Chapter 3

Database Modeling and Design
II. Database Modeling

Dr. Eng. Shady Aly

Data modeling

- A data model is abstract representation تمثيل مجرد of the data on which the IS application is to be based and independent of how the data are stored

Data modeling
نمذجة البيانات

- The focus of this chapter is on the conceptual, or logical design of the database (the design of a data model)

- The data model provides a representation of the entities in the enterprise, the attributes of those entities, and the relationships that exist among entities.

ENTITY-RELATIONSHIP (E-R) MODELING

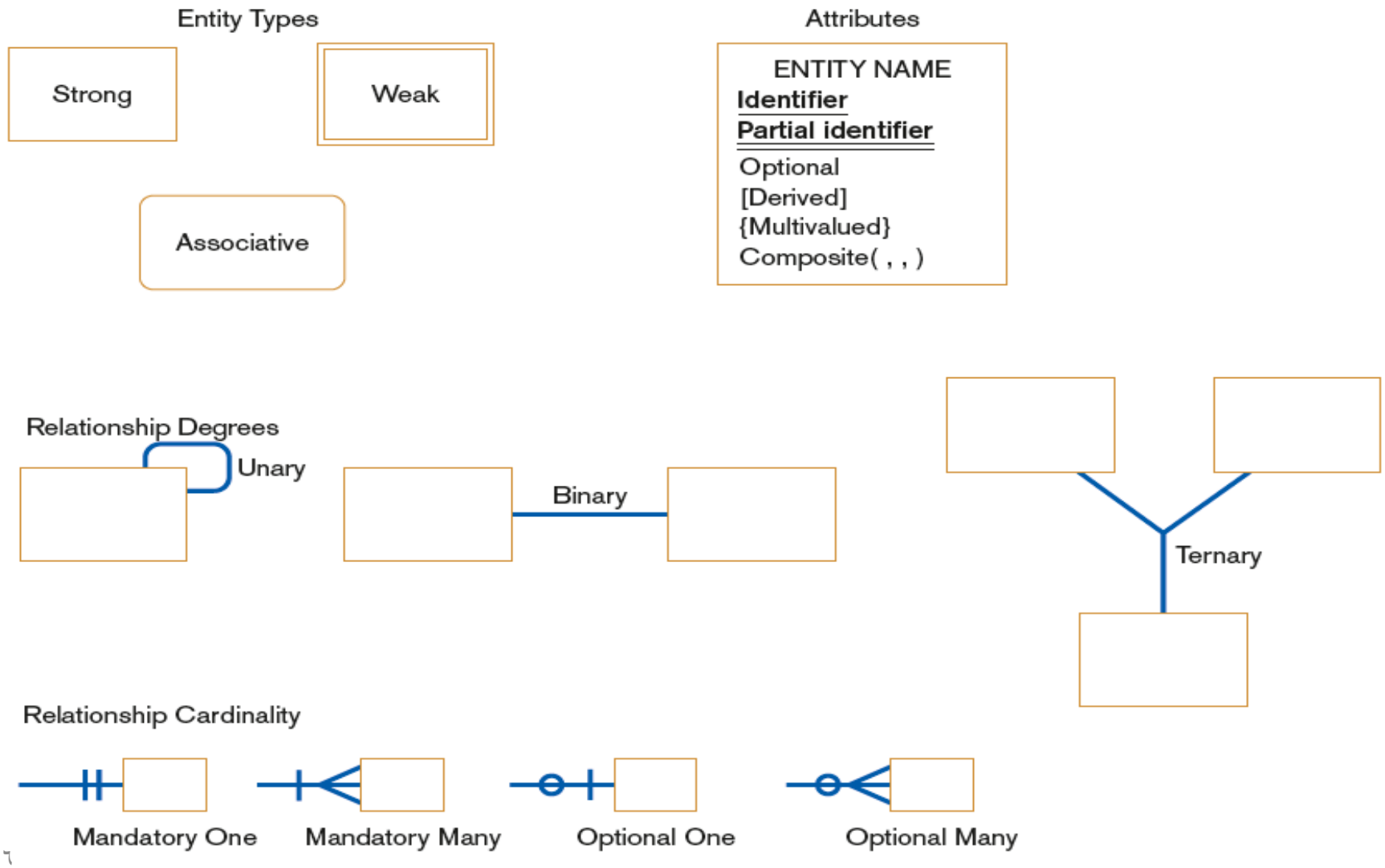
- The purpose of E-R modeling is to design a conceptual schema (model) of entities and their relationships

- The E-R models :
  - facilitate the process of communication among analysts, managers, and users of data
  - design a common data model that will accommodate the different needs of individuals and organizations within the enterprise
  - establish a logical model that can be implemented in a database management system (DBMS).

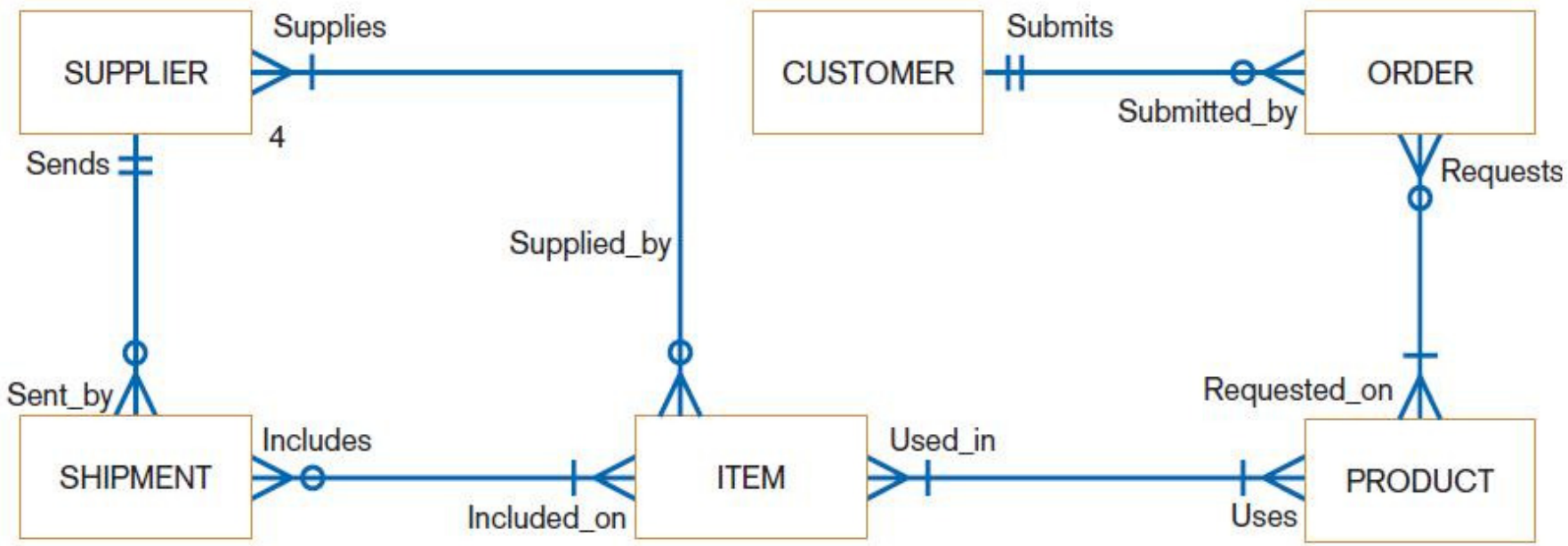
The E-R model

- The E-R model is expressed in terms of:
  - Data entities in the business environment.
  - Relationships or associations among those entities.
  - Attributes or properties (fields) of both the entities and their relationships.

Basic E-R notation



E-R modeling notations



E-R modeling

- **Entity**: a person, place, object, event or concept in the user environment about which data is to be maintained

- **Entity type**: collection of entities that share common properties or characteristics

- **Entity instance**: single occurrence of an entity type

E-R modeling

- **Attribute**: a named property or characteristic of an entity that is of interest to the organization
  - Naming an attribute: i.e. Vehicle_ID
  - Place its name inside the rectangle for the associated entity in the E-R diagram.

