Selected Queries Resource Utilization

Todor Ivanov, Matteo Pergolesi

September 11, 2018

Contents

1 Q08

1.1 Type

MapReduce/Python

1.2 Description

For online sales, compare the total sales monetary amount in which customers checked online reviews 3 days before making the purchase and that of sales in which customers did not read reviews. Consider only online sales for a specific category in a given year.

1.3 Performance

	Default (sec.)		No compr. (sec.)		Snappy (sec.)	
Query	**ORC**	**Parquet**	**ORC**	**Parquet**	**ORC**	**Parquet**
Q08	913	675	911	712	900	639

1.4 Phase 1 - Data preparation

- Temp table 1: dates
- Temp table 2: users_which_read_reviews (get webclicks and use python sessionize)
- Temp table 3: sales on desired date range

1.5 Phase 2 - Query

- Sum on sales_reviews and (sales_all - sales_review).
- Result is a single line for money amount comparison

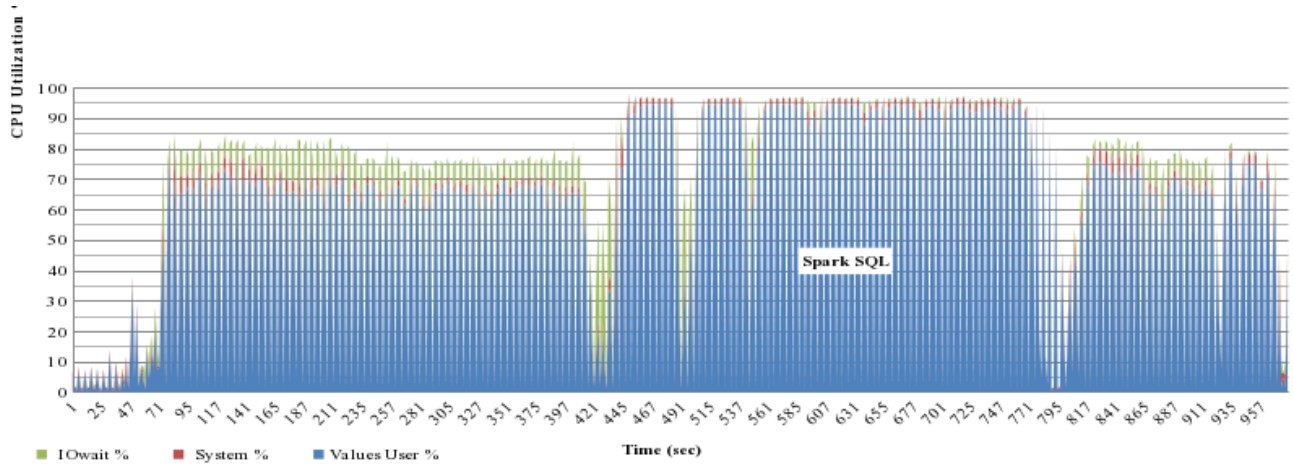
1.6 Source

https://git.io/vpFen

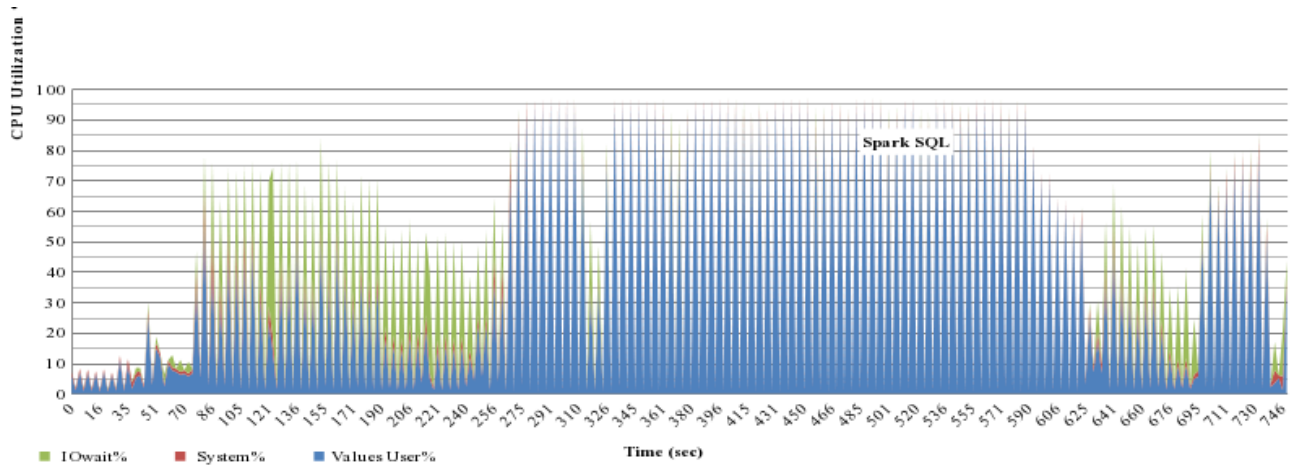
1.7 Notes

Phase 1 should be the interesting one (it's where you use file formats).

1.8 No compression - ORC vs Parquet

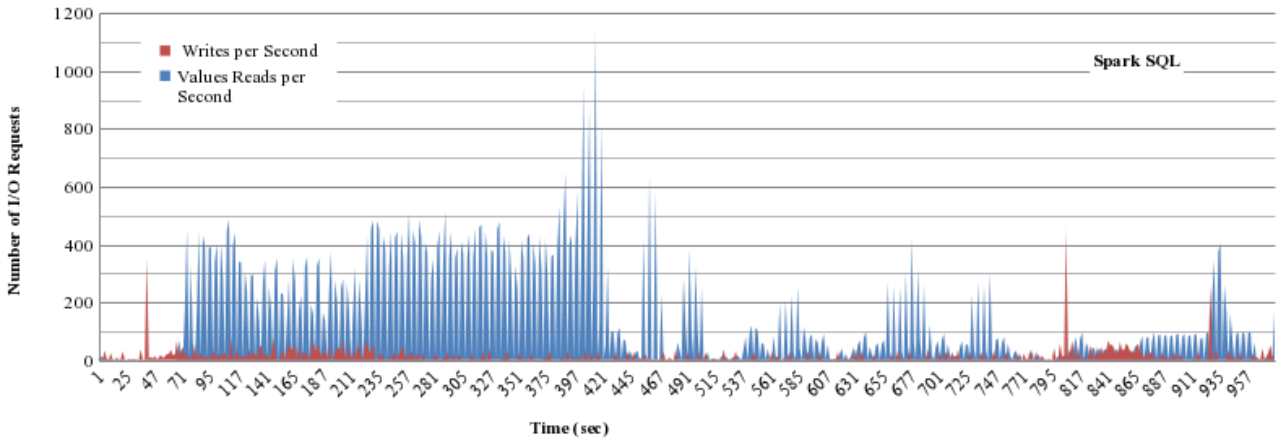


(a) ORC

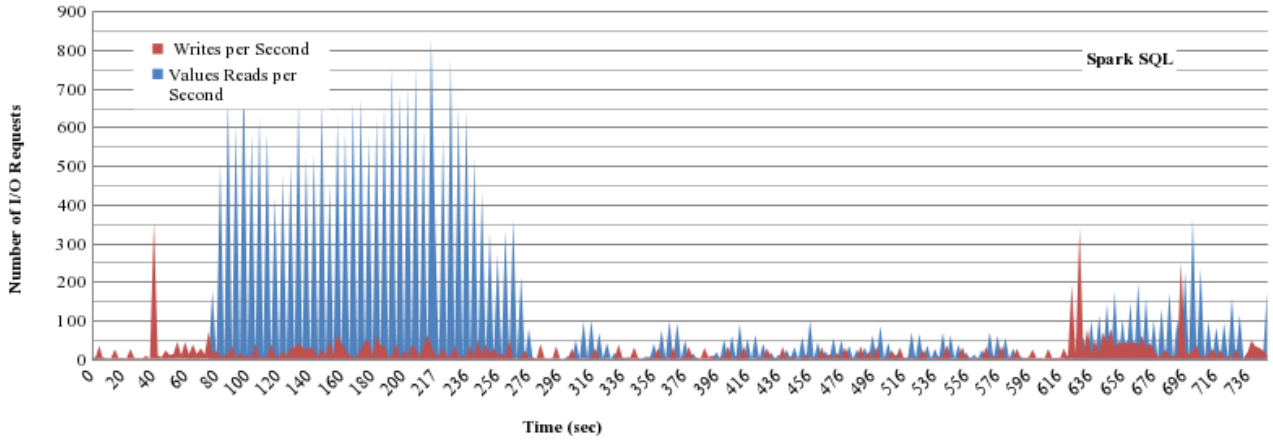


(b) Parquet

Figure 1: CPU: Lower utilization on Parquet (max is 80%)

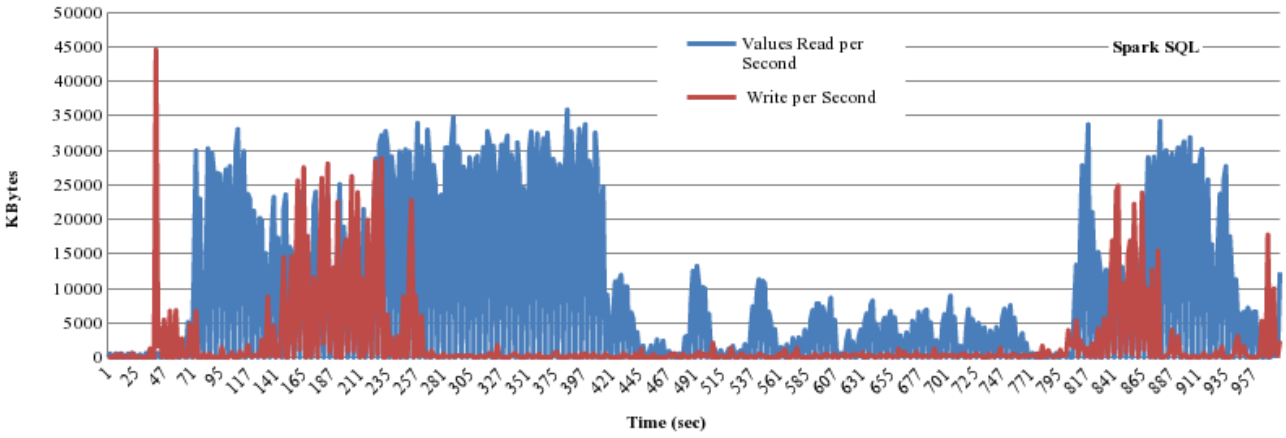


(a) ORC

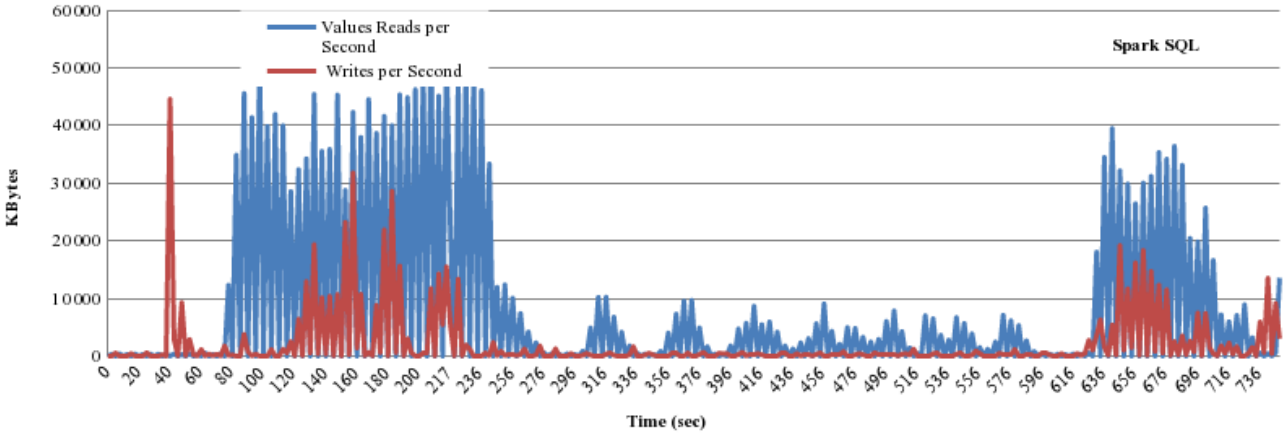


(b) Parquet

Figure 2: Disk Requests: More requests on Parquet.

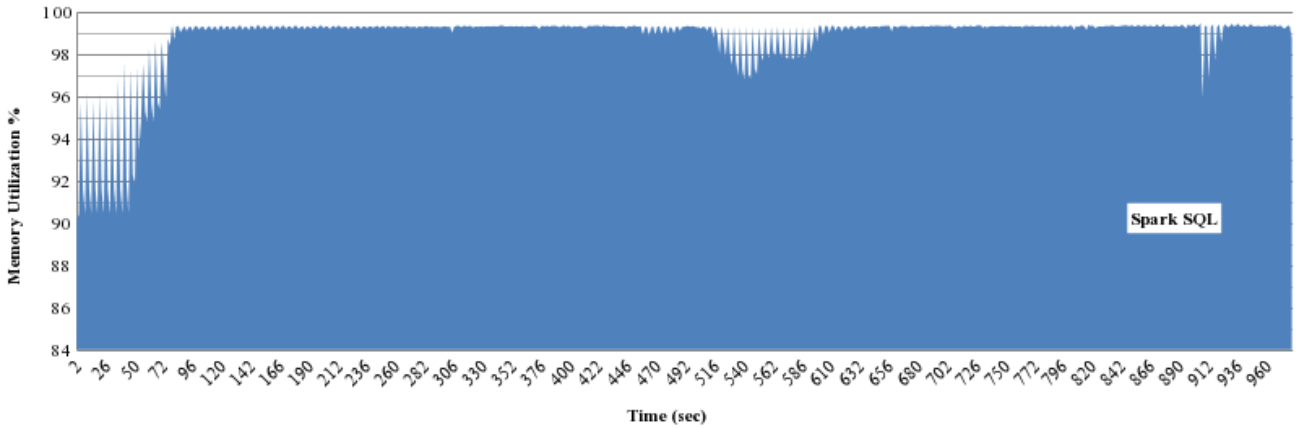


(a) ORC

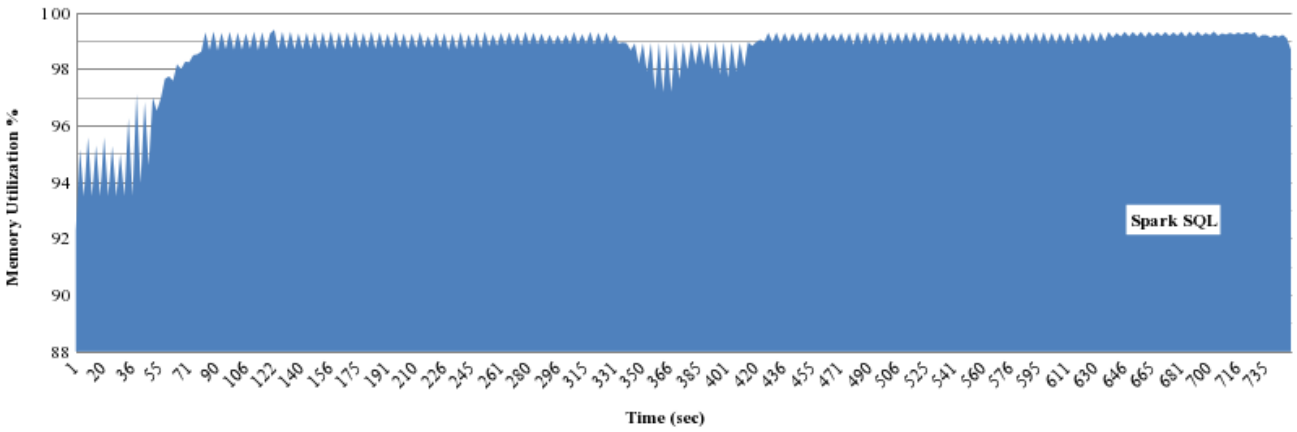


(b) Parquet

Figure 3: Disk Bandwidth: bandwidth utilization on Parquet.

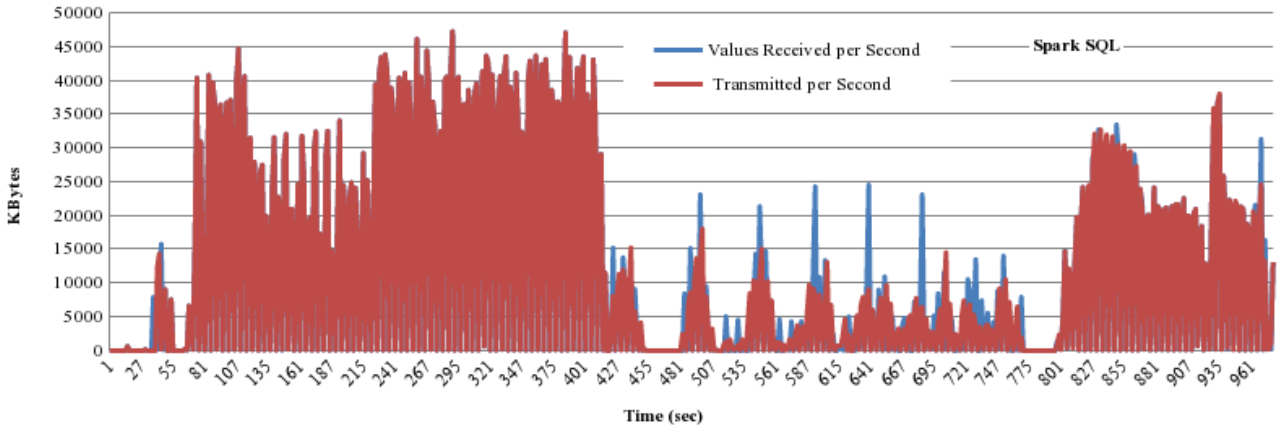


(a) ORC

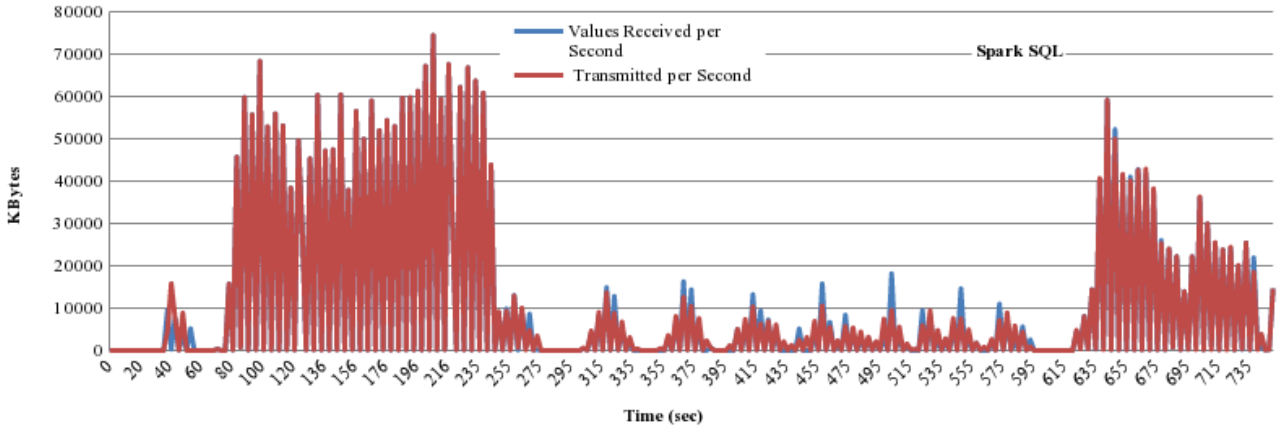


(b) Parquet

Figure 4: Free Memory.

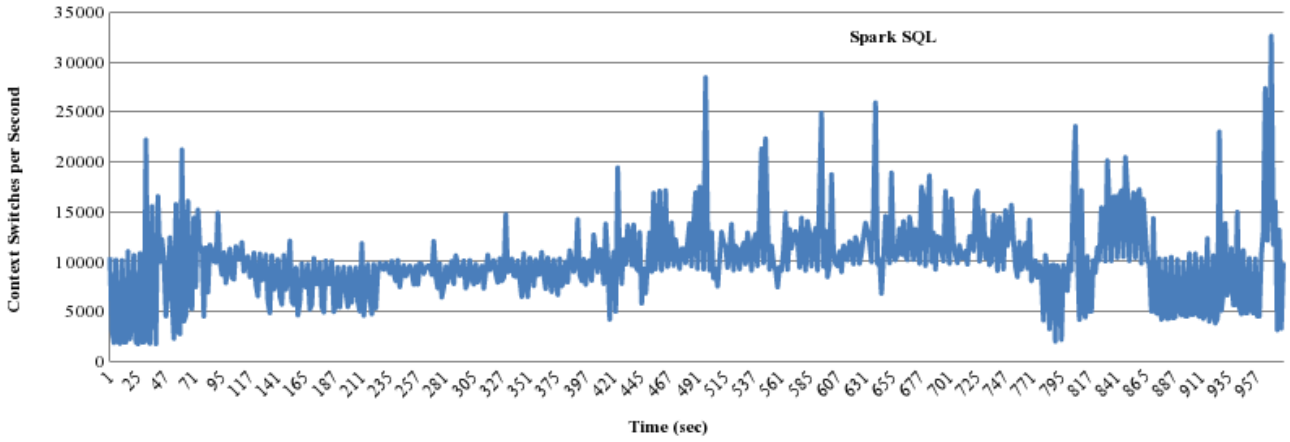


(a) ORC

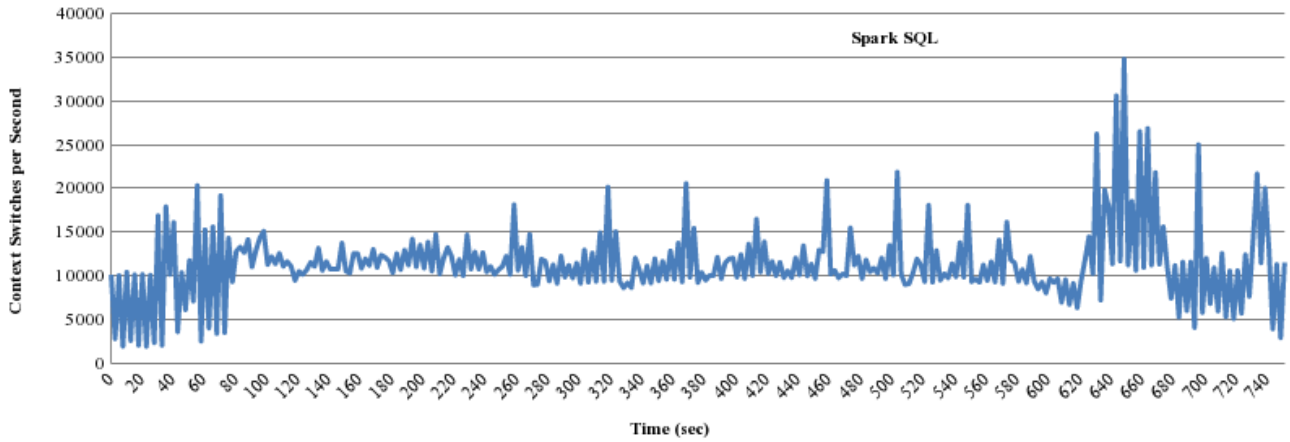


(b) Parquet

Figure 5: Network Utilization: More or less corresponding to DiskIO.



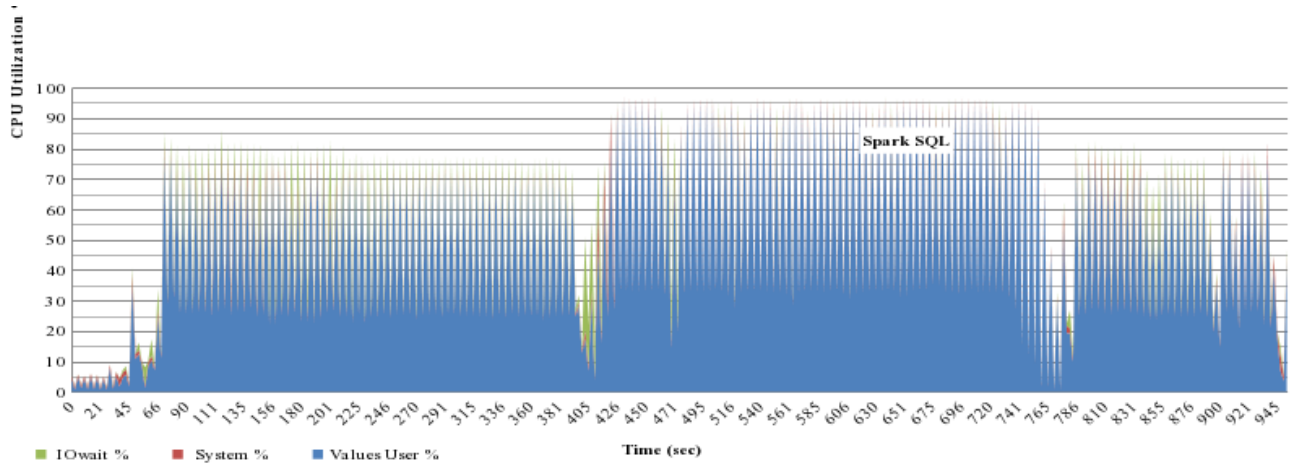
(a) ORC



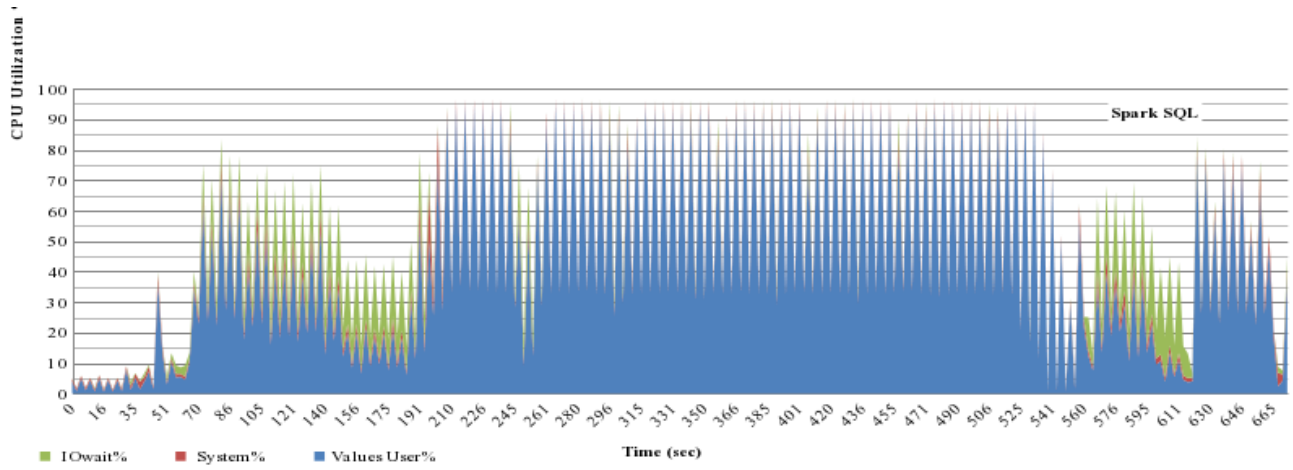
(b) Parquet

Figure 6: VMSTAT Context Switches

1.9 Snappy - ORC vs. Parquet

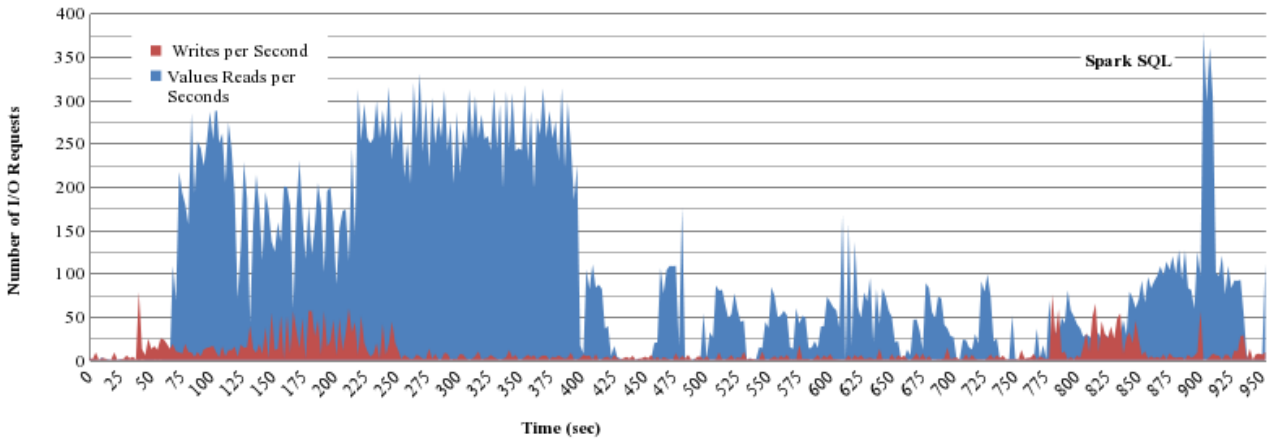


(a) ORC

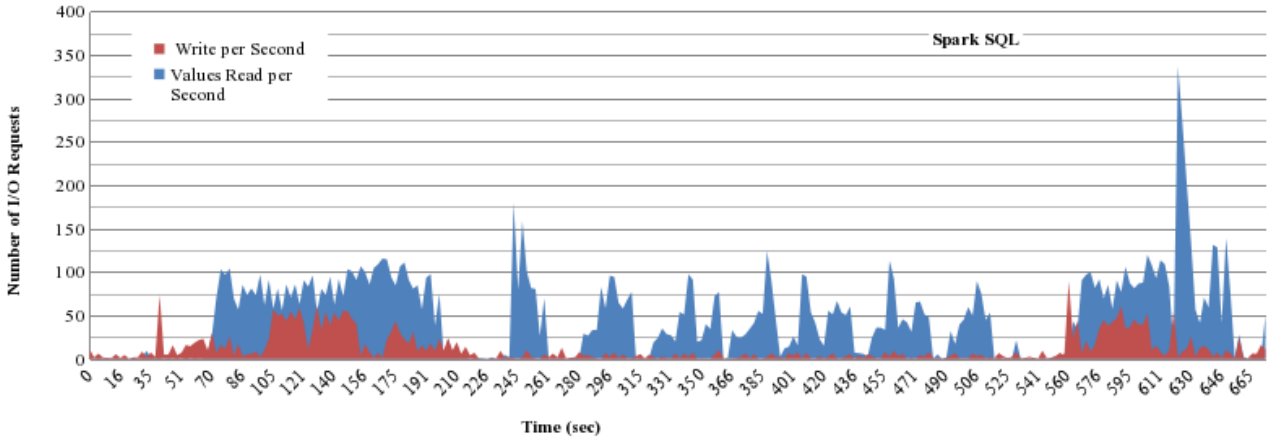


(b) Parquet

Figure 7: CPU: Low util on parquet (60%), short time for phase 1.

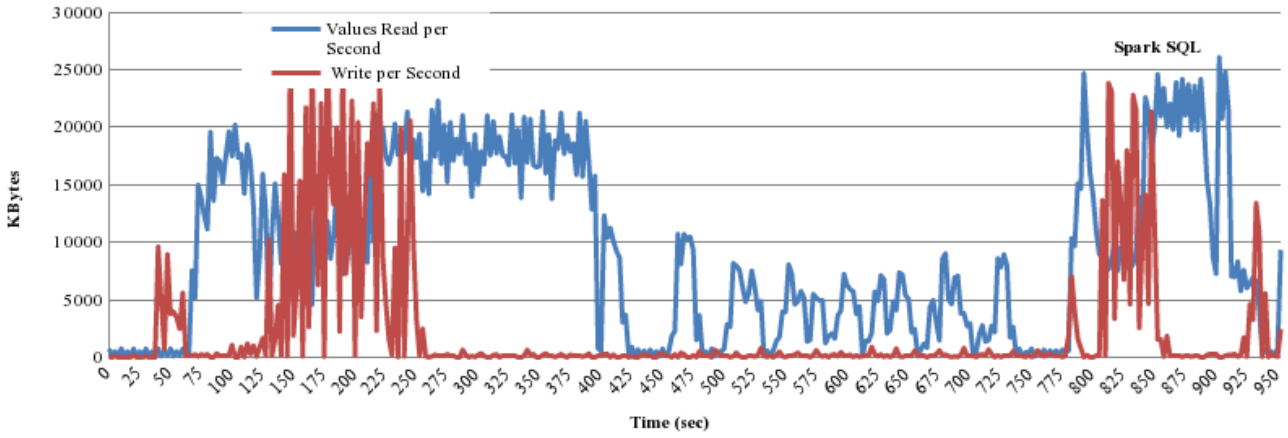


(a) ORC

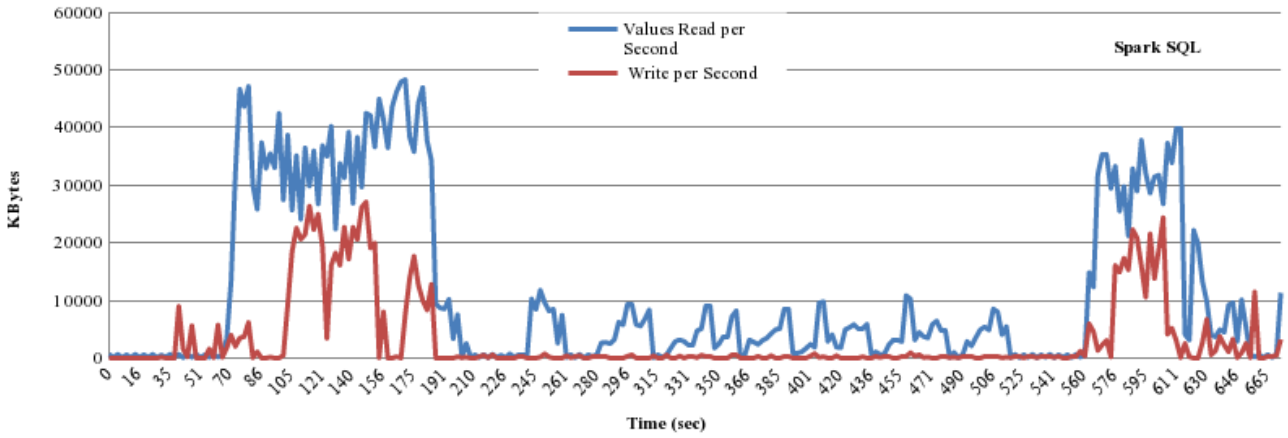


(b) Parquet

Figure 8: Disk Requests: Lower number of requests on Parquet.

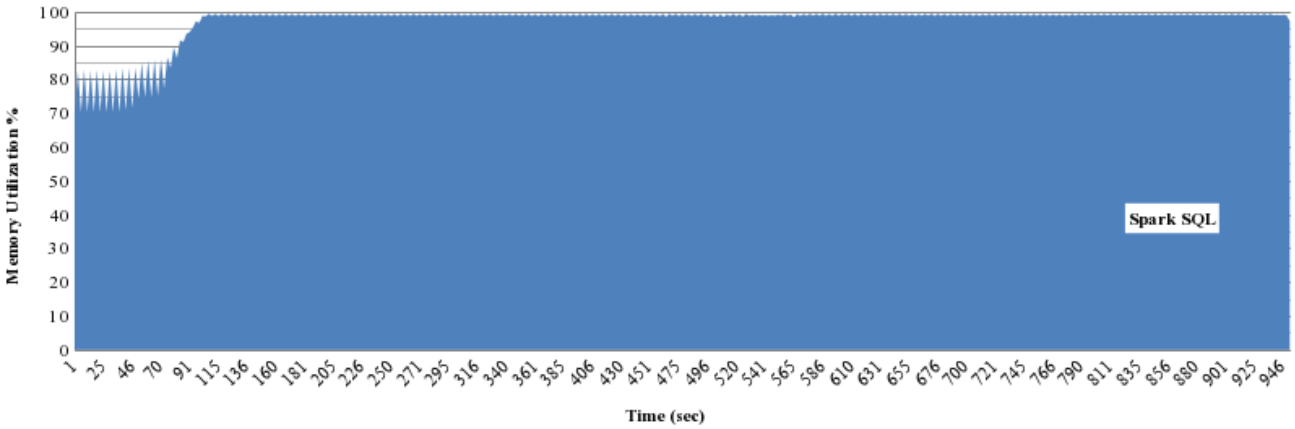


(a) ORC

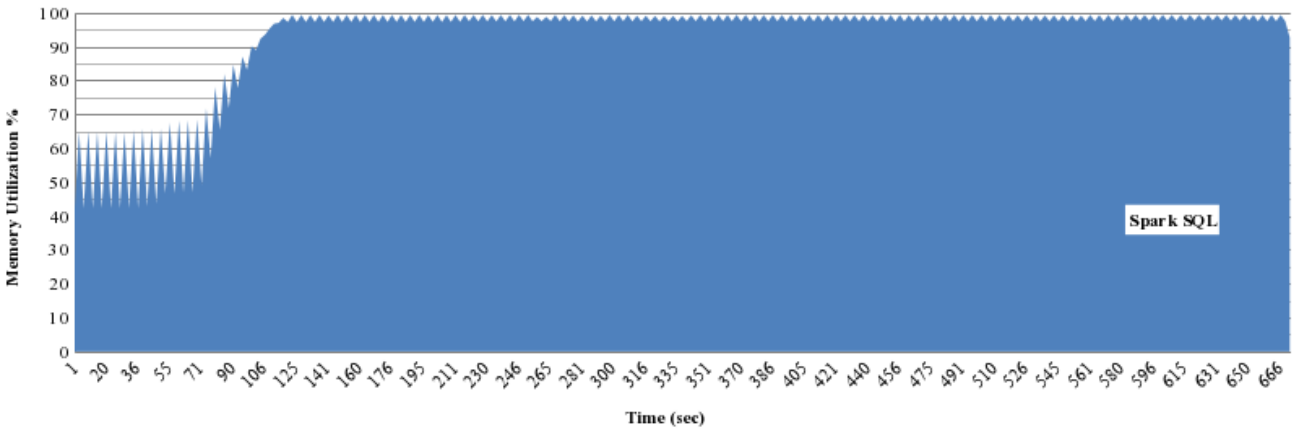


(b) Parquet

Figure 9: Disk Bandwidth: higher BW utilization on Parquet.

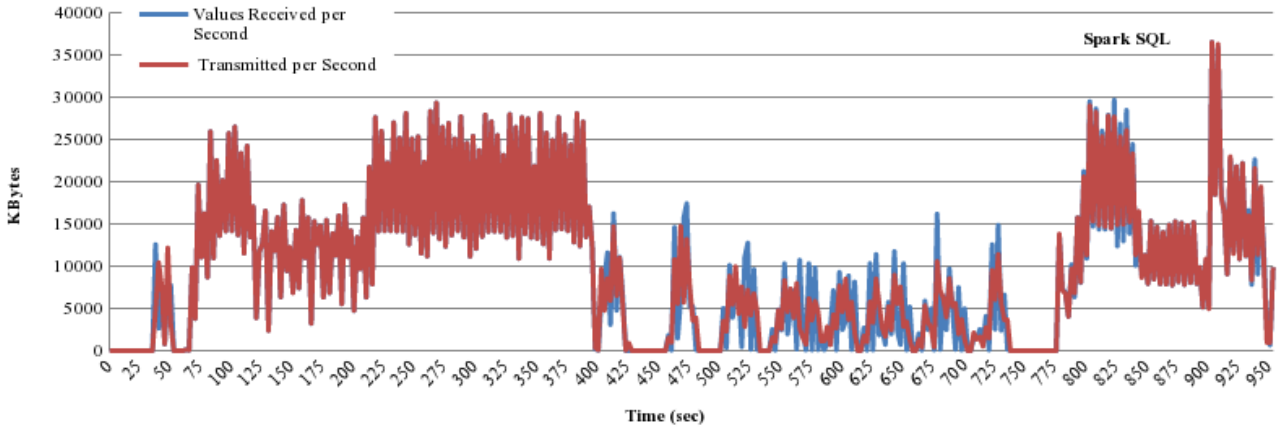


(a) ORC

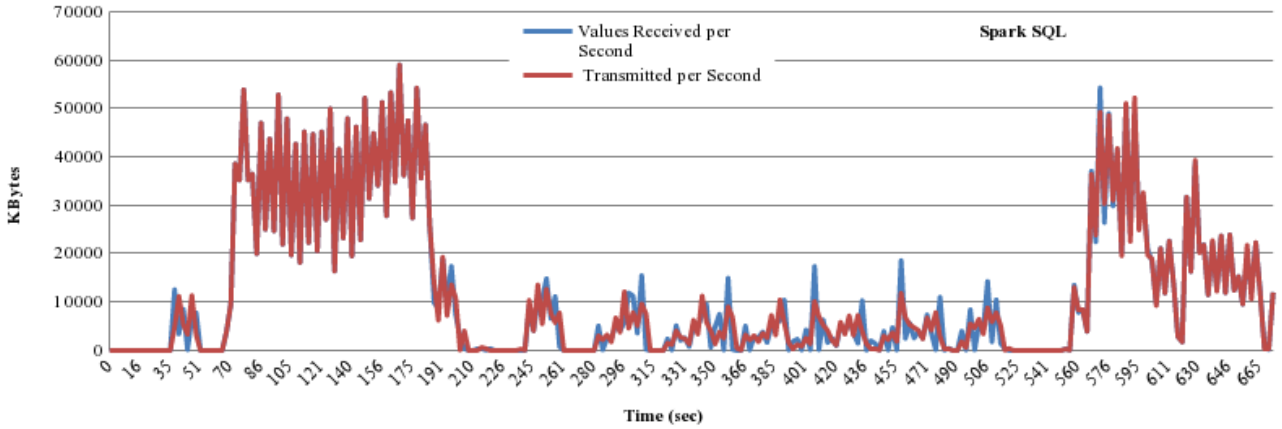


(b) Parquet

Figure 10: Free Memory.

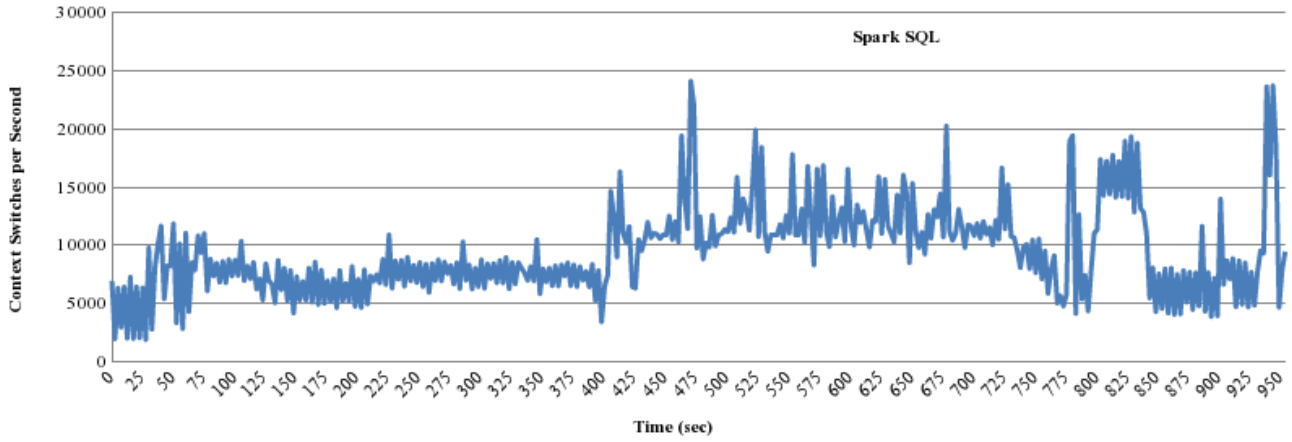


(a) ORC

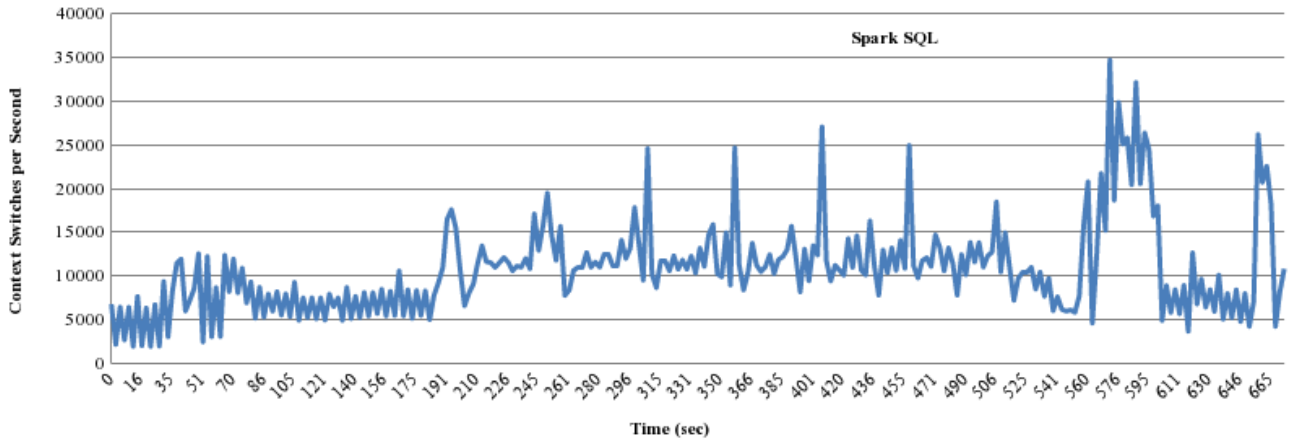


(b) Parquet

Figure 11: Network Utilization: more or less corresponding to DiskIO BW.



