

# 4

## Pemodelan Basis Data dengan EER ( PART 2 )

CSF2600700 - BASIS DATA

enhanced





## Acknowledgements

This slide is a modification to supplementary slide of “Database System”, 6th edition, Elmasri/Navathe, 2011: **Chapter 4** used in “Basis Data” course in academic years 2018/2019 in the Faculty of Computer Science, Universitas Indonesia.

Additional resources: presentation prepared by Prof Steven A. Demurjian, Sr  
(<http://www.engr.uconn.edu/~steve/courses.html>)

## Yang Sudah Dipelajari

1. Latar Belakang

2. Superclass/Subclass Relationship

3. Spesialisasi dan Generalisasi



# Outline

1. Latar Belakang

2. Superclass/Subclass Relationship

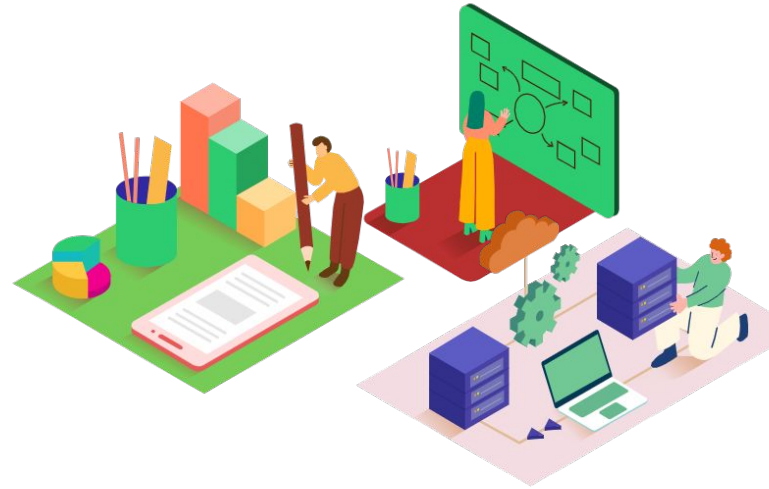
3. Spesialisasi dan Generalisasi

**4. Hierarchy dan Lattice**

5. Pemodelan dengan Categories

6. Higher Degree Relationship

7. Kapan Kita Menggunakan EER?

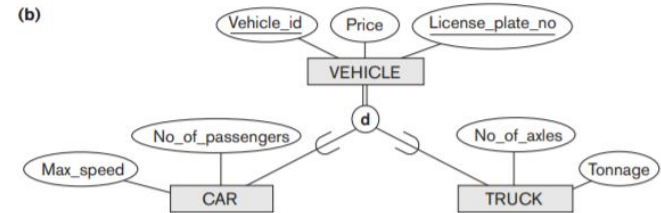


# Hierarchy dan Lattice

## Hierarchy

Satu subclass hanya berpartisipasi pada satu class/subclass relationship (satu subclass **hanya** memiliki satu superclass saja)

**Contoh:** VEHICLE dengan TRUCK dan CAR



## Lattice

- Satu subclass dapat berpartisipasi pada lebih dari satu class/subclass relationship

**Contoh:**

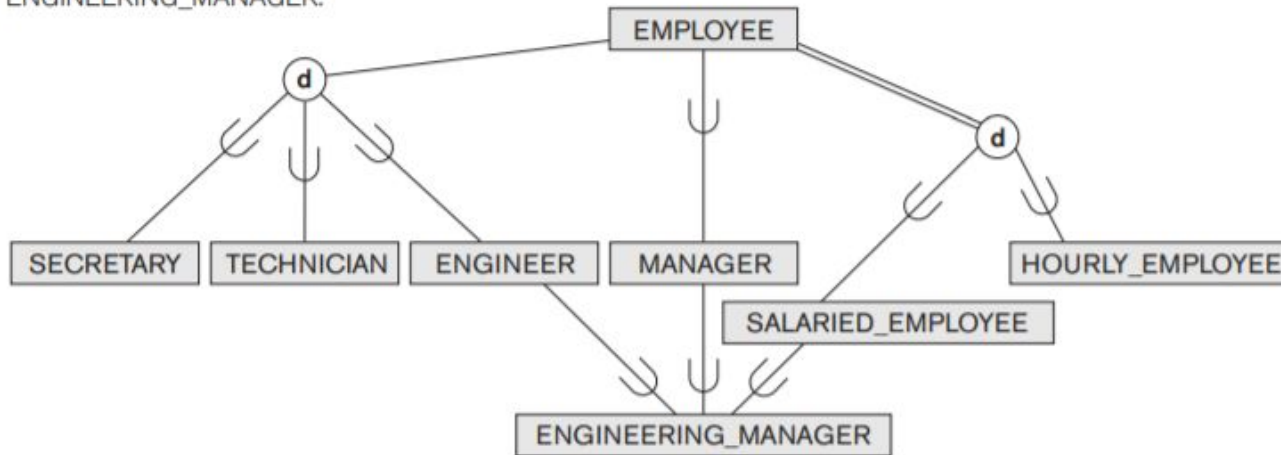
seorang Engineering Manager, haruslah seorang Engineer dan juga seorang Manajer

