2022 3rd International Conference for Emerging Technology (INCET)
Belgaum, India. May 27-29, 2022

SLC: Sliding Latency Coverage Factors for Optimal Performance Benchmarking of Storage Systems

Dr. Keshava Munegowda
Vice President, SecDB Engineering
Goldman Sachs
Bengaluru, India
keshava.gowda@gmail.com

Sanjay Kumar N.V
Associate Professor
Computer Science & Engineering
Tiptur, India
sanjaynv@gmail.com

2022 3rd International Conference for Emerging Technology (INCET) | 978-1-6654-9499-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/INCET54531.2022.9825170

***Abstract*—The Throughput in terms of bytes/seconds and records/seconds, the latencies range, average latency, median latency, quartiles and percentiles latencies are the de-facto factors for performance benchmarking of write and read operations of storage systems. This paper introduces the new factor called Sliding Latency Coverage(SLC). The SLC denotes the latencies range, median latency, latency quartiles and percentiles in a single unit factor. This paper also demonstrates the implementation of SLC factor as part of Storage Benchmark Kit (SBK). Even though implementation of SLC factor is specific to storage systems, but SLC factor can be used for performance analysis of any generic software system.**

Keywords—Benchmarking, Coefficient of Dispersion, Latency, Performance, Percentile , Quartile, SBK, SLC, Throughput.

I. INTRODUCTION

The SBK (Storage Benchmark Kit)[1][2][3] is an open-source software framework that we developed and it is used to measure the read and write performance of any storage system. The SBK supports performance benchmarking with various execution modes such as Burst mode (Maximum throughput mode), Throughput mode, Rate limiter mode, and End to End Latency mode. The SBK periodically delivers the benchmarking results such as write/read throughput, write/read latencies to the Grafana [4] analytics platform through Prometheus [5] monitoring systems to generate the performance graphs. The SBK initially developed with programming language Java [6] Version 8 [7] and later upgraded to Java version 17 [8]. In this paper Sliding Latency Coverage factors are defined and implemented as part of the SBK framework. To demonstrate the usage and significance of SLC factors the performance benchmarking of Ext4 file system (Extended File System , version 4) [9], LevelDB key value store [10] [11], RocksDB key value store [12] [13], and MinIO [14] [15] Distributed object storage system are conducted in this paper. During performance benchmarking, the data is stored in the local disk for Ext4 file system , RocksDB and LevelDB key value store. In case of MinIO distributed object storage the objects are sent/received to/from the remote hosted MinIO server [16].

II. SLC (SLIDING LATENCY COVERAGE) FACTORS DERIVATIONS

During Storage system performance benchmarking, SBK collects all the latency values and calculates the quartiles and percentiles. Percentiles are used to make large data set easier to interpret and to simplify data presentation by limiting all values to a 0 to 100 range. SBK outputs the latency percentiles in terms of 10th, 20th, 25th, 30th, 40th, 50th, 60th, 75th, 80th, 90th, 92.5th, 95th, 97.5th, 99th, 99.25th, 99.5th, 99.75th, 99.9th, 99.95th and 99.99th for every preconfigured time interval in the unit of seconds.

The SLC factors are introduced to combine all the percentile and quartile latency values of storage benchmarking ranging from 10^{th} percentile to 99.99^{th} percentile. The SLC factors denotes the latencies ranges too. While conducting maximum throughput (burst mode) performance benchmarking using SBK, its observed that most of the storage systems latencies drastically degrades after the median percentiles due to system load, hence we are defining SLC as 2 factors, Sliding Latency Coverage 1 (SLC 1) and Sliding Latency Coverage 2 (SLC 2).

To calculate SLC factors the percentile latency values are divided into two halves. The first half, percentile values from 10th percentile to all the percentile values which are less than and including median percentile and the second half is all the latency percentile values greater than median percentile to 99.99^{th} percentile. If the values of the latency percentile 10, 20, 25, 30, 40, 50, 60, 70, 75, 80, 90, 92.5, 95, 97.5, 99, 99.25, 99.5, 99.75, 99.99, then the value of the 6^{th} item 50 is a positional average value of the median.

median (m) = Value of $\left(\frac{(n+1)}{2}\right)$th term.

median denotes the central tendency which means the value of the middle item of series when it is arranged in ascending order.

Consider the range of latency values from x_1 to x_n in the ascending order, to calculate SLC factors it is divided into two ranges the first range is from x_1 to x_m and the second range is from x_{m+1} to x_n. x_1 denotes the lower latency percentile, x_m denotes the median (50^{th}) latency percentile, x_n denotes the highest latency percentile.

The average distance from x_1 to x_m calculated as

$$\left(\frac{(x_2 - x_1) + (x_3 - x_1) + (x_4 - x_1) + \ldots + (x_m - x_{m-1})}{m}\right)$$

Where m is the count of percentile values from x_1 to x_m.

Then SLC1 factor is denoted as

$$\mathrm{SLC1} = \frac{\left(\frac{(x_2-x_1)+(x_3-x_1)+(x_4-x_1)+\ldots+(x_m-x_{m-1})}{m}\right)}{x_1}$$

The above equation can be further simplified as below

$$\mathbf{SLC1} = \frac{1}{x_1.m}\sum_{i=2}^{m}(x_i - x_1)$$

$\sum =$ Symbol for Summation
x_i = Value of the i^{th} term
m = count of percentile values from x_1 to x_m

Similarly, SLC2 is derived as

$$\frac{\left(\frac{(x_{m+1}-x_m)+(x_{m+2}-x_m)+(x_{m+3}-x_m)+\ldots.+(x_n-x_m)}{n-m}\right)}{x_m}$$

This can be further simplified to below equation

$$\mathbf{SLC2} = \frac{\mathbf{1}}{x_m.(n-m)} \sum_{i=m+1}^{n} (x_i - x_{m+1})$$

n-m = count of percentile latency values from median percentile to the highest latency percentile value.

Note that SLC factors are similar to Coefficient of Dispersion (CD) but not exactly the same. Coefficient of Dispersion is a relative measure of finding the distances from before the median and after to the median. Whereas SLC1 is the relative measure of finding the distances from lower percentile latency values to median percentile latency value and SLC2 is the relative measure of finding the distances from median latency percentile values to maximum percentile latency value. The SBK implements the SLC1 and SLC2 formulas for both write and read performance benchmarking.

