Unit 9 □ Database structure, Organization and Search

Structure

9.0 Objectives

The objectives of the Unit are to :

- Explore concept of database architecture
- Enumerate different levels of database architecture
- Present an overview of components of a database system
- Explain database organization
- Discuss database search and advanced database search techniques

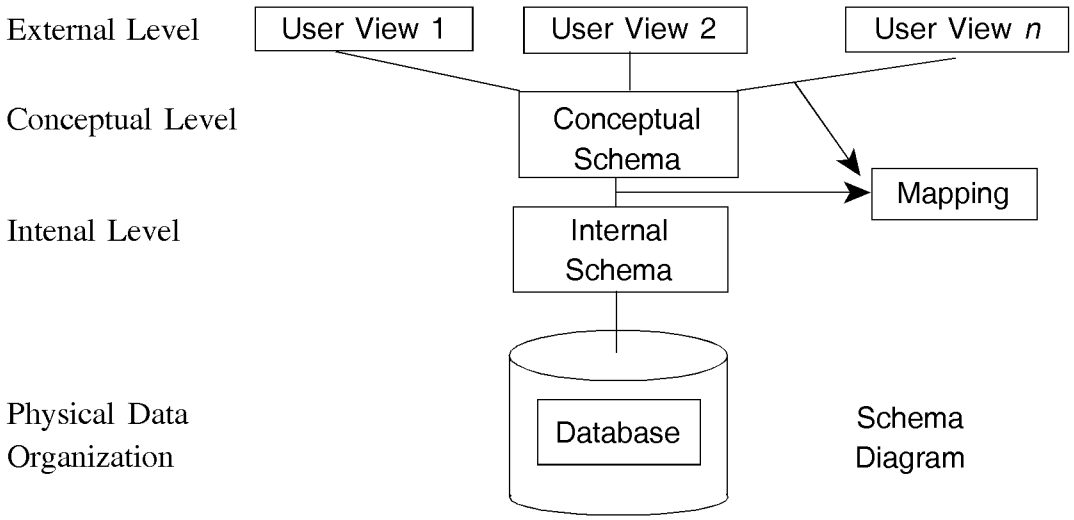
9.1 Introduction

The term database may be defined in several ways. However, the most meaningful way from the standpoint of the design of an information system is that it is a controlled collection of interrelated data. Implicit in this definition is the capability to define interrelationships among data, to minimize redundancy, to storage data in such a ways that can be retrieved to satisfy a variety of needs, and to modify the stored data as per the needs.

9.2 Database Architecture

It is necessary to understand certain key terms to appreciate the database architecture. The selected terms are :

- Database Schema (intension)
  - refer to a description of database
  - specify during database design
  - should not be changed
- schema diagrams
  - convention to display some aspect of a schema visually
- Schema construct
  - refers to each object in the schema (e.g. STUDENT)
- Database state (snapshot or extension)
  - refers to the actual data in the database at a specific time
  - changes any time we add or delete a record
  - vaild state : the state that satisfies the structure and constraints specified in the schema and is enforced by DBMS
- To define a new database, we specify its database schema to the DBMS (database is empty)
- Database is initialized when we first load it with data

