

www.phoronix-test-suite.com

spark tr

AMD Ryzen Threadripper 2990WX 32-Core testing with a ASUS ROG ZENITH EXTREME (2201 BIOS) and Gigabyte AMD Radeon RX 570 4GB on Ubuntu 22.04 via the Phoronix Test Suite.

Automated Executive Summary

B had the most wins, coming in first place for 59% of the tests.

Based on the geometric mean of all complete results, the fastest (B) was 1.08x the speed of the slowest (A).

The results with the greatest spread from best to worst included:

Apache Spark (Row Count: 10000000 - Partitions: 100 - Inner Join Test Time) at 1.699x
Apache Spark (Row Count: 10000000 - Partitions: 100 - Repartition Test Time) at 1.661x
Apache Spark (Row Count: 10000000 - Partitions: 100 - Broadcast Inner Join Test Time) at 1.642x
Apache Spark (Row Count: 20000000 - Partitions: 100 - Inner Join Test Time) at 1.587x
Apache Spark (Row Count: 20000000 - Partitions: 100 - Broadcast Inner Join Test Time) at 1.572x
Apache Spark (Row Count: 10000000 - Partitions: 500 - Broadcast Inner Join Test Time) at 1.547x
Apache Spark (Row Count: 10000000 - Partitions: 500 - Repartition Test Time) at 1.542x
Apache Spark (Row Count: 20000000 - Partitions: 100 - Repartition Test Time) at 1.504x
Apache Spark (Row Count: 10000000 - Partitions: 500 - Inner Join Test Time) at 1.482x
Apache Spark (Row Count: 1000000 - Partitions: 2000 - Repartition Test Time) at 1.447x.

Test Systems:

A

B

Processor: AMD Ryzen Threadripper 2990WX 32-Core @ 3.00GHz (32 Cores / 64 Threads), Motherboard: ASUS ROG ZENITH EXTREME (2201 BIOS), Chipset: AMD 17h, Memory: 32GB, Disk: Samsung SSD 970 EVO 500GB + 250GB Western Digital WDS250G2X0C-00L350, Graphics: Gigabyte AMD Radeon RX 570 4GB (1244/1750MHz), Audio: Realtek ALC1220, Monitor: MX279, Network: Intel I211 + Qualcomm Atheros QCA6174 802.11ac + Wilocity Wil6200 802.11ad

OS: Ubuntu 22.04, Kernel: 5.19.0-051900rc7-generic (x86_64), Desktop: GNOME Shell 42.2, Display Server: X Server + Wayland, OpenGL: 4.6 Mesa 22.0.1 (LLVM 13.0.1 DRM 3.47), Vulkan: 1.2.204, Compiler: GCC 11.2.0, File-System: ext4, Screen Resolution: 1920x1080

Kernel Notes: Transparent Huge Pages: madvise
Compiler Notes: --build=x86_64-linux-gnu --disable-vtable-verify --disable-werror --enable-bootstrap --enable-cet --enable-checking=release --enable-clocale=gnu --enable-default-pie --enable-gnu-unique-object --enable-languages=c,ada,c++,go,brig,d,fortran,objc,obj-c++,m2 --enable-libphobos-checking=release --enable-libstdcxx-debug --enable-libstdcxx-time=yes --enable-link-serialization=2 --enable-multiarch --enable-multilib --enable-nls --enable-objc-gc=auto --enable-offload-targets=nvptx-none=/build/gcc-11-gBFGDP/gcc-11-11.2.0/debian/tmp-nvptx/usr,amdgcn-amdhsa=/build/gcc-11-gBFGDP/gcc-11-11.2.0/debian/tmp-gcn/usr --enable-plugin --enable-shared --enable-threads=posix --host=x86_64-linux-gnu --program-prefix=x86_64-linux-gnu- --target=x86_64-linux-gnu --with-abi=m64 --with-arch-32=i686 --with-build-config=bootstrap-lto-lean --with-default-libstdcxx-abi=new --with-gcc-major-version-only --with-multilib-list=m32,m64,mx32 --with-target-system-zlib=auto --with-tune=generic --without-cuda-driver -v
Processor Notes: Scaling Governor: acpi-cpufreq schedutil (Boost: Enabled) - CPU Microcode: 0x800820d
Java Notes: OpenJDK Runtime Environment (build 11.0.15+10-Ubuntu-0ubuntu0.22.04.1)
Python Notes: Python 3.10.4
Security Notes: itlb_multihit: Not affected + l1tf: Not affected + mds: Not affected + meltdown: Not affected + mmio_stale_data: Not affected + retbleed: Mitigation of untrained return thunk; SMT vulnerable + spec_store_bypass: Mitigation of SSB disabled via prctl + spectre_v1: Mitigation of usercopy/swapgs barriers and __user pointer sanitization + spectre_v2: Mitigation of Retpolines IBPB: conditional STIBP: disabled RSB filling + srbds: Not affected + tsx_async_abort: Not affected

	A	B
Apache Spark - 40000000 - 100 - B.I.J.T.T (sec)	**43.02298204**	**31.92**
Normalized	74.19%	100%
Apache Spark - 40000000 - 100 - I.J.T.T (sec)	**43.76988138**	**35.77**
Normalized	81.72%	100%
Apache Spark - 40000000 - 100 - R.T.T (sec)	**40.979517765**	**30.69**
Normalized	74.89%	100%
Apache Spark - 40000000 - 100 - Group By Test Time (sec)	**31.884101243**	**29.730571283**
Normalized	93.25%	100%
Apache Spark - 40000000 - 100 - C.P.B.U.D (sec)	**4.43**	**4.44**
Normalized	100%	99.77%
Apache Spark - 40000000 - 100 - C.P.B (sec)	**62.136693946**	**61.95799001**
Normalized	99.71%	100%
Apache Spark - 40000000 - 100 - S.5.B.T (sec)	**43.910825271**	**46.03**
Normalized	100%	95.4%
Apache Spark - 40000000 - 500 - B.I.J.T.T (sec)	**35.542473101**	**33.950674349**
Normalized	95.52%	100%
Apache Spark - 40000000 - 500 - I.J.T.T (sec)	**38.805176603**	**34.60**
Normalized	89.16%	100%

spark tr

Apache Spark - 40000000 - 500 - R.T.T (sec)	35.936626267	31.040781082
Normalized	86.38%	100%
Apache Spark - 40000000 - 500 - Group By Test Time (sec)	33.81815958	33.168941005
Normalized	98.08%	100%
Apache Spark - 40000000 - 500 - C.P.B.U.D (sec)	4.53	4.79
Normalized	100%	94.57%
Apache Spark - 40000000 - 500 - C.P.B (sec)	62.40252554	62.332106635
Normalized	99.89%	100%
Apache Spark - 40000000 - 500 - S.5.B.T (sec)	46.704938593	48.940567345
Normalized	100%	95.43%
Apache Spark - 40000000 - 2000 - B.I.J.T.T (sec)	33.406302575	34.63
Normalized	100%	96.47%
Apache Spark - 40000000 - 2000 - I.J.T.T (sec)	36.89	37.377857332
Normalized	100%	98.69%
Apache Spark - 40000000 - 2000 - R.T.T (sec)	32.367778177	31.86
Normalized	98.43%	100%
Apache Spark - 40000000 - 2000 - Group By Test Time (sec)	32.790285393	31.62
Normalized	96.43%	100%
Apache Spark - 40000000 - 2000 - C.P.B.U.D (sec)	4.62	4.38
Normalized	94.81%	100%
Apache Spark - 40000000 - 2000 - C.P.B (sec)	61.990397429	62.13
Normalized	100%	99.78%
Apache Spark - 40000000 - 2000 - S.5.B.T (sec)	44.327354258	44.058118018
Normalized	99.39%	100%
Apache Spark - 40000000 - 1000 - B.I.J.T.T (sec)	31.340923075	33.61
Normalized	100%	93.25%
Apache Spark - 40000000 - 1000 - I.J.T.T (sec)	34.819192588	34.82076963
Normalized	100%	100%
Apache Spark - 40000000 - 1000 - R.T.T (sec)	31.12755649	32.348191608
Normalized	100%	96.23%
Apache Spark - 40000000 - 1000 - Group By Test Time (sec)	33.856953727	33.810705774
Normalized	99.86%	100%
Apache Spark - 40000000 - 1000 - C.P.B.U.D (sec)	4.43	4.42
Normalized	99.77%	100%
Apache Spark - 40000000 - 1000 - C.P.B (sec)	61.930554898	62.279023502
Normalized	100%	99.44%
Apache Spark - 40000000 - 1000 - S.5.B.T (sec)	44.735944448	44.49
Normalized	99.45%	100%
Apache Spark - 20000000 - 100 - B.I.J.T.T (sec)	26.429071267	16.81
Normalized	63.6%	100%
Apache Spark - 20000000 - 100 - I.J.T.T (sec)	28.857376046	18.178785652
Normalized	63%	100%
Apache Spark - 20000000 - 100 - R.T.T (sec)	27.337910713	18.18
Normalized	66.5%	100%
Apache Spark - 20000000 - 100 - Group By Test Time (sec)	14.753288032	13.83
Normalized	93.74%	100%
Apache Spark - 20000000 - 100 - C.P.B.U.D (sec)	4.46	4.46
Apache Spark - 20000000 - 100 - C.P.B (sec)	61.889646244	62.87
Normalized	100%	98.44%

spark tr

Apache Spark - 20000000 - 100 - S.5.B.T (sec)	31.155512825	25.802573232
Normalized	82.82%	100%
Apache Spark - 20000000 - 500 - B.I.J.T.T (sec)	23.545246024	17.11
Normalized	72.67%	100%
Apache Spark - 20000000 - 500 - I.J.T.T (sec)	24.462975514	17.636233312
Normalized	72.09%	100%
Apache Spark - 20000000 - 500 - R.T.T (sec)	23.552140763	17.014284711
Normalized	72.24%	100%
Apache Spark - 20000000 - 500 - Group By Test Time (sec)	14.918612745	14.69
Normalized	98.47%	100%
Apache Spark - 20000000 - 500 - C.P.B.U.D (sec)	4.52	4.488093463
Normalized	99.29%	100%
Apache Spark - 20000000 - 500 - C.P.B (sec)	62.025743606	62.074873258
Normalized	100%	99.92%
Apache Spark - 20000000 - 500 - S.5.B.T (sec)	30.738066039	24.016552673
Normalized	78.13%	100%
Apache Spark - 20000000 - 2000 - B.I.J.T.T (sec)	18.885806515	19.261739304
Normalized	100%	98.05%
Apache Spark - 20000000 - 2000 - I.J.T.T (sec)	20.098796966	19.49
Normalized	96.97%	100%
Apache Spark - 20000000 - 2000 - R.T.T (sec)	18.129978776	18.775430078
Normalized	100%	96.56%
Apache Spark - 20000000 - 2000 - Group By Test Time (sec)	16.642311492	15.041605602
Normalized	90.38%	100%
Apache Spark - 20000000 - 2000 - C.P.B.U.D (sec)	4.69	4.64
Normalized	98.93%	100%
Apache Spark - 20000000 - 2000 - C.P.B (sec)	62.508160211	62.105153355
Normalized	99.36%	100%
Apache Spark - 20000000 - 2000 - S.5.B.T (sec)	25.218516804	25.614620798
Normalized	100%	98.45%
Apache Spark - 20000000 - 1000 - B.I.J.T.T (sec)	17.533151598	18.23
Normalized	100%	96.18%
Apache Spark - 20000000 - 1000 - I.J.T.T (sec)	18.129485143	17.744465328
Normalized	97.88%	100%
Apache Spark - 20000000 - 1000 - R.T.T (sec)	17.444436259	18.60
Normalized	100%	93.79%
Apache Spark - 20000000 - 1000 - Group By Test Time (sec)	14.545897933	14.164335829
Normalized	97.38%	100%
Apache Spark - 20000000 - 1000 - C.P.B.U.D (sec)	4.635858138	4.65
Normalized	100%	99.7%
Apache Spark - 20000000 - 1000 - C.P.B (sec)	62.23774266	62.53
Normalized	100%	99.53%
Apache Spark - 20000000 - 1000 - S.5.B.T (sec)	24.55720381	25.13
Normalized	100%	97.72%
Apache Spark - 10000000 - 100 - B.I.J.T.T (sec)	15.102118238	9.20
Normalized	60.92%	100%
Apache Spark - 10000000 - 100 - I.J.T.T (sec)	17.083292218	10.053629991
Normalized	58.85%	100%
Apache Spark - 10000000 - 100 - R.T.T (sec)	16.178241522	9.740366086
Normalized	60.21%	100%

spark tr

Apache Spark - 10000000 - 100 - Group By Test Time (sec)	11.039362384	11.30869129
Normalized	100%	97.62%
Apache Spark - 10000000 - 100 - C.P.B.U.D (sec)	4.41	4.49
Normalized	100%	98.22%
Apache Spark - 10000000 - 100 - C.P.B (sec)	61.957279936	61.836282332
Normalized	99.8%	100%
Apache Spark - 10000000 - 100 - S.5.B.T (sec)	20.06391253	14.42
Normalized	71.87%	100%
Apache Spark - 10000000 - 500 - B.I.J.T.T (sec)	14.623652287	9.45
Normalized	64.62%	100%
Apache Spark - 10000000 - 500 - I.J.T.T (sec)	15.825329803	10.68
Normalized	67.49%	100%
Apache Spark - 10000000 - 500 - R.T.T (sec)	15.315967879	9.93
Normalized	64.83%	100%
Apache Spark - 10000000 - 500 - Group By Test Time (sec)	11.477514675	10.89
Normalized	94.88%	100%
Apache Spark - 10000000 - 500 - C.P.B.U.D (sec)	4.44	4.39
Normalized	98.87%	100%
Apache Spark - 10000000 - 500 - C.P.B (sec)	61.824250186	61.70
Normalized	99.8%	100%
Apache Spark - 10000000 - 500 - S.5.B.T (sec)	19.504123346	14.86
Normalized	76.19%	100%
Apache Spark - 10000000 - 2000 - B.I.J.T.T (sec)	11.376763629	11.97
Normalized	100%	95.04%
Apache Spark - 10000000 - 2000 - I.J.T.T (sec)	12.691831458	12.98
Normalized	100%	97.78%
Apache Spark - 10000000 - 2000 - R.T.T (sec)	10.668073982	10.94
Normalized	100%	97.51%
Apache Spark - 10000000 - 2000 - Group By Test Time (sec)	11.740644063	11.83
Normalized	100%	99.24%
Apache Spark - 10000000 - 2000 - C.P.B.U.D (sec)	4.444747603	4.53
Normalized	100%	98.12%
Apache Spark - 10000000 - 2000 - C.P.B (sec)	61.554074426	61.941561169
Normalized	100%	99.37%
Apache Spark - 10000000 - 2000 - S.5.B.T (sec)	15.295615927	15.97
Normalized	100%	95.78%
Apache Spark - 10000000 - 1000 - B.I.J.T.T (sec)	10.538025619	9.43
Normalized	89.49%	100%
Apache Spark - 10000000 - 1000 - I.J.T.T (sec)	10.950003509	11.04
Normalized	100%	99.18%
Apache Spark - 10000000 - 1000 - R.T.T (sec)	10.664056294	9.85
Normalized	92.37%	100%
Apache Spark - 10000000 - 1000 - Group By Test Time (sec)	10.691886696	10.913284688
Normalized	100%	97.97%
Apache Spark - 10000000 - 1000 - C.P.B.U.D (sec)	4.54	4.64
Normalized	100%	97.84%
Apache Spark - 10000000 - 1000 - C.P.B (sec)	61.57736862	61.86
Normalized	100%	99.54%

