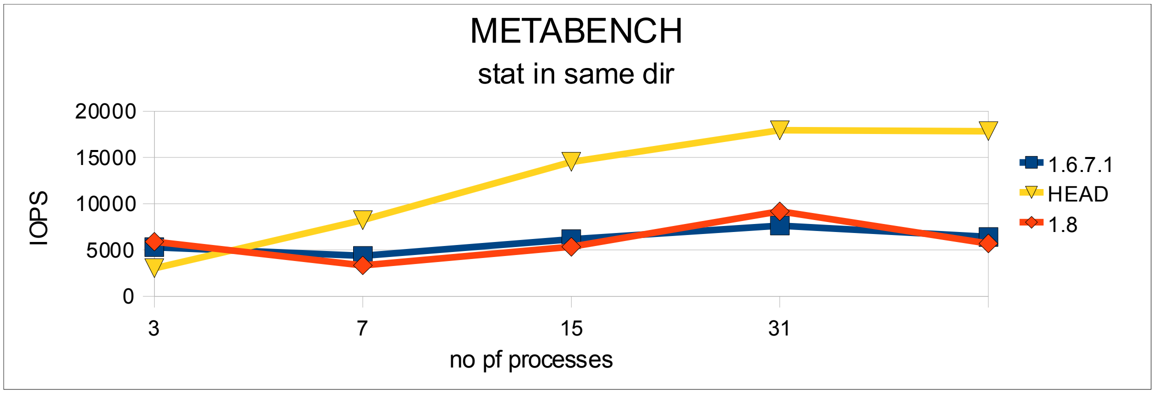


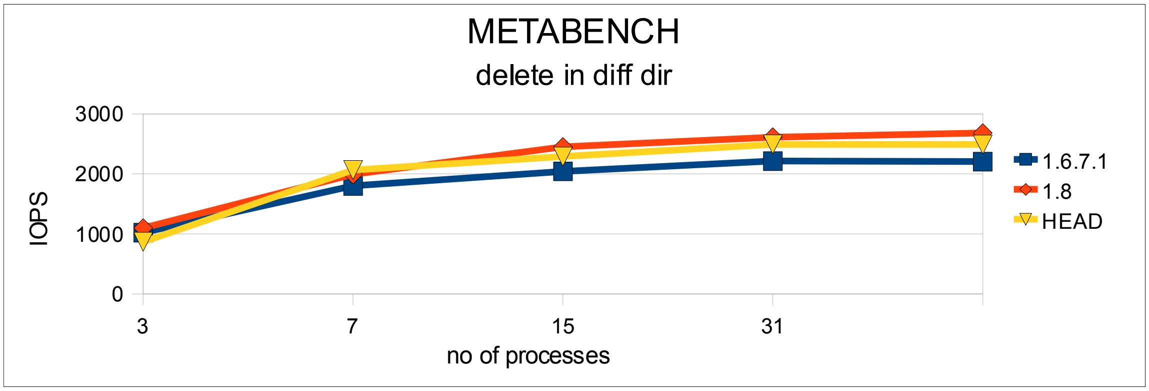
Overall 2.0 seems to perform better. In case file create the 1.6 better than 1.6. Unlink operation Performance is quite sporadic .

