BAHRIA UNIVERSITY, (Karachi Campus)

Department of Software Engineering

Assignment#01– Spring 2024

COURSE TITLE:	**DBMS**	COURSE CODE:	**CSC-220**
Class:	**BSE 4**	Shift:	**Morning**
Course Instructor:	**ENGR. BUSHRA FAZAL KHAN**	Assignment Date:	**26-Feb-2024**
Max. Marks:	**3 Points [CLO2]**	Assignment Due:	**01-Mar-2024**

NAME: Ali Agha

Enrollment #: 02-131222-107

BSE-4(A)

Q1) Explain the following Data Models along with advantages and disadvantages. Also give one real world example for each Model.

• Hierarchical Model • Network Model • Relational Model • ER Model • OODM • Extended ER Model • NoSQL • NewSQL

ANSWER

1. Hierarchical Model:

Description: In the hierarchical model, data is organized in a tree-like structure with a single root and multiple levels of child nodes. Each parent can have multiple children, but each child has only one parent. It was developed in 1970s under Second Generation, used mainly for early database systems. Even if the segments are connected as a chain-like structure by logical associations, then the instant structure can be a fan structure with multiple branches. We call the illogical associations as directional associations.

In the hierarchical model, segments pointed to by the logical association are called the child segment and the other segment is called the parent segment. If there is a segment without a parent is then that will be called the root and the segment which has no children are called the leaves.

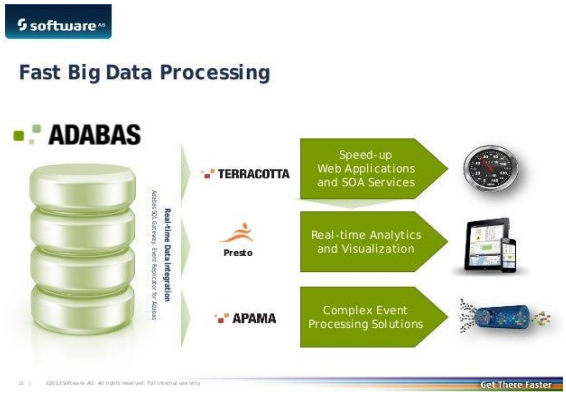
Advantages:

- Simple and easy to understand.
- Efficient for representing one-to-many relationships.

Disadvantages:

- Lack of flexibility (no many-to-many relationships).
- Difficult to modify the structure once data is inserted.

Example: **File system directories** (folders and subfolders), ADABAS.



2. <u>Network Model</u>:

Description: The network model extends the hierarchical model by allowing many-to-many relationships. It uses pointers to connect records. It was also developed in 1970s under Second Generation, used mainly for early database systems for navigation access. A unique feature of the network model is its schema, which is viewed as a graph where relationship types are arcs and object types are nodes.

Unlike other database models, the network model's schema is not confined to be a lattice or hierarchy; the hierarchical tree is replaced by a graph, which allows for more basic connections with the nodes.

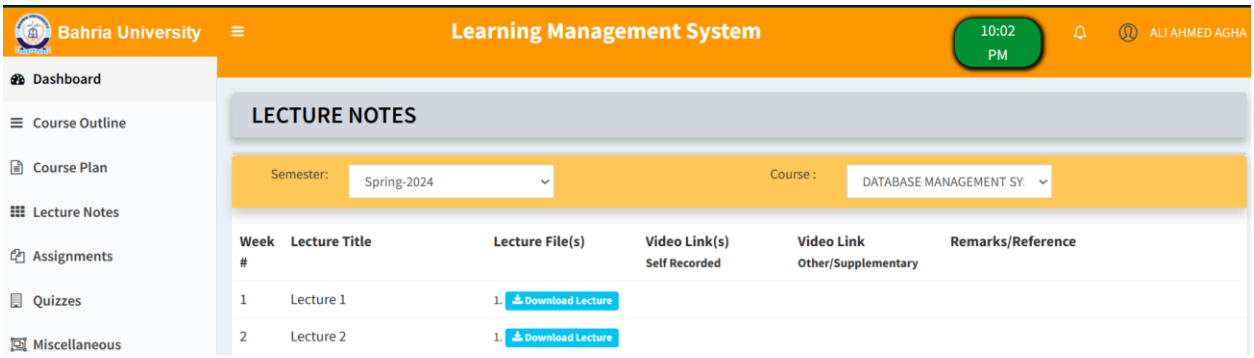
Advantages:

- Supports complex relationships.
- More flexible than the hierarchical model.

