Performance effects of building Ztocs with different values of span size

Summary

This document presents the results of running the performance test from the benchmarking framework for container images having several Soci indices built with different span size configurations. Based on https://github.com/awslabs/soci-snapshotter/commit/1628d6eac6cb9383f9538d0bb85de8a007b4f9a3, the initial span size was set to 4 MiB. There was a question if that's enough. This document tries to answer this question. For that, we look at the span sizes of 4, 8, 16 and 32 MiB to see if increasing the span size positively affects the total time to run the container workload to ready line.

The data was collected in the following way:

1. For every image in the Benchmarking results table 3 additional indices were built with span sizes 8, 16 and 32 MiB respectively.
2. For each image and span size, the performance test was executed using 10 samples (`make benchmarks`).

Looking at the data considering network calls to S3 (next section) it is evident, that for most of the workloads increasing the span size to 8 MiB is beneficial and improves the mean time to get to ready line by 1.75-10% (exceptions: gcchellocompile and rabbitmq) and max time to get to ready line by 1.85-26% (exceptions: gcchellocompile, mongo, glassfish, rabbitmq).
gcchellocompile appears to be an extreme case, where increasing span size negatively affect the performance (the difference is under 2s for 8MiB).
tensorflow cpu is the only image, where increasing the span size all the way to 32MiB shows the biggest gains.

Therefore, we can conclude that increasing the default span size to 8MiB will benefit most of the workloads with minimal harm to gcchellocompile, mongo, glassfish, rabbitmq.

For tensorflow image only it can be recommended to build an index with span size of 32MiB, since it produces the most benefits (29.3% improvement in mean time to ready line and 25.18% improvement in max time to ready line).

However, if we eliminate network latency (and in the case of ECR, the latency to download data from S3), the results are pretty different and there's a strong correlation between increasing the span size and worsening the performance. For the details on the approach, please refer to Appendix B.

Considering that there's a plan to put some effort into optimizing S3 gets, and given that the baseline performance with network latency eliminated is the best for span size 4MiB, it is recommended to keep the default span size as 4MiB and reevaluate the performance after S3 optimizations are complete.

