ISE 211 - Industrial Information Systems

Databases and Analysis

Lecture 3 - Chapter 3

Data Modelling (Continue)

İzmir University of Economics

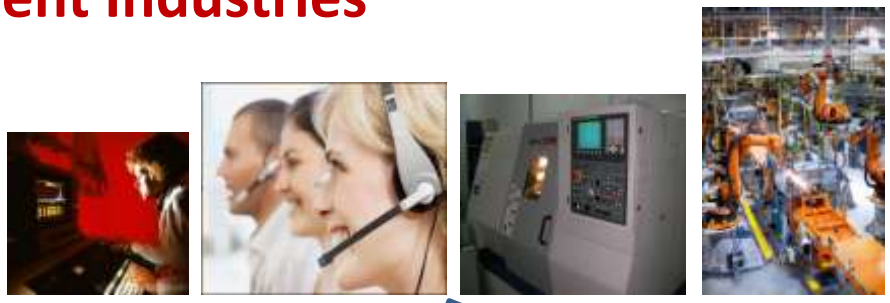
Halil POSACI

2011, İzmir

Agenda

- Summary Ch1
- Summary + Continue Ch3
  - Recursive entities – BOM summary
  - Super & subclass entity types
- Case: Company Database
  - Class work
- DB Mapping (Company DB)
  - Lab work
- HOMEWORK

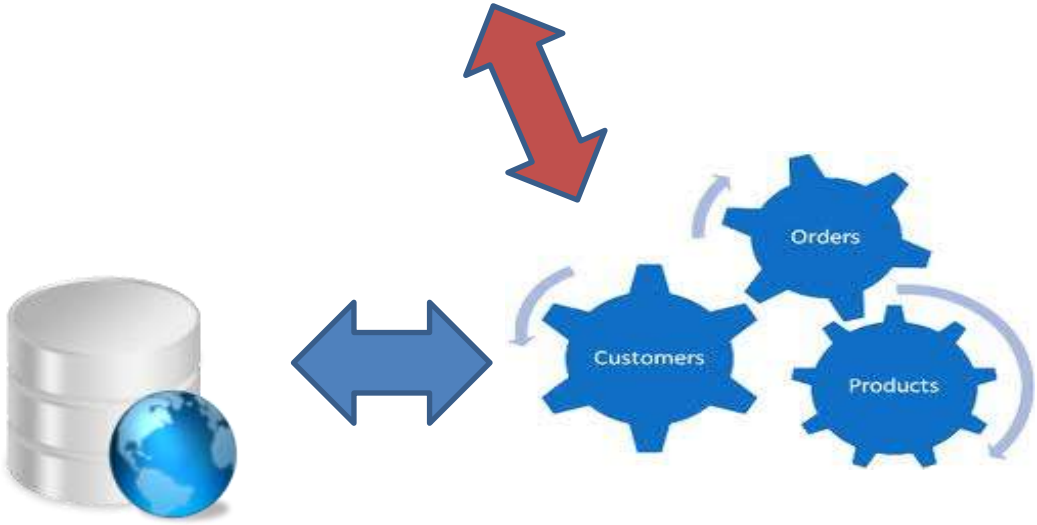
Different Design for Different Industries