III. RESULTS AND DISCUSSION

Along with the SLC factors implementations the SBK framework generates the read/write loads for benchmarking experiments. The performance benchmarking is conducted with varying data sizes for 100, 1000, 100000 bytes with a single Reader and Writer. The SLC1 and SLC2 factors are used to demonstrate the performance analysis of read/write latency variations. Most importantly SLC2 factor denotes the latency variations at higher latency percentiles and thus indicates the stability of the storage system too. For Performance benchmarking, the hardware and software configuration are listed in TABLE1.

TABLE I. HARDWARE AND SOFTWARE CONFIGURATION OF THE TEST SETUP

Components	Remarks
Number of CPU Cores	16 CPUs Each of CPU is 64 Bit, 3.60GHz (Giga Hertz)
Main Memory Size	16GB (Giga Bytes)
Hard Disk Size	SSD (Solid State Drive) of size 1TB (Tera Bytes)
Operating System	Ubuntu 20.04.3 LTS (Focal Fossa) [17] [18]
Linux Kernel	Version 5.11.0
SBK	Version 0.97 [3]
Java Virtual Machine (JVM)	Version 17 [8]

A. SLC Factors for Ext4 File System.

Figure 1 shows the variations in write latencies percentiles for a data set of size 100 bytes. The latency time is measured in nanoseconds. It is observed that from 10th percentile to 50th percentile the increase in latency value is minimal, hence SLC1 factor is derived as 0. Note that there is a sudden percentage increase after 99.95th percentile yielding value 3 as SLC2 factor.

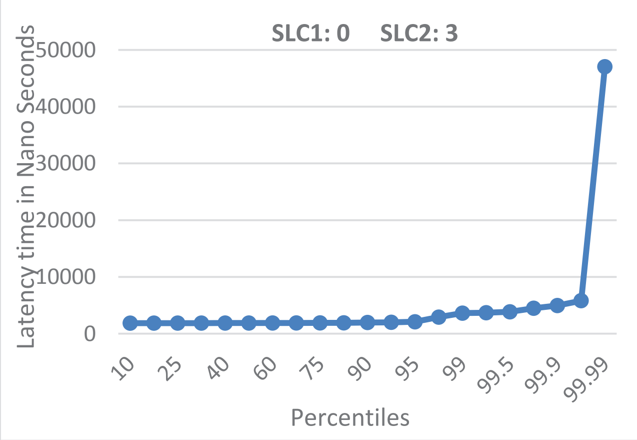

Fig. 1. Ext4 File System Write Latency variations for the data size : 100 Bytes

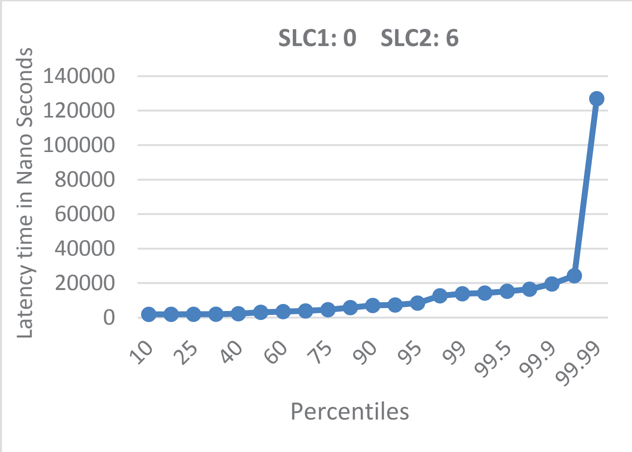

Fig. 2. Ext4 File System Write Latency variations for the data size : 1000 Bytes

In Figure 2 a small increase in latency value from the 10th percentile to the 50th percentile resulting in the SLC1 factor as 0. The SLC2 factor is 6 because there is a slight latency increase from the 60th percentile to 99.95th percentile, followed by a steep change at the 99.95th percentile. The value of SLC2 factors increases as the latency variation increases. Note that in Figure 1 , the latency degradation starts at 99.95th percentile but in Figure 2, the latency degradation starts at 60th percentile hence SLC2 factor in Figure 1 is less than SL2 factor of Figure 2.

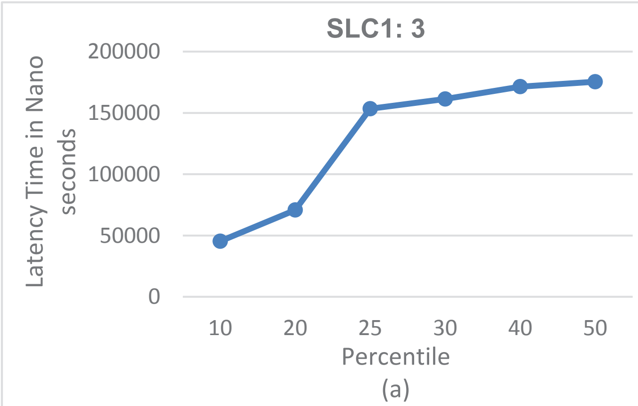

Fig. 3. (a). Ext4 File System SLC1 factor Write Latency variations for the data size : 1000 Bytes

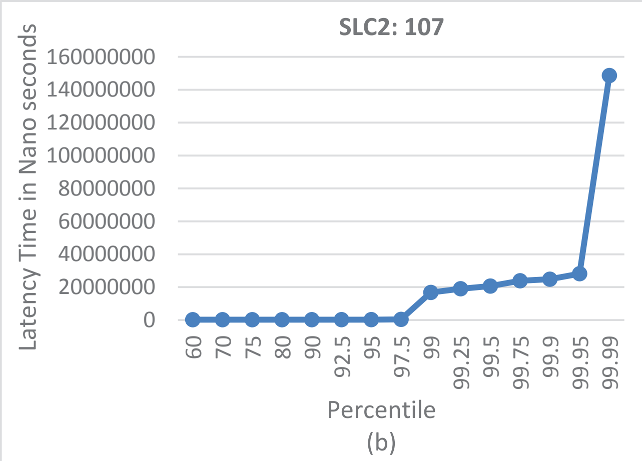

Fig. 3(b). Ext4 File System SLC2 factor Write Latency variations for the data size : 1000 Bytes

In Figure 3(a) slight increase in the write latency variations which gives the SLC1 factor as 3. In figure 3(b) latency start increasing from 97.5th percentile to 99.95th percentile and drastically increases from 99.95th percentile to 99.99th percentile, yielding SLC2 factor as 107. In this latency variation graph, we can observe that when the latency variation increases, the value of SLC factors increases, indicating the performance degradation. During File system write operations, the SLC2 value increases if the data size is higher.

