
BAB 8

MODEL DATA DAN ENTITY-RELATIONSHIP MODEL

8.1. Model Data

Model data adalah sekumpulan konsep yang terintegrasi untuk mendeskripsikan data, hubungan antar data dan batasan – batasannya dalam suatu organisasi. Model data merepresentasikan suatu organisasi. Model data harus menyediakan konsep dasar dan notasi yang memungkinkan perancang basis data dan pemakai untuk dapat mengkomunikasikan pemahamannya mengenai organisasi data.

8.1.1. Komponen model data.

Komponen model data dapat dikategorikan menjadi 3 (tiga) bagian yang meliputi:

- Bagian struktural, memuat sekumpulan aturan untuk melakukan konstruksi basis data (database).
- Bagian manipulasi, melakukan definisi tipe operasi yang diijinkan pada data, termasuk operasi yang digunakan untuk melakukan perubahan (update), atau membaca data (retrieve) dari basis data dan untuk melakukan perubahan struktur basis data.
- Sekumpulan aturan mengenai integritas, yang akan menjaga keakuratan dari data dalam basis data (database).

8.1.2. Kelompok Model Data

Terdapat 3 (tiga) kelompok model data, yaitu model data berbasis objek, model data berbasis record dan model data fisik.

a. Model data berbasis objek.

Pada model ini menjelaskan data pada tingkat konsepsi dan view. Pada model ini terdapat beberapa macam model, yaitu :

- ER model (*Entity relationship Model*).
- OO model (*Objek Oriented Model*).
- *Binary Model*.

- Model data semantik.
- Model data infologikal, dan
- Model data fungsional.

b. Model data berbasis *record*.

Pada model ini menjelaskan data pada tingkat konsepsi dan view, memakai seluruh struktur logik basis data dan menyediakan uraian tingkat tinggi dari implementasi. Terdiri dari sejumlah fixed format *record* dengan berbagai tipe.

Pada model ini teradapat 3 (tiga) macam tipe,yaitu :

- Model data *relational*.
- Model data *network*, dan
- Model data *hirarki*.

b.1. Model relational.

Pada model ini menggambarkan data dan relationship diantara data oleh suatu koleksi tabel, contoh nya:

Customer

Name	Street	City	Number
Lowery	Maple	Queens	900
Shiver	Nort	Bronx	556
Shiver	Nort	Bronx	647
Hodges	Sidehill	Brooklyn	801
Hodges	Sidehill	Brooklyn	647

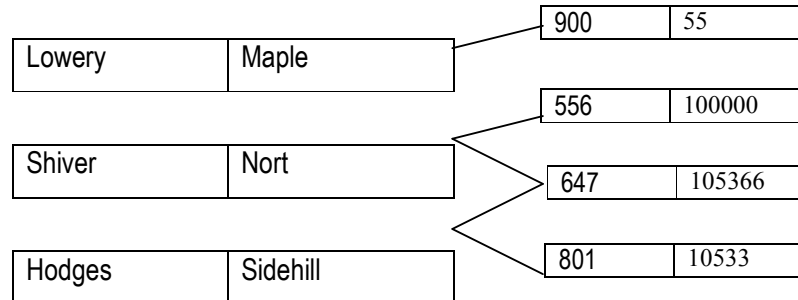
Account

Number	Balance
900	55
556	100000
647	1005366
801	10533

Gambar 8.1. Model *Relational*

b.2. Model *network* (jaringan)

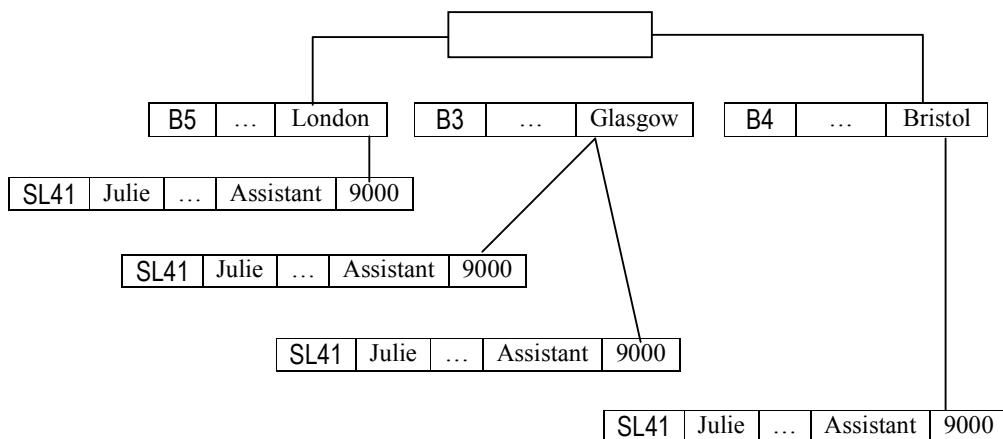
Pada model ini menggambarkan koleksi record – record dan relationship diantara data ditunjukkan oleh *link* (penghubung) yang dapat dipandang sebagai *pointer* (penunjuk), contoh nya:



