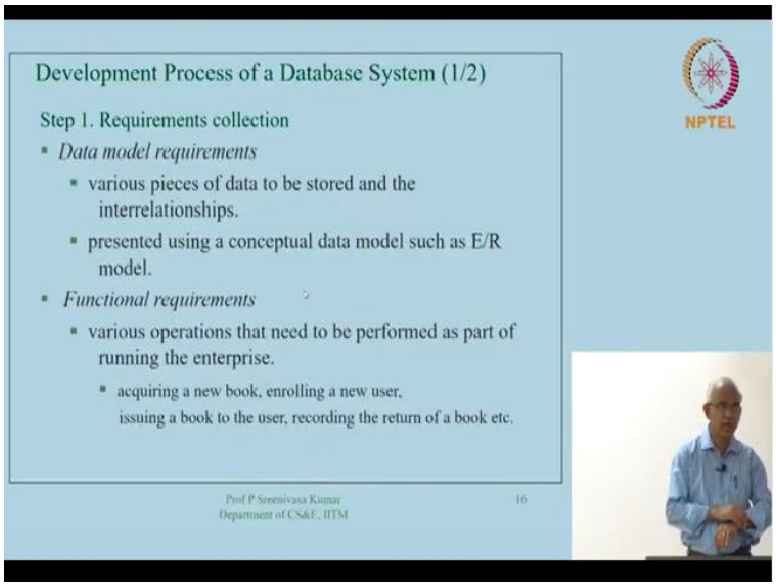
Database Systems
Prof. Sreenivasa Kumar
Computer Science and Engineering Department
Indian Institute of Technology-Madras

Lecture-03
RDBMS Architecture

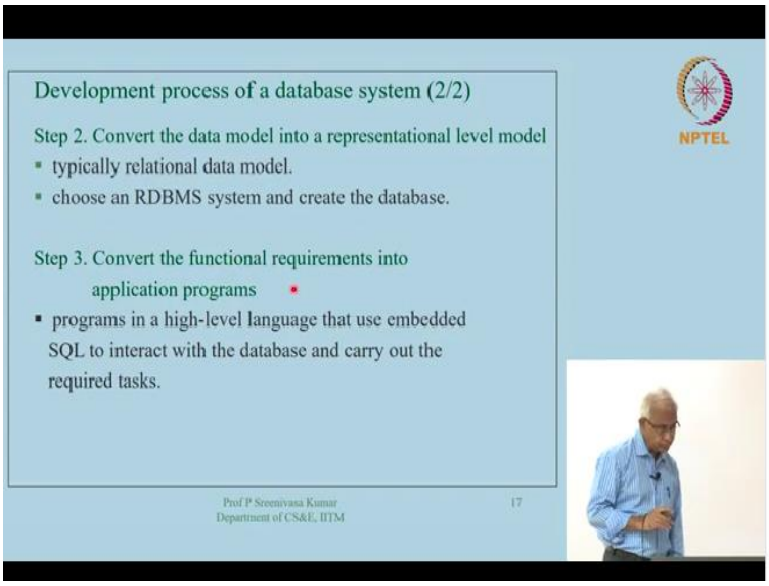
(Refer Slide Time: 00:16)



So, let us briefly recall what we were doing in the last lecture. So, we are looking at the development process of a database system. So, basically it has these steps that we need to first collect the requirements for the system. And then there are 2 kinds of requirements is that first thing is that we need to figure out what are the important entities and associations between them that we need to represent.

And then tell us about the various operations that need to be performed as part of the running the enterprise okay.

