

Mod-04 Lec-18 Data management

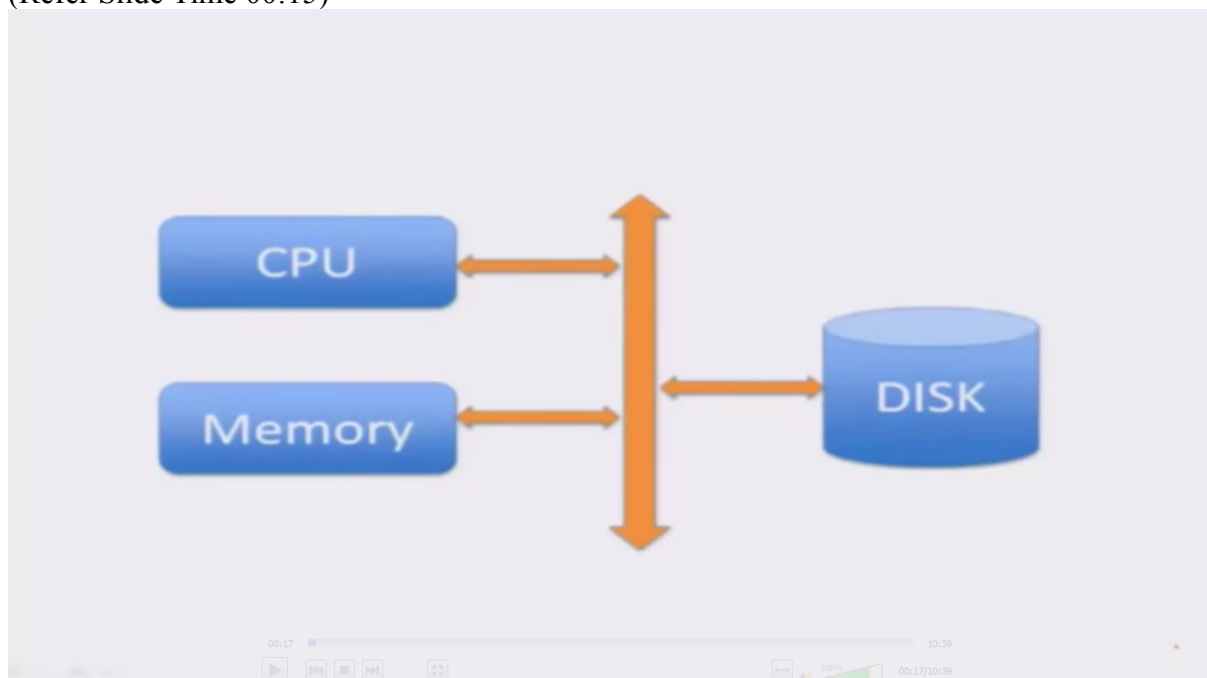
Data Management

Prabhakar TV
IIT Kanpur



Hi. My name is Prabhakar. I am with IIT Kanpur. Welcome to this lecture on Data Management. We will have a quick look into what's a DBMS.

(Refer Slide Time 00:15)



Here is a simplified model of a computer. There is a central processing unit. There is a memory here and there is a disk. Programs and data sitting on the disk, they are brought into

the main memory, this part and the CPU executes these programs. And after the execution is over, some of the data is written like on to the disk.

(Refer Slide Time 00:40)

Data

- Data on the disk is “persistent”
 - Typically magnetic storage
 - Solid state disks
- Does not vanish when the power goes or the program terminates
- If you want the data to persist you have to write it on the disk

Now the thing with this is if you want the data to persist after the program finishes execution or when the computer is turned off, you will have to put it on the disk. If the data is sitting in the main memory, it will go away when you turn the power off or when the program terminates. The disk is a persistent data store. The main memory is not. Typically, it's made of magnetic storage or these days you get solid state disks and when you put it on a disk, you get persistence.

(Refer Slide Time 01:16)

Abstractions



The diagram illustrates the concept of abstractions in computing. It shows four application icons (Word Processor, Image Processor, Language Processor, and DBMS) positioned above a red bar representing memory. Below the red bar is a blue bar representing binary data (010101100100101010010111100001101010101101011001001010001).

