

BAB 9

DATA MODELLING

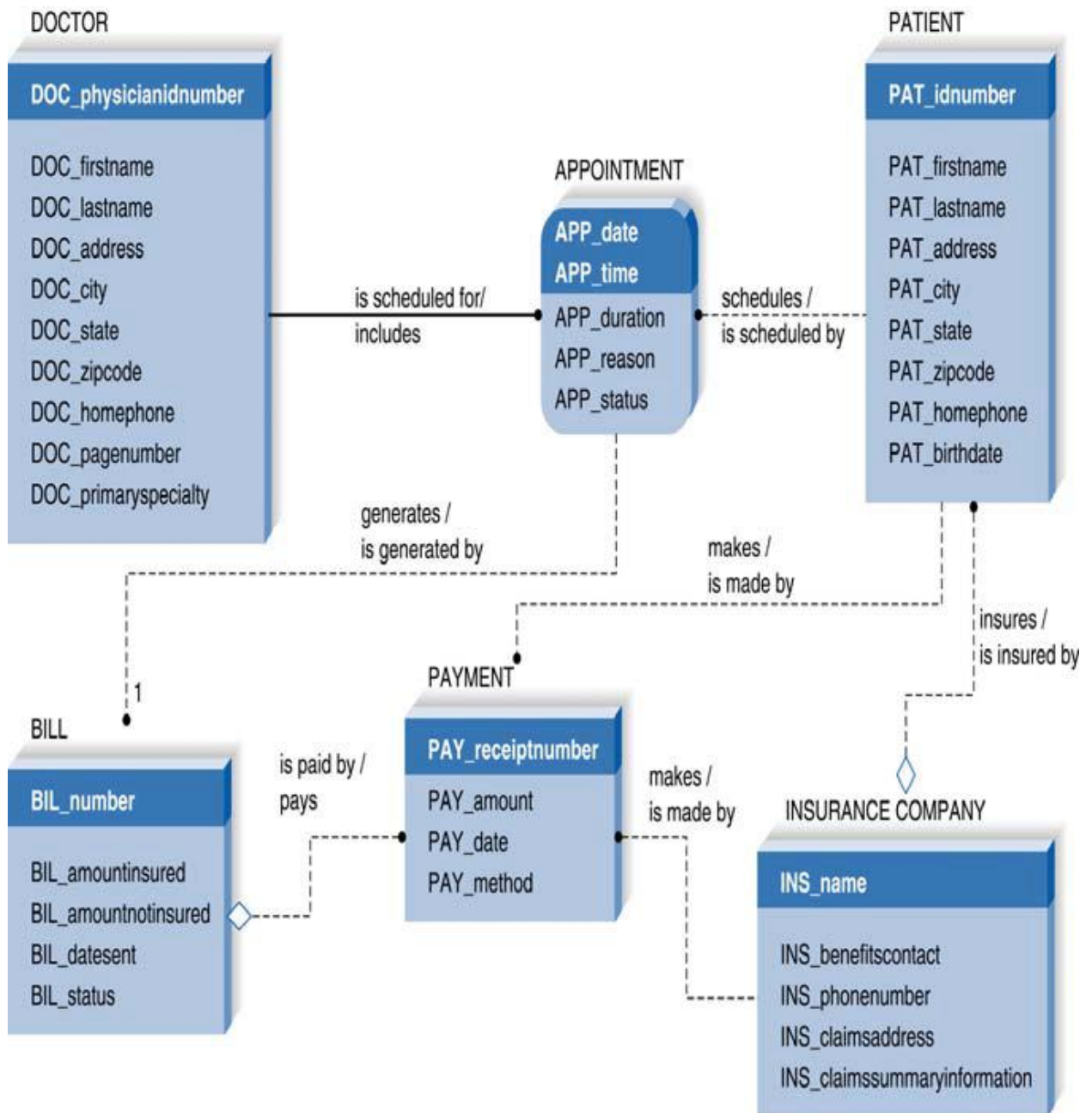
9.1 DATA MODEL

Data model adalah cara formal untuk menggambarkan data yang digunakan dan diciptakan dalam suatu sistem bisnis. Model ini menunjukkan orang, tempat atau benda dimana data diambil dan hubungan antar data tersebut. Data modelling juga dibedakan menjadi 2: **Logical data model**: Menunjukkan pengaturan data tanpa mengindikasikan bagaimana data tersebut disimpan, dibuat dan dimanipulasi. **Physical data model**: Menunjukkan bagaimana data akan disimpan sebenarnya dalam database atau file. Penyusunan model data harus seimbang dengan model proses. Salah satu cara pemodelan data adalah dengan ERD (Entity Relationship Diagram).

