







Entity-Relationship (ER) Model

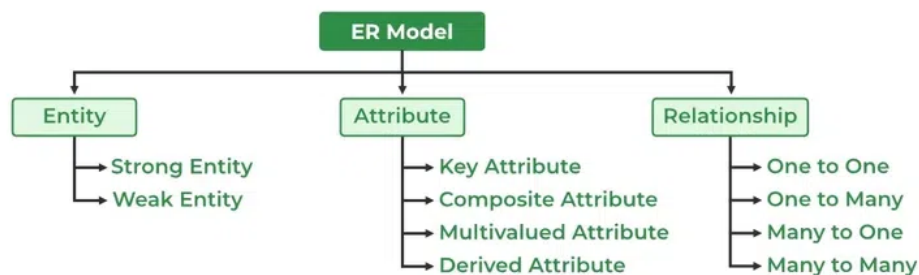
Notes by Mannan UI Haq

The Entity Relationship Diagram explains the relationship among the entities present in the database. ER models are used to model real-world objects like a person, a car, or a company and the relation between these real-world objects. In short, the ER Diagram is the structural format of the database.

Symbols Used in ER Model

Figures	Symbols	Represents
Rectangle		Entities in ER Model
Ellipse		Attributes in ER Model
Diamond		Relationships among Entities
Line		Attributes to Entities and Entity Sets with Other Relationship Types
Double Ellipse		Multi-Valued Attributes
Double Rectangle		Weak Entity

Components of ER Diagram



Entity

An Entity may be an object with a physical existence – a particular person, car, house, or employee – or it may be an object with a conceptual existence – a company, a job, or a university course.

1. Strong Entity

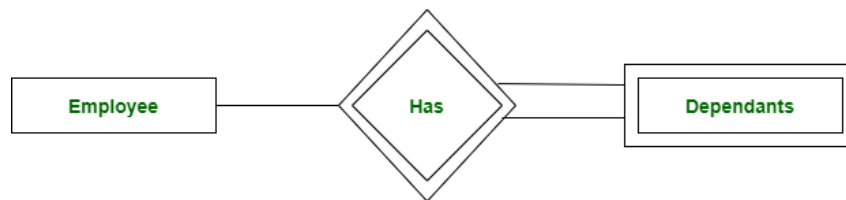
A Strong Entity is a type of entity that has a key Attribute. Strong Entity does not depend on other Entity in the Schema. It has a primary key, that helps in identifying it uniquely, and it is represented by a

rectangle.

2. Weak Entity

Some entity type exists for which key attributes can't be defined. These are called Weak Entity types. A weak entity type is represented by a Double Rectangle. The participation of weak entity types is always total. The relationship between the weak entity type and its identifying strong entity type is called identifying relationship and it is represented by a double diamond.

For Example, A company may store the information of dependents (Parents, Children, Spouse) of an Employee. But the dependents can't exist without the employee. So Dependent will be a **Weak Entity Type** and Employee will be Identifying Entity type for Dependent, which means it is **Strong Entity Type**.



Attributes

Attributes are the properties that define the entity type. For example, Roll_No, Name, DOB, Age, Address, and Mobile_No are the attributes that define entity type Student. In ER diagram, the attribute is represented by an oval.



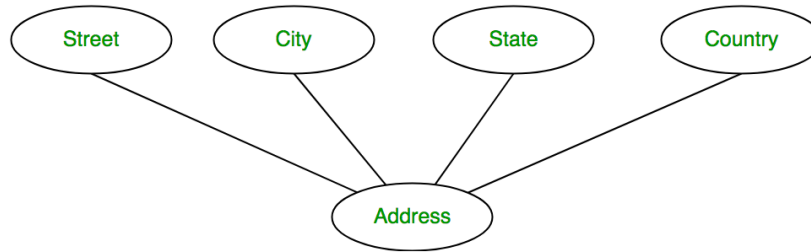
1. Key Attribute

The attribute which **uniquely identifies each entity** in the entity set is called the key attribute. In ER diagram, the key attribute is represented by an oval with underlying lines.



2. Composite Attribute

An attribute **composed of many other attributes** is called a composite attribute. In ER diagram, the composite attribute is represented by an oval comprising of ovals.



