B.Sc. (Computer Science)-Part-III Subject Name- "DBMS" Paper-I (Video Part –IV)



By- Prof. Dileep Kumar Sahu Assistant Professor

Department of Computer Application

Govt. Vishwanath Yadav Tamaskar Post Graduate Autonomous College, Durg (C.G.) Email ID: dileepksahu20@gmail.com

Unit-III: Concept of DBMS and Data Models

Content

- 1. E-R MODEL
- 2. ENTITY
- 3. ATTRIBUTE
- 4. RELATIONSHIP
- 5. RELATHIONSHIP CARDINALITY
- 6. STRONG AND WEAK ENTITY

Objective

- Describe E-R Model for a given problem;
- Describe the Entity and various types of Constraints

ENTITY RELATIONSHIP (ER) MODEL

The main features of ER model:

- Entity relationship model is a high-level conceptual data model.
- It allows us to describe the data involved in a real-world enterprise in terms of objects and their relationships.
- It is widely used to develop an initial design of a database.
- It provides a set of useful concepts that make it convenient for a developer to move from a basic set of information to a detailed and precise description of information that can be easily implemented in a database system.
- It describes data as a collection of entities, relationships and attributes.

ENTITY

What are entities?

- An entity is an object of concern used to represent the things in the real world, e.g., car, table, book, etc.
- An entity need not be a physical entity, it can also represent a concept in real world, e.g., project, loan, etc.
- It represents a class of things, not any one instance, e.g., 'STUDENT' entity has instances of 'Ramesh' and 'Mohan'.

Entity Set or Entity Type:

 A collection of a similar kind of entities is called an Entity Set or entity type.

ENTITY

Example:

- For the COLLEGE database objects of concern are Students, Faculty, Course and departments.
- The collections of all the students entities form a entity set STUDENT. Similarly collection of all the courses form an entity set COURSE.

Entity identifier key attributes

- An entity type usually has an attribute whose values are distinct for each individual entity in the collection.
- Such an attribute is called a key attribute and its values can be used to identify each entity uniquely.

ENTITY

Example:

- For the COLLEGE database objects of concern are Students, Faculty, Course and departments.
- The collections of all the students entities form a entity set STUDENT. Similarly collection of all the courses form an entity set COURSE.

Entity identifier key attributes

- An entity type usually has an attribute whose values are distinct for each individual entity in the collection.
- Such an attribute is called a key attribute and its values can be used to identify each entity uniquely.

STRONG ENTITY

Strong entity set: The entity types containing a key attribute are called strong entity types or regular entity types.

• EXAMPLE: The Student entity has a key attribute RollNo which uniquely identifies it, hence is a strong entity.

Attribute

What is an attribute?

- An attribute is a property used to describe the specific feature of the entity. So to describe an entity entirely, a set of attributes is used.
- For example, a student entity may be described by the student's name, age, address, course, etc.
- An entity will have a value for each of its attributes. For example for a particular student the following values can be assigned:
 - RollNo: 1234
 - Name: Supriya
 - Age: 18
 - Address: B-4, Mayapuri, Delhi.
 - Course: B.Sc. (H)

Attribute : Domain

- Domains:
- Each simple attribute of an entity type contains a possible set of values that can be attached to it. This is called the domain of an attribute. An attribute cannot contain a value outside this domain.
- EXAMPLE- for PERSON entity PERSON_ID has a specific domain, integer values say from 1 to 100.

Types of Attribute

- 1. Simple: The attribute that cannot be further divided into smaller parts and represents the basic meaning is called a simple attribute.
 - For example: The 'First name', 'Last name', age attributes of a person entity represent a simple attribute.
- 2. Composite: Attributes that can be further divided into smaller units and each individual unit contains a specific meaning.
 - For example:-The NAME attribute of an employee entity can be sub-divided into First name, Last name and Middle name.
- **3. Single valued:** Attributes having a single value for a particular entity.
 - For Example, Age is a single valued attribute of a student entity.
- **4. Multivalued:** Attributes that have more than one values for a particular entity is called a multivalued attribute.
 - Different entities may have different number of values for these kind of attributes.
 - For multivalued attributes we must also specify the minimum and maximum number of vales that can be attached. For Example phone number for a person entity is a multivalued attribute.
- 5. Stored: Attributes that are directly stored in the data base.
 - For example, 'Birth date' attribute of a person.
- 6. **Derived:** Attributes that are not stored directly but can be derived from stored attributes are called derived attributes.
 - For Example, The years of services of a 'person' entity can be determined from the current date and the date of joining of the person. Similarly, total salary of a 'person' can be calculated from 'basic salary' attribute of a 'person'.

