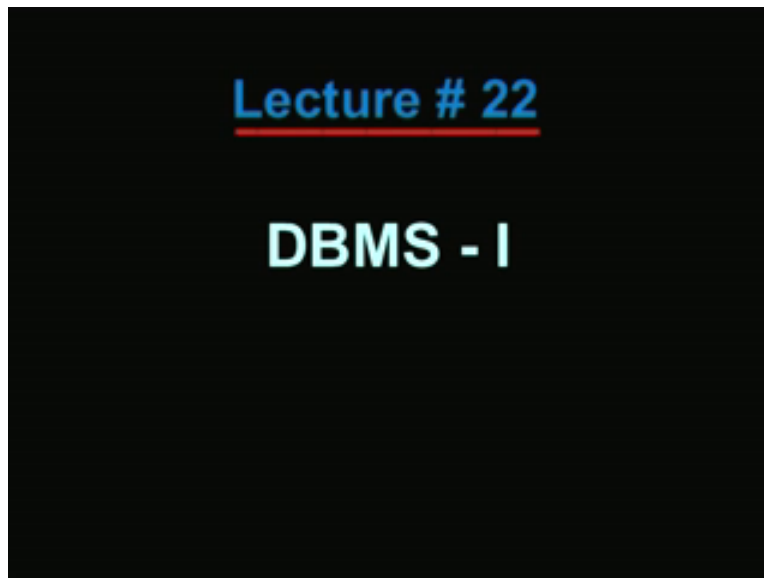


Management Information System
Prof. Biswajit Mahanty
Department of Industrial Engineering & Management
Indian Institute of Technology, Kharagpur

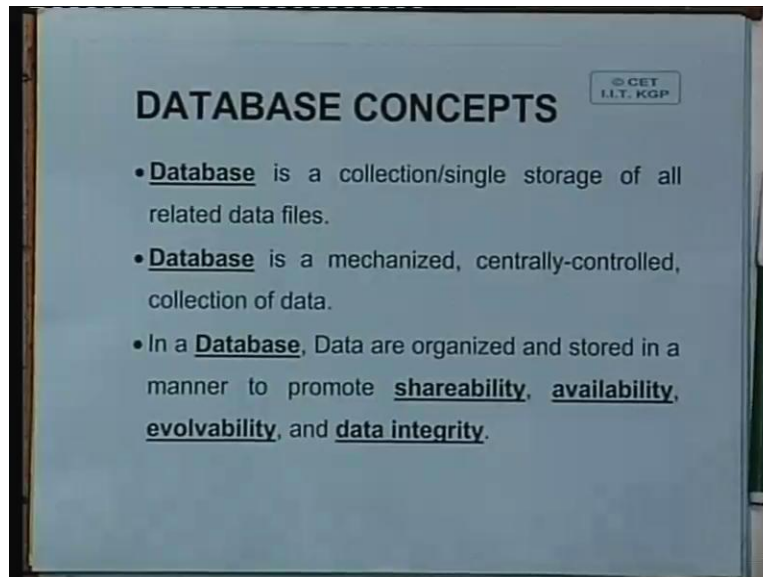
Lecture No. # 22
System Design – II
DBMS - I

(Refer Slide Time: 00:47)



Today let us begin a new chapter on the database concepts that is the database management systems.

(Refer Slide Time: 00:55)

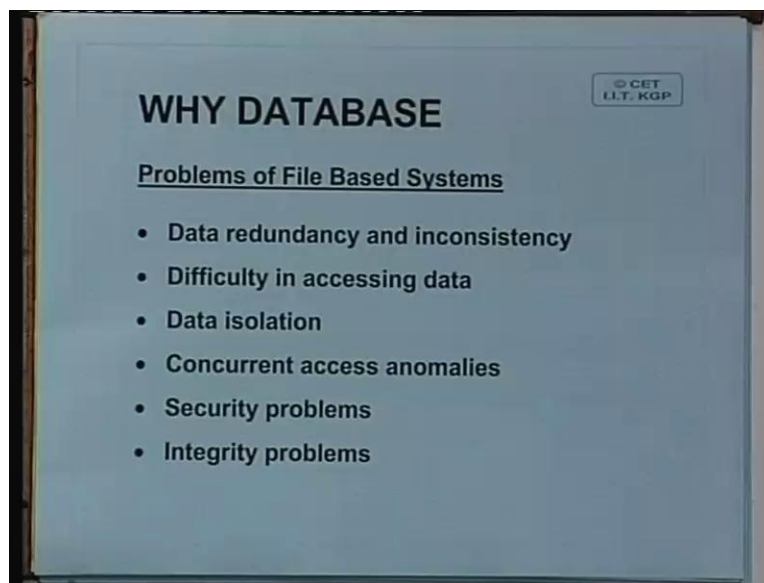


We shall go about this chapter in a particular manner. Initially I would like to discuss about the ideas of what is a database, how it is different from DBMS and DBA that is the database administrator, what are the objectives of database management systems? What are the different kinds of database management systems and why database? I mean what is wrong with the traditional file base systems. So this is how we begin. Afterwards we shall see how to design a database basically designing of database requires two kind of approaches. The first one you can say the entity relationship modeling and the second one we can say the normalization techniques right.