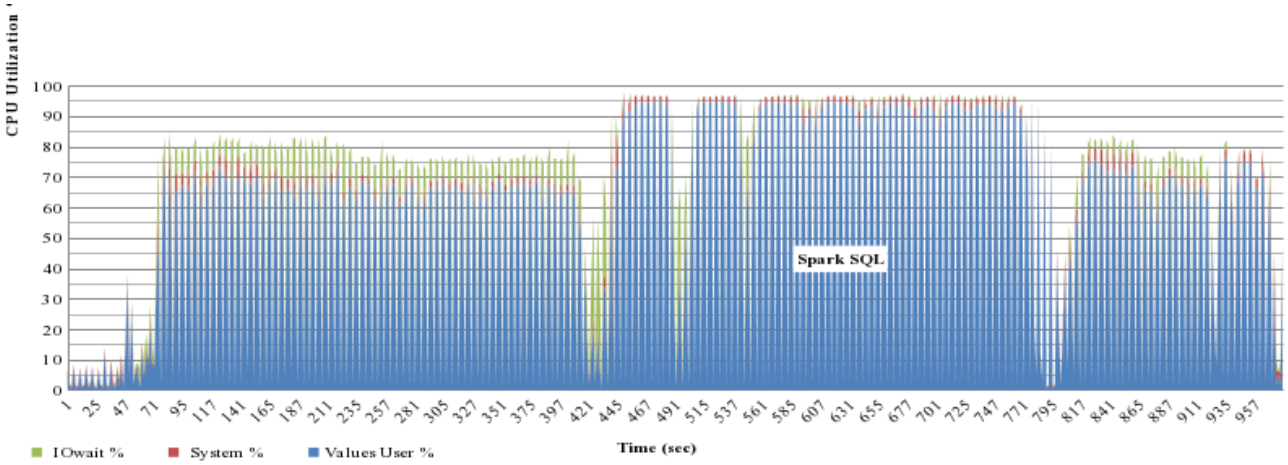
(a) ORC



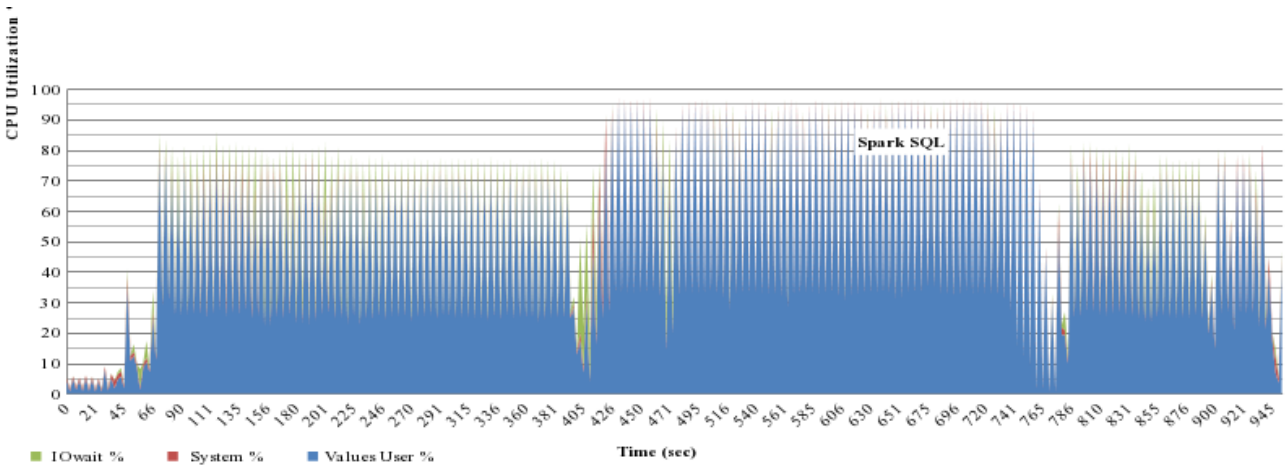
(b) Parquet

Figure 12: VMSTAT Context Switches

1.10 ORC - NoCompression vs. Snappy

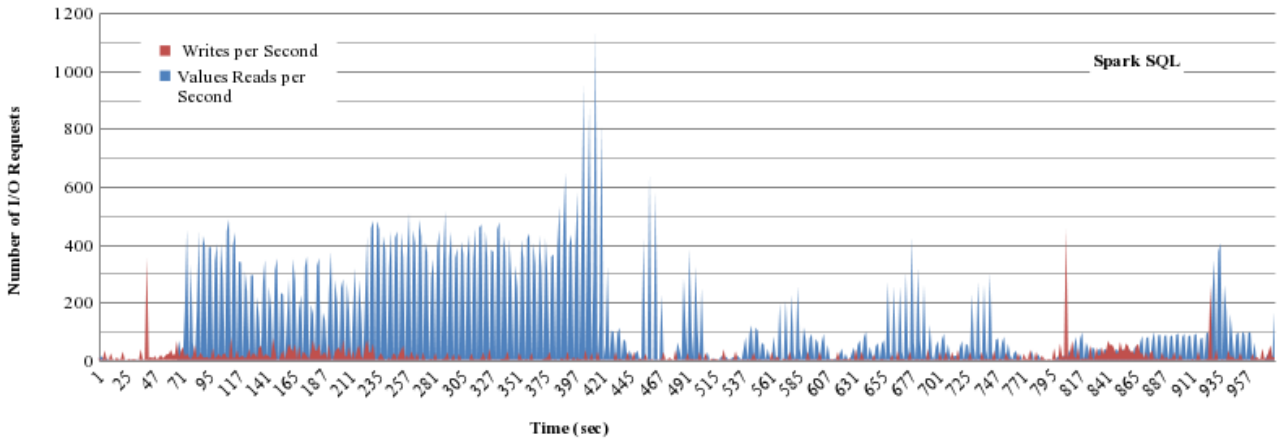


(a) NoCompression

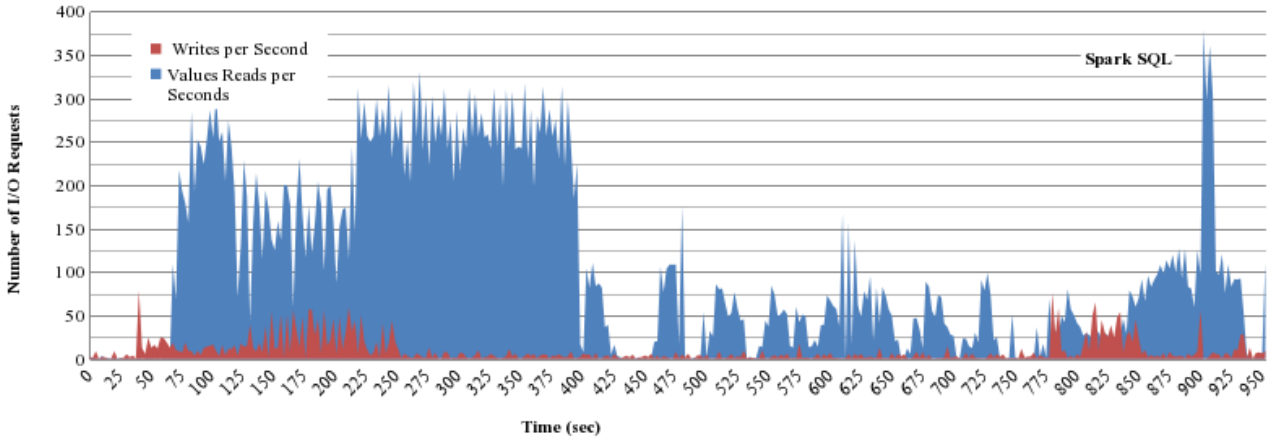


(b) Snappy

Figure 13: CPU: Same behavior.

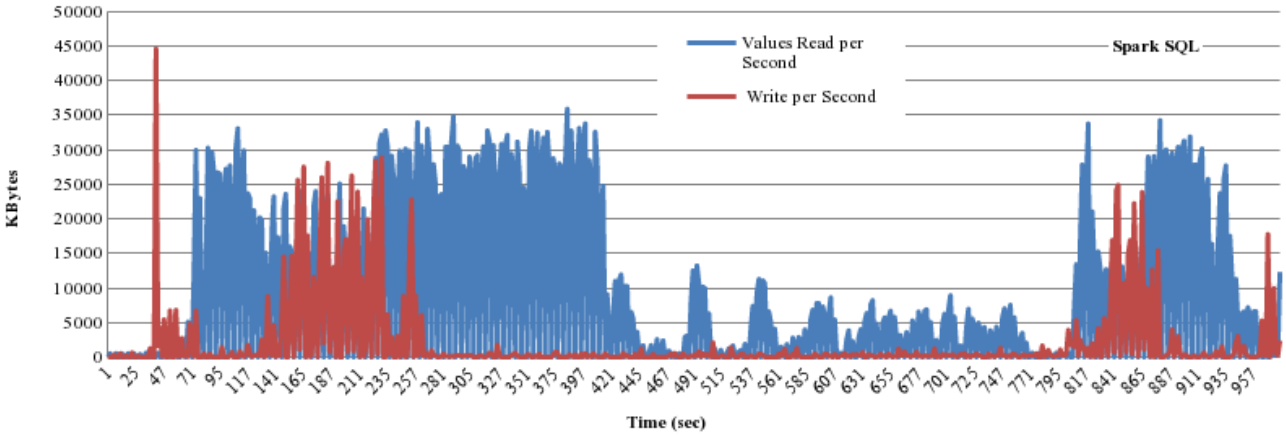


(a) NoCompress

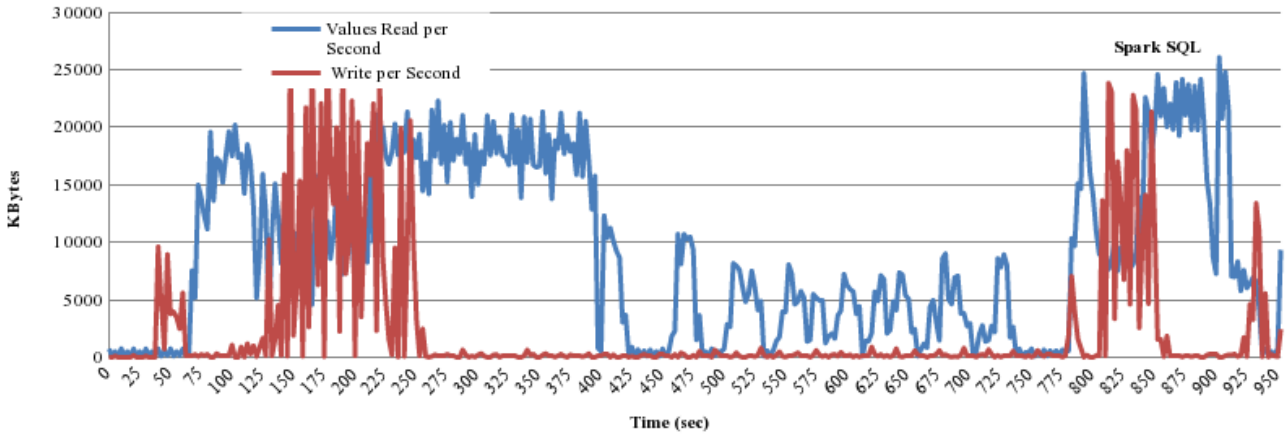


(b) Snappy

Figure 14: Disk Requests: Less requests with Snappy (good).

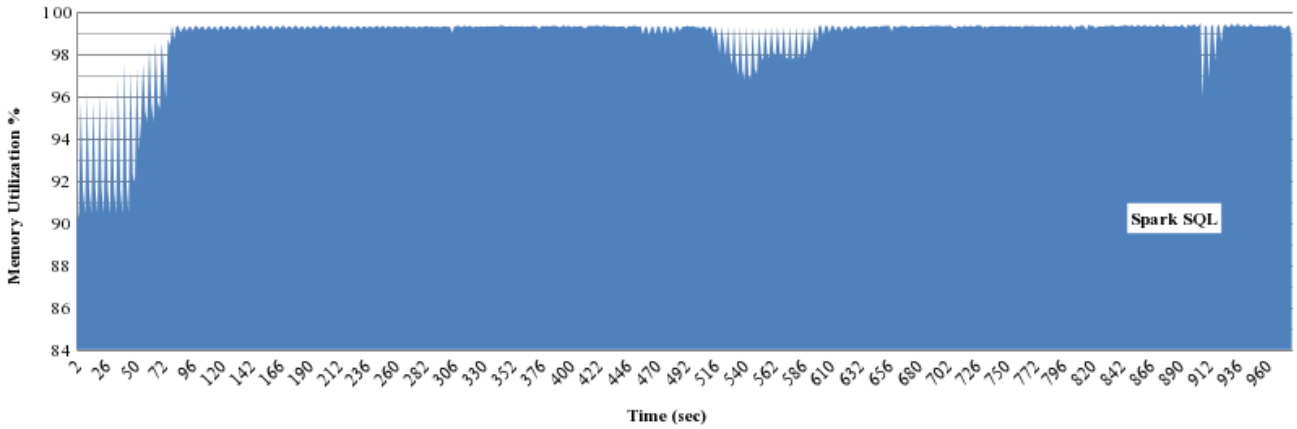


(a) NoCompress

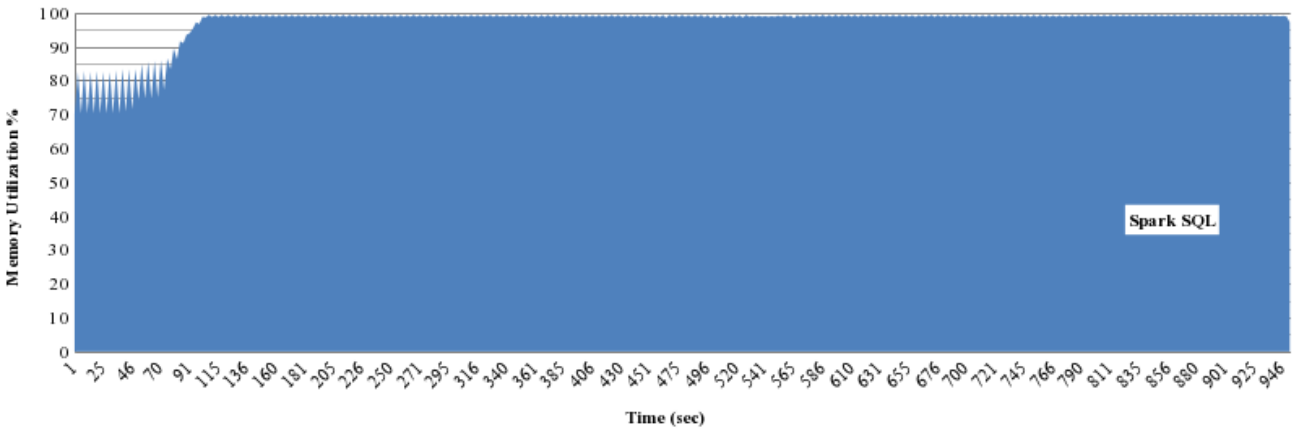


(b) Snappy

Figure 15: Disk Bandwidth: less bw utilization on snappy; goes up and falls to zero on no compression.

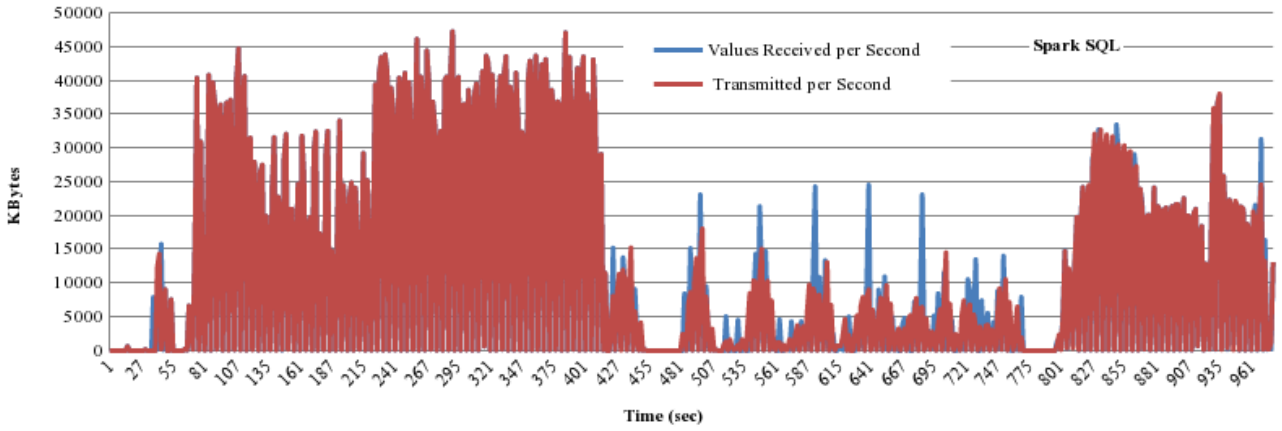


(a) No Compress

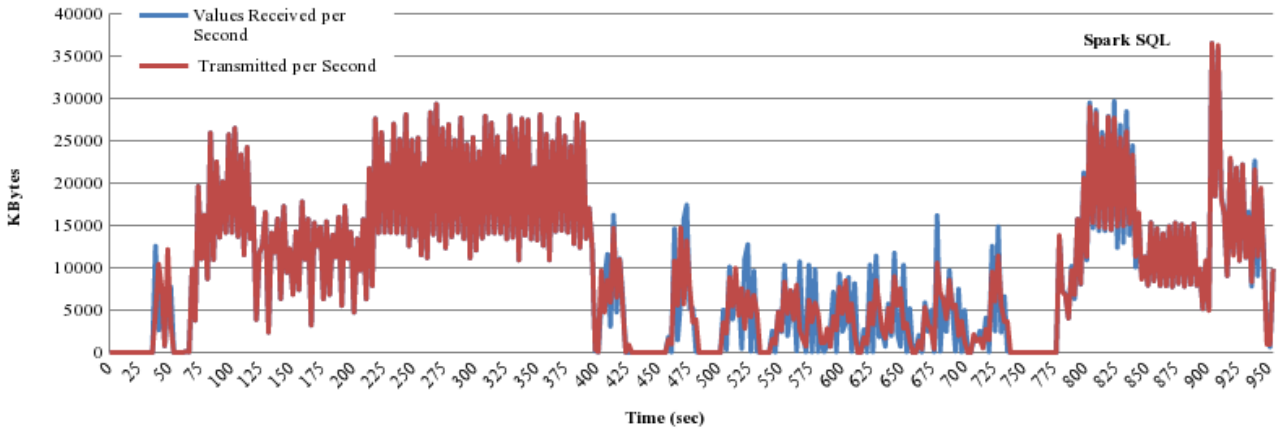


(b) Snappy

Figure 16: Free Memory.

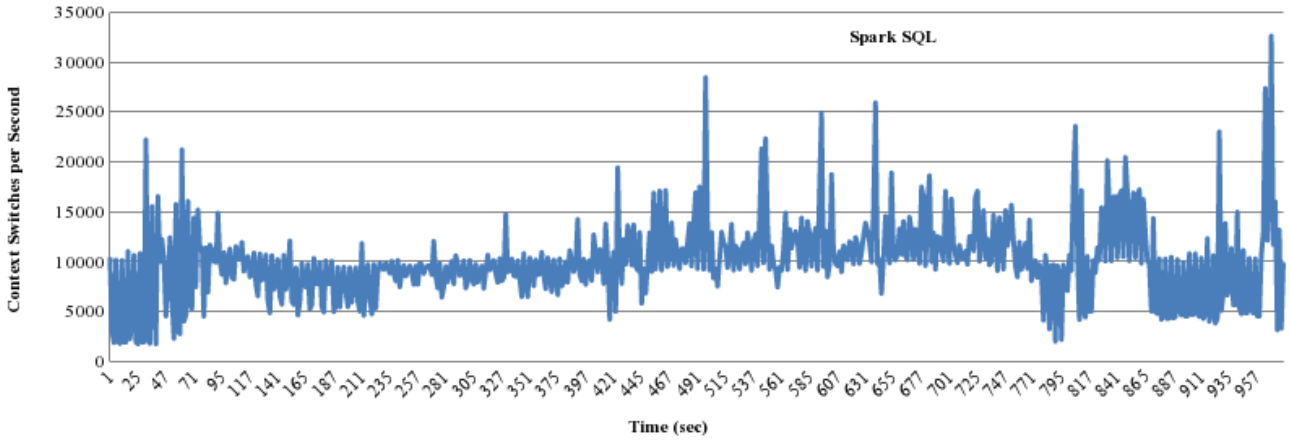


(a) NoCompress

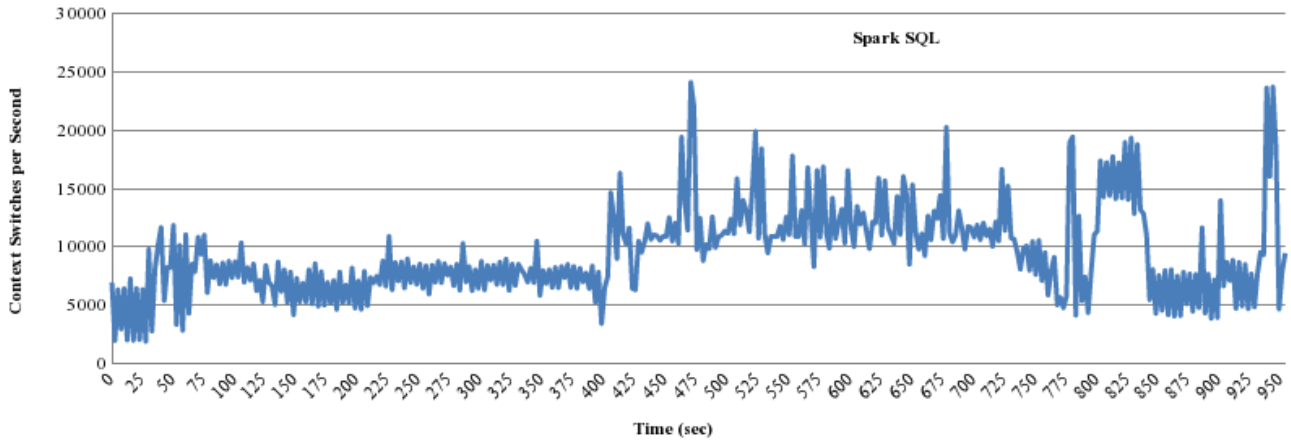


(b) Snappy

Figure 17: Network Utilization: Follows DiskIO BW.



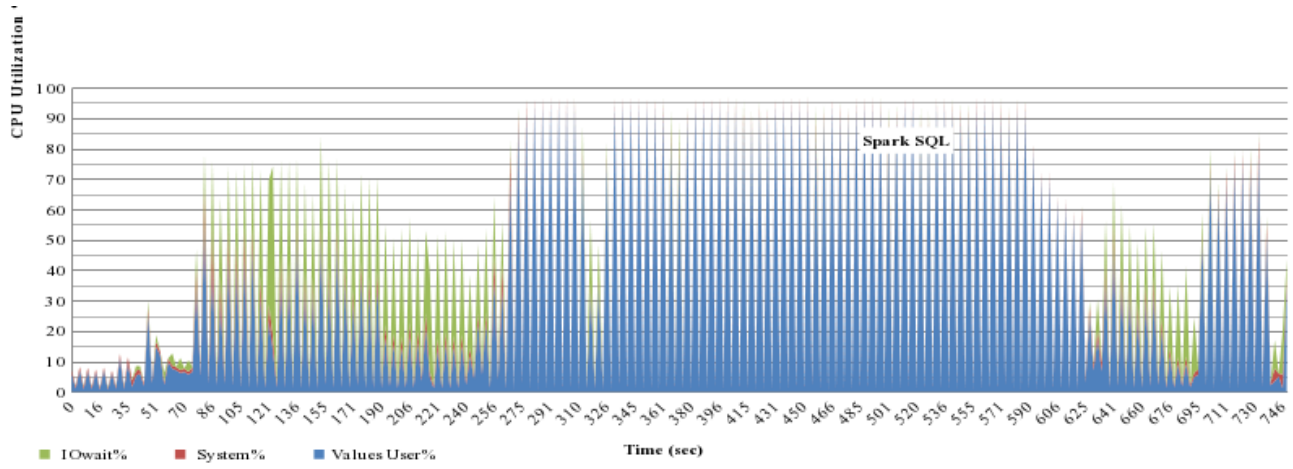
(a) NoCompress



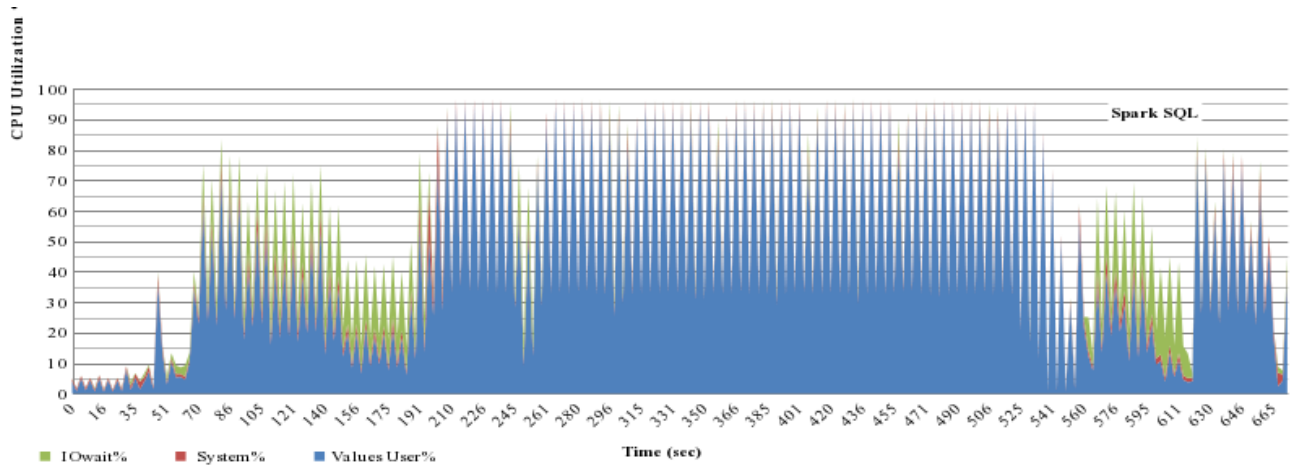
(b) Snappy

Figure 18: VMSTAT Context Switches

1.11 Parquet - NoCompression vs Snappy

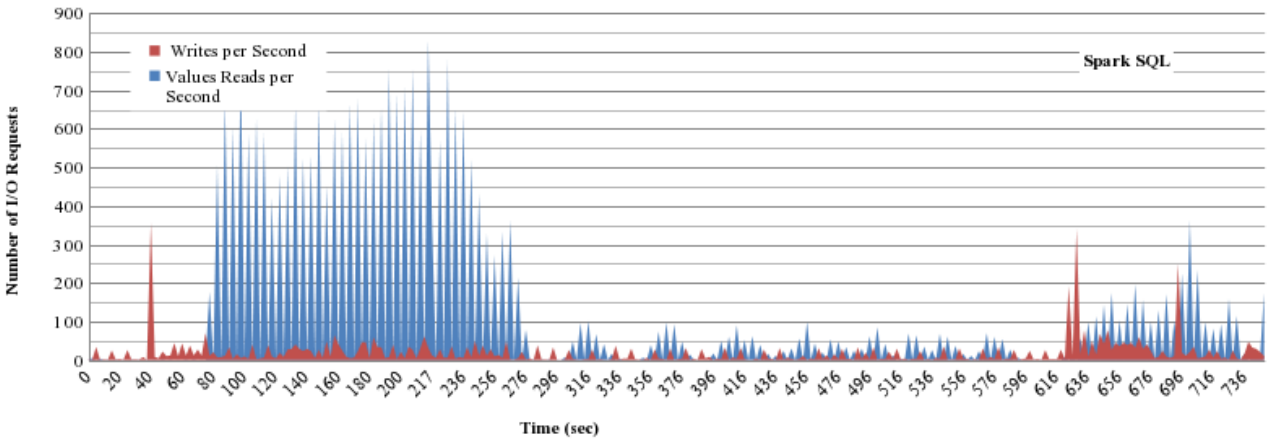


(a) NoCompression

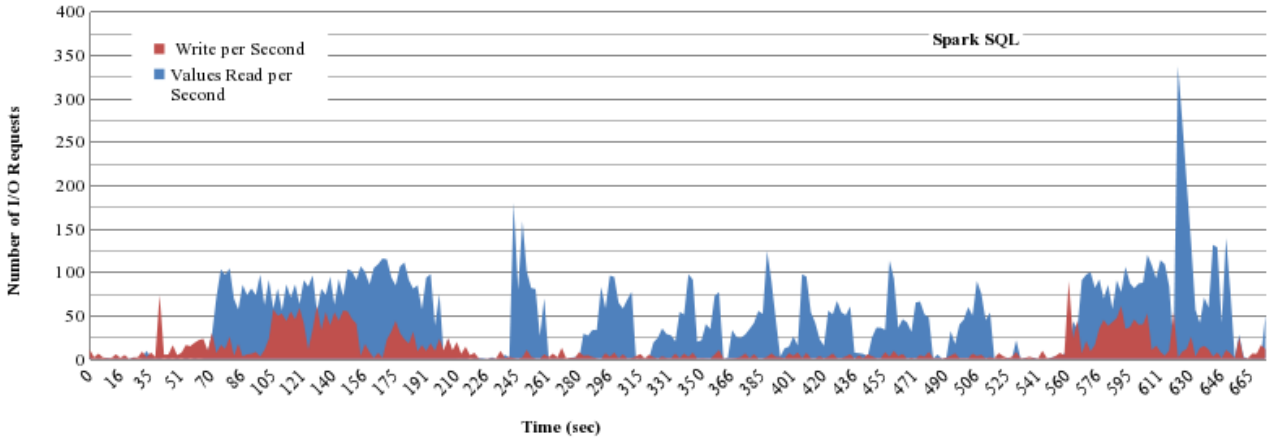


(b) Snappy

Figure 19: CPU: Same behavior (phase 1 on snappy takes less time).

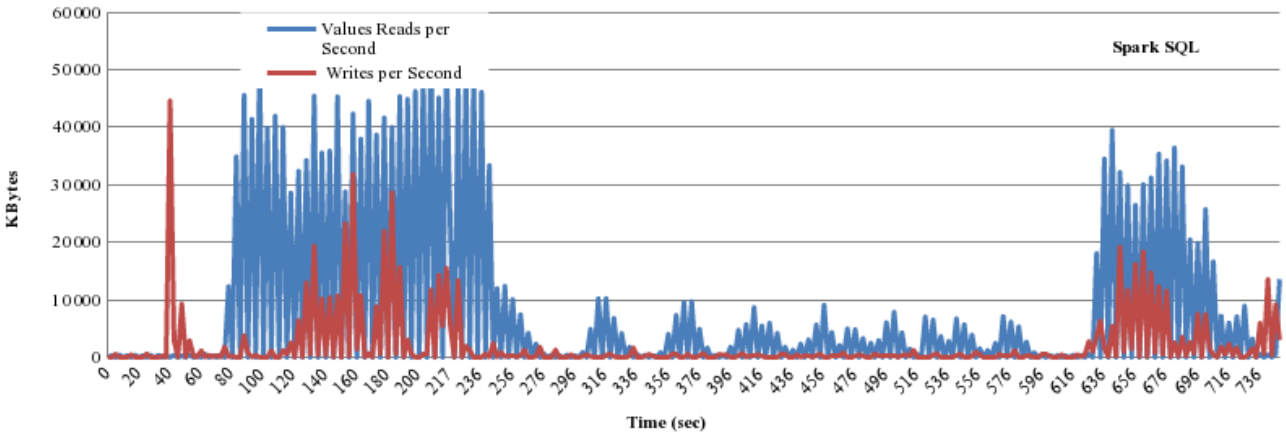


(a) NoCompression

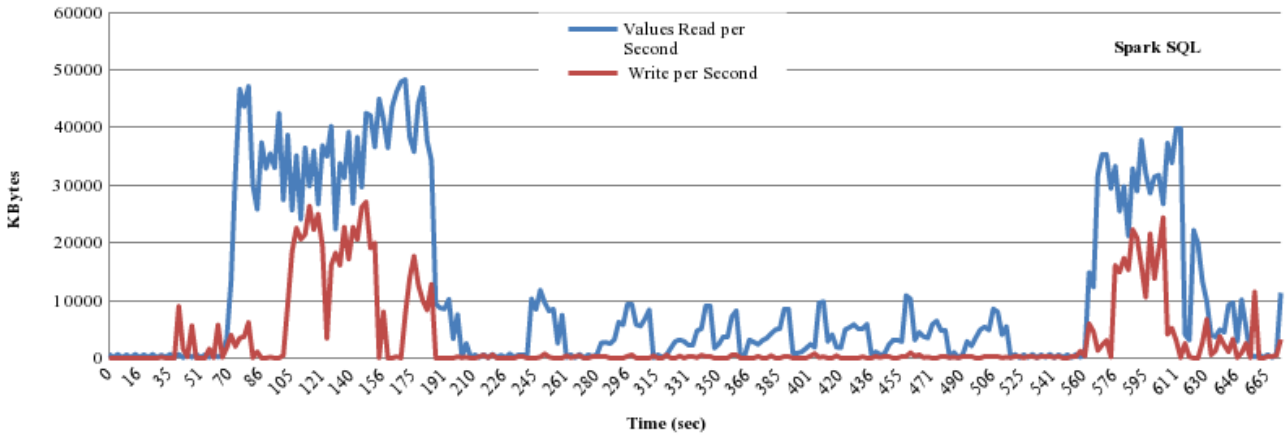


(b) Snappy

Figure 20: Disk Requests: So much less requests with Snappy!.

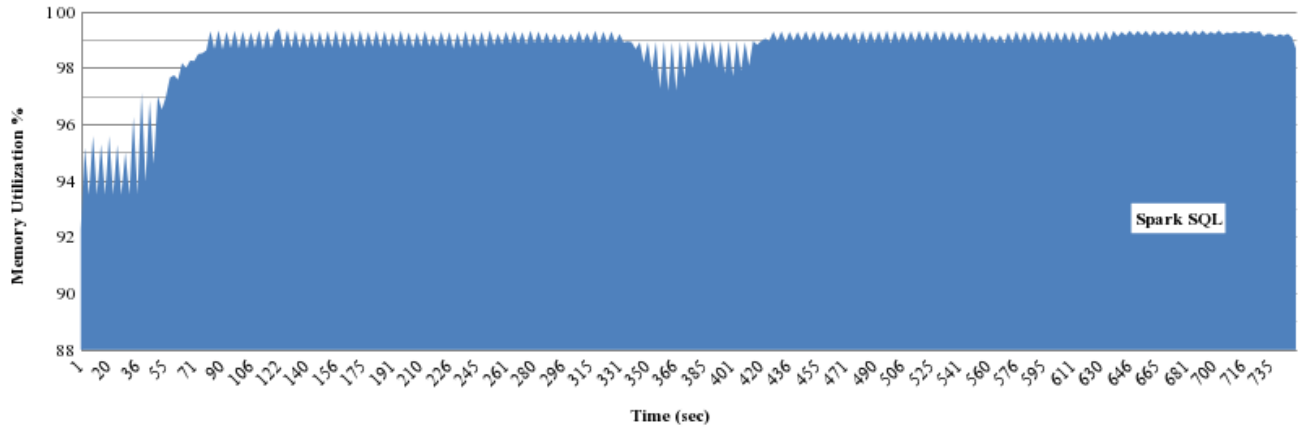


(a) NoCompression

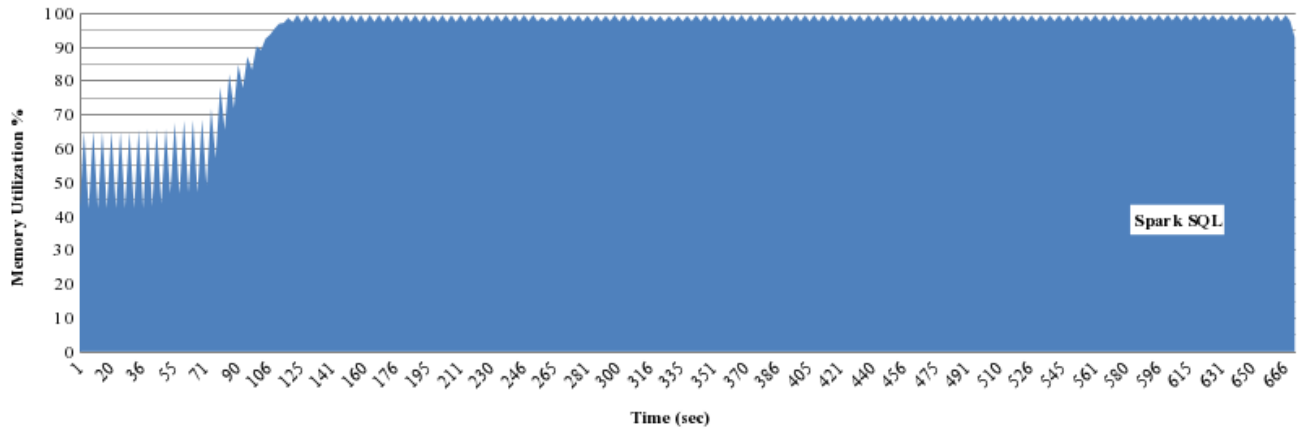


(b) Snappy

Figure 21: Disk Bandwidth: Less BW util with Snappy.

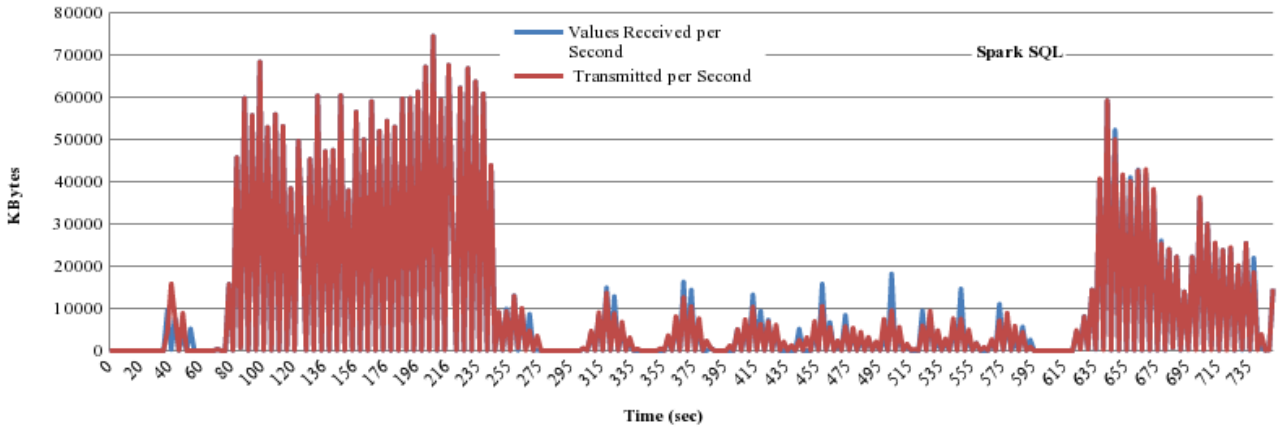


(a) No Compression

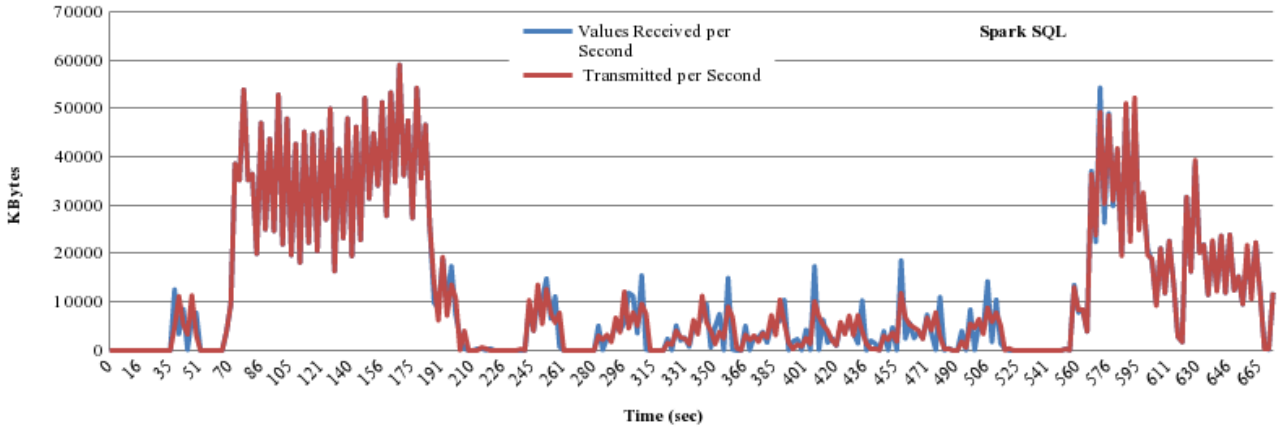


(b) Snappy

Figure 22: Free Memory.

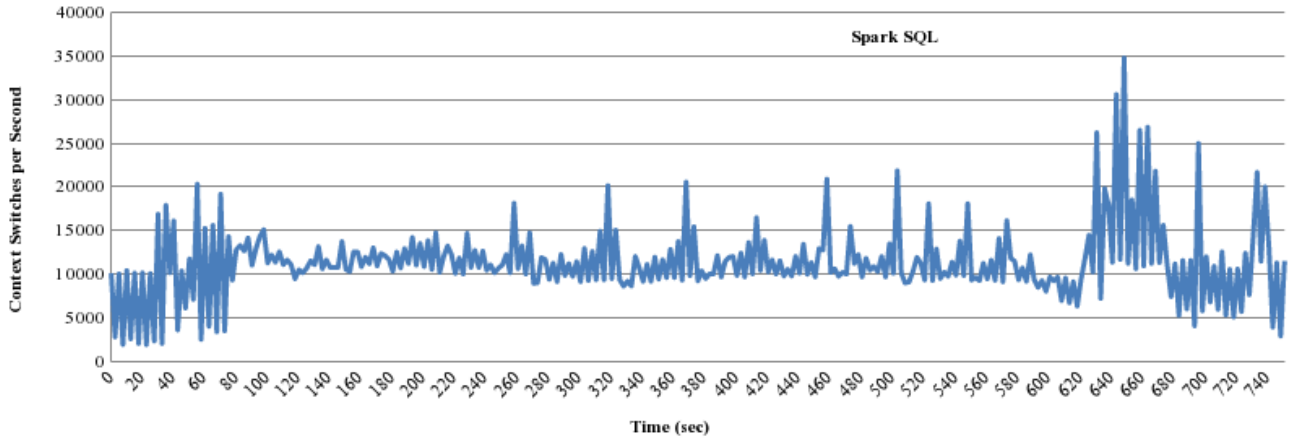


(a) NoCompression

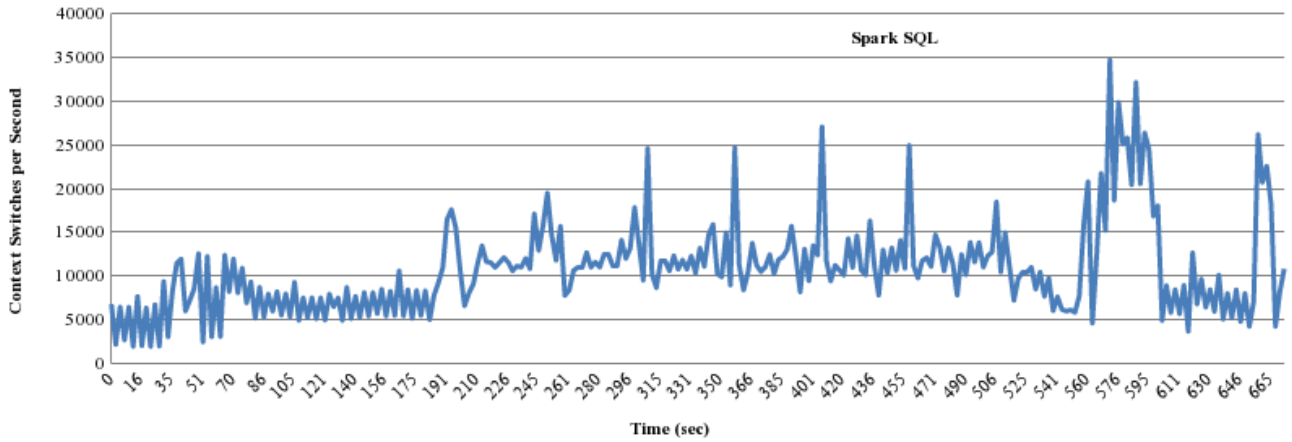


(b) Snappy

Figure 23: Network Utilization: Follows diskIO BW.



(a) NoCompression



(b) Snappy

Figure 24: VMSTAT Context Switches

2 Q10

2.1 Type

HiveQL/OpenNLP

2.2 Description

For all products, extract sentences from its product reviews that contain positive or negative sentiment and display for each item the sentiment polarity of the extracted sentences (POS OR NEG) and the sentence and word in sentence leading to this classification.

