Lecture 14: Entity Relationship Modelling

→ The Entity-Relationship Model
- Entities
- Relationships
- Attributes

→ Constraining the instances
- Cardinalities
- Identifiers
- Generalization

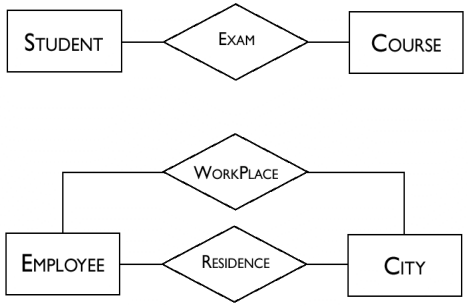
The Entity Relationship Model

→ Entity-Relationship Schema
- Describes data requirements for a new information system
- Direct, easy-to-understand graphical notation
- Translates readily to relational schema for database design
  - But more abstract than relational schema
  - E.g. can represent an entity without knowing its properties
- comparable to UML class diagrams

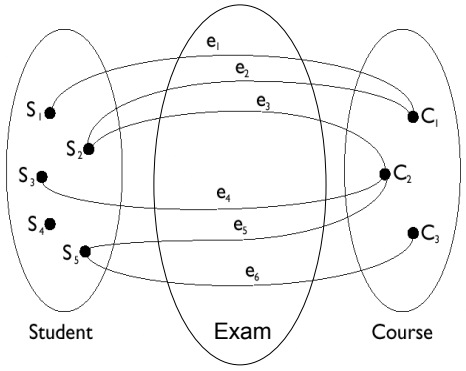
→ Entities:
- classes of objects with properties in common and an autonomous existence
  - E.g. City, Department, Employee, Purchase and Sale
- An instance of an entity is an object in the class represented by the entity
  - E.g. Stockholm, Helsinki, are examples of instances of the entity City

→ Relationships:
- logical links between two or more entities.
  - E.g. Residence is a relationship that can exist between the City and Employee
- An instance of a relationship is an n-tuple of instances of entities
  - E.g. the pair (Johanssen,Stockholm), is an instance in the relationship Residence.

Examples



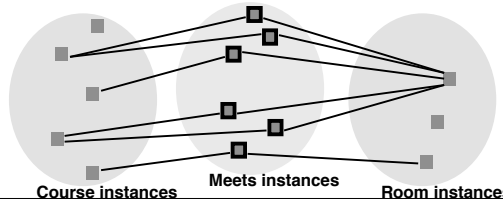
Example Instances for Exam



What Does An E-R Diagram Really Mean?

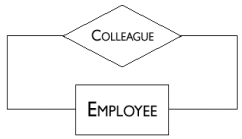
→ Course and Room are entities.
- Their instances are particular courses (eg CSC340F) and rooms (eg MB128)

→ Meets is a relationship.
- Its instances describe particular meetings.
- Each meeting has exactly one associated course and room

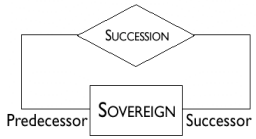


Recursive Relationships

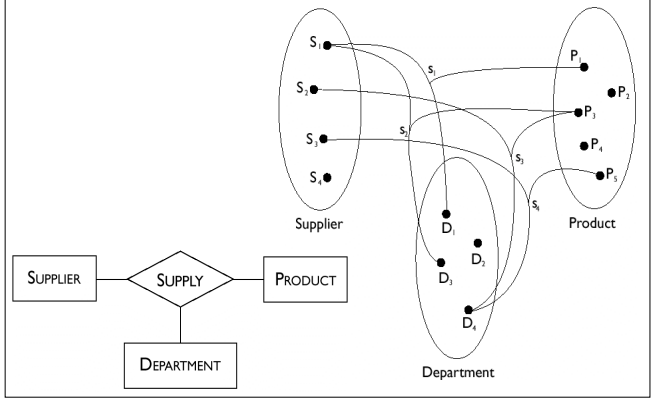
→ an entity can have relationships with itself…



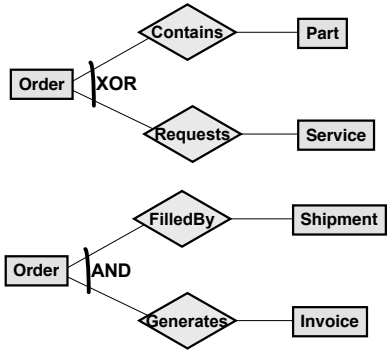
→ If the relationship is not symmetric…
- …need to indicate the two roles that the entity plays in the relationship.