Now therefore we take up the entity relationship modeling, first a particular diagram called entity relationship diagram that will be discussed and thereafter we shall take up normalization. See normalization is a very important topic and it can be covered in two ways the first approach is to cover normalization from a layman's view right. Suppose you do not want to go into the nitty gritty of the concepts behind it. But you can still understand normalization by using some very simple empirical rules. So that approach we shall take first and after we have built a little bit of understanding what is normalization and how normalization and entity relationship modeling gives the same results.

Then we shall go into the mathematical side of normalization right. Then what are the different kinds of normalization how to obtain them mathematically one after another and therefore how an efficient design of DBMS can be carried out towards the end we shall take up structured query language. Structured query language or SQL everybody knows is an very important component of any DBMS system so we shall take up that. Now this we may take quite some time may be say 5 to 6 lecture classes right. So this will be our coverage for the database management systems. Today let us see the initial concepts very beginning we have to have a definition of database. We can in a very simple language. We can call database is a collection or a single storage of all related data files is an integrated concept right is a mechanized centrally controlled collection of data. But we can put it differently in a database data are organized and stored in a manner to promote shareability, availability, evolvability and data integrity. What are these? We shall take up in detail after sometime. Now what is the problem of traditional file base systems?

(Refer Slide Time: 04:28)



See the whenever we have the traditional file base system first and foremost I think I would say the most important thing is the data redundancy and inconsistency. I think this is a very important point I would like to elaborate on this. See what really happens is that whenever you have a traditional file base systems you have the same data probably kept in many places, many

places right. That is usually called data redundancy you are keeping more data than is required. Then inconsistency. Inconsistency means when you are keeping the same data in different files probably they do not match right. So this is known as data inconsistency as I said, I would like to deliberate on them but first let us see the points one by one.

The second is difficulty in accessing data. Naturally whenever your data is there in number of files, it will be difficult to obtain a specific type of information. You see if you just want usually how it happens in a traditional sense wherever you have isolated applications. The isolated applications are developed with some specific requirements in mind. Say in an organizational context, let us say we have isolated applications of let us say payroll right payroll means the payments to the people workers officers all the other people. So it should have that what is the basic pay? What is the dearness allowance? What are the various deductions? How many days person has come to the office? Etcetera, etcetera, is there any over time earning, incentives etcetera. So that's the payroll part.

Then another could be suppose you think of an industrial situation where every job that is carried out in a shop floor. It requires a job card to be distributed to the workers right. So you have several workers anything that is to be produced a job card is given say a machining right. A job card is there and this job card specifically identifies codes reactivity and gives a job card. So we can well understand that we have a job card for every job that is done. So if the job cards come back to the computer section. Then the computer section knows what are the jobs that has been carried out and naturally which machines are used and how much time has been spend on this jobs right and what was the standard time that should have been utilized. A third application could be the machine performance right usually what happens all machines do not work all the time. So it may so happen that some machines are under maintenance some machines are working full time some machines are to be reinstalled or revised replaced so on.

All these data about various machines are kept in another application usually whenever the top management or the corporate office wants to do an analysis, they do not want isolated analysis. See isolated analysis reveals their own story in a summarized manner. What it may say that machine down time is 15 percent then workers incentive earning is 30 to 40 percent of their salary right. The situation of job card distribution is up to whatever was initially thought of in the

production plan right in the beginning of the year. This kind of results can be obtained by taking individual data from isolated applications. But can you find out what are the machines or who are the workers what are the machines they are working on where maximum incentives are being given try to understand.

We now want queries or answers to queries which are specifically related, what the workers are doing who are allotted to equipment where the down time is very high. Which jobs are being done on machines? Which are not working very well? Who are the people who are working on replaced machines right. So if you try to see these kind of reports, they actually tell many other stories which you do not know. Otherwise you see suppose you find that the people have earned incentives right. The jobs are done on equipment which are down. What does it mean? See the machines are down people have earned incentives and the jobs are apparently being done. What are these jobs? Where are they coming from?

So is it a falsification is it a deliberate falsification or what is it you see these kind of overall control figures can be obtained can be obtained not only to find faults not only to find faults, but also to see the trends in operation trends in operation. Say for example between the production and marketing what is the job of production to maximize production. What is the job of marketing to maximize sales but then the two may not match all the time when the production would like to produce their maximum. The maximum sales may not be at that time all right and the sales may be having. So much fluctuation that however the production may like to match they may not able to match it. So what will happen you have to have something called an inventory or a stock in between all right.