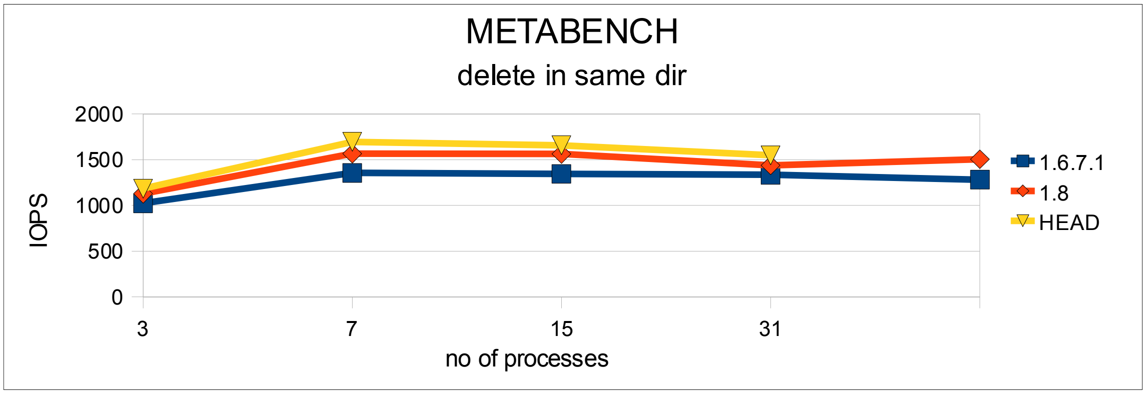

METABENCH
create in diff dir
ops
5000
4000
3000
2000
1000
0
3
7
15
31
no of processes
1.6.7.1
1.8
HEAD

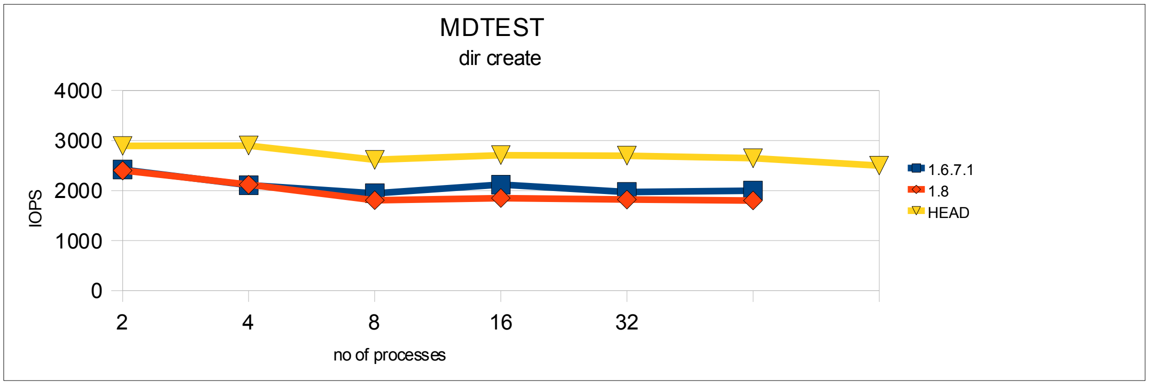

METABENCH
create in same dir
IOPS
4000
3000
2000
1000
0
3
7
15
31
no of processes
1.6.7.1
1.8
HEAD

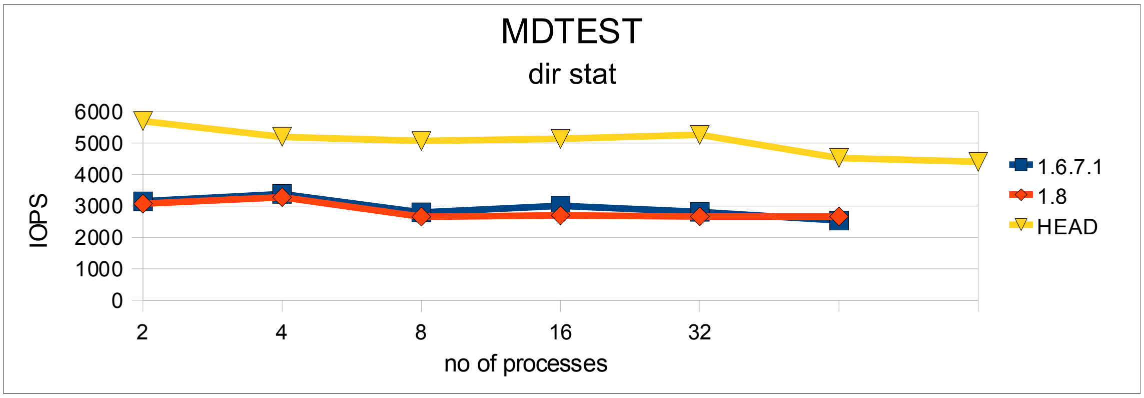

METABENCH
stat in diff dir
IOPS
15000
10000
5000
0
3
7
15
31
no of processes
1.6.7.1
1.8
HEAD

