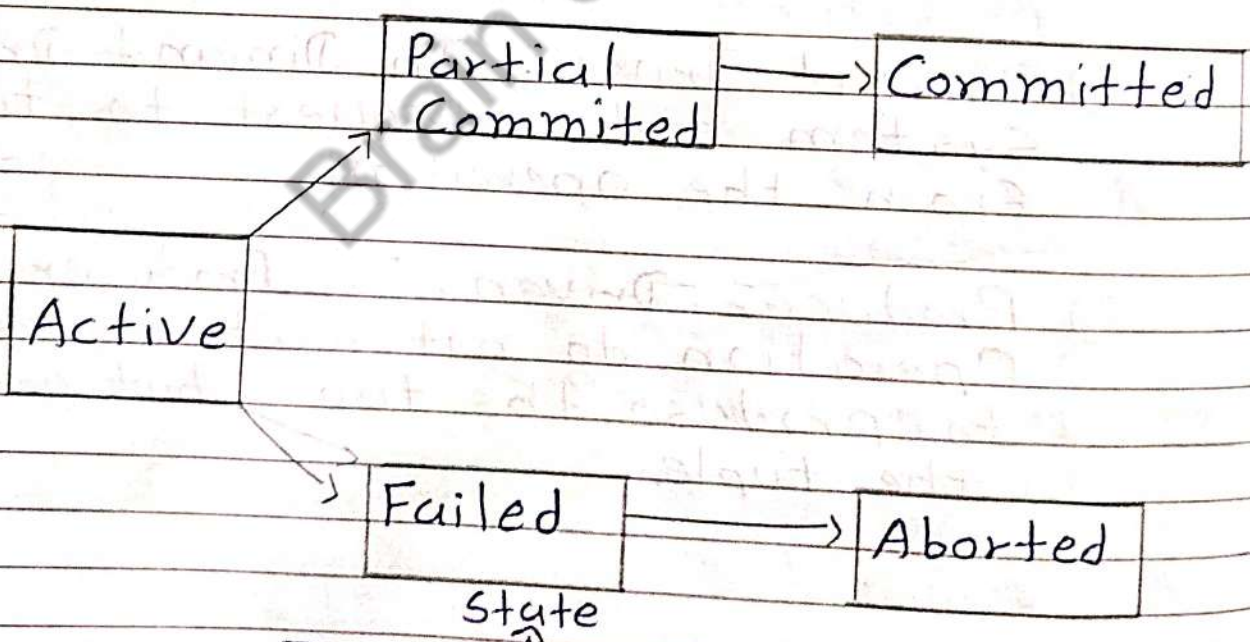
* Explane Transaction State Diagram.

Transaction is a sequence of operations performed as a single line.

There are Five State of Procesing Transaction.

1) Active
2) Partial Commited
3) Failed
4) Committed
5) Aborted.



1 Active : In this transaction is executed.

Transaction is the start to the active state.

2 Partial Commited:

In this state, transaction perform its Final operation.

3 Failed:

If transaction is not perform operation then transaction go in the Failed state.

4 Aborted:

After the Failed state, transaction is roll back and restored its the initial state.

5 Committed:

After the partial commited state, transaction is successfull complete and store in database.

* Explain schedule with its types.

Schedule is the process of Grouping Transactions.

Schedule is executed transaction in Chronological Order.

There are three types of Schedule.

1) Serial Schedule
2) Interleaved Schedule
3) Equivalent Schedule

1 Serial Schedule:

In this schedule, transaction are performed one by one.

Serial Schedule, Process transaction one by one.

In this schedule, no transaction starts until running transaction has ended.

In this schedule, One transaction perform at a time.

Ex.

T_1	T_2
R(A)	
A = A-50	
W(A)	
R(B)	
B = B+50	
W(B)	
Commit	
	R(A)
	A = A+50
	W(A)
	R(B)
	B = B-50
	W(B)
	Commit

2 Interleaved Schedule:

In this schedule, transaction are not performed one by one.

In Interleaved Schedule, transaction execution can switch between the transaction.

In Interleaved Schedule, Second transaction start before first transaction endes.

Ex.

T_1	T_2
R(A)	
A = A - 50	
W(A)	
	R(B)
	B = B + 50
	W(B)
	Commit
R(A)	
A = A + 100	
W(A)	
	R(B)
	B = B + 100
	W(B)
	Commit

3 Equivalent Schedule:

In this schedule, transaction gives same result.

In this schedule, transaction can be switch and one transaction can be serial but result of transaction same.