Now what is the exact value of the stock invariably in isolated applications? You will find the production will show one kind of stock and the marketing will show another kind of stock. The stock with production and the stock with marketing it is the same stock same finished product or finished goods stock you will find there is a difference. Why this difference because they are managing separately in two isolated applications to suit their purpose. Production would like to blame marketing that we are producing. But they are not taking it into their stock. They do not like to show it on the stock. Because if they show it on the stock because they are not selling they

cannot blame us on the other hand marketing will say we want lot of material they are not able to provide.

You see these battle goes on and therefore you know these kind of isolated applications in away better. Because they do not create problems you can still keep two values both are there you have the production side of inventory you have the marketing side of inventory. Although it is should be same but they are different and information system keeps two different values and its perfectly fine think about production and maintenance. Say you have an equipment. Equipment has failed at a certain point of time. Let us say on a Friday afternoon right you have tried to contact the maintenance people you could not contact fine.

You could contact the maintenance people only at the beginning of the next day probably Saturday or if Saturday and Sunday is a holiday probably on a Monday. The maintenance people came within a few minutes rectified the equipment and everything is fine. Now what is the down time is it according to the maintenance because they were informed on Monday. They would say the down time is only Monday so many minutes and if you ask the production people they would like to put the down time from Friday to Monday morning. Now how to resolve it if you allow two isolated applications production will keep their kind of log book maintenance will keep their kind of log book. You see very easy to match all right. But that is not right you have to have a single down time all right. So that, the analysis by taking production data and taking marketing data should be the same. Is it clear?

This is where I was telling you that the isolated application has you know tremendous difficulty or inefficiency. That is the difficulty of the traditional file base system. How to how to really do this you must have an integrated database right. If you have inventory there should be only one figure. If you have down time there should be only one figure all right. So that it is easy to you know understand or you can have an accuracy of data and a consistent data. That is the inconsistency and redundancy I was talking of. So you can understand all the related things. You can difficulty in accessing data if you want cross application information. For example, who are the workers who are earning incentives high incentives and what are the machines which were there working on this is difficult because you have isolated applications.

You have to merge that two most of an isolated applications are run by different sets of people. They do not understand their own language. The different languages I mean their data definitions may be different. What you are calling as inventory the other fellow may call stock right. So what you are calling as equipment number the other person may call machine identification. That machine identification and equipment number are one and the same thing who will tell. Actually the most difficult thing is you know realistic problem in any organization where the codes themselves are different. See suppose think of the students are given a roll number every student is having a roll number and all your academic marks and your grades or whatever CGPA etcetera. Everything is according to the roll number.

Now we want an application, suppose every hall has got a separate application which got your room number your mess bills and certain other details about you. Now we want to merge the two from a common application, we want all these things available at one go all right. Your hall related details as well as your institute related details. But suppose hall has given you a different identification number and not a roll number all right and that number is not available in the institute database. Tell me how do you combine the two extremely difficult impossible almost until unless you manually insert both the codes you make what is known as a mapping. You have to create a mapping extremely difficult extremely difficult. This is one of the very major problems that plays almost all industry more so in the materials management.

You will find all kinds of materials all kinds of codes and two different branches of the same organization do not use the same code they use a different kind of coding scheme. So the same material is having different codes different material has got sometime same code and all kinds of coding schemes. So it becomes a very major problem of rationalizing this kind of situation. So it is easy to say okay. Well, let us integrate. But it is not that easy to integrate lot of work has to be done before the integration can actually be carried out. Then data isolation data is available this is an example of probably of data isolation data is there. But isolated and could not be integrated. Another problem that usually happens is concurrent access anomalies.

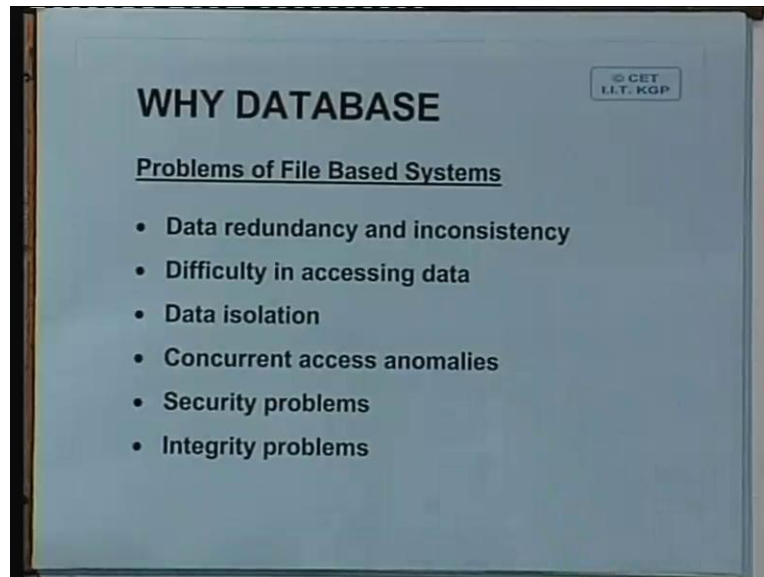
What is concurrent access? Suppose you have the same data same data in a particular file and two people from two different terminals or two different stations are trying to access it at the same time. What will happen? See lot of such problems can happen? Suppose it is a bank and

