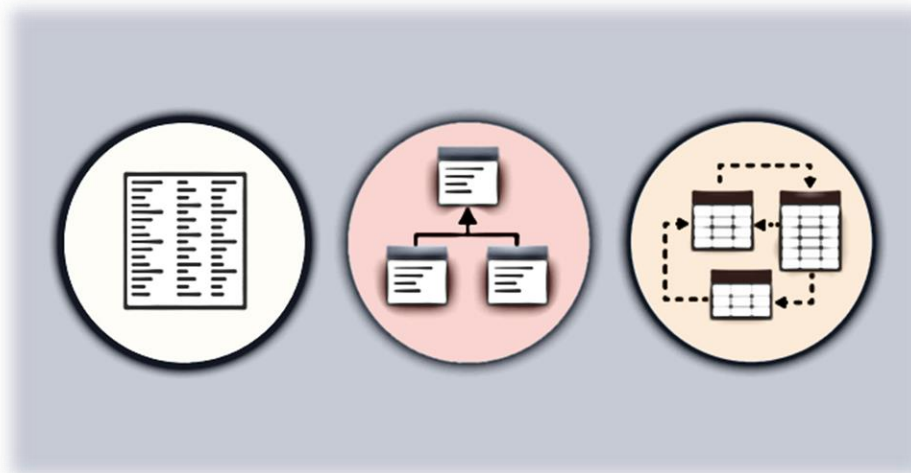


What is a Data Model?

Data model is a collection of conceptual tools for describing data, data relationships, data semantics and consistency constraints. That means a data model provides a way to describe the design of a database.

Data models are widely used in database management systems to demonstrate how data is related, stored, retrieved, and modified. We depict information using a set of symbols and terminology so that members of an organisation may understand and comprehend it before communicating.



Data model basic building blocks

Entity

An entity is anything about which data are to be collected and stored. An entity represents a particular type of object in the real world.

Entity Set

Set of entities of the same type that share the same properties are called as entity sets.

Attribute

An attribute is a characteristic of an entity.

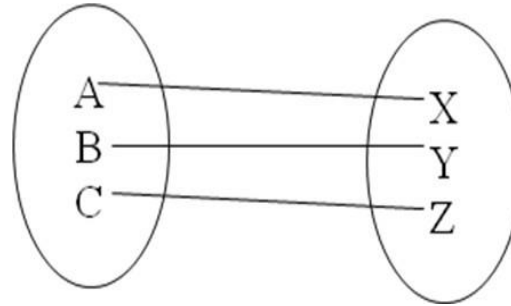
Constraints

A constraint is a restriction placed on the data. Constraints are important because they help to ensure data integrity.

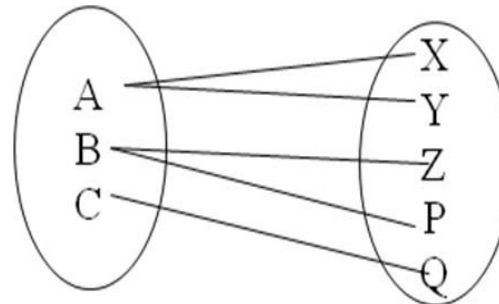
Relationships

A relationship describes an association among entities.
Different types of relationship are:

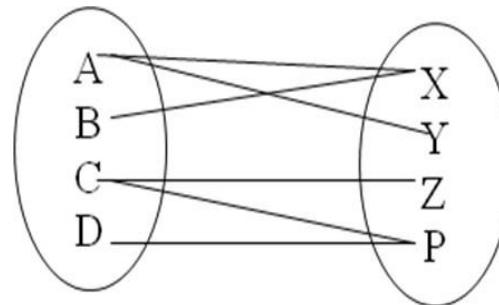
One to One (1:1) relationship:



One to Many (1:M) relationship:

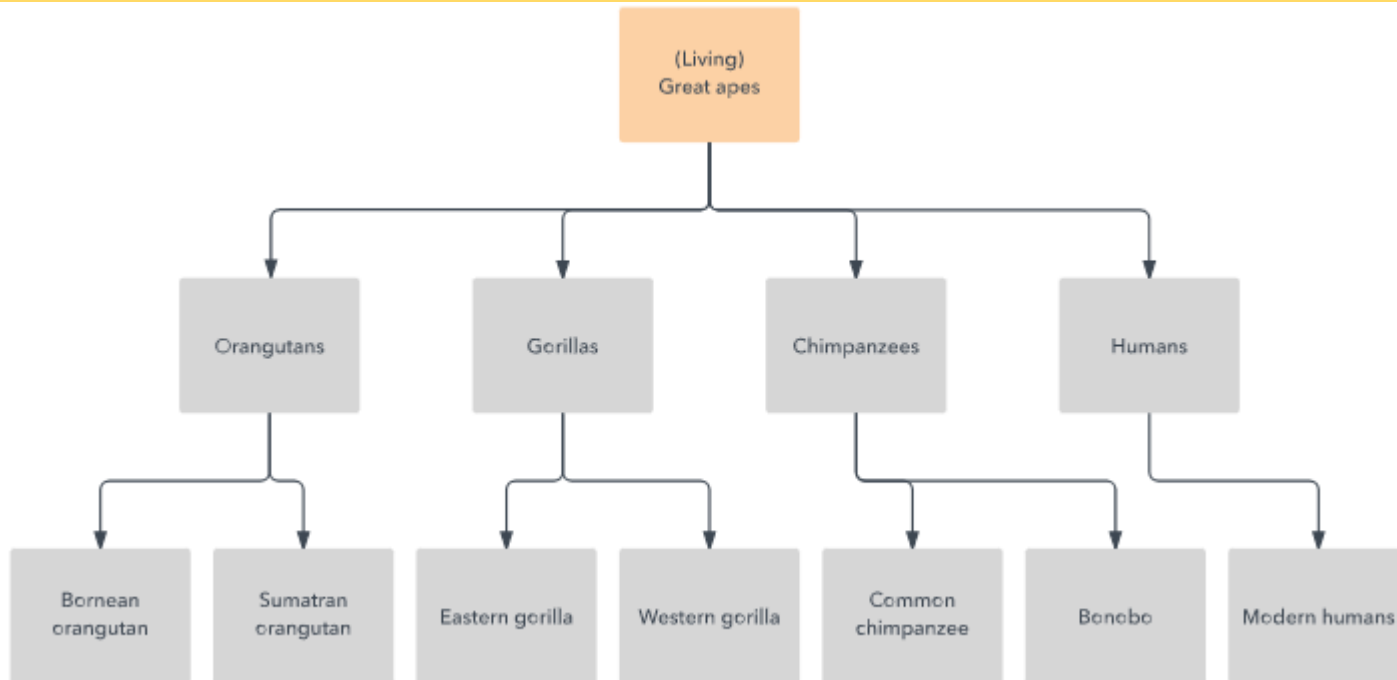


Many to Many (M:N) relationship:



Hierarchical model

The hierarchical model classifies the data into a tree-like structure with a single parent or root for each record. Sibling records are arranged in a certain sequence. This is the physical order in which the database is stored. This model is useful for explaining a wide range of real-world interactions.



It depicts a set of 1:M relationships between a parent and its children segments

Cont...

Advantages of Hierarchical Model

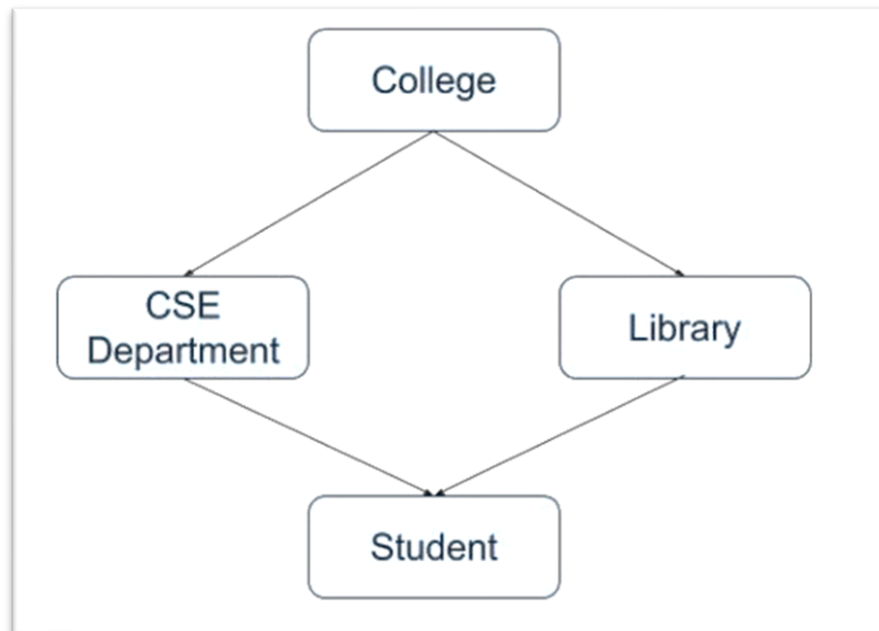
- It is very simple and fast to traverse through a tree-like structure.
- Any change in the parent node is automatically reflected in the child node so, the integrity of data is maintained.