2.3 Performance

	Default (sec.)		No compr. (sec.)		Snappy (sec.)	
Query	**ORC**	**Parquet**	**ORC**	**Parquet**	**ORC**	**Parquet**
Q10	1994	976	2026	1633	2070	3075

2.4 Phase 1

Selects all product reviews from "product_reviews" table and uses a UDF to extract sentiment. No temporary tables, works directly and only on file format data. Result of course is not compressed.

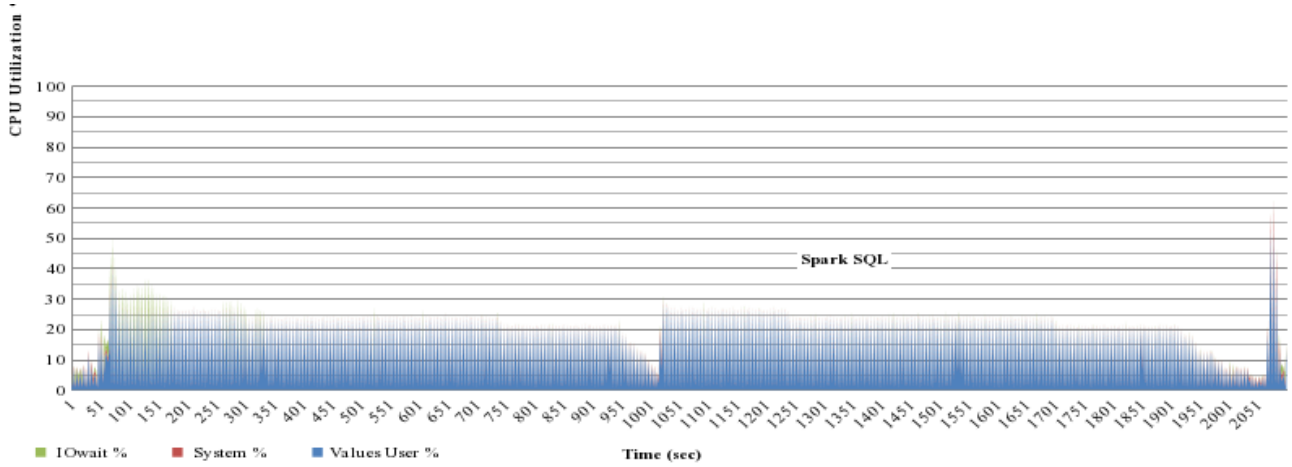
2.5 Source

GitHub: https://git.io/vpFeK

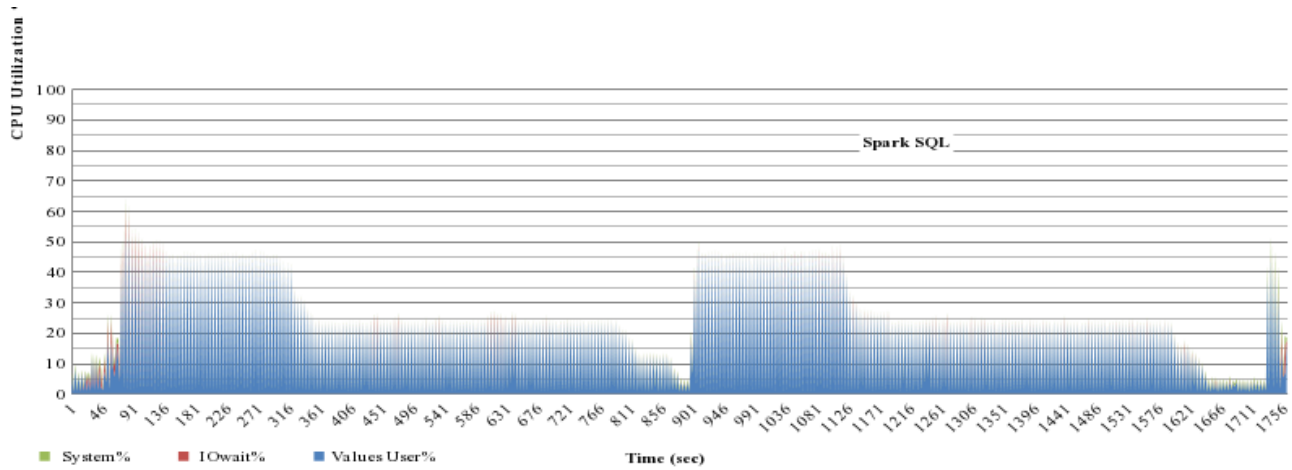
2.6 Notes

This type of queries perform better on uncompressed data. It is interesting to observe that with Snappy, ORC performs better for Q10. Check CPU and disk.

2.7 No compression - ORC vs Parquet

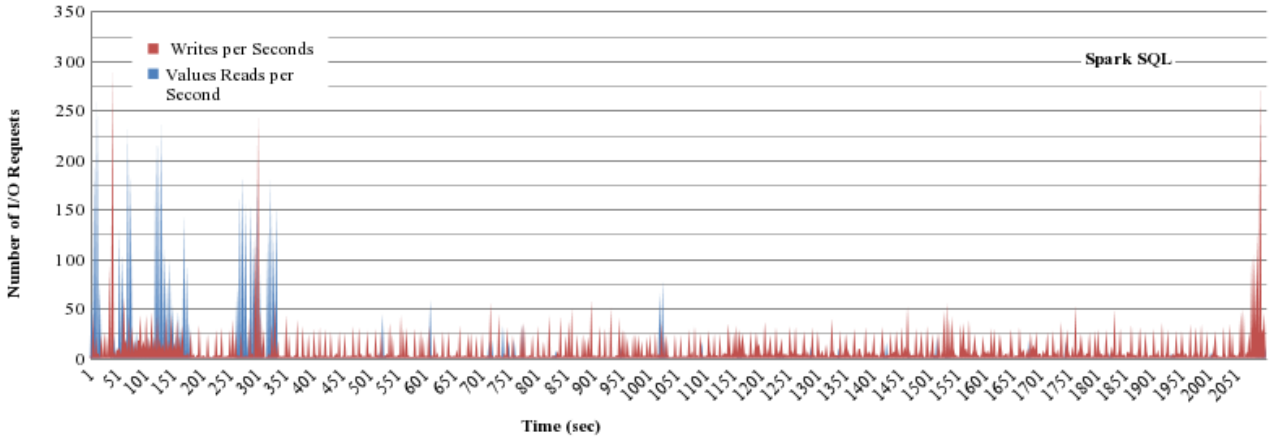


(a) ORC

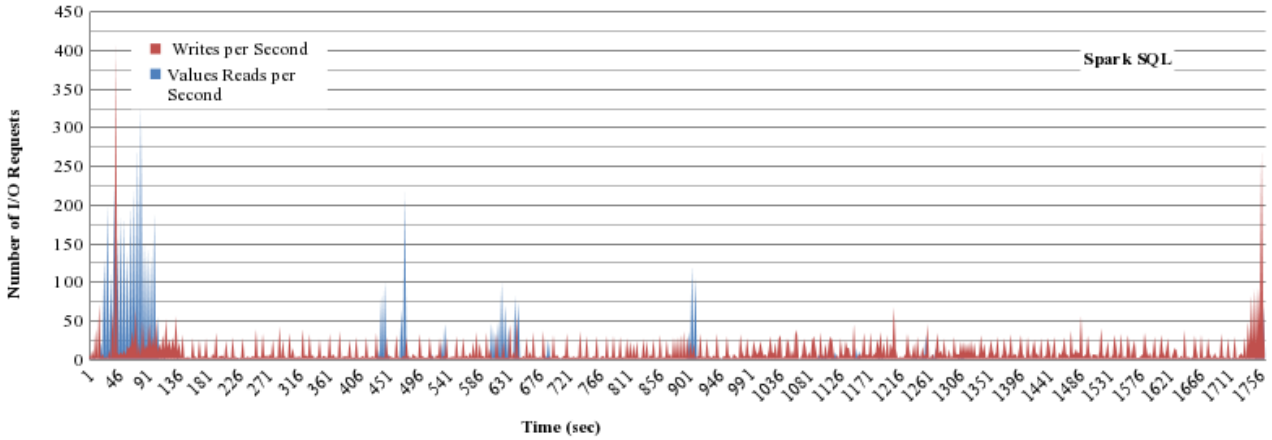


(b) Parquet

Figure 25: CPU: Lower utilization on Parquet (max is 80%)

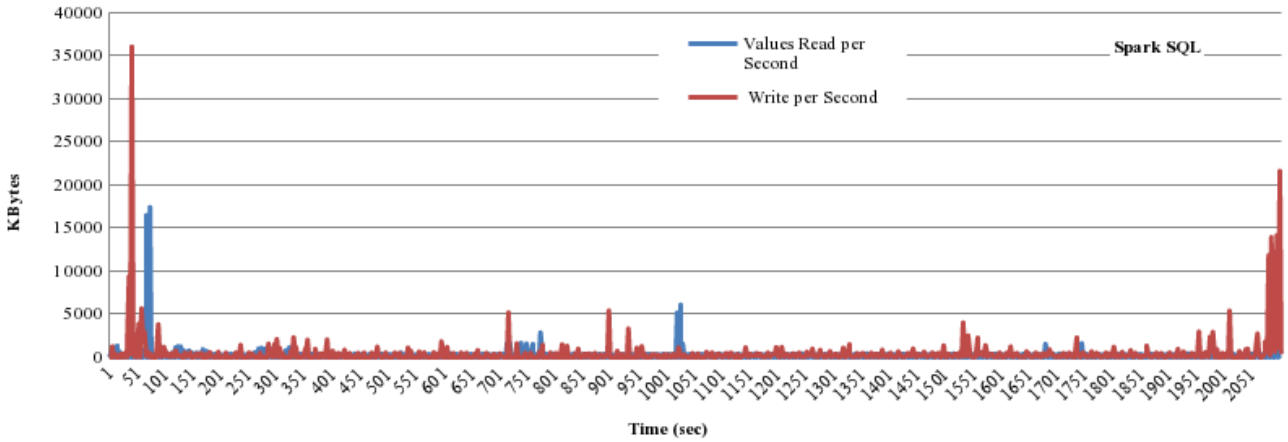


(a) ORC

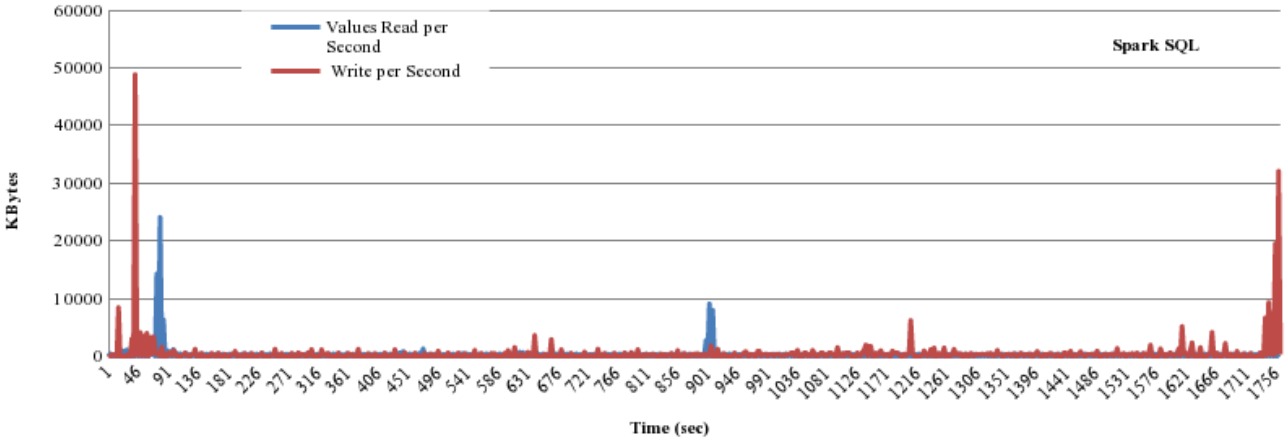


(b) Parquet

Figure 26: Disk Requests: More requests on Parquet.

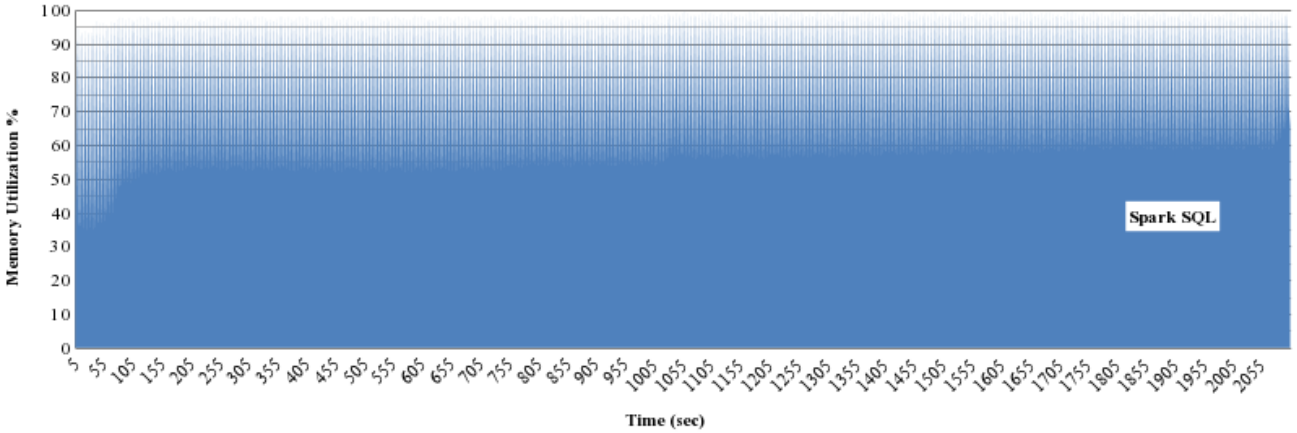


(a) ORC

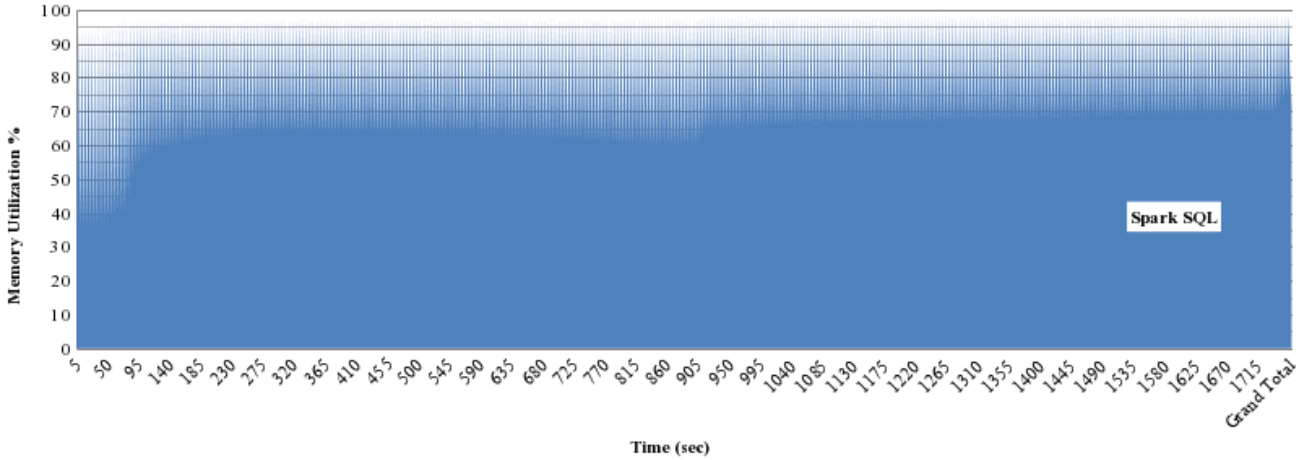


(b) Parquet

Figure 27: Disk Bandwidth: bandwidth utilization on Parquet.

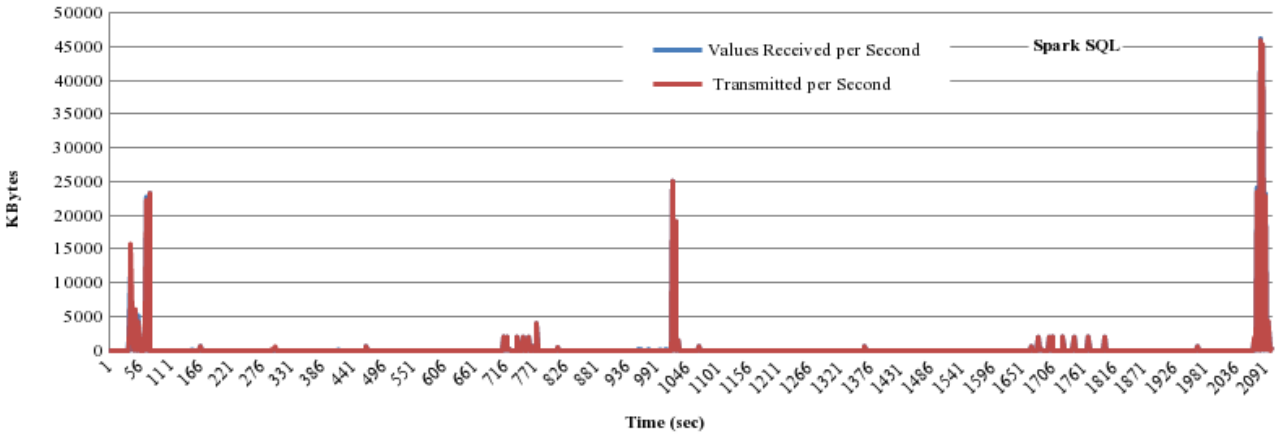


(a) ORC

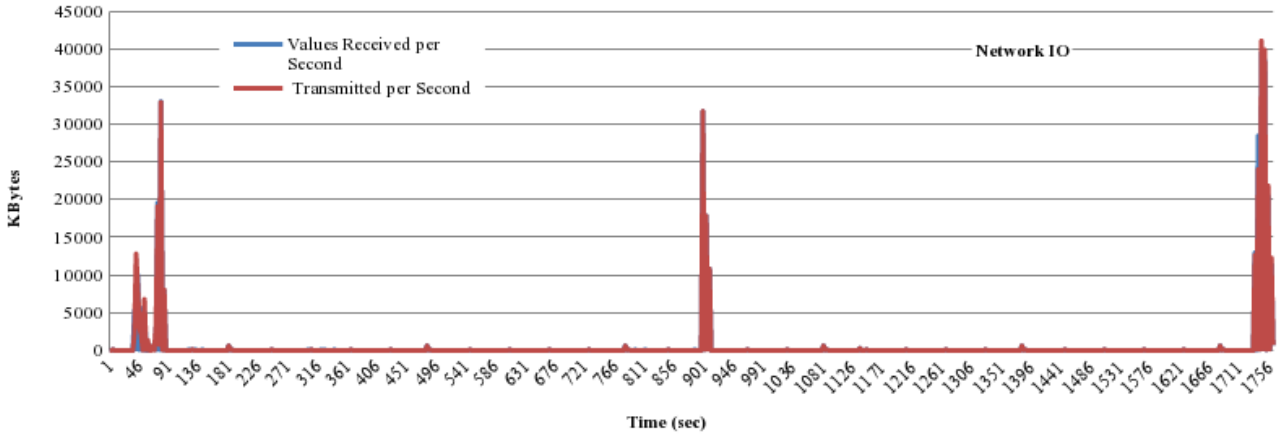


(b) Parquet

Figure 28: Free Memory.

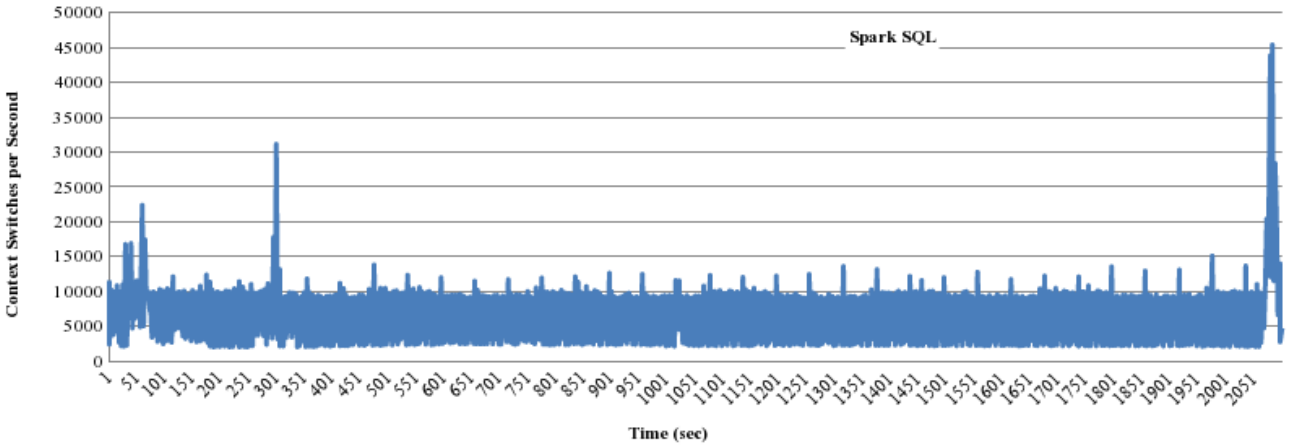


(a) ORC

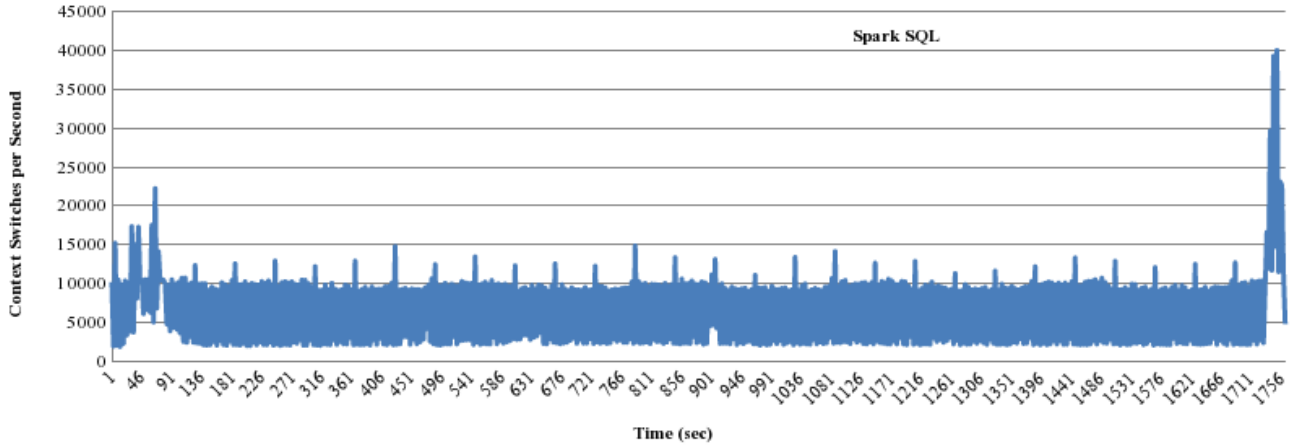


(b) Parquet

Figure 29: Network Utilization: More or less corresponding to DiskIO.



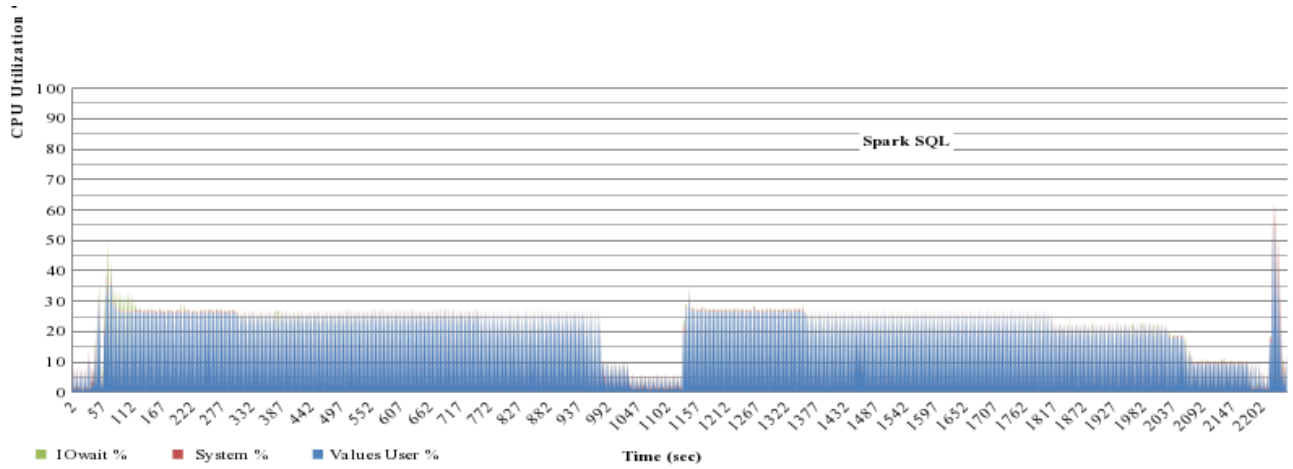
(a) ORC



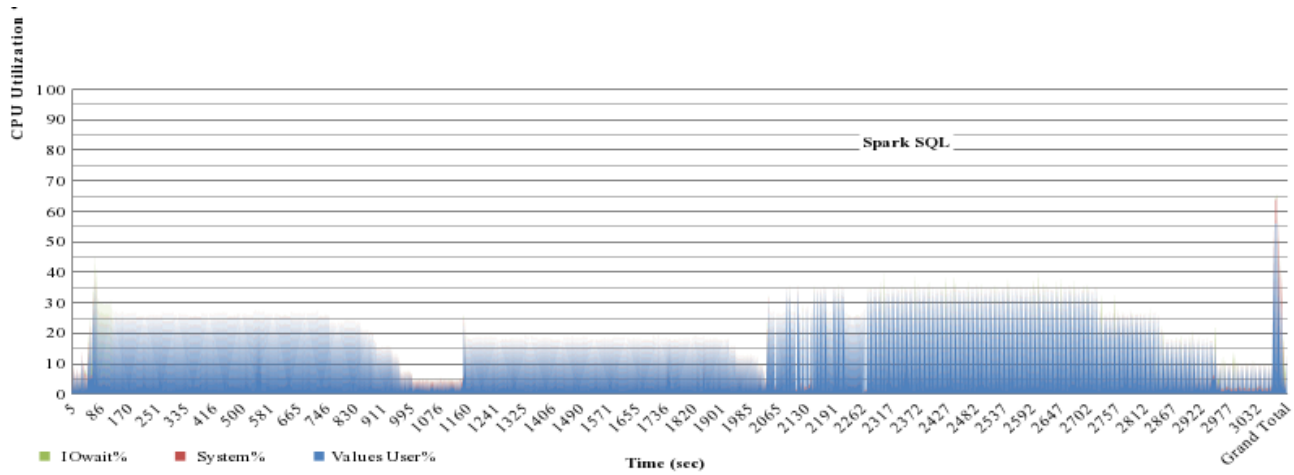
(b) Parquet

Figure 30: VMSTAT Context Switches

2.8 Snappy - ORC vs. Parquet

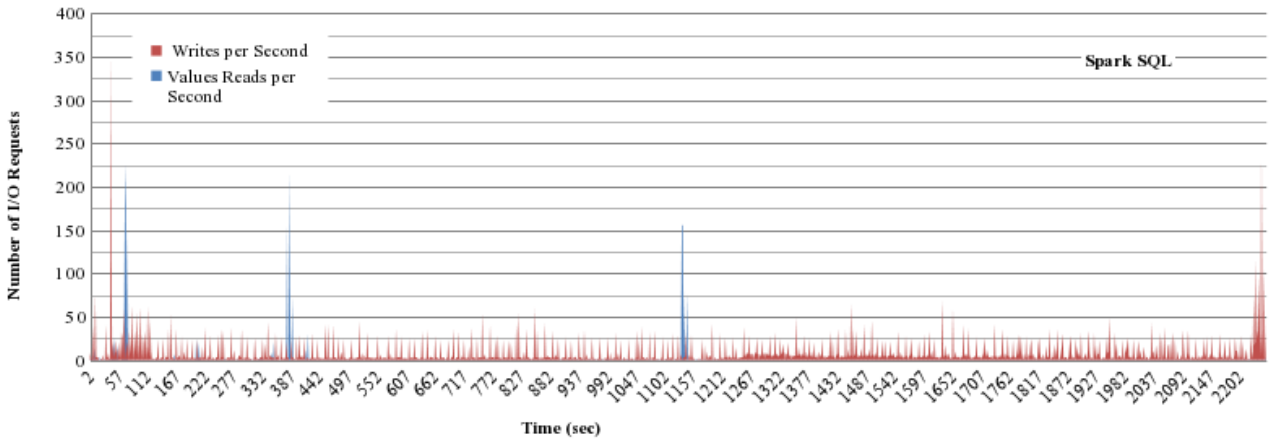


(a) ORC

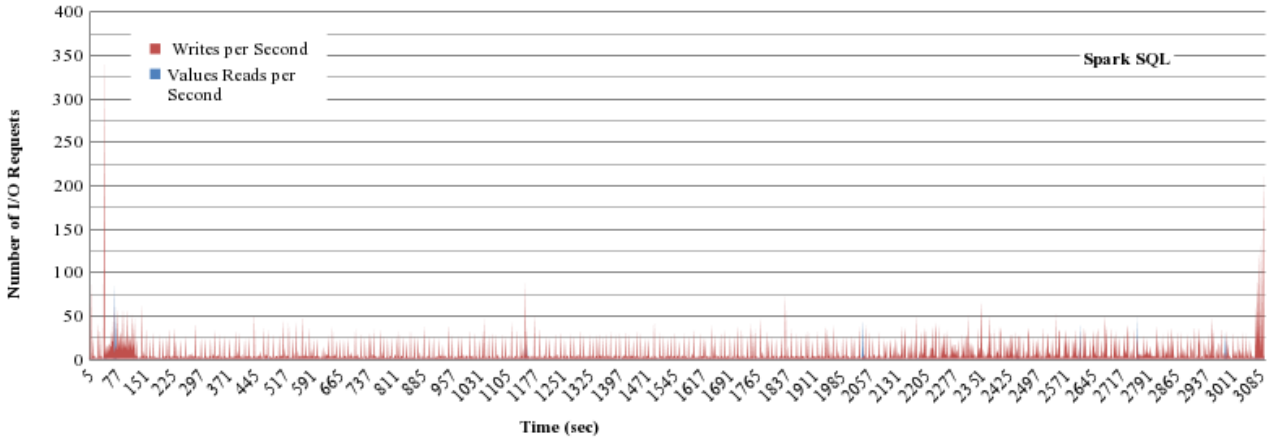


(b) Parquet

Figure 31: CPU: Low util on parquet (60%), short time for phase 1.

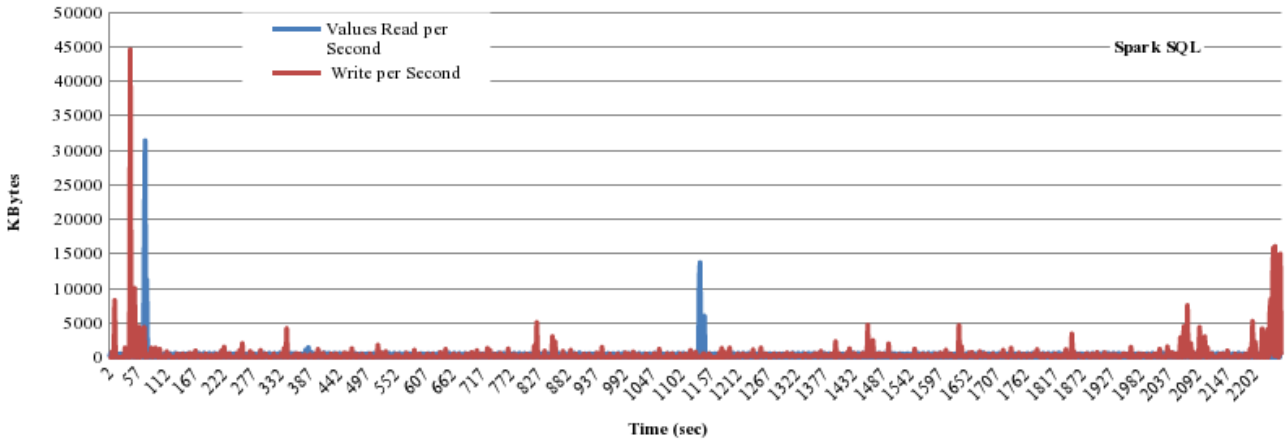


(a) ORC

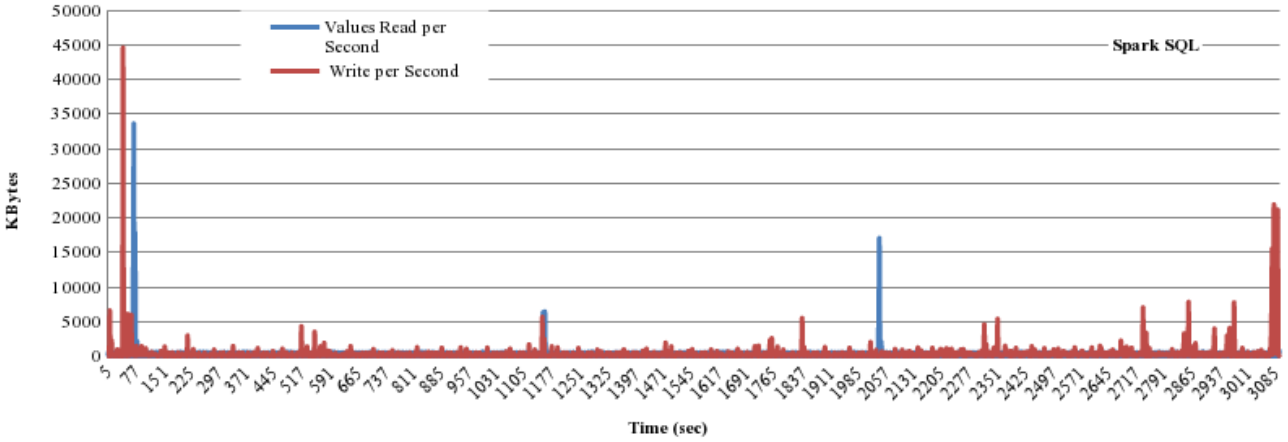


(b) Parquet

Figure 32: Disk Requests: Lower number of requests on Parquet.

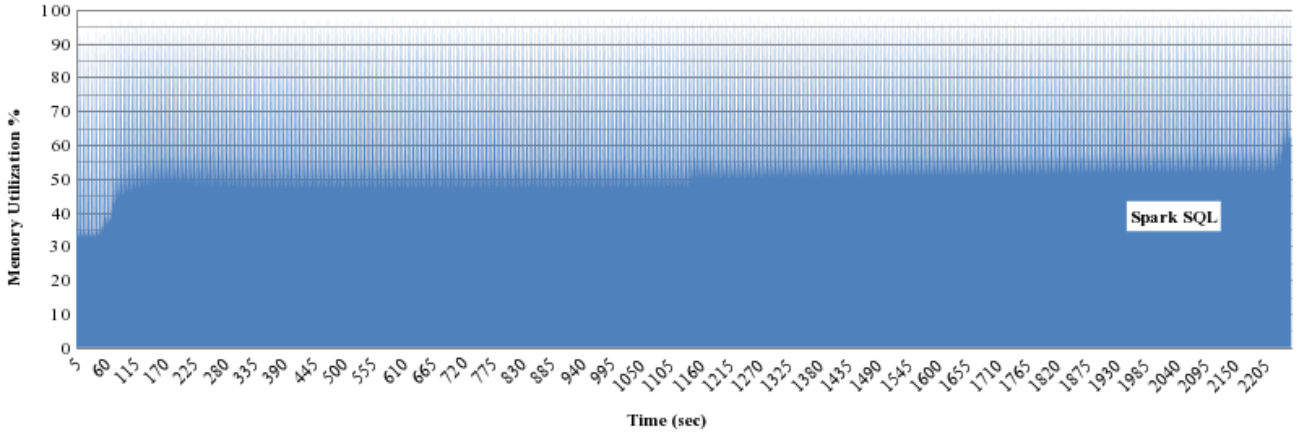


(a) ORC

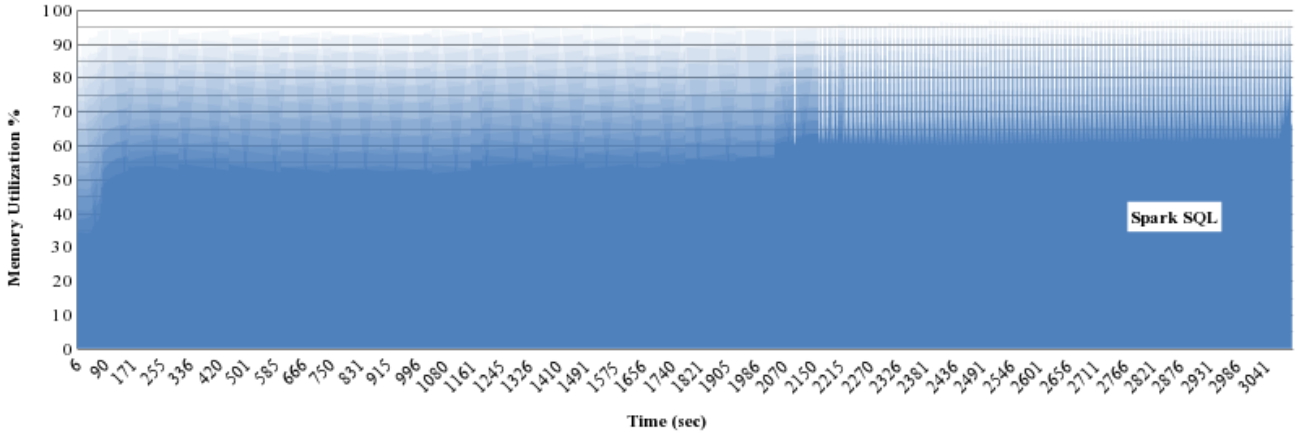


(b) Parquet

Figure 33: Disk Bandwidth: higher BW utilization on Parquet.

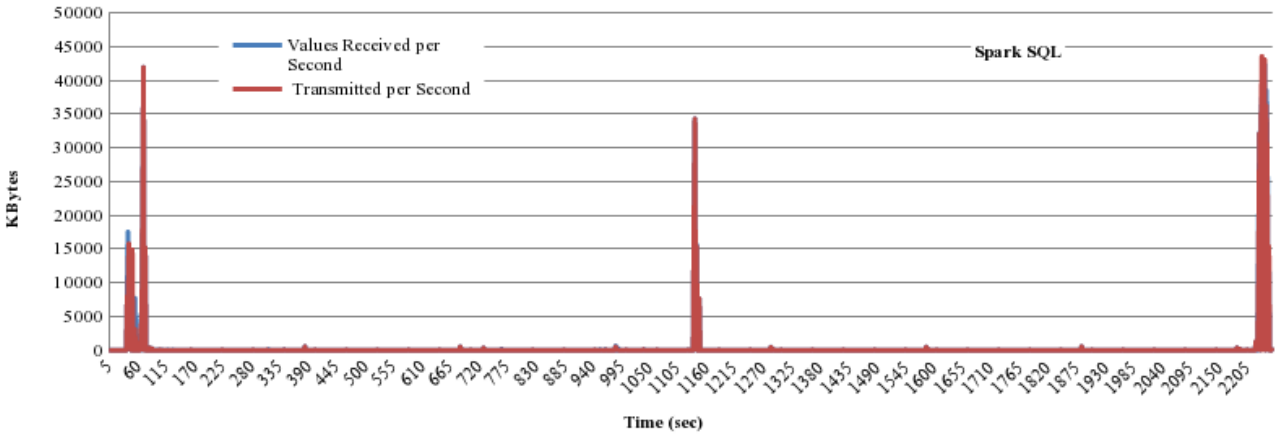


(a) ORC

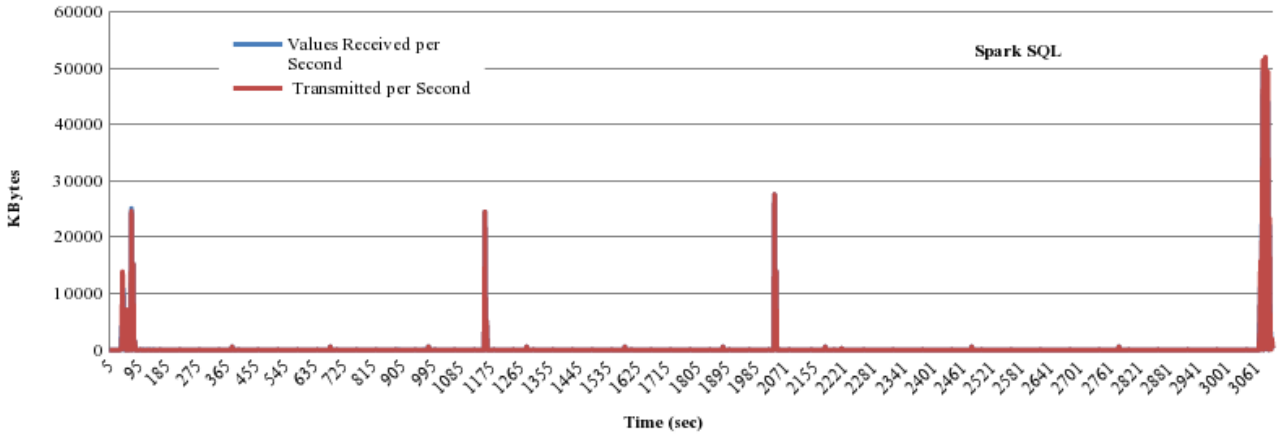


(b) Parquet

Figure 34: Free Memory.

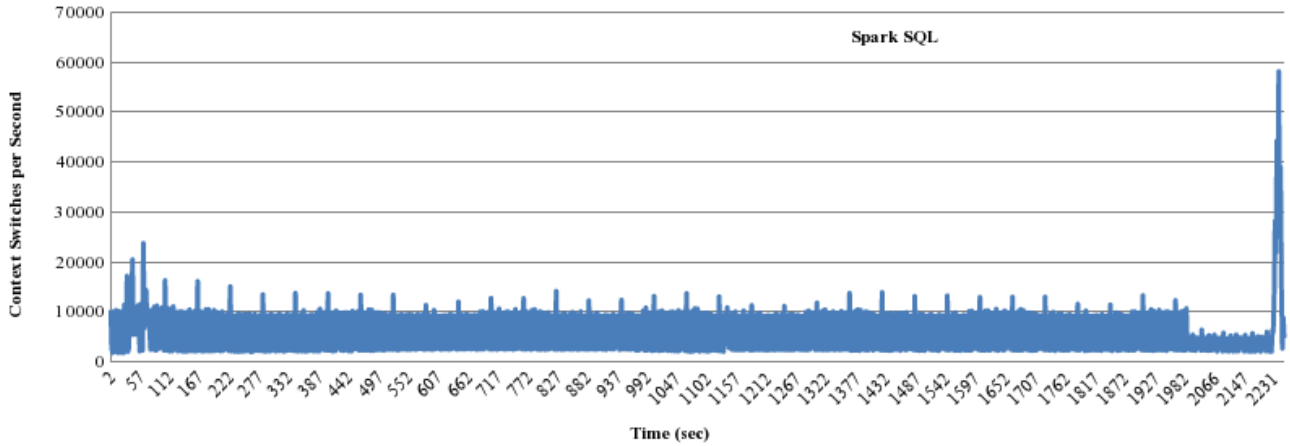


(a) ORC

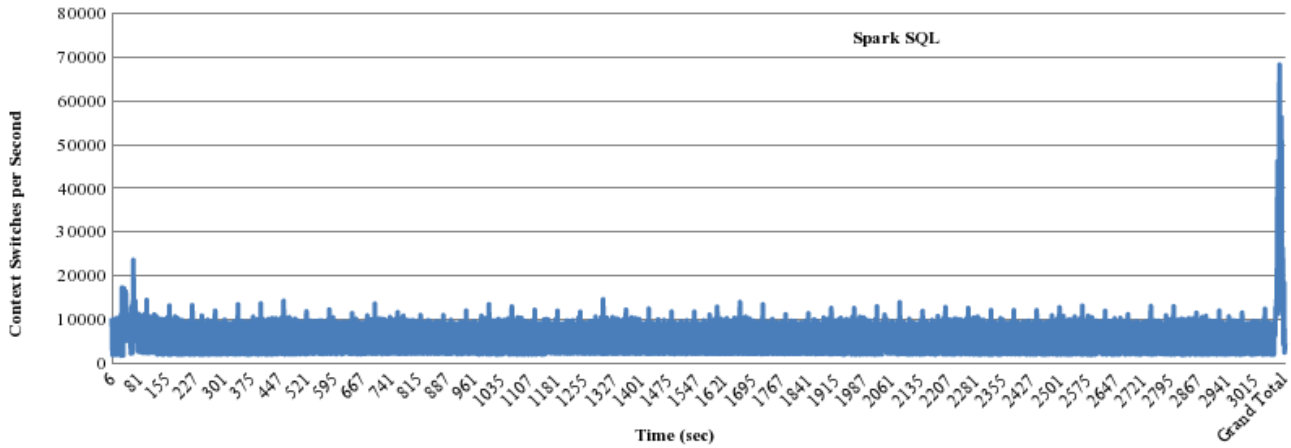


(b) Parquet

Figure 35: Network Utilization: more or less corresponding to DiskIO BW.