9.2 THE ENTITY-RELATIONSHIP DIAGRAM (ERD)

Apakah ERD itu?




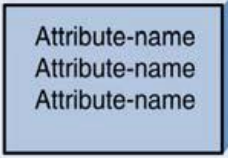

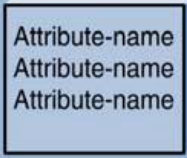
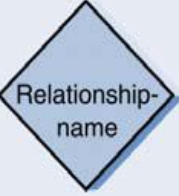
- 1) Gambar yang menunjukkan informasi dibuat disimpan dan digunakan dalam sistem bisnis.
- 2) Entitas biasanya menggambarkan jenis informasi yang sama
- 3) Garis yang menghubungkan antar entitas menunjukkan hubungan antar data.
- 4) ERD juga bisa digunakan untuk menunjukkan aturanaturan bisnis Menggunakan ERD untuk menunjukkan aturan bisnis
- 5) Aturan bisnis adalah batasan yang harus diikuti ketika sistem beroperasi.
- 6) Simbol ERD hanya menunjukkan satu instance dari entitas harus ada sebelum instance lain dari suatu entitas. Sebagai contoh: Seorang dokter harus ada sebelum perjanjian ketemu dengan dokter dibuat.
- 7) Simbol ERD dapat menunjukkan ketika salah satu instance dari suatu entitas dapat direlasikan dengan satu atau lebih instance dari entitas lainnya. Contohnya: satu dokter bisa memiliki banyak pasien, satu pasien bisa jadi hanya memiliki satu dokter utama.
- 8) Simbol ERD juga menunjukkan ketika eksistensi dari instance suatu entity adalah opsional untuk sebuah relasi dengan instance dari suatu entitas. Contohnya: Pasien mungkin memiliki atau mungkin tidak memiliki biaya asuransi. **Berikut ini contoh sebuah ERD**



Gambar 9.1 Contoh Diagram ERD

Elemen-elemen ERD

Seperti data flow diagram, ERD juga menggunakan simbol-simbol khusus untuk menggambarkan elemen-elemen ERD. Berikut ini simbol-simbol yang digunakan dalam ERD:

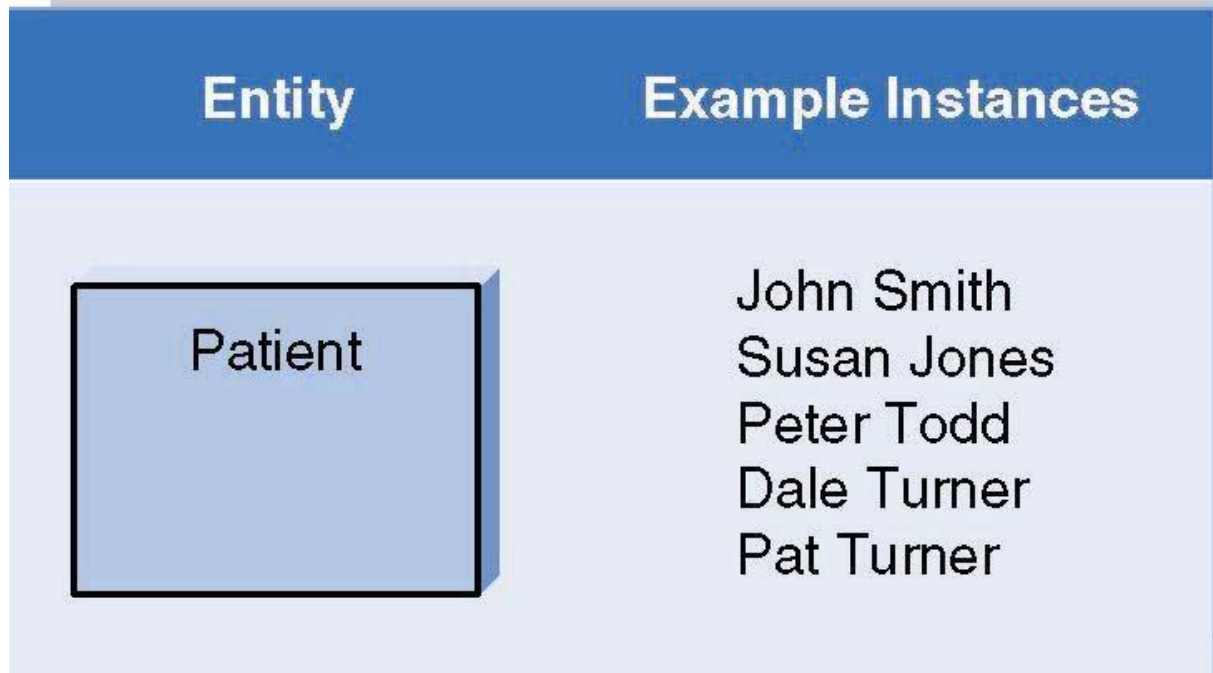
	IDEF1X	Chen	Information Engineering
<p>An ENTITY:</p> <ul style="list-style-type: none"> ✓ Is a person, place, or thing ✓ Has a singular name spelled in all capital letters ✓ Has an identifier ✓ Should contain more than one instance of data 	<p>ENTITY-NAME</p> 	<p>ENTITY-NAME</p> 	<p>ENTITY-NAME</p> 
<p>An ATTRIBUTE:</p> <ul style="list-style-type: none"> ✓ Is a property of an entity ✓ Should be used by at least one business process ✓ Is broken down to its most useful level of detail 	<p>ENTITY-NAME</p> 		<p>ENTITY-NAME</p> 
<p>A RELATIONSHIP:</p> <ul style="list-style-type: none"> ✓ Shows the association between two entities ✓ Has a parent entity and a child entity ✓ Is described with a verb phrase ✓ Has cardinality (1 : 1, 1 : N, or M : N) ✓ Has modality (null, not null) ✓ Is dependent or independent 	<p><u>Relationship-name</u></p>		<p><u>Relationship-name</u></p>