File System - Read Latency Variations for the Data sizes (100, 1000, 100000) Bytes.

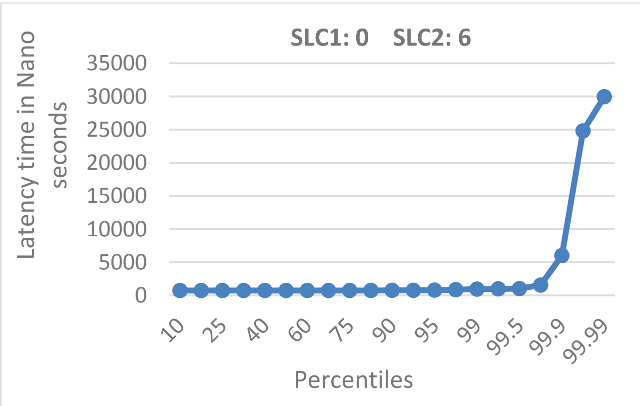

Fig. 4. File-System Ext4 Read SLC factor variations for the data size : 100 Bytes

As seen in Figure 4, the read latency variation is generally flat for data sizes of 100 bytes, yielding SLC1 factor as 0, with variation increasing slightly from 99.5th percentile to 99.99th percentile, as the latency variation rises, the SLC2 factor which gives the value as 6.

In the Figure 5 there is a small latency variation in SLC1 which derives SLC1 factor as 0. The variation in read latency is flat from 10th percentile to 99.5th percentile, then there is a mild increase until 99.95th percentile, followed by a steep slope till 99.99th percentile, due to a substantial change in the latency variation, which yields the value of SLC2 factor as 179.

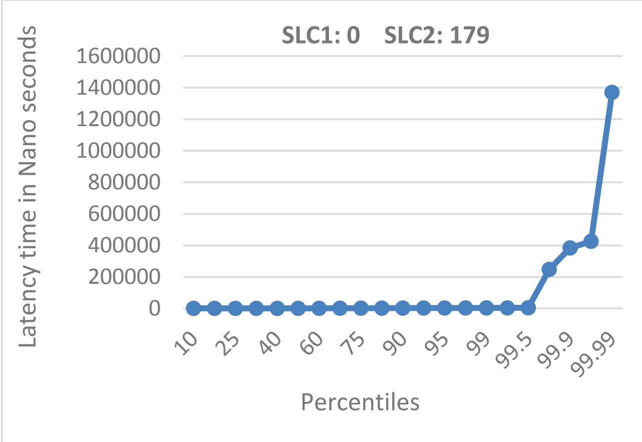

Fig. 5. Ext4 File System Read SLC factor variations for the data size : 1000 Bytes

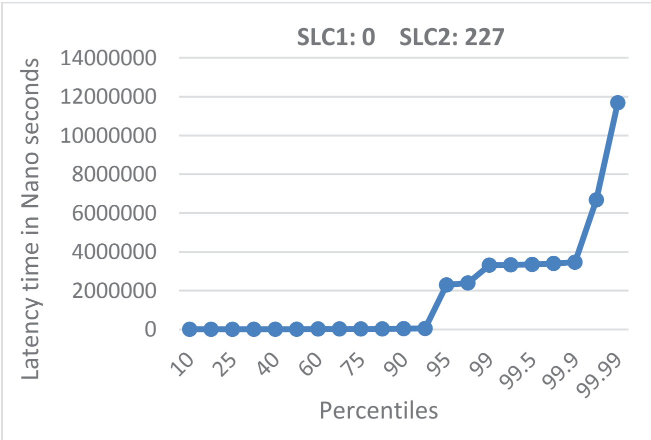

Fig. 6. Ext4 File System Read SLC factor variations for the data size : 100000 Bytes

Figure 6 shows variations in read latencies for a data set of 100000 bytes. It is observed that the latency value is minimal from 10th percentile to 50th percentile hence SLC1 factor is derived as 0. Note that latency variation goes up a little from 92.5th percentile 99.90th percentile, and a sudden rise in latency after the 99.90th percentile to 99.99th percentile yielding the value of SLC2 factor as 227. Since the huge SLC2 value indicates that performance degradation has happened after 90th percentile.

B. SLC Factors for LevelDB Key value store

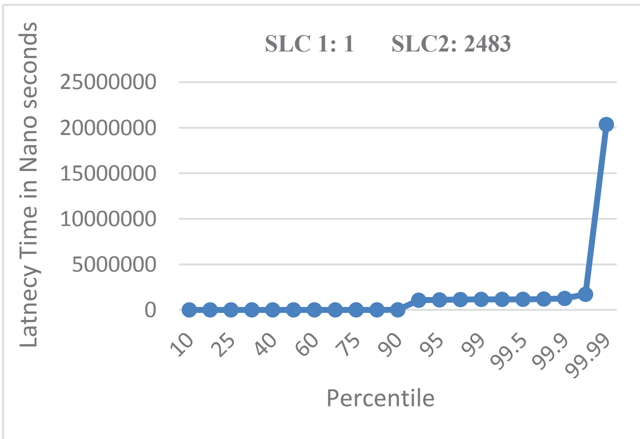

Fig. 7. LevelDB Write SLC factor variations for the data size : 100 Bytes

Figure 7 shows the write latency variation graph for the data set of 100 bytes. The latency time is measured in nanoseconds. It is observed that a modest increase in the latency variations yields the SLC1 factor value as 1. Note that the write operation takes a small increment in the latency variation from 90th percentile to 99.95th percentile, then there is a sudden spike occurs, yielding the maximum value of 2483 in the SLC2 factor. After 99.95th percentile, too much latency variations has occurred, implying that performance has deteriorated.

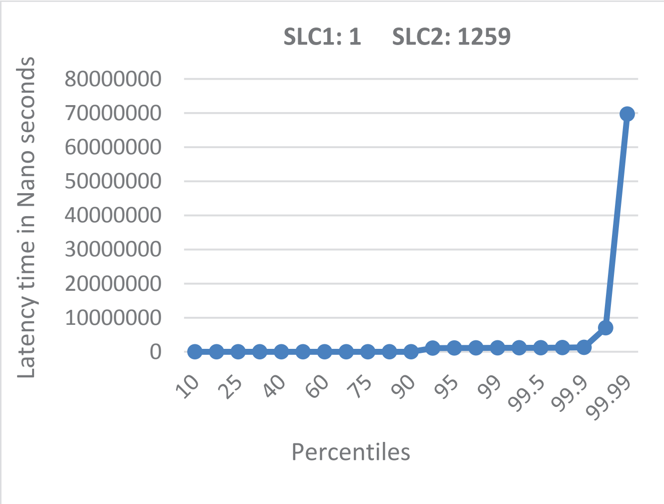

Fig. 8. LevelDB Write SLC factor variations for the data size : 1000 Bytes

Figure 8 shows write latency variations for data size of 1000 bytes. The variation in read latency is generally flat, with a small rise in latency value yielding SLC1 factor as 1. It is observed that there is a slight increase in the latency variation till 90th percentile, then a moderate increase from 92.5th percentile to the 99.95th percentile, and then it takes a sharp slope at 99.95th percentile to 99.99th percentile, yielding the SLC2 factor as 1259.

