DBMS college notes (1)

Topic → Basic, classifications of DBMS, DBMS Architecture

---

1) Basic

- **Data** → known fact that can be recorded & have an implicit meaning.
- **Database** → A collection of related data representing some aspects of real world.
- **Miniworld** → Some part of real world about which data is stored in database.
- **Database Management System** → General purpose software system that facilitates the purpose of defining, constructing & manipulating databases for various applications
  Eg: MySQL, Oracle

1. Defining database – specifying data types, structures & constraints for data
2. Constructing database or Load database – Process of storing data on some medium that is controlled by DBMS.
3. Manipulating a database – functions like querying, updating, insertions, deletions

3) Example of Database

Mini world for example = University Environment

Some mini-world entities:

1) Students
2) Courses
3) Sections (of courses)
4) (academic) Departements
5) INSTRUCTORS

above could be expressed in Entity-Relationship data model

Some mini-world Relationships :-

- SECTIONS are of specific COURSES
- STUDENTs take SECTIONs
- COURSES have prerequisite COURSES
- INSTRUCTORS teach SECTIONs
- STUDENTS major in DEPARTMENTS

it can be represented as ER model

Student	Name	Students no.	class	Major
	Smith	17	1	CS
	Brown	8	2	CS

Grade	Students No.	Section Identifier	Grade
	17	112	B
	17	119	C
	8	85	A

Courses	Course Name	Course No.	Credit Hours	Department
	Intro to Programming	CS1310	4	CS
	Data Structures	CS3320	4	CS
	Discrete Maths	MA2410	3	Maths
	Database	CS3380	3	CS

Prerequisite	Course Number	Prequisite
	CS3380	CS3320
	CS3380	MA2410
	CS3320	CS1310

Section	Section Identifier	Course No.	Semester	Year	Instructor
	85	MA2410	Fall	98	King
	92	CS1310	Fall	98	Andrew
	102	CS3320	Spring	99	Knuth
	112	MA2410	Fall	99	Chang
	119	CS1310	Fall	99	Stone

③ Relations

Relation_name	No_of_colums
STUDENT	4
COURSE	4
SECTION	5
GRADE	3
PREREQUISITE	2

Main characteristics of Database approach

1) Insulation betn program & data – Called program data independence. Allows changing data storage structures & operations without having to change DBMS access programs.

2) Data Abstraction – A data model is used to hide storage details & present the user with conceptual view of database.

3) Support of multiple views of data :- Each user may see a different view of database, which describe only data of interest of that user.

Database Users

Users are of 2 type

1. Those who actually use & control the content (called actor on the scene) eg: Database administrators, Database designers, End-users.
2. Those who enable database to be developed & DBMS software to be designed & implemented (called "Actors behind the scene") eg: System designers & Implementors, Tool Developers, operators & Maintenance Personnel.

DBMS ARCHITECTURE

- Data model → A set of concepts to describe structure of a database & certain constraints a database should poses.

Categories of data models :-

- Conceptual (high-level, semantic) data models :
  Provide concepts that are close to the way many users perceive data (Also called entity-based or object-based data models.)
- Physical (low-level, internal) data models :
  Provide concepts that describe details of how data is stored in computer.
- Implementation (representational) data models :
  Provide concepts that fall between the above two, balancing user views with some computer storage details.

Schemas vs instances :-

- Database Schema :- The description of a database. Includes description of data base structure & constraints that should hold on database.
- Schema Diagram :- A diagrammatic display of (some aspects of) database schema.
- Schema construct :- A component of schema or an object within the schema, eg :- STUDENT, COURSE
- Database instance :- The actual data stored in data at particular moment in time. Also called database state (or occurance)

⑤ Schema diagram

Schema is called 'intension', whereas state is called extension.

eg of Database State

COURSE

Name	Course No.	Credit	Department
Data Structure	CS 3320	4	CS
Discrete Maths	MA 2410	3	MATH

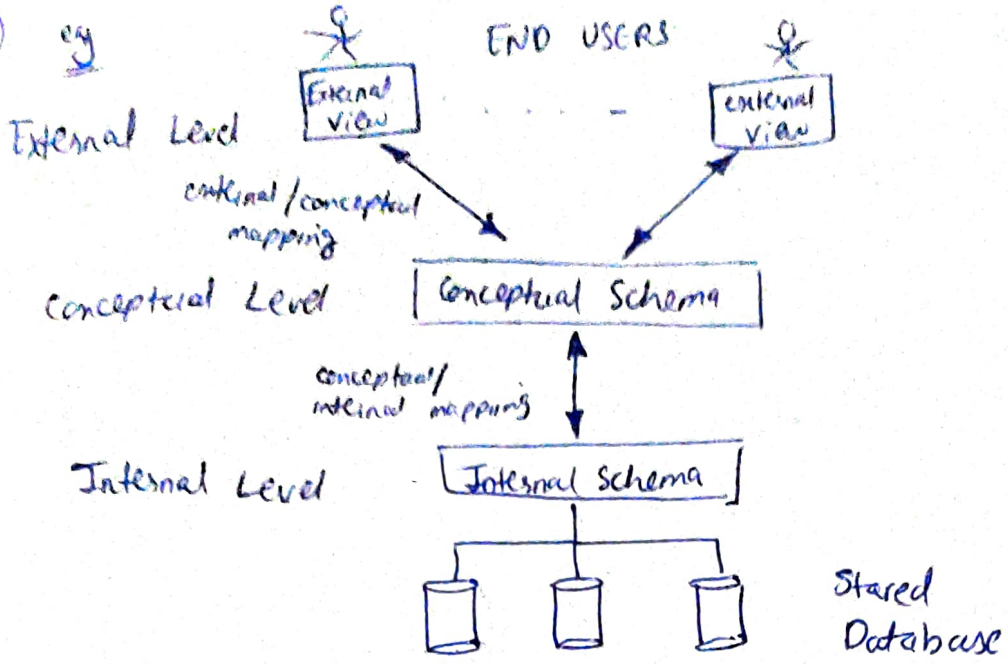
- **Three Schema Architecture**

Defines DBMS schemas at three levels.

1) Internal Schema :- at internal level to describe physical storage structures & access paths. Typically uses a physical data model.
2) Conceptual schema :- at conceptual level to describe the structure & constraints for whole database for a community of users. Uses a conceptual or an implementation data model.
3) External schemas :- at external level to describe various user views. Usually uses the same data model as conceptual level

→ Mappings among schema levels are needed to transform request & data. Program refer to an external schema, and are mapped by DBMS to internal schema for execution

⑥ eg



Data Independence

1) Logical Data Independence :- The capacity to change conceptual schema without having to change the external schemas and & their application programs.

2) Physical Data Independence :- The capacity to change internal schema without having to change conceptual schema

DBMS LANGUAGES

1) Data Definition Language (DDL): Used by DBA & database designers to specify conceptual schema of a database. In many DBMS, the DDL is also used to define internal & external schemas (views).

2) Data Manipulation Language (DML): Used to specify database retrievals & updates

- DML commands (data sublanguage) can be embedded in general-purpose programming language (host language) such as C or an Assembly language
- Alternatively, stand-alone DML commands can be applied directly (query Language)

⑦ Classification of DBMS

- Based on data model used :
  - Relational , Network , hierarchical
  - Object - oriented , Object - relational
- Other classifications :
  - Single - user (typically used with micro - computers) vs multi user .
  - Centralized (uses single computer with one database) vs distributed (uses multiple computers , multiple databases)