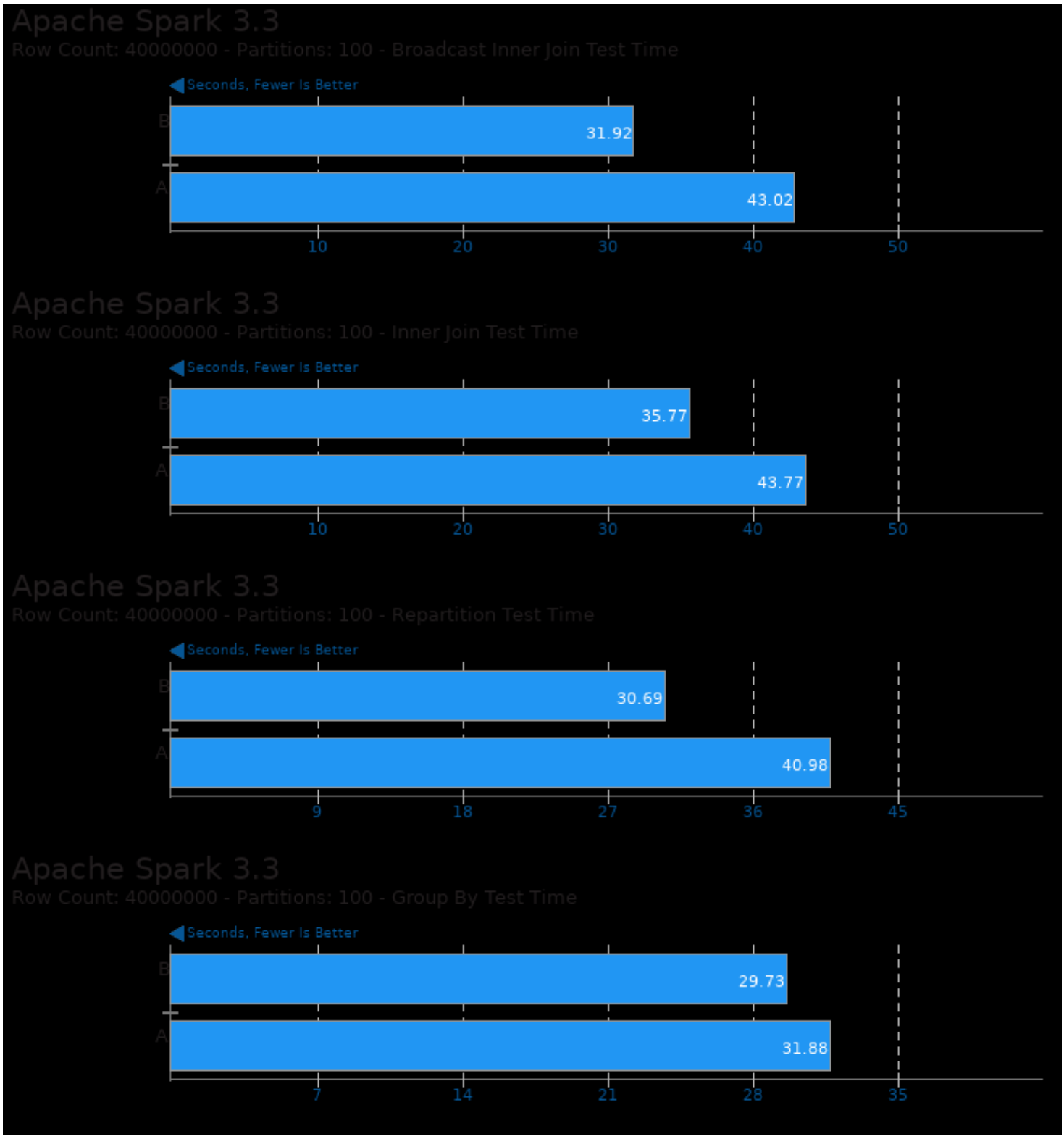
spark tr

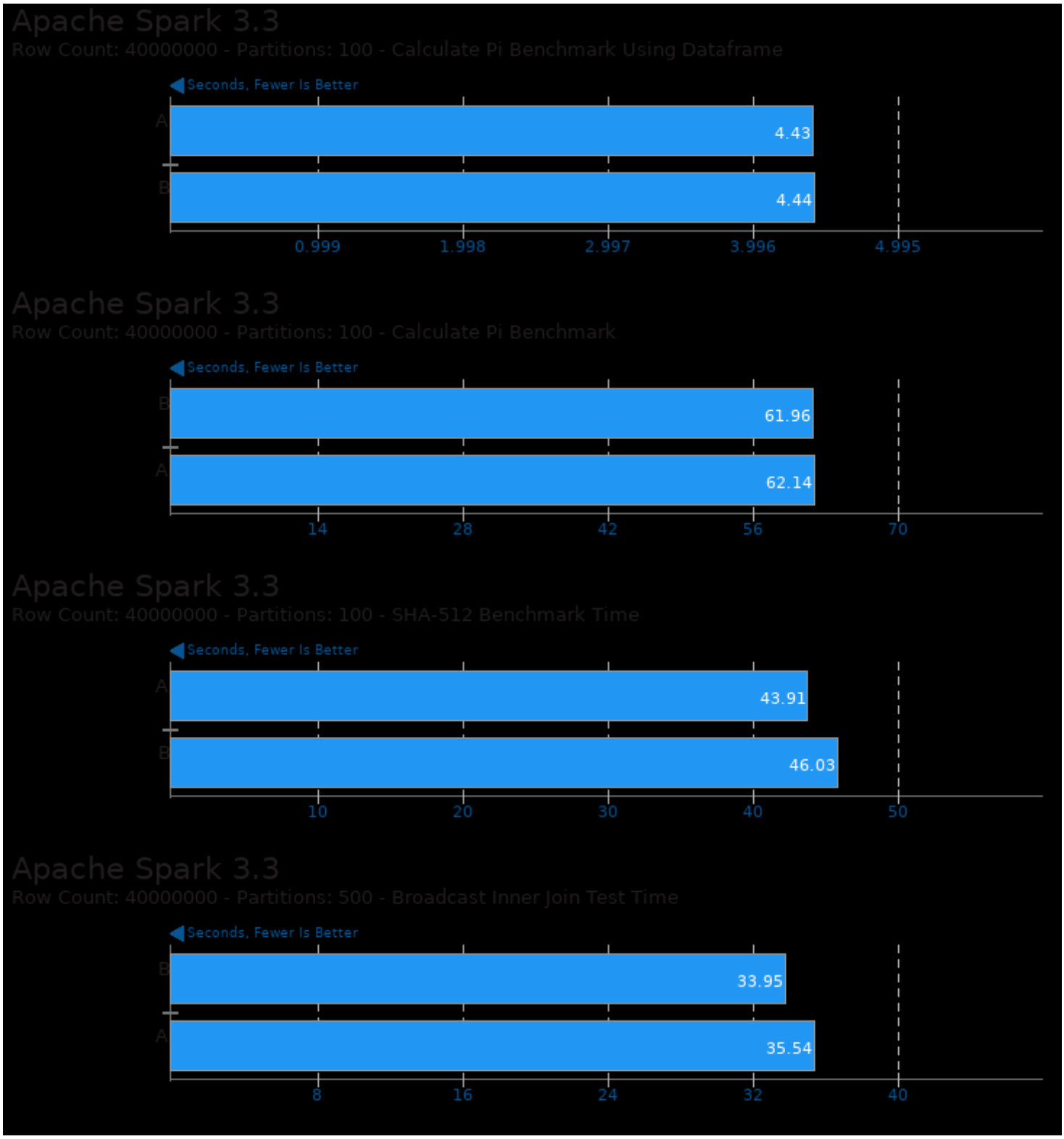
Apache Spark - 10000000 - 1000 - S.5.B.T (sec)	15.225220123	15.31529798
Normalized	100%	99.41%
SVT-AV1 - Preset 4 - Bosphorus 4K (FPS)	1.472	1.453
Normalized	100%	98.71%
Apache Spark - 1000000 - 2000 - B.I.J.T.T (sec)	4.015974333	3.16
Normalized	78.69%	100%
Apache Spark - 1000000 - 2000 - I.J.T.T (sec)	4.231091785	3.91
Normalized	92.41%	100%
Apache Spark - 1000000 - 2000 - R.T.T (sec)	6.15	4.25
Normalized	69.11%	100%
Apache Spark - 1000000 - 2000 - Group By Test Time (sec)	7.03	7.14
Normalized	100%	98.46%
Apache Spark - 1000000 - 2000 - C.P.B.U.D (sec)	4.50417597	4.37
Normalized	97.02%	100%
Apache Spark - 1000000 - 2000 - C.P.B (sec)	62.439872636	62.06
Normalized	99.39%	100%
Apache Spark - 1000000 - 2000 - S.5.B.T (sec)	5.99405918	5.74
Normalized	95.76%	100%
Apache Spark - 1000000 - 1000 - B.I.J.T.T (sec)	3.07	2.98
Normalized	97.07%	100%
Apache Spark - 1000000 - 1000 - I.J.T.T (sec)	3.47	3.22
Normalized	92.8%	100%
Apache Spark - 1000000 - 1000 - R.T.T (sec)	4.77	3.69
Normalized	77.36%	100%
Apache Spark - 1000000 - 1000 - Group By Test Time (sec)	6.69	6.67
Normalized	99.7%	100%
Apache Spark - 1000000 - 1000 - C.P.B.U.D (sec)	4.42	4.45
Normalized	100%	99.33%
Apache Spark - 1000000 - 1000 - C.P.B (sec)	62.428705739	61.558668692
Normalized	98.61%	100%
Apache Spark - 1000000 - 1000 - S.5.B.T (sec)	5.707415972	5.30
Normalized	92.86%	100%
Apache Spark - 1000000 - 500 - B.I.J.T.T (sec)	2.475242862	2.30
Normalized	92.92%	100%
Apache Spark - 1000000 - 500 - I.J.T.T (sec)	3.75	3.11
Normalized	82.93%	100%
Apache Spark - 1000000 - 500 - R.T.T (sec)	4.448490734	3.65
Normalized	82.05%	100%
Apache Spark - 1000000 - 500 - Group By Test Time (sec)	7.00	6.58
Normalized	94%	100%
Apache Spark - 1000000 - 500 - C.P.B.U.D (sec)	4.442555283	4.43
Normalized	99.72%	100%
Apache Spark - 1000000 - 500 - C.P.B (sec)	61.739942656	61.906716893
Normalized	100%	99.73%
Apache Spark - 1000000 - 500 - S.5.B.T (sec)	5.77	5.30
Normalized	91.85%	100%
Apache Spark - 1000000 - 100 - B.I.J.T.T (sec)	2.36	1.85
Normalized	78.39%	100%
Apache Spark - 1000000 - 100 - I.J.T.T (sec)	3.07470943	2.85
Normalized	92.69%	100%

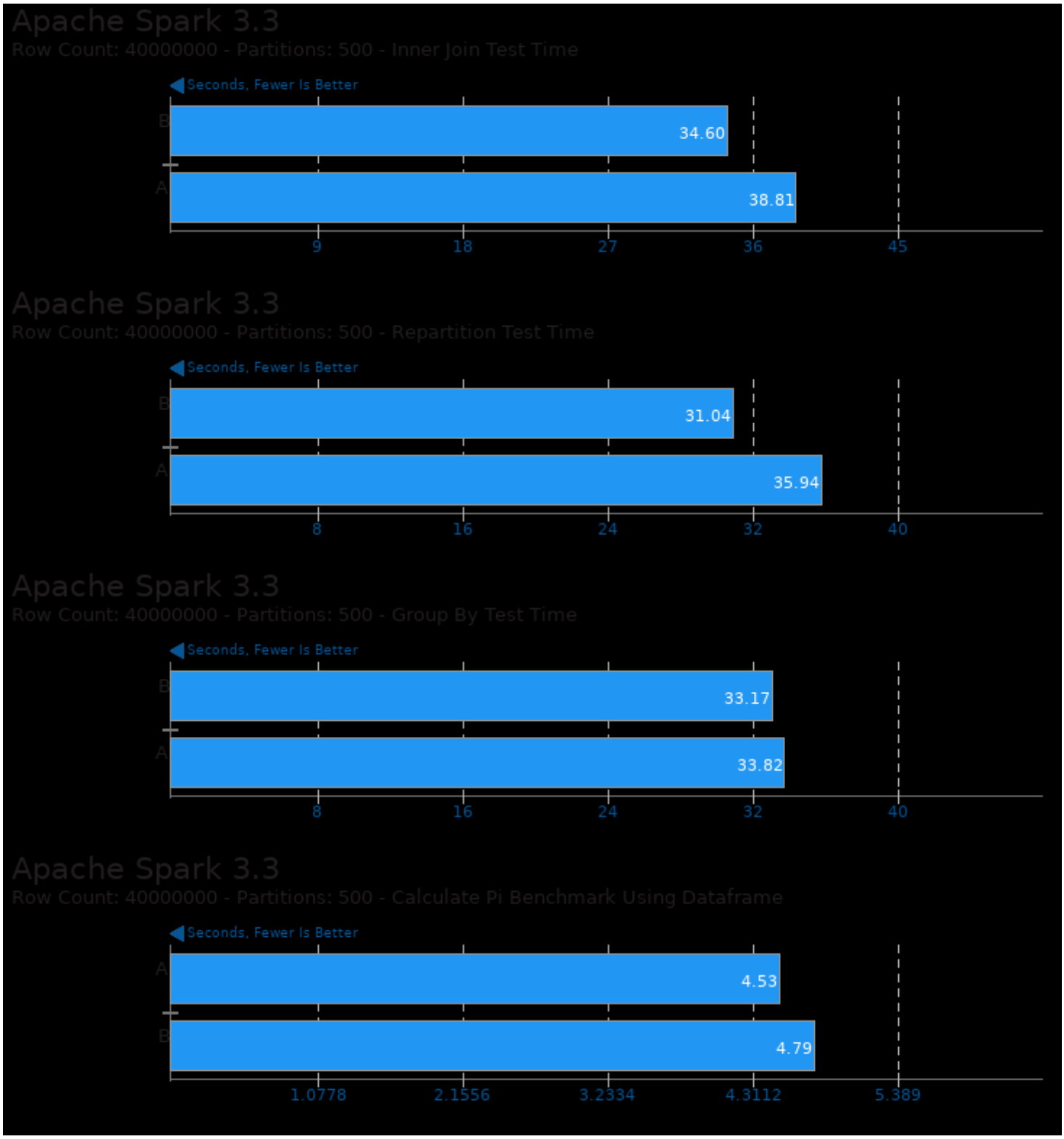
spark tr

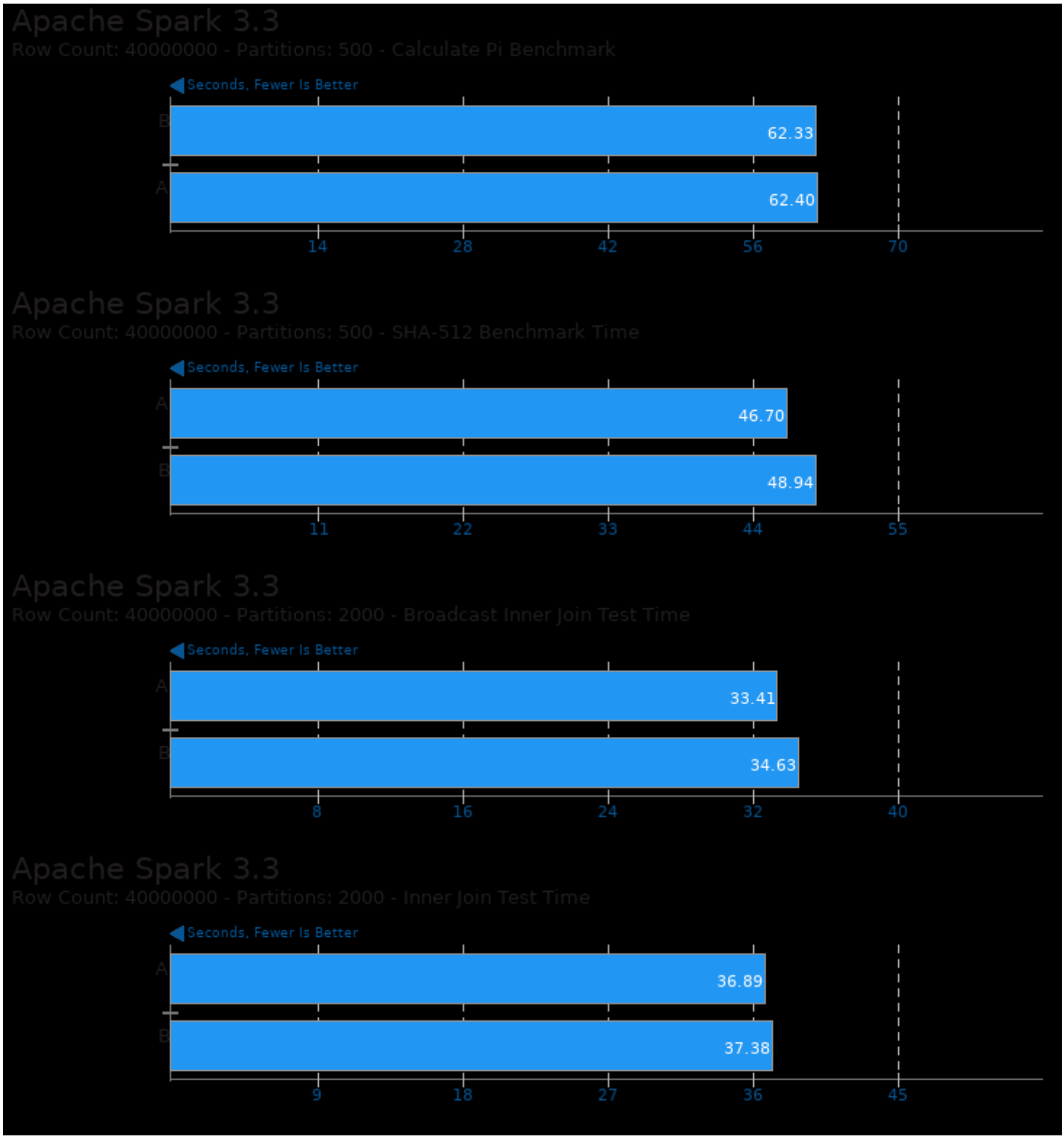
Apache Spark - 1000000 - 100 - R.T.T (sec)	**4.01**	**3.20**
Normalized	79.8%	100%
Apache Spark - 1000000 - 100 - Group By Test Time (sec)	**6.479692439**	**6.63**
Normalized	100%	97.73%
Apache Spark - 1000000 - 100 - C.P.B.U.D (sec)	**4.33**	**4.38**
Normalized	100%	98.86%
Apache Spark - 1000000 - 100 - C.P.B (sec)	**61.131040206**	**61.43**
Normalized	100%	99.51%
Apache Spark - 1000000 - 100 - S.5.B.T (sec)	**5.13080919**	**5.35**
Normalized	100%	95.9%
SVT-AV1 - Preset 4 - Bosphorus 1080p (FPS)	**3.568**	**3.547**
Normalized	100%	99.41%
SVT-AV1 - Preset 8 - Bosphorus 4K (FPS)	**28.875**	**28.259**
Normalized	100%	97.87%
SVT-AV1 - Preset 10 - Bosphorus 4K (FPS)	**48.462**	**50.882**
Normalized	95.24%	100%
SVT-AV1 - Preset 12 - Bosphorus 4K (FPS)	**60.767**	**61.899**
Normalized	98.17%	100%
SVT-AV1 - Preset 8 - Bosphorus 1080p (FPS)	**72.267**	**67.743**
Normalized	100%	93.74%
SVT-AV1 - Preset 10 - Bosphorus 1080p (FPS)	**150.077**	**142.495**
Normalized	100%	94.95%
SVT-AV1 - Preset 12 - Bosphorus 1080p (FPS)	**233.906**	**201.802**
Normalized	100%	86.27%

