

Date: 3/April/2024

DATE / /
PAGE

* Concurrency Control - Concurrency control is (multiple transaction) the management procedure that is required for controlling concurrent execution of the operation that is takes place on a data base.

* Concurrent Execution in DBMS -

In a multiuser system, multiple user can access and use the data base at a one time which is known as (same) concurrent execution of the data base. It means that the same data base is executed simultaneously on a multiuser system by different user.

* Transaction - It is a set of operation are used to perform a logical unit of work. A transaction generally represent change in data base. Each transaction generally represent both atomicity and consistency thus we require that transaction do not violate any database.

Consistency Constant - That is the database passed then a transaction started the database must be consistent when the transaction terminates.

However during the executions of transaction. It may necessary temporary to allow inconsistency. Since either the debit of and credit of must be done before the other.

is
re-
at

- * Read (A) - Read operation read the value of A from the data base and store in the buffer in the main memory.
- * Write (A) - Write operation writes the value of A back to the data base from the buffer.

• ACID properties -

imp
m
ie

ACID - Atomicity Consistency Isolation Durability

- * Atomicity - As a transaction is a set of logically related operation either all of them should be executed or none.

- * Consistency - If operation of debit and credit transaction on the same account are executed concurrently. It may leave the data base in an inconsistent state.

comi-

For example, $A = 1000$ and $B = 2000$

$R(A) \rightarrow A = A - 500$
 $W(A)$

ssed

$B = 2000 + 500$

operation

$W(B)$

It
istency.

Commit (process will be in RAM)

+

* Isolation - In isolation, multiple transactions runs parallel, so we convert parallel transaction in serial transaction.

* Durability - Once the database as committed a transaction the changes made by permanent.

* Transaction Strategy - Transaction Straight state through which transactions go during its lifetime. These are the stage which tell its current stage of the transaction and also tell how we will further to the processing in the transaction.

Types of transaction -

(1) Active State - When the instructions of the transaction are running then the transaction is in active state. If all the Read and write operation perform without any error or then it goes to partially committed stage. If any instructions fail then it goes to the fail state.

(2) Partially Committed - After the compilation of all the read and write operation the changes are made in main memory or local buffer. If the changes

are permanent on the database. Then the State ~~is~~ will change to committed state. In case of failure it will go to the failed state.

(3) Failed State - When any instructions of the transaction fails it goes to the failed state or if failures occur in making a permanent change on data on database.

7

DATE / /
PAGE

* Operators in DBMS- \bowtie (Join Operator)

(a) FAT (File Allocation Table) allocates memory.

(b) NTFS-

* Foreign Key - Master Table / Parent Table

Syntax - Create table <table_name>
(col1 <type> col2 <type> primary
key, col3 <type>);

* Detail Key / Child Key -

Create table <table_name>
(col1 <type> col2 <type> reference <Master table>
, col3 <type>);