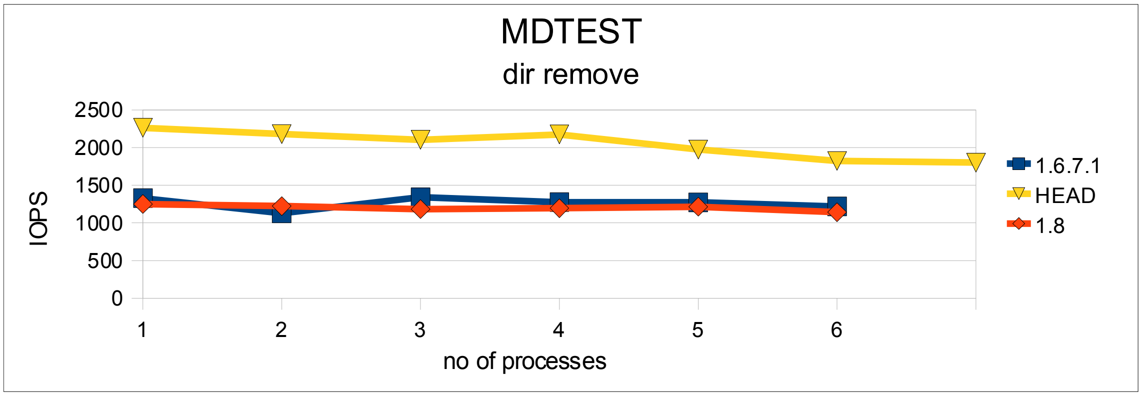

METABENCH
stat in same dir
IOPS
20000
15000
10000
5000
0
3
7
15
31
no pf processes
1.6.7.1
HEAD
1.8

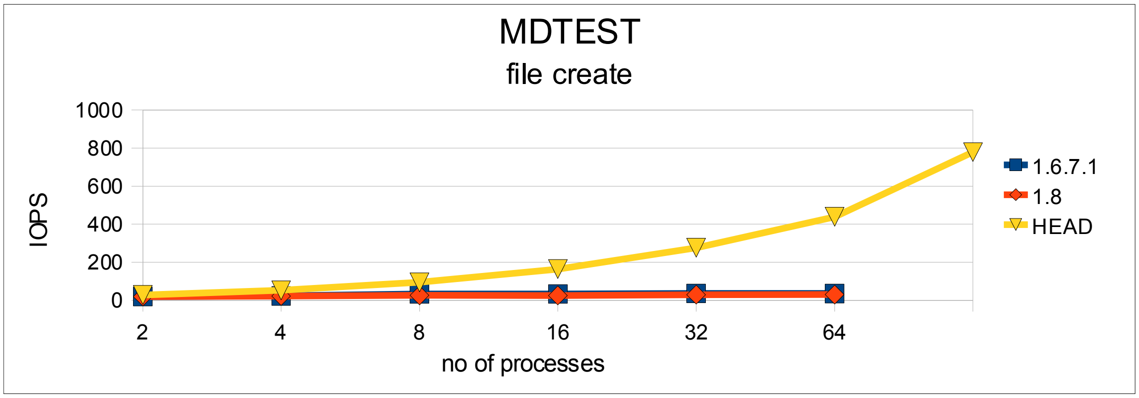

METABENCH
delete in diff dir
3000
2000
1000
0
IOPS
3
7
15
31
no of processes
1.6.7.1
1.8
HEAD