Gambar 8.2. Model *Network*

b.3. Model hirarki

Pada model ini sama seperti pada model *network* dalam hal data dan relationship diantara data digambarkan oleh record dan link. Perbedaannya adalah record disusun sebagai “*collection of tree*” dengan sembarang grafik. contoh nya:



Gambar 8.3. Model *Hirarki*

8.1.3. Database Instance & Schema

Database instance adalah data yang berada dalam basis data pada suatu saat tertentu. *Database schema* merupakan rancangan (deskripsi) keseluruhan *database*. Skema dispesifikasikan selama proses desain basis data yang tidak diharapkan untuk selalu berubah,

tetapi data aktual yang berada dalam basis data bisa selalu berubah (misalkan dengan adanya penambahan data (*insert*), penghapusan data (*delete*) dan lain sebagainya.

Skema kadang disebut juga sebagai intension dari *database*, sedangkan *instance* disebut dengan *extention* atau state dari *dabase*.

Konsep *database* schema berhubungan dengan definisi *type* (bentuk) dalam bahasa pemrograman, sedangkan konsep *database instance* berhubungan dengan definisi *variable* dalam bahasa pemrograman. Sistem basis data mempunyai banyak schema berdasarkan tingkat abstraksi yaitu:

- *Physical schema* → tingkat terendah.
- *Conceptual schema* → tingkat menengah.
- *Eksternal schema* (subschema) → tingkat tertinggi.

Sistem basis data sangat mendukung semua tingkat pada ketiga *schema* tersebut. Pada *schema* conceptual menggambarkan semua item data dan *relationship* diantara item – item data serta batasan – batasan integritasnya, hanya terdapat satu conceptual *schema* didalam suatu *database*. Skema internal merupakan gambaran lengkap dari model internal, meliputi definisi dari stored record, metode representasi, data field dan *indeks* serta skema *hashing* jika ada, hanya terdapat satu *schema* dalam suatu *database*.

8.1.4. Mapping.

DBMS (*database management system*) bertanggung jawab didalam pemetaan diantara tiga tipe skema tersebut dan *DBMS* menguji skema untuk konsistensi yang berarti harus menguji bahwa setiap skema eksternal diturunkan dari skema konseptual, dan harus menggunakan informasi didalam skema konseptual untuk memetakan antara masing – masing skema eksternal dan skema internal.

Skema konseptual berhubungan dengan skema internal yang melalui *conceptual / internal mapping*. Hal ini memungkinkan *DBMS* untuk menemukan aktual record atau kombinasi record dalam penyimpanan fisik yang merupakan sebuah logical record dalam skema konseptual, bersama dengan batasan – batasan yang harus dijaga dalam operasi untuk logical record tersebut.

Dimungkinkan juga perbedaan – perbedaan dalam nama entitas, nama atribut, urutan atribut, tipe data dan lain sebagainya. Setiap skema eksternal berhubungan dengan skema konseptual melalui *eksternal / konseptual mapping*. Hal ini memungkinkan DBMS memetakan nama – nama dalam pandangan pemakai kedalam bagian yang relevan dalam skema konseptual.

8.2. Entity Relationship Model

Model data entity relationship (E-R) adalah berdasarkan pada persepsi dari dunia nyata yang terdiri dari sekumpulan objek – objek dasar yang disebut entity, dan relationship antara objek – objek tersebut. Pembentuk Model E-R (*Entity Relationship*) pada dasarnya terdiri dari 2 komponen, yaitu : Entiti (*Entity*) / entitas dan Relasi (*Relation*) atau hubungan.

8.2.1. Entiti

Merupakan suatu objek yang dapat diidentifikasi secara unik dalam lingkungan pemakai, suatu yang penting bagi pemakai dalam konteks sistem yang akan dibuat.

Sekelompok entiti yang sejenis dan berada dalam lingkup yang sama membentuk sebuah Himpunan Entiti.

Contoh → Himpunan Entiti Mahasiswa

atribut entiti ←

NIM	Nama	Alamat	Tgl Lahir
980001	Ali Akbar	Jl. Merdeka No.10, Jakarta 40121	02-01-1979
980002	Budi Haryanto	Jl. Gajah Mada No.2, Jakarta 45123	06-10-1978
980003	Rini Susanti	Jl. Adil No.123, Bogor 43123	27-07-1977

Himpunan Entiti Mahasiswa

Gambar 8.4. Himpunan entiti mahasiswa

8.2.2. Atribut

Merupakan elemen bagian dari entiti yang berfungsi menjelaskan karakter entiti. Contoh dalam Entiti Mahasiswa ada atribut : Nim, Nama, Alamat, Telpn dan Tgl Lahir.

