International Islamic University Chittagong (IIUC)

Department of Computer and Communication Engineering

Program: **B.sc (Eng.)**
Course Code: **CCE–3505**
Type: **Theory**
Segment: **Final Term-Segment 8**

Baizid MD Ashadzzaman
Designation: **Adjunct Lecturer**
Email: **baizid.md.ashadzzaman@gmail.com**
Phone: **8801862420119**

Course Contents:

Contents	Note

Concurrency control, Recovery System and Distribute databases:

Concurrency Control

Concurrency Control in Database Management System is a procedure of managing simultaneous operations or transaction without conflicting with each other.

□ It ensures that database transactions are performed concurrently and accurately to produce correct results without violating data integrity of the respective Database. □ Concurrent access is quite easy if all users are just reading data.

Concurrency Control Protocols

Different concurrency control protocols offer different benefits between the amount of concurrency they allow and the amount of overhead that they impose. Following are the Concurrency Control techniques in DBMS:

- Lock-Based Protocols
- Timestamp-Based Protocols
- Validation Based Protocol

Lock-based Protocols

Lock Based Protocols in DBMS is a mechanism in which a transaction cannot Read or Write the data until it acquires an appropriate lock. Lock based protocols help to eliminate the concurrency problem in DBMS for simultaneous transactions by locking a particular transaction to a single user.

A lock is a data variable which is associated with a data item. This lock signifies that operations that can be performed on the data item. Locks in DBMS help synchronize access to the database items by concurrent transactions.

There are two types of locks used in databases.

Shared Lock : Shared lock is also known as read lock which allows multiple transactions to read the data simultaneously. The transaction which is holding a shared lock can only read the data item but it can not modify the data item.

Exclusive Lock : Exclusive lock is also known as the write lock. Exclusive lock allows a transaction to update a data item. Only one transaction can hold the exclusive lock on a data item at a time. While a transaction is holding an exclusive lock on a data item, no other transaction is allowed to acquire a shared/exclusive lock on the same data item.

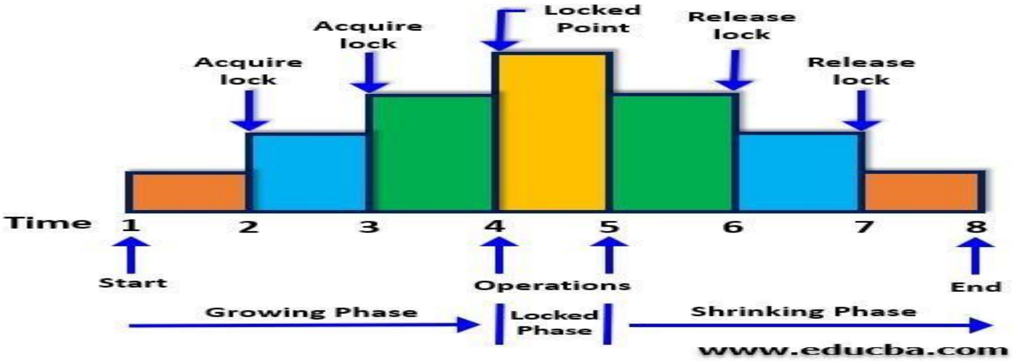
Types of Lock Based Protocol

There are two lock based protocol mostly used in database:

1. **Two Phase Locking Protocol :** Two Phase Locking Protocol also known as 2PL protocol is a method of concurrency control in DBMS that ensures serializability by applying a lock to the transaction data which blocks other transactions to access the same data simultaneously.Two Phase Locking protocol helps to eliminate the concurrency problem in DBMS.

This locking protocol divides the execution phase of a transaction into three different parts.

- In the first phase, when the transaction begins to execute, it requires permission for the locks.
- The second part is where the transaction obtains all the locks. When a transaction releases its first lock, the third phase starts.
- In this third phase, the transaction cannot demand any new locks. Instead, it only releases the acquired locks



The Two-Phase Locking protocol allows each transaction to make a lock or unlock request in two steps:

Growing Phase: In this phase transaction may obtain locks but may not release any locks. In this phase, the transaction starts acquiring locks before performing any modification on the data items. Once a transaction acquires a lock, that lock can not be released until the transaction reaches the end of the execution.

