Q4. Distributed Concurrency Control

- Concurrency controlling techniques ensure that multiple transactions are executed simultaneously while maintaining the ACID properties of transactions and serializability in schedules.

- Concurrency control is provided in a database to →

  → enforce isolation among transactions
  → preserve database consistency
  → resolve read-write and write-read conflicts

- Locking based Concurrency control protocols use the concept of locking data items

- A lock is a variable associated with a data item that determines whether read/write operations can be performed on the data item.

- The operation's access request is decided based on the compatibility of ~~two~~ lock modes

- A read lock is compatible with another read lock. A write lock is not compatible with another read or write lock.

- Locking based concurrency control systems can either use one-phase or two-phase locking protocols.

One Phase Locking Protocol →

- In this method, each transaction locks an item before use and releases the lock as soon as it finishes using it

- This locking method provides for maximum concurrency

- But it does not always enforce serializability.

Two Phase Locking Protocol →

- In this method all the locking operations occur before the first lock release or unlock operation.

- The transaction comprises of two phases →

  → In the first phase, the transaction acquires all the locks it needs and does not release any lock

  This is called the expanding or growing phase.

  → In the second phase, the transaction releases the locks and cannot request any new locks.

  This is called the shrinking phase.

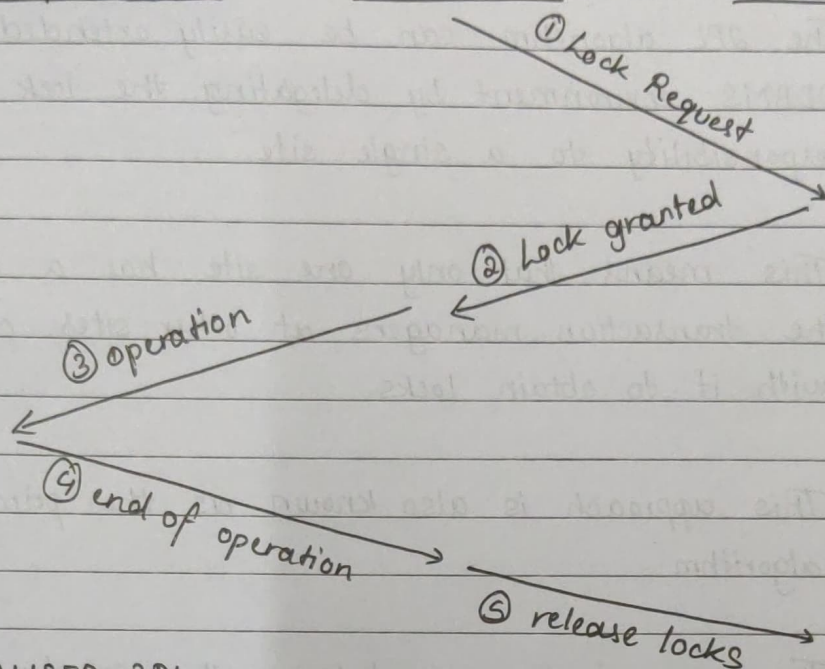Locks are usually maintained in a lock table.

Centralized 2PL →

- The 2PL algorithm can be easily extended to the DDBMS environment by delegating the lock management responsibility to a single site.

- This means that only one site has a lock manager, the transaction managers at other sites communicate with it to obtain locks.

- This approach is also known as the primary site 2PL algorithm

- The communication is between the coordinating TM, the Lock manager at the central site and data processors at other participating sites.

- The participating sites are those that store the data items on which the operation(s) is to be carried out.

- The transaction manager (C2PL - TM) is written as an algorithm that runs forever and waits until a message arrives either from the application (with a transaction operation) or from the lock manager.

- The data processors (DP) and the lock manager (C2PL - LM) algorithms are written as procedures that are called when needed.

Data processors   Coordinating   Participating
at participating sites   TM   TM

① Lock Request

② Lock granted

③ operation

④ end of operation
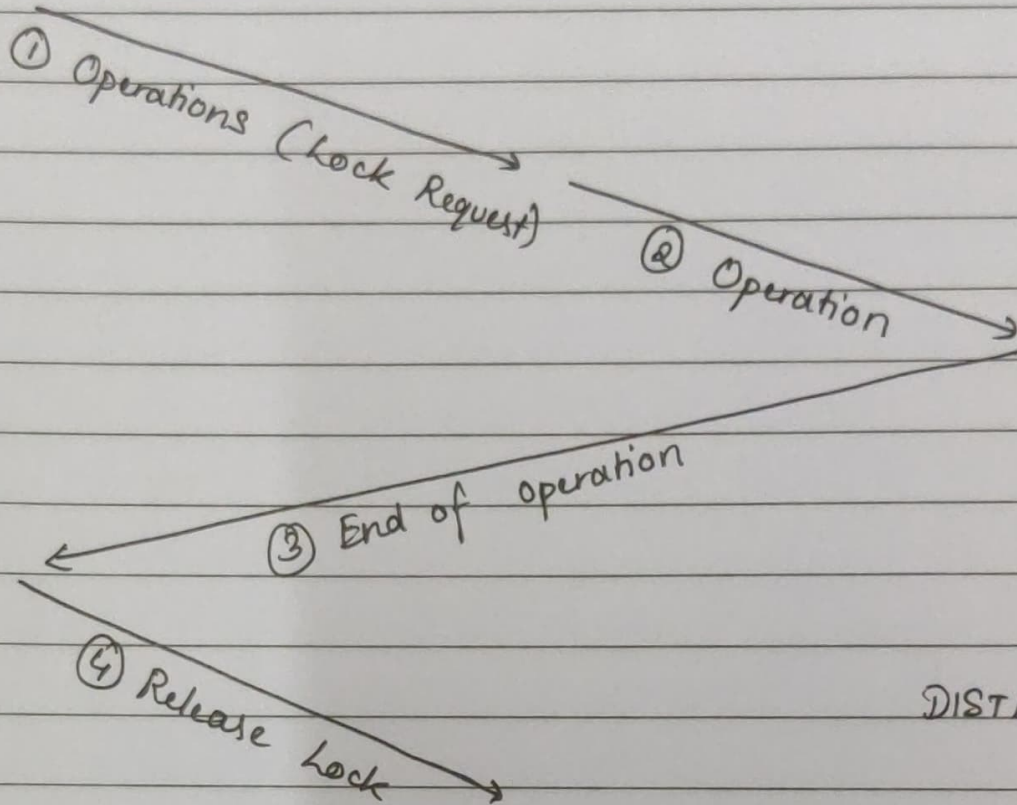
⑤ release locks

CENTRALISED 2PL

Distributed 2PL →

- D2PL requires the availability of lock manager at each site

- The D2PL–TM algorithm is similar to the C2PL–TM, with two major modifications

  → The messages sent to the central site lock manager in C2PL–TM are sent to all the sites participating sites in D2PL–TM.

  → Operations are not passed to the dataprocessors by the coordinating transaction manager, instead they are passed by the participating lock managers. So, the coordinating transaction manager does not wait for a lock request granted message

Coordinating TM	Participating Scheduler	Participating DM's

① Operations (Lock Request)

② Operation

③ End of operation

④ Release Lock

DISTRIBUTED

2PL