Dimana Nim merupakan PK (Primary Keys) dan Nama, Alamat, Telp dan Tgl Lahir merupakan atribut Deskriptif.

8.2.3. Relasi / Hubungan

Relasi menunjukkan adanya hubungan di antara sejumlah entiti yang berada dalam sejumlah himpunan entiti yang berbeda. **Misalkan** himpunan entiti Mahasiswa berhubungan dengan himpunan entiti Matakuliah.

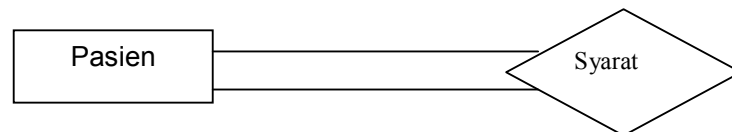
Himpunan Relasi : kumpulan semua relasi diantara entiti – entiti yang terdapat pada himpunan entiti – himpunan entiti tersebut. Istilah Himpunan relasi jarang digunakan, dan lebih sering disingkat dengan Relasi saja.

8.2.4. Varian Relasi

Relasi berfungsi menghubungkan antar entiti. Entiti yang berelasi tidak hanya terdiri dari dua relasi saja. Tetapi entiti bisa berelasi dengan dirinya sendiri atau berelasi lebih dari 2 entiti. Relasi ini dapat dikelompokkan menurut varian sebagai berikut :

✓ **Relasi Unary**

Relasi yang terjadi dari sebuah himpunan entiti ke himpunan entiti yang sama. Sering disebut sebagai Relasi Tunggal. Misalkan relasi yang terjadi pada Pasien dan Syarat. Relasi ini menunjukkan adanya persyaratan menjadi pasien rawat inap. Misalkan pasien hanya boleh mengikuti rawat inap bila telah terdaftar menjadi pasien rawat jalan.



Gambar 8.5. Relasi Unary

✓ **Relasi Binary**

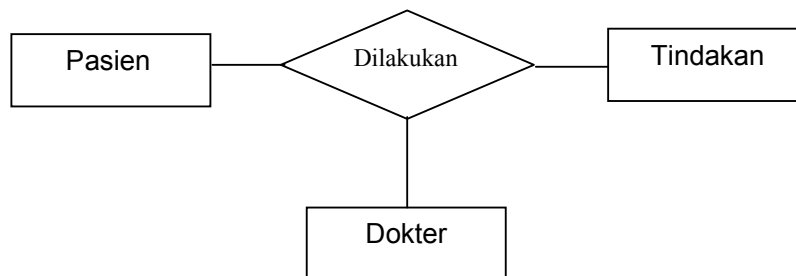
Relasi yang terjadi dari dua himpunan entiti. Relasi ini kerap terjadi dan paling banyak digunakan. Contoh relasi biner adalah relasi antara Pasien dengan Obat.



Gambar 8.6. Relasi Biner

✓ **Relasi Treenary**

Relasi yang terjadi dari hubungan3 (tiga) buah entiti. Contoh relasi ini adalah hubungan antara Pasien, Tindakan dan Dokter.



Gambar 8.7. Relasi *Treenary*

✓ **Relasi N-nary**

Relasi yang menghubungkan lebih dari 3 (tiga) buah entiti, dimasukkan dalam relasi multi entiti. Bentuk relasi semacam ini sedapat mungkin dihindari karena akan mengaburkan derajat relasi yang ada dan juga akan menyebabkan desain databasenya semakin kompleks.

8.2.5. Kardinalitas Relasi Biner

Model relasi entiti didasarkan pada persepsi dunia nyata yang terdiri dari himpunan obyek dasar yang disebut entiti dan relasi antar entiti. Entiti dapat diartikan suatu obyek yang dapat diidentifikasi secara unik, obyek dapat berupa orang, benda, peristiwa dan lainnya.

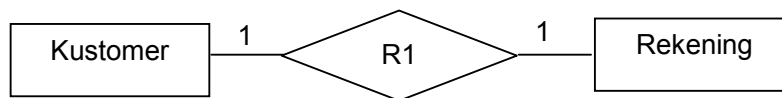
Entiti dikarakteriasi dan direpresentasikan dengan suatu gugus atribut. Contoh atribut dari Mahasiswa adalah : Nim, Nama, Alamat, Telpon, Tgl Lahir dan lainnya.