- Mengandung konsep multiple inheritance

## Contoh Lattice (1)

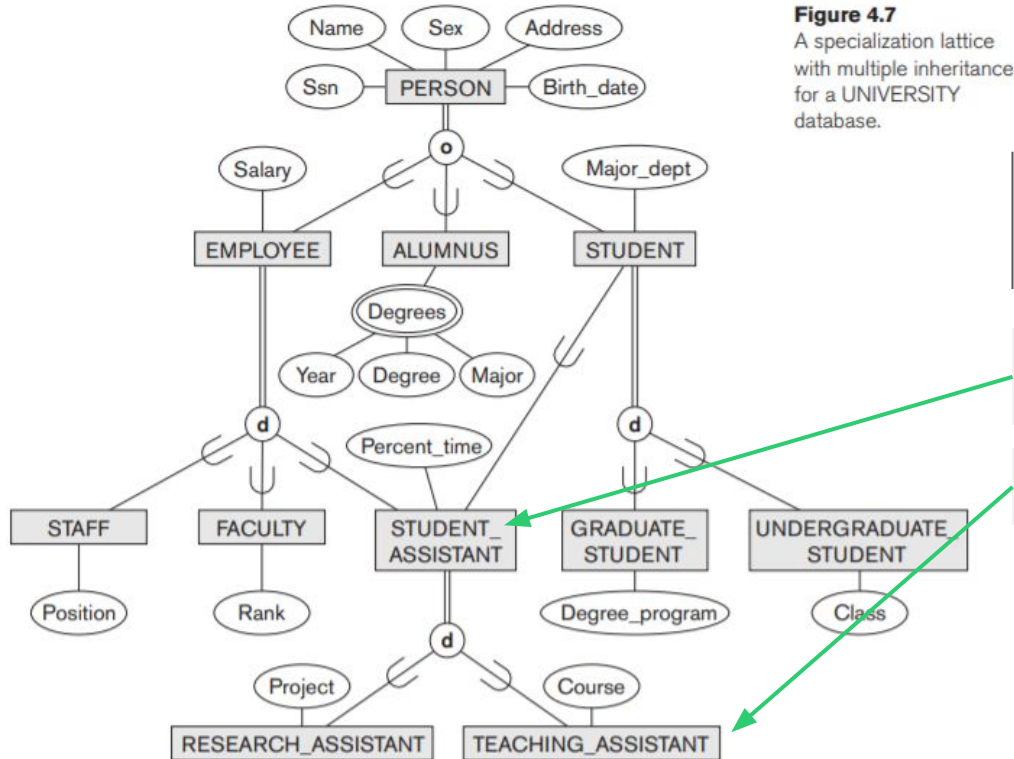
**Figure 4.6**

A specialization lattice with shared subclass  
ENGINEERING\_MANAGER.



**Engineering\_Manager** punya 3 relationship,  
namun ketiganya punya 1 superclass

## Contoh Lattice (2)



**Figure 4.7**

A specialization lattice with multiple inheritance for a UNIVERSITY database.

