**ST. XAVIER’S COLLEGE**

**(Affiliated to Tribhuvan University)**

Maitighar, Kathmandu



**Database Management System**

**Assignment #**

**Submitted by:**

Pratik Gautam  
013BSCCSIT029

**Submitted to:**

|  |  |
| --- | --- |
| **Er. Sanjay Kumar Yadav**  Lecturer  St. Xavier’s College |  |

**Transactions:**

A transaction is a set of changes that must all be made together. It is a program unit whose execution mayor may not change the contents of a database. Transaction is executed as a single unit. If the database was in consistent state before a transaction, then after execution of the transaction also, the database must be in a consistent. For example, a transfer of money from one bank account to another requires two changes to the database both must succeed or fail together.

You are working on a system for a bank. A customer goes to the ATM and instructs it to transfer Rs. 1000 from savings to a checking account. This simple transaction requires two steps:

• Subtracting the money from the savings account balance.

• Adding the money to the checking account balance.

The code to create this transaction will require two updates to the database. For example, there will be two SQL statements: one UPDATE command to decrease the balance in savings and a second UPDATE command to increase the balance in the checking account.

You have to consider what would happen if a machine crashed between these two operations. The money has already been subtracted from the savings account will not be added to the checking account. It is lost. You might consider performing the addition to checking first, but then the customer ends up with extra money, and the bank loses. The point is that both changes must be made successfully. Thus, a transaction is defined as a set of changes that must be made together.

**Transaction Recovery:**

A transaction has to abort when it fails to execute or when it reaches a point from where it can’t go any further. This is called transaction failure where only a few transactions or processes are hurt.

When a system crashes, it may have several transactions being executed and various files opened for them to modify the data items. Transactions are made of various operations, which are atomic in nature. But according to ACID properties of DBMS, atomicity of transactions as a whole must be maintained, that is, either all the operations are executed or none.

There are two types of techniques, which can help a DBMS in recovering as well as maintaining the atomicity of a transaction:

Maintaining the logs of each transaction, and writing them onto some stable storage before actually modifying the database.

Maintaining shadow paging, where the changes are done on a volatile memory, and later, the actual database is updated.

**System Recovery:**

A database may become inconsistent because of a transaction failure (abort), Database system failure (possibly caused by OS crash) or media crash (disk-resident data is corrupted). Database Recovery is the process of restoring the database and the data to a consistent state. This may include restoring lost data up to the point of the event (e.g. system crash).

The recovery system ensures the database contains exactly those updates produced by committed transactions i.e. atomicity and durability, despite failures. Two approaches are discussed here: Manual Reprocessing and Automated Recovery

**Media Recovery:**

Media recovery restores data files to a point in time before failure. The following is a list of media recovery options:

**Complete Media Recovery:**

Complete media recovery is the recovery of all damaged or missing database files and the application of all redo information. The database may need to be opened with the RESETLOGS option if a backup control file or new control file was created for the recovery. There are three types of complete media recoveries.

1. Closed database recovery
2. Open-database, offline table space recovery
3. Open database, offline table space data file recovery

**Incomplete Media Recovery:**

Incomplete media recovery is also called point-in-time recovery. Point in time recovery that is not continued to a complete recovery must be terminated by the OPEN RESETLOGS option. The database must be closed during incomplete recovery operations. There are three types of incomplete media recoveries.

1. Cancel based recovery
2. Time based recovery
3. Change based recovery

**Two Phases Commit:**

A feature of transaction processing systems that enables databases to be returned to the pre-transaction state if some error condition occurs. A single transaction can update many different databases. The two-phase commit strategy is designed to ensure that either all the databases are updated or none of them, so that the databases remain synchronized.

Database changes required by a transaction are initially stored temporarily by each database. The transaction monitor then issues a "pre-commit" command to each database which requires an acknowledgment. If the monitor receives the appropriate response from each database, the monitor issues the "commit" command, which causes all databases to simultaneously make the transaction changes permanent.