Disadvantages:

- Complexity in implementation.
- Difficult to maintain.

Example: **CODASYL (networked system) databases** used in early mainframe systems such as LMS.



3. Relational Model:

Description: The relational model organizes data into tables (relations) with rows (tuples) and columns (attributes). It uses keys to establish relationships. It was developed in mid 1970s under third Generation, used mainly for conceptual simplicity and ER-Modeling. The relational model uses a collection of tables to represent both data and the relationships among those data. Each table has multiple columns, and each column has a unique name. Tables are also known as relations. The relational model is an example of a record-based model. Record-based models are so named because the database is structured in fixed-format records of several types. Each table contains records of a particular type. Each record type defines a fixed number of fields, or attributes.

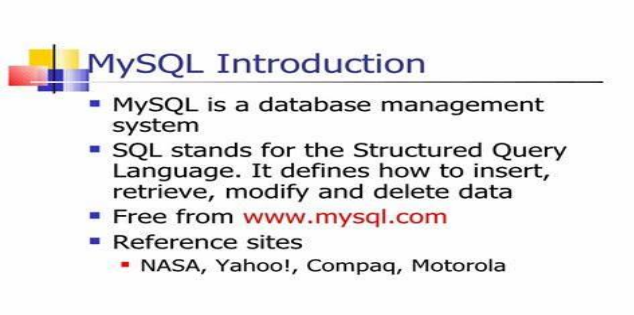
Advantages:

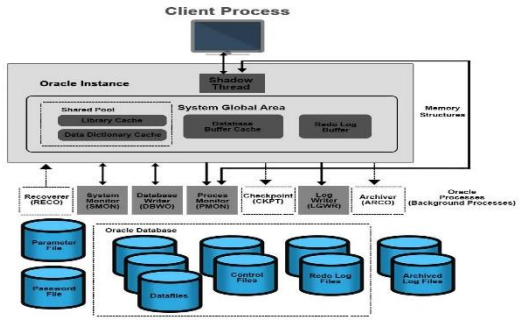
- Simplicity and ease of use.
- Data independence.

Disadvantages:

- Performance issues with large datasets.
- Lack of support for complex relationships.

Example: **MySQL**, **PostgreSQL**, and **Oracle** databases.





The relational model represents how data is stored in Relational Databases. A relational database consists of a collection of tables, each of which is assigned a unique name. Consider a relation STUDENT with attributes ROLL_NO, NAME, ADDRESS, PHONE, and AGE shown in the table.

4. ER Model (Entity-Relationship Model):

Description: The ER model represents entities, attributes, and relationships between them. It uses entity-relationship diagrams (ERDs) to visualize the database structure. It was developed by PETER CHEN in 1976.

The Entity Relational Model is a model for identifying entities to be represented in the database and representation of how those entities are related. The ER data model specifies enterprise schema that represents the overall logical structure of a database graphically.

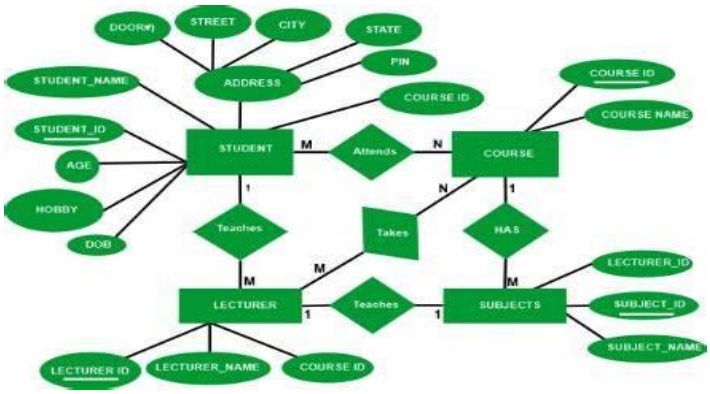
Advantages:

- Clear representation of data requirements.
- Useful for designing databases.
- ER diagrams are used to represent the E-R model in a database, which makes them easy to convert into relations (tables).
- ER diagrams provide the purpose of real-world modeling of objects which makes them intently useful.
- ER diagrams require no technical knowledge and no hardware support.