Disadvantages of Hierarchical Model

- Complex relationships are not supported.
- As it does not support more than one parent of the child node so if we have some complex relationship where a child node needs to have two parent node then that can't be represented using this model.
- If a parent node is deleted then the child node is automatically deleted.

Network Model

This paradigm is an expansion of the hierarchical model. It was the most prevalent model prior to the relational model. The main difference between this model and the hierarchical approach is that a record can have several parents. It replaces the hierarchical tree with a graph. In the sample below, we can see that node student has two parents, CSE Department and Library. This was previously not allowed in the hierarchical model.



Cont...

Advantages of Network Model

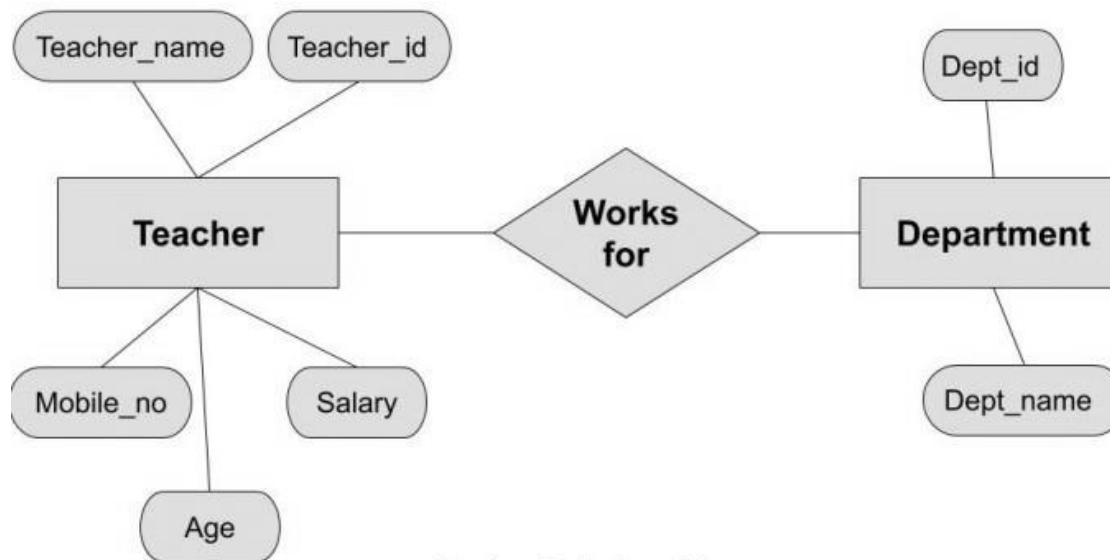
- When compared to the hierarchical approach, data may be obtained more quickly. This is because the data in the network model is more linked and there may be more than one path to a certain node. As a result, the information may be accessed in a variety of ways.
- Data integrity is present since there is a parent-child connection. Any changes made to the parent record are mirrored in the child record.

Disadvantages of Network Model

- As additional relationships are added, the system may get increasingly complicated. To operate with the model, a user needs have full understanding of it.
- Any modification, such as an update, deletion, or insertion, is quite difficult.

