

CHAPTER 3

DEPENDENSI DAN KUNCI RELASI

3.1 PENGENALAN

Teori relasional memasukan metoda untuk mengidentifikasi berbagai level redundansi dalam relasi. Istilah yang digunakan untuk mendefinisikan level redundansi adalah **form normal** relasi. Terdapat lima form normal. Relasi melanjutkan ke form normal yang lebih tinggi karena setiap bentuk redundansi dihilangkan. Tujuannya adalah untuk merubah setiap relasi kedalam bentuk normal yang lebih tinggi.

Dua property penting dibutuhkan untuk mendefinisikan form normal. Salah satu property berkenaan dengan dependensi antara atribut dan fakta dalam sistem. Tujuan perancangan adalah untuk meyakinkan bahwa tidak ada fakta yang disimpan lebih dari satu kali.

Properti yang lain adalah kunci relasi. Kunci relasi adalah sebuah himpunan atribut dimana nilai-nilainya menunjukkan baris-baris yang unik dalam relasi. Kunci relasi bergantung pada atribut dalam relasi dan keterikatan antara atribut atribut ini. Nanti kita akan melihat relasi non-redundan hanya menyimpan fakta tentang kunci relasi.

Chapter ini mendeskripsikan dependensi fungsional dan kunci relasi. Chapter berikutnya akan mendeskripsikan bagaimana dua properti digunakan untuk menentukan form normal sebuah relasi.

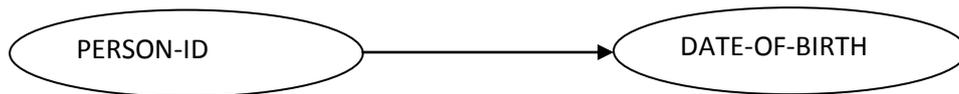
3.2 DEPENDENSI FUNGSIONAL

Secara informal dependensi fungsional adalah representasi formal dari fakta bernilai tunggal. Sebuah fakta, anda mungkin ingat, telah didefinisikan sebagai sebuah nilai yang dihubungkan dengan beberapa objek atau sebuah asosiasi antara dua objek. Dalam matematika, fakta ini direpresentasi oleh sebuah dependensi fungsional, yang menunjukan nilai unik dari satu atribut ditentukan oleh sebuah nilai yang diberikan dari atribut yang lain. Dependensi ini didefinisikan oleh sebuah pernyataan dependensi fungsional. Sebagai contoh, fakta bernilai tunggal bahwa seseorang mempunyai DATE-OF-BIRTH dapat direpresentasikan oleh pernyataan dependensi fungsional (DF):

PERSON-ID → DATE-OF-BIRTH

Statement ini berarti bahwa terdapat hanya satu nilai dari DATE-OF-BIRTH untuk sebarang nilai yang diberikan PERSON-ID. Dependensi fungsional dapat juga direpresentasikan oleh diagram dependensi fungsional seperti yang ditunjukkan dalam Gambar 3.1.

Gambar 3.1 Dependensi Fungsional



Sisi sebelah kiri dari sebuah dependensi fungsional disebut determinan (penentu). PERSON-ID adalah determinan dari dependensi fungsional di atas. Penting untuk dicatat bahwa sebuah nilai dari determinan menentukan hanya satu nilai dari sisi kanan dari dependensi fungsional. Pada contoh di atas sebuah nilai dari PERSON-ID menentukan satu nilai DATE-OF-BIRTH.

Sama pentingnya untuk dipahami bahwa jika lebih dari satu nilai dari satu atribut dapat diasosiasikan dengan satu nilai atribut yang lain maka tidak terdapat dependensi fungsional. Sebagai contoh, jika seseorang dengan sebuah PERSON-ID mempunyai sejumlah ADDRESSES maka tidak benar untuk menuliskan bahwa

PERSON-ID → ADDRESSES

Akan tetapi, jika seseorang mempunyai satu alamat maka:

PERSON-ID → ADDRESS

Adalah benar. Dengan demikian dependensi fungsional ditentukan secara alami oleh informasi. Suatu pernyataan dependensi fungsional boleh memasukan lebih dari satu dependensi. Sebagai contoh, jika seorang mempunyai satu DATE-OF-BIRTH ada satu ADDRESS, maka kedua dependensi ini dapat ditunjukkan oleh satu pernyataan dependensi:

PERSON-ID → DATE-OF-BIRTH, ADDRESS

Adalah hal yang memungkinkan untuk dua himpunan atribut untuk saling terikat (dependent). Jika setiap proyek mempunyai satu manajer, dan setiap manajer mengatur satu proyek, maka kedua dependensi fungsional berikut adalah benar:

PROJECT-NO → MANAGER

MANAGER → PROJECT-NO

Panah dua arah digunakan untuk mendeskripsikan saling dependensi ini dan kedua dependensi fungsional menjadi

PROJECT-NO ↔ MANAGER

Dependensi Fungsional pada lebih dari satu Atribut

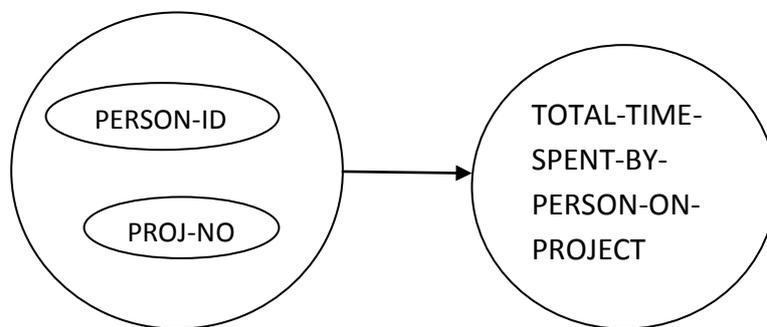
Kadang-kadang fakta disimpan kombinasi dari dua atau lebih atribut. Sebagai contoh, total waktu yang dihabiskan oleh seseorang pada proyek tertentu adalah sebuah fakta tentang hubungan antara seseorang dengan proyek. Disini tidak memadai untuk mengetahui hanya nilai PERSON-ID untuk mendapatkan nilai tunggal TOTAL-TIME-SPENT-BY-PERSON-ON-PROJECT. Ini disebabkan seseorang boleh bekerja pada satu proyek dan TOTAL-TIME-SPENT-BY-PERSON-ON-PROJECT akan berbeda untuk setiap proyek untuk orang tersebut. Dengan demikian kedua nilai PERSON-ID dan PROJ-NO dibutuhkan untuk mendapatkan nilai TOTAL-TIME-SPENT-BY-PERSON-ON-PROJECT.

Dependensi pada dua atau lebih atribut dinyatakan sebagai berikut:

PERSON-ID, PROJ-NO → TOTAL-TIME-SPENT-BY-PERSON-ON-PROJECT

Sekarang penentu (determinant) disusun lebih dari satu atribut. Dependensi fungsional yang demikian dapat juga dinyatakan dalam diagram dependensi seperti pada Gambar 3.2.

Gambar 3.2 Sebuah determinan dengan lebih dari satu atribut



Sekarang ke dua atribut dalam determinan dikelilingi lingkaran dan sebuah panah dari batas lingkaran yang berakhir pada atribut dimana nilai-nilai tunggalnya ditentukan oleh determinan tersebut.

Lagi, anda harus hati-hati untuk menggunakan aturan dependensi fungsional pada satu atribut dan dengan lebih dari satu atribut.

Dependensi Fungsional Penuh

Juga penting untuk dapat membedakan perbedaan antara dependensi fungsional dan dependensi fungsional penuh. Dependensi fungsional penuh haruslah tidak mempunyai atribut yang tidak dibutuhkan dalam determinan.

Untuk melihat perbedaan antara dependensi fungsional dengan dependensi fungsional penuh catat bahwa jika

PERSON-ID → DATE-OF-BIRTH

Maka itu juga benar bahwa

PERSON-ID, PERSON-NAME → DATE-OF-BIRTH

Tapi kita tak perlu tahu PERSON-NAME untuk mendapatkan DATE-OF-BIRTH; karena PERSON-ID sudah cukup. Dengan demikian

PERSON-ID, PERSON-NAME → DATE-OF-BIRTH

Tidaklah dependensi fungsional penuh karena kita tidak membutuhkan PERSON-NAME dalam determinan. Tapi

PERSON-ID → DATE-OF-BIRTH

Adalah dependensi fungsional penuh.

