Instructor Notes:

Instructor Notes:

Lesson Objectives

- In this lesson, you will learn:
  - What Is Database and DBMS?
  - Data Models
  - Properties of RDBMS

Instructor Notes:

What is Data?

- Data (plural of the word datum) is a factual information used as a basis for reasoning, discussion, or calculation
- Data may be numerical data which may be integers or floating point numbers, and non-numerical data such as characters, date etc.
- Data by itself normally doesn't have a meaning associated with it.
  - e.g.:- Krishnan ,01-jan-71,15-jun-05,50000

Instructor Notes:

Information – Related Data

- Related data is called as information.
- Information will always have a meaning and context attached to the data element.
- When we add meaning and context to the data it becomes information.
  - Employee name: Krishnan
  - Date of birth: 01-jan-71
  - Data of joining: 15-jun-05
  - Salary: 50000
  - Department number: 10

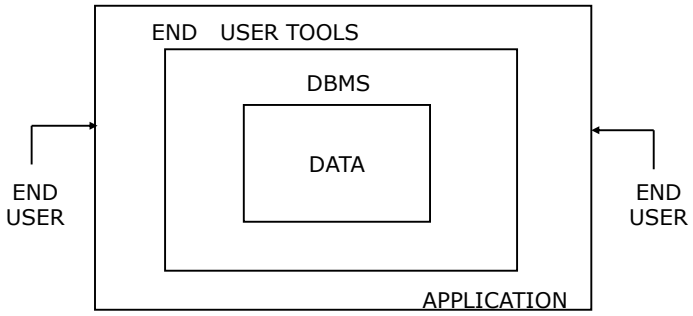
Instructor Notes:

Start with the need of Database. Talk about the 3 tier architecture and then move to the data storage.

Explain the importance of data management in brief and then introduce this slide. This will cover WHY part and then we come to WHAT part Explained on the slide.

Introduction to Database

- DATABASE - A set of inter-related data
- DBMS - A software that manages the data
- SCHEMA - A set of structures and relationships, which meet a specific need

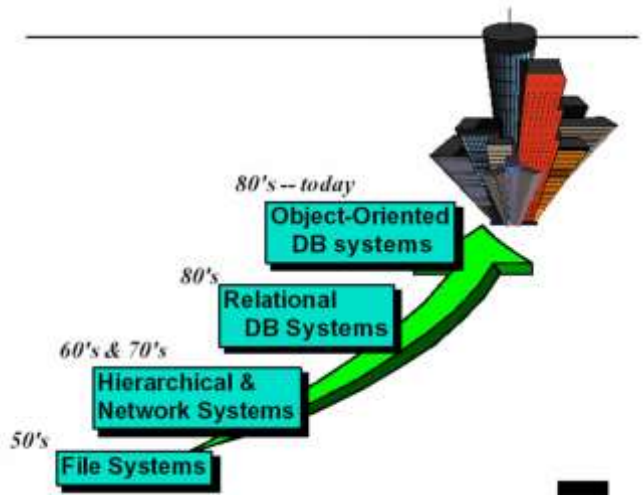


Introduction

A set of inter-related data is known as *database* and the software that manages it is known as *database management system* or DBMS. Hence DBMS can be described as "a computer-based record keeping system which consists of software for processing a collection of interrelated data". A set of structures and relationships that meet a specific need is called as a *schema*. The database is centrally managed by a person known as the database administrator or the DBA. The DBA initially studies the System and accordingly decides the types of data to be used, then the structures to be used to hold the data and the interrelationships between the data structures. He then defines data to the DBMS. The DBA also ensures the security of the database. The DBA usually controls access to the data through the user codes and passwords and by restricting the views or operations that the users can perform on the database.

Instructor Notes:

Evolution of Database



Instructor Notes:

Database Management System

- A database is a collection of logically related information.
- Database Management is the task of maintaining databases so that information is readily and accurately available.
- The software required to perform the task of database management is called a Database Management System (DBMS).

Instructor Notes:

Link this to the need of Database that you talked about before first slide. You can also compare redundancy and integrity problem in shared files.
Transaction control feature (on next slide) and SQL Language features can only be obtained in DBMS.

Introduction to Database

- Control of Data Redundancy
  - Same data is stored at number of places
  - DBMS helps in removing redundancies by providing means of integration.
- Sharing of Data
  - DBMS allow many applications to share the data.
- Maintenance of Integrity
  - Refers to the correctness, consistency and interrelationship of data with respect to the application, which uses the data.