you are having a traditional file traditional file to keep all the account details. Now you have gone there and you have you are trying to deposit say 1000 rupees at the same time. You have written a cheque of 1000 rupees or whatever two thousand rupees to a person who is trying to take that money. Now it so happened you are depositing that 1000 rupees. So the clerk is entering that 1000 rupees you have given and the person who is taking out 2000 rupees the clerk has given. The cashier has given him 2000 rupees and he is also entering that 2000 rupees deduction.

Now depending on which transaction goes faster goes fast, the results could be different. Suppose your bank presently has got only 1000 rupees all right. See lot of things can happen. Suppose the clerk who the deposit clerk enters, first then everything is fine. 2000, 2000 balanced. Nobody says anything. Suppose the withdrawal clerk enters first then the cheque will bounce. If you do not have so much money in the bank so sorry right. This will happen even with database but if both are concurrent at the same time what will happen? The program will hang all right. The program will hang it does not no because a two cannot access the same file and try to update the same file at the same time all right.

This is the problem of a traditional file without any specific arrangement what the database does database. Nowadays lots of things are happening. But the traditional idea is the database will lock the record. Whenever you are trying to update it the database locks the records and it does not allow anything else to be done on that record all right. Some other day will discuss. The basic idea is that whenever we are talking about a database the a transaction to the database. When the transaction is going on particularly if it is an add or delete kind of transaction no other transaction can continue all right. So specific arrangement has been made in the database to take care of concurrent access anomalies.

(Refer Slide Time: 22:17)



Then obviously there are security related problems if you have traditional files right. Traditional files security related problems like are anybody can easily take out data out of that. Whereas in database you can give specific security measures so that the data is secured. The integrity problems you see all integrity checks has to be done in a database in a traditional file through the program. Because the file will accept anything and everything all right. Say for example, suppose someone writes this date of birth is 29th February, 1983 all right see such a date does not exist all right. So in a database system, this will throw out this. Data should not be there right. Whereas if you keep this 29th February 1983, in a conventional database it will accept right. It will accept not only it will accepts.

So therefore how to check that this is an invalid date by a you have to write specific code in the program to check it. Whereas in today's database systems they have their integrity checks. You can include the integrity checks database will not accept such a date. But sometimes you have difficulty in such situations because I still remember. We had to face one such difficulty once right. The student wrote his date of birth is 29th February 1983. So when you wrote back that such a date is not possible the students send an affidavit signed by a magistrate. That his date of birth is really 29th February 1983. So we have to make special changes in the database to include him all right.

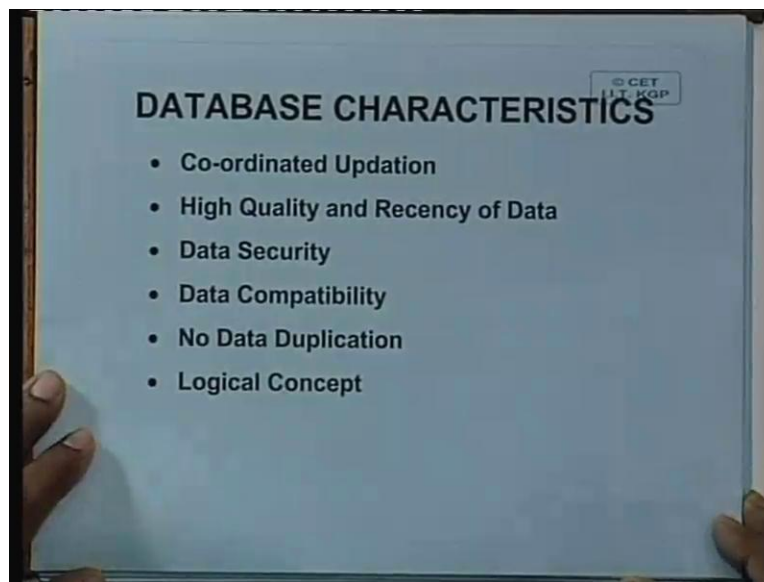
So these are sometimes I mean in other kind of problems. Now there are other kind of integrity problems we had traditional files let me share with you. One of the most conventional integrity problems in the earlier files was putting o instead of 0, you see probably we will never find in today's world. But I do not know even today those punching machines are still there. Suppose you have to enter huge amount of data. Huge amount of data in fact the some companies may approach some firms that only for data entry nothing but that in fact US Government sometimes approaches some Indian firms that look here are our some 1000 telephone directories. You simply enter it in computer validate and give up give us back the data.