Gambar 9.2 Elemen-elemen dari ERD

Keterangan :

Entity

Entitas bisa berupa orang, kejadian, atau benda dimana data akan dikumpulkan. Untuk menjadi sebuah entity, suatu objek harus menampilkan beberapa kali event. Sebagai contoh : Jika sebuah firma hanya memiliki 1 gudang, maka gudang tersebut bukan entitas. Tetapi jika perusahaan memiliki banyak gudang, maka gudang bisa menjadi entitas suatu entitas jika perusahaan ingin menyimpan data untuk setiap instance dari gudang.



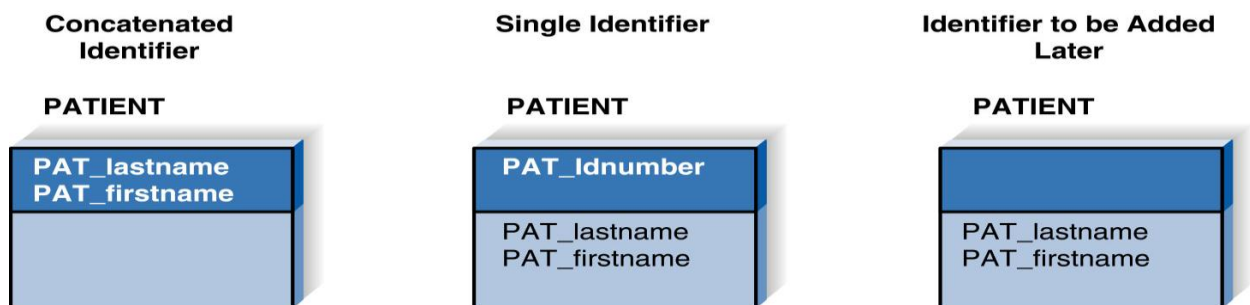
Gambar 10.3 Contoh Entitas dan Instance

Atribut

- 1) Informasi yang diambil tentang sebuah entitas
- 2) Hanya yang digunakan oleh organisasi yang dimasukkan dalam model
- 3) Nama atribut harus merupakan kata benda
- 4) Kadang nama entitas diletakkan di depan nama atribut untuk ketelitian.

Identifier

- 1) Satu atau lebih atribut dapat menjadi identifier entitas, yang secara unik mengidentifikasi setiap instance dari entitas.
- 2) Concatenated identifier (identifier gabungan) terdiri dari beberapa atribut.
- 3) Identifier bisa saja artifisial, seperti dengan membuat ID number
- 4) Identifier tidak akan dikembangkan sampai fase desain.