Characteristics of DBMS

Some of the characteristics of the DBMS have been discussed below:

Control of Data Redundancy

When the same data is stored in a number of files it brings in data redundancy. In such cases, if the data is changed at one place, the change has to be duplicated in each of the files.

The main disadvantages of data redundancy are**:**

Storage space gets wasted.

Processing time may be wasted as more data is to be handled.

Inconsistencies may creep in.

DBMS help in removing redundancies by providing means of integration.

Sharing of Data

DBMS allow many applications to share the data.

Maintenance of Integrity

Integrity of data refers to the correctness, consistency and interrelationship of data with respect to the application that uses the data. Some of the aspects of data integrity are:

Many data items can only take a restricted set of values.

Certain field values are not to be duplicated across records. Such restrictions, called *primary key* constraints can be defined to the DBMS.

Data integrity, which defines the relationships between different files, is called *referential integrity* rule, which can also be specified to the DBMS.

Instructor Notes:

Characteristics of DBMS

- Support for Transaction Control and Recovery
  - DBMS ensures that updates take place physically after a logical transaction is complete.
- Data Independence
  - In DBMS, the application programs are transparent to the physical organization and access techniques.
- Availability of Productivity Tools
  - Tools like query language, screen and report painter and other 4GL tools are available.

Support for Transaction Control and Recovery

Multiple changes to the database can be clubbed together as a single 'logical transaction'. The DBMS will ensure that the updates take place physically only when the logical transaction is complete.

Data Independence

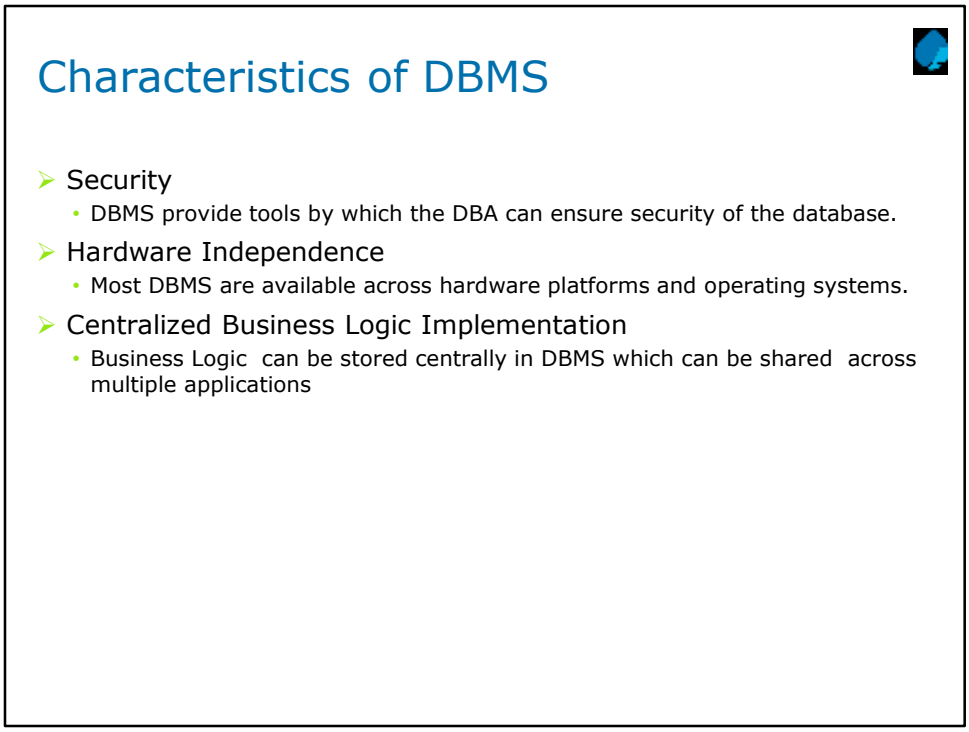
In conventional file based applications, programs need to know the data organization and access technique to be able to access the data. This means that if you make any change in the way the data is organized you will also have to take care to make changes to the application programs that apply to the data. In DBMS, the application programs are transparent to the physical organization and access techniques.

Availability of Productivity Tools

Tools like query language, screen and report painter and other 4GL tools are available. These tools can be utilized by the end-users to query, print reports etc. SQL is one such language, which has emerged as standard.

Instructor Notes:

Explain why security is essential. Give example like different roles in an Organization would like to access different data from the database.
E.g.. Accounts people should have access To payroll data but other employees should not be given access to it.



Security
DBMSs provide tools by which the DBA can ensure security of the database.

Hardware Independence
Most DBMSs are available across hardware platforms and operating systems. Thus the application programs need not be changed or rewritten when the hardware platform or operating system is changed or upgraded.

