

# RANCANG BANGUN ALGORITMA TEKNIK NORMALISASI RELASI UNTUK PERANCANGAN BASIS DATA SAINS DIRGANTARA TERPADU DIRGANTARA LAPAN

Sarwito Agung Nugroho  
Pusat Pemanfaatan Sains Antariksa - LAPAN  
Jl. Dr. Djundjunaan 133 Bandung 40173  
Email : [agung@bdg.lapan.go.id](mailto:agung@bdg.lapan.go.id)

## Abstrak

*Normalisasi basis data merupakan proses penting yang dilakukan pada tahapan perancangan model relasional basis data untuk suatu system informasi. Pada tahapan ini dilakukan pemeriksaan terhadap setiap relasi pada perancangan struktur data file untuk mengetahui apakah suatu basis data akan mengalami redundansi atau tidak. Pemeriksaan tersebut meliputi pengujian berbagai aturan teknik normalisasi yakni normalisasi bentuk I, II, III dan normalisasi Boyce-Codd. Untuk membantu mempermudah proses normalisasi basis data terutama bagi masyarakat awam maka perlu dibuat suatu perangkat lunak normalisasi basis data. Perangkat lunak ini diharapkan dapat membantu para peneliti dan perekayasa untuk merancang model basis data relasional. Hasil kajian ini ternyata cukup efektif dalam membantu tahapan perancangan basis data. Kajian Algoritma Perangkat lunak ini diharapkan akan menjadi kajian lengkap untuk pembangunan perangkat lunak TNBD (Teknik Normalisasi Basis Data).*

*Kata Kunci : Rancang Bangun, Algoritma, Normalisasi, Relasi, Basis data*

## 1. PENDAHULUAN

Normalisasi basis data merupakan langkah penting dalam perancangan model data suatu perancangan basis data. Tujuan pros normalisasi adalah untuk mendapatkan mendapatkan suatu rancangan basis data yang efisien dalam penyimpanan data dan efektif dalam pengaksesannya serta mengurangi terjadinya kehilangan informasi atau bahkan terjadinya redundansi / duplikasi data sehingga menyulitkan pada proses retrieving dan editing suatu record dan deleting. Proses normalisasi basis data ini meliputi pengaturan berbagai relasi data sehingga memenuhi syarat berbagai bentuk normal ( Normal Form ) yakni bentuk normal I , bentuk normal II , bentuk normal III serta bentuk normal Boyce-Codd. Proses normalisasi merupakan proses yang cukup panjang mulai 100

dari penentuan atribut atau field kunci dari suatu relasi data sampai pemecahan suatu relasi menjadi sub-sub relasi yang tidak mengandung redundansi sehingga memperkecil kapasitas memori bagi suatu file tetapi

tidak akan menghilangkan informasi kebergantungan antar atribut data. Proses ini membutuhkan ketrampilan dan pengetahuan yang cukup tentang aljabar relasional dan system pengaksesan basis data. Untuk mempermudah dan mempersingkat waktu pada tahapan perancangan basis data itu maka dirancang suatu perangkat lunak normalisasi basis data ini.

## 2. TINJAUAN PUSTAKA

### 2.1 Model Data

Model basis data relasional merupakan suatu model data yang menggambarkan relasi-relasi antar file serta struktur data yang ada pada setiap file dibuat dalam pada perancangan basis data. Model data adalah kumpulan berbagai piranti koseptual untuk menggambarkan data, relasi antar data dan makna/ semantic data serta konsistensi konstrain. Model data juga harus menggambarkan hubungan antar obyek pada dunia nyata serta berbagai lingkungan obyek data. Model data yang dirancang pada akhirnya menghasilkan suatu skema relasi yang merepresentasikan suatu basis data sehingga memungkinkan untuk dilakukannya proses penyimpanan data yang efisien tanpa adanya duplikasi data.

