

RocksDB Festival

Supported by IITP, StarLab.

August 2, 2021

송인호, 한예진

inhoinno@dankook.ac.kr , hbb97225@naver.com

TeamName : 멘탈모델을 만들고 싶어요

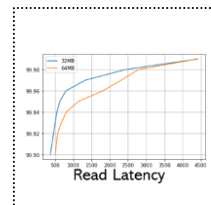
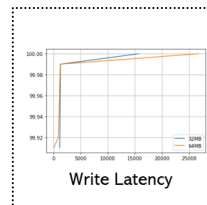
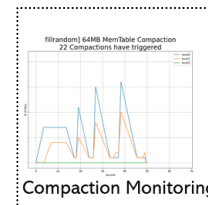
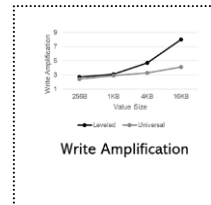
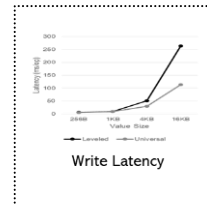
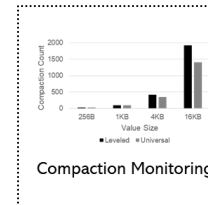
Contents

■ Last Week

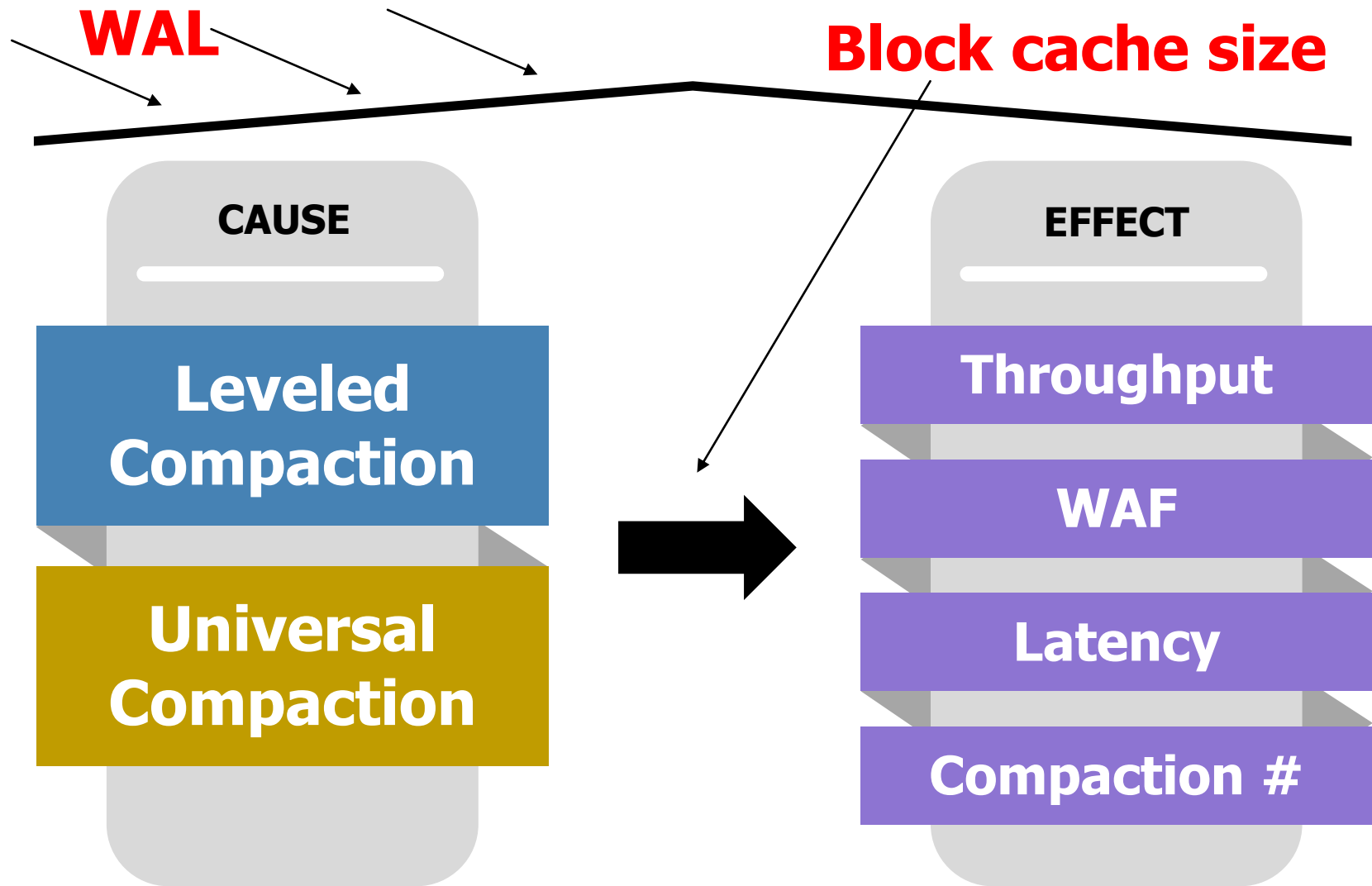
- ✓ Leveled vs Universal Compaction Comparison
 - Throughput
 - Write Amplification
 - Latency distribution + # of Compactions

■ This Week

- ✓ Leveled vs Universal Compaction Comparison
 - Block cache size – Hit ratio
 - Write-Ahead-Log – Throughput, Latency



LVL vs Univ Compaction Comparison

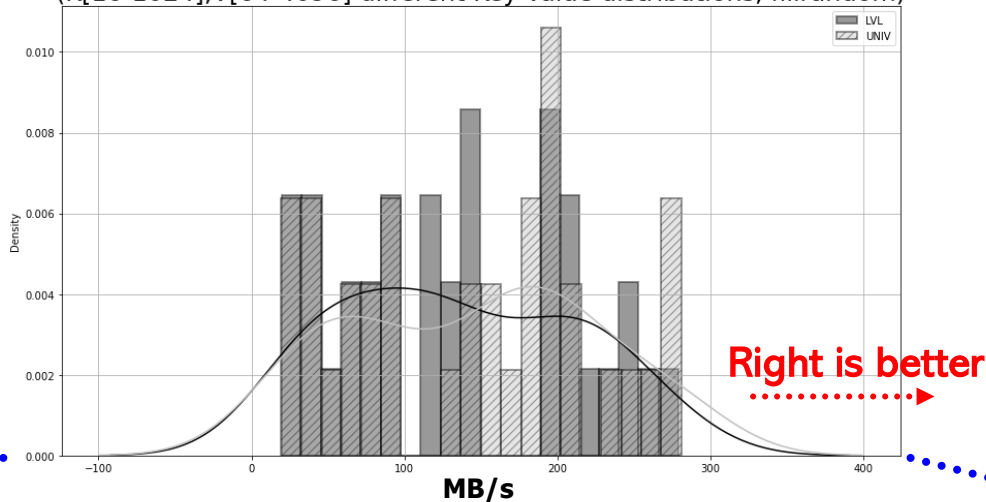


LVL vs Univ Write Throughput: WAL_OFF

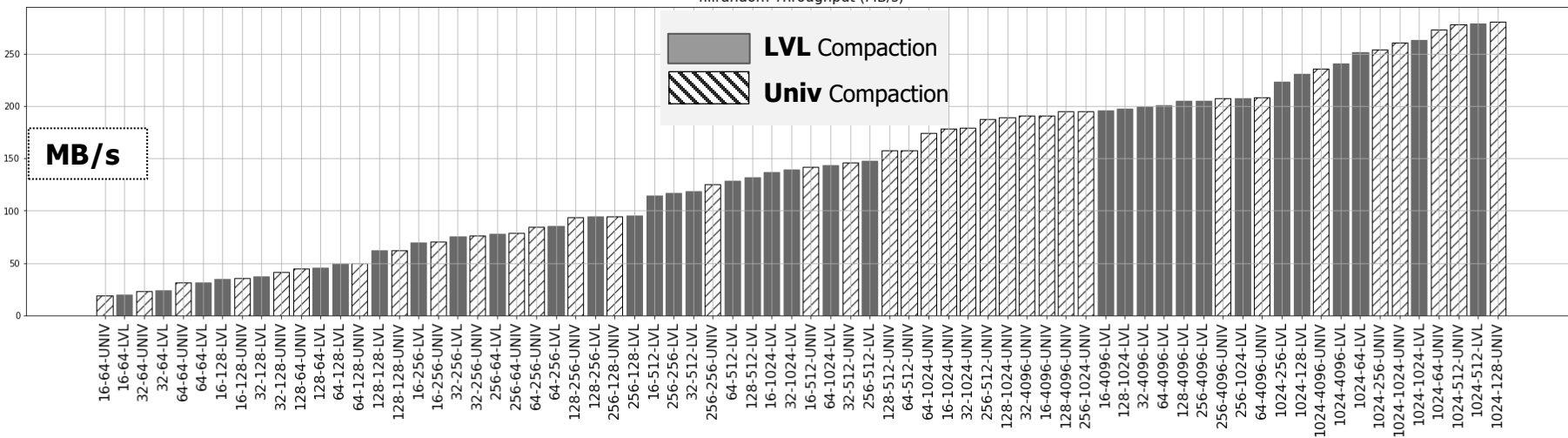
Fillrandom

Key [16, 32, 64, 128, 256, 1024]
Value [64, 128, 256, 512, 1024, 4096]
DB_Size 2.4GB
Storage Samsung 512GB 860 Pro
File System Ext4
CPU Intel(R) Core(TM) i5-4440 CPU @ 3.10GHz

Throughput comparison
 between different compaction Style
 (K[16-1024],V[64-4096] different Key-Value distributions, fillrandom)