Now you have to then employ lot of semi-skilled people and is it really necessary for such work to have sophisticated Pentium 4 computers. Not necessary you can have simple punching machines which may be coasting very less. It has simply a type writer you type it out and it goes into some small cassettes which can later on be converted to some magnetic tapes right. This is an old technology but this technology is still very much there. Now in the earlier days what used to happened some people wherever there is a zero they use to by mistake enter an o. So what will happen you have an o instead of a 0. So suppose something is 900 and he entered it has 9 o o. But look at this 900 is numeric 9 o o is not numeric. So if you want to multiply 900 by 5 it will do 9 o o into 5 nothing will happen. In fact the program will go you know crashing it will stop.

Now try to understand if you do not have a validation check in your program for each and every such data item think of a payroll program. In a payroll it handles with nearly 50 to 100 variables. So those 50 to 100 variables if you do not give in your Cobol program numeric check for each of these variables. Suppose these particular variables numeric check is missing what may happen? When in the date of the night 10000 employee data is being processed their pay slip is being printed right. Suddenly the program stopped. See you have to print 10000 pay slips 2000 has been printed another 8000 is pending the implication could be very high. Because next day morning the pay slips could not be distributed then the company will face a little bit of difficulty is it not.

So there the computer people has to be very, very alert right. See these are practical situations right at the date of the night the immediately what will happen the operations people. They will call up the programmer or the analyst he has to come running to the center, find fault with the Cobol program recompile it and rerun the program these are part of life all right. So these could be situations if in the old days. Today what will happen? If the person enters 9 0 0, the database will simply not accept it. So the data will be thrown out see please understand throwing one data is not that bad even throwing 10 data is not that bad. What will happen? Ten workers slip could not be printed but you can still print 9990. You see in the middle of night if the whole program stops then nothing is printed. Instead if a few errors are there you can always take care of ten special cases by manually or by any other means all right. So these are the things which are very, very important.

(Refer Slide Time: 29:32)



So on the other side which compare to the, oh, oh, okay. I will just tell just one of you inform. So database characteristics on the other side are the first one is coordinated updation right. So whenever you are updating the updation is done in a coordinated manner. That means not only we have between the two file systems but also the say between production and marketing you can decide who will enter what all right. Then high quality and recency of data high quality and recency of data, data security data compatibility no data duplication and a logical concept of

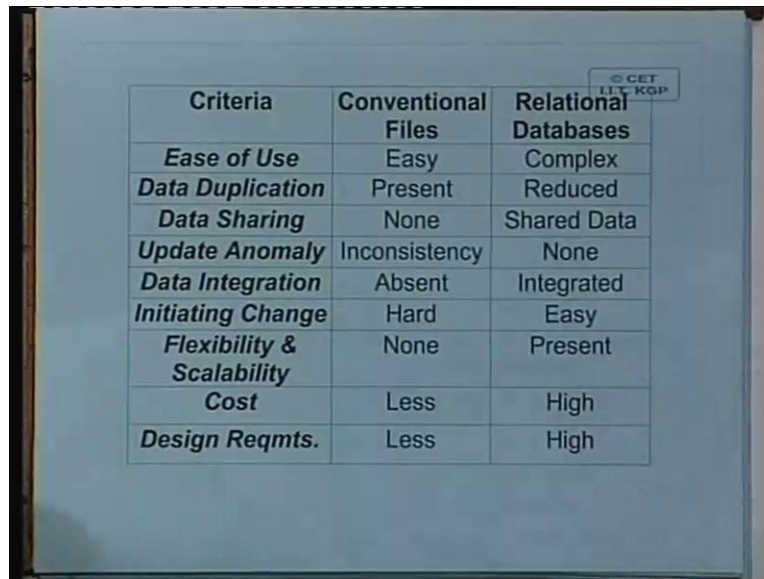
data. So I will explain them one by one. The first one is the coordinated updation that whenever you are updating you are since you have a common field then all the entries are taken together.

The high quality and recency of data recency of data means see if you have multiple files any update you may have to do in all the files many a time what happens whenever you have isolated applications suppose the personal department has all the employee details all right. Now if these data you are also keeping in individual departments okay fine. So you think why I have to go the personal database all the time let me keep my own data. So every department keeps their own personal data unfortunately what usually happens after sometime the personal department because they are a regular department that is their job. They will update their data as and when new things are coming for example a person has unpromotion.

A person has got increments a person has got new degrees etcetera, etcetera all are being updated. Whereas individual departments they it is not their responsibility all right. So they do not enter because they are not entering their data will remain old all right. This is why the question of recency of data comes in. So the idea is you integrate the data all right and with a specific instruction who will maintain the data. The basic concept again is data should be maintained where or data should be updated where it is generated as near to the point of generation all right. With specific permission should be granted that means if I am responsible for updating the data only I will update the data others will not have the access of updation all right.