Equivalent schedule, give same result when both transaction can be perform with two different method.

Ex.

T_1	T_2
R(A)	
A = A - 50	
W(A)	
R(B)	
B = B + 50	
W(B)	
Commit	
	R(C)
	C = C + 30
	W(C)
	Commit

T_1	T_2
R(A)	
A = A - 50	
W(A)	
	R(B)
	B = B + 50
	W(B)
	Commit
R(C)	
C = C + 30	
W(C)	
Commit	

Both Transaction Gives same Result.

* Explaine Two phase commit Protocol.

Two phase commit protocol insures that all the particeipants perform the same action.

In this Protocol there are two participants.

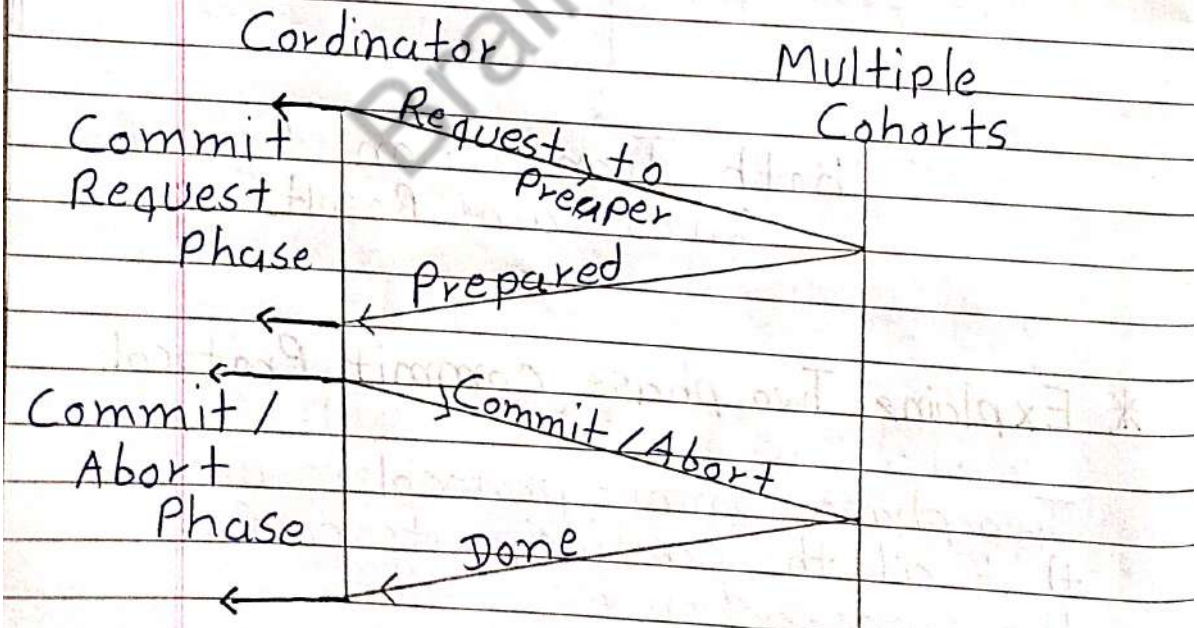
1) Cordinator
2) Cohorts

1 Cordinator : Cordinator cordinates with all the particiPent.

2 Cohorts : Cohorts is participet in the action.

Two phase commit Protocol consist Two phase.

1) Commit Request Phase
2) Commit / Abort Phase.



1 Commit Request Phase:

In this phase, Cordinator request the multiple cohorts to prepar the request.

Multiple Cohorts accept the cordinator 's request.

In this phase, Multiple Cohorts prepar the request and Cordinates with cordinator.

So, In this phase request is prepar.

2 Commit / Abort Phase:

In this phase, Cordinator send the action to the cohorts,

If Cohorts is ready then they accept the action and do commit and send done message.

Else, Cohorts is not ready then they abort the action.

So, In this phase request commit or Abort.

* Explain Database Recovery with its types.

Database Recovery is very important to maintain database.

In some case, transacation may not reach to commit or Abort point.

Database Recovery is the process the recovering database and Data to a consitant state.

Database Recovery may include restore the lost data upto the point.

Database Recovery consist only one type.

1) Logbased Recovery Method.

1 Logbased Recovery Method:

Logbased Recovery Method containt sequence of log records.

This log records sequence contains,

1) Start of Transaction
2) Transacation - id
3) Record - id
4) Type of Operation
5) Old value, new value
6) Commit Transacation.