Disadvantages:

- Doesn’t directly map to a physical database.
- Limited support for implementation details.

Example: Designing a university database with entities like **Student**, **Course**, and **Professor**, ER Model for university, Bank etc.



5. OODM (Object-Oriented Data Model):

Description: The OODM treats data as objects with attributes and methods. It combines object-oriented programming concepts with database management. It was developed in mid 1980s under fourth Generation. Web databases became common in that time. In Object Oriented Data Model, data and their relationships are contained in a single structure which is referred as object in this data model. In this, real world problems are represented as objects with different attributes. All objects have multiple relationships between them. Basically, it is combination of Object Oriented programming and Relational Database Model

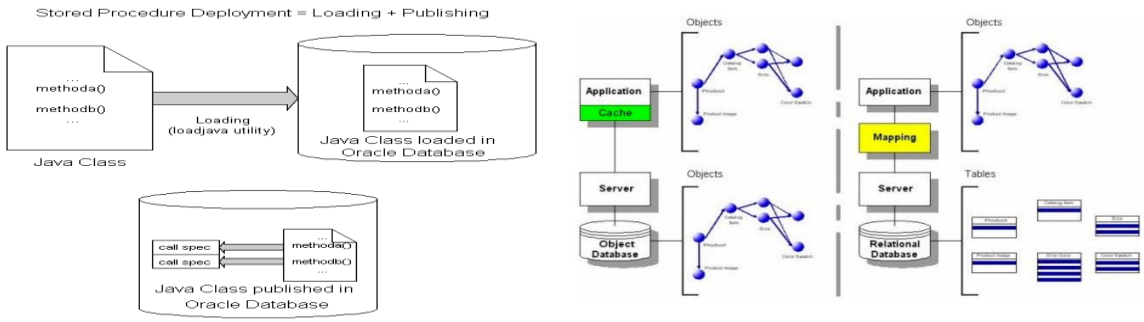
Advantages:

- Supports complex data structures.
- Encapsulation and inheritance.

Disadvantages:

- Lack of standardization.
- Performance overhead.

Example: **Java Object Database (db4o), db, db2**.



Db40 Database for java

6. Extended ER Model:

Description: Extends the ER model by adding additional constructs like generalization/specialization, aggregation, and weak entities. Idea was put forward by PETER CHEN in 1976.

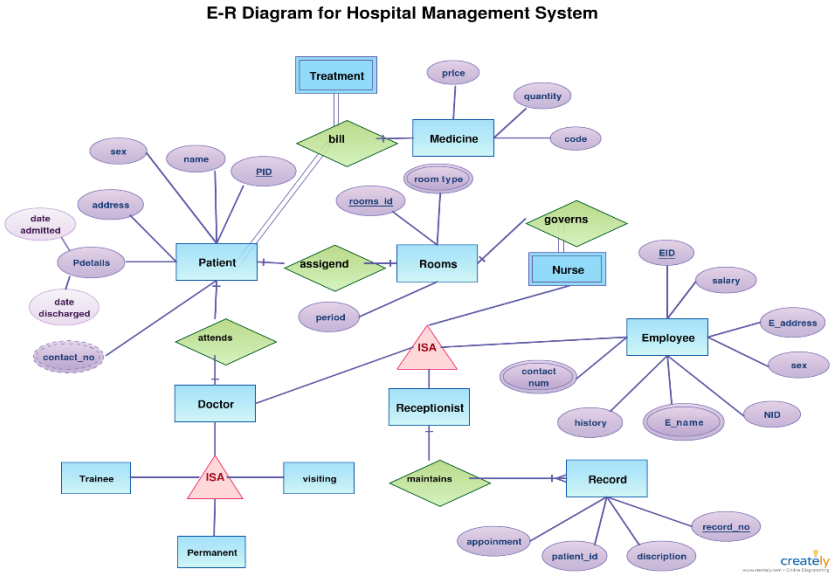
Advantages:

- Enhanced modeling capabilities.
- Better representation of real-world scenarios.

Disadvantages:

- Increased complexity.
- May not be supported by all DBMS.

Example: Modeling a **hospital management system** with patients, doctors, and medical records.



7. NoSQL (Not Only SQL):

Description: NoSQL databases are non-relational and provide flexible schema design. They handle large volumes of unstructured or semi-structured data. It is an approach to database design that focuses on providing a mechanism for storage and retrieval of data that is modeled in means other than the tabular relations used in relational databases. Instead of the typical tabular structure of a relational database, NoSQL databases house data within one data structure. Since this non-relational database design does not require a schema. It was developed in 1998.