Benchmarking results

1. BENCHMARKING WITH ACCOUNTING FOR NETWORK CALLS TO S3

	A	B	C	D	E	F	G	H	I
1	image	image size, MiB	span size, MiB	mean total time, s	mean total time diff, %	min total time, s	max total time, s	max total time diff, %	soci index sha
2	tomcat	252	4	6.19	0.00%	5.76	7.01	0.00%	sha256:f6fab4cfb60142401062e50b02b7d22c37c3a3dcd55511ece6893e440fbd84aa
3			8	5.86	5.33%	5.57	6.18	11.84%	sha256:62d96589f4cd8cc4eaa714d43c9bbe68d06c086458592852cff778d98e0c4eb4
4			16	6.35	-2.58%	6	7.29	-3.99%	sha256:bc15ca01ebdedb5246356b1dc7520df2adf02d3cd5247b0e0b399ef26ac3372b
5			32	7.35	-18.74%	7.01	9.08	-29.53%	sha256:6ec9cefca8c8f97daac5a10f66eb4b18b51c3c4d5b1f6b1052dda7561b25e29d
6	rethinkdb	48	4	2.23	0.00%	1.99	3.53	0.00%	sha256:50fa6303f78dabc5f6d60bcac24019590f9a005309264d9ad9d0c3cf1a4f9687
7			8	2.1	5.83%	1.94	2.58	26.91%	sha256:0a957f0e69545b2459bd837ff3867204243b1a0552880271519fcd4f56ce913d
8			16	2.03	8.97%	1.89	2.42	31.44%	sha256:94b3f6ca920a9436b683eeb361291741d136e176163049bd8bd38d9d5492cd5d
9			32	2.23	0.00%	2.11	2.87	18.70%	sha256:e7b9af7bb83b56a0efcb3258984dfdd0c4a2bc67f62a184dcbab0085c29a3c4f
10	python	352	4	3.07	0.00%	2.74	4.58	0.00%	sha256:fdde13a373d92092d3208538103c7cdebe0b09aaa2b60c580be387fee3e37318
11			8	2.99	2.61%	2.79	3.45	24.67%	sha256:49d1944e69d21c017d6179991951961e3356a776b78fcecacb1a330bb484a891e
12			16	3.31	-7.82%	3.07	4.28	6.55%	sha256:3fa6de3ccf6dd7bb9f1dad7396af0440de62286cc627959af6ed7b8890b55a2d
13			32	4.03	-31.27%	3.74	4.91	-7.21%	sha256:65dd368cfc36ebcc2f28d5d57e635a925ed0cf1b7eb1f210f16408b7a9ab8b12
14	mongo	232	4	5.36	0.00%	5.19	5.55	0.00%	sha256:a58e81941726547c2f2560c1771d3887131b9528141c93b7a1aa36a0d649006b
15			8	5.01	6.53%	4.79	5.91	-6.49%	sha256:e3d0fd7cb58ed4fc792af04294ce37e91be7bf20a85de35eb1502aeb2db820c2
16			16	4.67	12.87%	4.51	5.06	8.83%	sha256:569b1680be222121e214e4f2e3ec58da7c958f7166b243c053bf1f0d8a992a6a
17			32	4.53	15.49%	4.41	4.84	12.79%	sha256:076c966caa12a08f52f3f5842c018ea35a1184140c236392861f5e6eaac69fae
18	jetty	256	4	6.62	0.00%	6.07	10.18	0.00%	sha256:9b9a777ac59528f3ead96d438740dc3b362a73916235f1cd64e572d90d9069a3
19			8	6.34	4.23%	5.98	7.52	26.13%	sha256:8d69ff58b7be185b0b7022e69a15a09c10f37e33bc824a459619aa1e18888b0f
20			16	6.71	-1.36%	6.43	7.87	22.69%	sha256:f51bbe504084822bd8362d96e5f8fb6c1189c31cb3b9ce131dd40ea3c3337a42
21			32	7.67	-15.86%	7.28	9.25	9.14%	sha256:c1502b141e965cb12c3fe3bed1aa189a948d27f47bb42794d90e7e9fe2f02e7b
22	jenkins	290	4	12.6	0.00%	11.9	15.03	0.00%	sha256:bb37643d4db4b493b673ec51a51ff17f638ae052212ffeddc413c23413d16b83
23			8	12.25	2.78%	11.76	13.11	12.77%	sha256:b692624103f396bd79463f96d8ad08c38466bf30d247faf3ad1cbe210495cef6
24			16	12.16	3.49%	11.86	13.07	13.04%	sha256:f5057084c15081007a01e9905d930dc2ca262f453f4c6ddabbda906711525600
25			32	12.68	-0.63%	12.39	13.66	9.12%	sha256:6ccb0c38d3242ae7d03697c3a0c3a0cacfae079e2ccd1e105600c4f23cb1b4f3
26	glassfish	351	4	10.35	0.00%	10.13	10.79	0.00%	sha256:6c17bc83dff53d0556bf6e6d8bcd94517a0532cc0b2c589b988b53f0d873da40
