INTRODUCTION

Presenter: Ben B. K. Ayawli
Contact: 0243374387
Wechat contact: bbkayawli
Main Areas to be covered

- Introduction To Database Systems
- The Entity-relationship Model
- The Relational Model
- Relational Algebra And Calculus
- Storing Data: Disks And Files
- Database Design
- Schema Refinement And Normal Forms
- Security
- Transaction Management
- Crash Recovery
TOPIC

Introduction To Database Systems
Definition of a database

- A database is an integrated collection of related data records, files, and other objects.
- Examples:
  - Students records
  - Staff records
  - Etc.
Database Management Systems

- A collection of programs that enables the user to create and maintain a database.
- A software package with computer programs that control the creation, maintenance, and use of a database.
- DBMS, is software designed to assist in maintaining and utilizing large collections of data.
• It facilitates the process of defining, constructing and manipulating database for various applications.

• **Defining a database** involves specifying the data types, structures and constraints for the data to be stored in the database.

• **Constructing the database** is the process of storing the data itself on some storage medium that is controlled by the DBMS.

• **Manipulating a database** includes such functions as querying the database to retrieve specific data updating the database to reflect change and generation of reports from the data.
Facilities provided by DBMSs

- A DBMS provides facilities for:
  - controlling data access
  - enforcing data integrity,
  - managing concurrency control,
  - recovering the database after failures and restoring it from backup files
  - maintaining database security
Database Servers

– Database servers are dedicated computers that hold the actual databases and run only the DBMS and related software.

– Database servers are usually multiprocessor computers, with generous memory and RAID disk arrays used for stable storage.

– DBMSs are found at the heart of most database applications.

– Modern DBMSs typically rely on a standard operating system to provide these functions.
Problems with Traditional Systems

The traditional systems developed using COBOL have many shortcomings which has led to the development of DBMS.

- Each application identifies its data needs and decides on the physical representation of its data, which resulted in redundant storage of data in inconsistent format.
- The inconsistency among replicated data and difficulty to correlate data across different applications.
- Limited access to data available with the system.
- Any change in requirements entitled modifications to programs, which constitutes heavy maintenance and modification costs.
Advantage of using a DBMS

1. Controlling redundancy
2. Restricting unauthorized access
3. Providing persistent storage for program object and data structures
4. Providing multiple user interface
5. Presenting complex relationships among data
6. Enforcing integrity constraints
7. Providing backup and recovery
Advantages of DBMS

1. **Data independence**: Application programs should be as independent as possible from details of data representation and storage. The DBMS can provide an abstract view of the data to insulate application code from such details.

2. **Efficient data access**: A DBMS utilizes a variety of sophisticated techniques to store and retrieve data efficiently. This feature is especially important if the data is stored on external storage devices.
Advantages of DBMS (cont)

3. Data integrity and security: If data is always accessed through the DBMS, the DBMS can enforce integrity constraints on the data.

– For example, before inserting salary information for an employee, the DBMS can check that the department budget is not exceeded.
– Also, the DBMS can enforce access controls that governs the visibility of specified data to different classes of users.
4. Data administration: When several users share data, centralizing the administration of data can offer significant improvements.

– Experienced professionals who understand the nature of the data being managed, and how different groups of users use it, can be responsible for organizing the data representation to minimize redundancy and for fine-tuning the storage of the data to make retrieval efficient.
5. Concurrent access and crash recovery:

- A DBMS schedules concurrent accesses to the data in such a manner that users can think of the data as being accessed by only one user at a time.
- Also DBMS protects users from the effects of system failures.
History

• Databases have been in use since the earliest days of electronic computing. Originally DBMSs were found only in large organizations with the computer hardware needed to support large data sets.
A number of general-purpose database systems emerged as computers grew in speed and capability, by the mid-1960s there were a number of such systems in commercial use.

Interest in a standard began to grow, and Charles Bachman, author of one such product, the Integrated Data Store (IDS), founded the "Database Task Group" within CODASYL, the group responsible for the creation and standardization of COBOL.
The Codasyl (COnference on DAta SYstems Languages) approach was based on the "manual" navigation of a linked data set which was formed into a large network.

When the database was first opened, the program was handed back a link to the first record in the database, which also contained pointers to other pieces of data.