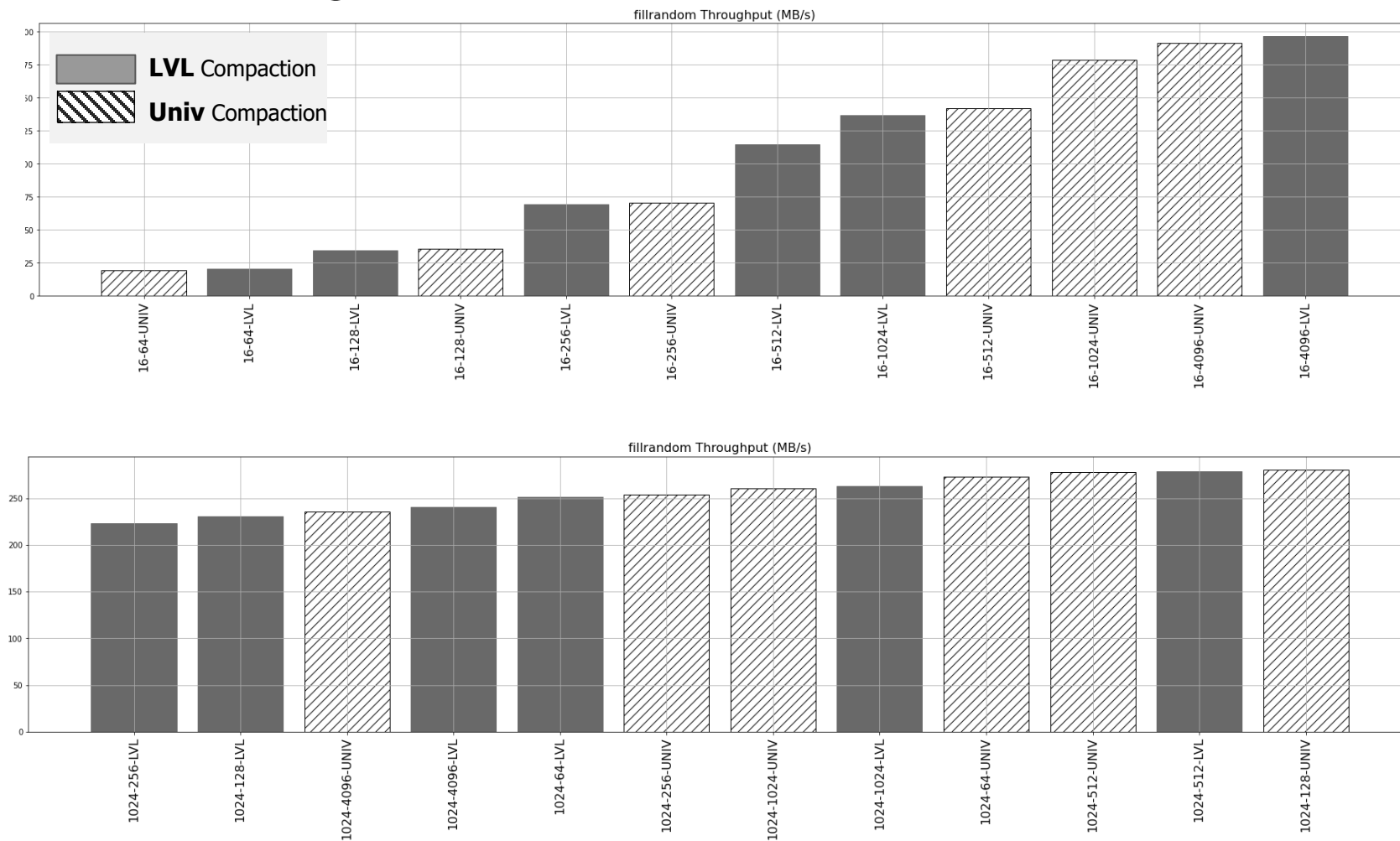


fillrandom Throughput (MB/s)



LVL vs Univ Write Throughput: WAL_OFF

■ Write Throughput: WAL_OFF - K16, 1024 / V[64-4096]



LVL vs Univ Read Throughput: WAL_OFF

Readrandom

--use_existing_db

Key [16, 32, 64, 128, 256, 1024]

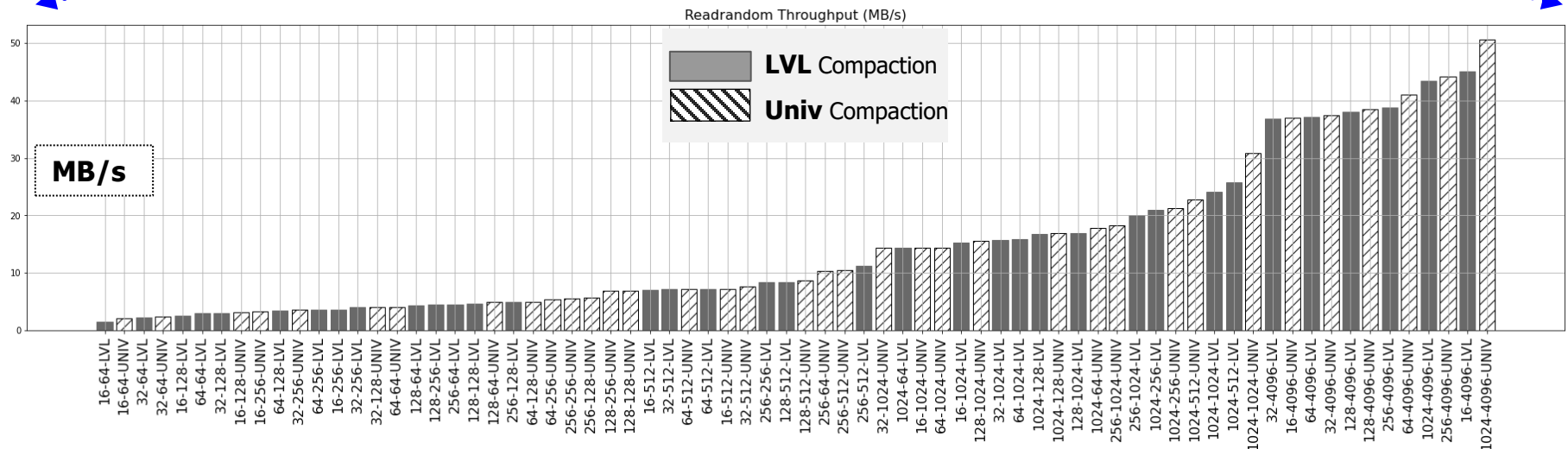
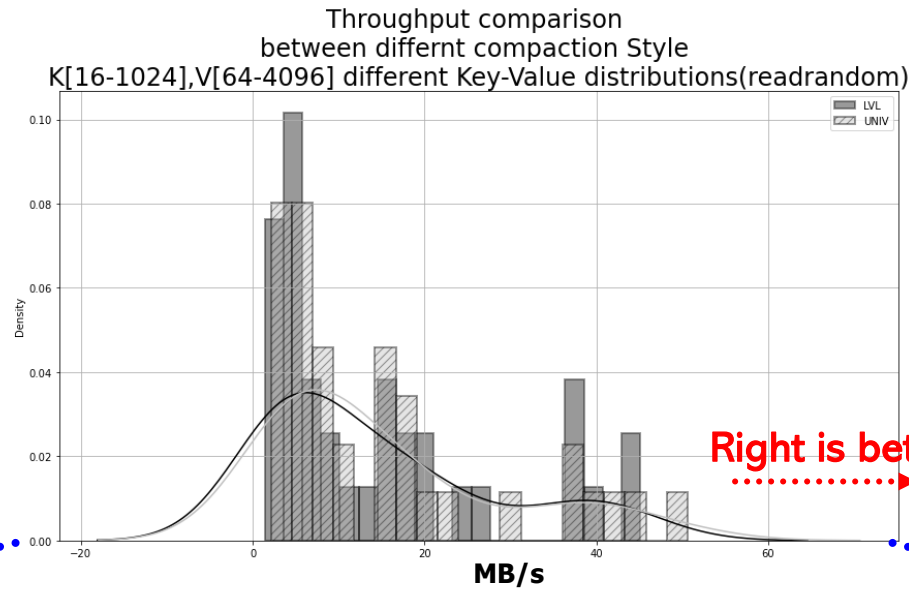
Value [64, 128, 256, 512, 1024, 4096]

DB_Size 2.4GB

Storage Samsung 512GB 860 Pro

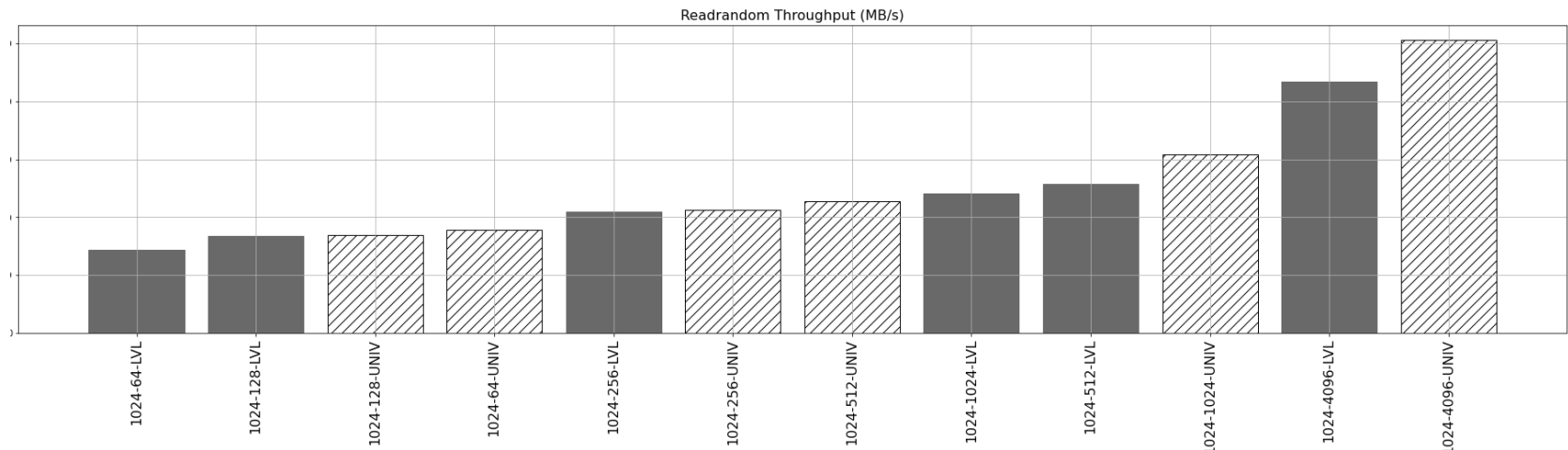
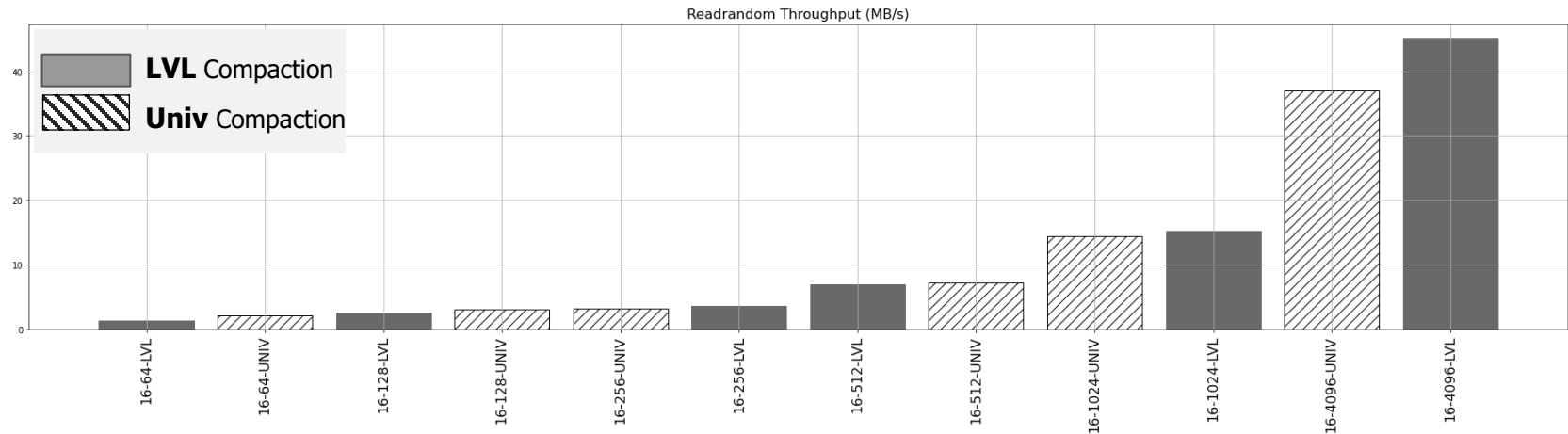
File System Ext4