(Refer Slide Time: 01:07)

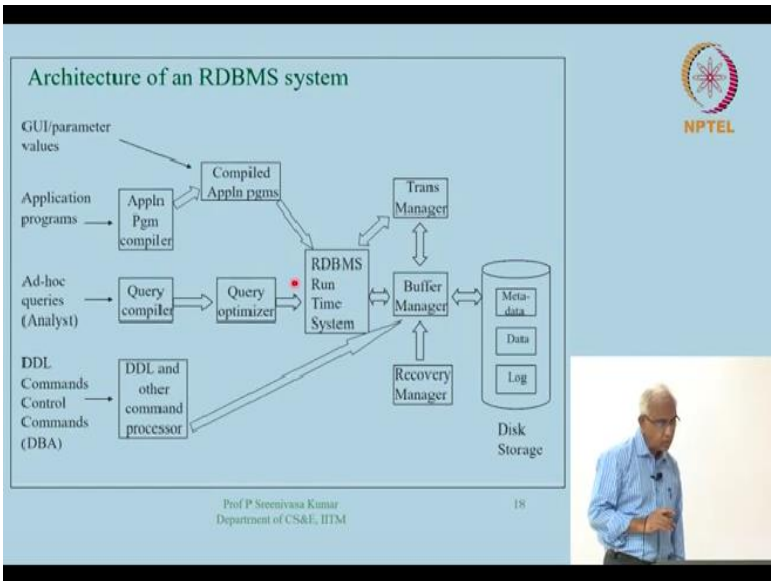


So these requirements need to be collected and then we convert this into a representational level data model which is relational data model and choose the RDBMS, chose means so there are lots of providers of RDBMS right now Oracle, DB2 and Microsoft SQL server so many RDBMS systems are there. So we can choose one of them and then create the database and convert this functional requirements that we are talking about into what are called application programs.

So, these are programs in high level language that use SQL in order to interact with the database system and modify the database system as per requirements of this functionality of the enterprise. The day to day operations of the enterprise are captured by this functional requirements okay. So, in this lecture we will focus a lot on the various subsystems of a database management system okay.

As I was telling you the DBMS is a pretty complex piece of software. And it has lots of these components, lots of various, so we will look at each of these components a little bit in detail.

(Refer Slide Time: 02:49)



In some sense, this picture that I am going to show here gives you a complete you know kind of introduction to this entire course because this has the various components of this particular system architecture are going to be covered in various modules of this course and so, this kind of gives you a complete introduction to the entire set of modules that are going to come up in this in this course okay.

Now, the ways things in this picture, is this picture clear, visible okay at this place, we have a picture that shows a disk. So, this is the disk storage. So, this is where ultimately the database and all its related information is going to be stored. In fact, there can be several databases that will be stored. So, we can roughly divide that into metadata, data, and log, we will talk about log a little while later.

Basically, metadata is the schema information about the database system all the structure information, the names of relations, names of attributes, their data types, other constraints that they are required to satisfy all that information is this metadata. And then we have actual data that will be the various tuple instances that are there as part of the relations, all actual pieces of data names, roll numbers and things like that all those phone numbers, etc. all the various bits of data.

So, that is the data and then we also have log will talk about a little later. Now at this moment, we will let us let us focus here. First we will try to understand as to how the system a particular database system will be set up, and then how it will be used right. So we will first focus on how to set it up, in order to set it up just now we looked at, you know the steps involved in setting up a database system, right.

So after the first step we have all the requirements in our hands. And typically these requirements are going to be represented in a high level conceptual level model, like an ER diagram. And then we sit with the stakeholders, engage them in a discussion and explain clearly the requirements as we have understood from the discussions and get a go ahead from the stakeholders.

This is a very important step because we need to have the clarity about the requirements before we proceed to build the system. If there is some ambiguity in the requirements and we have not been able to have a clear discussion and convey what we have understood about their requirements, and if there is a misunderstanding at this stage, then it is likely to affect the design of the entire information system.

And then might actually cause a lot of problems later on when we build the system. So, for any engineering enterprise is design is a very important thing, right. So, at the design stage we must pay a lot of attention and then engaged with the end user and get clarity into the whole process and clearly ah convey what we have understood and whether we have understood what we have understood clearly on all this, we should ensure and get a go ahead from the end users.