Gambar 10.4 tipe-tipe identifier

Berikut ini contoh penggunaan identifier dalam sebuah case repository untuk suatu atribut :

The screenshot shows a form for defining a case repository attribute. The fields are as follows:

- Attribute Name:** PAT _ idnumber
- Aliases:** Patient ID Number, Social Security Number, SSN
- Entity:** PATIENT
- Description:** The ID number assigned by the doctor's office that uniquely identifies a patient.
- Data Type:** Alpha-Numeric
- Identifier:** Yes
- Notes:** Usually, the patient's social security number is used. When a patient does not have a social security number, the number assigned by the patient's insurance provider is used. If the patient does not have an insurance provider, a driver's license number is used.
- Notes:** If the patient's insurance provider number is used and it is greater than 9 characters, the first 9 characters are used to create this attribute. Zeros are added to the number if it is less than 9 characters.

Gambar 9.5 Case repository untuk atribut patient_SSN

Relationships

- 1) Hubungan antar entitas

- 2) Entitas pertama dalam relationship disebut entitas induk, entitas kedua disebut sebagai entitas anak.
- 3) Relationship harus mempunyai nama yang berupa kata kerja
- 4) Relationship berjalan 2 arah. Sebagai contoh, jika dimiliki dua entitas yaitu buku dan toko buku maka bisa dibuat beberapa relationship, diantaranya:
 - a) Toko buku memesan buku
 - b) Toko buku menampilkan buku
 - c) Toko buku menstock buku
 - d) Toko buku menjual buku
 - e) Toko Buku mengembalikan buku.

Relationship *memesan*, *menampilkan*, *menstock*, *menjual* dan *mengembalikan* mendefinisikan hubungan yang relevan antara buku dan toko buku.