Sama halnya, jika setiap proyek mempunyai satu manajer maka

PROJECT-NO, PART-NO → QTY-USED

Adalah dependensi fungsional penuh tapi

PROJECT-NO, PART-NO, MANAGER → QTY-USED

Tidak karena kita tidak membutuhkan manajer dalam determinan untuk mendapatkan QTY-USED

3.3 NOTASI UNTUK MENGGAMBARKAN RELASI

Disamping penggunaan tabel, relasi dapat juga digambarkan dengan menggunakan notasi matematika. Notasi secara matematika memasukan masing-masing atribut relasi dan dependensi fungsional antara mereka.

Gambar 3.3 relasi ASSIGNMENTS

ASSIGNMENT		
PERSON-ID	DATE-OF-BIRTH	DEPT-NAME
P1	7 July 89	Accounts
P3	3 June 91	Sales
P2	11 Feb 85	Accounts
P4	22 Apr 84	Sales
P5	17 May 96	Sales

Sebagai contoh, misalkan kita mengambil relasi, ASSIGNMENTS, dalam Gambar 3.3. Relasi ini menunjukkan bagaimana seseorang ditugaskan pada department. Relasi ASSIGNMENTS digambarkan dalam notasi matematika dalam dua bagian:

$$\text{ASSIGNMENTS} = (\{\text{attribute}\}, \{\text{daftar dependensi fungsional}\})$$

Yang menjadi

$$\text{ASSIGNMENTS} = (\{\text{PERSON-ID, DEPT-NAME, DATE-OF-BIRTH}\}, \{\text{PERSON-ID} \rightarrow \text{DEPT-NAME, DATE-OF-BIRTH}\})$$

Ini menunjukkan bahwa ASSIGNMENTS mempunyai tiga atribut, PERSON-ID, DEPT-NAME dan DATE-OF-BIRTH. Terdapat juga dua dependensi dalam atribut tersebut. Satunya adalah PERSON-ID menentukan DATE-OF-BIRTH dan yang lainnya PERSON-ID menentukan DEPT-NAME. Masing-masing dependensi ini ditunjukkan oleh satu pernyataan dependensi fungsional. Banyak contoh dalam atau di luar buku ini menggunakan notasi aljabar sebagai pengganti nama sebenarnya untuk mengurangi jumlah penulisan dalam contoh. Penggunaan notasi ini, suatu relasi dengan tiga atribut, A, B, dan C dapat dispesifikasikan sebagai

$$R1 = (\{A, B, C\}, \{AB \rightarrow C, B \rightarrow C\})$$

3.4 DEPENDENSI BANYAK NILAI

Kadang-kadang anda mungkin menemukan bahwa dependensi fungsional tidak dapat menyatakan ekspresi suatu hubungan antar atribut. Sebagai contoh:

Setiap orang, diidentifikasi oleh PERSON-ID, dapat memiliki banyak skill, diidentifikasi oleh SKILL. Setiap skill dapat dimiliki oleh banyak orang.

Dalam kasus ini, masing-masing

$$\text{PERSON-ID} \rightarrow \text{SKILL}, \text{ atau}$$
$$\text{SKILL} \rightarrow \text{PERSON-ID}$$

adalah benar.

Beberapa metodologi menggunakan konsep dependensi banyak nilai untuk mendefinisikan bahwa sebuah atribut selalu diasosiasikan dengan sebuah himpunan yang diberikan dari nilai atribut yang lain. Dengan demikian, fakta bahwa seseorang mempunyai banyak skill dinyatakan sebagai

$$\text{PERSON-ID} \twoheadrightarrow \text{SKILL}$$

Disini \twoheadrightarrow menyatakan dependensi banyak nilai.

Dependensi banyak nilai didefinisikan awalnya hanya untuk relasi. Dalam cara ini mereka berbeda dari dependensi fungsional yang bebas (independen) dari relasi.

Banyak metodologi, akan tetapi, sekarang menggunakan dependensi banyak nilai dalam hal tertentu sebagai dependensi fungsional, yaitu keluar dari konteks relasi.

Suatu dependensi banyak nilai mengatakan bahwa suatu nilai tunggal dari sebuah himpunan atribut, A, menentukan sebuah himpunan nilai dari atribut lain. Lebih lanjut, himpunan nilai ini adalah independent sama dari sebarang dependensi lain dari atribut A.

