



Modul III

Build Logical Data Model

DBD 2018

By
Tim Asisten 2018



BUILD LOGICAL DATA MODEL

Deskripsi.

MEMETAKAN RELASI UNTUK LOGICAL DATA MODEL

User memastikan bahwa model yang dibuat telah sesuai dengan representasi dari perusahaan.

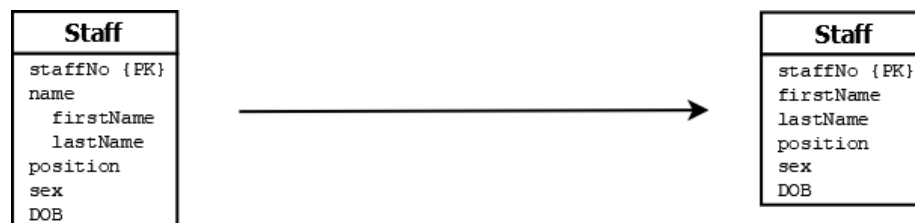
Objective : Membuat relasi untuk logical data model sehingga dapat merepresentasikan entitas, relasi, dan attribute yang telah didefinisikan.

How : Mengidentifikasi entitas “parent” dan “child” serta menentukan **Primary Key** dan **Foreign Key** pada suatu relasi.

Tahapan :

1. Strong Entity Types

Setiap *strong entity* dalam *data model*, buatlah relasi yang berisi seluruh atribut dalam *entity*. Untuk *composite attribute*, seperti nama, masukkan hanya komponen simple atributnya (*firstName* dan *lastName*) dalam relasi.



2. Weak Entity Types

Setiap *weak entity* dalam *data model*, buatlah relasi yang berisi seluruh atribut dalam entitas. *Primary Key* dari *weak entity* sebagian atau seluruhnya diturunkan dari *owner entity*, sehingga untuk mengidentifikasi *Primary Key* dari *weak entity* tidak bias dilakukan sampai seluruh relasi dengan *owner entities* telah dipetakan.



3. One-to-many (1:*) Binary Relationship Types

Setiap 1:* *binary relationship*, *entity* pada salah satu sisi relasi merupakan *parent entity* dan *entity* pada salah satu sisi yang lain merupakan *child entity*. Untuk merepresentasikan relasi ini, salin *Primary Key attributes* dari *parent entity* kedalam *child entity* untuk menjadi *Foreign Key*.

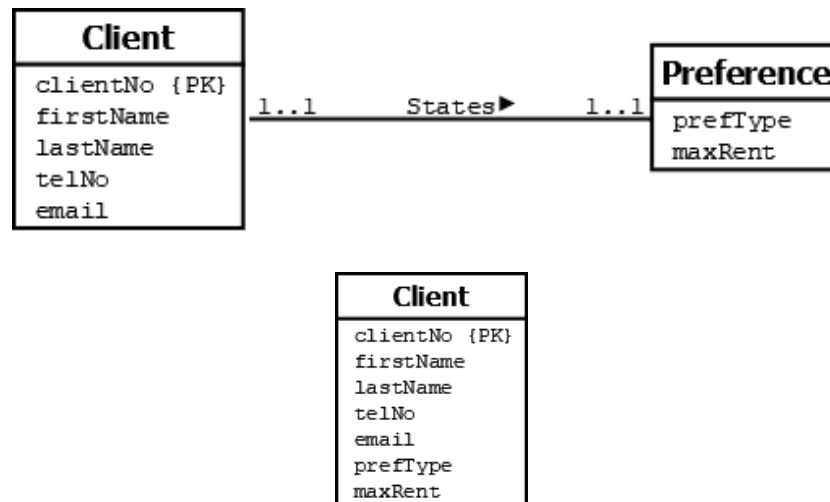


4. One-to-one (1:1) Binary Relationship Types

Membuat relasi untuk merepresentasikan 1:1 relationship sedikit lebih kompleks, karena *cardinality* tidak dapat digunakan untuk membantu mengidentifikasi *parent* dan *child entity* dalam relasi. Sebagai gantinya, *participation constraints* digunakan untuk membantu memutuskan mana *entity* yang paling merepresentasikan relasi. Berikut cara untuk mempertimbangkan *participation constraint* dalam membuat relasi:

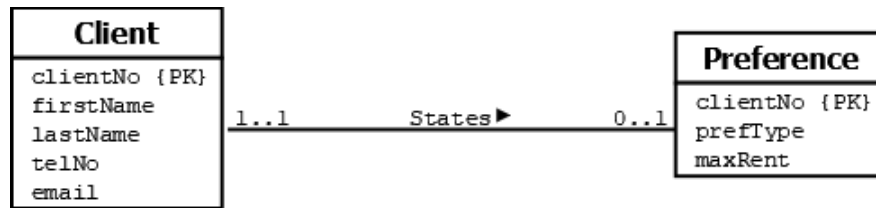
a.) Mandatory participation on both sides of 1:1 relationship

