Database Management Systems
MIT 22033

Introduction
By S. Sabraz Nawaz

Microsoft Certified Professional
Recommended Reading…


❖ Murach’s SQL Server 2008 for Developers

❖ Any book that is on SQL Server development as well as any book that is on database
Introduction

• What is a database?
A database is a collection of related data (examples??)

• A database management system (DBMS) is a software package designed to create and maintain databases (examples?)
Introduction (contd...) 

- The DBMS is a general-purpose software system facilitates the processes of 
  - Defining  
    - Specifying the data types, structures, and constraints for the data to be stored in the database. 
  - Constructing  
    - The process of storing the data itself on some storage medium that is controlled by the DBMS. 
  - Manipulating  
    - Manipulating a database includes such functions as querying the database to retrieve specific data, updating the database to reflect changes, and generating reports from the data.
Examples

Today, DBMSs are an essential part of any Enterprise Information System

- University Database
- Credit Card Processing Systems
- Airline Reservation Systems
- Banking System
Example: University DB

- University database maintaining information concerning:
  - Students,
  - Courses, and
  - Grades
  - …
Example: University DB (contd…)

- It shows the database structure and a few sample data.
- It is organized as five files.
  - The STUDENT file – data on each student,
  - The COURSE file – data on each course,
  - The SECTION file – data on each section of a course,
  - The GRADE_REPORT file – data on grades that students receive in the various sections they have completed, and
  - The PREREQUISITE file – data on prerequisite of each course.
Need for a DBMS

- Traditional File System provided by the Operating System is insufficient to meet the requirements of enterprise applications scenario:

  A company has a large collection (500 GB) of data on database. This data is accessed concurrently by several employees. Questions about the data must be answered quickly, changes made to the data by different users must be applied consistently, and access to certain parts of the data must be restricted.
Advantages of using a DBMS

- Data independence
- Efficient data access
- Data integrity and security
- Easy data administration
- Concurrent access, recovery from crashes
- Reduced application development time
- ...
DBMS is described at three levels of abstraction:

- **External Schema**
  many views describe how users see the data

- **Conceptual Schema**
  defines logical structure

- **Physical Schema**
  describes the files and indexes used

Note:
Information about the schemas is stored in the system catalog
Conceptual schema

- describes the stored data in terms of the data model of the DBMS

- in a relational DBMS, the conceptual schema describes all relations that are stored in the database
Levels of Abstraction in a DBMS (contd)

- **Physical schema**
  - describe storage details
  - summarizes how the relations described in the conceptual schema are actually stored on secondary storage devices such as disks and tapes
  - decide what file organizations used to store the relations
  - create indexes to speed up data retrieval operations
Levels of Abstraction in a DBMS (contd)

- **External schemas**
  - allow data access to be customized (and authorized) at the level of individual users or groups of users
  - any given database has exactly one conceptual schema and one physical schema because it has just one set of stored relations, but it may have several external schemas
Data independence

- Application programs are independent of data representation and storage details.
- The structure of data files is stored in the DBMS catalog separately from the access programs.
  
  E.g. a file access program may be written in such a way that it can access only STUDENT records of the structure.
Data independence (cont)

- To add another piece of data to each STUDENT record, say Birthdate,
  - need to change the description of STUDENT records in the catalog to reflect the inclusion of the new data item Birthdate; no program changed.
  - The next time a DBMS program refers to the catalog, the new structure of STUDENT records will be accessed and used.
Efficient data access

- DBMS utilises sophisticated techniques to store and retrieve data efficiently, including support for very large files, index structures and query optimisation.

- Storage methods can be improved without changing the application programs.
Data integrity and security

❖ DBMS can enforce integrity constraints on the data
e.g., checking salary increase against dept budget

❖ Access controls govern what data is visible to different class of users
Easy data administration

- Centralising administration of data shared among many users
  - data managed by professionals

- Organise data to
  - meet user needs
  - minimise redundancy

- Fine tuning of storage for efficient retrieval
Concurrent access and Crash recovery

- Concurrent accesses are scheduled by DBMS
  - users can think of the data as being accessed by one user at a time

- DBMS protects users from the effects of system failure
Reduced application development time

- DBMS supports many functions common to applications that access the database
- These applications are likely to be more robust than applications developed from scratch because many important tasks are handled by DBMS instead of being implemented by the application
When NOT to use a DBMS

- High initial investment (DBMS is an expensive software package)
- Applications use small amounts of data
- Lack of resources (disk space, memory, etc.) to support a database
- Single-user applications
- Overhead for flexible querying, security, concurrent access & crash recovery is not required
A **data model** is a collection of high-level data description constructs used to model the application domain.

