ARIA
APPLIED RESEARCH IN ACTION

IOMemTable — Memory-Optimized Index-Organized Table for LSM-Tree Storage

Toward a better trade-off of index performance for Hybrid Transactional / Analytical Processing (HTAP) workloads

Deyu Liu and Duy Tran

Qizhen Zhang and Niv Dayan
ACADEMIC SUPERVISORS

Kelvin Ho
INDUSTRY SUPERVISOR

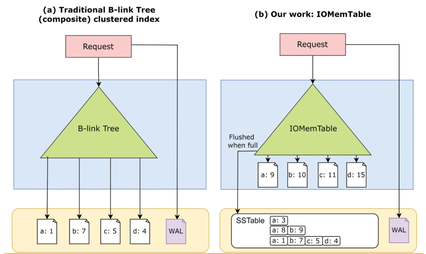

Figure 1: Overview of IOMemTable design as compared to a traditional index.

Unlike a traditional B-Tree composite clustered index, IOMemTable keeps all indexed and non-indexed attributes at the leaf nodes of the B-Link tree, enabling faster inserts, efficient point and range query of hot data, and future extension to an log-structured merge tree design (on-disk components not yet realized).

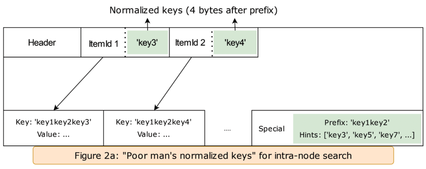

Figure 2a: "Poor man's normalized keys" for intra-node search

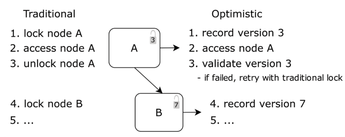

Figure 2b: Comparison of a down-link traversal in a B-Link tree using traditional latching vs. optimistic latching

PROJECT SUMMARY

Our project implements IOMemTable, a memory-optimized Index-Organized Table (IOT) [1] in Huawei openGauss database management system. IOT is a disk-based ACID-compliant storage option, essentially a clustered composite B-Link Tree index storing all attribute data in leaf nodes. We explore IOMemTable's potential to be the "level 0" data structure for a log-structured merge-trees [2] storage engine. Building on IOT allows us to maintain strong transactional guarantees, achieve decent write throughput and fast read performance of hot data required for HTAP workloads, and explore trade-offs different from existing LSM tree implementations.

We first implemented (1) a dedicated memory arena for full control of buffer pool policy, and (2) pointer swizzling [3] to eliminate buffer-to-block mapping overhead, among other required changes for transactional semantics. These changes only yielded an 8% improvement in TPC-C results compared to highly-optimized IOT baselines, and two primary bottlenecks emerged: slow intra-node search and node locking overhead.

Inspired by in-memory order-preserving data structures like Masstree [4], we focus on two optimizations: (1) speeding up intra-node search by storing normalized keys in a cache-friendly manner, while using a hint array to narrow down search range [5], and (2) improving concurrency with optimistic latching [6]. Micro-benchmarking shows a 1.25x throughput improvement from intra-node search optimizations in a single-threaded point query workload, and optimistic latching achieves a 2.29x improvement for a 20-threaded point query workload. Currently at about 50% of Masstree's performance, we continue to optimize IOMemTable, aiming to be competitive with its performance on insert, point, and range queries.

REFERENCES

[1] Jagannathan Srinivasan, Souripriya Das, Chuck Freiwald, Eugene Inseok Chong, Mahesh Jagannath, Aravind Yalamanchi, Ramkumar Krishnan, Anh-Tuan Tran, Samuel DeFazio, and Jayanta Banerjee. 2000. Oracle8i Index-Organized Table and Its Application to New Domains. In VLDB 2000, Proceedings of 26th International Conference on Very Large Data Bases, September 10-14, 2000, Cairo, Egypt, Morgan Kaufmann, 285–296.

[2] Patrick E. O'Neil, Edward Cheng, Dieter Gawlick, and Elizabeth J. O'Neil. 1996. The Log-Structured Merge-Tree (LSM-Tree). Acta Informatica 33, 4 (1996), 351–385.

[3] Goetz Graefe, Haris Volos, Hideaki Kimura, Harumi A. Kuno, Joseph Tucek, Mark Lillibridge, and Alistair C. Veitch. 2014. In-Memory Performance for Big Data. Proc. VLDB Endow. 8, 1 (2014), 37–48.

[4] Yandong Mao, Eddie Kohler, and Robert Tappan Morris. 2012. Cache craftiness for fast multicore key-value storage. In European Conference on Computer Systems, Proceedings of the Seventh EuroSys Conference 2012, EuroSys '12, Bern, Switzerland, April 10-13, 2012, ACM, 183–196.

[5] Goetz Graefe. 2011. Modern B-Tree Techniques. Found. Trends Databases 3, 4 (2011), 203–402.

[6] Jan Böttcher, Viktor Leis, Jana Giceva, Thomas Neumann, and Alfons Kemper. 2020. Scalable and robust latches for database systems. In 16th International Workshop on Data Management on New Hardware, DaMoN 2020, Portland, Oregon, USA, June 15, 2020, ACM, 2:1-2:8.

HUAWEI

Computer Science
UNIVERSITY OF TORONTO
Master of Science in Applied Computing