# Introduction to SET08104

SET08104 Database Systems

# Introduction

Before Databases:

- Each application suite had independent master files.
  - Duplication of data could lead to inconsistencies
  - Common master files had integrity and security problems.
- Data structuring techniques to exploit random access disks made data manipulation techniques complicated.
  - Subroutines (a step towards DBMS) gave general routines to manipulate data.
  - To use subroutines required low-level data knowledge.

# Database Approach

- Information in the database is subdivided into two concepts:
  - Schema
  - Data
- Schema is the concept of how information relates to other pieces of information, and how information should be grouped.
- Data is the concept of the actual information users want to store in the database. You can only store data in structures which the schema provides, so we must ensure that the schema is correct.

# A Table…

- The main schema “container” concept is called a table.

| Name | Address | Date of Birth | Salary |
|---|---|---|---|
| Jim Smith | 1 Apple Lane | 1/3/1991 | 11000 |
| Jon Greg | 5 Pear St | 7/9/1992 | 13000 |
| Bob Roberts | 2 Plumb Road | 3/2/1990 | 12000 |

- Columns: facts about each object in the database
- Rows: different objects of the same classification
- Each column is of a particular type.
  - NAME – string holding 12 characters
  - ADDRESS – string holding 12 characters
  - Date of Birth – a date meaning an age >18 & <100
  - SALARY – a number greater than 0
- These types of rules and more can all be built into the database by adding rules to the schema.

# Users

There are three broad classes of user:

1. the application programmer, responsible for writing programs in some high-level language such as COBOL, C++, etc.
2. the end-user, who accesses the database via a query language
3. the database administrator (DBA), who controls all operations on the database

# Database Architecture

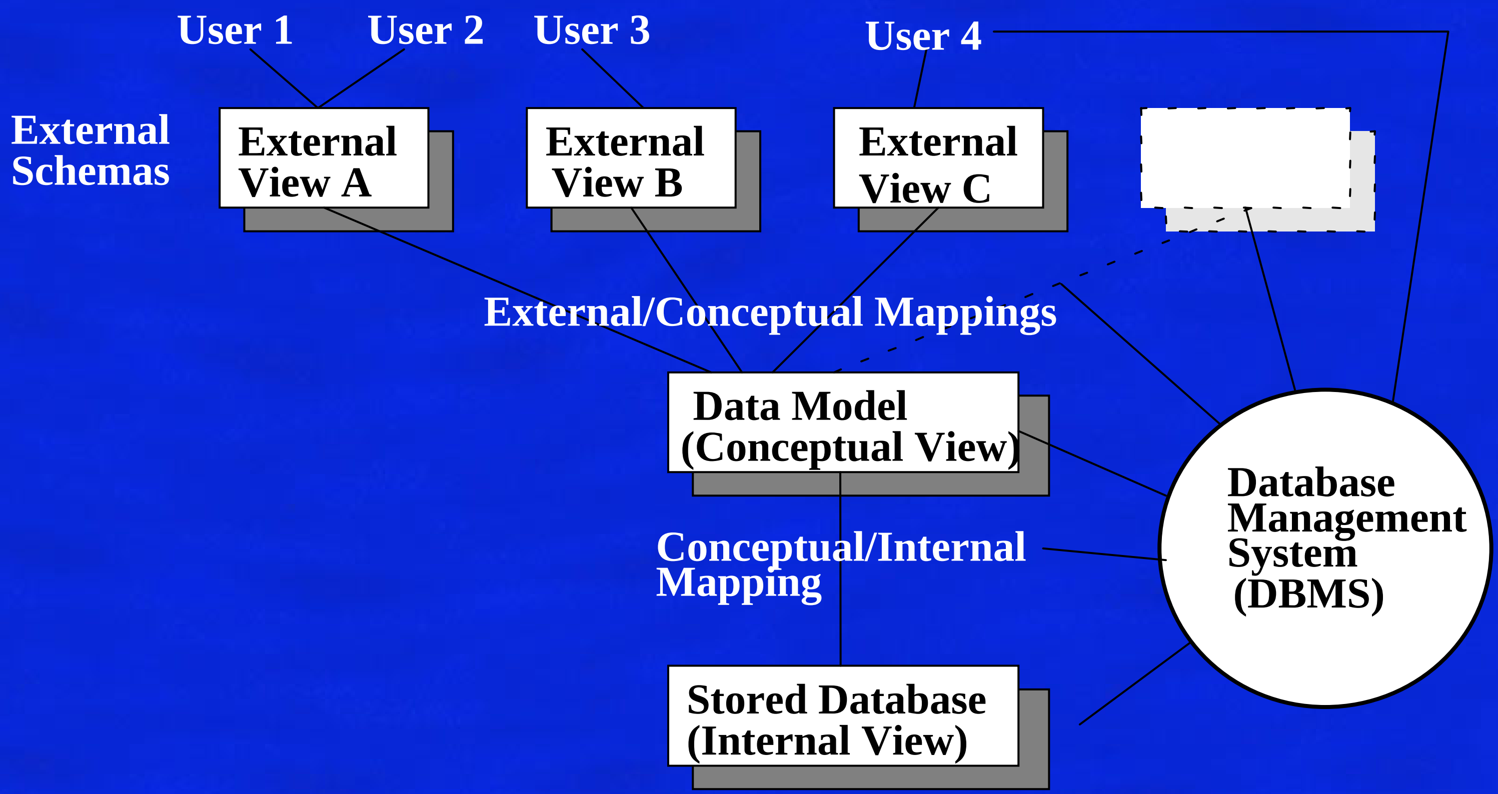
DBMSs do not all confirm to the same architecture.

