

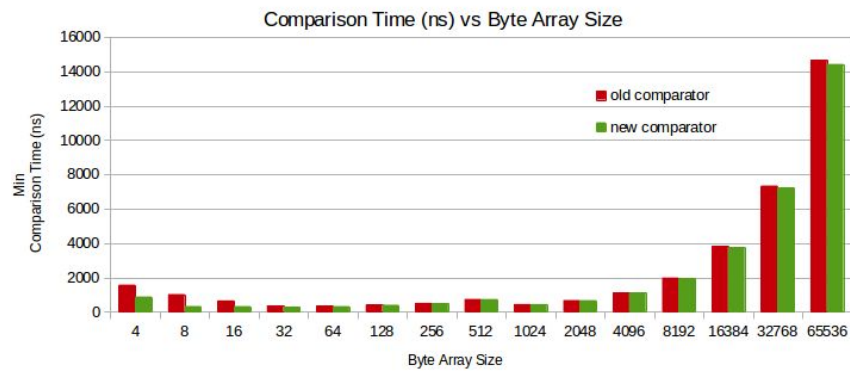
# Micro Benchmarks for byte array comparator

## Summary

- Results are calculated over 10,000 comparison iterations for each bytearray size
- New Comparator is substantially faster for byte array sizes from 4 to 64 / 128
- For larger byte array comparisons (4K onwards) new comparator time is equivalent to slightly slower in some cases

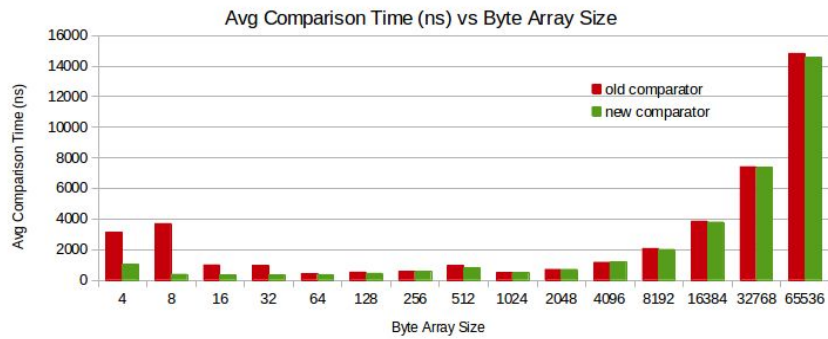
	old comparator time(ns)			new comparator time(ns)		
byte array size	min	avg	max	min	avg	max
4	1574	3147.35	55725	885	1046.48	6299
8	1024	3700.05	4071	328	386.83	572
16	660	1002.66	8919	322	361	512
32	361	981.95	13617	287	337.71	396
64	358	437.61	2082	322	369.94	580
128	413	531.91	1683	401	458.01	729
256	514	578.54	2148	503	575.06	1940
512	742	953.63	16815	735	833.25	3425
1024	446	484.58	580	446	484.75	705
2048	676	709.45	838	671	707.18	881
4096	1127	1171.34	1513	1110	1208.9	3750
8192	1989	2079.99	6849	1959	2010.16	2224
16384	3809	3859.63	4077	3740	3796.95	3867
32768	7341	7419.4	7808	7237	7376.28	11804
65536	14660	14834.28	19282	14420	14562.62	18475

**Min (10-68% faster for smaller arrays, almost equivalent for larger arrays)**



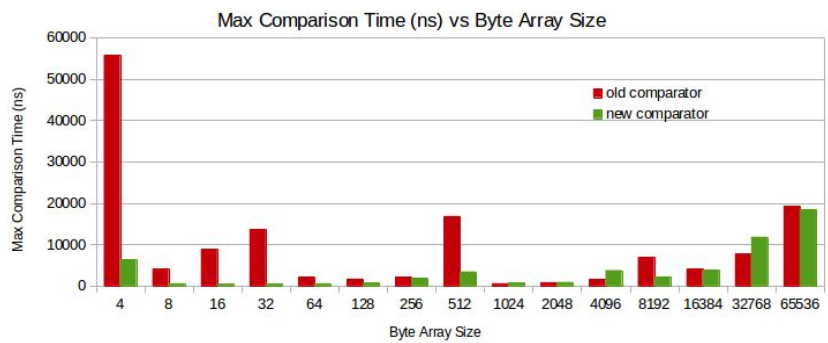
byte array size	old comparator time(ns)	new comparator time(ns)	%fast (new)
4	1574	885	44%
8	1024	328	68%
16	660	322	51%
32	361	287	20%
64	358	322	10%
128	413	401	3%
256	514	503	2%
512	742	735	1%
1024	446	446	0%
2048	676	671	1%
4096	1127	1110	2%
8192	1989	1959	2%
16384	3809	3740	2%
32768	7341	7237	1%
65536	14660	14420	2%

**Average (14-90% faster for smaller arrays, almost equivalent for larger arrays)**



byte array size	old comparator time(ns)	new comparator time(ns)	%fast (new)
4	3147.35	1046.48	67%
8	3700.05	386.83	90%
16	1002.66	361	64%
32	981.95	337.71	66%
64	437.61	369.94	15%
128	531.91	458.01	14%
256	578.54	575.06	1%
512	953.63	833.25	13%
1024	484.58	484.75	0%
2048	709.45	707.18	0%
4096	1171.34	1208.9	-3%
8192	2079.99	2010.16	3%
16384	3859.63	3796.95	2%
32768	7419.4	7376.28	1%
65536	14834.28	14562.62	2%

