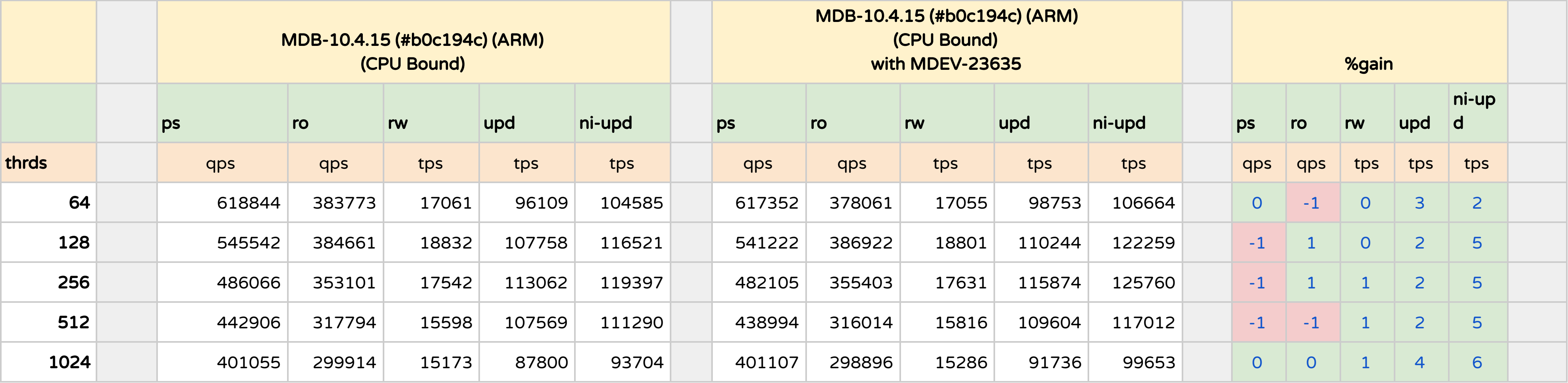
MDEV-23635

1. Proposed MDEV-23635 introduces a notional delay using spin-loop while the flow attempts to register presence of reader by decrement lock-word. Currently the flow enters a tight loop trying till it succeeds. This tight loop is eased out using the said notional delay.
2. Given MDEV-23633 is already folded to the trunk (10.4.15) we will use that as baseline for evaluation of the MDEV-23635.
3. As per the chart below MDEV-23635 seems to have a good positive effect with improvement upto 6%.

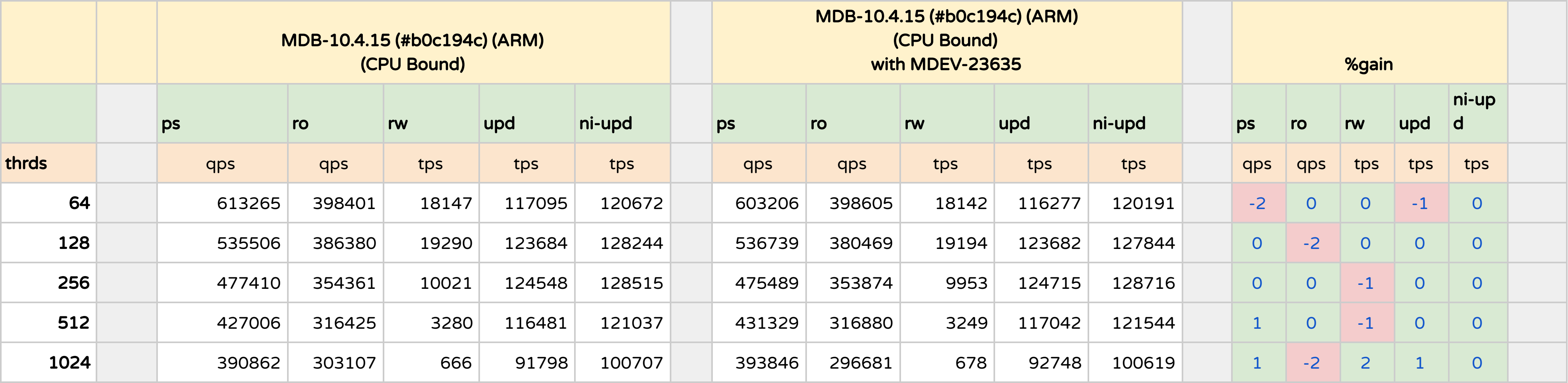
Configuration: ARM BMS (64 cores = 58 (server) + 6 (client), 80 GB buffer pool, 69 GB data, 40 GB redo log) sysbench: uniform