- **Candidate key**: an attribute (or combination of attributes) that uniquely identifies each instance of an entity type

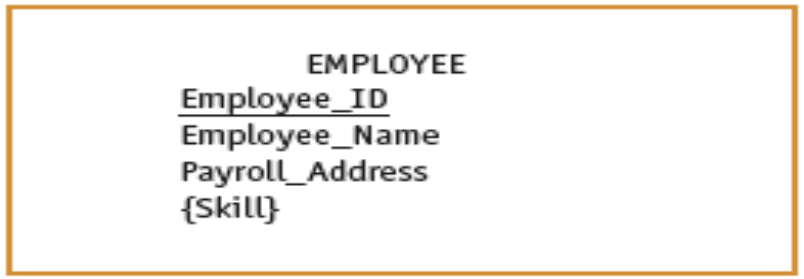
- **Identifier (Primary key)**: a candidate key that has been selected as the unique, identifying characteristic for an entity type

E-R modeling

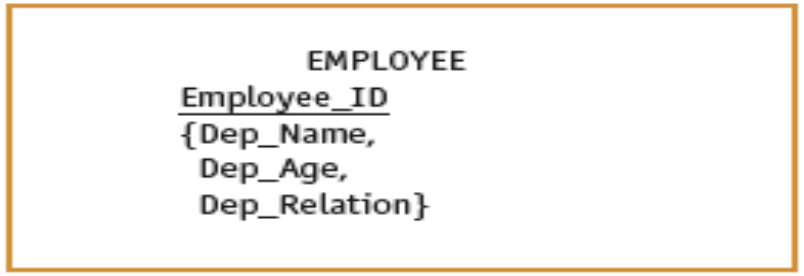
- **Multi-valued attribute**: an attribute that may take on more than one value for each entity instance

- **Repeating group**: a set of two or more multi-valued attributes that are logically related

E-R modeling



(a) Multivalued attribute skill



(b) Repeating group of dependent data

(c) Weak entity for dependent data

E-R modeling

- **Required attribute**: an attribute that must have a value for every entity instance
- **Optional attribute**: an attribute that may not have a value for every entity instance
- **Composite attribute**: an attribute that has meaningful component parts
- **Derived attribute**: an attribute whose value can be computed from related attribute values

E-R modeling

- **Relationship**: an association between the instances of one or more entity types that is of interest to the organization

- **Degree**: the number of entity types that participate in a relationship

- Relationship degrees:
  - **Unary relationship**: a relationship between the instances of one entity type. Also called a *recursive relationship*
  - **Binary relationship**: a relationship between instances of two entity types. Most common type of relationship encountered in data modeling
  - **Ternary relationship**: a simultaneous relationship among instances of three entity types

E-R modeling

- **Cardinality**: the number of instances of entity B that can (or must) be associated with each instance of entity A

- **Minimum** Cardinality
  - The minimum number of instances of entity B that may be associated with each instance of entity A

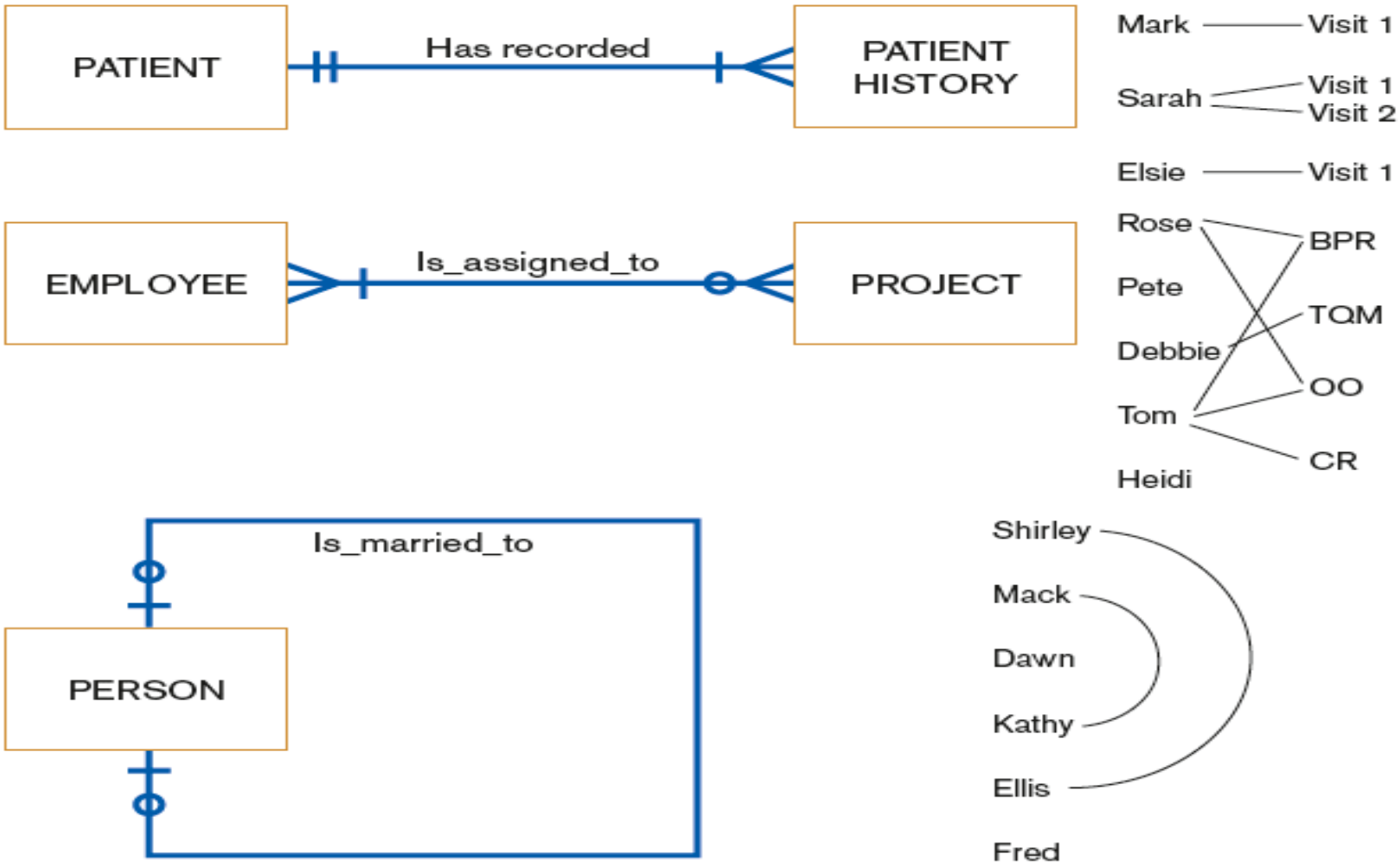
- **Maximum** Cardinality
  - The maximum number of instances of entity B that may be associated with each instance of entity A

E-R modeling

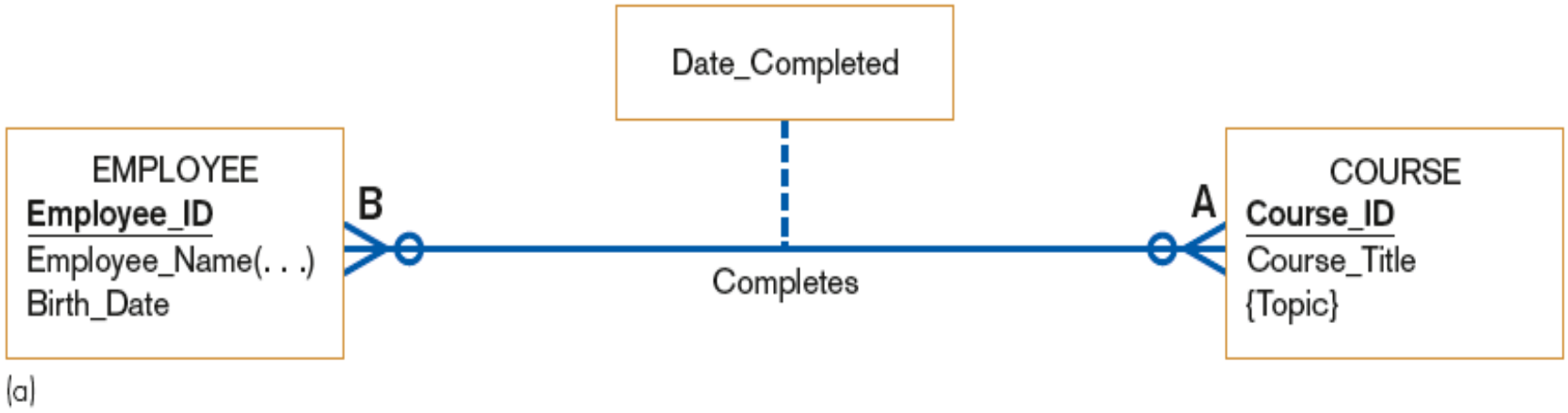
- **Mandatory vs. Optional** Cardinalities
  - Specifies whether an instance must exist or can be absent in the relationship.

- **Associative Entity**: an entity type that associates the instances of one or more entity types and contains attributes that are peculiar to the relationship between those entity instances
  - Sometimes called a gerund
- The data modeler chooses to model the relationship as an entity type.