So will come to that, then security of data can be there data will be compatible production and marketing will now see the same value of inventory they cannot see two different values of inventory. So data duplication as far as possible we should see to it that data duplication is to the minimum and the logical concept. See for every physical database you can define a number of logical databases. I will come to this point will explain in some detail.

(Refer Slide Time: 33:27)



Criteria	Conventional Files	Relational Databases
<i>Ease of Use</i>	Easy	Complex
<i>Data Duplication</i>	Present	Reduced
<i>Data Sharing</i>	None	Shared Data
<i>Update Anomaly</i>	Inconsistency	None
<i>Data Integration</i>	Absent	Integrated
<i>Initiating Change</i>	Hard	Easy
<i>Flexibility & Scalability</i>	None	Present
<i>Cost</i>	Less	High
<i>Design Reqmts.</i>	Less	High

Now this is a comparison between the conventional files and relational databases. The criteria suppose ease of use right. We can say easy complex see the relational database is not that easy to use. So this is a one difficulty and data duplication present reduced fine. So conventional files are known for data duplication data sharing conventional files none relational databases shared data shared across applications update anomaly. There is an update anomaly and data is inconsistent and in relational databases. There is no data anomaly should be data integration absent integrated initiating change is hard in conventional files. If you want to make a change it is difficult whereas it is easy in the databases flexibility and scalability. What is flexibility? Flexibility means change of fields change of data items etcetera, etcetera and what is scalability?

What is scalability? Scalability means upgrading; upgrading by adding new features all right. The scalability considerations are very important. Whenever you take up a database development right say for example if you are thinking of today suppose your role numbers look at your role numbers it is an 8 digit role number. Few days back we have had only a six digit role number all right. See when we have let us say allocated codes for departments allocated codes for department allocated codes for department. That time we had given something like 52, 40 etcetera, etcetera. Different numeric codes today we are giving alpha codes e, e, i, m, etcetera, etcetera right. So the point is at that time while the codes were allocated at the times when the

codes are allocated. We should have seen that there should be provisions for doing these things in an efficient manner. I do not know whether you are getting the picture. Let me let me give you another example.

See the railway reservation system right. Sometimes what happens a particularly in the summer time there is a huge rush and there is a rumor sometime that some summer special trains will be announced all right. Now in view of these that there might be a summer special train lot of people they book ticket in the existing train and been waitlisted. See usually waitlist may within 50 to 100. But if people know that there will be an summer special train. So let me have a booking anyway. So suppose there are some 750 wait listing the earliest software, these that the railway people had there was a difficulty that they kept an arbitrary upper limit of wait listing passengers all right. And there was a little bit of anomaly somewhere in the design right. So what happens the 750th person or even more has been registered taken ticket is given. But the system crashed all right.

So when this system crashed what will happen for next two days nobody got the facility. Is okay? It is not the system crashed means system crashed in Kharagpur the system crashed, that is all over right. So this was an particular incident I am just sighting from newspaper. So what probably had happened when the initial system was thought of that time they could not imagine that it could go up so high all right. The people thought okay. Waitlisted passenger means hundred can it be 750 too high. But how high is high very difficult to tell. See when the computers or personal computers first came the our very own Bill Gates who wrote the you know disk operating system or DOS. Original DOS was written by him MS-DOS. So he kept an upper memory of RAM as six hundred and forty kb think about it.

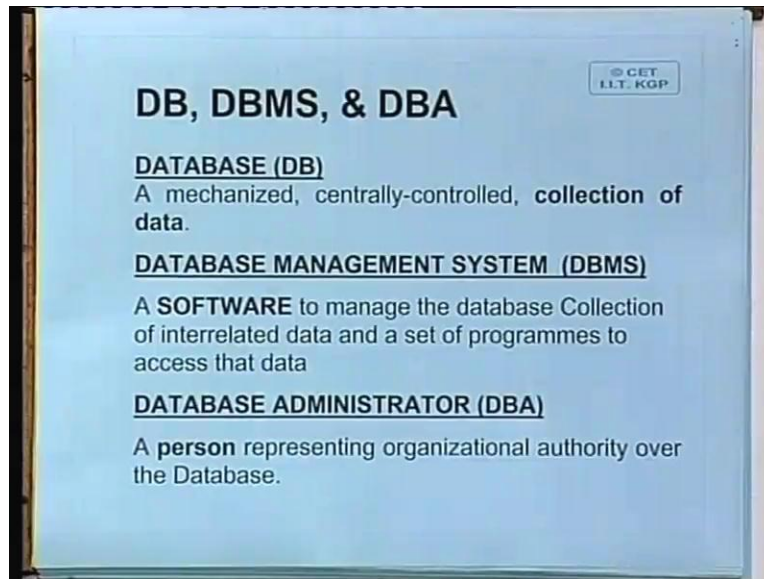
Today even 640 Mb we can think of right when he took an upper limit of 640 Kb. Today it is 1000 times higher 640 Mb. So much difference can be there and particularly in database design. You must always remember that it is the exception that is important rather than the rule. Suppose you want to design a database for student results see you are going and talking to the academic section people. Say you ask that um how about failures of B-tech students the result will be what failure nobody fails in IIT all right. Nobody fails because 99 percent or 99.99 percent of B-tech students pass nobody fails. So we assume nobody fails. No, no, sometimes one or two student

fails in how many subjects oh some one or two subjects like that. So by thinking or believing him completely you keep a provision in your database the total number of backlog subjects that a student can have should be between 0 and 9.

