Entity relationship approach to knowledge base systems: description of terms, not...

Entity: An entity can be referred to as a " common noun" like a person, object or a concept in the user environment about which the data is maintained by the organization. An entity is notated using a rectangular box. Entities are used to define set of business rules with the help of an ER diagram. The following is an example of how an entity is used to define business rules. For example, on the above figure, student and class are entities.

Strong Entity: Entity types whose instances can be uniquely identified are known as strong entity. Strong entities have a unique identifier which is underlined with a single line. For example, employee could be a strong entity as an employee ID can uniquely identify a person.

Weak Entity: Entity types whose instances cannot be uniquely identified are known as weak entity and are represent as one rectangle embedded in another and partial identifiers are underlined with 2 lines. For example, an employee who is enrolled in a health insurance plan through work can have dependents who are also eligible for the plan. The dependent is an example of a weak entity as the dependent wouldn't be eligible for the plan if the employee wasn't working in the first place. A dependent needs an employee to work in order to be enrolled in the plan and hence can be said to be a Weak entity.

Associative Entity: When 2 entities are associated in a many to many relationship, and the attribute that belongs to a relationship and not an entity type, then you take that attribute and make it a separate entity. The result of interaction between 2 entities is basically known as the associative

entity. It is dependent on both entities. Associative entities are represented by the round edged rectangles. For example, an employee (entity containing employee ID, name etc.) can do a certain project (another entity containing Project #, Project type etc.). For a particular project, there can be an associative entity called " Employee Project Engagement" which will now have attributes from both Employee and Project Entities such as Employee ID, Project ID, etc.

Relationship: Relationship is what links 2 entities together and can be represented by a single line that links 2 rectangular boxes to define how they are related to each other. In the above figure drawn for (i), the line in between the rectangular box shows the relationship between Student and Classes. In this case, it represents the fact the one student can take many classes.

Degree of Relationship: The number of entity types that are associated in a given relation is called the degree of that relation. If a single entity is involved, it's referred to as unary, binary for 2 different entities and ternary for 3 different entities related to each other. An example would be EMPLOYEEE manages another EMPLOYEE. Instances would be many but type is only one. From the figure above drawn for (i), it is an example of a binary relationship.

Cardinalities: Cardinalities refer to how many instances of one type is associated with the number of instances of other entity types involved. Cardinalities can be of four types that include Mandatory One (notated with

double vertical lines on the relationship line), Mandatory Many (notated with a single vertical like followed by a crowfoot on the relationship line), Optional One (notated using a " o" followed by a single vertical line on the relationship line) and Optional Many (notated using a " o" followed by a crowfoot on the relationship line).

Mandatory One: Example could be an enrolled student (entity) MUST take at least 1 course (another entity) to maintain enrolment status.

Mandatory Many: Example could be an enrolled International student (entity) MUST take at least 3 courses (another entity) to maintain full-time enrollment status.

Optional One: Example could be a person (entity) getting married to another person. Meaning, a person may choose to not marry at all or marry one person but not more.

Optional Many: Example could be a domestic student (entity) who can take 0 courses (another entity) to defer a semester or may simply chose to take many courses and carry on with semester.

Attribute: It's the characteristic of an entity (about which organizations wishes to retain data) and can be of 4 types.

Simple Attribute: A basic property which can be recorded. Example can be student ID.

Composite Attribute: Attributes that can be broken down in to component attributes. For example, name can be broken down in to Suffix, First Name, Middle Name and Last Name.

Multi Valued Attribute: It is used when you want to capture more than one value at a single instant. Example could be employee skills.

Derived Attribute: An attribute that is derived from the values associated with other attributes. An example could be Age of employee which can be derived from Date of Birth and Current Date.