

# RocksDB Festival

## RF5\_Team\_LayOut

Supported by IITP, StarLab.

August 23, 2021

Minguk Choi, Jungwon Lee, Guangxun shin

[koreachoi96@gmail.com](mailto:koreachoi96@gmail.com), [gardenlee960828@gmail.com](mailto:gardenlee960828@gmail.com), [guangxun0621@naver.com](mailto:guangxun0621@naver.com)

Docks

- 1. Performance (vs Titan)
  - ✓ 0. Workload: Fill\_random
  - ✓ 1. Throughput
  - ✓ 2. Write Amplification
  - ✓ 3. Space Amplification
  
- 2. Future Work

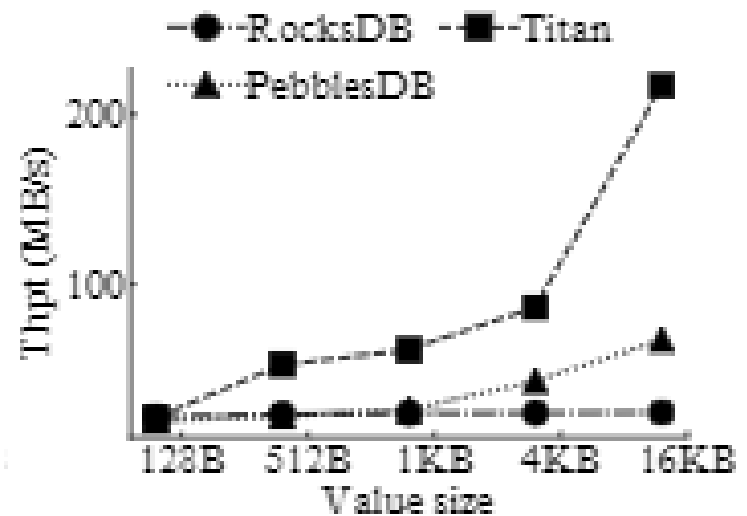
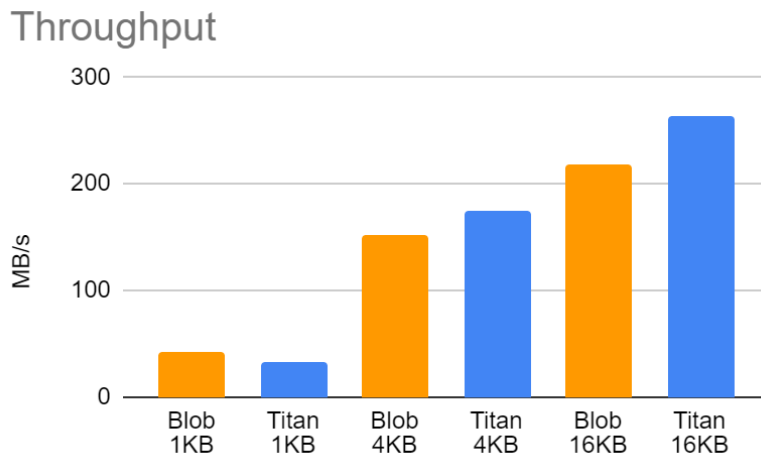
## ■ 0. Workload

- ✓ db\_bench, ldb, sst\_dump
- ✓ Fill\_random
- ✓ Key Size: 100B
- ✓ MemTable = BlobFile = 64MB(default)
- ✓ Value: 1KB/4KB/16KB
- ✓ SST\_file: 5.7MB/1.5MB/0.4MB
  - $$\text{SST}_{\text{size}} = \text{Blob}_{\text{size}} * \frac{\text{key}_{\text{size}}}{\text{key}_{\text{size}} + \text{value}_{\text{size}}}$$
- ✓ Compression: None
- ✓ Blob Garbage Collection: True
- ✓ Workload Size: 100GB