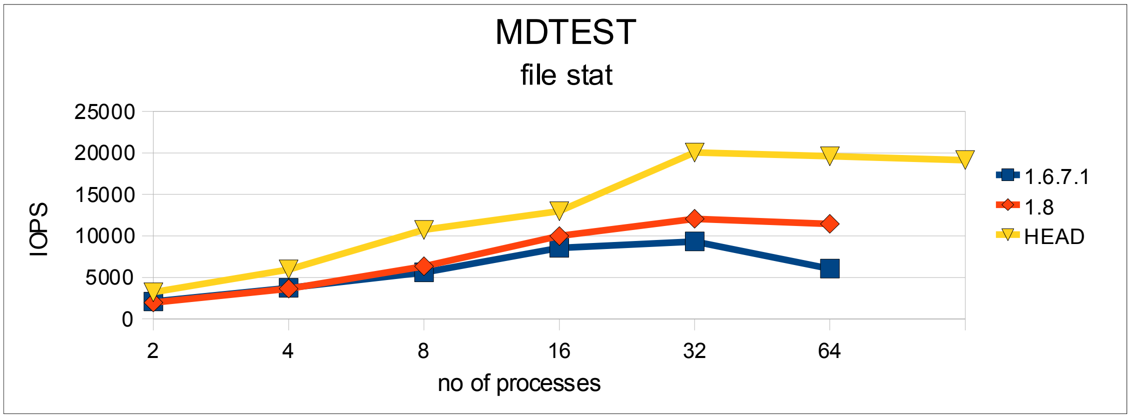

METABENCH
delete in same dir
2000
1500
1000
500
0
IOPS
3
7
15
31
no of processes
1.6.7.1
1.8
HEAD

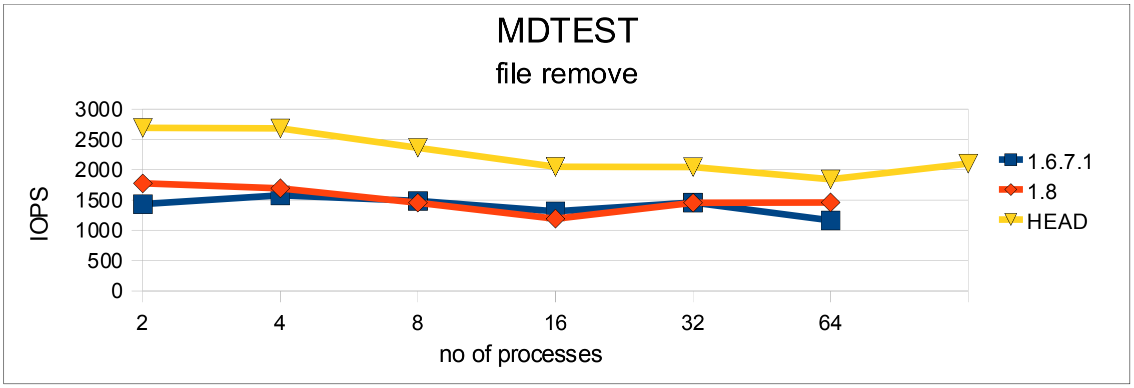

MDTEST
dir create
4000
3000
2000
1000
0
IOPS
2
4
8
16
32
no of processes
1.6.7.1
1.8
HEAD

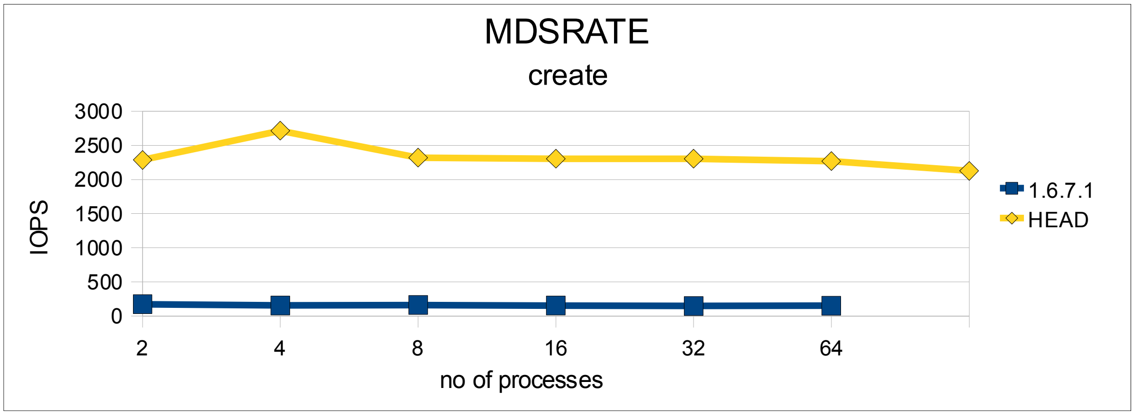

MDTEST
dir stat
6000
5000
4000
3000
2000
1000
0
IOPS
2
4
8
16
32
no of processes
1.6.7.1
1.8
HEAD

