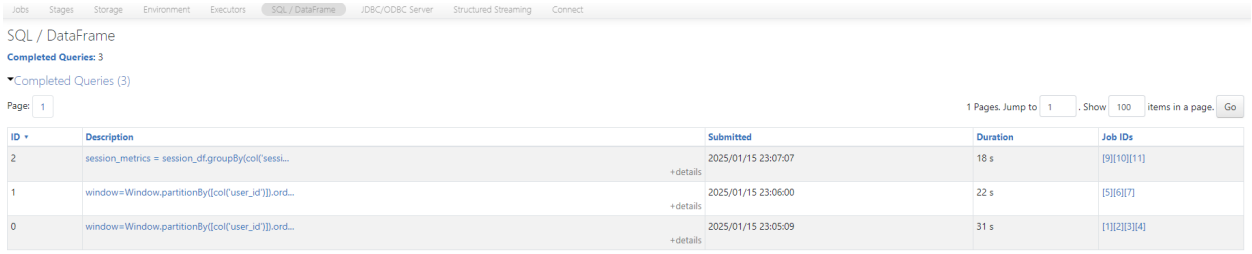
Pre-Optimization:
There are total of 12 jobs created for the pipeline. The total execution time amounts to 71 seconds.


Jobs Stages Storage Environment Executors SQL / DataFrame JDBC/ODBC Server Structured Streaming Connect

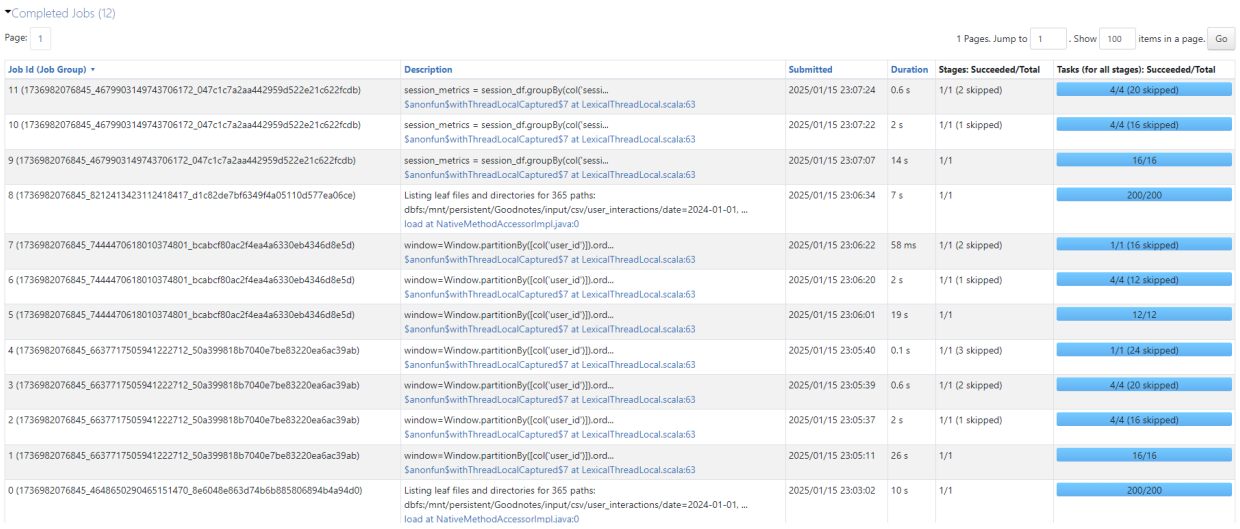
SQL / DataFrame

Completed Queries: 3

Completed Queries (3)

Page: 1 — 1 Pages. Jump to 1 . Show 100 items in a page. Go

ID	Description	Submitted	Duration	Job IDs
2	session_metrics = session_df.groupBy(col('sessi... +details	2025/01/15 23:07:07	18 s	[9][10][11]
1	window=Window.partitionBy([col('user_id')]).ord... +details	2025/01/15 23:06:00	22 s	[5][6][7]
0	window=Window.partitionBy([col('user_id')]).ord... +details	2025/01/15 23:05:09	31 s	[1][2][3][4]


Completed Jobs (12)

