

Computer Science & IT

Database Management System



Relational Model & Normal Forms

Lecture No. 01



By- Vishal Sir

Topics to be Covered



✓
Topic

Syllabus

✓
Topic

Introduction to DBMS

✓
Topic

Relational Database Model

✓
Topic

Functional dependency



Topic : Syllabus



✓ ➤ Relational Model & Normal Forms

✓ ➤ Query Languages

✓ ➤ Transactions and concurrency control

Not in the
Syllabus of
DA' paper.

✓ ➤ File Organization & Indexing

✓ ➤ ER model



Topic : Introduction to DBMS



Database : Database is organized collection of information

Information can be organized in multiple ways, i.e. we have multiple database models.

From GATE
Point of view
We will discuss

(i) Relational Model
& (ii) ER Model.

eg { Relational Model
ER model
Network Model
Hierarchical Model
Object oriented Model

DBMS : It is the software used to manage & access the database efficiently.

- ① "File system" can be used to manage and access the database, but file system fails to do the task efficiently if database is too large.
- ② Database files are stored in the non-volatile memory i.e., secondary memory.
- ③ Unit of transfer between secondary memory and main memory is "One disk block".

IO Cost:- IO cost of a "record access" can be defined as number of secondary memory disk blocks that needs to be transferred from secondary memory to main memory in order to access that record from the database

• { it is stored in the }
 { blocks of the disk }

{ B₁

{ B₂

{ B₃

{ B₄

Student.txt file

Sid	Sname	-----
S ₁ S100		
S150 S ₄		
S227 S15		
S225 S ₃		
⋮	⋮	⋮
S ₂ S11		

"other information related to student"

→ Retrieve record of the student
 with Sid = S240

{ Manually
 search the
 record }

↓
 It will be
 time Consuming

{ write a
 Program to
 search within
 the file }

↓
 To write a program
 we need to know
 the physical details
 of the file

Let 'N' blocks
 of the disk are
 required to
 store the file
 Student.txt

{ B_N
 (Too large)

{ it is stored in the }
blocks of the disk }

Student.txt file

"other information related to student"

B₁

B₂

B₃

B₄

Sid	Sname	-----
S ₁ S ₁₀₀		
S ₁₅₀ S ₄		
S ₂₂₇ S ₁₅		
S ₂₂₅ S ₃		
⋮ ⋮ ⋮ ⋮ ⋮	⋮ ⋮ ⋮ ⋮ ⋮	⋮ ⋮ ⋮ ⋮ ⋮
S ₂ S ₁₁		

→ Retrieve record of the student with Sid = S₂₄₀

Even if we have written a program to search within the file, then we will transfer the blocks of the file from secondary memory to main memory (one-by-one) and we will search for Sid: S₂₄₀.

In worst case we may have to transfer all the blocks of the file from secondary mem. to main memory { High IO Cost }

Let 'N' blocks of the disk are required to store the file Student.txt

B_N
(Too large)

{ it is stored in the }
blocks of the disk }

Student.txt file

"other information related to student"

	Sid	Sname	-----
B ₁	S ₁ S100		
B ₂	S150 S ₄		
B ₃	S227 S15		
B ₄	S225 S ₃		
	⋮	⋮	⋮
	⋮	⋮	⋮
	⋮	⋮	⋮
	⋮	⋮	⋮
	⋮	⋮	⋮
	S ₂ S11		

User U₁: Wants to update (write)
the record with Sid = S₁₅
if Sid = S₄

User U₂: Wants to update (write)
the record with Sid = S₃

Users wants to update different
records of the file, so They
should be allowed to execute
Concurrently.

Let 'N' blocks
of the disk are
required to
store the file
Student.txt

B_N
(Too large)

{ it is stored in the blocks of the disk }

Student.txt file

"other information related to student"

(w.r.t. File System)

	Sid	Sname	-----
B ₁	S ₁ S100		
B ₂	S150		
B ₃	S ₄ S227		
B ₄	S15 S225		
	S ₃		
	S ₂ S11		

User U₁

User U₂

lock (Student.txt)

it will be granted

Update (S15)

in between User U₂ request for the lock on file

lock (Student.txt)

This request will be denied because file is already locked by U₁

User U₁ & U₂ wants to update different records, but still they are not allowed to execute concurrently.
∴ Low Concurrency level using file system

Let 'N' blocks of the disk are required to store the file Student.txt

B_N (Too large)

File System vs DBMS :-

File System

vs

DBMS

✓ ① If we want to access the record from the file using a program, then in order to write the program we need to know physical details of the file.

✓ ② More IO Cost

✓ ③ Low Concurrency:
{ In file system locking is at file level }

① DBMS uses 3-tier architecture.

3-tier architecture of DBMS

Provides,

→ (i) Data abstraction:

{ It allows us to hide the physical details of database from External User }

→ (ii) Data independence:

{ Changes performed at lower level will not affect higher levels }

✓ ② less IO Cost

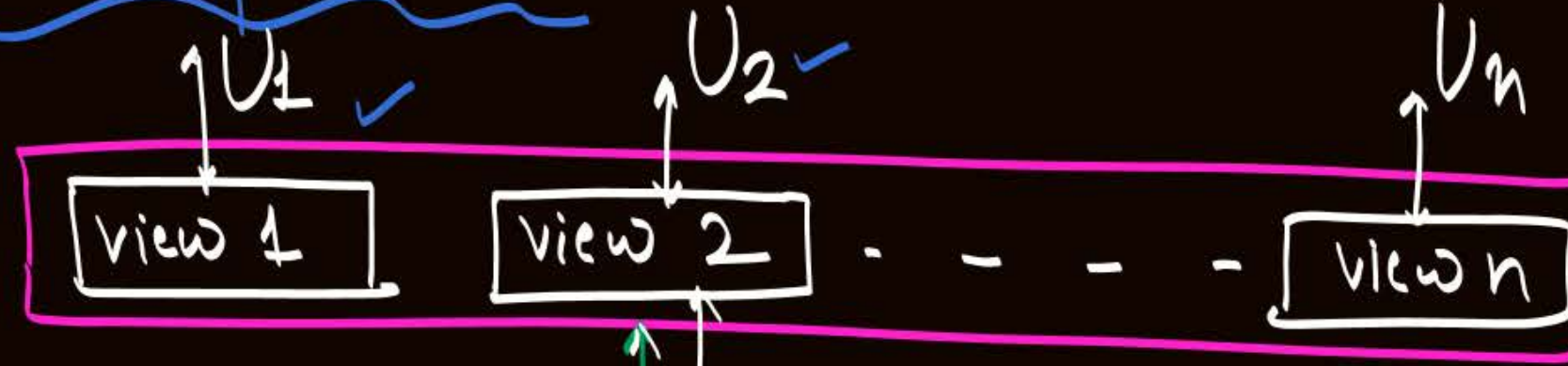
{ It uses indexing }

③ High Concurrency level:

{ In DBMS, locking is at record level }

→ 3-tier Architecture of DBMS:-

External level



Third party
trusted
Software





2 mins Summary



Topic

Syllabus

Topic

Introduction to DBMS

Topic

Relational Database Model

Topic

Functional dependency

Next Class

THANK - YOU