And once we do that we typically convert that into convert the information that we have gathered into the representational data model as I was just now telling which is the relational data model, because once we convert them relational data model, the RDBMS will come into picture. So, RDBMS as you all know, provides you or implements the SQL standard. So implements the SQL standard. So, SQL provides you what are called commands that are DDL commands that are data definition sub language.

We do not this is part of a SQL itself earlier it used to be called data definition sub language. So, basically there are statements like you will be already heard about this statements like create table is a command which you actually create the relation etc. So, these are the kind of commands that are part of the DDL sub language as part of the SQL standard and using that we will lay out we will actually set up the skeletal structure of the relational database system.

So, all the relations that need to be present will all be designed and then all. So, there is a DDL and command processor that will take up this set of SQL commands that will create the relational structure and convert that into low level details and then actually you know, set up the required information structures back in the disk. So, you can see that the RDBMS has a compiler.

So it has to understand SQL right. So, there is an SQL compiler that has to be part of the RDBMS system and SQL statements will be converted internally into a convert has to be processed and they have to be converted into appropriate information structures on the disk okay. So, that is the who does this. It is the database administrators, the database administrator will set up this skeletal structure of the database.

And then we will see how the database will get used okay. So right now there is not any data in the in the database just the structure. So, what are the very all the tables are kind of empty, but we have all the table structures and the constraints that need to be satisfied by these columns and these details are all in place. We will of course, later study in detail as to how exactly these commands looked like okay.

Now let us go to the top portion of this picture here. The typical use of the information systems that you would have encountered when you know interact with various systems in the real world is that there are people who will you know gather some important pieces of information from us. And then you know interact with information system and provide some service for us.

Like, for example, when you go to the library, they will ask you, okay, where is your ID card and what is the book that you want to borrow, okay, these are 2 pieces of information, they will ask, and then they will key in this information, and then say that, okay, yes, this issue, this particular

in the fact that this book has been issued to you is now recorded. And so you can take this right. So what they are essentially doing here is use the information system.

But then they are interacting with that through a graphical user interface, feeding some parameter values and invoking some program inside that okay. So these programs are what I have just now told you as these are the ones that are called application programs, the application programs are the ones that kind of provide all the functionality that is required for the day to day operations of the enterprise okay.

So, if you focus on library again. So, the various operations that it is involved on a day to day basis would be enrolling users, issuing books, recording returns, collecting fines, procuring books, recording the book has arrived, etc. So, these are the various things right. So for each of these things, there is going to be a high level program that will be written. And it is written in either C, Java, C++ etc. and as part of that program, it will somehow invoke SQL.

And interact with the database in order to send those in pieces of information that he has collected to the database okay. So, that is why application programs go through an application program compiler. So, now these application programs have SQL inside it, right. So obviously, it has to be pre processed, so that you can remove all the SQL commands from it and then send the remaining program to the usual you know C compiler or C++ compiler.

Let stick to one language let say C. And then the program has to be compiled, but then it all there are these embedded SQL commands. So obviously, these SQL has to be replaced by appropriate function calls, so that the program becomes complete, right. So these function who provides the function who I mean the RDBMS vendor, RDBMS vendor would have provided these implemented function interface, and these functions have to be then used and then the program has to be compiled into an executable okay.

So the user, the application program developers, they are called application programmers. The application programmers would use a high level language like C and use SQL and submit that thing as a application program. But then the RDBMS system has to use a normal C compiler and

then the functions that are been provided by the RDBMS. So RDBMS vendors would supply these library functions that will be useful for implement in these SQL commands that are used inside the host language programs.

So, these so together this will be compiled into a compile application program and you keep them in store to say, okay. So, these are also kind of, you know, stored inside on the on the stored programs, right. So, when these people behind the counters are operating what they will, what they are essentially doing is invoking one of these application programs and supply the parameters to them.

So, we will use them, you know so we can probably call them as parametric users. So we basically use the system by simply supplying the parameters, they probably do not have a full idea about what the system, how the system actually works and things like that they do not need to actually these are the people who are operating the system okay. So, when they do that, this compile application programs have to be taken up by this RDBMS runtime system.

