Cesar Santiago

ID: 970403634

COP4710 Database Systems

Pf. J. Teichert

## **Checkpoint 11.1:**

1) The first mapping rule is to map strong entities. To do that, one must develop a new table for each strong entity and make the indicated key the primary key of the said table. If there is more than one indicated key then one must choose which to make the primary key through induction.

2) Weak entities depend on strong entities, in the sense that without strong entities weak entities cannot exist.

3) After the weak entities, you should map the relationships between the strong entities. Firstly the M:N relationships and then the 1:M then the 1:1.

4) You would map such entities the same way you map a weak entity from a strong entity. In such cases, the weak entities from which the other weak entities are depending from should be considered the strong entity.

5) You would end up with a lot of repeated registers. You want the N entity to hold the data of the 1 entity because the 1-side holds less data than the N-side.

6) The participation makes it so that an entity from the N-side might not exist as data in the database. So mapping it to a table that is apart from the side with participation makes it so if more information needs to be related it does not conflict with information that already exists.

## **Checkpoint 11.2:**

1) If a table has three key attributes that shape their name similar to three different tables, this hints that all three tables hold a relationship together, and therefore is a ternary relationship.

2) Look for tables that share the primary keys of another table.

3) If there is a single large table that appears to have more than one flag and several values corresponding to the flags, it is probably an overlapping generalization/specialization relationship.

4) To determine whether there are shared subclass entities, we need to follow reverse engineering rules 5a–5c, and we need data to help us determine if the subclasses are actually shared.

5) If the primary key of a table appears on other tables as a foreign key, there could be a category or union type of relationship in which the superclasses have different primary keys; hence, a surrogate key was created, which was included in the other superclass tables as the foreign key.

## **Chapter 11 Exercises:**

Do exercise 11.2.

11.2)