E-R modeling


PATIENT
Has recorded
PATIENT HISTORY
Mark
Visit 1
Sarah
Visit 1
Visit 2
Elsie
Visit 1
EMPLOYEE
Is_assigned_to
PROJECT
Rose
Pete
Debbie
Tom
Heidi
BPR
TQM
OO
CR
Is_married_to
PERSON
Shirley
Mack
Dawn
Kathy
Ellis
Fred

E-R modeling

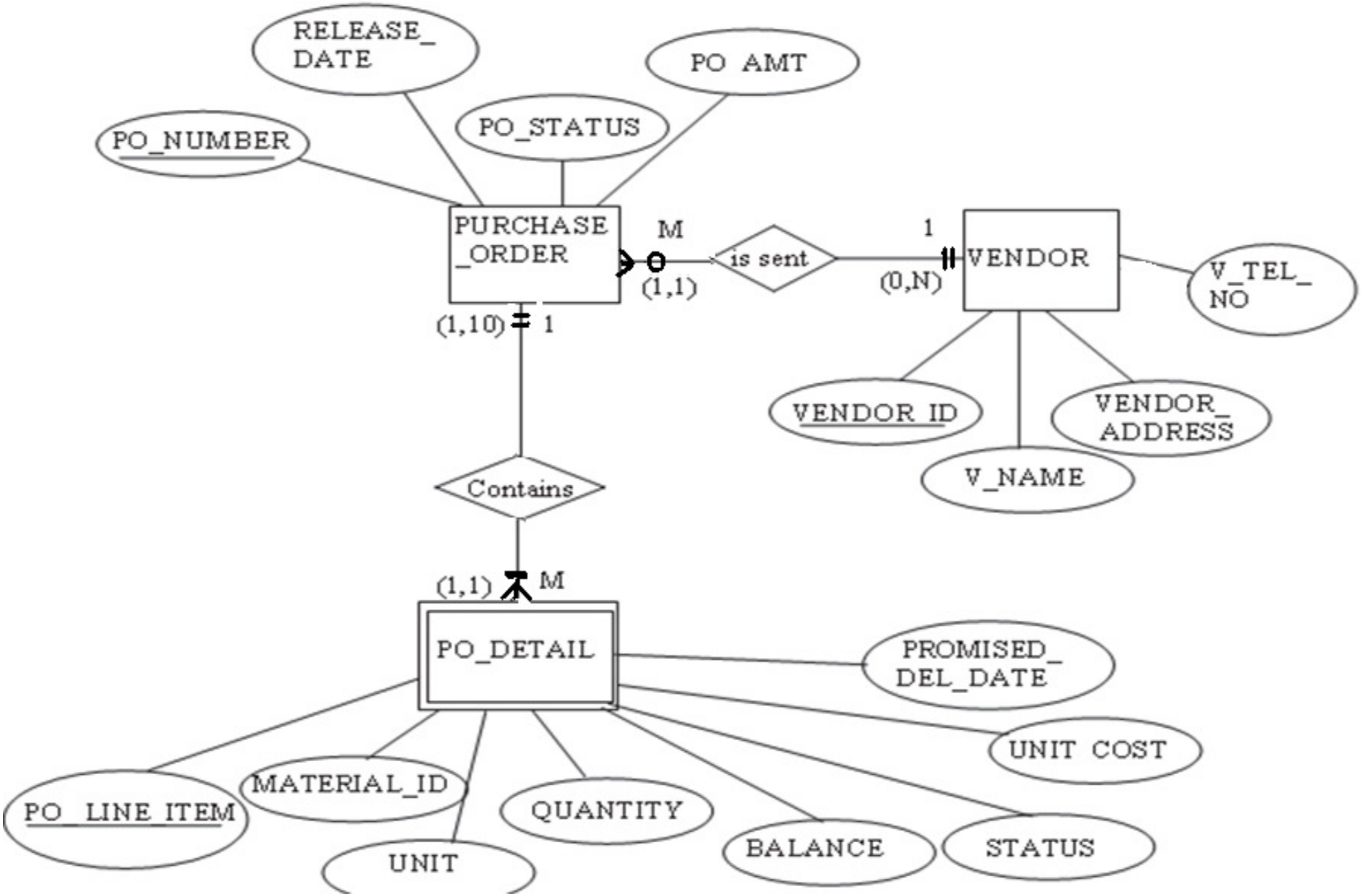


(a)

(b)

(c)

E-R MODELING PRIMITIVES example



E-R MODELING PRIMITIVES example

- The E-R diagram uses a box to represent an entity set (PURCHASE_ORDER, PO_DETAIL, and VENDOR)

- E-R diagrams distinguish between weak and strong entities. An entity is ***weak if its existence*** is dependent on the existence of another entity. An example of this occurs in the case of PO_DETAIL. PO_DETAIL is dependent on the existence of PURCHASE_ORDER.

E-R MODELING PRIMITIVES e3xample – Cont.

- A **weak entity** is sometimes referred to as a *child in a parent-child relationship.*
- *The primary key of the weak (child)* entity is at least in part derived from the parent entity.
- A weak entity is shown in an E-R diagram by using a double-sided rectangle for the entity.
- A **strong entity** is an entity that its existence is not dependent on some other entity, and is represented by a single-sided rectangle.

E-R MODELING PRIMITIVES – Cont.

- The attributes are represented by ovals with a line joining them to the entity to which they apply

- Primary keys are indicated by underlining the attribute name (PO_NUMBER is the primary key attribute of the entity PURCHASE_ORDER, and VENDOR_ID is the primary key for VENDOR.

- foreign keys are not shown (VENDOR_ID is the foreign key that links PURCHASE_ORDER to the entity VENDOR)

E-R MODELING PRIMITIVES – Cont.

- When tables are implemented, a table must contain, as a foreign key, the primary key of any table to which it is related.

- It is unnecessary to show the attribute VENDOR_ID with the entity PURCHASE_ORDER.

- VENDOR_ID will be inherited by PURCHASE_ORDER when the tables are created.

- The entity PO_DETAIL will inherit the attribute PO_NUMBER from PURCHASE_ORDER, as part of its composite primary key (because PO_DETAIL is a weak entity.

E-R MODELING PRIMITIVES – Cont.

- A composite attribute is an attribute that can be further broken down into more specific attributes (example: VENDOR_ADDRESS in the entity VENDOR, can be subdivided into street, city, state, and zip code).. such a breakdown is necessary.

- An attribute may be described as being either **single valued or multi-valued**.

- A **single-valued** attribute will have only one entry in each instance of that attribute (For example, there is only one purchase order number to be entered in each entry of the attribute PO_NUMBER.)

- Example attribute that have more than one value is like the attribute V_TEL_NO of the entity VENDOR.

E-R MODELING PRIMITIVES- Cont.

- The relationship is a description of the association or link between entities.
- Relationships have three components: connecting arcs, diamonds, and cardinality.
- Arcs connect entities that are related to each other. VENDOR is related to PURCHASE_ORDER and PURCHASE_ORDER is related to PO_DETAIL.
- A verb or verb phrase is used to express the general relationship, and is placed in a diamond on the arc. The phrases "VENDOR *is sent PURCHASE_ORDER" and "PURCHASE_ORDER contains PO_DETAIL"*

E-R MODELING PRIMITIVES- Cont.

- Cardinality expresses the number of entity occurrences associated with one occurrence of the related entity. Cardinality is indicated by numbers or symbols written at the ends of the relationship arc.

Three cardinalities are common on E-R diagrams:

One-to-many: (1 : M)

One-to-one: (1 : 1)

Many-to-many: (M : N)

E-R MODELING PRIMITIVES- Cont.

- The cardinality 1 : M means that each instance of one entity is associated with zero, one, or many instances of the entity on the M side of the relationship.

“One vendor is sent 0,1 or many purchase orders.”

“One purchase order contains 0,1 or many purchase order details.”

E-R MODELING PRIMITIVES- Cont.

- In general, the one-to-many relationship is common

- The many-to-many relationship is difficult to handle in a database query, is usually expanded into a series of one-to-many relationships in the final database tables.