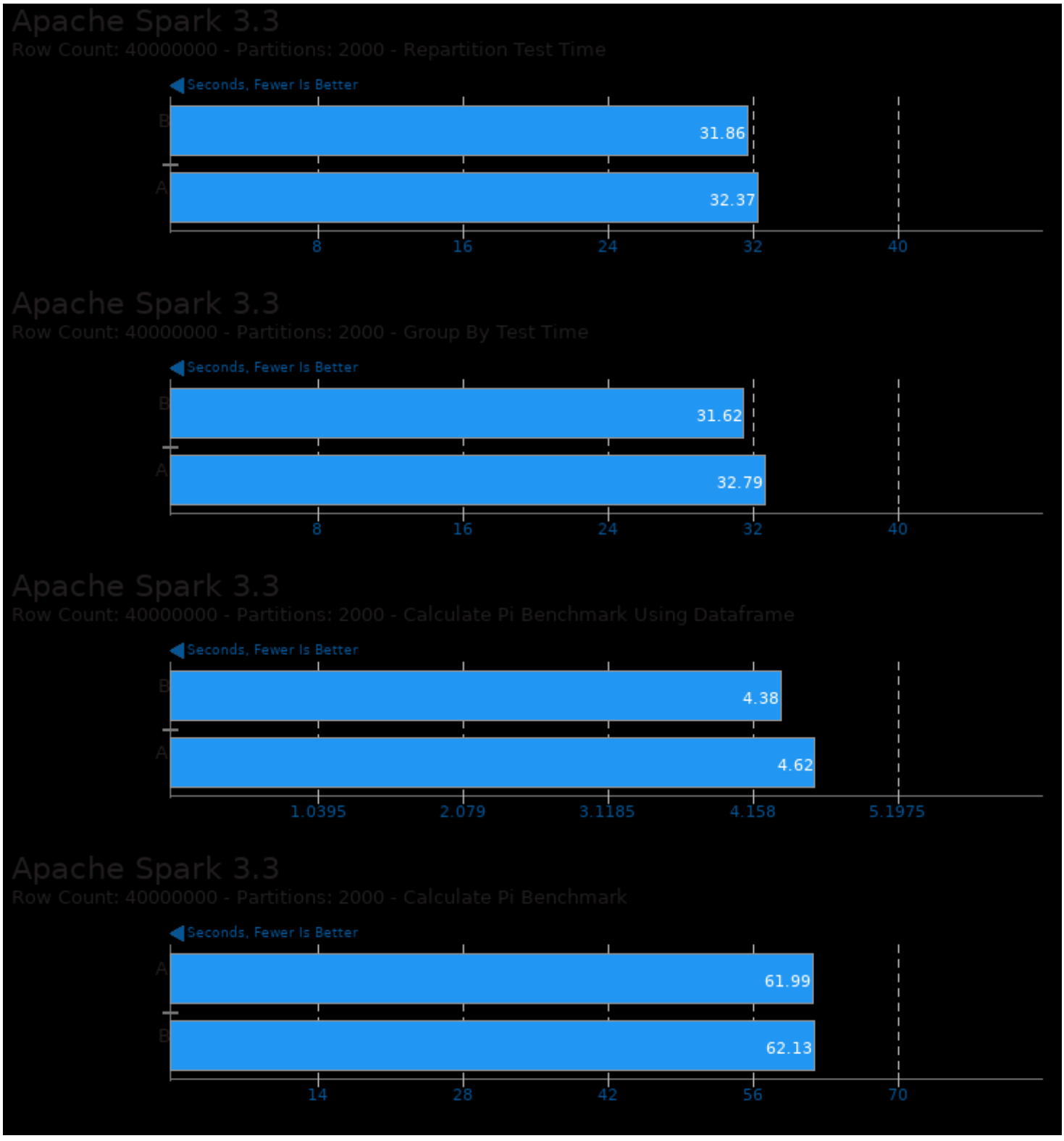
spark tr

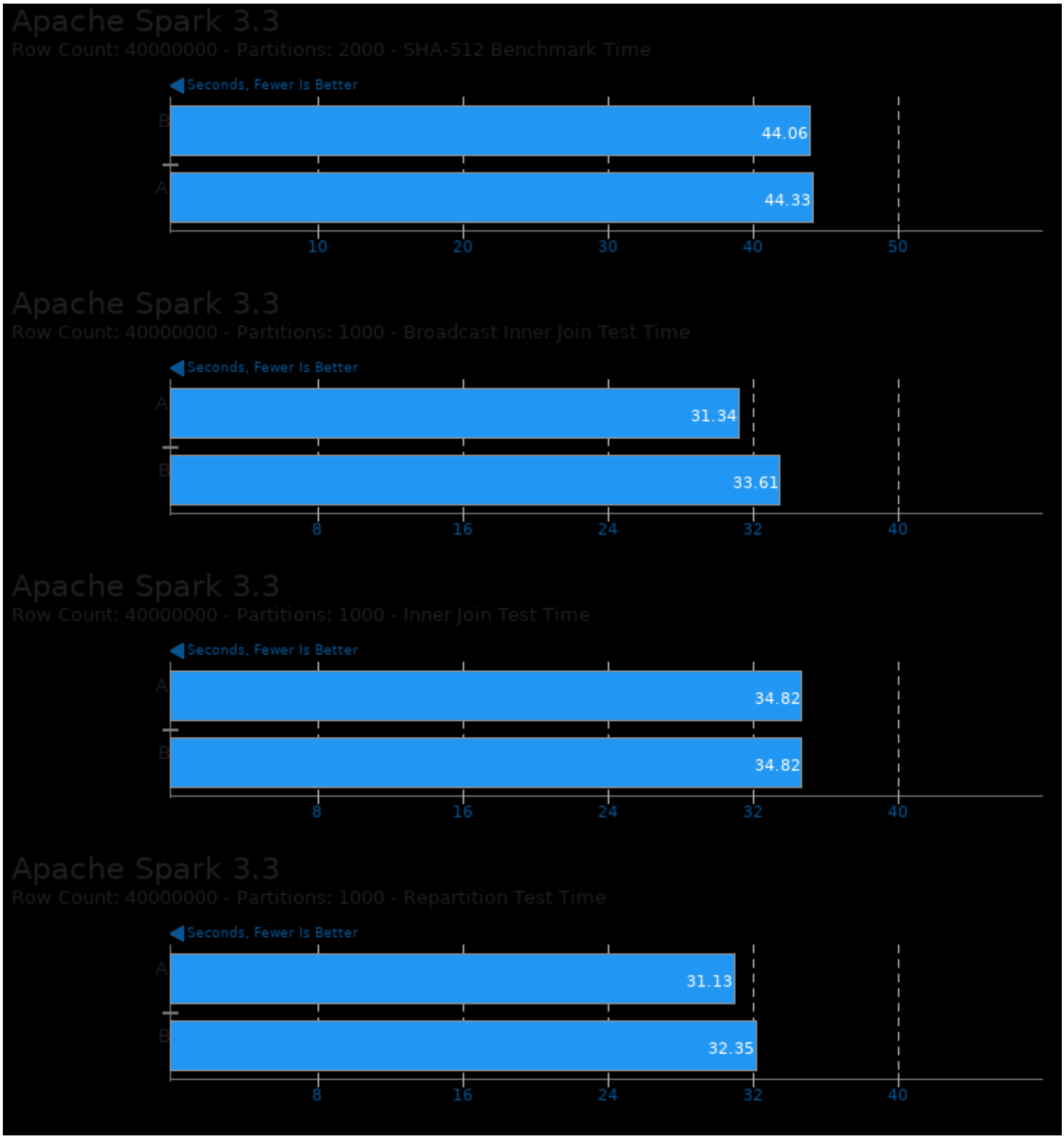

Apache Spark 3.3
Row Count: 40000000 - Partitions: 100 - Broadcast Inner Join Test Time
Seconds, Fewer Is Better
B
31.92
A
43.02
10
20
30
40
50
Apache Spark 3.3
Row Count: 40000000 - Partitions: 100 - Inner Join Test Time
Seconds, Fewer Is Better
B
35.77
A
43.77
10
20
30
40
50
Apache Spark 3.3
Row Count: 40000000 - Partitions: 100 - Repartition Test Time
Seconds, Fewer Is Better
B
30.69
A
40.98
9
18
27
36
45
Apache Spark 3.3
Row Count: 40000000 - Partitions: 100 - Group By Test Time
Seconds, Fewer Is Better
B
29.73
A
31.88
7
14
21
28
35


Apache Spark 3.3
Row Count: 40000000 - Partitions: 100 - Calculate Pi Benchmark Using Dataframe
Seconds, Fewer Is Better
A
4.43
B
4.44
0.999
1.998
2.997
3.996
4.995
Apache Spark 3.3
Row Count: 40000000 - Partitions: 100 - Calculate Pi Benchmark
Seconds, Fewer Is Better
B
61.96
A
62.14
14
28
42
56
70
Apache Spark 3.3
Row Count: 40000000 - Partitions: 100 - SHA-512 Benchmark Time
Seconds, Fewer Is Better
A
43.91
B
46.03
10
20
30
40
50
Apache Spark 3.3
Row Count: 40000000 - Partitions: 500 - Broadcast Inner Join Test Time
Seconds, Fewer Is Better
B
33.95
A
35.54
8
16
24
32
40

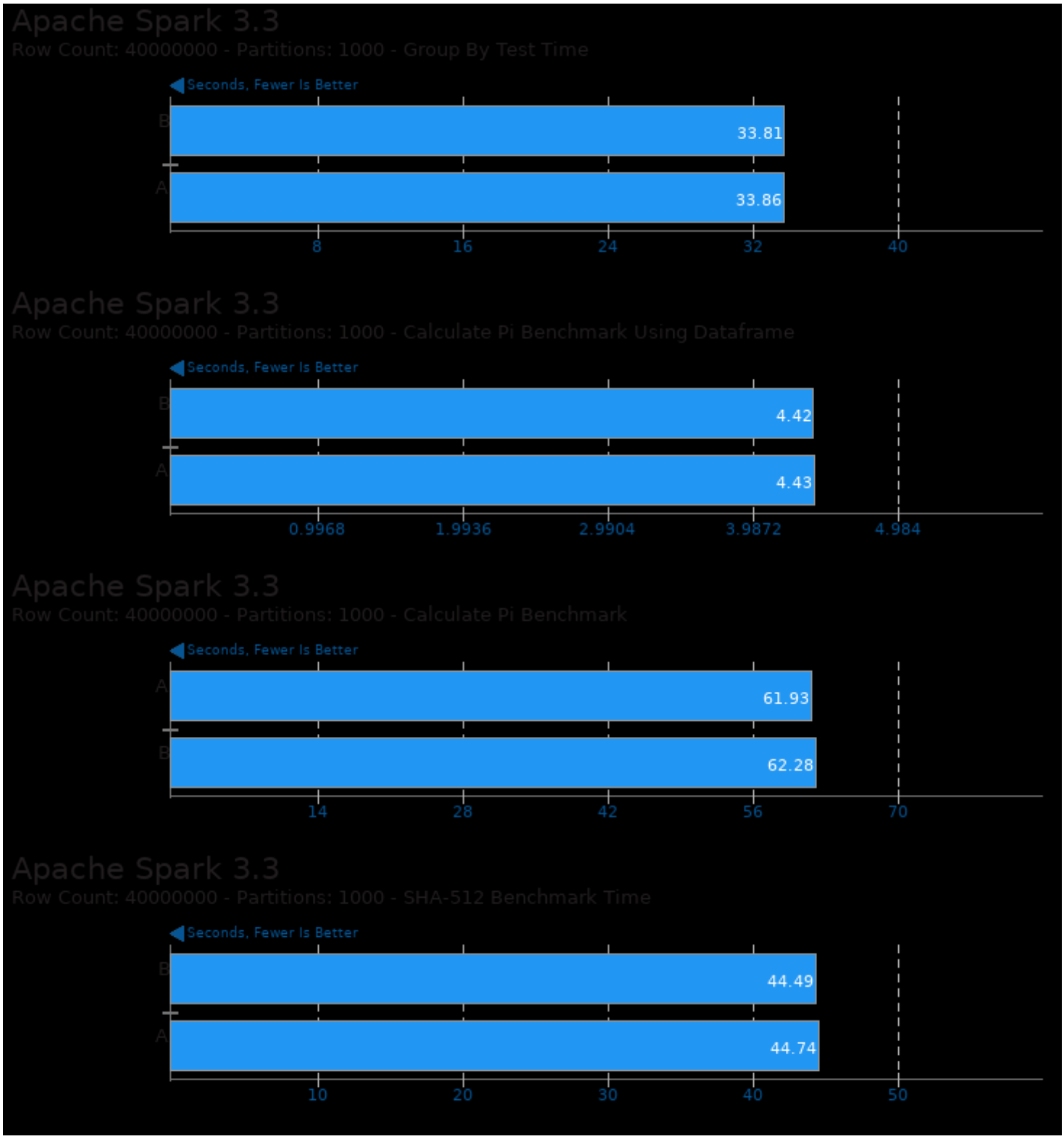

Apache Spark 3.3
Row Count: 40000000 - Partitions: 500 - Inner Join Test Time
Seconds, Fewer Is Better
B
34.60
A
38.81
9
18
27
36
45
Apache Spark 3.3
Row Count: 40000000 - Partitions: 500 - Repartition Test Time
Seconds, Fewer Is Better
B
31.04
A
35.94
8
16
24
32
40
Apache Spark 3.3
Row Count: 40000000 - Partitions: 500 - Group By Test Time
Seconds, Fewer Is Better
B
33.17
A
33.82
8
16
24
32
40
Apache Spark 3.3
Row Count: 40000000 - Partitions: 500 - Calculate Pi Benchmark Using Dataframe
Seconds, Fewer Is Better
A
4.53
B
4.79
1.0778
2.1556
3.2334
4.3112
5.389

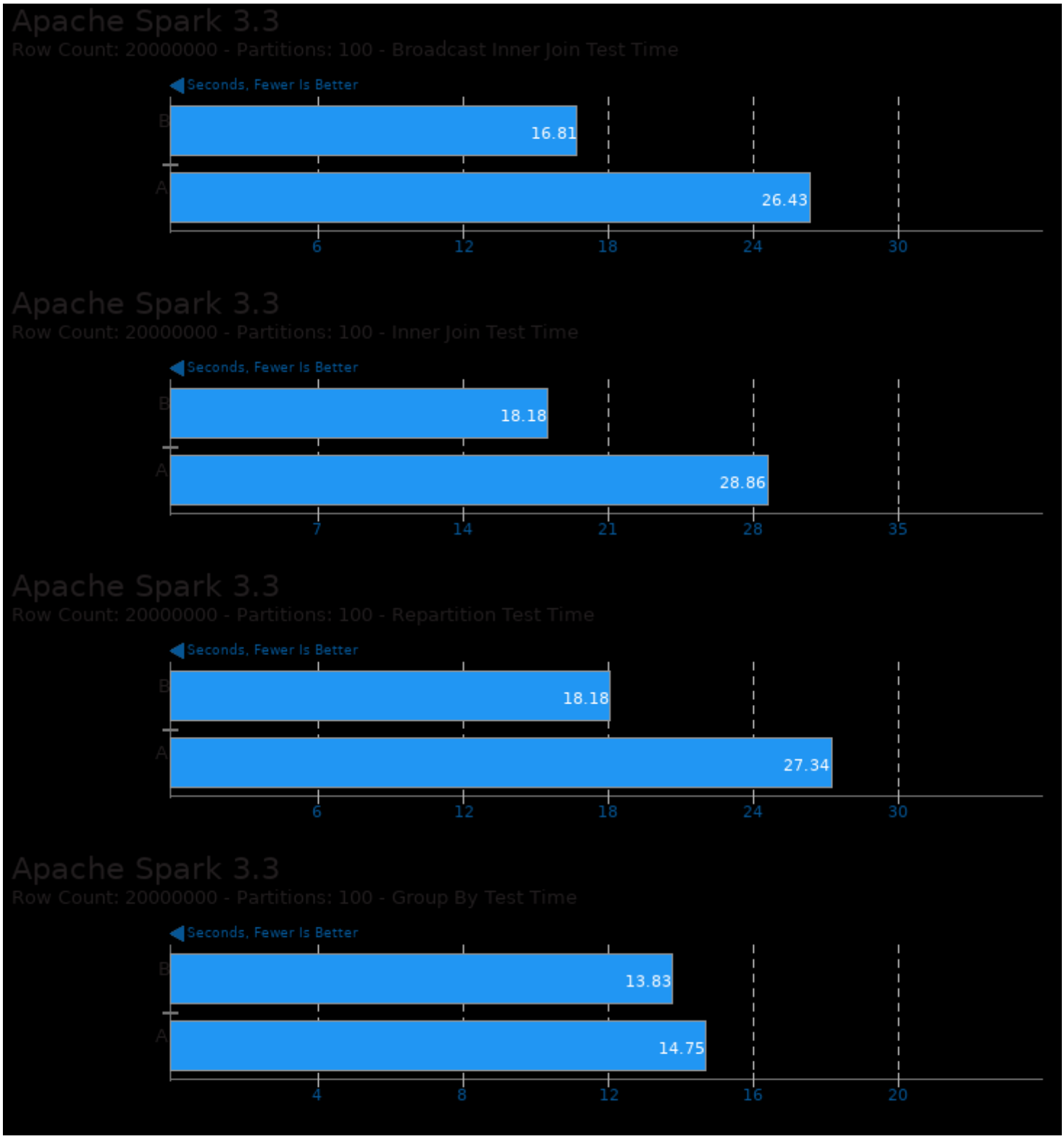

Apache Spark 3.3
Row Count: 40000000 - Partitions: 500 - Calculate Pi Benchmark
Seconds, Fewer Is Better
B
62.33
A
62.40
14
28
42
56
70
Apache Spark 3.3
Row Count: 40000000 - Partitions: 500 - SHA-512 Benchmark Time
Seconds, Fewer Is Better
A
46.70
B
48.94
11
22
33
44
55
Apache Spark 3.3
Row Count: 40000000 - Partitions: 2000 - Broadcast Inner Join Test Time
Seconds, Fewer Is Better
A
33.41
B
34.63
8
16
24
32
40
Apache Spark 3.3
Row Count: 40000000 - Partitions: 2000 - Inner Join Test Time
Seconds, Fewer Is Better
A
36.89
B
37.38
9
18
27
36
45


Apache Spark 3.3
Row Count: 40000000 - Partitions: 2000 - Repartition Test Time
Seconds, Fewer Is Better
B
31.86
A
32.37
8
16
24
32
40
Apache Spark 3.3
Row Count: 40000000 - Partitions: 2000 - Group By Test Time
Seconds, Fewer Is Better
B
31.62
A
32.79
8
16
24
32
40
Apache Spark 3.3
Row Count: 40000000 - Partitions: 2000 - Calculate Pi Benchmark Using Dataframe
Seconds, Fewer Is Better
B
4.38
A
4.62
1.0395
2.079
3.1185
4.158
5.1975
Apache Spark 3.3
Row Count: 40000000 - Partitions: 2000 - Calculate Pi Benchmark
Seconds, Fewer Is Better
A
61.99
B
62.13
14
28
42
56
70

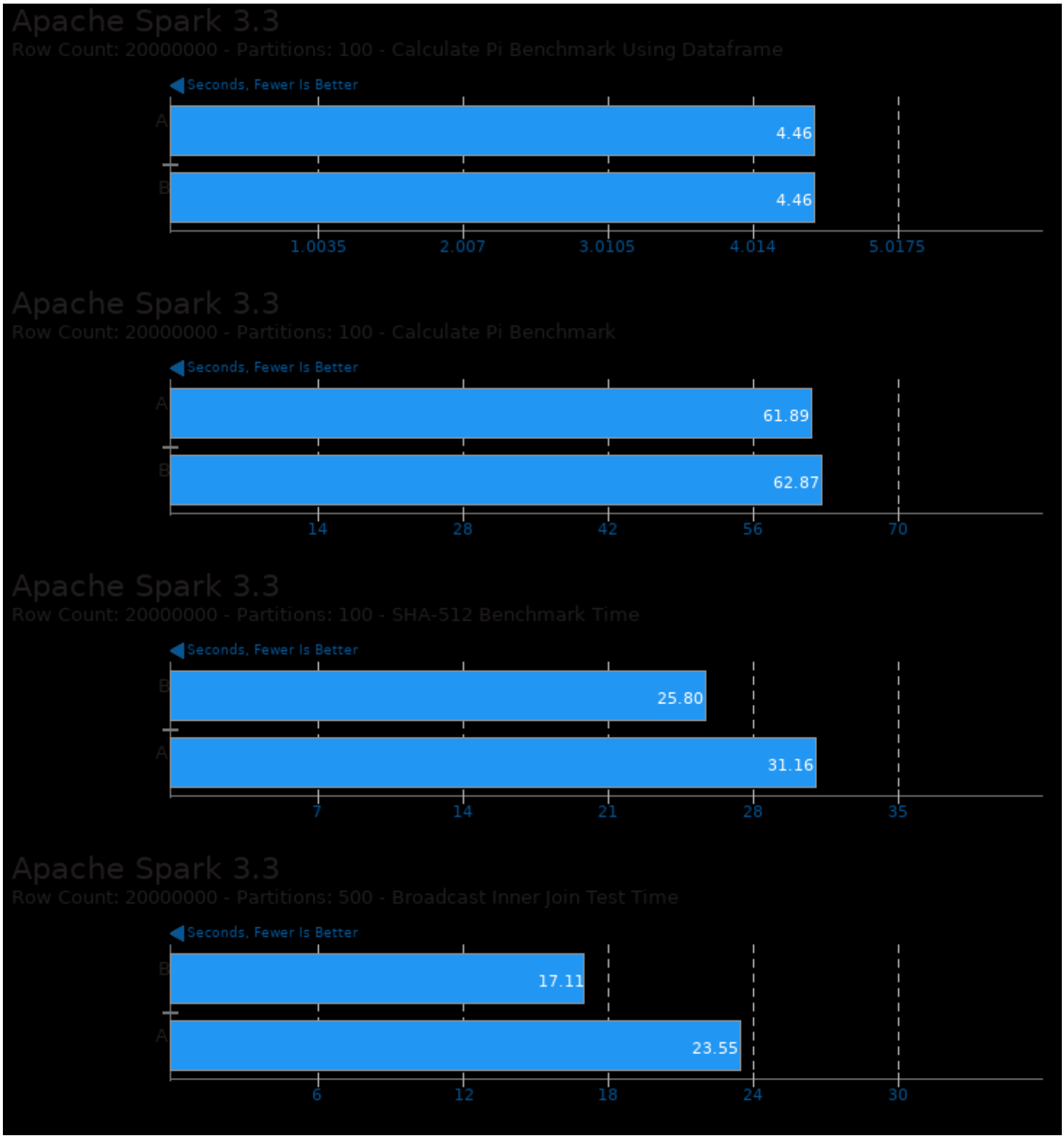

Apache Spark 3.3
Row Count: 40000000 - Partitions: 2000 - SHA-512 Benchmark Time
Seconds, Fewer Is Better
B
44.06
A
44.33
10
20
30
40
50
Apache Spark 3.3
Row Count: 40000000 - Partitions: 1000 - Broadcast Inner Join Test Time
Seconds, Fewer Is Better
A
31.34
B
33.61
8
16
24
32
40
Apache Spark 3.3
Row Count: 40000000 - Partitions: 1000 - Inner Join Test Time
Seconds, Fewer Is Better
A
34.82
B
34.82
8
16
24
32
40
Apache Spark 3.3
Row Count: 40000000 - Partitions: 1000 - Repartition Test Time
Seconds, Fewer Is Better
A
31.13
B
32.35
8
16
24
32
40