Page: 1 — 1 Pages. Jump to 1 . Show 100 items in a page. Go

Job Id (Job Group)	Description	Submitted	Duration	Stages: Succeeded/Total	Tasks (for all stages): Succeeded/Total
11 (1736982076845_467990314974370617 2_047c1c7a2aa442959d522e21c622fcdb)	session_metrics = session_df.groupBy(col('sessi... $anonfun$withThreadLocalCaptured$7 at LexicalThreadLocal.scala:63	2025/01/15 23:07:24	0.6 s	1/1 (2 skipped)	4/4 (20 skipped)
10 (1736982076845_467990314974370617 2_047c1c7a2aa442959d522e21c622fcdb)	session_metrics = session_df.groupBy(col('sessi... $anonfun$withThreadLocalCaptured$7 at LexicalThreadLocal.scala:63	2025/01/15 23:07:22	2 s	1/1 (1 skipped)	4/4 (16 skipped)
9 (1736982076845_467990314974370617 2_047c1c7a2aa442959d522e21c622fcdb)	session_metrics = session_df.groupBy(col('sessi... $anonfun$withThreadLocalCaptured$7 at LexicalThreadLocal.scala:63	2025/01/15 23:07:07	14 s	1/1	16/16
8 (1736982076845_821241342311241841 7_d1c82de7bf6349f4a05110d577ea06ce)	Listing leaf files and directories for 365 paths: dbfs:/mnt/persistent/Goodnotes/input/csv/user_interactions/date=2024-01-01, ... load at NativeMethodAccessorImpl.java:0	2025/01/15 23:06:34	7 s	1/1	200/200
7 (1736982076845_744447061801037480 1_bcabcf80ac2f4ea4a6330eb4346d8e5d)	window=Window.partitionBy([col('user_id')]).ord... $anonfun$withThreadLocalCaptured$7 at LexicalThreadLocal.scala:63	2025/01/15 23:06:22	58 ms	1/1 (2 skipped)	1/1 (16 skipped)
6 (1736982076845_744447061801037480 1_bcabcf80ac2f4ea4a6330eb4346d8e5d)	window=Window.partitionBy([col('user_id')]).ord... $anonfun$withThreadLocalCaptured$7 at LexicalThreadLocal.scala:63	2025/01/15 23:06:20	2 s	1/1 (1 skipped)	4/4 (12 skipped)
5 (1736982076845_744447061801037480 1_bcabcf80ac2f4ea4a6330eb4346d8e5d)	window=Window.partitionBy([col('user_id')]).ord... $anonfun$withThreadLocalCaptured$7 at LexicalThreadLocal.scala:63	2025/01/15 23:06:01	19 s	1/1	12/12
4 (1736982076845_663771750594122271 2_50a399818b7040e7be83220ea6ac39ab)	window=Window.partitionBy([col('user_id')]).ord... $anonfun$withThreadLocalCaptured$7 at LexicalThreadLocal.scala:63	2025/01/15 23:05:40	0.1 s	1/1 (3 skipped)	1/1 (24 skipped)
3 (1736982076845_663771750594122271 2_50a399818b7040e7be83220ea6ac39ab)	window=Window.partitionBy([col('user_id')]).ord... $anonfun$withThreadLocalCaptured$7 at LexicalThreadLocal.scala:63	2025/01/15 23:05:39	0.6 s	1/1 (2 skipped)	4/4 (20 skipped)
2 (1736982076845_663771750594122271 2_50a399818b7040e7be83220ea6ac39ab)	window=Window.partitionBy([col('user_id')]).ord... $anonfun$withThreadLocalCaptured$7 at LexicalThreadLocal.scala:63	2025/01/15 23:05:37	2 s	1/1 (1 skipped)	4/4 (16 skipped)
1 (1736982076845_663771750594122271 2_50a399818b7040e7be83220ea6ac39ab)	window=Window.partitionBy([col('user_id')]).ord... $anonfun$withThreadLocalCaptured$7 at LexicalThreadLocal.scala:63	2025/01/15 23:05:11	26 s	1/1	16/16
0 (1736982076845_464865029046515147 0_8e6048e863d74b6b885806894b4a94d0)	Listing leaf files and directories for 365 paths: dbfs:/mnt/persistent/Goodnotes/input/csv/user_interactions/date=2024-01-01, ... load at NativeMethodAccessorImpl.java:0	2025/01/15 23:03:02	10 s	1/1	200/200

While creating user_interactions_df there are 200 tasks generated which means that spark is reading from many small files. The optimization is to repartition data while writing such reading the data becomes efficient.