Ex When Transacation start $\langle T_i \text{ start} \rangle$

- Before T_i exeutes write(x), A log record.

$$\langle T_i, x, V_1, V_2 \rangle$$

where,
T_i = Transacation
x = name of Attribute
V_1 = Old name
V_2 = new name

- For Undo Transacation,

$$\langle T_i, x, V_1, V_2 \rangle$$

- For Redo Transacation,

$$\langle T_i, x, V_1, V_2 \rangle$$

- T_i Finishes $\langle T_i \text{ commit} \rangle$

There are Two Types of Log based Recovery Method.

1) Immediate database Modification
2) Deferred database Modification

1 Immediate Database Modification:

In this modification, Database update without it reach to the commit point.

In this modification, Database change without do any type of commit.

Ex.

$\langle T_1 \text{ start} \rangle$
$\langle T_1, A, 500, 400 \rangle$
$\langle T_1, B, 600, 700 \rangle$

Here, A = 400, B = 700 change in database.

2 Deferred Database Modification:

In this modification, Database update untill it reach to the commit point.

In this modification, Database change untill we do any type of Commit.

Ex. <T, start>
<T, A, 400>
<T, B, 700>
<T, commit>

Here A = 400 and B = 700
change in database.

* Explain DBMS Concurrency Control with its Problems and Protocols.

In a ~~mutti~~ multi-user system, multiple user can access and use the same database at a time, this problem, problem remove by DBMS Concurrency Control.

DBMS Concurrency follow interleaved manner execution.

Using DBMS concurrency control all the type of ACID Properties.

DBMS Concurrency Control maintain the Database.

=> This are the Different Concurrency control problem.

1) Lost Update Problem
2) Dirty Read Problem
3) Unrepeatable Read Problem.

1 Lost Update Problem:

Lost Update Problem is also called W-W conflict.

This problem occurs when, Two Transacation perform Read, write operation in different database using interleaved manner.

In this transacation, we can lost different data in database.

Ex.

T_X	T_Y
R(A) = 300	
A = A - 50	
	R(A) = 300
	A = A + 100
W(A)	
	W(A)

In above, Transacation, we perform write, read operation.

Here, Transacation T_x perform R(A) operation and untill the transacation T_x perform W(A) operation.

After that using T_x database, T_y perform R(A) operation and T_y Read A = 300.

But updated value is A = 250.

So, A = 250 value is lost in database. This is called Lost update Problem.

2 Dirty Read Problem:

Dirty Read Problem is also called W-R Conflict.

This problem occurs when, One transaction update an item in database and somehow the transacation fails and before gets data rollback the second transaction take updated value, this is called Dirty Read Problem.

Ex. T_x	T_y
R(A) = 300	
A = A + 50	
W(A) = 350	
	R(A) = 350
Rollback	

Here, T_x transaction performes Read and write operation and some how transaction get failed and rollback.

T_y transaction Read this failed transacation value.

This is called Dirty Read Problem.

3 Unrepeatable Read Problem:

Unrepeatable Read Problem is also called W-R conflict.

This problem occures when, two transacation perform in database and One transacation Read

Two different value in database, this is called Unrepeatable Read Problem.

Ex.

T_x	T_y
R(A) = 300	
	R(A) = 300
	A = A + 50 = 350
	W(A) = 350
R(A) = 350	

Here, Transacation T_x perform Read operation and Transacation T_y perform Read and write operation.

Transacation T_x perform two Read operation and get two different value.

This problem is called Unrepeatable Read Problem.

This Three are Basic DBMS Concurrency Control Problem.

=> This are the different DBMS Concurrency Control Protocols:

1) Lock Based
2) Time Based
3) Validation Based

1 Lock Based:

In this Protocols, any transacation cannot Read or write data untill it require lock.

Every transaction require lock to perform operation.

There are two type of Lock.

(a) Shared lock
(b) Exclusive lock.

(a) Shared lock: This lock is perform only read operation.

In this lock, No Transacation update in database.

In this lock, we cannot write data in the database.

(b) Exclusive lock: This lock is perform Read and write both operation.

In this lock, we can update the transacation in database.

In this lock, we can Read and write data in database.

There are Four types of Exclusive lock.

(i) Simplistic lock
(ii) Pre-claiming lock
(iii) Two-phase locking
(iv) Strict Two-phase locking

(i) Simplistic lock: This lock is work with three steps.

step - 1 - It allows the transacation aquires the lock

step - 2 - Complete transacation

step - 3 - Redise the lock.