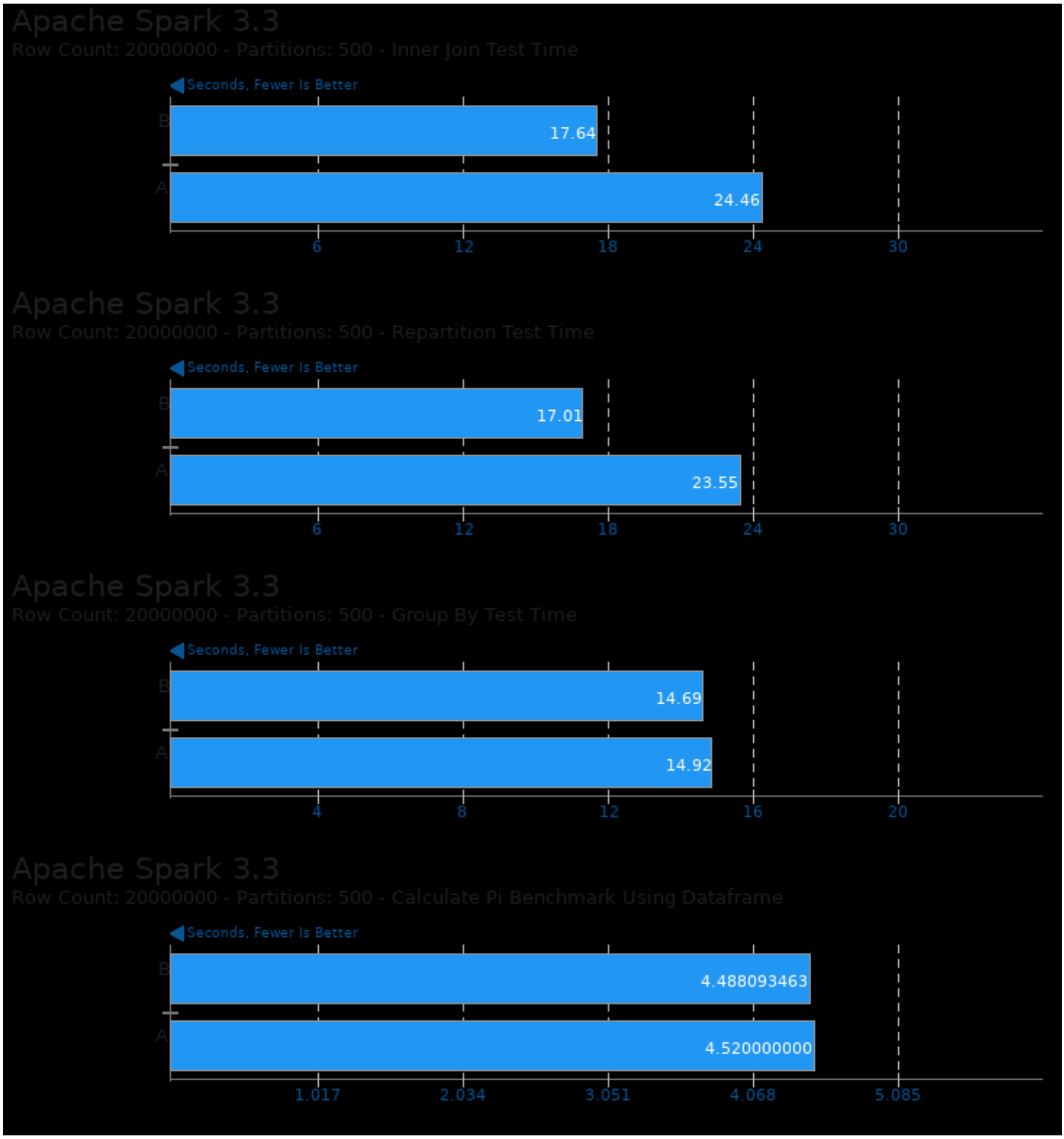
spark tr

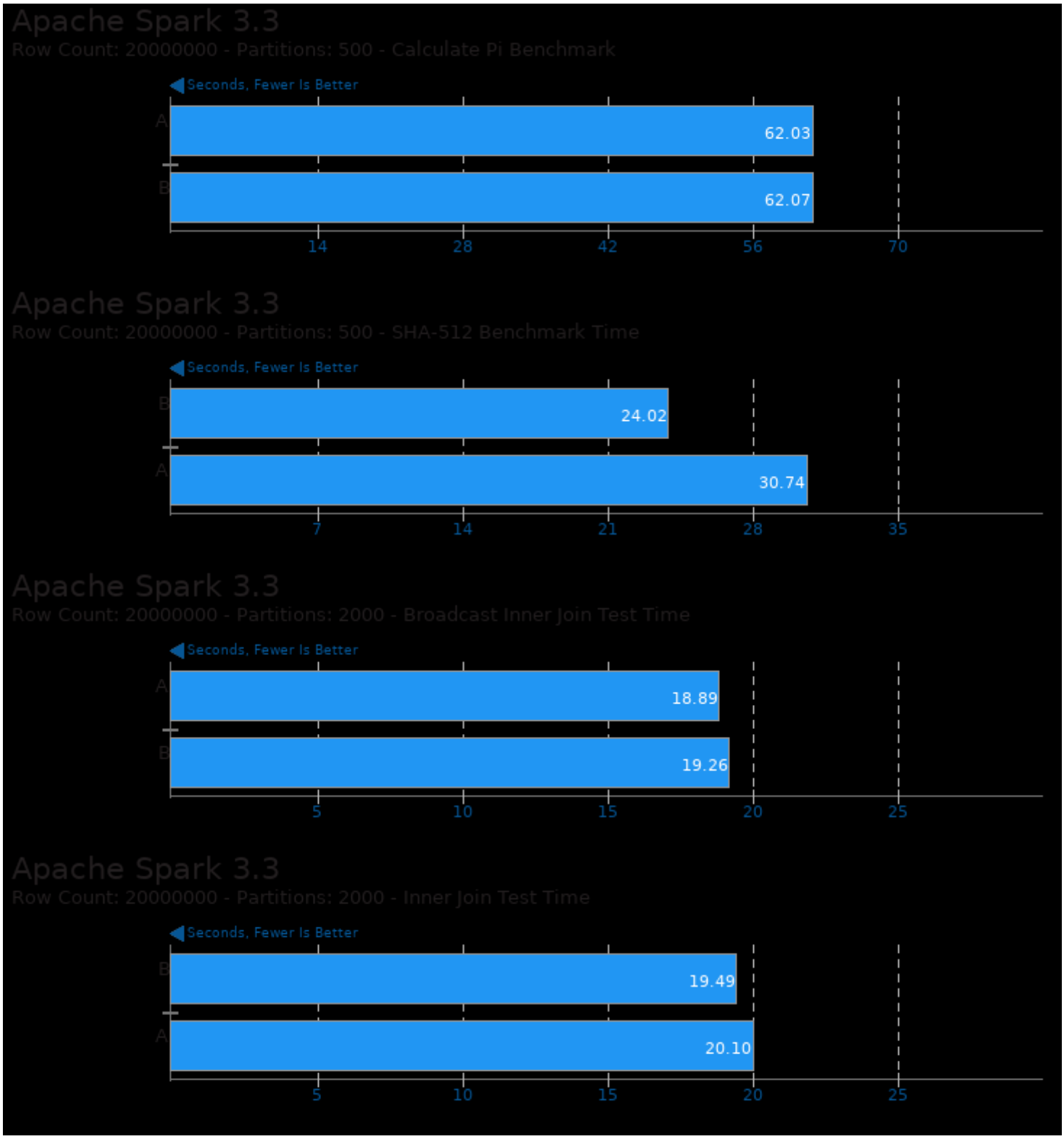

Apache Spark 3.3
Row Count: 40000000 - Partitions: 1000 - Group By Test Time
Seconds, Fewer Is Better
B
33.81
A
33.86
8
16
24
32
40
Apache Spark 3.3
Row Count: 40000000 - Partitions: 1000 - Calculate Pi Benchmark Using Dataframe
Seconds, Fewer Is Better
B
4.42
A
4.43
0.9968
1.9936
2.9904
3.9872
4.984
Apache Spark 3.3
Row Count: 40000000 - Partitions: 1000 - Calculate Pi Benchmark
Seconds, Fewer Is Better
A
61.93
B
62.28
14
28
42
56
70
Apache Spark 3.3
Row Count: 40000000 - Partitions: 1000 - SHA-512 Benchmark Time
Seconds, Fewer Is Better
B
44.49
A
44.74
10
20
30
40
50

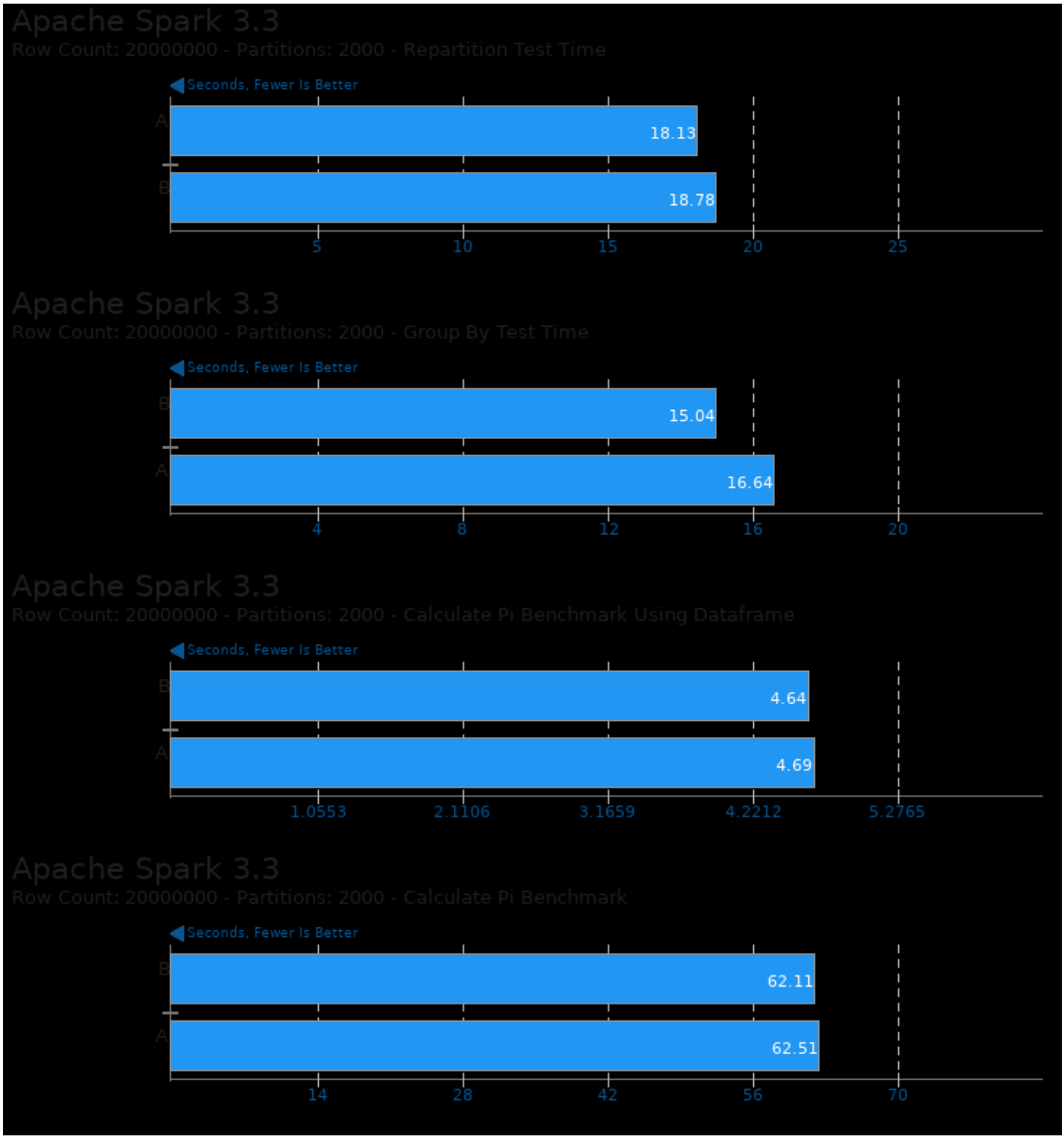

Apache Spark 3.3
Row Count: 20000000 - Partitions: 100 - Broadcast Inner Join Test Time
Seconds, Fewer Is Better
B
16.81
A
26.43
6
12
18
24
30
Apache Spark 3.3
Row Count: 20000000 - Partitions: 100 - Inner Join Test Time
Seconds, Fewer Is Better
B
18.18
A
28.86
7
14
21
28
35
Apache Spark 3.3
Row Count: 20000000 - Partitions: 100 - Repartition Test Time
Seconds, Fewer Is Better
B
18.18
A
27.34
6
12
18
24
30
Apache Spark 3.3
Row Count: 20000000 - Partitions: 100 - Group By Test Time
Seconds, Fewer Is Better
B
13.83
A
14.75
4
8
12
16
20

spark tr

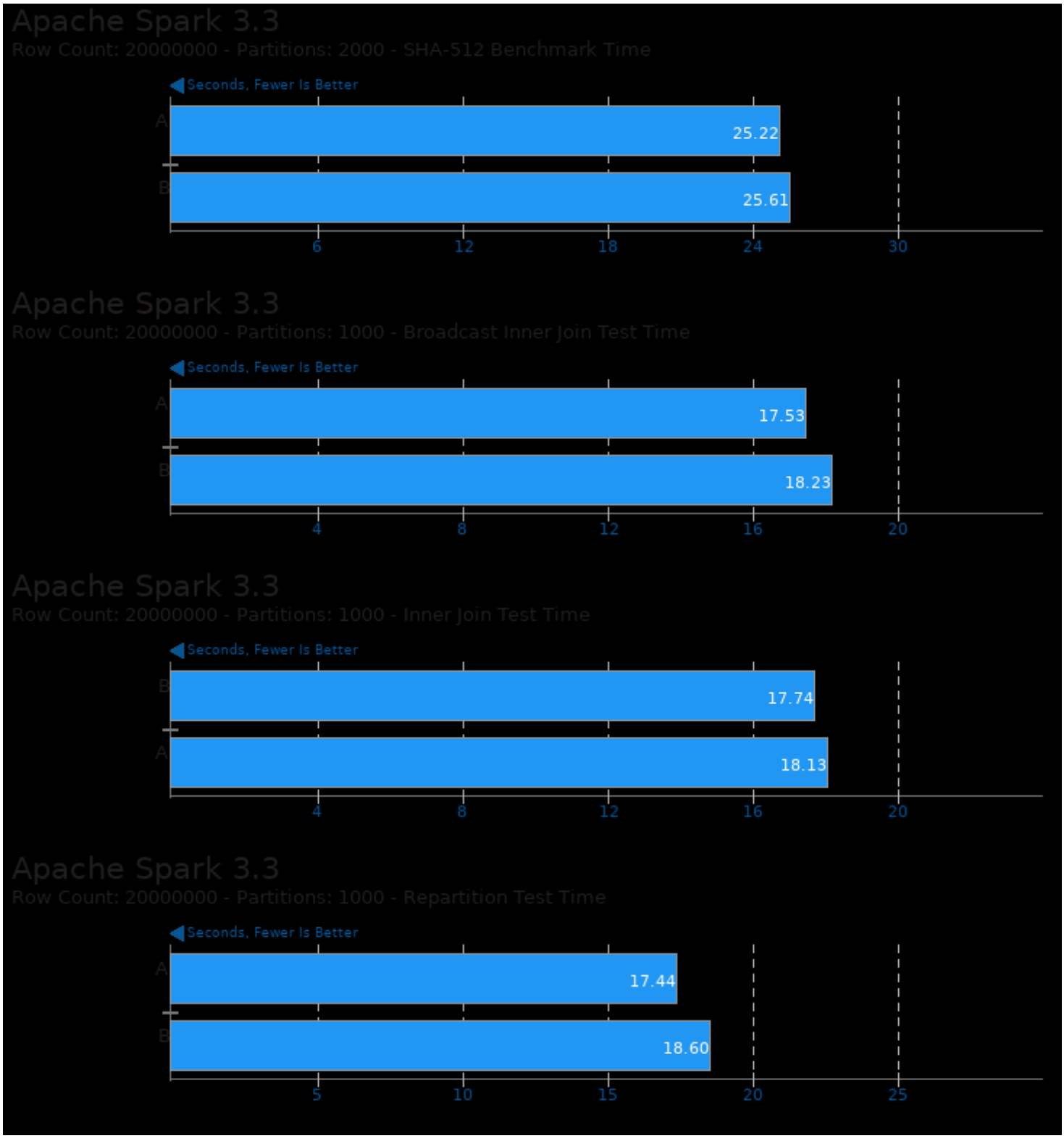

Apache Spark 3.3
Row Count: 20000000 - Partitions: 100 - Calculate Pi Benchmark Using Dataframe
Seconds, Fewer Is Better
A
4.46
B
4.46
1.0035
2.007
3.0105
4.014
5.0175
Apache Spark 3.3
Row Count: 20000000 - Partitions: 100 - Calculate Pi Benchmark
Seconds, Fewer Is Better
A
61.89
B
62.87
14
28
42
56
70
Apache Spark 3.3
Row Count: 20000000 - Partitions: 100 - SHA-512 Benchmark Time
Seconds, Fewer Is Better
B
25.80
A
31.16
7
14
21
28
35
Apache Spark 3.3
Row Count: 20000000 - Partitions: 500 - Broadcast Inner Join Test Time
Seconds, Fewer Is Better
B
17.11
A
23.55
6
12
18
24
30


Apache Spark 3.3
Row Count: 20000000 - Partitions: 500 - Inner Join Test Time
Seconds, Fewer Is Better
B
17.64
A
24.46
6
12
18
24
30
Apache Spark 3.3
Row Count: 20000000 - Partitions: 500 - Repartition Test Time
Seconds, Fewer Is Better
B
17.01
A
23.55
6
12
18
24
30
Apache Spark 3.3
Row Count: 20000000 - Partitions: 500 - Group By Test Time
Seconds, Fewer Is Better
B
14.69
A
14.92
4
8
12
16
20
Apache Spark 3.3
Row Count: 20000000 - Partitions: 500 - Calculate Pi Benchmark Using Dataframe
Seconds, Fewer Is Better
B
4.488093463
A
4.520000000
1.017
2.034
3.051
4.068
5.085

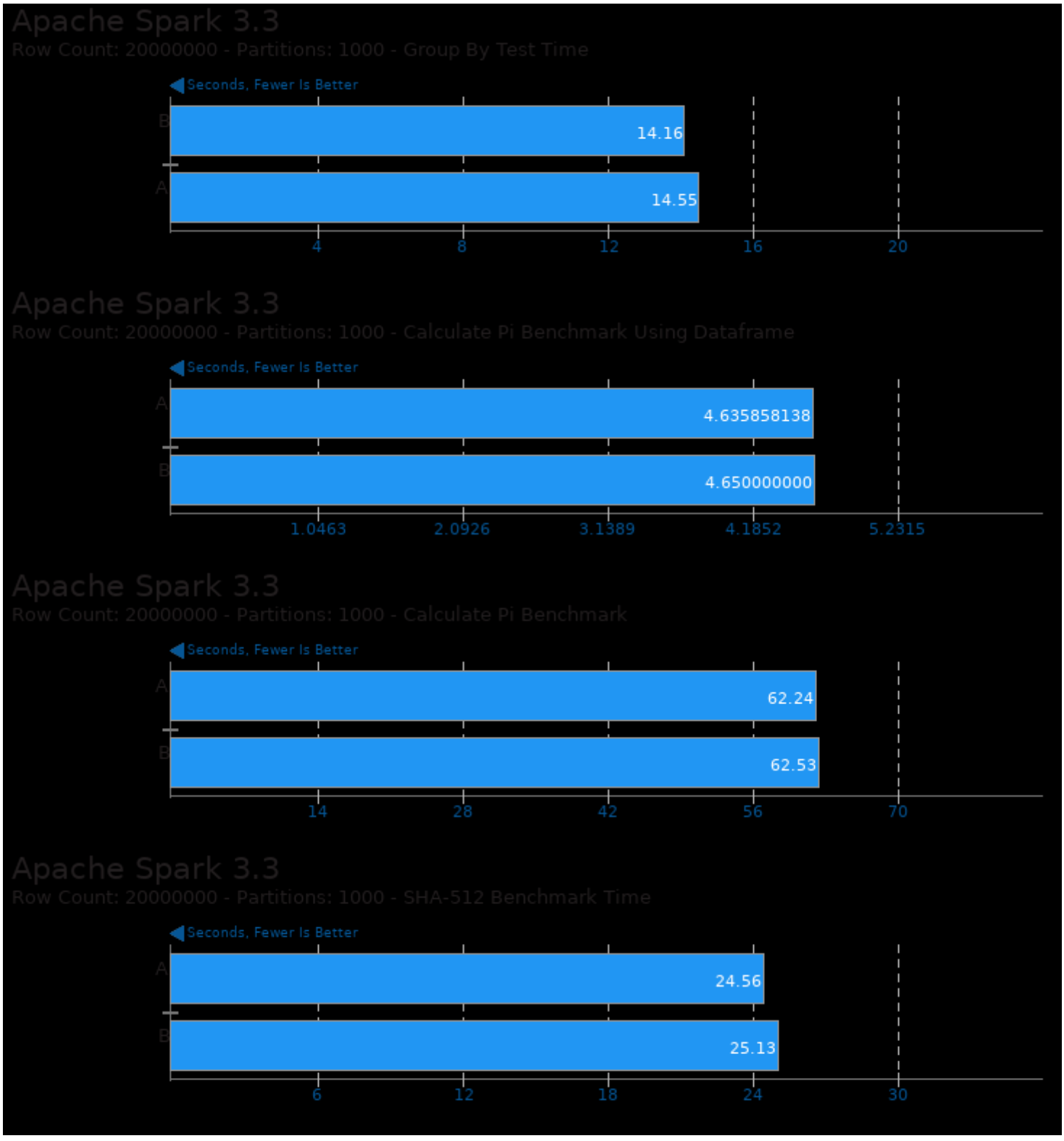

Apache Spark 3.3
Row Count: 20000000 - Partitions: 500 - Calculate Pi Benchmark
Seconds, Fewer Is Better
A
62.03
B
62.07
14
28
42
56
70
Apache Spark 3.3
Row Count: 20000000 - Partitions: 500 - SHA-512 Benchmark Time
Seconds, Fewer Is Better
B
24.02
A
30.74
7
14
21
28
35
Apache Spark 3.3
Row Count: 20000000 - Partitions: 2000 - Broadcast Inner Join Test Time
Seconds, Fewer Is Better
A
18.89
B
19.26
5
10
15
20
25
Apache Spark 3.3
Row Count: 20000000 - Partitions: 2000 - Inner Join Test Time
Seconds, Fewer Is Better
B
19.49
A
20.10
5
10
15
20
25