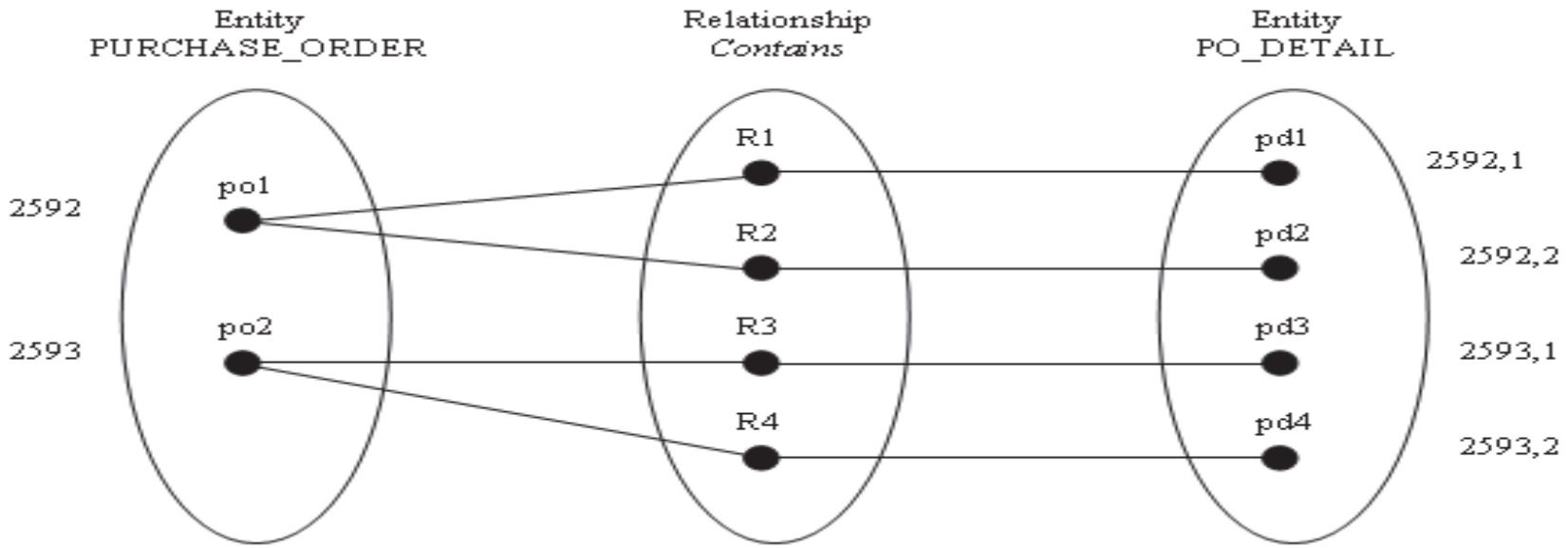
E-R MODELING PRIMITIVES- Cont.

- A useful graphical tool that assists in visualizing the cardinality of a relationship is called a semantic net diagram.

- If instances of one entity connect to many instances of another entity, it indicates the existence of a 1 : M relationship.

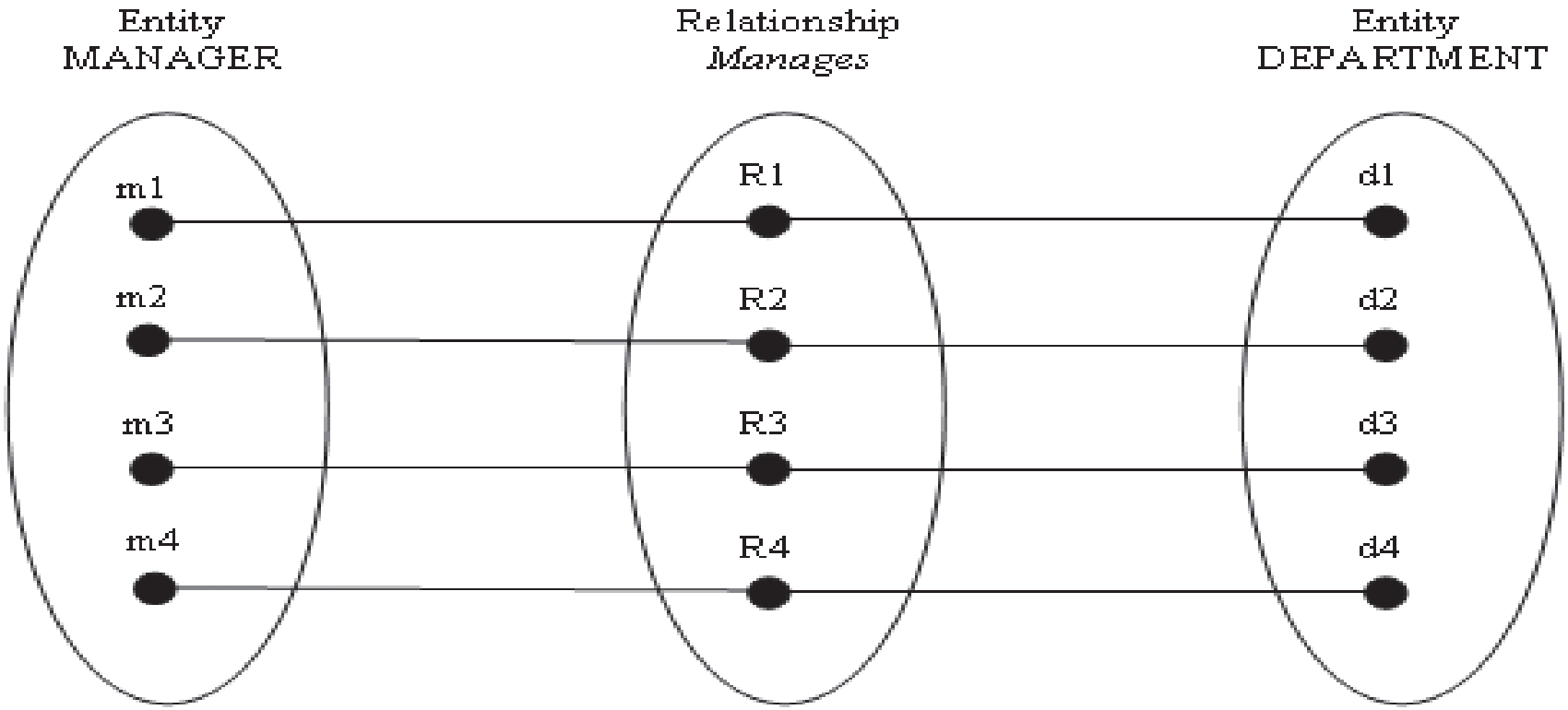
E-R MODELING PRIMITIVES- Cont.

- A diagram for the relationship between PURCHASE_ORDER and PO_DETAIL.



1 : M relationship in a semantic net diagram.

E-R MODELING PRIMITIVES- Cont.



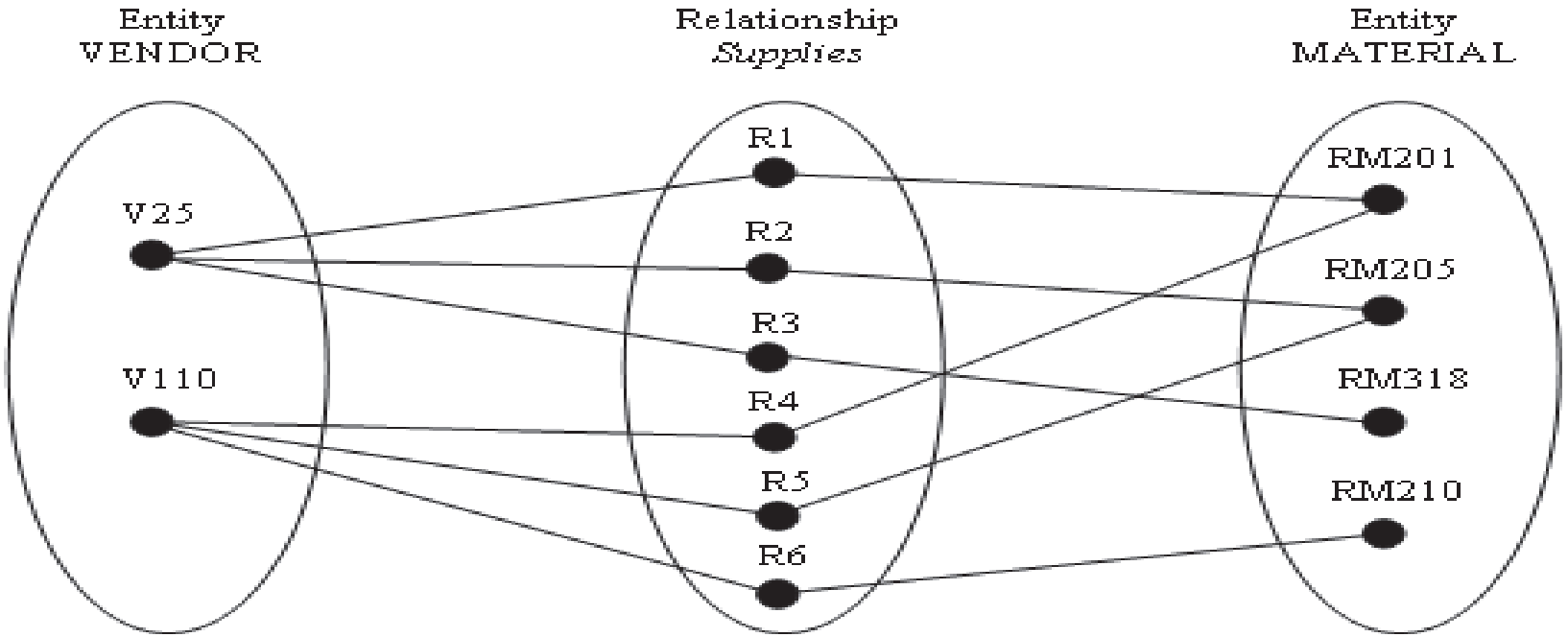
- 1 : 1 relationship in a semantic net diagram

E-R MODELING PRIMITIVES- Cont.