Pada relasi 1:1 dengan *mandatory participation* pada kedua sisi dengan satu atau lebih atribut, maka atribut ini dapat digabungkan. Sebagai catatan, penggabungan dua *entity* dapat dilakukan ketika tidak ada relasi secara langsung antara dua *entity* yang mencegah, seperti 1:* relationship (semisal entitas terkait melalui dua hubungan yang berbeda).



b.) Mandatory participation on one side of 1:1 relationship

Pada kasus ini kita dapat melakukan identifikasi terhadap *parent* dan *child entities* untuk relasi 1:1 menggunakan *participation constraints*. *Entity* dengan *optional participation* pada relasi dipilih menjadi *parent entity* dan satunya menjadi *child entity*. Seperti yang dijelaskan sebelumnya, salin *Primary Key* dari *parent entity* kepada *child entity* untuk menjadi *Foreign Key* yang merepresentasikan relasi kedua *entity*.

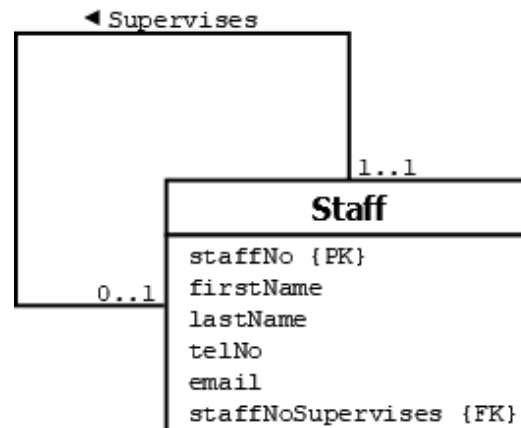


c.) *Optional* participation on *both* sides of 1:1 relationship

Pada kasus ini kita dapat memilih *parent* dan *child entity* dengan sewenang-wenang, kecuali kita menemukan relasi lebih yang dapat membantu kita untuk membuat keputusan yang lain.

5. One-to-one (1:1) Recursive Relationship Types

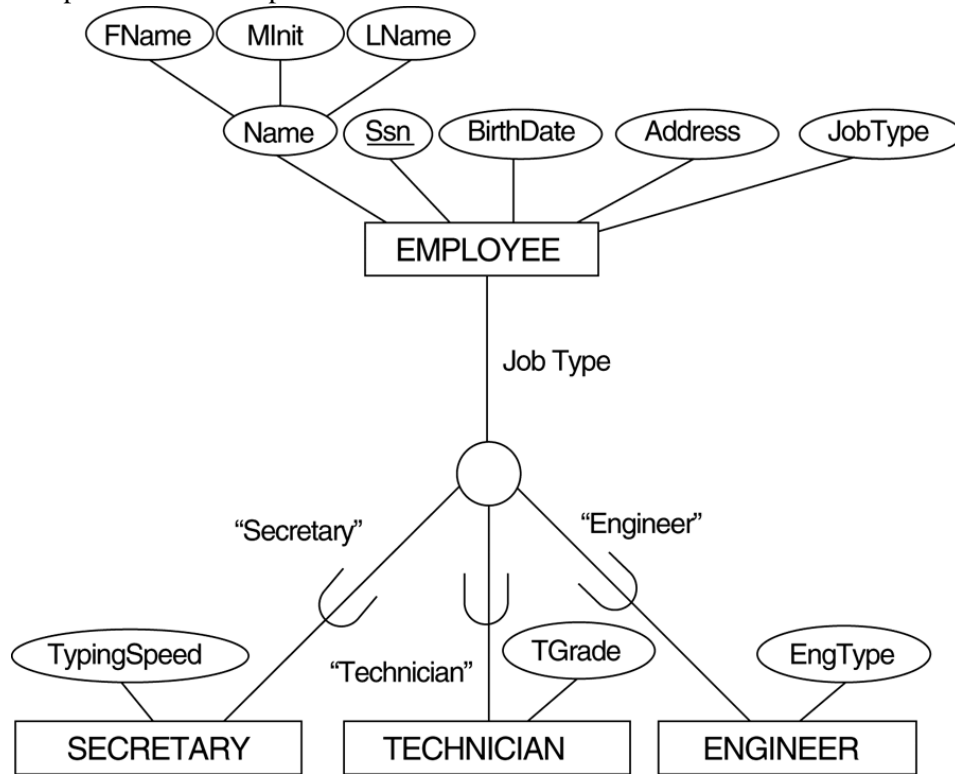
Pada *recursive relationship*, ikuti aturan *participation* yang dijelaskan sebelumnya. Untuk 1:1 *recursive relationship* dengan *mandatory participation* pada kedua sisi, representasikan *recursive relationship* sebagai sebuah relasi dengan menyalin *Primary Key* kedalam *entity* dan ubah namanya yang dapat mengindikasikan representasi dari relasi tersebut.



