BAB III

BASIS DATA RELASIONAL

3.1. SISTIM BASIS DATA RELASIONAL

Sistim basis data merupakan suatu kumpulan file-file yang saling berkaitan dan suatu himpunan program-program yang memungkinkan para pemakai untuk menjalankan dan memodifikasi file-file tersebut.

Tujuan umum sistim basis data adalah untuk menyediakan gambaran secara abstrak tentang data tersebut. Hal ini merupakan sistim yang menyimpan rincian-rincian tertentu tentang bagaimana data disimpan dan diperoleh. Akan tetapi agar sistim dapat dipakai maka data harus dipanggil secara efesien .

Hal-hal yang berkaitan dengan efesiensi bergantung pada perancangan struktur file yang menyimpan data-data dalam basis data. Karena sistim basis data sering digunakan oleh orang yang belum menguasai komputer, maka untuk menghindari kesulitan ditentukan beberapa level/tingkatan abstraksi basis data yang diantaranya :

1. Tingkat fisik (physical level)

2. Tingkat konseptual (conseptual level)

3. Tingkat view (view level)

3.1.1. Model File

Untuk menggambarkan struktur basis data perlu didefenisikan model file. Model file adalah kumpulan konsep-konsep untuk menggambarkan data, hubungan-hubungan data dan sesuatu tentang data. Model file ini dibedakan dalam tiga kelompok yang masing-masing mempunyai tujuan tertentu. Diantara ketiga kelompok Model file itu adalah :

3.1.1.1. Model Object - Based Logikal

Model ini digunakan dalam penggambaran file pada tingkat konseptual dan tingkat view. Model ini ditandai oleh fakta yang memberikan kemampuan struktur yang baik dan fleksibel dan model ini menetapkan sesuatu mengenai file secara eksplisit.

Beberapa dari model ini adalah :

1. Model Relasi - Entitas

2. Model Biner

3. Model Data Semantik

4. Model Infologika

Dalam tulisan ini hanya dijelaskan model Relasi - Entitas sebagai gambaran dari model Object - Based Logikal karena mempunyai kemudahan-kemudahan sebagai model file yang sesuai untuk rancangan basis data.

Model file Relasi-Entitas didasarkan pada kenyataan yang terdapat dalam pengumpulan obyek-obyek dasar yang disebut entitas dan relasi antara obyek-obyeknya. Entitas adalah sebuah obyek yang ada dan dapat dibedakan dengan obyek yang lain, sedang relasi adalah hubungan antara tiap - tiap entitas.

3.1.1.2. Model Record-Based Logikal

Model ini digunakan dalam penggambaran data pada tingkat konseptual dan tingkat view. Perbedaannya dengan model Object-Based Logikal adalah model ini digunakan untuk menentukan struktur logika yang menyeluruh dari basis data, model ini tidak menyediakan fasilitas untuk menentukan file secara eksplisit. Yang termasuk model ini antara lain :

1. Model Relasional

   Dalam model Relasional, file dan relasi antar file digambarkan sebagai kumpulan tabel yang mempunyai sejumlah kolom dengan nama-nama tertentu dan antar kolom menyajikan suatu relasi.

2. Model Network

   File pada model Network digambarkan melalui kumpulan record-record dan relasi antar file digambarkan melalui penghubung (link) yang dapat dipandang sebagai penunjuk. Record-record dalam basis data diolah sebagai kumpulan grafik.

3. Model Hierarchical

   Model ini sama dengan model Network dalam pengertian file dan relasi antar file, digambarkan melalui record dan link. Perbedaannya dalam pengolahan record yang dibuat sebagai diagram (grafik pohon/tree).

3.1.1.3. Model Physic

Model ini digunakan untuk menggambarkan data pada tingkat fisik. Perbedaan dengan model-model logika adalah file disimpan secara rinci dan tampak sangat rumit. Yang termasuk model ini adalah :

1. Model Unifying (menyatu)

2. Model kerangka memori (frame memori).

Untuk membatasi pembahasan maka model ini tidak akan dibahas.

Selanjutnya tulisan ini akan menjelaskan file-file yang termasuk dalam model Relasional untuk menggambarkan abstraksi basis data pada tingkat konseptual. Untuk menggambarkan basis data secara keseluruhan maka dibuat struktur file yang sederhana yang hany memuat beberapa atribut dan antara struktur file yang satu dengan yang lain masih bisa dihubungkan kembali sehingga mencerminkan struktur basis data secara menyeluruh.

3.2. STRUKTUR BASIS DATA RELASIONAL

Suatu Basis data relasional terdiri dari kumpulan tabel-tabel yang masing-masing diberi nama khusus dimana diantara tabel-tabel tersebut terdapat hubungan satu sama lainnya sehingga membentuk Struktur Basis data relasional. Suatu baris dalam tabel menyajikan keterkaitan antara himpunan-himpunan nilai, karena itu Struktur ini diturunkan dari konsep relasi pada himpunan.

Definisi : 3.1

Diberikan suatu himpunan D_1, D_2, ..., D_n R adalah suatu relasi pada n Himpunan dengan tupel t = (d_1, d_2 ,..., d_n) sedemikian sehingga $d_1 \in D_1$, $d_2 \in D_2$,....... $d_n \in D_n$. Himpunan D_1, D_2, ..., D_n adalah domain dari R dengan n adalah derajat dari relasi R.

Pada gambar 3.1 menjelaskan relasi R yang berderajat lima, kelima domain menyajikan himpunan nilai-nilai yaitu D_1,D_2,D_3,D_4,D_5 sehingga dalam bentuk tabel relasi R dapat dibuat sebagai berikut :

A_1	A_2	A_3	A_4	A_5
d_{11}	d_{12}	d_{13}	d_{14}	d_{15}
d_{21}	d_{22}	d_{23}	d_{24}	d_{25}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
d_{m1}	d_{m2}	d_{m3}	d_{m4}	d_{m5}

Gambar 3.1 Relasi R

Jadi R adalah relasi dari himpunan D_1, D_2, D_3, D_4, D_5 dan merupakan himpunan bagian dari hasil ganda kartesius $D_1 X D_2 X D_3 X D_4 X D_5$.

Secara umum tabel yang terdiri dari n kolom adalah himpunan bagian dari :

$$\overset{n}{\underset{i=1}{X}} D_i$$

Yaitu hasil ganda kartesius dari n domain.

Dalam membentuk relasi R dari himpunan domain $D_1, D_2, \ldots\ldots, D_n$ dipergunakan beberapa atribut dan nama-nama atribut ini didapatkan dari domain yang memungkinkan terbentuknya relasi R.

Pandang U adalah sebuah skema relasi, suatu himpunan dari skema relasi $\{R_1, R_2, \ldots, R_n\}$ adalah pemisahan dari U jika :

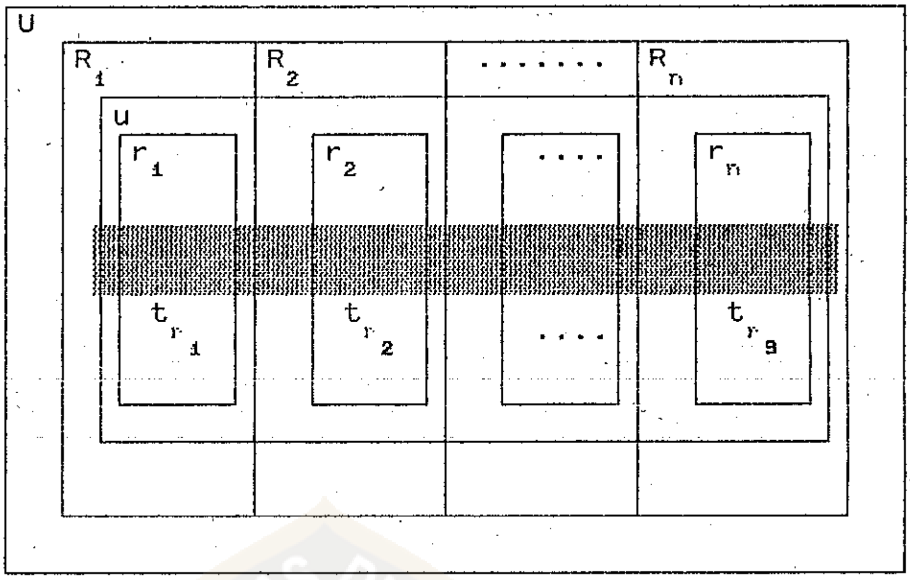
$$\bigcup_{i=1}^{n} R_i = U$$

Yaitu setiap atribut dalam U muncul paling sedikit satu dalam R_i, $1 \le i \le n$.

u adalah relasi pada skema U dan $r_i = \Pi_{R_i}(u)$, $1 \le i \le n$, yaitu $\{ r_1, r_2, \ldots, r_n\}$ adalah sistim basis data yang dihasilkan dari pemisahan U dalam $\{ R_1, R_2, \ldots, R_n\}$ sehingga menghasilkan :

$$u \subseteq \bigotimes_{i=1}^{n} r_i$$

Dengan pengertian, anggaplah tupel t pada relasi u, bila ditentukan $r_1, r_2, \ldots, r_n$, tupel t menimbulkan satu tupel t_i pada setiap r_i, $1 \le i \le n$. Dan tupel-tupel ini bergabung membentuk t, ketika ditentukan $r_1 \otimes r_2 \otimes \ldots \otimes r_n$.



