COMPUTER ORGANIZATION AND ARCHITECTURE

Lecture 5

Entity Relationship Modelling

Dr Victoria Mukami

INTRODUCTION

We have learnt and dwelt on the introduction of databases for quite a bit. Now we start working with various diagrams and concepts that we learned. This lecture focuses on the entity-relationship diagram. We mainly look at the types of entities and how to identify various relation types and attributes. Next, we review the role of attributes on relationships, and we review the structural constraints as related to connectivity and cardinality. Finally, we will build an entity-relationship diagram.

Learning objectives

By the end of this topic, you should be able to:

1. Distinguish the various entity types
2. Identify the various relation types and attributes
3. Determine the role of attributes on relationships
4. Understand structural constraints as related to connectivity and cardinality
5. Build and ERD diagram

OVERVIEW

In lecture three, we did a review of the database analysis and design which included the various phases of database development. One of the phases of the database lifecycle is the database design has an ERD diagram as an output specifically for the conceptual design. Why do we create ERD diagrams? There is a quote that states, that it is better to plan than not to plan at all. ERD diagrams are our foundation when creating the design for our database. ERD diagrams encourage the designer to be creative on paper while also being able to correct any inadequacies and errors from the ERD diagram. In this lecture, we focus on the components of the ERD diagram, and we will learn how to draw an ERD diagram.

ENTITY-RELATIONSHIP DIAGRAM CONCEPTS

The ERD as mentioned is an output of the conceptual design within the database design model. It is a representation of the conceptual database and is based on how a user views the database. Let us first review the notations commonly used when creating ERDs. Later we look at entities, attributes, relationships and cardinality regarding their symbols within the ERD.

Notations

There are three major entity-relationship notations. These include the Chen notation, Crow's foot notation and the class diagram notation [1]. We will review all three here but will mainly use the Crow's foot in our examples.

1. Chen's Notation

This notation is represented by entities as squares, relationships as a diamond with connectivities being written next to each entity box [1]. The relationship name is written within the diamond as represented in Figure 1.

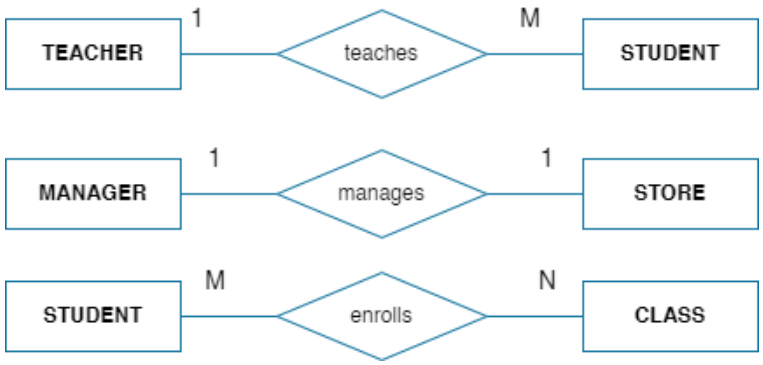

Figure 1: Chen's Notation

2. Crow's Foot Notation

This notation gets its name from the three-prong symbol used to show the many relationships side. The three-prong matches the crow's foot. The entity is still shown in a box but features various attributes while the connectivities are represented by symbols [1]. The relationship is represented by a word on the two joining lines as represented in Figure 2.

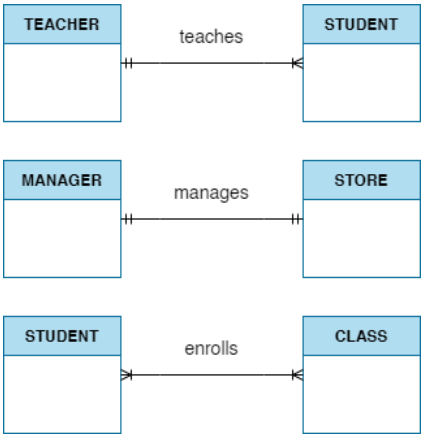

Figure 2: Crow's Foot Notation

3. Class Diagram Notation

The entities are represented by the boxes while connectivities are represented by lines with symbols [1]. Relationships are represented by names on both sides of the relationship as shown in Figure 3.