ACCESS

DBMS

ORACLE
MS SQL
DB/2 – INFORMIX
SYBASE
MYSQL



Layers Of Software

Business Logic
Software pieces

User /Machine
Interface Layer

Businesss Logic
Layer

DB Layer

Aim Of

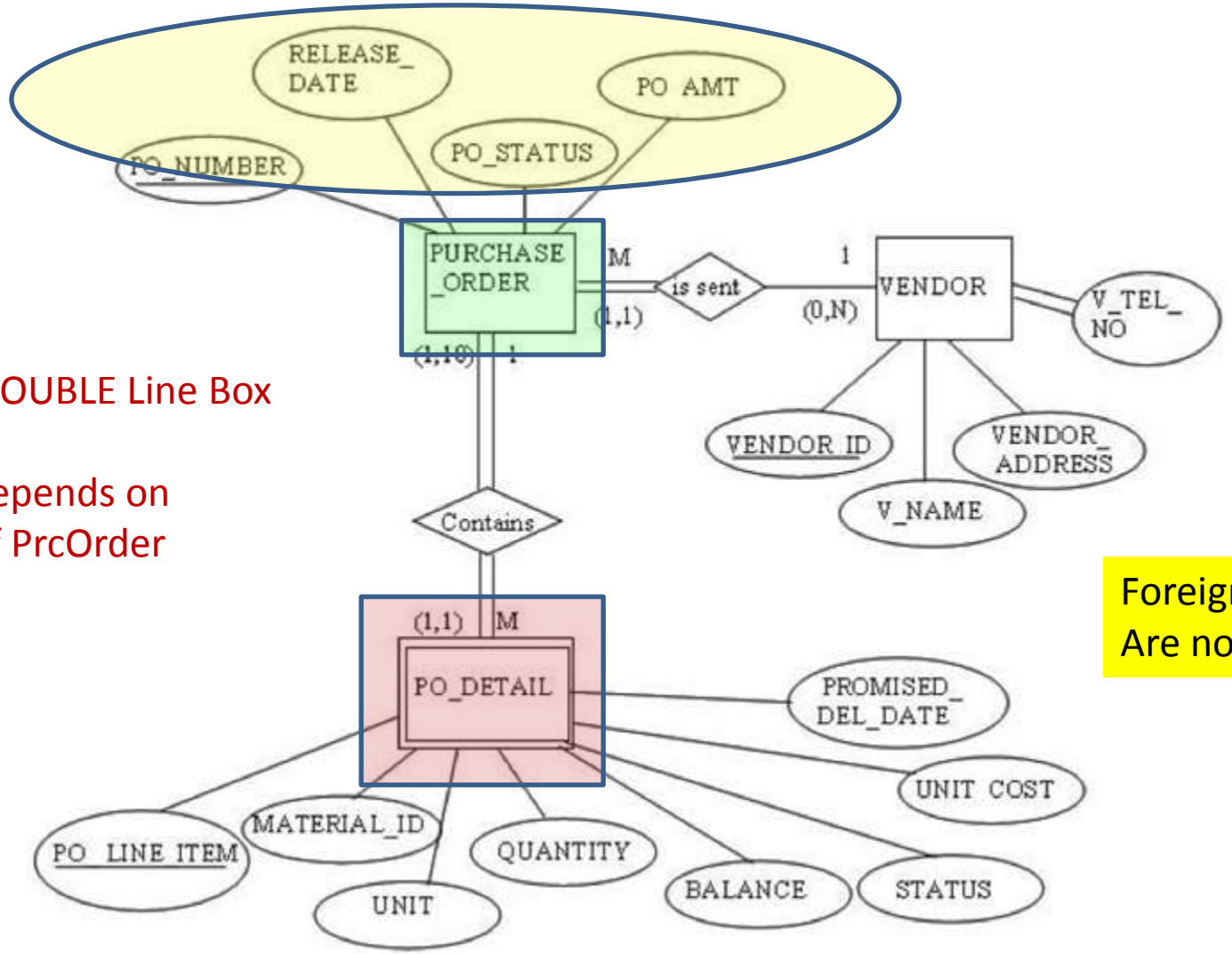
E-R Diagramming

1. Communicate – Interact
2. Design over a data model
3. Logical Model for Implementation of DB

Perfect Model
Less error possibility
in BL & UI – SW Layer

By a perfect model and good DB xor DBMS selection
you may put most of the Business rules into DB itself