Gambar 3.2 Relasi U dengan pemisahan
$\{R_1, R_2, \ldots\ldots\ldots, R_n\}$

Misal F^+ mewakili himpunan klosure pada basis data, pemisahan $\{R_1, R_2, \ldots\ldots\ldots, R_n\}$ dari skema relasi U adalah pemisahan yang tidak hilang untuk U jika untuk setiap relasi u pada skema U yang berlaku menurut F^+ maka :

$$u = \bigotimes_{i=1}^{n} r_i$$

$$u = \bigotimes_{i=1}^{n} \Pi_{R_i}(u)$$

Jadi suatu relasi R memperlihatkan hubungan dari beberapa atribut dari domain-domain yang sama atau berlainan. setiap relasi dari basis data relasional disyaratkan memenuhi kondisi bahwa setiap nilai berelasi dalam satu tupel. Kondisi yang demikian disebut Normalisasi. Hal ini diperlihatkan secara langsung dari definisi relasi dimana suatu relasi minimal menunjukkan hubungan dua elemen x dan y dengan $x \in V$ dan $y \in W$. Jadi tidak mungkin relasi dibentuk tanpa elemen.

Sejalan dengan itu untuk membentuk rancangan sistim basis data relasional kedalam suatu bentuk normal tertentu perlu diperhatikan ketergantungan antar atribut-atribut pembentuk relasi. Jadi suatu normalisasi senantiasa beracuan pada sifat ketergantungan secara fungsional dari data dalam sistim basis data relasional.

Sesuai dengan lingkup pembahasan yaitu membawa struktur basis data relasional kebentuk normal ketiga (3NF) dan bentuk normal Boyce-Codd (BCNF) maka teori ketergantungan fungsional-lah yang akan ketengahkan sebagai penyusun normalisasi.

3.3. KETERGANTUNGAN FUNGSIONAL (FUNCTIONAL DEPENDENCIES)

Definisi : 3.2

$R\ [A_1\ A_2\ \ldots.\ A_n]$ adalah relasi dan V,W adalah himpunan bagian dari $\{A_1, A_2, \ldots, A_n\}$ $V \longrightarrow W$ adalah ketergantungan fungsional W pada V jika dan hanya jika pada relasi r (R) yang manapun untuk semua pasangan tupel-tupel t_1 dan t_2 pada relasi r(R) sedemikian sehingga $t_1(V) = t_2(V)$ maka juga berlaku $t_1(W) = t_2(W)$.

Contoh : 3.1

Diketahui suatu relasi r (R) sebagai berikut :

A	B	C	D
a_1	b_1	c_1	d_1
a_1	b_2	c_1	d_2
a_2	b_2	c_2	d_2
a_2	b_3	c_2	d_3
a_3	b_3	c_2	d_3

Gambar 3.3 Relasi r(R)

Skema relasi diatas menunjukan ketergantungan antara atribut C pada A dan dapat ditulis sebagai $A \longrightarrow C$. Jika diperhatikan, terdapat dua tupel dari A yang mempunyai nilai a_1 dan C mempunyai nilai c_1, dengan cara sama juga terdapat dua tupel dari A dengan nilai a_2 dan dari C dengan nilai c_2. Tetapi skema relasi diatas tidak menunjukkan ketergantungan fungsional $C \longrightarrow A$ sebab pada tupel

t_1 = (a_2,b_3,c_2,d_3) dan t_2 = (a_3,b_3,c_2,d_3)

mempunyai nilai c_2 yang sama tetapi pada atribut A terdapat a_2 dan a_3 yang berbeda.

3.3.1. Implikasi Logis Dari Ketergantungan

R adalah relasi dan A, B, C adalah atribut dari relasi R. Bila diketahui ketergantungan fungsional $A \longrightarrow B$ dan $B \longrightarrow C$ maka terdapat ketergantungan fungsional $A \rightarrow C$

Hal diatas dapat dijelaskan sebagai berikut :

Misal r(R) adalah relasi yang memenuhi $A \longrightarrow B$ dan $B \longrightarrow C$. Andaikan dua tupel t_1 dan t_2 dalam relasi r(R) sehingga t_1 dan t_2 memenuhi untuk komponen A tetapi tidak memenuhi komponen C. Karena t_1 dan t_2 memenuhi komponen A maka t_1 dan t_2 juga memenuhi komponen B. Sebab $A \longrightarrow B$. Kemudian dari ketergantungan fungsional $B \longrightarrow C$, bila t_1 dan t_2 memenuhi komponen B juga memenuhi pada komponen C jadi kontradiksi sehingga t_1 dan t_2 adalah dua tupel dari relasi r(R) yang memenuhi dalam komponen A dan komponen C sehingga $A \longrightarrow C$ merupakan ketergantungan fungsional C pada A.

Secara umum jika F adalah suatu himpunan ketergantungan fungsional untuk skema relasi R dan $V \longrightarrow W$ adalah suatu ketergantungan fungsional. F menunjukkan implikasi logis $V \longrightarrow W$ dan ditulis $F \Longrightarrow V \longrightarrow W$ jika setiap relasi r(R) memenuhi ketergantungan dalam F maka juga memenuhi ketergantungan fungsional $V \longrightarrow W$.

Contoh : 3.2

Diketahui F terdiri dari $A \rightarrow B$ dan $B \rightarrow C$ maka implikasi logis dari F adalah $A \rightarrow C$ dan ditulis $\{ A \rightarrow B , B \rightarrow C \} \Longrightarrow A \rightarrow C$

Suatu closure F^+ yaitu himpunan ketergantungan fungsional yang dibentuk oleh implikasi logis dari F jadi $F^+ = \{ V \longrightarrow W \mid F \Longrightarrow V \longrightarrow W \}$

Contoh : 3.3

R [A,B,C] dan $F = \{ A \rightarrow B , B \rightarrow C \}$

maka $F^+ = \{A \rightarrow C \mid F \Longrightarrow A \rightarrow C \}$ atau F^+ terdiri dari semua ketergantungan $V \rightarrow W$ sedemikian sehingga :

1. V terdiri dari A, misalnya ABC $\longrightarrow$ AB, AB $\longrightarrow$ BC atau A $\longrightarrow$ C.
2. V terdiri dari B tetapi tidak memuat A, misalnya BC $\longrightarrow$ B, B $\longrightarrow$ C atau B $\longrightarrow \emptyset$
3. V $\longrightarrow$ W adalah satu dari dua ketergantungan C $\longrightarrow$ C atau C $\longrightarrow \emptyset$ (V tidak terdiri dari A maupun B).

3.3.2. Aksioma-aksioma Ketergantungan Fungsional

hukum-hukum berikut ini disebut aksioma Armstrong. Dengan asumsi bahwa diberikan suatu skema relasi dengan U adalah himpunan semesta dari atribut-atribut dan suatu himpunan ketergantungan fungsional F dari atribut- atribut yang berada dalam U. Hukum-hukum Inferensinya adalah :

Aksioma : 3.1

Refleksif. Jika $W \subseteq V \subseteq U$ maka V $\longrightarrow$ W adalah diimplikasikan secara logis oleh F dengan hukum ini memberikan ketergantungan trivial dan penggunaan hukum ini hanya tergantung pada U bukan pada F.

Aksioma : 3.2

Argumentasi . Jika $V \longrightarrow W$ adalah ketergantungan fungsional dari relasi R dengan atribut-atribut dalam U dan $Z \subseteq U$ maka $VZ \longrightarrow WZ$. VZ adalah bentuk singkat dari $V \cup Z$.

Aksioma : 3.3

Transitip. Jika $V \longrightarrow W$ dan $W \longrightarrow Z$ adalah ketergantungan fungsional dari relasi R dengan atribut dalam U, maka $V \longrightarrow Z$ juga merupakan ketergantungan fungsional dari relasi R.

Contoh : 3.4

Diketahui relasi R [A B C] dan ketergantungan fungsional F = { $AB \longrightarrow C$, $C \longrightarrow A$ } akan ditunjukkan $BC \longrightarrow ABC$.

1. diberikan $C \longrightarrow A$.
2. Dari Aksioma (2) argumentasi. dengan B maka $BC \longrightarrow AB$.
3. diberikan $AB \longrightarrow C$.

4. Dari aksioma (2) argumentasi. dengan AB maka AB $\longrightarrow$ ABC.

5. Dari aksioma (3) transitip. antara 2 dan 4 maka BC $\longrightarrow$ ABC.

Lemma : 3.1

Jika ketergantungan fungsional V $\longrightarrow$ W didapat dari himpunan ketergantungan fungsional F dengan menggunakan aksioma Armstrong, maka V $\longrightarrow$ W merupakan ketergantungan fungsional yang benar dalam relasi R dengan himpunan ketergantungan F.

Bukti :

Aksioma 3.1 Refleksif. aksioma ini menjelaskan bahwa tidak ada relasi r(R) dengan dua tupel yang memenuhi V tetapi tidak memenuhi beberapa himpunan bagian dari V.

Aksioma 3.2 Argumentasi. misal relasi r(R) memenuhi V $\longrightarrow$ W berarti tupel-tupel yang memenuhi V juga memenuhi W. Andaikan t_1 dan t_2 adalah dua tupel yang memenuhi VZ tetapi tidak memenuhi WZ, berarti tupel t_1 dan t_2 tidak memenuhi dari Z akibatnya t_1 dan t_2 tidak memenuhi W kontradiksi dengan ketentuan diatas bahwa $V \longrightarrow W$ memenuhi relasi r(R). Jadi yang benar t_1 dan t_2 memenuhi VZ dan WZ atau $VZ \longrightarrow WZ$.

