Wenkang Xu

wenymedia@gmail.com | terminal.im | WenyXu

Experiences

Google Summer of Code 2022 March 2022 – Fall 2022

Casbin Mentor Casbin-Mesh/neo

- Leads the development of **Casbin Neo**(new engine option), a Casbin-compatible engine.
- Introduces the Synchronization Adaptive Radix Tree, 2-4x faster than BadgerDB's lock-free version Skip List, and 4-6x faster than Std Sync Map.
- Builds the all new expression evaluation system, 2-8x faster than the origin dependency.
- Introduces the MVVC(Multi-Version Timestamp Ordering) Index

Open-Source Contributions

Senrok/yadal Fall 2022 - Now

Yet Another Data Access Layer: Accessing S3, POSIX in the same way. senrok/yadal

- Leads the development of the **yadal** Project, deeply inspired Databend's OpenDAL and Databricks's Delta lake paper.

Casbin Community September 2020 – Fall 2022

Casbin-Mesh Maintainer Casbin-Mesh

- Leads the development of **Casbin Mesh**, A scalable authorization layer built on Casbin and Raft consensus algorithm.

Personal Projects WenyXu

- **sync-adaptive-radix-tree** : An implementation of the Adaptive Radix Tree with Optimistic Lock Coupling.

Projects

PingCAP's Talent-plan/TinySQL July 2022 – September 2022

Minimum Viable Product of TiDB

- Completes Join table SQL parser.
- Implemented the Online Schema Change based on Google F1 paper.
- Introduces Count-Min Sketch.
- Implemented the predicate pushdown algorithm for aggregate expression (System-R like optimizer operator).
- Implemented the dynamic programming version join reordering algorithm.
- Implemented the vectorization for several executors.
- Implemented the parallel Hash join & aggregate algorithm.
- Implemented the client-side two-phase commit protocol based on Percolator paper.

CMU Database Group's Bustub January 2022 – March 2022

CMU 15-445 Course Project

- Implemented the buffer pool manager based on the LRU replacement mechanism.
- Implemented the Extendible Hash table.
- Implemented the iterator model exectors.
- Implemented the concurrecy control based on lock manager.

PingCAP's Talent-plan/TinyKV November 2021 – January 2022

A superset of MIT 6.824 Course Raft-KV Project

- Implemented a standalone storage engine.
- Implemented Raft Leader election, Log replication, Leader transfer, Configration changing, Peer changing algorithms.
- Implemented the KV Store based on Multi-Raft.
- Implemented the server-side two-phase commit protocol based on Percolator paper.

Skills

Programming Languages: Golang, Node.js, C++, Rust, Python
Tech Skills: Key-Value Storage Systems, Distributed Systems