CS648 Project(Winter 2020): RocksDB Row-store

Adithya Murali

Student ID: 20814040

Summary

This project implements row-store using RocksDB where each row in a relation is stored separately instead of being stored as a bunch of rows in the form of a buffer. This project is built on top Assignment 2.

Implementation Details

This project was implemented by modifying the storage engine used in Assignment 2. Relations beginning with “rdb_” are stored using row-store method. Relations are stored normally otherwise.

Each relation is stored in a separate RocksDB database. The following meta-data about each relation is also stored on the same database.

- NUM_BLOCK – Stores the number of blocks in the relation. Initialized to 0 when a relation is created.
- “%dB” – Stores the number of tuples in the given block. For example, if block 10 of a relation contains 100 tuples, the key is “10B” and its value is 100.

Each tuple is then stored as follows:

- 10^{th} tuple of 42^{nd} block is stored with key as “10B42” and its value being the tuple’s attributes stored in the same way as it was stored on the buffer. HeapTupleHeader structure contains the meta-data(t_ctid) regarding the block and the offset of a given tuple.

Each tuple is read as follows:

- The input buffer is initialized as a Page. Each row is then read from the required block and added to the page using PageAddItem() function. The number of rows in that block is stored as meta-data as mentioned above.

Rbd.c and smgr.c files were modified to implement the above changes. The code has been commented to explain its workings (Search for CS648).