Sebagai contoh dari dependensi banyak nilai dari dua atribut pada atribut lain adalah:

PERSON-ID $\rightarrow\rightarrow$ SKILL-ID, DATE-ACQUIRED

Disini PERSON-ID menentukan himpunan SKILL-ID bersama dengan tanggal dimana setiap skill diperoleh setiap orang (person)

Hubungan antara Dependensi Banyak Nilai dengan Dependensi Fungsional

Terdapat hubungan antara dependensi fungsional dengan dependensi banyak nilai. Seringkali suatu dependensi fungsional dapat menyatakan fakta sama sebagai dependensi banyak nilai.

Contoh:

Dependensi fungsional

PROJECT-ID, PART-ID \rightarrow QTY-USED

mengatakan bahwa sebuah proyek menggunakan sebuah QTY-USED yang diberikan dari sebuah part.

Ini juga dapat diekpresikan sebagai

PROJECT-ID $\rightarrow\rightarrow$ PART-ID, QTY-USED

Yang mengatakan bahwa untuk setiap proyek yang diberikan terdapat sebuah himpunan part yang digunakan oleh proyek dan quantity (jumlah) dari part yang digunakan.

Metodologi cenderung menggunakan dependensi fungsional ketika memungkinkan dan tidak menggunakan dependensi banyak nilai untuk menunjukkan fakta yang dapat ditunjukkan oleh dependensi fungsional. Dengan demikian aturannya adalah untuk mengambil semua dependensi fungsional pertama kali dan kemudian menambahkan dependensi banyak nilai untuk menunjukkan informasi yang hilang.

Dengan demikian, anda mungkin mempunyai pernyataan seperti setiap orang mempunyai banyak kendaraan,

PERSON-ID $\rightarrow\rightarrow$ REG-NO

Bagaimana, jika DATE-PURCHASED juga

PERSON-ID,REG-NO→DATE-PURCHASED

3.5 KUNCI RELASI

Sebagai tambahan pada dependensi fungsional, properti lain yang harus dimengerti tentang relasi adalah kunci relasi. Nilai dari kunci relasi mengidentifikasi sebuah baris uniq dalam suatu relasi. Kunci relasi dibuat dari satu atau lebih atribut relasi. Atribut dalam kunci relasi mempunyai sifat-sifat berikut:

- Himpunan nilai atribut yan diberikan mengidentifikasi hanya satu baris dalam relasi
- Tidak ada subset dari atribut dalam kunci relasi yang juga adalah kunci relasi
- Kunci relasi tidak boleh mempunyai nilai nol

Gambar 3.4 menunjukkan satu contoh kunci relasi. Ini menunjukkan bahwa PERSON-ID adalah kunci relasi dari relasi PERSONS. Terdapat paling banyak satu baris dalam relasi dengan nilai yang diberikan oleh PERSON-ID. Dengan demikian untuk contoh terdapat hanya satu baris dalam PERSONS dengan PERSON-ID='P3'. NAME bukan kunci relasi karena boleh jadi terdapat banyak nama orang yang sama, sehingga terdapat lebih dari satu baris yang mempunyai nilai NAME yang sama.

Gambar 3.4 Relasi PERSONS

PERSONS		
PERSON-ID	DATE-OF-BIRTH	NAME
P1	7 July 79	Joe
P3	3 June 89	Mary
P2	11 Feb 65	Andrew
P4	22 Apl 84	Joe
P5	17 May 96	Jill

DATE-OF-BIRTH bukanlah kunci relasi PERSONS karena lebih dari satu orang dapat lahir pada tanggal yang sama. Sehingga lebih dari satu baris dapat mempunyai nilai yang sama dalam hal DATE-OF-BIRTH.

Penting untuk diingat bahwa kunci relasi adalah sebuah kunci untuk sembarang himpunan dari nilai relasi dan tidak hanya untuk himpunan nilai pda waktu tertentu yang diberikan. Dengan demikian kunci relasi mungkin muncul pada relasi di atas yaitu DATE-OF-BIRTH adalah kunci relasi karena pada waktu ini, setiap baris mempunyai nilai DATE-OF-BIRTH yang berbeda. Akan tetapi, adalah mungkin pada akhirnya terdapat dua orang mempunyai DATE-OF-BIRTH yang sama dalam database. Karena dua baris dengan nilai sama DATE-OF-BIRTH dapat terjadi, maka DATE-OF-BIRTH tidak dapat menjadi kunci relasi.