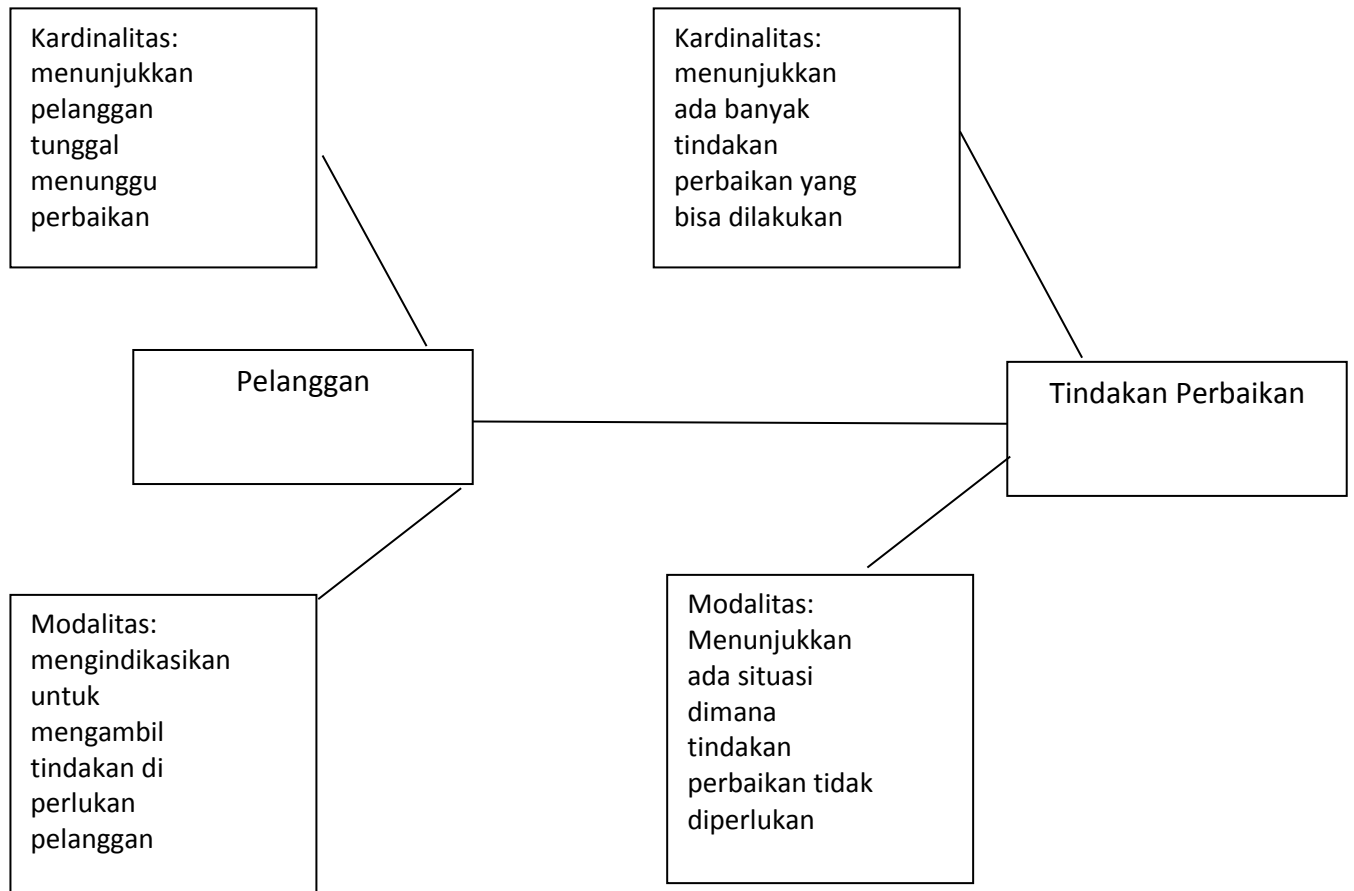
Cardinalitas

- 1) Kardinalitas mengacu pada berapa kali instance dari satu entitas dapat berelasi dengan instance lain di entitas yang berbeda.
- 2) Satu instance dalam 1 entitas mengacu pada satu dan hanya satu instance pada entitas lainnya (1:1)
- 3) Satu instance dalam suatu entitas mengacu ke satu atau lebih instance yang berelasi (1:N)
- 4) Satu atau lebih instance dalam suatu entitas mengacu pada satu atau lebih instance pada entitas yang berelasi (M:N).

Modalitas

- 1) Mengacu pada apakah suatu instance dari entitas anak dapat ada tanpa suatu relasi dengan instance dari entitas induk atau tidak.
- 2) **Not Null** berarti bahwa suatu instance pada entitas yang berelasi harus ada untuk suatu instance dari entitas lain untuk disebut valid
- 3) **Null** berarti bahwa tidak ada instance dalam entitas yang berelasi yang diperlukan untuk instance pada relasi lain untuk dikatakan valid.

Berikut ini contoh penggunaan kardinalitas dan modalitas dalam suatu Relationship:



Gambar 9.6 Frame Metadata

Data Dictionary dan Metadata

- 1) Metadata adalah informasi yang tersimpan yang berisi komponen dari model data
- 2) Metadata disimpan dalam data dictionary sehingga bias dibagi dengan developer dan user melalui SDLC
- 3) Data dictionary yang lengkap dan bisa dibagi membantu meningkatkan kualitas dari sistem yang sedang dikembangkan.

Berikut ini tabel frame umum dari sebuah metadata:

Sebuah entitas yang bias dideskripsikan sebagai:	Sebuah Atribut yang bisa Dideskripsikan sebagai:	Sebuah Relationship yang Bisa dideskripsikan sebagai:
1) Nama 2) Definisi 3) Catatan Khusus 4) Kontak User 5) Kontak Analis	1) Nama 2) Deskripsi 3) Alias 4) Contoh nilai 5) Nilai yang bisa 6) diterima 7) Format 8) Tipe 9) Catatan khusus	1) Frase kata kerja 2) Entitas induk 3) Entitas anak 4) Definisi 5) Kardinalitas 6) Modalitas

MEMVALIDASI ERD

Untuk membuat ERD, diperlukan latihan dan jam terbang, Ada beberapa pedoman yang perlu diperhatikan untuk membuat ERD diantaranya:

- 1) Entitas harus memiliki banyak kejadian/realitas
- 2) Hindari penggunaan atribut yang tidak perlu
- 3) Berilah label yang jelas untuk semua komponen
- 4) Pasangkan kardinalitas dan modalitas yang jelas dan benar
- 5) Pecah atribut menjadi level serendah mungkin yang diperlukan
- 6) Label harus merefleksikan istilah-istilah bisnis yang umum
- 7) Asumsi harus disebutkan dengan jelas.

Normalisasi

Normalisasi adalah teknik yang digunakan untuk memvalidasi model data. Serangkaian aturan diberlakukan pada data model logik untuk meningkatkan pengaturannya. Biasanya digunakan 3 aturan.

Langkah-langkah normalisasi

Berikut ini adalah langkah-langkah yang digunakan untuk melakukan normalisasi terhadap

	diperlukan penambahan relasi untuk menghubungkan entitas baru dan lama. Tidak: Model data dalam bentuk 2NF (2 Normal Form)
--	--

data

model yang telah kita peroleh :