CPU Intel(R) Core(TM) i5-4440 CPU @ 3.10GHz



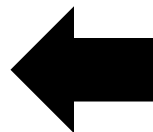
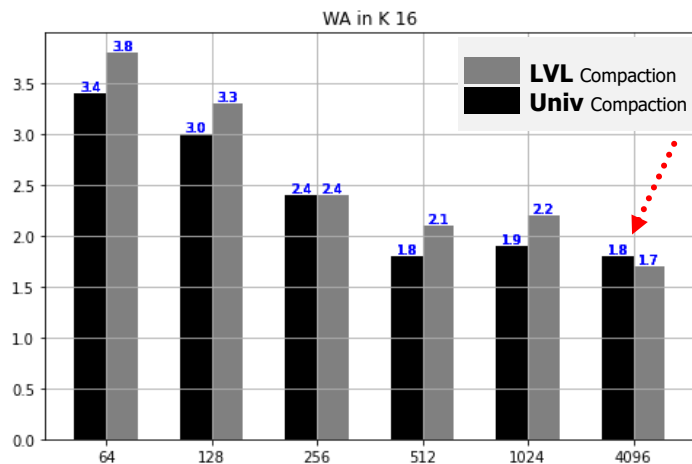
LVL vs Univ Read Throughput: WAL_OFF

■ Read Throughput: WAL_OFF - K16, 1024 / V[64-4096]

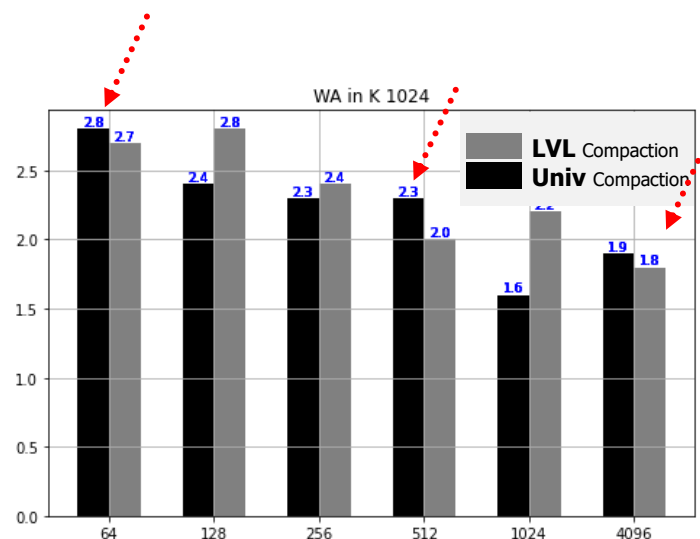
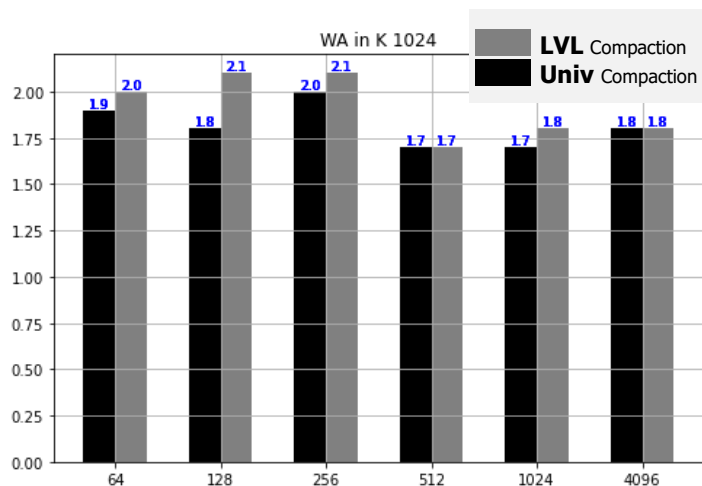
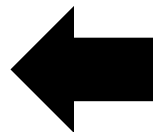
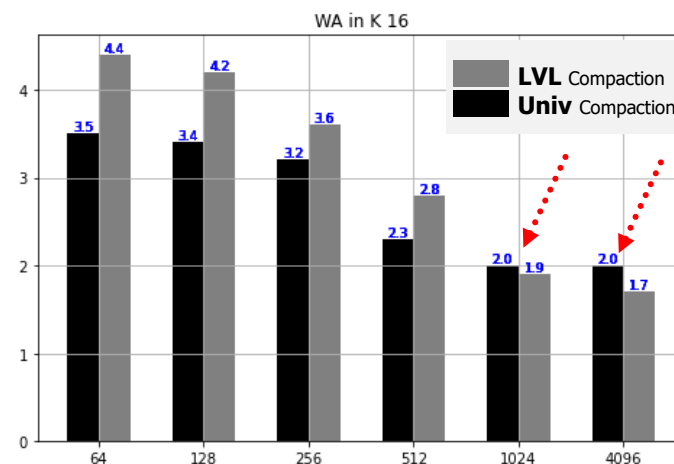


