Databases and Information Systems CS303

Database design : ER Diagrams
01-09-2023

Recap

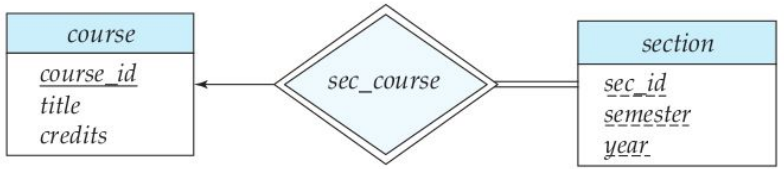
- Entity sets
- Relationship sets
- Attribute sets

Weak Entity sets

- Weak Entity is an Entity set that does not have sufficient attributes to form a Primary Key
- Entity that has a primary key is called a strong entity
- Weak entity is always associated with a strong entity called identifying entity set (Owner entity set)
- The relationship that associates the weak entity set with its Owner entity set is called the Identifying relationship
- Identifying relationship is Many-to-One from Weak entity to Strong entity

course_id	sec_id	semester	year	building	room_number	time_slot_id
BIO-101	1	Summer	2009	Painter	514	B
BIO-301	1	Summer	2010	Painter	514	A
CS-101	1	Fall	2009	Packard	101	H
CS-101	1	Spring	2010	Packard	101	F
CS-190	1	Spring	2009	Taylor	3128	E
CS-190	2	Spring	2009	Taylor	3128	A
CS-315	1	Spring	2010	Watson	120	D
CS-319	1	Spring	2010	Watson	100	B
CS-319	2	Spring	2010	Taylor	3128	C
CS-347	1	Fall	2009	Taylor	3128	A
EE-181	1	Spring	2009	Taylor	3128	C
FIN-201	1	Spring	2010	Packard	101	B
HIS-351	1	Spring	2010	Painter	514	C
MU-199	1	Spring	2010	Packard	101	D
PHY-101	1	Fall	2009	Watson	100	A

section

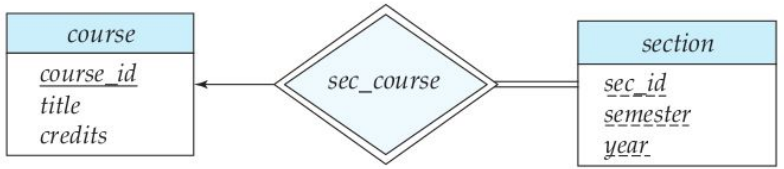


Weak Entity sets

- Discriminator of a weak entity is a set of attributes that allows distinguishing the weak entities that depend on the same owner entity
  - Also called Partial Key
- The discriminator of a weak entity is underlined with a dashed font.
- Identifying relationship is depicted by a double diamond.
- Weak entity set always ‘totally participates’ in the identifying relationship

course_id	*sec_id*	*semester*	*year*	*building*	*room_number*	*time_slot_id*
BIO-101	1	Summer	2009	Painter	514	B
BIO-301	1	Summer	2010	Painter	514	A
CS-101	1	Fall	2009	Packard	101	H
CS-101	1	Spring	2010	Packard	101	F
CS-190	1	Spring	2009	Taylor	3128	E
CS-190	2	Spring	2009	Taylor	3128	A
CS-315	1	Spring	2010	Watson	120	D
CS-319	1	Spring	2010	Watson	100	B
CS-319	2	Spring	2010	Taylor	3128	C
CS-347	1	Fall	2009	Taylor	3128	A
EE-181	1	Spring	2009	Taylor	3128	C
FIN-201	1	Spring	2010	Packard	101	B
HIS-351	1	Spring	2010	Painter	514	C
MU-199	1	Spring	2010	Packard	101	D
PHY-101	1	Fall	2009	Watson	100	A

