CAS CS 561: Data Systems Architectures
Data-intensive Systems and Computing Lab
Department of Computer Science
College of Arts and Sciences, Boston University
http://bu-disc.github.io/CS561/

CS561 Spring 2021 - Research Project

Title: *Benchmarking Dual B+-trees for Near-Sorted Workloads*

Background: B+-tree is an index data structure that stores data pointers only in leaf nodes, and only pivot pointers in the internal nodes. Additionally, the leaf nodes are also linked to provide ordered access to the records.

Problem: A general purpose of an index is to bring "order" to the data, for easier access. Hence, for an already ordered/nearly-ordered workload, an index is expected to perform the least amount of work required. B+-trees however, do not specially leverage existing "orderness" in the data, and perform the same insert procedure as in a general case. To increase the speed of inserts, we explore the idea of maintaining two B+-trees – while inserting an element, if it is in-order with respect to the tree, will be inserted in the first tree, else will be inserted into the second tree. There are a few obvious drawbacks of course, for example, a query will have to now scan two indexes rather than one, so will be more expensive.

Objective: The objective of the project is to benchmark the dual-index data structure against differently sorted workloads. The workflow for this is as the following.

(a) Implement a simple dual-tree data structure – One tree inserts elements if in order, and the other inserts out-of-order elements.
(b) A simple in-memory version of the B+-tree will be given to you. Improvements to the B+-tree are encouraged. It is required to identify the modifications or additions required to support the dual-tree system (for example, a way to recognize elements if sorted or not).
(c) Benchmark the performance for inserts and point queries, differently sorted workloads from the generator provided in Project0.

For more details, contact Aneesh Raman during the office hours or by email.

[1] S. Ben-Moshe, Y. Kanza, E. Fischer, A. Matsliah, M. Fischer, and C. Staelin. **Detecting and Exploiting Near-Sortedness for Efficient Relational Query Evaluation**. In Proceedings of the International Conference on Database Theory (ICDT), pages 256–267, 2011