Entity-Relationship Model

The Entity-Relationship Model, sometimes known as the ER Model, is a high-level data model diagram. We express the real-world problem in visual form in this model to make it easier for stakeholders to grasp. The ER diagram also makes it extremely simple for developers to comprehend the system. The ER diagram is a visual tool used to depict an ER Model. The ER diagram is made up of three parts:



Entity-Relationship Model

Cont...

Advantages of Entity-Relationship Model

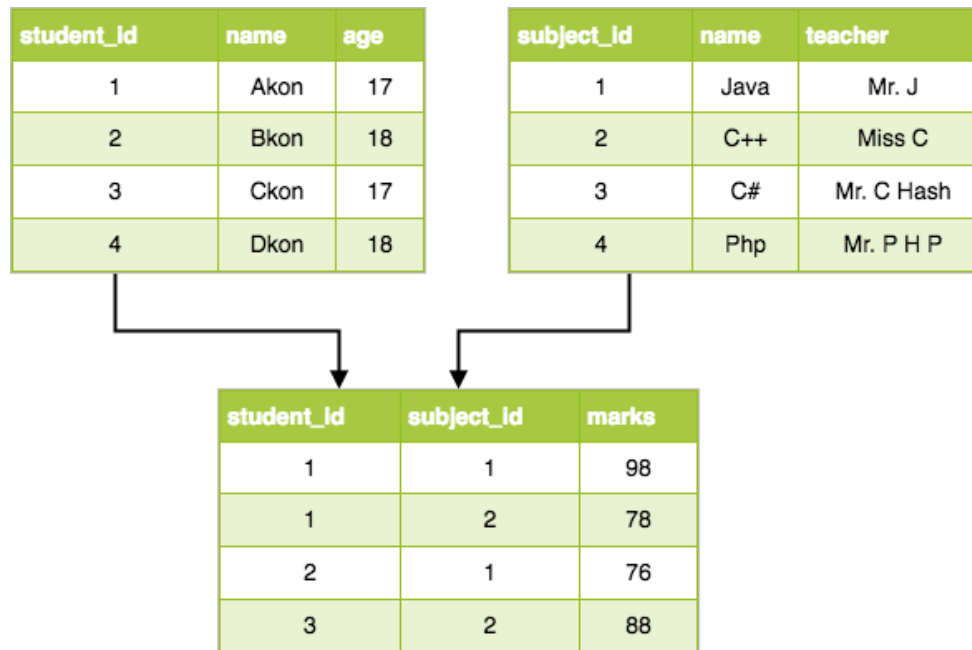
- Simple: Conceptually ER Model is very easy to build. If we know the relationship between the attributes and the entities we can easily build the ER Diagram for the model.
- Effective Communication Tool : This model is used widely by the database designers for communicating their ideas.
- Easy Conversion to any Model : This model maps well to the relational model and can be easily converted relational model by converting the ER model to the table. This model can also be converted to any other model like network model, hierarchical model etc.

Disadvantages of Entity-Relationship Model

- No industry standard for notation: There is no industry standard for developing an ER model. So one developer might use notations which are not understood by other developers.
- Hidden information: Some information might be lost or hidden in the ER model. As it is a high-level view so there are chances that some details of information might be hidden.

Relational Model

The most common model is the Relational Model. The data in this model is kept in the form of a two-dimensional table. All data is saved in the form of rows and columns. Tables are the foundation of a relational paradigm. In the relational paradigm, the tables are also referred to as relations. Example: We have an Employee table in this example.



Cont...

Advantages of Relational Data Model

- Simple: This model is more simple as compared to the network and hierarchical model.
- Scalable: This model can be easily scaled as we can add as many rows and columns we want.
- Structural Independence: We can make changes in database structure without changing the way to access the data. When we can make changes to the database structure without affecting the capability to DBMS to access the data we can say that structural independence has been achieved.

Disadvantages of Relational Data Model

- Hardware Overheads: For hiding the complexities and making things easier for the user this model requires more powerful hardware computers and data storage devices.
- Bad Design: As the relational model is very easy to design and use. So the users don't need to know how the data is stored in order to access it. This ease of design can lead to the development of a poor database which would slow down if the database grows.