- In an M : N relationship, the number of instances of a relationship is greater than the number of related instances in an entity set

- Consider the relationship between two entities: VENDOR and MATERIAL. A vendor *supplies material to a company. One vendor may supply more than one* material. Also, a specific material may be supplied by more than one vendor.

E-R MODELING PRIMITIVES- Cont.



- M : N Relationship in a semantic net diagram.

E-R MODELING PRIMITIVES- Cont.

- Cardinality limits are shown as numbers in parentheses.

- In the relationship between PURCHASE_ORDER and PO_DETAIL, the cardinality limits for PO_DETAIL are (1,1), which means that each PO_DETAIL will be associated with one purchase order.

- Similarly, the cardinality limits of PURCHASE_ORDER is (1,10), which means that there will be a minimum of 1 and a maximum of 10 line items associated with one purchase order.

E-R MODELING PRIMITIVES- Cont.

- Cardinality limits are used based on the way in which the business is operated (business rules of the enterprise)

- Business rules also affect the degree of an entity's participation in a relationship.

E-R MODELING PRIMITIVES- Cont

- The optional entity is shown by placing a single circle near the optional entity.

- A relationship is mandatory (إلزامية) if one entity's occurrence requires the participation of the other entity in the relationship. This is like the relationship between PURCHASE_ORDER and PO_DETAIL.

E-R MODELING PRIMITIVES- Cont

- A cross line arc is placed next to Employee cardinality one to indicate that the participation is mandatory in the relationship.

THE DEGREE OF A RELATIONSHIP

- The degree of a relationship is a measure of the number of entities sharing the same association

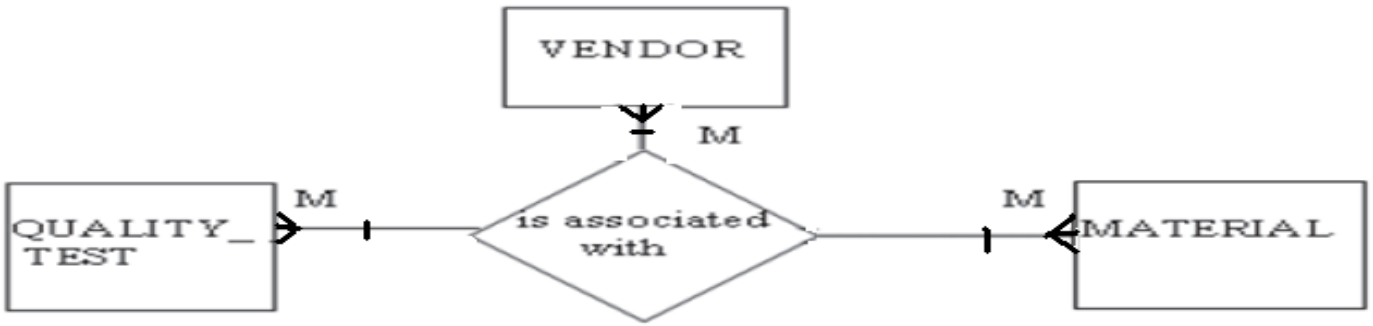
- There are four cases: binary relationships, ternary relationships, n-ary relationships, and unary relationships

- The binary relationship is most common, and it exists when two entities have an associated relationship. For example, VENDOR and PURCHASE_ORDER share a binary relationship, as does PURCHASE_ORDER and PO_DETAIL.

THE DEGREE OF A RELATIONSHIP- Cont.

- A *ternary relationship exists when three entities share a common relationship.*

- Example: different vendors may require different inspection Levels (such as “minimum,” “normal,” and “restricted”)



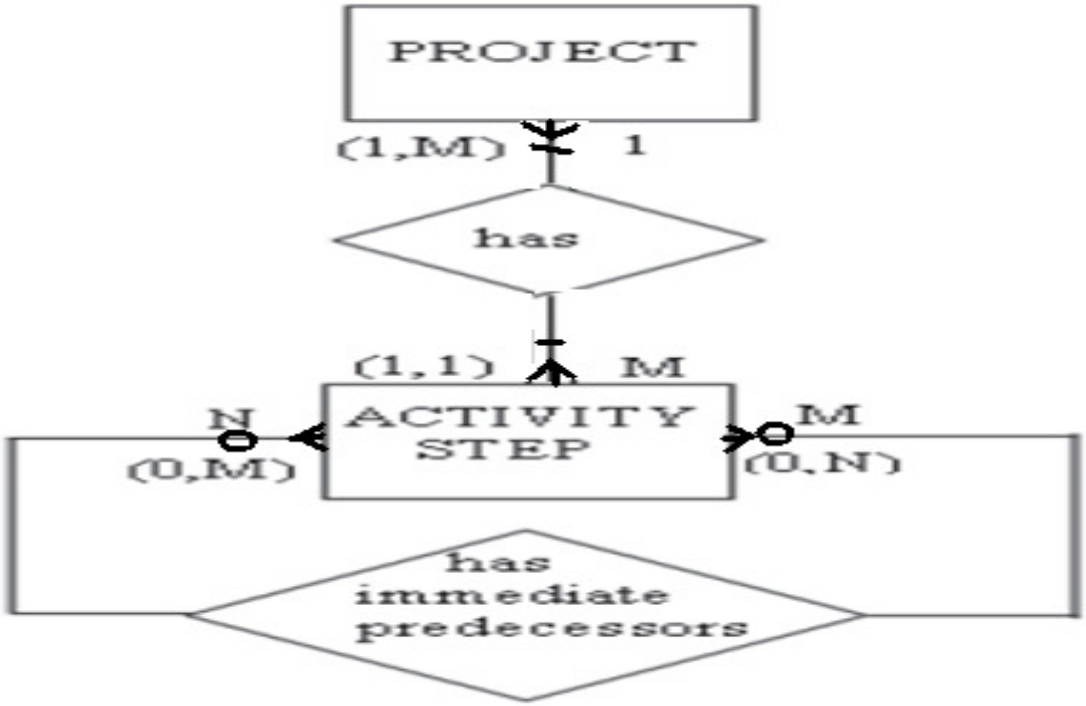
THE DEGREE OF A RELATIONSHIP- Cont.

- When more than three entities share a relationship, it is known as n-ary. This situation seldom occurs and can be ignored for our purposes.

- A *unary relationship exists when an association is maintained within a single entity (i.e., the* entity has a relationship with itself (recursive relationship)).

THE DEGREE OF A RELATIONSHIP- Cont.

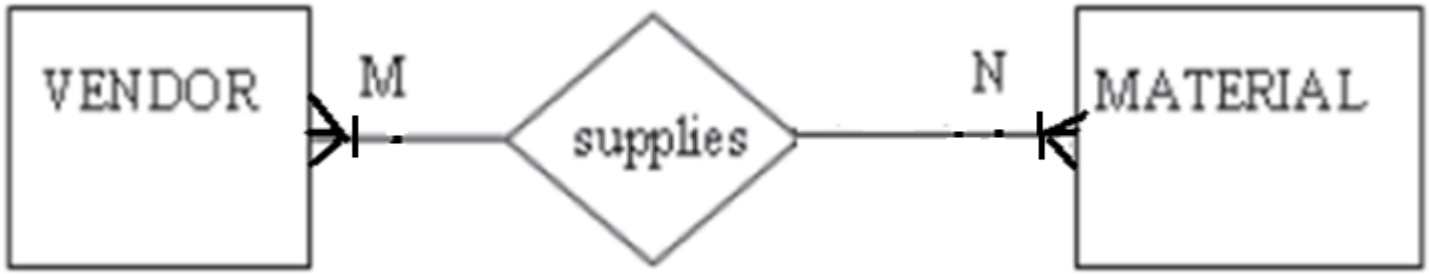
- A project is defined as a sequence of activity steps.
- These activity steps are sequential tasks that must be performed in order to complete the entire project.



COMPOSITE ENTITIES

- In the course of developing an E-R diagram, M : N relationships are often encountered.