### 2.2 Entitas, Relasi dan Atribut

Entitas adalah obyek yang ada dan dapat dibedakan dari obyek-obyek yang lainnya baik bersifat nyata atau abstrak. Peralatan ionosonda, peneliti dan SPD adalah contoh dari berbagai entitas nyata dan transaksi pengamatan, transaksi pengaksesan data adalah entitas-entitas abstrak. Relasi data dapat berbentuk suatu entitas atau kaidah kaitan antar berbagai entitas dan secara konsep merupakan gambaran suatu struktur file. Relasi digambarkan sebagai suatu pasangan terurut berbentuk  $R^n$ , dimana n banyak atribut atau field yang ada pada relasi atau record file tersebut. Contoh suatu relasi pada basis data sains dirgantara LAPAN SPD ( ID\_SPD, Nama\_SPD, Alamat, Tahun\_pembangunan SPD) (1) Relasi akan digambarkan sebagai Tabel yang berbentuk matrik berdimensi dua dengan kolom berupa field dan baris merupakan record data. Digambarkan sebagai :

FILE SPD ( Stasiun Pengamat Dirgantara )

ID_SPD	Nama_SPD	Alamat_SPD	Tahun	Letak_Geografi

Atribut merupakan nama dari property atau sifat karakter yang di anggap penting pada suatu obyek. Kumpulan berbagai atribut akan mengkonstruksi suatu relasi secara konseptual atau suatu obyek secara nyata. Contoh atribut adalah ID\_SPD, Nama\_SPD ...dll. Atribut pada penyimpanan data dinyatakan sebagai field atau suatu kolom pada tabel diatas.

### 2.3 Atribut Kunci

Atribut kunci merupakan atribut yang dapat membedakan suatu pasangan terurut atau record dengan record yang lain. Secara teknis dengan diketahuinya nilai dari atribut kunci maka suatu record data dapat ditemukan dari jutaan record yang ada pada suatu file.

#### Kunci Relasi ( Relation Key )

X dinyatakan sebagai suatu himpunan bagian dari himpunan atribut-atribut A dari relasi  $R(A_1, A_2, A_3, \dots, A_n)$  yang memenuhi :

1.  $X \rightarrow A_1, A_2, A_3 \dots A_n$
2. Tidak ada Y himpunan bagian dari X sehingga  $y \rightarrow A_1, A_2, A_3 \dots A_n$

ID\_SPD merupakan kunci relasi SPD

### 1.4 Kebergantungan Fungsional

Konsep kebergantungan fungsional diperkenalkan oleh CODD [Codd 70] untuk mengkarakteristik mekanisme dekomposisi suatu relasi tanpa kehilangan informasi yakni :

#### Kebergantungan Fungsional ( Functional Dependency )

Jika  $R(A_1, A_2, A_3 \dots, A_n)$  suatu skema relasi dan X , Y merupakan himpunan bagian dari

$A = \{ A_1, A_2, A_3 \dots A_n \}$  dimana  $X \rightarrow Y$  atau Y bergantung secara fungsional terhadap X maka untuk setiap relasi  $r \in R$  dan setiap pasangan terurut  $t_1$  dan  $t_2$  berlaku :

$$\pi x(t_1) = \pi x(t_2) \Rightarrow \pi y(t_1) = \pi y(t_2)$$

$ID\_SPD \rightarrow Nama\_SPD$  dan  $ID\_Alat, Nama\_Alat \rightarrow Fungsi\_alat$

## 3. PROSES NORMALISASI

Suatu Proses penentuan apakah suatu relasi dalam model basis data relasional mengandung redundansi atau tidak merupakan proses normalisasi. Proses ini pada hakekatnya memeriksa kesesuaian antara kebergantungan fungsional yang terkandung pada suatu relasi dengan berbagai syarat kebergantungan fungsional yang memenuhi bentuk normal I, bentuk normal II , bentuk normal III serta bentuk normal Boyce Codd.

Pemeriksaan ini berjalan secara hirarki artinya bentuk normal ke  $i+1$  membutuhkan prasyarat bentuk normal ke  $i$  untuk  $i = 1, 2$ . Jadi suatu bentuk normal ke  $i$  akan membutuhkan prasyarat bentuk normal sebelumnya. Adapun konsep dasar bentuk normal adalah sebagai berikut :

### Bentuk Normal ke I

Sebuah relasi R dinyatakan memenuhi bentuk normal ke satu jika dan hanya jika setiap  $r \in R$  memiliki nilai tunggal atau atomic.