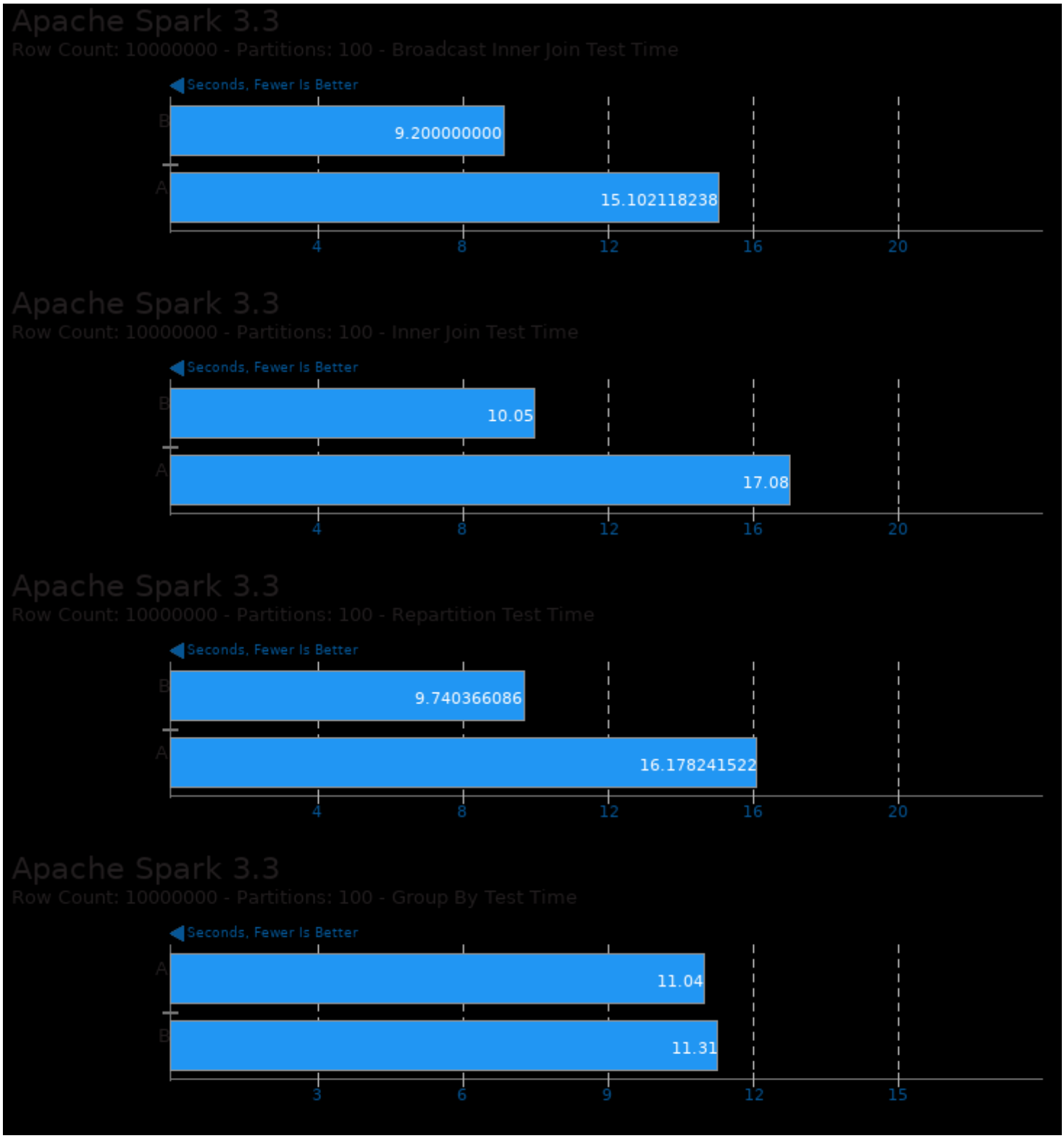

Apache Spark 3.3
Row Count: 20000000 - Partitions: 2000 - Repartition Test Time
Seconds, Fewer Is Better
A
18.13
B
18.78
5
10
15
20
25
Apache Spark 3.3
Row Count: 20000000 - Partitions: 2000 - Group By Test Time
Seconds, Fewer Is Better
B
15.04
A
16.64
4
8
12
16
20
Apache Spark 3.3
Row Count: 20000000 - Partitions: 2000 - Calculate Pi Benchmark Using Dataframe
Seconds, Fewer Is Better
B
4.64
A
4.69
1.0553
2.1106
3.1659
4.2212
5.2765
Apache Spark 3.3
Row Count: 20000000 - Partitions: 2000 - Calculate Pi Benchmark
Seconds, Fewer Is Better
B
62.11
A
62.51
14
28
42
56
70

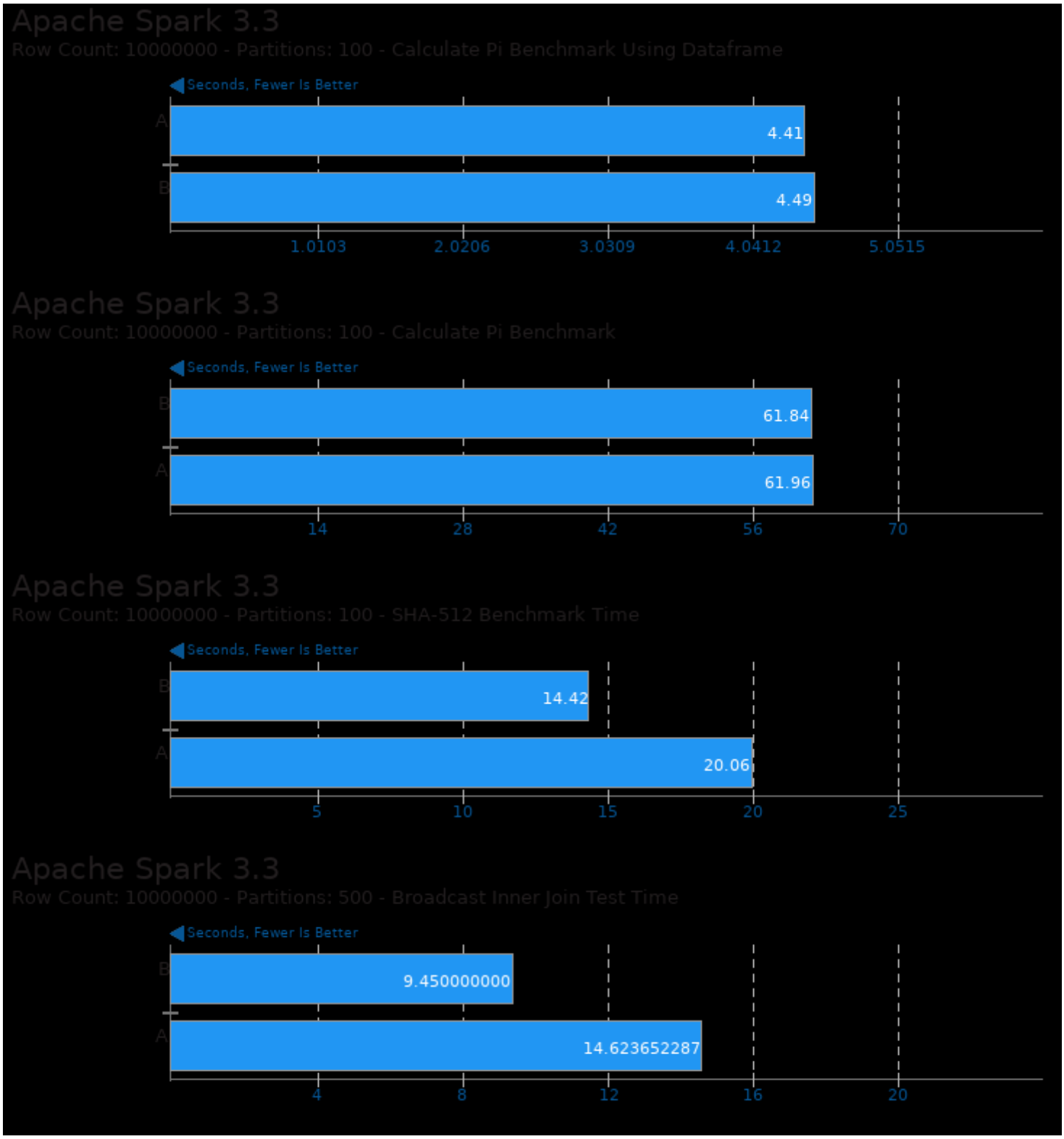
spark tr


Apache Spark 3.3
Row Count: 20000000 - Partitions: 2000 - SHA-512 Benchmark Time
Seconds, Fewer Is Better
A
25.22
B
25.61
6
12
18
24
30
Apache Spark 3.3
Row Count: 20000000 - Partitions: 1000 - Broadcast Inner Join Test Time
Seconds, Fewer Is Better
A
17.53
B
18.23
4
8
12
16
20
Apache Spark 3.3
Row Count: 20000000 - Partitions: 1000 - Inner Join Test Time
Seconds, Fewer Is Better
B
17.74
A
18.13
4
8
12
16
20
Apache Spark 3.3
Row Count: 20000000 - Partitions: 1000 - Repartition Test Time
Seconds, Fewer Is Better
A
17.44
B
18.60
5
10
15
20
25

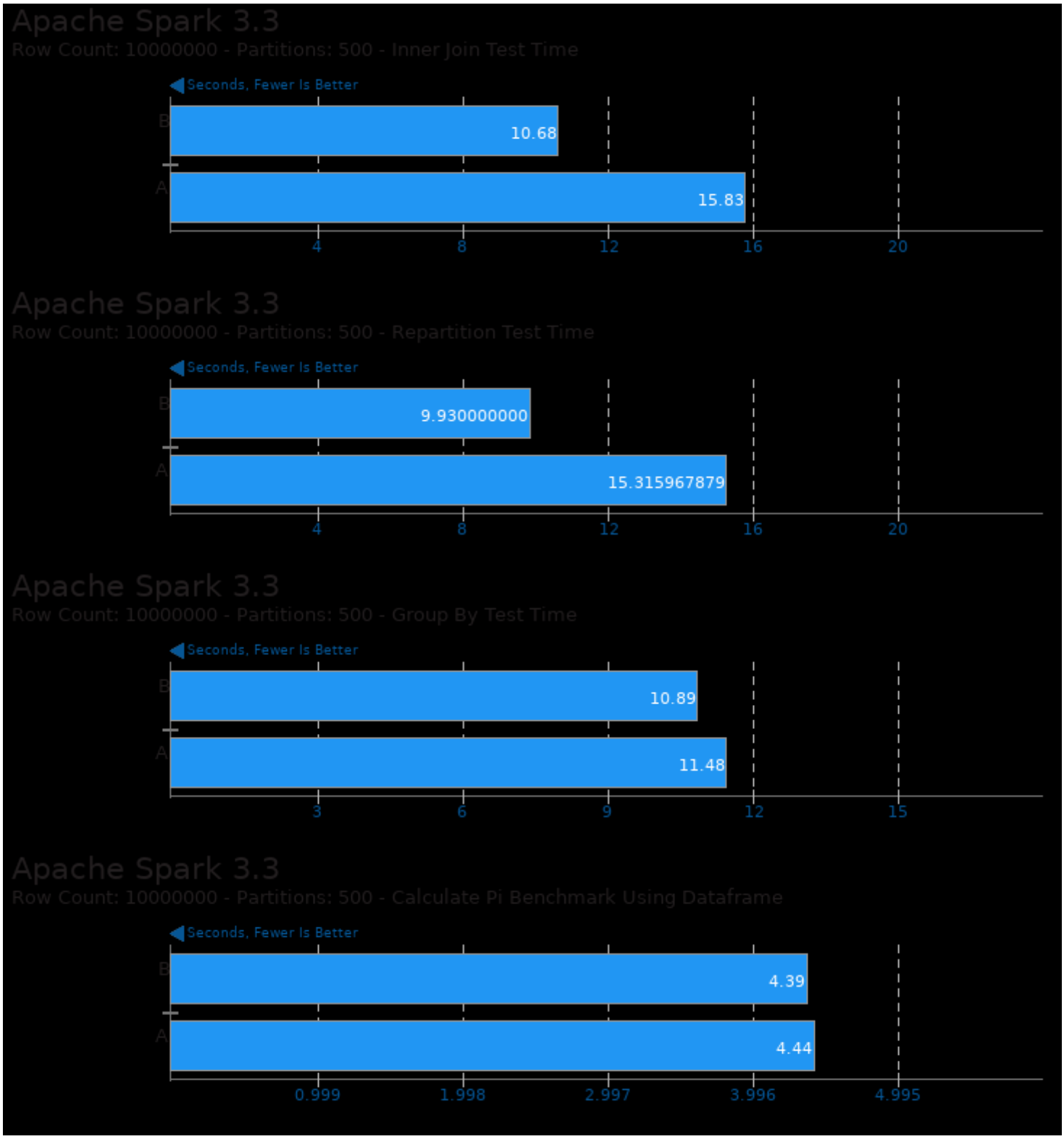
spark tr

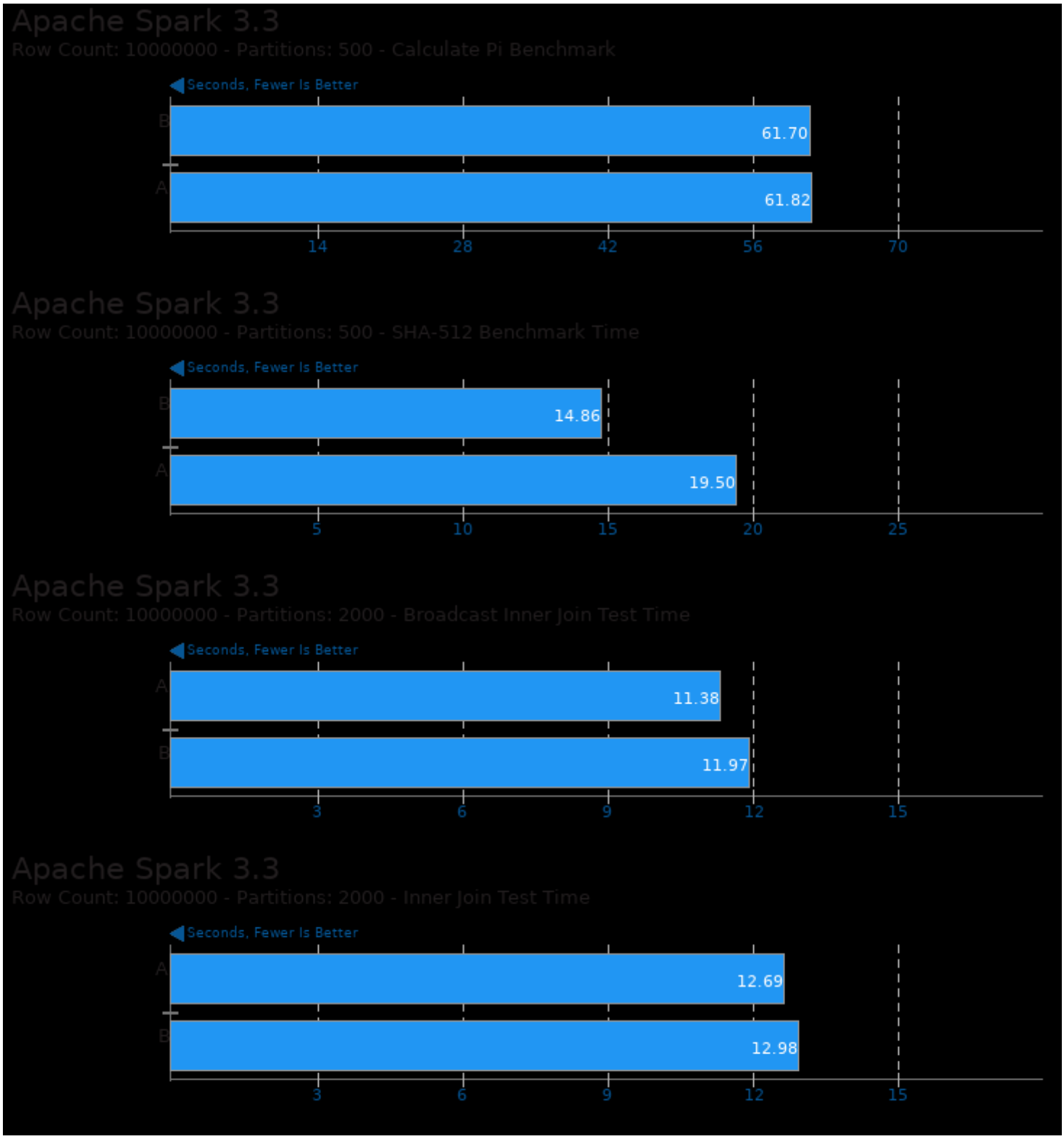

Apache Spark 3.3
Row Count: 20000000 - Partitions: 1000 - Group By Test Time
Seconds, Fewer Is Better
B
14.16
A
14.55
4
8
12
16
20
Apache Spark 3.3
Row Count: 20000000 - Partitions: 1000 - Calculate Pi Benchmark Using Dataframe
Seconds, Fewer Is Better
A
4.635858138
B
4.650000000
1.0463
2.0926
3.1389
4.1852
5.2315
Apache Spark 3.3
Row Count: 20000000 - Partitions: 1000 - Calculate Pi Benchmark
Seconds, Fewer Is Better
A
62.24
B
62.53
14
28
42
56
70
Apache Spark 3.3
Row Count: 20000000 - Partitions: 1000 - SHA-512 Benchmark Time
Seconds, Fewer Is Better
A
24.56
B
25.13
6
12
18
24
30