LVL vs Univ WAF Comparison:WAL_OFF

■ WAF: WAL_OFF

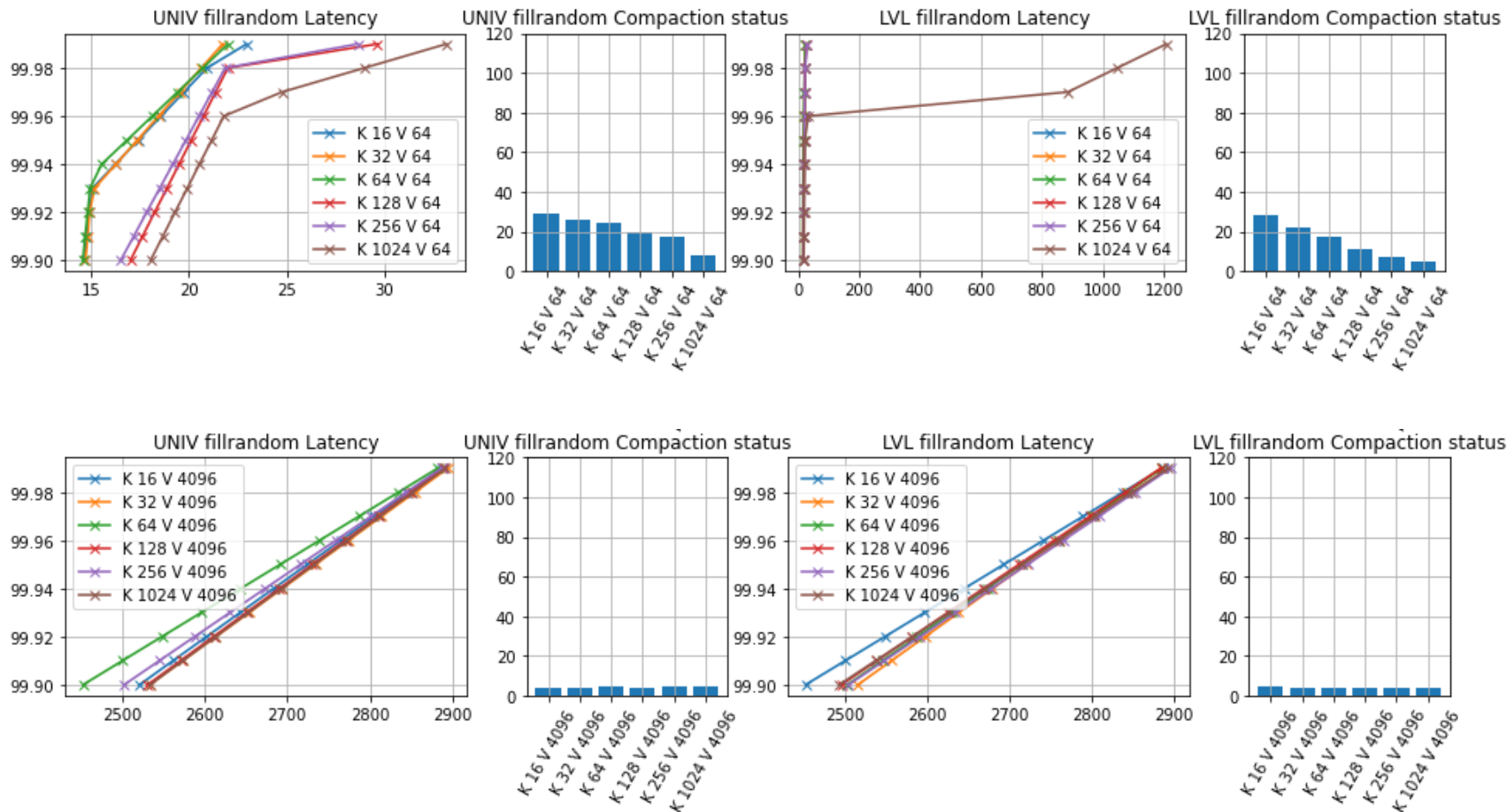


■ WAF: WAL_ON



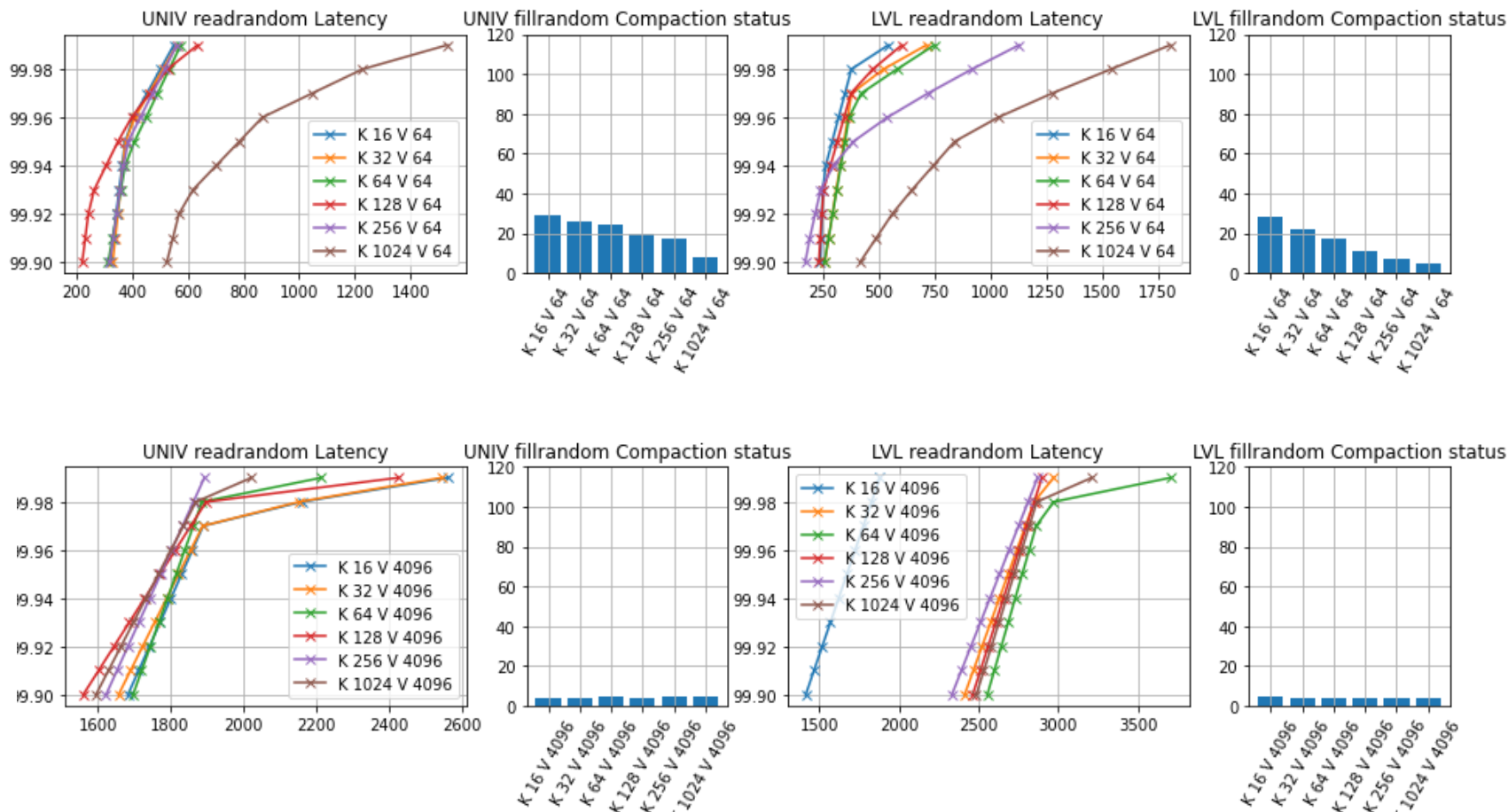
LVL vs Univ # of Compactions , latency Comparison

■ Fillrandom latency 99.99%: WAL_OFF



LVL vs Univ # of Compactions , latency Comparison

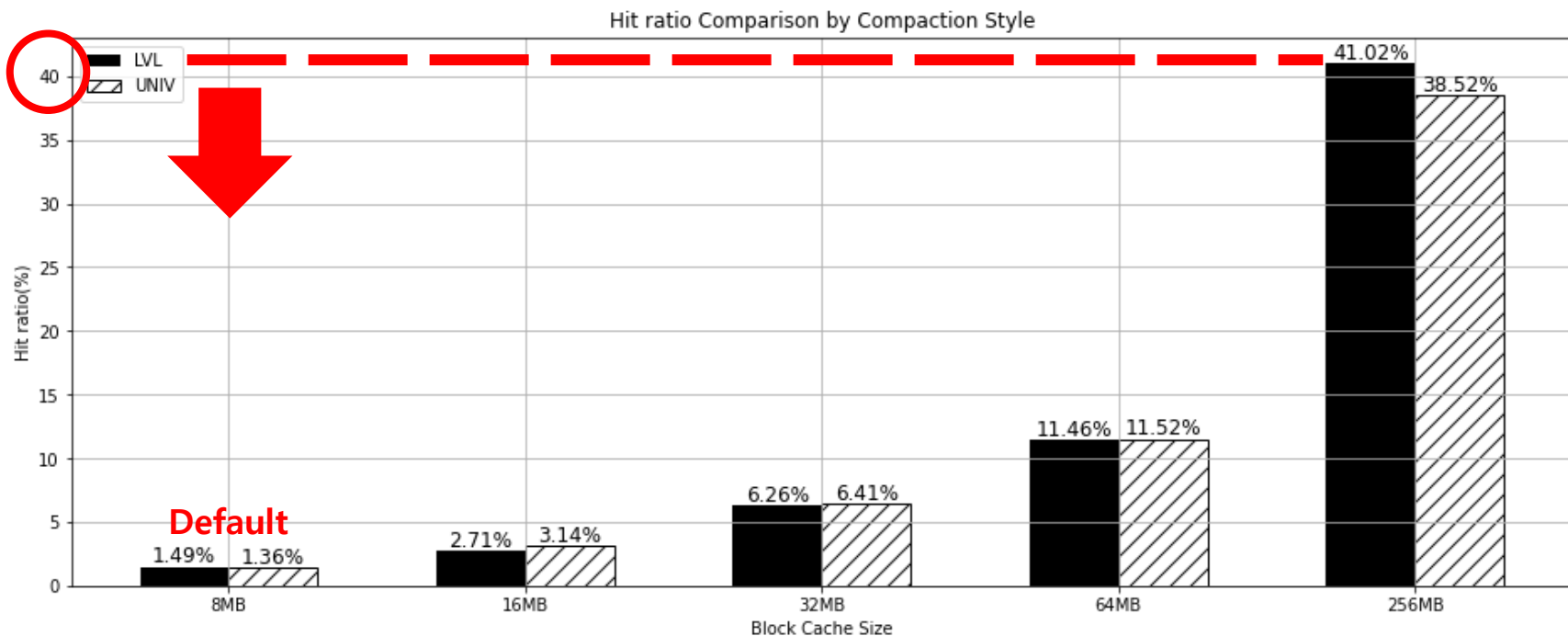
■ Readrandom latency 99.99%: WAL_OFF



LVL vs Univ Cache Hit ratio Comparison

■ Block Cache Hit ratio Comparison

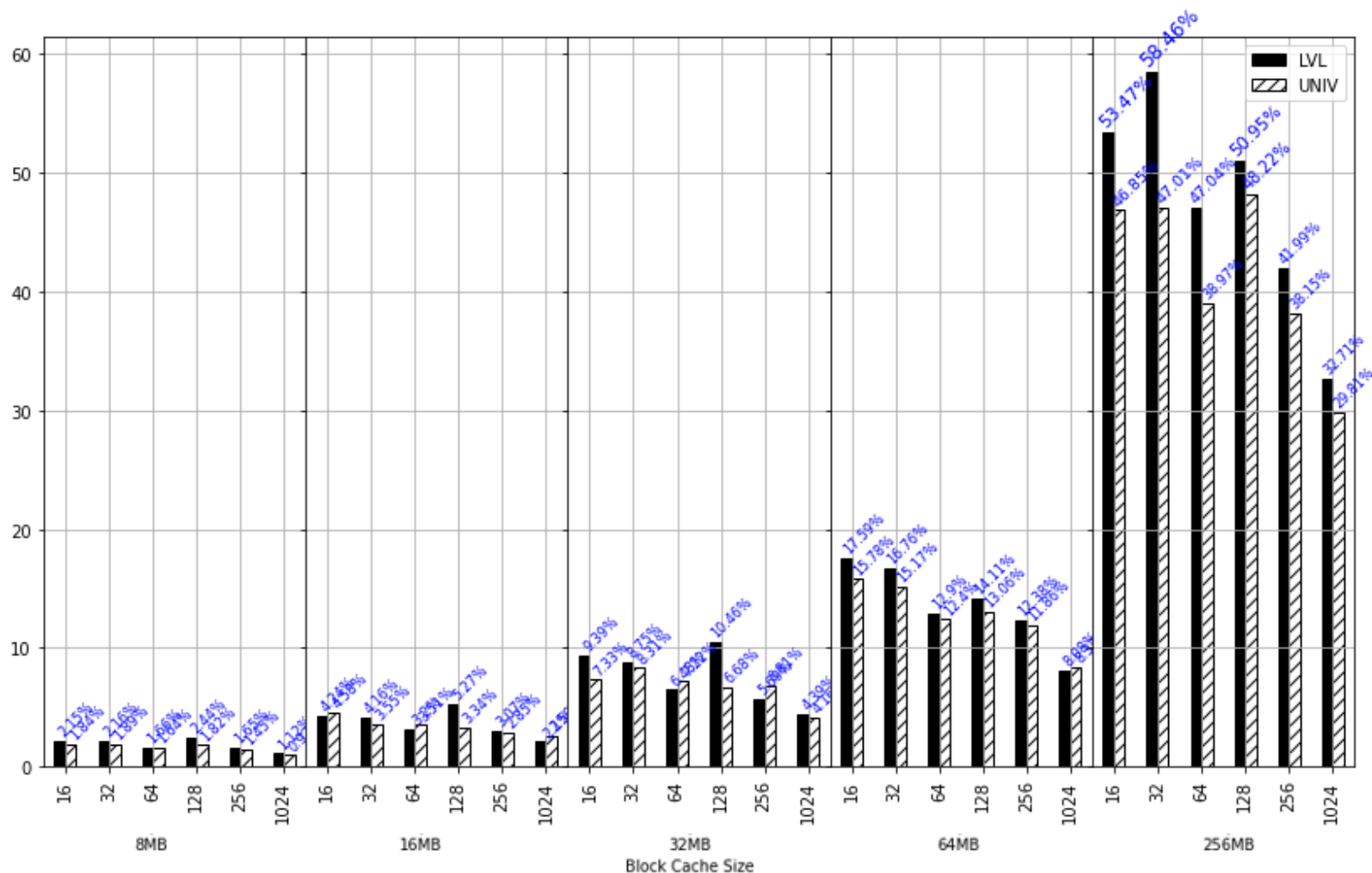
✓ 8MB, 16MB, 32MB, 64MB, 256MB → 512MB, 1GB, 2GB, 4GB 실험 중..



☞ Hit ratio is less than 50% under block cache size == 256MB

LVL vs Univ Cache Hit ratio Comparison

■ Block Cache Hit ratio Comparison



LVL vs Univ Cache Hit ratio Comparison

Readrandom

--use_existing_db

Key [16, 32, 64, 128, 256, 1024]

Value [64, 128, 256, 512, 1024, 4096]