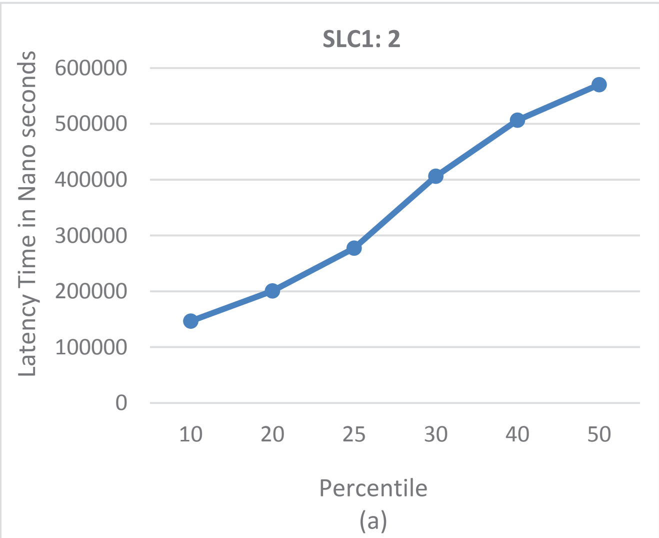

Fig. 9. (a). LevelDB Write SLC1 factor variations for the data size : 100000 Bytes

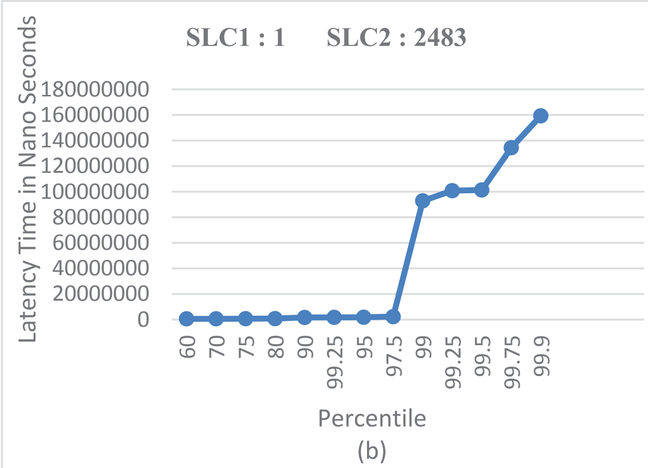

Fig. 9(b). LevelDB Write SLC2 factor variations for the data size : 100000 Bytes

In Figure 9(a) the SLC1 factor as 2 exhibits a slight increase in the latency variation. In Figure 9(b) the latency variation increases gradually from the 60th percentile to the 97.5th percentile, before it takes a steep slope at the 97.50th percentile. Huge value of SLC2 factor indicates that there is a very high latency, indicating the poor performance.

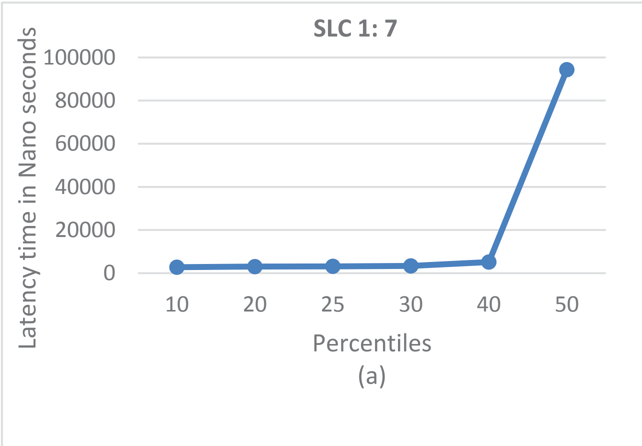

Fig. 10. (a). LevelDB Read SLC1 factor variations for the data size : 100 Bytes

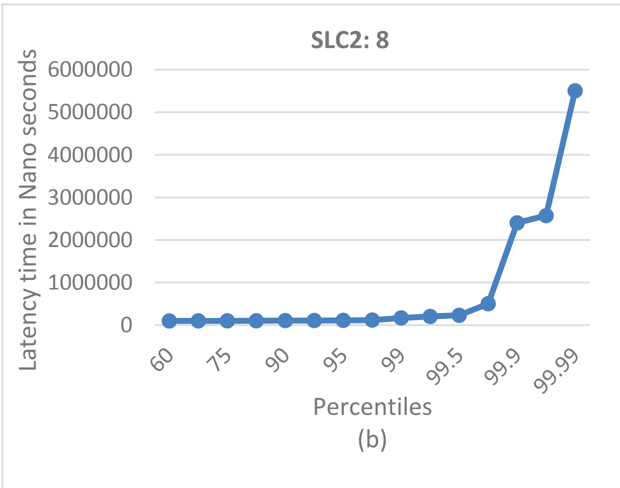

Fig. 10(b). LevelDB Read SLC2 factor variations for the data size : 100 Bytes

In Figure 10(a) depicts the LevelDB read latency variations for the data set of 100000 bytes, with a slight increase in latency variation from 10th percentile to the 50th percentile, resulting in an SLC1 factor of 7. In Figure 10(b) The latency variation increases little from 60th percentile to 99.75th percentile, then increases dramatically from the 99.75th percentile to the 99.99th percentile, providing SLC2 factor as 8.