- When this happens, the analyst can break down the M : N relationship into a series of 1 : M relationships using a **composite entity.**



COMPOSITE ENTITIES

Entity Set: VENDOR

VENDOR ID	V NAME	V STREET	V CITY	V STATE	V ZIP
V110	Jersey	2 Main St.	Patterson	NJ	07055
V25	General	125 Common	Boise	ID	44830
V250	Spices	25 Salty Lane	East Hampton	NY	10027
V75	Pasta Supply,	34 Henry St.	Philadelphia	PA	09098

Entity Set: MATERIAL

MATERIAL ID	MATL DESCRIPTION
RM201	Carrots, whole
RM202	Carrots, diced, 1/4 inch
RM205	Potatoes, Eastern,
RM210	Peas, shelled
RM211	Tomatoes, whole
RM310	Garlic, whole
RM311	Garlic powder
RM318	Salt, iodized
RM308	Onion salt
RM305	Paprika
RM340	Sugar, bulk
RM805	Olive oil
RM810	Vinegar, white

COMPOSITE ENTITIES

Entity Set: VENDOR

VENDOR ID	V NAME	V STREET	V CITY	V STATE	V ZIP
V110	Jersey	2 Main St.	Patterson	NJ	07055
V25	General	125 Common	Boise	ID	44830
V250	Spices	25 Salty Lane	East Hampton	NY	10027
V75	Pasta Supply,	34 Henry St.	Philadelphia	PA	09098

Entity Set: MATERIAL

MATERIAL ID	MATL DESCRIPTION
RM201	Carrots, whole
RM202	Carrots, diced, 1/4 inch
RM205	Potatoes, Eastern, whole
RM210	Peas, shelled
RM211	Tomatoes, whole
RM310	Garlic, whole
RM311	Garlic powder
RM318	Salt, iodized
RM308	Onion salt
RM305	Paprika
RM340	Sugar, bulk
RM805	Olive oil
RM810	Vinegar, white

Entity Set: VENDOR_MATL_XREF

VENDOR ID	MATERIAL ID
V25	RM201
V25	RM205
V25	RM318
V25	RM340
V110	RM201
V110	RM202
V110	RM205
V110	RM210
V110	RM211
V250	RM310
V250	RM311
V250	RM318
V250	RM340
V250	RM308
V250	RM305
V25	RM805
V25	RM810

Super-class and sub-class entity types

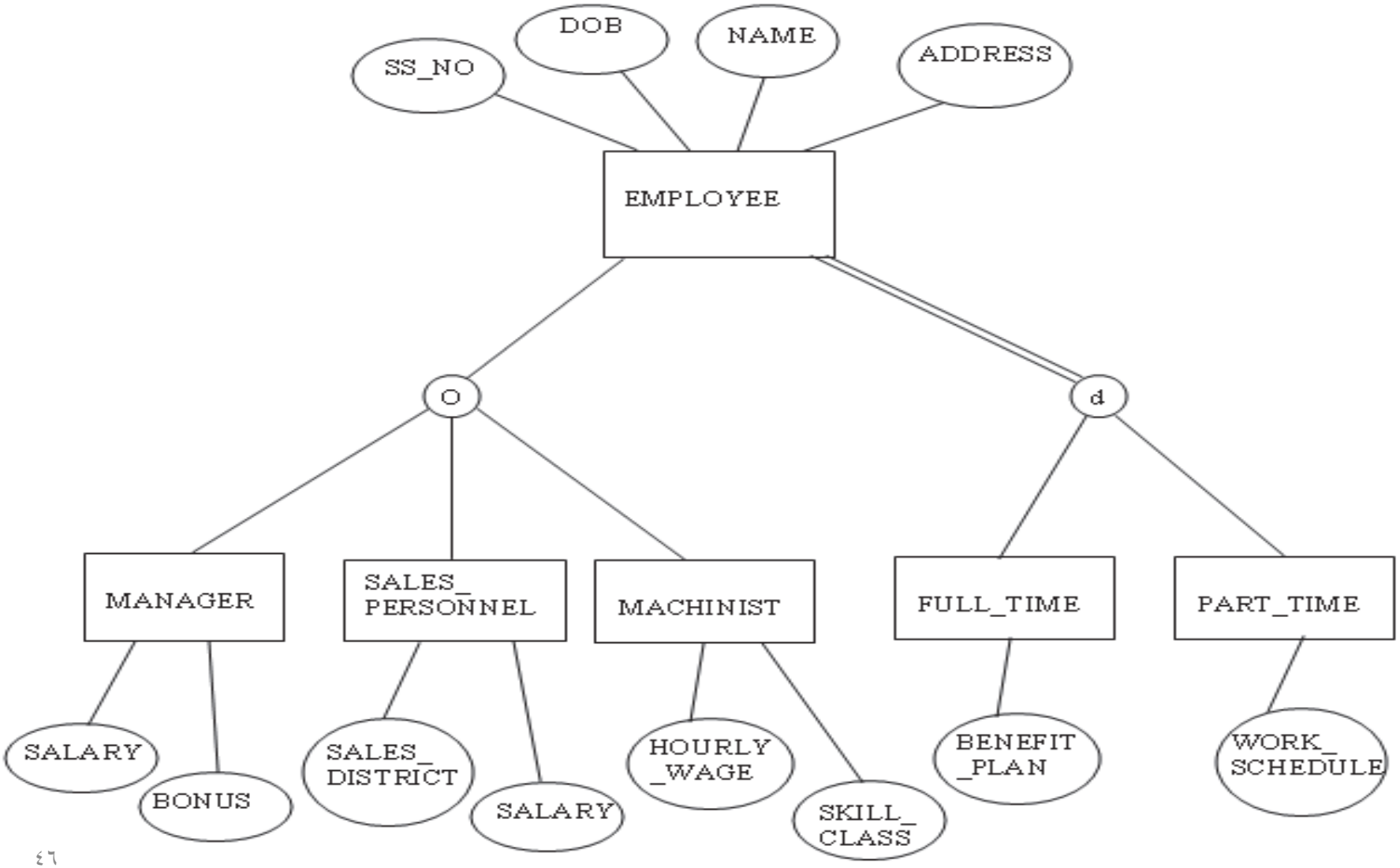
- A super-class is an entity type that represents a set of similar people, places, or things that have common attributes but represent distinct subclasses.

- The subclass is an entity that is a member of a super-class but has distinct attributes that are not in common with all other members of the super-class.

Superclass and subclass entity types

- Super-class and subclass represent the type and subtypes relationship
- The subclass entity types takes all attributes of their of super-class
- Raw materials, subassemblies, final assemblies, office materials, repair and maintenance materials, and so forth are subclass or subtypes of super-class entity called material