Shrinking Phase: In this phase, a transaction may release locks but not obtain any new lock.In this phase, the transaction releases all the acquired locks once it performs all the modifications on the data item. Once the transaction starts releasing the locks, it can not acquire any locks further.

Lock Point

The Point at which the growing phase ends, i.e., when a transaction takes the final lock it needs to carry on its work

2. **Strict Two Phase Locking Protocol :** It is almost similar to the two phase locking protocol the only difference is that in two phase locking the transaction can release its locks before it commits, but in case of strict two phase locking the transactions are only allowed to release the locks only when they performs commits.

Timestamp based Protocol

Timestamp based Protocol in DBMS is an algorithm which uses the System Time or Logical Counter as a timestamp to serialize the execution of concurrent transactions. The Timestamp-based protocol ensures that every conflicting read and write operations are executed in a timestamp order.

- It uses system time to determine the time stamp of the transaction. This is the most commonly used concurrency protocol.
- Lock-based protocols help you to manage the order between the conflicting transactions when they will execute.
- Timestamp-based protocols manage conflicts as soon as an operation is created.

Example:

Suppose there are three transactions T1, T2, and T3.

T1 has entered the system at time 0010

T2 has entered the system at 0020

T3 has entered the system at 0030

Priority will be given to transaction T1, then transaction T2 and lastly Transaction T3.

Deadlock detection and Recovery

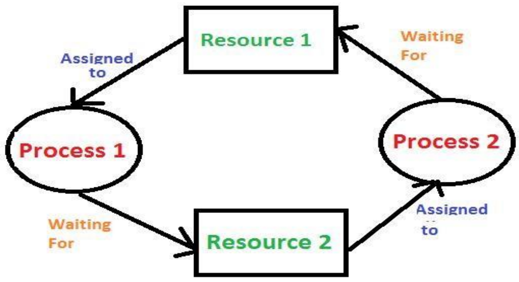
Deadlock

Deadlock refers to a specific situation where two or more transactions are waiting for each other to release a resource in a circular chain.

Deadlock detection algorithms, such as the Wait-For Graph, are used to identify deadlocks, and recovery algorithms, such as the Rollback and Abort algorithm, are used to resolve them. **Deadlock Detection :**

1. If resources have a single instance –

In this case for Deadlock detection, we can run an algorithm to check for the cycle in the Resource Allocation Graph. The presence of a cycle in the graph is a sufficient condition for deadlock.



In the above diagram, resource 1 and resource 2 have single instances. There is a cycle R1 → P1 → R2 → P2. So, Deadlock is Confirmed.

2. Wait-For Graph Algorithm –

The Wait-For Graph Algorithm is a deadlock detection algorithm used to detect deadlocks in a system where resources can have multiple instances. The algorithm works by constructing a Wait-For Graph, which is a directed graph that represents the dependencies between processes and resources.

Deadlock Recovery :
A traditional operating system such as Windows doesn't deal with deadlock recovery as it is a time and space-consuming process. Real-time operating systems use Deadlock recovery.

1. **Killing the process –**

Killing all the processes involved in the deadlock. Killing process one by one. After killing each process check for deadlock again and keep repeating the process till the system recovers from deadlock.

2. **Resource Preemption –**

Resources are preempted from the processes involved in the deadlock, and preempted resources are allocated to other processes so that there is a possibility of recovering the system from the deadlock.

3. **Concurrency Control –** Deadlocks can occur in concurrent systems when two or more processes are blocked, waiting for each other to release the resources they need. This can result in a system-wide stall, where no process can make progress. Concurrency control mechanisms can help prevent deadlocks by managing access to shared resources and ensuring that concurrent processes do not interfere with each other.

Source: https://www.studocu.com/in/document/rajasthan-technical-university/database-management-system/dbms2-best-lecture-notes-on-data-base-managementsystem2/43103288