Instructor Notes:

Discuss the various definitions of a Model.

Explain how it can help them to explore the requirements further.

Definition of a Model

- An integrated collection of concepts for describing data, relationships between data, and constraints on the data used by an organization.
- A representation of 'real world' objects and events, and their associations.
- It attempts to represent the data requirements of the organization that you wish to model
- Modeling is an integral part of the design and development of any system.
- A correct model is essential.

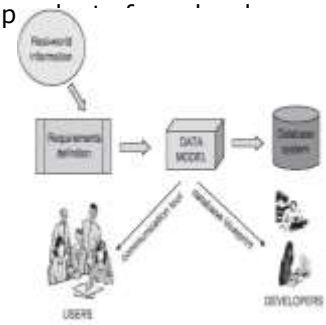
What is a model?
A model serves two primary purposes:
As a true representation of some aspects of the real world, a model enables clearer communication about those aspects.
A model serves as a blueprint to shape and construct the proposed structures in the real world.
So, what is a data model? A data model is an instrument that is useful in the following ways:
1) A model helps the users or stakeholders clearly understand the database system that is being implemented. It helps them understand the system with reference to the information requirements of an organization.
2) It enables the database practitioners to implement the database system exactly conforming to the information requirements.

A data model, therefore, serves as a critical tool for communication with the users and it also serves as a blueprint of the database system for the developers.
Without a proper data model, an adequate database system cannot be correctly designed and implemented. A good data model forms an essential prerequisite for any successful database system. Unless the data modelers represent the information requirements of the organization in a proper data model, the database design will be totally ineffective.

Instructor Notes:

What is Data Modeling?

- Data modeling is a technique for exploring the data structures needed to support an organization's information need.
- It would be a conceptual representation or a replica of the data structure required in the database system.
- A data model focuses on which data is required and how the data should be organized.
- At the conceptual level, the data model is indep[illegible] or software constraints.



What is Data Modeling?

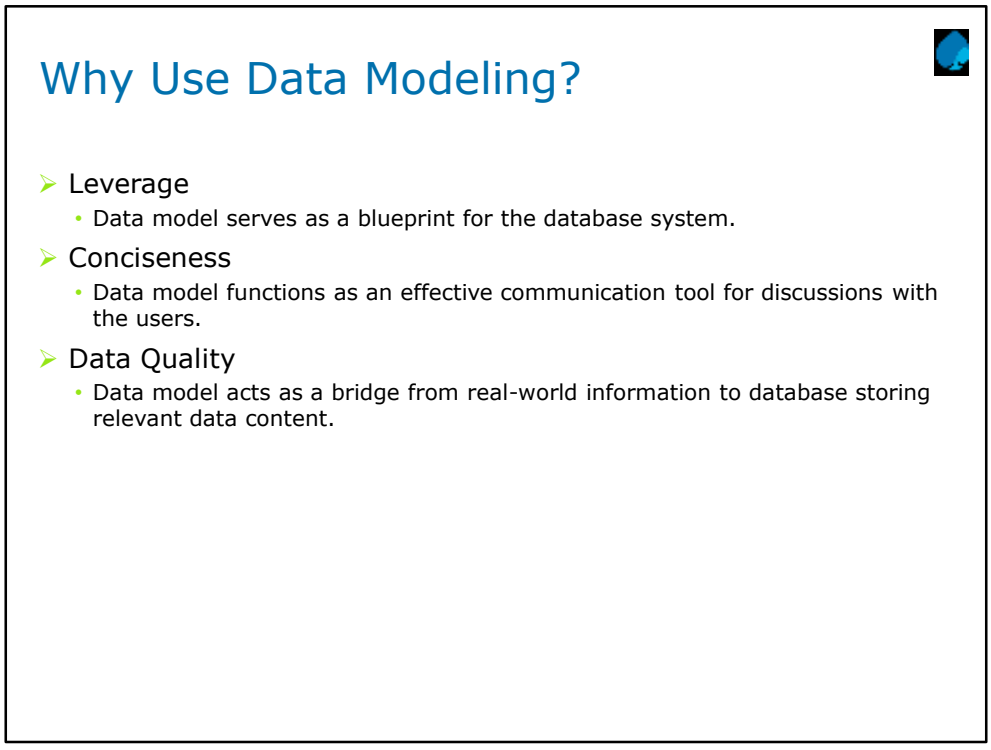
At this level, the data model is generic; it does not vary whether you want to implement an object-relational database, a relational database, a hierarchical database, or a network database.