So here I have a set of abstractions on what happens on the disk and in the computer. So we know that everything is represented as a set of zeros and ones. When it's on the disk, it's a string of zeros and ones. Now there is a file manager, which looks at these zeros and ones and gives an abstraction of a set of files. It clangs them together saying these are one file, these sets of zeros and ones are another file and so on.

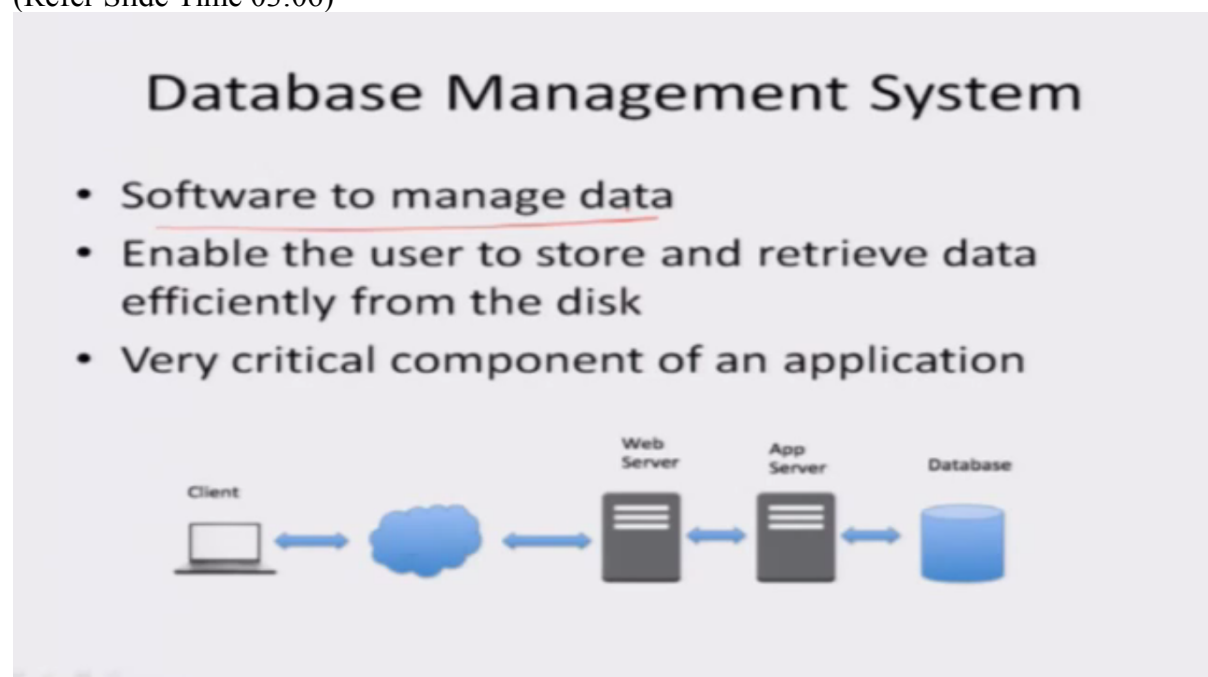
So at this level I have a sequence of files. I can look at them as files. You all know about this, right? When you use the disk explorer you can see various files and directories and so on.

Now some of these files may be documents, may be Word documents or spreadsheets and so on. So we need a word processor to be able to figure out that it is a document to see and manipulate. Similarly, some of them may be images, may be JPEG images or something else and some other software is needed to figure out that this file is storing an image.

And then we have something like language processors, the program, the file maybe a program and a language processor will understand that this is a program written in a programming language and then compile it and execute it and so on. So this is another piece of software I use to manipulate my files.

And then there is this thing called database management system, which can take a file and provide you an abstraction of a DBMS. So the database management system is the software, which can interpret the strings of zeros and ones that are written on the disk as values, which you and I can understand.