	0 normal form
Apakah ada atribut yang memiliki nilai ganda untuk satu instance dari suatu entitas?	Ya: Hilangkan atribut yang berulang dan grup yang berulang. Buat entitas yang menggambarkan atributatributnya. Biasanya diperlukan penambahan relasi untuk menghubungkan entitas baru dan lama. Tidak : Model data ada dalam bentuk 1NF (1 Normal Form)
	1 Normal Form
Apakah identifier terdiri dari lebih dari satu atribut? Jika ya. Apakah nilai atribut tergantung hanya pada satu bagian dari identifier?	Ya: Hilangkan ketergantungan parsial. Hilangkan atribut suatu entitas dimana nilai-nilai mereka tergantung pada ke semua identifier. Biasanya

	2 Normal Form
Apakah ada nilai-nilai atribut yang tergantung pada entitas yang bukan identifier?	<p>Ya: Hilangkan ketergantungan transitif atau entitas turunan. Pindahkan atribut ke entitas dimana atribut tersebut bergantung pada identifier. Biasanya diperlukan penambahan relasi untuk menghubungkan entitas baru dan lama.</p> <p>Tidak : Model data ada dalam bentuk 3NF (1 Normal Form)</p>
	3 Normal Form

Unnormalized Entity

Mulai dengan entitas dari model data log

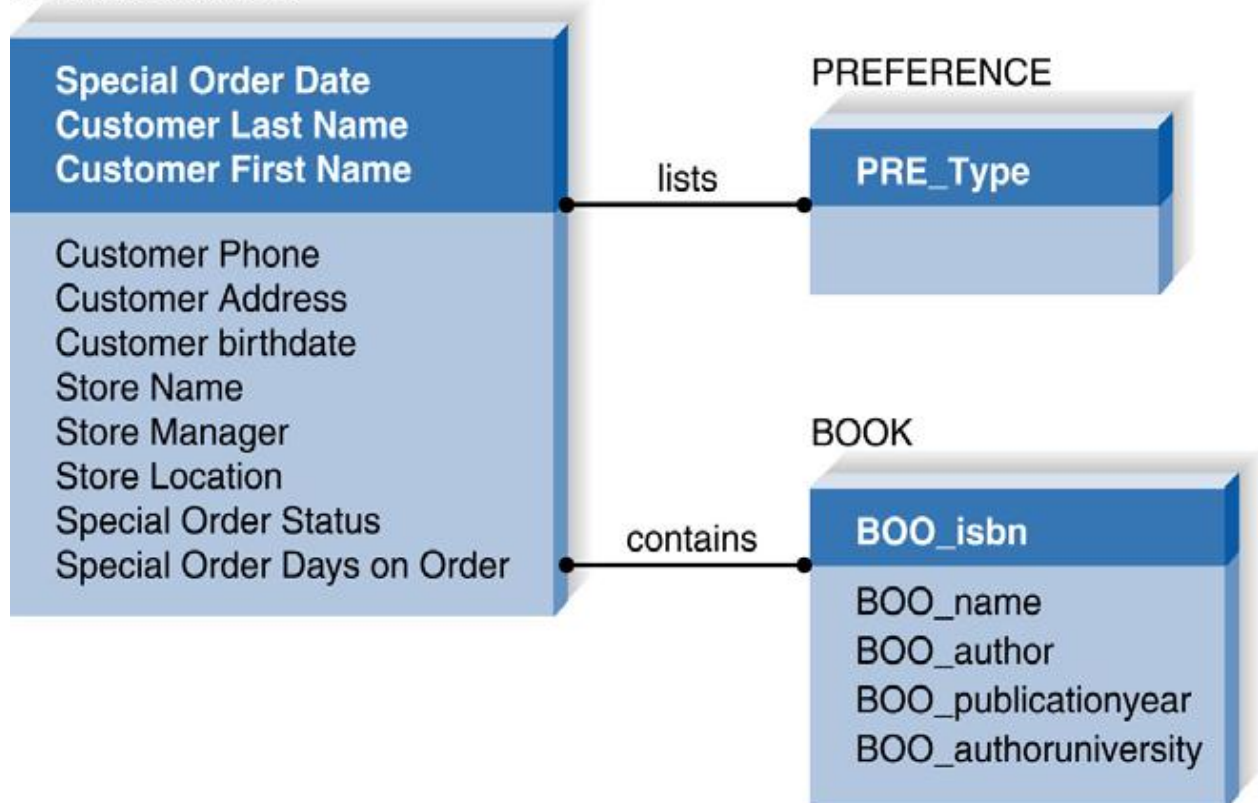
SPECIAL ORDER

Special Order Date
Customer Last Name
Customer First Name
Customer phone
Customer Address
Customer birthdate
Customer Book Preferences
Book ISBN1
Book Name1
Book Author1
Book Publication Year1
Book Author University1
Book ISBN2
Book Name2
Book Author2
Book Publication Year2
Book Author University2
Book ISBN3
Book Name3
Book Author3
Book Publication Year3
Book Author University3
Store Name
Store Manager
Store Location
Special Order Status
Special Order Days on Order