section

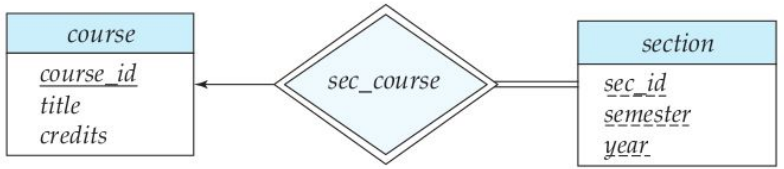


Weak Entity sets

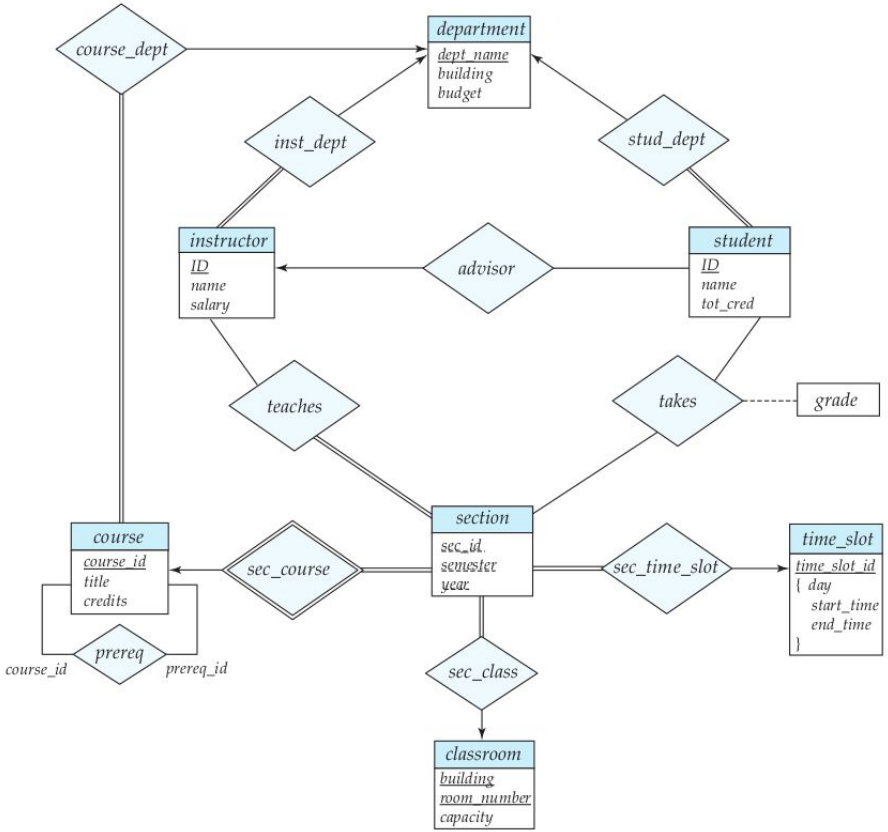
- Weak entities can be alternatively described as composite attribute of the owner entity set
- Weak entity can also have multiple owner entity sets
- Weak entities can participate in other relationships with other entity sets

course_id	sec_id	semester	year	building	room_number	time_slot_id
BIO-101	1	Summer	2009	Painter	514	B
BIO-301	1	Summer	2010	Painter	514	A
CS-101	1	Fall	2009	Packard	101	H
CS-101	1	Spring	2010	Packard	101	F
CS-190	1	Spring	2009	Taylor	3128	E
CS-190	2	Spring	2009	Taylor	3128	A
CS-315	1	Spring	2010	Watson	120	D
CS-319	1	Spring	2010	Watson	100	B
CS-319	2	Spring	2010	Taylor	3128	C
CS-347	1	Fall	2009	Taylor	3128	A
EE-181	1	Spring	2009	Taylor	3128	C
FIN-201	1	Spring	2010	Packard	101	B
HIS-351	1	Spring	2010	Painter	514	C
MU-199	1	Spring	2010	Packard	101	D
PHY-101	1	Fall	2009	Watson	100	A

section



E - R diagram for the University Database



E-R Diagrams to Relational Schema

Strong Entity sets

- Suppose E is a strong entity set with simple attributes a_1 a_2.... a_n
  - We can have a schema called E with n distinct attributes
  - Primary key of the schema will be the set of attributes that form the primary key of E

- If E has a composite attribute A (a_1 a_2.... a_m) then create one column for each of the simple attribute separately.