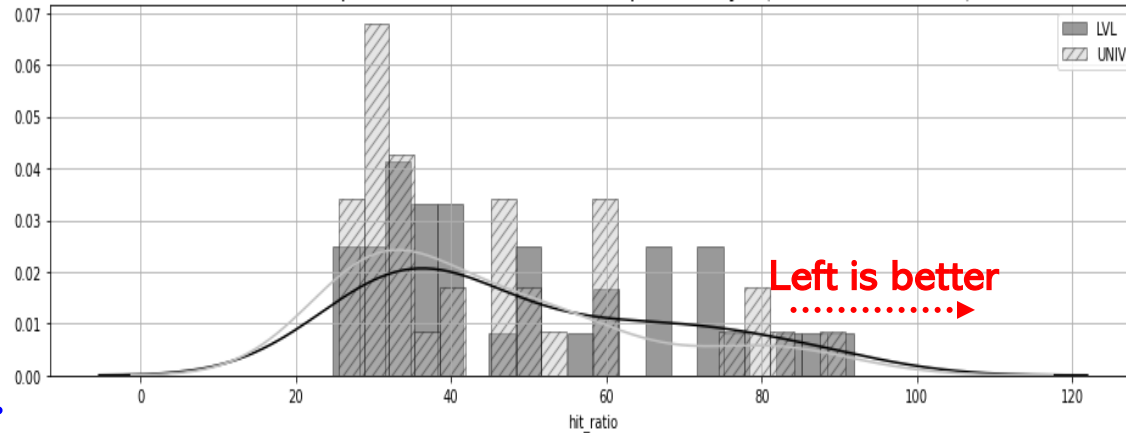
Entries 500 0000

Storage Samsung 512GB 860 Pro

File System Ext4

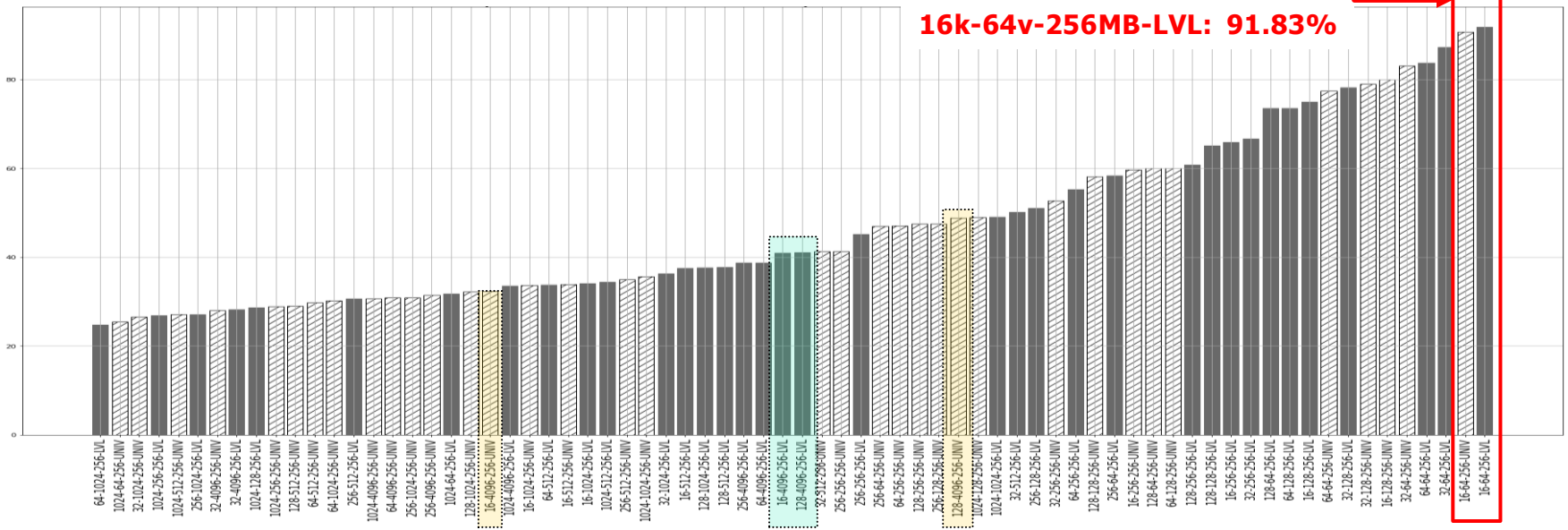
CPU Intel(R) Core(TM) i5-4440 CPU @ 3.10GHz

Hit ratio comparison between different compaction Style (Block Cache 256MB)



16k-64v-256MB-UNIV: 90.79%

16k-64v-256MB-LVL: 91.83%



Mental Model

■ Quantative Experiment

- ✓ LvL vs Univ
 - KV distribution
 - Throughput, latency, Hit ratio, QPS(queries per second)
 - SST Table Size
 - Throughput, latency
 - WAL off
 - Adjusting block cache size

■ Qualitative Experiment

- ✓ Level Compaction's weak point
 - Write Amplification
 - Write Stall

→ Methods to overcome

1~2 Week

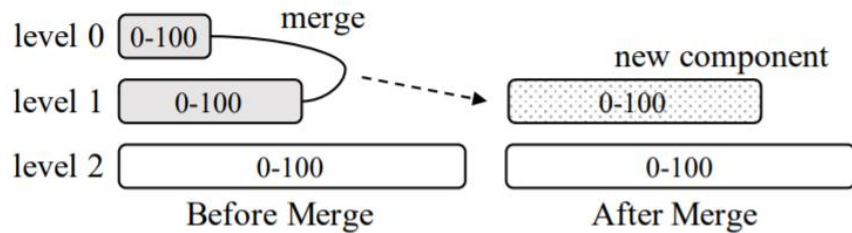
- ✓ Universal Compaction's weak point
 - Read Amplification
 - Space Amplification

→ Methods to overcome

1~2 Week

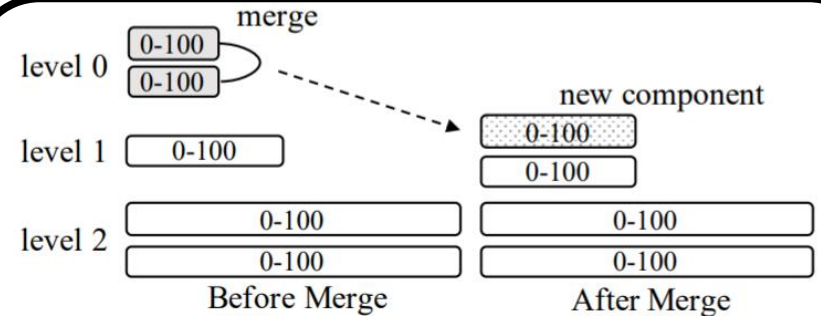
→ New Idea!

Mental Model



(a) Leveling Merge Policy: one component per level

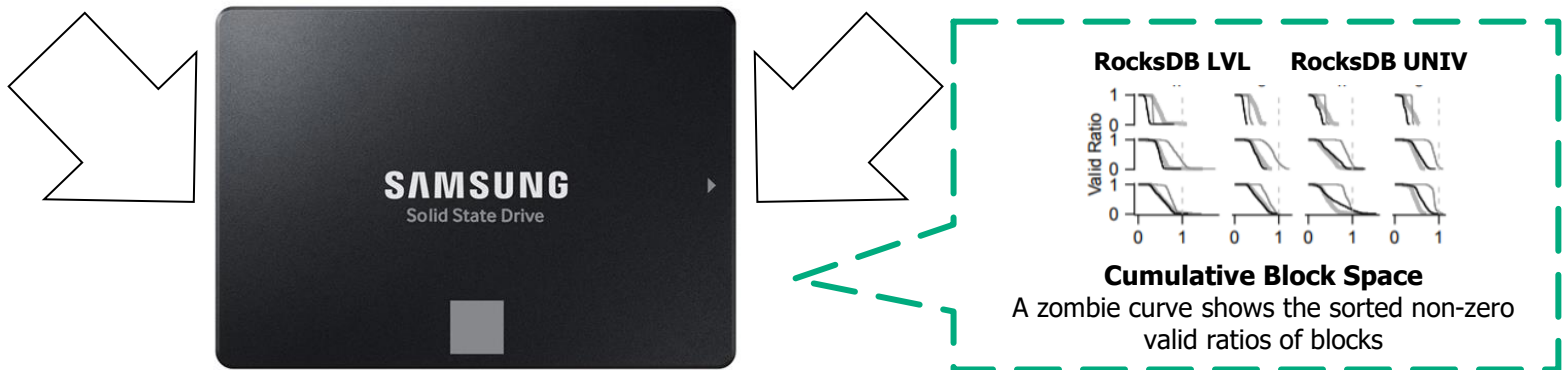
Level Compaction



(b) Tiering Merge Policy: up to T components per level

Universal Compaction

Fig: Luo, Chen, and Michael J. Carey. "LSM-based storage techniques: a survey." *The VLDB Journal* 29.1 (2020): 393-418.



Jun He, Sudarsun Kannan, Andrea C. Arpaci-Dusseau, Remzi H. Arpaci-Dusseau, The Unwritten Contract of Solid State Drives, EuroSys '17

<https://github.com/junhe/wiscsee>

Discussion



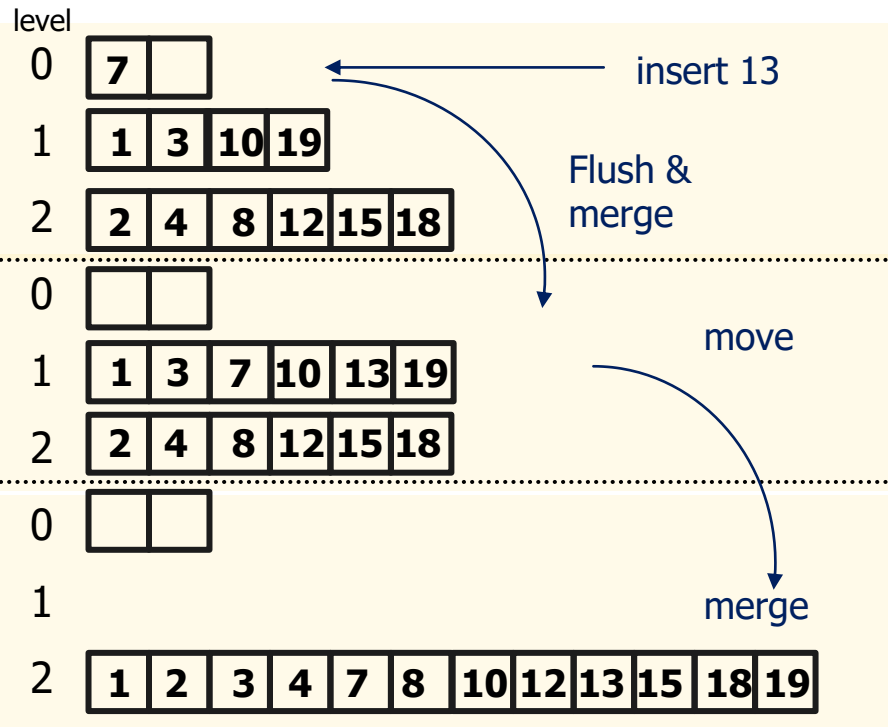
Last Week

Compaction Style

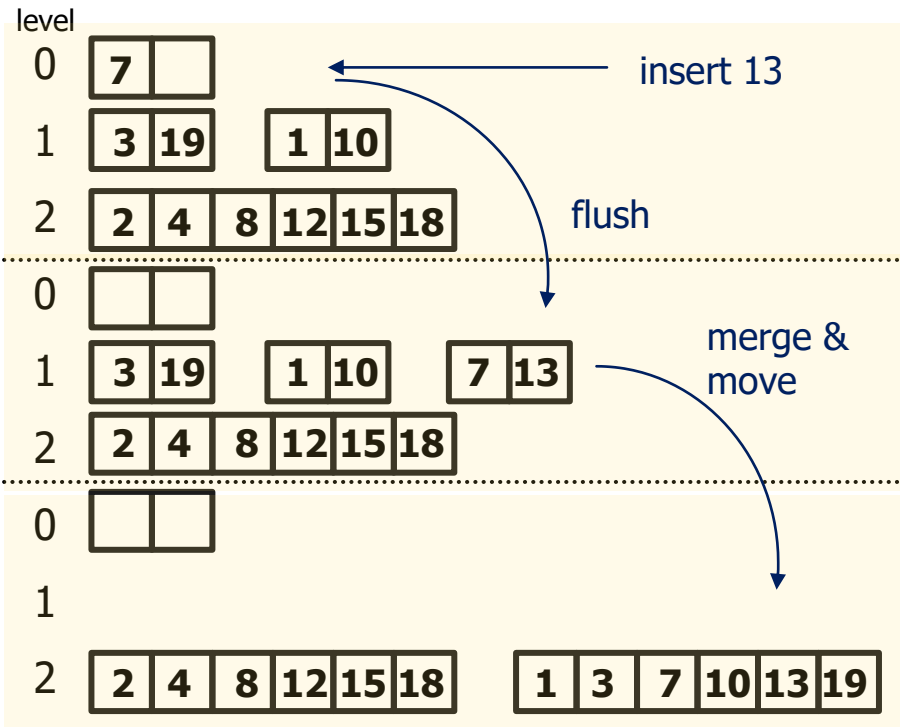
■ Leveled Compaction, Universal Compaction