Ternary Relationships



AND/XOR Relationships



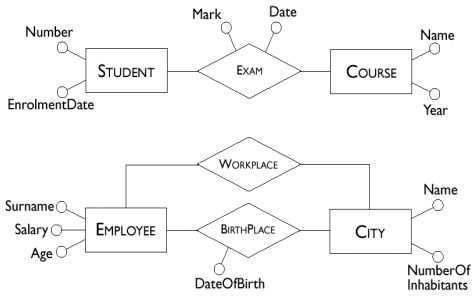
"Each Order either contains a part or requests a service, but not both"

"For any given order, whenever there is at least one invoice there is also at least one shipment and vice versa"

Attributes

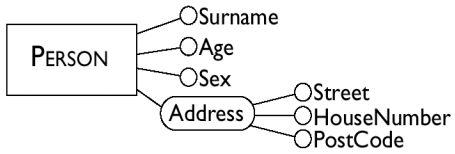
→ associates with each instance of an entity (or relationship) a value belonging to a set (the domain of the attribute).

- The domain determines the admissible values for the attribute.

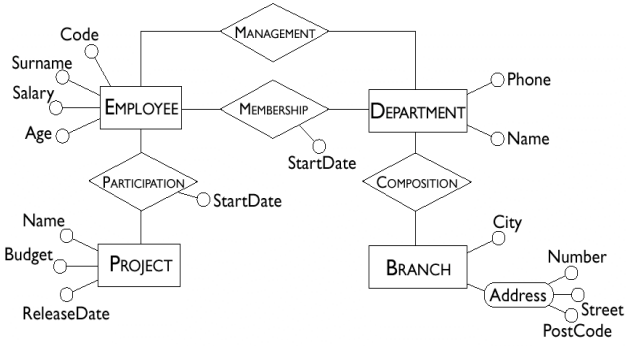


Composite Attributes

→ These group attributes of the same entity or relationship that have closely connected meanings or uses.



Schema with Attributes



Cardinalities

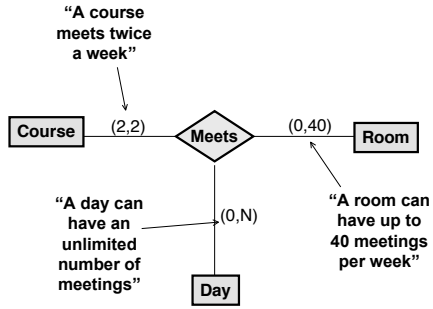
→ Cardinalities constrain participation in relationships

- maximum and minimum number of relationship instances in which an entity instance can participate.
- E.g.

→ cardinality is any pair of non-negative integers (a,b)

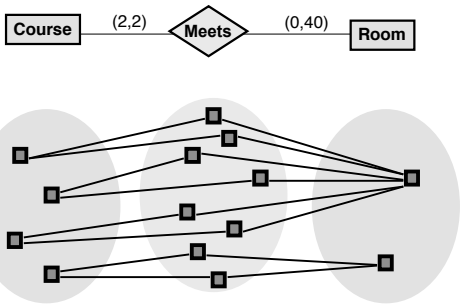
- such that a≤b.
- If a=0 then entity participation in a relationship is optional
- If a=1 then entity participation in a relationship is mandatory.
- If b=1 each instance of the entity is associated at most with a single instance of the relationship
- If b="N" then each instance of the entity is associated with an arbitrary number of instances of the relationship.

Cardinality Example

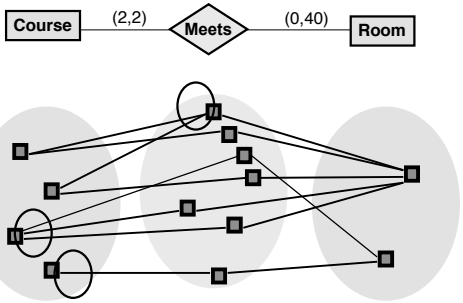


Instantiating ER diagrams

→ An ER diagram specifies what states are possible in the world being modeled

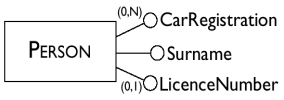


Illegal Instantiations

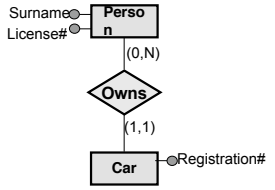


Cardinalities of Attributes

→ Attributes can also have cardinalities
- To describe the minimum and maximum number of values of the attribute associated with each instance of an entity or a relationship.
- The default is (1,1)
- Optional attributes have cardinality (0,1)