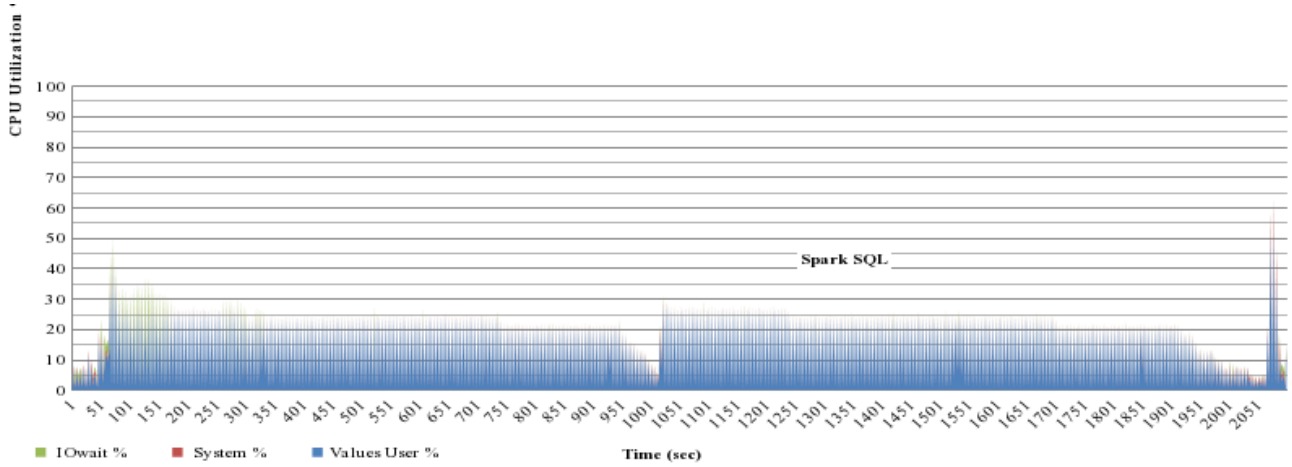
(a) ORC



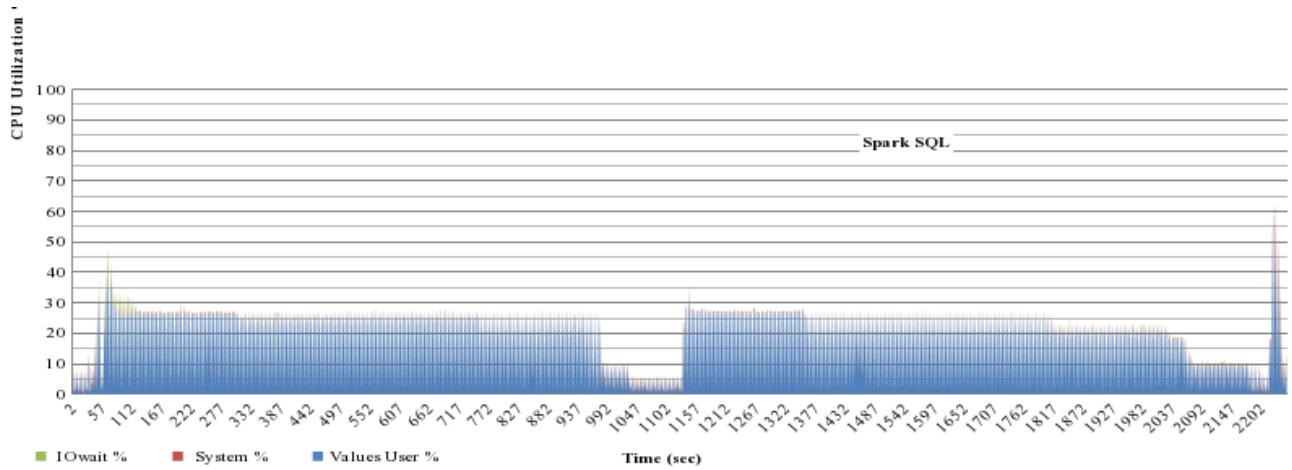
(b) Parquet

Figure 36: VMSTAT Context Switches

2.9 ORC - NoCompression vs. Snappy

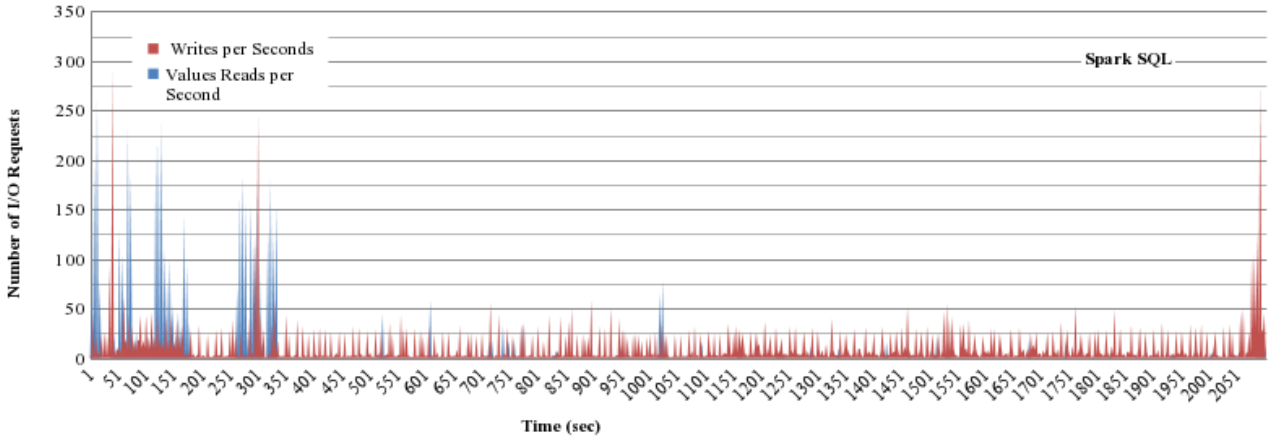


(a) NoCompression

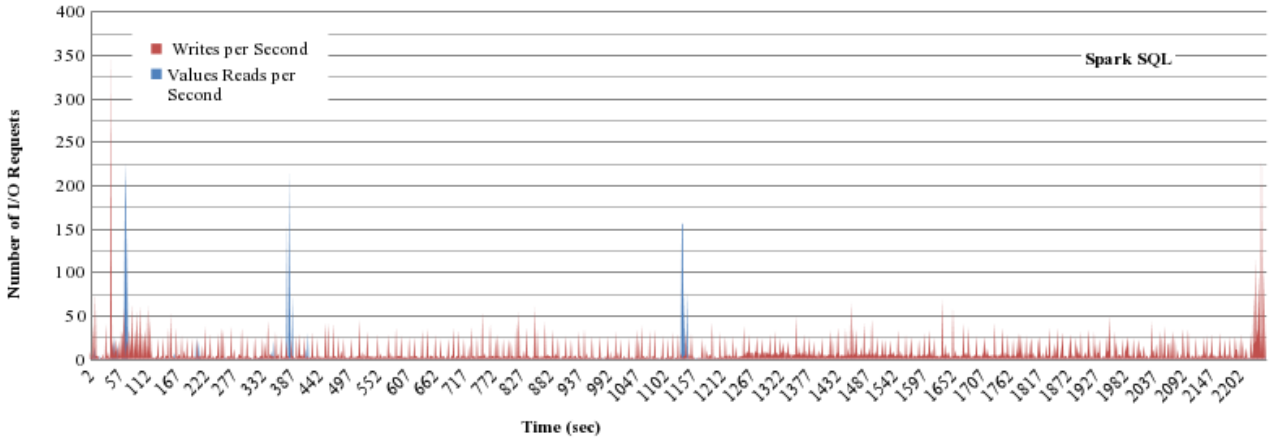


(b) Snappy

Figure 37: CPU.

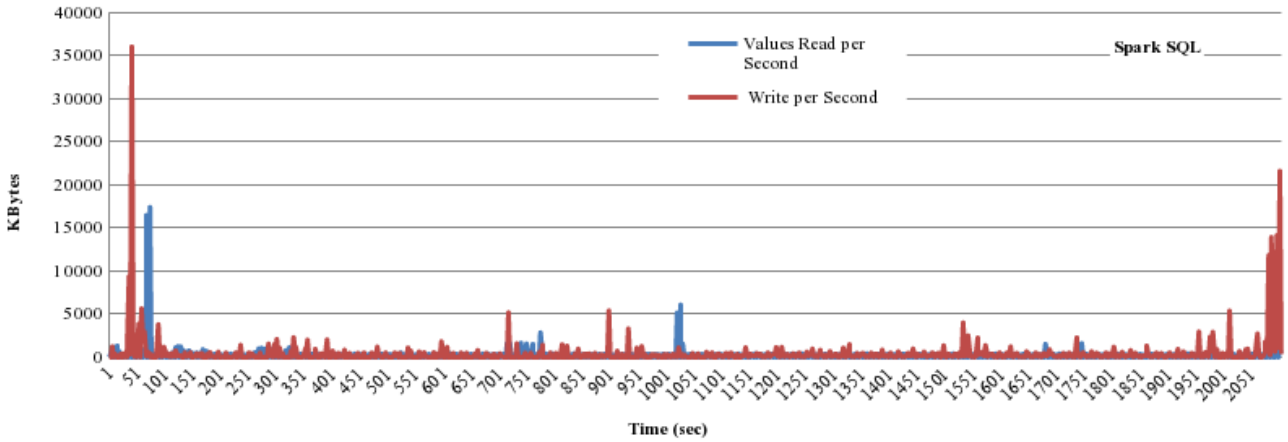


(a) NoCompress

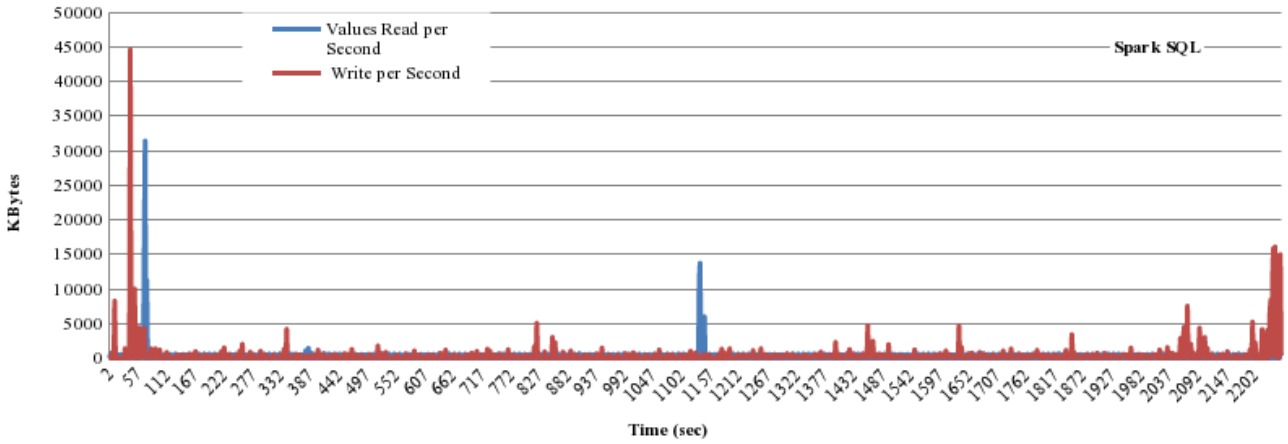


(b) Snappy

Figure 38: Disk Requests.

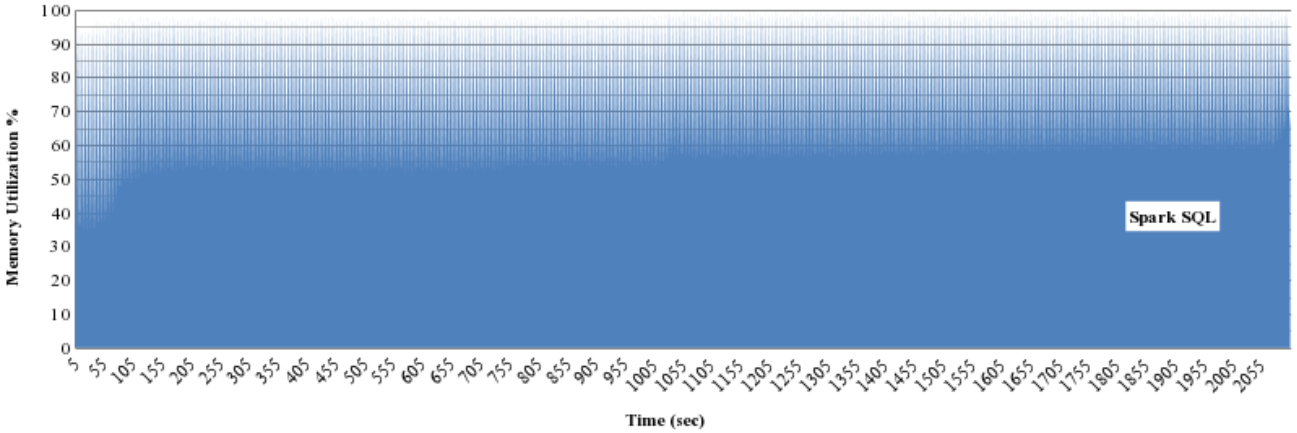


(a) NoCompress

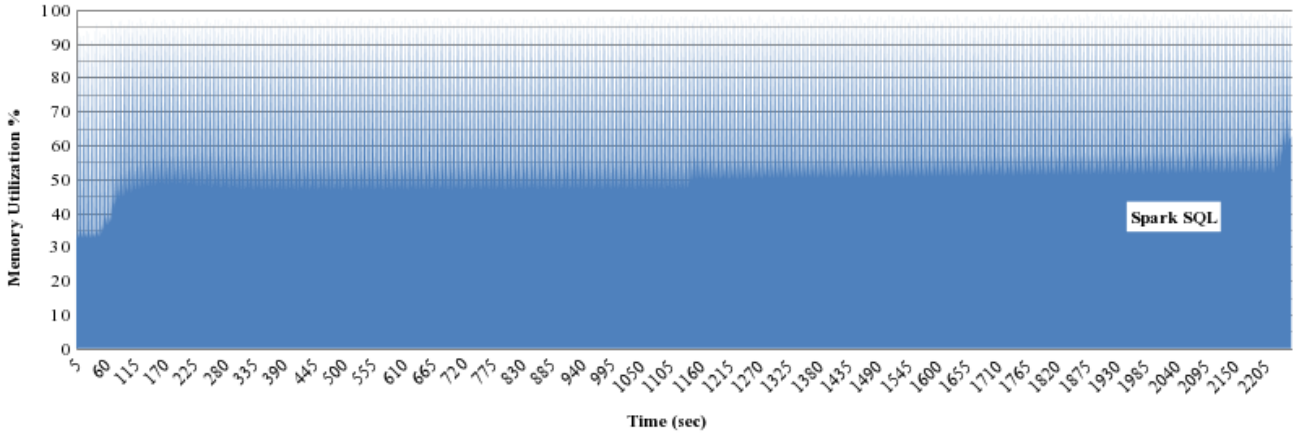


(b) Snappy

Figure 39: Disk Bandwidth.

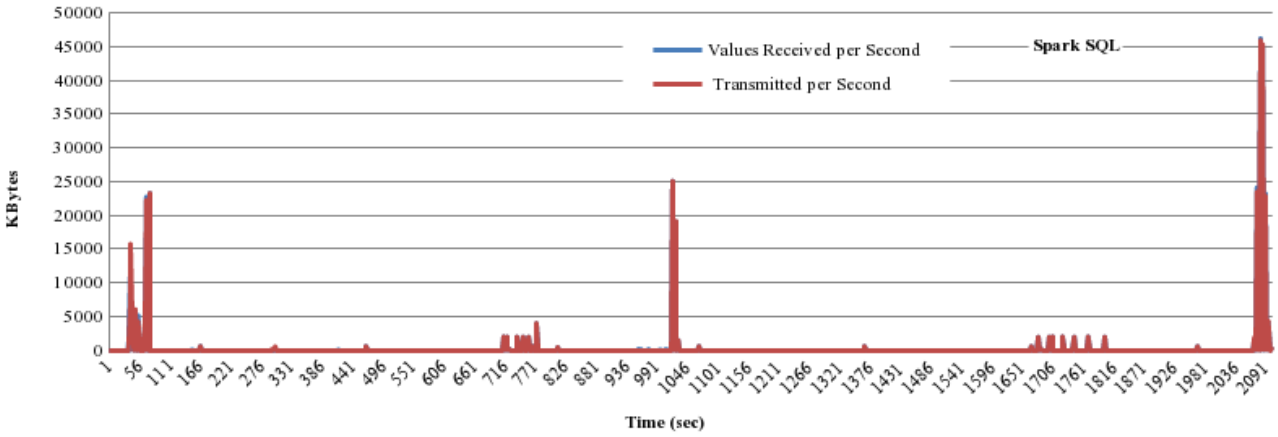


(a) NoCompress

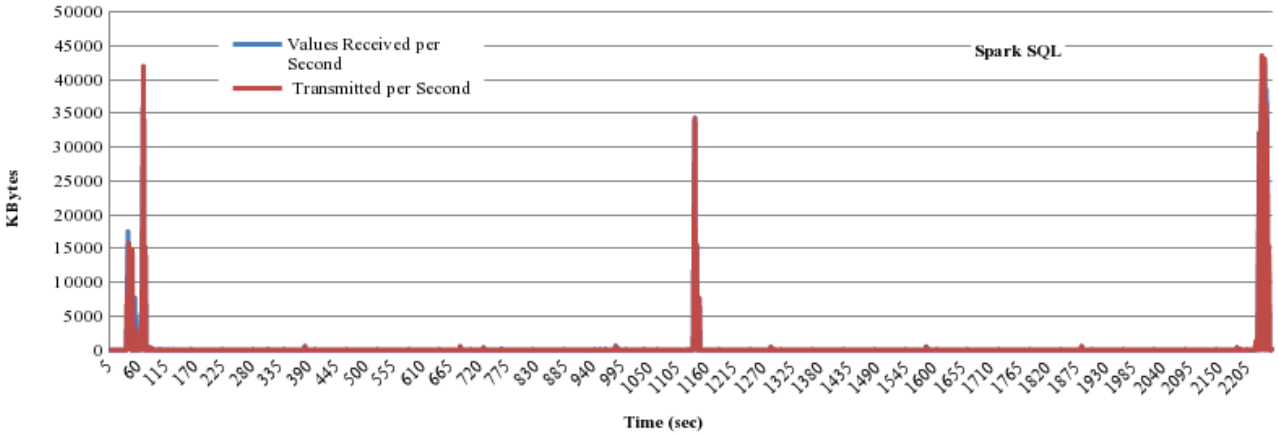


(b) Snappy

Figure 40: Free Memory.

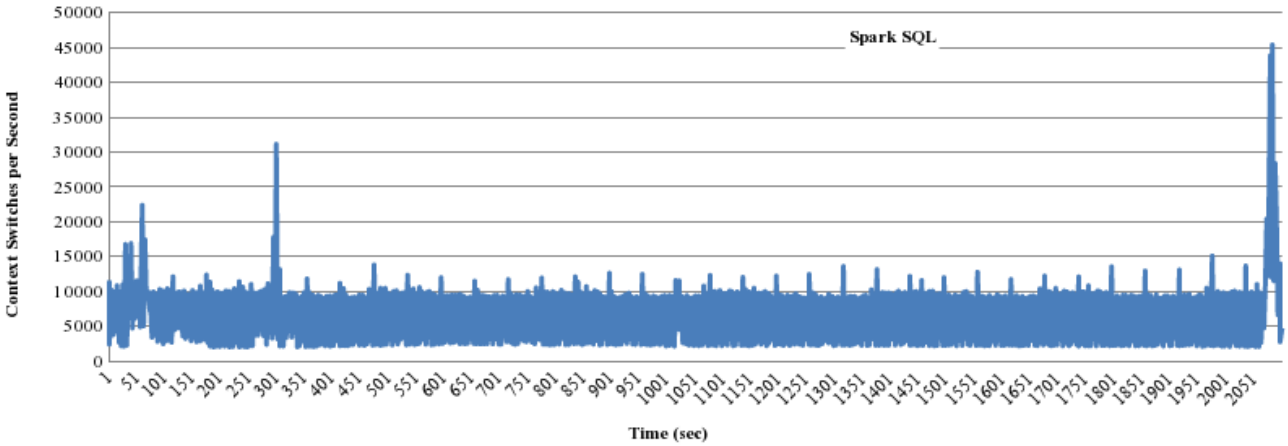


(a) NoCompress

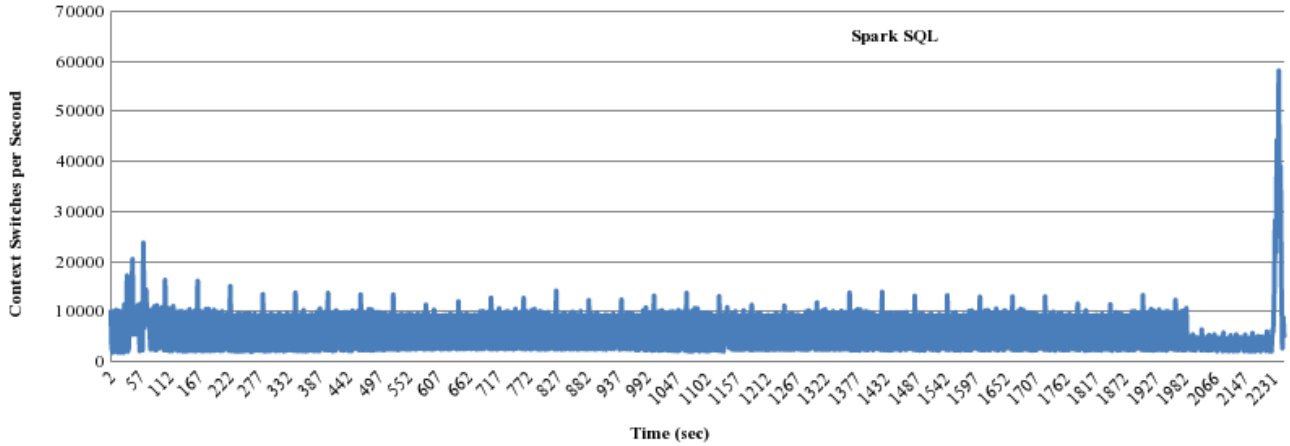


(b) Snappy

Figure 41: Network Utilization.



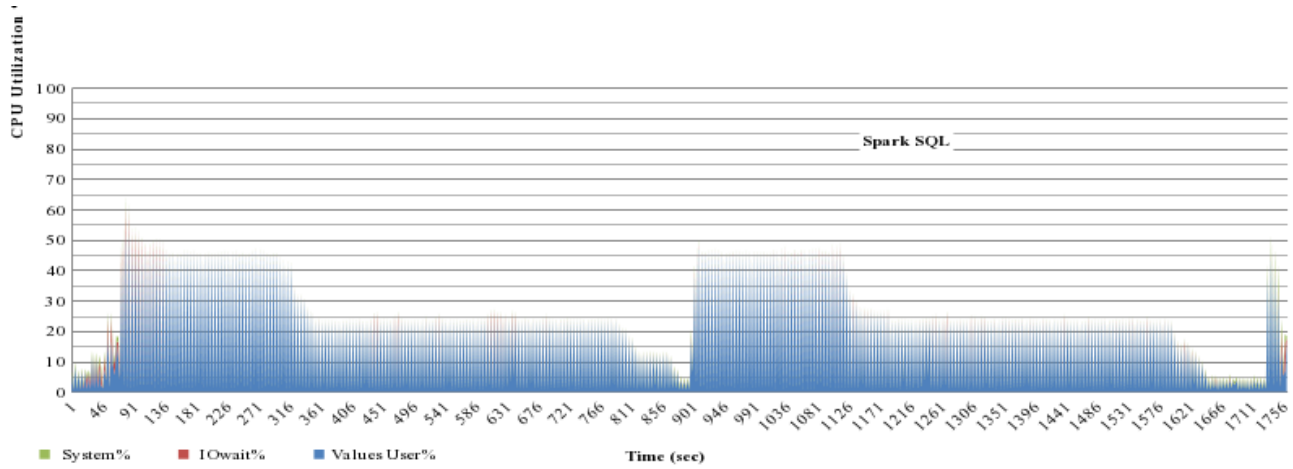
(a) NoCompress



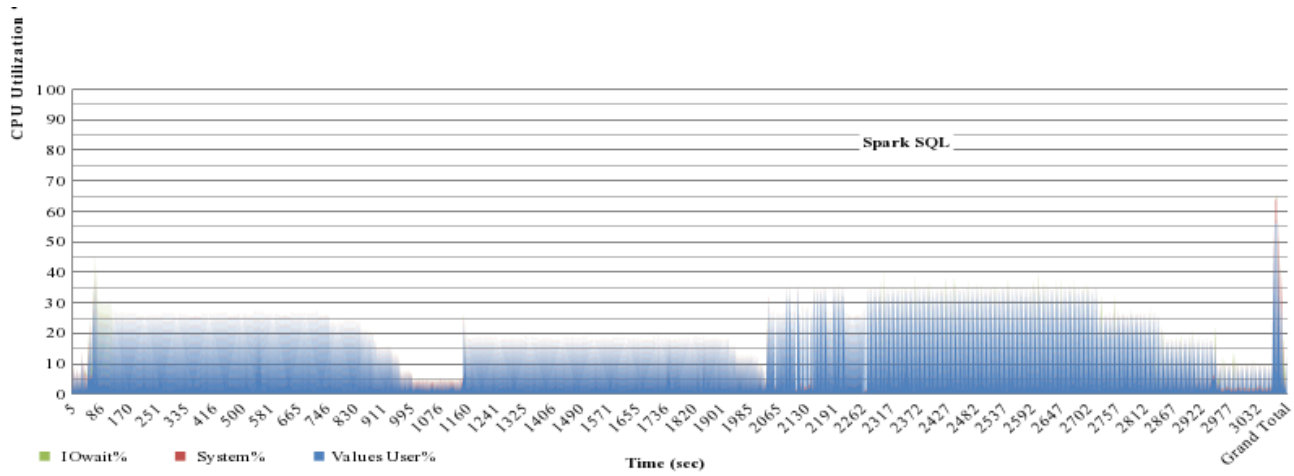
(b) Snappy

Figure 42: VMSTAT Context Switches

2.10 Parquet - NoCompression vs Snappy

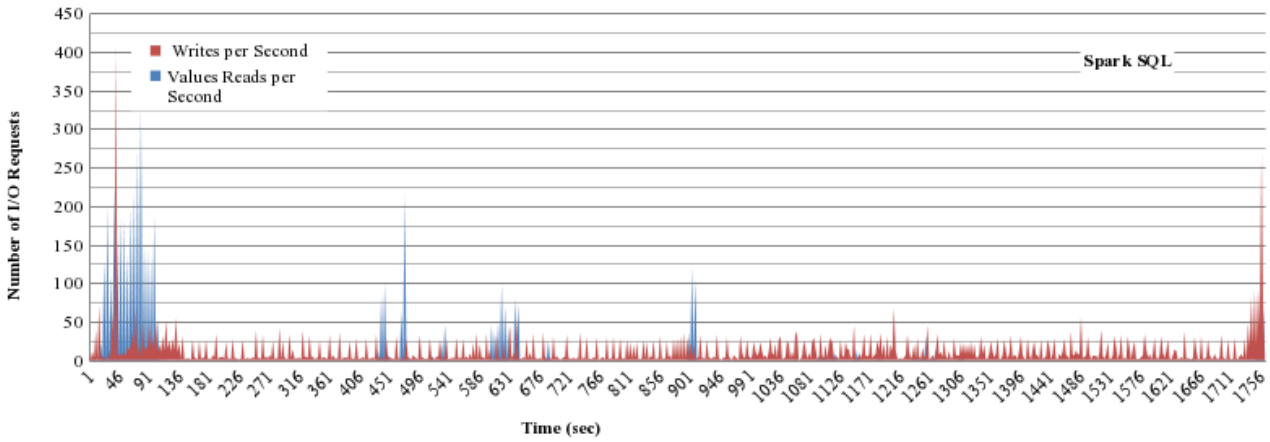


(a) NoCompression

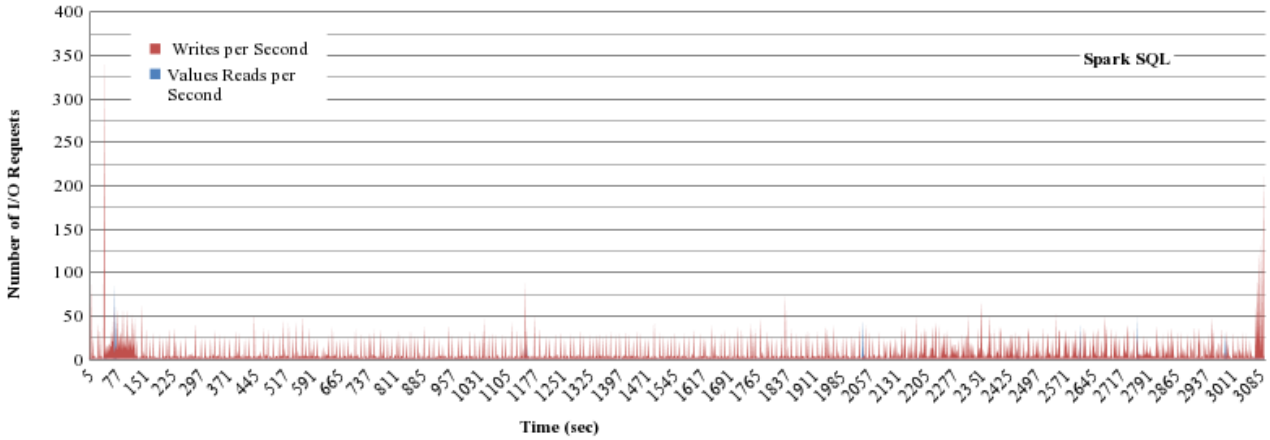


(b) Snappy

Figure 43: CPU.

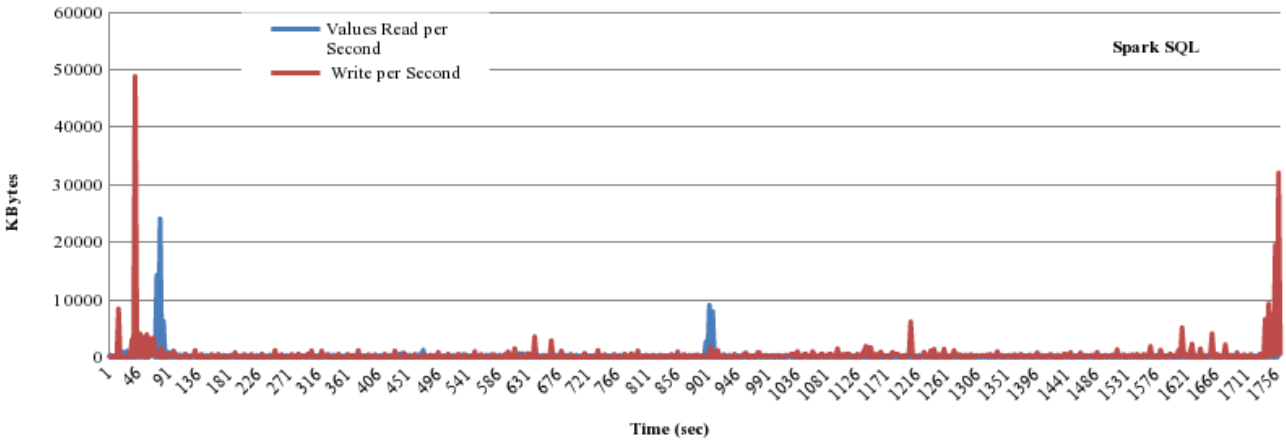


(a) NoCompression

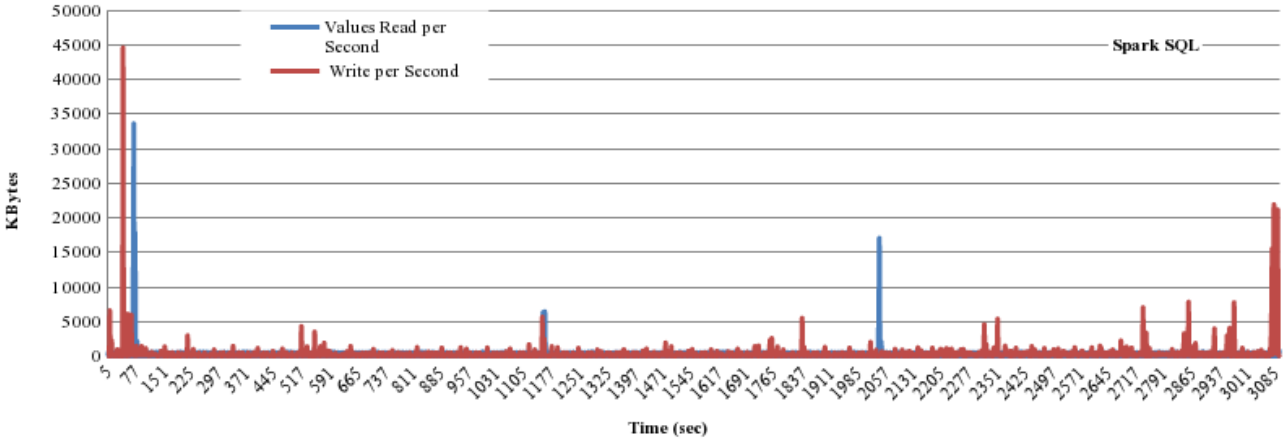


(b) Snappy

Figure 44: Disk Requests.

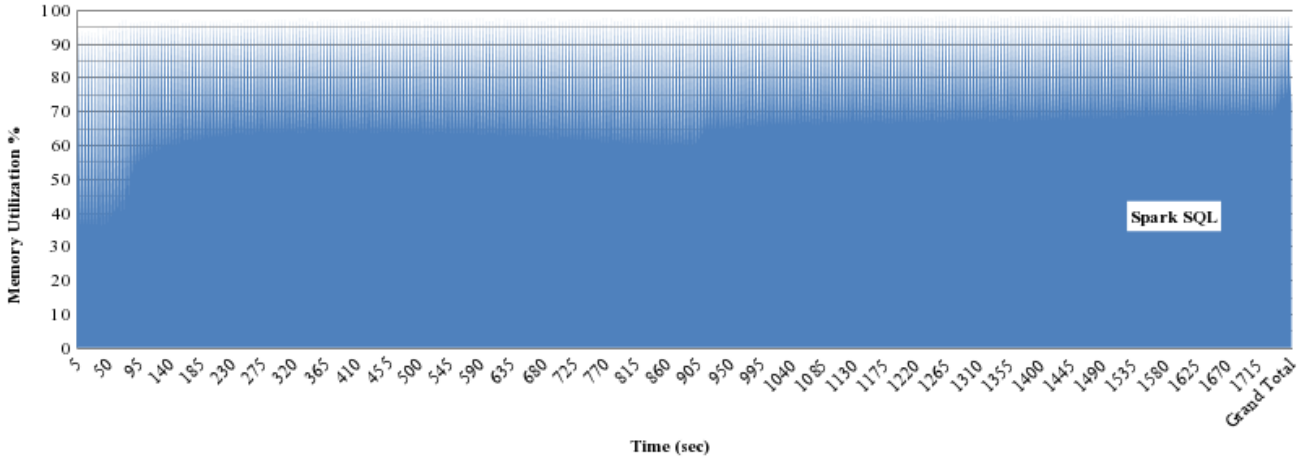


(a) NoCompression

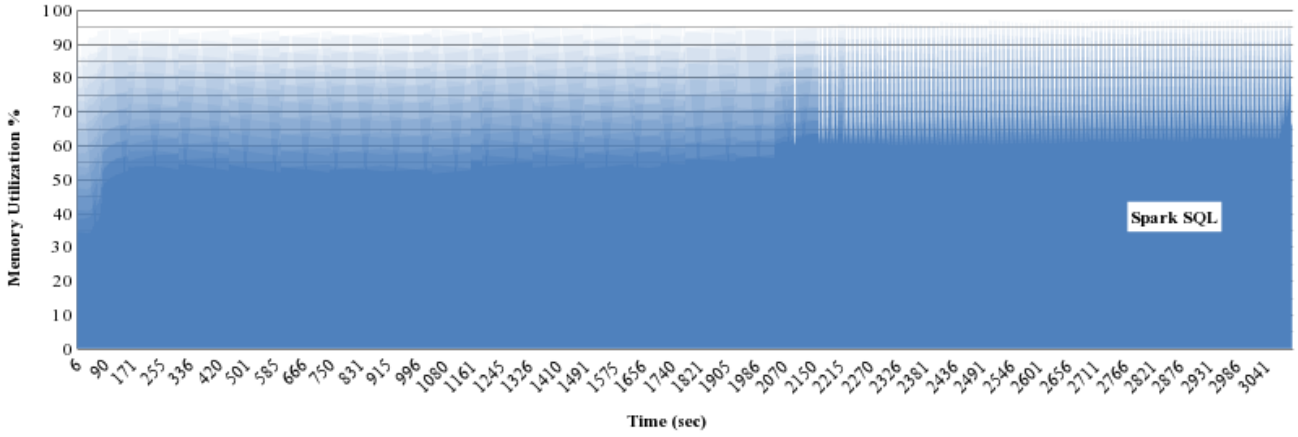


(b) Snappy

Figure 45: Disk Bandwidth.

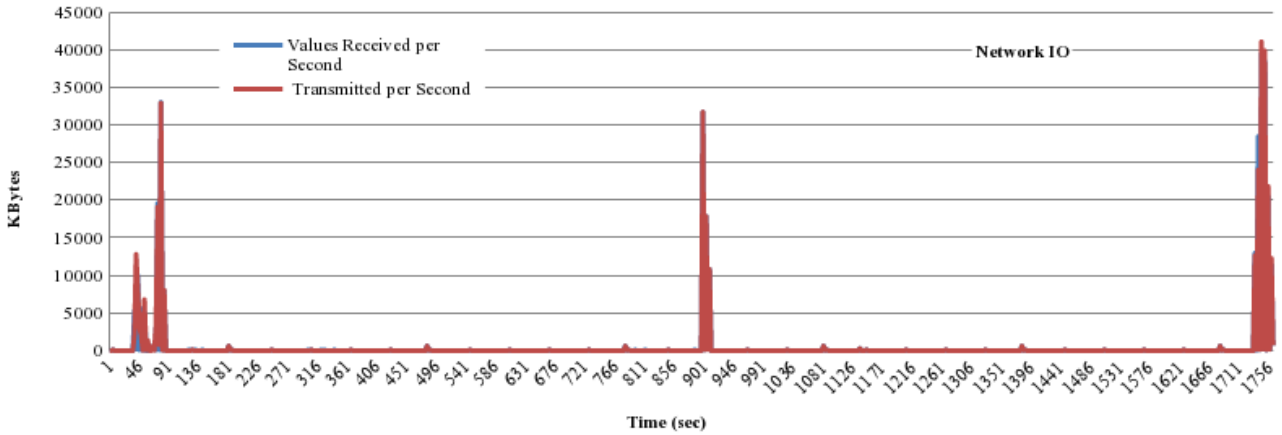


(a) NoCompression

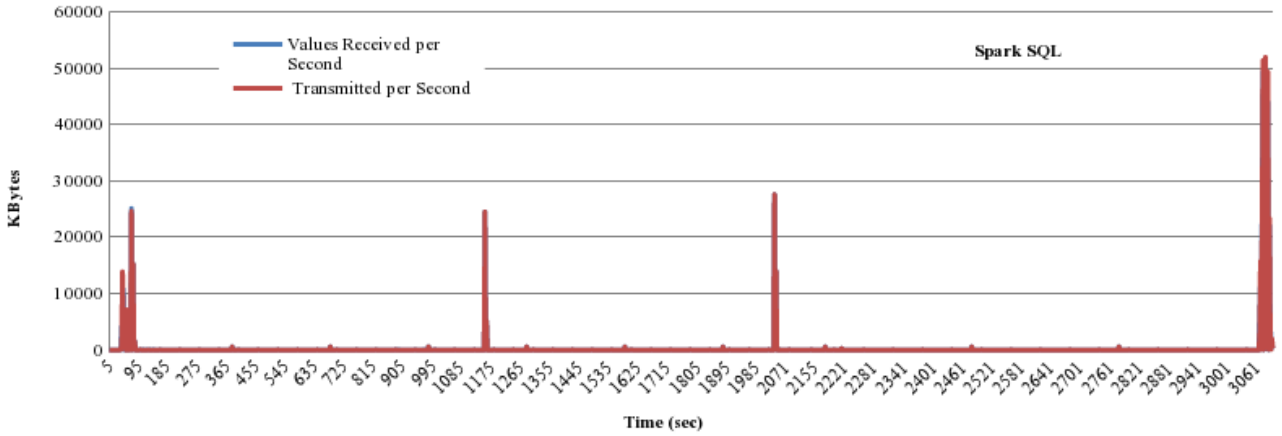


(b) Snappy

Figure 46: Free Memory.

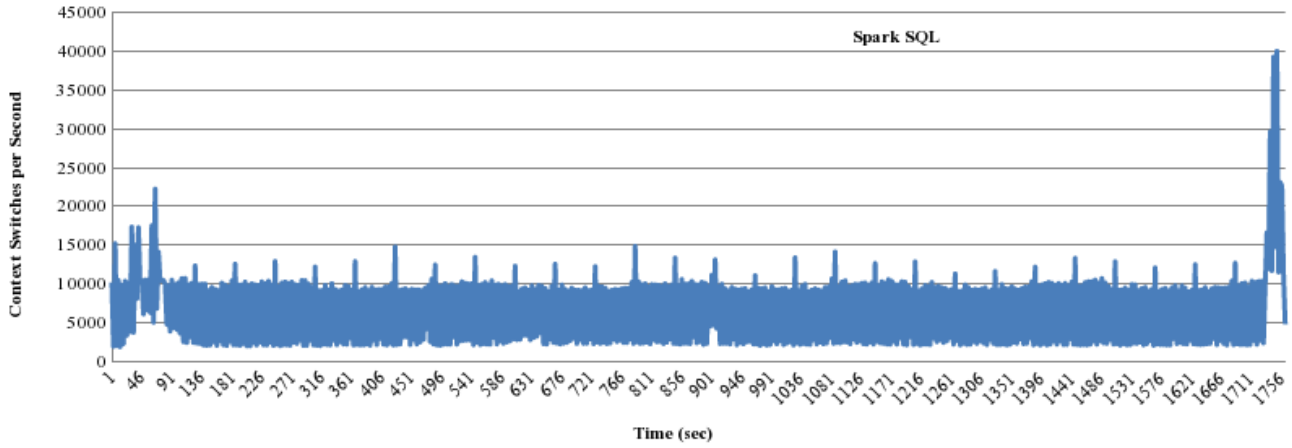


(a) NoCompression

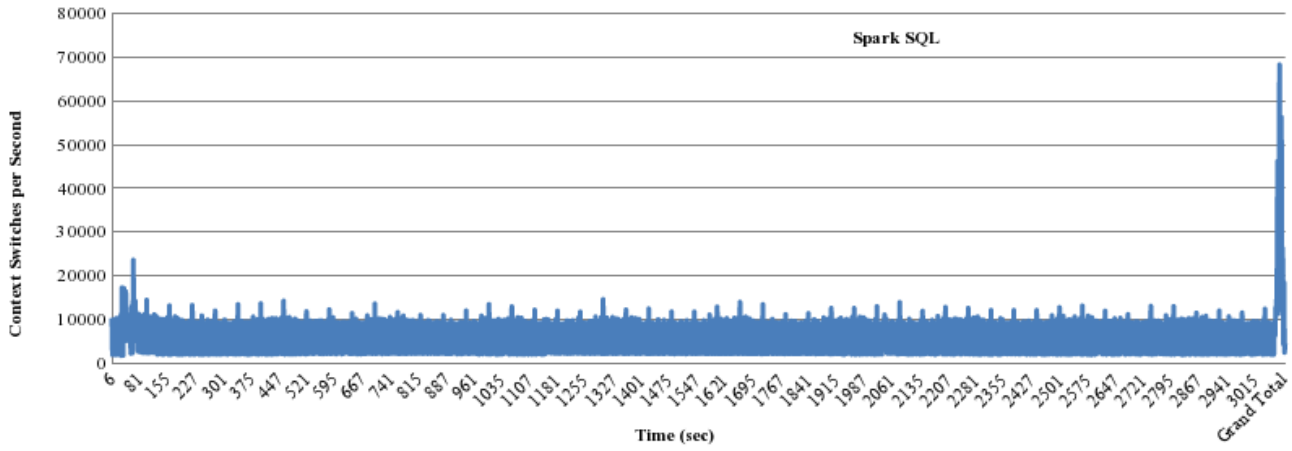


(b) Snappy

Figure 47: Network Utilization.



(a) NoCompression



(b) Snappy

Figure 48: VMSTAT Context Switches

3 Q12

3.1 Type

Pure HiveQL

3.2 Description

Find all customers who viewed items of a given category on the web in a given month and year that was followed by an in-store purchase of an item from the same category in the three consecutive months.

3.3 Performance

	Default (sec.)		No compr. (sec.)		Snappy (sec.)	
Query	**ORC**	**Parquet**	**ORC**	**Parquet**	**ORC**	**Parquet**
Q12	331	105	331	118	314	122

3.4 Phase 1

There are no temp tables. The query gets users with the specified requirements from physical store and web store and joins them. Then, it shows the result.