(Refer Slide Time 03:06)

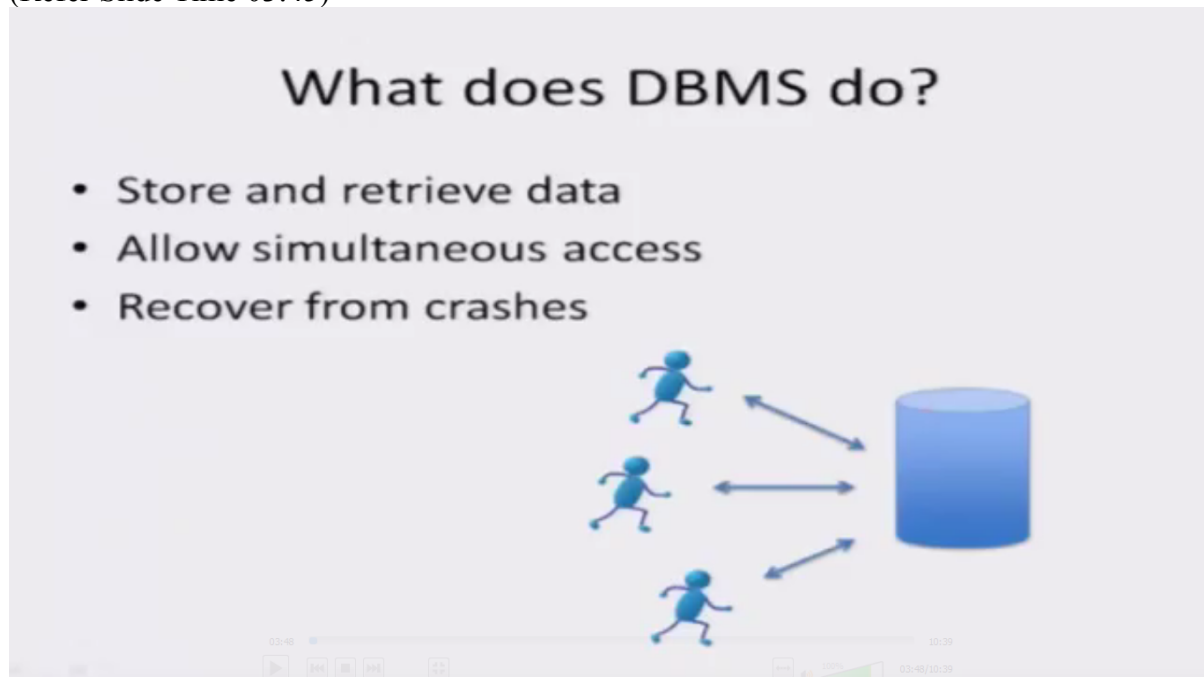


So essentially it is some software to manage data. It will allow you to store and retrieve data from the disk. The disk is very critical. Efficiently means without spending too much time and as you would accept, it's a very essential part of any application. If the data I manage,

process, input, type in and so on is not available when I come back tomorrow, the application is no good.

We have seen in the web lecture, this is the architecture of a website and there is this database sitting at the back deep inside storing all the important and relevant information and history.

(Refer Slide Time 03:45)



So what does the DBMS do? Typically, it stores and retrieves data. You can write into the DBMS and read from the DBMS. This is the simplest of things that it does.

There is another complicated function that it does. Let's say two people simultaneously want to read or write into the disk. To give an example, there is one ticket, a train ticket from Delhi to Kanpur. Two people want to buy that. I have to permit only one person to buy the ticket. I can't issue the ticket to both people. So this is called concurrent access.

So a DBMS engine has built-in mechanisms to permit multiple people to access simultaneously.

The third function that a DBMS does is to recover from crashes. By this what we mean is you know computer systems fail, the hard disk crashes and power goes off in between. Sometimes the software crashes, but if I have bought a train ticket, it should not happen because the train, the computer has crashed, my reservation is lost. Right?

There should be a mechanism to recover or let's say I'm transferring money from this account to that account. In the middle of this transfer, if the power fails, the money either should be transferred or not transferred at all. You cannot deduct money from my account and not add it to your account. Either you do not deduct from my account at all or if you deduct, then you should add and in between if there is a crash, you have to make sure that either it is not deducted at all or it is transferred from my account to your account.

So these are very important functions of a database engine and this is essentially what it does. It's actually quite a complicated piece of software. It takes a huge amount of code and competence to build a database engine.