Weak Entity – Child Entity
Strong Entity And Attributes



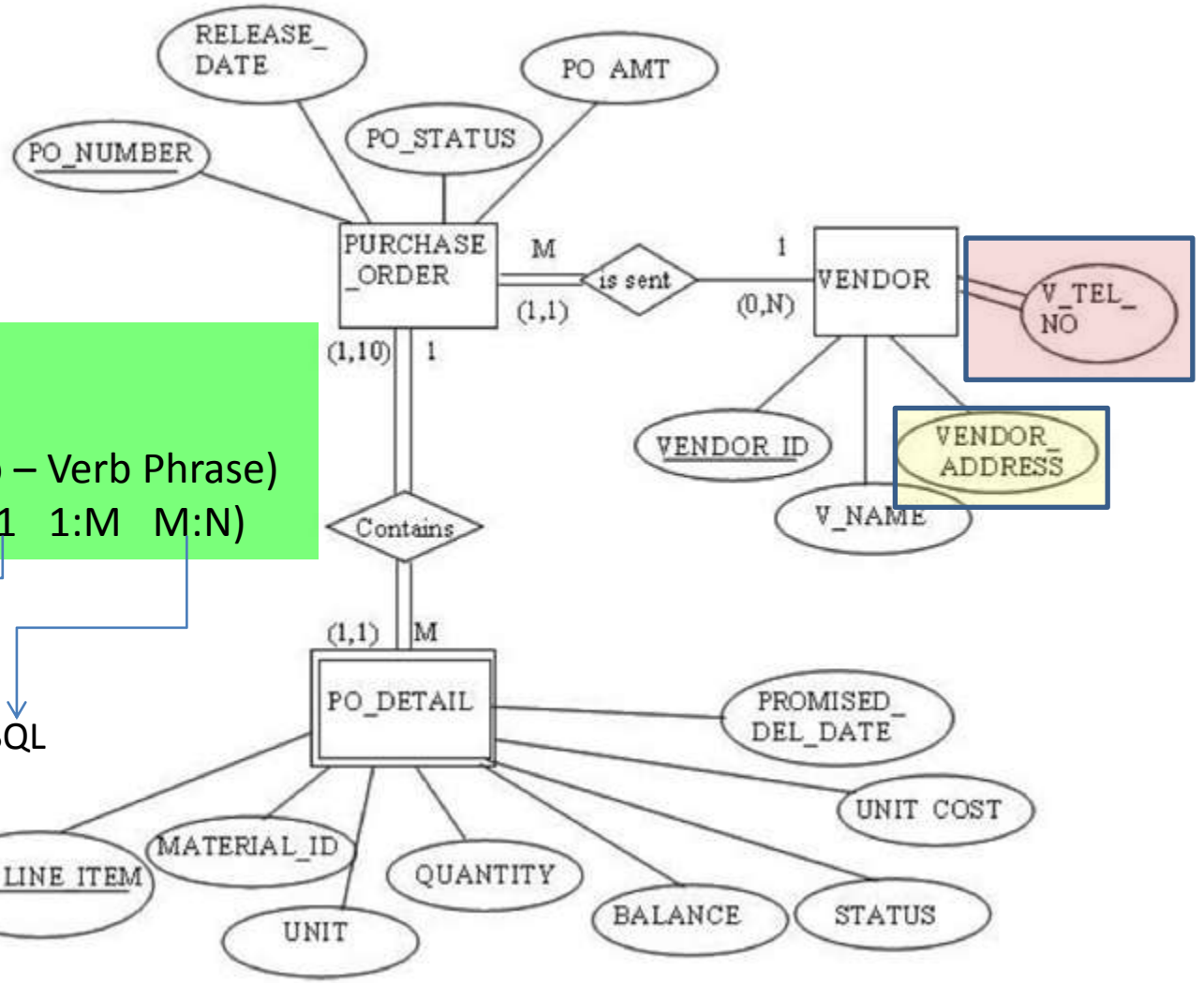
Shown by DOUBLE Line Box

Existence depends on existance of PrcOrder

Foreign Keys Are not Shown

Primary Key

Composite – Multi Valued Attributes
Relationships

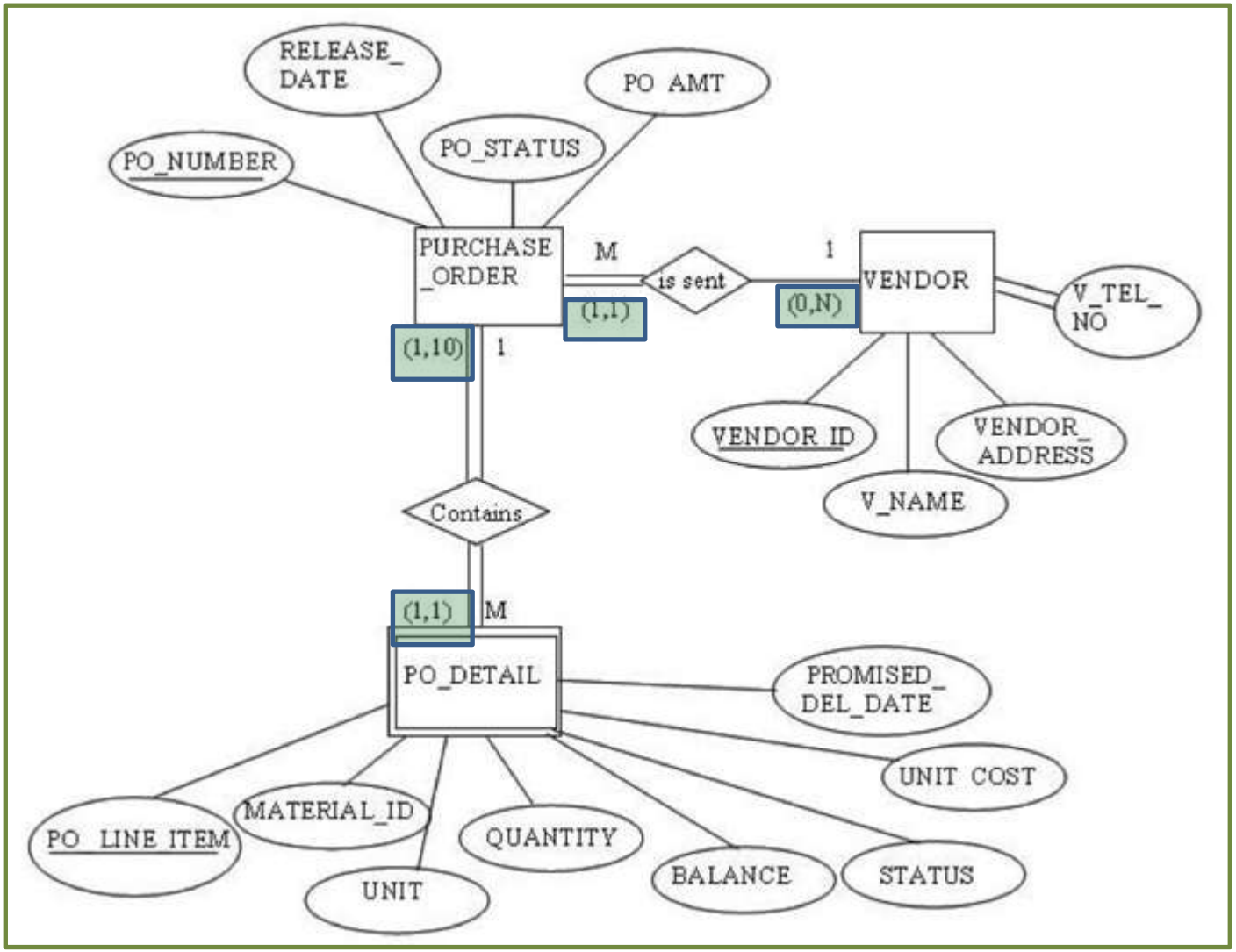


Relationships
- Arcs
- Diamonds (Verb – Verb Phrase)
- Cardinalities (1:1 1:M M:N)

Combine tables

Hard to handle in SQL

Relationships - Cardinality Limits



ER Diagramming

Recursive Entities - Unary

Production Example

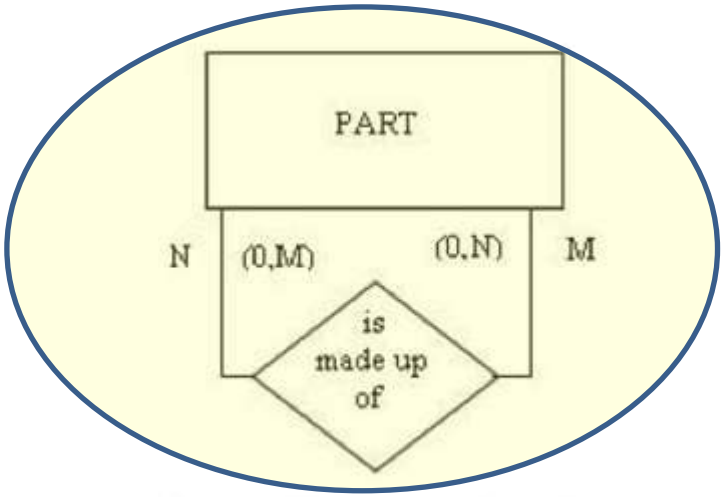
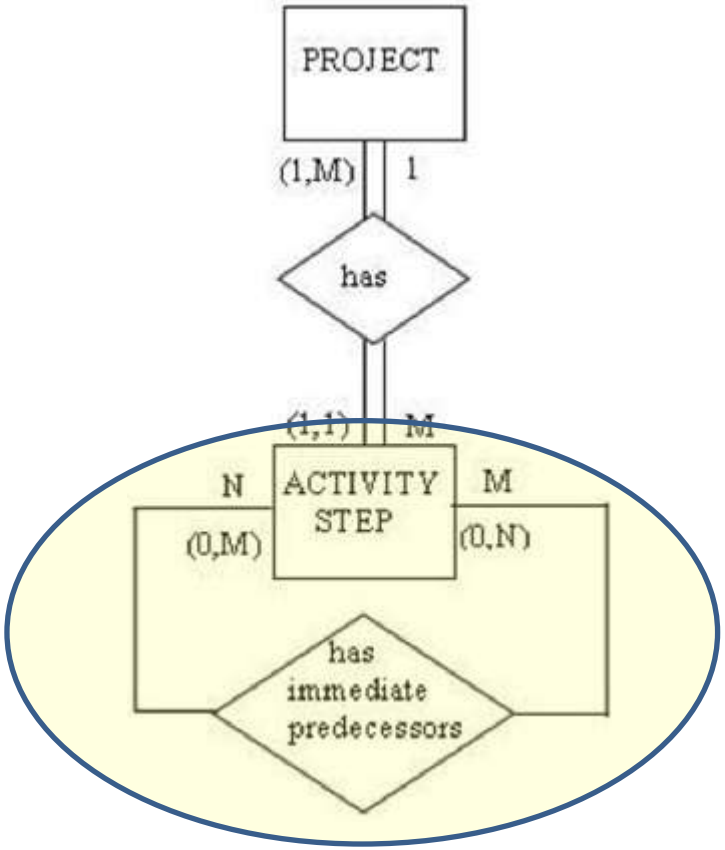


Figure 3.14 A recursive entity.