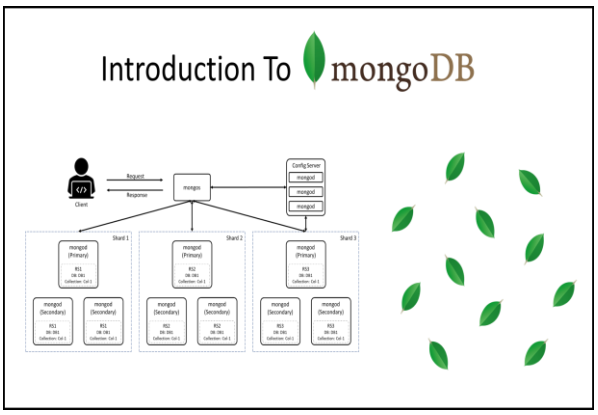
Advantages:

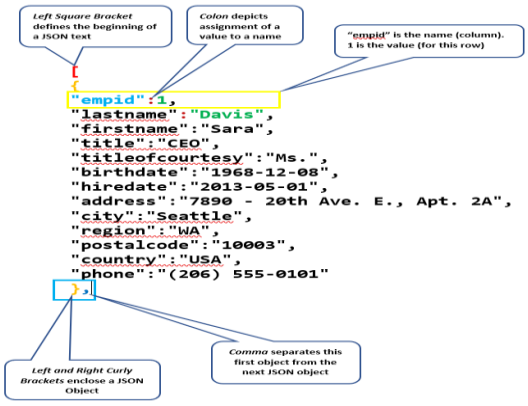
- Scalability and high performance.
- Schema flexibility.

Disadvantages:

- Lack of ACID (Atomicity, Consistency, Isolation, and Durability) transactions.
- Limited query capabilities.

Example: **MongoDB**, used for storing JSON-like documents.





Json Document in SQL

8. NewSQL:

Description: NewSQL databases combine the benefits of NoSQL (scalability) with the ACID properties of traditional relational databases. NewSQL is a class of relational database management systems that seek to provide the scalability of NoSQL systems for online transaction processing (OLTP) workloads while maintaining the ACID guarantees of a traditional database system. The term was first used by 451 Group analyst Matthew Aslett in a 2011 research paper discussing the rise of a new generation of database management systems.

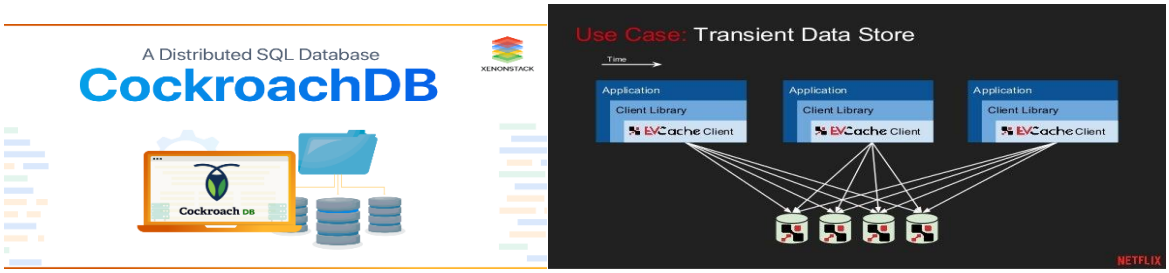
Advantages:

- Scalability without compromising consistency.
- ACID compliance.

Disadvantages:

- Limited adoption compared to traditional RDBMS.

Example: **CockroachDB**, designed for global scalability, Netflix Netflix uses NoSQL databases to store and manage massive amounts of data, including customer profiles, viewing histories, and content recommendations.



COCKROACH DB example for Netflix database

Q2) Explain the RDBMS rules given by E. F. Codd.

ANSWER

Description:

Codd's rules are proposed by a computer scientist named Dr. Edgar F. Codd and he also invent the relational model for database management. These rules are made to ensure data integrity, consistency, and usability. This set of rules basically signifies the characteristics and requirements of a relational database management system (RDBMS). In this article, we will learn about various Codd's rules.