To find any particular record the programmer had to step through these pointers one at a time until the required record was returned.
• There was, essentially, no concept of "find" or "search".
• Solutions were found to many of these problems
  – Prime Computer created a CODASYL compliant DBMS based entirely on B-Trees that circumvented the record by record problem by providing alternate access paths
  – They also added a query language that was very straightforward
IBM also had their own DBMS system in 1968, known as IMS (IBM Information Mgt System).

IMS was generally similar in concept to Codasyl, but used a strict hierarchy for its model of data navigation instead of Codasyl's network model.

Both concepts later became known as navigational databases due to the way data was accessed.

IMS is classified as a hierarchical database.
Edgar Codd was unhappy with the navigational model of the Codasyl approach, notably the lack of a "search" facility.

In 1970, - a new approach to database construction that eventually culminated in the groundbreaking for a Relational Model of Data for Large Shared Data Banks.
Instead of records being stored in some sort of linked list of free-form records as in Codasyl, Codd's idea was to use a "table" of fixed-length records.

The relational model solved this by splitting the data into a series of normalized tables (or relations), with optional elements being moved out of the main table to where they would take up room only if needed.
Late - 1970s SQL DBMS

- IBM -- System R - based on Codd's concepts
  - work then started on multi-table systems in which the data could be split so that all of the data for a record (some of which is optional) did not have to be stored in a single large "chunk".

- IBM developed a true production version of System R, known as SQL/DS, and, later, Database 2 (DB2).

- Sybase, Informix, NonStop SQL and eventually Ingres

- Microsoft SQL Server is actually a re-built version of Sybase, and thus, INGRES
Late - 1970s SQL DBMS (cont)

- Only Larry Ellison's Oracle started from a different chain, based on IBM's papers on System R.
- Stonebraker went on to apply the lessons from INGRES to develop a new database, Postgres, which is now known as PostgreSQL.
- PostgreSQL is often used for global mission critical applications (the .org and .info domain name registries use it as their primary data store, as do many large companies and financial institutions).
- In Sweden, Codd's paper was also read and Mimer SQL was developed from the mid-70s at Uppsala University.
1980s - Object-Oriented Databases

- Programmers and designers began to treat the data in their databases as objects.
  - That is to say that if a person’s data were in a database, that person’s attributes, such as their address, phone number, and age, was now considered to belong to that person instead of being extraneous data.
- Allows for relations between data to be relations to objects and their attributes and not to individual fields.
- Focus on increasing reliability and access speeds.
- The practice of indexing, which is used by almost every operating system from Windows to the system that operates Apple iPod devices.
In the 21st century a new trend of NoSQL databases was started. Those non-relational databases are significantly different from the classic relational databases. They do not require fixed table schemas, avoid join operations by storing denormalized data. Can be classified as document-oriented databases. The most popular software in this category include: memcached, Redis, MongoDB, CouchDB, Apache Cassandra and HBase.
DATA MODELS

- Data modeling is a method used to define and analyze data requirements needed to support the business processes of an organization and by defining the data structures and the relationships between data elements.
Levels of Data models

• Composed of three levels of modeling:
  – The semantic model (Conceptual model)
  – The Logical model
  – The physical model
The semantic model

• Ways of interpreting and analysing the theories of logic
• Defining the meaning of data within the context of its interrelationships and constraints with other data
• It is an abstraction which defines how the stored symbols relate to the real world.
• Thus, the semantic model must be a true representation of the real world.
• A semantic model consists of entity classes, representing kinds of things of significance in the domain, and relationships assertions about associations between pairs of entity classes.
• A semantic model specifies the kinds of facts or propositions that can be expressed using the model.
The logical model

- Describes the system information, as represented by a particular data manipulation technology type:
  - e.g. flat file system
  - Hierarchical DBMS (IMS),
  - Network dbms (IDMS, IDS2,..),
  - Relational DBMS (DB2, ORACLE, SQL SERVER, MySQL...).
Logical model (cont)

– A logical model consists of descriptions of:

• **entities** called
  – « segments » in hierarchical DBMS,
  – « records » in network DBMS,
  – « tables » in relational DBMS) and

• **attributes** called
  – « data » in hierarchical and network DBMS,
  – « columns » in relational DBMS),

