

# Degrees of Relationship Types

In this lesson, we will learn about the different degrees of relationship types.

## WE'LL COVER THE FOLLOWING



- Degrees of relationship types
  - The Unary (recursive) relationship type
  - The Binary relationship type
  - The Ternary relationship type

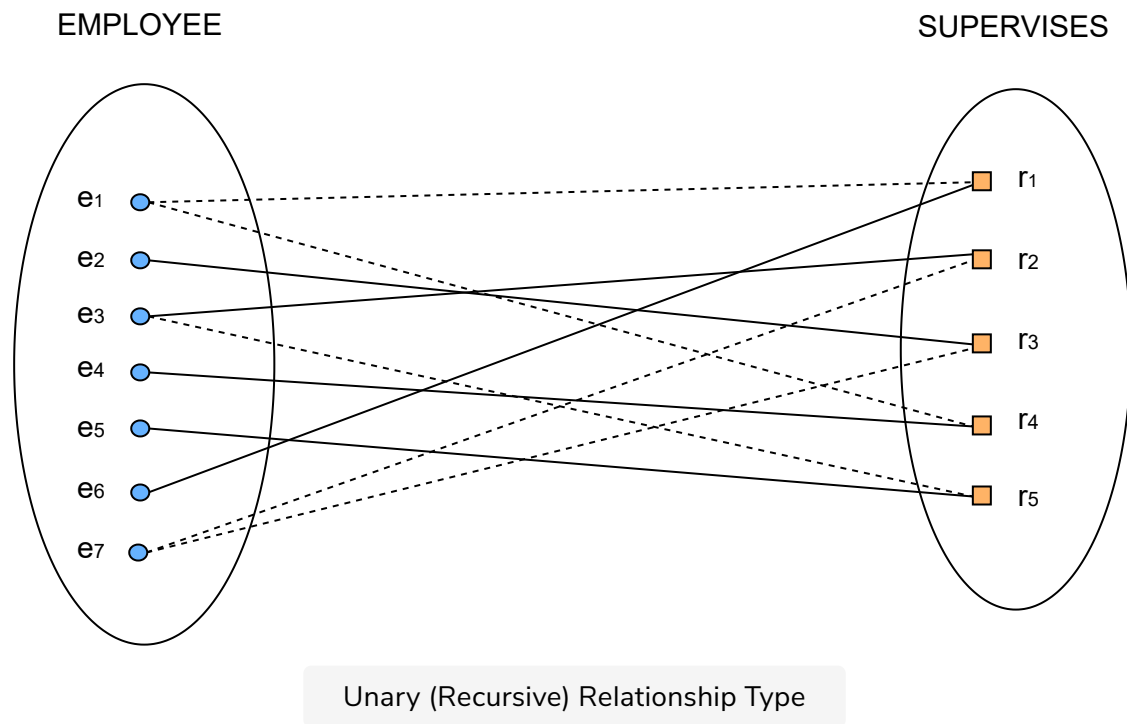
## Degrees of relationship types #

Consider the scenario where an employee works on multiple projects, so for a single EMPLOYEE entity the number of PROJECT entities he/she is associated with is multiple. Similarly, a PROJECT entity can have multiple EMPLOYEE entities that work on it. We represent this situation in our ER model through the degrees of a relationship.

The degree of a relationship type is the **number of participating entities types**. We will focus on mainly three types of degrees:

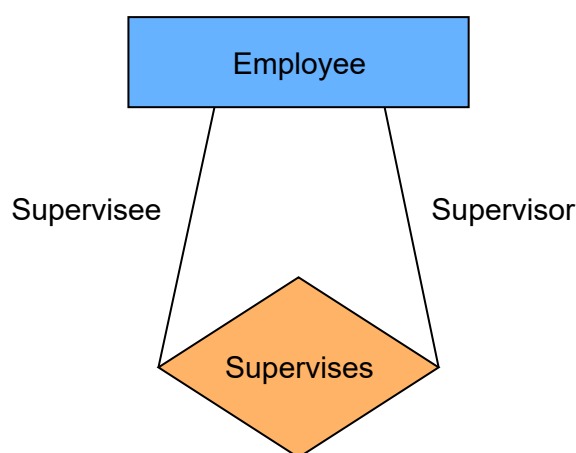
### The Unary (recursive) relationship type #

The unary relationship type involves only one entity type. However, the same entity type participates in the relationship type in different roles. For example, The SUPERVISES relationship type relates an employee to a supervisor, where both employee and supervisor entities are members of the same EMPLOYEE entity set. Hence, the EMPLOYEE entity type participates twice in SUPERVISION: once in the role of supervisor, and once in the role of the supervisee. This concept is illustrated below:



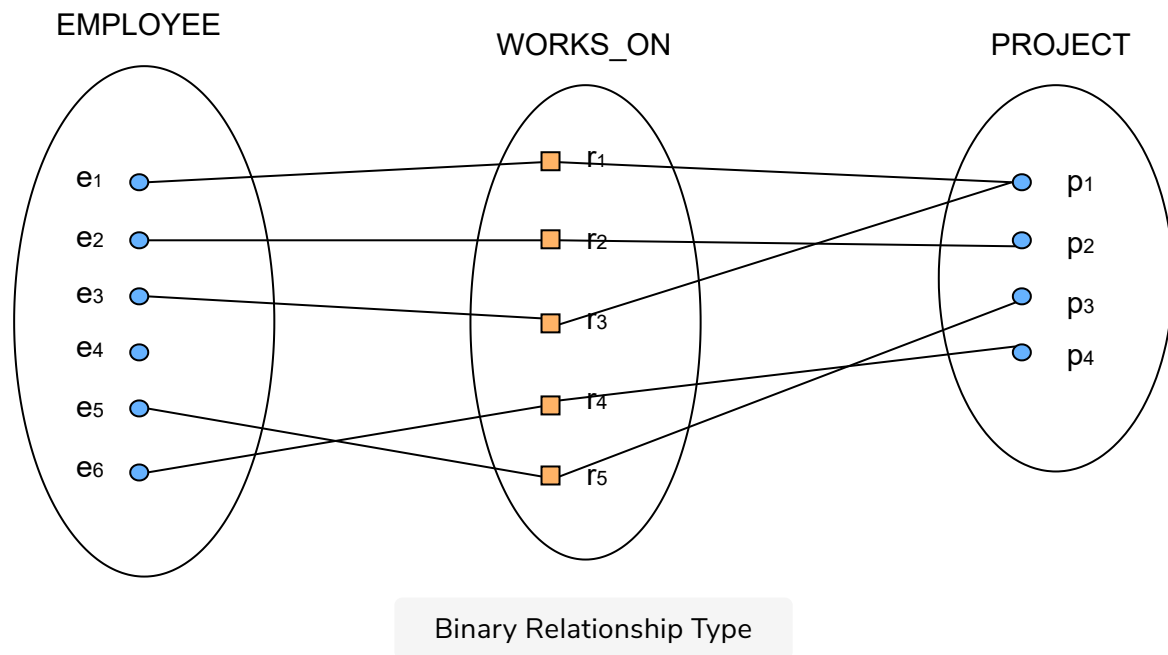
Each **relationship instance**  $r_i$  in SUPERVISES associates two different employee entities  $e_j$  and  $e_k$ , one of which plays the role of supervisor and the other the role of the supervisee. In the figure above, the dotted lines represent the supervisor role, and the solid lines represent the supervisee role; hence,  $e_1$  supervises  $e_4$  and  $e_6$ ,  $e_7$  supervises  $e_3$  and  $e_2$ .

In the case of ER diagram we represent unary relationship types as:



## The Binary relationship type #

This relationship type has two entity types linked together. This is the most common relationship type. For example, consider a relationship type WORKS\_ON between the two entity types EMPLOYEE and PROJECT, which associates each employee with the project he/she is working on. This relationship is expanded upon in the diagram below:



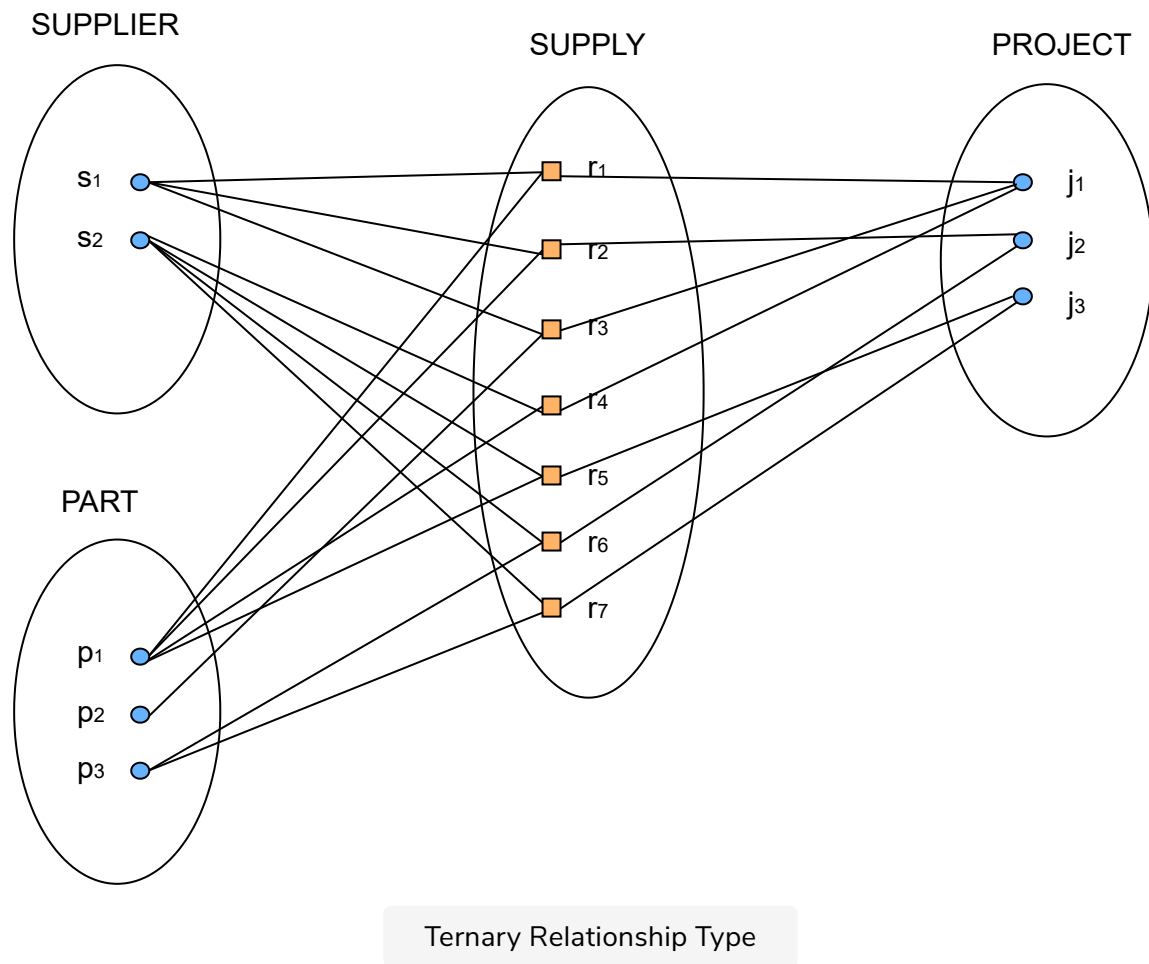
In the figure above each relationship instance  $r_i$  is shown connected to the **EMPLOYEE** and **PROJECT** entities that participate in  $r_i$ . In the mini-world represented by this figure, the employees  $e_1$  and  $e_3$  work on project  $p_1$ , the employee  $e_2$  works on project  $p_2$ , and so on.

In the case of ER diagram we represent the binary relationship type as:



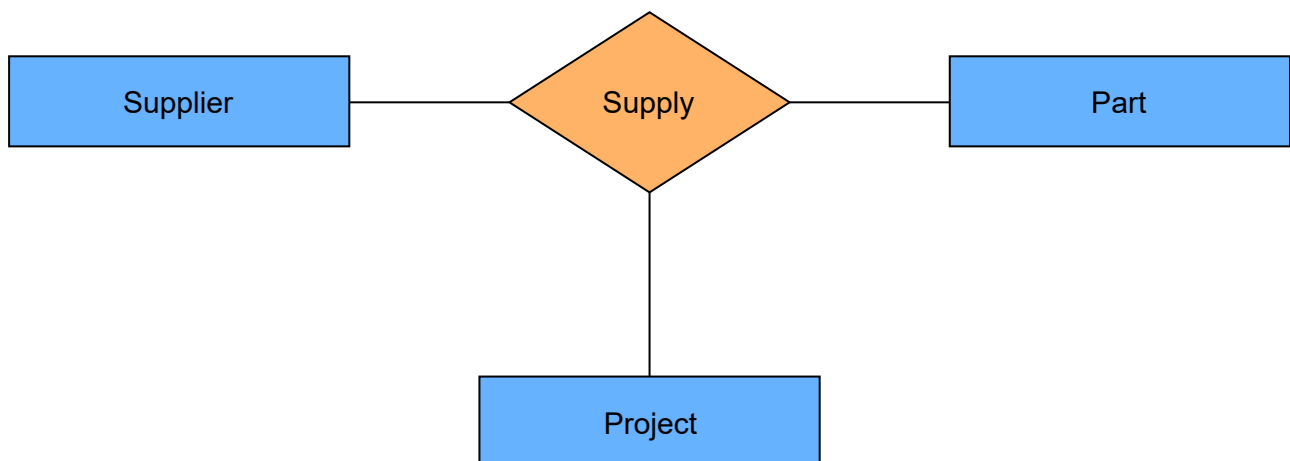
## The Ternary relationship type #

If there are three entity types linked together, the relationship is called a ternary relationship. An example of a ternary relationship is **SUPPLY**, shown in the figure below, where each relationship instance  $r_i$  associates three entities—a supplier  $s$ , a part  $p$ , and a project  $j$ .



In the above diagram, we observe that a supplier  $s_i$  supplies part  $p_j$  to a project  $j_k$ . So we see that the supplier  $s_1$  supplies part  $p_1$  to both projects  $j_1$  and  $j_2$ , while also supplying  $p_2$  to project  $j_1$ . Similarly, other relationships can be determined as well.

In the case of an ER diagram we represent ternary relationship type as:



In the next lesson, we will discuss the constraints on binary relationship types.

