

Technical Information Sheet

SQL on Hadoop benchmarks using TPC-DS query set

This document supports the “Kognitio SQL on Hadoop benchmarks using the TPC-DS query set” whitepaper.¹

It contains the following information:

- Benchmark Architecture – details of the 12 node system used in all the benchmarks
- Benchmark Deployment – Overview and schematic of the benchmarks for each platform: Impala, Kognitio and Spark
- Individual Query Timings – The query timings for each of the 99 TPC-DS queries on each of the platforms

Benchmark Architecture

The 3 benchmarks: Impala, Kognitio and Spark were executed on the same 12 node system. The hardware utilised was standard commodity hardware. This system had Cloudera CDH 5.8.2 installed deployed with one edge (master) node and 11 worker nodes. Each node is a standard ProLiant BL465c G7 blade with the following specifications:

- 550GB available disk
- 256GB RAM.
- 24 cores

YARN has 210GB per node allocated for container memory; a system total of 2.26TB and 264 vcores available for containers.

Benchmark Deployment

Each of the benchmarks was run on the system with the other 2 systems stopped. This allowed the platform to utilise all of the available resources available to it during the benchmark.

In all cases the 1TB TPC-DS data set was generated using the data generator (dsdgen) provided as part of the TPC-DS benchmarking tool suite.

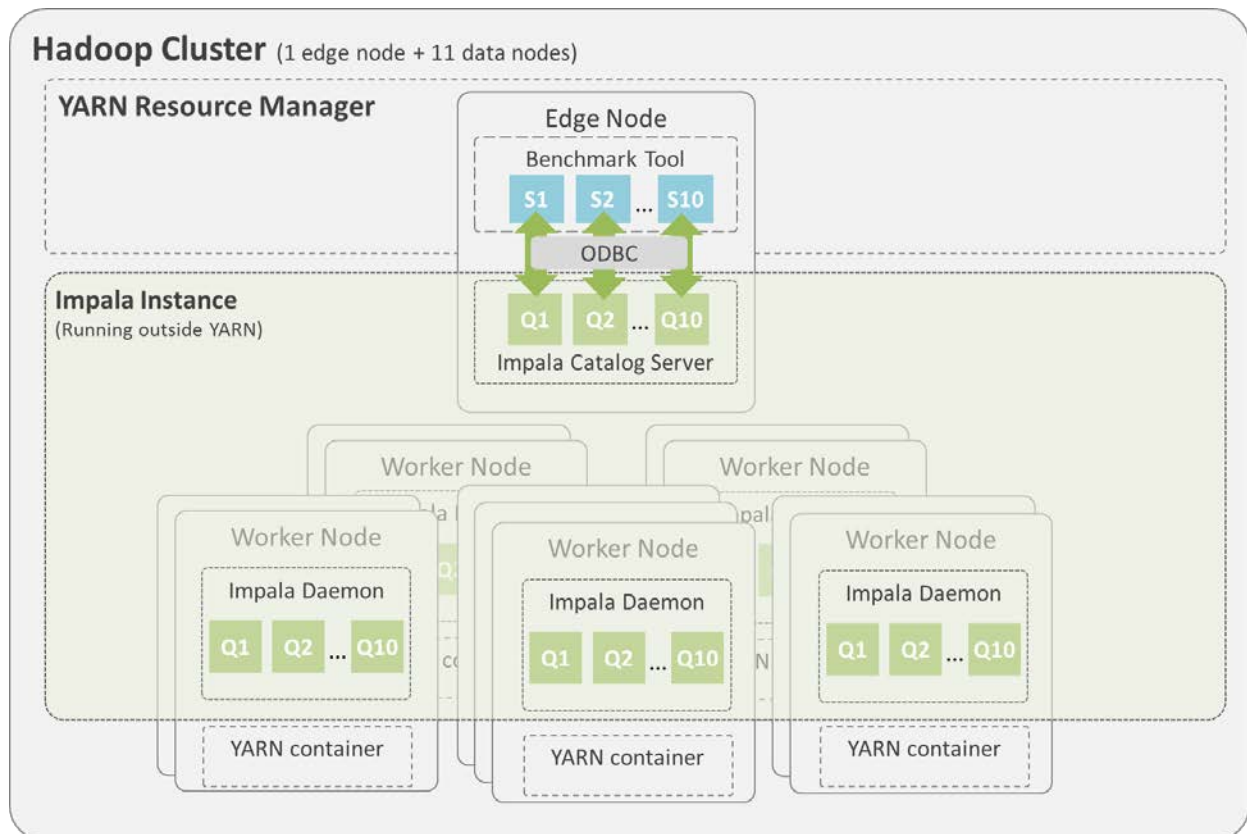
In all cases the TPC-DS query generation tool (dsqgen) was utilised to generate the queries. This tool generates a script for each query stream that randomises the order of the 99 queries in each script. As well as randomising the order of the queries the tool is designed to insert randomised values for parameters in each of the queries. This ensures the benchmark is a truly mixed workload. For more details on how this tool works see <http://www.tpc.org/tpcds/>.