First Normal Form (1NF)

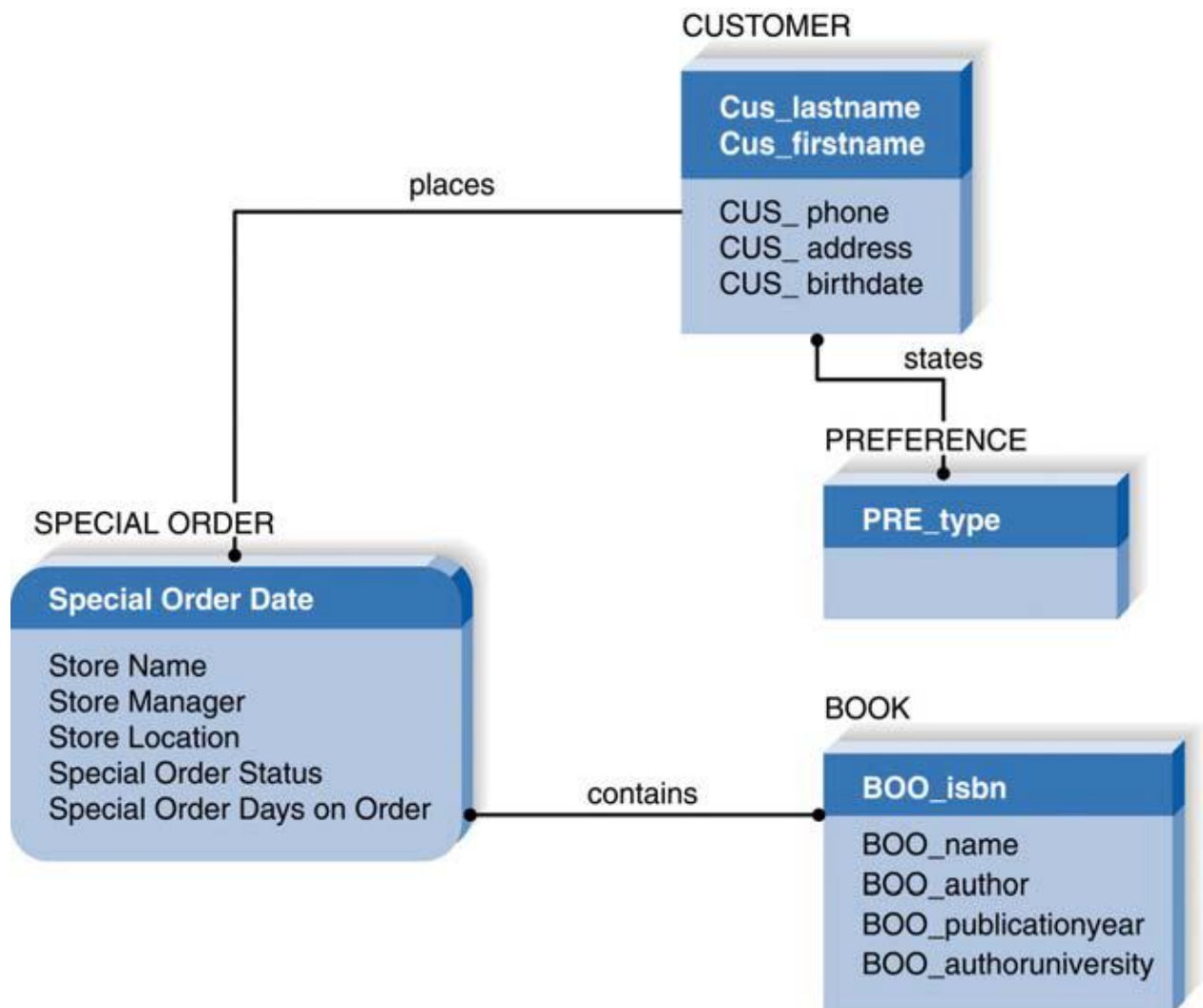
Cari kelompok-kelompok entitas yang berulang dan pisahkan ke dalam entitas yang berbeda.

