



ER Model

Instructor : Nitesh Kumar Jha

niteshjha@soa.ac.in

ITER,S'O'A(DEEMED TO BE UNIVERSITY)

July 2018

Review

■ The ER Model

Content

- The ER Model
- The ER Diagram

Extended ER Features

- The extended ER features are
 - Specialization
 - Generalization
 - Aggregation

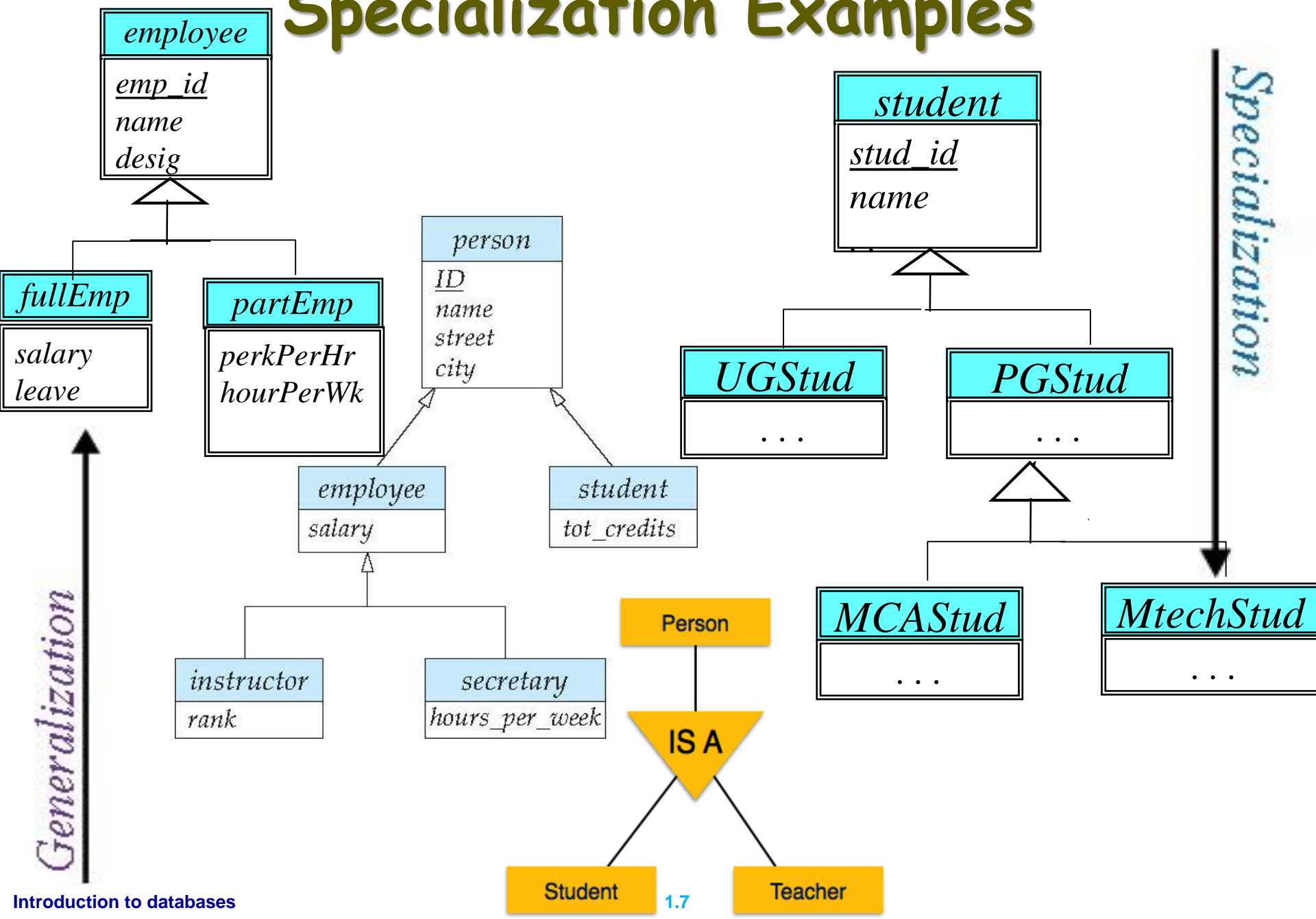
Specialization

- An entity set may allow further sub-grouping within it based on the distinctive features (attributes)
- Specialization is a technique of designating sub-grouping of entities within an entity set based upon the distinct attributes that the entity possess
- Specialization is a refinement approach that stems from an entity set (higher level entity set) in order to form multiple entity sets (lower level entity set)
- The entities in the lower level entity sets are the subsets of the entities in its higher level entity set.
- This is known as '*IS A*' relationship.
- Ex1. Tiger *is an* Animal, Cheetah *is an* Animal
- Also be referred as “*Superclass – subclass*” relationship

Generalization