Aksioma 3.3 Transitip. dengan implikasi logis bahwa jika $A \longrightarrow B$ dan $B \longrightarrow C$ maka $A \longrightarrow C$ memenuhi relasi r(R) dengan himpunan ketergantungan fungsional F.

□

Lemma : 3.2

a. Hukum Union (gabungan)

$\{ V \longrightarrow W, V \longrightarrow Z \} \Longrightarrow V \longrightarrow WZ$

b. Hukum Pseudotransitif

$\{ V \longrightarrow W, YW \longrightarrow Z \} \Longrightarrow YV \longrightarrow Z$

c. Hukum Dekomposisi (pemisahan)

Jika $V \longrightarrow W$ dan $Z \subseteq W$ maka $V \longrightarrow Z$

Bukti :

a. Diberikan $V \longrightarrow W$ dengan menambah V sehingga menjadi $V \longrightarrow VW$ dan diberikan juga $V \longrightarrow Z$ dengan menambah W maka menjadi $V W \longrightarrow Z W$ dengan hukum transitip didapat $V \longrightarrow VW$, $VW \longrightarrow ZW$ maka $V \longrightarrow ZW$.

b. Diberikan $V \longrightarrow W$, dengan menambah Y, didapat $YV \longrightarrow YW$ dan diberikan juga $YW \longrightarrow Z$, maka dengan hukum transitip didapat $YV \longrightarrow Z$.

c. Diberikan $Z \subseteq W$, dengan hukum Refleksif $W \longrightarrow Z$ dan diberikan $V \longrightarrow W$, maka menurut hukum transitip $V \longrightarrow Z$

Akibat dari hukum union dan dekomposisi didapatkan jika $A_1, A_2, \ldots A_n$ adalah atribut-atribut, $V \longrightarrow A_1, A_2, \ldots A_n$ jika dan hanya jika $V \longrightarrow A_i$, $1 \leq i \leq n$.

Definisi : 3.3 Closure

F adalah himpunan ketergantungan fungsional pada atribut-atribut semesta U dan V himpunan bagian U. V^+ adalah closure dari V yaitu himpunan atribut-atribut A sedemikian sehingga $V \longrightarrow A$ yang dapat disimpulkan dari F dengan aksioma-aksioma Armstrong.

Lemma : 3.3

Ketergantungan fungsional $V \longrightarrow W$ memenuhi aksioma-aksioma Armstrong jika dan hanya jika $W \subseteq V^+$.

Bukti :

Misal $W = \{ A_1, A_2, \ldots, A_n \}$ dan $W \subseteq V^+$

Dari definisi closure V^+, $V \rightarrow A_i$, $1 \leq i \leq n$

Dari hukum union pada Lemma 3.2, maka $V \longrightarrow A_1 A_2 \ldots A_n$ atau $V \longrightarrow W$. Kemudian dari hukum dekomposisi untuk setiap i $V \longrightarrow A_i$, sehingga $A_i \subseteq V^+$ maka $V \longrightarrow W$ dan $W \subseteq V^+$.

Teorema : 3.1

Setiap ketergantungan fungsional $V \longrightarrow W$ didalam F dipenuhi oleh relasi r(R) dengan aksioma Armstrong, sebaliknya setiap ketergantungan fungsional $V \longrightarrow W$ diluar F tidak dipenuhi oleh relasi r(R) atau F tidak berimplikasi secara logis pada

$V \longrightarrow W$.

Bukti :

F adalah himpunan ketergantungan dibawah atribut-atribut himpunan U dan misalnya $V \longrightarrow W$ tidak dapat diturunkan dari aksioma-aksioma. Mengingat relasi r(R) dengan dua tupel pada gambar 3.2

Atribut pada V^+				Atribut yang lain			
1	1	....	1	1	1	.....	1
1	1	....	1	0	0	.....	0

Gambar 3.2
Relasi r(R) Tidak Menunjukkan $F \Longrightarrow V \longrightarrow W$

Pertama ditunjukkan bahwa semua ketergantungan dalam F adalah dipenuhi oleh r(R). Misal $T \longrightarrow Y$ dalam F tetapi tidak dipenuhi oleh r(R) maka $T \subseteq V^+$ atau dua tupel yang lain pada r(R) tidak memenuhi atribut-atribut pada T. Dan hal ini tidak bertentangan dengan $T \longrightarrow Y$ sebab Y juga bukan himpunan bagian dari V^+.

Untuk A adalah atribut pada Y yang tidak dalam V^+ menurut lemma 3.3 $T \subseteq V^+$ maka $V \longrightarrow T$. Dari ketergantungan $T \longrightarrow Y$ dalam F oleh hukum Transitip didapat $V \longrightarrow Y$ Kemudian dengan hukum Refleksif didapat $Y \longrightarrow A$ dan dengan hukum transitip didapat $V \longrightarrow A$, tetapi menurut definisi closure A didalam V^+ padahal diasumsikan A tidak didalam V^+ sehingga kontradiksi jadi yang benar setiap $T \longrightarrow Y$ dalam F adalah dipenuhi oleh r(R).

Sekarang akan ditunjukkan bahwa $V \longrightarrow W$ tidak dipenuhi oleh relasi r(R). Andaikan $V \longrightarrow W$ dipenuhi oleh relasi r(R), berarti $W \subseteq V^+$. Dua tupel yang berlainan pada r(R) dipenuhi V tetapi tidak dipenuhi W. Tetapi menurut lemma 3.3, bahwa $V \longrightarrow W$ dapat diturunkan dari aksioma-aksioma Armstrong, sehinggà kontradiksi. jadi yang benar $V \longrightarrow W$ tidak dipenuhi oleh r(R) didalam ketergantungan F. Setiap $V \longrightarrow W$ yang tidak termasuk dalam F oleh aksioma Armstrong, maka F tidak berimplikasi logis pada $V \longrightarrow W$ sehingga aksioma tersebut adalah lengkap.

□

Suatu pendefinisian yang equivalen dari closure V^+ adalah himpunan atribut A sedemikian sehingga $F \Longrightarrow V \longrightarrow A$ sehingga F^+ adalah implikasi logis dari F.

3.4. MENGHITUNG CLOSURE

Misal diketahui himpunan ketergantungan fungsional $F = \{ A \rightarrow B_1, A \rightarrow B_2, \ldots A \rightarrow B_n \}$ maka closure F^+ dari F memuat semua ketergantungan $A \longrightarrow W$ dengan W adalah himpunan bagian dari $\{B_1, B_2, \ldots B_n\}$ yang dapat dinyatakan secara logis oleh F. Cara semacam ini merupakan penyederhanaan dari pada menggunakan definisi formal dari ketergantungan fungsional yang mana bila F besar prosesnya akan panjang dan sulit.

Untuk menghitung closure V^+ dari himpunan atribut pada V dapat menggunakan algoritma sebagai berikut :

Algoritma : 3.1

1. Diketahui (input) :
   - Himpunan berhingga dari atribut-atribut U
   - Suatu himpunan ketergantungan fungsional F pada U.
   - Dan himpunan $V \subseteq U$.
2. Ditanyakan (output) :
   - Closure V^+ dari V dengan memperhatikan F.
3. Metode :
   - Dalam menghitung barisan himpunan dari atribut-atribut $V^{(0)}, V^{(1)}$, ... digunakan dengan hukum-hukum :

     3.1. $V^{(0)} = V$

     3.2. $V^{(i+1)} = V^{(i)}$ ditambah himpunan atribut-atribut A sedemikian sehingga ketergantungan $W \longrightarrow Z$ dalam F, A dalam Z dan $W \subseteq V^{(i)}$.

$V = V^{(0)} \subseteq V^{(1)} \subseteq \ldots \subseteq V^{(i)} \subseteq \ldots \subseteq U$

dan U berhingga untuk i sangat besar sehingga $V^{(i)} = V^{(i+1)} = V^{(i+2)} = \ldots\ldots\ldots$ dan untuk i yang ini maka $V^{+} = V^{(i)}$.

Contoh : 3.5

Misal $F = \{ AB \longrightarrow C,\ C \longrightarrow A,\ BC \longrightarrow D,$
$ACD \longrightarrow B,\ D \longrightarrow EG,\ BE \longrightarrow C,$
$CG \longrightarrow BD,\ CE \longrightarrow AG \}$

dan $V = \{B,D\}$

Dengan algoritma 3.1

$V^{(0)} = BD,$

untuk menghitung $V^{(1)}$,diperhatikan ketergantungan yang bagian kirinya B, D atau BD jadi didapat $D \longrightarrow EG$, kemudian BD digabung dengan E dan G sehingga didapat $V^{(1)} =$ BDEG. Untuk menghitung $V^{(2)}$ diperhatikan ketergantungan yang bagian kirinya B, D, E, G atau salah satu dari gabungannya, sehingga didapat $D \longrightarrow EG$ dan

$BE \longrightarrow C$ maka $V^{(2)} =$ BCDEG . Untuk menghitung $V^{(3)}$ dengan cara sama didapat $V^{(3)}$ = ABCDEG yaitu himpunan semua atribut dari hal diatas maka $V^{(3)} = V^{(4)} = \ldots\ldots.$

Jadi $(BD)^{+}$ = ABCDEG.

Teorema : 3.2

Algoritma 3.1 merupakan penghitung closure V^{+} yang baik/tepat .

