Lebanese University **Faculty of Media and Documentation** **- Section 1 -** **Computer Science** **First Year - Semester 1** **Database 1**		**الجامعة اللبنانية** **كلية الإعلام والتوثيق** **- الفرع الأول -** **Chapter 4: Enhanced ER (EER)**

1. Enhanced ER model / Extended ER model

The ER model is the abstract representation of a database structure that defines:

- Entities in a database.
- Attributes that they had.
- Relationships between them.

ER model is supported with the additional semantic concepts is called as : Extended / Enhanced Entity Relationship Model or EER.

EER diagrams are advanced database diagrams very similar to regular ER diagrams which represent requirements and complexities of complex databases.

As the complexity of data increased in the late 1980s, it became more and more difficult to use the traditional ER Model for database modelling. Hence some improvements or enhancements were made to the existing ER Model to make it able to handle the complex applications better.

2. EER model includes following concepts that are used to represent complex data relationships, inheritance, and abstraction in databases.

- **Subclass and Superclass**
- **Specialization**
- **Generalization**
- **Inheritance**
- **Aggregation**
- **Union or category**

2.1 Subclass and Superclass

- **Superclass** is a higher-**level entity** set that has common attributes: is an entity type that has a relationship with one or more subtypes. An entity cannot exist in database merely by being member of any super class.
- **Subclass** is a **lower-level entity** set that inherits attributes and relationships from its superclass but also has its own specific attributes or relationships: is a group of entities with unique attributes.

Subclass and Superclass relationship leads the concept of **Inheritance**.

The relationship between subclass and superclass is denoted with symbol .

Example: Science is a Super class which has subclasses like Physics, Chemistry, Biology.

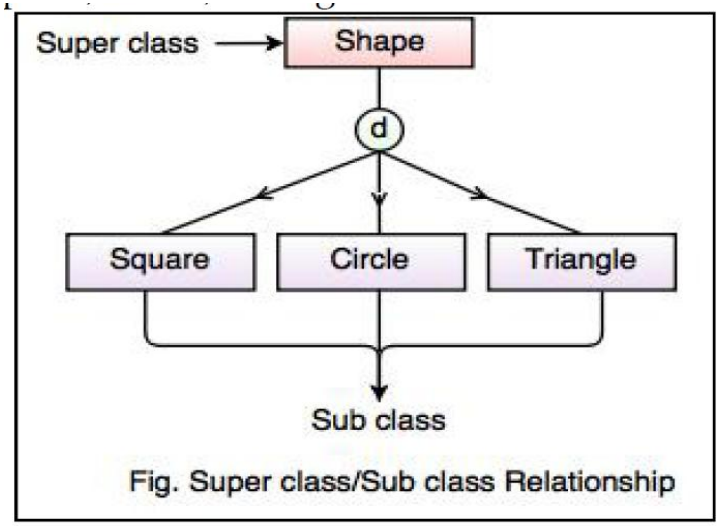


Fig. Super class/Sub class Relationship

Example: Shape super class is having sub groups as Square, Circle, Triangle.

2.2 Generalization and Specialization

Generalization (Bottom-down) and Specialization (Top-down) are common relationships added as enhancements to the classical ER model. A subclass (specialized class) inherits from a superclass (generalized class), similar to object-oriented concepts. This is best understood using IS-A relationships like “Technician IS-A Employee” or “Laptop IS-A Computer.” These are tools for organizing and simplifying data by abstracting or specifying entity relationships.

ISA (Inheritance Special Attributes)

triangles indicate the subclass relationship.

- Generalization is the process of generalizing the entities which contain the properties of all the generalized entities.
- It is a bottom-up approach, in which two lower level entities combine to form a higher level entity.
- Generalization is the reverse process of Specialization.
- It defines a general entity type from a set of specialized entity type.
- It minimizes the difference between the entities by identifying the common features.

Example: Here we have three sets of employees: Secretary, Technician and Engineer. The employee is a super-class of the rest three sets of individual sub-class is a subset of Employee set.

Generalization Constraints

- Constraints are application for generalization and specialization.