Small syntax changes were done such as adding aliases for derived tables, direct swaps for functions (i.e. CAT for ||) and editing when reserved words used but query rewriting was not allowed.

The 3 schematics indicating how each of the platforms was deployed follow.

¹ You can download the whitepaper from: <http://bit.ly/kognitio-benchmark-wp>

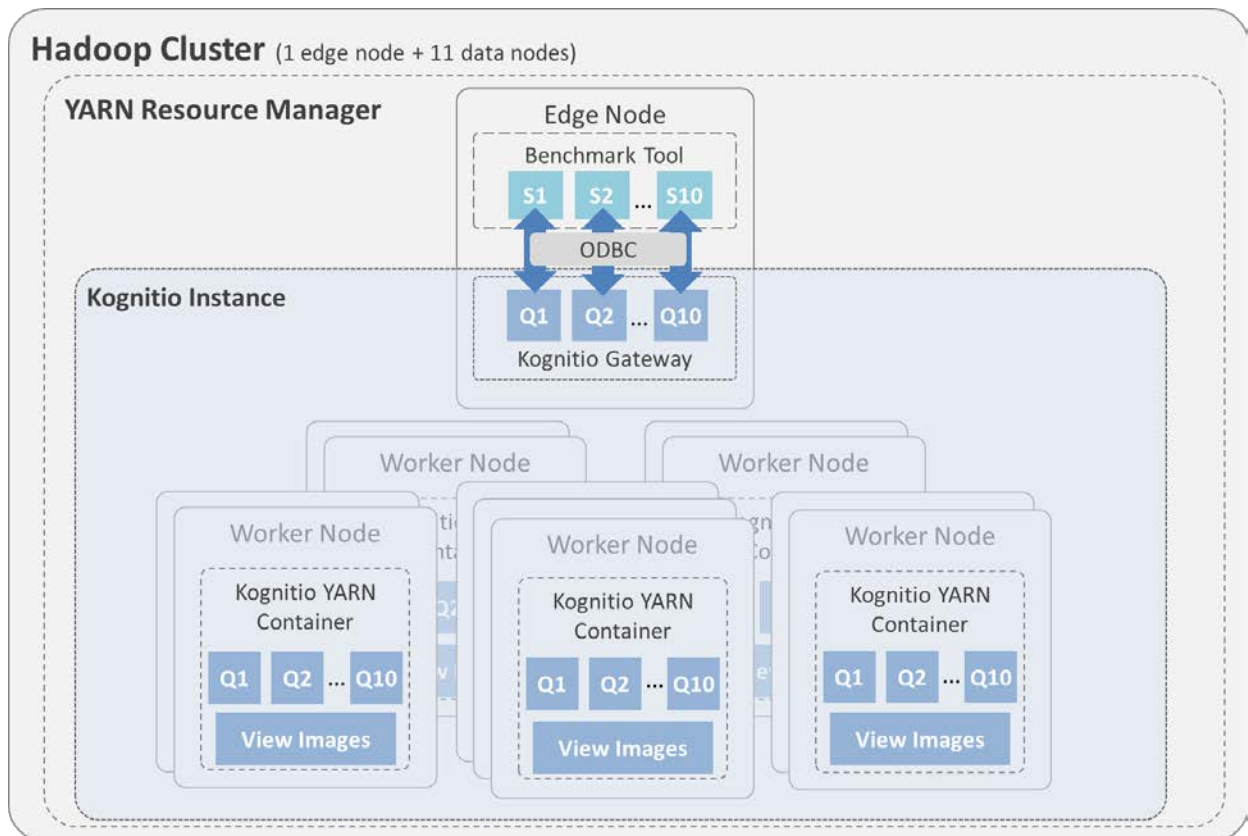
Impala



Notes:

- Impala 2.6.0 version was used. This is the version shipped by Cloudera in CDH 5.8.2
- Impala was run outside of the YARN resource manager on Hadoop.
- Data was held in Hive parquet formatted files. The largest tables were partitioned on the columns most commonly used in joins.
- Statistics for Hive and Impala were both gathered.
- Queries were submitted from the edge node using the `impala-shell` command line tool for each of the randomised query streams in the benchmark

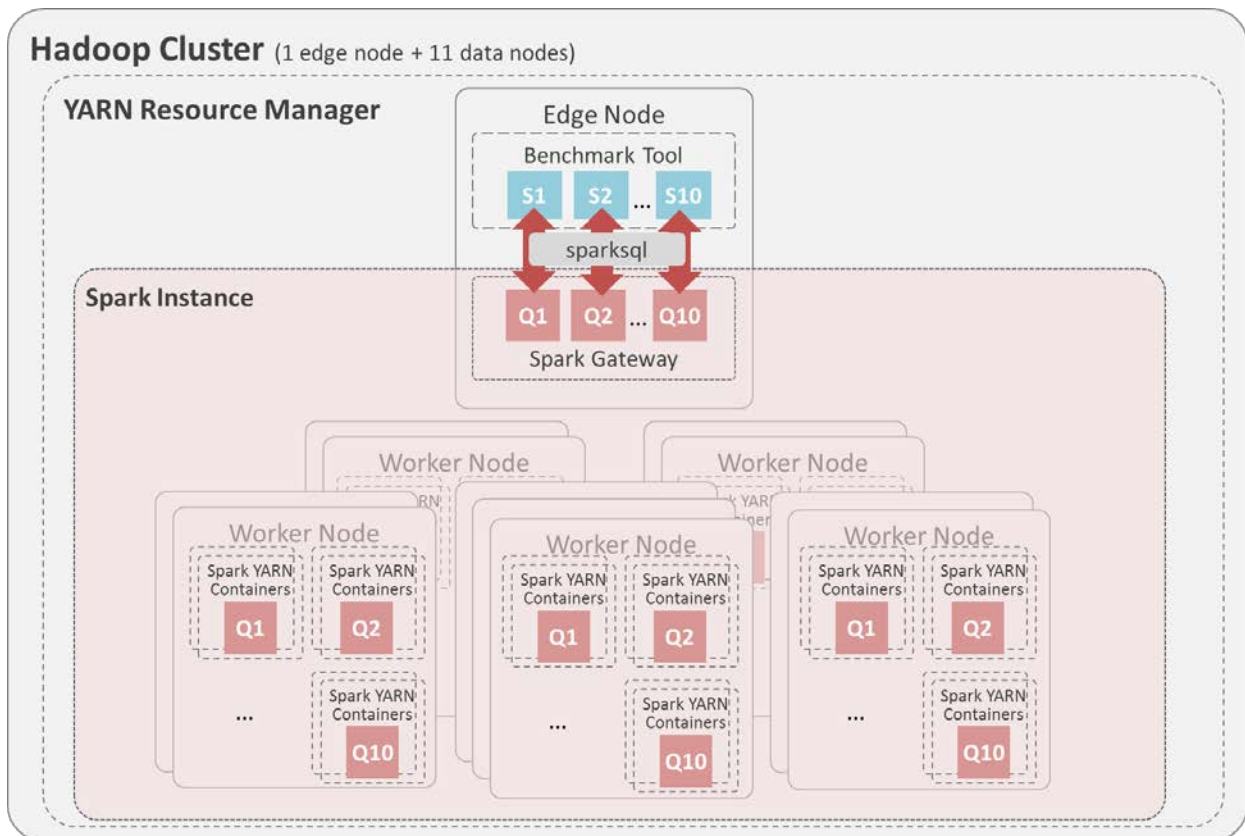
Kognitio



Notes:

- Kognitio version 8.1.50-rel20170105 was used This is the current version available for download at <http://kognitio.com/on-hadoop/> .
- Kognitio is deployed within the YARN resource manager.
- Data was held in Kognitio RAM view images. The larger data sets were hashed on the columns most commonly used in the joins. These reside within the Kognitio YARN containers and can be utilised by multiple queries.
- Kognitio statistics were collected on all views.
- Queries were submitted from the edge node using the Kognitio command line tool wxsubmit for each of the randomised query streams in the benchmark.
- Each query is executed within all containers in the remaining RAM available (not utilised by view images)

Spark



Notes:

- Spark version used was 2.0. This version was in beta test on Cloudera and is available here: <https://blog.cloudera.com/blog/2016/09/apache-spark-2-0-beta-now-available-for-cdh/>
- Spark was deployed within the YARN resource manager on Hadoop.
- Data was held in Hive parquet formatted files. The largest tables were partitioned on the columns most commonly used in joins.
- Statistics for Hive were gathered.
- Queries were submitted from the edge node using the sparksql command line tool for each query in the randomised query streams of the benchmark
- When a query is submitted to Spark it requests containers from YARN to satisfy that query depending on the resources it estimates that it requires. This can lead to many containers on each node for each query. The vast majority of the errors encountered in the Spark benchmark were when it could not obtain the resources from YARN it requested.