- If E has derived attributes, they are not represented in the relational schema

- If E has a multi-valued attribute:
  - Create a new schema with one column for the multi-valued along with the primary key of E
  - Primary Key will be all the attributes of the new schema
  - Also add foreign key constraint for primary key part of E that references the table of E

Weak Entity sets

- Suppose E is a weak entity set with simple attributes a_1 a_2.... a_n and F is the identifying strong entity:
    - We can have a schema called E with n distinct attributes of E along with the primary key attributes of F
    - Primary key of the schema E will be the set of partial attributes of E along with the primary key attributes of F
    - Create a foreign key constraint for the primary key attributes of F

Relationship sets

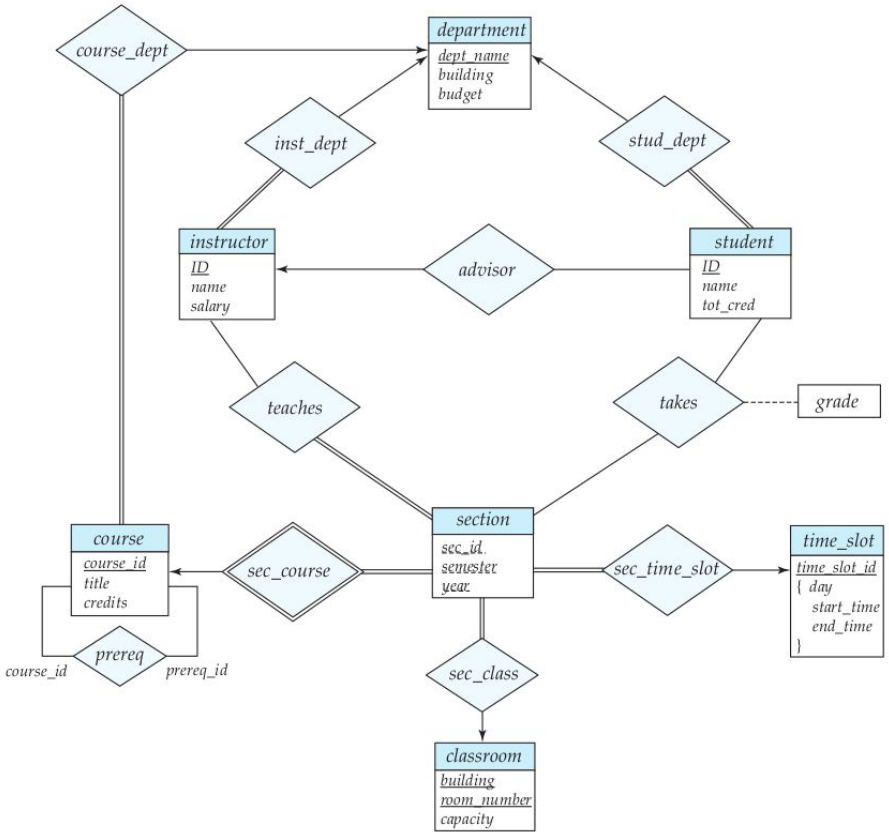
- Suppose R is a relationship set that involves the entity sets E_1 E_2.... E_n
  - We can have a schema called R with attributes to be the union all the primary key attributes of the involving entity sets
  - Primary key will be all the attributes together
  - Create Foreign key constraints for each of the participating entity sets

- We can have better primary keys for Binary Relationship sets:
  - Many-to-Many : Union of primary keys of both participating entity sets
  - One-to-One : Primary key of any one of the entity set
  - One-to-Many (or) Many-to-One: Primary key of the ‘Many’ entity set

