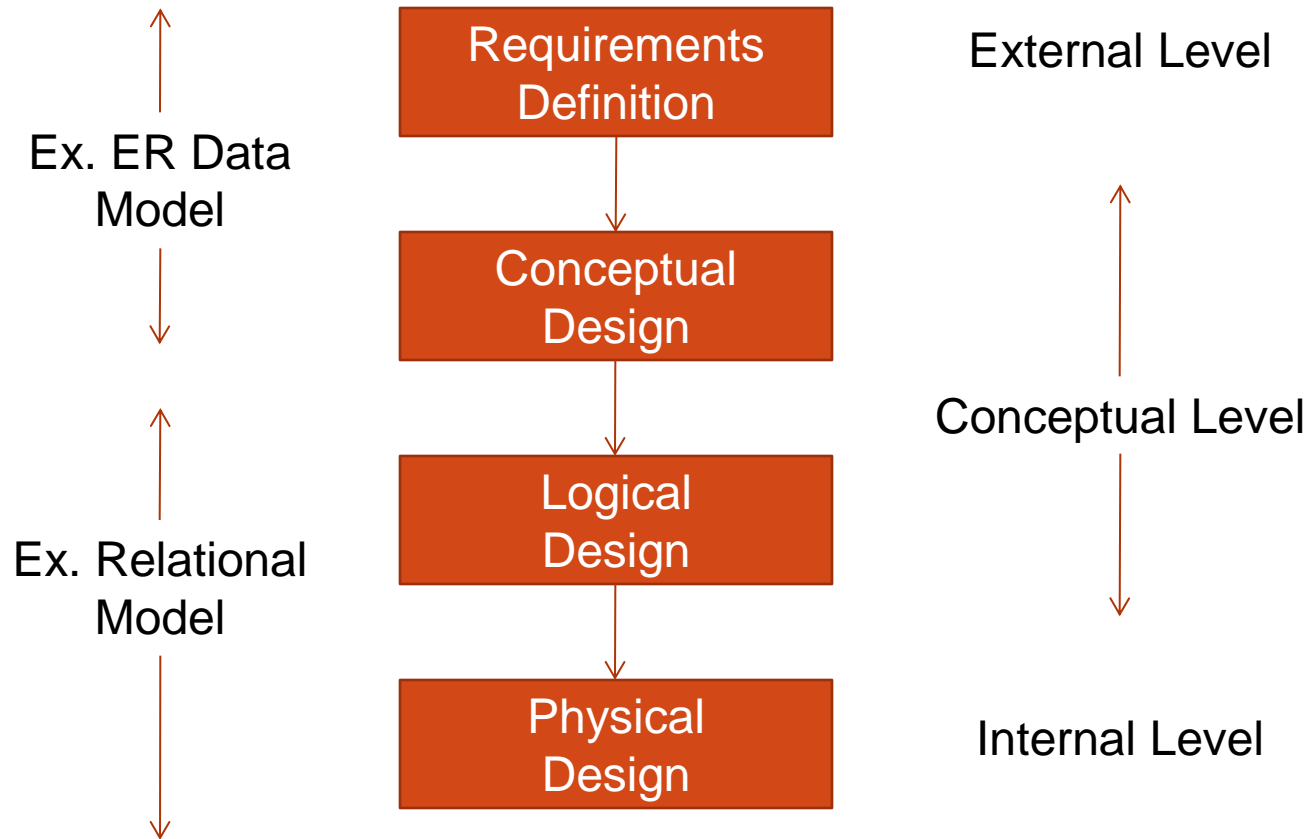


# Database Design II

TPI4210  
Sistem dan Teknologi Informasi  
[e-tp.ub.ac.id](http://e-tp.ub.ac.id)

# Database Design Life Cycle



# Properties of Relations

Sebuah *relation* memenuhi beberapa karakteristik dasar:

- Setiap baris adalah unik, tidak boleh ada 2 *tuple* atau lebih yang sama
- Setiap kolom memiliki nama
- Semua nilai dalam sebuah kolom adalah nilai dari sebuah *attribute*
- Urutan baris dan kolom tidak mutlak
- Setiap entri adalah tunggal, setiap *cell* mengandung sebuah entri tunggal
- Setiap nilai dapat diakses melalui nama tabel, *primary key* untuk baris bersangkutan dan nama kolom

