Javier Perez

CS 457

Project 4

Design Document

Atomicity

For this project, I implement the atomicity property of transactions by using the method described in the Assignment description. When an existing process wants to access a table, it checks to see if a file called "tablename_lock.txt" exists in the working directory. If it does exist, then the requested command for that table is disregarded and not saved as a committable action. If it does not exist, then the lock file will be created and have all the contents of the source table copied into it. This allows for the requested command to be saved as a committable action into the lock file.

Transactions

At a high-level, transactions are implemented with the help of the lock file mentioned in the implementation of the atomicity property of transactions. When a process wants to request a transaction, it must first declare the beginning of a transaction. This command initializes the database management system to begin tracking changes using a list of file names with committable changes. Once the process submits a transaction command, the lock file is referenced before giving the process access to the specified file. When a process successfully accesses a file, the files contents are copied over to the lock file, where any changes to the source file are instead redirected to the lock file and the source files name is recorded in a list of tracked changes. Once the process decides to commit its transaction, the contents of the lock file are

copied over into the source files stored in the list of tracked changes, with the final action being the deletion of the lock file.

Compilation and Execution:

To compile the code, you first want to make sure you have Python 3.10 or higher installed and added to your path. Additionally, you'll want to be using a Windows or Linux machine. Windows can be version 10 or 11, and for Linux this program was tested on an Ubuntu 22.04.1 LTS version. Take the zip file and extract it into a clean directory where you want the databases to be saved in. Once you have the files extracted, you can then open a command line and switch the current working directory into the one containing the extracted files. When inside that directory, you will then use the command "python japerez_pa4.py". This will start the program in its user interactive mode, where you will be able to submit commands into the command line. In the case that you want to use a test script, you would first use the command ".exit" to terminate the interactive mode and return to the command line interface. Next you would input the command "python japerez_pa4.py testscript.sql". Once you press enter, the program will automatically open, process, and output the results of the commands within that test script.