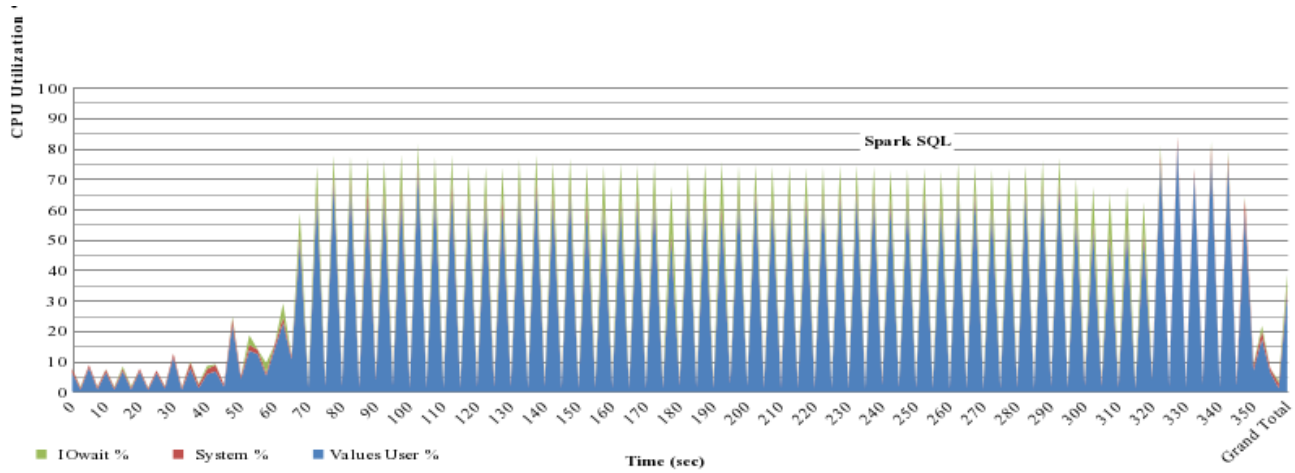
3.5 Source

GitHub: https://git.io/vpFey

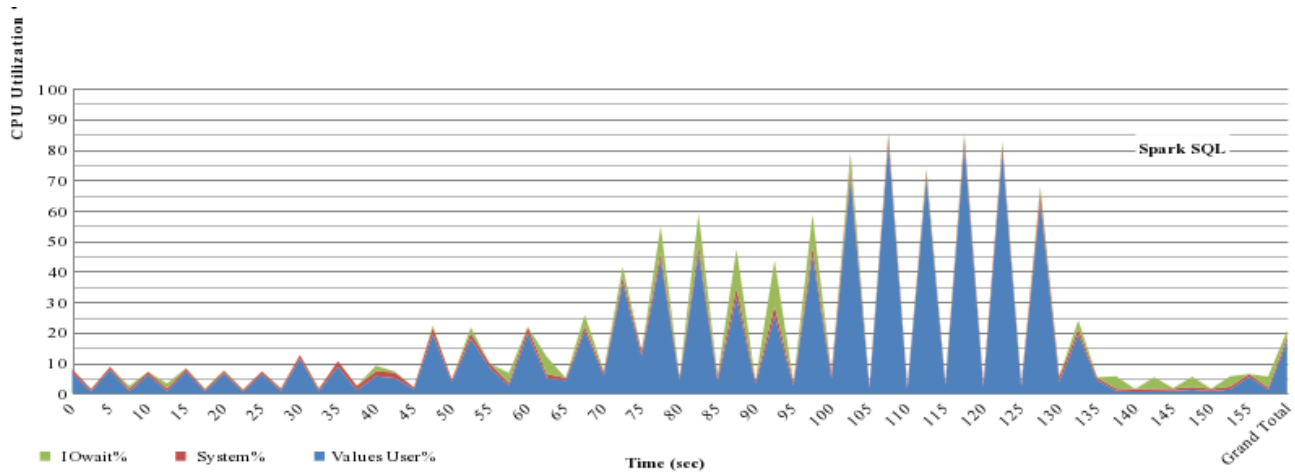
3.6 Notes

...

3.7 No compression - ORC vs Parquet

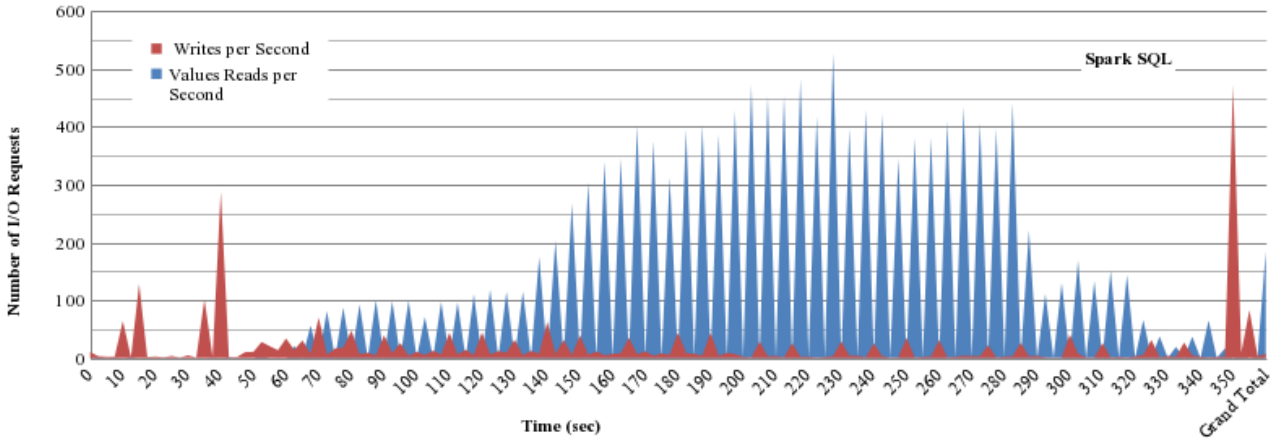


(a) ORC

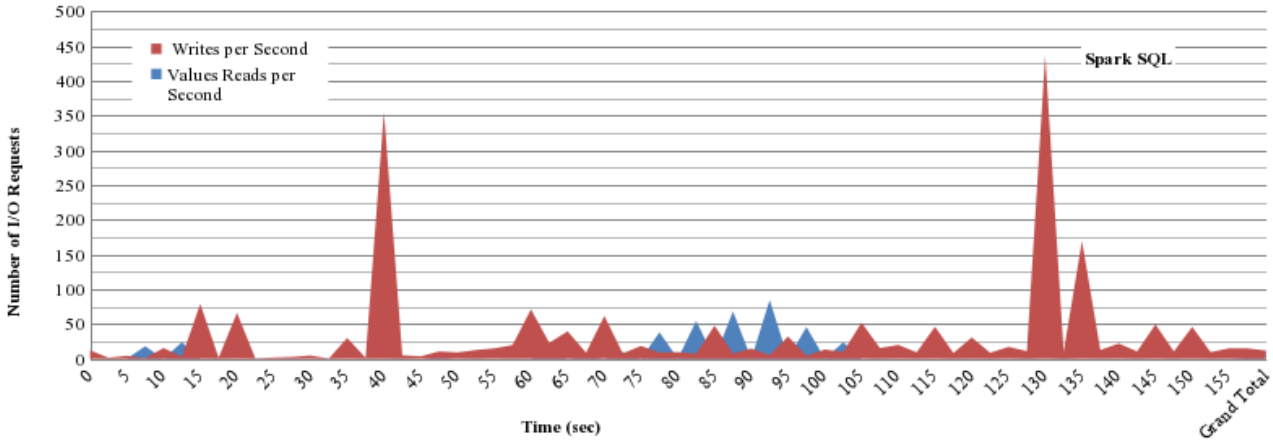


(b) Parquet

Figure 49: CPU: Lower utilization on Parquet (max is 80%)

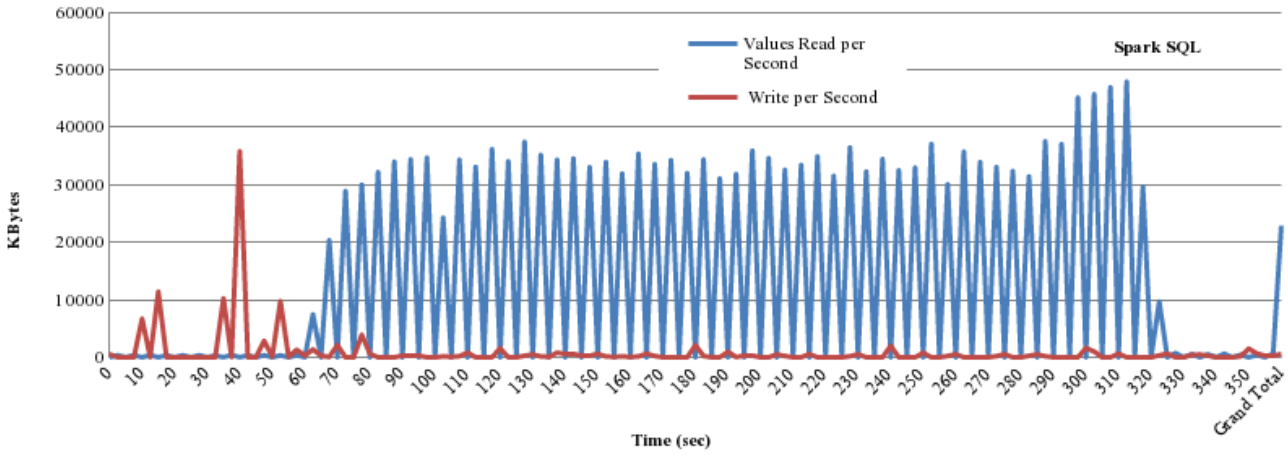


(a) ORC

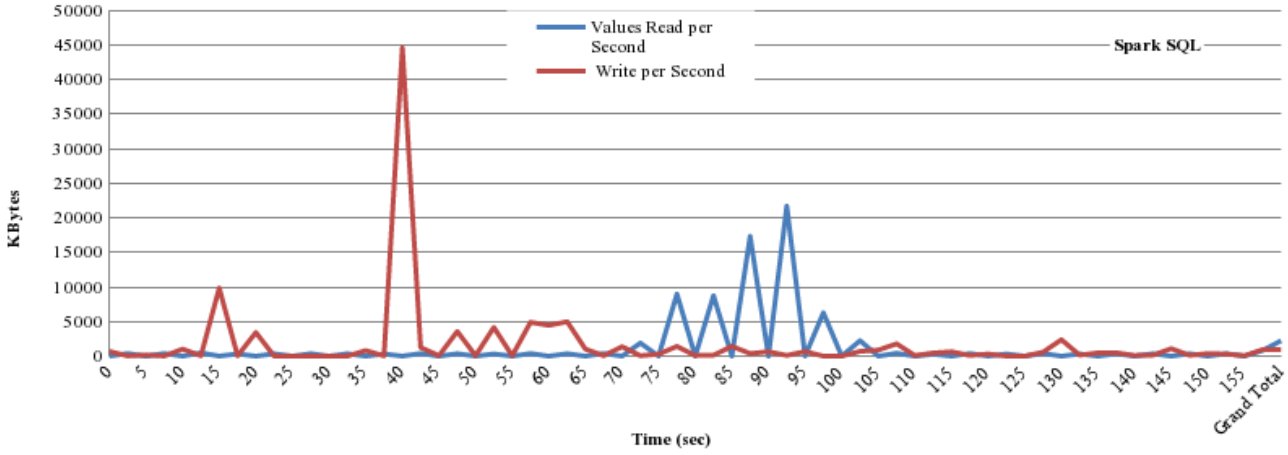


(b) Parquet

Figure 50: Disk Requests: More requests on Parquet.

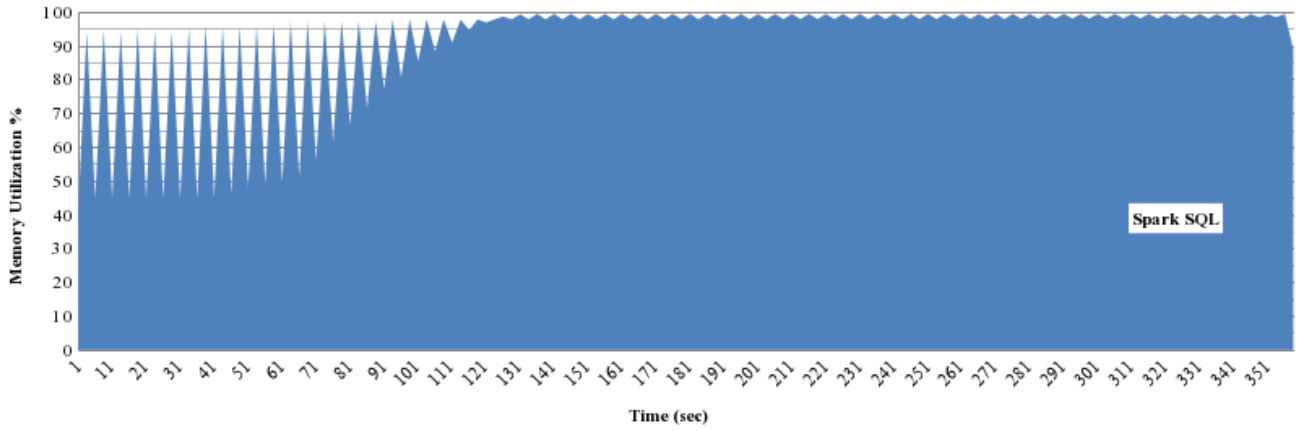


(a) ORC

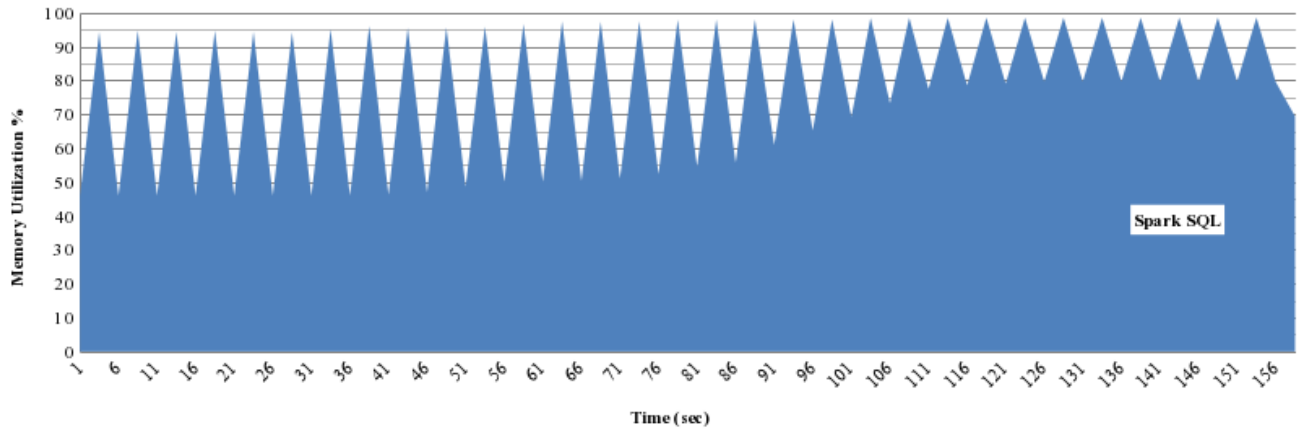


(b) Parquet

Figure 51: Disk Bandwidth: bandwidth utilization on Parquet.

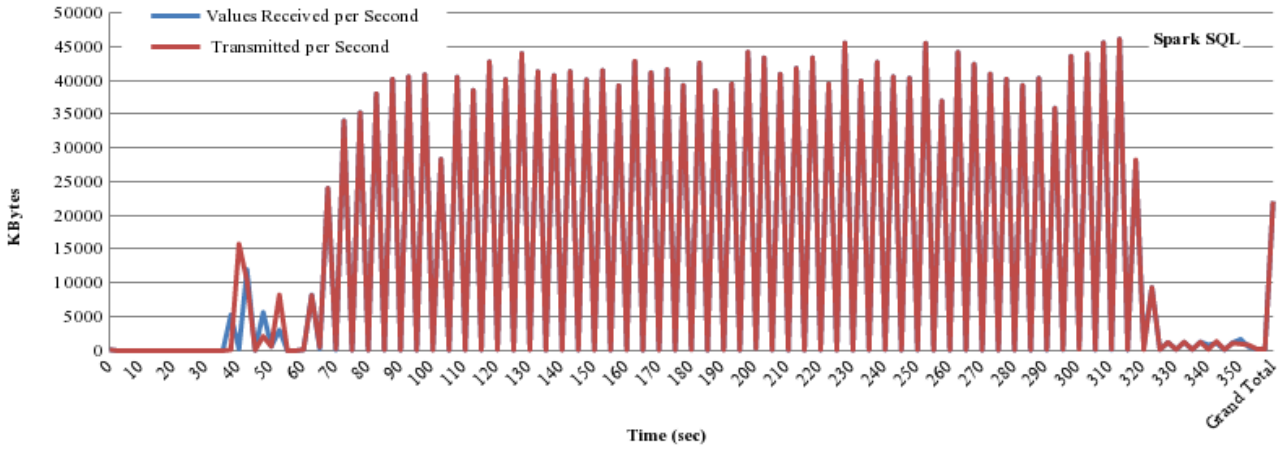


(a) ORC

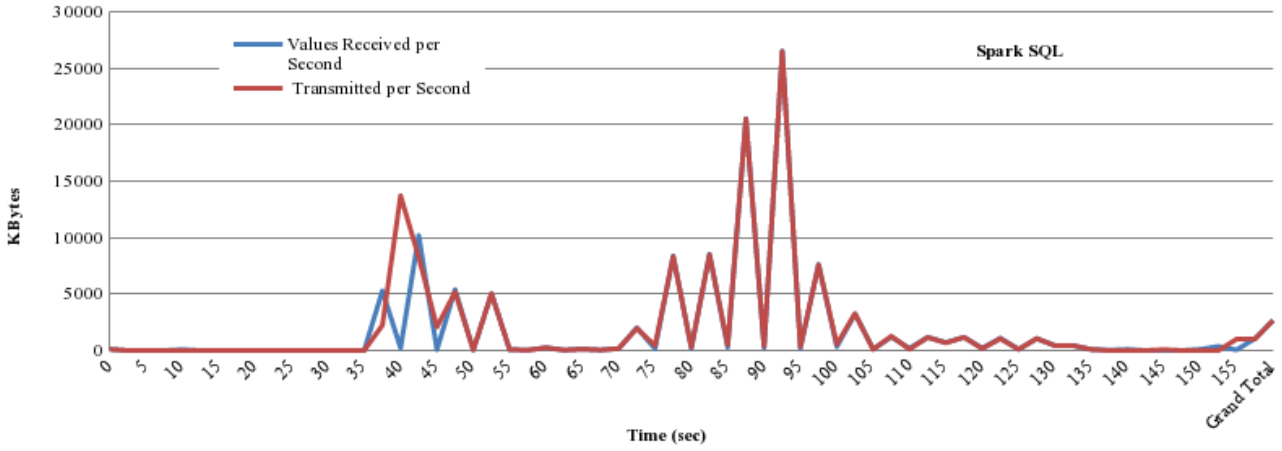


(b) Parquet

Figure 52: Free Memory.

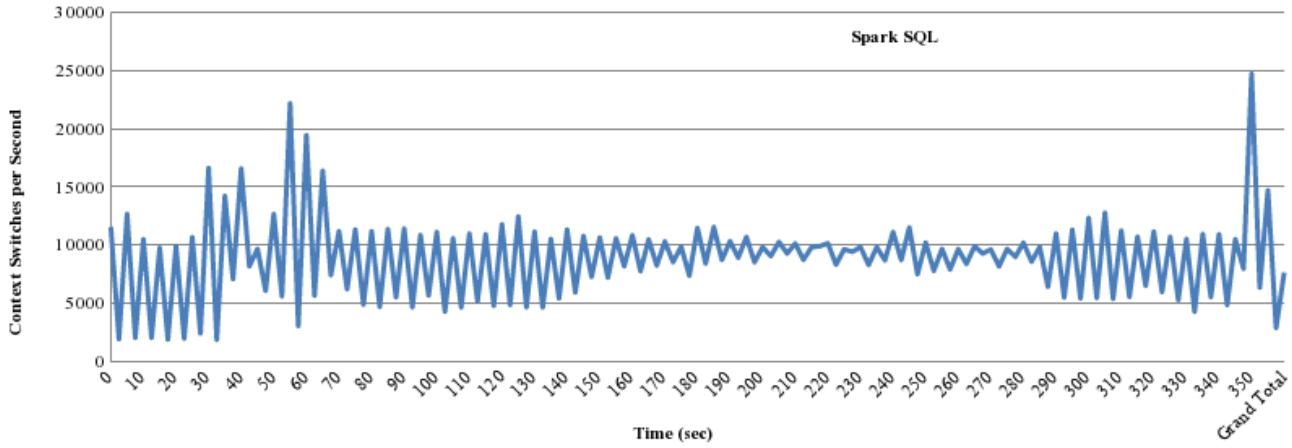


(a) ORC

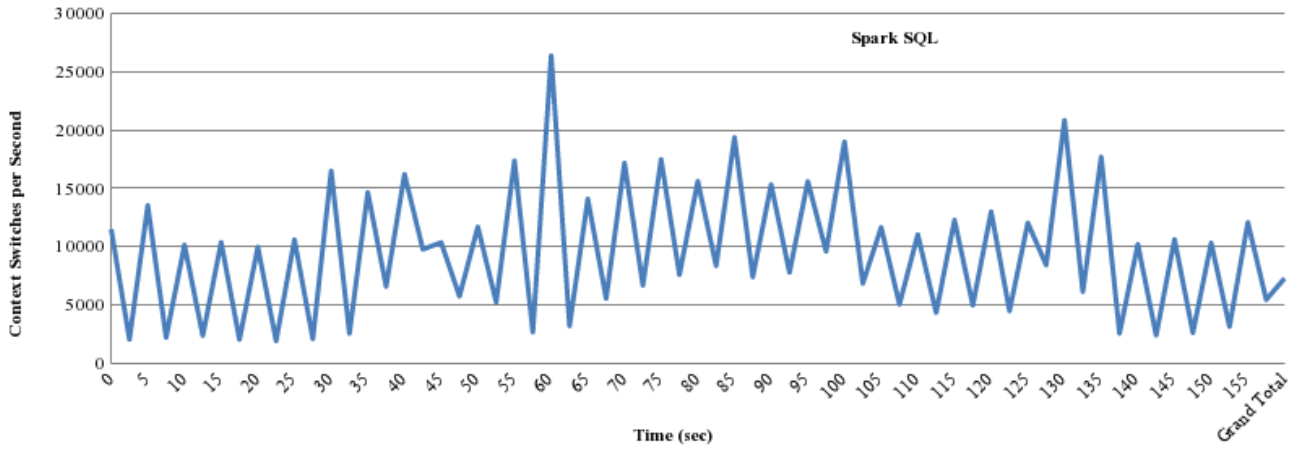


(b) Parquet

Figure 53: Network Utilization: More or less corresponding to DiskIO.



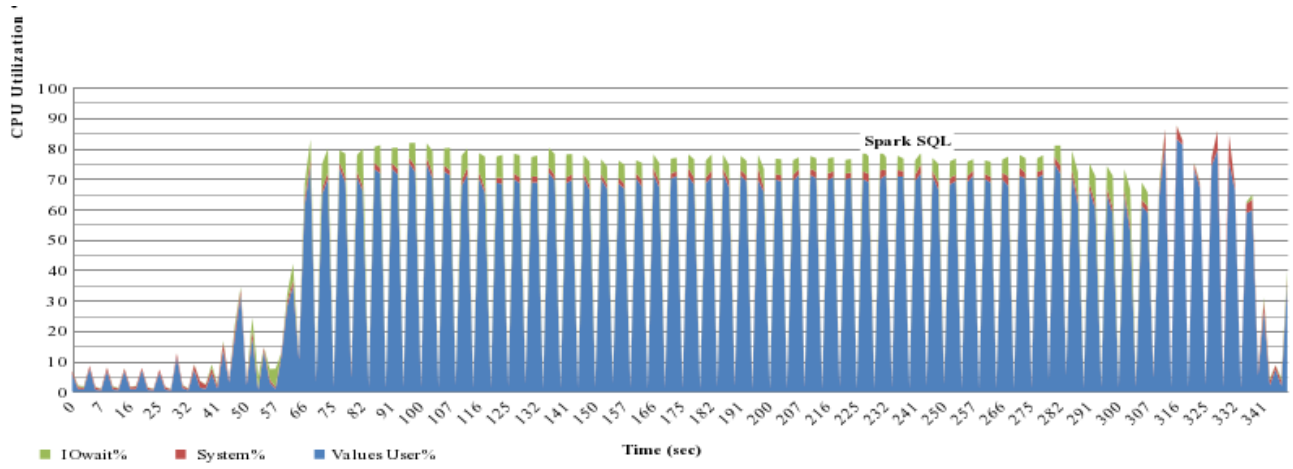
(a) ORC



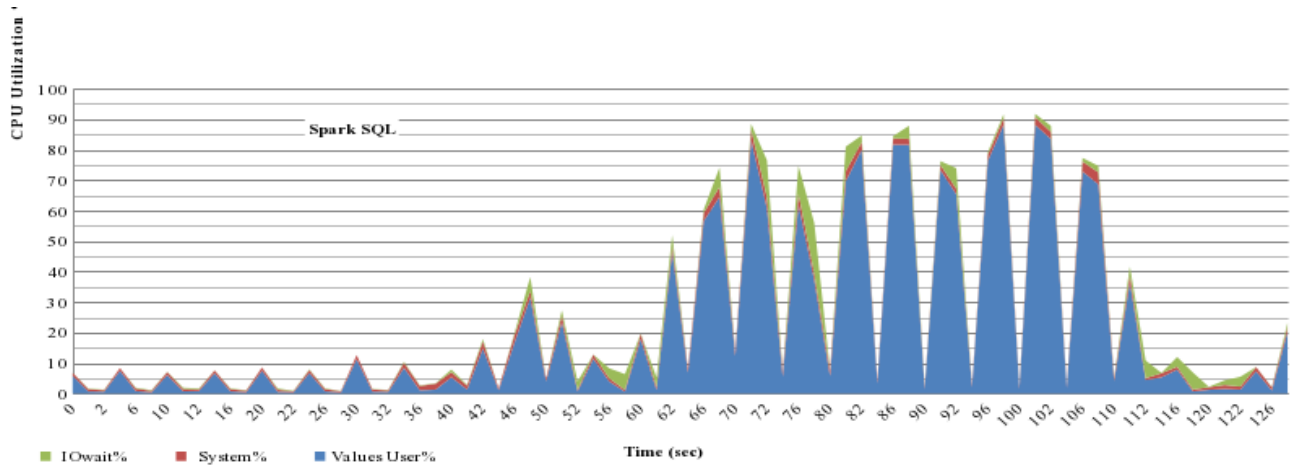
(b) Parquet

Figure 54: VMSTAT Context Switches

3.8 Snappy - ORC vs. Parquet

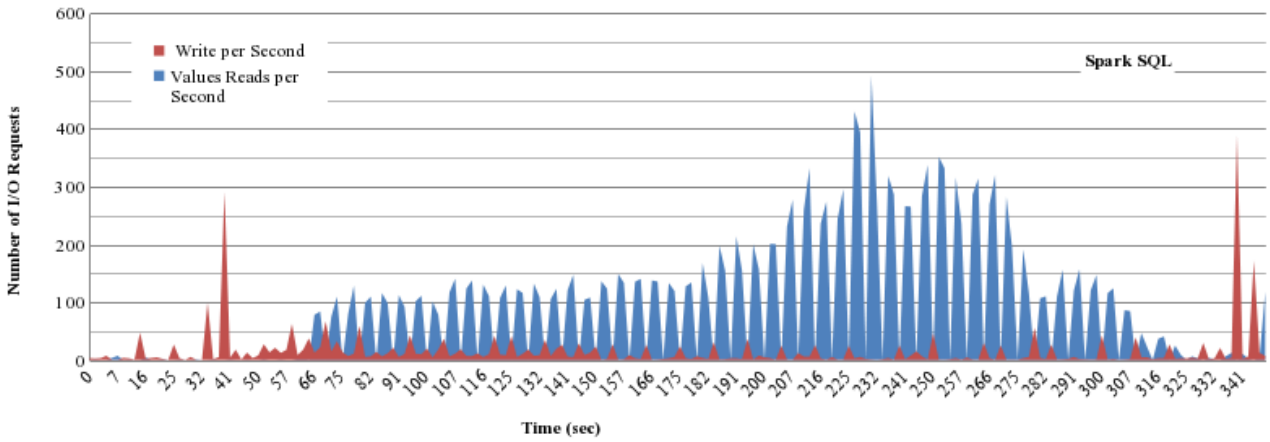


(a) ORC

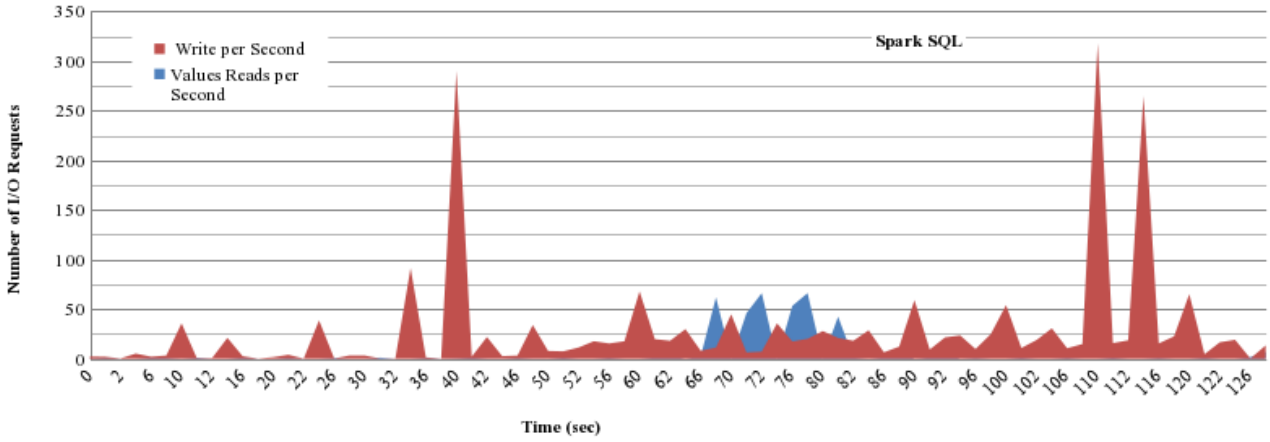


(b) Parquet

Figure 55: CPU: Low util on parquet (60%), short time for phase 1.

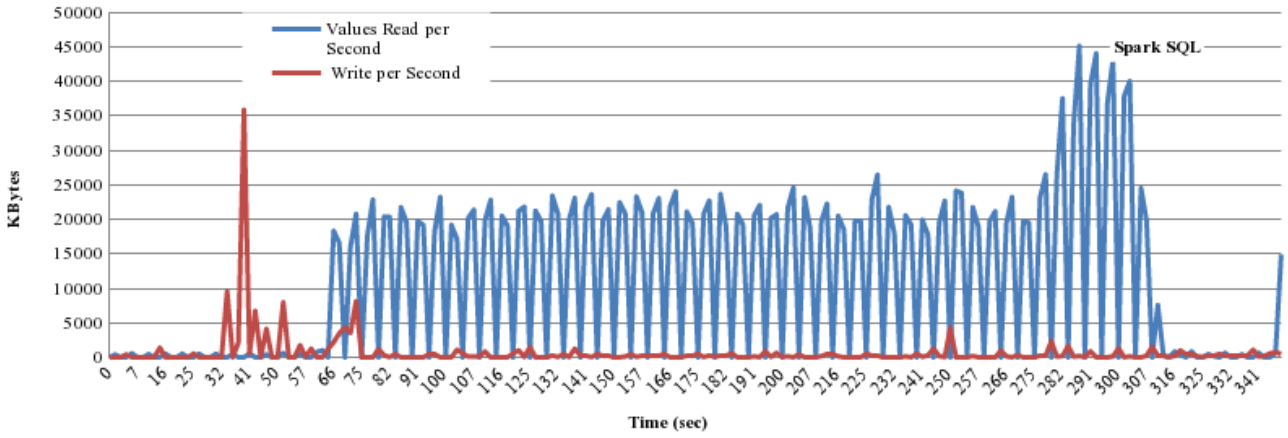


(a) ORC

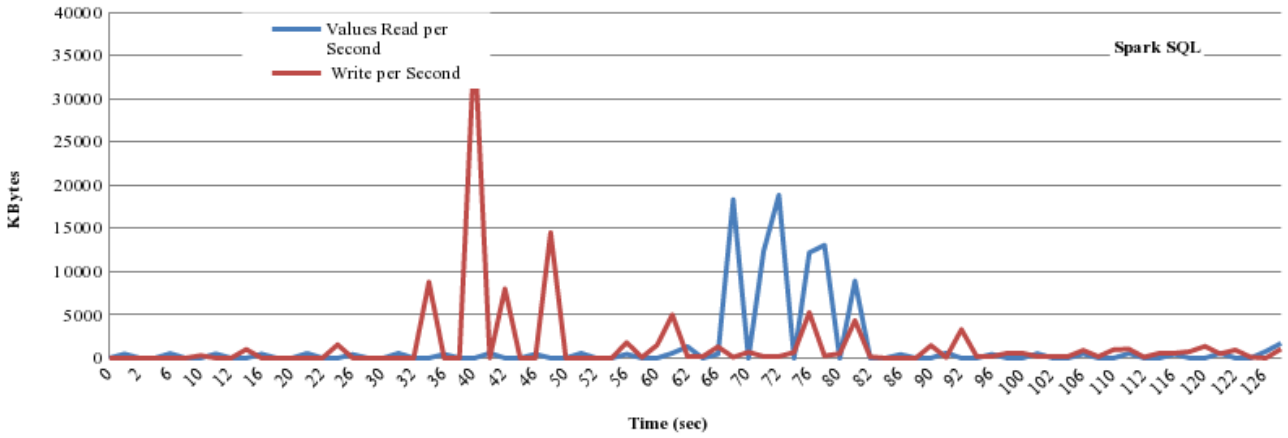


(b) Parquet

Figure 56: Disk Requests: Lower number of requests on Parquet.

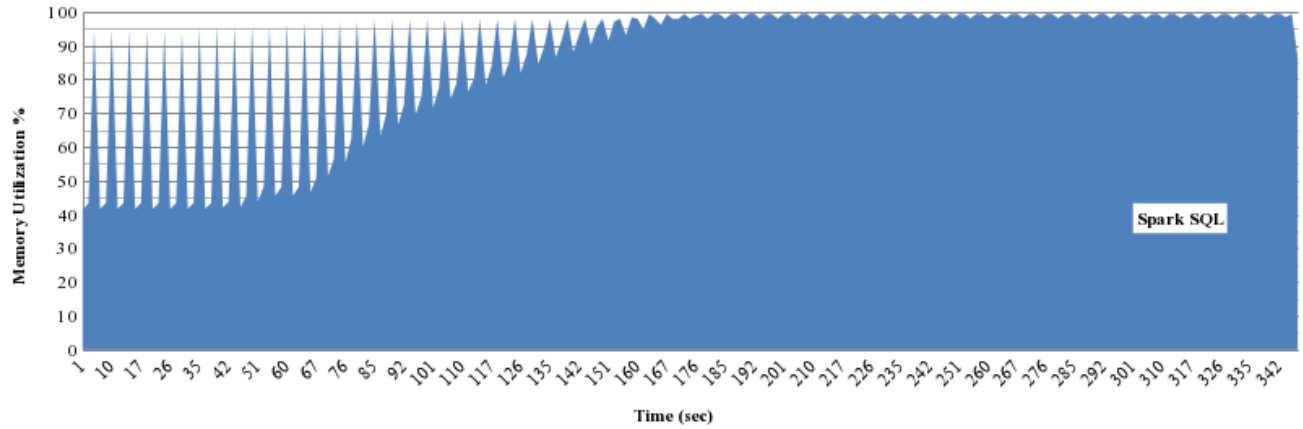


(a) ORC

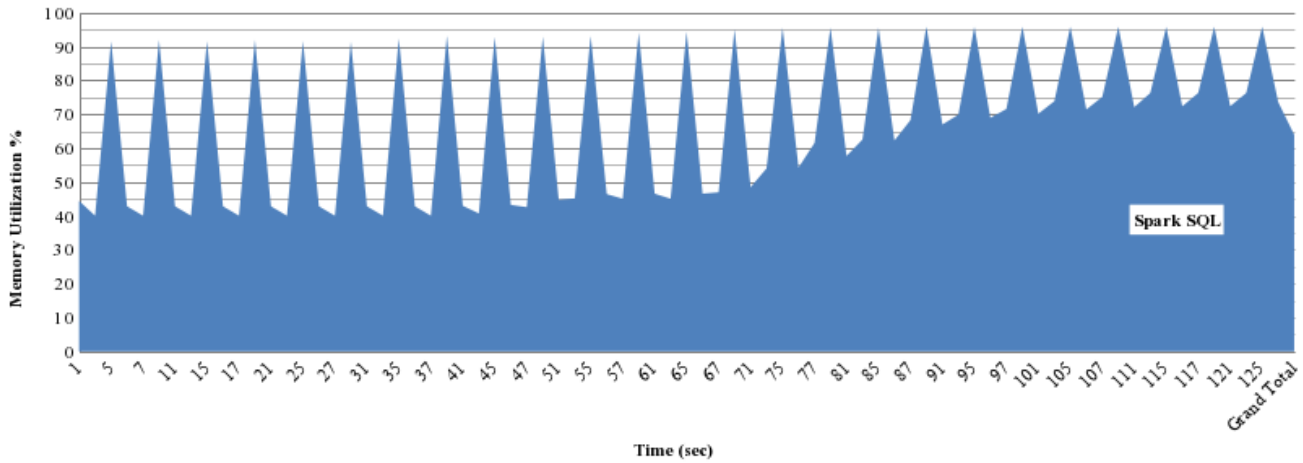


(b) Parquet

Figure 57: Disk Bandwidth: higher BW utilization on Parquet.

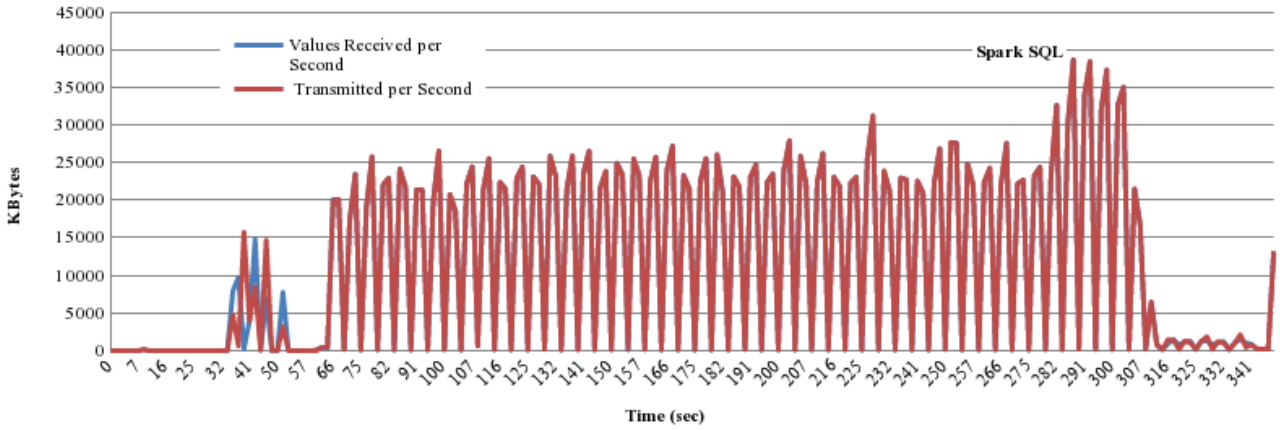


(a) ORC

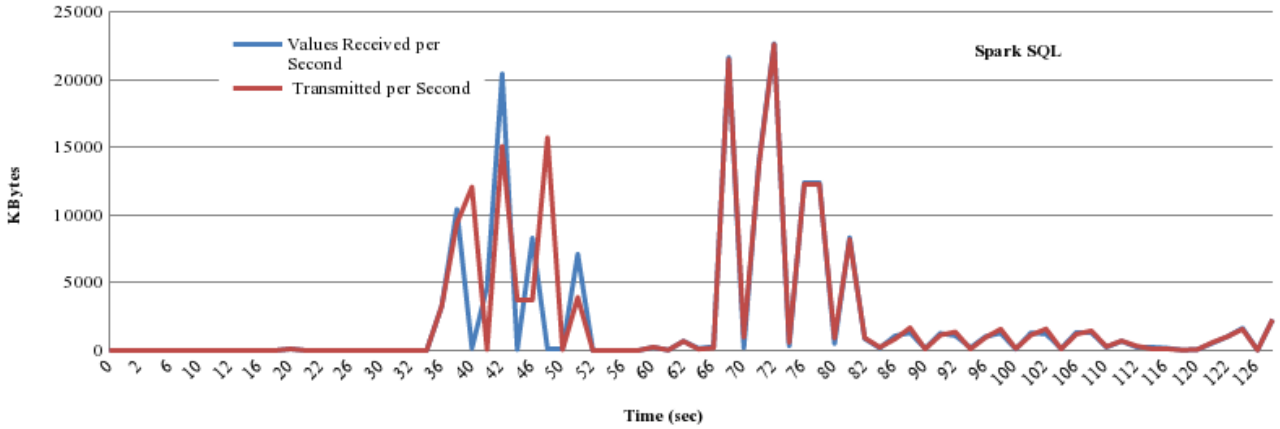


(b) Parquet

Figure 58: Free Memory.

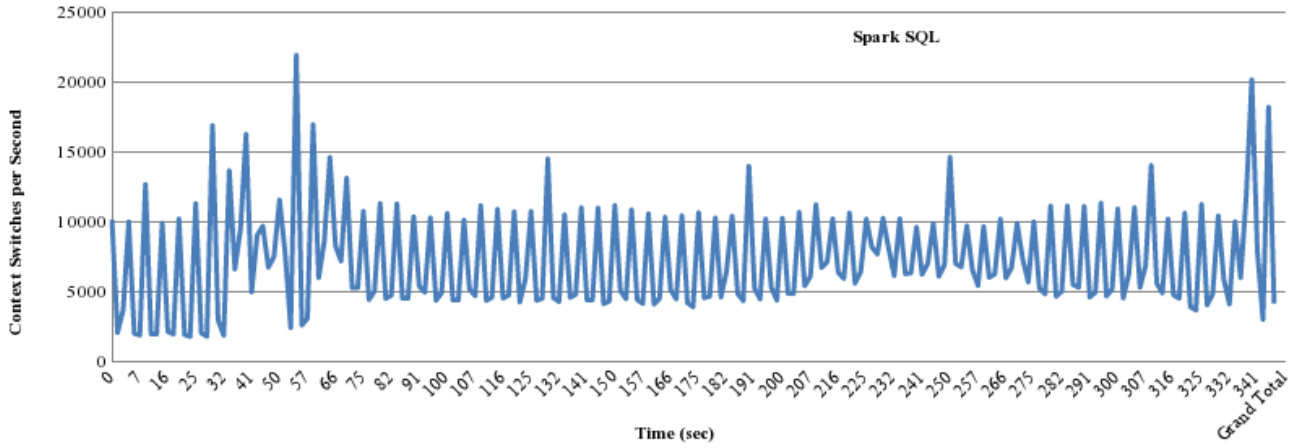


(a) ORC

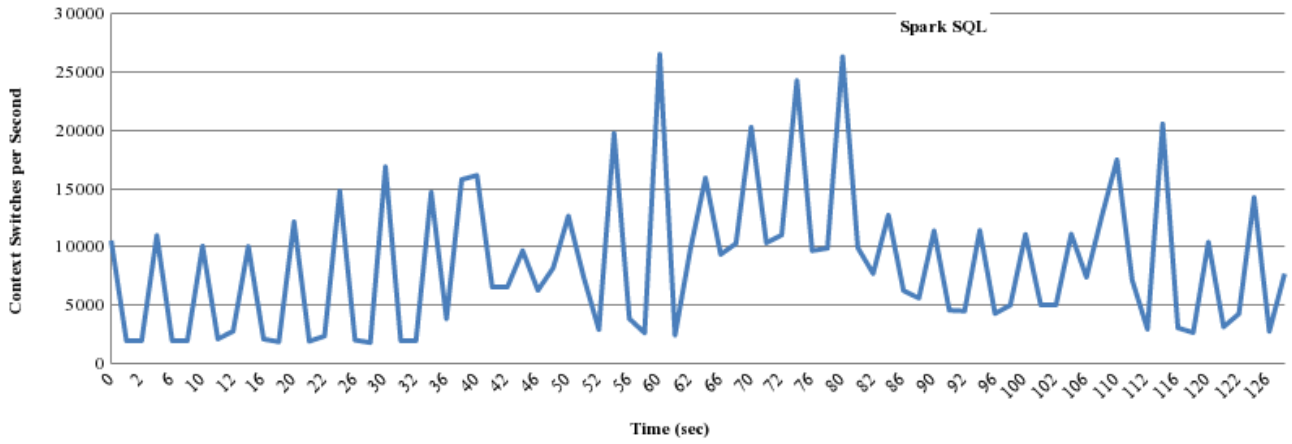


(b) Parquet

Figure 59: Network Utilization: more or less corresponding to DiskIO BW.



