A logo of a university

Description automatically generated

**كلية العلوم**

**قــســــــــــم الانظمة الطبية الذكية**

**Lecture: (2)**

**Subject: Database Systems**

**Level: Second**

**Lecturer:** **Asst. Lecturer Qusai AL-Durrah**

**Data Model**

Is a relatively simple representation, usually graphical, of more abstraction of a more complex real- world object event, a data model represents data structures and their characteristics, relations, constraints, transformations, and, other constructs with purpose of supporting a specific problem domain.

# The Importance of Data Models

Data models can facilitate interaction among the designer, the applications programmer, and the end user, a well- developed data model can even faster improved understanding of the organization for which the database design is developed.

# Data Model Basic Building Blokes

**Entity:** is anything (a person, a please, a thing, or an event) about which data are to be collected and stored. An entity represents a particular type of object in the real-world entities are "distinguishable" that is each entity occurrence is unique and distinct. For example, a CUSTOMER entity would have many distinguishable customer occurrences such as Ahmed, Ali, etc.

**Attribute:** is a characteristic of an entity. For example, a CUSTOMER entity would bed described by a attribute such as customer last name, customer first name, its equivalent of fields in the file systems

**Relationship:** describes an association among entities for example a relationship exists between customers and agents that can be described as follows: an agent can serve many customers, and each customer may be served by one agent.

Data models use three types of relationships:

* one-to-many (1:M or 1..\*)relationships

A painter paints many different paintings, but each one of them is painted by only one painter.

* Many-to-many (M: N or \*..\*) relationship

An employee may learn many job skills many be learned by many.

* one-to-one( 1:1 OR 1..1) relationship

A retail company's management structure may require that each of its stores be managed by a single employee. In turn, each store manager, who is an employee, managed only a single store.

**Constraint**: is a restriction placed on the data. Constraints are important because they help to ensure data integrity. And it's normally expressed in the form of rules.

* An employee's salary must have values that are between 6000 and 350000
* Each class must have one and only one teacher.

# Business Rules

From a database point of view, the collection of data becomes meaningful only when it reflects properly defined business rules. It’s a brief, precise, and ambiguous description of a policy, procedure, or principle within a specific organization.

Business rules, derived from a detailed description of organizations operations, help to create and enforce actions within that organizations environment, business rules must be rendered in writing and updated to reflect any change in the organizations operational environment, and it must be easy to understand, for example:

* A customer may generate many invoices.
* An invoice is generated by only one customer.
* A training session can not be scheduled for fewer than 10 employees or for more than 30 employees.

Business rules is essential to db design for several reasons:

* They help standardize the company's view of data.
* They can be a communications tool between users and designers.
* They allow the designer to understand the nature, role, and scope of the data.
* They allow the designer to understand business processes.
* They allow the designer to develop appropriate relationship participation rules and constraints and to create an accurate data model.

# Translating Business Rules into Data Model

* A noun in business rule will translate into an entity in the model.
* Verb associating nouns will translate into a relationship among the entities.

Example: "a customer may generate many invoices"

* Customer and invoice are objects of interest for the environment and should be represented by their respective entities.
* There is a "generate" relationship between customer and invoice.

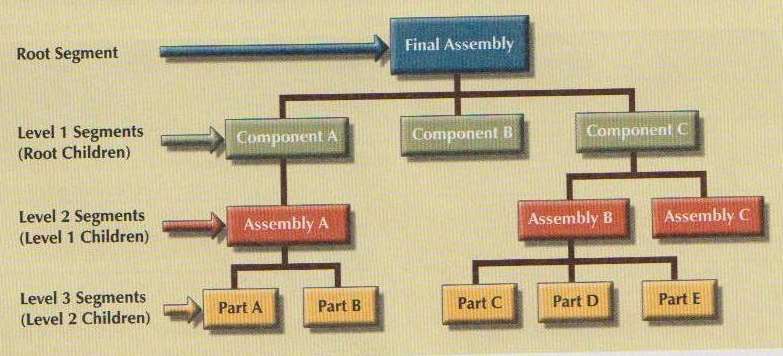
# Data Models Type

The quest for better data management has led to several different models that attempt resolve the file systems critical shortcomings.