Rule 1: The Information Rule

All information, whether it is user information or metadata, that is stored in a database must be entered as a value in a cell of a table. It is said that everything within the database is organized in a table layout.

Rule 2: The Guaranteed Access Rule

Each data element is guaranteed to be accessible logically with a combination of the table name, primary key (row value), and attribute name (column value).

Rule 3: Systematic Treatment of NULL Values

Every Null value in a database must be given a systematic and uniform treatment.

Rule 4: Active Online Catalog Rule

The database catalog, which contains metadata about the database, must be stored and accessed using the same relational database management system.

Rule 5: The Comprehensive Data Sublanguage Rule

A crucial component of any efficient database system is its ability to offer an easily understandable data manipulation language (DML) that facilitates defining, querying, and modifying information within the database.

Rule 6: The View Updating Rule

All views that are theoretically updatable must also be updatable by the system.

Rule 7: High-level Insert, Update, and Delete

A successful database system must possess the feature of facilitating high-level insertions, updates, and deletions that can grant users the ability to conduct these operations with ease through a single query.

Rule 8: Physical Data Independence

Application programs and activities should remain unaffected when changes are made to the physical storage structures or methods.

Rule 9: Logical Data Independence

Application programs and activities should remain unaffected when changes are made to the logical structure of the data, such as adding or modifying tables.

Rule 10: Integrity Independence

Integrity constraints should be specified separately from application programs and stored in the catalog. They should be automatically enforced by the database system.

FIGURE 3.3 An illustration of integrity rules

Table name: CUSTOMER **Database name: Ch03_InsureCo**
Primary key: CUS_CODE
Foreign key: AGENT_CODE

CUS_CODE	CUS_LNAME	CUS_FNAME	CUS_INITIAL	CUS_RENEW_DATE	AGENT_CODE
10010	Ramas	Alfred	A	05-Apr-2012	502
10011	Dunne	Leona	K	16-Jun-2012	501
10012	Smith	Kathy	W	29-Jan-2013	502
10013	Olowski	Paul	F	14-Oct-2012	
10014	Orlando	Myron		28-Dec-2012	501
10015	O'Brian	Amy	B	22-Sep-2012	503
10016	Brown	James	G	25-Mar-2013	502
10017	Williams	George		17-Jul-2012	503
10018	Farriss	Anne	G	03-Dec-2012	501
10019	Smith	Olette	K	14-Mar-2013	503

Table name: AGENT (only five selected fields are shown)
Primary key: AGENT_CODE
Foreign key: none

AGENT_CODE	AGENT_AREACODE	AGENT_PHONE	AGENT_LNAME	AGENT_YTD_SLS
501	713	228-1249	Alby	132735.75
502	615	882-1244	Hahn	138967.35
503	615	123-5589	Okon	127093.45

SOURCE: Course Technology/Cengage Learning

Rule 11: Distribution Independence

The distribution of data across multiple locations should be invisible to users, and the database system should handle the distribution transparently.

Rule 12: Non-Subversion Rule

If the interface of the system is providing access to low-level records, then the interface must not be able to damage the system and bypass security and integrity constraints.

REFERENCES:

https://www.bing.com/ck/a?!&&p=3aa00a0df224ebadJmltdHM9MTcwOTc2OTYwMCZpZ3VpZD0wNzQ1YmI0Yi04ZWZjLTY5MDAtMjM2Zi1hODg1OGYyNzY4NTEmaW5zaWQ9NTQ1OA&ptn=3&ver=2&hsh=3&fclid=0745bb4b-8efc-6900-236f-a8858f276851&psq=Q2)+Explain+the+RDBMS+rules+given+by+E.+F.+Codd&u=a1aHR0cHM6Ly93d3cuZ2Vla3Nmb3JnZWVrcy5vcmcvY29kZHMtcnVsZXMtaW4tZGJtcy8&ntb=1

json documents - Search (bing.com)

https://www.bing.com/ck/a?!&&p=1e996353fc6657cbJmltdHM9MTcwOTc2OTYwMCZpZ3VpZD0wNzQ1YmI0Yi04ZWZjLTY5MDAtMjM2Zi1hODg1OGYyNzY4NTEmaW5zaWQ9NTE5NQ&ptn=3&ver=2&hsh=3&fclid=0745bb4b-8efc-6900-236f-a8858f276851&psq=codasyl+database&u=a1aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvQ09EQVNZTA&ntb=1