Artinya Suatu relasi R dikatakan memenuhi bentuk normal ke I apabila dalam setiap record atau pasangan terurut tidak terdapat pengulangan grup nilai atau hanya satu dan hanya satu harga.

Contoh kasus 1 :

Relasi untuk Entitas SPD ( Stasiun Pengamat Dirgantara ) berisi informasi mengenai SPD yang dimiliki LAPAN. Terdapat 5 atribut pada relasi Entitas SPD yakni ID\_SPD ( nomor SPD) merupakan kunci relasi, Nama SPD dan Alamat SPD serta Letak geografi SPD sebagai atribut non key atau atribut deskriptif /pelengkap relasi. Bentuk tabel relasi ini adalah :

ID_SPD	Nama_SPD	Alamat SPD	Tahun	Letak Geografi
010279	SPD Ponti I	Siantan Hulu, Pontianak kalbar	1979	00 BB 15' BT
010279	SPD Ponti II	Siantan hulu, Pontianak Kalbar	1980	00 BB 15 ' BT

Relasi ini tidak memenuhi bentuk normal I karena terdapat 2(dua) nilai atau pengulangan nilai pada atribut Nama\_SPD, Alamat\_SPD , Tahun pembangunan SPD dan Letak geografi SPD.

Untuk membuat bentuk relasi SPD memenuhi bentuk normal I maka ID\_Stasiun yang merupakan Kunci relasi harus dibedakan antara SPD ponti I dan SPD Ponti II menjadi 010279 dan 010280 bagi ID\_SPD.

Contoh kasus 2 :

Relasi Peralatan Pengamat dirgantara berisi tentang berbagai informasi mengenai peralatan yang ada distasiun atau yang dipasang pada beberapa stasiun sekaligus. Relasi ini memiliki paling tidak 9 (Sembilan) atribut yakni ID\_peralatan, Nama\_alat, Fungsi\_alat, ID\_SPD1, Nama\_SPD1, ID\_SPD2, Nama\_SPD2, ID\_SPD3, Nama\_SPD3.

Nama-nama SPD dan ID SPd adalah lokasi dimana alat tersebut dipasang / di install dan maksimal 3 SPD berbeda.

ALAT( ID\_Alrat, Nama\_Alrat, Fungsi \_Alrat, (ID\_SPD ,Nama\_SPD)\*) atau dengan bentuk lengkap

ALAT(ID\_Alrat,Nama\_alrat,Fungsi\_alrat ,ID\_SPD1, Nama\_SPD1, ID\_SPD2, Nama\_SPD2, ID\_SPD3, Nama\_SPD3)

ID\_alrat → Nama\_Alrat, Fungsi\_alrat, (ID\_SPD, Nama\_SPD)\*

Terlihat bahwa terdapat repeating grup/ pengulangan nilai grup yakni ID\_SPD dan Nama\_SPD. Sebagai akibatnya penyimpanan record data alat menjadi tidak efisien karena jika suatu alat hanya dipasang pada satu SPD saja maka akan terdapat field ID\_SPD dan Nama\_SPD yang kosong

### Bentuk Normal ke II

Suatu Relasi R dinyatakan memenuhi bentuk normal II jika dan hanya jika

1. R memenuhi bentuk normal I
2. Setiap atribut yang bukan bagian atribut kunci tidak bergantung fungsional kepada bagian dari atribut kunci

Bentuk normal II pada hakekatnya mensyaratkan adanya ketentuan bahwa setiap atribut yang bukan atribut kunci harus bergantung fungsional secara penuh terhadap atribut kunci. Jika atribut kunci hanya terdiri dari satu dan hanya satu atribut tanpa memiliki atribut lain maka jika relasi R memenuhi bentuk normal I maka secara otomatis akan memenuhi bentuk normal II.