At the next level down, a data model is a logical model relating to the particular type of database relational, hierarchical, network, and so on. This is because in each of these types, data structures are perceived differently.

If you proceed further down, a data model is a physical model relating to the particular database management system (DBMS) you may use to implement the database.

Instructor Notes:

Explain how a model will help in understanding and communicating requirements better.



Why Use Data Modeling?

Leverage: The key reason for giving special attention to data organization is the leverage. A small change to a data model may have a major impact on the whole system. Therefore, you can opt for modifying the data model instead of the system. For the most commercial information systems, the programs are far more complex. Also, considerable time is consumed in specifying and constructing them, as compared to the database. However, their contents and structures are heavily influenced by the database design.

Conciseness: A data model is a very powerful tool for establishing requirements and capabilities of information systems. Its valuable because of its conciseness. It implicitly defines a whole set of screens, reports, and processes needed to capture, update, retrieve, and delete the specified data. The data modeling process can tremendously facilitate our understanding of the essence of business requirements.

Data Quality: The data held in a database is usually a valuable business asset built up over a long period. Inaccurate data (poor data quality) reduces the value of the asset and can be expensive or impossible to correct. Frequently, problems with data quality can be traced back to a lack of consistency in (a) defining and interpreting data, and (b) implementing mechanisms to enforce the definitions.

Instructor Notes:

Data modeling is a technique of creating a conceptual schema based on the entities identified in the problem domain and finally arriving at the physical database schema

Database modeling is how the data and its relationship is organized in the database .

Data Models

- It is a specification of how the data in a database is structured and accessed .
- Common models are
  - Flat Model
  - Hierarchical
  - Network
  - Relational

Instructor Notes:

The instructor can talk about Object relation model also , if he /she is comfortable

Data Models

- Flat Model
  - Data is stored in an array of two dimensions
- Hierarchical model
  - Data and the relationships among them are represented in the form of a tree structure ,.
- Network model
  - Data and the relationships among them are represented in the form of records and links.
- Relational model
  - Data is stored in tables and the relationship among them is represented in common column called foreign key

Instructor Notes:

The Relational Model

- Relational Model – developed by Dr. E. F. Codd at IBM in the late 1960s
- He was looking for ways to solve the problems with the existing models
- Relational Model - core concept is of a table (also called a relation) in which all data is stored
- Each table is made up of
  - records (horizontal rows also known as tuples) and
  - fields (vertical columns also known as attributes)

Instructor Notes:

Make it clear that TABLE itself means RELATION

The Relational Model

- Because of lack of linkages relational model is easier to understand and implement.

Student Table

Scode	Sname
S1	A
S2	B

Course Table

Ccode	Cname
C1	Physics
C2	Chemistry
C3	Maths
C4	Biology

Marks Table

Ccode	Scode	Marks
C1	S1	65
C2	S1	78
C3	S1	83
C4	S1	85
C3	S2	83
C4	S2	85

Instructor Notes:

Possibilities in Relational Model

- INSERT
  - Inserting a course record or student record poses no problems because tables are separate
- DELETE
  - Deleting any record affects only a particular table
- UPDATE
  - Update can be done only to a particular table

Instructor Notes:

The Relational DBMS

- **Examples of Relational Tables**

DEPT table

Deptno	Dname	Loc
10	Accounting	New York
20	Research	Dallas
30	Sales	Chicago
40	Operations	Boston

ß 'row' or 'tuple'

á

'column' or 'attribute'

EMPLOYEE table

Empno	Empname	Job	Mgr	Deptno
7369	Smith	Clerk	7902	20
7499	Allen	Salesman	7839	30
7566	Jones	Manager	7839	20
7839	King	President		10
7902	Ford	Analyst	7566	20

Instructor Notes:

Properties of Relational Data Structures

- Tables must satisfy the following properties to be classified as relational.
  - Entries of attributes are single valued
  - Entries of attribute are of the same kind.
  - No two rows are identical
  - The order of attributes is unimportant
  - The order of rows is unimportant
  - Every column can be uniquely identified.

Relational tables have six properties, which must be satisfied for any table to be classified as relational. These are :

1. *Entries of attributes are single valued*

Entry in every row and column position in a table must be single valued. This means columns do not contain repeating groups

2. *Entries of attribute are of the same kind*

Entries in a column must be of same kind. A column supposed to store sal of a employee should not store comm.

3. *No two rows are identical*

Each row should be unique, this uniqueness is ensured by the values in a specific set of columns called the primary key.

4. *The order of attributes is unimportant*

There is no significance attached to order in which columns are stored in the table. A user can retrieve columns in any order.