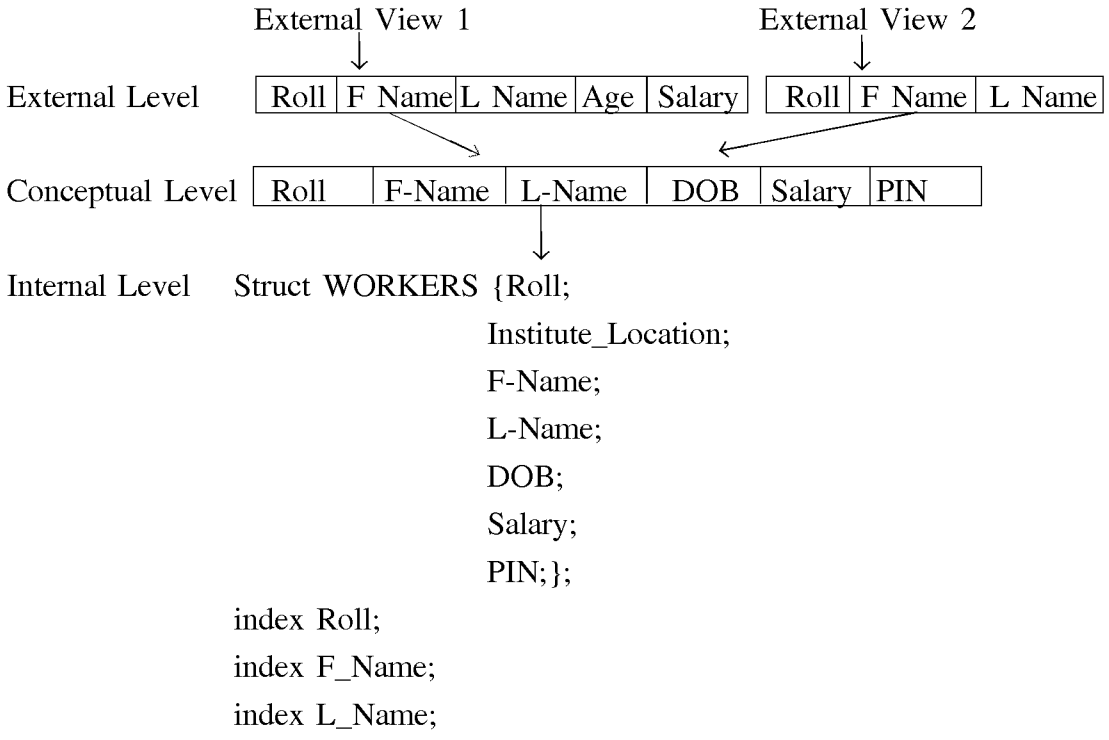


The salient objectives of the database systems are :

- Insulation of data from programme
- Support of multiple user views
- Use of catalogue to store the database schema

To ensure these basic characteristics of the database, the architecture of a database can be grouped into three separate levels of descriptions :

- **Conceptual :** It is the logical description of the database. This overall description is known as a Schema. Describes the structure of the entire database, hides the details of physical storage structures. Concentrates on the describing entities, data types, relationships, operations, and constratints. High-level data models or an implementation data model may be used here. It provides concepts closer to the way many users
  - Perceive data
  - Uses concepts such as entities, attributes, and relationships
    - Entity : a real world object or concepts
      (e.g., Project)
    - Attribute : properties that describes objects
      (e.g., Project_Name)
    - Relationships : an interaction or links among entities
      (e.g., works-on.)
- **External :** Subsets of the schema that contain only the data needed for particular applications provide user views. The sub schemas provide a description at the external level. Includes a number of external schema or user views, each view describes subset of database needed by a particular user. High Level data model or an implementation data model can be used here.
- **Internal :** The description of the physical storage structures on a specific computer system is the internal description. It is concerned with the actual storage representation of data and its relationships. It describes structure of the database (data storage and access path)



Three Level Database architecture

This three level description is a convenient framework. Three-schema is used to implement data independence. This allows changes to be made at one level with minimum impact on the other levels. It ensures different types of data independence, they are :

- Logical data
  - Refers to the immunity of the external schemas to changes in the conceptual schema (e.g., add new record or field)
- Physical data
  - Refers to the immunity of the conceptual schema to changes in the internal schema (e.g., add new access path should not void existing ones)

9.2.1 Components of Database Management System

The basic components of database management systems are :

- Data Description Language (DDL) to translate the schema written in a source language into object schema, thereby creating a logical and physical layout for the database.
- Procedures for adding/deleting/modifying/retrieving etc. Data