3. Multivalued Attribute

An attribute consisting of more than one value for a given entity. In ER diagram, a multivalued attribute is represented by a double oval.



4. Derived Attribute

An attribute that can be derived from other attributes of the entity type is known as a derived attribute. e.g.; Age (can be derived from DOB). In ER diagram, the derived attribute is represented by a dashed oval.

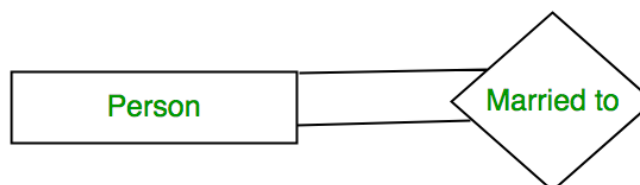


Relationship Type and Relationship Set

A Relationship Type represents the association between entity types. In ER diagram, the relationship type is represented by a diamond and connecting the entities with lines.



1. Unary Relationship: When there is only ONE entity set participating in a relation, the relationship is called a unary relationship. For example, one person is married to only one person.



2. Binary Relationship: When there are TWO entities set participating in a relationship, the relationship is called a binary relationship. For example, a Student is enrolled in a Course.



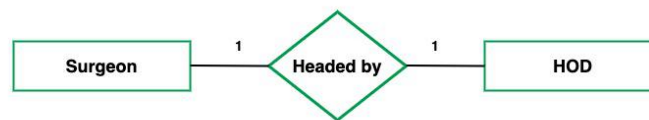
3. Ternary Relationship: When there are n entities set participating in a relation, the relationship is called an n-ary relationship.

Cardinality

The number of times an entity of an entity set participates in a relationship set is known as cardinality. Cardinality can be of different types:

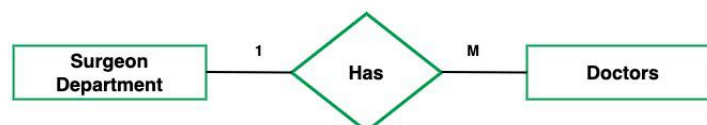
1. One-to-One: When each entity in each entity set can take part only once in the relationship, the cardinality is one-to-one. Let us assume that a male can marry one female and a female can marry one male. So the relationship will be one-to-one.

The number of tables that can be used is 2.



2. One-to-Many: In one-to-many mapping as well where each entity can be related to more than one entity. Let us assume that one surgeon department can accommodate many doctors. So the Cardinality will be 1 to M. It means one department has many Doctors.

The total number of tables that can be used is 3.



3. Many-to-One:



4. Many-to-Many: When entities in all entity sets can take part more than once in the relationship cardinality is many to many. Let us assume that a student can take more than one course and one course can be taken by many students. So the relationship will be many to many.

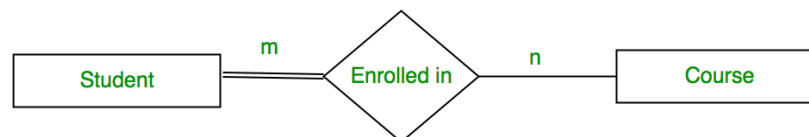
The total number of tables that can be used in this is 3.



Participation Constraint

1. Total Participation – Each entity in the entity set must participate in the relationship. If each student must enroll in a course, the participation of students will be total. Total participation is shown by a double line in the ER diagram.

2. Partial Participation – The entity in the entity set may or may NOT participate in the relationship. If some courses are not enrolled by any of the students, the participation in the course will be partial.



Note: The upper notation we used for cardinality is also called **Chen notation**.

Cross-Foot Notation for Cardinality

There are two marks that indicate Cross-Foot. The first mark, closest to the end of the line, represents the maximum number of times an instance of an entity can be associated with other instances

- **One time** – A short, perpendicular line

- **Many** – A three-pronged line

The second mark, behind the first, represents the minimum number of times an instance of an entity can be associated with other instances. The minimum can only be zero or one, and they are referred to as '**optional**' or '**mandatory**', respectively.

- **Optional** – An open circle
- **Mandatory** – A short, perpendicular line