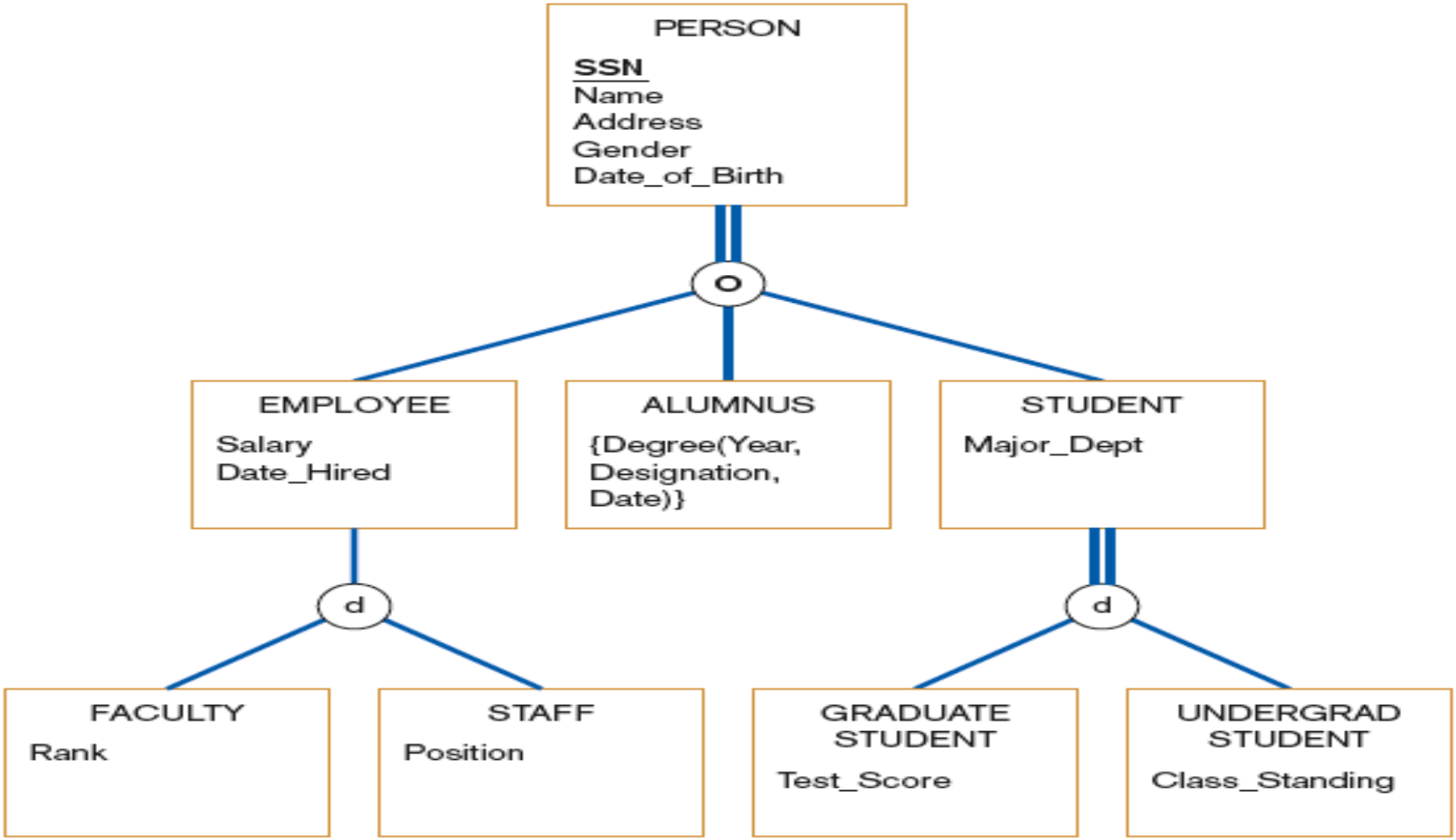
Super-class and subclass entity types



Super-class and subclass entity types

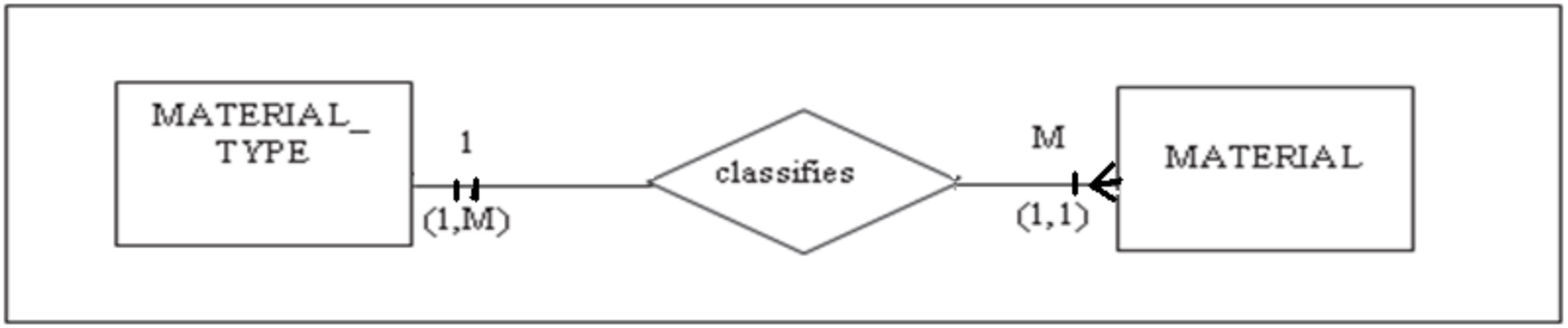
- Employee is superclass can be either FULL_TIME & PART_TIME in one classification.
- Employee is superclass can be also one of or more subclasses of MANAGER & MACHINIST & SALES_PERSONNEL in another classification.
- The code “d” indicates distinct. This means that any instance of the super-class entity may be in one and only one of the subclass entities.
- code “o” indicates “overlapping.” This means that any instance of the superclass entity may be in more than one subclass entity.

Super-class and subclass entity types



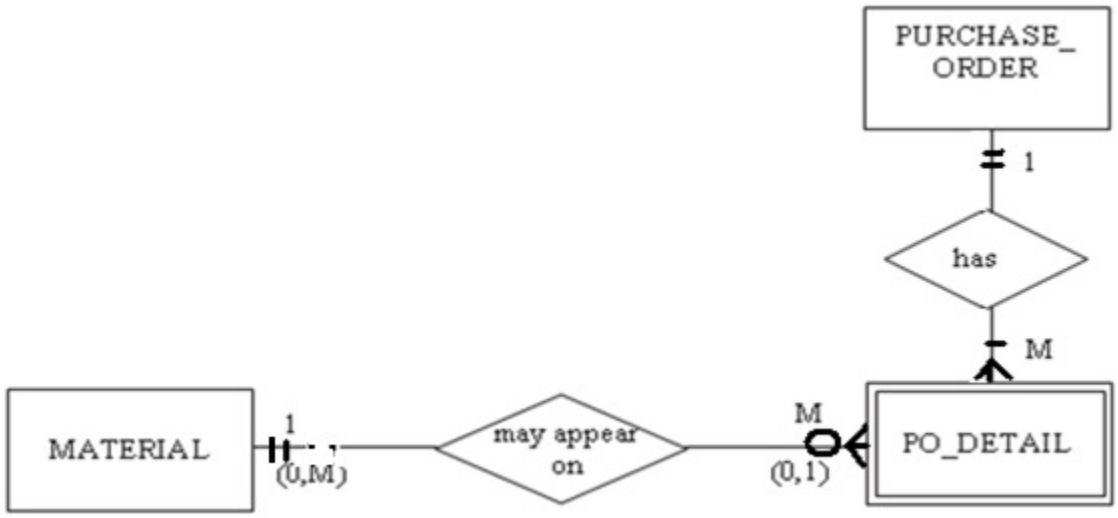
Case study data model (Inventory system)

1. The department is responsible for the inventory of three material types: raw material, finished goods, and supplies. Each material, identified by a material ID, belongs to one material type. Each material type classifies many different materials.



Case study data model –Cont.

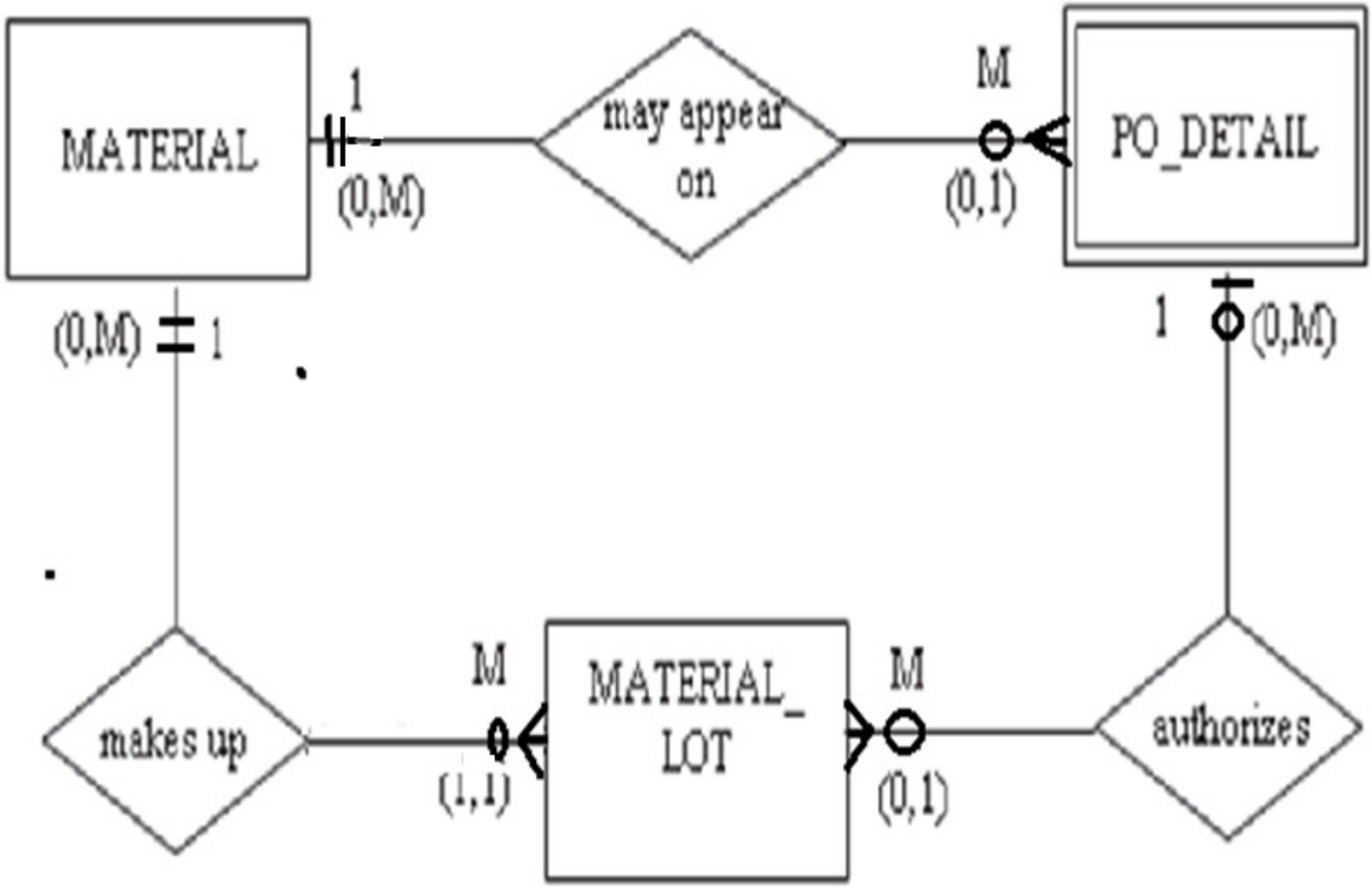
2. When a raw material or supply is ordered, it is done using a purchase order, where it appears on the purchase order detail. Purchase orders are also used to order capital equipment and services.