And it has to actually run the required SQL functions through these back end and then ensure that the data on the disk changes appropriately right. So all that is involved in this particular chain of activity. Now okay, let us now look at we will probably come back to this again. But let us look at this. What is this, is apart from the commands that are put inside an application programs.

Usually the database system will have another interface through which we can write and submit through which we can submit SQL commands, SQL queries. Like for example, you might suddenly you know let us say we have you know, how many number of computer science books have been issued out in the last 1 month, somebody wants to know okay. And so this SQL later we will see will provide you the appropriate language means to translate this into a query.

And that SQL query will have to be managed by the you have to be put through a query compiler and then appropriately run on the data so that we get the results back and then give them there is a study analyst that who is analyzing as to the usage of the of the database. How okay. Now here

is another interesting thing SQL as we will later see is a declarative language is a largely declarative language.

So, what do we mean by declarative language is that it gives you a way of specifying what you want without actually telling how that information has to be obtained okay, it gives us such a kind of a, you know, mechanism so that you can specify what is the information that you want from the information from the database system without exactly specifying how that information has to be, you know actually put together from the database.

The information that you are looking for, might be actually lying in multiple relations, multiple files and things like that. So, those are low level, those are other details, but you know SQL gives you a way of indicating as to how indicating what exactly is required without telling how it has to be obtained. Now, the how part of obviously is important. So, what exactly happens internally is that SQL queries will be converted into what are called relational algebraic expressions.

They will be converted internally into relational algebraic expressions. So we will when we go to the relational data model will study what this relational algebra is all about. But at this stage, we can just understand that it is something like alphabetic expression, we have a algebraic relational algebraic expression. And when you run, execute the relational algebraic expression will get the results that we want we are looking for okay.

Now, the reason why we actually use relational algebraic representation for the queries is that once we have a algebraic expression, you know just like an arithmetic expression can be executed in multiple ways right. If you have a bunch of arithmetic operators and then there is a arithmetic expression, you can imagine that there are multiple ways of actually evaluating that, right. You might evaluate this operator first, then probably operate a sub expression later the first and then etc.

There are multiple orders in which you can evaluate the expression. In a similar way once you convert this into a relational algebra expression, there will be multiple ways of executing that. And some of them will be fast, some of them will be slow and things like that. And so there is an

opportunity for us to look at these possibilities of running that, evaluating that expression and optimize the expression.

In such a way that we will lose as small amount of time as possible to actually get the information. So, the query optimizer module that is there here essentially contributes to this activity that it will focus on this relational algebra expressions that are coming as a result of this SQL that are being SQL queries that are being submitted here. And then we will see whether there is an opportunity for improving the what is the plan in using which we can run that particular query.

And then it will give it as to that particular method to the runtime system, so that it can actually run the query in that particular way using the data in the database and then give back the results okay. So that is so we are going to focus, we are going to study this relational algebra and then how exactly this relation algebraic operations are actually implemented by the RDBMS runtime system etc. with later on in the course okay.

Now, any questions, so in this picture we looked at the how the database is first set up, and how queries can be run on the database right. And then what are these application programs, application programs are to be compiled into compile application programs and then store so that they can be repeatedly invoked to carry out the day to day operations of the enterprise. Now, since we are repeatedly invoking these programs, we have to be very, very careful about how these programs run.

They should run correctly, and they should run efficiently. So thorough testing of this application programs is essential before we adopt them as correct application programs. Now, what you can see here is that once the database has been set up, it is through these repeated invocations of these compiled application programs, the data actually gets accumulated into the database right. So, for example, as I was telling you about the library information system.

So, as day to day operations keep happening, a lot of issue a lot of data about who has issued what book and what book is inside the library, what book is not there in the library, all that you

know, will now started getting recorded into the database here, and it currently reflects the current state of the affairs of the library right okay. Now, let us go a little bit more into this other parts of the database management system.

Now, in order to understand this is transaction manager, so he has just written as trans manager, but it is transaction manager, I gave a brief idea about transaction server, but essentially a transaction is a logical unit of work that has to be done in entirety okay. So, the nature of that work is such that you either do it completely or do not do it at all. That is what we say as a logical unit.