# Throughput

## ■ 1. Throughput

- ✓ Titan is slightly better than BlobDB
- ✓ Similar with DiffKV Paper

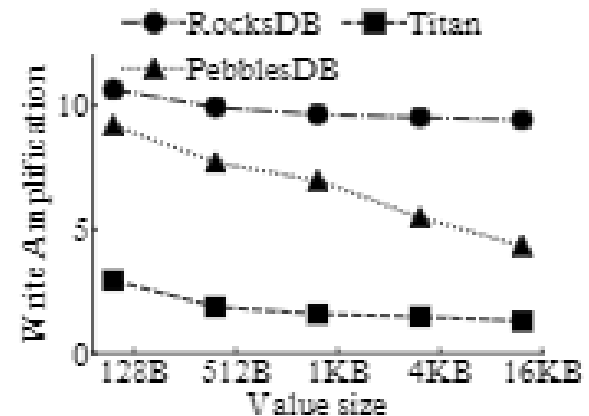
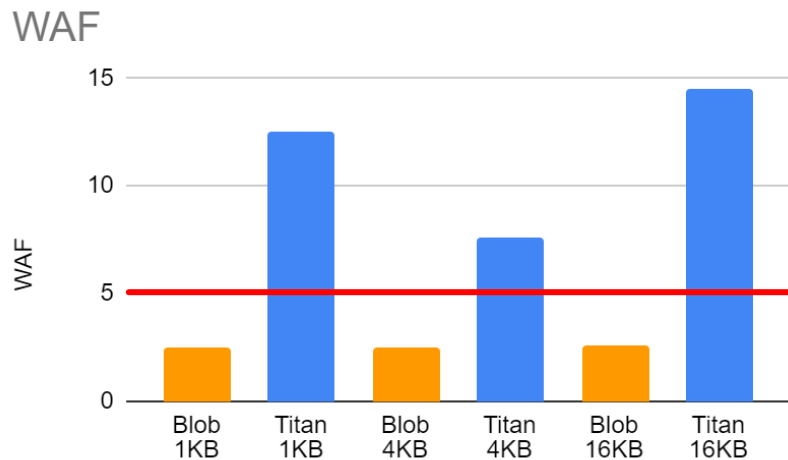


(b) Write throughput

# Write Amplification

## ■ 2. WAF

- ✓ **Blob db is Way better** than Titan
  - Blob = 2.5 / 2.5 / 2.6
  - Titan = 12.5 / 7.6 / 14.5
    - **Way worse than expectation...**
- ✓ But in Diffkv Paper, it's different
  - Titan WAF is under 5.



(a) Write amplification

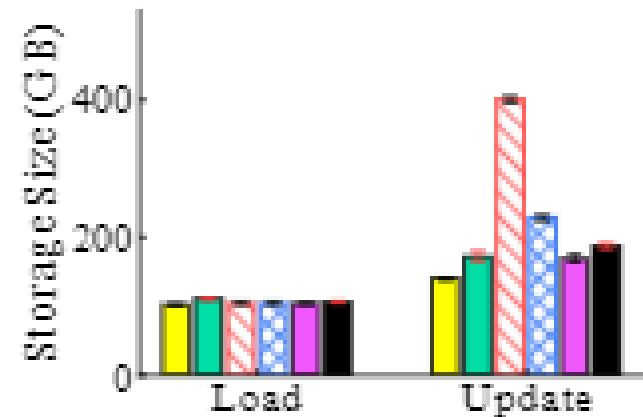
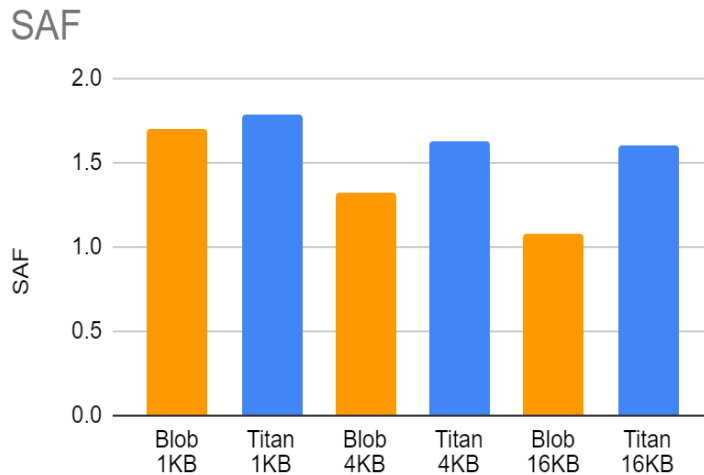
# Space Amplification

## ■ 3. SAF

$$✓ \text{ SAF} = \frac{\text{DB\_size}}{\text{Data\_size}}$$

- DB\_size = size of DB after db\_bench
- Data\_size = size of valid entries, **after entire compaction**  
= [num of entries in sst file] x [size of (key+value)] after entire compaction

✓ **BlobDB** is better than TitanDB



(c) Space usage.

Legend: RocksDB (Yellow), Titan (BG-GC) (Blue with dots), Titan (FG-GC) (Purple)

## 4. Future Work

---

### ■ 1. Performance

✓ Blob vs Titan + vs Rocks

- Write random
- Range scan

### ■ 2. Why Titan WAF is way worse than expectation

# Q & A

---

