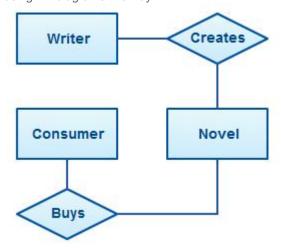# Entity Relationship Diagram

An Entity Relationship Diagram (ERD) is a visual representation of different data using conventions that describe how these data are related to each other. For example, the elements writer, novel, and consumer may be described using ER diagrams this way:

ER diagram with basic objects

In the diagram, the elements inside rectangles are called entities while the items inside diamonds denote the relationships between entities. This ER diagram tutorial for beginners covers most things related to ER diagram, for quick navigation use the links below.

- ER Diagram Usage
- History of ER diagrams
- ER Diagrams Symbols and Notations
- How to Draw ER Diagrams
- ER Diagram Templates
- Benefits of ER Diagrams

ER Diagrams Usage

While able to describe just about any system, ER diagrams are most often associated with complex databases that are used in software engineering and IT networks. In particular, ER diagrams are frequently used during the design stage of a development process in order to identify different system elements and their relationships with each other. For example, an inventory software used in a retail shop will have a database that monitors elements such as purchases, item, item type, item source and item price. Rendering this information through an ER diagram would be something like this:
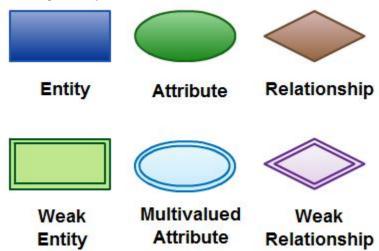
ER diagram example with entity having attributes

In the diagram, the information inside the oval shapes are attributes of a particular entity.

History of ER Diagrams

ER diagrams are visual tools that are used in the Entity-Relationship model initially proposed by Peter Chen in 1976 to create a uniform convention that considers both relational database and network views. Chen envisioned the ER model as a conceptual modeling approach that views real world data as systems of entities and relationships. Entities are data objects that maintain different relationships with each other. Additionally, entities are also described further using attributes.

Since 1976, the ER model has been expanded and is sometimes used in business management, product development, and strategy formulations. However, database design remains its primary application.

ER Diagram Symbols and Notations

Elements in ER diagrams

There are three basic elements in an ER Diagram: entity, attribute, relationship. There are more elements which are based on the main elements. They are weak entity, multivalued attribute, derived attribute, weak relationship and recursive relationship. Cardinality and ordinality are two other notations used in ER diagrams to further define relationships.

Entity

An entity can be a person, place, event, or object that is relevant to a given system. For example, a school system may include students, teachers, major courses, subjects, fees, and other items. Entities are represented in ER diagrams by a rectangle and named using singular nouns.
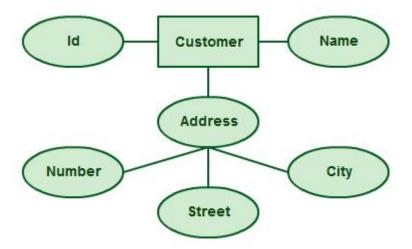
Weak Entity

A weak entity is an entity that depends on the existence of another entity. In more technical terms it can defined as an entity that cannot be identified by its own attributes. It uses a foreign key combined with its attributed to form the primary key. An entity like order item is a good example for this. The order item will be meaningless without an order so it depends on the existence of order.

Weak Entity Example in ER diagrams

Attribute

An attribute is a property, trait, or characteristic of an entity, relationship, or another attribute. For example, the attribute Inventory Item Name is an attribute of the entity Inventory Item. An entity can have as many attributes as necessary. Meanwhile, attributes can also have their own specific attributes. For example, the attribute "customer address" can have the attributes number, street, city, and state. These are called composite attributes. Note that some top level ER diagrams do not show attributes for the sake of simplicity. In those that do, however, attributes are represented by oval shapes.

Attributes in ER diagrams, note that an attribute can have its own attributes (composite attribute)
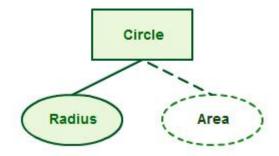
Multivalued Attribute

If an attribute can have more than one value it is called an multivalued attribute. It is important to note that this is different to an attribute having its own attributes. For example a teacher entity can have multiple subject values.

Example of a multivalued attribute

Derived Attribute

An attribute based on another attribute. This is found rarely in ER diagrams. For example for a circle the area can be derived from the radius.

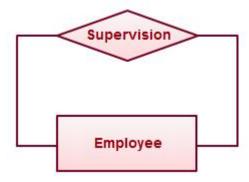*Derived Attribute in ER diagrams*

Relationship

A relationship describes how entities interact. For example, the entity "carpenter" may be related to the entity "table" by the relationship "builds" or "makes". Relationships are represented by diamond shapes and are labeled using verbs.

