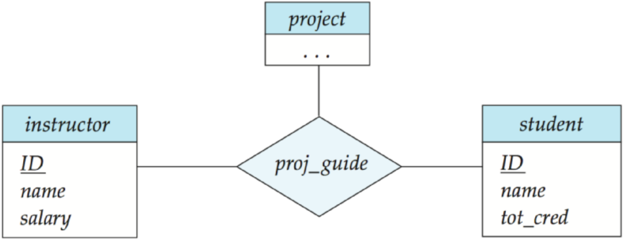
Database Management Systems (CSN-351)
Entity Relationship Model (contd.)

BTech 3rd Year (CS) + Minor + Audit

Instructor: **R**anita **B**iswas
Department of Computer Science and Engineering
Indian Institute of Technology Roorkee
Roorkee, Uttarakhand - 247 667, India

E-R Diagram with a Ternary Relationship



Cardinality Constraints on Ternary Relationship

- At most one arrow is allowed out of a ternary (or greater degree) relationship to indicate a cardinality constraint.
- If there are more than one arrow, there are multiple ways of defining the meaning.

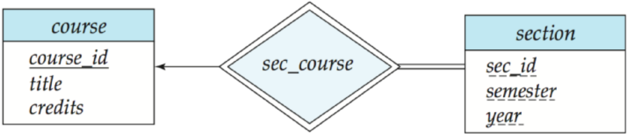
Weak Entity Sets

- An entity set that does not have a *primary key* is referred to as a weak entity set.
- The existence of a weak entity set depends on the existence of an *identifying entity set*.
  - It must relate to the identifying entity set via a *total, one-to-many relationship* set from the identifying to the weak entity set.
  - Identifying relationship depicted using a *double diamond*.

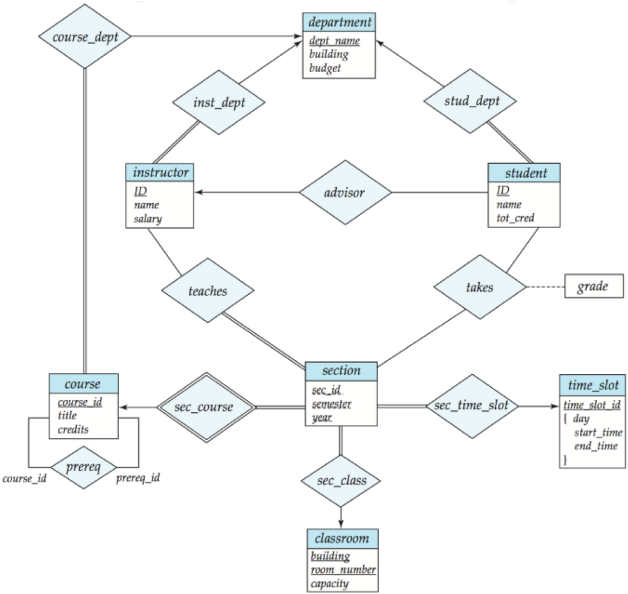
Weak Entity Sets (contd.)

- The *discriminator* (or *partial key*) of a weak entity set is the set of attributes that distinguishes among all the entities of a weak entity set that depend on one particular strong entity.

- The *primary key of a weak entity set* is formed by the primary key of the strong entity set on which the weak entity set is existence dependent, plus the weak entity set's discriminator.



E-R Diagram for a University

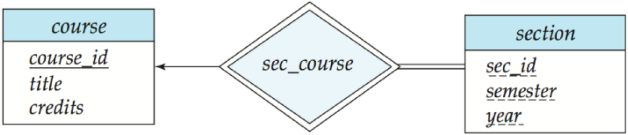


Reduction to Relation Schemas

- Entity sets and relationship sets can be expressed uniformly as *relation schemas* that represent the contents of the database.
- A database which conforms to an E-R diagram can be represented by *a collection of schemas.*
- For each entity set and relationship set there is a *unique schema* that is assigned the name of the corresponding entity set or relationship set.
- Each schema has a number of *columns* (generally corresponding to attributes), which have unique names.