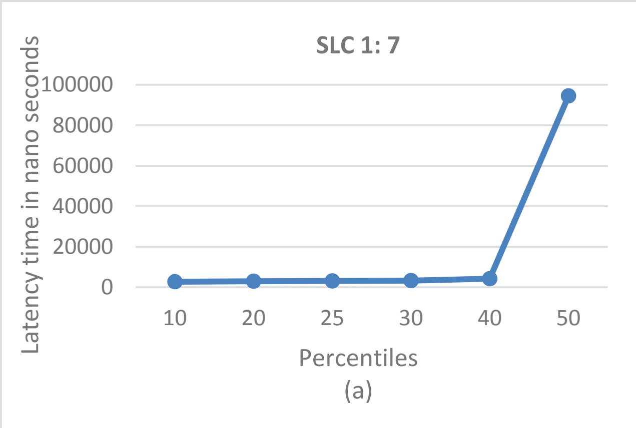

Fig. 11. (a). LevelDB Read SLC1 factor variations for the data size : 1000 Bytes

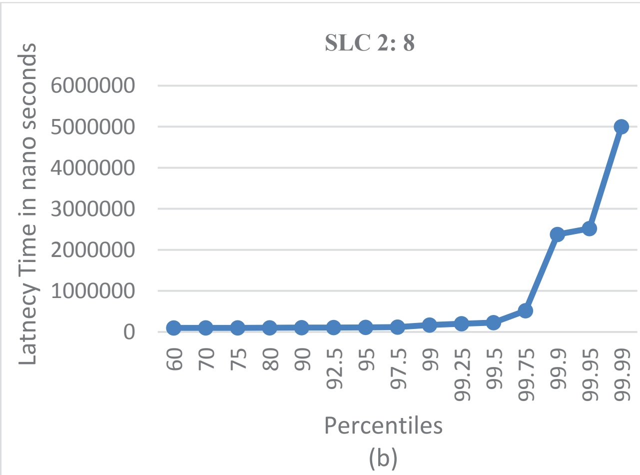

Fig. 11. (b). LevelDB Read SLC2 factor variations for the data size : 1000 Bytes

In Figure 11(a) A small increase in the latency variation in the LevelDB read operation for 100 bytes, giving the SLC1 factor as 7. In Figure 11(b), From the 60th percentile to 99.75th percentile, the latency variation is small, but from 99.75th percentile to 99.99th percentile, the latency variation keeps rising, which yields SLC2 factor as 8.

Figure 12(a) depicts the LevelDB read latency variations for 100000 bytes of data set, with a slight increase in latency variation from 10th percentile to 50th percentile, resulting in SLC1 factor as 7. In Figure 12(b) The latency variation grows slowly from the 60th percentile to 99.75th percentile, then rapidly from 99.75th percentile to 99.99th percentile, yielding an SLC2 factor as 8. Note that in LevelDB, if the data size is more then SLC factor values are minimal but latency values are high but if the data size is low then it records the lower latency values but SLC factors are higher indicating the larger latency range and variations.

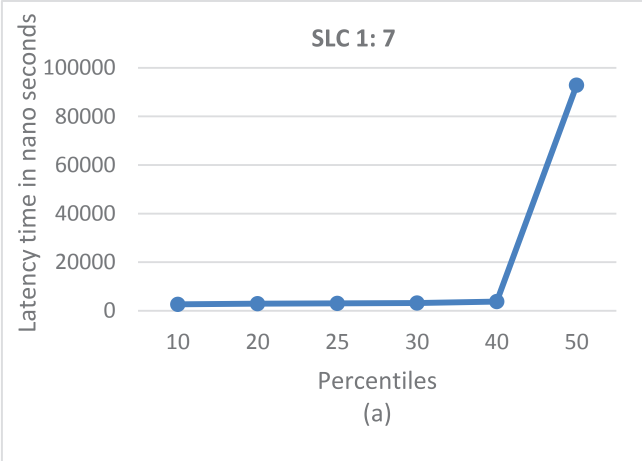

Fig. 12(a). LevelDB Read SLC1 factor variations for the data size : 100000 Bytes

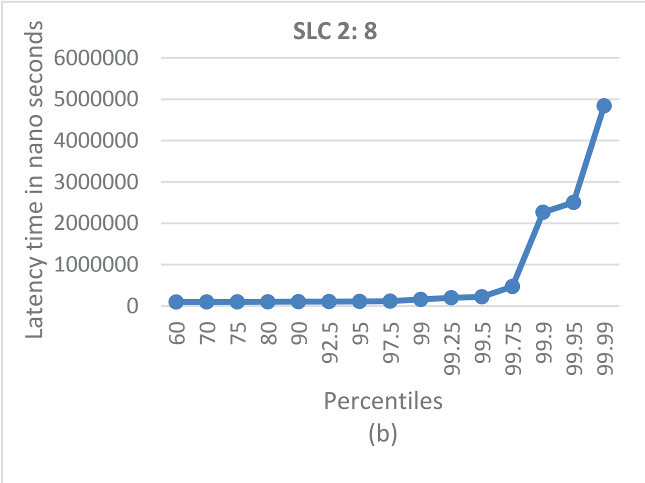

Fig. 12. (b). LevelDB Read SLC2 factor variations for the data size : 100000 Bytes

C. *SLC Factors for the RocksDB Key value store*

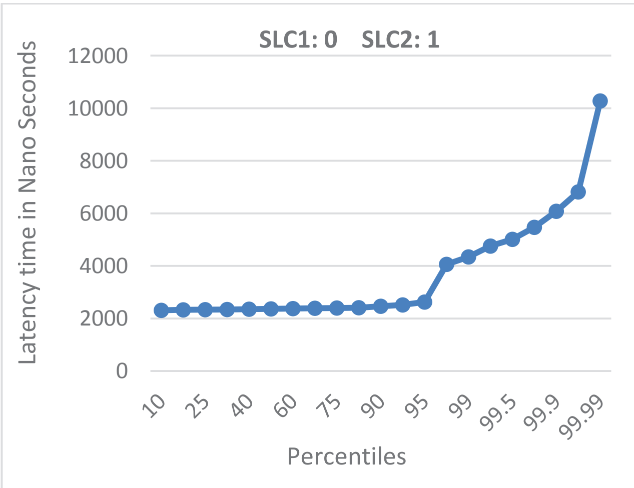

Fig. 13. RocksDB Write SLC factor variations for the data size :100 Bytes

Figure 13 shows variations in write latencies for a data set of 100 bytes. It is observed that from 10th percentile to 50th percentile increase in the latency is very small hence SLC1 factor is derived as 0. Note that the progressive increase in the latency from 70th percentile to 99.99th percentile yields value 3 as SLC2 factor. If the latency variation curve gap is small, it indicates that the system is providing stable performance, resulting in SLC factors with minimal difference.