Individual Query Times for single stream @ 1TB

Query Number	Execution Time (S)		
	Impala	Kognitio	Spark
1	7.8	3.1	55.0
2	53.4	9.9	56.4
3	6.5	2.4	63.9
4	123.8	95.7	737.53
5	no ROLLUP support	6.0	125.8
6	163.5	12.2	157.7
7	12.7	4.0	72.6
8	no INTERSECT support	5.2	76.0
9	sub query error	22.7	329.2
10	sub query error	7.2	timeout error
11	60.2	60.7	368.57
12	3.1	2.7	64.4
13	long running	8.1	timeout error
14	no INTERSECT support	47.9	timeout error
15	14.1	3.2	571.1
16	14.2	11.7	timeout error
17	44.0	3.7	453.8
18	no ROLLUP support	9.0	83.6
19	9.0	5.3	95.5
20	4.8	3.2	85.0
21	4.1	0.8	36.0
22	no ROLLUP support	2.7	49.9
23	sub query error	81.9	timeout error
24	sub query error	51.0	689.7
25	35.9	3.2	443.7

Query Number	Execution Time (S)		
	Impala	Kognitio	Spark
26	8.7	2.9	49.6
27	no GROUPING SETS support	3.9	71.2
28	33.3	13.7	199.5
29	69.5	3.3	448.4
30	6.5	10.9	65.4
31	40.4	26.7	132.4
32	9.1	2.4	49.5
33	13.7	18.9	86.2
34	12.4	5.3	70.5
35	sub query error	20.5	timeout error
36	no GROUPING SETS support	5.2	72.8
37	10.6	1.4	51.6
38	no INTERSECT support	15.5	92.8
39	17.6	3.5	57.8
40	19.2	2.1	149.0
41	no correlated predicates	0.8	19.3
42	6.6	1.9	61.6
43	9.9	3.1	62.2
44	sub query error	4.0	137.9
45	sub query error	3.0	84.7
46	19.7	5.0	78.8
47	76.4	14.7	133.9
48	long running	8.5	timeout error
49	21.5	3.2	136.5
50	37.2	3.2	281.7

Individual Query Times for single stream @ 1TB (continued)

Query Number	Execution Time (S)		
	Impala	Kognitio	Spark
51	7.8	3.1	104.4
52	6.5	2.0	52.2
53	6.6	4.9	57.9
54	sub query error	11.1	161.5
55	6.7	2.1	50.6
56	13.7	18.5	76.9
57	48.9	11.4	83.0
58	sub query error	6.7	69.9
59	57.9	9.1	74.7
60	14.3	13.5	74.8
61	17.2	15.2	73.8
62	10.0	2.1	52.0
63	6.7	4.9	57.3
64	99.5	13.6	long running
65	56.2	7.2	127.0
66	10.8	2.8	81.9
67	no ROLLUP support	1,060.0	556.4
68	15.2	4.4	67.7
69	21.3	9.1	timeout error
70	no GROUPING SETS support	10.6	66.1
71	19.3	7.0	70.1
72	868.5	124.5	741
73	8.9	3.3	56.7
74	56.1	22.8	175.5
75	78.8	16.0	133.3

Query Number	Execution Time (S)		
	Impala	Kognitio	Spark
76	18.2	5.0	65.2
77	no ROLLUP support	7.2	incorrect join
78	304.8	70.7	594.4
79	14.2	6.3	67.9
80	no ROLLUP support	6.2	512.8
81	5.7	11.8	68.6
82	16.1	3.0	87.1
83	15.7	3.9	48.9
84	74.2	6.3	51.4
85	33.5	8.5	timeout error
86	no GROUPING SETS support	3.3	51.9
87	no EXCEPT support	15.7	90.4
88	21.0	20.4	69.5
89	7.3	5.9	207.2
90	3.8	2.1	46.0
91	3.1	6.9	61.0
92	4.7	1.6	52.6
93	88.3	2.9	322.7
94	11.2	6.7	591.9
95	31.8	22.4	462.0
96	4.5	2.4	45.5
97	67.7	10.9	95.9
98	8.4	6.0	171.8
99	23.4	3.2	54.0
Queries Run	73	99	89
Fastest Query Count	6	92	1