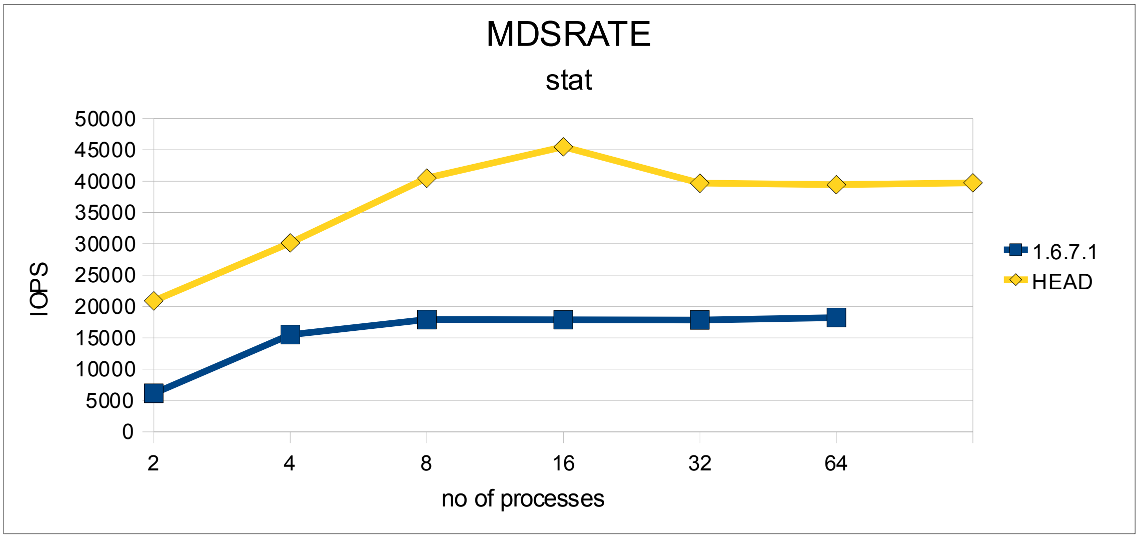

MDTEST
dir remove
IOPS
2500
2000
1500
1000
500
0
1
2
3
4
5
6
no of processes
1.6.7.1
HEAD
1.8