Project Management Example



ER Diagramming
Recursive Entities - Unary

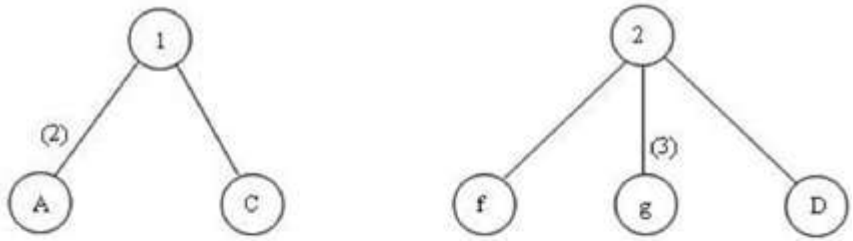
- BOM example
  - May not be so easy to convert 1-N

Bill of Materials Matrix

	1	2	A	D	B	C	g	e	f
1			2			1			
2				1			3		1
A					1			2	
D					2	1			
B						2	1		
C								1	3
g									
e									
f									

Item Level Assignments

	Level			
0	**1**	**2**	**3**	**4**
1	A	B	C	e
2	D		g	f



Two end products (1 and 2)

Four subassemblies (A, B, C, and D)

ER Diagramming
Recursive Entities - BOM Example

Bill of Materials Matrix

	1	2	A	D	B	C	g	e	f
1			2			1			
2				1			3		1
A					1			2	
D					2	1			
B						2	1		
C								1	3
g									
e									
f									

Software may consider Tact Time

To reduce in production inventory

Item Level Assignments

	Level			
0	1	2	3	4
1	A	B	C	e
2	D		g	f

TABLE PART

PART_NO	PART_DESC	DRAWING_NO
1	Product 1	102-23
2	Product 2	110-20
A	Subassembly A	290-10
B	Subassembly B	220-05
C	Subassembly C	256-01
D	Subassembly D	245-90
e	Component e	335-23
f	Component f	304-20
g	Component g	356-90

TABLE PART_COMPONENT

PART_NO	PART_COMP_NO	PART_COMP_QTY
1	A	2
1	C	1
2	D	1
2	f	1
2	g	3
A	B	1
A	e	2
B	C	2
B	g	1
C	e	1
C	f	3
D	B	2
D	C	1

Figure 3.15 PART tables using the BOM matrix.

ER Diagramming
Recursive Entities - BOM Example

Software can not consider Tact Time

High in process inventory
İf tact-time is high

TABLE PART

PART_NO	PART_DESC	DRAWING_NO
1	Product 1	102-23
2	Product 2	110-20
A	Subassembly A	290-10
B	Subassembly B	220-05
C	Subassembly C	256-01
D	Subassembly D	245-90
e	Component e	335-23
f	Component f	304-20
g	Component g	356-90

TABLE PART_COMPONENT

PART_NO	SUB_PART_NO	SUB_PART_QTY
1	A	2
1	B	2
1	C	5
1	e	9
1	f	15
1	g	2
2	B	3
2	C	7
2	D	1
2	e	7
2	f	21
2	g	6
A	B	1
A	C	2
A	e	4
A	f	6
A	g	1
B	C	2
B	e	2
B	f	6
B	g	1
C	e	1
C	f	3
D	B	2
D	C	5
D	e	5
D	f	15
D	g	2

Total Requirements Matrix

	1	2	A	D	B	C	g	e	f
1	1		2		2	5	2	9	15
2		1		1	3	7	6	7	21
A			1		1	2	1	4	6
D				1	2	5	2	5	15
B					1	2	1	2	6
C						1		1	3
γ							1		
α								1	
β									1

Figure 3.16 PART tables using total-requirements matrix.

ER Diagramming

Super class and Subclass Entity Types

To simplify your diagram
Use few entities as possible

Specialization circle
Overlapping, Distinct

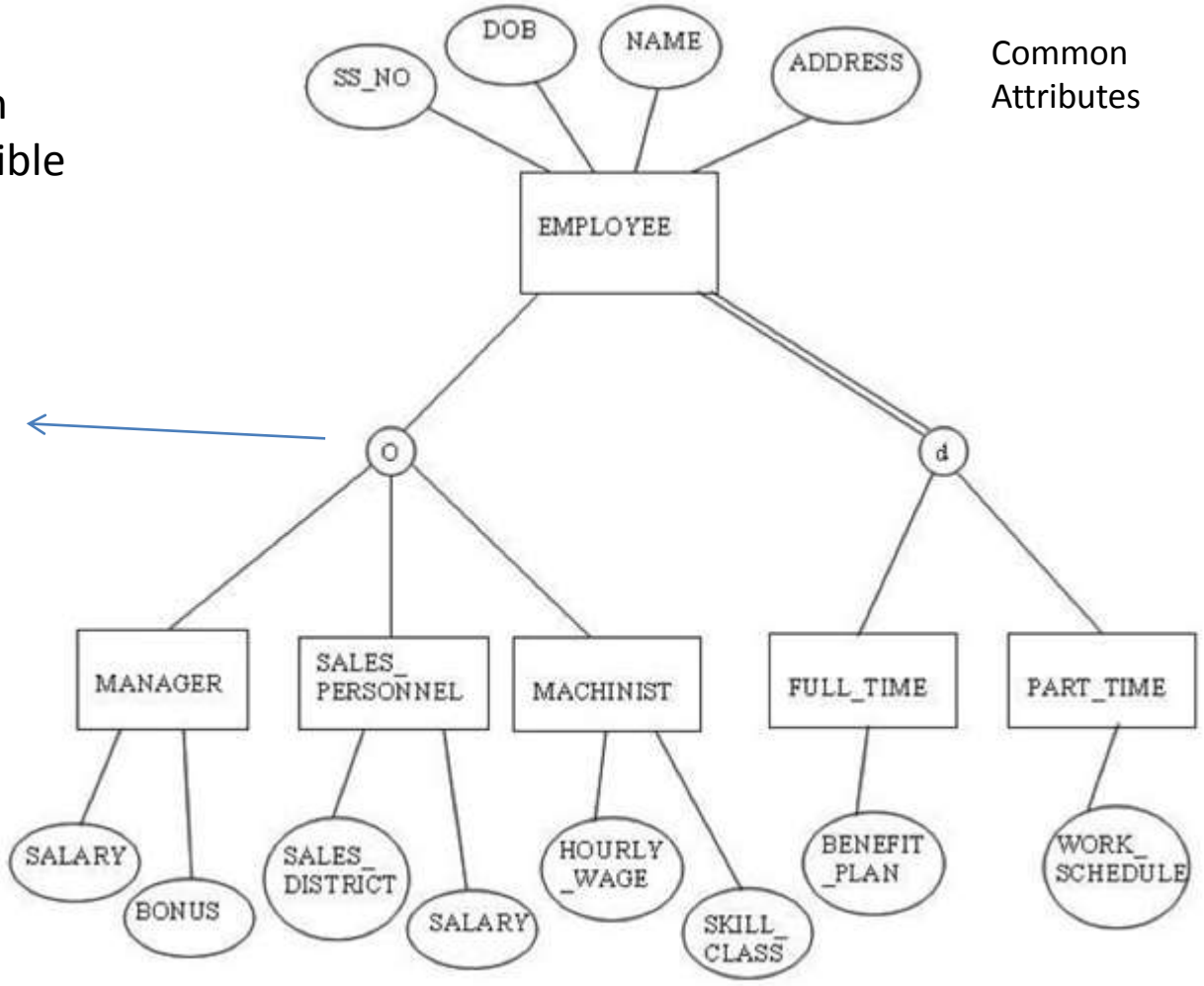
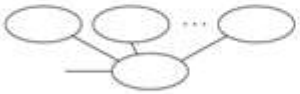
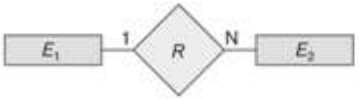
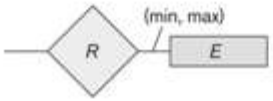