6. Superclass / Subclass Relationship Types

Setiap *superclass/subclass relationship*, identifikasi *superclass entity* sebagai *parent entity* dan *subclass entity* sebagai *child entity*. Berikut macam-macam pilihan dalam merepresentasikan relasi:

a.) Multiple relations – Superclass & subclass



(a) EMPLOYEE

<u>SSN</u>	FName	MInit	LName	BirthDate	Address	JobType
------------	-------	-------	-------	-----------	---------	---------

SECRETARY

<u>SSN</u>	TypingSpeed
------------	-------------

TECHNICIAN

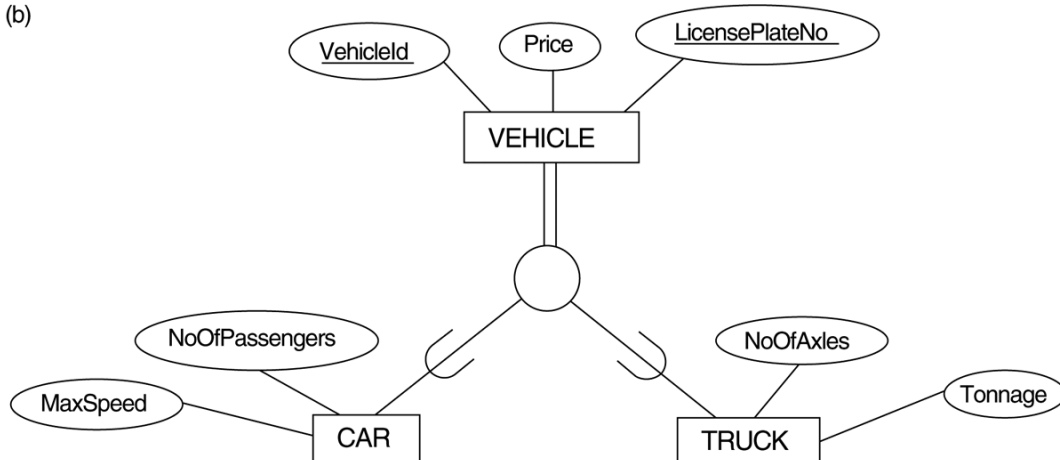
<u>SSN</u>	TGrade
------------	--------

ENGINEER

<u>SSN</u>	EngType
------------	---------

b.) Multiple relations – Subclass relations only

(b)



(b) CAR

<u>VehicleId</u>	LicensePlateNo	Price	MaxSpeed	NoOfPassengers
------------------	----------------	-------	----------	----------------

TRUCK

<u>VehicleId</u>	LicensePlateNo	Price	NoOfAxles	Tonnage
------------------	----------------	-------	-----------	---------

c.) Single relation – one type attribute

(c) EMPLOYEE

<u>SSN</u>	FName	MInit	LName	BirthDate	Address	JobType	TypingSpeed	TGrade	EngType
------------	-------	-------	-------	-----------	---------	---------	-------------	--------	---------

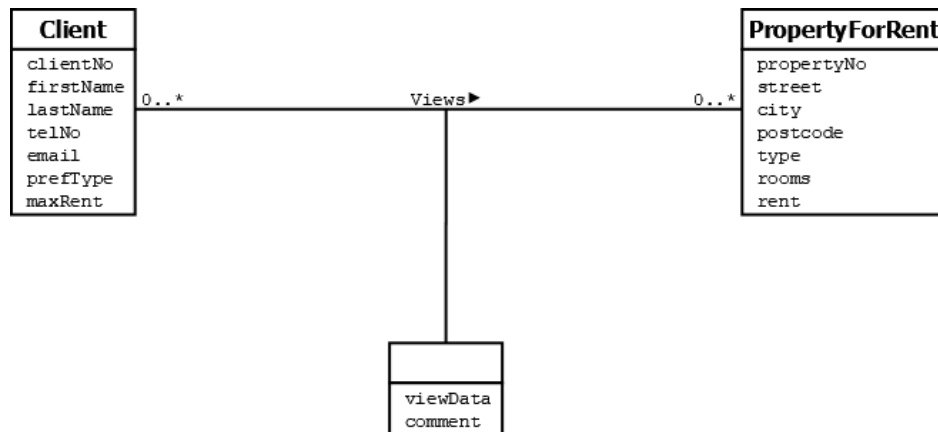
d.) Single relation – mutiple type attribute