- Generalization is the approach of synthesizing multiple entity set (lower level) in order to form a higher level entity set based upon common features (attributes) possessed by the lower level entity sets
- Generalization stems from multiple lower level entity sets and forms a single higher level entity set.
- In ER diagram it is represented in same way as specialization.
- Ex. Given: Cat, Tiger, Wolf, Lion, elephant entity sets, we extract and use all the common features available in all entity sets to form another Animal entity set.

Specialization Examples



Attribute Inheritance

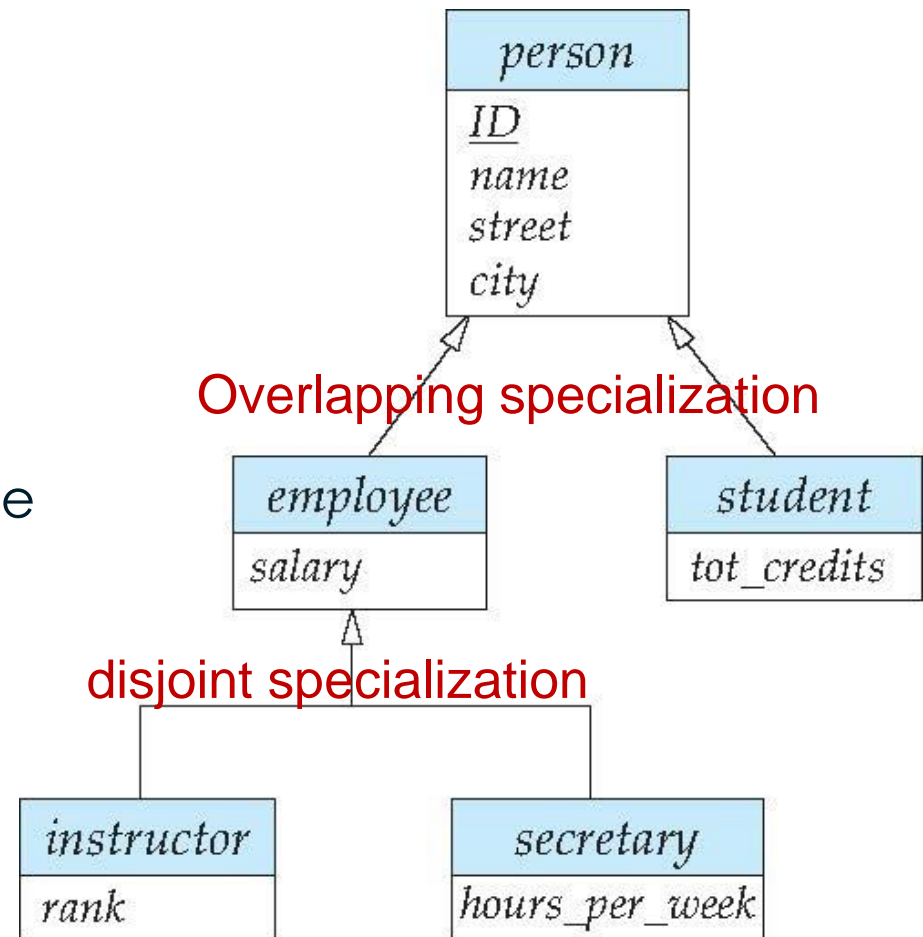
- All the attributes including the primary key of the higher level entity set are inherited/derived and associated to all lower level entity sets
 - Ex. *student* and *employee* inherit the attributes of *person*.
 - Thus, *student* is described by its *ID*, *name*, and *address* attributes, and additionally a *tot_cred* attribute;
 - *employee* is described by its *ID*, *name*, and *address* attributes, and additionally a *salary* attribute.

