DATA MODELS

The evolution of data models began with the need to overcome the limitations of file-based systems. Early file systems were plagued by redundancy, inconsistency, poor security, and the difficulty of accessing related data. As organizations grew and data volume increased, it became necessary to establish more structured ways of organizing information. A data model provides a logical view of how data is arranged, how it is connected, and how it can be accessed. It shapes the architecture of a database and influences how efficiently data is stored and retrieved.

Over time, several major data models emerged: the hierarchical model, the network model, the relational model, and the object-oriented model. Each of these represents a different philosophy of organizing data, and each addresses the shortcomings of earlier models in its own way.

a) Hierarchical Model

The hierarchical data model organizes records in a tree structure *i.e.*, hierarchy of parent and child records relationships. The hierarchical structure contains levels, or segments. Within the hierarchy, a higher layer is perceived as the parent of the segment, directly beneath it is the child. The hierarchical model depicts a set of one-to-many (1:M) relationships between a parent and its children's segments. (Each parent can have many children, but each child has only one parent).

Advantage: Hierarchical model relates well to anything that works through a one-to-many relationship. This model is good for hierarchies such as employees in an organization or an inventory of plant specimen in a museum. In addition, data at the top of the hierarchy is accessed with great speed.

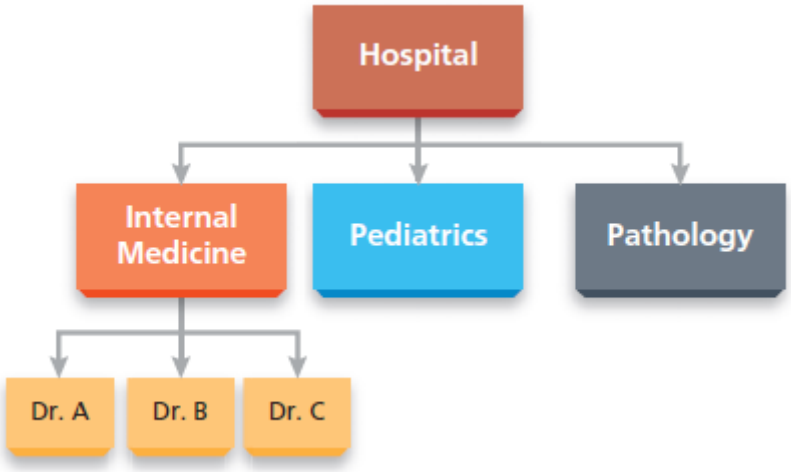

Fig 1: Hierarchical data model

Disadvantage: This type of data model permits each child a relationship with only one parent, and relationships or linkages between children are not permitted, even if they make sense from a logical standpoint. This limitation is circumvented by a repetition of data, which adds to the size of the database. Searching for specific data requires the DBMS to run through the entire data from top to bottom until the required information is found, making queries very slow. The lower the required data in the hierarchy, the longer it takes to retrieve it.

b) Network Model

The network model was created to represent complex data relationships more effectively than the hierarchical model, to improve database performance, and to impose a database standard. Unlike the hierarchical model, the network model allows a record to have more than one parent. It supports many-to-many relationships and can be visualized as a cobweb or interconnected network of records.

Advantages: Network model is conceptually simple and easy to design. The data access is easier and more flexible as compared to a hierarchical model. The main advantage of a network database is that it can handle more complex data because of its many-to-many relationship. This makes it efficient for accessing related data, especially in large, interlinked systems like transportation or telecommunications. It

allows for a more natural modelling of relationships between records or entities, as opposed to the hierarchical model.

Disadvantages: The main disadvantage is that all the records in the database need to be maintained using pointers, making the whole database structure very complex. The insertion, deletion and updating operations of any record require an adjustment of a large number of pointers. Network databases are difficult to use by first time users. Structural changes to the database are very difficult to implement. Difficulties are encountered while making alterations to the database because entering new data may necessitate altering the entire database.