27			8	10.09	2.51%	9.53	10.86	-0.65%	sha256:18bb6a4e2ff43ea80fbed9a4dc21f1d40666c0eeca73456b72ecb6ed16219ba4
28			16	10.57	-2.13%	10.21	11.91	-10.38%	sha256:5a29fcb8aef40596d03fc631f91973f33f82dd9b4893491ca8e5159362905ecd
29			32	10.31	0.39%	9.99	11.26	-4.36%	sha256:553f14aa8a134f195dcb0c27eafc5c446efcea4ecf026aaff95d5349860677fc
30	ghost	158	4	11.46	0.00%	10.52	12.79	0.00%	sha256:3ec28440ef7f685f9d325973acecb31425deb6993144964d95282b645b1afa70
31			8	11.26	1.75%	10.46	12.53	2.03%	sha256:1acc0cc28121ed5a4b66daf9f18c65d486ea29bd06e11624cc0b13d0eab29428
32			16	11.15	2.71%	10.52	13.47	-5.32%	sha256:1f3cdf51ff65c00c453c03c1363c4eb56688a7f2f02e75042b008f8c1a5c5d10
33			32	10.7	6.63%	10.43	11.79	7.82%	sha256:9c3e6e700bd29ff4e5e4837ceddf54f2e5105085060bc4ca162b66732eee881a
34	gcchellocompile	449	4	3.89	0.00%	3.72	4.08	0.00%	sha256:381093aa887aceb8561f99d96abda88ac68f284843fca64f5db66d703fca888e
35			8	4.43	-13.88%	4.2	5.65	-38.48%	sha256:c8d35b84e22a6fe971c61741567fc4a0486d933538a54c4e6f836de73a86d871
36			16	5.02	-29.05%	4.69	6.22	-52.45%	sha256:e71680da0a10f255e1b51ad816dbf76e6a8cb67d99f8da5697a507ea41a2815f
37			32	7.55	-94.09%	6.92	9.66	-136.76%	sha256:a0f8dcb2e92cd08a878ffb0caeb1ccf548d89c4fdc2f04b1e12a06461f6e7457
38	tensorflow	451	4	19.73	0.00%	18.21	27.44	0.00%	sha256:ca132cd2f28296658f70d70e780e860b52cda1afd58353d3d80ee69cfebf6786
39			8	17.72	10.19%	17.01	18.87	31.23%	sha256:9b2975a67542aa482a9aca2e9cbe8fa3c8b20bfa3020cf8b2cb2cc9f8dc64e58
40			16	17.25	12.57%	16.65	19.72	28.13%	sha256:9b59adc8232c4443f216b5c028e544457e6300a68a942ebb73f2bd5c886af163
41			32	13.95	29.30%	12.93	20.53	25.18%	sha256:143f021c68b18b2c748b01ab0a444ebf9778cdb55b98aabc5240771d621944c3
42	drupal	189	4	4.48	0.00%	4.14	5.41	0.00%	sha256:75ce50d1b5e2a27248d609d82ba153d165ad7e8d193a79b9b5c0c8a52fffbc93
43			8	4.2	6.25%	3.98	5.31	1.85%	sha256:d85671468d0d1d4e20d84a5c35a661f375857fb7903d9ac96028f3f3e142a72d
44			16	4.53	-1.12%	4.34	5.22	3.51%	sha256:80f63d437a6a105a66c898e3c90415966b4668c72a203cea01423a9d829ec7cc
45			32	5.1	-13.84%	4.79	5.9	-9.06%	sha256:56474d6288ada1679708c51e9ada97f0310cafa618e0f94a7903c5f38b8797cc
46	redis	42	4	1.79	0.00%	1.65	2.45	0.00%	sha256:5f9898b7b82296187dfab6c27d9997190ee35ecde22ad27d4c7e6461d569b715
47			8	1.71	4.47%	1.61	1.85	24.49%	sha256:9df57765a5a854c0425c62288c79d50db2894652025bf332f8930c8687cbbe47
48			16	1.75	2.23%	1.63	2.01	17.96%	sha256:3b71fd43ecd965621a14f92a57a1cacb53b19df12a404facf4892506488d14d5
49			32	1.72	3.91%	1.63	1.82	25.71%	sha256:4e65b6dbbaa88dd71ccbd9ec05d9214f76403d94e0c3d7e0c041be158e7e63e2
50	rabbitmq	101	4	13.8	0.00%	13.59	14.16	0.00%	sha256:0f58d24142516f2f538f2a60d24dcb70fe0d04b33a5935d41e142c5d647b0426
51			8	13.83	-0.22%	13.37	15.67	-10.66%	sha256:5afa71493ac32dba60e20af571266d4e7955579ad7729249d7360e570bd730eb
52			16	13.41	2.83%	13.12	14.19	-0.21%	sha256:bd98c6a3259a52c9f6125482bf14c8113640735a74bf0712b4a93603ebf46dd7
53			32	13.54	1.88%	13.37	14.16	0.00%	sha256:a83ccbd8def554f5cfde10bbd898b35095f12b7472e3ad3baabb314734a16c9a