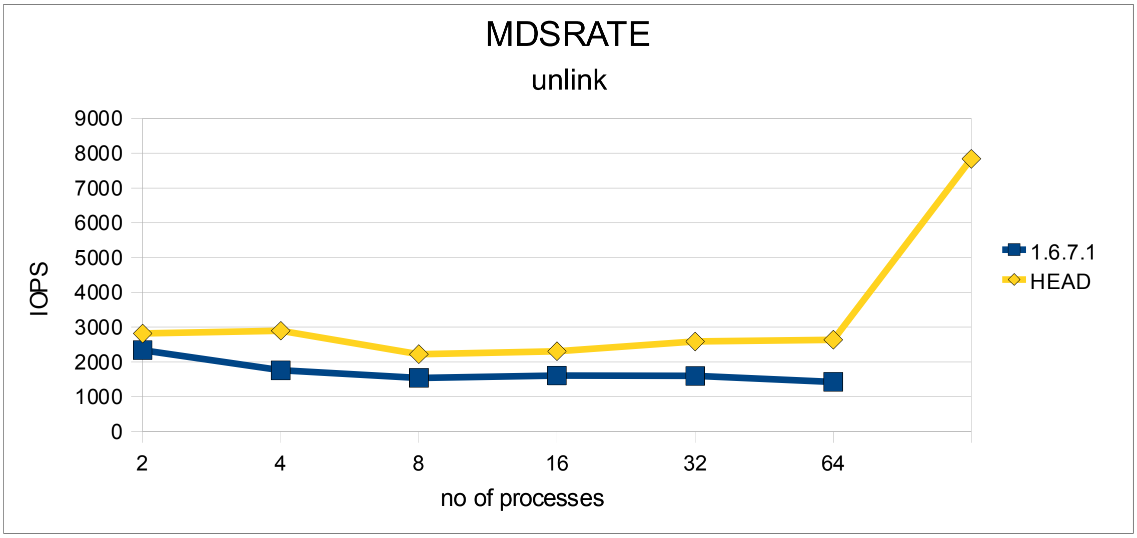

MDTEST
file create
IOPS
1000
800
600
400
200
0
2
4
8
16
32
64
no of processes
1.6.7.1
1.8
HEAD


MDTEST
file stat
IOPS
25000
20000
15000
10000
5000
0
2
4
8
16
32
64
no of processes
1.6.7.1
1.8
HEAD


MDTEST
file remove
IOPS
3000
2500
2000
1500
1000
500
0
2
4
8
16
32
64
no of processes
1.6.7.1
1.8
HEAD


MDSRATE
create
IOPS
3000
2500
2000
1500
1000
500
0
2
4
8
16
32
64
no of processes
1.6.7.1
HEAD


MDSRATE
stat
IOPS
50000
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
2
4
8
16
32
64
no of processes
1.6.7.1
HEAD


MDSRATE
unlink
IOPS
9000
8000
7000
6000
5000
4000
3000
2000
1000
0
2
4
8
16
32
64
no of processes
1.6.7.1
HEAD

Multi-client Numbers

1.6.7.1		METABENCH				
Operations Vs No of processes	**1**	**1**	**3**	**7**	**15**	**31**	**63**
create diff dir	2530.92	2503.41	6631.4	11520.49	14838.08	12675.66	12552.66
create same di	2570.83	2553.18	5822.98	4804.24	4771.8	4727.27	4975.05
STAT diff dir	5600.07	6153.29	17969.62	16492.97	25574.04	12556.65	12574.2
STAT same dir	5505.75	6348.24	6104.29	9243.97	15431.78	15396.31	15472.1
del diff dir	1129.57	1506.3	2905.95	5647.74	7221.43	6703.56	6891.34
del same dir	1058.55	1488.72	2424.18	2737.26	2724.41	2714.67	5414.55

		MDTEST					
Operations Vs No of processes	**1**	**2**	**4**	**8**	**16**	**32**	**64**
dir create	2450.89	3740.8	3775.5	3877.02	3687.56	3583.36	3591.88
dir stat	3624.91	7893.12	14653.73	20319.04	26250.06	25969.64	25607.55
dir remove	1390.76	1494.35	1650.69	1653.72	1649.07	1629.26	1603.85
file create	18.52	43.74	85.31	134.57	240.03	390.97	577.62
file stat	2183.13	4647.94	9153.08	12544.33	18217.67	18846.18	18648.59
file remove	1721.07	1318.95	1487.04	1162.53	1237.69	1253.97	1282.59

		MDSRATE					
Ops per sec Vs No of processes	**1**	**2**	**4**	**8**	**16**	**32**	**64**
create	195.57	344.95	823.29	1995.1	1356.4	1770.67	2103.83
stat	13858.41	25355.78	48294.2	93635.83	153186.56	31959.14	26002.98
unlink	2040.27	1788.24	1608.49	1340.69	1434.39	1328.6	1446.44

Single Client Numbers

1.6.7.1		METABENCH				
Operations Vs No of processes	**1**	**1**	**3**	**7**	**15**	**31**	**63**
create diff dir	2591.89	2535.28	3825.37	3105.24	3324.86	3176.41	
create same di	2602.79	2568.24	3468.23	2744.26	2698.32	2670.88	
STAT diff dir	6195.61	5499.11	5771.93	6828.31	6976.14	4985.14	
STAT same dir	5904.63	5303.08	4371.33	6168.92	7639.22	6405.82	
del diff dir	1024.06	1020.24	1799.39	2041.35	2213.75	2204.53	
del same dir	987.41	1022.35	1354.87	1344.18	1334.48	1280.88	

