Class Notes - CS482 - Chapter 4

Main Questions:

1. What is the ER-approach?
2. What are the components of an ER-diagram?
3. How to translate an ER-diagram to a relational database schemas?

Question 1. ER-approach is a design methodology; it is not a data model (Why?); can be applied in different data models; it is the first step in the process of database design;

Question 2. *entity types*, *relationship types*, and *integrity constraints*.

Entity: object; has attributes; attributes can be *set-valued*.

Entity type: set of similar entities; can have keys; represented by a rectangle in ER-diagrams; attributes attached in eclipses; set-valued attributes are in double eclipse; entity type could be viewed as a *class* in OO-design.

Schema: the schema of an entity type consists of the name of the entity type, its set of attributes, and the key constrains;

Relationship: used to related entities to each other; involves one, two, or more entitites; has attributes; has roles (can be explicitely or implicitely named);

Relationship type: set of similar relationships; has keys; cardinality constraints; represented by diamond; attached attributes; connect to entity types in the relationship;

Schema of a relationship type: set of attributes, set of roles (with their corresponding entity types), ans set of constraints;

Cardinality constraint: involves an entity type (say E) and a relationship (say A) on a role (say R); is of the form $\min, \max$ where $0 \leq \min \leq \max$ and $\max$ could be $*$ (infinity); It states that each entity of type E has to appear in at least $\min$ and at most $\max$ relationships of type A.

Need to differentiate between entity and entity type, relationship and relationship type, relationship and relation in relational data model.

Special cases: entity type hierarchies, participation constraints, part-of relationships

Entity type hierarchies: a special type of relationship between two entity types (say R, R'); every entity of type R is also an entity of type R'; every attribute in R' is also an attribute in R; represents by a triangle with ISA is the name of the relationship type. Also referred to as *subtyping* relationship; can be a disjoining or covering relationships;

Participation constraints: a requirement on the entity type E which is involved in the role R of a relationship A; this says that for each entity e of type E there is a relationship r of A such that e participates in r in the role R;

Part-of relationships: relationship between two entity types E and E' of which one (E') is part of the other (E); if an entity of type E yields to exist then two possibilities arise: its components in E' continue to exist (*non-exclusive*) or yield to exist (*exclusive*); no special feature needs to be added for the non-exclusive case; for the exclusive case, the entity type E' and the relationship type related E and E' are called *weak entity type* and *weak relationship type*; represented by double rectangles and diamonds, respectively.

Question 3. Two algorithms for converting entities and relationships into relations in relational database schemas.

Entity type: name, attributes, keys; need to add set-valued attributes to key;

Relationship type: name, attributes, primary keys (NOT NULL requirements) of all the entities involved in the relationship; foreign key constraints for all the primary keys;

Special cases:

ISA *relationships*: the primary key of the super type plus attributes; subtypes can be combined into one single relation when disjoined; supertype can be splitted into different relations when covering constraint is satisfied;

Participation cosntraints: an inclusion dependency (foreign key, assertion, or trigger)

Part-of relationships: three cases; (i) non-exclusive part-of, subparts can be shared between different whole — normal translation of a relationship into a relation; (ii) non-exclusive part-of, subparts can be part of at most one whole — can merge the weak entity with the part-of relationship to create a relation which has the primary key of the master entity (NULL possible) as part of its attributes and a foreign key refer to the master entity; (iii) exclusive part-of: similar to participation and key constraints.