Participation Inheritance

- A lower-level entity set also inherits participation in the relationship sets in which its higher-level entity participates.
 - Participation inheritance applies through all tiers of lower-level entity sets.
 - *Ex.* suppose the *person* entity set participates in a relationship *person_dept* with *department*.
 - Then, the *student*, *employee*, *instructor* and *secretary* entity sets, which are subclasses of the *person* entity set, also implicitly participate in the *person_dept* relationship with *department*.
 - The above entity sets can participate in any relationships in which the *person* entity set participates.

Overlapping and Disjoint Constraints

- Specialization is said to be **overlapping** if there exists at least one entity in the higher entity set that belongs to more than one lower level entity sets, otherwise called **disjoint**.
- Overlapping Specialization
 - If an entity belong to multiple specialized entity sets
- Disjoint Specialization
 - if it must belong to at most one specialized entity set.



Membership Constraint

■ Condition-defined Membership

- A membership is confirmed if an entity satisfies an explicit condition.
- Ex. *student* has an attribute *student_type*. Whose value can be 'undergraduate' or 'graduate'
- Then all entities that satisfy the condition *student type* = "undergraduate" are included in *UGStud*.

■ User-defined Membership

- after 3 months of employment, university employees are assigned to one of four work teams.
- The teams are represented as *four* lower-level entity sets of the higher-level employee entity set.
- A given employee is assigned to a specific team entity set manually.

Completeness Constraint

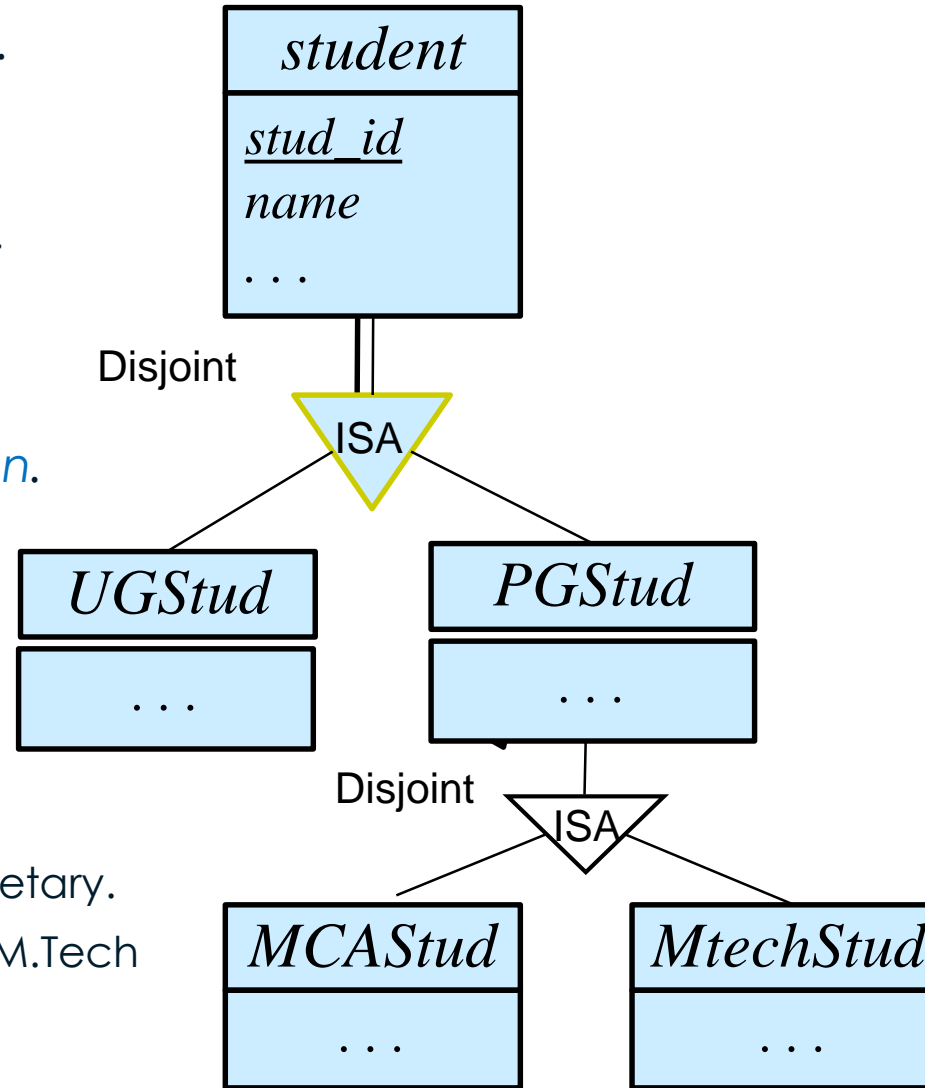
■ Total generalization or specialization.