✓ Sorted Level vs Sorted Run

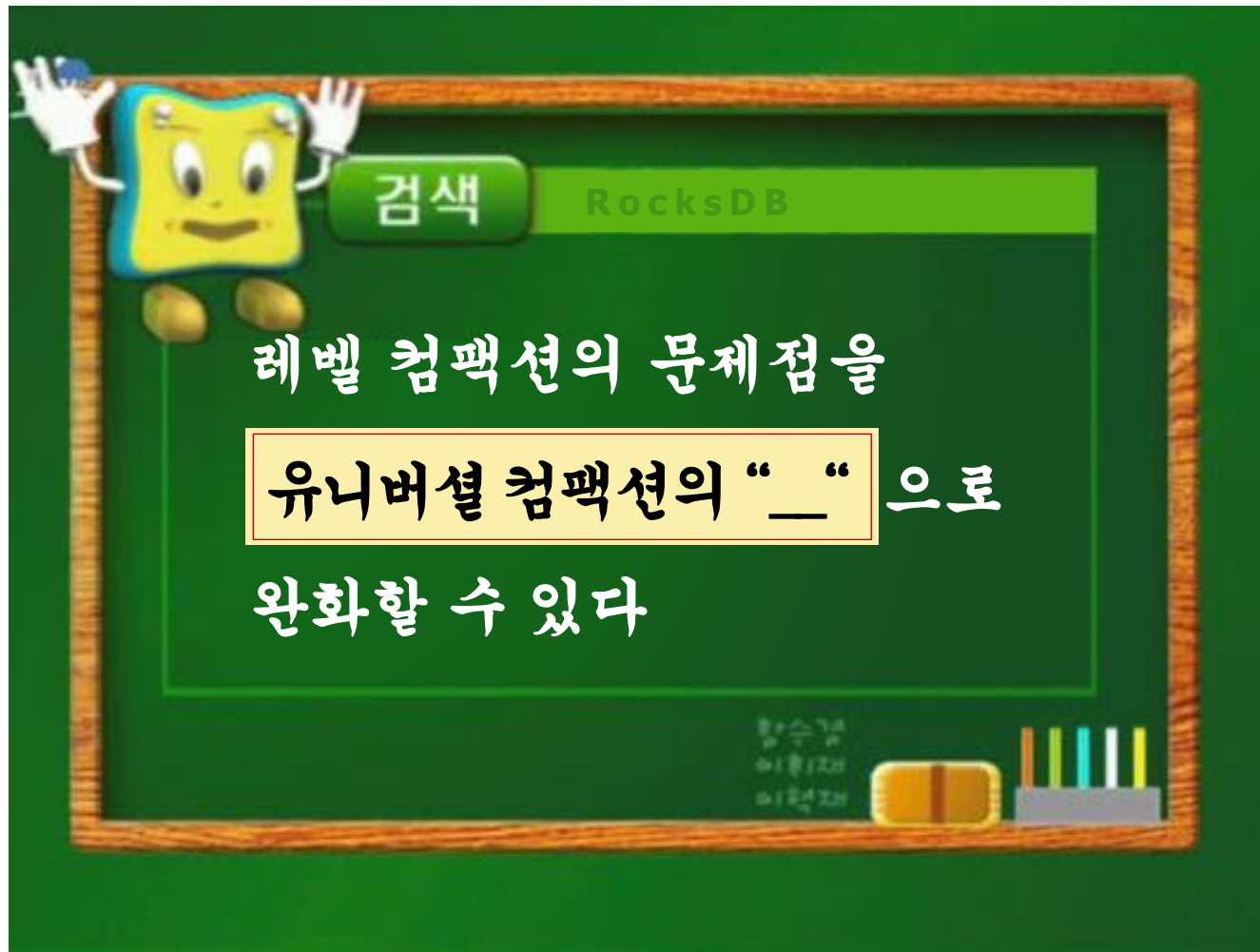
Example of LeveledCompaction



Example of Universal Compaction



Mental Model



LVL vs Univ Throughput Comparison

Fillrandom

Key [16, 32, 64, 128, 256, 1024]

Value [64, 128, 256, 512, 1024, 4096]

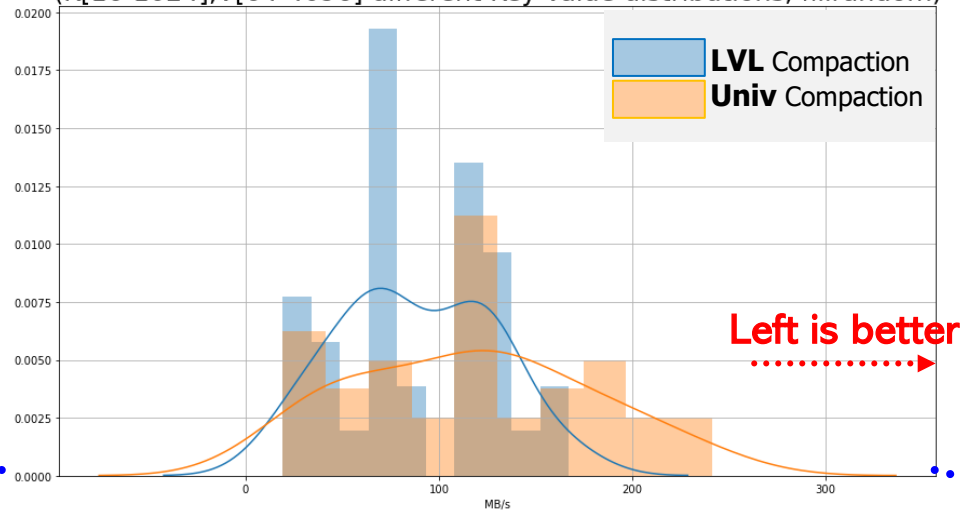
Entries 500 0000

Storage Samsung 1TB 860 Pro

File System Ext4

CPU Intel(R) Core(TM) i7-10700K CPU @ 3.80GHz

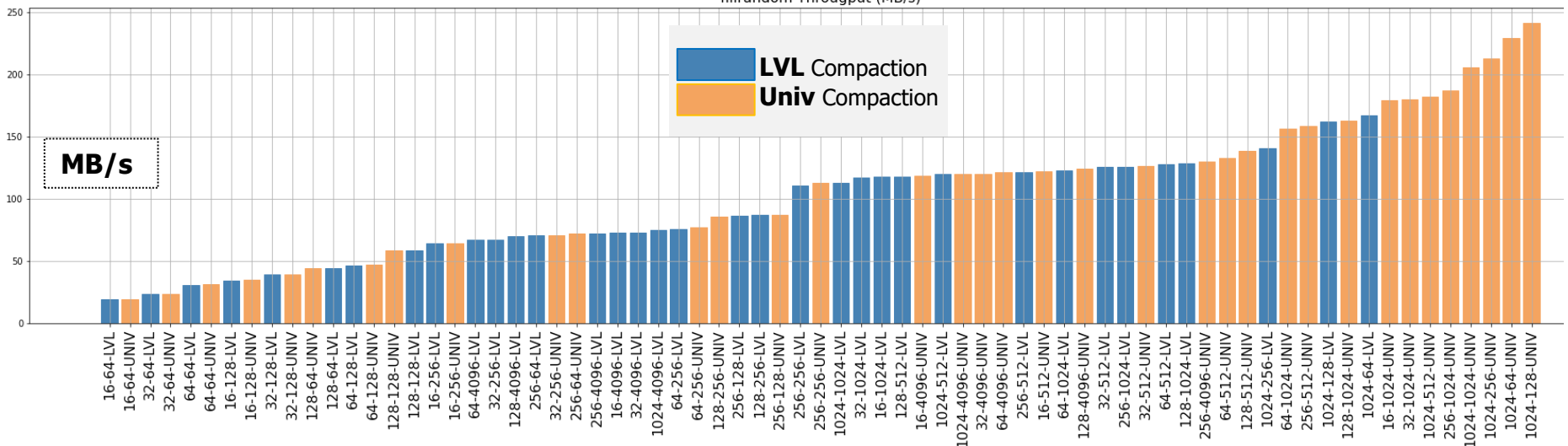
Throughput comparison
between different compaction style
(K[16-1024],V[64-4096] different Key-Value distributions, fillrandom)



fillrandom Throughput (MB/s)

MB/s

LVL Compaction
Univ Compaction



LVL vs Univ Throughput Comparison

Throughput comparison
between different compaction style

K[16-1024], V[64-4096] different Key-Value distributions(readrandom)

Readrandom

--use_existing_db

Key [16, 32, 64, 128, 256, 1024]