Satu entity mungkin ada di beberapa **subclass**. Misal graduate student sekaligus teaching assistant

**Multiple inheritance!** Namun attribute dari PERSON hanya diwariskan 1 kali

**Leaf node:** tidak punya subclass

# Outline

1. Latar Belakang

2. Superclass/Subclass Relationship

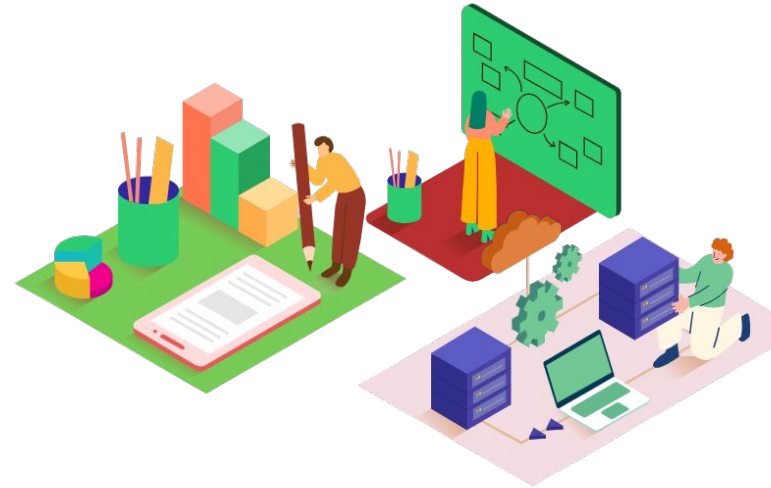
3. Spesialisasi dan Generalisasi

4. Hierarchy dan Lattice

5. Pemodelan dengan Categories

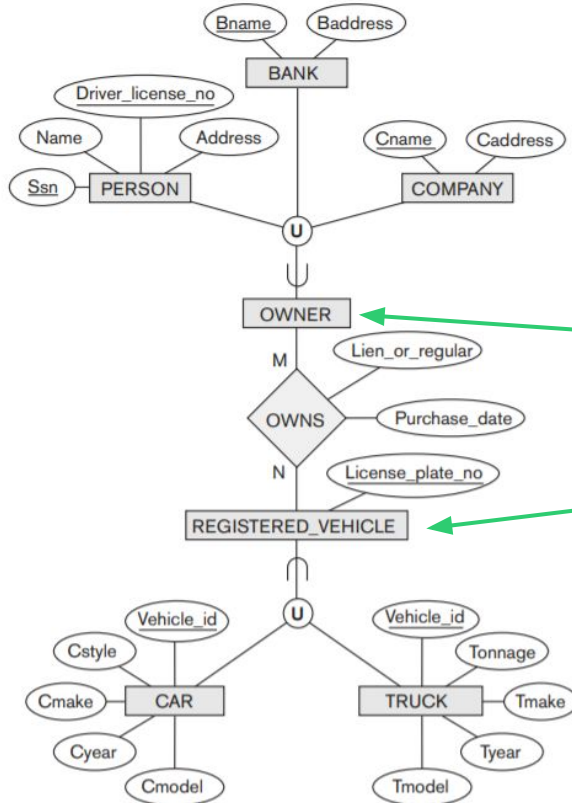
6. Higher Degree Relationship

7. Kapan Kita Menggunakan EER?





# Union Type dengan Menggunakan Category



Satu subclass memiliki satu relationship dengan 3 buah superclass: disebut sebagai **union type** atau **category**

OWNER merupakan **union subclass** dari COMPANY, BANK, PERSON

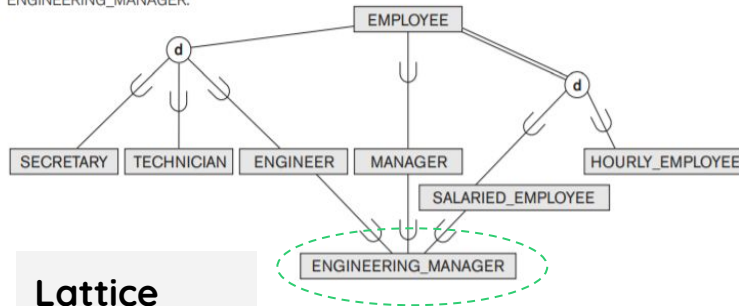
REGISTERED\_VEHICLE merupakan **union subclass** dari TRUCK dan CAR