Bukti :

Pertama ditunjukkan dengan induksi matematik pada J jika A berada dalam $V^{(j)}$ maka A dalam V^{+} untuk J = 0 maka A dalam V dan juga $V \longrightarrow A$ untuk $J > 0$ dan asumsi $V^{(j-1)}$ hanya terdiri atribut dalam V^{+} .

Misal A berada dalam $V^{(j)}$ karena A dalam Z dengan $W \longrightarrow Z$ dalam F dan $W \subseteq V^{(j-1)}$.

Kemudian dengan lemma 3.3, untuk $V \longrightarrow W$ oleh hukum Transitip $V \longrightarrow W$ dan $W \longrightarrow Z$ maka $V \longrightarrow Z$. Oleh hukum Refleksif $Z \longrightarrow A$ lalu Transitip maka didapat $V \longrightarrow A$. Jadi A berada dalam V^{+} . □

3.5. COVER DARI HIMPUNAN KETERGANTUNGAN

Suatu himpunan ketergantungan F dan G dikatakan equivalen jika $F^+ = G^+$. Jika F dan G adalah equivalen maka dikatakan F cover G dan G cover F. Untuk menguji apakah F dan G equivalen sebagai berikut :

Untuk setiap ketergantungan $W \longrightarrow Z$ dalam F diuji apakah $W \longrightarrow Z$ dalam G^+ dengan algoritma 3.1, untuk menghitung W^+ dan kemudian diuji apakah $Z \subseteq W^+$. Jika ketergantungan $W \longrightarrow Z$ dalam F tetapi tidak dalam G^+ maka $F^+ \neq G^+$.

Setiap ketergantungan dalam F dan G maka setiap ketergantungan $T \longrightarrow Y$ dalam F^+ juga berada dalam G^+. Sebagai bukti bahwa $T \longrightarrow Y$ dalam G^+ bisa dikerjakan seperti diatas yaitu $W \longrightarrow Z$ dalam G^+ Padahal $T \longrightarrow Y$ berada dalam F^+. Jadi jelas terbukti bahwa jika $T \longrightarrow Y$ dalam F^+ maka juga dalam G^+ . Dengan cara sama jika ketergantungan berada dalam G pasti juga dalam F.

Lemma : 3.4

Setiap himpunan ketergantungan fungsional

F adalah ditutup (covered) oleh himpunan ketergantungan G. Yang didalamnya tidak ada bagian kanan yang memiliki lebih dari satu atribut.

Bukti :

G adalah himpunan ketergantungan $V \longrightarrow A$ sedemikian sehingga untuk $V \longrightarrow W$ didalam F, A berada dalam W. Maka $V \longrightarrow A$ dihasilkan dari $V \longrightarrow W$ oleh hukum dekomposisi (pemisahan) sehingga $G \subseteq F^{+}$. Pada hal lain jika $W = A_1A_2...A_n$ maka $V \rightarrow W$ didapat dari bentuk $V \rightarrow A_1, V \rightarrow A_2, ... V \longrightarrow A_n$.

□

Suatu himpunan ketergantungan F' disebut minimal jika memenuhi sifat sebagai berikut :

1. Setiap bagian kanan dari ketergantungan dalam F adalah atribut tunggal.
2. Komplemen dari $V \longrightarrow A$ dalam F yaitu himpunan $F - \{V \rightarrow A\}$ equivalen dengan F.
3. Komplemen dari $V \longrightarrow A$ dalam F dan $Z \subset V$

adalah himpunan $F - \{V \rightarrow A\} \cup \{Z \rightarrow A\}$ equivalen dengan F.

Sifat-sifat diatas memberikan suatu kepastian bahwa :

1. Tidak ada ketergantungan dalam F yang redundansi.
2. Tidak ada atribut pada bagian kiri ketergantungan yang redundansi.
3. Tidak ada atribut pada bagian kanan ketergantungan juga redundansi sebab hanya satu atribut.

Teorema : 3.3

Setiap himpunan ketergantungan F equivalen dengan himpunan ketergantungan minimal F'.

Bukti :

Oleh lemma 3.4 diasumsikan bahwa tidak ada bagian kanan ketergantungan dalam F yang memiliki lebih satu atribut . Untuk memenuhi sifat (2), menyatakan setiap ketergantungan $V \longrightarrow W$ dalam F dan jika $F - \{ V \longrightarrow W \}$ equivalen dengan F maka ditempuh dengan menghilangkan $V \longrightarrow W$ dari F dengan catatan bahwa ketergantungan pada tingkat yang berbeda dapat dihasilkan dari eliminasi pada himpunan ketergantungan yang berbeda. Sebagai contoh :

$F=\{A \rightarrow B,\ B \rightarrow A,\ B \rightarrow C,\ A \rightarrow C,\ C \rightarrow A\}$ dengan mengeliminasi $B \rightarrow A$ dan $A \rightarrow C$ atau $B \rightarrow C$ selanjutnya pada sifat (3) menyatakan setiap atribut sisanya dalam F dan setiap pada bagian kiri dieliminasi serta memiliki himpunan atribut yang equivalen. Misalnya sebagai contoh :

$F = \{ AB \rightarrow C,\ A \rightarrow B,\ B \rightarrow A \}$ dengan mengeliminasi A atau B dari $AB \rightarrow C$ sehingga $F' = \{B \rightarrow C,\ A \rightarrow B,\ B \rightarrow A\}$.

□

Contoh : 3.6

Diketahui himpunan ketergantungan F yang terdiri dari :

$AB \longrightarrow C$ $BE \longrightarrow C$

$C \longrightarrow A$ $CG \longrightarrow B$

$BC \longrightarrow D$ $CG \longrightarrow D$

$ACD \longrightarrow B$ $CE \longrightarrow A$

$D \longrightarrow E$ $CE \longrightarrow G$

$D \longrightarrow G$

Dari hal diatas :

$CE \longrightarrow A$ adalah redundansi sebab diimplikasikan oleh $C \longrightarrow A$ dan $CG \longrightarrow B$ adalah redundansi sebab $CG \longrightarrow D$, $C \longrightarrow A$ dan $ACD \longrightarrow B$ sehingga implikasinya $CG \longrightarrow B$ yang dapat diuji dengan menghitung $(CG)^+$. Sedang $ACD \longrightarrow B$ dapat digantikan oleh $CD \longrightarrow B$ sebab $C \longrightarrow A$ maka satu cover minimal dari F adalah F' =

$AB \longrightarrow C$ $D \longrightarrow G$

$C \longrightarrow A$ $BE \longrightarrow C$

$BC \longrightarrow D$ $CG \longrightarrow D$

$CD \longrightarrow B$ $CE \longrightarrow G$

$D \longrightarrow E$

dan cover minimal lain dengan mengeliminasi $CE \longrightarrow A$, $CG \longrightarrow D$ dan $ACD \longrightarrow B$ adalah F' =

$AB \longrightarrow C$ $\qquad D \longrightarrow G$

$C \longrightarrow A$ $\qquad BE \longrightarrow C$

$BC \longrightarrow D$ $\qquad CG \longrightarrow B$

$D \longrightarrow E$ $\qquad CE \longrightarrow G$

catatan bahwa dua cover minimal memiliki jumlah ketergantungan yang berbeda.

3.6. PEMISAHAN RELASI (DECOMPOSITION OF RELATION)

Pemisahan relasi U [A_1 A_2 .... A_n] adalah kumpulan α = { R_1, R_2, ... R_k } yaitu himpunan-himpunan bagian dari U sedemikian sehingga : $U = R_1 \cup R_2 \cup \ldots \cup R_k$ yang dalam hal ini tidak disyaratkan bahwa antara R_i saling berpotongan.

3.7. PENGGABUNGAN YANG TIDAK HILANG

Jika relasi U dipisahkan menjadi α dan F adalah himpunan ketergantungan. Maka suatu pemisahan penggabungan yang tidak hilang jika untuk relasi u(U) memenuhi F :

$$u = \Pi_{R_1}(u) \; (X) \; \Pi_{R_2}(u) \; (X) \; \ldots\ldots \; \Pi_{R_k}(u)$$

u adalah natural join dari proyeksinya pada R_i .

Jika α = { R_1, R_2, ... R_k } maka m_α adalah pemetaan yang didefinisikan :

$$m_\alpha (u) = \mathop{(X)}_{i=1}^{k} \Pi_{R_i}(u)$$

yaitu join dari proyeksi u pada relasi dalam α sehingga gabungan yang tidak hilang dapat diekspresikan dengan notasi himpunan tupel. Jika t adalah tupel pada u, t(V) adalah tupel pada atribut V, yaitu komponen-komponen V.

Contoh : 3.7

$\Pi_x (u)$

Dapat diekspresikan dengan :

{ t(x) | t dalam u }

Lemma : 3.5

Jika U adalah relasi, α = { R_1,R_2, ... R_k} suatu pemisahan dari U, u suatu relasi untuk U dan $r_i = \Pi_{R_i}(u)$ maka :

a. $u \subseteq m_\alpha (u)$

b. Jika $s = m_\alpha (u)$, maka $\Pi_{R_i}(s) = r_i$

c. $m_\alpha(m_\alpha(u)) = m_\alpha(u)$

Bukti :

a. Jika t dalam u, maka untuk setiap i, $t_i = t(R_i)$ dalam r_i. Dengan definisi natural join t dalam $m_\alpha(u)$, sebab t dipenuhi t_i dalam atribut-atribut R_i untuk setiap i.

b. Untuk $u \subseteq s$, dari (a) $\Pi_{R_i}(u) \subseteq \Pi_{R_i}(s)$ sehingga $r_i \subseteq \Pi_{R_i}(s)$ kemudian untuk menunjukkan $\Pi_{R_i}(s) \subseteq r_i$, dimisalkan untuk suatu i sehingga t_i dalam $\Pi_{R_i}(s)$ maka terdapat tupel t dalam s sedemikian sehingga $t(R_i) = t_i$ dengan t dalam s, terdapat tupel l_i dalam r_i sehingga untuk setiap i, $t(R_i) = l_i$.