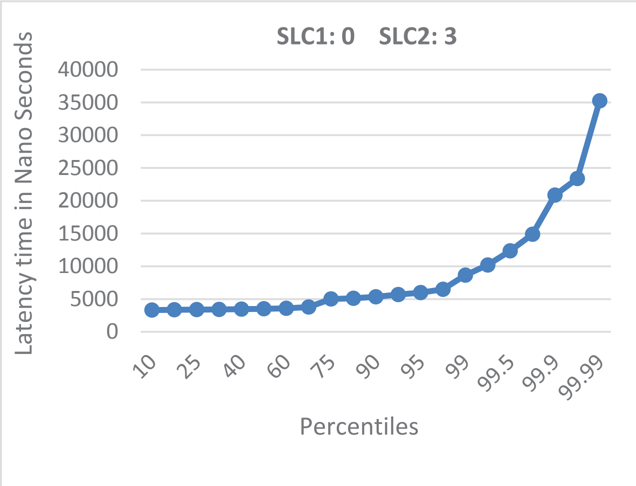

Fig. 14. RocksDB Write SLC factor variations for the data size : 1000 Bytes

Figure 14 depicts RocksDB write latency variations for a data set of 1000 bytes. Since the increase in the latency values from 10^{th} percentile to 50^{th} percentile is very low, the SLC1 factor is derived as 0. the gradual increase in latency from the 70^{th} to the 99.99^{th} percentile, yielding 3 as the SLC2 factor.

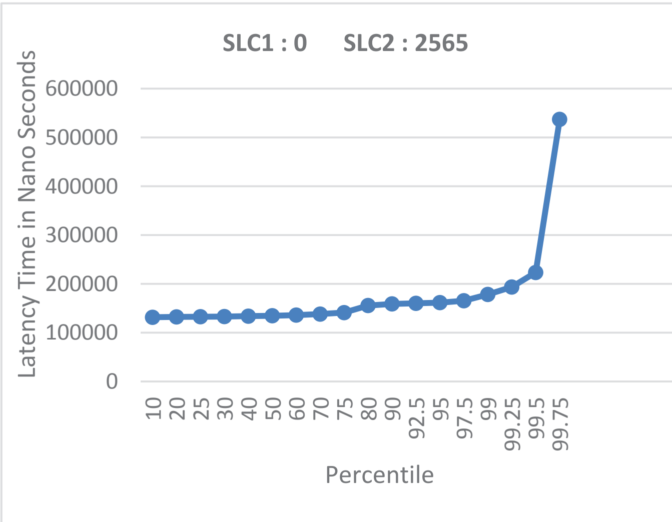

Fig. 15. RocksDB Write SLC factor variations for the data size : 100000 Bytes

Figure 15 shows RocksDB write latency variations for data size of 100000 bytes. The variation in write latency is generally flat, with a minimal increase in latency value yielding SLC1 factor as 0. It is observed that there is a slight increase in the latency variation till 99.5^{th} percentile, then a rapid increase from 99.5^{th} percentile to 99.75^{th} percentile. Since there is too much latency variations, resulting in an increase in the value of SLC2 factor as 2565, indicating the performance degradation after 99.5^{th} percentile.

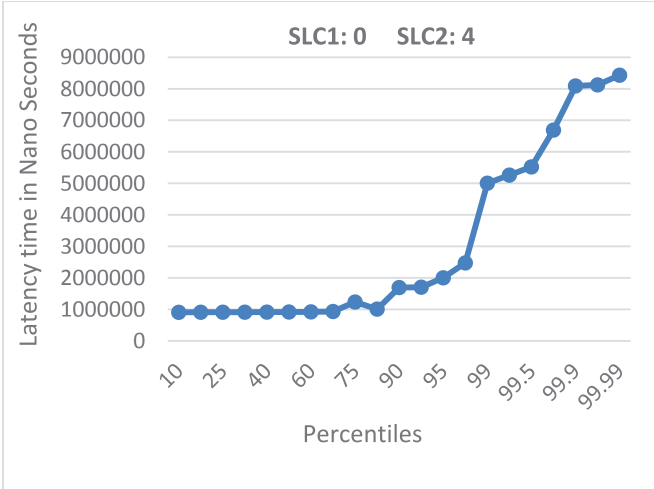

Fig. 16. RocksDB Read SLC factor variations for the data size : 100 Bytes

Figure 16 shows variations in read latencies for data set of 100 bytes. It is observed that the variation in read latency is generally flat, the increase in latency value from 10^{th} percentile to 50^{th} percentile is minimal, hence SLC1 factor is derived as 0. Note that latency values gradually increasing from 60^{th} percentile to 99.99^{th} percentile yielding the value of SLC2 factor as 4. The difference between each of the latency values are minimal hence the SLC2 factor value is low.

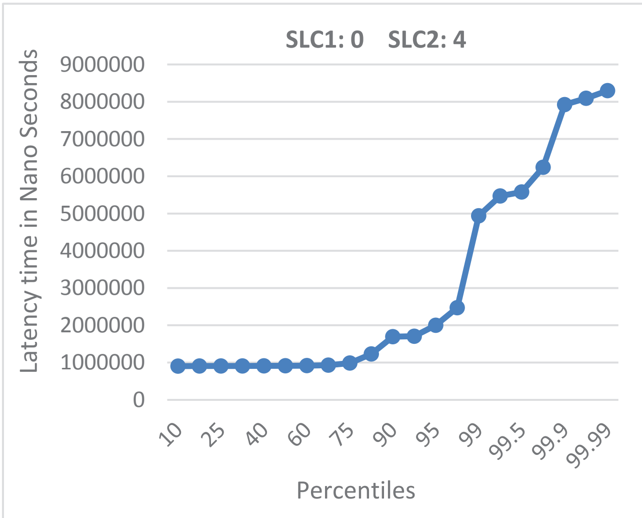

Fig. 17. RocksDB Read SLC factor variations for the data size : 1000 Bytes.

Figure 17 shows the read latency variation graph for data size of 1000 bytes. The read latency variation is generally flat from 10^{th} percentile to 50^{th} percentile which gives the value of SLC1 factor as 0. Note that latency variation increasing gradually from 60^{th} percentile to 99.99^{th} percentile yielding the value of SLC2 factor as 4.