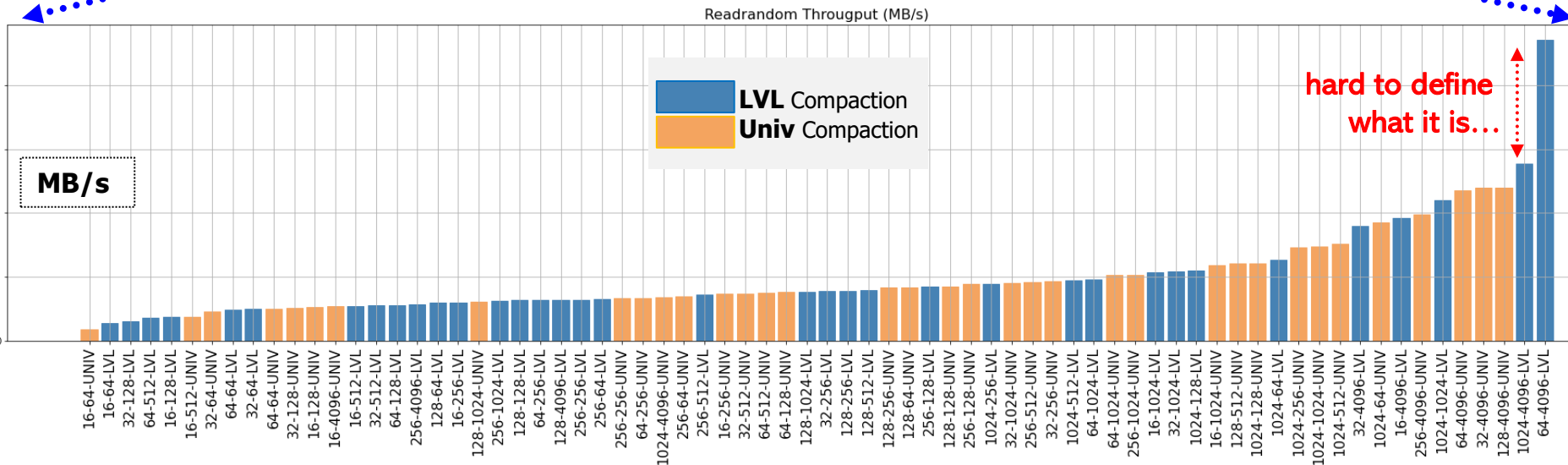
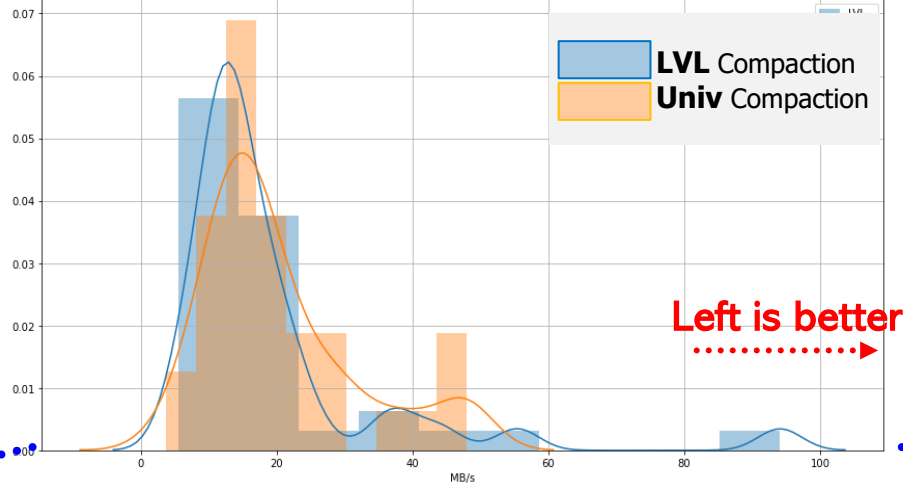
Value [64, 128, 256, 512, 1024, 4096]

Entries 500 0000

Storage Samsung 1TB 860 Pro

File System Ext4

CPU Intel(R) Core(TM) i7-10700K CPU @ 3.80GHz



LVL vs Univ Throughput Comparison

Readrandom

--use_existing_db

Key [16, 32, 64, 128, 256, 1024]

Value [64, 128, 256, 512, 1024, 4096]

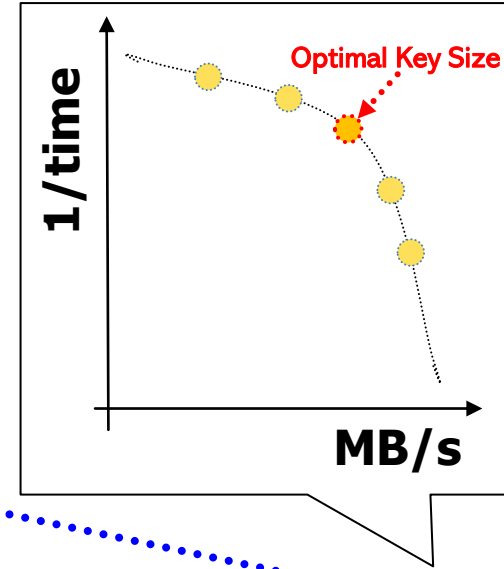
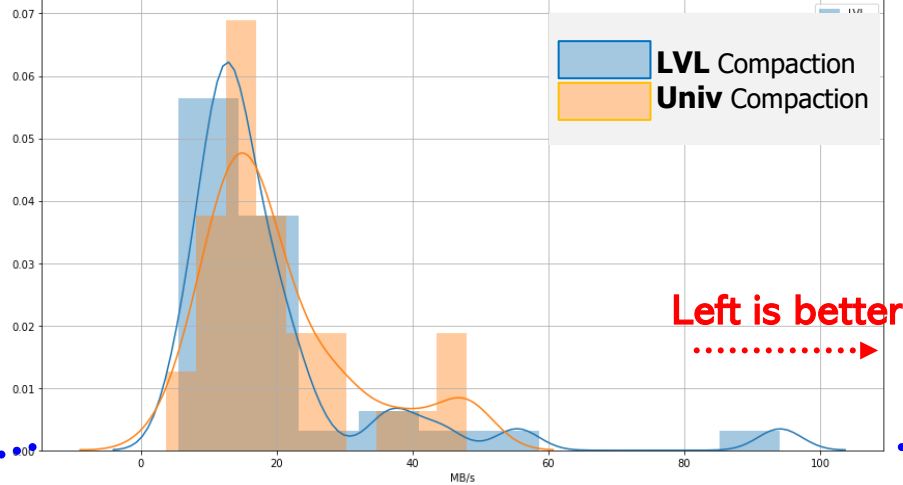
Entries 500 0000

Storage Samsung 1TB 860 Pro

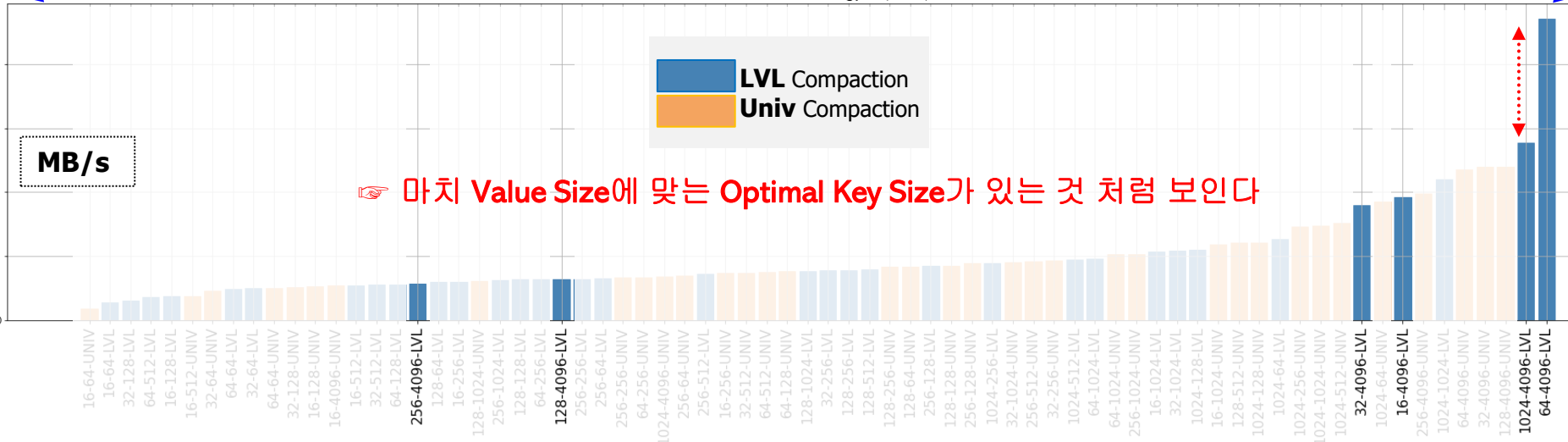
File System Ext4

CPU Intel(R) Core(TM) i7-10700K CPU @ 3.80GHz

Throughput comparison
between different compaction style
K[16-1024], V[64-4096] different Key-Value distributions(readrandom)

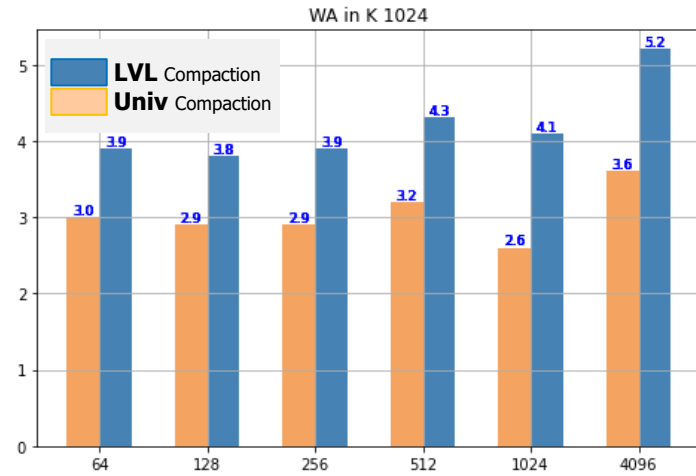
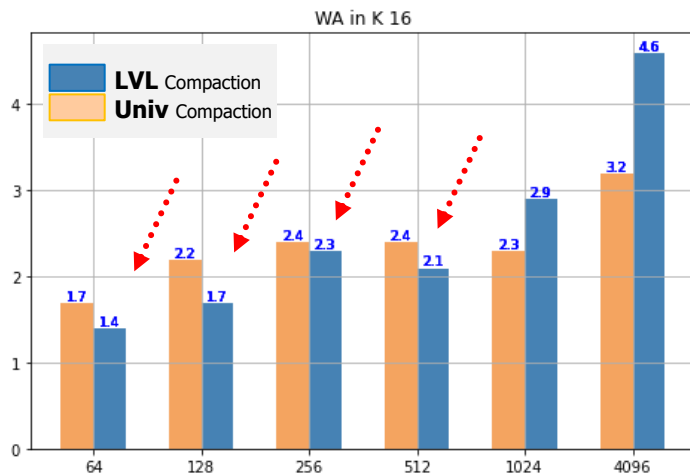
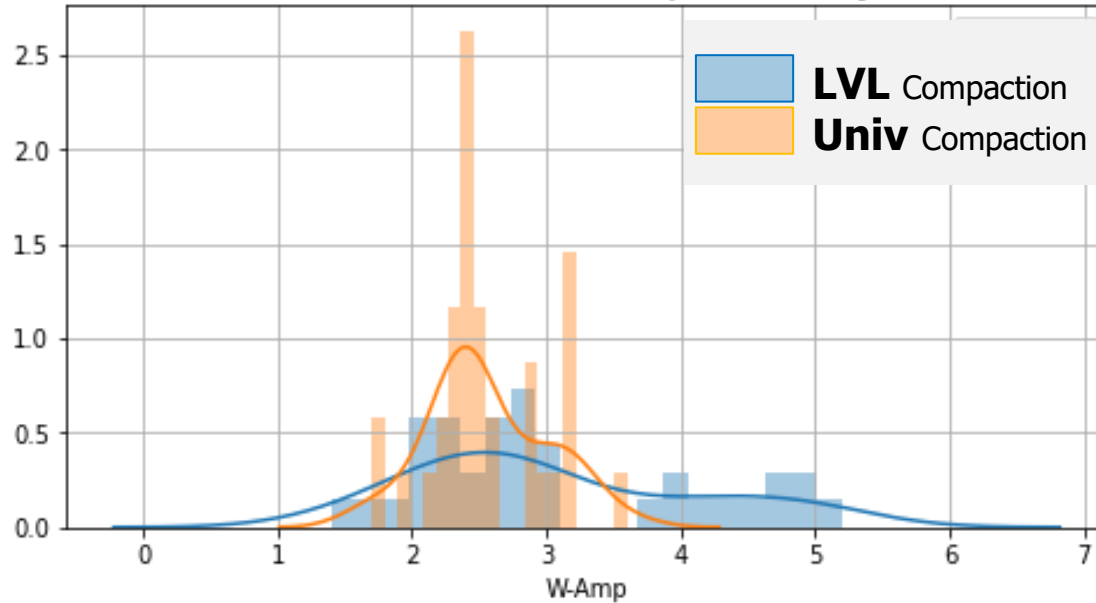