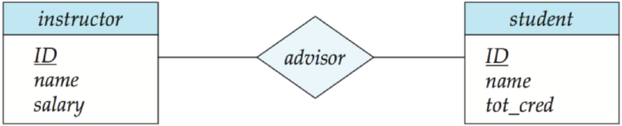
Representing Entity Sets With Simple Attributes

- A strong entity set reduces to a schema with the same attributes $student(\underline{ID}, name, tot\_cred)$
- A weak entity set becomes a table that includes a column for the primary key of the identifying strong entity set $section(\underline{course\_id}, \underline{sec\_id}, \underline{sem}, \underline{year})$



Representing Relationship Sets

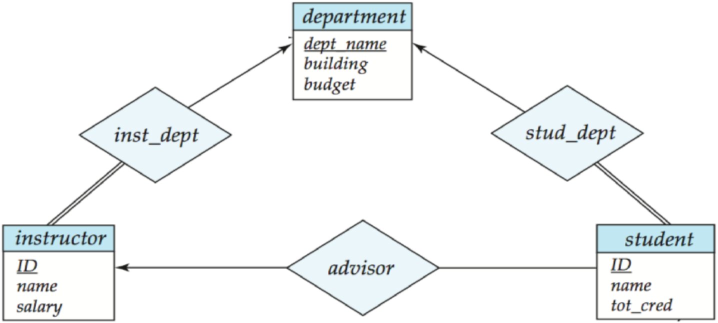
- A many-to-many relationship set is represented as a schema with attributes for the primary keys of the two participating entity sets, and any descriptive attributes of the relationship set.
- Example: schema for relationship set advisor
  $advisor(\underline{s\_id}, \underline{i\_id})$



Redundancy of Schemas

- Many-to-one and one-to-many relationship sets that are total on the many-side can be represented by adding an extra attribute to the *many* side, containing the primary key of the *one* side

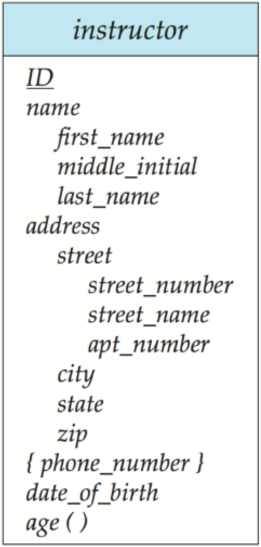
- Example: Instead of creating a schema for relationship set *inst_dept*, add an attribute *dept_name* to the schema arising from entity set instructor.



Redundancy of Schemas (contd.)

- For one-to-one relationship sets, either side can be chosen to act as the *many* side
- If participation is partial on the *many* side, replacing a schema by an extra attribute in the schema corresponding to the *many* side could result in null values
- The schema corresponding to a relationship set linking a weak entity set to its identifying strong entity set is redundant.
    - Example: The *section* schema already contains the attributes that would appear in the *sec_course* schema

Composite Attributes



instructor

- <u>ID</u>
- name
  - first_name
  - middle_initial
  - last_name
- address
  - street
    - street_number
    - street_name
    - apt_number
  - city
  - state
  - zip
- { phone_number }
- date_of_birth
- age ( )

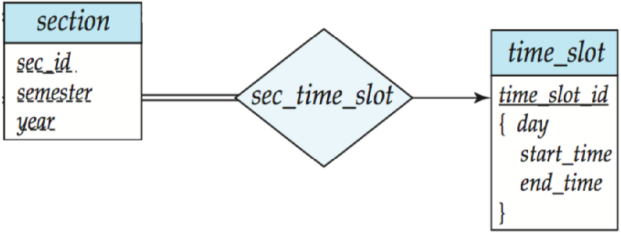
- Composite attributes are flattened out by creating a separate attribute for each component attribute
  - Example: given entity set instructor with composite attribute name with component attributes *first_name* and *last_name* the schema corresponding to the entity set has two attributes *name_first_name* and *name_last_name*
- Ignoring multivalued attributes, extended instructor schema is $instructor(\underline{ID}, first\_name, middle\_initial, last\_name, street\_number, street\_name, apt\_number, city, state, zip\_code, date\_of\_birth)$