- Each higher-level entity must belong to a lower-level entity set.
- *Ex.* The *student* generalization is total: All student entities must be either *post graduate* or *undergraduate*.

■ Partial generalization or specialization.

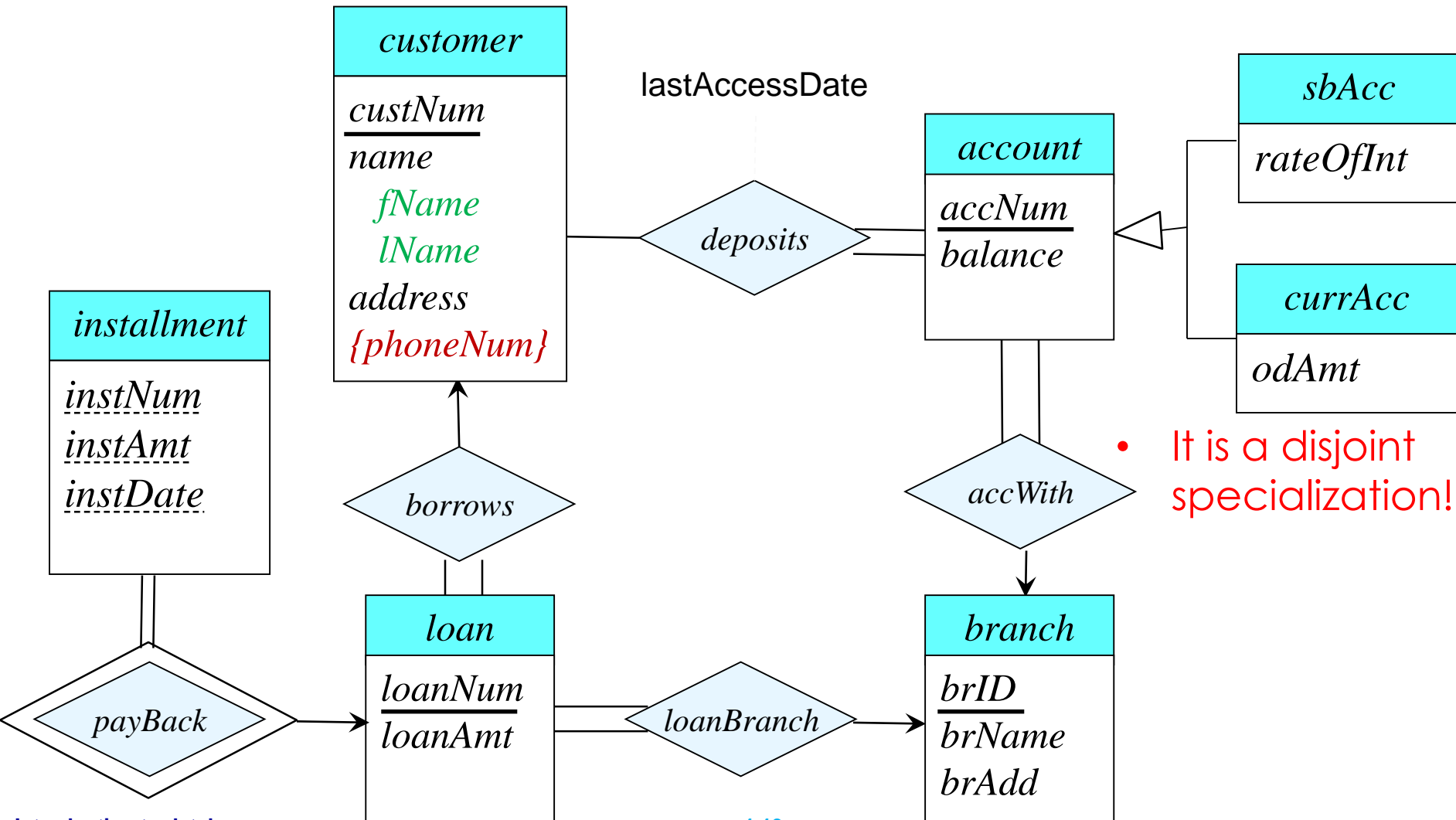
- Some higher-level entities may not belong to any lower-level entity set.
- Partial generalization is the default.