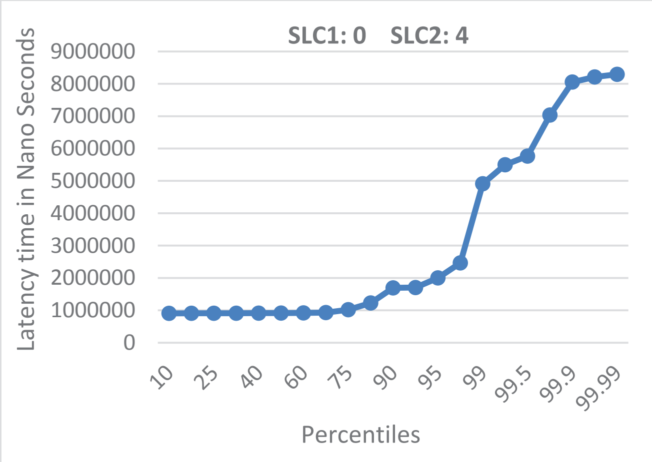

Fig. 18. RocksDB Read SLC factor variations for the data size : 100000 Bytes.

Figure 18 shows the read latency variation graph for the data size of 100000 bytes. It is observed that variations in read latencies are generally flat, and an increase in latency value is small resulting in SLC1 factor as 0. Because of the progressive increase in latency variation from 60^{th} percentile to 99.99^{th} percentile, the value of SLC2 factor is calculated as 4.

Note that if the latency variation curve gap is minimal, the performance is stable, resulting in low values for SLC factors.

D. SLC Factors for the MinIO Distributed Object Storage system :

Since Remote hosted MinIO server [11] has been used for read/write operations which includes the network latencies , we have used milliseconds as a time unit for latency measurement.

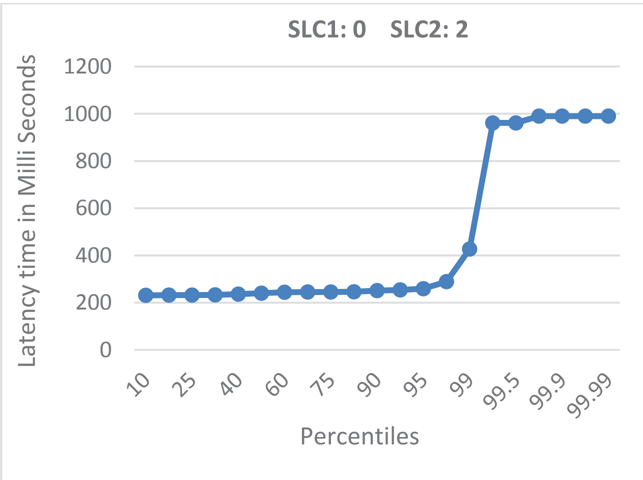

Fig. 19. MinIO Write SLC factor variations for the data size : 100 Bytes

Figure 19 shows the MinIO Write Latency variation for the data size of 100 bytes. The latency variation graph is generally flat from 10^{th} percentile to 50^{th} percentile, implying that an increase in the latency value is minimum, and so the SLC1 factor is derived as 0. Note that the latency values increased a little bit from the 95^{th} to the 99.75^{th} percentile, resulting in an SLC2 factor value as 2. When the minimal latency values differ, the SLC factors have a small value, resulting in the stable performance.

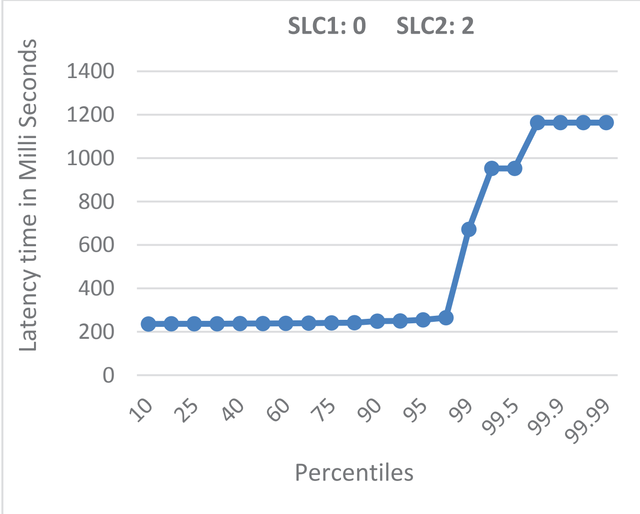

Fig. 20. MinIO Write SLC factor variations for the data size : 1000 Bytes

In figure 20 for data size of 1000 bytes, the read latency variation is generally flat which yields SLC1 factor as 0. The latency value increases minimally from the 95^{th} percentile to the 99.75^{th} percentile, yielding SLC2 factor as a minimal value 2.

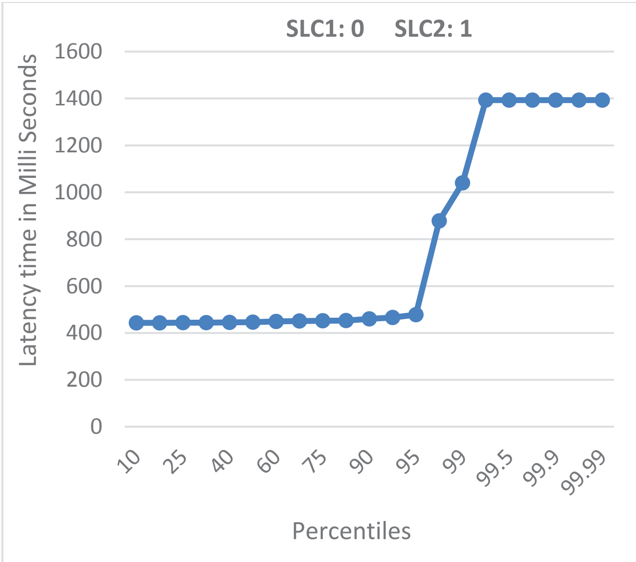

Fig. 21. MinIO Write SLC factor variations for the data size : 100000 Bytes.

Figure 21 shows the variations in write latencies for a data set of 100000 bytes. It is observed that from 10th percentile to 50^{th} percentile the increase in latency value is minimal hence SLC1 factor is derived as 0. However, there is a small increase in latency value, from the 95^{th} to the 99.25^{th} percentile, resulting in an SLC2 factor as minimal value 1.

Variations in read latencies for a 100-byte data set as shown in Figure 22. An increase in latency value from 10th percentile to 50^{th} percentile is small, hence the SLC1 factor is derived as 0. from 80^{th} percentile to 99^{th} percentile, a small increase in the latency values yielding SLC2 factor as 3.