Index	Task ID	Attempt	Status	Locality level	Executor ID	Host	Logs	Launch Time	Duration	GC Time	Shuffle Read Fetch Wait Time	Shuffle Remote Reads	Input Size / Records	Shuffle Write Time	Shuffle Write Size / Records	Errors
0	200	0	SUCCESS	PROCESS_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:05:11	8 s		0.0 ms	0.0 B	4 MiB / 69781	0.1 s	751.8 KiB / 69556	
1	201	0	SUCCESS	PROCESS_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:05:11	8 s		0.0 ms	0.0 B	4 MiB / 68885	59.0 ms	744.3 KiB / 68660	
2	202	0	SUCCESS	PROCESS_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:05:11	8 s		0.0 ms	0.0 B	3.9 MiB / 68198	70.0 ms	734.2 KiB / 67998	
3	203	0	SUCCESS	PROCESS_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:05:11	8 s		0.0 ms	0.0 B	3.9 MiB / 67477	0.1 s	728.7 KiB / 67281	
4	204	0	SUCCESS	PROCESS_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:05:19	6 s	0.2 s	0.0 ms	0.0 B	3.9 MiB / 66940	32.0 ms	722.1 KiB / 66727	
5	205	0	SUCCESS	PROCESS_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:05:19	6 s	0.2 s	0.0 ms	0.0 B	3.8 MiB / 66485	69.0 ms	717.2 KiB / 66267	
6	206	0	SUCCESS	PROCESS_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:05:19	6 s	0.2 s	0.0 ms	0.0 B	3.8 MiB / 66049	72.0 ms	714.6 KiB / 65872	
7	207	0	SUCCESS	PROCESS_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:05:19	6 s	0.2 s	0.0 ms	0.0 B	3.8 MiB / 65671	40.0 ms	709 KiB / 65460	
8	208	0	SUCCESS	PROCESS_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:05:25	6 s		0.0 ms	0.0 B	3.8 MiB / 65287	36.0 ms	706.6 KiB / 65101	
9	209	0	SUCCESS	PROCESS_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:05:25	6 s		0.0 ms	0.0 B	3.8 MiB / 64871	33.0 ms	703.4 KiB / 64686	
10	210	0	SUCCESS	PROCESS_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:05:25	5 s		0.0 ms	0.0 B	3.7 MiB / 64463	28.0 ms	697.2 KiB / 64252	
11	211	0	SUCCESS	PROCESS_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:05:26	6 s		0.0 ms	0.0 B	3.7 MiB / 63982	35.0 ms	694.4 KiB / 63793	
12	212	0	SUCCESS	PROCESS_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:05:31	6 s	0.2 s	0.0 ms	0.0 B	3.7 MiB / 63310	21.0 ms	687 KiB / 63136	
13	213	0	SUCCESS	PROCESS_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:05:31	5 s	0.2 s	0.0 ms	0.0 B	3.6 MiB / 62613	23.0 ms	681.5 KiB / 62445	
14	214	0	SUCCESS	PROCESS_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:05:31	5 s	0.2 s	0.0 ms	0.0 B	3.6 MiB / 61491	18.0 ms	669.8 KiB / 61323	
15	215	0	SUCCESS	PROCESS_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:05:31	1 s		0.0 ms	0.0 B	773 KiB / 13037	25.0 ms	163.8 KiB / 13025	

Spark is not efficiently reading the data and small input size are being read into these 16 partitions. We can solve this by coalescing the partitions to half the size while writing the data such that it reads into less number of partitions with efficient input size/records. In the next step while wholestagecodegen sparks coalesced the data into 4 partitions.

Index	Task ID	Attempt	Status	Locality level	Executor ID	Host	Logs	Launch Time	Duration	GC Time	Shuffle Read Fetch Wait Time	Shuffle Remote Reads	Shuffle Write Time	Shuffle Write Size / Records	Shuffle Read Size / Records	Errors
0	218	0	SUCCESS	PROCESS_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:05:37	2 s		0.0 ms	0.0 B	53.0 ms	2.1 MiB / 235157	2.6 MiB / 244243	
1	219	0	SUCCESS	PROCESS_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:05:37	2 s		0.0 ms	0.0 B	37.0 ms	2.1 MiB / 234797	2.6 MiB / 243972	
2	217	0	SUCCESS	PROCESS_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:05:37	2 s		0.0 ms	0.0 B	38.0 ms	2.2 MiB / 239408	2.6 MiB / 248561	
3	216	0	SUCCESS	PROCESS_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:05:37	2 s		0.0 ms	0.0 B	42.0 ms	2.2 MiB / 249110	2.7 MiB / 258806	

Post-Optimization:

There are total of 12 jobs created for the pipeline. The total execution time amounts to 35 seconds which is half the time of pre-optimization make it 50% efficient just by reading from a columnar file format like Parquet.

