

Associative Entities

by Sophia



WHAT'S COVERED

Associative entities can be used to model complex relationships in the database, maintain data integrity, and enhance data management and retrieval efficiency. In this lesson, you will explore the use of associative entities in an entity-relationship diagram (ERD), in two parts. Specifically, this lesson will cover:

1. Introduction

2. Movie Ratings Example

1. Introduction

ERDs represent many-to-many relationships using an **associative entity**, also called a junction or bridge entity, that bridges the gap between related entities. By establishing one-to-many relationships between related entities, the associative entity effectively breaks down the many-to-many relationship into two one-to-many relationships.

Due to the many-to-many nature of the relationship, associative entities are particularly useful in situations where a direct one-to-many relationship is neither appropriate nor feasible. An associative table helps avoid redundancy in the strong entities with which it is associated, which in turn contributes to database normalization.

An associative entity has multiple one-to-many relationships with other entities and is on the "many" side of each of those relationships. In other words, the attributes involved in each of those relationships are foreign keys in the associative entity.

Rather than having a primary key (that is, a single attribute that uniquely identifies each record), they typically have a composite key. A **composite key** is a combination of two or more foreign keys, the combination of which is unique for each record.

⇒ **EXAMPLE** A university database contains two entities: Students and Courses. A single student may enroll in more than one course, and a single course may have multiple enrolled students. Instead of directly connecting Students and Courses in a many-to-many relationship, an associative entity named Enrollment

would be introduced instead. As part of the Enrollment entity, foreign keys would be associated with both Students and Courses as well as other attributes like Enrollment Date or Grade.



THINK ABOUT IT

The use of a composite key in an associative entity is not mandatory; a database designer might choose to add a new attribute such as ID# to an associative entry specifically to be a unique value in each record. However, using a composite key is more efficient because it does not increase the entity size. In part, this decision depends on whether the table will be referenced from other tables. If it is, creating a new primary key makes more sense. Otherwise, the table that refers to the associative entity would need to link all of the foreign keys, which can be redundant.



TERMS TO KNOW

Associative Entity

An entity that exists to convert a many-to-many relationship between two other entities into two one-to-many relationships in order to normalize the database.

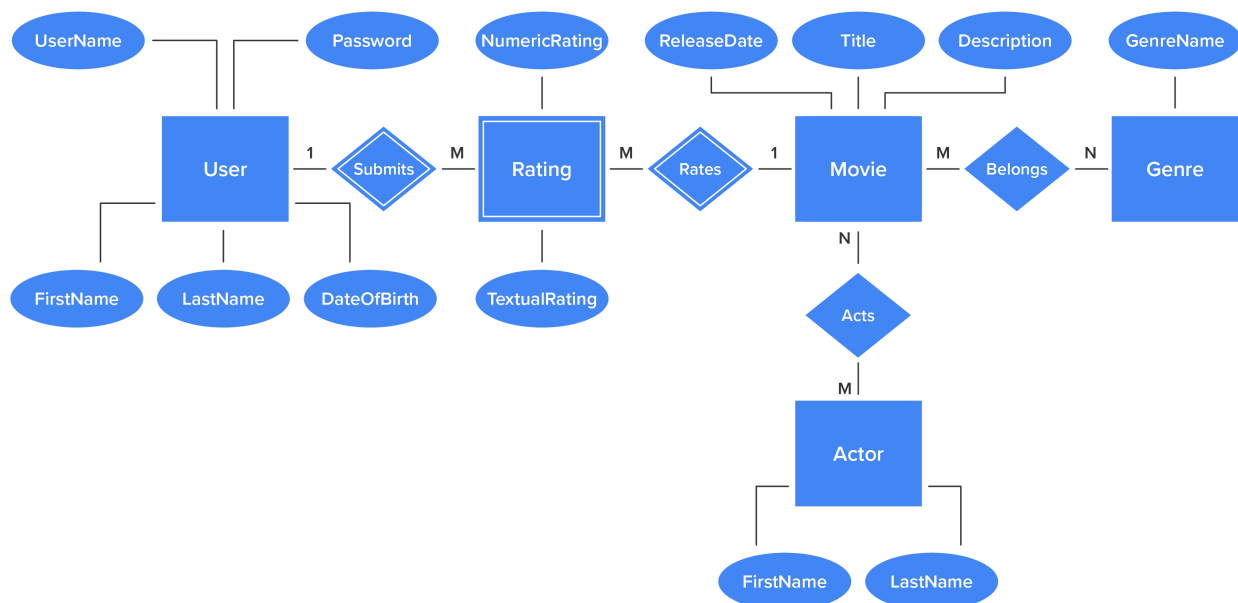
Composite Key

A key that consists of two or more attributes, the combination of which uniquely identifies a record.

2. Movie Ratings Example

At the conceptual level, which is where you currently are with your movie ratings database model, it is valid to have many-to-many relationships (M:N), and they are frequently used in the ER modeling process. During the implementation into a logical model, though, you are required to create your associative entities.

⇒ **EXAMPLE** For example, if you look at the Movie and Actor relationship in your movie ratings database, you see that it is currently a many-to-many (M:N) relationship:



⇒ **EXAMPLE** You could create a new associative entity called **Role** to indicate the role that an actor plays in a movie. It would have **MovieID** and **ActorID** foreign keys in it, and together these would be the composite key. It could also have one or more additional attributes that exist only in the **Role** table, such as **RoleName**.



SUMMARY

In this lesson **introduction**, you learned how an associative entity brings value to a database by converting a many-to-many relationship between two other entities into two one-to-many relationships. Associative entities can help you model complex relationships, maintain data integrity, and make data retrieval more efficient.

You learned that the difference between an associative entity and a strong entity is that an associative entry does not have a primary key; instead, it has multiple one-to-many relationships with other entities, and the foreign keys involved in those relationships work together to form a composite key in which the combination is unique for each record.

In applying this knowledge to the **movie ratings example**, you learned that the **Rating** entity is an associative entry, and that in order to normalize the database, we will need to create associative entities that connect the **Movie** entity to the **Actor** entity, and also that connect the **Movie** entity to the **Genre** entity.

Source: THIS TUTORIAL WAS AUTHORED BY DR. VINCENT TRAN, PHD (2020) AND Faithe Wempen (2024) FOR SOPHIA LEARNING. PLEASE SEE OUR TERMS OF USE.



TERMS TO KNOW

Associative Entities

An entity that exists to convert a many-to-many relationship between two other entities into two one-to-many relationships in order to normalize the database.

Composite Key

A key that consists of two or more attributes, the combination of which uniquely identifies a record.