Tutorial on LSM-trees

Database System Technology

Niv Dayan

Question 1 - picking a storage engine

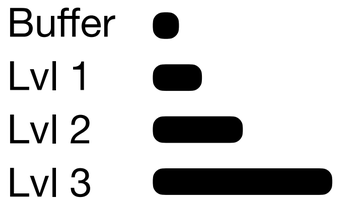
We would like to choose KV-store engine between BerkeleyDB vs. RocksDB for a workload consisting of 10% random gets and 90% random puts. BerkeleyDB uses a B-tree. RocksDB uses a leveled LSM-tree with size ratio T=10 and a buffer size of $P=2^{26}$ entries. Assume $N=2^{40}$ and $B=2^{7}$. All internal nodes fit in memory. We are using a disk drive. What is your choice and why?

ORACLE
BERKELEY DB

RocksDB

Question 2 - tuning an LSM-tree

A friend tells you they switched from using a B-tree to a basic LSM-tree (with size ratio 2), yet write-amplification actually increased. There are $N=2^{40}$ entries, and the memtable size P and block size B are both $B=P=2^5$ entries. Explain why this happened. Identify three tuning options for reducing write-amplification and the trade-off of each one of them.



Question 3 - clustered vs. unclustered LSM-trees

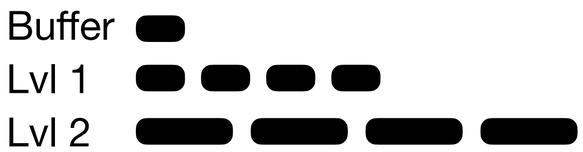
In a clustered LSM-trees, the values are stored within the LSM-tree alongside their keys. Another approach is an unclustered LSM-tree, which stores values in an append-only file and indexes them using key-pointer pairs from within the LSM-tree. What's the impact of clustered vs. unclustered LSM-trees on put/get/scan performance. Propose adjusted cost models. Assume a basic LSM-tree (size ratio T=2), insertions only (no deletes or updates), and a 1 page memtable. Let K be the number of key-pointer pairs fitting into a page, and let B be the number of key-value pairs fitting in a page. Based on your models, identify cases where each of the two approaches shines.

Question 4 - space-amplification

An LSM-tree can store multiple versions for a given entry, where only the most recent is considered up-to-date and the rest are obsolete. The obsolete entries are eventually discarded during compaction, but meanwhile they consume space. This phenomenon is known as space-amplification, defined as: (physical space taken up) / (logical data size). Quantify space-worst-case amplification for a leveled vs. tiered LSM-tree with size ratio T between any two adjacent levels. Assume all levels are totally full and all entries are equally sized.

e.g., T=4

Tiered



Leveled