5. *The order of rows is unimportant*

There is no significance attached to the order in which rows are stored in the table. A user can retrieve rows in any order.

6. *Every column can be uniquely identified.*

Each column is identified by its name and not its position. A column name should be unique in the table.

Instructor Notes:

What is Data Integrity?

- Data integrity is the assurance that data is consistent, correct, and accessible
- Two important steps in planning tables are to identify valid values for a column and to decide how to enforce the integrity of the data in the column
- Data integrity falls into these categories
  - Entity Integrity
  - Domain Integrity
  - Referential Integrity
  - User Defined Integrity

Instructor Notes:

Types of Data Integrity

- Entity Integrity
  - Entity integrity ensures that no records are duplicated and that no attributes that make up the primary key are NULL
  - It is one of the properties necessary to ensure the consistency of the database.
- Domain Integrity
  - Domain integrity is the validity of entries for a given column
  - Domain integrity can be enforced by restricting the type, the format, or the range of possible values.

Entity integrity defines a row as a unique entity for a particular table. Entity integrity enforces the integrity of the identifier column(s) or the primary key of a table (through indexes, UNIQUE constraints, PRIMARY KEY constraints, or IDENTITY properties).

Domain Integrity

You can enforce domain integrity by restricting the type (through data types), the format (through CHECK constraints and rules), or the range of possible values (through FOREIGN KEY constraints, CHECK constraints, DEFAULT definitions, NOT NULL definitions, and rules).

Instructor Notes:

Types of Data Integrity

- Referential Integrity
  - Referential integrity preserves the defined relationships between tables when records are entered or deleted
  - The referential integrity rule : If a foreign key in table A refers to the primary key in table B, then every value of the foreign key in table A must be null or must be available in table B
- User-defined integrity:
  - Refers to a set of rules specified by a user, which do not belong to the entity, domain, and referential integrity categories

In Microsoft SQL Server, referential integrity is based on relationships between foreign keys and primary keys or between foreign keys and unique keys (through FOREIGN KEY and CHECK constraints). Referential integrity ensures that key values are consistent across tables. Such consistency requires that there be no references to nonexistent values and that if a key value changes, all references to it change consistently throughout the database.

Instructor Notes:

How to implement Data Integrity?

- Data integrity is maintained by:
  - Applying Constraints
  - Applying Rules
  - Using User defined Types

Instructor Notes:

Different database vendors will have different flavors of SQL . T-SQL from Microsoft and Sybase Inc , SQLPLUS from Oracle etc .
Also talk about ANSI-SQL

T-SQL Languages

- SQL is a special-purpose language used to define, access, and manipulate data
- SQL is nonprocedural language, it only describes the necessary components like tables and desired results without specifying exactly how those results should be computed
- SQL comes in many flavors
- Microsoft SQL Server makes use of Transact-SQL

Instructor Notes:

T-SQL Sub Language

- The Data Definition Language (DDL)
- Data Manipulation Language (DML)
- Data Control Language (DCL)
- Transactional Control Language (TCL)

Instructor Notes:

T-SQL Sub Language - DDL

- It is the subset of SQL which contains the commands used to create and destroy databases and database objects
- DDL includes the commands for handling tasks such as creating tables, indexes, views, and constraints
- The commands are
  - CREATE
  - ALTER
  - DROP
  - TRUNCATE

Instructor Notes:

T-SQL Sub Language - DML

- It is the subset of SQL used to access and manipulate data contained within the data structures previously defined via DDL
- The Commands are:
  - INSERT
  - UPDATE
  - DELETE
  - MERGE
  - [SELECT]

Instructor Notes:

T-SQL Sub Language - DCL

- It is the subset of SQL Commands that control a database, including administering privileges and committing data
- It is used to create roles, permissions, and to control access to database.
- The Commands are
  - GRANT
  - REVOKE
  - DENY

Instructor Notes:

T-SQL Sub Language - Transactional Control Language

- It is used to manage different transactions occurring within a database.
- The commands are
  - COMMIT
  - ROLLBACK
  - SAVE TRANSACTION

Instructor Notes:

None

Summary

- In this lesson, you have learnt:
  - A set of inter-related data is known as database and the software that manages it is known as database management system or DBMS
  - Different data models are Hierarchical Model, Network Model and Relational Model
  - Data integrity is the assurance that data is consistent, correct, and accessible

Instructor Notes:

Review Question

- Question 1: A set of structures and relationships that meet a specific need is called as a _____.
- Question 2: What are the characteristics of DBMS?
- Question 3: What are the various types on Data Integrity?
- Question 4: What are the SQL Sublanguages?