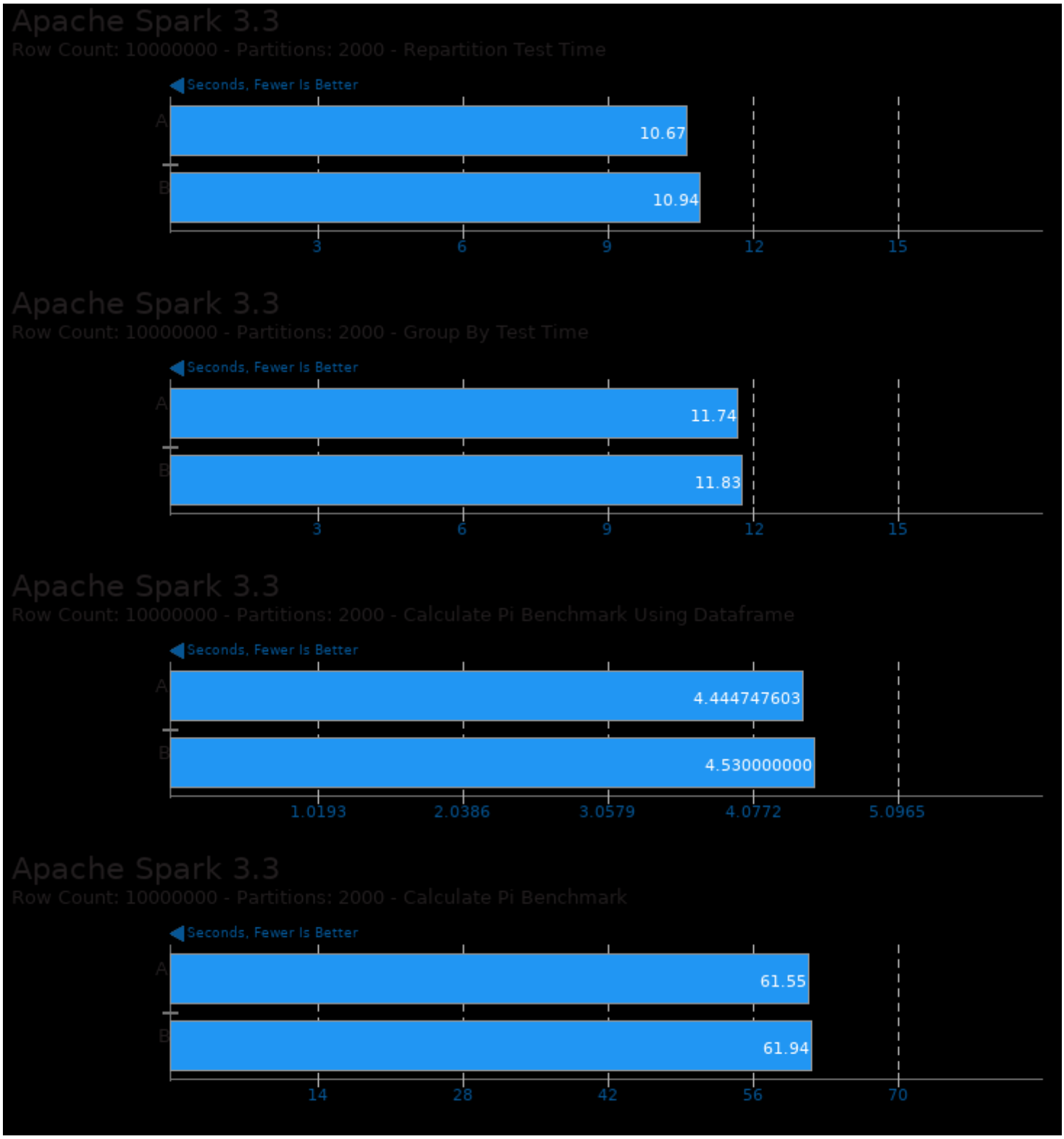

Apache Spark 3.3
Row Count: 10000000 - Partitions: 100 - Broadcast Inner Join Test Time
Seconds, Fewer Is Better
B
9.200000000
A
15.102118238
4
8
12
16
20
Apache Spark 3.3
Row Count: 10000000 - Partitions: 100 - Inner Join Test Time
Seconds, Fewer Is Better
B
10.05
A
17.08
4
8
12
16
20
Apache Spark 3.3
Row Count: 10000000 - Partitions: 100 - Repartition Test Time
Seconds, Fewer Is Better
B
9.740366086
A
16.178241522
4
8
12
16
20
Apache Spark 3.3
Row Count: 10000000 - Partitions: 100 - Group By Test Time
Seconds, Fewer Is Better
A
11.04
B
11.31
3
6
9
12
15


Apache Spark 3.3
Row Count: 10000000 - Partitions: 100 - Calculate Pi Benchmark Using Dataframe
Seconds, Fewer Is Better
A
4.41
B
4.49
1.0103
2.0206
3.0309
4.0412
5.0515
Apache Spark 3.3
Row Count: 10000000 - Partitions: 100 - Calculate Pi Benchmark
Seconds, Fewer Is Better
B
61.84
A
61.96
14
28
42
56
70
Apache Spark 3.3
Row Count: 10000000 - Partitions: 100 - SHA-512 Benchmark Time
Seconds, Fewer Is Better
B
14.42
A
20.06
5
10
15
20
25
Apache Spark 3.3
Row Count: 10000000 - Partitions: 500 - Broadcast Inner Join Test Time
Seconds, Fewer Is Better
B
9.450000000
A
14.623652287
4
8
12
16
20

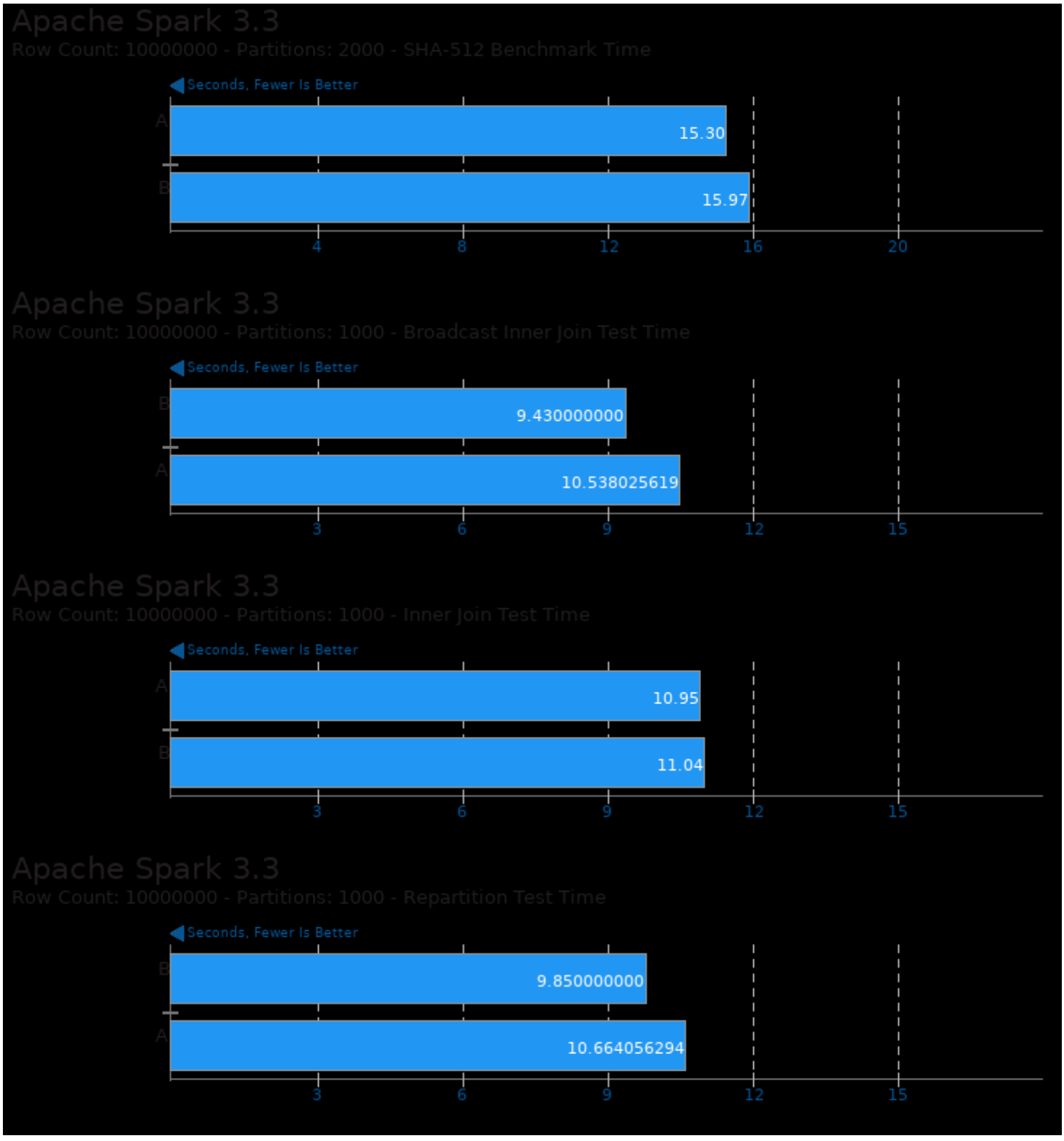
spark tr


Apache Spark 3.3
Row Count: 10000000 - Partitions: 500 - Inner Join Test Time
Seconds, Fewer Is Better
B
10.68
A
15.83
4
8
12
16
20
Apache Spark 3.3
Row Count: 10000000 - Partitions: 500 - Repartition Test Time
Seconds, Fewer Is Better
B
9.930000000
A
15.315967879
4
8
12
16
20
Apache Spark 3.3
Row Count: 10000000 - Partitions: 500 - Group By Test Time
Seconds, Fewer Is Better
B
10.89
A
11.48
3
6
9
12
15
Apache Spark 3.3
Row Count: 10000000 - Partitions: 500 - Calculate Pi Benchmark Using Dataframe
Seconds, Fewer Is Better
B
4.39
A
4.44
0.999
1.998
2.997
3.996
4.995

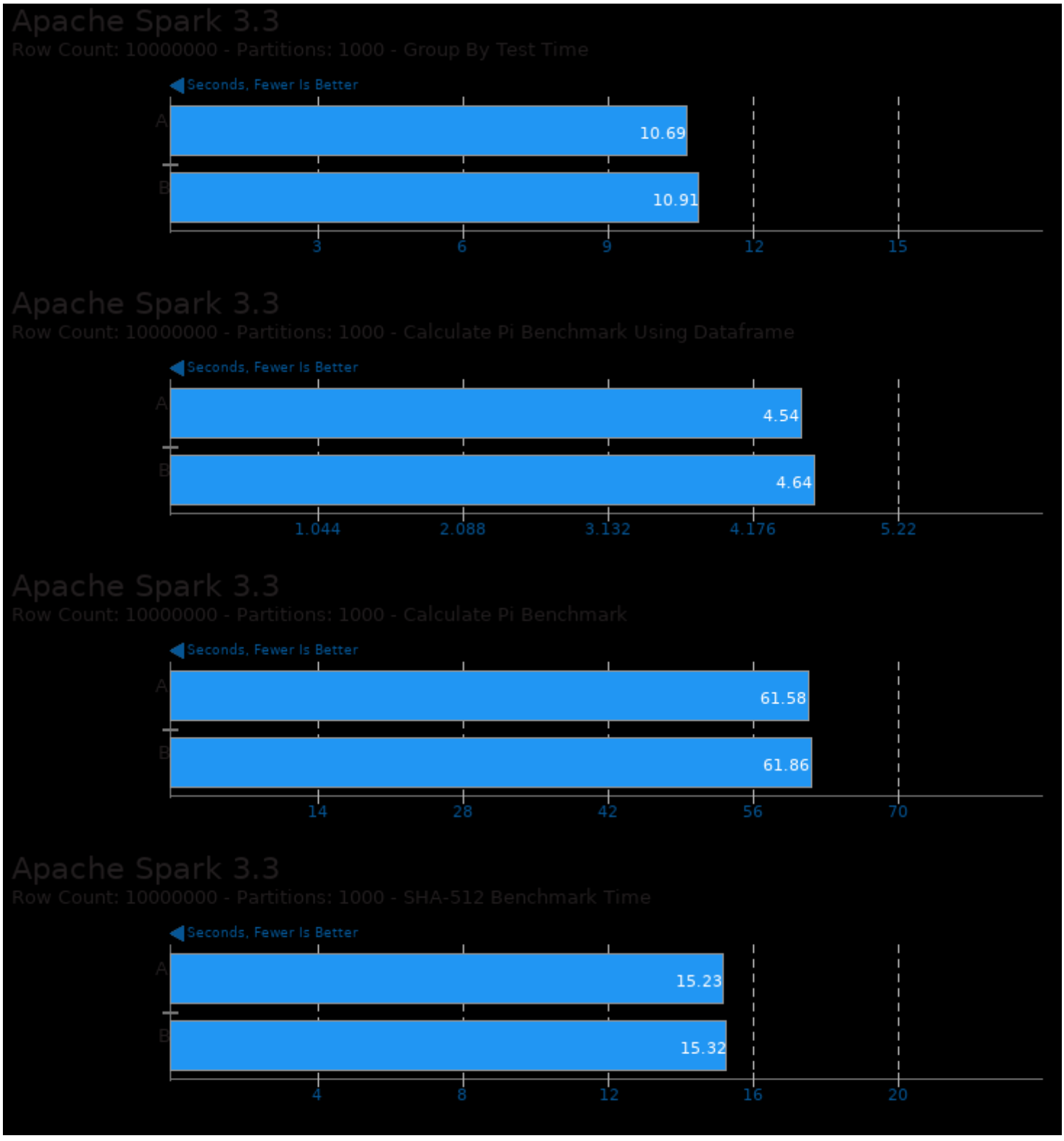

Apache Spark 3.3
Row Count: 10000000 - Partitions: 500 - Calculate Pi Benchmark
Seconds, Fewer Is Better
B
61.70
A
61.82
14
28
42
56
70
Apache Spark 3.3
Row Count: 10000000 - Partitions: 500 - SHA-512 Benchmark Time
Seconds, Fewer Is Better
B
14.86
A
19.50
5
10
15
20
25
Apache Spark 3.3
Row Count: 10000000 - Partitions: 2000 - Broadcast Inner Join Test Time
Seconds, Fewer Is Better
A
11.38
B
11.97
3
6
9
12
15
Apache Spark 3.3
Row Count: 10000000 - Partitions: 2000 - Inner Join Test Time
Seconds, Fewer Is Better
A
12.69
B
12.98
3
6
9
12
15

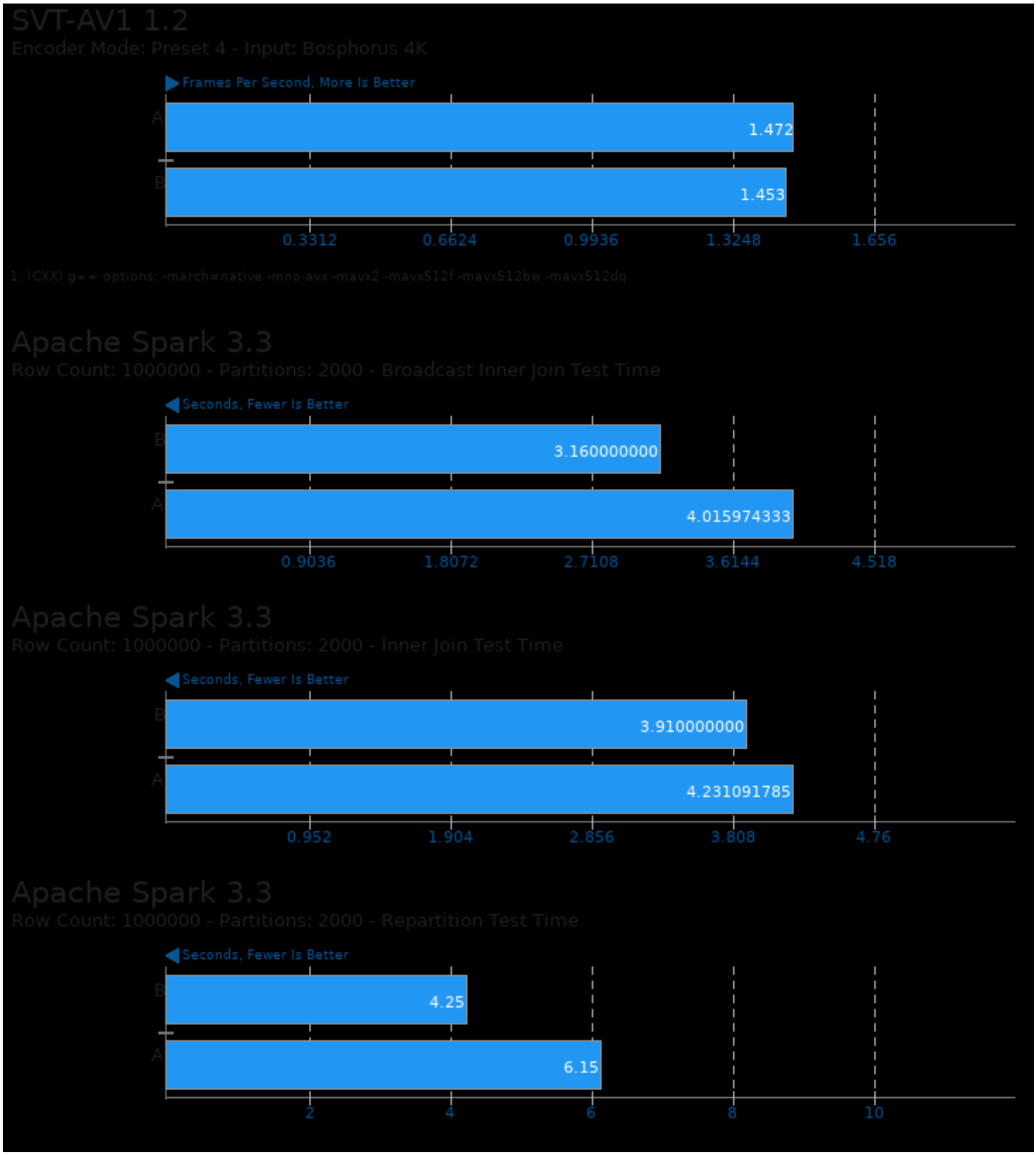

Apache Spark 3.3
Row Count: 10000000 - Partitions: 2000 - Repartition Test Time
Seconds, Fewer Is Better
A
10.67
B
10.94
3
6
9
12
15
Apache Spark 3.3
Row Count: 10000000 - Partitions: 2000 - Group By Test Time
Seconds, Fewer Is Better
A
11.74
B
11.83
3
6
9
12
15
Apache Spark 3.3
Row Count: 10000000 - Partitions: 2000 - Calculate Pi Benchmark Using Dataframe
Seconds, Fewer Is Better
A
4.444747603
B
4.530000000
1.0193
2.0386
3.0579
4.0772
5.0965
Apache Spark 3.3
Row Count: 10000000 - Partitions: 2000 - Calculate Pi Benchmark
Seconds, Fewer Is Better
A
61.55
B
61.94
14
28
42
56
70