Using Relationships in Entity Relationship Diagrams

Recursive Relationship

If the same entity participates more than once in a relationship it is known as a recursive relationship. In the below example an employee can be a supervisor and be supervised, so there is a recursive relationship.

Example of a recursive relationship in ER diagrams

Cardinality and Ordinality

These two further defines relationships between entities by placing the relationship in the context of numbers. In an email system, for example, one account can have multiple contacts. The relationship in this case follows a "one to many" model. There are number of notations used to present cardinality in ER diagrams. Chen, UML, Crow's foot, Bachman are some of the popular notations. Creately supports Chen, UML and Crow's foot notations.The following example uses UML to show cardinality.

Cardinality in ER diagrams using UML notation

Tips on How to Draw ER Diagrams

Because ER diagrams are simple enough to understand, just about anyone can create them. However, two different ER diagrams describing the same system may still be radically different in terms of their simplicity, completeness, and efficiency at communicating the system. In other words, there are good ER diagrams and there are poor ones.
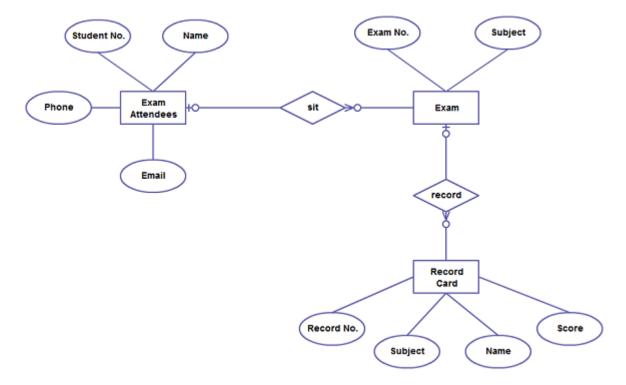
Because this ER tutorial focuses on beginners below are some tips that will help you build effective ER diagrams:

- Identify all the relevant entities in a given system and determine the relationships among these entities.
- An entity should appear only once in a particular diagram.
- Provide a precise and appropriate name for each entity, attribute, and relationship in the diagram. Terms that are simple and familiar always beats vague, technical-sounding words. In naming entities, remember to use singular nouns. However, adjectives may be used to distinguish entities belonging to the same class (part-time employee and full time employee, for example). Meanwhile attribute names must be meaningful, unique, system-independent, and easily understandable.
- Remove vague, redundant or unnecessary relationships between entities.
- Never connect a relationship to another relationship.
- Make effective use of colors. You can use colors to classify similar entities or to highlight key areas in your diagrams.

You can draw entity relationship diagrams manually, especially when you are just informally showing simple systems to your peers. However, for more complex systems and for external audiences, you need diagramming software such as Creately's to craft visually engaging and precise ER diagrams. The ER diagram software offered by Creately as an online service is pretty easy to use and is a lot more affordable than purchasing licensed software. It is also perfectly suited for development teams because of its strong support for collaboration.
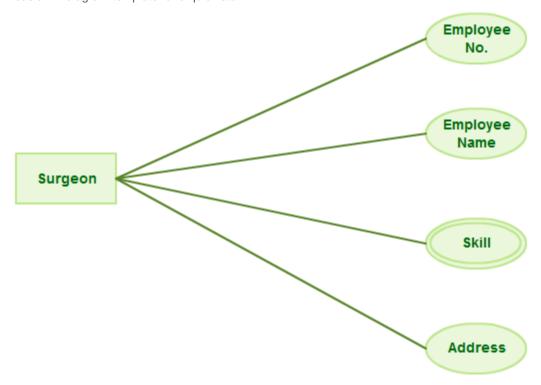
ER Diagram Templates

Below are some ER diagram templates so you can get started quickly. Clicking on the image and in the new page that opens click the "Use as Template" button.

ER Diagram Template of exam database (Click on the image to use as template)

A basic ER diagram template for a quick start

Basic ER Diagram template (Click to use as template)

Benefits of ER diagrams

ER diagrams constitute a very useful framework for creating and manipulating databases. First, ER diagrams are easy to understand and do not require a person to undergo extensive training to be able to work with it efficiently and accurately. This means that designers can use ER diagrams to easily communicate with developers, customers, and end users, regardless of their IT proficiency. Second, ER diagrams are readily translatable into relational tables which can be used to quickly build databases. In addition, ER diagrams can directly be used by database developers as the blueprint for implementing data in specific software applications. Lastly, ER diagrams may be applied in other contexts such as describing the different relationships and operations within an organization.