Relationship sets

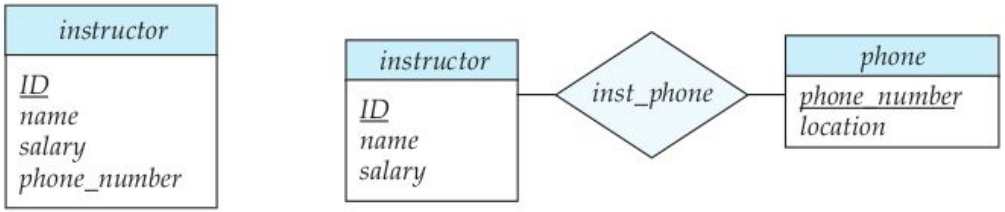
- Identifying relationship sets are redundant
  - So no need to create a new schema
- Many-to-one relationship can be combined with the schema of the ‘Many’ entity set
- One-to-one relationship can be combined with the schema of the either of the entity set
- Foreign Key constraints should be appropriately defined

Design the Schema for the E - R diagram for the University Database (exercise)



E-R Diagram Design issues

- Entity versus Attributes
  - Some attributes can be designed an independent entity set rather than attribute of an entity
  - Some attributes cannot be designed an independent entity set
- How to decide?
  - Depends on the application at hand

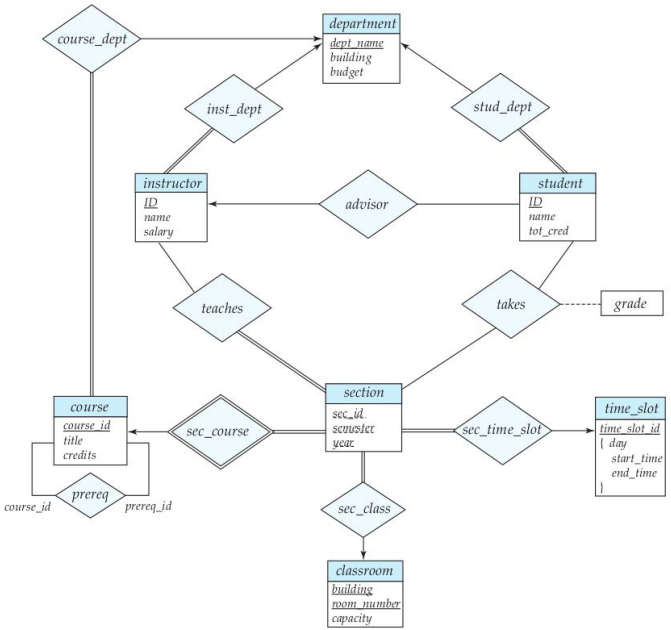


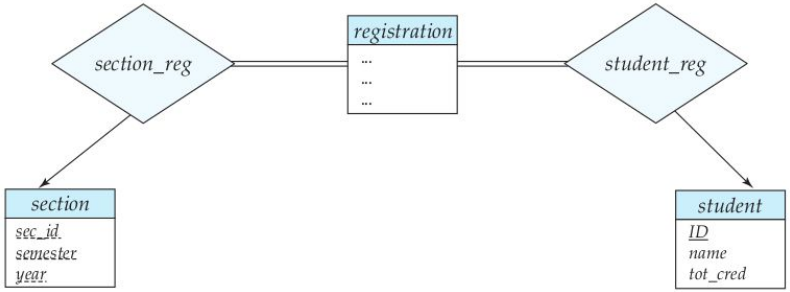
E-R Diagram Design issues

- Common mistakes in encoding relationship sets:
  - Using primary key of one entity set as an attribute of another entity set (instead of relationship)
    - Example: Having instructor ID as attribute of Student (since every student is advised by a single instructor)
  - Making primary key of participating entity sets as attributes of a relationship schema (without making them part of primary key of the relationship schema)
    - Since they are already implicitly used to create the relationship schema