So you keep a provision for one digit. That is all now suppose due to whatever reason a given student has failed in ten subjects. What will happen because you have a single digit provision his data could not be kept in the database. See it is not that his data could not be kept the problem is immediately. You do not know what to do is it not you see people who are working in computer sections are very busy all the time. Now if they have to make a database change see any database change is risky you must understand the importance of the thing because you have the live data of all the students all their current results which are meticulously entered data is validated and entire results is there tomorrow someone wants a transcript you have to print and give him.

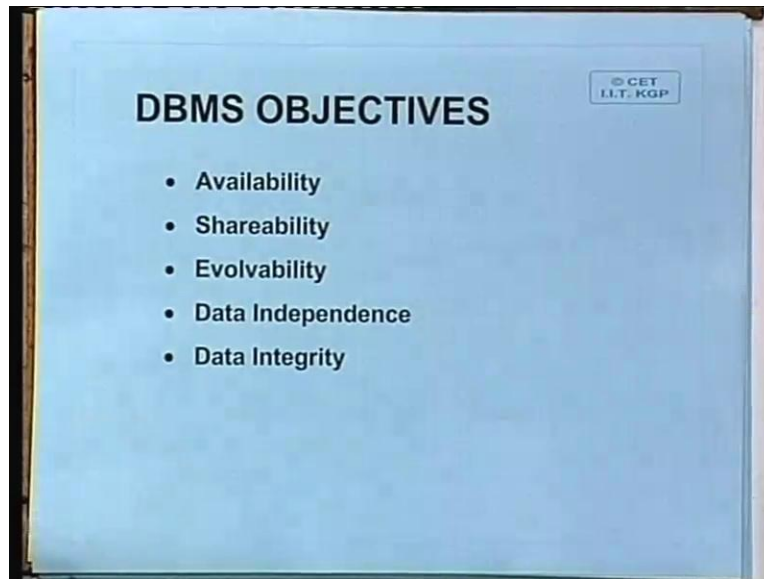
Now suppose you want to make a database design change and suppose something goes wrong that means you have to go back to your back up tapes recover all the data and it will be a crisis situation is it all right. So all the time you must keep in mind these sort of things this is actually called evolvability will discuss later. So those are things then cost and design requirements like it should be you know if you want lot of facilities you have to pay for it today's sometimes some sophisticated oracle software's would be very costly. In fact you know more facility want the better plat form you want more is the price. Suppose you want oracle for personal computers it may not be that costly. But if you want oracles for UNIX base systems you want it for the Sun systems it will be very costly right. So those are the some of the things.

(Refer Slide Time: 42:45)



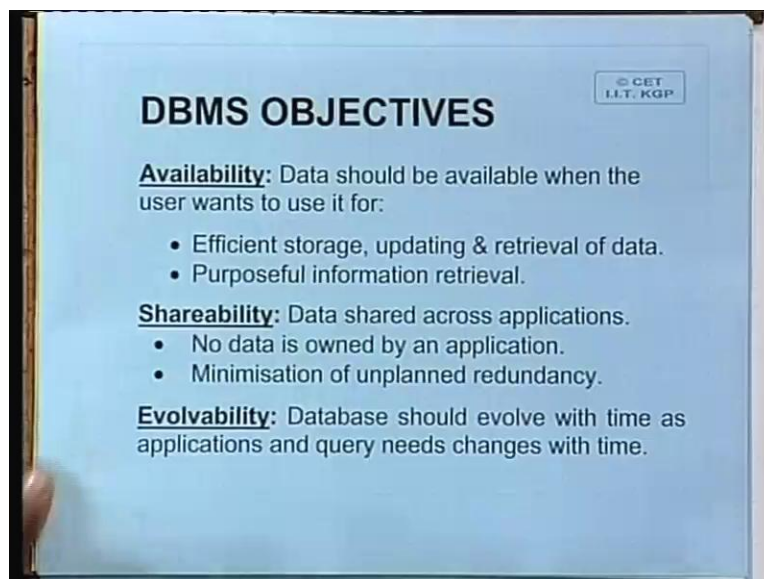
Now between database, database management systems and database administrator. See the first one the database is a mechanized centrally controlled collection of data I have already told you the DBMS is actually a software right. So DB is the data part the collection of data DBMS is a software to manage the database collection of a interrelated data and a set of programs to access that data and database administrator is a person is a person right. Whenever you have a database you must have a database administrator who is actually representing the organizational authority over the database. That means if you have a database that database administrator will look after the activities of database. Means he will see to it that the data is entered and data is managed data is maintained in a perfect manner. So this is the difference between the database, database management system and database administrator will come back to this little later.