Ex. The employee generalization is partial. Some employees are there who are neither an instructor nor a secretary.
Ex: All PG Students may not be MCA or M.Tech Students



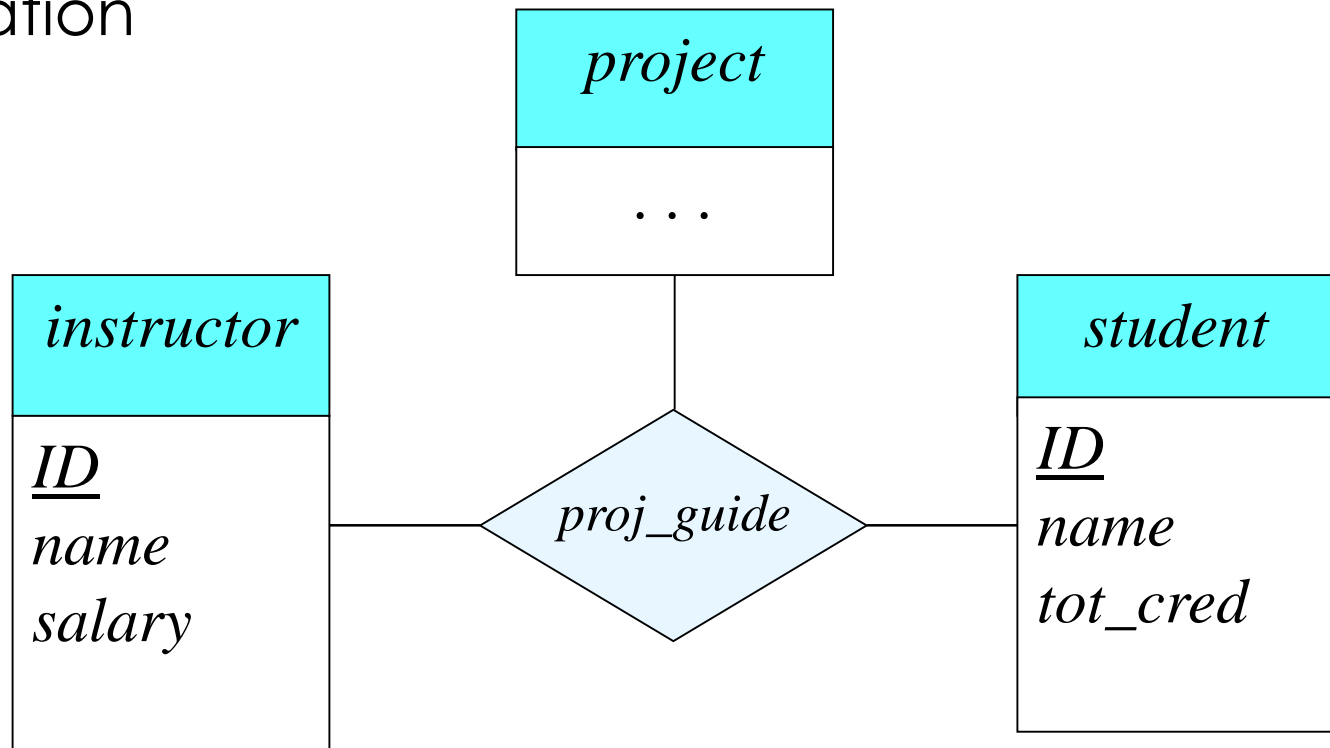
Specialization/Generalization

- Design two sub entity sets of **account** as **sbAcc** and **currAcc**.