2. BENCHMARKING WITH ALL DATA BEING LOCAL

	A	B	C	D	E	F	G	H	I
1	image	image size, MiB	span size, MiB	mean total time, s	mean total time diff, %	min total time, s	max total time, s	max total time diff, %	soci index sha
2	tomcat	252	4	3.35	0.00%	3.3	3.4	0.00%	sha256:f6fab4cfb60142401062e50b02b7d22c37c3a3dcd55511ece6893e440fbd84aa
3			8	3.56	-6.27%	3.51	3.61	-6.18%	sha256:62d96589f4cd8cc4eaa714d43c9bbe68d06c086458592852cff778d98e0c4eb4
4			16	4.07	-21.49%	4	4.17	-22.65%	sha256:bc15ca01ebdedb5246356b1dc7520df2adf02d3cd5247b0e0b399ef26ac3372b
5			32	4.67	-39.40%	4.61	4.75	-39.71%	sha256:6ec9cefca8c8f97daac5a10f66eb4b18b51c3c4d5b1f6b1052dda7561b25e29d
6	rethinkdb	48	4	1.35	0.00%	1.32	1.4	0.00%	sha256:50fa6303f78dabc5f6d60bcac24019590f9a005309264d9ad9d0c3cf1a4f9687
7			8	1.44	-6.67%	1.41	1.47	-5.00%	sha256:0a957f0e69545b2459bd837ff3867204243b1a0552880271519fcd4f56ce913d
8			16	1.47	-8.89%	1.43	1.51	-7.86%	sha256:94b3f6ca920a9436b683eeb361291741d136e176163049bd8bd38d9d5492cd5d
9			32	1.6	-18.52%	1.58	1.64	-17.14%	sha256:e7b9af7bb83b56a0efcb3258984dfdd0c4a2bc67f62a184dcbab0085c29a3c4f
10	python	352	4	2.46	0.00%	2.38	2.56	0.00%	sha256:fdde13a373d92092d3208538103c7cdebe0b09aaa2b60c580be387fee3e37318
11			8	2.57	-4.47%	2.48	2.62	-2.34%	sha256:49d1944e69d21c017d6179991519161e3356a776b78fcecacb1a330bb484a891e
12			16	2.85	-15.85%	2.76	2.94	-14.84%	sha256:3fa6de3ccf6dd7bb9f1dad7396af0440de62286cc627959af6ed7b8890b55a2d
13			32	3.38	-37.40%	3.24	3.61	-41.02%	sha256:65dd368cfc36ebcc2f28d5d57e635a925ed0cf1b7eb1f210f16408b7a9ab8b12
14	mongo	232	4	3.41	0.00%	3.38	3.48	0.00%	sha256:a58e81941726547c2f2560c1771d3887131b9528141c93b7a1aa36a0d649006b
15			8	3.5	-2.64%	3.47	3.56	-2.30%	sha256:e3d0fd7cb58ed4fc792af04294ce37e91be7bf20a85de35eb1502aeb2db820c2
16			16	3.5	-2.64%	3.45	3.55	-2.01%	sha256:569b1680be222121e214e4f2e3ec58da7c958f7166b243c053bf1f0d8a992a6a
17			32	3.49	-2.35%	3.46	3.53	-1.44%	sha256:076c966caa12a08f52f3f5842c018ea35a1184140c236392861f5e6eaac69fae
18	jetty	256	4	3.66	0.00%	3.59	3.71	0.00%	sha256:9b9a777ac59528f3ead96d438740dc3b362a73916235f1cd64e572d90d9069a3
19			8	3.92	-7.10%	3.87	3.98	-7.28%	sha256:8d69ff58b7be185b0b7022e69a15a09c10f37e33bc824a459619aa1e18888b0f
20			16	4.36	-19.13%	4.32	4.44	-19.68%	sha256:f51bbe504084822bd8362d96e5f8fb6c1189c31cb3b9ce131dd40ea3c3337a42
21			32	4.95	-35.25%	4.88	5.05	-36.12%	sha256:c1502b141e965cb12c3fe3bed1aa189a948d27f47bb42794d90e7e9fe2f02e7b
22	jenkins	290	4	8.94	0.00%	8.79	9.19	0.00%	sha256:ee48edaaaf9e22f4954f9e16278bb972be61f50f2ec5e9a71febc30f9f2f0f64
23			8	9.039	-1.11%	8.876	9.23	-0.44%	sha256:ee47cfc102b7bd061321bb4cea1026f824764612640909576b5052879cf5ecf3
24			16	9.4	-5.15%	9.25	9.57	-4.13%	sha256:a95a4631da98d659a94fbc566184c974503ee9adf3d3f610d614a4205f3a6ee9
25			32	9.84	-10.07%	9.66	10.1	-9.90%	sha256:e3af8964ce64e0869bb3d4abfeb74125e8162a36dbfce2ce48b46920b4443ff0
26	glassfish	351	4	6.76	0.00%	6.65	6.87	0.00%	sha256:4d85b01b075816bb6031b271e5932a7bb652924b9fc4734326d9efccd38a35ef