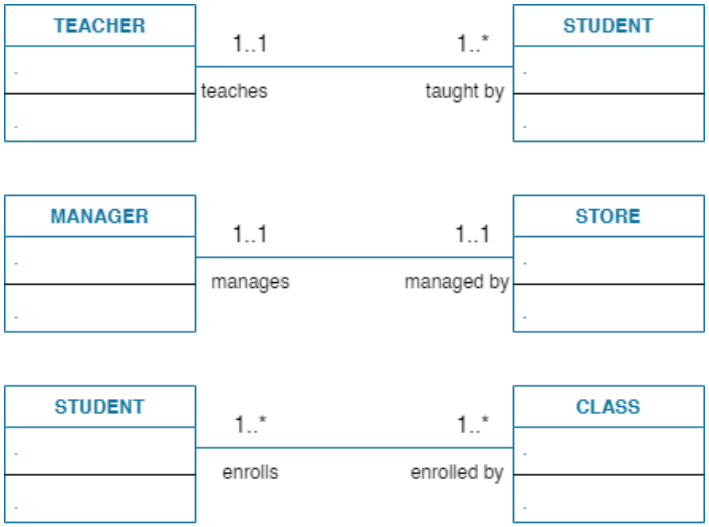

Figure 3: Class Diagram Notation

Entity

This is a person, place, or thing that the data is recorded about. Entities are considered to represent a real-world objects. Entities are represented by boxes. A **Weak entity** is considered weak as it can only exist based on a main or strong entity. Both a (strong/normal) entity (department) and a weak entity (course) are represented in Figure 4 . Figure 4 is a representation of Chen's notation; however, the representation of weak entities is identified by the attributes listed within a Crow's foot diagram.

Figure 4: Crow's Foot Representation of Attributes

Attribute

This is a characteristic of the entity. Within the Chen notation, attributes are represented as ovals within Figure 5 while the Crow's foot represents them in words within the entity box as shown within the specific attributes. A teacher's attributes may include TA_FNAME, TA_LNAME, TA_GENDER, TA_DOB e.t.c.

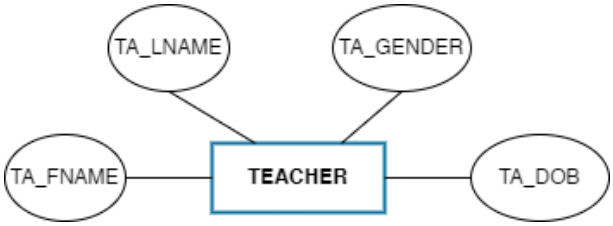

Figure 5: Crow's Foot Representation of Attributes

There are four types of attributes, and these include:

1. Key Attributes (Identifiers)
2. Composite Identifiers
3. Multivalued Attributes
4. Derived Attributes

1. Key Attributes (Identifiers)

This is an attribute that uniquely identifies an entity. Another name for identifiers as primary keys. For instance, as a citizen of Kenya, you can be uniquely identified by a national ID. As a bank customer, you can be uniquely identified by your account number. Within the crow's foot, the key attribute is represented by the words PK and is underlined. For instance, the primary key for teachers is PF_NO in Figure 6.

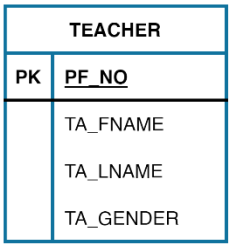

Figure 6: Key Attributes Representation

2. Composite Identifiers

A composite attribute is an attribute that is a combination of other attributes. For instance, a bank customer may have an address that is made up of the P.O Box, Postal Code, City, and Country.