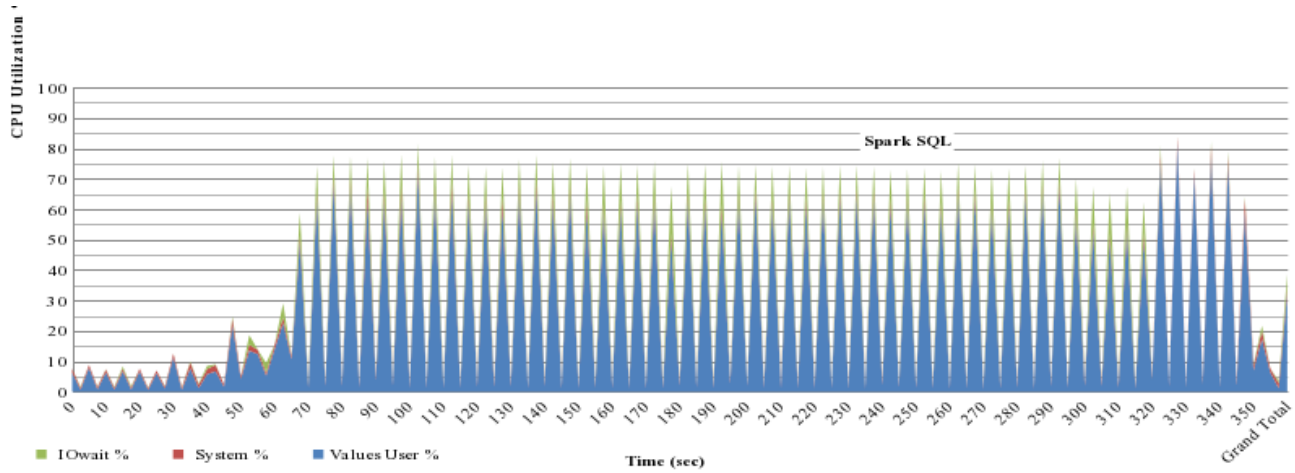
(a) ORC



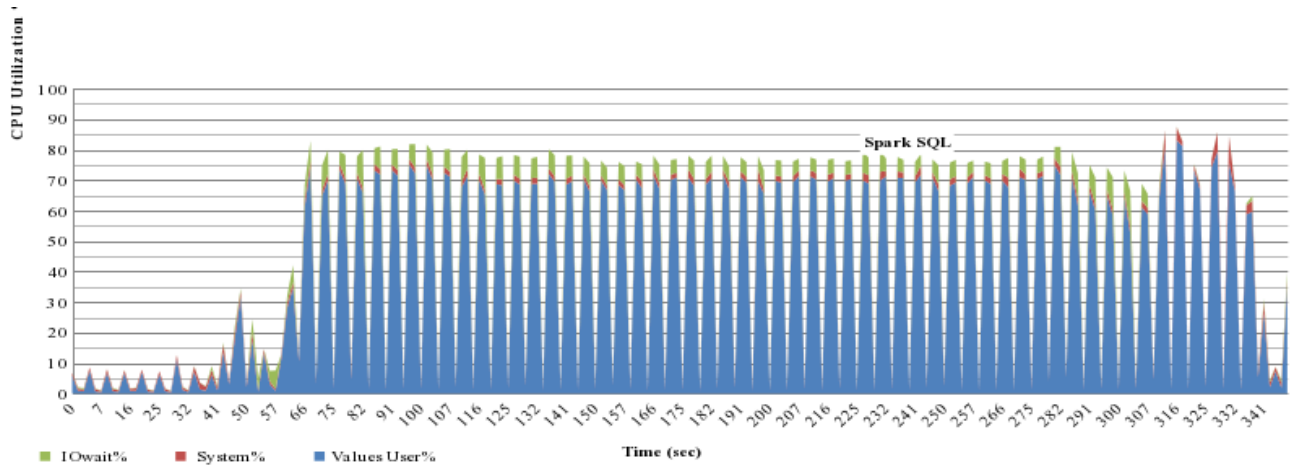
(b) Parquet

Figure 60: VMSTAT Context Switches

3.9 ORC - NoCompression vs. Snappy

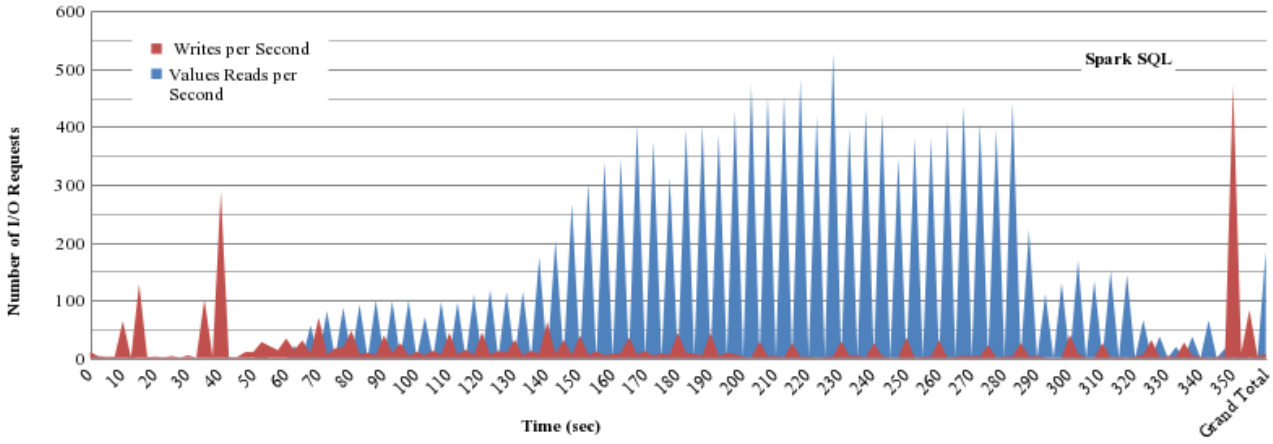


(a) NoCompression

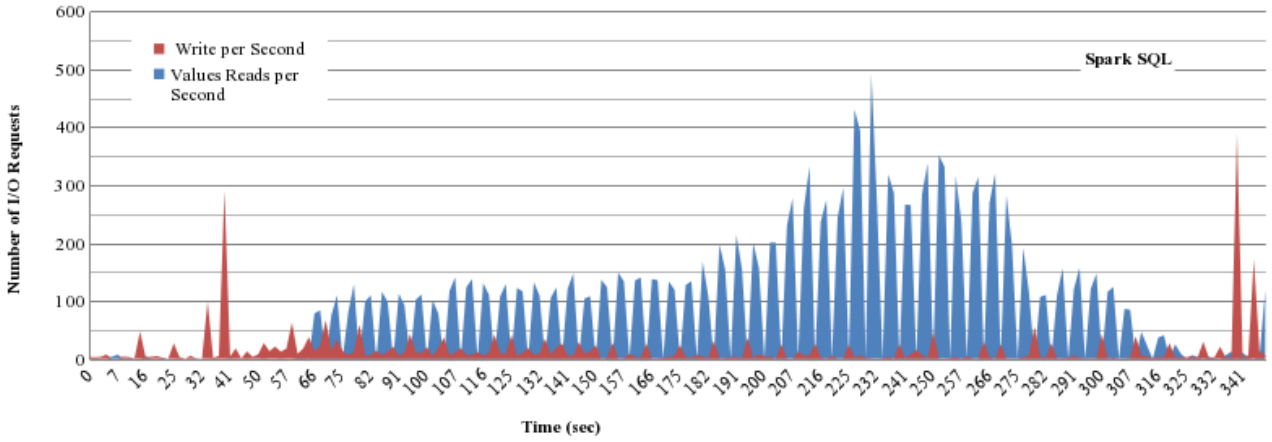


(b) Snappy

Figure 61: CPU.

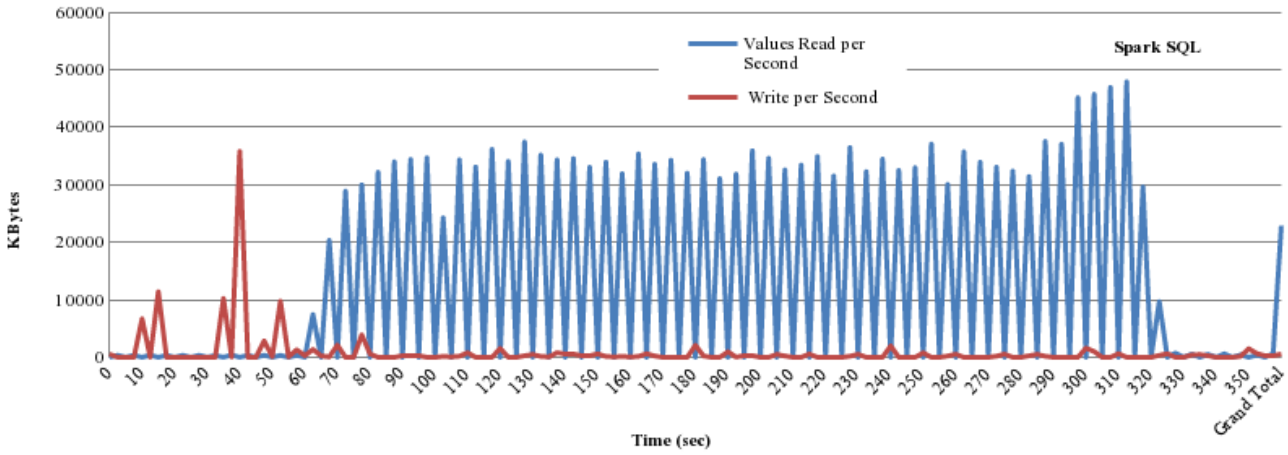


(a) NoCompress

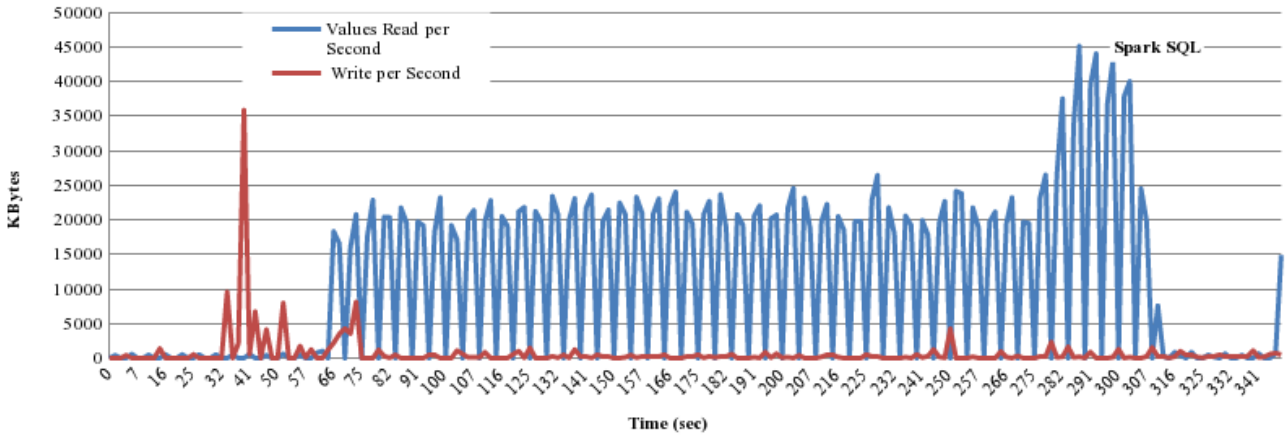


(b) Snappy

Figure 62: Disk Requests.

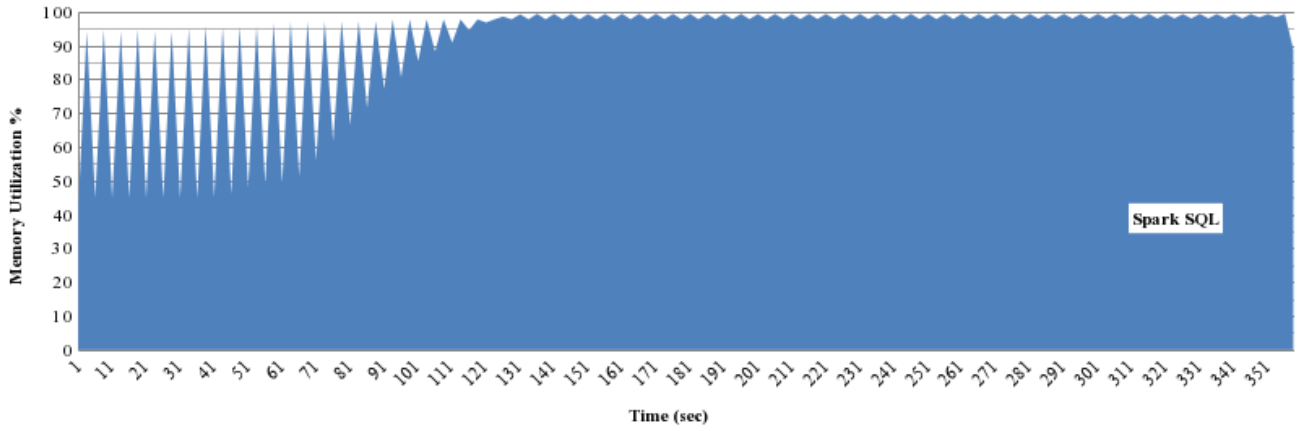


(a) NoCompress

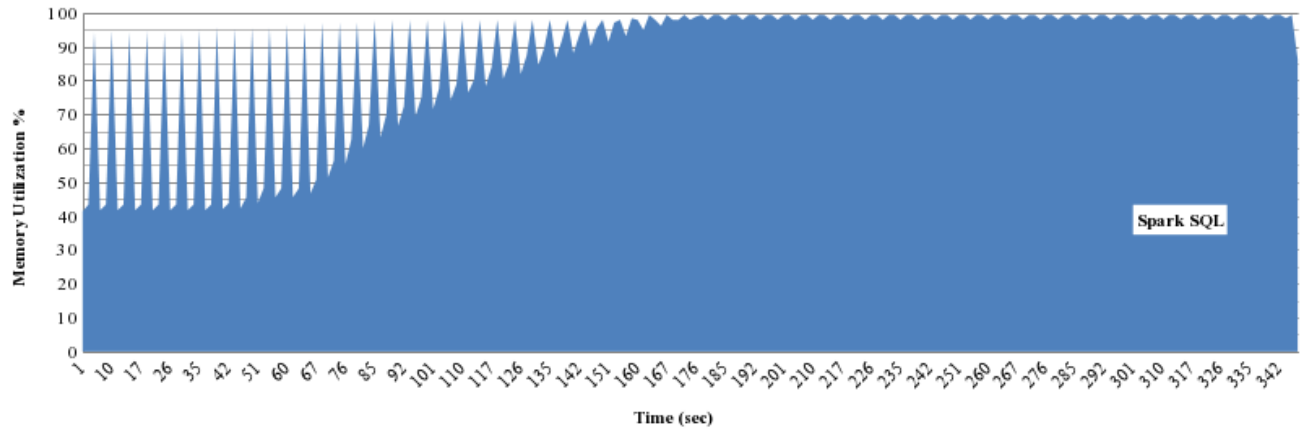


(b) Snappy

Figure 63: Disk Bandwidth.

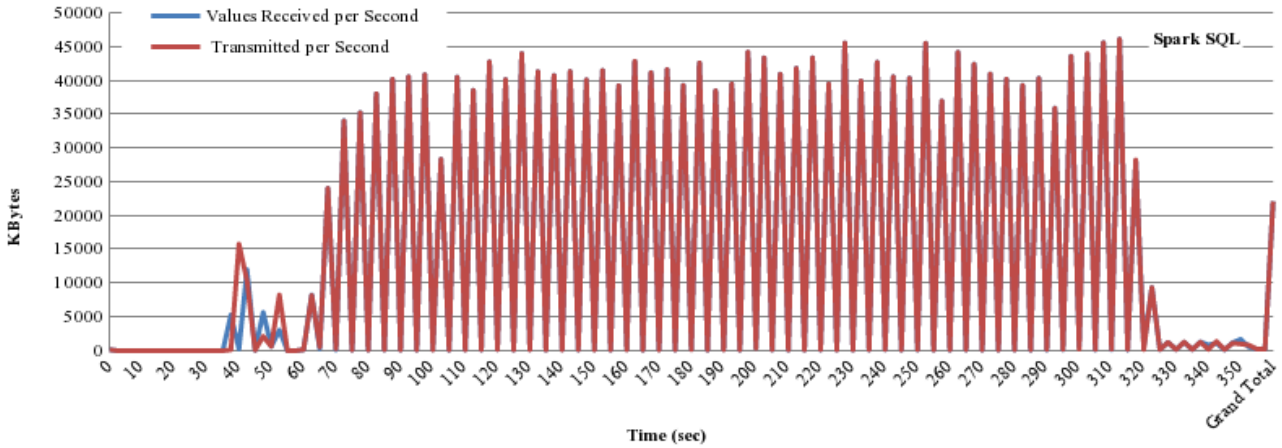


(a) NoCompress

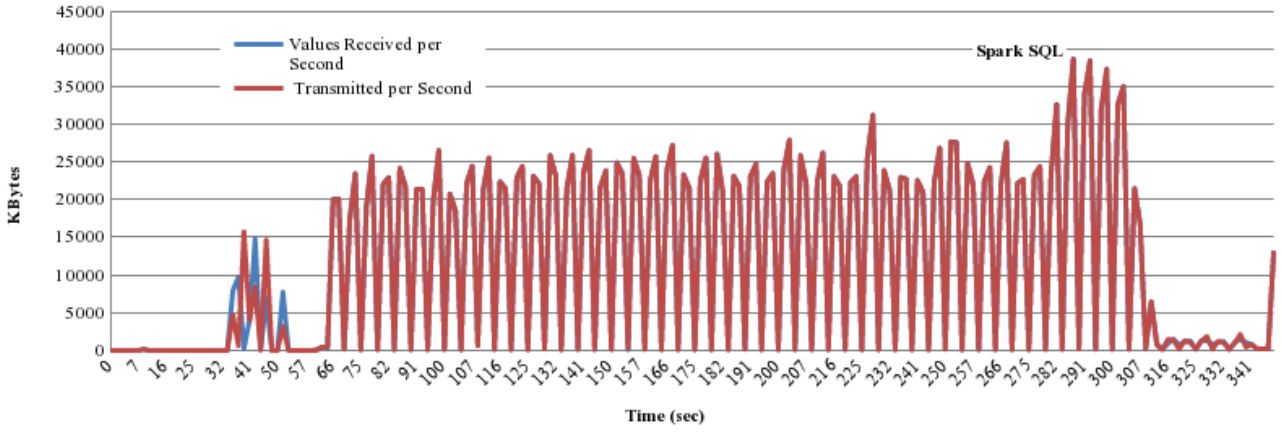


(b) Snappy

Figure 64: Free Memory.

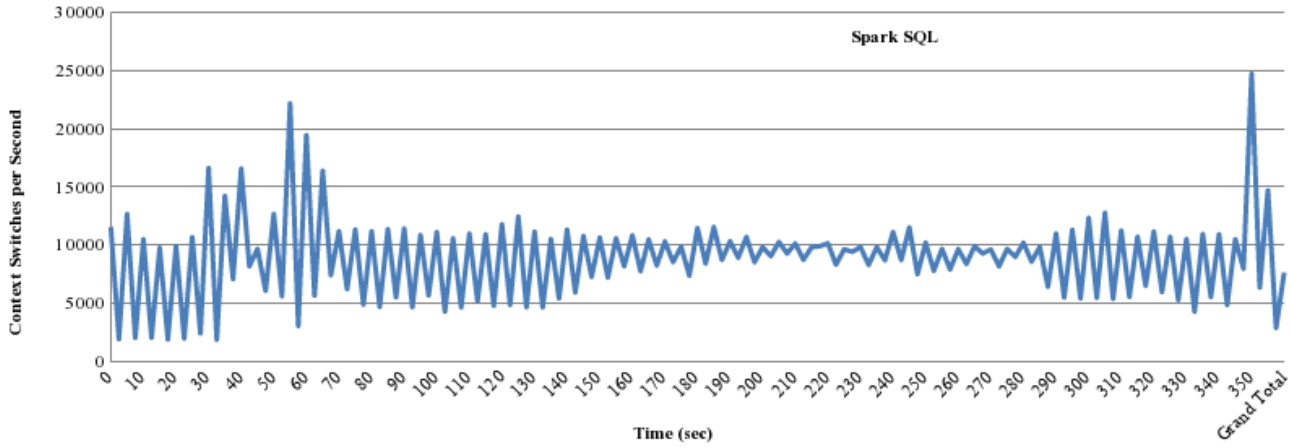


(a) NoCompress

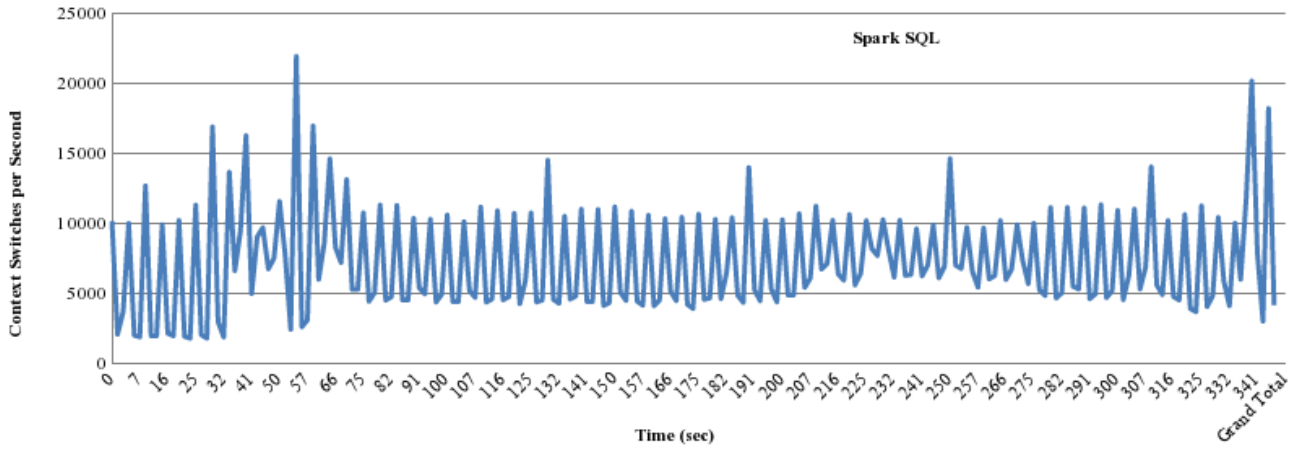


(b) Snappy

Figure 65: Network Utilization.



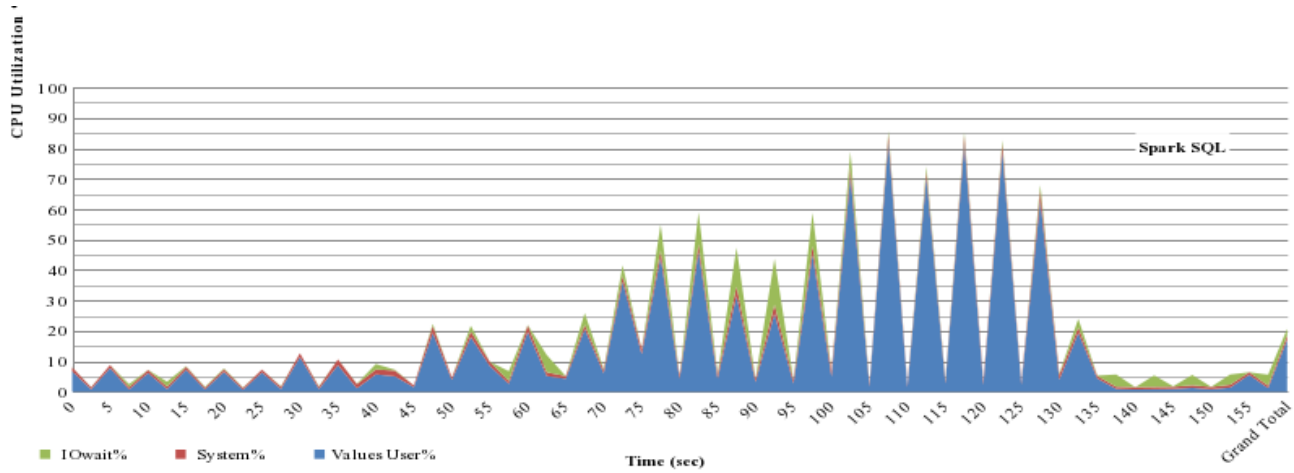
(a) NoCompress



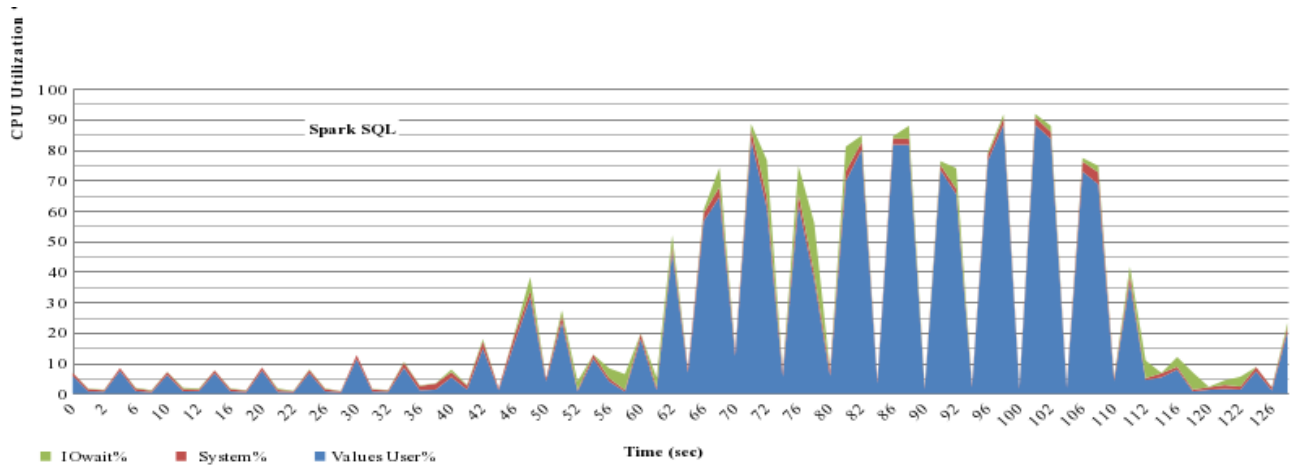
(b) Snappy

Figure 66: VMSTAT Context Switches

3.10 Parquet - NoCompression vs Snappy

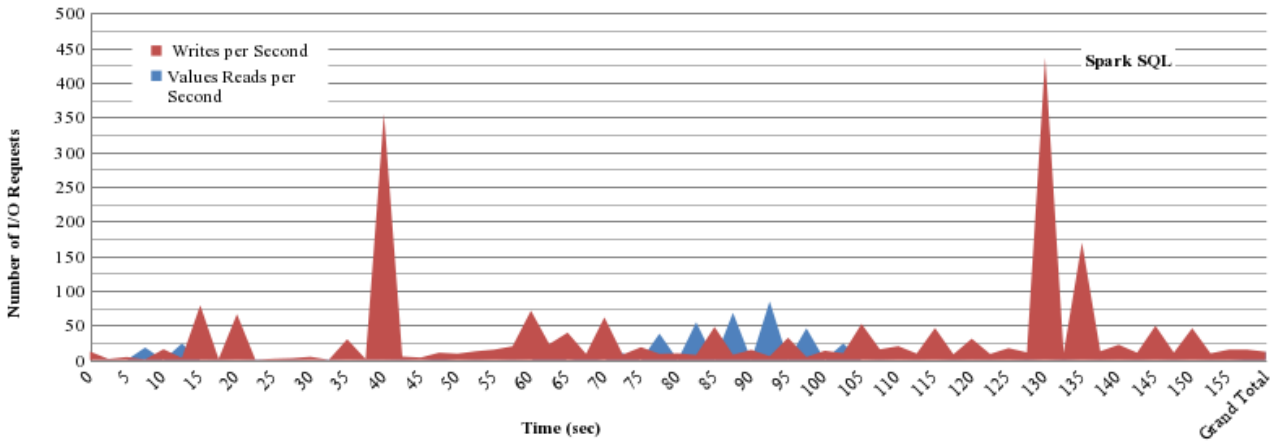


(a) NoCompression

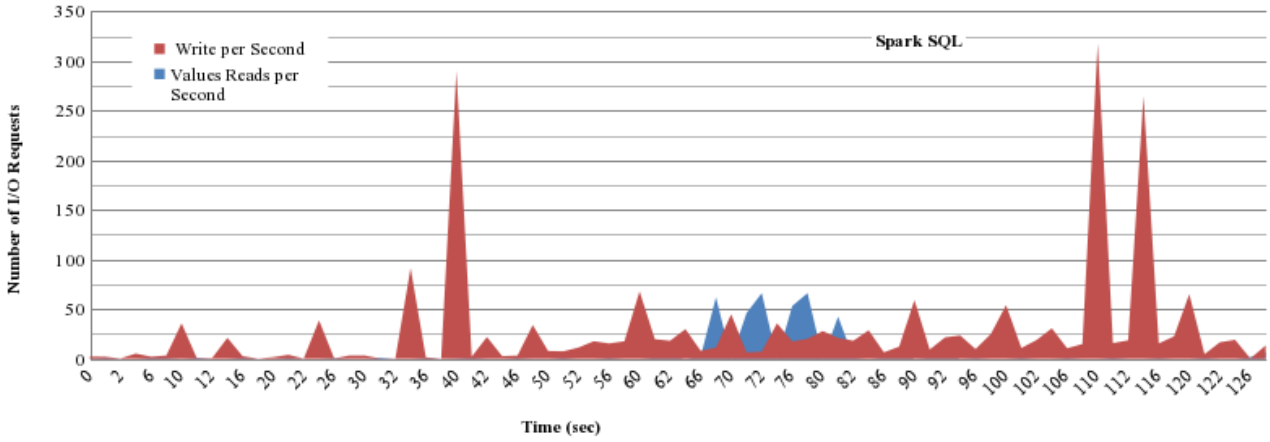


(b) Snappy

Figure 67: CPU.

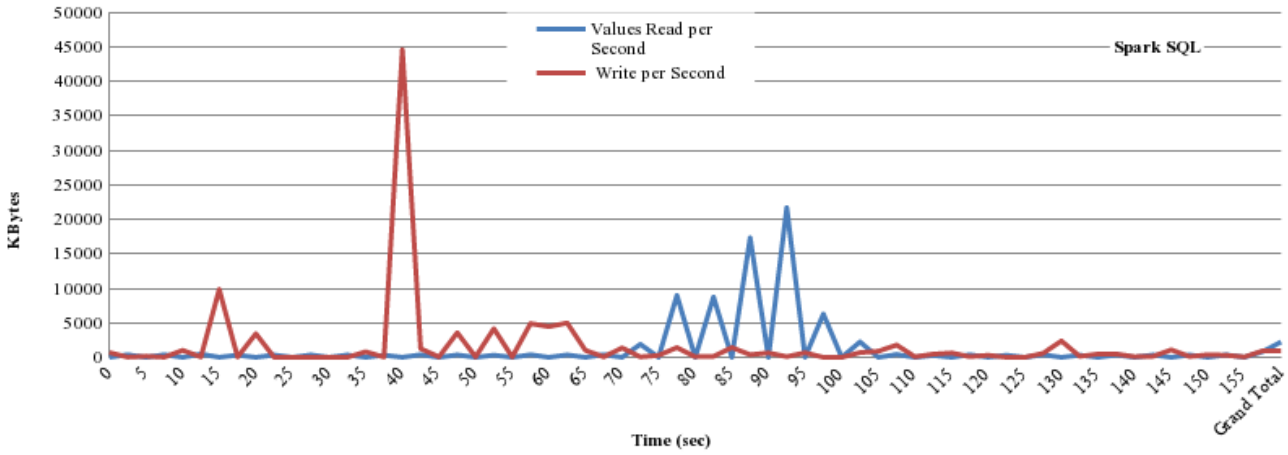


(a) NoCompression

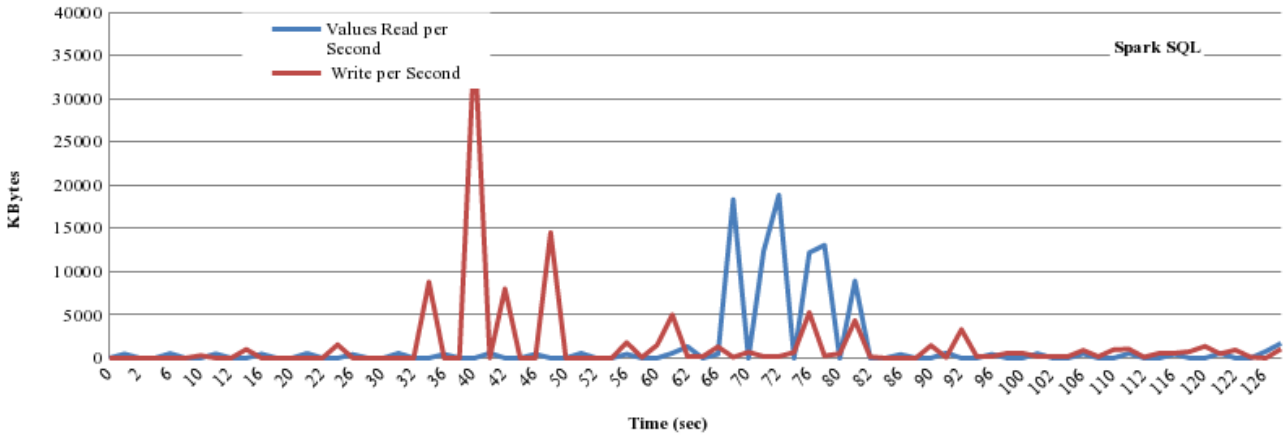


(b) Snappy

Figure 68: Disk Requests.

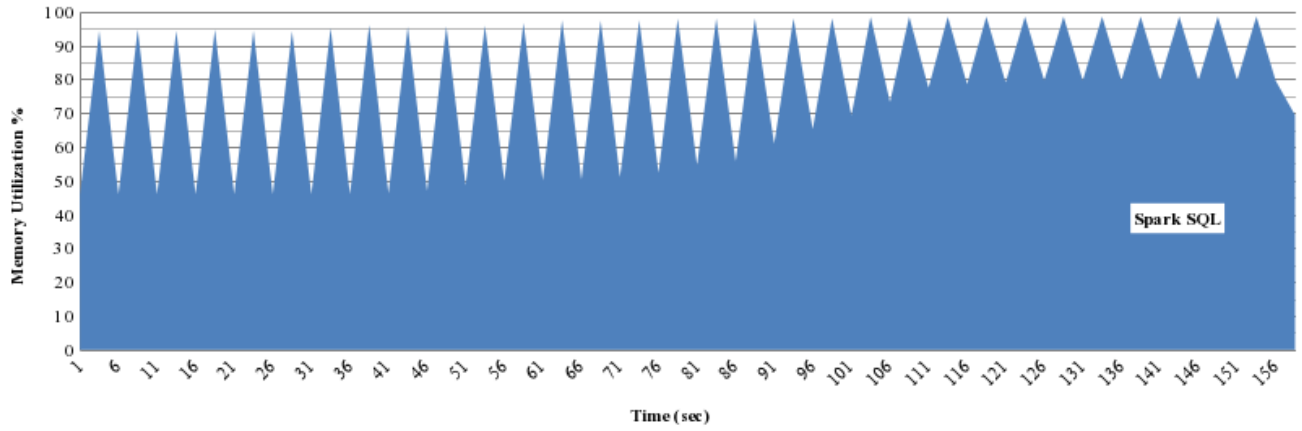


(a) NoCompression

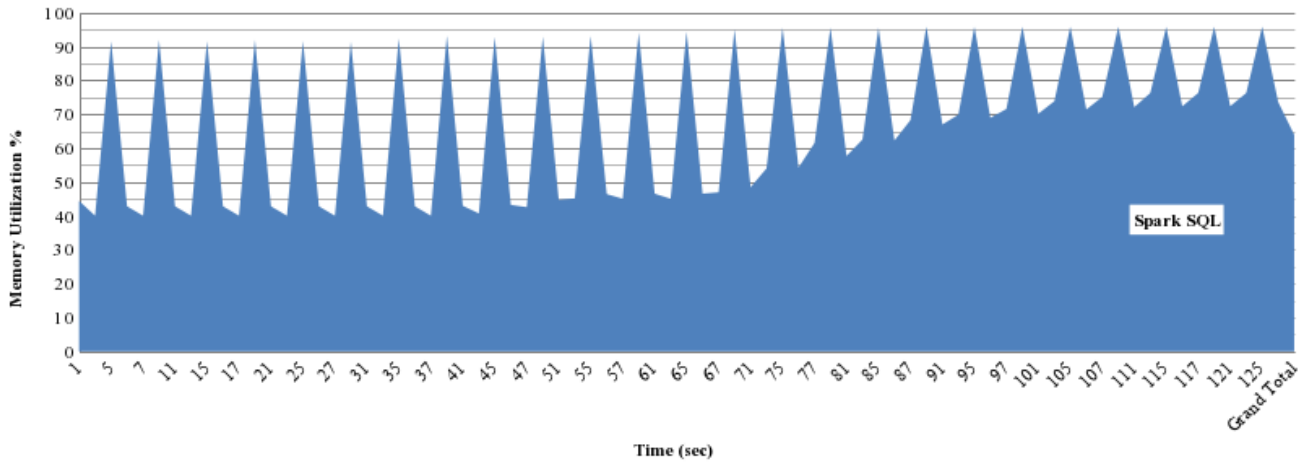


(b) Snappy

Figure 69: Disk Bandwidth.

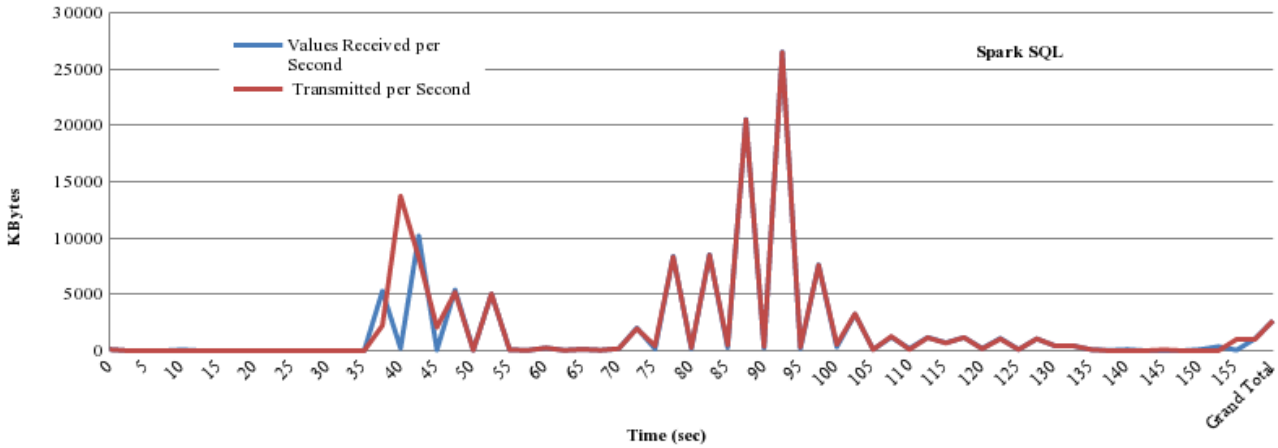


(a) NoCompression

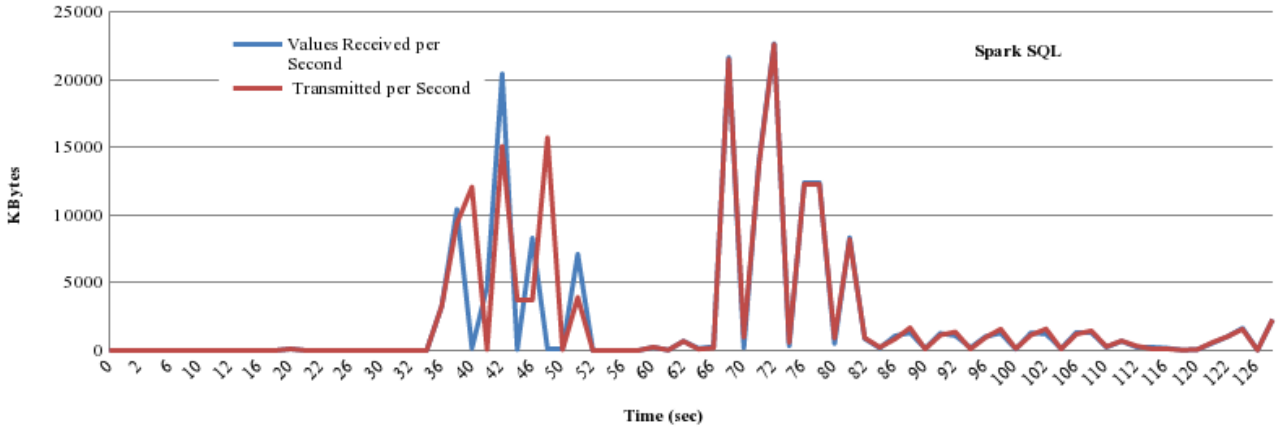


(b) Snappy

Figure 70: Free Memory.

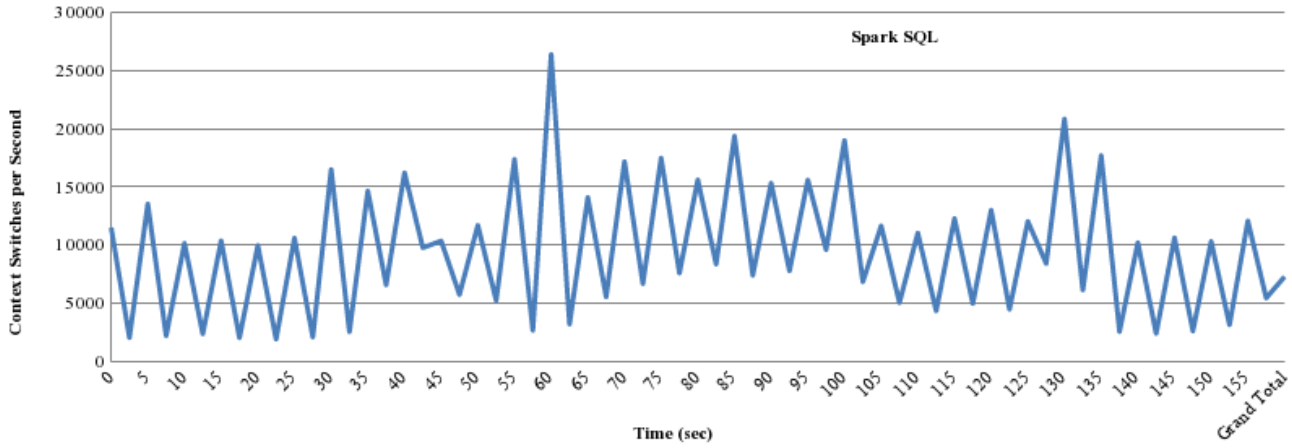


(a) NoCompression

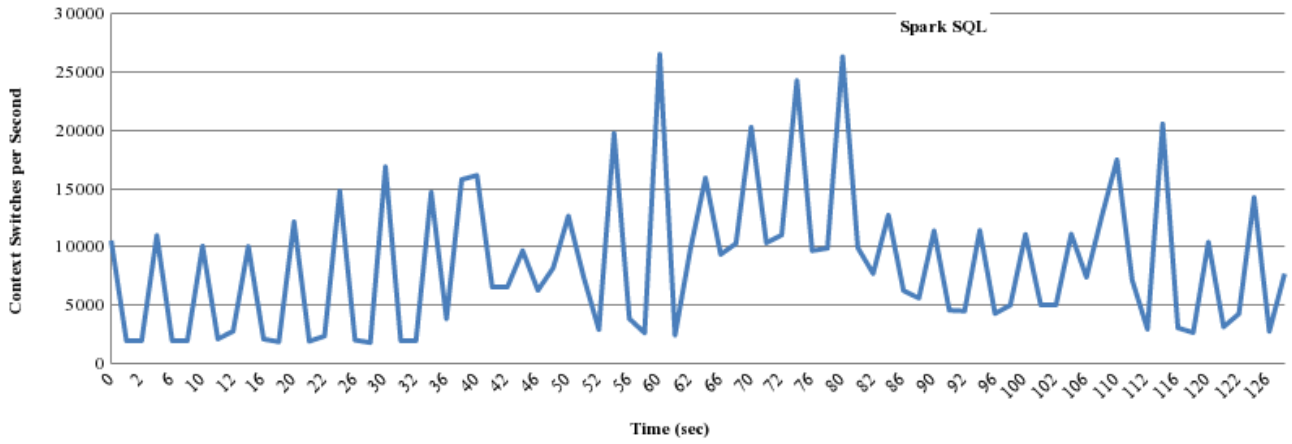


(b) Snappy

Figure 71: Network Utilization.