27			8	7.08	-4.73%	6.92	7.26	-5.68%	sha256:94e23d6ce13b4418f8d5b7c880e9fc162f39ddb434514d85eeaadcf9840e59d0
28			16	7.65	-13.17%	7.46	7.94	-15.57%	sha256:2bb55adf42d6b66ce146b36206337ab43535bc28cc157c971dc5721746c6f705
29			32	7.8	-15.38%	7.68	7.97	-16.01%	sha256:7715e2bfc0666b9d86c00dab5d77d2c54f0b56df96ad7c287135fcb095abc628
30	ghost	158	4	7.94	0.00%	7.82	8.04	0.00%	sha256:3ec28440ef7f685f9d325973acecb31425deb6993144964d95282b645b1afa70
31			8	8.31	-4.66%	8.16	9.16	-13.93%	sha256:1acc0cc28121ed5a4b66daf9f18c65d486ea29bd06e11624cc0b13d0eab29428
32			16	8.83	-11.21%	8.72	9.05	-12.56%	sha256:1f3cdf51ff65c00c453c03c1363c4eb56688a7f2f02e75042b008f8c1a5c5d10
33			32	8.9	-12.09%	8.85	8.95	-11.32%	sha256:9c3e6e700bd29ff4e5e4837ceddf54f2e5105085060bc4ca162b66732eee881a
34	gcchellocompile	449	4	3.04	0.00%	2.88	3.17	0.00%	sha256:381093aa887aceb8561f99d96abda88ac68f284843fca64f5db66d703fca888e
35			8	3.38	-11.18%	3.29	3.53	-11.36%	sha256:c8d35b84e22a6fe971c61741567fc4a0486d933538a54c4e6f836de73a86d871
36			16	3.83	-25.99%	3.59	3.98	-25.55%	sha256:e71680da0a10f255e1b51ad816dbf76e6a8cb67d99f8da5697a507ea41a2815f
37			32	5.25	-72.70%	5.15	5.39	-70.03%	sha256:a0f8dcb2e92cd08a878ffb0caeb1ccf548d89c4fdc2f04b1e12a06461f6e7457
38	tensorflow	449	4	11.13	0.00%	10.98	11.25	0.00%	sha256:ca132cd2f28296658f70d70e780e860b52cda1afd58353d3d80ee69cfebf6786
39			8	11.61	-4.31%	11.44	11.8	-4.89%	sha256:9b2975a67542aa482a9aca2e9cbe8fa3c8b20bfa3020cf8b2cb2cc9f8dc64e58
40			16	12.16	-9.25%	12.09	12.2	-8.44%	sha256:9b59adc8232c4443f216b5c028e544457e6300a68a942ebb73f2bd5c886af163
41			32	13.35	-19.95%	13.21	13.53	-20.27%	sha256:143f021c68b18b2c748b01ab0a444ebf9778cdb55b98aabc5240771d621944c3
42	drupal	189	4	3.13	0.00%	3.06	3.25	0.00%	sha256:75ce50d1b5e2a27248d609d82ba153d165ad7e8d193a79b9b5c0c8a52fffbc93
43			8	3.29	-5.11%	3.23	3.35	-3.08%	sha256:d85671468d0d1d4e20d84a5c35a661f375857fb7903d9ac96028f3f3e142a72d
44			16	3.63	-15.97%	3.57	3.7	-13.85%	sha256:80f63d437a6a105a66c898e3c90415966b4668c72a203cea01423a9d829ec7cc
45			32	3.99	-27.48%	3.86	4.05	-24.62%	sha256:56474d6288ada1679708c51e9ada97f0310cafa618e0f94a7903c5f38b8797cc
46	redis	42	4	1.03	0.00%	1.025	1.069	0.00%	sha256:5f9898b7b82296187dfab6c27d9997190ee35ecde22ad27d4c7e6461d569b715
47			8	1.23	-19.42%	1.21	1.28	-19.74%	sha256:9df57765a5a854c0425c62288c79d50db2894652025bf332f8930c8687cbbe47
48			16	1.38	-33.98%	1.37	1.42	-32.83%	sha256:3b71fd43ecd965621a14f92a57a1cacb53b19df12a404facf4892506488d14d5
49			32	1.45	-40.78%	1.38	1.51	-41.25%	sha256:4e65b6dbbaa88dd71ccbd9ec05d9214f76403d94e0c3d7e0c041be158e7e63e2
50	rabbitmq	101	4	12.13	0.00%	12.02	12.25	0.00%	sha256:0f58d24142516f2f538f2a60d24dcb70fe0d04b33a5935d41e142c5d647b0426
51			8	12.14	-0.08%	12.05	12.26	-0.08%	sha256:5afa71493ac32dba60e20af571266d4e7955579ad7729249d7360e570bd730eb
52			16	12.29	-1.32%	12.22	12.45	-1.63%	sha256:bd98c6a3259a52c9f6125482bf14c8113640735a74bf0712b4a93603ebf46dd7
53			32	12.42	-2.39%	12.34	12.49	-1.96%	sha256:a83ccbd8def554f5cfde10bbd898b35095f12b7472e3ad3baabb314734a16c9a