- Data model hides the low-level storage details.
- Most commercial database systems are based on the **relational data model**.
However, it is easier to use a **semantic data model** to model an application domain. A well-known semantic data model is the Entity Relationship (ER) Model.
In relational data model, the main construct is a relation. A relation has fields that belong to it which contain the name & data type of each field.

A description of data in terms of a data model is called the schema.

Every relation has a schema, which describes the name of the relation, name of each attribute (field or column), and the type of each column.

e.g.

Students(sid: string, name: string, login: string, age: integer, gpa: real)
An example instance of the Students relation:

<table>
<thead>
<tr>
<th>SID</th>
<th>Name</th>
<th>Login</th>
<th>Age</th>
<th>GPA</th>
</tr>
</thead>
<tbody>
<tr>
<td>53666</td>
<td>SaNa</td>
<td><a href="mailto:sana@email.com">sana@email.com</a></td>
<td>33</td>
<td>3.9</td>
</tr>
<tr>
<td>53668</td>
<td>Kumar</td>
<td><a href="mailto:kumar@fmail.com">kumar@fmail.com</a></td>
<td>29</td>
<td>3.2</td>
</tr>
<tr>
<td>53670</td>
<td>Sanath</td>
<td><a href="mailto:sanath@matara.com">sanath@matara.com</a></td>
<td>41</td>
<td>3.8</td>
</tr>
</tbody>
</table>
In addition to relational data model…

- Hierarchical model
- Network model
- Object oriented model
- Object relational model
With reference to a fictitious university database
1. What is the name of the student with student ID 123456?
2. What is the average salary of lecturers who teach the course with course ID CS564?
3. How many students are enrolled in course CS564?
4. Is there any student with a GPA less than 3.0 enrolled in course CS564?

These questions involving the data stored in a DBMS are called queries
Queries In A DBMS...

- A DBMS provides a specialized language, called the **query language** in which queries can be posed.
- **Relational calculus** is a formal query language based on mathematical logic.
- **Relational algebra** is another formal query language, based on a collection of **operators** for manipulating relations, which is equivalent in power to the calculus.
Database Design Process
Database design process can be divided into 6 major steps: (given in Raghu’s text book)
Requirements Analysis

• This step answers the following question:
  “What users want from the database?”
  - what is going to be stored in the database
  - what applications are going to be built on top of the database
  - what are the most frequently asked queries
Result:

A well-written concise document enumerating the user’s requirements
Requirements Analysis…

For example: a library database…
Data to be stored can be…

• Record of all books in the library

• Record of members of the library
  o Students
  o Faculty
  o Other members

• Record members’ borrowing information
Some applications on top of the database can be…

- Renewal service (may be on-line)
- Borrowing-Lending service
- Resource reservation system (may be on-line)
- Resource request service (may be on-line)
The information gathered in the requirements analysis phase is used to create a high-level description of the data in a conceptual data model. *(Semantic Data Model, e.g. E-R Diagram)*
In this step, we determine the DBMS to implement the database & also the data model

We utilize the conceptual schema created in the previous step and convert it into a schema of a particular data model (e.g. Relational Database Schema)
The schema created by the logical database design phase is further refined for potential problems such as redundancies (e.g. Normalization)
Physical Database Design

In this step, performance criteria are taken into consideration and further enhancements to the schema & creation of indexes are considered.
In this step, different user groups and their roles are identified. Appropriate levels of access are then provided to the data ensuring that users have access to only the necessary data.
The story is…

- Requirements Analysis
- Conceptual Database Design
- Logical Database Design
- Schema Refinement
- Physical Database Design
- Security Design

ER Diagram

Conceptual Schema or Logical Schema

Normalization
1. What is a data model? Describe the data models with example

2. Clearly explain the architecture of a DBMS

   - Submission Deadline 07/12/2012
   - Use IEEE Referencing Style
Questions