Figure 3.17 Superclass and subclass hierarchy.

E-R Notations Summary (Elmasri)

There may be differences in notation sometimes

Symbol	Meaning
	Entity
	Weak Entity
	Relationship
	Indentifying Relationship
	Attribute
	Key Attribute
	Multivalued Attribute
	Composite Attribute
	Derived Attribute
	Total Participation of E_2 in R
	Cardinality Ratio 1: N for E_1:E_2 in R
	Structural Constraint (min, max) on Participation of E in R

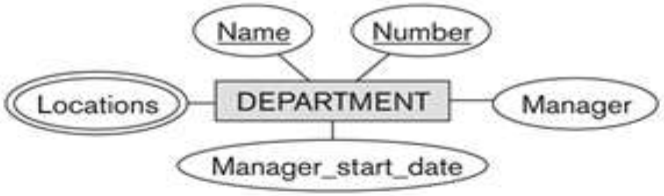
Case – Company Database

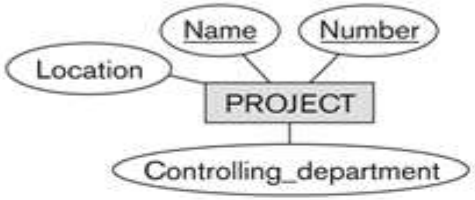
- We need to create a database schema design based on the following (simplified) **requirements** of the COMPANY Database:
  - The company is organized into DEPARTMENTs. Each department has a name, number and an employee who *manages* the department. We keep track of the start date of the department manager. A department may have several locations.
  - Each department *controls* a number of PROJECTs. Each project has a unique name, unique number and is located at a single location.

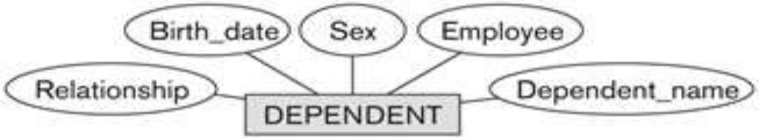
Case – Company Database

- We store each EMPLOYEE's social security number, address, salary, gender, and birthdate.
  - Each employee *works for* one department but may *work on* several projects.
  - We keep track of the number of hours per week that an employee currently works on each project.
  - We also keep track of the *direct supervisor* of each employee.
- Each employee may *have* a number of DEPENDENTs.
  - For each dependent, we keep track of their name, gender, birthdate, and relationship to the employee.