SPECIAL ORDER



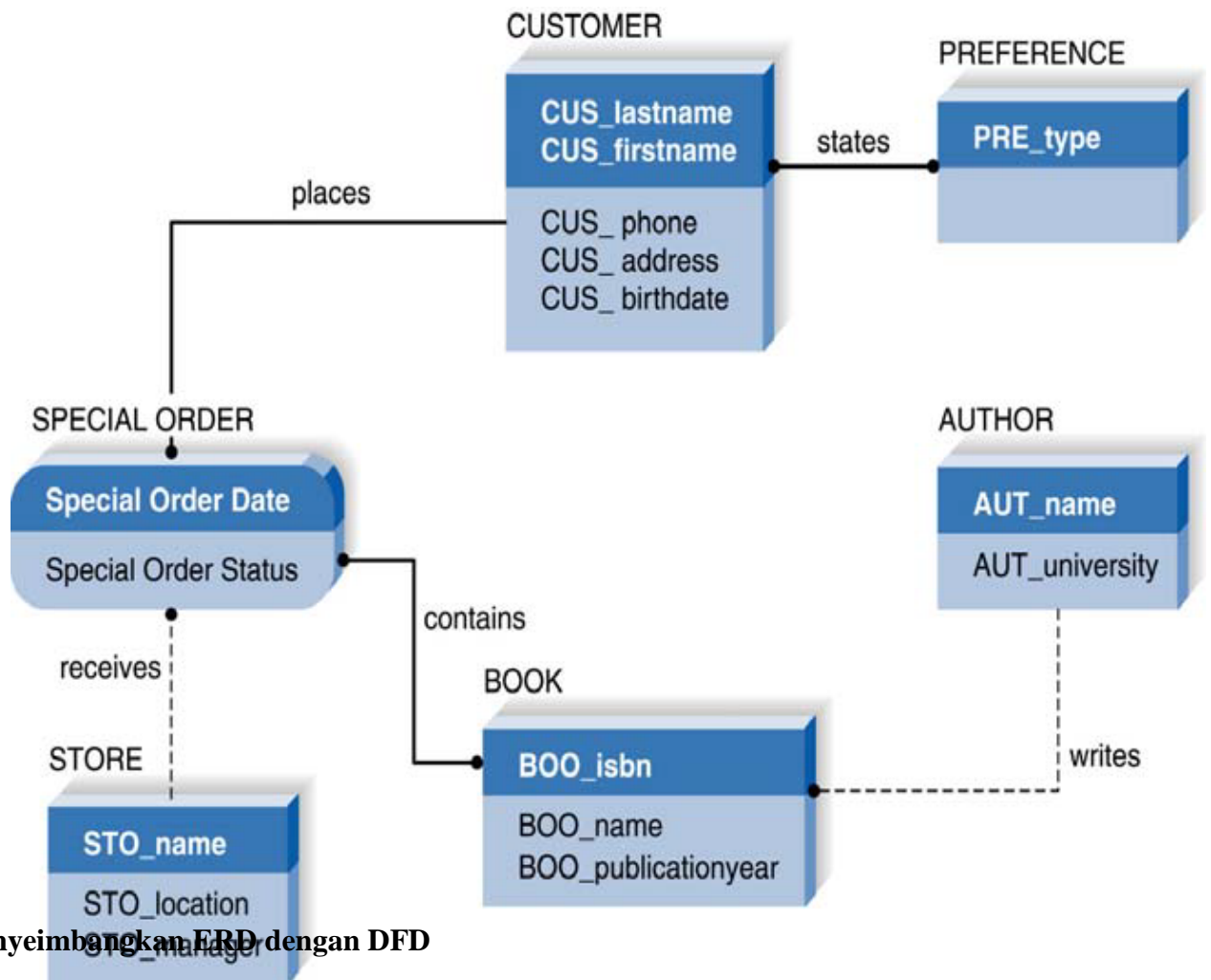
Second Normal Form (2NF)

Jika ada entitas yang memiliki identifier gabungan, cari atribut yang hanya bergantung pada identifier . Jika ditemukan pindahkan ke entitas baru.



Third Normal Form (3NF)

Cari atribut yang bergantung hanya pada atribut lain yang bukan merupakan identifier. Jika ditemukan pindahkan menjadientitas baru, juga pindahkan atribut-atribut yang dirasa perludipindahkan.



Menyeimbangkan ERD dengan DFD

Semua aktifitas analisa merupakan aktifitas-aktifitas yang saling berkaitan, termasuk proses modelling dan data modelling. Proses model akan berisi dua hal data flow dan data store. Komponen data dalam DFD ini harus diseimbangkan dengan ERD dimana Data store diseimbangkan dengan entitas dan elemen data diseimbangkan dengan atribut. Untuk mempermudah, banyak tool CASE yang menyediakan fitur untuk mengecek ketidakseimbangan.