Object-Oriented(OO) Model

A database, according to this paradigm, is a collection of objects, or reusable software parts, with related characteristics and operations.

- Object-oriented databases are classified into numerous types: A multimedia database contains media, such as photos, that would be impossible to store in a relational database.
- Any object in a hypertext database can link to any other object. It is effective for organizing a large amount of heterogeneous data, but it is not perfect for numerical analysis.
- Because it integrates but is not limited to tables, the object-oriented database model is the most well-known post-relational database paradigm. These types of models are often known as hybrid database models.

Employee
Attributes
Name
Job_Title
Phone_No
Salary
Dept_ID
Methods
Get Hired
Change Number

OODs are most often used with object-oriented programming languages like Java, Kotlin, C#, Node JS (React), and Swift.

Cont...

Advantages of Object Oriented Data Model

- Complex data sets can be saved and retrieved quickly and easily.
- Object IDs are assigned automatically.
- Works well with object-oriented programming languages.
- Semantic content is added
- Support for complex objects
- Visual representation includes semantic content

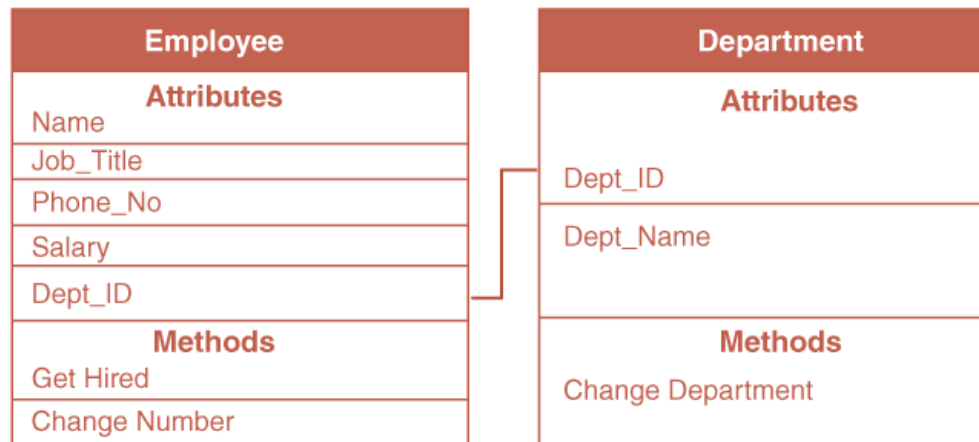
Disadvantages of Object Oriented Data Model

- Object databases are not widely adopted.
- In some situations, the high complexity can cause performance problems.
- High system overheads slow transactions.

Object-Relational (OR) Model

As the name implies, it is a hybrid of the relational and object-oriented models. This paradigm was created to bridge the gap between the object-oriented and relational models. We can have many sophisticated features, such as the ability to create complicated data types based on our needs utilising current data types. The issue with this paradigm is that it may become complicated and difficult to manage.

As a result, a thorough comprehension of this paradigm is essential.



OO vs OR Data Model

Object-Oriented database

- A database that represents information in the form of objects as inspired by Object oriented programming.
- Depends on OOP.
- In object oriented database, relationships are represented by references via the object identifier (OID).
- Handles larger and complex data than RDBMS.
- Less efficient.
- The data management language is typically incorporated into a programming language such as #C++.

VS

Object-Relational database

- A database that depends on Relational data model and Object oriented data model.
- Hybrid of relational and object oriented model.
- In object relational database, connections between two relations are represented by foreign key attributes in one relation that reference the primary key of another relation.
- Handles comparatively simpler data.
- Improvement and comparatively more efficient.
- There are data manipulation languages such as SQL, QUEL and QBE which are based on relational calculus.

Semi-structured Model

The semi-structured model emerged from the relational model.

In this paradigm, we can't tell the difference between data and schema. Web-based data sources, for example, when we cannot distinguish between the website's structure and data. Some entities in this model may be missing characteristics, while others may have an additional attribute. This approach allows for greater data storage flexibility. It also provides qualities with flexibility.

A few examples of semi-structured data sources are emails, XML and other markup languages, binary executables, TCP/IP packets, zipped files, data integrated from different sources, and web pages.