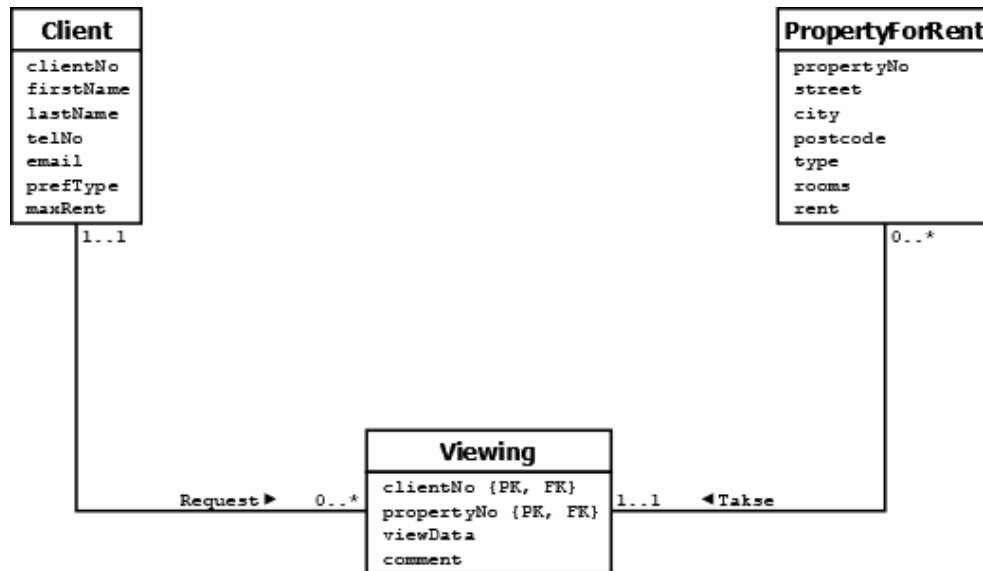
(d) PART

<u>PartNo</u>	Description	MFlag	DrawingNo	ManufactureDate	BatchNo	PFlag	SupplierName	ListPrice
---------------	-------------	-------	-----------	-----------------	---------	-------	--------------	-----------

7. Many-to-many (*:*) Binary Relationship Types

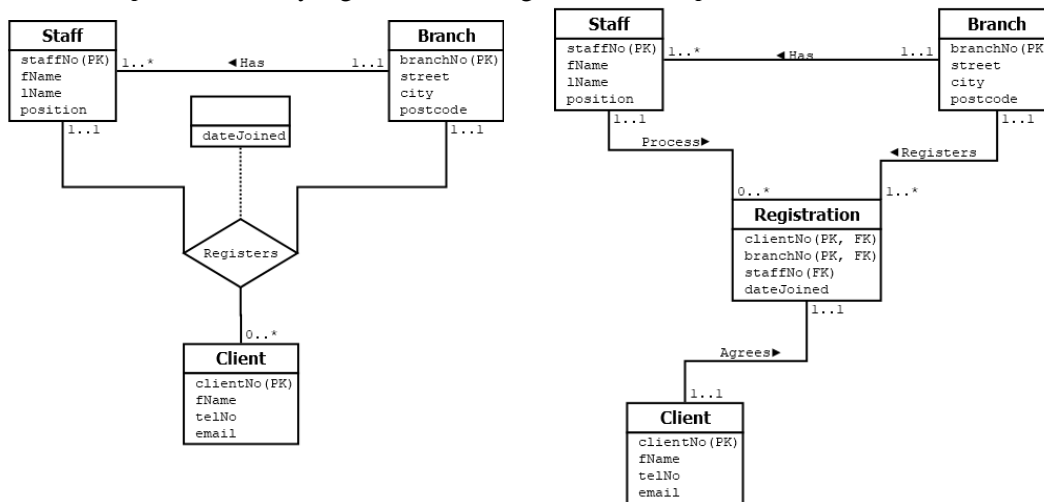
Setiap *: * *binary relationship*, buatlah relasi yang merepresentasikan *relationship* dan berisi semua atribut bagian dari *relationship*. Salin *Primary Key* dari *entities* yang ada dalam pada relasi kedalam relasi baru sebagai *Foreign Key*.





8. Complex Relationship Types (N-ary relationships)

Untuk setiap *complex relationship (N-ary relationships)* buat sebuah relasi yang merepresentasikan *relationship* dan atribut yang berkaitan dengan *relationship* tersebut.



9. Multi-valued Attributes

Untuk setiap *multivalued attribute* dalam sebuah entitas buat sebuah relasi baru yang merepresentasikan *multi-valued attribute* tersebut, termasuk *primary key* dari *entity* dalam relasi baru yang digunakan sebagai *foreign key*. Kecuali *multi-valued attribute* tersebut merupakan *alternate key* dari *entity*. *Primary key* pada relasi baru merupakan kombinasi dari *multi-valued attribute* dan *primary key* dari *entity*.