Average Query Times for 10 streams @ 1 TB

Query Number	Execution Time (S)		
	Impala	Kognitio	Spark
1	21.4	16.8	171.2
2	80.7	83.2	159.6
3	74.0	23.9	127.6
4	long running on all platforms		
5	no ROLLUP support	66.9	312.6
6	long running	80.0	544.0
7	158.2	29.7	206.4
8	no INTERSECT support	37.3	182.2
9	sub query error	126.8	long running
10	sub query error	55.9	timeout error
11	long running on all platforms		
12	23.2	14.4	369.9
13	long running	35.1	timeout error
14	no INTERSECT support	long running	timeout error
15	61.7	20.5	536.7
16	108.9	67.5	timeout error
17	200.8	37.8	1,082.0
18	no ROLLUP support	49.3	291.1
19	101.9	31.4	287.7
20	35.3	21.3	280.4
21	14.8	6.0	109.2
22	no ROLLUP support	22.7	347.2
23	sub query error	long running	timeout error
24	sub query error	632.3	long running
25	180.0	37.7	1,178.0

Query Number	Execution Time (S)		
	Impala	Kognitio	Spark
26	92.6	21.4	112.1
27	no GROUPING SETS support	26.6	144.1
28	101.8	74.7	202.2
29	173.9	34.9	991.5
30	13.7	71.6	338.6
31	114.0	159.1	275.9
32	88.4	27.0	184.8
33	85.1	113.4	227.7
34	117.5	26.0	182.0
35	sub query error	140.3	timeout error
36	no GROUPING SETS support	33.9	177.3
37	30.0	17.5	152.8
38	no INTERSECT support	88.5	273.9
39	35.7	19.9	437.6
40	76.3	14.1	438.8
41	no correlated predicates	8.1	62.9
42	62.7	23.9	134.6
43	71.5	24.0	195.0
44	sub query error	25.0	133.0
45	sub query error	21.9	467.1
46	168.3	34.6	208.5
47	127.2	101.2	321.3
48	long running	59.1	timeout error
49	114.3	28.6	372.2
50	145.0	29.0	633.1

Average Query Times for 10 streams @ 1 TB (continued)

Query Number	Execution Time (S)		
	Impala	Kognitio	Spark
51	160.1	70.5	447.0
52	67.5	19.5	163.3
53	93.1	24.3	143.0
54	sub query error	49.4	457.8
55	67.8	21.7	134.5
56	80.0	109.3	213.8
57	85.8	71.9	199.2
58	sub query error	46.4	257.1
59	93.5	68.6	223.0
60	87.7	123.5	239.2
61	106.2	73.7	223.7
62	27.7	16.4	120.9
63	82.6	25.5	145.4
64	332.1	77.4	long running
65	142.6	56.5	447.4
66	69.8	26.7	206.9
67	no ROLLUP support	long running	
68	167.2	30.0	241.6
69	96.9	67.7	timeout error
70	no GROUPING SETS support	76.9	164.4
71	180.9	36.9	200.7
72	long running on all platforms		
73	90.0	24.1	210.8
74	106.2	140.3	503.0
75	276.8	74.9	462.1

Query Number	Execution Time (S)		
	Impala	Kognitio	Spark
76	123.2	30.8	170.5
77	no ROLLUP support	57.6	incorrect join
78	long running on all platforms		
79	142.3	32.5	193.7
80	no ROLLUP support	50.2	long running
81	18.0	67.6	272.0
82	60.9	28.2	250.4
83	21.8	25.3	122.7
84	86.2	42.3	206.4
85	78.5	50.1	timeout error
86	no GROUPING SETS support	20.8	110.6
87	no EXCEPT support	114.6	269.5
88	91.3	183.7	226.0
89	87.3	27.5	611.4
90	24.2	17.6	117.3
91	7.0	35.3	116.5
92	24.8	17.7	146.0
93	150.6	20.0	773.9
94	63.6	37.6	long running
95	95.9	97.0	1,094.8
96	73.7	22.2	106.4
97	131.8	71.3	339.6
98	70.4	38.7	530.7
99	65.5	28.2	121.6
Queries Run	68	92	79
Fastest Query Count	12	80	0