		MDTEST					
Operations Vs No of processes	**1**	**2**	**4**	**8**	**16**	**32**	**64**
dir create	2420.02	2109.42	1945.49	2119.6	1972	1999.72	
dir stat	3151.42	3381.14	2796.75	3010.86	2812.42	2536.93	
dir remove	1328.73	1127.37	1341.73	1275.41	1272.85	1217.39	
file create	19.56	23.36	33.77	33.17	35.76	35.47	
file stat	2095.88	3750.64	5602.21	8561.02	9334.2	6061.55	
file remove	1429.62	1576.3	1483.24	1312.29	1457.96	1163.61	

		MDSRATE					
Ops per sec Vs No of processes	**1**	**2**	**4**	**8**	**16**	**32**	**64**
create	173.36	155.98	161.74	153.37	147.9	151.96	
stat	6121.68	15490.99	17893.04	17873.49	17841.88	18229.56	
unlink	2338.46	1757.23	1541.22	1606.36	1597.04	1422.91	

Multi-client Numbers

1.8		METABENCH					
Operations Vs No of processes	**1**	**1**	**3**	**7**	**15**	**31**	**63**
create diff dir	2407.65	2467.97	6646.26	12479.9	15539.1	14724.06	12247.42
create same di	2440.59	2455.69	5644.92	4711.28	4884.37	4570.75	4767.24
STAT diff dir	5106.02	6038.07	12405.24	20161.14	12190.68	19169.27	10612.83
STAT same dir	5153.85	6250.56	6493.33	10408.76	16212.84	16458.41	16685.33
del diff dir	1089.56	1789.46	4538.2	7863.97	10229.74	9847.21	9035.06
del same dir	1105.48	1779.62	3016.73	3329.56	3398.84	3316.29	3335.29

		MDTEST					
Operations Vs No of processes	**1**	**2**	**4**	**8**	**16**	**32**	**64**
dir create	2355.35	3415.11	3811.99	3617.57	3474.58	3429.81	3413.25
dir stat	3552.07	7710.19	15112.14	20543	28818.61	28459.76	27663.21
dir remove	1430.87	1430.95	1557.04	1566.16	1564.57	1542	1522.01
file create	18.88	28.56	64.14	95.91	163.67	212.29	340.49
file stat	2259.6	4579.79	8804.02	15705.94	27655.1	24751.47	27142.92
file remove	1820.8	1260.78	1505.59	1498.14	1658.24	1672.79	1781.08

		MDSRATE					
Ops per sec Vs No of processes	**1**	**2**	**4**	**8**	**16**	**32**	**64**
create	32.89	136.66	265.62	402.14	234.41	358.83	567.18
stat	22289.17	36789.41	73146.83	133107.41	226421.53	40889.84	32026.89
unlink	1888.2	2061.04	2841.12	2275.05	3499.33	1769.84	1179.8

1.8.0 perf numbers

Single client numbers

1.8		METABENCH					
Operations Vs No of processes	**1**	**1**	**3**	**7**	**15**	**31**	**63**
create diff dir	1896.62	2057.21	3712.5	3254.94	3212.3	3210.56	
create same di	1905.48	2058.14	2993.33	2690.18	2590.27	2558.95	
STAT diff dir	3474.11	5984.28	6008.97	5529.29	5643.94	6751.81	
STAT same dir	3745.32	5901.75	3334.22	5345.75	9174.47	5707.22	
del diff dir	1046.14	1096.91	1992.14	2449.76	2612.14	2680.57	
del same dir	981.81	1127.93	1565.21	1562.49	1437.49	1504.4	

		MDTEST					
Operations Vs No of processes	1	2	4	8	16	32	64
dir create	2399.9	2119.97	1807.02	1848.84	1819.77	1799.83	
dir stat	3074.3	3281.65	2658.81	2697.82	2667.68	2668.12	
dir remove	1250.99	1223.46	1183.44	1195.13	1214.7	1141.29	
file create	19.68	22.45	26.08	25.39	29.41	29.93	
file stat	1967.16	3641.87	6345.05	9991.68	12055.66	11444.73	
file remove	1773.2	1693.18	1453.85	1190.43	1453.73	1459.82	