# THE HIERARCHICAL MODEL

It was developed in 1960s to manage large amounts of data for complex manufacturing projects Apollo rocket that landed on the moon in 1969. its basic logical structure is represented by an upside-down tree- its structure contains levels, or segments. Segment is the equivalent of a file systems record type. The top layer (the root) is perceived as the parent of the segment (each parent can have many children, but each child has only one parent)

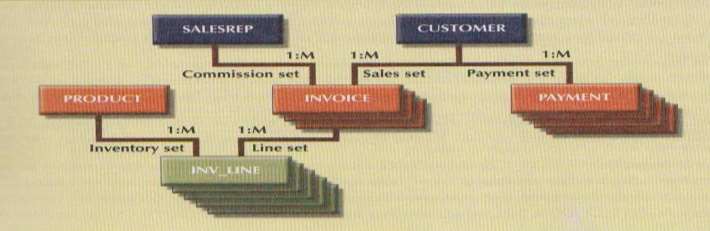
The hierarchical model had limitations: it was complex to implement, it was difficult to manage, and it lacked structural independence. Also, many common data relationships do not conform to the 1: M from, and there were no standards for how to implement the model.



# The Network Model

The network model (1970) was created to represent complex data relationships more effectively than the hierarchical model, in the network model, the user perceives the network database as a collection of records in 1:M relationship. However, unlike the hierarchical model, the network model allows a record to have more than one parent. In network db terminology, relationship is called a set. Each set is composed of at least two record types: an owner record and a member record. a set represents a 1:M relationship between the owner and the member.

Disadvantages is the lack of ad hoc query capability put heavy pressure on programmers to generate the code required to produce even the simplest report, and although the existing databases provided limited data independence, any structural change in the database still could produce have in all application programs that drew data from the database.



**The Relational Model (1970)**

The relational model consist of:

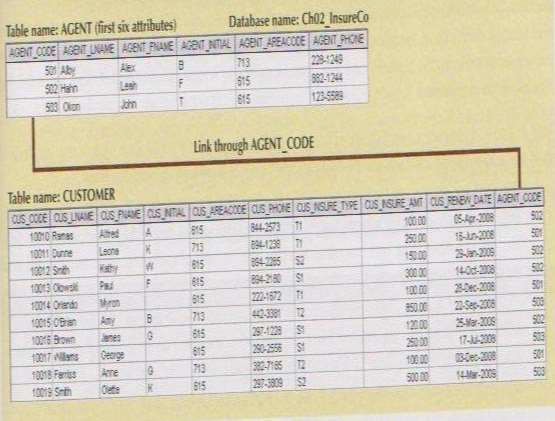
* Relation or table as a matrix.
* Tuple: raw in relation.
* Attribute: column in relation.

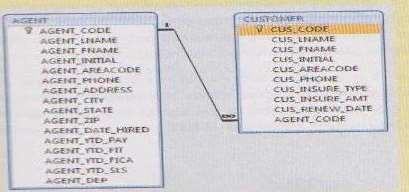
Advantages

* Tables are related to each other through the sharing of common attribute.
* The tables are independent of another; we can easily associate the data between tables.

Provides a minimum level of controlled redundancy.

* The relationship type(1:1, 1:M or M:N) is often show in relational model.
* The relational model is powerful and flexible in query language.

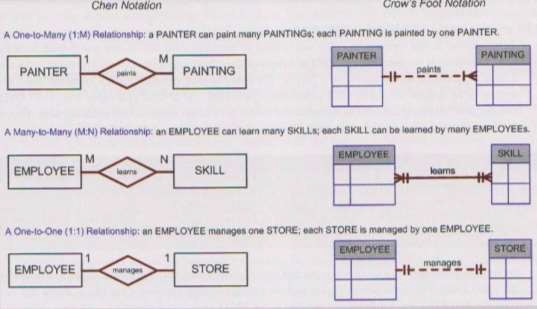




# The Entity Relationship Model

Peter Chen first introduced the ER data model in 1976: it was the graphical representation of entities and their relationships in a database structure the quickly become popular because it complemented the relational data model complex.

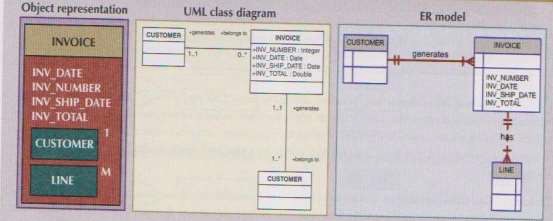
* entity is represented by rectangle
* Relationships: describe associations among data. The ER model uses the term connectivity it may be 1:1 or 1:M or M:N



# The Object-oriented (OO) Model

In which both data and their relationship are contained in single structure known as an object .The OO data model is based on the following components:

* Object: may be considered equivalent to an ER models entity. The objects semantic content is defined through several of the items in this list
* Attributes: describe the properties of an object.
* Class is a collection of similar objects with shared structure (attributes) and behavior(methods) .
* Classes are organized in a class hierarchy are upside-down tree in which each class has only one parent.
* Inheritance is the ability of an object within the class hierarchy to inherit the attributes and methods of the classes above it.



# The development of data models