• **data access keys**, 

• **type of links between entities** (called « sets » in network DBMS, « foreign keys » in relational DBMS) among other things;
The physical model

- Describes the physical means by which data are stored in a particular DBMS product (flat files, XML files, IMS, IDS2, IDMS, ORACLE, DB2, ...).
- This is concerned with partitions, CPUs, tablespaces, and the like.
Types of database models

Five

- Hierarchical model
- Network model
- Relational model
- Deductive model
- Object Oriented model
The hierarchical model

- The first DBMS model
- Data is sorted hierarchically, using a downward tree.
- This model uses pointers to navigate between stored data.
- The hierarchical data model organizes data in a tree structure.
- There is a hierarchy of parent and child data segments.
- This structure implies that a record can have repeating information, generally in the child data segments - data in a series of records, which have a set of field values attached to it.
The hierarchical model (cont)

• It collects all the instances of a specific record together as a record type.
• These record types are the equivalent of tables in the relational model, and with the individual records being the equivalent of rows.
• To create links between these record types, the hierarchical model uses Parent Child Relationships.
• These are a 1:N mapping between record types.
This is done by using trees, like set theory used in the relational model, "borrowed" from Maths.

- For example, an organization might store information about an employee, such as name, employee number, department, salary.

- The organization might also store information about an employee's children, such as name and date of birth. The employee and children data forms a hierarchy, where the employee data represents the parent segment and the children data represents the child segment. If an employee has three children, then there would be three child segments associated with one employee segment.
In a hierarchical database the parent-child relationship is one to many. This restricts a child segment to having only one parent segment.

This structure is simple but nonflexible because the relationship is confined to a one-to-many relationship.
Hierarchical model
The network model

- Like the hierarchical model, this model uses pointers toward stored data. However, it does not necessarily use a downward tree structure.
- Unlike the hierarchical structure, it can relate to many records and accesses them by following one of several paths.
- Thus, the structure allows for many-to-many relationships.
- In 1971, the Conference on Data Systems Languages (CODASYL) formally defined the network model.
Network model (cont)

- The basic data modeling construct in the network model is the set construct.
- A set consists of an owner record type, a set name, and a member record type.
- A member record type can have that role in more than one set, hence the multiparent concept is supported.
- An owner record type can also be a member or owner in another set.
- The data model is a simple network, and link and intersection record types (called junction records by IDMS) may exist, as well as sets between them.
Network model (cont)

• The CODASYL network model is based on mathematical set theory.
• The network model's original inventor was Charles Bachman, and it was developed into a standard specification published in 1969 by the CODASYL Consortium.
The relational model (RDBMS)

- The data is stored in two-dimensional tables (rows and columns).
- The tables of records can be connected by common key values.
- The model is not easy for the end user to run queries with because it may require a complex combination of many tables.
- A table is a collection of records and each record in a table contains the same fields.
Relational model (cont)

- Certain fields may be designated as keys, which means that searches for specific values of that field will use indexing to speed them up.
- Where fields in two different tables take values from the same set, a join operation can be performed to select related records in the two tables by matching values in those fields.
- Because the relationships are only specified at retrieval time, relational databases are classed as dynamic database management system.
- The RELATIONAL database model is based on the Relational Algebra.
Relational model
The deductive model

- Data is represented as a table just as in relational model, but is manipulated using predicate calculus
The object model (ODBMS)

- Data is stored in the form of objects, which are structures called classes that display the data within.
- The fields are instances of these classes.
- The object-oriented structure has the ability to handle graphics, pictures, voice and text; and other types of data, without difficulty unlike the other database structures.
- Object DBMSs add database functionality to object programming languages.
- Object DBMSs extend the semantics of the C++, Smalltalk and Java object programming languages to provide full-featured database programming capability, while retaining native language compatibility.
• A major benefit of this approach is the unification of the application and database development into a seamless data model and language environment.
• Applications require less code, use more natural data modeling, and code bases are easier to maintain.
• Object developers can write complete database applications with a modest amount of additional effort.
Assignment

1. Differentiate between Database and Database management systems

2. Briefly explain the following:
   a. Defining a database
   b. Constructing a database
   c. Manipulating a database

3. Identify and discuss FIVE benefits of using DBMS

4. Give TWO differences between the hierarchical database model and the network database model

5. What is the basic difference between Relational database model and the deductive database model?