Case study data model –Cont.

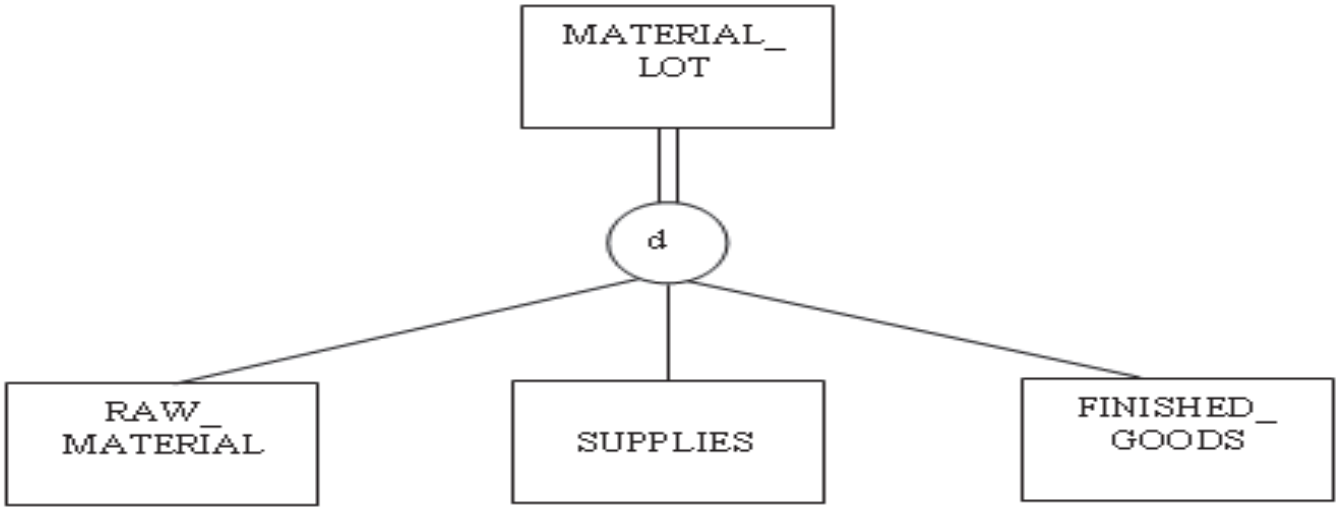
3. When the materials or supplies ordered on a purchase order arrive or when finished goods are produced, they represent MATERIAL. They are called material lots, and each is assigned a unique identification called a material lot number. For each material, there may be zero, one, or many material lots. Each material lot is for one and only one material.

Case study data model –Cont.



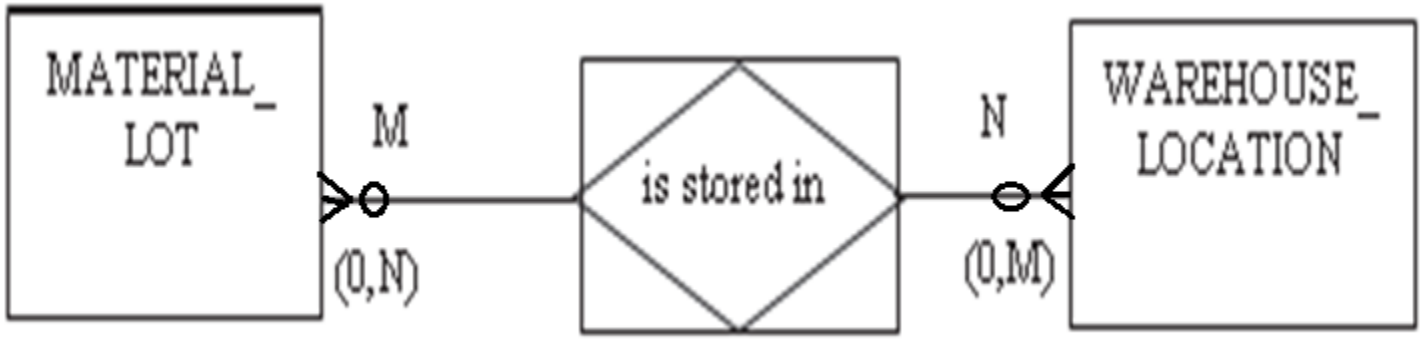
Case study data model –Cont.

4. The entity MATERIAL_LOT includes inventoried raw materials, supplies, and finished goods. These three categories are non-overlapping and each instance of a material lot must appear in one and only one category.



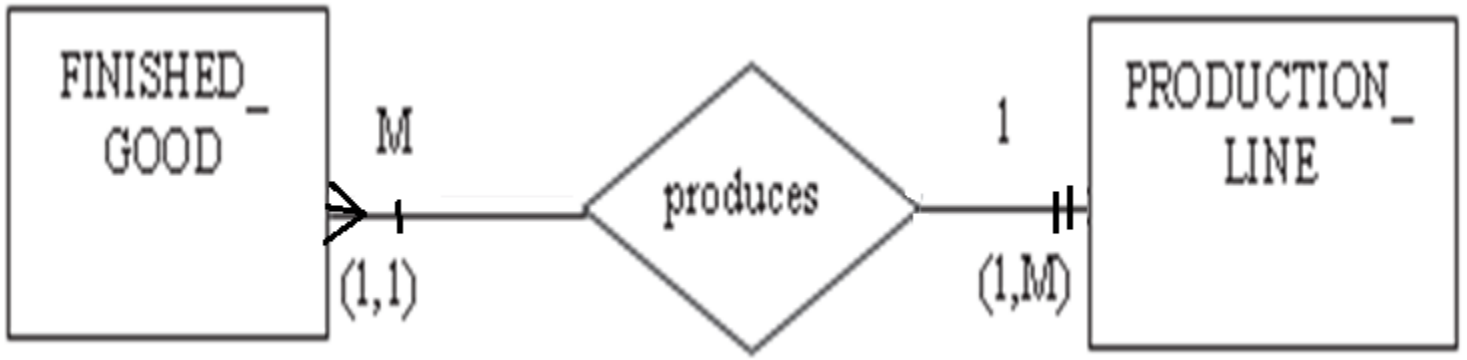
Case study data model –Cont.

5. A material lot is stored in a warehouse location. A warehouse location may store zero, one, or many material lots. If a material lot is too large to be stored in one warehouse location, it may have to occupy more than one location. A material lot may not be assigned to a warehouse location at some points in time.



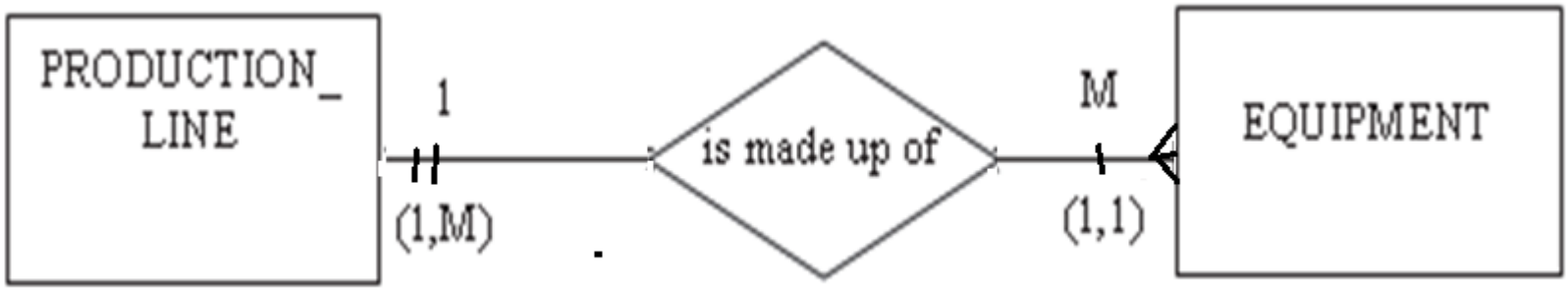
Case study data model –Cont.

6. If a material lot is finished goods, it was produced on a production line. A production line can produce more than one material lot of material, and each lot of finished goods will be produced on one and only one production line.



Case study data model –Cont.

7. A production line is made up of one or more pieces of equipment, each with its own equipment ID. Each piece of equipment belongs to one and only one production line.



Case study data model –Cont.

8. Materials may have quality control tests associated with them. Some raw materials and all supplies are not subject to any quality control tests. Some quality control tests are performed on more than one material, and some materials require more than one quality control test.

-

Case study data model –Cont.