		MDB-10.4.15 (#b0c194c) (ARM) (CPU Bound)						MDB-10.4.15 (#b0c194c) (ARM) (CPU Bound) with MDEV-23635						%gain					
		ps	ro	rw	upd	ni-upd		ps	ro	rw	upd	ni-upd		ps	ro	rw	upd	ni-upd	
thrds		qps	qps	tps	tps	tps		qps	qps	tps	tps	tps		qps	qps	tps	tps	tps	
64		618844	383773	17061	96109	104585		617352	378061	17055	98753	106664		0	-1	0	3	2	
128		545542	384661	18832	107758	116521		541222	386922	18801	110244	122259		-1	1	0	2	5	
256		486066	353101	17542	113062	119397		482105	355403	17631	115874	125760		-1	1	1	2	5	
512		442906	317794	15598	107569	111290		438994	316014	15816	109604	117012		-1	-1	1	2	5	
1024		401055	299914	15173	87800	93704		401107	298896	15286	91736	99653		0	0	1	4	6	

4. Let’s continue to analyze the patch using other workloads. With zipfian there is no regression.

Configuration: ARM BMS (64 cores = 58 (server) + 6 (client), 80 GB buffer pool, 69 GB data, 40 GB redo log) sysbench: zipfian/1.5



		MDB-10.4.15 (#b0c194c) (ARM) (CPU Bound)						MDB-10.4.15 (#b0c194c) (ARM) (CPU Bound) with MDEV-23635						%gain					
		ps	ro	rw	upd	ni-upd		ps	ro	rw	upd	ni-upd		ps	ro	rw	upd	ni-upd	
thrds		qps	qps	tps	tps	tps		qps	qps	tps	tps	tps		qps	qps	tps	tps	tps	
64		613265	398401	18147	117095	120672		603206	398605	18142	116277	120191		-2	0	0	-1	0	
128		535506	386380	19290	123684	128244		536739	380469	19194	123682	127844		0	-2	0	0	0	
256		477410	354361	10021	124548	128515		475489	353874	9953	124715	128716		0	0	-1	0	0	
512		427006	316425	3280	116481	121037		431329	316880	3249	117042	121544		1	0	-1	0	0	
1024		390862	303107	666	91798	100707		393846	296681	678	92748	100619		1	-2	2	1	0	

5. Now let’s try with IO Bound workload. IO bound workload also shows improvement. A noise with point-select is due to heavy flushing from previous update and non-update workload. With improved throughput in update/non-update, flushing is more stressed for the patched case.

Configuration: ARM BMS (64 cores = 58 (server) + 6 (client), 35 GB buffer pool, 69 GB data, 40 GB redo log) sysbench: uniform

	MDB-10.4.15 (#b0c194c) (ARM) (IO Bound)					MDB-10.4.15 (#b0c194c) (ARM) (IO Bound) with MDEV-23635					%gain				
	ps	ro	rw	upd	ni-upd	ps	ro	rw	upd	ni-upd	ps	ro	rw	upd	ni-upd
thrds	qps	qps	tps	tps	tps	qps	qps	tps	tps	tps	qps	qps	tps	tps	tps
64	269361	227500	8397	64992	64933	260534	227139	8477	65371	65593	-3	0	1	1	1
128	345281	274011	6875	45749	52012	344549	273847	6972	46189	53662	0	0	1	1	3
256	425736	320200	2825	39016	49910	425202	320676	2967	39623	49716	0	0	5	2	0
512	412559	293445	890	32670	49377	410248	292001	1045	32415	48383	-1	0	17	-1	-2
1024	371565	268852	484	25808	36802	370339	265189	728	25801	38652	0	-1	50	0	5

6. So the patch is surely helping 10.4.x but what about 10.5.x? 10.5.x already has existing flushing regression so the testing should be done to take some direction only. While update-workload has shown quite good improvement but there are a couple of regressions too and unfortunately it is difficult to predict if these regressions are from increased flushing activity or flushing regression that 10.5.x has. This prompted us to re-run the things for the 2nd time. Regressing point reduced but still present. Improvement too looks impressive with numbers crossing 10% for read-write workload.

Configuration: ARM BMS (64 cores = 58 (server) + 6 (client), 80 GB buffer pool, 69 GB data, 40 GB redo log) sysbench: uniform

	MDB-10.5.6 (#d25f806) (ARM) (CPU Bound)					MDB-10.5.6 (#d25f806) (ARM) (CPU Bound) with MDEV-23635					%gain				
	ps	ro	rw	upd	ni-upd	ps	ro	rw	upd	ni-upd	ps	ro	rw	upd	ni-upd
thrds	qps	qps	tps	tps	tps	qps	qps	tps	tps	tps	qps	qps	tps	tps	tps
64	568369	355819	15044	125864	204496	552668	362316	14774	134359	198569	-3	2	-2	7	-3
128	523722	375366	18867	126177	227088	524056	377917	18719	147278	233683	0	1	-1	17	3
256	481119	345365	17234	183657	213585	472189	346461	17173	166246	229162	-2	0	0	-9	7
512	436176	314529	15308	210129	204110	429755	311510	15254	225026	195717	-1	-1	0	7	-4
1024	389534	300373	13424	203444	170846	387843	294673	13896	230837	184859	0	-2	4	13	8