(ii) Pre-claiming Lock: Pre-claiming lock is also work with this three steps.

Step-1 - Evaluate the transacation list and count lock for the transacation.

Step-2 - Request DBMS For lock.

step-3 - If all lock are granted then transacation execute

else all lock are not granted then untill the rollback to all locks are granted.

(iii) Two-phase locking: This lock is containt two phase.

1) Growing 2) Shrinking.

This lock is perform three step:

step-1 - Seeks Permission for the lock it requires.

step-2 - Evaluate First transacation and acquires second lock for ~~see~~ second

Transacation.

Step-3- Realise First lock.

Ex.

	T_1	T_2
0	Lock-S(A)	
1	-	L-S(A)
2	L-X(B)	-
3	-	-
4	UnL-(A)	
5	-	L-X(C)
6	Un.L-(B)	
7		Un Lo.(A)
8		Un Lo.(C)
9		

In above Example,

Growing Phase,

For T_1 - 1 to 3
T_2 - 2 to 6

Shrinking Phase,

For T_1 = 5 to 7
T_2 = 8 to 9

(iv) Strict - Two Phase Locking: This locking containt only Growing phase.

This lock is also work with this three step.

step-1 - Lock Aquired

step-2 - Evaluation Transacation.

step-3 - Not realise untill without commit.

2 Timestamp Order Protocol:

This protocal Perform Read and write transaction to ~~the~~ cheking This two step:

- step: 1 - T_i Perform R(x) operation.

If W_T(x) > W_T(T_i) -> Reject

If W_T(x) <= W_T(T_i) -> Update

- step: 2 - T_i Perform W(x) operation.

If R_T(T_i) < R_T(x) -> Reject

If R-T(T_i) < W_T(x) -> Reject

3 Validation Based Protocol:

This Protocol consist this three phase.

- Read Phase : Transacation is read and perform all the operation using tempoary variable location without change in database.

- Validation Phase : Verified the Data in a tempoary variable location.

- Write Phase: After Verification, all the operation update in database otherwise rollback Transacation.

DBMS Concurrency Control

- Problem
  - Lost Update Problem
  - Dirty Read Problem
  - Unreapeatable Read Problem
- Protocol
  - Lock Based
    - Shared Lock
    - Exclusive Lock
      - Simplistic lock
      - Pre-claiming lock
      - Two-phase locking
      - Strict-Two-phase locking
  - Timestemp Based
  - Validation Based

* Explain Serializability.

A schedule is serialized if it is equivalent to a serial schedule.

In Serializability multiple transacation are allow.

It refers to the sequence of action such as read, write, commit are performed in a serial manner.

There are two types of Serializability.

1) Confilict Serializability
2) View Serializability.

1 Confilict Serializability:

This are the four confilict Instruction.

(a) If we perform Read and Read operation in two transacation then no confilict acuures.

(b) If we perform Read and write Operation in two transacation then confilict acuures.

(c) If we performs Write and Read operation in two transacation then confilict acuures.

(d) If we performs Write and write operation in two transacation then confilict acuures.

2 View Serializability :

There are three type of Read view serializability.

(a) Intial Read:

An intial read of both schedules must be same.

Ex.

S_1		S_2		S_3	
T_1	T_2	T_1	T_2	T_1	T_2
R(A)			R(A)		W(A)
	W(A)	W(A)		R(A)	

Here, S_3 and S_1 is view equivalent. If S_1 schedule transacation R(A) and S_3 schedule transacation is also R(A).

(b) Updated Read:

An updated read of both schedules must be same.

Ex.

S_1

T_1	T_2	T_3
WCA)		
	WCA)	
		RCA)

S_2

T_1	T_2	T_3
	WCA)	
WCA)		
		RCA)

In schedule S_1, If T_3 read the RCA) which is update by T_2 Transacation WCA).

In schedule S_2, If T_3 Read the RCA) which is update by T_1 Transacation WCA)

So, Schedule S_1 and S_2 is not View Equivalent.

(c) Final write:

Final write of both schedules must be same.

If one transacation write operation in T_i then second transacation write operation also in T_i

Ex.

S_1			S_2		
T_1	T_2	T_3	T_1	T_2	T_3
W(A)				R(A)	
	R(A)		W(A)		
		W(A)			W(A)

Here, Schedule S_1 transacation T_3 write Final operation and schedule S_2 transacation T_3 is also write Final operation.

So, S_1 and S_2 is a View equivalent.