3. Multivalued Attributes

Sometimes an attribute can have several values. For instance, a student may have several phone numbers or even several email addresses. The Crow's foot model does not identify multivalued entities. Later, we will learn when working with normalization that multivalued attributes should be broken down into single-valued attributes.

4. Derived Attributes

Also known as computed attributes. A derived value changes and is calculated from other attributes. The values of derived attributes are not stored within the table/database but are normally calculated or derived using an algorithm [1]. For instance, if a human resource manager wanted to know how many teachers are set to retire within the next two years, then they can derive that by subtracting the date of birth from the current date. Figure 7 is a representation of Retirement based on DOB.

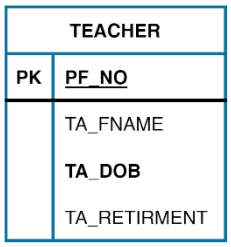

Figure 7: Derived Attributes

Relationship

This is a connection between entities. There are three relationships mainly used and are notated in different ways. These include one-to-many (1:M), many-to-many (M: N), and one-to-one (1:1) relationships. Figure 1 (Chen's Notation), Figure 2 (Crow's Foot Notation) and Figure 3 (Class Diagram Notation) are representations of the relationship representation. We need to define two new terms: connectivity and cardinality. Connectivity describes the relationship classification while cardinality is the minimum and the maximum number of entity occurrences [1].

For instance, in Figure 2 one and only one teacher can teach many students. In the case of a university, this may be represented as 1,1 teacher can teach 5,20 students. This means that a minimum of 1 and a maximum of 1 teacher can teach a minimum of 5 and a maximum of 20 students.

ERD CREATION

Building an ERD is considered an iterative process that follows the following activities [1]

1. Create a detailed narrative of the business operations
2. Identify business rules based on 1 above
3. Identify entities and relationships from 2 above
4. Develop an initial ERD
5. Identify attributes and any identifiers that describe the entities
6. Revise and redraw the ERD

We will work on each of these rules using a case study. Step 1 above where a detailed narrative is present would be extracted during the database planning and design phases of the database lifecycle. Assuming that the business narrative was used and business rules identified we will then work from step two.

Identify Business Rules

A business rule is defined as a brief, precise and unambiguous description of a procedure or principle within a business [1]. Just because they are called business rules doesn't mean they only apply to businesses, but they also apply to organizations of all kinds [1]. We will use the following business rules for the case study.

1. Bustani Vocational Training College (VTC) is a training centre that has three teaching departments: Computing, Business and Arts. Each department is headed by a Chair who is also an instructor. Each Chair can only head one department.
2. Each department offers many courses. For instance, the computing department offers courses such as digital literacy, graphic design, and beginners programming.
3. Each department has one or more Instructors. Of those instructors only, one heads the department.
4. Instructors are part of the employees of Bustani VTC. An instructor is a member of the employees, and the employees have one to many instructors.
5. Each instructor may teach one to a maximum of three courses.
6. A student may enrol into courses and can enrol on a maximum of five courses. For instance, a student within the computing department is hosted by the computing department and can only take courses approved by the department.
7. A student is hosted by one department. A student can only belong to one department and each department has many students.
8. Each course is also hosted by a specific department. Each department may have many courses.

Identify Entities and Relationships

This step aims to identify all the entities and any relationships. Looking at the business rules we can easily see that the following are the entities.

- DEPARTMENT
- EMPLOYEE
- STUDENT
- INSTRUCTOR
- COURSE

Based on the entities above we can easily indicate and determine the relationships between these entities as shown in Table 1 [1].

Table 1: Derived entities and relationships

ENTITY	RELATIONSHIP	CONNECTIVITY	ENTITY
DEPARTMENT	employs	1:M	INSTRUCTOR
DEPARTMENT	has	1:M	STUDENT
DEPARTMENT	hosts	1:M	COURSES

INSTRUCTOR	chairs	1:1	DEPARTMENT
INSTRUCTOR	is an	1:1	EMPLOYEE
INSTRUCTOR	teaches	1:M	COURSE
STUDENT	takes	1:M	COURSE

Develop initial ERD

Once all the entities and relationships have been identified then the initial ERD diagram can be created. This diagram does not yet have attributes or identifiers. Figure 8 represents the initial ERD diagram.