Contoh kasus :

*ID\_Alrat, Nama\_Alrat → Fungsi\_alrat dan Nama\_Alrat → Fungsi\_Alrat*

Relasi Alat (ID\_Alrat, Nama\_Alrat, Fungsi\_ALrat, Keterangan\_ALrat )

Dimana atribut kunci yang terdiri ID\_Alrat dan Nama\_Alrat. Atribut Fungsi Alat bergantung fungsional kepada bagian atribut kunci yakni atribut Nama\_alrat. Sehingga relasi Alat tidak memenuhi bentuk normal II.

Agar relasi alat memenuhi bentuk normal II maka haruslah dilakukan proses dekomposisi. Relasi Alat harus dipecah menjadi 2(dua) buah sub relasi yang berbeda tanpa kehilangan berbagai informasi yang berguna.

### Bentuk Normal ke III

Suatu Relasi R dinyatakan memenuhi bentuk norma ke III jika dan hanya jika

1. Relasi R memenuhi bentuk normal ke II
2. Setiap atribut yang bukan bagian atribut kunci tidak bergantung fungsional terhadap atribut bukan kunci lainnya.

Bentuk normal III mensyaratkan relasi R harus memenuhi bentuk normal II dan tidak adanya transitivitas diantara atribut-atribut yang ada pada relasi R.

Bentuk Normal Boyce-Codd

Suatu Relasi R dikatakan memenuhi bentuk normal boyce-Codd (BNBC) jika setiap kebergantungan fungsional elementer yang ada merupakan

kebergantungan fungsional antara atribut kunci dan atribut deskriptif ( bukan kunci )

Bentuk normal Boyce\_Codd menyatakan bahwa setiap kebergantungan fungsional yang ada pada relasi R hanyalah berbentuk  $A \rightarrow B$  dimana A atribut kunci dan B atribut bukan kunci. Secara konseptual hal ini menjelaskan bahwa satu-satunya kebergantungan fungsional merupakan kebergantungan fungsional yang sangat kuat antara atribut kunci dan atribut bukan kunci.

### 3. RANCANG BANGUN STRUKTUR DATA

Pada bagian ini akan dikaji struktur data yang akan dikembangkan bagi perangkat lunak yang akan dibangun yakni berbentuk struktur data link-list dengan tipe *Multi link list*. Struktur data ini digunakan untuk menyimpan data berbagai atribut suatu relasi sembarang dilengkapi dengan berbagai fungsi kebergantungan fungsional yang tercakup pada relasi R.

1.  $Ptr\ atribut = ^{atribut};$

$Atribut = record$

$Attr : string;$

$Next : Ptr\_atribut;$

$End;$

2.  $Ptr\ fungsi = ^{K\_fungsi}$

$K\_fungsi = record$

$Domain : Ptr\_atribut;$

$Next : ptr\_fungsi$

$End;$

3.  $Ptr\ relasi = ^{relasi};$

$Relasi = record$

$Relasi : ptr\ atribut$

$Fungsi : ptr\ fungsi$

$Next : : ptr\ relasi$

$End;$

### 4. RANCANG BANGUN ALGORITMA

Rancang bangun algoritma merupakan tahapan penentuan bentuk logis dari perangkat lunak yang akan dibangun. Algoritma perangkat lunak yang akan dibangun terdiri dari 4 (empat) buah algoritma utama yakni

1. Algoritma penentuan atribut kandidat kunci primer
2. Algoritma Normalisasi 1NF kedalam 2NF
3. Algoritma Normalisasi 2NF kedalam 3NF

#### 4. Algoritma Normalisasi 3NF kedalam BCNF

##### 1. Algoritma Penentuan Kandidat Kunci Primer

Algoritma penentuan kandidat key dilakukan dengan cara melakukan pemeriksaan semua kebergantungan fungsional yang ada pada skema relasi. Sebagai langkah awal dicari atribut-atribut yang merupakan super key dari relasi kemudian ditentukan atribut man saja yang akan menjadi kandidat key.