**Figure 4.8**  
Two categories (union types): OWNER and REGISTERED\_VEHICLE.

# Perbedaan Category dengan Lattice

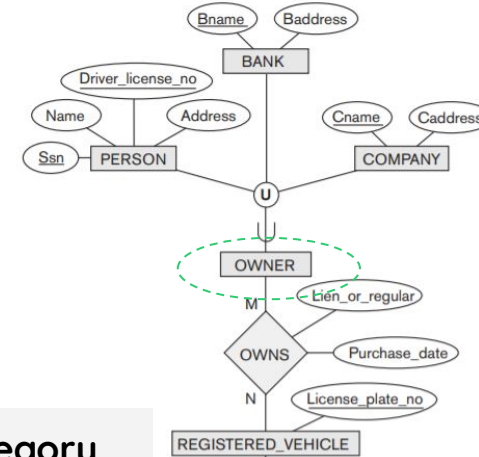
Figure 4.6

A specialization lattice with shared subclass ENGINEERING\_MANAGER.



## Lattice

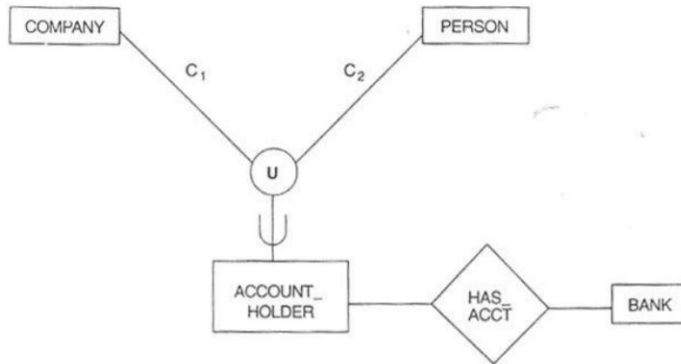
- Engineering\_Manager **harus ada pada semua** superclass: Manager, Engineer, Salaried\_Employee
- Engineering\_Manager: **mewarisi semua attribute** dari superclasses



## Category

- Owner harus ada pada **salah satu** dari ketiga superclasses
- Owner mewarisi **attribute tertentu saja**, tergantung dari superclass-nya