Jadi $t(R_i) = l_i$ berada dalam r_i dan akibatnya $\Pi_{R_i}(s) \subseteq r_i$.

Jadi didapat $\Pi_{R_i}(s) = r_i$.

c. Jika $s = m_\alpha(u)$ maka dengan (b) ,

$\Pi_R(s) = r_i$ sehingga :

$$m_{\alpha}(s) = \bigotimes_{i=1}^{k} r_i = m_{\alpha}(u).$$

□

Jika untuk setiap r_i yaitu relasi untuk R_i dan $s = \bigotimes_{i=1}^{k} r_i$ maka $\Pi_{R_i}(s)$ tidak perlu sama dengan r_i.

Contoh : 3.8

Jika : R_1 [AB] , R_2 [BC]

r_1 [a b] dan r_2 [b_1c_1 b_2c_2]

maka : s [a b c], dan

$\Pi_{BC}(s)$ [b_1c_1] $\neq r_2$ meskipun secara umum

$\Pi_{R_i}(s) \subseteq r_i$.

R_1	A	B
r_1	a_1	b_1

R_2	B	C
r_2	b_1	c_1
	b_2	c_2

$R_1 (X) R_2$	A	B	C
s	a_1	b_1	c_1

Uji Penggabungan yang tidak hilang

Algoritma : 3.2

- Diketahui (input) :

  Suatu relasi R [$A_1A_2....A_n$].

  Himpunan ketergantungan fungsional F, dan pemisahan α = {R_1,R_2, ...R_k}

- Ditanya (output) :

  pemisahan α = {R_1,R_2, ...R_k} adalah pemisahan gabungan yang tidak hilang.

- Motode penyelesaian :

  Dibentuk tabel dengan n kolom dan k baris, kolom ke-j berkorespodensi dengan atribut A_j dan baris ke-i korespodensi dengan relasi R_i. Pada baris ke- i dan kolom ke-j diberi simbol a_j jika A_j didalam R_i, namun bila tidak ditulis b_{ij}. Dengan memperhatikan setiap ketergantungan V ⟶ W dalam F, untuk setiap kolom dari atribut V, jika dapat ditemukan dua baris pada atribut W, sehingga nilai yang satu a_j maka yang lain juga disamakan menjadi a_j, tetapi jika keduanya berbentuk b_{ij} dan b_{ej} maka keduanya dapat dibuat b_{ij} atau b_{ej}. Jika pada akhir modifikasi baris dari tabel diperoleh $a_1, a_2, \ldots a_k$ pada satu baris maka penggabungan ini adalah penggabungan yang tidak hilang.

Contoh : 3.9

R [ABCDE]

R_1 [AD], R_2 [AB], R_3 [BE], R_4 [CDE]

R_5 [AE]

Ketergantungan fungsional :

$A \rightarrow C$, $B \rightarrow C$, $C \rightarrow D$, $DE \rightarrow C$, $CE \rightarrow A$

Dari relasi dan pemisahan yang diketahui dapat dibuat tabel sebagai berikut :

nilai a_1 terdapat pada kolom 1 baris 1,2,5

nilai a_2 terdapat pada kolom 2 baris 2,3

nilai a_3 terdapat pada kolom 3 baris 4

nilai a_4 terdapat pada kolom 4 baris 1,4

nilai a_5 terdapat pada kolom 5 baris 3,4,5

sedang nilai-nilai pada baris i dan kolom j yang lain adalah b_{ij} seperti terlihat pada tabel :

A	B	C	D	E
a_1	b_{21}	b_{13}	a_4	b_{15}
a_1	a_2	b_{23}	b_{24}	b_{25}
b_{31}	a_2	b_{33}	b_{34}	a_5
b_{41}	b_{41}	a_3	a_4	a_5
a_1	b_{51}	b_{53}	b_{54}	a_5

Dari ketergantungan $A \longrightarrow C$, maka b_{13}, b_{23} dan b_{53} dapat diganti dengan b_{13}.

dari ketergantungan $B \longrightarrow C$ maka b_{33} dapat pula diganti b_{13} sehingga tabelnya menjadi

A	B	C	D	E
a_1	b_{21}	b_{13}	a_4	b_{15}
a_1	a_2	b_{13}	b_{24}	b_{25}
b_{31}	a_2	b_{13}	b_{34}	a_5
b_{41}	b_{41}	a_3	a_4	a_5
a_1	b_{51}	b_{13}	b_{54}	a_5

Dari ketergantungan C $\rightarrow$ D, maka b_{24}, b_{34} dan b_{54} dapat diganti dengan a_4.

Dari ketergantungan DE $\rightarrow$ C, maka b_{13} dapat diganti dengan a_3 .

Dari ketergantungan CE $\rightarrow$ A, maka a_{13} dan a_{41} dapat diganti dengan a_1.

Sehingga pada akhirnya didapat baris ketiga dimana semua nilai yaitu $a_1 a_2 a_3 a_4 a_5$ Jadi penggabungan tersebut adalah tidak hilang .

A	B	C	D	E
a_1	b_{12}	a_3	a_4	b_{15}
a_1	a_2	a_3	a_4	b_{25}
a_1	a_2	a_3	a_4	a_5
a_1	b_{42}	a_3	a_4	a_5
a_1	b_{52}	a_3	a_4	a_5

Teorema : 3.4

Algoritma 3.2 adalah penguji yang tepat untuk pemisahan gabungan yang tidak hilang.

Bukti : Misal tabel akhir yang dihasilkan oleh algoritma 3.2 tidak memiliki suatu baris dengan semua elemennya a_j. Tabel tersebut dianggap sebagai relasi r dari R, barisnya adalah tupel-tupel, a_j dan b_{ij} adalah simbol yang dipilih dari domainnya pada atribut A_j dan relasi r memenuhi ketergantungan F.

Misal ditentukan $r \neq m_\alpha(r)$ maka r tidak memuat tupel $a_1, a_2, \ldots a_n$, tetapi untuk setiap R_i terdapat tupel t_i dalam r yaitu tupel pada baris ke i sedemikian sehingga t_i (R_i) memuat semua elemen - elemen dengan simbol $a_1, a_2 \ldots a_n$

Sehingga penggabungan dari $\Pi_{R_i}(r)$ memuat tupel-tupel dengan semua elemennya $a_1, a_2 \ldots a_n$.

Jadi tupel tersebut sesuai dengan ti untuk setiap i dan pemisahan memiliki gabungan yang hilang sebaliknya jika tabel akhir dari hasil algoritma 3.2 memiliki elemen $a_1, a_2 \ldots a_n$ pada suatu barisnya atau paling sedikit terdapat elemen a_i pada atribut dari R_i dan elemen b_{ij} ada atribut yang lain . Dengan asumsi bahwa relasi r dari R memenuhi ketergantungan F yang mana pernyataan tersebut dapat disingkat sebagai :

$$r = \{ a_1, a_2 \ldots a_n \mid (\exists\, b_{11}) \ldots\ldots (\exists\, b_{kn}) \\ ((R(w_1) \wedge \ldots\ldots \wedge R(w_k)) \}$$

dan didefinisikan suatu pemetaan $m_\alpha(r)$

$$m_\alpha(r) = \{a_1, a_2 \ldots a_n \mid R(a_1, a_2 \ldots a_n) \wedge \ldots\}$$

jadi $m_\alpha(r) \subseteq r$, dilain pihak, lemma 3.5.a menyatakan $r \subseteq m_\alpha(r)$

jadi didapatkan $r = m_\alpha(r)$ dan pemisahan α adalah memiliki gabungan yang tidak hilang.

Teorema : 3.5.

Jika $\alpha = (R_1, R_2)$ adalah pemisahan dari R dan F adalah himpunan ketergantungan fungsional, maka α memiliki gabungan yang tidak hilang dengan memenuhi F. Jika dan hanya jika $(R_1 \cap R_2) \rightarrow (R_1 - R_2)$ atau $(R_1 \cap R_2) \rightarrow (R_2 - R_1)$.

(Catatan : ketergantungan ini tidak perlu dalam F tetapi harus dalam F^+)

Bukti :

$\Longrightarrow$ Dengan algoritma 3.2 didapatkan tabel sebagai berikut :

	$R_1 \cap R_2$	$R_1 - R_2$	$R_2 - R_1$
baris 1	a	a	b
baris 2	a	b	a

Karena $R_1 \cap R_2 = \{ A \mid A \in R_1 \wedge A \in R_2 \}$ maka pada kolom ke satu semua elemennya a.

Karena $R_1 - R_2 = \{ A \in R_1 \wedge A \in R_2 \}$ maka pada kolom ke dua elemen baris ke satu a, sedang baris ke dua b, dan dengan cara sama untuk kolom ke tiga.

Karena diketahui α adalah pemisahan gabungan yang tidak hilang maka : pada kolom 2 dan 3 nilai b dapat diganti a.