# Relational Database

- Relational Database adalah kumpulan dari beberapa relation yang sudah di normalisasi.
- Normalisasi adalah sebuah tahap dalam desain database.
- Contoh Relational Database:

**order(order-id,date,)**

**order-line(order-id,product-id,quantity)**

**product-id(product-id, description, unit-price)**

# Normalization

**An important step in Logical  
Database Design**

# Data Normalization

- Sebuah teknik untuk membuat *relation-relation* dengan attribute seperti yang diinginkan berdasarkan kebutuhan data perusahaan.
- Sebuah metode formal untuk menganalisis *relation-relation* berdasarkan *primary key* (atau *candidate key*) dan *functional dependency*

# Data Normalization

- *Relation* harus dinormalisasi untuk menghindari beberapa jenis anomali yang mungkin terjadi pada saat update data.
- Normalisasi adalah serangkaian teknik sistematis yang mampu memperbaiki data model secara progresif.
- “Perburuan “ entitas yang tersembunyi

# Un-normalized Data

Sales(Cus-id, Name, Ord-no, O-date, Prod-id, Descrip, Qty-ord)

Cus-id	Name	Ord-no	O-date	Prod-id	Descrip	Qty-ord
C23	G.Gold	O56	15/4	P92	Chisel	6
C23	G.Gold	O56	15/4	P38	Plane	14
C75	R.Red	O57	16/4	P19	Saw	3
C19	B.Blue	O58	16/4	P33	Punch	24
C19	B.Blue	O58	16/4	P38	Plane	9
C23	G.Gold	O59	17/4	P19	Saw	10



# What's wrong with the Sales Relation?

- *Relation* ini menyimpan .....
- data customer, order dan product
- data penjualan
- Pengulangan data
- pencatatan berulang:
  - G.Gold memiliki cust-id 23
  - Order O56 dilakukan pada 15 april
  - Product P19 adalah Saw
- terbuangnya space pada media penyimpanan
- inkonsistensi data, terdapat beberapa versi data

# The Update Anomalies of the “Sales” File

- *Modification anomaly* – perubahan nilai sebuah *attribute* mengakibatkan perubahan pada seluruh record dimana *attribute* itu ada.  
Ex. G.Gold diganti H.Gold
- *Insertion anomaly* – input data baru tertunda  
Ex. seorang customer baru tidak bisa di input sebelum ia memasukkan order
- *Deletion anomaly* – jika suatu record dihapus, maka semua data yang tersimpan dalam record tersebut akan hilang.  
Ex. data customer akan hilang jika order dihapus

# The Update Anomalies

*Modification anomaly*



**Employees' Skills**

Employee ID	Employee Address	Skill
426	87 Sycamore Grove	Typing
426	87 Sycamore Grove	Shorthand
519	94 Chestnut Street	Public Speaking
519	96 Walnut Avenue	Carpentry

*Insertion anomaly*



**Faculty and Their Courses**

Faculty ID	Faculty Name	Faculty Hire Date	Course Code
389	Dr. Giddens	10-Feb-1985	ENG-206
407	Dr. Saperstein	19-Apr-1999	CMP-101
407	Dr. Saperstein	19-Apr-1999	CMP-201

424	Dr. Newsome	29-Mar-2007	?
-----	-------------	-------------	---

*Deletion anomaly*



**Faculty and Their Courses**

Faculty ID	Faculty Name	Faculty Hire Date	Course Code
389	Dr. Giddens	10-Feb-1985	ENG-206
407	Dr. Saperstein	19-Apr-1999	CMP-101
407	Dr. Saperstein	19-Apr-1999	CMP-201

424	Dr. Newsome	29-Mar-2007	?
-----	-------------	-------------	---

# Necessary Terminology

## *Functional dependence*

- Sebuah *attribute* B dikatakan tergantung secara fungsional (*functionally dependent*) pada *attribute* A jika setiap nilai A terkait dengan satu nilai B (*exactly*).
- A dan B mungkin merupakan sebuah *composite attribute* (Ex: Address, domain berbeda-beda dalam satu attribute)
- Dapat juga dikatakan jika A menentukan B, sehingga A adalah *determinant* bagi B