- The three-level architecture forms the basis of modern database architectures.
  - agreement with the ANSI/SPARC study group on Database Management Systems.
- The architecture for DBMSs is divided into three general levels:
  1. external
  2. conceptual
  3. internal

# Architecture cont...

1. the external level : concerned with the way individual users see the data
2. the conceptual level : can be regarded as a community user view - a formal description of data of interest to the organisation, independent of any storage considerations.
3. the internal level : concerned with the way in which the data is actually stored

# Architecture cont…



User 1
User 2
User 3
User 4
External Schemas
External View A
External View B
External View C
External/Conceptual Mappings
Data Model (Conceptual View)
Conceptual/Internal Mapping
Stored Database (Internal View)
Database Management System (DBMS)

# External View

- A user is anyone who needs to access some portion of the data.
  - Access via a 3GL,COBOL, etc query language
  - All access methods include a data sub-language (DSL).
- A DSL is a combination of two languages:
  - a data definition language (DDL)
  - a data manipulation language (DML)
- Each user sees the data in terms of an external view
  - Defined by an external schema, consists of external record descriptions, and understands the mapping between external schema and the conceptual level.

# Conceptual View

- An abstract representation of the entire information content of the database.
- It is in general a view of the data as it actually is.
- It consists of multiple occurrences of multiple types of conceptual record
- To achieve data independence, the definitions of conceptual records must involve information content only.
- The conceptual schema, as well as definitions, contains authorisation and validation procedures.

# Internal View

- This is a very low-level representation of the entire database
- It is at one remove from the physical level
- The internal view is described by the internal schema:
  - defines the various types of stored record
  - what indices exist
  - how stored fields are represented
  - what physical sequence the stored records are in
- In effect, the internal schema is the storage definition structure.

# Mappings

- The conceptual/internal mapping:
  - defines conceptual and internal view correspondence
  - specifies mapping from conceptual records to their stored counterparts
- An external/conceptual mapping:
  - defines a particular external and conceptual view correspondence

- A change to the storage structure definition means that the conceptual/internal mapping must be changed accordingly, so that the conceptual schema may remain invariant, achieving physical data independence.
- A change to the conceptual definition means that the conceptual/external mapping must be changed accordingly, so that the external schema may remain invariant, achieving logical data independence.

# DBMS

The database management system (DBMS) is the software that:

- handles all access to the database
- is responsible for applying the authorisation checks and validation procedures

# Conceptually…

Conceptually what happens is:

1. A user issues an access request, using some particular DML.
2. The DBMS intercepts the request and interprets it.
3. The DBMS inspects in turn the external schema, the external/conceptual mapping, the conceptual schema, the conceptual internal mapping, and the storage structure definition.
4. The DBMS performs the necessary operations on the stored database.