Relationships

What Are Relationships?

- A relationship can be defined as:
 - a connection or set of associations, or
 - a rule for communication among entities:

Example: In college the database, the association between student and course entity,

i.e., "Student opts course" is an example of a relationship.

Relationship sets: A relationship set is a set of relationships of the same type.

• For example, consider the relationship between two entity sets student and course. Collection of

all the instances of relationship opts forms a relationship set called relationship type.

Degree

- The degree of a relationship type is the number of participating entity types.
- The relationship between two entities is called binary relationship.
- A relationship among three entities is called ternary relationship.
- Similarly relationship among n entities is called n-ry relationship.

Cardinality specifies the number of instances of an entity associated with another entity participating in a relationship.

- Based on the cardinality binary relationship can be further classified into the following categories:
- **One-to-one**: An entity in A is associated with at most one entity in B, and an entity in B is associated with at most one entity in A.
 - **Example :** Relationship between college and principal.



- One college can have at the most one principal and one principal can be assigned to only one college.
- we can define the relationship between university and Vice Chancellor

- **One-to-many**: An entity in A is associated with any number of entities in B. An entity in B is associated with at the most one entity in A.
- **Example :** Relationship between department and faculty.



One department can appoint any number of faculty members but a faculty member is assigned to only one department.

- Many-to-one: An entity in A is associated with at most one entity in B. An entity in B is associated with any number in A.
- Example : Relationship between course and instructor. An instructor can teach various courses but a course can be taught only by one instructor.



- Many-to-many: Entities in A and B are associated with any number of entities from each other.
- **Example:** Taught_by Relationship between course and faculty.
- One faculty member can be assigned to teach many courses and one course may be taught by many faculty members.



Relationship between book and author.

One author can write many books and one book can be written by more than one authors.



More about Entities and Relationships

Recursive relationships

When the same entity type participates more than once in a relationship type in different roles, the relationship types are called recursive relationships.

Participation constraints

The participation Constraints specify whether the existence of an entity depends on its being related to another entity via the relationship type. There are 2 types of participation constraints:

Total: When all the entities from an entity set participate in a relationship type, is called total participation, for example, the participation of the entity set student in the relationship set must 'opts' is said to be total because every student enrolled must opt for a course.

Partial: When it is not necessary for all the entities from an entity set to participate in a relationship type, it is called partial participation. For example, the participation of the entity set student in 'represents' is partial, since not every student in a class is a class representative.

STRONG ENTITY

Strong entity : The entity type containing a key attribute are called strong entity or regular entity.

- It is denoted by a single rectangle.
- A strong entity always has the **primary key** in the set of attributes that describes the strong entity.
- Types of strong entities together forms the Strong Entity Set.
- It indicates that each entity in a strong entity set can be uniquely identified.
- EXAMPLE: The Student entity has a key attribute **RollNo** which uniquely identifies it, hence is a strong entity.

WEAK ENTITY

- Entity types that do not contain any key attribute, and hence cannot be identified independently, are called weak entity types.
- A weak entity can be identified uniquely only by considering some of its attributes in conjunction with the primary key attributes of another entity, which is called the <u>identifying owner</u> entity.
- A weak entity is denoted by the **double rectangle**.
- Weak entity do not have the primary key instead it has a partial key that uniquely discriminates the weak entities

Strong Key Vs Weak Key: Example



DIFFERENCE BETWEEN STRONG KEY ANDWEAK ENTITY

S.NO	STRONG ENTITY	WEAK ENTITY
1	Strong entity always has primary key.	While weak entity has partial discriminator key.
2	Strong entity is not dependent of any other entity.	Weak entity is depend on strong entity.
3	Strong entity is represented by single rectangle.	Weak entity is represented by double rectangle.
4	Two strong entity's relationship is represented by single diamond.	While the relation between one strong and one weak entity is represented by double diamond.
5	Strong entity have either total participation or not.	While weak entity always has total participation.

Thank You