Appendix A. Configuration for the benchmarking environment

The benchmarking was run on the dev desktop with the following configuration:

- Host type: m4.2xlarge
- RAM: 32GiB
- snapshotter's commit id: 192f026b6267a08241cfe7e9e9f1346b81a49ccc

Contents of soci_config.toml:

```
image_service_path = "/tmp/containerd-grpc/containerd.sock"
[cri_keychain]
enable_keychain = true
```

Appendix B. Approach on running benchmarking tests with elimination of network calls

First download and start the local container registry:

```
REGISTRY_IMAGE=ghcr.io/oci-playground/registry:v3.0.0-alpha.1
docker run -d -p 5000:5000 --restart always --name registry $REGISTRY_IMAGE
```

Push every image in the local registry:

```
ECR_IMAGE=<image from csv list>
IMAGE=localhost:5000/<image name>
sudo docker pull $ECR_IMAGE
sudo docker tag $ECR_IMAGE $IMAGE
sudo docker push $IMAGE
```

Build indices for every image in the local registry and push them there:

```
sudo ctr i pull $IMAGE --plain-http
sudo out/soci create $IMAGE --span-size <desired span size>
sudo out/soci push $IMAGE --plain-http
```

Note: benchmarking framework will not work with the local registry out of the box, so I made some changes to make it work. The changes can be found as the part of the following commit: https://github.com/vkuzniet/soci-snapshotter/commit/11b4eced6230c3922df1646ccf23f6a9ce00108b

After applying the patch outlined above, just update `/benchmarks/singleImage.csv` with the references on the newly created images (starting with `localhost:5000/`) and execute `make benchmarks`.