So you are kind of the think of the whole thing as one atomic operation. It has to be done completely or do not do it. For example this is best explained using the example of a bank. So if you are transferring 1000 rupees from As second to the Bs account, it has to be done in its entirety you cannot reduce 1000 rupees from As account and not credited to the Bs account right.

It has to be done otherwise A would end up losing money and B would not get the money and the transaction is not complete right. So, such kind of units are what are called transactions. Now, it is very important for the database to recognize these operations as atomic pieces of operations and ensure that they are run in their entirety. Now, of course in practice, these programs have multiple steps in inside them.

First, you have to check whether As account has at least 1000 rupees or more and at least does it have the required amount and then you have to reduce 1000 rupees from the fellows balance and then upgrade the balance and then open our whatever these account and add it and then store it as Bs balance etc. There are multiple steps involved and a database actually might fail the system that is operating this whole programs might fail anytime.

But then the RDBMS takes the responsibility saying that these pieces of these things that you declare as transactions will be run in their entirety. So, in order to do this the RDBMS has to take certain measures internally. So, that in case a system failure occurs some transaction which has

been partially done okay would not have been recorded in the database. So, it has to take measures.

But that if a transaction has been partially completed then its effect is not there in the in the database that is sitting on the on the disk okay and if some transaction has completely run and from your side you have given a go ahead saying that yes, I have done the transfer, then whatever the required changes have to be permanently recorded in the database right. So, all this, the database has to ensure.

So, in order to do that, it uses what is called a log okay, a log is some system, we will again have to go into the details later on there are multiple kinds of logs that one can maintain is essentially a system where information about the updates that are being done to the database are recorded in a separate place. So that when there is some issue of this kind, where you know, you have to either undo the operations of certain transactions or etc.

It can be done making use of this information that is stored in the in the log okay. So log is some kind of mechanism or in fact it is a file where we will store appropriate information about the running of the transactions. So, that when there is a situation of this kind where some failure has occurred, and we have to come to bring the database to a consistent state information from the law can be made yourself. So it is the recovery manager that comes into the picture when there is a when there is some kind of a crash in the in the system.

A recovery manager is the one that gets the control of the database system after it recovers from a crash. A crash might occur a power might go or a disk might fail. So you may have to stop the operations of the database suddenly. So at that time once you have restored the situation, the recovery manager is the one that gets control first from the, it is the first one that gets the control of the whole system.

And what it does is to be basically check as to what was the situation of the database system at the time of crash and what was the information that was stored on the log and taking all these things into consideration, it will bring the database system to a consistent state, so that it is

normal operations can be the issue okay. So that is the role of a recovery manager. Now, we also have what is called a buffer manager here.

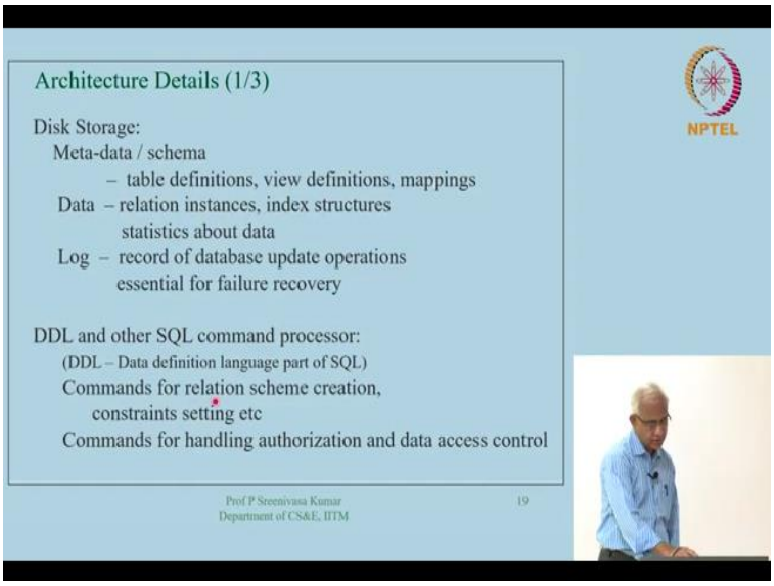
The reason is that though the okay the entire data is actually stored on the disk right. And the data is actually stored as files. So the RDBMS will require to update parts of these files to reflect whatever it is doing. So it has to keep on requesting chunks of the files from the disk. So, there is a buffer manager which will buffer all the information on memory and then keep growing.