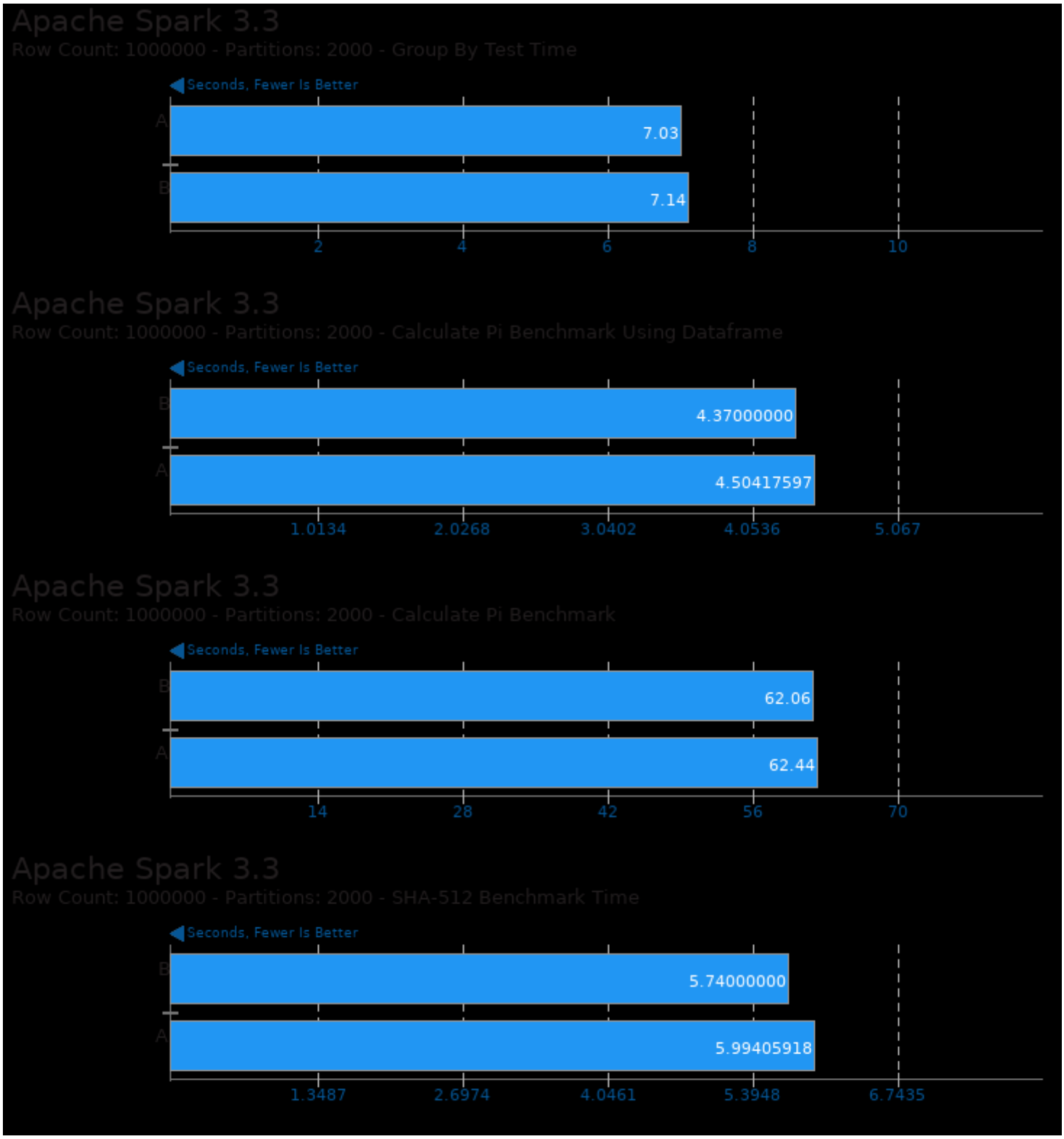
spark tr

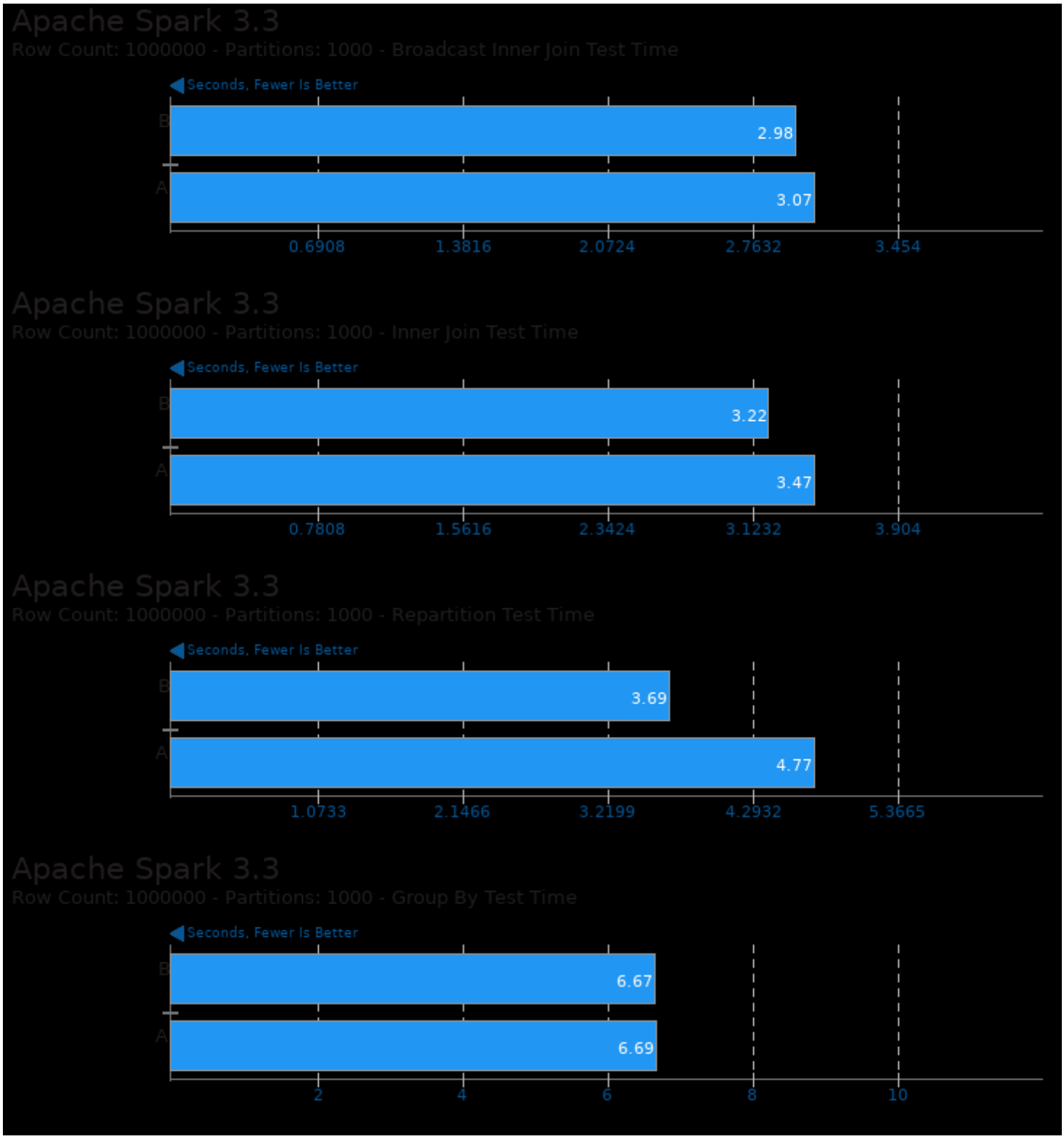

Apache Spark 3.3
Row Count: 10000000 - Partitions: 2000 - SHA-512 Benchmark Time
Seconds, Fewer Is Better
A
15.30
B
15.97
4
8
12
16
20
Apache Spark 3.3
Row Count: 10000000 - Partitions: 1000 - Broadcast Inner Join Test Time
Seconds, Fewer Is Better
B
9.430000000
A
10.538025619
3
6
9
12
15
Apache Spark 3.3
Row Count: 10000000 - Partitions: 1000 - Inner Join Test Time
Seconds, Fewer Is Better
A
10.95
B
11.04
3
6
9
12
15
Apache Spark 3.3
Row Count: 10000000 - Partitions: 1000 - Repartition Test Time
Seconds, Fewer Is Better
B
9.850000000
A
10.664056294
3
6
9
12
15

spark tr

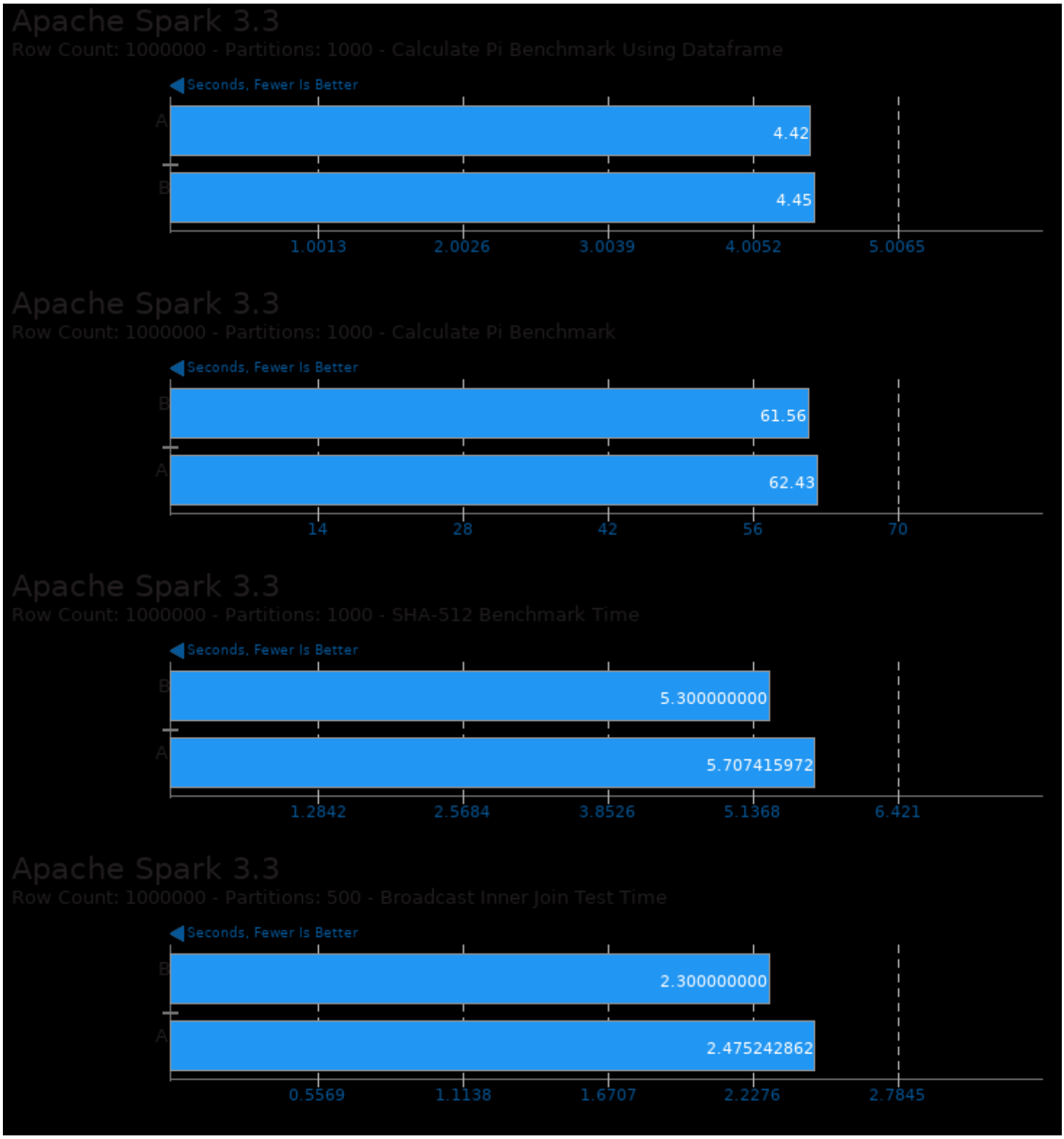

Apache Spark 3.3
Row Count: 10000000 - Partitions: 1000 - Group By Test Time
Seconds, Fewer Is Better
A
10.69
B
10.91
3
6
9
12
15
Apache Spark 3.3
Row Count: 10000000 - Partitions: 1000 - Calculate Pi Benchmark Using Dataframe
Seconds, Fewer Is Better
A
4.54
B
4.64
1.044
2.088
3.132
4.176
5.22
Apache Spark 3.3
Row Count: 10000000 - Partitions: 1000 - Calculate Pi Benchmark
Seconds, Fewer Is Better
A
61.58
B
61.86
14
28
42
56
70
Apache Spark 3.3
Row Count: 10000000 - Partitions: 1000 - SHA-512 Benchmark Time
Seconds, Fewer Is Better
A
15.23
B
15.32
4
8
12
16
20


SVT-AV1 1.2
Encoder Mode: Preset 4 - Input: Bosphorus 4K
Frames Per Second, More Is Better
A
1.472
B
1.453
0.3312
0.6624
0.9936
1.3248
1.656
1: (CXX) g++ options: -march=native -mno-avx -mavx2 -mavx512f -mavx512bw -mavx512dq
Apache Spark 3.3
Row Count: 1000000 - Partitions: 2000 - Broadcast Inner Join Test Time
Seconds, Fewer Is Better
B
3.160000000
A
4.015974333
0.9036
1.8072
2.7108
3.6144
4.518
Apache Spark 3.3
Row Count: 1000000 - Partitions: 2000 - Inner Join Test Time
Seconds, Fewer Is Better
B
3.910000000
A
4.231091785
0.952
1.904
2.856
3.808
4.76
Apache Spark 3.3
Row Count: 1000000 - Partitions: 2000 - Repartition Test Time
Seconds, Fewer Is Better
B
4.25
A
6.15
2
4
6
8
10

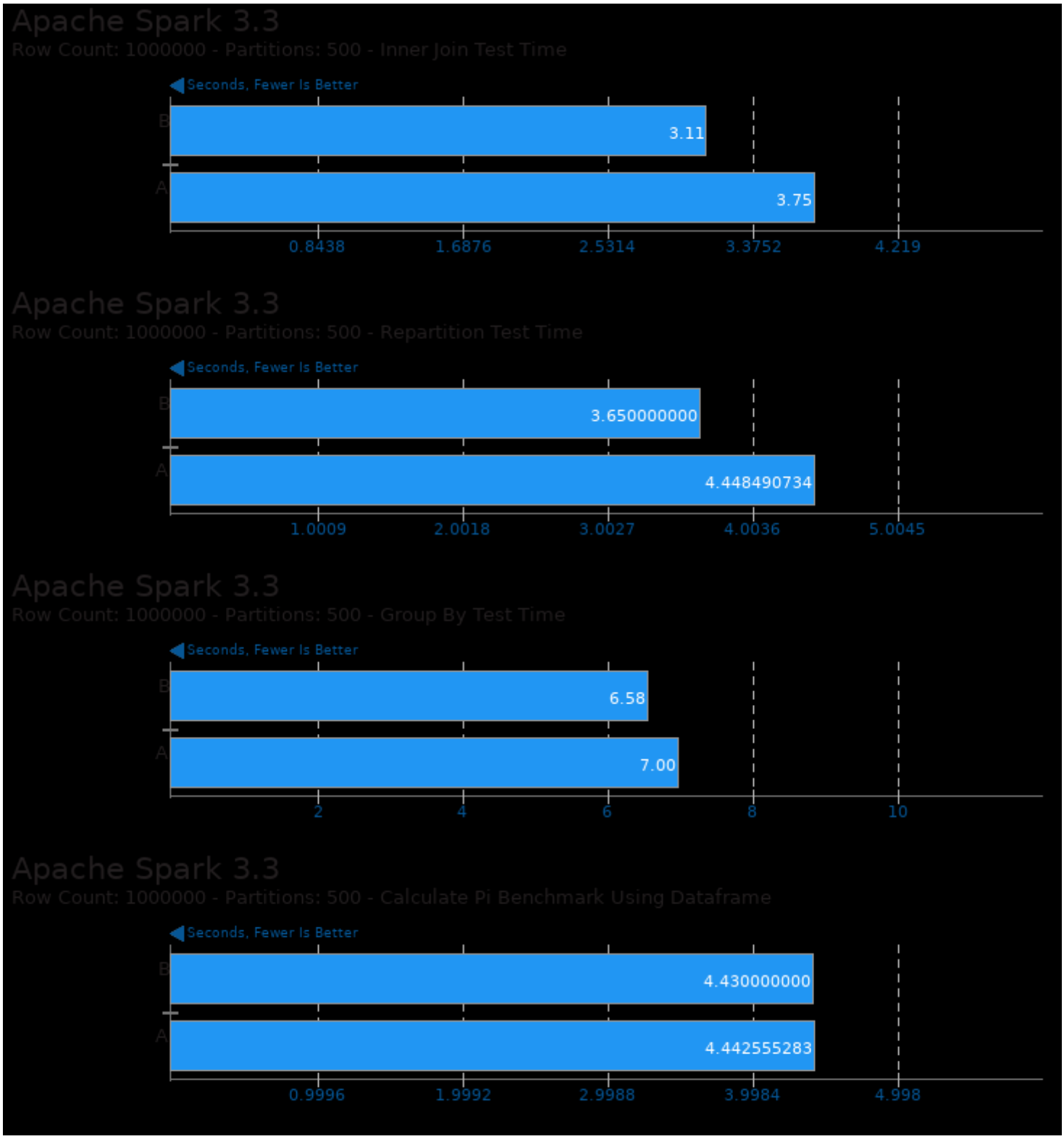

Apache Spark 3.3
Row Count: 1000000 - Partitions: 2000 - Group By Test Time
Seconds, Fewer Is Better
A
7.03
B
7.14
2
4
6
8
10
Apache Spark 3.3
Row Count: 1000000 - Partitions: 2000 - Calculate Pi Benchmark Using Dataframe
Seconds, Fewer Is Better
B
4.37000000
A
4.50417597
1.0134
2.0268
3.0402
4.0536
5.067
Apache Spark 3.3
Row Count: 1000000 - Partitions: 2000 - Calculate Pi Benchmark
Seconds, Fewer Is Better
B
62.06
A
62.44
14
28
42
56
70
Apache Spark 3.3
Row Count: 1000000 - Partitions: 2000 - SHA-512 Benchmark Time
Seconds, Fewer Is Better
B
5.74000000
A
5.99405918
1.3487
2.6974
4.0461
5.3948
6.7435