###### **Algoritma Penentuan kandidat key**

IS : Result kosong, Super Key kosong  $\alpha, \beta$

FS : Super key menyimpan atribut super key, kandidat key berisi super key yang menjadi kandidat key

###### **Algoritma**

$i \leftarrow 0$ ;

###### **Repeat**

*Key*  $\leftarrow \alpha$  {pembentuk  $\alpha \rightarrow \beta$  Pada  $F_i$ }

*Result*  $\leftarrow$  *Key*

**While** ( Nilai result berubah ) and ( result  $\neq$  Kumpulan atribut di R ) **do**

###### **Begin**

For setiap  $\beta \rightarrow \delta$  pada  $F$  do

if  $\beta \subseteq$  *Result* then *result*  $\leftarrow$  *result union*  $\delta$

###### **End**

$i \leftarrow i + 1$

**Until** (  $i >$  maks( $F$ ) ) or ( *result* = kumpulan atribut di R )

If *result* = kumpulan atribut R then *Super\_key*  $\leftarrow$  *key*

for  $j \leftarrow 1$  to maks( $F$ ) do

for  $i \leftarrow 1$  to maks ( $F$ )do

if Super key  $\not\subseteq$  super key then kandidat key  $\leftarrow$  kandidat key union super key

## Algoritma Normalisasi 1NF menjadi 2NF

### Algoritma Normalisasi 1NF menjadi 2NF

IS : T1 dan T2 kosong, key berisi kandidat key dari relasi R

FS : T1 berisi atribut relasi yang bergantung penuh pada key, T2 berisi atribut relasi yang tidak bergantung penuh pada key

#### ALGORITMA

$T1 \leftarrow \{ \} ; T2 \leftarrow \{ \}$

$Key \leftarrow Kandidat\_key(F)$

**For** Setiap  $\alpha$  merupakan bagian dari atribut R **do**

**If**  $\alpha$  merupakan bagian dari Key

**then**  $T1 \cup \alpha$

**else if**  $\alpha$  bergantung secara transitif pada key

**then**  $T2 \leftarrow$  Atribut kebergantungan Fungsional  $\alpha$

**else**  $T1 \leftarrow T1 \cup \alpha$

## Algoritma Normalisasi 2NF menjadi 3NF

### Algoritma Normalisasi 2NF menjadi 3NF

IS : T1 dan T2 kosong, key berisi kandidat key dari relasi R

FS : T1 berisi atribut relasi yang bergantung penuh pada key, T2 berisi atribut penuh yang bergantung transitif pada key

#### ALGORITMA

$T1 = \{ \}$

$T2 = \{ \}$

$Key \leftarrow Kandidat\_key(F)$

**For** setiap  $\alpha$  bagian dari atribut R **Do**

**If**  $\alpha$  merupakan bagian dari key

**Then**  $T1 = T1 \cup \alpha$

**Else IF**  $\alpha$  bergantung fungsional transitif pada key

**Then** T2

$=$  Atribut kebergantungan fungsional pada key

**Else**  $T1 = T1 \cup \alpha$

**Endif**

**Endif**

**End for**

## 2. Algoritma normalisasi 3NF menjadi BCNF

### **Algoritma normalisasi 3NF menjadi BCNF**

IS : Result berisi atribut atribut pada relasi R

FS : Result berisi atribut relasi hasil dekomposisi ke dalam BCNF

#### **Algoritma**

```
Result ← {R}
Done ← false
While not(done) Do
  IF skema R pada result belum BCNF
    Then begin
      IF (α →
        β bukan trivial) and (α bukan kandidat key dan α ∪ β ≠ ∅
        )
        Then Result ← (result - R1) ∪ (R1 - β) ∪ (α, β)
        Else done ← true
      End if
    Endif
  Endwhile
```

## 5. KESIMPULAN

Rancang bangun perangkat lunak normalisasi basis data TNBD dapat digunakan sebagai dasar untuk pembangunan perangkat lunak TNBD dan akan memberikan kemudahan dan percepatan waktu dalam tahapan analisis dan perancangan system sesuai metodologi waterfall.

## DAFTAR PUSTAKA

[BERNSTEIN 74] ,Bernstein P A, Synthesizing Third Normal Form Relations From Dependencies, 1976

[FAT 99 ] ,Fathansyah, Basis Data, 1999.