**Max (10-97% faster for smaller arrays, for larger arrays in some cases 5-150% slower)**

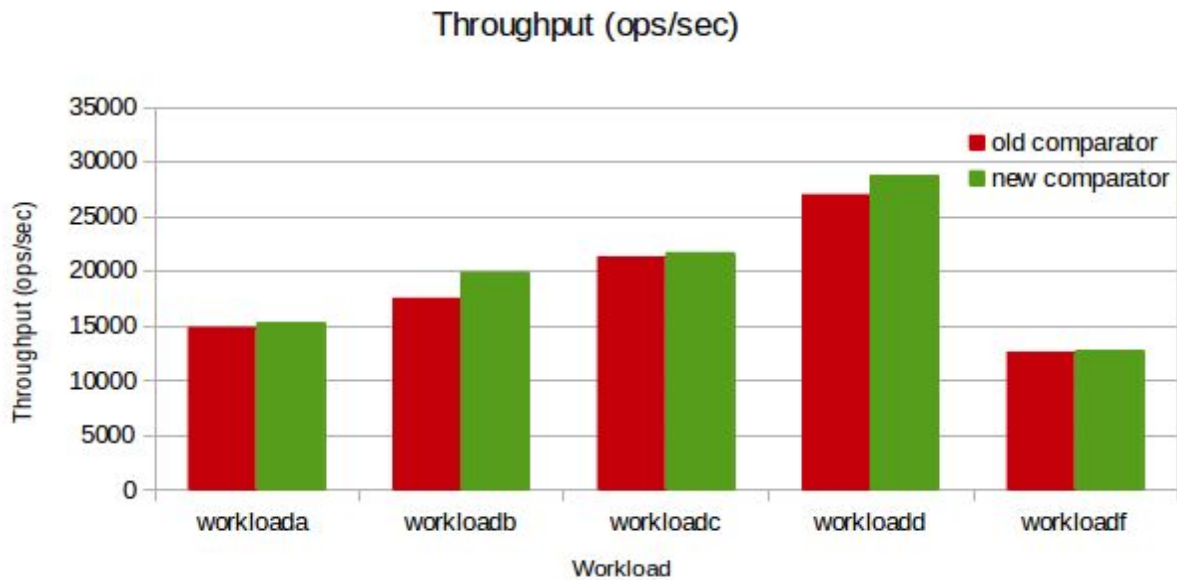


byte array size	old comparator time(ns)	new comparator time(ns)	%fast (new)
4	55725	6299	89%
8	4071	572	86%
16	8919	512	94%
32	13617	396	97%
64	2082	580	72%
128	1683	729	57%
256	2148	1940	10%
512	16815	3425	80%
1024	580	705	-22%
2048	838	881	-5%
4096	1513	3750	-148%
8192	6849	2224	68%
16384	4077	3867	5%
32768	7808	11804	-51%
65536	19282	18475	4%

# YCSB results

Test params -p recordcount=100000000 -p operationcount=10000000

Summary (1-14% higher throughput with new comparator)



		Runtime (ms)		Throughput (ops/sec)	
YCSB	description	old comparator	new comparator	old comparator	new comparator
workloada	50/50 reads and writes	683692	676404	14842	15235
workloadb	95/5 reads/write	608819	509912	17507	19879
workloadc	100% read.	471862	465034	21287	21687
workloadd	Read latest workload	370859	349403	26971	28750
workloadf	Read-modify-write	795686	792563	12585	12671

## Details (Results averaged over multiple iterations)

YCSB	commons-jar	RunTime (ms)	Throughput (ops/sec)
workloada	old	735603	13594.2
Iteration#2		572080	17480
Iteration#3		743394	13451.8
	avg	683692	14842
50/50 reads and writes	new	648895	15410.8
Iteration#2		546317	18304.3
Iteration#3		834001	11990.3
	avg	676404	15235
	improvement	7288ms	3%
workloadb	old	477110	20959.5
Iteration#2		508336	19672
Iteration#3		841011	11890.4
	avg	608819	17507
95/5 reads/write	new	452565	22096.2
Iteration#2		482457	20727.2
Iteration#3		594715	16814.7
	avg	509912	19879
	improvement	98907ms	14%
workloadc	old	433658	23059.6
Iteration#2		471231	21221
Iteration#3		510698	19581
	avg	471862	21287
100% read.	new	416822	23991
Iteration#2		456603	21900.8
Iteration#3		521677	19168.9
	avg	465034	21687
	improvement	6828ms	2%
workloadd	old	367803	27188.4
Iteration#2		365672	27346.9

Iteration#3		379102	26378.1
	avg	370859	26971
Read latest workload	new	321379	31115.9
Iteration#2		378888	26393
Iteration#3		347943	28740.3
	avg	349403	28750
	improvement	21456ms	7%
workloadf	old	754668	13250.8
Iteration#2		812423	12308.8
Iteration#3		819967	12195.6
	avg	795686	12585
Read-modify-write	new	725656	13780
Iteration#2		802105	12467.1
Iteration#3		849928	11765.7
	avg	792563	12671
	improvement	3123ms	1%