Problem

- How to Modify the model s.t. an **Instructor** guiding a **student** on a **project** can file a monthly evaluation report containing **instructor**, **student** and **project** information

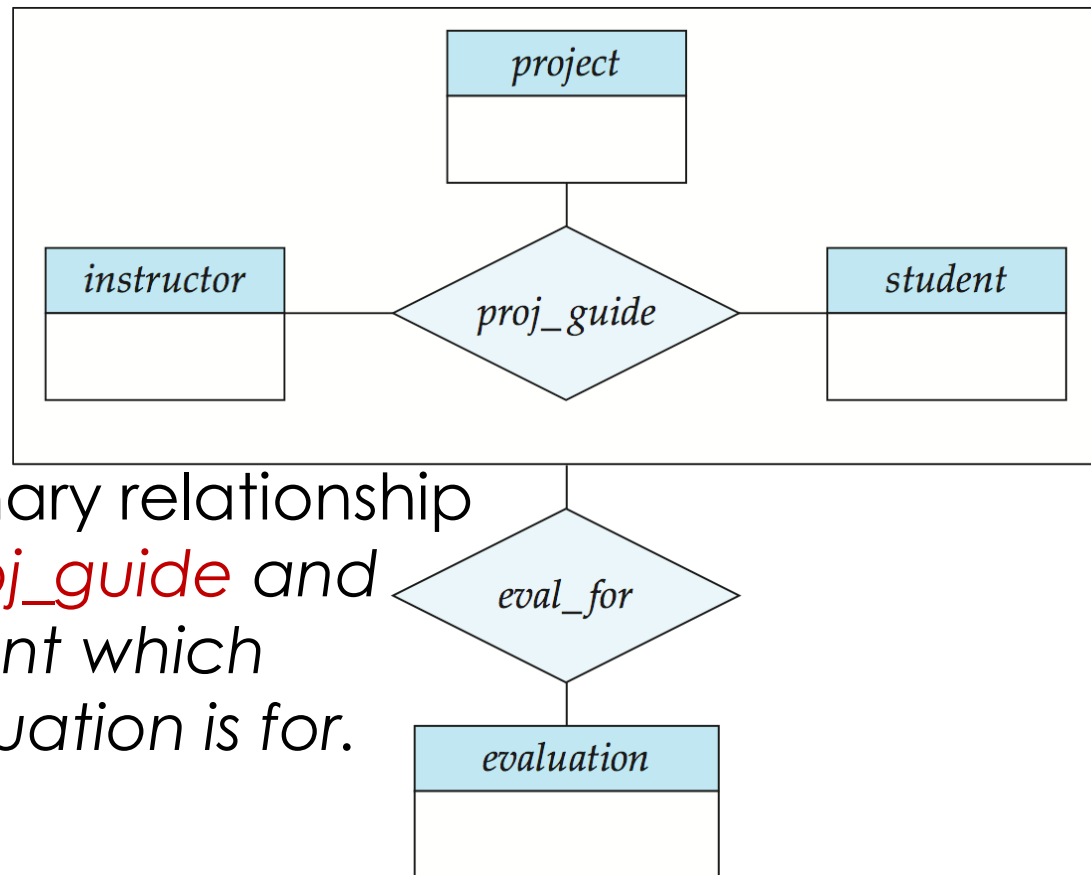


Aggregation -I

- It can express relationship among relationship sets
- Aggregation facilitates to treat a relationship set (along with all of its associated entity sets) as a higher level entity set and can participate in another relationship set.
 - i.e. Aggregation is an abstraction through which relationships are treated as higher-level entities.

