

Research Papers related to LSM-trees

Background

LSM-based Storage Techniques: A Survey

Evolution of Development Priorities in Key-value Stores Serving Large-scale Applications:
The RocksDB Experience

Characterizing, Modeling, and Benchmarking RocksDB Key-Value Workloads at Facebook

P1: HyperClockCache vs. LRUCache

FIFO Queues Are All You Need for Cache Eviction

P2: MemTable implementations

Breaking Down Memory Walls: Adaptive Memory Management in LSM-based Storage Systems

P3: LSM caching systems

AC-Key: Adaptive Caching for LSM-based Key-Value Stores

Leaper: A Learned Prefetcher for Cache Invalidation in LSM-tree based Storage Engines

Calcspar: A Contract-Aware LSM Store for Cloud Storage with Low Latency Spikes

SAS-Cache: A Semantic-Aware Secondary Cache for LSM-based Key-Value Stores

Range Cache: An Efficient Cache Component for Accelerating Range Queries on LSM-Based Key-Value Stores

AdCache: Adaptive Cache Management with Admission Control for LSM-tree Key-Value Stores

Improving Range Scan Performance in LSM-trees with Group Caching

P4: Filters

Monkey: Optimal Navigable Key-Value Store

Mnemosyne: Dynamic Workload-Aware BF Tuning via Accurate Statistics in LSM trees

ElasticBF: Elastic Bloom Filter with Hotness Awareness for Boosting Read Performance in Large Key-Value Stores

Cuckoo Filter: Practically Better Than Bloom

Ribbon Filter: Practically Smaller Than Bloom and Xor

SuRF: Practical Range Query Filtering with Fast Succinct Tries

Rosetta: A Robust Space-Time Optimized Range Filter for Key-Value Stores

SNARF: A Learning-Enhanced Range Filter

bloomRF: On Performing Range-Queries in Bloom-Filters with Piecewise-Monotone Hash Functions and Prefix Hashing