(a) NoCompression



(b) Snappy

Figure 72: VMSTAT Context Switches

4 Q25

4.1 Type

HiveQL/Spark MLlib

4.2 Description

Customer segmentation analysis: Customers are separated along the following key shopping dimensions: recency of last visit, frequency of visits and monetary amount. Use the store and online purchase data during a given year to compute. After model of separation is build, report for the analysed customers to which "group" they where assigned

4.3 Performance

	Default (sec.)		No compr. (sec.)		Snappy (sec.)	
Query	**ORC**	**Parquet**	**ORC**	**Parquet**	**ORC**	**Parquet**
Q25	462	359	453	363	440	346

4.4 Phase 1

Selects desired customers from store and web sales and puts them in a TEMP table.

4.5 Phase 2

Users are grouped in a second TEMP table with the correct format value for the ML algorithm.

4.6 Phase 3

The KMeansClustering Spark class runs with the aforementioned table as input. The final result is written in a new table.

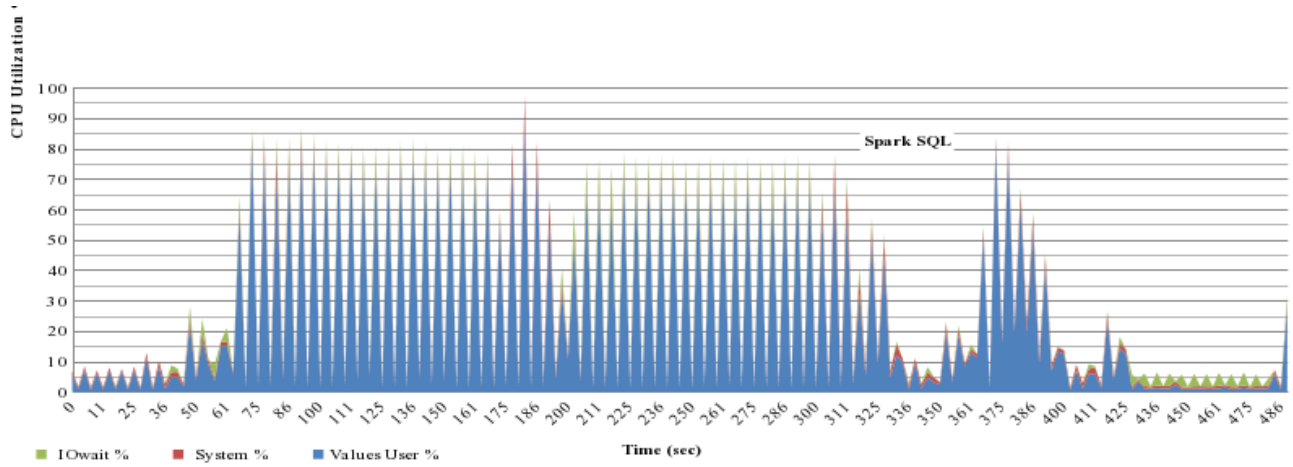
4.7 Source

GitHub: https://git.io/vpFve

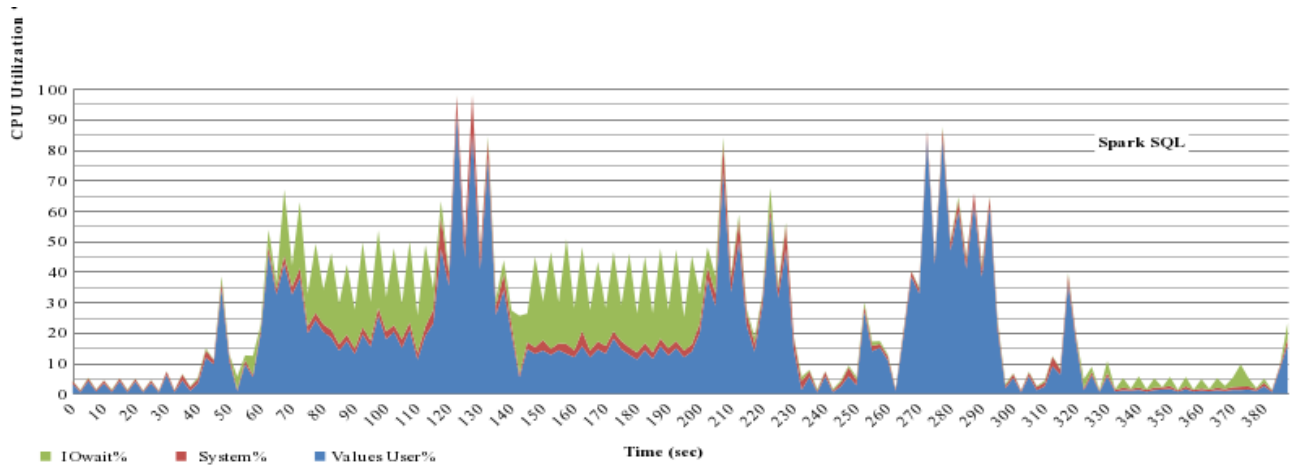
4.8 Notes

...

4.9 No compression - ORC vs Parquet

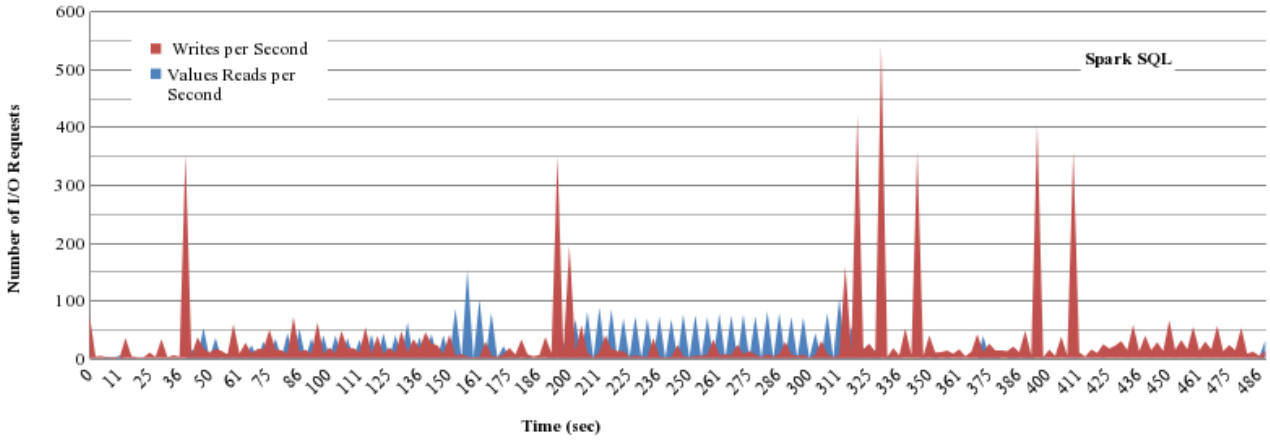


(a) ORC

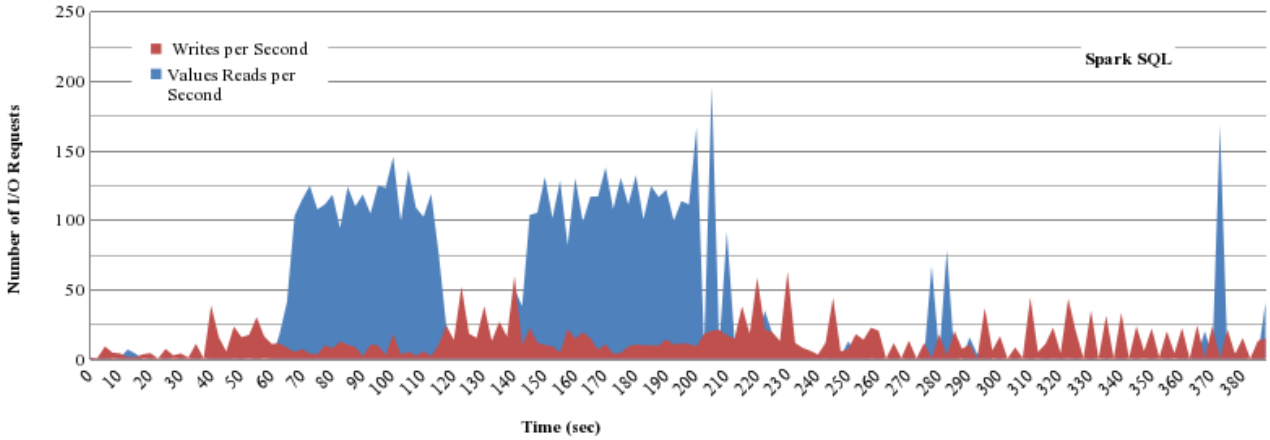


(b) Parquet

Figure 73: CPU: Lower utilization on Parquet (max is 80%)

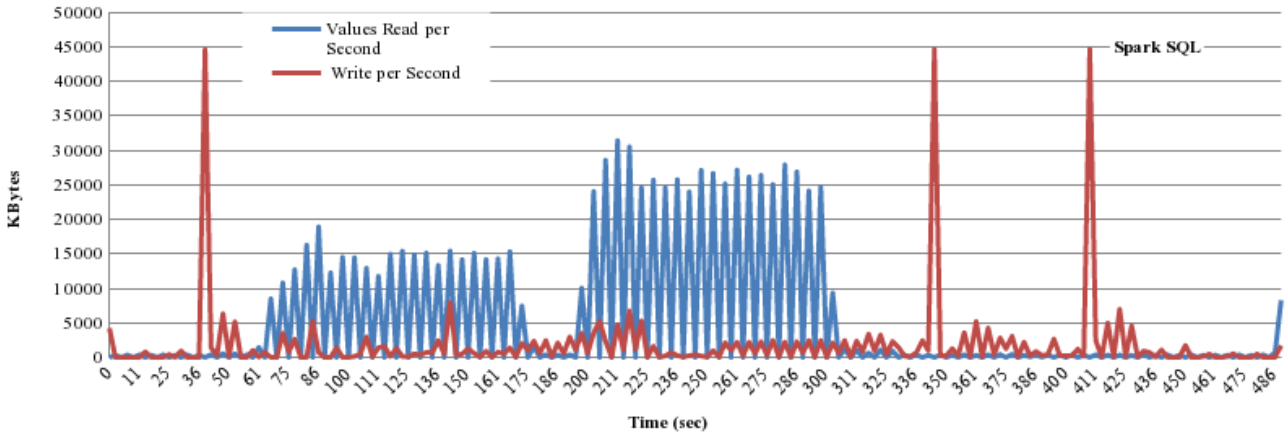


(a) ORC

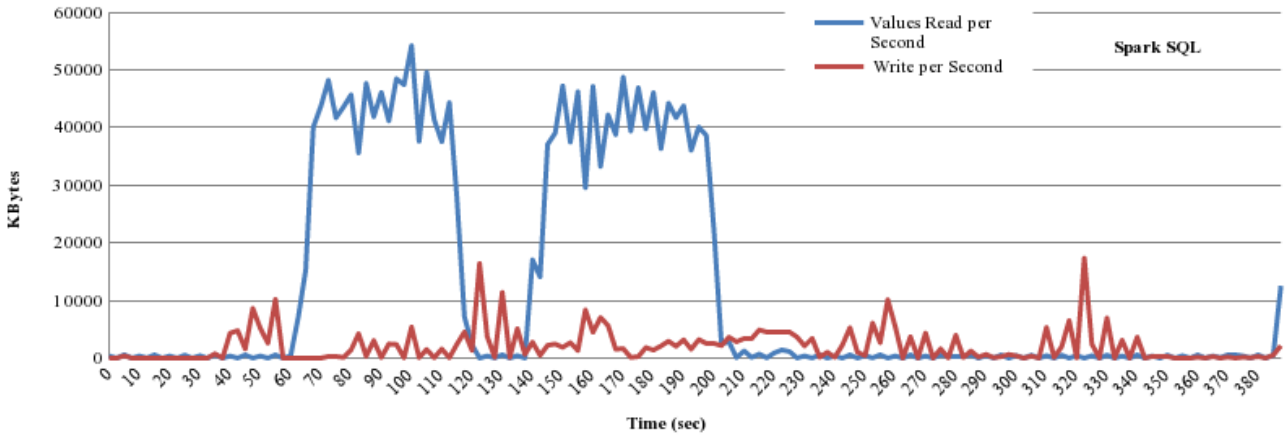


(b) Parquet

Figure 74: Disk Requests: More requests on Parquet.

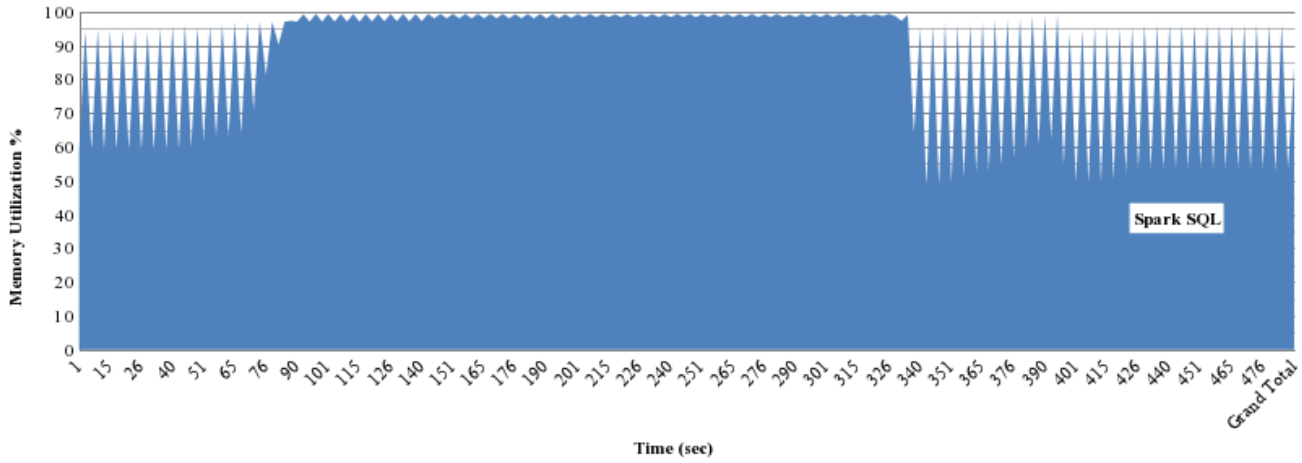


(a) ORC

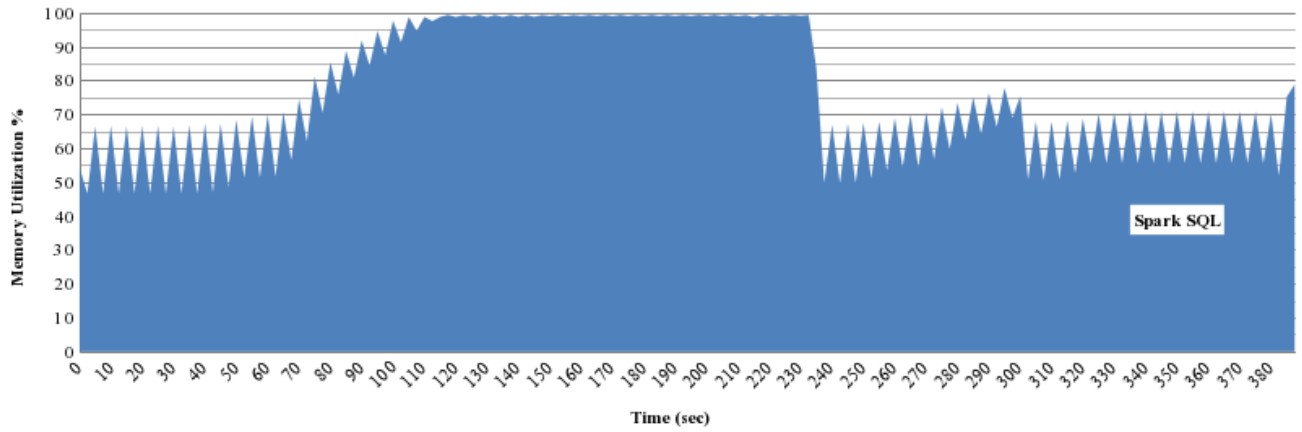


(b) Parquet

Figure 75: Disk Bandwidth: bandwidth utilization on Parquet.

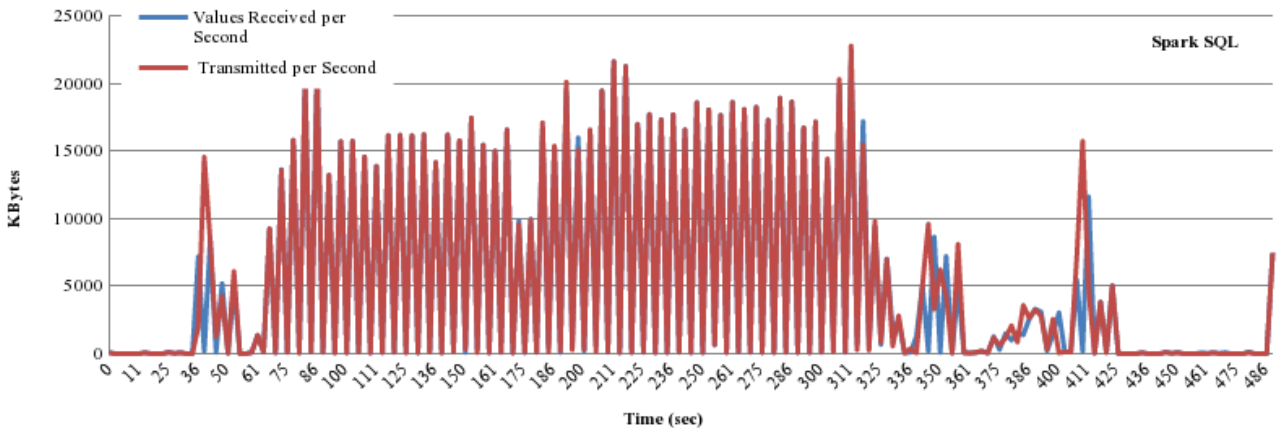


(a) ORC

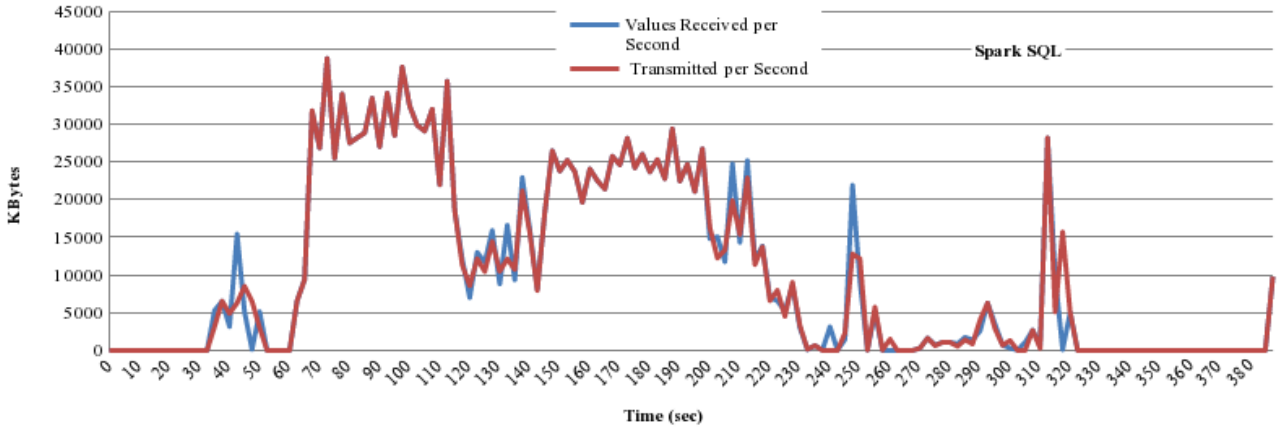


(b) Parquet

Figure 76: Free Memory.

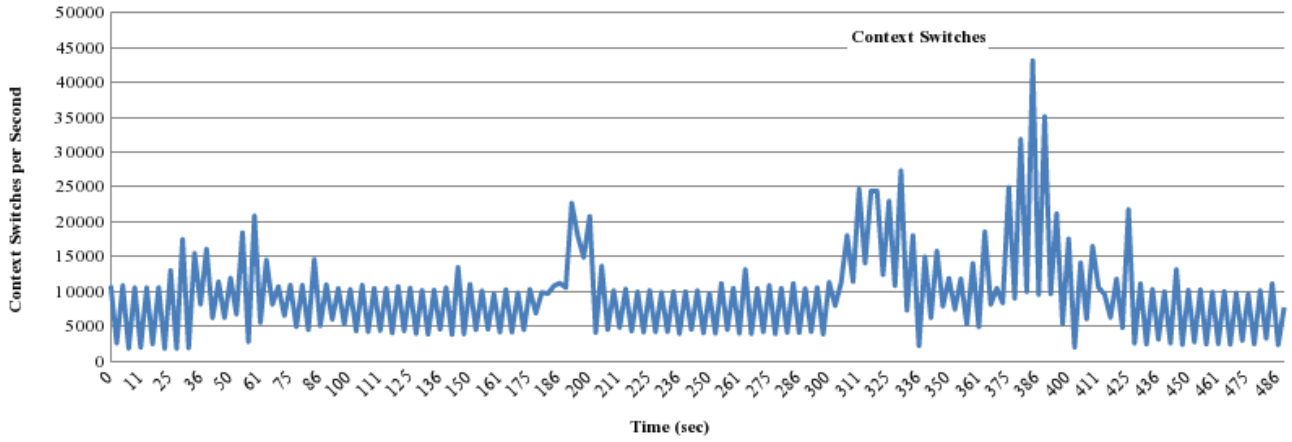


(a) ORC

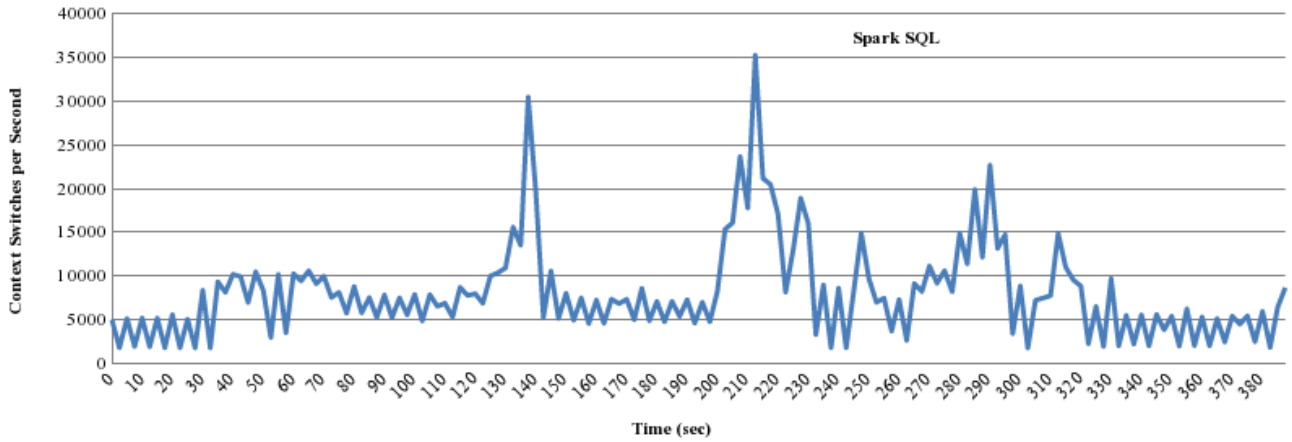


(b) Parquet

Figure 77: Network Utilization: More or less corresponding to DiskIO.



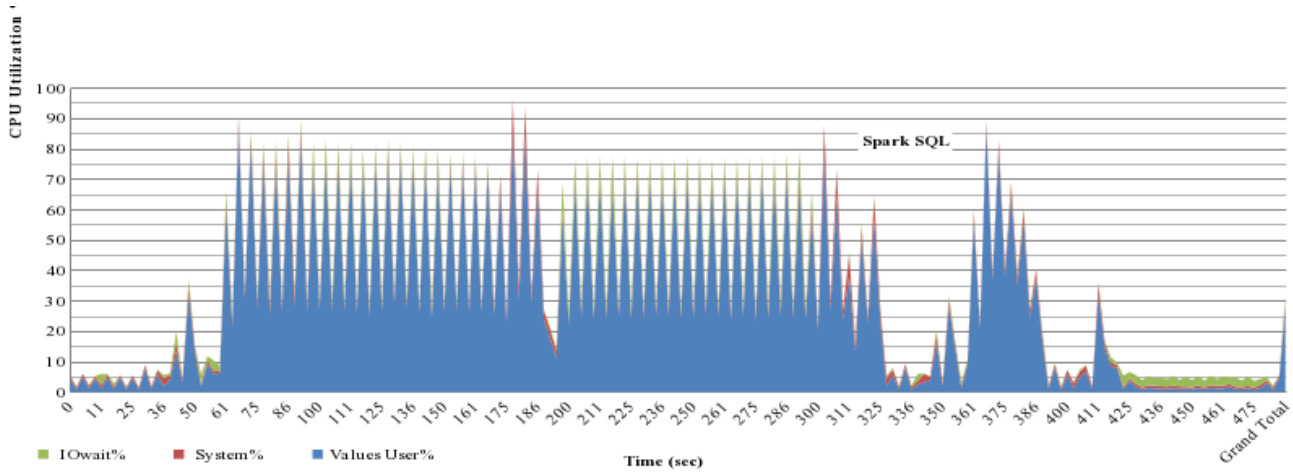
(a) ORC



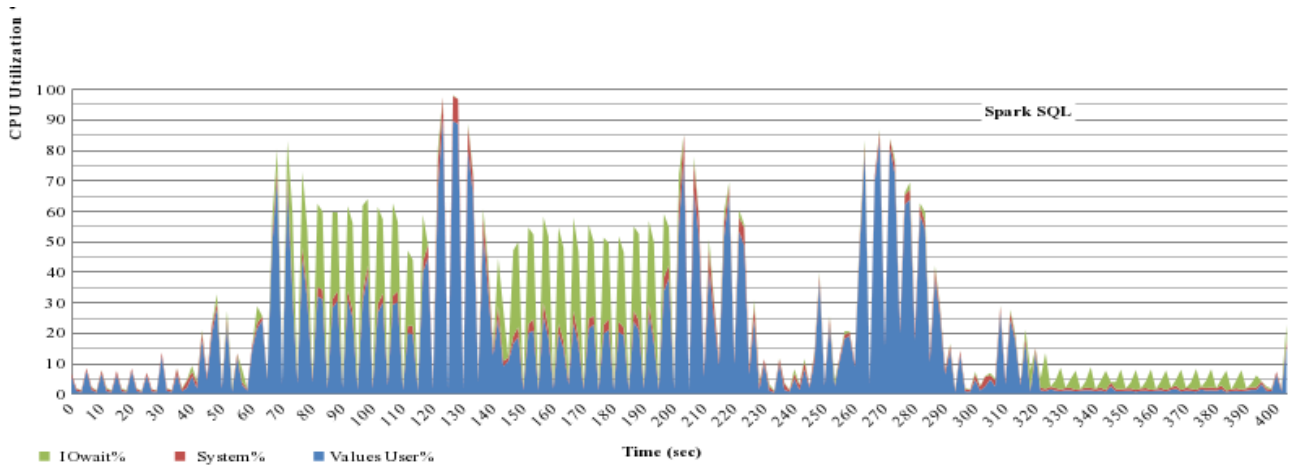
(b) Parquet

Figure 78: VMSTAT Context Switches

4.10 Snappy - ORC vs. Parquet

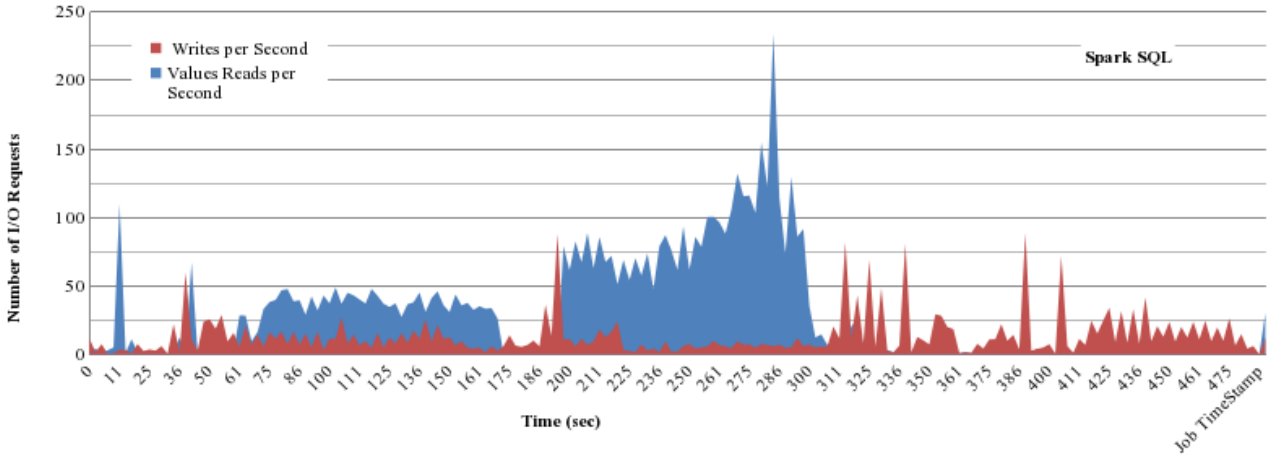


(a) ORC

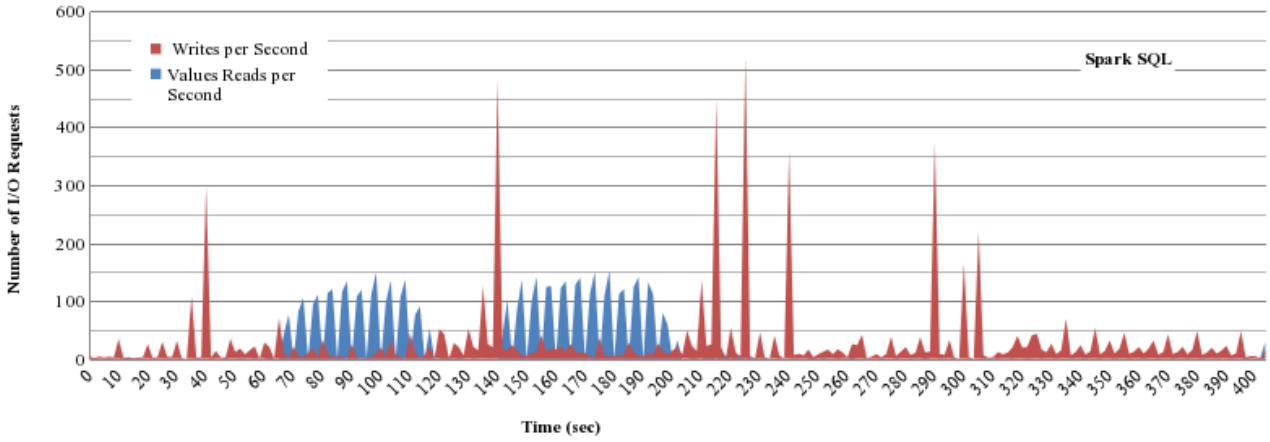


(b) Parquet

Figure 79: CPU: Low util on parquet (60%), short time for phase 1.

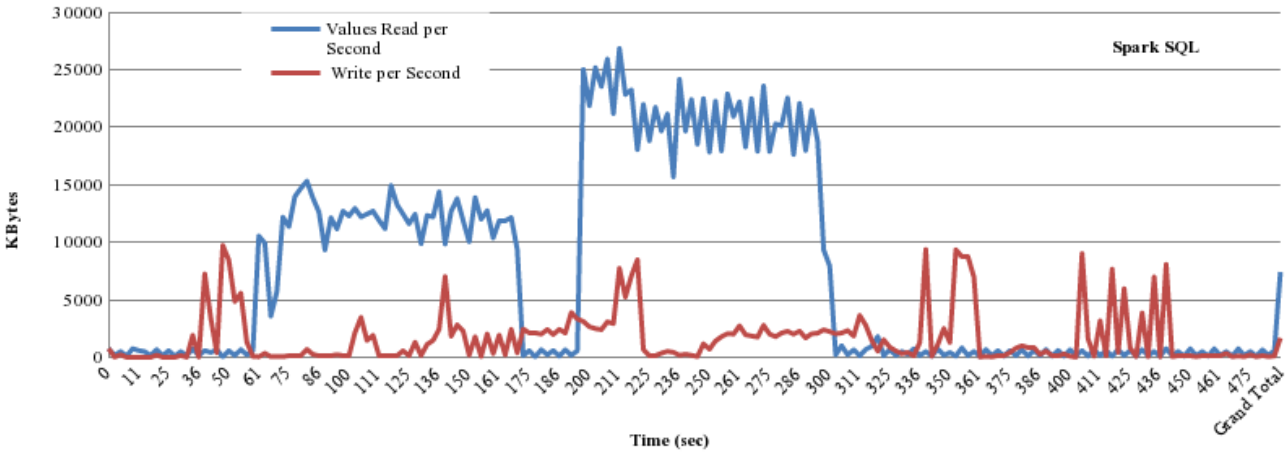


(a) ORC

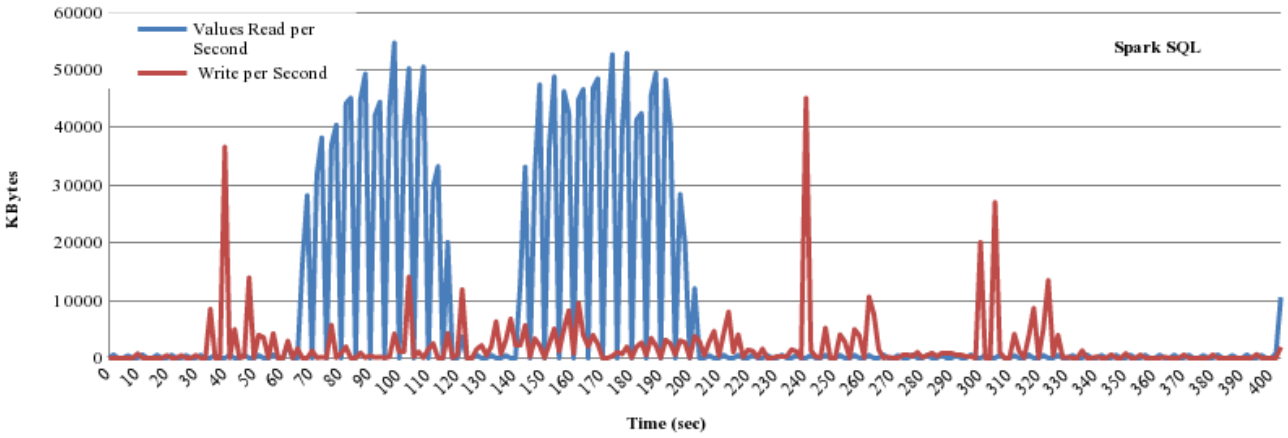


(b) Parquet

Figure 80: Disk Requests: Lower number of requests on Parquet.

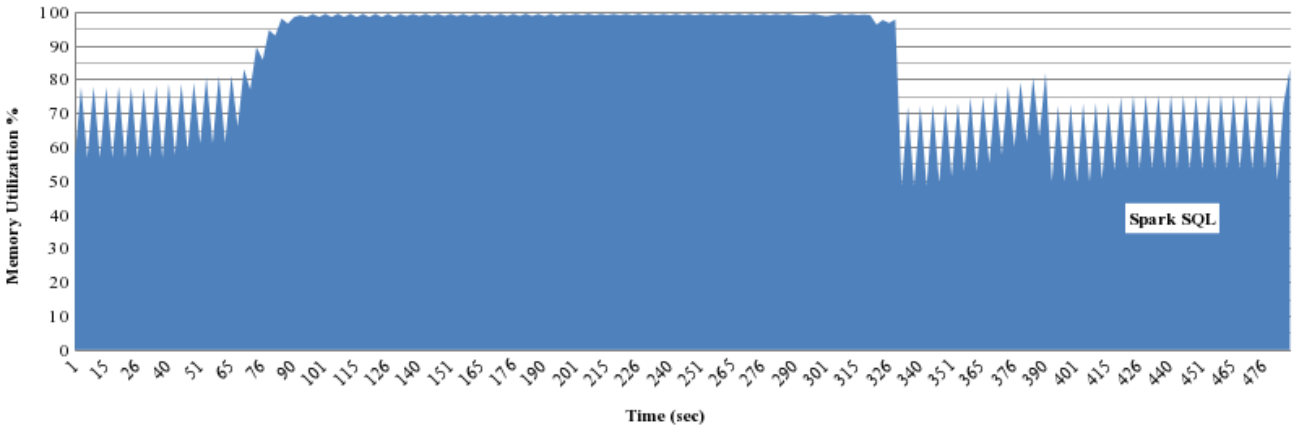


(a) ORC

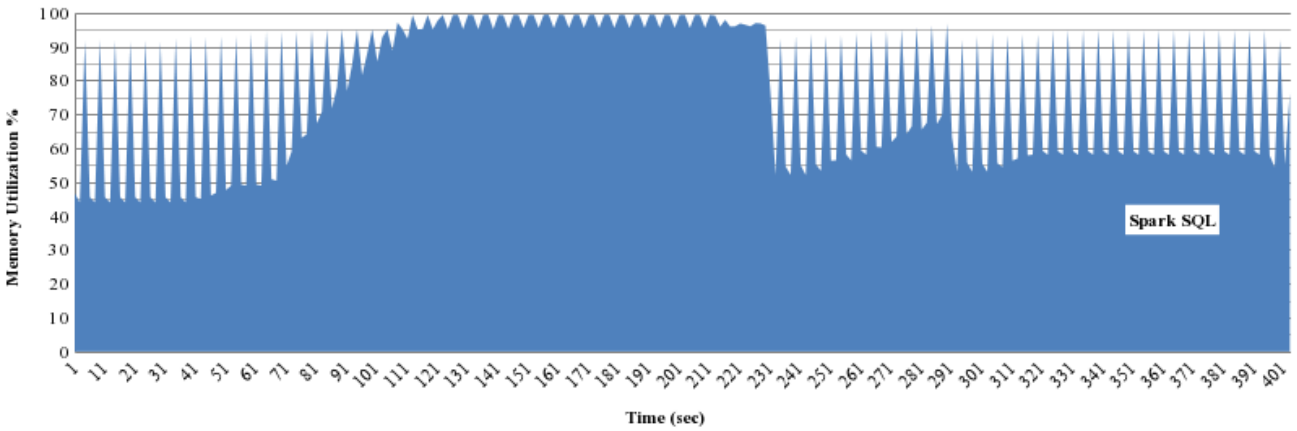


(b) Parquet

Figure 81: Disk Bandwidth: higher BW utilization on Parquet.

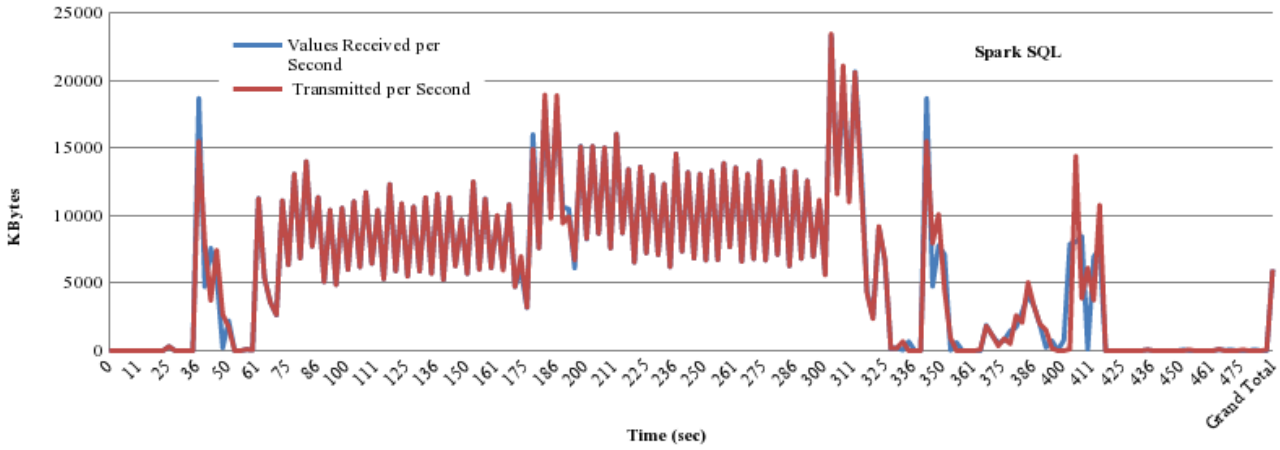


(a) ORC

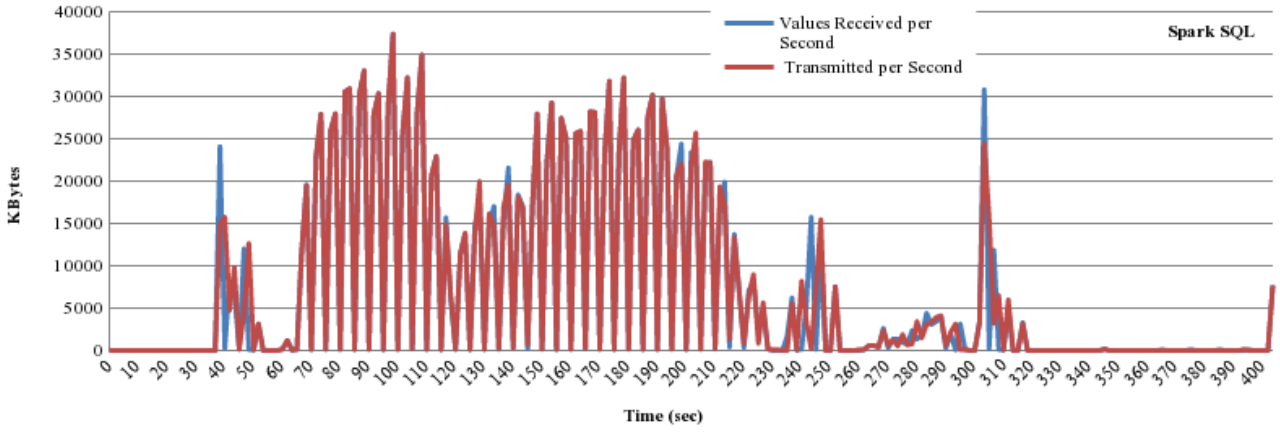


(b) Parquet

Figure 82: Free Memory.

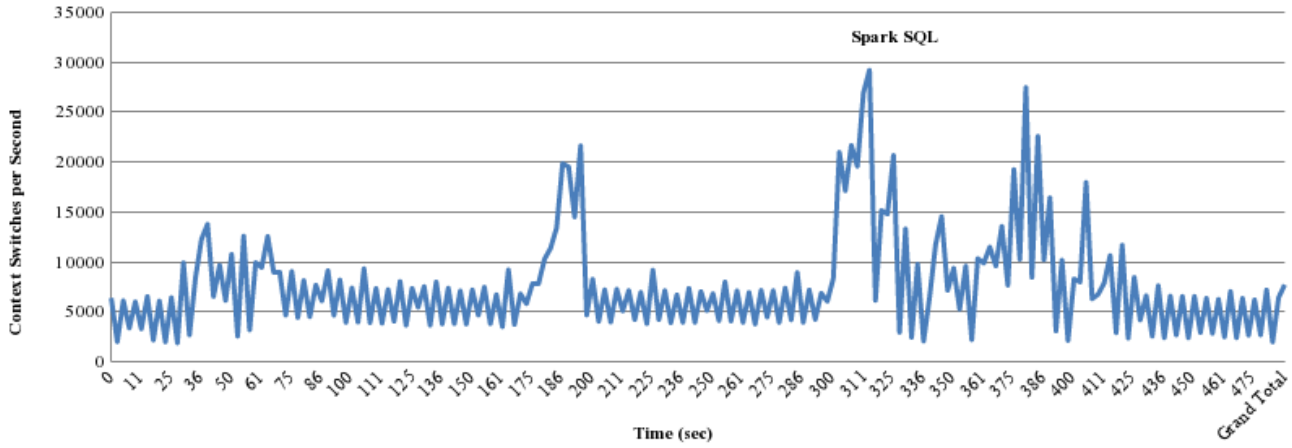


(a) ORC

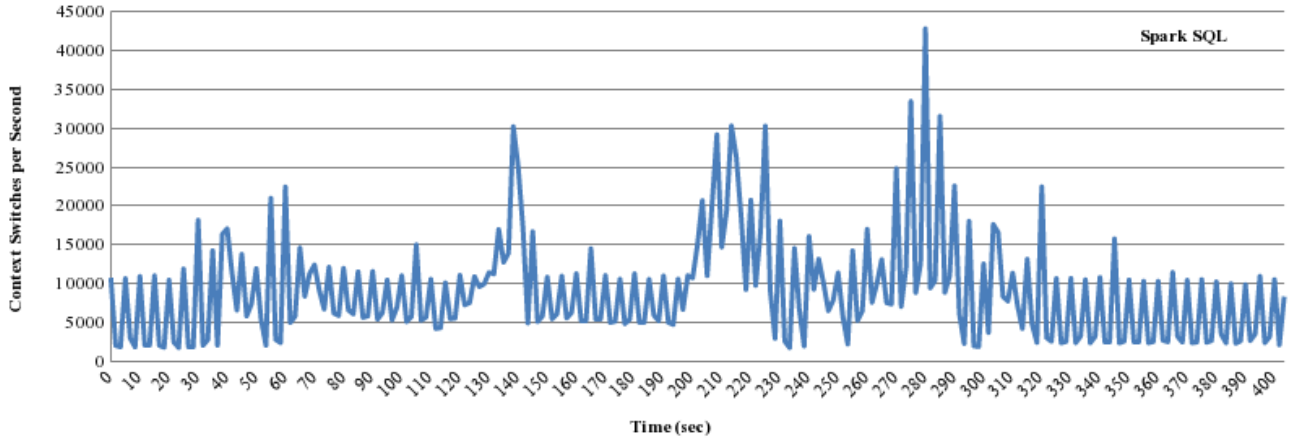


(b) Parquet

Figure 83: Network Utilization: more or less corresponding to DiskIO BW.