There are three categories of constraints

I. Determine which entities can be member of a given lower ‑level entity set

- **Conditional defined**
- **User Defined**

II. Determine entities belong to more than one lower level entity set

- **Disjoint**
- **Overlapping**

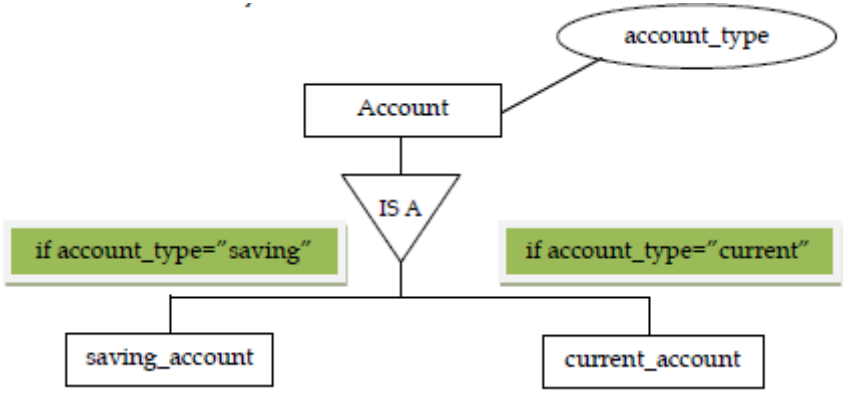
III. Determine participation/belonging of higher level entity

- **Total generalization/specialization**
- **Partial generalization /specialization**

Constraints I. Determine which entities can be member of a given lower –level entity set

i. Conditional defined

Condition is defined, based on that members of lower-level entities is evaluated or joined.

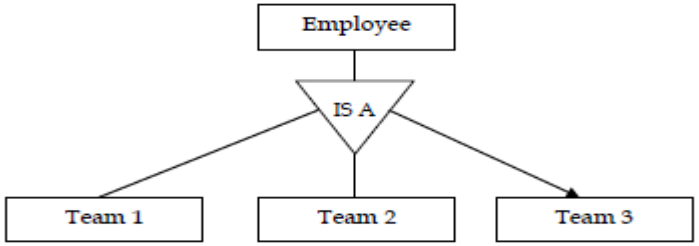


ii. User Defined

-Type of the constraints / members of entities are defined by the user

-If an employee joins an entity “employee”, Team Incharge will assign to any one of the 3 teams.

- It is defined by the user.



Constraints II. Determine entities belong to more than one lower level entity set

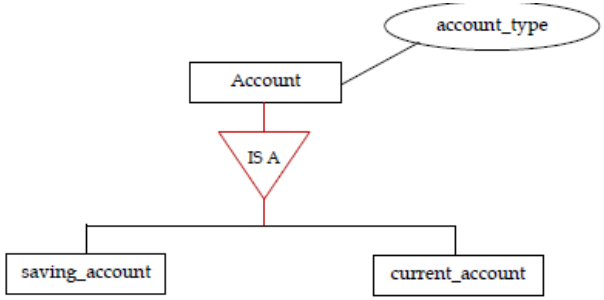
i. Disjoint

- An entity can belong to only one subclass.

- It is represented user the ER notation



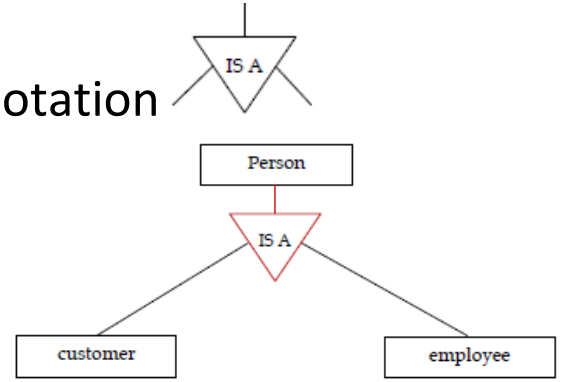
Example:



User can join only one of the lower-level entity.

ii. Overlapping

- **An entity can belong to multiple subclasses.**
- It is represented user the ER notation



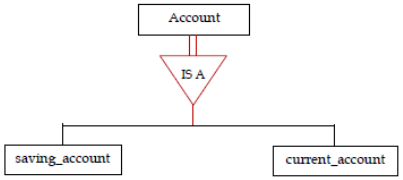
- A person can be customer & employee.
- Can participate in more than one entity of lower level.

Constraints III. Determine participation/belonging of higher level entity

i. Total generalization/specialization

- Each entity at higher level must belong to a lower-level entity set.
- Total generalization is represented using double line

Example: Each account must belong to any one of the lower level enity.



ii. Partial generalization /specialization

- Some entities of higher level may not belong to any lower level entity set.
- Partial generalization is default

2.4 Aggregation

- Aggregation is an abstraction through which relationship are treated as higherlevel entities.
- Aggregation is used to avoid redundant information.
- Aggregation is a process that represent a relationship between a whole object and its component parts.
- It abstracts a relationship between objects and viewing the relationship as an
- object.
- It is a process when two entity is treated as a single entity.