So, you will some of you might have done the operating systems course. So, you have studied this paging mechanism right, paging while dealing with the disk systems, the disk systems are slow and so you want to store chunks of the files into the main memory. So, that operations in the main memory can carry on but then as operations update as the memory operations you know happen.

And then these information is updated, it has to be stored back into the into the database right. So, this is how so, it is in that context we will get what is called a buffer management. Now, this is very interesting to see that the RDBMS has to play some part of the operating systems roll because it has to basically implement this paging mechanism there okay. And we will later on study as to why, you know it cannot really rely on the operating system in order to do this paging mechanism.

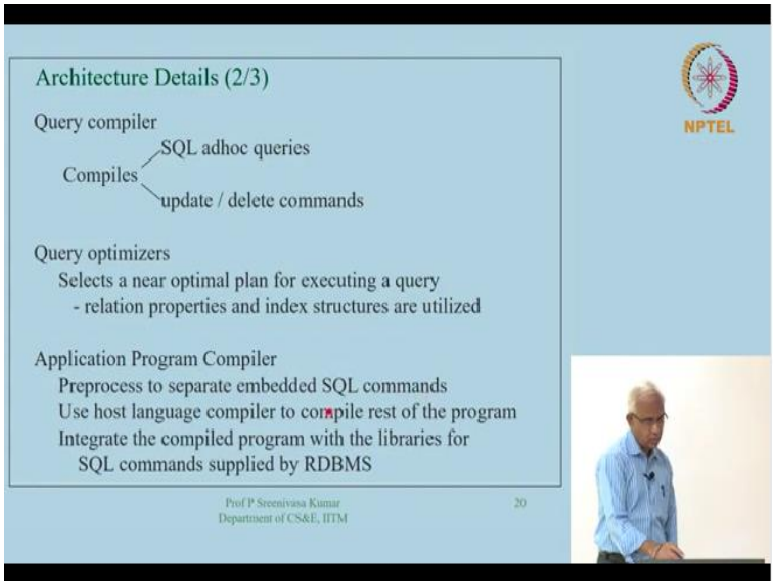
It has to have it really needs to have a tighter control on how information flows from the memory buffers to the disk refers to the disk. We will study it as part of this transaction management and recovery module. Good so, I suppose you got a fair picture of what are the various subsystems that are involved in a RDBMS system.

(Refer Slide Time: 35:11)



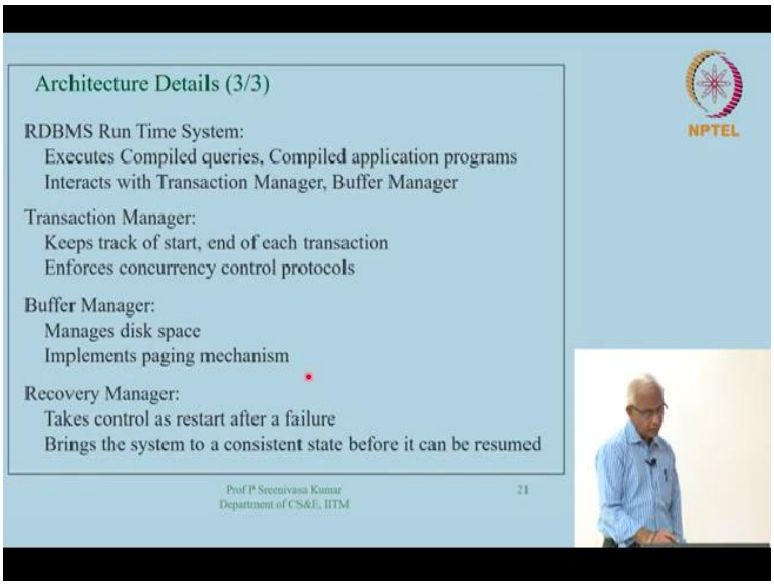
Now, whatever I have been talking about is there in the next few slides, like what is disk is storing and then what does the DDL, SQL command processor will do. It creates the relational schema, the constraints. It also handles authorization and data access control.