(Refer Slide Time 05:43)

Relational Data Bases

- Data is modeled as “Relations”
- These are like “Tables”

So when data is stored in a computer, it is stored as a set of relations. This is called a relational model. These database engines typically are relational database engines. Now relations are like nothing but tables. Here is an example.

(Refer Slide Time 06:01)

Relational Data Bases

- Data is modeled as “Relations”
- These are like “Tables”

Employee

Emp No.	Name	Dept	Date of Joining
134	Murali	Admin	05/05/2005
221	Deepak	Media	07/08/2007
333	Revathy	Devops	19/06/2011
146	Gaurav	Devops	08/05/2005
356	Sunil	Devops	25/07/2011

Let's say I have an organization. There are employees in the organization. I want to store their data in the DBMS. So I create a table, a relation called employee. now this employee relation

has these kinds of information. It stores employee number, the name of the employee, the department in which the employee is working and the date of joining.

So here if you see 221 is the employee number. Deepak is the name of the employee. He works in media technologies and he joined in 2007. So I have a large number of records like this. So this is called a table or a relation.

(Refer Slide Time 06:42)



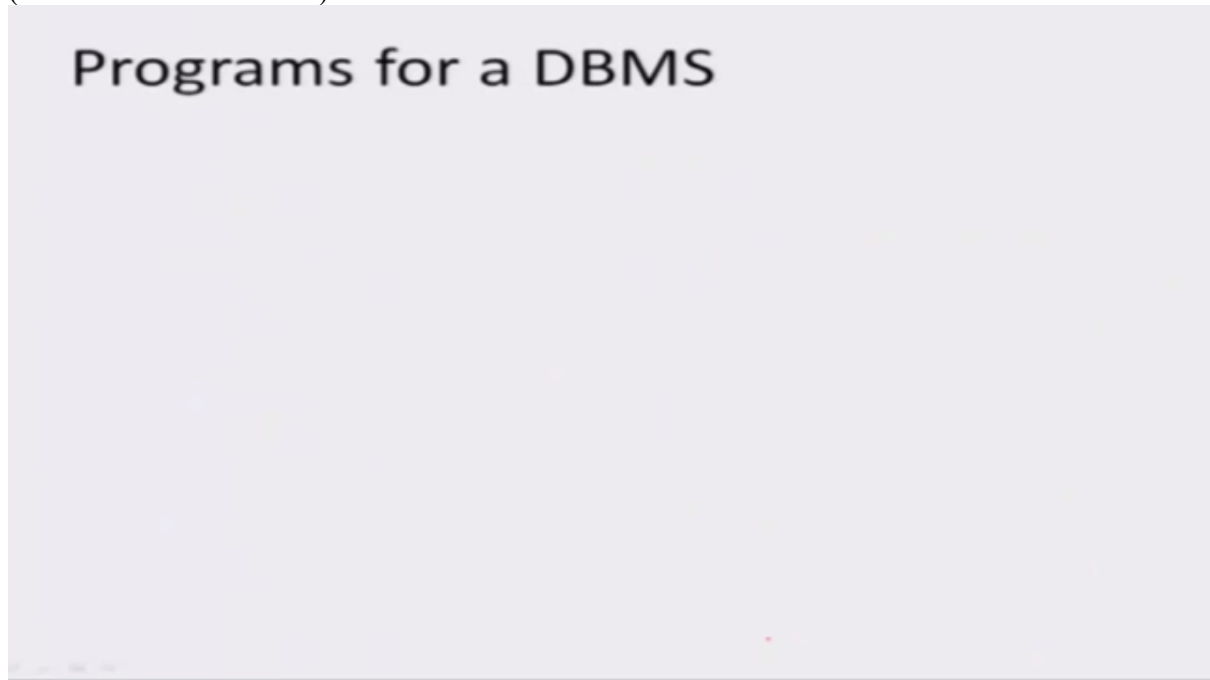
So one row in the relation is called a tuple and these columns are called attributes. So a relation will have many attributes and a large number of tuples or rows in it.

(Refer Slide Time 06:56)

The screenshot shows a video player window with a slide titled "The database is a collection of tables". The slide contains three tables of varying sizes and colors (blue, orange, red, green) representing different tables. A red bracket groups these tables under the label "Schema". The video player interface includes a progress bar at the bottom, showing the video is at 07:22 out of 10:39. The taskbar at the bottom shows several open applications, including VLC media player, Google Chrome, and Express Scribe.