E-R Diagram Design issues

- Entity versus Relationship :
  - When should an object be represented as entity set and when as relationship set?
  - Suppose registration of a student in a section should be recorded:



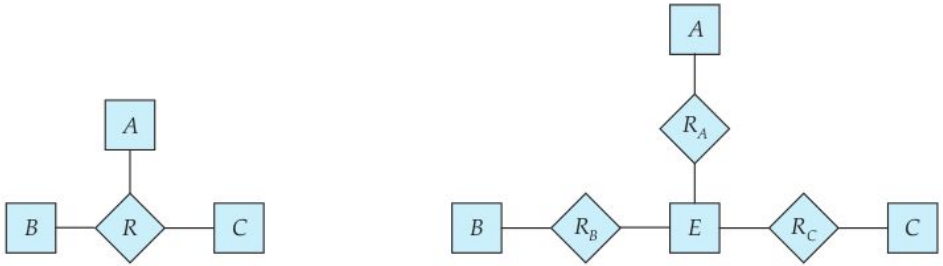


E-R Diagram Design issues

- Entity versus Relationship :
  - Possible guideline:
    - Relationship set designates an ‘action’ that happens between two entity sets

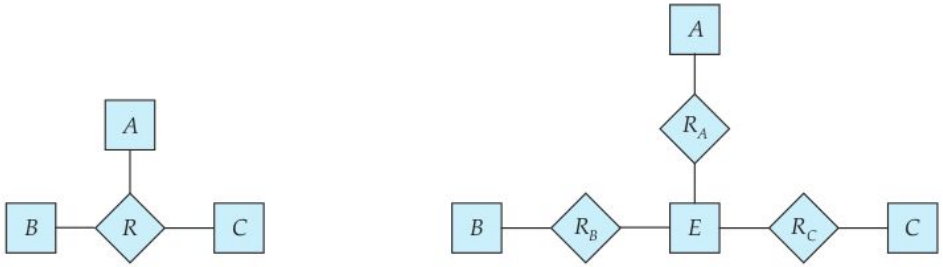
E-R Diagram Design issues

- Binary versus n-ary Relationship sets:
  - Non-binary relationship sets can be described as a combination of several binary relationship sets



E-R Diagram Design issues

- Binary versus n-ary Relationship sets:
  - Should we always restrict to Binary relationships ?
    - n-ary relationship set intuitively shows that more than two entity sets participate in the relationship
    - Increases the number of entity sets and relationship sets in the design
    - Constraints on the n-ary relationship set may not be transferable to binary relationship sets



E-R Diagram Design issues

- Placement of Relationship attribute
  - Attribute of relationship affects where it gets placed in the schema design.
  - Should be carefully designed

Extended E-R Diagrams Features

E-R Diagram extensions

- Basic ER diagram might not express all required features
  - Specialization, Generalization, Higher/Lower level entity sets, Attribute inheritance
  - Aggregation

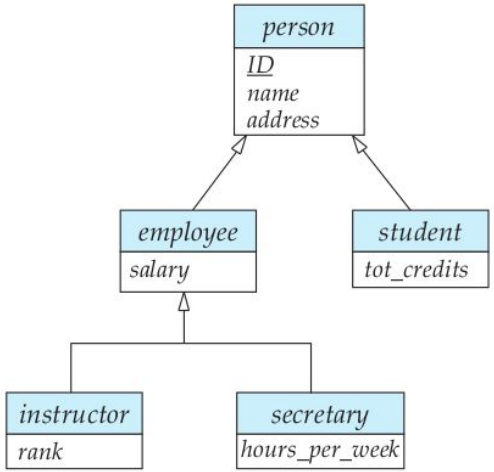
E-R Diagram Design issues : Specialization

- An entity set may include subgroupings of entities that are distinct in some way from other entities in the set
  - Entity set Person may be further classified as student or employee
    Both student and employee inherit attributes of person, but they will have additional attributes

- Designating subgroupings within an entity set is called specialization.