Readrandom Throughput (MB/s)



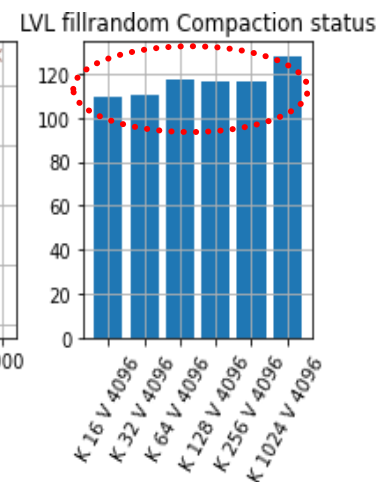
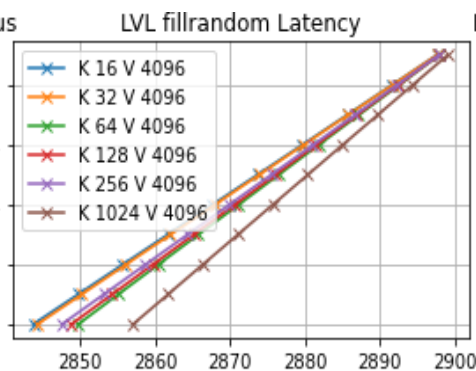
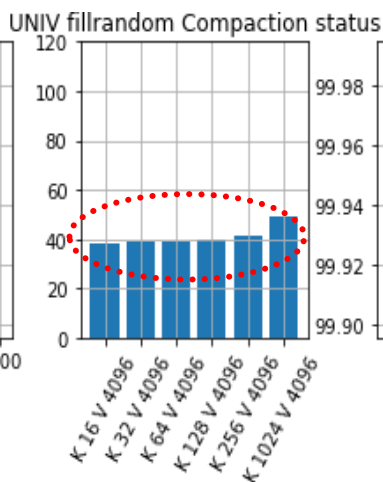
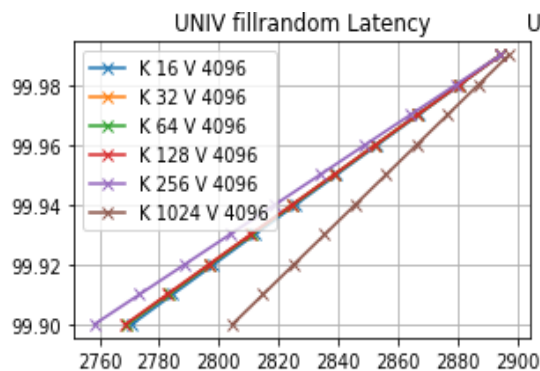
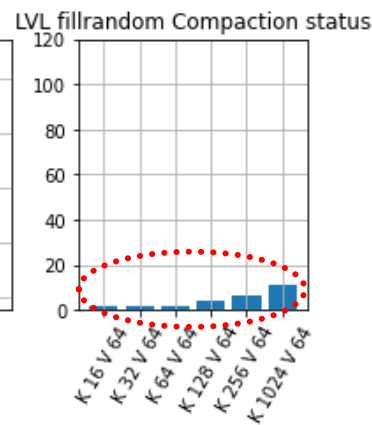
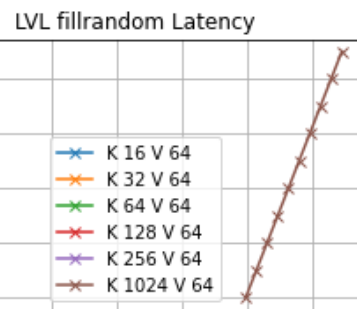
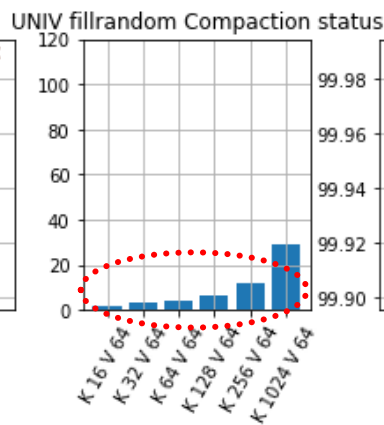
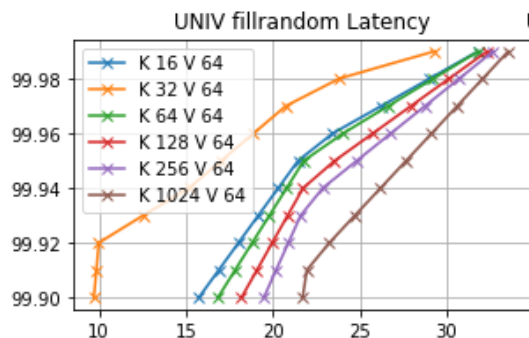
LVL vs Univ WAF Comparison

Write Amplification comparison
between different compaction Style



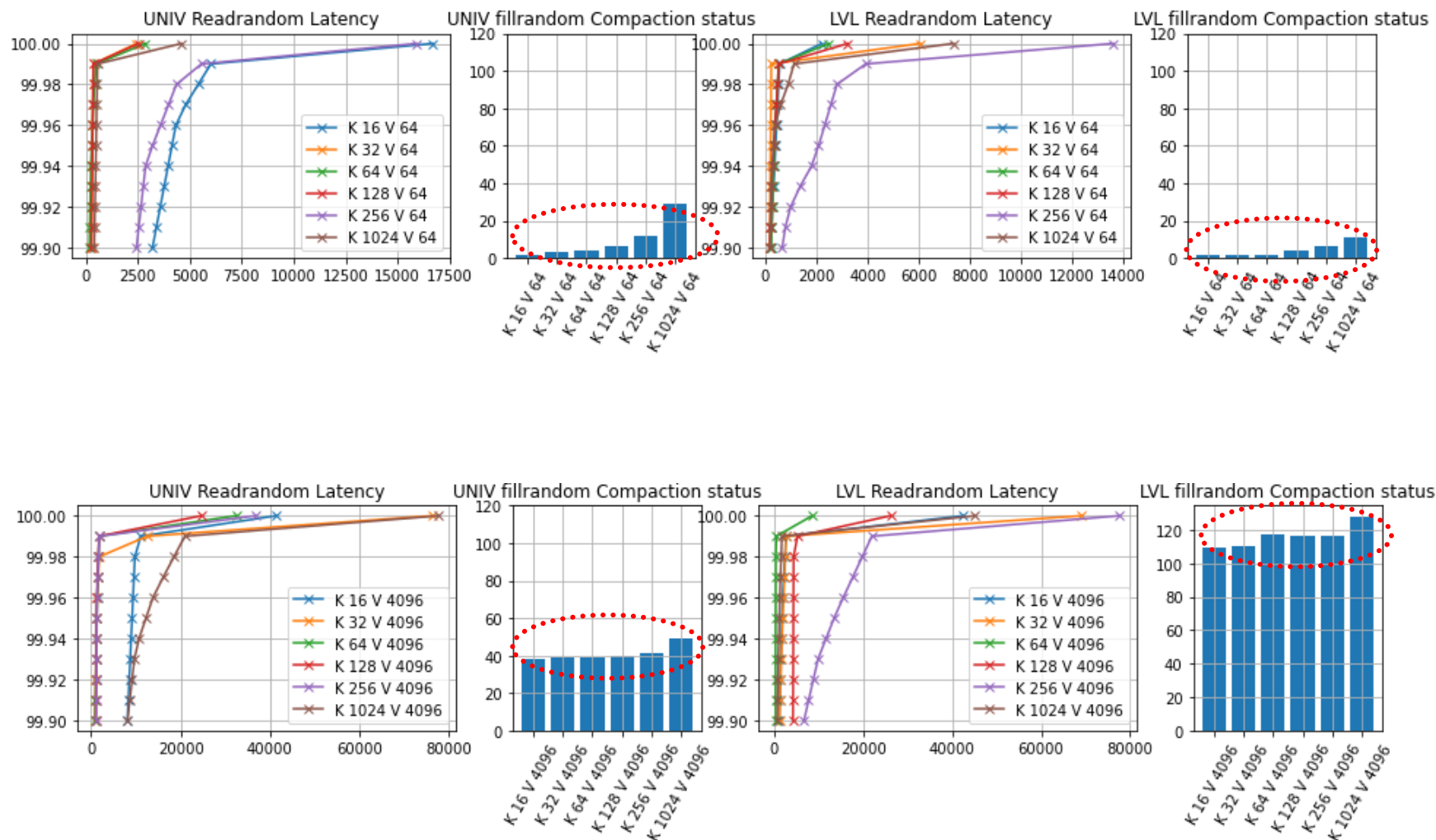
LVL vs Univ # of Compactions , latency Comparison

■ Fillrandom

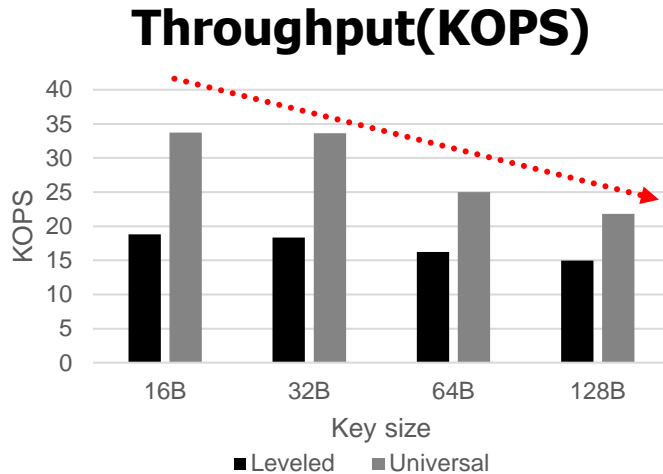


LVL vs Univ # of Compactions / latency Comparison

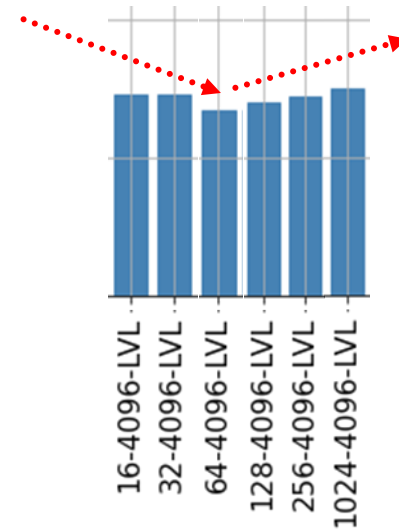
Readrandom



■ Issue on Last week



Throughput(MB/s)



☞ 한 쪽은 **OPS**, 한 쪽은 **MB/s** 임.

즉, **KV Size**가 늘어날 수록 **MB/s**는 늘어나고, **OPS**는 줄어들게 됨