- Query Language Translator (QLT) to translate queries written in a source language to a sequence of procedure calls for retrieving data from the database.
- System Software for file and buffer management, report generation, controlling concurrency, recovery from failure and to enforce security.
    - DBMS Interfaces :
        - Menu-based interfaces
        - Graphical Interfaces
        - Forms-based interfaces
        - Natural Languages
        - Interfaces for parametric users
        - Interfaces for DBA
    - Database System Environment/Tools
        - DBMS component Modules : E.g., Data Mgr, DDL Compiler, Query Compiler, etc
        - DBMS utilities : E.g., Backup
        - DBMS Communication Facilities : E.g., LAN
        - Case Tool : Rational Rose
        - Application development Environments : E.g., JBuilder

9.3 Database organization

Databases generally have one of two basic forms-the single-file database or the multi-file relational database. Single-file databases are often called "flat file" systems and relational databases are frequently known as "structured" databases. The type of database system or tools that require for an application depends on a number of factors, such as :

- The complexity of the data involved eg plain text, images, sound files
- The quantity of data to be stored and processed
- Whether the data needs to be accessed and amended by more than one person
- Whether data needs to be imported from or exported to the IT Systems

9.3.1 Types of Database Organization

The four major types of the database organization are :

- Flat
- Hierarchical
- Relational
- Object-oriented

9.3.1.1 Flat Database

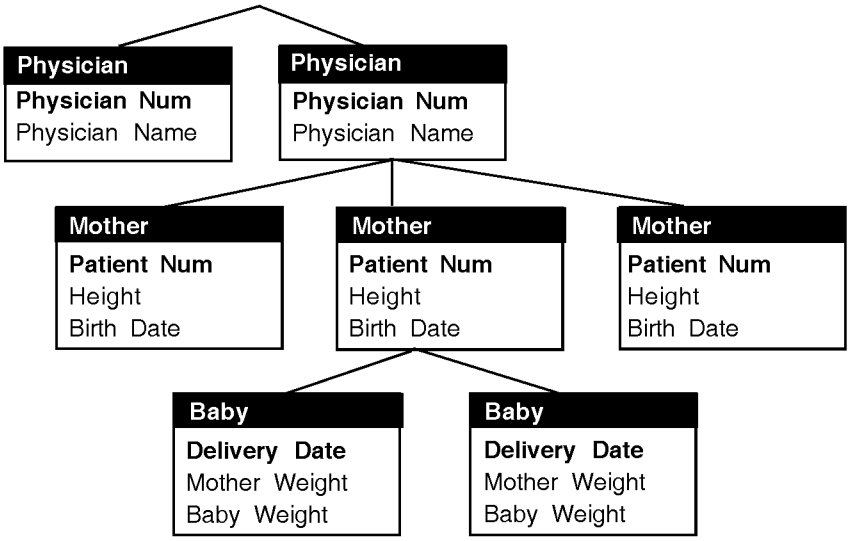
This database contains a single kind of record with a fixed number of fields. Notice the repetition of data, and thus an increased chance of errors.

Record No	Title	Publisher	Country	Phone
1	Serials Management	Elsevier	USA	789-896
2	Library Trends	Elsevier	USA	789-896
3	Granthagar	BLA	India	2442-6896

9.3.1.2 Hierarchical Database

This database depends on hierarchical relationships among different types of data. Points to be noted are :

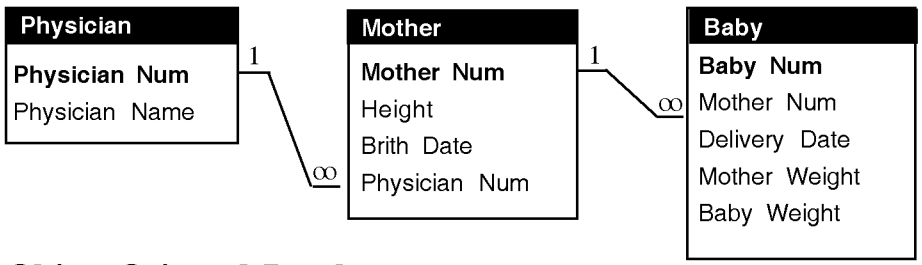
- Can be very easy to answer some questions, but very difficult to answer others
- If one-to-many relationship is violated (e.g., a patient can have more than one physician) then the hierarchy becomes a network



9.3.1.3 Relational Database

Data are organized as logically independent tables. Features of a relational database are :

- Natural
- Not so strongly biased towards specific questions
- Expresses relationships by means of shared data rather than explicit pointers
- Theoretical basis : relational algebra, calculus;
- Operations on tables (Join, Project, Select) to form new tables



9.3.11.4 Object-Oriented Database

Object-oriented analysis is another way to model the world, involving abstraction, encapsulation, modularity and hierarchy (with inheritance). An object consists of data and methods. Classes are used to group objects which have the same types of data and the same methods.