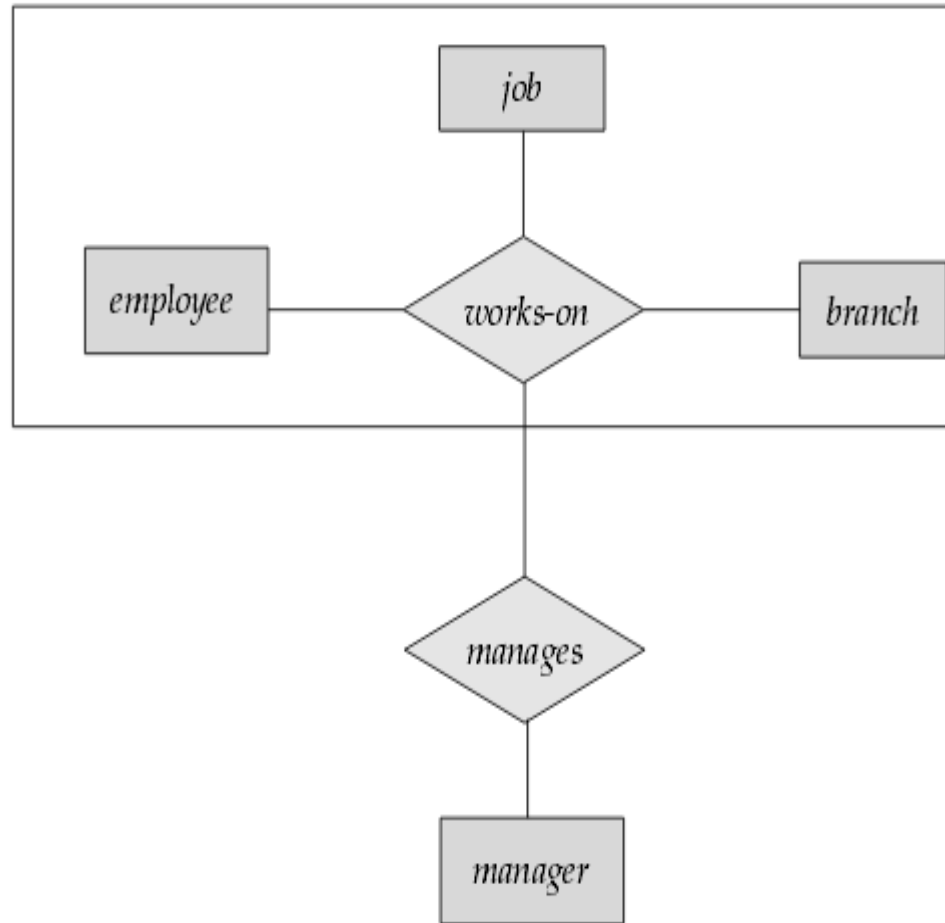
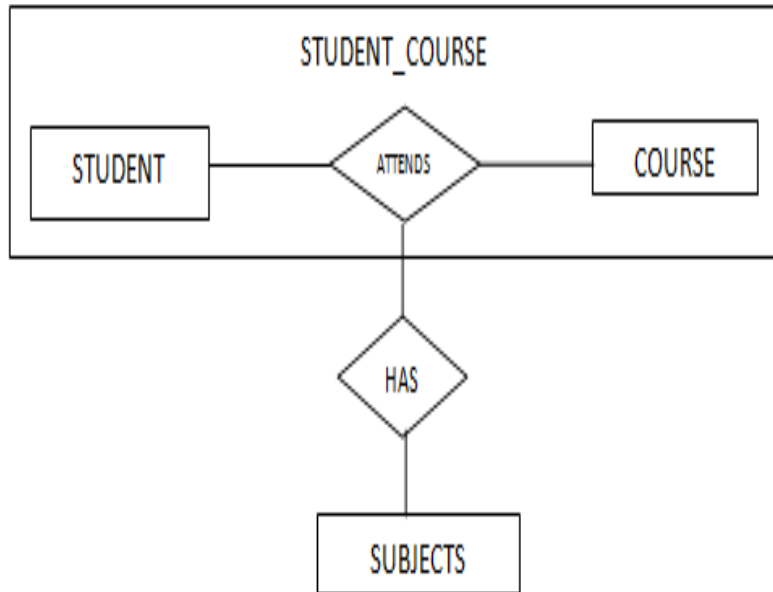
Aggregation -II

- The relationship set *proj_guide* is represented as a higher-level entity set called *proj_guide*.
- Such an entity set is treated in the same manner as is any other entity



- We then create a binary relationship *eval_for* between *proj_guide* and *evaluation* to represent which combination an evaluation is for.

Aggregation -III



ER DIAGRAM WITH AGGREGATION

Thank You