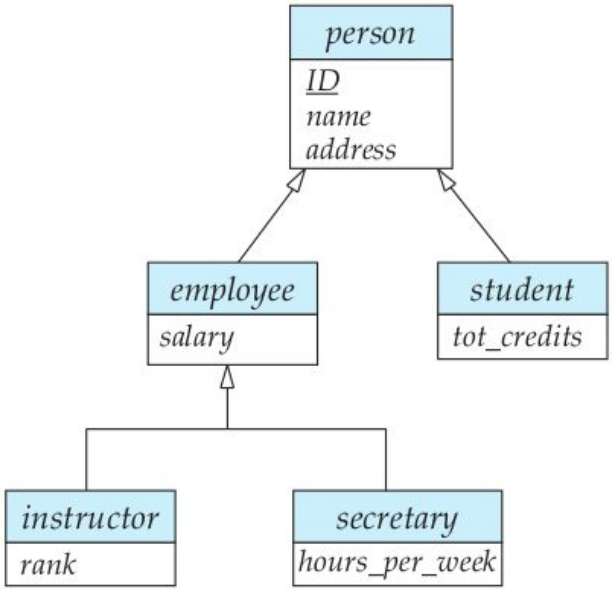
E-R Diagram Design issues : Specialization

- Specialization is applied repeatedly to refine the design
  - Employee is further divided into instructor and secretary



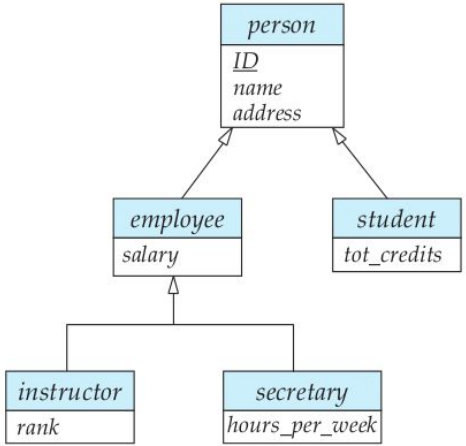
E-R Diagram Design issues : Generalization

- Multiple entities are synthesized into higher level entities (bottom-up)



E-R Diagram Design issues : Attribute inheritance

- Specialization and Generalization leads to attribute inheritance
- Attributes of higher-level are inherited to the lower level
- Lower level entity also inherits the participating relationship sets of the higher level entity



E-R Diagram Design issues : Constraints on Generalization

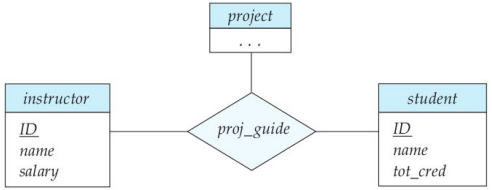
- There can be constraints on which entity of the higher level can be part of which entity set of the lower level
    - Condition defined: Membership is evaluated on the basis of whether or not an entity satisfies an explicit condition or predicate.
    - User defined : User assigns entities to a given entity set
- Another way of classification:
    - Disjoint : An entity belong to no more than one lower-level entity set
    - Overlapping : The same entity may belong to more than one lower-level entity set within a single generalization.

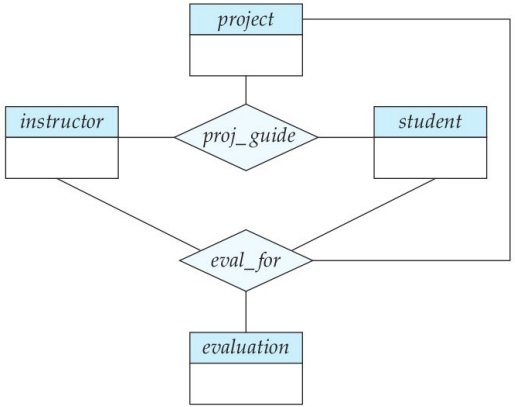
E-R Diagram Design issues : Constraints on Generalization