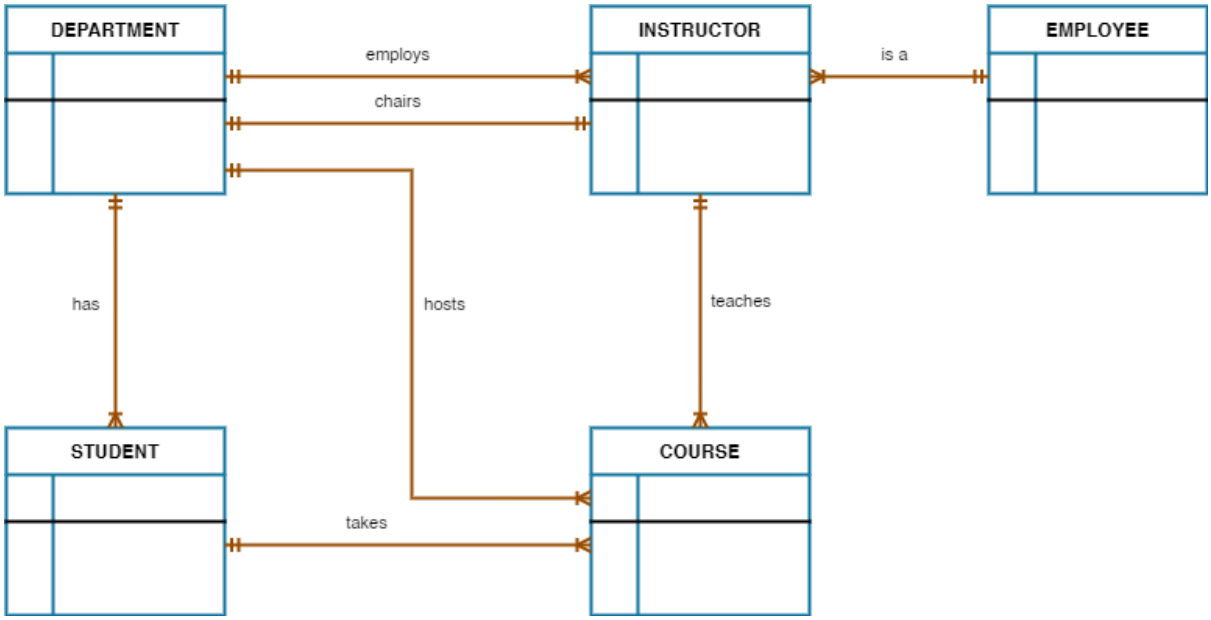

Figure 8: Initial ERD diagram for Bustani VTC

Identify attributes and identifiers

It is imperative that the various attributes and identifiers (primary keys for each entity are shown. While the business rules might not have the attributes, the narrative may offer this. Additionally, the database designer may document what sort of records need to be stored including the various attributes. For our case study, some of the attributes and primary keys are shown in Figure 9.

Revise and Redraw ERD

Based on attributes then we would draw the final ERD as shown in Figure 9. You will notice we added a new entity that is a combination of the STUDENT and COURSE

entities. The new entity is called REGISTRATION. We will discuss the necessity when we work on normalization in the next lecture.