Case – Company Database Initial Steps

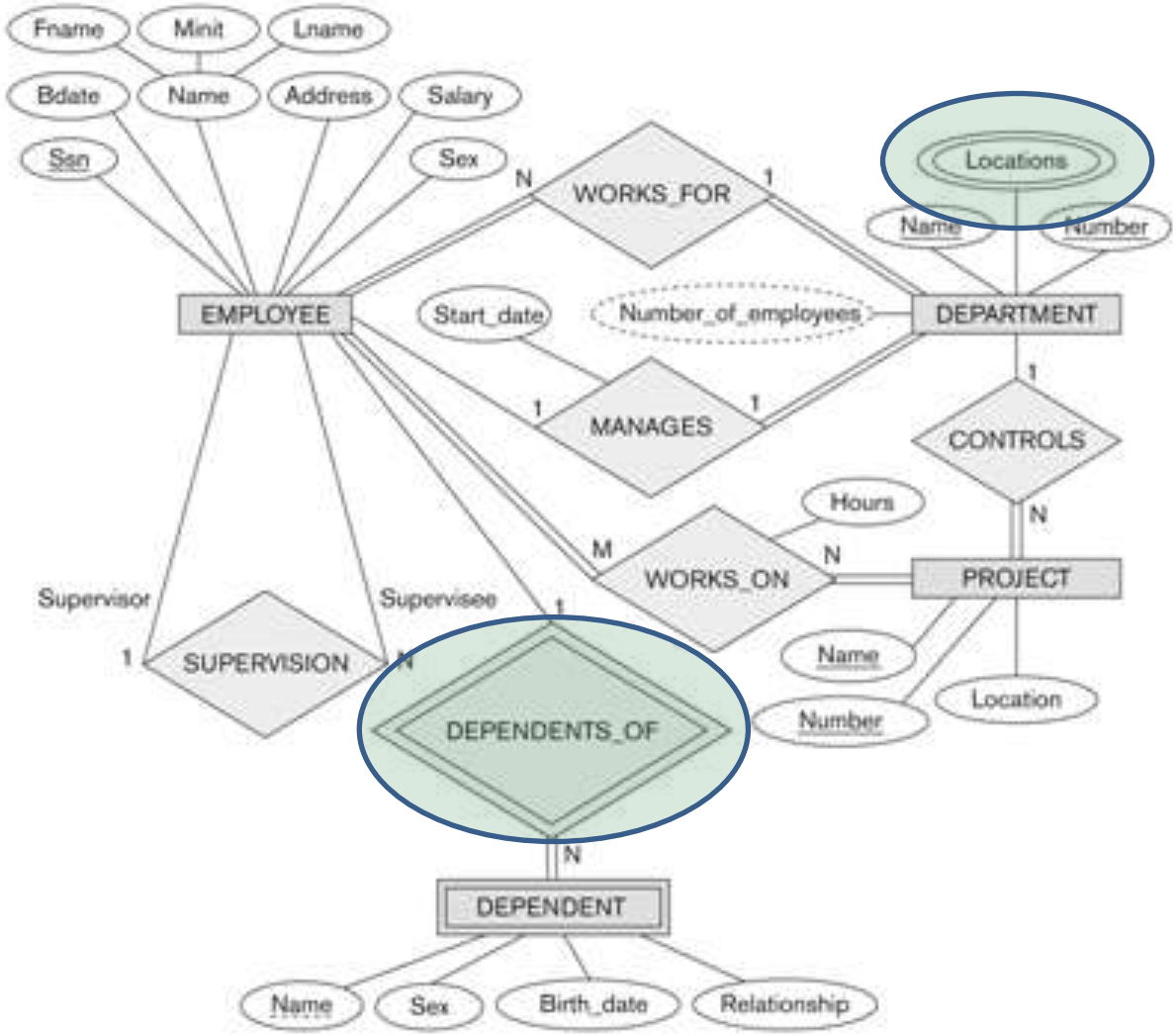




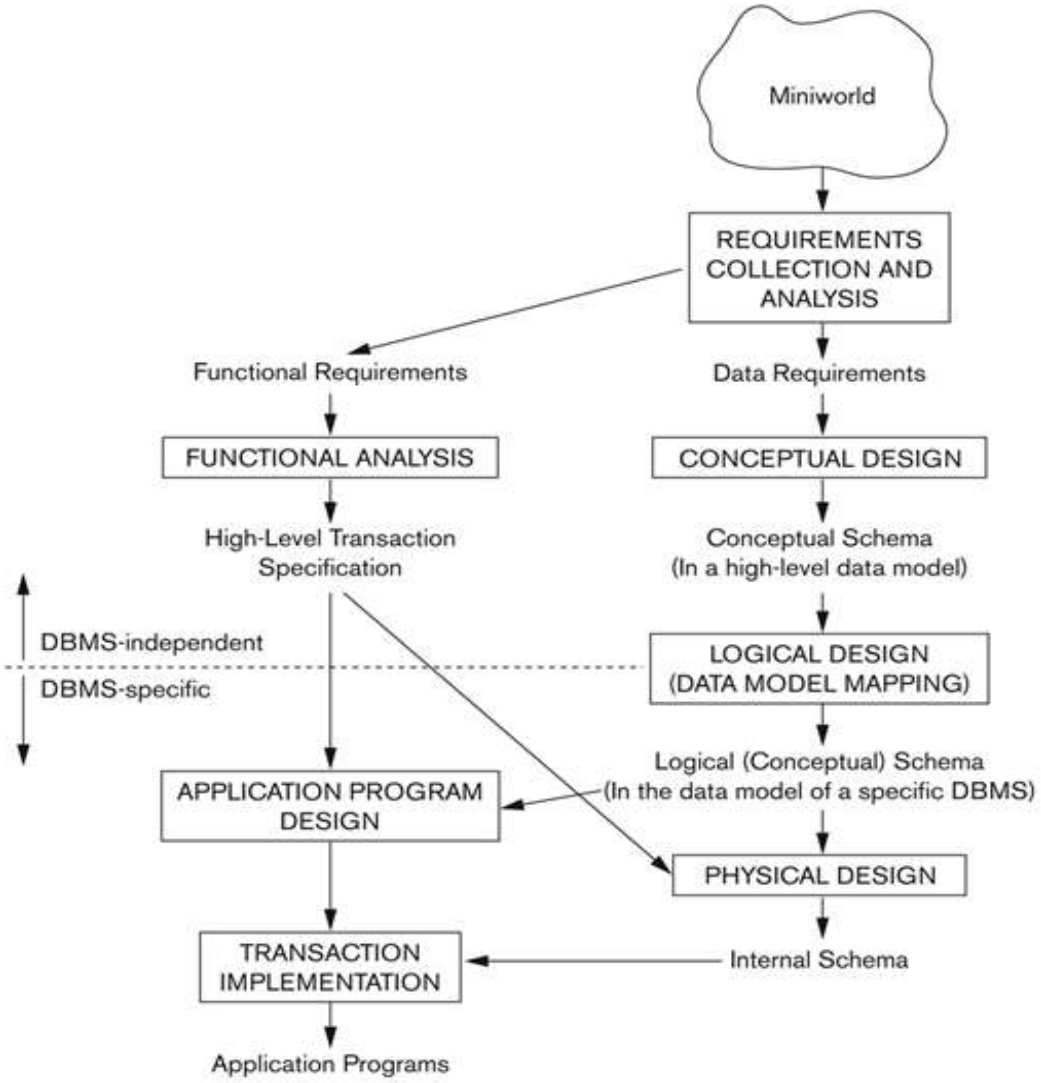


Case – Company Database

There may be differences in notation sometimes



DB Design Process



ER – DB Mapping Process

- **ER-to-Relational Mapping Algorithm**
  - Step 0: General rules for simplicity
  - Step 1: Mapping of Regular Entity Types
  - Step 2: Mapping of Weak Entity Types
  - Step 3: Mapping of Binary 1:1 Relation Types
  - Step 4: Mapping of Binary 1:N Relationship Types.
  - Step 5: Mapping of Binary M:N Relationship Types.
  - Step 6: Mapping of Multi valued attributes.
  - Step 7: Mapping of N-ary Relationship Types.

- **Mapping EER Model Constructs to Relations**
  - Step 8: Options for Mapping Specialization or Generalization.
  - Step 9: Mapping of Union Types (Categories).

ER – DB Mapping Process

Step 0 – General Rules

- Be Fast, Simple, Regular, In coordination with others.
  - ✓ Do not do anything by yourself (try to work in 3)
  - ✓ If you do (for being fast) share with 2 of others face to face
  - ✓ Send info to other team members
- Decide about Naming Rules with your team

  Example:
  - ✓ Table Names: All Capital with underscore
  - ✓ Attribute Names: Pascal Case having first 3 consonant letters of the table
  - ✓ Use same Primary key name as foreign key
  - ✓ SW variable names: Camel Case having 2 consonant letters of project.
  - ✓ Key fields should be first fields
  - ✓ Foreign keys should be just after key fields

ER – DB Mapping Process

Step 1 – Mapping Regular Entities

- ✓ Make first field: key attribute and keep it short
- ✓ Map all attributes in logical sequence as real life
- ✓ For the **M** site attributes take the primary key of **1** site as foreign key
- ✓ Reduce 1-1 relation as field in shorter table side (use your logic)
- ✓ Use another table for multi valued attributes

- Step 1: (Elmasri)

  For each regular (strong) entity type E in the ER schema, create a relation R that includes all the simple attributes of E.