- Completeness constraint : Whether or not an entity in the higher-level entity set must belong to at least one of the lower-level entity sets within the generalization/specialization.
    - Total : Each higher-level entity must belong to a lower-level entity set
    - Partial : Some higher-level entities may not belong to any lower-level entity set

E-R Diagram Design issues : Aggregation

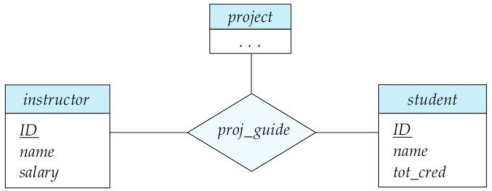
- E-R Diagram cannot express relationship among relationship sets
    - Suppose each project is evaluated by the advisor every month:

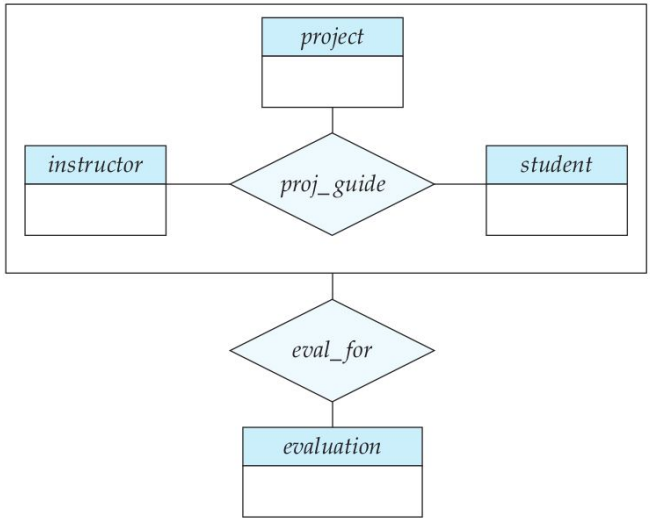




E-R Diagram Design issues : Aggregation

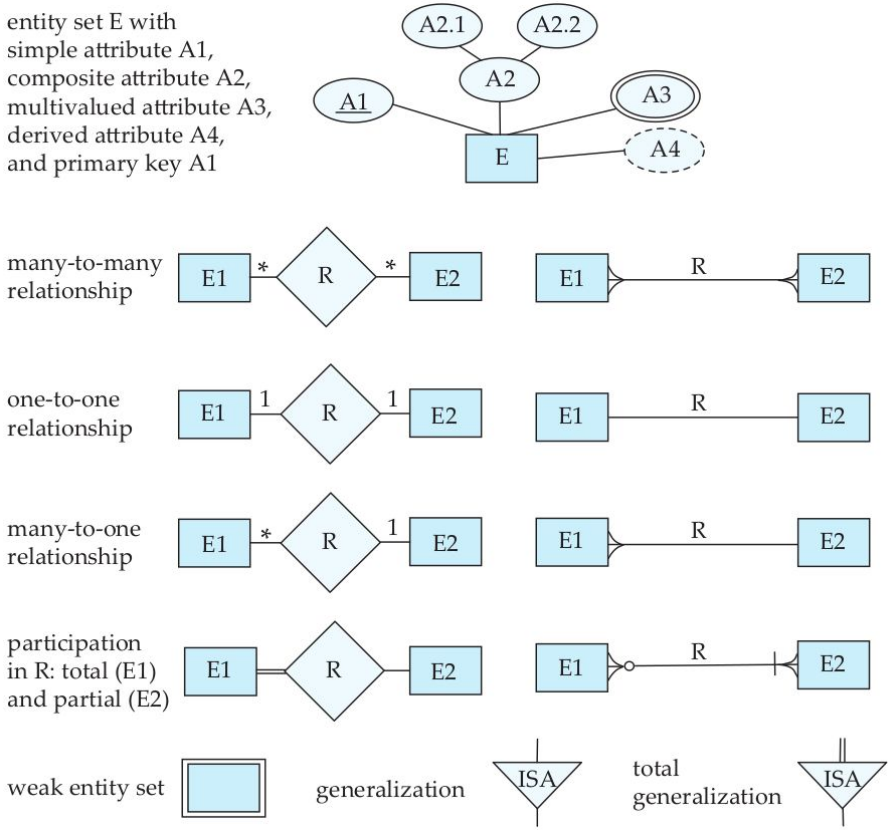
- Involves redundancy
- Solution: Treat relationships as entities in higher level