(Refer Slide Time: 35:38)



The query compiler compiles SQL queries and also update delete commands. The query optimizer selects a optimal plan for executing a query. As I was telling you the queries the presented as a relational algebraic expression. Then this is the application program compiler that the processes the application programs uses the host language compiler, and also integrates the compiled program with the libraries of the SQL supplied by the RDBMS.

(Refer Slide Time: 36:20)

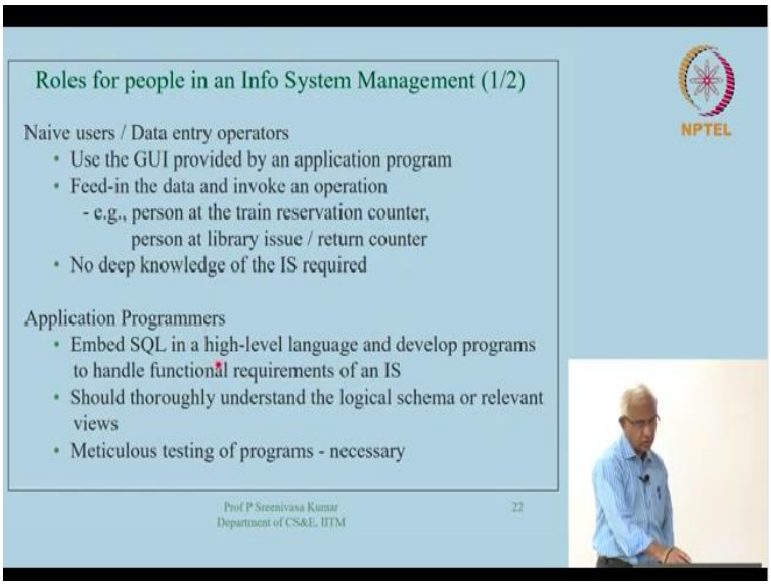


And the runtime system executes the compile queries come and they compile application programs, etc. The transaction manager keeps track of interacts with the runtime system. It keeps track of the start end of each of these units of work, the transactions and it enforces what is called concurrency control protocols. We will study them later in detail later. Basically, what it stands for is that when multiple transactions are submitted to the DBMS system.

They have to be concurrently processed, but with the database has to ensure that at any point of time the database is in a consistent state. Then buffer manager manages the disk space kind of implements the paging mechanism. The recovery manager as I was telling you takes control as after the restart of the system after a failure and brings the system to a consistent state before it can be used for normal operations okay.

Now let us look at briefly look at the various people that are involved. What are the roles for people here okay.

(Refer Slide Time: 37:45)