Menurut definisi ketergantungan fungsional maka :

$(R_1 \cap R_2) \longrightarrow (R_1 - R_2)$

dan $(R_1 \cap R_2) \longrightarrow (R_2 - R_1)$.

dari sebab $(R_1 \cap R_2) \longrightarrow (R_1 - R_2)$ maka,

jika $t_1(R_1 \cap R_2) = t_2(R_1 \cap R_2)$ terdapat :

$t_1(R_1 - R_2) = t_1(R_1 - R_2)$ dan karena

$(R_1 \cap R_2) \longrightarrow (R_2 - R_1)$ maka :

$t_1(R_1 \cap R_2) = t_2(R_1 \cap R_2)$ terdapat

$t_1(R_2 - R_2) = t_2(R_2 - R_1)$.

Jadi semua elemen-elemen suatu tupel adalah sama yaitu a. Karena terdapat nilai a semua pada satu baris maka memenuhi algoritma 3.2 dan pemisahan α adalah penggabungan yang tidak hilang.

□

Contoh : 3.10

R [ABC]

$F = \{ A \longrightarrow B \}$

$\alpha_1 = \{ AB, AC \}$ $\alpha_2 = \{ AB, BC \}$

1. $AB \cap AC = A$

   $AB - AC = B$

   karena $A \longrightarrow B$ maka α_1 adalah pemisahan gabungan yang tidak hilang.

2. sedangkan :

   $AB \cap BC = B$

   $AB - BC = A$

   karena $B \not\rightarrow A$ maka α_2 adalah bukan pemisahan gabungan yang tidak hilang.

Cara lain :

1. Misal r =

A	B	C
a_1	a_2	b_{13}
a_1	a_2	a_3

$F = \{ A \longrightarrow B \}$

$t_1(A) = t_2(A)$

$$t_1(B) = t_2(B)$$

$$\Pi_{AB}(r) = \begin{array}{cc} A & B \\ \hline a_1 & a_2 \end{array}$$

$$\Pi_{AC}(r) = \begin{array}{cc} A & C \\ \hline a_1 & b_{13} \\ a_1 & a_3 \end{array}$$

$$m_{\alpha_1}(r) = \Pi_{AB}(r) \text{ (X) } \Pi_{AC}(r)$$

$$= \begin{array}{ccc} A & B & C \\ \hline a_1 & a_2 & b_{13} \\ a_1 & a_2 & a_3 \end{array}$$

Karena $r = m_{\alpha_1}(r)$ maka α_1 adalah pemisahan gabungan yang tidak hilang.

2. Untuk $\alpha_2 = \{ AB, BC \}$ $F = \{A \rightarrow B\}$

$$r = \begin{array}{ccc} A & B & C \\ \hline a_1 & a_2 & b_{11} \\ b_{12} & a_2 & a_3 \end{array}$$

$$\Pi_{AB}(r) = \begin{array}{cc} A & B \\ \hline a_1 & a_2 \\ b_{12} & a_2 \end{array}$$

$$\Pi_{BC}(r) = \begin{array}{cc} B & C \\ \hline a_2 & b_{11} \\ a_2 & a_3 \end{array}$$

$$m_{\alpha_2}(r) = \Pi_{AB}(r) \ (X) \ \Pi_{BC}(r)$$

$$= \begin{array}{ccc} A & B & C \\ \hline a_1 & a_2 & b_{11} \\ a_1 & a_2 & a_3 \\ b_{12} & a_2 & b_{11} \\ b_{12} & a_2 & a_3 \end{array}$$

$r \neq m_{\alpha_2}(r)$ jadi α_2 bukan pemisahan gabungan yang tidak hilang.

□

3.8. PEMISAHAN YANG MENJAGA KETERGANTUNGAN

Sifat lain yang penting dari pemisahan relasi R kedalam $\alpha = \{ R_1, R_2 \ldots R_k)$ adalah adanya implikasi antara ketergantungan F untuk R dengan proyeksi ketergantungan F dari relasi R_i.

Secara formal, proyeksi F dari relasi R_i ditulis dengan notasi $\Pi_{R_i}(F)$ yaitu himpunan ketergantungan $V \longrightarrow W$ adalah F^+ sedemikian sehingga $VW \subseteq R_i$. Pemisahan α disebut menjaga ketergantungan F jika Union semua ketergantungan dalam $\Pi_{R_i}(F)$ berimplikasi logis terhadap semua ketergantungan dalam F.

Pengujian dari menjaga ketergantungan.

Pada prinsipnya untuk menguji pemisahan $\alpha = \{ R_1, R_2 \ldots R_k \}$ menjaga himpunan ketergantungan F adalah dengan menghitung F^+ dan proyeksinya pada semua R_i, $1 \leq i < k$.

Kemudian diuji apakah Union semua $\Pi_{R_i}(F)$ adalah cover himpunan F.

Algoritma : 3.3

- Diketahui (input) :

  Pemisahan $\alpha = \{ R_1, R_2 \ldots R_k \}$

  himpunan ketergantungan F.

- Ditanya, (Output)

  menentukan apakah α menjaga ketergantungan

- Metode :

  Misal : $G = \bigcup_{i=1}^{k} \Pi_{R_i} (F)$

  Untuk menguji apakah G merupakan Cover F, perhatikan setiap $V \longrightarrow W$ dalam F dan menghitung V^+ berdasarkan G yang berisi W.

  dengan mendefinisikan operasi R pada himpunan atribut Z yang memenuhi himpunan ketergantungan F untuk menggantikan Z dengan $Z \cup ((Z \cap R)^+ \cap R)$ yaitu closure yang memenuhi F. Operasi ini menghubungkan Z dengan Atribut A sedemikian sehingga $(Z \cap R) \longrightarrow A$ ada didalam Π_{R_i} (F) .

penghitungan V^+ yang memenuhi G dimulai dengan V dan diteruskan melalui relasi-relasi R_i sehingga menghasilkan V^+ secara formal algoritma ini adalah :

Z = V

While mengganti Z menjadi do

for i = 1 to k

$Z = Z \cup ((Z \cap R_i)^+ \cap R_i)$

Jika W himpunan bagian Z maka V $\longrightarrow$ W dalam G.

Jika setiap V $\longrightarrow$ W dalam F adalah berada dalam G maka pemisahan α menjaga ketergantungan.

Contoh : 3.11

R [A B C D]

α = { AB, BC, CD }

F = { A $\longrightarrow$ B, B $\longrightarrow$ C, C $\longrightarrow$ D, D $\longrightarrow$ A }

Himpunan ketergantungan F didalam F^+ yaitu semua atribut menentukan hubungan fungsional dengan semua atribut lainnya.

Proyeksi F terhadap AB didapatkan ketergantungan $A \longrightarrow B$ dan $B \longrightarrow A$ dengan cara sama $B \longrightarrow C$ dan $C \longrightarrow B$ dalam $\Pi_{BC}(F)$ dan $C \rightarrow D$ dan $D \rightarrow C$ dalam $\Pi_{CD}(F)$ dari ketergantungan $D \longrightarrow C$, $C \longrightarrow B$ dan $B \longrightarrow A$ didapatkan $D \longrightarrow A$. Kemudian dengan algoritma 3.3 ditunjukkan $D \rightarrow A$ diturunkan secara logis dari :

$G = \Pi_{AB}(F) \cup \Pi_{BC}(F) \cup \Pi_{CD}(F)$

dimulai dengan mengambil :

$Z = \{ D \}$

Untuk ini digunakan operasi CD sebab

$D \in \{ CD \}$

$Z = \{ D \} \cup (\{D\} \cap \{CD\}^+ \cap \{CD\})$

$= \{ D \} \cup (\{D\}^+ \cap \{CD\})$

$= \{ D \} \cup (\{ABCD\} \cap \{CD\})$

$= \{ D \} \cup \{CD\}$

$= \{CD\}$

dengan operasi BC

$Z = \{CD\} \cup (\{CD\} \cap \{BC\}^+ \cap \{BC\})$

$= \{CD\} \cup (\{ABCD\} \cap \{BC\})$

$= \{CD\} \cup \{BC\} = \{ BCD \}$

dengan Operasi AB

$Z = \{BCD\} \cup (\{BCD\} \cap \{AB\}^+ \cap \{AB\})$

$= \{BCD\} \cup (\{ABCD\} \cap \{AB\})$

$= \{BCD\} \cup \{AB\}$

$= \{ABCD\}$

Sehingga dengan memperhatikan G,

$\{ D \}^+ = \{ ABCD \}$ yang memuat A maka G berimplikasi dengan A, $G \rightarrow A$, maka pemisahan α adalah menjaga ketergantungan F sebab anggota-anggota F yang lain didalam G^+.

$G^+ = \{A \rightarrow B,\ B \rightarrow A,\ B \rightarrow C,\ D \rightarrow C,\ C \rightarrow D \}.$

Teorema : 3.6.

Jika $V \longrightarrow W$ didalam G maka algoritma 3.3 penentu yang tepat untuk sifat menjaga ketergantungan.

Bukti :

Misal $V \longrightarrow W$ dalam G maka algoritma 3.3 merupakan urutan tahap-tahap untuk mendapatkan closure V^+ yang memenuhi G, dan semua atribut W termuat dalam G . Setiap langkah-langkah itu merupakan penerapan dari ketergantungan dalam G dan maka dari itu ketergantungan tersebut dalam $\Pi_{R_i}(F)$.