(Refer Slide Time: 44:05)



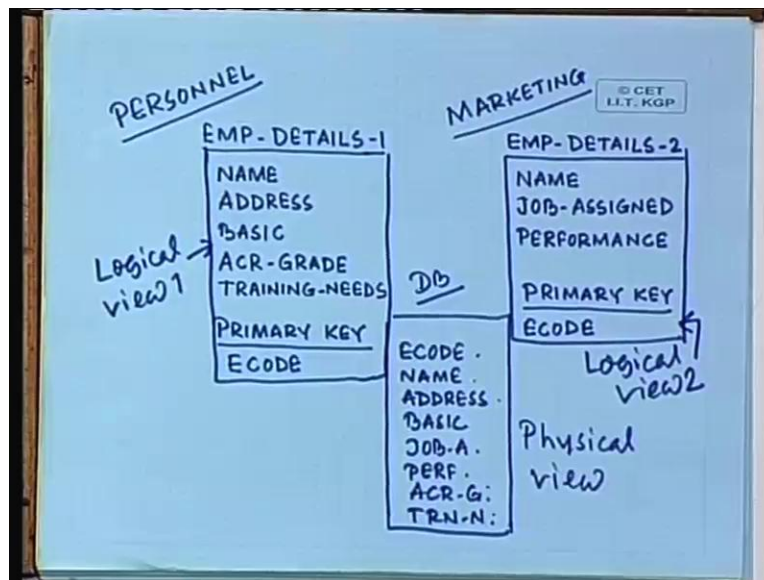
Then let us see the objectives of database management system. So these are the 5 fundamental objectives of a database management system. First one availability, shareability, evolvability, data independence and data integrity right. These are the five objectives major objectives of the database management system. Let us discuss them one after another.

(Refer Slide Time: 44:40)



The first one availability data should be available when the user wants to use it for efficient storage updating and retrieval purposeful information retrieval. Shareability data shared across applications right. No data is owned by an application minimization of unplanned redundancy okay. So these are things and database should evolve with time as applications and query needs changes with time right. So this is what I was talking about while talking about scalability that is database should evolve with time as applications and query needs changes with time. Now these availability, shareability and evolvability. Let us take an example and try to understand in a slightly better manner should I okay right.

(Refer Slide Time: 48:31)



Now I have missed something. I have missed ecode. So ecode also is a part of this databases. Now I have written here two sets of employee details. Now actually employee details are not these only so small usually an in employee details we have somewhere between 50 to 100 fields usually right. So I have written only 5 probably. So it is just a very, very, very, what you say sample. Let us just take only two departments. Now similarly there may be other departments the personnel department keeps employee data. That is the name address his basic pay the ACR grade and the training needs obviously employee code or ecode is the primary key. Now look at these. These ACR grade is a confidential information ACR means annual confidential report.

What was the grade given to these employee in his annual confidential report and what are the training these particular employee needs. This is what is there in the personal department details.

Whereas the marketing department keeps the employee information basically code name job assigned and performance all right. Now see what is what is the need of these basically here you have the employee code you have the employee name what is a job given to him and what is his performance is it okay. Now look at this if they are two isolated files all the difficulties which we have discussed will come in the marketing data may not be updated the name in these two files may be different for the same code two names may be found all these various update anomalies may actually come in all right. So suppose we want to make it an integrated application moment we want to integrate the applications. So we can create a third file which will should have ecode, name, address, basic, job assigned, performance, ACR grade and training needs all right.

So all the fields will come into this. So this is our integrated DB. So integrated DB is having all these employee code name address basic job assigned performance ACR grade training needs. But if I implement these there will be problems. What will be the problem? The marketing people may get access to ACR grade. The marketing people may get access to training needs. Similarly the internal performance reports which the marketing people are writing personal people know about it. They do not want it to happen is it okay. So one of the basic needs of the database system is that see data should be shared we talked about availability data should be available. See we are sharing the data but at the same time we want our data to be available. But we do not want our confidential data to be known to others.

So answer to these is we call it the physical view and it should be possible for every physical view we should be able to create logical views fine. So for every physical view if we can create these two logical views and these logical views are given to personnel and marketing so we are achieving the same thing what we were having earlier. The data is shared the data is available we are getting advantages of databases. But when personnel department looks at these data set, it does not find job assigned or performance right. When the marketing department looks it does not find ACR training needs. In fact if you want, you can give him something like address and basic some additional information which can be shared. Is it okay? So this is the advantage. Fine. We stop here will continue the discussion next class.