  Choose one of the key attributes of E as the primary key for R.

  If the chosen key of E is composite, the set of simple attributes that form it will together form the primary key of R.

Example: We create the relations EMPLOYEE, DEPARTMENT, and PROJECT in the relational schema corresponding to the regular entities in the ER diagram.

  – SSN, DNUMBER, and PNUMBER are the primary keys for the relations EMPLOYEE, DEPARTMENT, and PROJECT as shown.

ER – DB Mapping Process

Step 2 – Mapping Weak Entities

- ✓ Same as regular entity
- ✓ Do not forget foreign key as primary key of strong table

- **Step 2: (Elmasri)**
  - For each weak entity type W in the ER schema with owner entity type E, create a relation R & include all simple attributes (or simple components of composite attributes) of W as attributes of R.
  - Also, include as foreign key attributes of R the primary key attribute(s) of the relation(s) that correspond to the owner entity type(s).
  - The primary key of R is the *combination of* the primary key(s) of the owner(s) and the partial key of the weak entity type W, if any.
- **Example:** Create the relation DEPENDENT in this step to correspond to the weak entity type DEPENDENT.
  - Include the primary key SSN of the EMPLOYEE relation as a foreign key attribute of DEPENDENT (renamed to ESSN).
  - The primary key of the DEPENDENT relation is the combination {ESSN, DEPENDENT_NAME} because DEPENDENT_NAME is the partial key of DEPENDENT.

ER – DB Mapping Process

Step 3 – Mapping 1:1 Relations

✓ Reduce it as a field in short table side

- **Step 3: (Elmasri)**
  - For each binary 1:1 relationship type R in the ER schema, identify the relations S and T that correspond to the entity types participating in R.
- There are three possible approaches:

1- Foreign Key approach: Choose one of the relations-say S-and include a foreign key in S the primary key of T. It is better to choose an entity type with total participation in R in the role of S.

Example: 1:1 relation MANAGES is mapped by choosing the participating entity type DEPARTMENT to serve in the role of S, because its participation in the MANAGES relationship type is total.

2- Merged relation option: An alternate mapping of a 1:1 relationship type is possible by merging the two entity types and the relationship into a single relation. This may be appropriate when both participations are total.

3- Cross-reference or relationship relation option: The third alternative is to set up a third relation R for the purpose of cross-referencing the primary keys of the two relations S and T representing the entity types.

ER – DB Mapping Process

Step 4 – Mapping 1:N Relations

✓ Include 1 side primary key as foreign key in side N

- Step 4: (Elmasri)
  - For each regular binary 1:N relationship type R, identify the relation S that represent the participating entity type at the N-side of the relationship type.
  - Include as foreign key in S the primary key of the relation T that represents the other entity type participating in R.
  - Include any simple attributes of the 1:N relation type as attributes of S.
- Example: 1:N relationship types WORKS_FOR, CONTROLS, and SUPERVISION in the figure.
  - For WORKS_FOR we include the primary key DNUMBER of the DEPARTMENT relation as foreign key in the EMPLOYEE relation and call it DNO.

ER – DB Mapping Process

Step 5 – Mapping M:N Relations

- ✓ Reduce to 1:N
- ✓ Use combined key in X-ref table (Primaries of M&N).
  - ✓ You may use another integer primary id, if DBMS do not permit combined as primary.

- Step 5: (Elmasri)
  - For each regular binary M:N relationship type R, *create a new relation* S to represent R.
  - Include as foreign key attributes in S the primary keys of the relations that represent the participating entity types; *their combination will form the primary key* of S.
  - Also include any simple attributes of the M:N relationship type (or simple components of composite attributes) as attributes of S.
- Example: The M:N relationship type WORKS_ON from the ER diagram is mapped by creating a relation WORKS_ON in the relational database schema.
  - The primary keys of the PROJECT and EMPLOYEE relations are included as foreign keys in WORKS_ON and renamed PNO and ESSN, respectively.
  - Attribute HOURS in WORKS_ON represents the HOURS attribute of the relation type. The primary key of the WORKS_ON relation is the combination of the foreign key attributes {ESSN, PNO}.

ER – DB Mapping Process

Step 6 – Mapping Multi valued Attribute

✓ Create another short table

- Step 6: (Elmasri)
  - For each multi valued attribute A, create a new relation R.
  - This relation R will include an attribute corresponding to A, plus the primary key attribute K-as a foreign key in R-of the relation that represents the entity type of relationship type that has A as an attribute.
  - The primary key of R is the combination of A and K. If the multi valued attribute is composite, we include its simple components.
- **Example:** The relation DEPT_LOCATIONS is created.
  - The attribute DLOCATION represents the multi valued attribute LOCATIONS of DEPARTMENT, while DNUMBER-as foreign key-represents the primary key of the DEPARTMENT relation.
  - The primary key of R is the combination of {DNUMBER, DLOCATION}.