# Database Administrator

The database administrator (DBA) is responsible for overall control of the database system. Responsibilities include:

- deciding the information content of the database, i.e. identifying the entities of interest to the enterprise and the information to be recorded about those entities. This is defined by writing the conceptual schema using the DDL
- deciding the storage structure and access strategy, i.e. how the data is to be represented by writing the storage structure definition. The associated internal/conceptual schema must also be specified using the DDL

# DBA cont...

- liaising with users, i.e. to ensure that the data they require is available and to write the necessary external schemas and conceptual/external mapping (again using DDL)
- defining authorisation checks and validation procedures. Authorisation checks and validation procedures are extensions to the conceptual schema and can be specified using the DDL

# DBA cont...

- defining a strategy for backup and recovery. For example periodic dumping of the database to a backup tape and procedures for reloading the database for backup. Use of a log file where each log record contains the values for database items before and after a change and can be used for recovery purposes
- monitoring performance and responding to changes in requirements, i.e. changing details of storage and access thereby organising the system so as to get the performance that is ‘best for the enterprise’

# Facilities and Limitations

Facilities offered by DBMSs vary. All DBMSs should provide the following advantages over conventional systems:

- independence of data and program
- data shareability and non-redundancy of data
- integrity
- centralised control
- security
- performance and Efficiency

# Data Independence

- This is a prime advantage of a database
- In conventional systems applications are data-dependent
- For example, if a file is stored in indexed sequential form then an application must know
  - that the index exists
  - the file sequence (as defined by the index), and

The internal structure of the application will be built around this knowledge. If, for example, the file was to be replaced by a hash-addressed file major modifications would have to be made to the application.

# Data Independence cont...

- Such an application is data-dependent
- it is undesirable to allow applications to be data-dependent
- the DBA must have the freedom to change storage structure or access strategy in response to changing requirements without having to modify existing applications.
- Data independence can be defines as ‘The immunity of applications to change in storage structure and access strategy’.

# Data Redundancy

- In non database systems each application has its own private files
  - This can often lead to redundancy in stored data, with resultant waste in storage space.
- in a database the data is integrated
  - the database may be thought of as a unification of several otherwise distinct data files, with any redundancy among those files partially or wholly eliminated.
- Data integration is regarded as an important characteristic
  - The avoidance of redundancy should be an aim, however, it is not a requirement.

# Redundancy cont...

Redundancy is

- direct if a value is a copy of another
- indirect if the value can be derived from other values:
  - simplifies retrieval but complicates update
  - conversely integration makes retrieval slow and updates easier

# Redundancy cont...

- Data redundancy can lead to inconsistency in the database unless controlled.
  - the system should be aware of any data duplication
  - a DB with uncontrolled redundancy can be in an inconsistent state
  - a fact represented by only one entry cannot result in inconsistency.

# Data Integrity

This describes the problem of ensuring that the data in the database is accurate...

- inconsistencies between two entries representing the same ‘fact’ give an example of lack of integrity (caused by redundancy in the database).
- integrity constraints can be viewed as a set of assertions to be obeyed when updating a DB to preserve an error-free state.
- even if redundancy is eliminated, the DB may still contain incorrect data.
- integrity checks which are important are checks on data items and record types.

# Integrity cont...

Integrity checks on data items can be divided into 4 groups:

1. type checks
2. redundancy checks
3. range checks
4. comparison checks

# Integrity cont...

- A record type may have constraints on the total number of occurrences, or on the insertions and deletions of records.
  - for example in a patient database there may be a limit on the number of Xray results for each patient
  - or the details of a patients visit to hospital must be kept for a minimum of 5 years before it can be deleted

# Integrity cont...

- Centralized control of the database helps maintain integrity
  - permits the DBA to define validation procedures to be carried out whenever any update operation is attempted (update covers modification, creation and deletion).
- Integrity is important in a database system
  - an application run without validation procedures can produce erroneous data which can then affect other applications using that data.