□

3.9. NOTASI KUNCI

Jika R adalah suatu relasi dengan atribut $A_1, A_2, \ldots, A_n$ dan F adalah himpunan ketergantungan fungsional. V adalah himpunan bagian dari $A_1, A_2, \ldots, A_n$ adalah kunci dari R jika :

1. $V \rightarrow A_1A_2 \ldots A_n$ berada dalam closure F^+
2. Tidak ada $W \subseteq V$ sedemikian sehingga $W \rightarrow A_1A_2 \ldots A_n$ berada dalam closure F^+.

Suatu kunci V dari relasi R merupakan salah satu atau beberapa atribut dalam R sehingga setiap nilai dari kunci V berelasi dengan setiap tupel t dari relasi R. Dengan demikian setiap nilai dari kunci tidak boleh berharga nol, sebab bila nilai suatu atribut adalah nol maka ini dianggap sebagai nilai yang kosong, sehingga tidak mungkin dapat membuat relasi dengan himpunan kosong, dan nilai-nilai dari atribut kunci adalah tunggal. hal ini disebabkan bila tidak tunggal maka akan terdapat tupel-tupel pengulangan yang merupakan suatu redundansi dan bukan rancangan sistem basis data yang baik. Jadi terdapat pemetaan yang satu-satu antara nilai-nilai atribut kunci dengan tupel-tupel dalam relasi R. atua suatu kunci dapat didefinisikan sebagai berikut :

Definisi : 3.4

R adalah suatu relasi, suatu himpunan bagian dari R yaitu V adalah kunci Utama dari relasi R jika pada beberapa relasi r(R) yang manapun untuk semua $t_1(r) \neq t_2(r)$ yaitu tupel-tupel r(R). Sedemikian sehingga $t_1(V) \neq t_2(V)$ atau tidak ada dua tupel dari relasi r(R) yang manapun dapat mempunyai nilai yang sama pada atribut kunci V.

Disamping adanya kunci utama pada suatu relasi R terdapat juga atribut yang merupakan calon kunci. Jika $r_1(R)$ dan $r_2(R)$ adalah dua relasi dari relasi R. Dan Z adalah calon kunci bagi relasi R maka Z merupakan kunci bagi salah satu relasi $r_1(R)$ atau $r_2(R)$.Jadi Z bukan kunci bagi setiap relasi r(R) dalam relasi R.

Karena adanya suatu pemetaan satu-satu antara nilai-nilai atribut kunci dengan tupel-tupel pada relasi R maka jika $r_i(R)$ adalah pemisahan dari relasi R dan t_i (r) adalah tupel-tupel pada $r_i(R)$ untuk setiap tupel t pada R dapat ditentukan tupel- tupel $t_i(r)$ dari $r_i(R)$.

Dengan menggunakan notasi kunci yaitu memasukkan suatu atribut kunci utama dalam beberapa relasi yang didapat dari pemisahan suatu relasi maka akan didapatkan relasi semula seandainya diadakan penggabunga relasi tersebut jadi bila V adalah atribut kunci maka $V \in r_i(R)$ dan $R = \bigcup_{i=1}^{n} r_i(R)$.

Contoh : 3.12.

Misal R adalah relasi dengan empat atribut yaitu A_1, A_2, A_3, A_4 dan ditulis

R $[A_1\ A_2\ A_3\ A_4]$.

dari relasi R dapat dibuat sebanyak 2^4 relasi-relasi pemisahan dari R .Namun untuk membentuk kembali relasi R tidak perlu menggunakan semua relasi pemisahan dari R, tetapi cukup menggunakan beberapa relasi saja yang memungkinkan untuk mengkaitkan relasi - relasi pemisahan tersebut. Untuk itu jika diambil relasi pemisahan adalah

r_1 = R $[A_1 A_2]$, r_2 = R $[A_1 A_3]$, r_3 = R $[A_1 A_4]$.

sehingga relasi R dapat dibentuk dengan Union semua r_i (R).

$$R = \bigcup_{i=1}^{3} r_i(R).$$

Definisi : 3.5.

Atribut A dalam relasi R disebut prime jika A adalah anggota dari suatu kunci untuk R dan jika A bukan anggota dari suatu kunci maka A disebut non prime.

Contoh : 3.13

R [ABCD]

F = { AB → C, B → D, BC → A }

Maka

AB dan BC adalah kunci

A, B, C adalah prime

D adalah non Prime

3.10. NORMALISASI DENGAN KETERGANTUNGAN FUNGSIONAL

Mengingat tujuan perancangan basis data relasional maka mengharuskan suatu relasi memiliki bentuk-bentuk normal tertentu. Bentuk normal yang dimaksud adalah bentuk relasi antar atribut-atribut sedemikian sehingga redundansi yang terdapat pada suatu relasi tertentu dapat direduksi yang pada akhirnya didapatkan bentuk relasi yang efektif, dalam arti tidak ada entitas-entitas yang berulang dan setiap entitas dari suatu atribut berelasi dengan entitas dari atribut yang lain pada satu tupel.

Beberapa bentuk normal yang dikenal yaitu :

Bentuk normal pertama (1NF), bentuk normal kedua (2NF), bentuk normal ketiga (3NF), bentuk normal Boyce-Codd (BCNF) , bentuk normal keempat (4NF), bentuk normal kelima (5NF). namun dalam tulisan ini akan dijelaskan beberapa diantaranya dan pada khususnya bentuk normal ketiga (3NF) dan bentuk normal Boyce-codd (BCNF). Sebab pada bentuk normal tersebut didasari teori yang dibahas dalam tulisan ini juga, yaitu : Ketergantungan fungsional dan pengertian tentang Notasi kunci .

Definisi : 3.6

Suatu relasi berada dalam bentuk normal ke tiga (3NF) jika dan hanya jika untuk setiap tupel dari R memuat nilai kunci utama.

Definisi diatas dapat juga dikatakan bahwa relasi R berada dalam bentuk normal ke tiga (3NF) jika V $\longrightarrow$ A adalah ketergantungan dalam R, dan A tidak dalam V, sehingga V merupakan kunci utama atau A adalah prime.

Jika $V \longrightarrow A$ ketergantungan dalam R maka satu dari dua hal berikut dipenuhi yaitu :

1. V himpunan bagian dari kunci.
2. V himpunan bagian dari bukan kunci.

Pada hal pertama, $V \longrightarrow A$ disebut "Ketergantungan sebagian" (partial) dan pada hal kedua disebut "Ketergantungan transitif", hal yang kedua ini dikarenakan bahwa jika W adalah kunci, maka $W \longrightarrow V \longrightarrow A$ adalah barisan non-trivial dari ketergantungan sebab $V \not\subset W$, sedang $A \not\subset V$ dan $A \not\subset W$ dan A adalah non-prime.

Jika relasi R tidak memiliki ketergantungan sebagian meskipun memiliki ketergantungan transitip maka R disebut berada dalam bentuk normal kedua (2NF) atau didefinisikan sebagai berikut :

Definisi : 3.7.

Relasi R berada dalam bentuk normal ke dua (2NF) jika dan hanya jika setiap atribut yang bukan kunci bergantung penuh pada kunci utama.

3.10.1. Pemisahan yang menjaga ketergantungan didalam 3NF.

Algoritma :3.4.

- Diketahui : (Input)

  Suatu relasi R

  himpunan ketergantungan F

- Ditanya (Output)

  Pemisahan yang menjaga ketergantungan dari R sedemikian sehingga setiap relasi dalam (3NF) dengan memenuhi proyeksi dari F pada R.

- Metode :

  Jika suatu atribut dari R tidak dimasukkan dalam ketergantungan dari F baik pada bagian kiri atau kanan maka atribut tersebut berelasi dengan dirinya sendiri.

  Jika satu dari ketergantungan dalam F memuat semua atribut-atribut dari R maka menghasilkan relasi R itu sendiri. Oleh karena itu pemisahan α menghasilkan relasi VA untuk setiap ketergantungan $V \longrightarrow A$ didalam F. Jadi jika :

$V \longrightarrow A_1$, $V \longrightarrow A_2$, $V \longrightarrow A_n$ dalam F maka relasi yang dihasilkan dari pemisahan α memuat $V, A_1, A_2 ..., A_n$.

□

Teorema : 3.7.

Algoritma 3.5 menghasilkan pemisahan yang tidak hilang dalam bentuk normal ke tiga (3NF)

Bukti :

Selama ketergantungan diproyeksikan ke dalam cover dari F maka pemisahannya akan menjaga ketergantungan. Akan ditunjukkan bahwa suatu relasi [WB] untuk setiap ketergantungan $W \longrightarrow B$ dalam cover minimal, berada dalam 3NF.

Andaikan $V \longrightarrow A$ tidak memenuhi 3NF dari WB sehingga A tidak dalam V, V bukan kunci utama [WB] dan A non prime, karena $VA \subseteq WB$ dan $V \longrightarrow A$ yang secara logis diturunkan dari F, mengingat ini maka terdapat dua kemungkinan :

1. $A = B$

Untuk A tidak dalam V, $V \subseteq W$ dan V bukan kunci utama untuk [WB], karena $V \subseteq W$ maka $V \longrightarrow B$. Kemudian karena $A = B$ maka $V \longrightarrow A$, jadi $W \longrightarrow B$ menjadi cover minimum, padahal diketahui $W \longrightarrow B$ adalah bagian dari cover minimum sehingga kontradiksi.