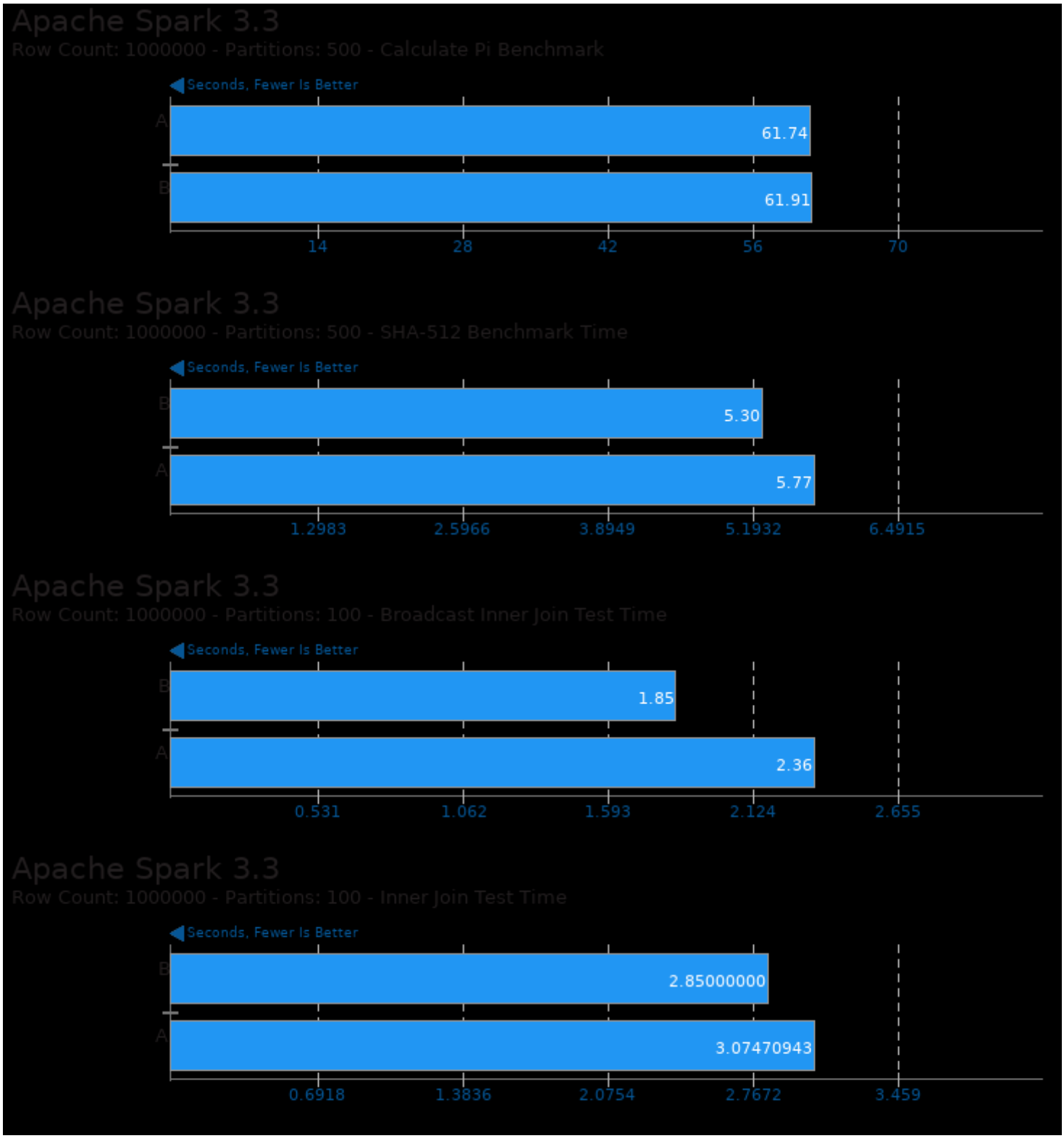

Apache Spark 3.3
Row Count: 1000000 - Partitions: 1000 - Broadcast Inner Join Test Time
Seconds, Fewer Is Better
B
2.98
A
3.07
0.6908
1.3816
2.0724
2.7632
3.454
Apache Spark 3.3
Row Count: 1000000 - Partitions: 1000 - Inner Join Test Time
Seconds, Fewer Is Better
B
3.22
A
3.47
0.7808
1.5616
2.3424
3.1232
3.904
Apache Spark 3.3
Row Count: 1000000 - Partitions: 1000 - Repartition Test Time
Seconds, Fewer Is Better
B
3.69
A
4.77
1.0733
2.1466
3.2199
4.2932
5.3665
Apache Spark 3.3
Row Count: 1000000 - Partitions: 1000 - Group By Test Time
Seconds, Fewer Is Better
B
6.67
A
6.69
2
4
6
8
10

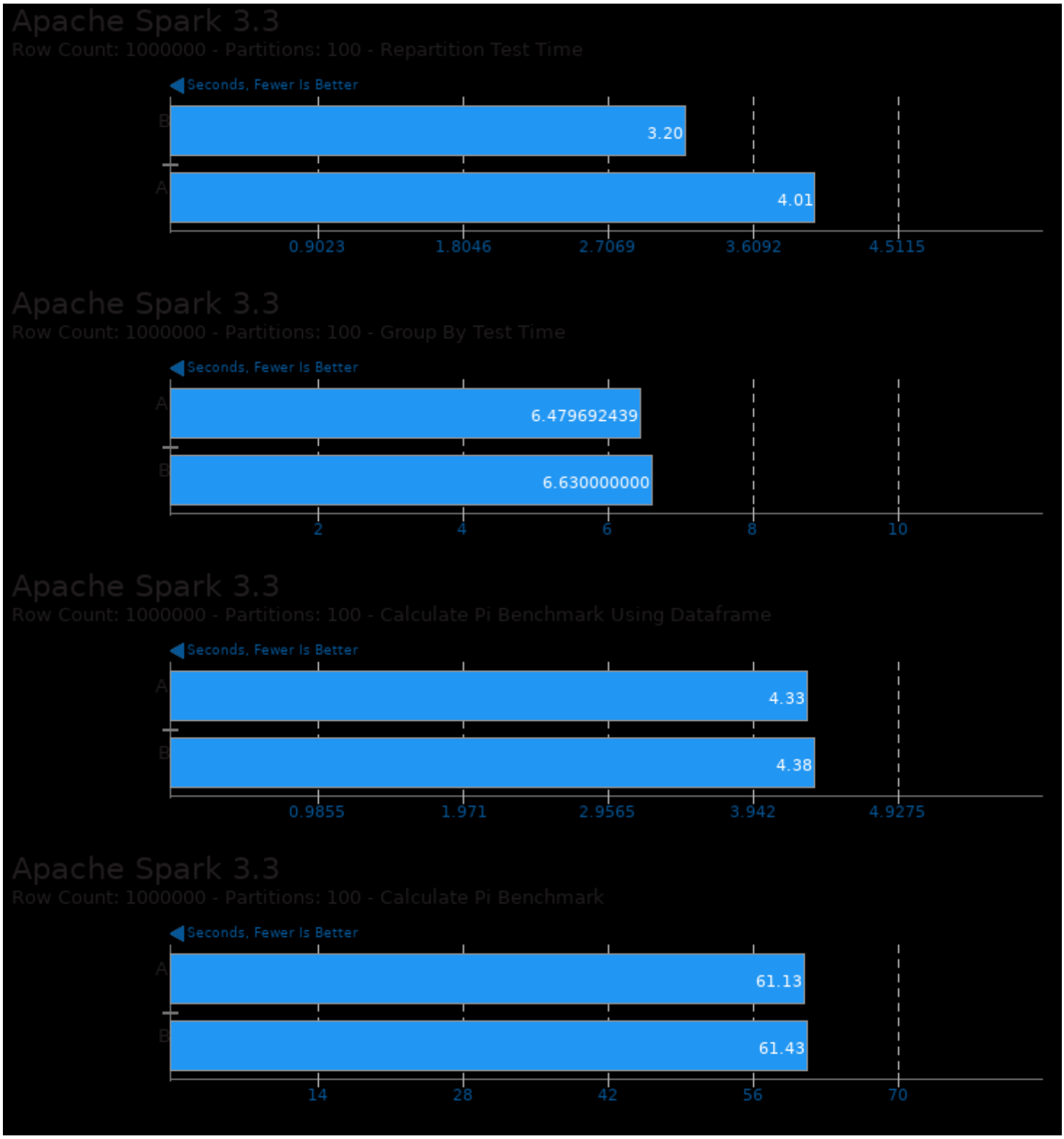
spark tr


Apache Spark 3.3
Row Count: 1000000 - Partitions: 1000 - Calculate Pi Benchmark Using Dataframe
Seconds, Fewer Is Better
A
4.42
B
4.45
1.0013
2.0026
3.0039
4.0052
5.0065
Apache Spark 3.3
Row Count: 1000000 - Partitions: 1000 - Calculate Pi Benchmark
Seconds, Fewer Is Better
B
61.56
A
62.43
14
28
42
56
70
Apache Spark 3.3
Row Count: 1000000 - Partitions: 1000 - SHA-512 Benchmark Time
Seconds, Fewer Is Better
B
5.300000000
A
5.707415972
1.2842
2.5684
3.8526
5.1368
6.421
Apache Spark 3.3
Row Count: 1000000 - Partitions: 500 - Broadcast Inner Join Test Time
Seconds, Fewer Is Better
B
2.300000000
A
2.475242862
0.5569
1.1138
1.6707
2.2276
2.7845

spark tr

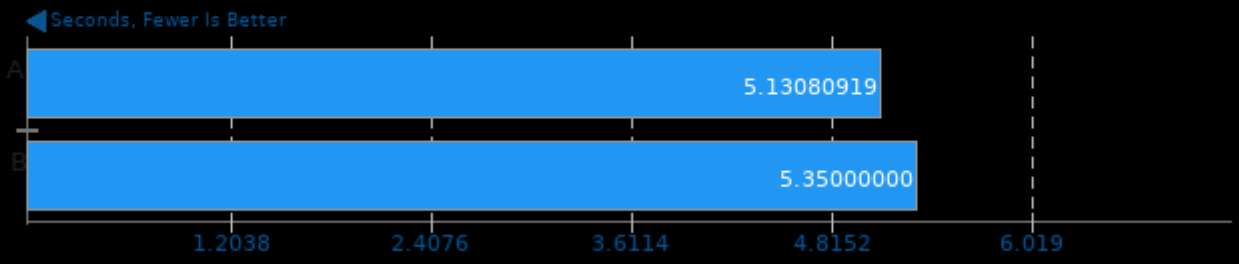

Apache Spark 3.3
Row Count: 1000000 - Partitions: 500 - Inner Join Test Time
Seconds, Fewer Is Better
B
3.11
A
3.75
0.8438
1.6876
2.5314
3.3752
4.219
Apache Spark 3.3
Row Count: 1000000 - Partitions: 500 - Repartition Test Time
Seconds, Fewer Is Better
B
3.650000000
A
4.448490734
1.0009
2.0018
3.0027
4.0036
5.0045
Apache Spark 3.3
Row Count: 1000000 - Partitions: 500 - Group By Test Time
Seconds, Fewer Is Better
B
6.58
A
7.00
2
4
6
8
10
Apache Spark 3.3
Row Count: 1000000 - Partitions: 500 - Calculate Pi Benchmark Using Dataframe
Seconds, Fewer Is Better
B
4.430000000
A
4.442555283
0.9996
1.9992
2.9988
3.9984
4.998


Apache Spark 3.3
Row Count: 1000000 - Partitions: 500 - Calculate Pi Benchmark
Seconds, Fewer Is Better
A
61.74
B
61.91
14
28
42
56
70
Apache Spark 3.3
Row Count: 1000000 - Partitions: 500 - SHA-512 Benchmark Time
Seconds, Fewer Is Better
B
5.30
A
5.77
1.2983
2.5966
3.8949
5.1932
6.4915
Apache Spark 3.3
Row Count: 1000000 - Partitions: 100 - Broadcast Inner Join Test Time
Seconds, Fewer Is Better
B
1.85
A
2.36
0.531
1.062
1.593
2.124
2.655
Apache Spark 3.3
Row Count: 1000000 - Partitions: 100 - Inner Join Test Time
Seconds, Fewer Is Better
B
2.85000000
A
3.07470943
0.6918
1.3836
2.0754
2.7672
3.459


Apache Spark 3.3
Row Count: 1000000 - Partitions: 100 - Repartition Test Time
Seconds, Fewer Is Better
B
3.20
A
4.01
0.9023
1.8046
2.7069
3.6092
4.5115
Apache Spark 3.3
Row Count: 1000000 - Partitions: 100 - Group By Test Time
Seconds, Fewer Is Better
A
6.479692439
B
6.630000000
2
4
6
8
10
Apache Spark 3.3
Row Count: 1000000 - Partitions: 100 - Calculate Pi Benchmark Using Dataframe
Seconds, Fewer Is Better
A
4.33
B
4.38
0.9855
1.971
2.9565
3.942
4.9275
Apache Spark 3.3
Row Count: 1000000 - Partitions: 100 - Calculate Pi Benchmark
Seconds, Fewer Is Better
A
61.13
B
61.43
14
28
42
56
70

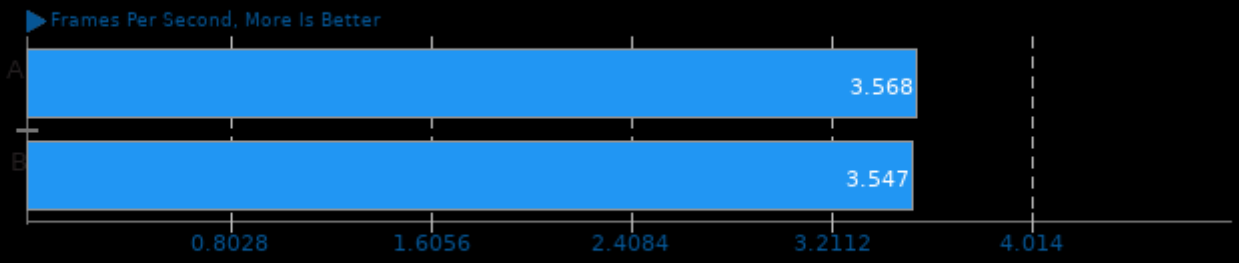
Apache Spark 3.3

Row Count: 1000000 - Partitions: 100 - SHA-512 Benchmark Time



SVT-AV1 1.2

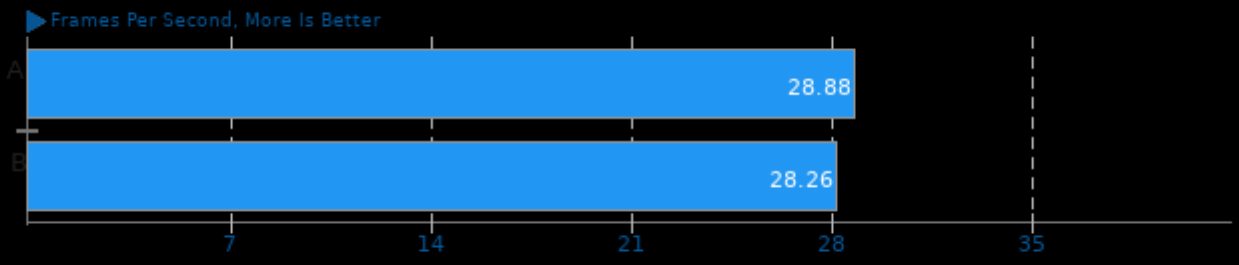
Encoder Mode: Preset 4 - Input: Bosphorus 1080p



1. (CXX) g++ options: -march=native -mno-avx -mavx2 -mavx512f -mavx512bw -mavx512dq

SVT-AV1 1.2

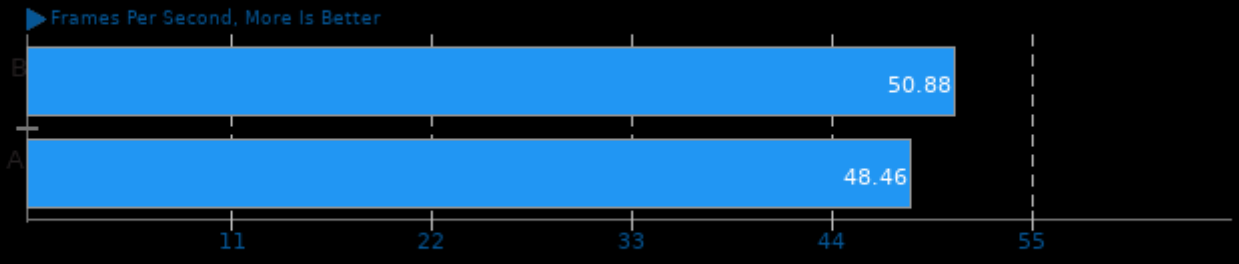
Encoder Mode: Preset 8 - Input: Bosphorus 4K



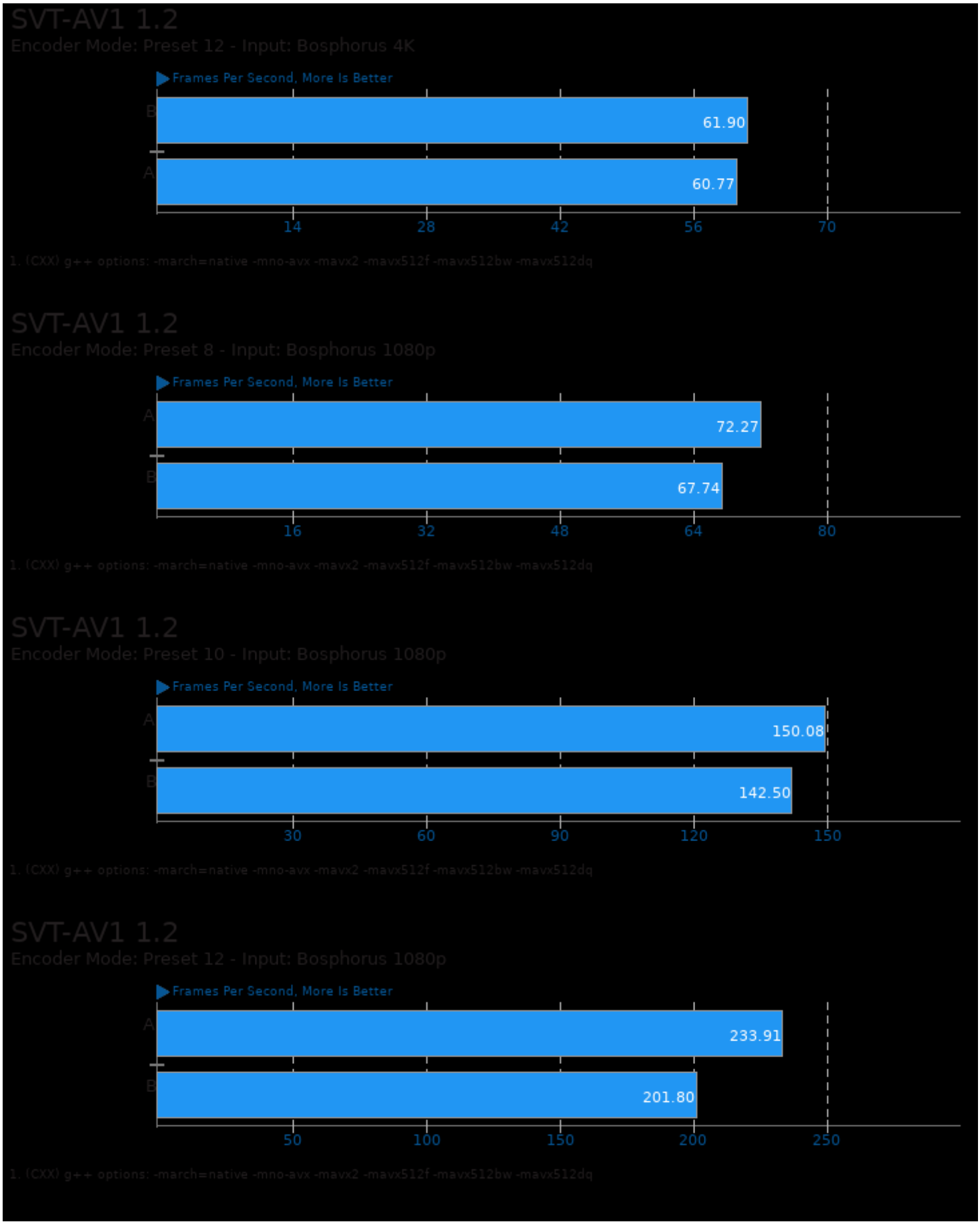
1. (CXX) g++ options: -march=native -mno-avx -mavx2 -mavx512f -mavx512bw -mavx512dq

SVT-AV1 1.2

Encoder Mode: Preset 10 - Input: Bosphorus 4K



1. (CXX) g++ options: -march=native -mno-avx -mavx2 -mavx512f -mavx512bw -mavx512dq


SVT-AV1 1.2
Encoder Mode: Preset 12 - Input: Bosphorus 4K
Frames Per Second, More Is Better
B
61.90
A
60.77
14
28
42
56
70
1. (CXX) g++ options: -march=native -mno-avx -mavx2 -mavx512f -mavx512bw -mavx512dq
SVT-AV1 1.2
Encoder Mode: Preset 8 - Input: Bosphorus 1080p
Frames Per Second, More Is Better
A
72.27
B
67.74
16
32
48
64
80
1. (CXX) g++ options: -march=native -mno-avx -mavx2 -mavx512f -mavx512bw -mavx512dq
SVT-AV1 1.2
Encoder Mode: Preset 10 - Input: Bosphorus 1080p
Frames Per Second, More Is Better
A
150.08
B
142.50
30
60
90
120
150
1. (CXX) g++ options: -march=native -mno-avx -mavx2 -mavx512f -mavx512bw -mavx512dq
SVT-AV1 1.2
Encoder Mode: Preset 12 - Input: Bosphorus 1080p
Frames Per Second, More Is Better
A
233.91
B
201.80
50
100
150
200
250
1. (CXX) g++ options: -march=native -mno-avx -mavx2 -mavx512f -mavx512bw -mavx512dq

spark tr

This file was automatically generated via the Phoronix Test Suite benchmarking software on Thursday, 28 March 2024 12:35.