# Basic Concept and Definitions in Relationships

A relationship type is a meaningful association between (or among) entity types. The phrase meaningful association implies that the relationship allows us to answer questions that could not be answered given only the entity types. A relationship type is denoted by a line labeled with the name of the relationships, as in the example shown in Figure 2-10a, or with two names, as in Figure 2-1.We suggest you use a short, descriptive verb phrase that is meaningful to the user in naming the relationship.(We say more about naming and defining relationships later in this section.)

A relationship instance is an association between (or among) entity instances, where each relationship instance associates exactly one entity instance from each participating entity type (Elmasri and Navathe，1994）.For example ,in figure 2-10b,each of the 10 lines in the figure represents a relationship instance between one employee and one course ,indicating that the employee has completed that course.for example the line between Employee Ritchie to Course Perl is one relationship instance.

**ATTRIBUTES ON RELATIONSHIPS** It is probably obvious to you that entities have attributes,but attributes may be associated with a many-to-many(or one-to-one)relationship,too.For example,suppose the organization wishes to record the date(month and year)when an employee completes each course. This attribute is named Date Completed. For some sample data, see Table 2-1.

Where should the attribute Date Completed be placed on the E-R diagram? Referring to Figure 2-10a,you will notice that Date Completed has not been associated with either the EMPLOYEE or COURSE entity.That is because Date Completed is a property of the relationship Completes,rather than a property of either entity.In other words,for each instance of the relationship Completes,there is a value for Date relationship line.Other attributes might be added to this relationship if appropriate,such as Course Grade,Instructor,and Room Location.

It is interesting to note that an attribute cannot be associated with a one-to-many relationship,such as Carries in Figure 2-5.For example,consider Dependent Date,similar to Date Completed above ,for when the DEPENDENT begins to be carried by the EMPLOYEE.Because each DEPENDENT is associated with only one EMPLOYEE,such a date is unambiguously a characteristic of the DEPENDENT(i.e,for a given DEPENDENT,Dependent Date cannot vary by EMPLOYEE) .So,if you ever have the urge to associate an attribute with a one-to-many relationship,”stept away from the relationship!”

**ASSOCITIVE ENTITIES** The presence of one or more attributes on a relationship suggests to the designer that the relationship should perhaps instead be represented as an entity type. To emphasize this point ,most E-R drawing tools require that such attributes be placed in an entity type. An associative entity is an entity type that associates the instances of one or more entity types and contains attributes that are peculiar to the relationship between those entity instances. The associative entity CERTIFICATE is represented with the rectangle with rounded corners, as shown in Figure 2-11b. Most E-R drawing tools do not have a special symbol for an associative entity. Associative entities are sometimes referred to as gerunds, because the relationship name (a verb) is usually converted to an entity name that is a noun. Note in Figure 2-11b that there are no relationship names on the lines between an associative entity and a strong entity. This is because the associative entity represents the relationship. Figure 2-11c shows how associative entities are drawn using Microsoft Visio, which is representative of how you would draw an associative entity with most E-R diagramming tools. In Visio, the relationship lines are dashed because CERTIFICATE does not include the identifiers of the related entities in its identifier. (Certificate number is sufficient.)

How do you know whether to convert a relationship to an associative entity type? Following are four conditions that should exist:

1.All the relationships for the participating entity types are

“many”relationships.

2.The resulting associative entity type has independent meaning to end users and, preferably, can be identified with a single-attribute identifier.

3.The associative entity has one or more attributes in addition to the identifier.

4.The associative entity participates in one or more relationships independent of the entities related in the associated relationship.

Figure 2-11b shows the relationship Completes converted to an associative entity type.In this case,the training department for the the company has decided to award a certificate to each employee who completes a course . Thus,the entity is named CERTIFICATE,which certainly has independent meaning to end users.Also,each certificate has a number(Certificate Number)that serves as the identifier. The attribute Date Completed is also included.Note also in Figure 2-11b and the Visio version of Figure 2-11c that both EMPLOYEE and COURSE are mandatory participants in the two relationships with CERTIFICATE .This is exactly what occurs when you have to represent a many-to-many relationship(Completes in Figure 2-11a) as two one-to-many relationships (the ones associated with CERTIFICATE in Figures 2-11b and 2-11c).

Notice that converting a relationship to an associative entity has caused the relationship notation to move.That is,the “many” cardinality now terminates at the associative entity,rather than at each participating entity type.In Figure 2-11,this shows that an employee,who may complete one or more courses(notation A in Figure 2-11a);may be awareded more than one certificate (notation A in Figure 2-11b);and that a course,which may have one or more employees complete it(notation B in Figure 2-11a),may have many certificate awareded(notation B in Figure 2-11b).See Problem and Exercise 20 for an interesting variation on Figure 2-11a, which emphasizes the rules for when to convert a many-to-many relayionship,such as Completes,into an associative entity.