2. $A \neq B$

Untuk W adalah kunci utama [WB] sehingga terdapat $Z \subseteq W$ yang menjadi kunci utama WB karena A dalam W dengan asumsi $A \neq B$ dan A tidak berada dalam Z sebab A non prime, maka dari $Z \subseteq W$, $Z \longrightarrow B$ termasuk dalam $W \longrightarrow B$ dan $W \longrightarrow B$ menjadi Cover minimum, sehingga Kontradiksi.

□

3.10.2. Pemisahan kedalam 3NF dengan penggabungan yang tidak hilang dan menjaga ketergantungan.

Suatu pemisahan $\alpha = \{ R_1, R_2 \ldots R_k \}$ dari relasi R sehingga α memiliki gabungan yang tidak hilang, dan $\beta = \{ S_1, S_2, \ldots S_m \}$ adalah suatu pemisahan yang menjaga ketergantungan dari F dengan setiap S_i berada dalam 3NF . Jika V adalah kunci dari R maka suatu pemisahan $\beta = \{ S_1, S_2, \ldots S_m \}$ dari V dalam 3NF adalah memiliki dua sifat tersebut.

Teorema : 3.8.

β Berada dalam pemisahan 3NF dari R yang dibentuk oleh algoritma 3.5 dan V menjadi kunci untuk R, kemudian $\gamma = \beta \cup \{ V \}$ adalah pemisahan dari R dengan seluruh relasi dalam 3NF. Pemisahan ini mempunyai sifat menjaga ketergantungan dan merupakan gabungan yang tidak hilang.

Bukti :

Ketergantungan transitip dan sebagian dalam V mengimplikasikan : bahwa himpunan bagian sejati V menentukan secara fungsional V dan bukan R, sehingga V tidak akan menjadi kunci, kemudian V sendiri menjadi anggota β dalam 3NF, sehingga γ menjaga ketergantungan terhadap β.

Dengan algoritma 3.2, bahwa suatu baris pada relasi V semuanya menjadi a dan dengan memperhatikan $A_1, A_2, \ldots A_k$ yaitu atribut-atribut dari $R \longrightarrow V$, dan diketahui Closure V^+.

Dari algoritma 3.7. yang pada akhirnya semua atribut digabungkan, dimana V sebagai kunci dari R. Dengan induksi ditunjukkan bahwa kolom ke i berkorespondensi dengan A_i, dan baris dari V adalah himpunan a_i.

untuk i = 0 adalah sebagai dasar.

untuk i - 1 maka A_i digabungkan dengan V^+, sebab dari beberapa ketergantungan $W \longrightarrow A_i$ dimana $W \subseteq V \cup \{ A_1, A_2, \ldots . A_{i-1} \}$

Sehingga WA_i dalam γ dan baris pada WA_i semuanya a_i setelah kolom dari V_i untuk

$A_1, A_2, \ldots A_{i-1}$ dibuat a maka nilai-nilai pada baris V_i juga semuanya a_i.

□

Definisi : 3.8.

Suatu relasi R dalam BCNF jika untuk semua ketergantungan yang memenuhi R yaitu $V \rightarrow W$, dimana $V \subseteq R$ dan $W \subseteq R$ paling sedikit memenuhi salah satu dari :

1. $V \rightarrow W$ adalah ketergantungan trivial.
2. V adalah kunci utama untuk relasi R.

Bila R tidak dalam BCNF, dapat dibuat pemisahan R kedalam $\alpha = \{ R_1, R_2, \ldots, R_k \}$ dimana R_i, $i = 1, 2, \ldots k$ berada dalam BCNF yang juga memiliki sifat pemisahan gabungan yang tidak hilang namun tidak menjaga ketergantungan.

Lemma : 3.6.

1. Misal R adalah suatu relasi dengan ketergantungan fungsional F. Suatu pemisahan $\alpha = \{ R_1, R_2, \ldots, R_k \}$ dari R

yang tidak hilang dalam F. Untuk suatu bagian ke i yaitu $F_i = \Pi_{R_i}(F)$ dan $\beta = \{S_1, S_2, \ldots, S_m\}$ adalah pemisahan R kedalam :

$\{R_1, R_2, \ldots, R_{i-1}, S_1, S_2, \ldots, S_m, R_{i+1}, \ldots, R_k\}$

adalah tak hilang dalam F.

2. Misal R, F dan α dari (1), dan diketahui pemisahan

$\gamma = \{R_1, R_2, \ldots, R_{i-1}, R_{i+1}, \ldots, R_k\}$ dari

R adalah di dalam pemisahan α , maka γ juga bersifat tidak hilang dalam F.

Bukti :

a. Jika relasi r adalah untuk menunjukkan R, dan proyeksinya r_j untuk setiap R_j , kemudian r_i diproyeksikan ke S_p untuk setiap S_p yang memiliki ketergantungan tak hilang, maka bila S_p digabungkan akan diperoleh r_i dengan cara sama penggabungan r_j akan diperoleh r.

b. Jika relasi r dari R diproyeksikan

kepada R_i, $i = 1,2, \ldots n$ maka bila digabungkan kembali proyeksi R_i akan diperoleh r, padahal dari lemma 3.5 (a) $r \subseteq m_r(r)$, sehingga $r = m_r(r)$. Jadi γ adalah pemisahan yang tak hilang.

□

Algoritma : 3.5

Pemisahan gabungan yang tak hilang dalam BCNF.

- Diketahui (input) :

  relasi R

  ketergantungan fungsional F

- Ditanya (output)

  Pemisahan gabungan yang tak hilang dari R dalam BCNF (dimulai dari ρ memuat R).

- Metode :

  Misal ρ adalah pemisahan dari R dan memiliki sifat gabungan yang tak hilang yang memenuhi F. Jika S adalah suatu relasi dalam ρ dan S tidak dalam BCNF. Suatu ketergantungan $V \longrightarrow A$ dalam S, dimana V bukan kunci utama S dan A $\not\subset$ V, misal S_1 berisi A dan atribut-atribut dari V dan S_2 berisi semua atribut-atribut S kecuali A, $S_1 \subset S$ dan $S_2 \subset S$ maka V = S − A, sehingga V adalah kunci utama S.

Dengan teorema 3.5 pemisahan S ke dalam S_1 dan S_2 memiliki sifat pemisahan gabungan yang tidak hilang dalam proyeksi ketergantungan dari S, untuk ($S_1 \cap S_2$) = V , ($S_1 - S_2$) = A maka ($S_1 \cap S_2$) $\rightarrow$ ($S_1 - S_2$), sehingga menurut Lemma 3.6 S_1 dan S_2 pemisahan dari S yang tidak hilang. Untuk membuat pemisahan yang berada dalam BCNF dimulai dari ρ yang memuat R sendiri kemudian dibuat pemisahan sehingga menjaga sifat gabungan yang tidak hilang.

□

Contoh : 3.14

R [MDJRST]

dengan M = mata kuliah R = ruang

D = dosen S = siswa

J = jam T = tingkat

F = ketergantungan fungsional

M $\longrightarrow$ D = Setiap mata kuliah diampu satu dosen.

JR $\longrightarrow$ M = Hanya ada satu mata kuliah yang berlangsung dalam ruangan pada waktu tertentu.

JD $\longrightarrow$ R = Dosen dapat berada hanya dalam satu ruangan pada suatu waktu tertentu.

MS $\longrightarrow$ T = Semua siswa memiliki satu tingkat dalam semua mata kuliah.

JS $\longrightarrow$ R = Siswa dapat berada hanya dalam satu ruangan pada suatu waktu tertentu.

- Dari semua ketergantungan fungsional diatas adalah tidak ada yang trivial.
- JS adalah satu-satunya kunci untuk [MDJRST] sebab :

JS $\longrightarrow$ R

JS $\longrightarrow$ R dan JR $\longrightarrow$ M $\Longrightarrow$ JS $\longrightarrow$ M

$JS \rightarrow M$ dan $M \rightarrow D \Rightarrow JS \rightarrow D$

$JS \rightarrow M$ dan $MS \rightarrow T \Rightarrow JS \rightarrow T$

jadi $JS \rightarrow RMDT$

Pandang ketergantungan $MS \rightarrow T$, dimana MS bukan kunci, sehingga relasi [MDJRST] dipisahkan menjadi [MST] dan [MDJRS] sebab [MST] $\cap$ [MDJRS] = MS dan [MST]− [MDJRS] = T. Sehingga $MST \rightarrow MDJRS$.

- dalam pemisahan [MST] terdapat ketergantungan $MS \rightarrow T$ dan MS sebagai kunci.
- dalam pemisahan [MDJRS] terdapat ketergantungan :

$M \rightarrow D$ $SD \rightarrow R$

$JR \rightarrow M$ $JS \rightarrow R$

dan JS sebagai kunci

Jadi [MST] dalam BCNF

kemudian [MDJRS] dipisahkan lagi, bila dipilih $M \rightarrow D$ sebagai ketergantungan dalam [MDJRS], maka pemisahaan [MDJRS] akan menjadi [MD] dan [MJRS] dengan masing-masing ketergantungan adalah :

untuk [MD] yaitu M → D dan M sebagai kunci untuk [MJRS] yaitu MJ → R, JS → R, JR → S dan JS sebagai kunci. Jadi [MD] dalam BCNF. Kemudian [MJRS] dipisahkan menggunakan MJ → R sehingga menjadi [MJR] dan [MJS], masing-masing dengan ketergantungan untuk [MJR] adalah MJ → R dan MJ sebagai kunci, serta untuk [MJS] adalah JS → M dan JS adalah kunci.