## Partial Category



**Partial category:**  
dapat berpartisipasi ataupun tidak  
pada relationship

## Total Category

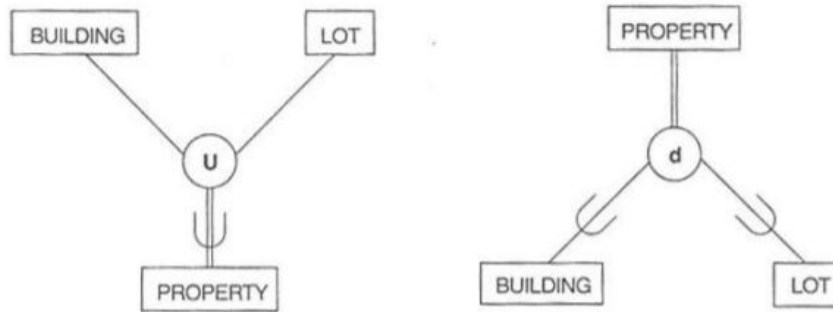


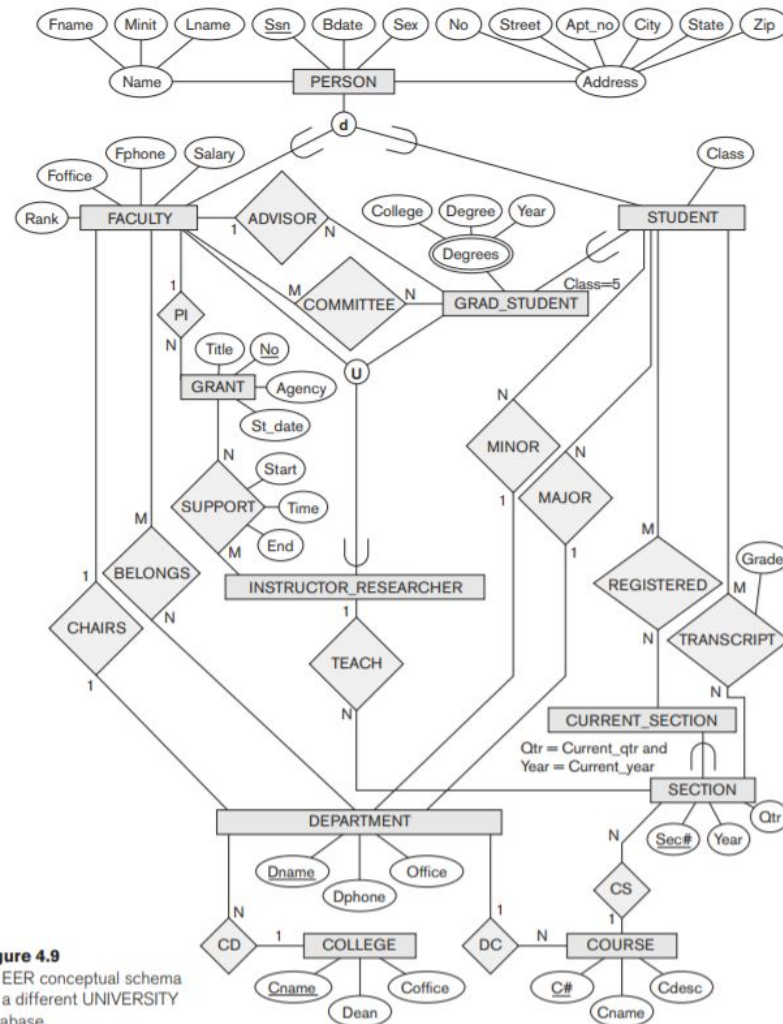
Figure 4.9 Total and partial categories. (a) Partial category `ACCOUNT_HOLDER` that is a subset of the union of two entity types `COMPANY` and `PERSON`. (b) Total category `PROPERTY` and a similar generalization.

Harus **merupakan salah satu superclasses**

**Contoh:** A building and a lot must be a member of `PROPERTY`

Dapat direpresentasikan sebagai generalization (d), khususnya jika kemiripannya banyak