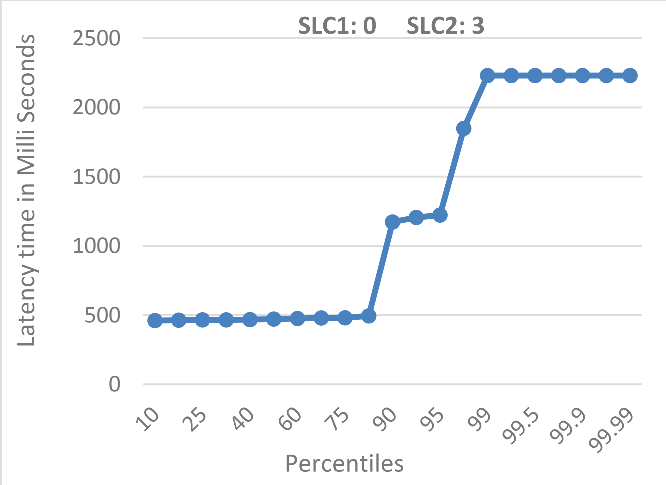

Fig. 22. MinIO Read SLC factor variations for the data size : 100 Bytes

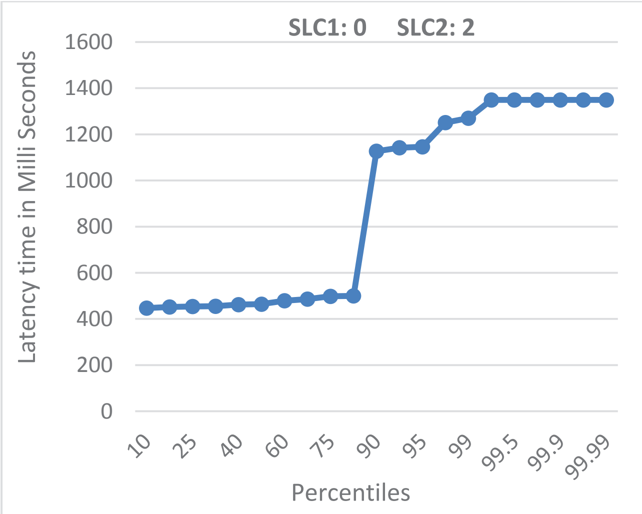

Fig. 23. MinIO Read SLC factor variations for the data size : 1000 Bytes

Figure 23 shows the variations in read latencies for the data set of 1000 bytes. The latency value increases by the smallest amount from 10th percentile to 50th percentile, which indicates the latency graph as generally flat, yielding the SLC1 factor as 0. From 80th percentile to 99.25th percentile, a small increase in the latency values yielding SLC2 factor as 2.

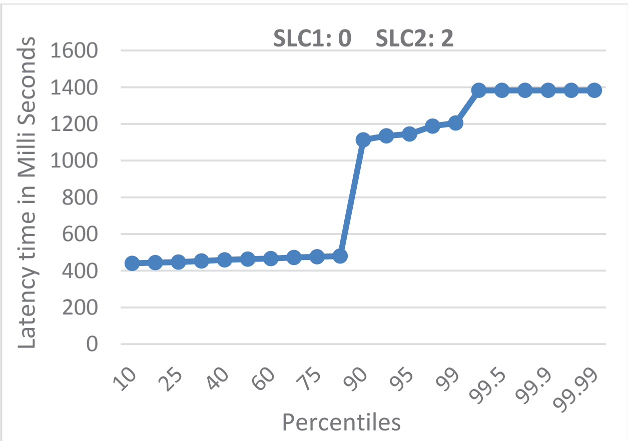

Fig. 24. MinIO Read SLC factor variations for the data size : 100000 Bytes

In Figure 24 depicts a small latency variation in SLC1 which derives SLC1 factor as 0. The read latency variation is generally flat from 10th percentile to 80th percentile, then increases from the 80th percentile to the 99.25th percentile, giving the SLC2 factor as 2.

IV. CONCLUSION

This paper defines and implements the SLC factors to understand the behavior of latencies range and latency percentiles variations in the performance benchmarking of any storage system. The performance benchmarking experiments conducted on the Ext4 file system, RocksDB, LevelDB, and MinIO storage systems are conducted using the SBK framework, demonstrates how we can study and analyze the latency range and variations using SLC factors. In the latency variation graphs it is observed that more the verticality of the latency graph, then more the SLC factor value. If the latency values are accumulated towards the higher latency percentile value, then also SLC factor increases. But if the latency values are accumulated towards the lower latency value, then the SLC factor decreases. If there are two storage systems to compare but having the same lower and higher latency then the SLC factors indicate the latency variations and thus the stability of the storage systems.

V. REFERENCES

[1] Munegowda K., Sanjay Kumar N.V. "Design and Implementation of Storage Benchmark Kit", In: Emerging Research in Computing, Information, Communication and Applications. Lecture Notes in Electrical Engineering, vol 790, Springer, Singapore. https://doi.org/10.1007/978-981-16-1342-5_5 , 2022.

[2] Storage Benchmark Kit (SBK) : https://github.com/kmgowda/SBK , 2022.

[3] SBK Releases : https://github.com/kmgowda/SBK/releases/tag/0.97 , March 2022.

[4] Grafana Website : https://grafana.com/ , 2022.

[5] Prometheus Website : https://prometheus.io/ , 2022.

[6] Herbert Schildt, "Java Complete Reference", 12th Edition, November 2021.

[7] Java Version 8 , website : https://www.java.com/en/download/help/java8.html

[8] Java Version 17, Release notes : https://www.oracle.com/news/announcement/oracle-releases-java-17-2021-09-14/

[9] Extended File system , Version 4 (Ext4) : https://ext4.wiki.kernel.org/index.php/Main_Page , 2022.

[10] Luo, C., Carey, M.J. "LSM-based storage techniques: a survey", The VLDB Journal, Volume 29, 393–418, 2020. https://doi.org/10.1007/s00778-019-00555-y

[11] LevelDB , Git hub : https://github.com/google/LevelDB , 2022.

[12] Siying Dong, Andrew Kryczka, Yanqin Jin, Michael Stumm "RocksDB: Evolution of Development Priorities in a Key-value Store Serving Large-scale Applications". ACM Transactions on Storage, Volume 17, Issue 4, Article 26, November, 2021.

[13] RocksDB , website : http://rocksdb.org/ , 2022.

[14] RocksDB, Git hub : https://github.com/facebook/rocksdb , 2022.

[15] MinIO , website : https://min.io/ , 2022.

[16] MinIO Remote Server , Website : https://play.min.io , 2022.

[17] Richard Peterson, Ubuntu 20.04 LTS Desktop: Applications and Administration, Jun 2020.

[18] Ubuntu Website : https://releases.ubuntu.com/20.04/ , 2022.