I had manually set the number of partition via colaesce(4) and insted of 16 partitions like before it read a good chunk of data into these 4 partitions.

Index	Task ID	Attempt	Status	Locality level	Executor ID	Host	Logs	Launch Time	Duration	GC Time	Shuffle Read Fetch Wait Time	Shuffle Remote Reads	Input Size / Records	Shuffle Write Time	Shuffle Write Size / Records	Errors
0	666	0	SUCCESS	NODE_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:42:56	22 s		0.0 ms	0.0 B	10.9 MiB / 241749	24.0 ms	2.5 MiB / 241295	
1	667	0	SUCCESS	NODE_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:42:56	25 s		0.0 ms	0.0 B	12 MiB / 266412	17.0 ms	2.7 MiB / 265979	
2	668	0	SUCCESS	NODE_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:42:56	25 s		0.0 ms	0.0 B	11.8 MiB / 259635	18.0 ms	2.6 MiB / 259175	
3	669	0	SUCCESS	NODE_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:42:56	23 s		0.0 ms	0.0 B	10.6 MiB / 232204	28.0 ms	2.4 MiB / 231769	

The downstream reads are actually reading less amount of data and thus untlimately leads to decrease in computation time.

Input Size / Records: 0.0 B / 1000000
Shuffle Write Size / Records: 10.3 MiB / 1000000
Associated Job Ids: 17

▸ DAG Visualization
▸ Show Additional Metrics
▸ Event Timeline

Summary Metrics for 4 Completed Tasks

Metric	Min	25th percentile	Median	75th percentile	Max
Duration	1 s	2 s	2 s	2 s	2 s
GC Time	76.0 ms	76.0 ms	76.0 ms	76.0 ms	76.0 ms
Input Size / Records	0.0 B / 230484	0.0 B / 242891	0.0 B / 259908	0.0 B / 266717	0.0 B / 266717
Shuffle Write Size / Records	2.4 MiB / 230484	2.5 MiB / 242891	2.7 MiB / 259908	2.7 MiB / 266717	2.7 MiB / 266717
Shuffle Write Time	26.0 ms	33.0 ms	45.0 ms	54.0 ms	54.0 ms

▾ Aggregated Metrics by Executor

Show 20 entries Search:

Executor ID	Logs	Address	Task Time	Total Tasks	Failed Tasks	Killed Tasks	Succeeded Tasks	Excluded	Input Size / Records	Shuffle Write Size / Records
driver		10.177.200.134:35449	6 s	4	0	0	4	false	0.0 B / 1000000	10.3 MiB / 1000000

Showing 1 to 1 of 1 entries Previous

Tasks (4)

Show All entries Search:

Index	Task ID	Attempt	Status	Locality level	Executor ID	Host	Logs	Launch Time	Duration	GC Time	Shuffle Read Fetch Wait Time	Shuffle Remote Reads	Input Size / Records	Shuffle Write Time	Shuffle Write Size / Records
0	679	0	SUCCESS	NODE_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:43:27	1 s	76.0 ms	0.0 ms	0.0 B	0.0 B / 230484	54.0 ms	2.4 MiB / 230484
1	680	0	SUCCESS	NODE_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:43:27	2 s	76.0 ms	0.0 ms	0.0 B	0.0 B / 266717	26.0 ms	2.7 MiB / 266717
2	681	0	SUCCESS	NODE_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:43:27	2 s	76.0 ms	0.0 ms	0.0 B	0.0 B / 259908	33.0 ms	2.7 MiB / 259908
3	682	0	SUCCESS	NODE_LOCAL	driver	ip-10-177-200-134.eu-west-1.compute.internal		2025-01-15 23:43:27	2 s	76.0 ms	0.0 ms	0.0 B	0.0 B / 242891	45.0 ms	2.5 MiB / 242891

Conclusion: Reading from Parquet/ORC indirectly benefits the pipeline to execute faster saving us on computational cost and writing to Parquet/ORC would save us storage costs on the cloud as the data is compressed and stored making the file size smaller compared to text based file format.