- Abstraction : Consider only features which are necessary for the problem at hand. For example, a person may be defined by the ability to ask for the ID number, age, height and weight
- Encapsulation : The internal structure of an object is hidden. For example, all we know is that we can ask for the age, not whether the age is (1) periodically updated or (2) calculated on demand from birth date and current date.
- Modularity : Grouping classes and objects into 'cohesive and loosely coupled modules'. For example, the classes person and cat might be implemented within the same module because they share many things (like how to calculate age).
- Hierarchy : Objects are instances of classes, e.g., John is an instance of the class patient. Classes and objects form two different hierarchies.
  - Class hierarchy ('kind') : e.g., patient is a kind of person. A class inherits (and specializes) the characteristics of its parent class.
  - Object hierarchy ('part') : e.g., the object John. height is a part of the object John.

9.3.2 Strategies for Database Organization

Strategy	Reliability	Expandability	Communication	Manageabilities	Data Integrity
Centralized database resides in one location on host; data values may be distributed to geographically dispersed users for local processing.	Poor	Poor	Very High	Very Good	Excellent
Replicated					
Distributed snapshot databases Copy of portion of the central database created by extraction and replication for use at remote sites.	Good	Very Good	Low to Medium	Very Good	Medium
Replicated, distributed database. Data are replicated and synchronized at multiple sites.	Excellent	Very Good	Medium	Medium	Medium to very Good
Partitioned					
Distributed nonintegrated databases Independent databases that can be accessed by applications on remote computers.	Good				
Distributed, integrated database. Data span multiple computers and software.	Very Good	Very Good	Low to medium	Difficult to very Difficult	Very poor

9.4 Database Design

Database design is the process of developing a database structure from user requirements. Database design is a complex process since database is a shared system and it has to meet the requirements of different users within the accepted data integrity, data security and data privacy. Actual design depends on the particular DBMS being used. However, there are certain steps which are independent of any software.

9.4.1 Design Process

One of the best ways to understand database design is to start with an all-in-one, flat-file table design and add some sample data to see what happens. Identify problems caused by the initial design by analyzing the sample data, Modify the design to eliminate the problems, test some more sample data. check for problems, and re-modify, continuing this process until a consistent and problem free design developed. The major steps involved in developing a computer based application system may be summarized as follows :

9.4.1.1 Analysis Phase

This phase is software independent. The goal is to formulate precise system requirement definitions through the following activities :

- External system activities
  - Define information needs
  - Determine required inputs
  - Define data relationships
- Interface system activities
  - Define computer outputs
  - Define enquiry requirements
  - Idenfine input sources
  - Define data entry procedures
- Internal system activities
  - Specify data validation requirements
  - Define data protection requirements
  - Specify data validation rules

9.4.1.2 Database Design Phase

The requirements identified in the Analysis Phase have to be oriented according to the potential of the software. This phase involves :

- Defining data items
- Defining data entry form/worksheet
- Defining indexing techniques
- Defining display and output formats

9.4.1.3 Database creation and Maintenance Phase

This phase allows the creation and modification of records in the currently selected database. The various maintenance activities can also be performed.

The phase consists of :

- Data entry/editing
- Creation of index files
- Taking backup of the database

9.4.1.4 Functional Phase

The database application must support searching/information retrieval and production of different outputs on suitable media.