Relasi PERSONS mempunyai sebuah kunci dengan satu atribut saja. Ini juga perlu dalam banyak kasus untuk kunci yang dibuat lebih dari satu atribut untuk mengidentifikasi baris-baris unik. Sebagai

contoh, relasi WORK dalam Gambar 3.5 mempunyai sebuah kunci relasi dibuat dari dua atribut, PERSON-ID dan PROJ-NO.

Gambar 3.5 Relasi WORK

WORK		
PERSON-ID	PROJ-NO	TOTAL-TIME-SPENT-BY-PERSON-ON-PROJECT
P1	PROJ1	20
P3	PROJ1	16
P2	PROJ2	35
P2	PROJ3	42
P3	PROJ2	17
P2	PROJ1	83
P4	PROJ3	41

Nilai dari masing PERSON-ID dan PROJ-NO perlu untuk mengidentifikasi sebuah baris unik. Tidak mencukupi untuk menggunakan hanya nilai dari PERSON-ID atau nilai dari PROJ-NO saja untuk menemukan baris unik.

Beberapa Relasi dapat Mempunyai Lebih dari Satu Kunci

Relasi CONSULTATIONS dalam Gambar 3.6 menggambarkan konsultasi oleh dokter. Diasumsikan bahwa setiap konsultasi melibatkan satu dokter dan satu pasien. Relasi CONSULTATION mempunyai dua kunci. Yaitu {DOCTOR-NO, TIME-OF-VISIT} dan {PATIENT-NO, TIME-OF-VISIT}

Gambar 3.6 Relasi CONSULTATIONS

CONSULTATIONS		
DOCTOR-ID	PATIENT-ID	TIME-OF-CONSULTATION
Dr. Jones	Ms. James	10.00 am June 5
Dr. Jones	Mr. Able	11.15 am May 25
Dr. Smith	Ms. James	2.00 pm Apr 11
Dr. Ajax	Mr. Able	12.15 pm Aug 17
Dr. Jones	Ms. James	3.00 pm July 7

{DOCTOR-NO, TIME-OF-VISIT} adalah sebuah kunci relasi karena seorang dokter hanya dapat mempunyai satu konsultasi pada satu waktu.

{PATIENT-NO, TIME-OF-VISIT} adalah sebuah kunci karena seorang pasien dapat mempunyai hanya satu konsultasi pada satu waktu.

Anda harus mencatat bahwa {DOCTOR-NO, PATIENT-NO} bukanlah kunci relasi karena pasien yang sama mungkin mengunjungi docter yang sama lebih dari satu kali.

Dependensi Fungsional dan Kunci Relasi

Penting untuk diingat bahwa fakta atau dependensi fungsional adalah properti dari sistem. Mereka muncul dari data. Kunci relasi juga ditentukan oleh data dalam relasi. Sebagai contoh, fakta bahwa seseorang hanya mempunyai satu DATE-OF-BIRTH dan NAME akan berarti bahwa PERSON-ID adalah kunci relasi dalam relasi PERSON dalam Gambar 3.4.

Sama halnya fakta bahwa seorang dokter dapat melakukan paling banyak satu konsultasi pada satu waktu menyebabkan kunci relasi dalm relasi CONSULTATION dalam Gambar 3.5. Dengan demikian, harusnya mungkin untuk menentukan kunci relasi dari dependensi fungsional.

Kunci Asing

Istilah kunci asing juga digunakan secara meningkat dalam rancangan. Kunci asing adalah sebuah himpunan atribut dalam satu relasi sedemikian sehingga himpunan atribut ini bukanlah kunci relasi dalam relasi ini tapi sebuah kunci relasi pada relasi lain.

Dengan demikian, sebagai contoh, misalkan dua relasi dalam Gambar 3.3 dan 3.4 berada dalam satu database. Dalam kasus tersebut PERSON-ID adalah kunci relasi dari relasi PERSONS. Ini bukanlah sebuah kunci relasi dari relasi PROJECT-DATA tapi ini adalah kunci asing dari relasi PROJECT-DATA. Perhatikan bahwa PERSON-ID adalah bagian dari kunci relasi PROJECT-DATA.