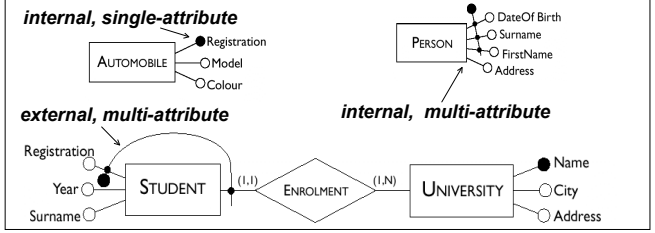
→ Multi-valued attribute cardinalities are problematic
- Usually better modelled with additional entities linked by one-to-many (or many-to-many) relationships



Identifiers (also known as "keys")

→ How to uniquely identify instances of an entity?

- An identifier may formed by one or more attributes of the entity itself
- If attributes of an entity are not sufficient to identify instances unambiguously, other entities can be involved in the identification
- A relationships is identified using identifiers for all the entities it relates
  - E.g. the identifier for the relationship (Person-) Owns(-Car) is a combination of the Person and Car identifiers.



Notes on Identifiers

→ Identifiers and cardinality:

- An identifier can involve one or more attributes, provided that each has (1,1) cardinality
- An external identifier can involve one or more entities, provided that each is a member of a relationship to which the entity to identify participates with cardinality (1,1)

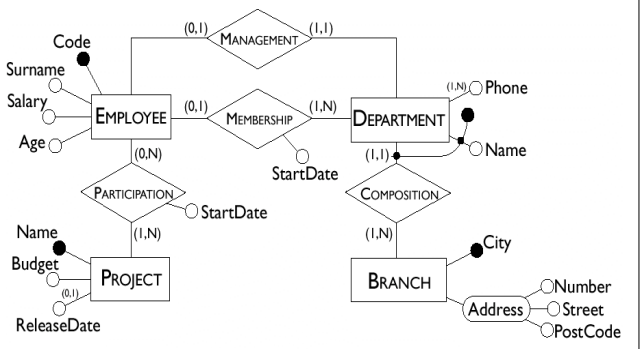
→ Cycles

- An external identifier can involve an entity that is in its turn identified externally, as long as cycles are not generated;

→ Multiple identifiers

- Each entity must have at least one (internal or external) identifier
- An entity can have more than one identifier
  - Note: if there is more than one identifier, then the attributes and entities involved in an identification can be optional (minimum cardinality equal to 0).

Schema with Identifiers



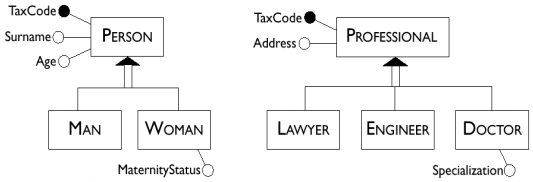
Modeling an Application with Identifiers

→ Identifiers provide an important modelling tool

- E.g. Assume we want a database storing information about lecture meetings.
- If we use the identifier <coursename,day,hour> for the Meeting entity.
  - This says there can only be one meeting at any one time for a given course name, day, hour; we can't have two sections of the same course meeting at the same day+hour.
- If we use only <coursename> as identifier for Meeting.
  - This says that there can only be one meeting per given course name (unreasonable!)
- If we use <courseinstructor,room> as identifier for Meeting
  - we are stating that there can only be one meeting for a given instructor+room combination, so an instructor must have all her meetings in different rooms!
- If we use <courseinstructor> by itself as identifier for Meeting
  - We are stating that each instructor participates in at most one meeting (unreasonable!)

Generalizations

→ Show "is-a" relationships between entities



→ Inheritance:

- Every instance of a child entity is also an instance of the parent entity
- Every property of the parent entity (attribute, identifier, relationship or other generalization) is also a property of a child entity

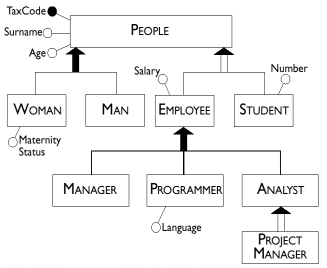
Types of Generalizations

→ Total generalizations:

- ...every instance of the parent entity is an instance of one of its children
- Shown as a solid arrow
- (otherwise: Partial, shown as an unfilled arrow)

→ Exclusive generalizations:

- ...every instance of the parent entity is at most an instance of one of its children
- (otherwise: overlapping)



The E-R Meta-Model (as an E-R Diagram)