9.5 Database Search

The method used to store, find and retrieve the data from a database is called access method/information retrieval. A query language is used for retrieval of data from a database. The user specifies the requirements in terms of database query language syntax. The main emphasis is on the data is required and not on how it is located. Each DBMS contains procedures that allow a computer programme to access the data from the system. Certain general issues need to be considered at this level, they are :

- Access Control
- Semantics of data
- Variety of responses

9.5.1 Semantics of Data

Familiarity with the structure of records and fields will help to pinpoint specific information quickly. In certain cases, these details may not be avilable directly from the database schema, and if they are avilable the data definitions are not sufficient to understand meaning of the data items. To avoid this difficulty a data dictionary is usually required to support a database query language, and provide details of the data to be retrieved.

9.5.2 Variety of Responses

Ensuring search capabilities is not enough. Search system must take into account the variety of possible responses to a query. User expects some data to be retrieved against his/her query. However, under certain conditions, no data may be retrieved. Appropriate messages must be displayed under "no data" situations. This is certainly different from messages for different user and/or system error during execution of a query.

9.5.3 Search Techniques

There are various types of information database : bibliographic, numeric, textual and graphical. To be useful, databases have to be structured in some fairly definite ways. Such structure provides organization for massive quantities of data and allows rapid access to specific pieces of information. Each record, in turn, is made up of fields, which are specific categories of data about the item-the title of a book, the author of an articale, the publisher (imprint), the publication date. Familiarity with the structure of records and fields will help to pinpoint specific information quickly. The following table presents an overview of certain advanced search techniques :

Type of Search	purpose	When to Use	Example
Boolean	Specifies multiple words in any field, in any order	You will want to use Boolean searches most of the time you are doing any keyword searches -unless your search term is very new and/or unusual, you will retrieve too much information	Tagore AND Rain (narrows search so that records must have both terms) Tagore OR Rain (Broadens search, records can have either Tagore or Rain)
Truncation	A truncation symbol tells the database to find any words that begin with the letters user typed, rgardless of how the words end	Use truncation when you have a keyword term that has many similar forms which may bring up additional information related to your search topic	Truncation symbols vary depending which database you are using. Griffin & Firstsearch: medic* ProQuest: medic? If you are not sure what truncation symbol a database uses, look for the "search tips" or "help" link.
Proximity	Proximity operators allow you to locate one word within a certain distance of another.	use proximity operators when you are searching for keywords, which should appear very close together in a database record to adequately address your topic.	Proximity operators vary depending which database you are using. Griffin: america within 3 econom* FirstSearch: america w3 econom*

			ProQuest: america w/3 econom? If you are not sure what proximity operators a database uses, look for the "search tips" or "help" link.
Limiting / Narrowing	Nearly all databases will allow you to limit your search in some way. This is very helpful in weeding out resources are not useful to you for some reason.	You will generally want to put some limiting criteria on most of your searches just to save yourelf the time of looking at less than-helpful resources.	Limiting options vary by database. Look for pull-down menus, check-boxes, etc. while you are searching

References and Further Reading List

1 2005 Data structure (http://en.wikipedia.org/wiki/Data structure). Sept 2005. Last visited : 23/10/2005

2 2005 Chapple (Mike). Database normalization basics. (http://databases. about.com/od/sepecificproducts/a/normalization. htm). visited last : 23/10/2005

3 2000 Database models. (http;//www.frickcpa.com/ss7/Theory Models. asp#File). 2000. Visited last : 23/10/2005

4 1996 Majumder (Arun K) and Bhattacharyya (Pritimoy). Database management systems.. New Delhi : Tata McGraw-Hill, 1996.

5 1994 Elmasri (Ramez) and Navathe (Shamkant B). Fundamentals of database systems. California : Benjamin/Cummings, 1994.

6 1985 Date (CJ). An Introduction to database systems. 3rd ed. New Delhi : Narosa, 1985.

7 1983 Date (CJ). Database : a primer. Reading : Addison-Wesley, 1983.

8 1985 Caryln (Shamlin). A user's guide for defining software requirements. Wellesley : QED, 1985.

9.6 Exercise

1. What is database architecture? Define different important terms associated with the concept of database architecture.
2. Describe components or database management systems.
3. Discuss different types of database organization.