Multivalued Attributes

- A multivalued attribute M of an entity E is represented by a separate schema EM.
    - Schema EM has attributes corresponding to the primary key of E and an attribute corresponding to multivalued attribute M.
    - Example: Multivalued attribute *phone_number* of instructor is represented by a schema: $inst\_phone(\underline{ID}, \underline{phone\_number})$
    - Each value of the multivalued attribute maps to a separate tuple of the relation on schema EM.
        - For example, an instructor entity with primary key 22222 and phone numbers 456-7890 and 123-4567 maps to two tuples: (22222, 456-7890) and (22222, 123-4567)

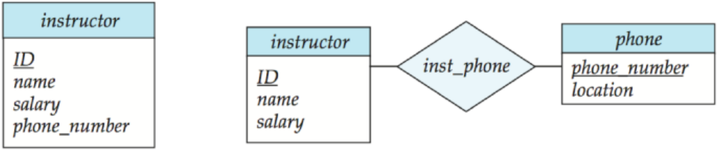
Multivalued Attributes (contd.)

- Special case: entity *time_slot* has only one attribute other than the primary-key attribute, and that attribute is multivalued.
  - Optimization: Don't create the relation corresponding to the entity, just create the one corresponding to the multivalued attribute
  - $time\_slot(\underline{time\_slot\_id}, \underline{day}, \underline{start\_time}, end\_time)$
  - Limitation: *time_slot* attribute of section (from *sec_time_slot*) cannot be a foreign key due to this optimization



Design Issues

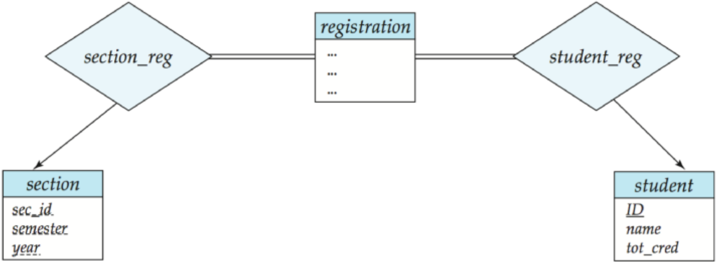
- Use of entity sets vs. attributes



- Use of phone as an entity allows extra information about phone numbers (plus multiple phone numbers)