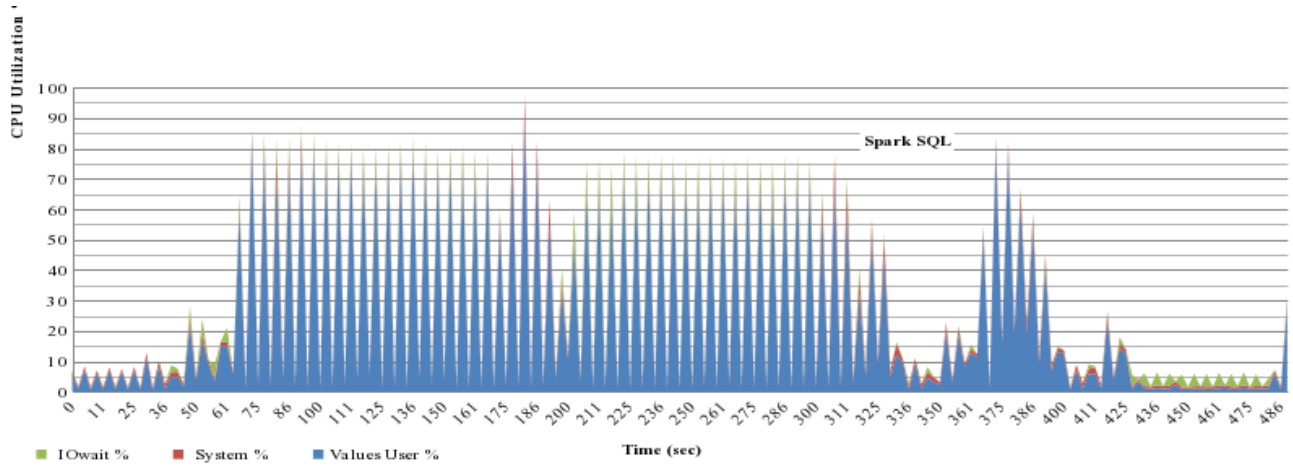
(a) ORC



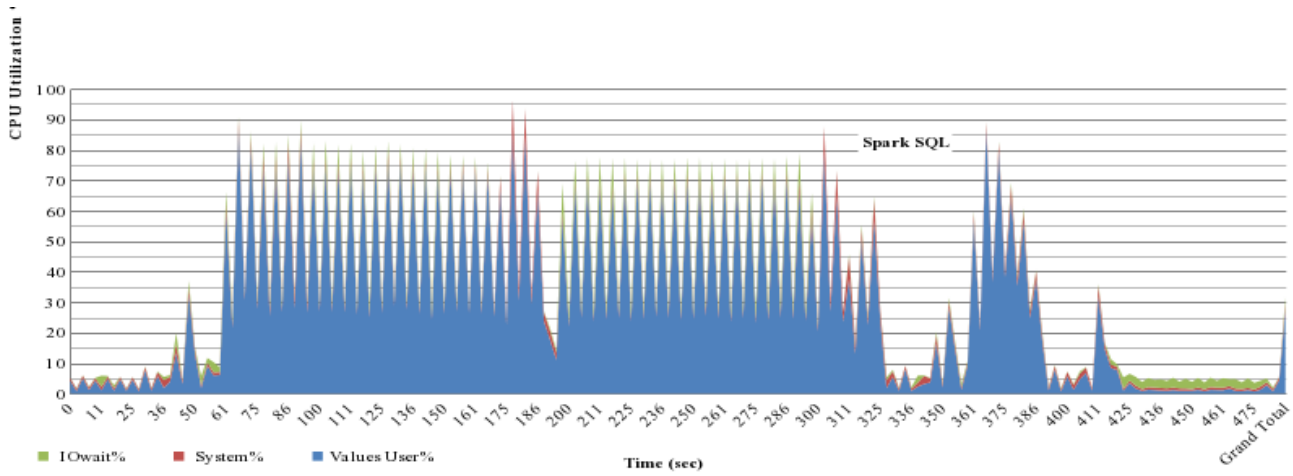
(b) Parquet

Figure 84: VMSTAT Context Switches

4.11 ORC - NoCompression vs. Snappy

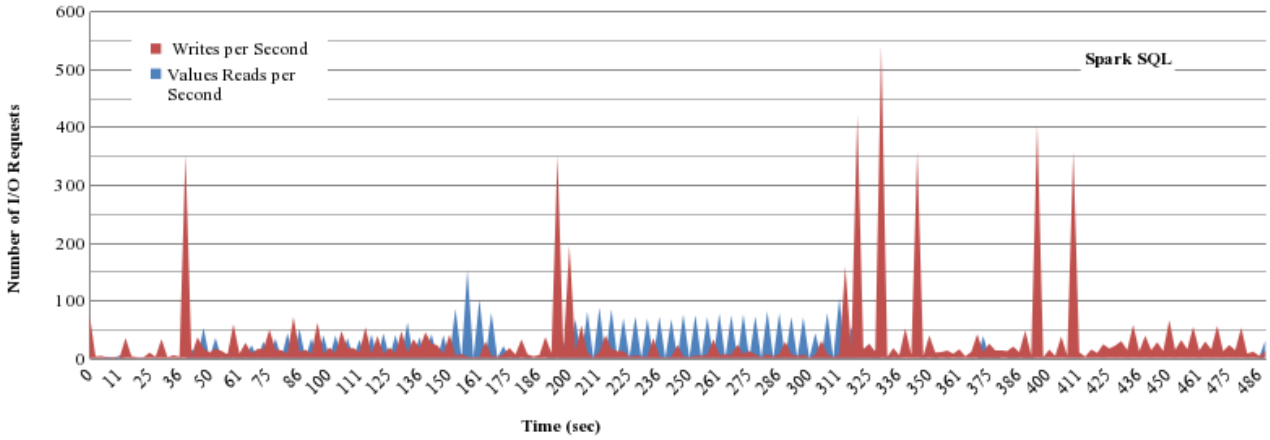


(a) NoCompression

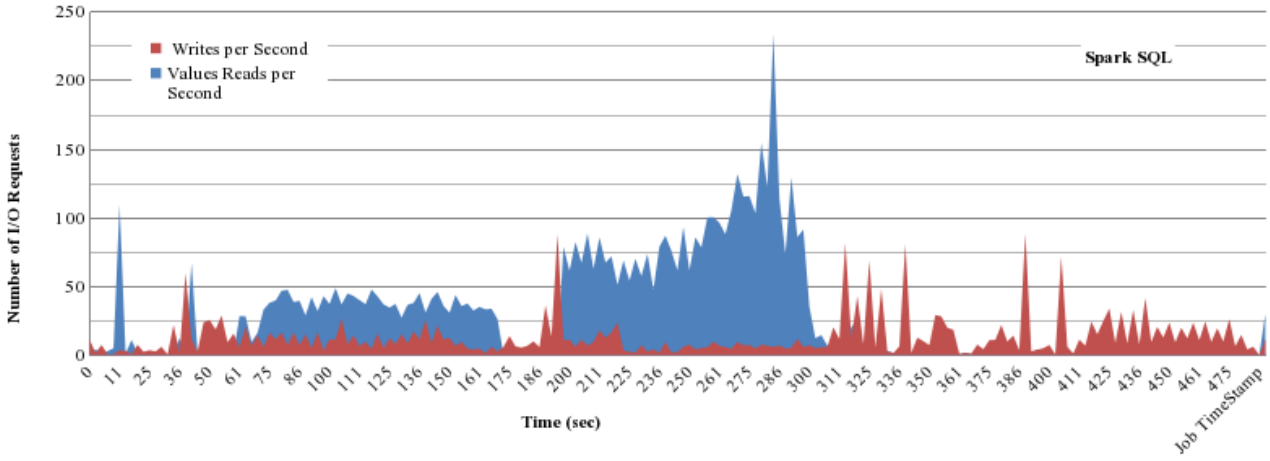


(b) Snappy

Figure 85: CPU.

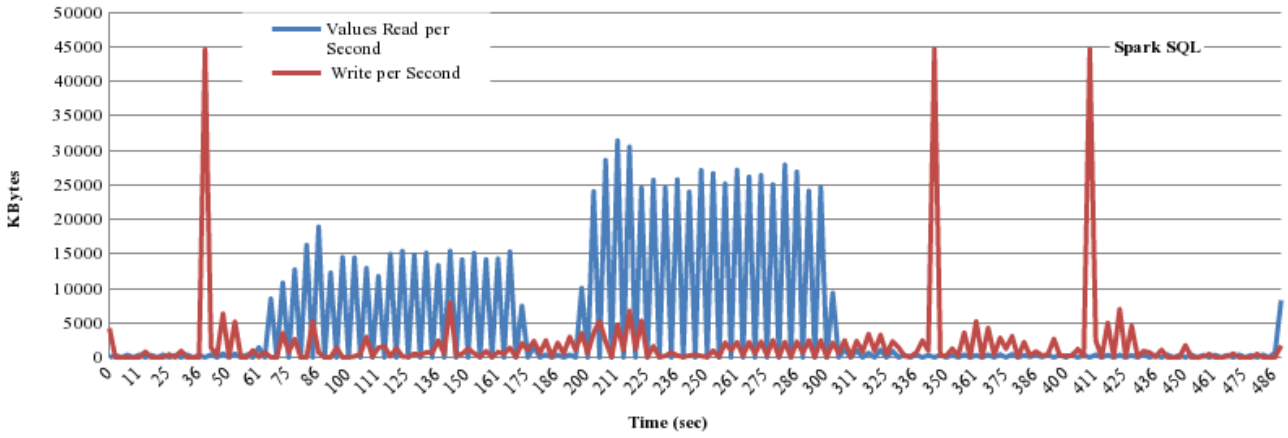


(a) NoCompress

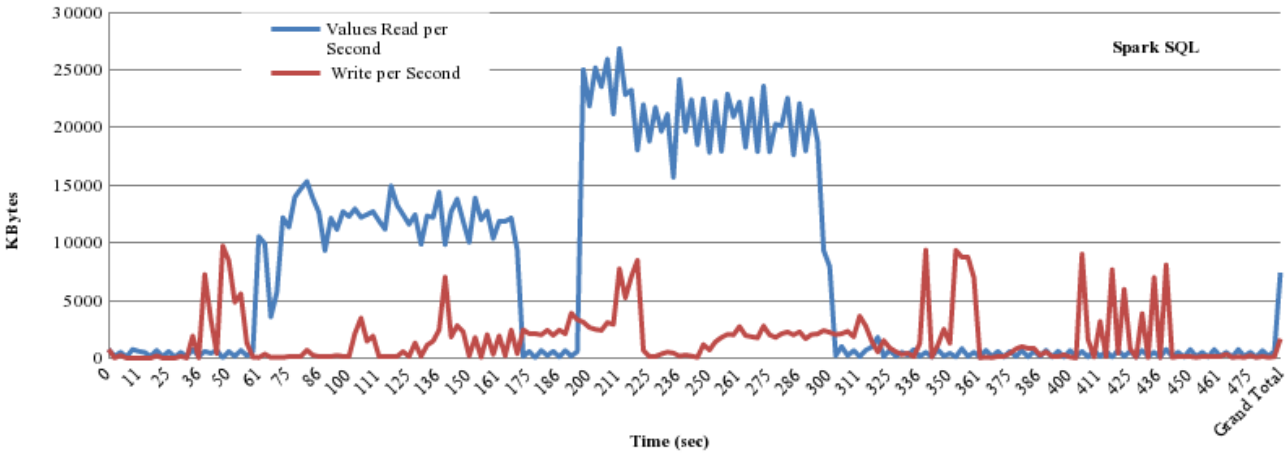


(b) Snappy

Figure 86: Disk Requests.

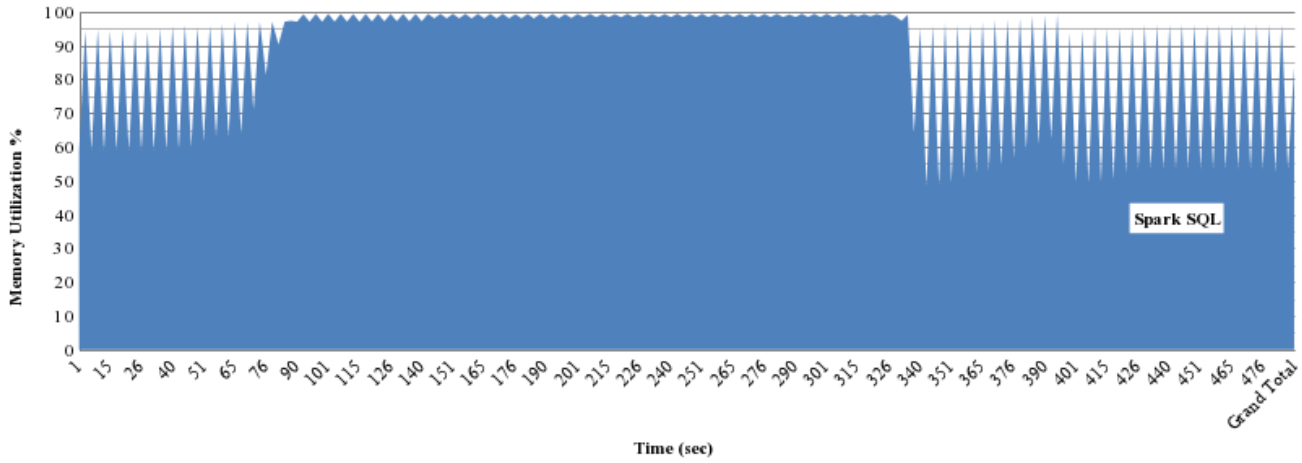


(a) NoCompress

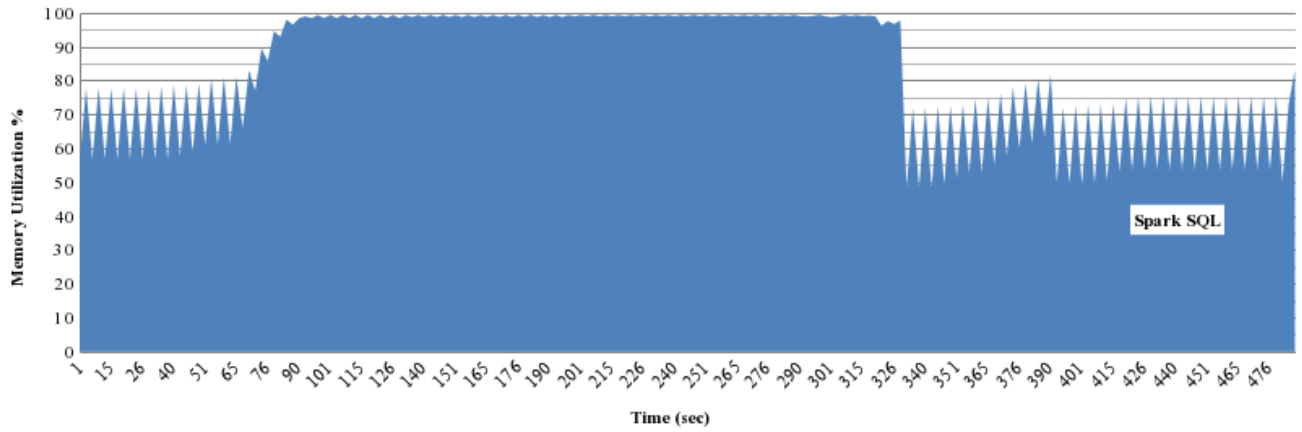


(b) Snappy

Figure 87: Disk Bandwidth.

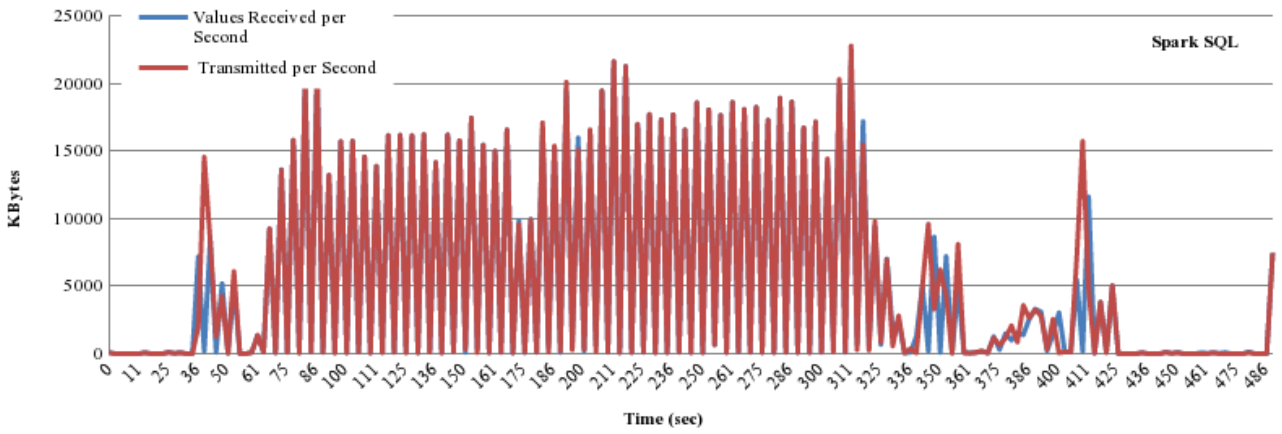


(a) NoCompress

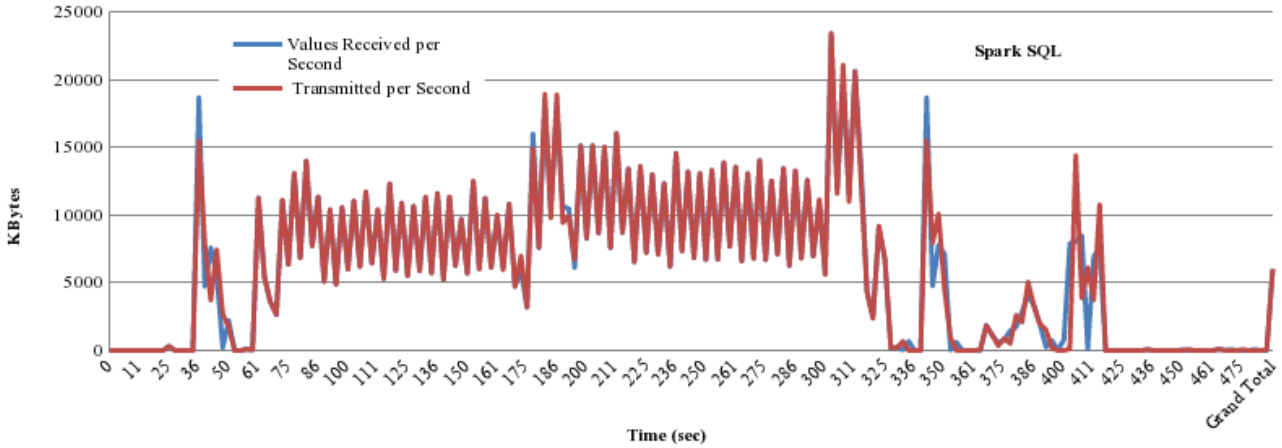


(b) Snappy

Figure 88: Free Memory.

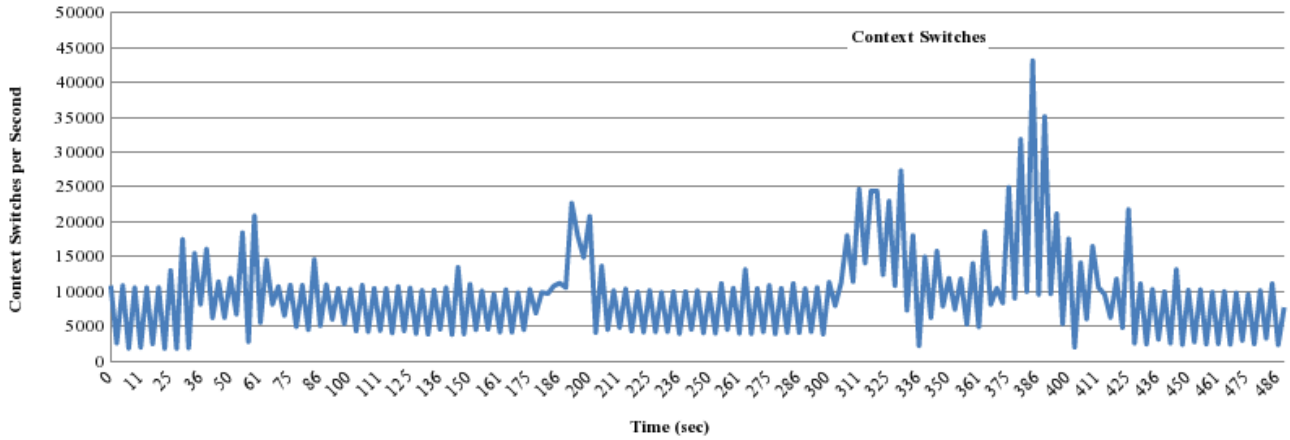


(a) NoCompress

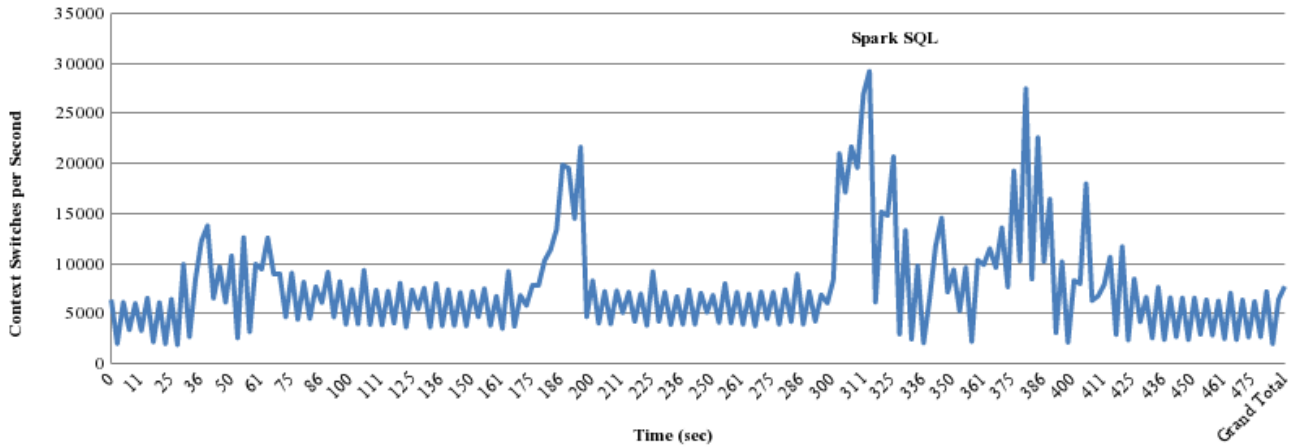


(b) Snappy

Figure 89: Network Utilization.



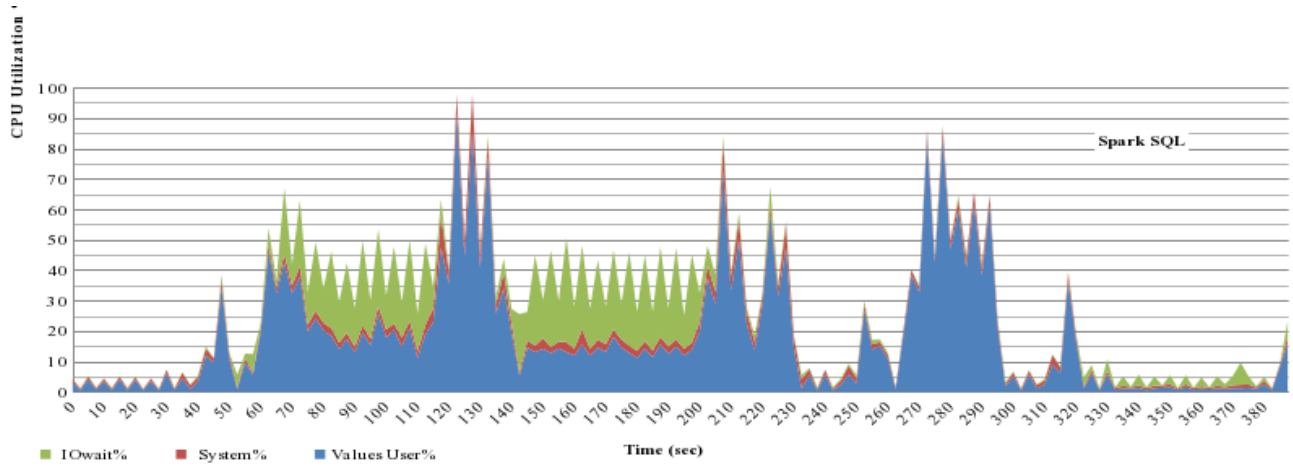
(a) NoCompress



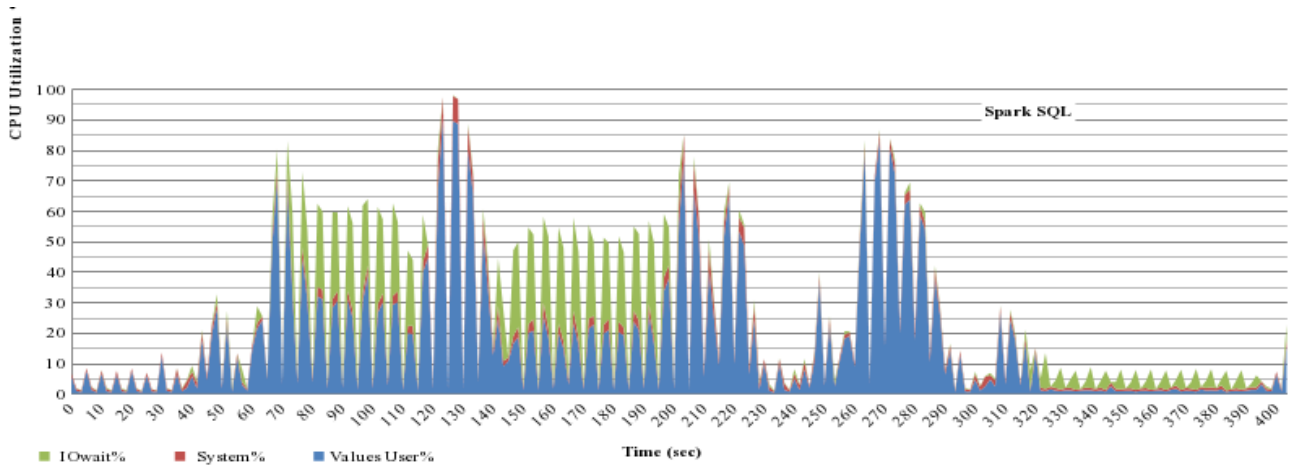
(b) Snappy

Figure 90: VMSTAT Context Switches

4.12 Parquet - NoCompression vs Snappy

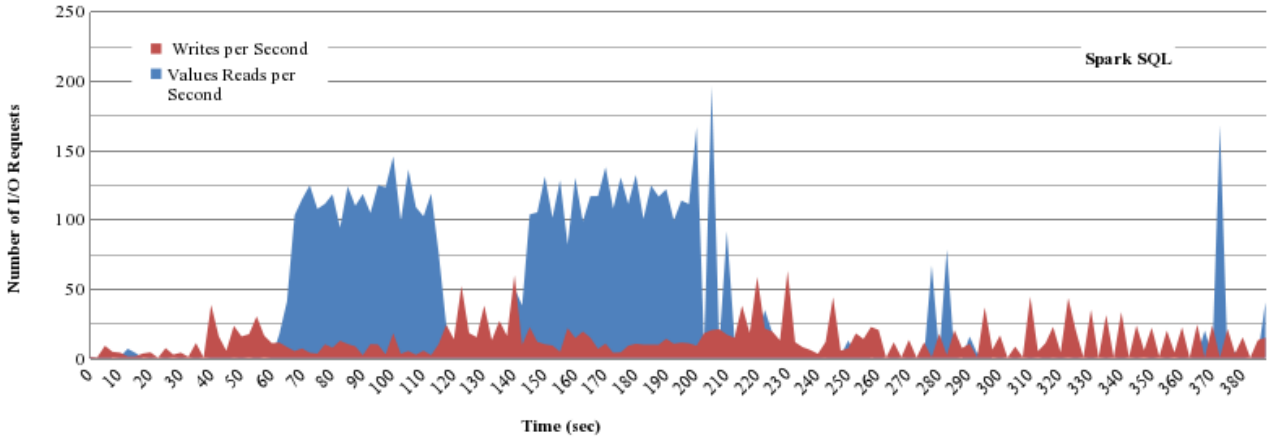


(a) NoCompression

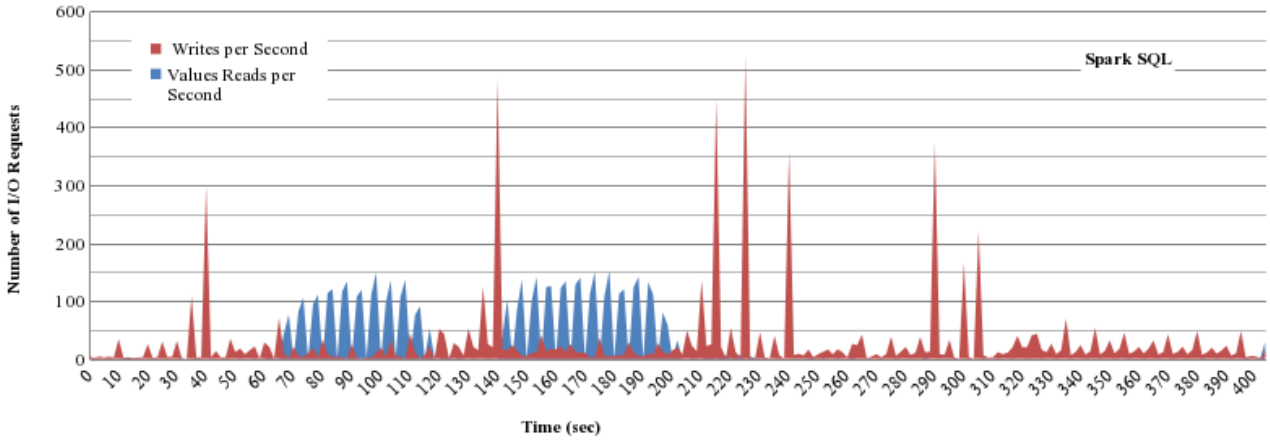


(b) Snappy

Figure 91: CPU.

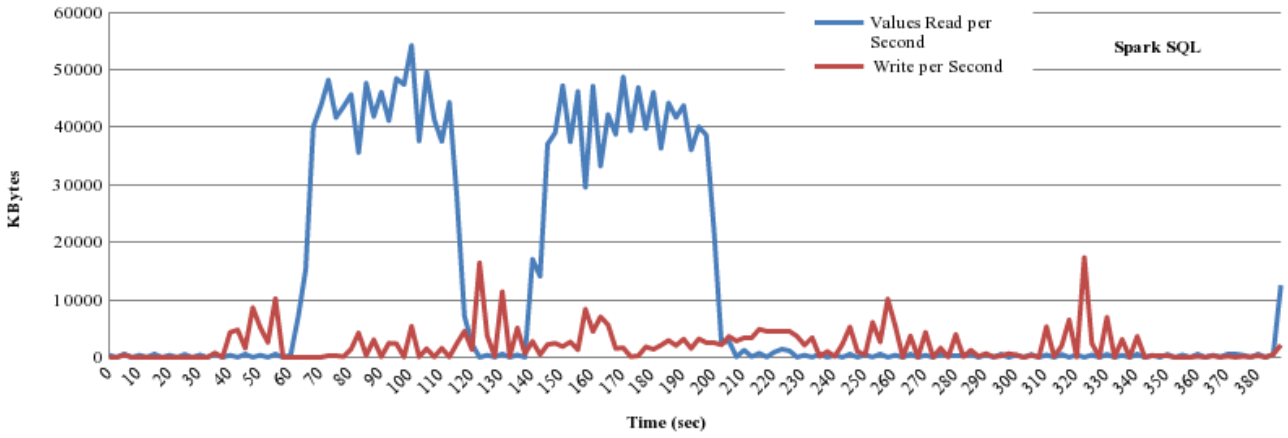


(a) NoCompression

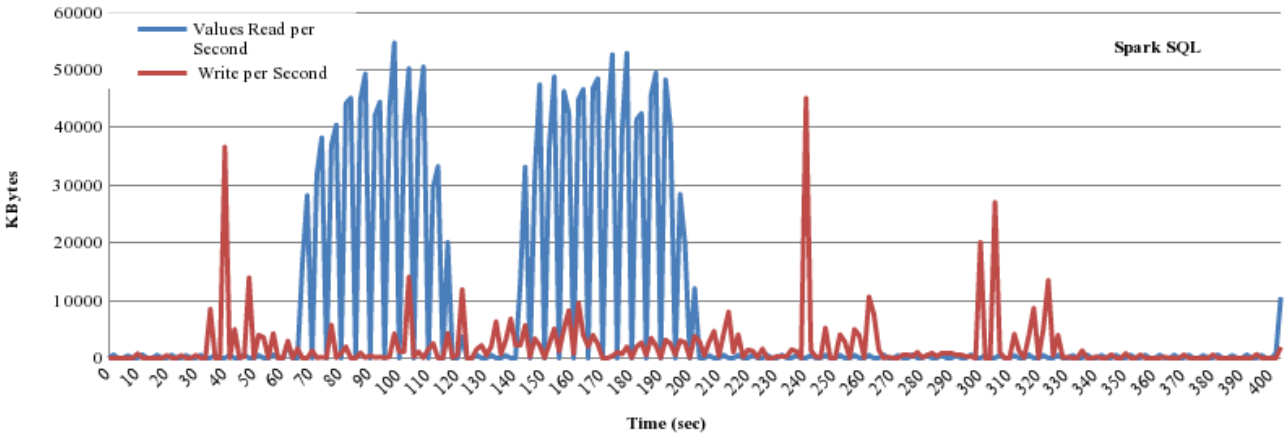


(b) Snappy

Figure 92: Disk Requests.

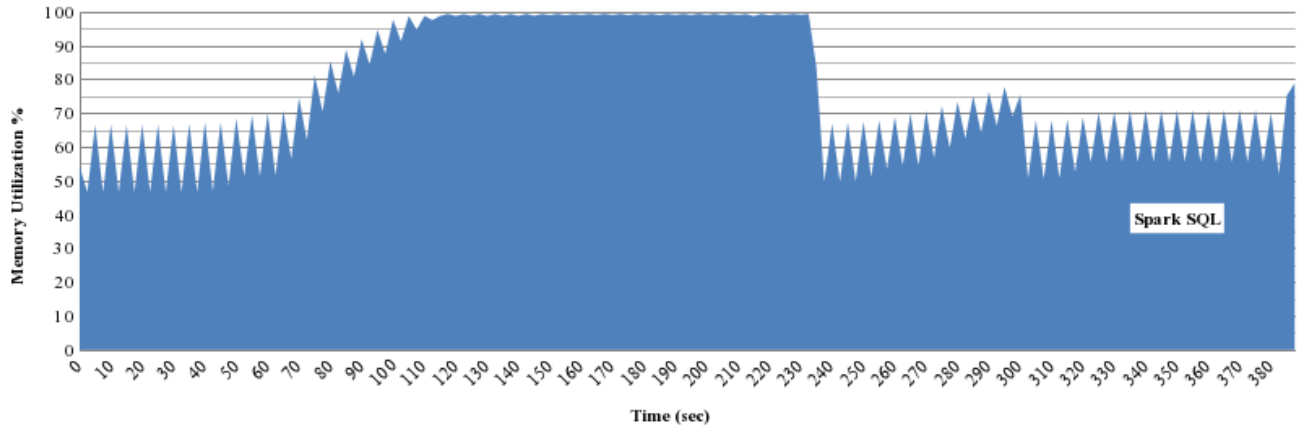


(a) NoCompression

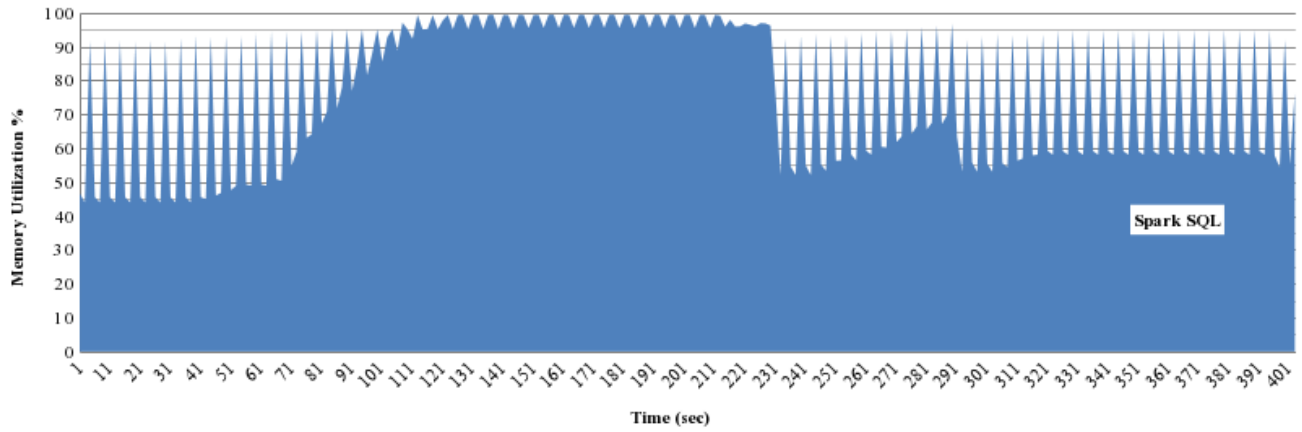


(b) Snappy

Figure 93: Disk Bandwidth.

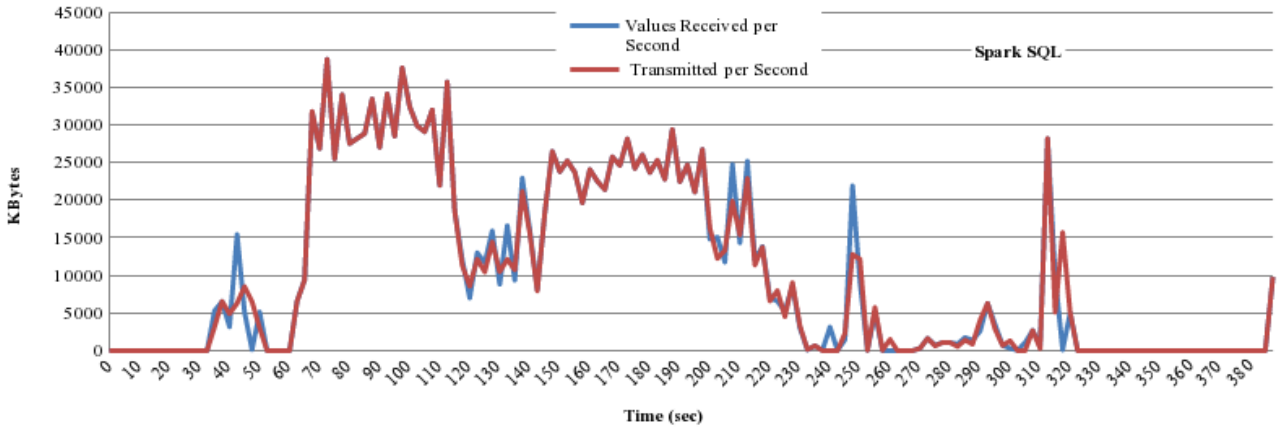


(a) NoCompression

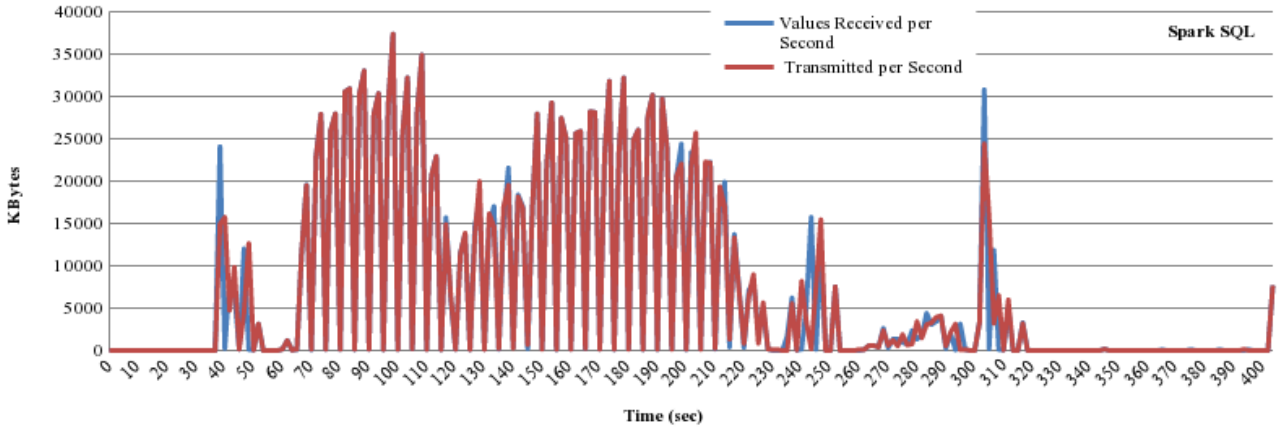


(b) Snappy

Figure 94: Free Memory.

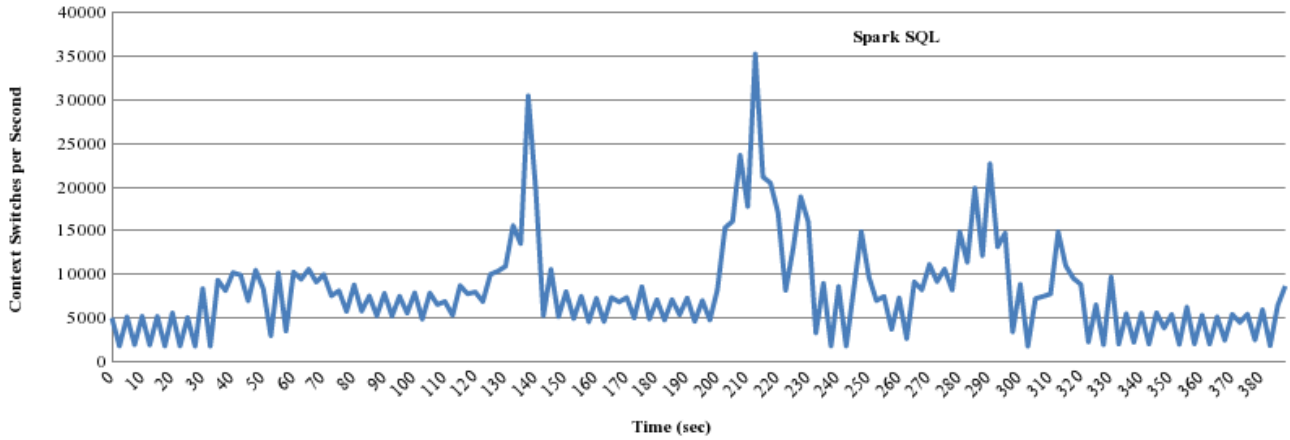


(a) NoCompression

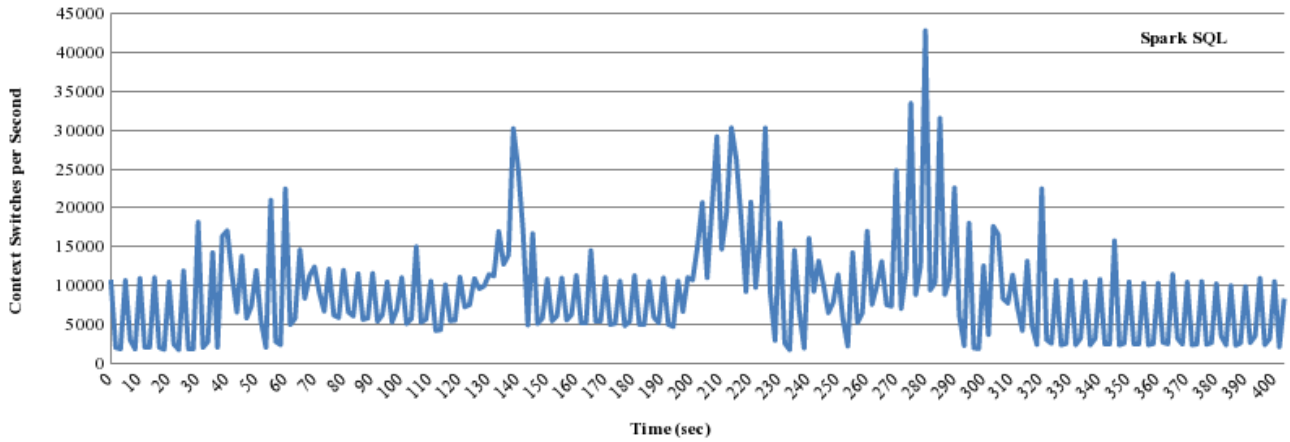


(b) Snappy

Figure 95: Network Utilization.



(a) NoCompression



(b) Snappy

Figure 96: VMSTAT Context Switches