ER – DB Mapping Process

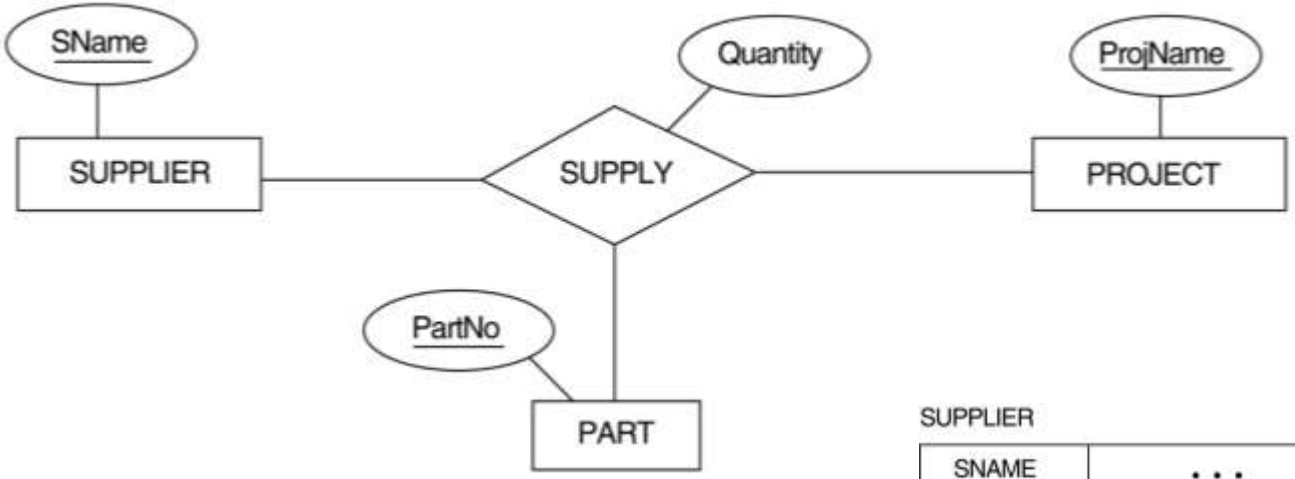
Step 7 – Mapping N-ary Relations

- ✓ Use x-ref table(s) (Convert 1:N)
- ✓ Use combined key in X-ref table (Primaries of other entities).
  - ✓ You may use another integer primary id, if DBMS do not permit combined as primary.

- **Step 7: (Elmasri)**
  - For each n-ary relationship type R, where n>2, create a new relationship S to represent R.
  - Include as foreign key attributes in S the primary keys of the relations that represent the participating entity types.
  - Also include any simple attributes of the n-ary relationship type (or simple components of composite attributes) as attributes of S.
- **Example:** The relationship type SUPPY in the ER on the next slide.
  - This can be mapped to the relation SUPPLY shown in the relational schema, whose primary key is the combination of the three foreign keys {SNAME, PARTNO, PROJNAME}

ER – DB Mapping Process

Step 7 – Mapping N-ary Relations



SUPPLIER

SNAME	. . .

PROJECT

PROJNAME	. . .

PART

PARTNO	. . .

SUPPLY

SNAME	PROJNAME	PARTNO	QUANTITY

ER – DB Mapping Process

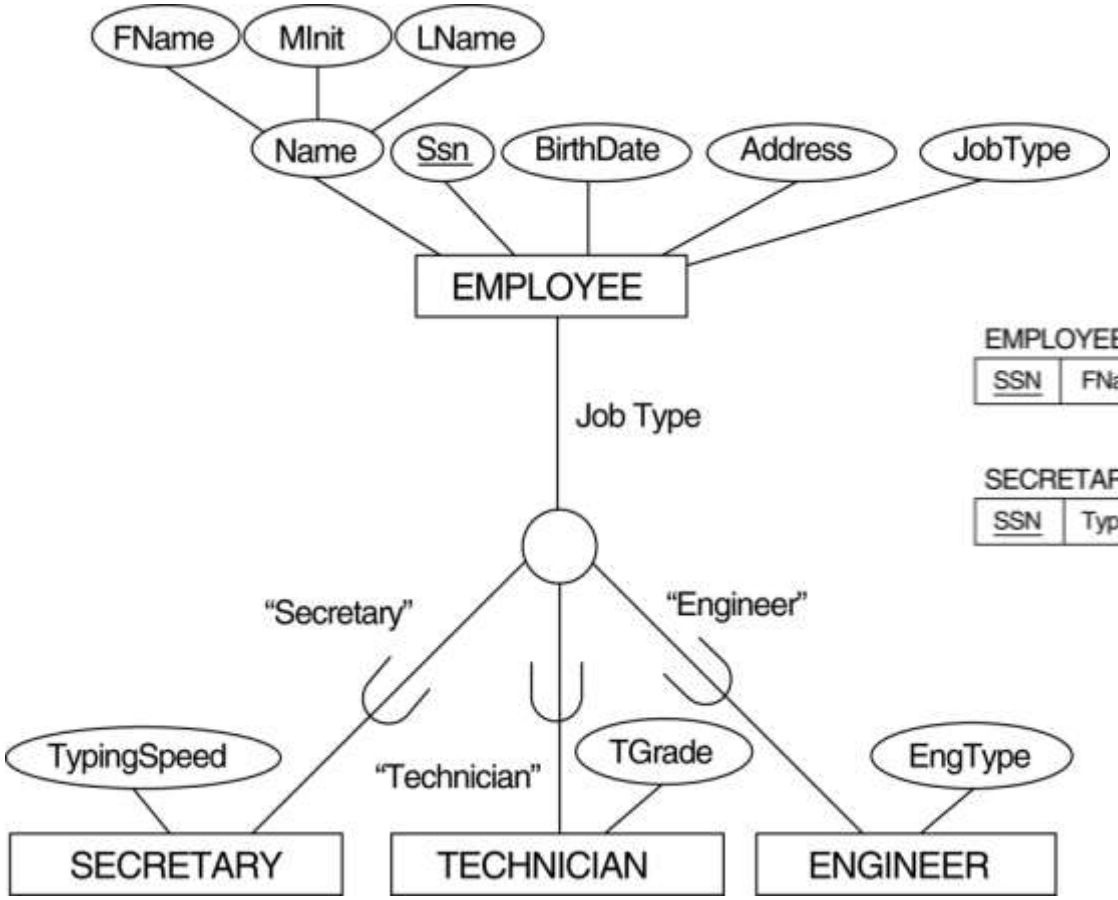
Step 8 – Mapping Sub-Super Classes

✓ You may use sub tables if subclasses have additional attributes

- **Step8: Options for Mapping Specialization or Generalization.**
  - Convert each specialization with m subclasses {S1, S2,….,Sm} and generalized super class C, where the attributes of C are {k,a1,…an} and k is the (primary) key, into relational schemas using one of the four following options:
    - Option 8A: Multiple relations-Super class and subclasses
    - Option 8B: Multiple relations-Subclass relations only
    - Option 8C: Single relation with one type attribute
    - Option 8D: Single relation with multiple type attributes

ER – DB Mapping Process

Step 8 – Mapping Sub-Super Classes



EMPLOYEE

SSN	FName	MInit	LName	BirthDate	Address	JobType

SECRETARY

SSN	TypingSpeed

TECHNICIAN

SSN	TGrade

ENGINEER

SSN	EngType

Classwork - Homework

- Do reverse engineering on company database
  - Write in your own words (on A4) what is happening
- In Lab: Map The database
  - If you cannot finish, mail it from lab computer to yourself after completion send it to hposaci@quiztechnology.com
- Homework Case Study:
  - Solve 3.8 from the book.
- For Midterm & Final: Observe 3.3 to 3.7 also consider mapping to database entities and attributes (columns)

Questions

Questions?

532 2877127

hposaci@quiztechnology.com