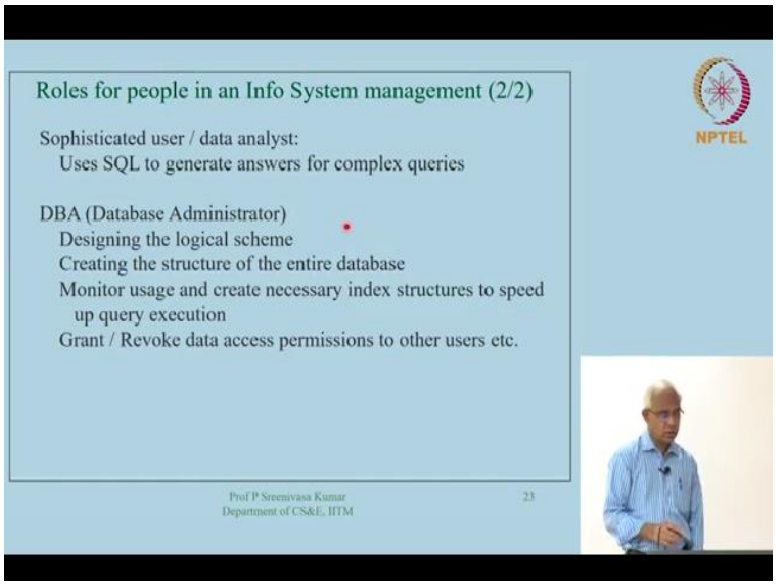


So there is this name users are database data entry operators. I was standing there also. So they basically use the GUI provided by the application programs, feeding the data and invoke in operation. So, these are the people who and whom we encounter at the train reservation counters or the library issue written counter etc., they invoke the application programs, they do not need to have deep knowledge of the information system.

They only have to know how to invoke this application programs, then we have application programmers who have to be thoroughly knowledgeable about the logical scheme of the entire database. And then they are the ones that are responsible for translating the functional requirements of the database into application programs. So they have to use high level language and develop programs to handle functional requirements.

And meticulous testing of these programs is absolutely necessary because they will repeatedly made use of.

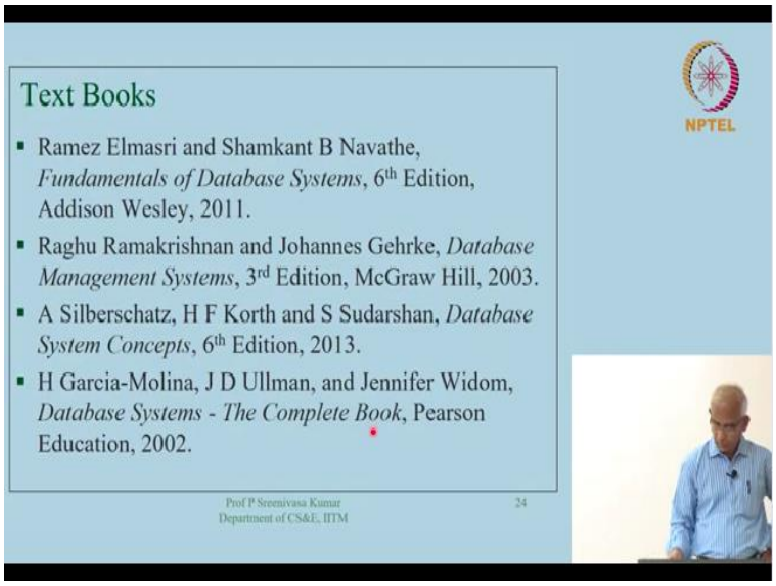
(Refer Slide Time: 39:00)



Then there is a sophisticated users or data analysts who have a good idea about complete idea about what is the database structure is, they use SQL to generate answers for the complex queries. And then we have the database administrator. The DBA is also called database. DBA is responsible for designing this logical scheme, creating the structure of the entire database monitor its usage.

And if necessary do some performance tuning by creating necessary infrastructures to speed up some application programs. And this person is also is very important in the sense that he or she would grant in our revoke data access permissions for all the other users. So this has to be the very trustworthy person because he has complete control of the entire database. So, these are the various roles that people play in the information systems. Good. So, with this I we have come to the end of the introduction module of this particular course.

(Refer Slide Time: 40:23)



Here as the set of s that you can use . These slides of course will be available to you so that you can look it up later on. It is kind of interesting to us to observe that in the 3 of the 4 books we have an Indian author Shamkant Navathe from Georgia Tech. Raghu Ramakrishnan used to be at Wisconsin. Sudarshan is at IIT Bombay. So, all these books are available. The last book is also available as all of them as available as Indian nations.

So, while we are talking about the various concepts in the class, I urge you to keep reading the books okay, so with that we will stop here.