# Contoh Skema EER untuk Basis Data Universitas



**Figure 4.9**  
An EER conceptual schema  
for a different UNIVERSITY  
database.

# Outline

1. Latar Belakang

2. Superclass/Subclass Relationship

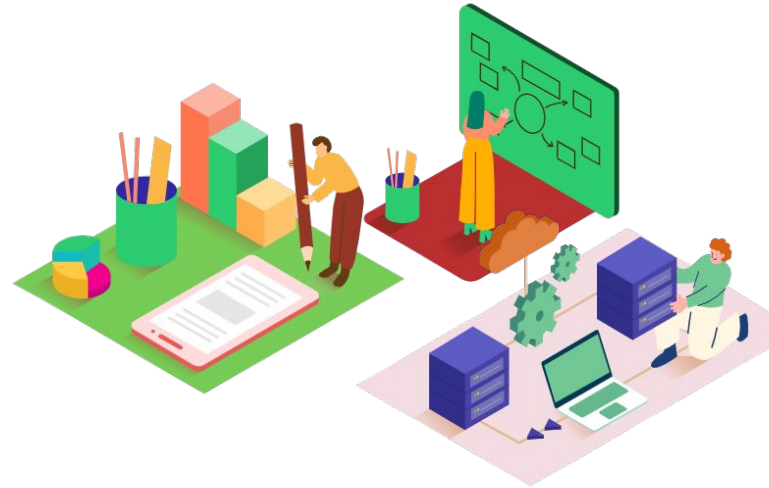
3. Spesialisasi dan Generalisasi

4. Hierarchy dan Lattice

5. Pemodelan dengan Categories

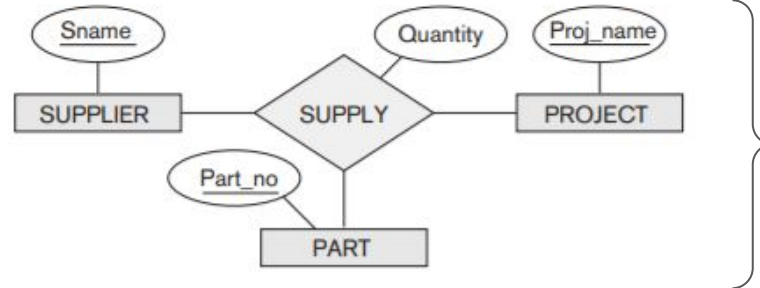
6. Higher Degree Relationship

7. Kapan Kita Menggunakan EER?

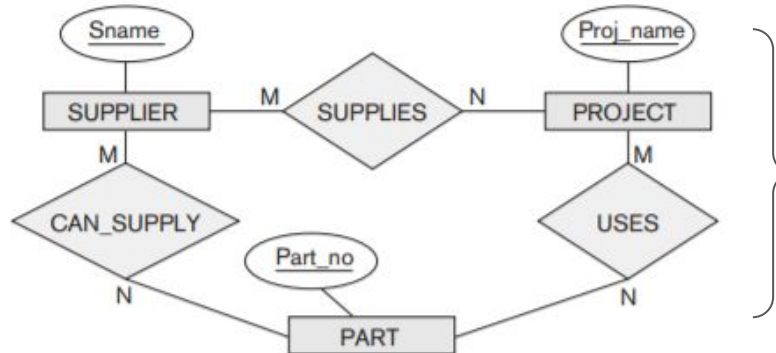


## Higher Degree Relationship (1)

Dua skema ini  
beda maknanya!



**Ternary relationship type:**  
menghubungkan 3 entity types



Tiga **binary relationship type:**  
CAN\_SUPPLY, USES, SUPPLIES

## Higher Degree Relationship (2)

Higher degree relationship tampak kompleks, bagaimana menyederhanakannya?

### Opsi 1. Higher degree relationship sebagai weak entity

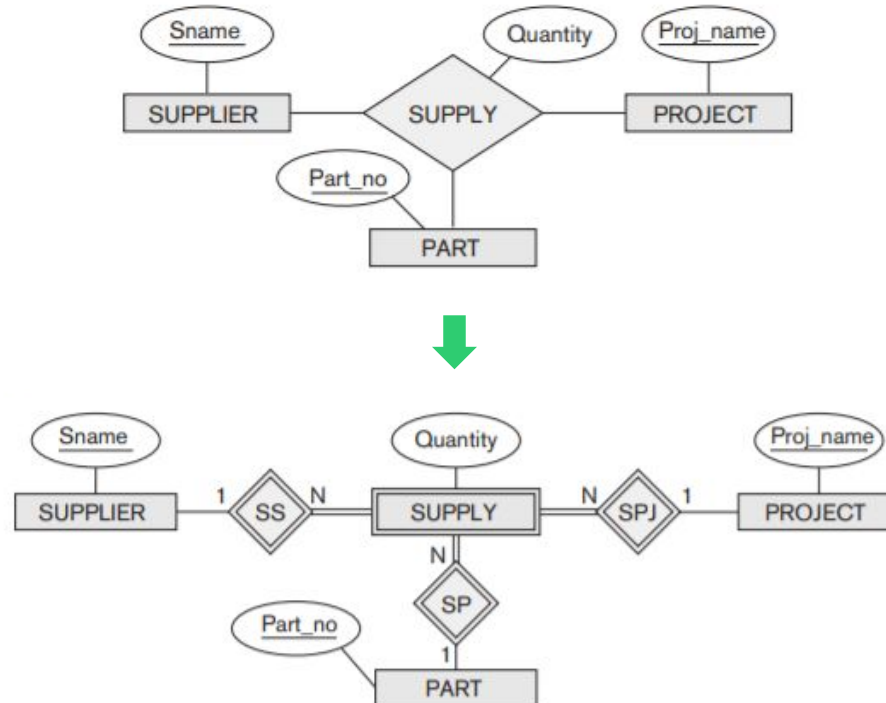
- Merepresentasikan *higher degree relationship* sebagai weak entity type yang berhubungan ke owner entity types
- Mengandung binary (identifying) relationship