# Normalization

- Normalisasi terdiri dari beberapa langkah
- Setiap langkah memperbaiki data model dan mengurangi *update anomaly*
- Secara resmi normalisasi yang dilakukan meliputi
  - First Normal Form (1NF)
  - Second Normal Form (2NF)
  - Third Normal Form (3NF)
  - Boyce-Codd Normal Form (BCNF)
  - Fourth & Fifth Normal Form (4NF, 5NF)
  - Higher Normal Forms

# First Normal Form (1NF)

- 1NF didesain untuk menghilangkan *repeating groups* (perulangan dalam *attribute* atau kelompok *attribute*).
- kelompok yang terulang diberi simbol { } dalam relation

**CUSTOMER(cus-id,surname,initials,{order-no,order-date})**

Kombinasi order-no, order-date dapat terjadi 0, 1 atau beberapa kali untuk seorang customer, yang menyebabkan redundancy dan update anomaly

# First Normal Form (1NF)

- Buat sebuah relation baru dari item yang terulang dan key dari relation asal.

CUSTOMER(cus-id, surname, initials, {order-no, order-date})

Menjadi .....

CUSTOMER(cus-id, surname, initials)

ORDER(cus-id, order-no, order-date)

Relation diatas kini dalam bentuk 1NF

*“perpotongan tiap baris dan kolom berisi hanya satu nilai”*

# Second Normal Form (2NF)

- 2NF didesain untuk *menghilangkan partial key dependency*
- *Partial key dependency* terjadi bila
  - *primary key* dalam *relation* dalam bentuk *composite key*
  - ada *attribute non primary key* yang secara fungsional tergantung pada hanya sebagian dari *primary key*
- *Partial key dependency* menyebabkan terjadinya *update anomaly*



# Second Normal Form (2NF)

- *Partial key dependency:*

**ORDER-LINE(ord-no, prod-id, description, quantity)**

- Dalam *relation* ini, *prod-code* adalah satu-satunya yang menentukan *description*, sehingga tidak dibutuhkan seluruh bagian *primary key* untuk mengidentifikasi *attribute description*. sehingga ini adalah bentuk *partial key dependency*
- Sebuah relasi dalam bentuk 2NF jika:
  - sudah dalam 1NF dan
  - seluruh *partial key dependency* sudah dihilangkan

# Second Normal Form (2NF)

Untuk mendapatkan 2NF

**ORDER-LINE(ord-no, prod-id, description, quantity)**

- Buat sebuah *relation* baru dari *attribute* yang tidak memerlukan *primary key* secara penuh

**ORDER-LINE(ord-no,prod-id,quantity)**

**PRODUCT(prod-id,description)**

- *Relation* dalam 2NF sudah dalam 1NF dan setiap *attribute non primary key* tergantung secara penuh pada *primary key*

# Third Normal Form (3NF)

- 3NF didesain untuk menghilangkan *transitive key dependency*
- *Transitive key dependency* muncul apabila:  
terdapat *attribute* yang ternyata lebih tepat diidentifikasi oleh attribute lain daripada oleh *primary key*
- *Transitive key dependency* menyebabkan terjadinya update anomaly

# Third Normal Form (3NF)

- *Transitive key dependency*

**ORDER(ord-no,ord-date,cus-id,name,street,suburb)**

name, street, suburb lebih tepat diidentifikasi oleh cust-id daripada oleh *primary key relation* (ord-no)

- Untuk mendapatkan 3NF:

buat sebuah *relation* dari *attribute* yang teridentifikasi oleh attribute lain selain *primary key*

**ORDER(ord-no,ord-date,cus-id)**  
**CUSTOMER(cus-id,name,street,suburb)**

# Third Normal Form (3NF)

Relation dalam 3NF jika:

- Sudah dalam bentuk 2NF dan
- Semua *transitive key dependency* telah dihilangkan

Self study:

- Fourth and Fifth Normal Form
- Boyce-Codd Normal Form
- Higher Normal Forms