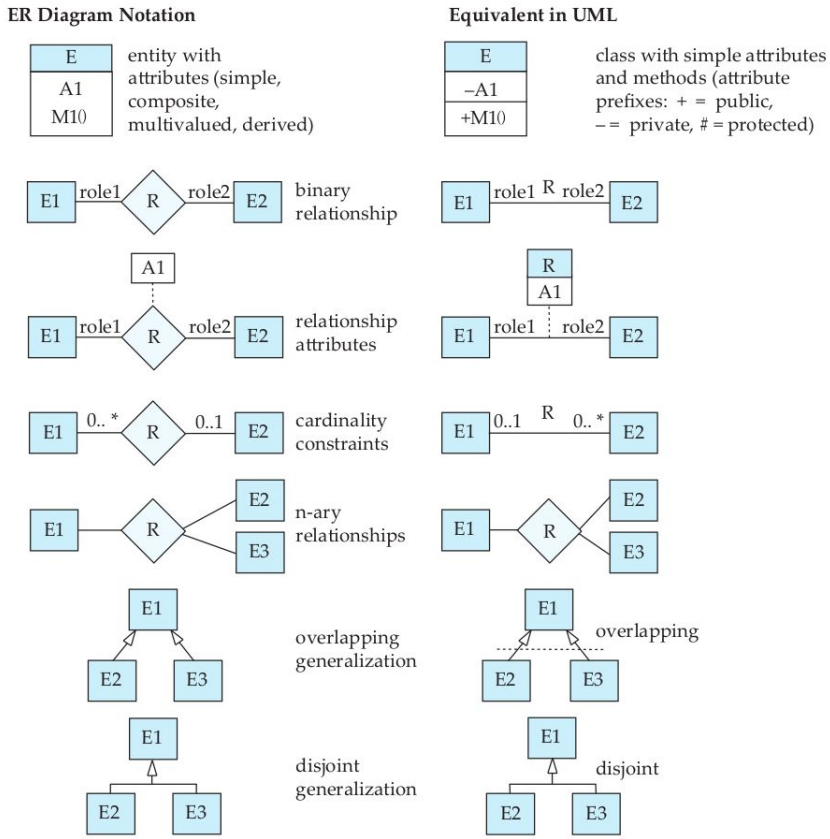
Other Equivalent Representations

E-R Diagram : Alternative notations



E-R Diagram : UML notation

- Unified Modelling Language
  - Class Diagram : Similar to ER Diagram
  - Use case Diagram : Shows interaction between users and the system
  - Activity Diagram : Depicts flow of tasks between various components of the system
  - Implementation Diagram : Depicts components and their interconnections of software and hardware tools



Other Aspects of Database Design

Other Aspects of Database Design

- There are other things to consider apart from schema design
  - Data Constraints: SQL allows many constraints
    (Primary Key, Foreign Key, Unique, Check., Assertion)
  - Usage and Performance Requirements :
    - Throughput : Number of queries that can be processed on average per unit time
    - Response time : amount of time a single transaction takes to complete
  - Authorization Requirements
  - Data Flow and Workflow : Database is a part of larger application
    Database might have to also store these Workflow sequences

Things to consider while designing

- Room for the enterprise to evolve
- Distinction between permanent constraints and constraints that might change
- Ease of conversion of data across schemas
  - Helpful for migration across platforms
- Care for needs and preferences of both end users and database administrators

Reference:

Database System Concepts by Silberschatz, Korth and Sudarshan
Chapter 7