### Opsi 2. Higher degree relationship sebagai identifying relationship type

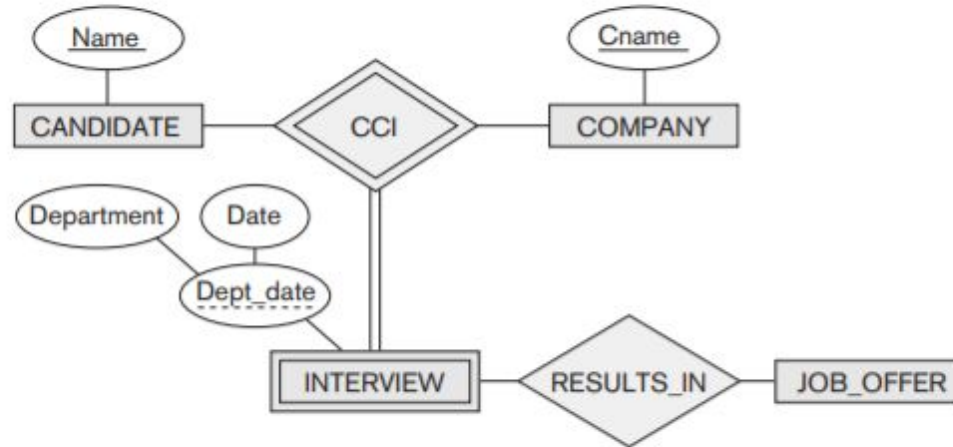
- Sebuah ternary relationship type dengan sebuah weak entity type dan dua buah owner entity type



## Ternary Relationship sebagai Weak Entity Type



## Ternary Relationship sebagai Identifying Relationship Type



**Figure 3.19**

A weak entity type **INTERVIEW** with a ternary identifying relationship type.

# Outline

1. Latar Belakang

2. Superclass/Subclass Relationship

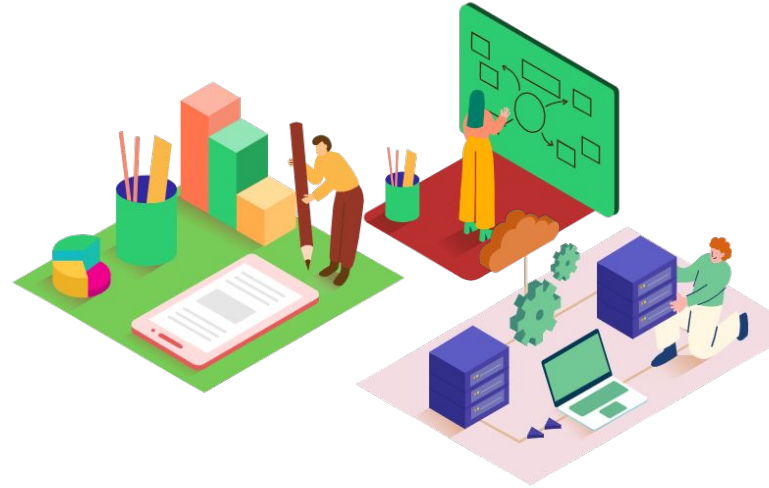
3. Spesialisasi dan Generalisasi

4. Hierarchy dan Lattice

5. Pemodelan dengan Categories

6. Higher Degree Relationship

7. Kapan Kita Menggunakan EER?



## Kapan Kita Menggunakan Model EER? (1)

- Sebagian besar proyek basis data tidak perlu fitur-fitur model berorientasi obyek yang ada pada EER
- Tujuan pemodelan data konseptual adalah untuk menghasilkan sebuah **model yang sederhana dan mudah dimengerti**
- Jangan menggunakan class/subclass relationship yang kompleks jika tidak diperlukan
- Penggunaan model EER menawarkan keuntungan dibandingkan model ER jika digunakan pada **kondisi yang tepat**

