CS 2550 – PRINCIPLES OF DATABASE SYSTEMS (SPRING 2015)
DEPT. OF COMPUTER SCIENCE, UNIVERSITY OF PITTSBURGH

Term Project: *myTRC*

Release: Feb. 12, 2015 Due: 11:59 PM, Feb. 26, 2015.

Goal

The objective of the project is to develop the Transactional Row Column store *myTRC* that efficiently supports concurrent execution of OLTP (i.e., transactions) and OLAP (i.e., aggregate queries) workloads. *myTRC* will provide limited *transactional* support. By limited support we mean that it does not support full durability. In addition to serializable and atomic access, it will also provide the standard uncontrolled access to files.

Description

myTRC will consist of three components: *Data Manager, Scheduler,* and *Transaction Manager*. Likewise, you will have a separate deadline to complete each component. In the first phase (i.e., this assignment) you are required to implement only the Data Manager.

Data Manager

For this project, all records (tuples) have the following fields (schema):

- ID: 4-byte integer (Primary Key)
- ClientName: 16-byte long string
- Phone: 12-byte long string

You are responsible to implement the following operations for the first phase:

- R table val: Retrieve the record with ID=val in table. If table does not exist, the read is aborted.
- *M table val*: Retrieve the record(s) which have val as area code in Phone attribute in table. If table does not exist, the read is aborted.
- *G table val*: Counts the number of customers which have val as area code in Phone attribute in table. If table does not exist, the group-by-count is aborted.
- *I table (t)*: Insert the new record t into table. If table does not exist, it is created.
- *D table*: Delete table.

The operations R, I and D should be performed on the row store, and operations M and G on the column store. All operations should be called from a script file. An example of a script file (assuming we have tables X and Y):

```
-----------------------------------------------
     R X 13
     R Y 7
     I Y (5, John, 412-111-2222)
     M Y 609
     I X (2, Thalia, 412-656-2212)
     G X 412
     D Y
-----------------------------------------------
```

Disk Organization

Records are kept on persistent storage (disk) in *directed files* in a *columnar* fashion, meaning that for each attribute of a table is kept in a separate file. Thus, you will have to keep separate files for ID, ClientName, and Phone of a table and separate sets of files for each table. Each directed file consists of slotted pages of size 512 bytes each. Every inserted tuple is hashed based on its primary key using hash function h(x) = x mod 16. Conflicts are resolved with chaining by appending new slotted pages at the end of the file as necessary.

Main Memory Organization

The records will have to be brought to main memory (database buffer) to be accessed. However, the number of buffer pages available in database buffer is limited and should be specified at the beginning of each execution. You are required to implement Least Recently Used (LRU) page replacement mechanism to swap pages in and out.

The available database buffer space in should be equally divided. The first half should be used to mirror the data using row store, and the second half to mirror the data using column store. Assume page size equal to 512 bytes. Recall, the operations R, I and D should be performed on the row store, and operations M and G on the column store. All updates made to the row store should be propagated to the disk when corresponding pages from database buffer get swapped out, or when the script execution completes.

All meta-data and control structures (such as page tables) in main memory that are needed for efficient pricessing are kept outside of the database buffer. As opposed to the database buffer, there is always sufficient space for meta-data and control structures.

In the following phases of this project your job will be to keep the data synchronized and consistent between the row store, the column store and the disk while executing multiple transactions simultaneously. In this phase you just need to make sure that a single script file can execute from the beginning to the end producing correct output (log) and leaving the database in a consistent state.

Logging

Data Manager should keep a log file in which it can record all its actions. You need to record all Data Manager actions such as performed operations, creating new pages and swapping existing pages in and out of main memory. Following is the accepted logging format:

```
R X 13
SWAP OUT T-X P-6 B-11
SWAP OUT T-X P-2 B-11
SWAP IN  T-X P-8 B-13
Read: 13, John, 412-222-3333
I Y (18, Bob, 412-111-2222)
CREATE T-Y P-15 B-2
SWAP IN T-Y P-15 B-2
Inserted: 18, Bob, 412-111-2222
D Y
Deleted: Y
M X 609
MRead: 16, Tim, 609-222-3333
MRead: 19, Jim, 609-222-3333
G X 412
GCount: 89
```

Note that T-X P-6 B-11 means Table X, Page 6, Hash-Bucket 11.

Logging can significantly help you with debugging your project. In following phases these log records will be modified and utilized to restore the database to the previous consistent state if a transaction is aborted.

Implementation

You have the option of implementing your prototype *myTRC* in any language and on Windows or Unix-based operating system. You will be required to demonstrate your system and submit an electronic copy of your code, log and data files. Use the provided sample data as a template. It is your responsibility to make sure *myTRC* will work with any data.

Submission Requirements

- You must work in teams of 2. You will have to declare the members and team name (if you wish) of your team by **Saturday, February 14, 2015**. Send e-mail to cs2550-staff@cs.pitt.edu with your team name and the name and pitt username of each member, CCing the email to all members of the team.
- All submissions are electronic via the homework submission page (link available in course web page). You must submit your assignment before the due date (**Thursday, 11:59pm, February 26, 2015**).
- Additionally, you are required to give a demo of each phase of the project. The demo of the first phase will be held on *Thursday, March 5, 2015*. You will be provided with a URL to register for the demo.

Academic Honesty

The work in this project is to be done *independently* as a group. Discussions with other groups on the assignment should be limited to understanding the statement of the problem. Cheating in any way, including giving your work to someone else will result in an F for the course and a report to the appropriate University authority.