NOTE: 1.8 MDSRATE numbers haven't been collected.
Thus the graphs you see are for 1.6 and HEAD only

Multi-client Number

HEAD		METABENCH					
Operations Vs No of processes	1	1	3	7	15	31	63
create diff dir		3139.52	7107.3	11773.87	16390.9	11948.53	
create same dir		3021.82	6926.14	11048.5	12126.65	9548.81	
STAT diff dir		6186.86	17270.01	13763.58	16499.02	12864.15	
STAT same dir		2849.94	7822.7	14780.49	21201.54	19664.6	
del diff dir		2107.24	4928.36	7573.2	9003.77	6017.67	
del same dir		1618.37	3639.53	5496.01	6253.51		

		MDTEST					
Operations Vs No of processes	1	2	4	8	16	32	64
dir create	2705.19	3820.92	6970.55	7092.12	5602.78	7517.27	6095.82
dir stat	5424.69	11858.07	19785.32	25610.76	25778.78	17833.84	12750.14
dir remove	2275.02	3278.32	5298.73	6577.27	6496.66	6413.95	5661.03
file create	28.03	50.56	89.12	155.13	257.19	887.04	900.76
file stat	3220.86	7011.43	11857.57	20130.87	25016.13	20037.11	14031.7
file remove	2576.87	2816.2	4505.54	6176.25	7250.04	7014.45	4309.65

		MDSRATE					
Ops per sec Vs No of processes	1	2	4	8	16	32	64
create	2471.54	4538.32	7509.69	11208.99	11638.18	10970.24	11070.77
stat	20723.21	38718.54	71319.94	136389.06	237378.41	21183.94	14156.72
unlink	3102.04	3776.54	5709.29	3396.75	2331.78	2114.8	2224.84

2.0 perf numbers

Single Client numbers

HEAD		METABENCH				
Operations Vs No of processes	1	1	3	7	15	31	63
create diff dir	2991.55	2661.17	4391.4	4736.58	4296.04	4249.43	
create same di	3111.83	2759.02	2655.19	2217.16	2241.65	2267.82	
STAT diff dir	4240.56	4287.84	9748.93	13826.53	12711.5	6786.64	
STAT same dir	3236.69	3029.19	8227.08	14521.62	17932.49	17835.44	
del diff dir	856.38	866.99	2065.08	2289.87	2491.64	2490.41	
del same dir	1084.31	1182.76	1694.34	1656.77	1547.36		

		MDTEST				
Operations Vs No of processes	1	2	4	8	16	32	64
dir create	2892.38	2899.43	2615.3	2708.05	2696.35	2647.2	2499.35
dir stat	5702.13	5201.53	5070.14	5135.94	5262.92	4527.65	4407.73
dir remove	2263.01	2179.44	2102.84	2174.45	1974.58	1823.06	1800.66
file create	28.07	52.16	94.45	163.72	276.43	438.82	778.78
file stat	3204.8	5955.09	10723.58	12976.37	20040.93	19595.67	19131.52
file remove	2689.42	2680.86	2360.45	2047.2	2042.46	1839.56	2100.35

		MDSRATE				
Ops per sec Vs No of processes	1	2	4	8	16	32	64
create	2285.71	2714.29	2320.61	2303.03	2303.03	2268.66	2125.87
stat	20876.34	30145.27	40482.83	45482.41	39694.8	39435.95	39736.99
unlink	2814.81	2895.24	2223.06	2304.69	2582.87	2636.77	7838.84

Notes/Observations during the test runs
In case of MDTEST, due the test param “-y = option sync file after every write”, the runs took little longer than expected
Profiling is done with lustre-iokit/stats-collect. Dstat and oprofile has also been collected for most of the tests
There are some problems noted with 2.0 runs, especially METABENCH, a bug(19452) has been raised for the same.
There has been a soft-lock up bug(19398) seen during umount on 2.0.
The lustre versions used doesn't include Lian Zhen scalability patches for LNET.
No issues/problems seen on 1.8 or 1.6.7.1