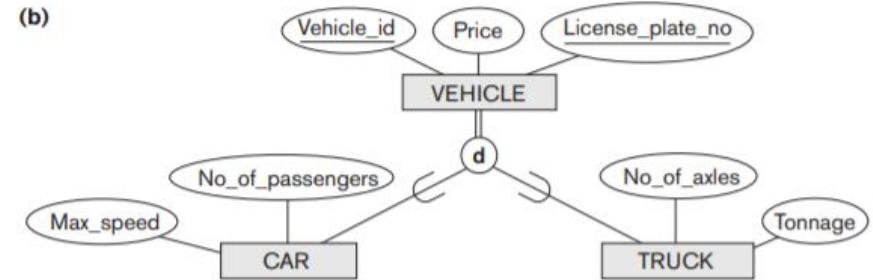


## Kapan Kita Menggunakan Model EER? (2)

Model EER perlu digunakan jika domain yang dimodelkan secara alamiah bersifat *object-oriented*, *inheritance* akan mereduksi kompleksitas perancangan

Gunakan EER pada situasi:

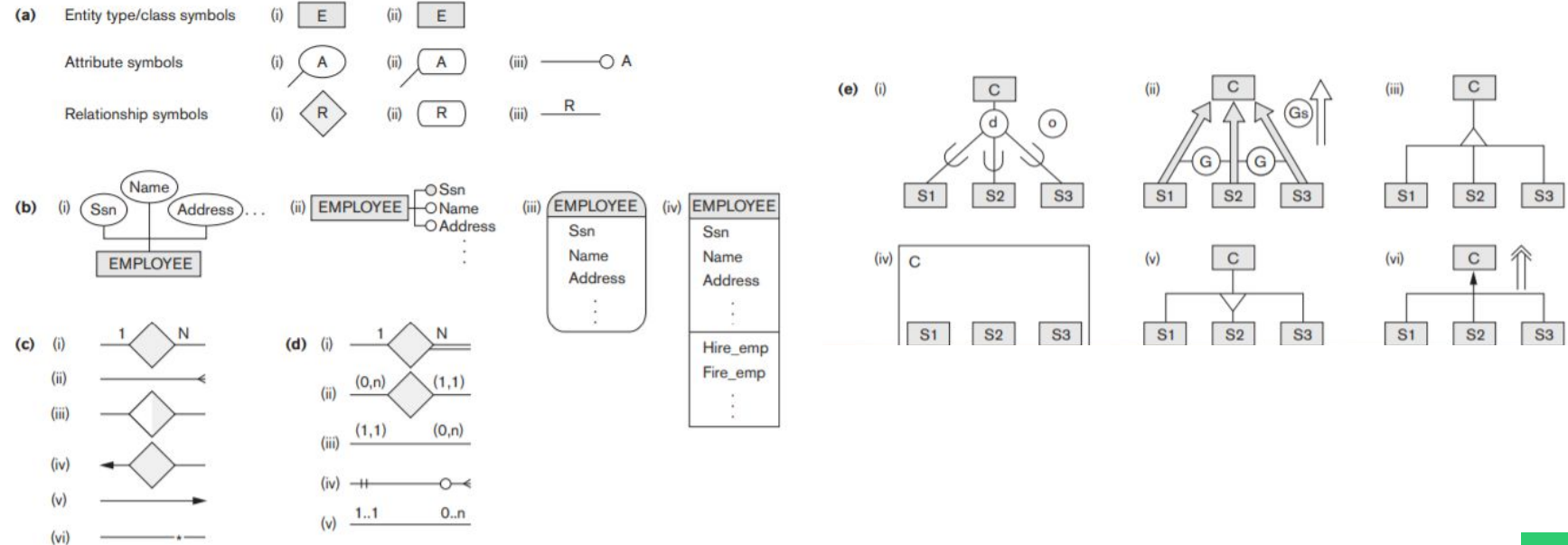
- Ketika penggunaan attribute inheritance dapat **mereduksi penggunaan null** pada suatu single entity relation (yang mengandung multiple subclasses)
- Subclass dapat digunakan untuk secara eksplisit memodelkan dan menamai subset dari entity yang berpartisipasi pada relationship-nya sendiri (dimana **subclass lain dalam superclass yang sama tidak berpartisipasi** pada relationship tersebut)



# Alternative Diagrammatic Notations

**Figure A.1**

Alternative notations. (a) Symbols for entity type/class, attribute, and relationship. (b) Displaying attributes. (c) Displaying cardinality ratios. (d) Various (min, max) notations. (e) Notations for displaying specialization/generalization.



# Q&A

