

ENTITY - RELATIONSHIP MODEL

An entity-relationship model describes data in terms of the following:

1. Entities
2. Relationship between entities
3. Attributes of entities.
4. Identifiers

ERM is graphically represented using an entity-relationship diagram (E-R Diagram).

Entity relationship diagram

Entity: An entity is an object that exists and which is distinguishable from other objects. An entity can be a person, a place, an object, an event, or a concept which an organization wishes to maintain data. The following are some examples of entities:

Person: STUDENT, EMPLOYEE, CLIENT

Object: COUCH, AIRPLANE, MACHINE

Place: CITY, ROOM, WAREHOUSE

Concept: PROJECT, ACCOUNT, COURSE

An **entity type** defines a collection of entities that have same attributes. An **entity instance** is a single item in this collection. An **entity set** is a set of entity instances. For example, STUDENT is an **entity type**; a student with ID number ISI070 is an **entity instance**; and a collection of all students is an **entity set**.

In E-R diagram, an entity name should be a concise **singular noun** that captures the unique characteristics of the entity type. An E-R diagram depicts an entity type using a **rectangle** with the name of the entity inside, as shown below.

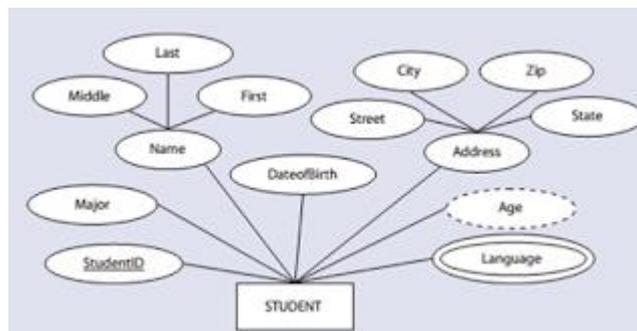


Attribute: An attribute is a characteristic of an entity type. Some attributes of common entity types include the following:

STUDENT = { Student ID, Name, Address, Phone, Email, DOB }

ACCOUNT = { Account Number, Account Type, Date Opened, Balance }

E-R diagrams depict an attribute inside an **ellipse** and connect the ellipse with a line to the associated entity type. The diagram below indicates some of the possible attributes in an E-R diagram for the entity STUDENT.

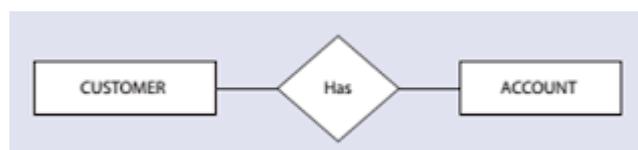


Types of attributes

1. **Simple** or an **atomic attribute**, such as City or State, cannot be further divided into smaller components.
2. **Composite attribute** can be divided into smaller subparts in which each subpart represents an **independent attribute**. **Name** and **Address** are the only **composite attributes** in the diagram above.
3. **Single-valued attribute**: most attributes have a single value. For example, a student has only one date of birth and one identification number.
4. **Multi-valued attribute**: have more than one value for an entity instance. For example, student attribute *language* above, which stores the name of languages a student can speak, is a *multi-valued attribute*.
5. **Stored attribute** is an attribute whose value cannot be derived from the values of other attributes.
6. **Derived attribute**: the value of a derived attribute can be determined by analyzing other attributes. For example in the figure above, *Age* is a derived attribute because its value can be derived from the **current date** and the attribute *DateofBirth*.
7. **A key attribute** (or identifier) is a single attribute or a combination of attributes that uniquely identify an individual instance of an entity type. No two instances within an entity set can have the same key attribute value. StudentID is the key attribute in the picture above.

Relationship

Entities in an organization do not exist in isolation but are related to each other. Students take courses and each STUDENT entity is related to the COURSE entity. Faculty members teach courses and each FACULTY entity is also related to the COURSE entity. Consequently, the STUDENT entity is related to the FACULTY entity through the COURSE entity. In an E-R diagram, we represent relationship types with *diamond-shaped boxes* connected by straight lines to the rectangles that represent participating entity types. A *relationship type* is a given name that is displayed in this diamond-shaped box and typically takes the form of a present tense verb or verb phrase that describes the relationship. A relationship is an association among several entities. The figure below is an E-R diagram that shows a relationship between entities ACCOUNT and CUSTOMER.



Degree of a relationship

The number of entity sets that participate in a relationship is called the degree of relationship. The 3 most common degrees of relationship are unary (degree 1), binary (degree 2) and ternary (degree 3).

Unary relationship: A unary relationship exists when there is an association between two instances of the same entity type. For example, in an organization, an employee will manage one or more employee.

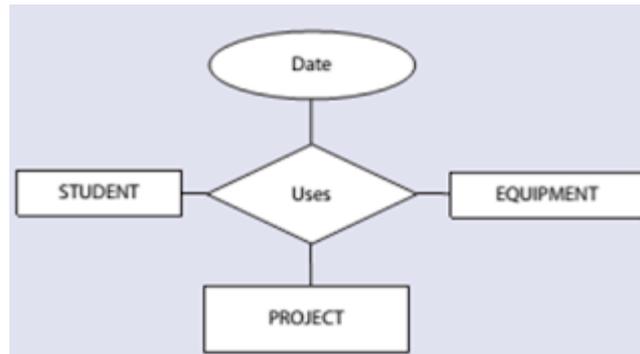


Binary relationship: A binary relationship exists when there is an association between two instances of two different entity types. For example, in a secondary school, a binary relationship

exists between a Subject (SUBJECT entity) and a Teacher (TEACHER entity); a teacher teaches (a) subject.



Ternary Relationship: A ternary relationship exists when there is an association between three instances of three different entity types. For example, consider a student using certain equipment for a project. In this case, the STUDENT, PROJECT, and EQUIPMENT entity types relate to each other with ternary relationships: a student checks out equipment for a project.



Cardinality of a relationship

The cardinality is the number of occurrences in one entity which are associated (or linked) to the number of occurrences in another entity. For example, if the entity types A and B are connected by a relationship, then the *maximum cardinality* represents the maximum number of instances of entity B that can be associated with any instance of entity A. There are three types of relationship:

1. **One-to-One Relationship:** In a one-to-one relationship, at most one instance of entity B can be associated with a given instance of entity A and vice versa. For example an employee is assigned a parking space or an employee manages the department.



2. **One-to-Many (or many-to-one) Relationship:** In a one-to-many relationship, many instances of entity B can be associated with a given instance of entity A. However, only one instance of entity A can be associated with a given instance of entity B. For example, many employee works in a department or a school has many employees.



3. **Many-to-Many Relationship:** In a many-to-many relationship, many instances of entity A can be associated with a given instance of entity B, and, likewise, many instances of entity B can be associated with a given instance of entity A. For example, many employees work on many projects.



TOPIC 2: RELATIONAL MODEL