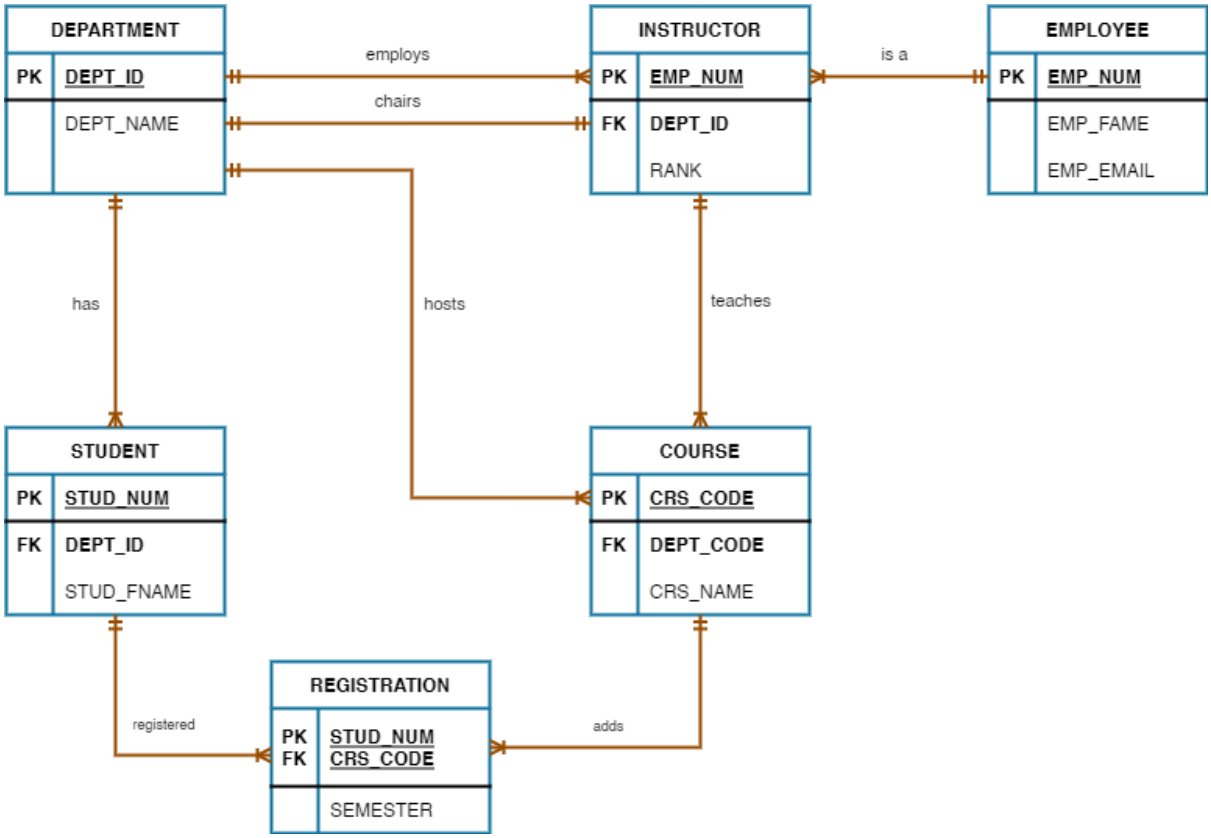

Figure 9: Final ERD

SUMMARY

During this lecture, we have done a review of ERD diagrams. We started by reviewing some ERD concepts such as entities, attributes and relationships. We reviewed the four types of attributes. We also looked at the three most popular notations i.e., Chen's, Crow's foot, and the class diagram notation. We then learnt about the process of drawing an ERD diagram. It is necessary to follow specific steps, and this was introduced. Finally, we were able to come up with an ERD diagram based on specified business rules.

DISCUSSION TOPIC

We follow up with the discussion topic from the previous lectures. We are building an ERD system for a lands office. Come up with the various business rules, identify the entities and attributes, and create the initial ERD. Thereafter, identify the attributes

and primary keys then create the final ERD. Work with your peers and peer review each other's work to see the different perspectives that each of you had.

REFERENCES

[1] Database systems: design, implementation, and management, Coronel, C., & Morris, S, Cengage Learning, 2019.

[2] Database Systems: A Practical Approach to Design, Implementation, and Management, Connolly, T., & Begg, C., Pearson, 2015.

[3] Fundamentals of database systems, Elmasri, R., & Navathe, S. B., Pearson Education Limited, 2016.