And finally a database, if you want to build an application, its database is a collection of tables. Any organization will have a large number of tables. It will have information about employees. It will have information about inventory it has. It will have information about the projects, the number of rooms. It has what each room has and so on as an example. So a DBMS stores data in a set of tables which is also called a schema.

(Refer Slide Time 07:26)



Suppose I have data in the DBMS. How do I write programs for it? You know some programming languages, right? You must have heard of them at least. C, Java, Basic and so on.

So DBMSes use something called SQL. People pronounce it as sequel, stands for structured query language. Let's see an example of how SQL works. It's quite a simple language actually.

(Refer Slide Time 07:46)

Programs for a DBMS

- SQL
- Structured Query Language

Employee

Emp No.	Name	Dept	Date of Joining
134	Murali	Admin	05/05/2005
221	Deepak	Media	07/08/2007
333	Revathy	Devops	19/06/2011
146	Gaurav	Devops	08/05/2005
356	Sunil	Devops	25/07/2011

Let's say I have this employee table and I want to find out who are the employees working in the Devops department.

(Refer Slide Time 07:57)

Programs for a DBMS

- SQL
- Structured Query Language

Employee

Emp No.	Name	Dept	Date of Joining
134	Murali	Admin	05/05/2005
221	Deepak	Media	07/08/2007
333	Revathy	Devops	19/06/2011
146	Gaurav	Devops	08/05/2005
356	Sunil	Devops	25/07/2011

```
Select Name
From Employee
Where Dept = "Devops"
```

So I can write a query like this. Select the name of the employee from the Employee table where the department value equals to Devops. Select Name from Employee where Department equals to Devops.

So what does this query do? it goes and looks at the table, checks for all those rows where the department value is Devops. So these three rows qualify. The first and the second row do not qualify. So I get the answer as Revathy, Gaurav and Sunil.

(Refer Slide Time 08:37)

Programs for a DBMS

- SQL
- Structured Query Language

Employee			
Emp No.	Name	Dept	Date of Joining
134	Murali	Admin	05/05/2005
221	Deepak	Media	07/08/2007
333	Revathy	Devops	19/06/2011
146	Gaurav	Devops	08/05/2005
356	Sunil	Devops	25/07/2011

Select Name
From Employee
Where Dept = "Devops"

Revathy ✓
Gaurav ✓
Sunil ✓

So this is how you can extract information from the database.

So shall I give you a small homework? So write an SQL query to select the employees whose employee number is 221. Who are the employees whose employee number is 221? What would be the SQL query?

(Refer Slide Time 09:00)



Okay. If you look at the DBMS software, there are a large number of software which allow you to build and manage database management systems. So the relational database management systems are the ones which have been invoked for a long time and typically, when we say a DBMS, we mean a relational database.

(Refer Slide Time 09:17)

DBMS Software

Relational DBMS

- MySQL
- MariaDB
- SQLite
- Ingres, Postgres
- Oracle
- DB2
- SQLServer

I have a small list here. MySQL is probably very very popular and large number of people know about it. This is an open source free database. MariaDB is open source and free. It's a fork out of MySQL. They are almost similar. Then SQLite is a database you will see on your Android phones, for example.

Ingres, Postgres are also very well-known database engines which have been around for a while. These are commercial database management systems. Oracle, DB2 and SQLServer. This is from Microsoft. This is from IBM and this is from Oracle Corporation.

A database engine actually is a very sophisticated piece of software and it is very expensive. These days people are talking about NoSQL database management systems in the last few years especially when data has become very big and e-commerce and web-based systems are coming into place.

(Refer Slide Time 10:15)

DBMS Software

Relational DBMS

- MySQL
- MariaDB
- SQLite
- Ingres, Postgres
- Oracle
- DB2
- SQLServer

NoSQL DBMS

- MongoDB
- CouchDB
- Neo4J

These are some examples of NoSQL databases. MongoDB, CouchDB, Neo4J and so on. But today when we say a database, we mean a relational database and these are the major relational database products.

So that's a quick introduction to what is the database engine. Thank you.

(Refer Slide Time 10:37)

Thank you