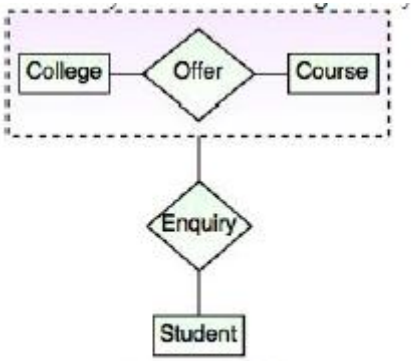


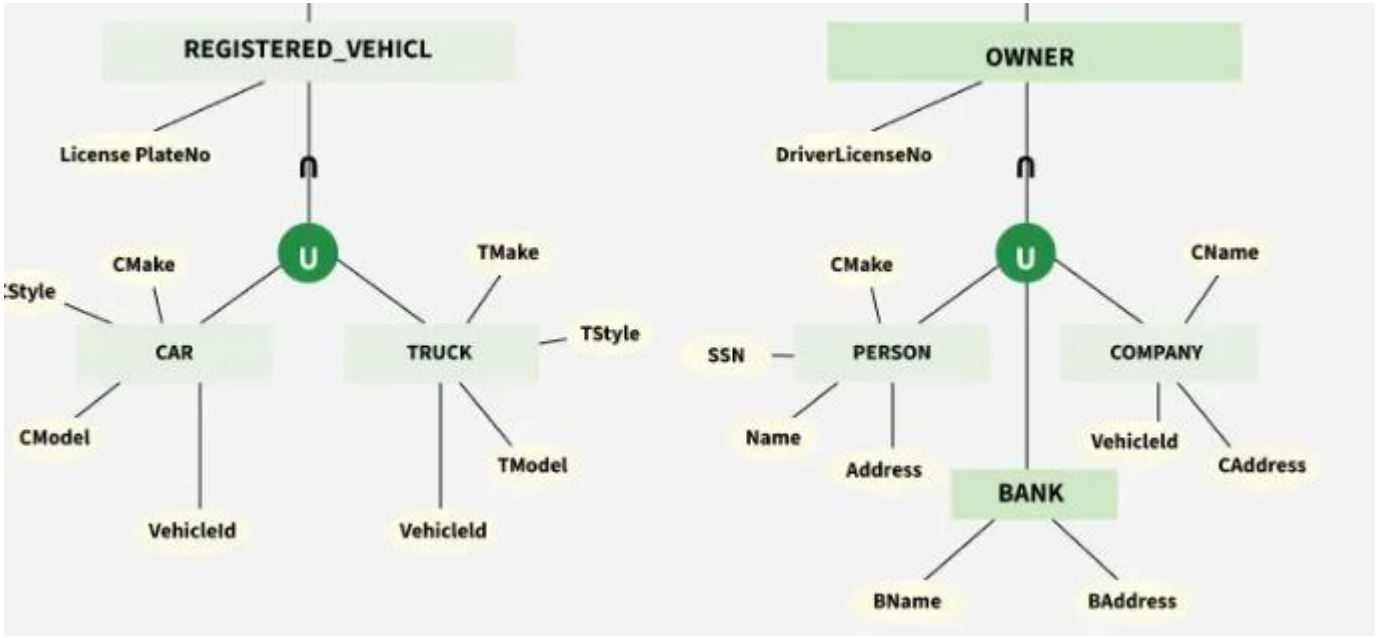
Fig. Aggregation

In the above example, the relation between College and Course is acting as an Entity in Relation with Student.

2.5 Category or Union

A Category (or Union Type) is a subclass that is derived from two or more superclasses that may not be related. It allows the model to represent an entity that can be a member of more than one entity set.

Example: Set of Library Members is UNION of Faculty, Student and Staff. A union relationship indicates either type; for example, a library member is either Faculty or Staff or Student. Below are two examples that show how UNION can be depicted in ERD - Vehicle Owner is UNION of PERSON and Company andRTO Registered Vehicle is UNION of Car and Truck.



There might be some confusion in Sub-class and UNION. Consider an example in above figure Vehicle is super-class of CAR and Truck. In the example, Vehicle is a superclass of Car and Truck, which normally implies inheritance of attributes. However, in the RTO-registered case, Car and Truck form a union without inheriting from Vehicle, each has independent attributes.