Configuration: ARM BMS (64 cores = 58 (server) + 6 (client), 80 GB buffer pool, 69 GB data, 40 GB redo log) sysbench: uniform

	MDB-10.5.6 (#d25f806) (ARM) (CPU Bound)					MDB-10.5.6 (#d25f806) (ARM) (CPU Bound) with MDEV-23635					%gain				
	ps	ro	rw	upd	ni-upd	ps	ro	rw	upd	ni-upd	ps	ro	rw	upd	ni-upd
thrds	qps	qps	tps	tps	tps	qps	qps	tps	tps	tps	qps	qps	tps	tps	tps
64	561657	357952	15097	130526	196894	574935	361196	15055	151930	200338	2	1	0	16	2
128	520051	378404	18879	139647	200211	523114	379155	18339	161469	203696	1	0	-3	16	2
256	482545	349997	17243	127646	242540	473978	351609	17173	145383	218320	-2	0	0	14	-10
512	429042	313883	15415	159194	189480	427543	307360	15378	171715	215920	0	-2	0	8	14
1024	385864	297835	13757	216886	186207	390894	297629	14002	211918	182515	1	0	2	-2	-2

7. So assuming we will proceed with this patch for ARM on 10.4.x and 10.5.x we now need to evaluate if this patch should be also enabled for x86.

Configuration: x86-BMS (28 cores = 22 (server) + 6 (client), 80 GB buffer pool, 69 GB data, 40 GB redo log) sysbench=uniform

		MDB-10.4.15 (#b0c194c) (x86) (CPU Bound)						MDB-10.4.15 (#b0c194c) (x86) (CPU Bound) with MDEV-23635						%gain					
		ps	ro	rw	upd	ni-upd		ps	ro	rw	upd	ni-upd		ps	ro	rw	upd	ni-upd	
thrds		qps	qps	tps	tps	tps		qps	qps	tps	tps	tps		qps	qps	tps	tps	tps	
64		553411	309527	12161	52899	62298		557491	307970	12158	53306	63536		1	-1	0	1	2	
128		539244	301453	11952	51007	56449		544356	300298	11934	51175	56885		1	0	0	0	1	
256		522150	288916	12195	46066	49791		525451	288039	12188	46047	49982		1	0	0	0	0	
512		501643	272532	11919	43003	46311		505972	271776	11901	43104	46360		1	0	0	0	0	
1024		463302	257553	10890	39187	43974		463053	256944	10977	39215	43910		0	0	1	0	0	

Configuration: x86-BMS (28 cores = 22 (server) + 6 (client), 80 GB buffer pool, 69 GB data, 40 GB redo log) sysbench=zipfian/1.5

		MDB-10.4.15 (#b0c194c) (x86) (CPU Bound)						MDB-10.4.15 (#b0c194c) (x86) (CPU Bound) with MDEV-23635						%gain					
		ps	ro	rw	upd	ni-upd		ps	ro	rw	upd	ni-upd		ps	ro	rw	upd	ni-upd	
thrds		qps	qps	tps	tps	tps		qps	qps	tps	tps	tps		qps	qps	tps	tps	tps	
64		601080	330484	13615	62572	66876		604578	330830	13652	62862	67550		1	0	0	0	1	
128		590295	324118	13507	54108	56771		594763	323782	13513	54063	57188		1	0	0	0	1	
256		573779	309022	8260	48790	50216		577844	308897	8301	49039	50610		1	0	0	1	1	
512		542765	290617	3409	45647	47243		545513	290563	3437	45901	47583		1	0	1	1	1	
1024		474344	274388	966	43408	45726		464260	273992	968	43590	46224		-2	0	0	0	1	

8. With x86, 10.4.x series showed some improvement that means it is safe to adapt it for 10.4.x (x86). What about 10.5.x.

Configuration: x86-BMS (28 cores = 22 (server) + 6 (client), 80 GB buffer pool, 69 GB data, 40 GB redo log) sysbench=uniform

		MDB-10.5.6 (#d25f806) (x86) (CPU Bound)						MDB-10.5.6 (#d25f806) (x86) (CPU Bound) with MDEV-23635						%gain					
		ps	ro	rw	upd	ni-upd		ps	ro	rw	upd	ni-upd		ps	ro	rw	upd	ni-upd	
thrds		qps	qps	tps	tps	tps		qps	qps	tps	tps	tps		qps	qps	tps	tps	tps	
64		500822	278042	12941	134700	147584		502002	280808	12813	135508	157108		0	1	-1	1	6	
128		504481	281830	13682	154891	166183		506134	284940	13655	157516	177072		0	1	0	2	7	
256		501904	275149	13592	172168	207368		503858	277238	13570	170814	207170		0	1	0	-1	0	
512		484269	262617	13103	173626	174410		486417	263572	13147	171024	177173		0	0	0	-1	2	
1024		472622	249520	12450	155584	150412		473391	249378	12464	154698	148439		0	0	0	-1	-1	

VERDICT: Patch is safe to be applied for ARM + x86 on 10.4.x and 10.5.x