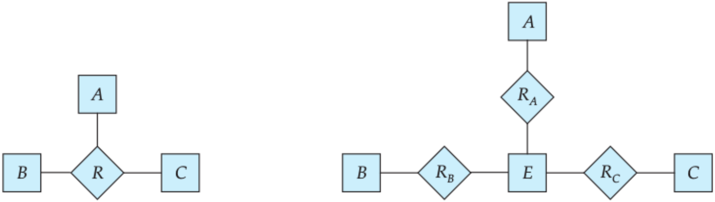
Design Issues

- Use of entity sets vs. relationship sets



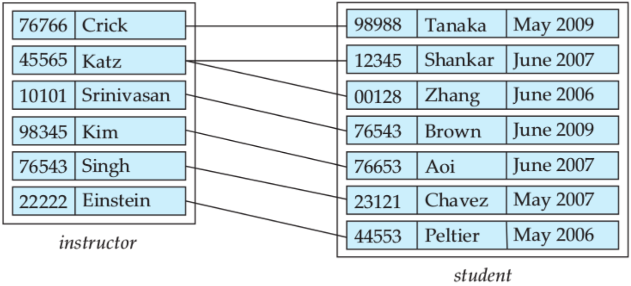
Design Issues

- Binary versus n-ary relationship sets



Design Issues

- Placement of relationship attributes



76766	Crick
45565	Katz
10101	Srinivasan
98345	Kim
76543	Singh
22222	Einstein

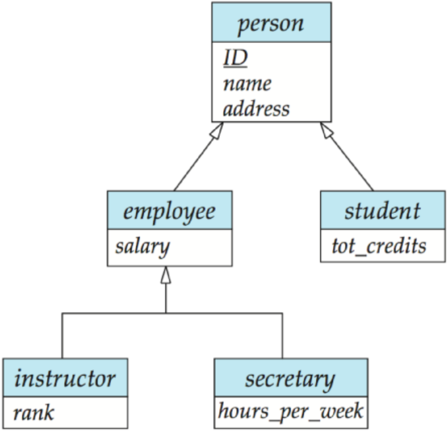
instructor

98988	Tanaka	May 2009
12345	Shankar	June 2007
00128	Zhang	June 2006
76543	Brown	June 2009
76653	Aoi	June 2007
23121	Chavez	May 2007
44553	Peltier	May 2006

student

Extended E-R Features: Specialization

- Example of Specialization



Specialization (contd.)

- Top-down design process; we designate subgroupings within an entity set that are distinctive from other entities in the set.

- These subgroupings become lower-level entity sets that have attributes or participate in relationships that do not apply to the higher-level entity set.

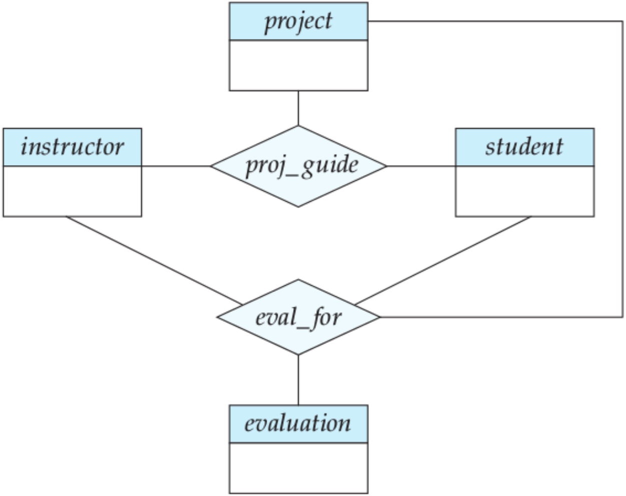
- Depicted by a triangle component labeled ISA
  (e.g., instructor "is a" person).

- *Attribute inheritance* — a lower-level entity set inherits all the attributes and relationship participation of the higher-level entity set to which it is linked.

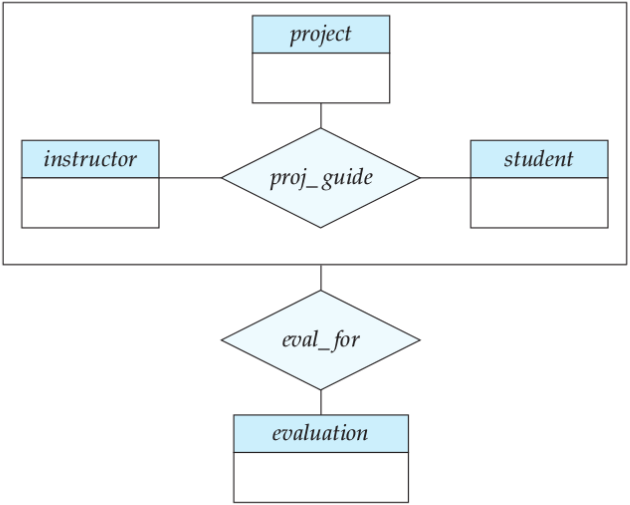
Extended E-R Features: Generalization

- A bottom-up design process — combine a number of entity sets that share the same features into a higher-level entity set.
- Specialization and generalization are simple inversions of each other; they are represented in an E-R diagram in the same way.
- The terms specialization and generalization are used interchangeably.

Extended E-R Features: Aggregation



Aggregation (contd.)


project
instructor
proj_guide
student
eval_for
evaluation