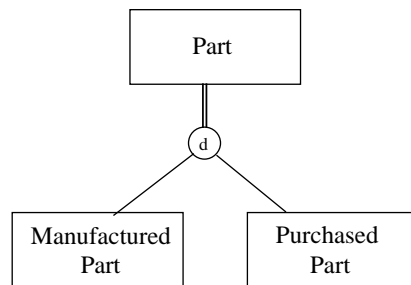


## Ch. 4 – Enhanced ER Model

- more complex requirements than traditional
  - o design
  - o www indexing, etc.
- need extra semantic data modeling concepts

Subclasses/Superclasses/Inheritance (hand out product & airport examples)

- subclass of an entity type
  - o subgrouping by which additional semantics exist for subclass
  - o subclass/superclass relationship – also called IS-A relationship
  - o member entity of subclass represents SAME real-world entity as member of superclass
  - o ex:

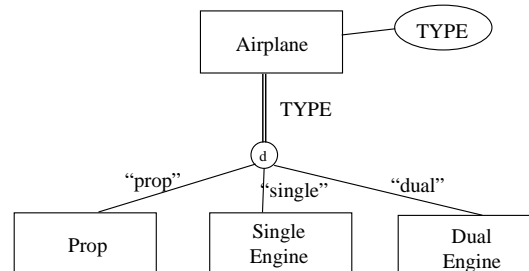


- o subclass member is same entity as superclass member – in a specific role
  - o entity can be member of more than one subclass
    - ex: university DB – grad student is a person, also a student, also an employee
  - o all members of superclass do not need to be members of a subclass
- type inheritance
    - o subclass inherits all attributes and relationship participation from its superclass
  - Specialization / Generalization
    - o Specialization – process of defining set of subclasses of an entity type
    - o Subclasses defined on basis of some distinguishing characteristic of members of the superclass
      - Ex: part relationship – specialization based on whether part was manufactured or bought
      - Use subset symbol to indicate direction of specialization
    - o Subclasses can have:
      - Specific attributes – unit cost / standard lot size
      - Specific relationship types – raw materials made of

- specialization allows us to:
  - define set of subclasses of an entity
  - define specific attributes and relationship types
- Generalization – reverse process
  - Start with distinct entity types and find common features and generalize into single superclass
  - Ex: define entity types car, truck, van – notice that they all have VIN, Price, license#
  - Generalize to form vehicle entity type
- Which way is best?
  - Depends on how the DB is designed – on semantics
  - Many times a combination – start in middle – work both ways

#### Constraints and Characteristics of SP and GE:

- How to determine membership of entity in a subclass:
  - Some specializations use value of an attribute to determine which subclass a member belongs to
    - Predicate defined subclass – membership in a subclass based on the value of a predicate
    - Ex: TYPE = “single”

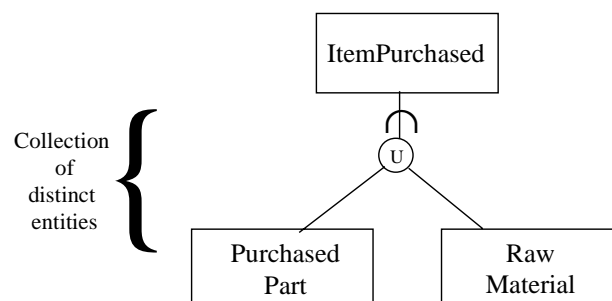


- TYPE = “single”, “prop”, etc. is the defining predicate
- each subclass could have a different predicate, possibly on a different attribute
- Constraint: members of PROP subclass must satisfy the predicate TYPE=“prop”
  - All entities of AIRPLANE whose attrib value of TYPE is “prop” must belong to the subclass
- Attribute-defined specialization – all subclasses have membership condition on same attribute – above example – all on TYPE
- User-defined specialization – no condition specified – up to user to determine how to specialize

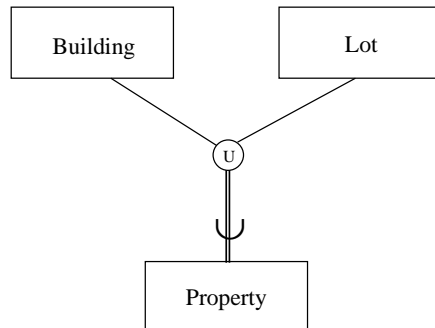
- Disjointness constraint – subclasses of a specialization must be disjoint
  - o Denoted with a “d” in the circle
  - o Each entity can be a member of at most one subclass
  - o If attribute-defined and attribute is single-valued – disjointness is automatic
- Otherwise – overlap can occur – same entity can be in more than one subclass
  - o O in circle (for overlap)
  - o Ex: Univ. DB – secretary subclass of person, student subclass of person – can represent the same entity
- Completeness constraint
  - o Total – every member of superclass must be in a subclass
    - Ex: every airplane has to be one of the subclasses
    - Double-line from superclass to circle denotes this in diagram
  - o Partial – some entities don’t have to belong to a subclass, just superclass
    - Ex: employee entity (custodian, eg) no represented by one of the specializations (secretary, technician, engineer)
- Insertion and deletion rules
  - o Deleting entity from superclass => automatic deletion from all subclasses
  - o Inserting entity in superclass
    - If predicate-defined – automatically inserted in corresponding subclass
    - If total specialization – requires insertion in at least one subclass
- Hierarchy / lattice
  - o Hierarchy – each subclass participates in only one superclass specialization
  - o Lattice – subclass can have more than one superclass
  - o Multiple inheritance – (show student lattice – fig. 4.7)
    - Attributes existing in more than one superclass – inherited only once
    - Some inheritance mechanisms don’t allow mult. Inh.

Union Types – using categories

- Modeling a single superclass/subclass relationship with more than one superclass
  - o superclasses represent different entities
  - o subclass represents collection – subset of the union of distinct entity types
  - o subclass called – union type (or category)
  - o ex: ItemPurchased is a subclass of the collection of entities in the union



- category (union) can be total or partial:
  - o total – every superclass entity must be a member of category subclass
  - o double-line from circle to union category indicates total union subclass
    - ex:



- o partial – some superclass entities do not participate in this union
  - ex: every company is not necessarily an account holder