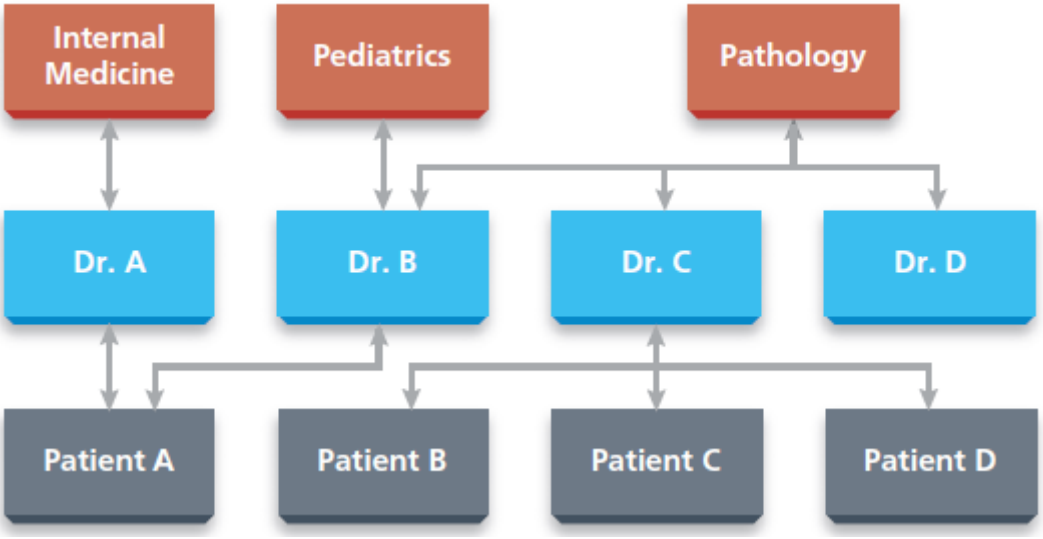

Fig 2: Network Data Model

c) Relational Model

The relational model was introduced in 1970 by E. F. Codd (of IBM). The relational model foundation is a mathematical concept known as a relation. To avoid the complexity of abstract mathematical theory, you can think of a relation (sometimes called a table) as a matrix composed of intersecting rows and columns. Each row in a relation is called a tuple. Each column represents an attribute.

Hierarchical and network databases require the user to pass a hierarchy in order to access needed data. These databases connect to the data in different files by using

common data numbers or a key field. In relational database, there is a relationship between data and that is stored in the form of the table of rows and columns. Every individual field represents the data value. Every row has its unique key field and these key fields are used to connect one table to another one. Examples include Oracle Database, MySQL, Microsoft SQL Server, and IBM DB2.

Advantages: Data stored as tables made up of rows and columns is easy to understand because it resembles familiar spreadsheet structures. Information can be separated into multiple tables containing logically related attributes, which allows large datasets to be segmented into manageable units. This greatly improves data organization, management, and retrieval.

Because tables can be linked using keys, relationships between different types of data are easy to define, query, and maintain. Security can also be implemented effectively. Sensitive information can be placed in separate tables with their own authorization rules, thereby restricting access to specific users.

The relational model supports a high level of data independence because changes to the logical structure of a table do not necessarily affect applications that access the data. It minimizes redundancy through normalization and reduces unnecessary replication. Additionally, relational DBMSs provide strong support for backup and recovery, making data protection and restoration more efficient and reliable.

Students

StudentID	LastName	FirstName	BirthDate
54001	Chong	Kevin	12/01/1987
65222	Danelli	Douglas	01/05/1986
54555	Burton	Stephanie	11/12/1978
25553	Washington	Nikia	10/02/1981
96887	Perez	Louis	07/25/1982

Fig 3: Relational data model

Disadvantages: A major constraint, and therefore disadvantage of relational model is its reliance on machine performance. If the number of tables between which relationships are to be established is large, then the performance in responding to the

SQL queries is affected. Also, relational models are often less flexible when handling unstructured or semi-structured data, such as documents, multimedia, or social media content. Because data in relational databases needs to fit into predefined tables and columns, handling data that doesn't fit neatly into this format can be cumbersome

d) Object -Oriented Data Model

As software systems became more complex, the relational model began to show limitations, especially for applications involving multimedia, scientific simulations, engineering designs, and real-time systems. The object-oriented data model emerged to integrate database capabilities with object-oriented programming concepts. In this model, data is represented as objects. An object contains both its data (attributes) and the operations (methods) that act on the data. This reflects how real-world entities behave. For example, a Car object contains attributes such as model, color, and speed, and it also contains methods such as accelerate(), brake(), or displaySpeed(). These objects can be combined into larger objects through relationships such as "is-a" or "is-part-of".

Object-oriented databases are more flexible and adaptable because they support inheritance, encapsulation, and polymorphism. They are useful for systems where data and behavior need to be packaged together, such as CAD systems, multimedia applications, robotics, and virtual reality.

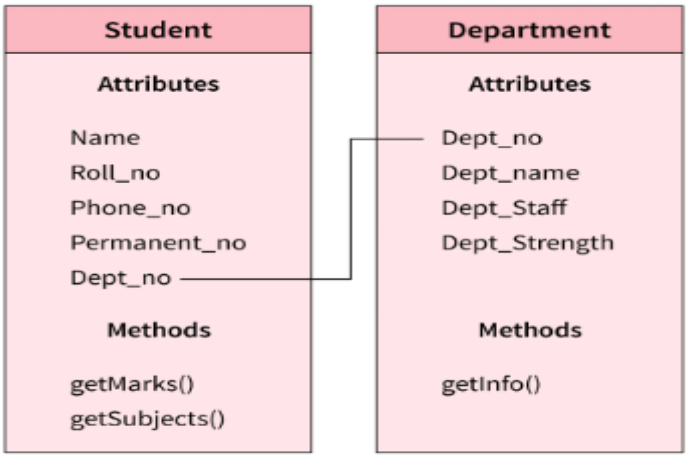

Fig 4: Object – Oriented data model