Sekelompok entiti yang mempunyai karakteristik entiti yang sama disebut Himpunan Entiti. Contoh himpunan entiti adalah, himpunan entiti Kustomer, himpunan entiti Rekening, himpunan entiti Bank, dan lainnya. Dari beberapa himpunan tadi mungkin terjadi relasi, misalkan relasi Kustomer dengan Rekening (Kustomer mempunyai rekening)

Khusus Relasi Biner, relasi antar anggota dari dua himpunan entiti yang terlibat dapat bersifat :

✓ Relasi 1-1 (one to one relationship)

Setiap entiti dari suatu himpunan entiti tertentu diasosiasikan atau direlasikan dengan tepat satu entiti dari himpunan yang lain.

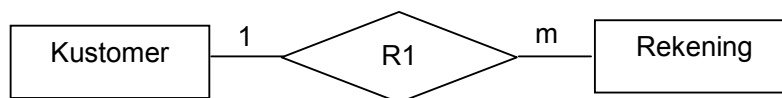


R1 = Mempunyai

Gambar 8.8. Relasi 1 - 1

✓ Relasi 1 – m (one to many relationship)

Setiap entiti dari suatu himpunan entiti direlasikan dengan satu atau lebih entiti anggota himpunan yang lain. Sebaliknya satu entiti himpunan yang lain direlasikan tepat dengan satu entiti anggota himpunan pasangannya.

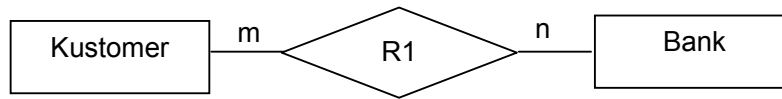


R1 = Mempunyai

Gambar 8.9. Relasi 1 – m

✓ Relasi m - n (many to many relationship)

Setiap entiti dari suatu himpunan entiti direlasikan dengan satu atau lebih entiti dari himpunan yang lain dan berlaku pula sebaliknya.



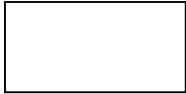

R1 = Pinjam

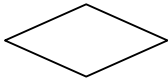

Gambar 8.10. Relasi m – n

Sifat ketiga relasi tersebut disebut sebagai kardinalitas relasi biner. Alat bantu yang banyak dipakai dalam mendiskripsikan pola hubungan (relasi) antar entiti adalah Entity Relationship Diagram atau Diagram Relasi Entiti atau lebih sering disingkat dengan ERD.

8.2. 6. Entity Relationship Diagram (ERD)

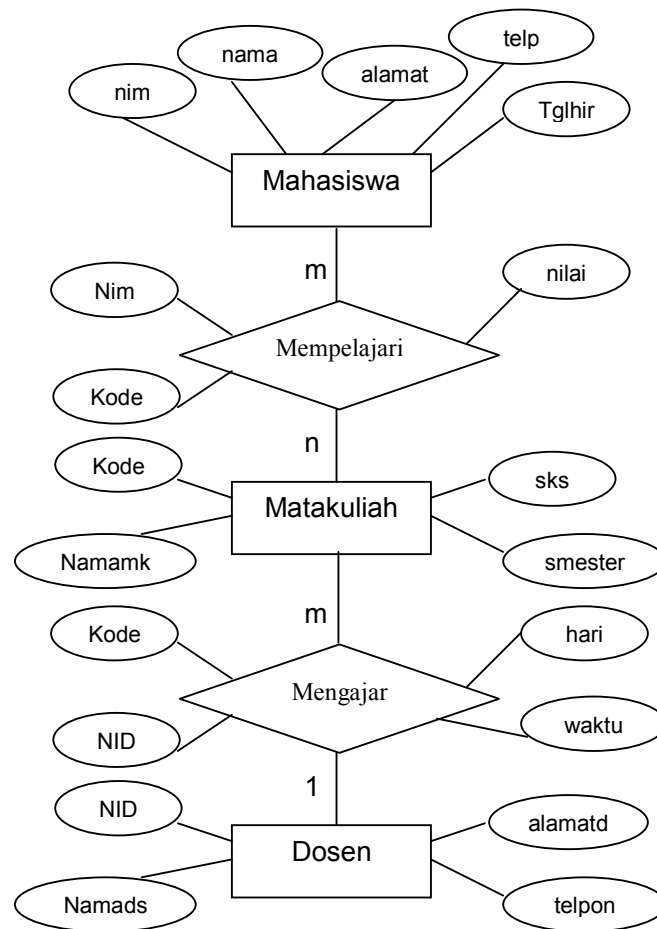
ERD berisi komponen - komponen himpunan entiti dan himpunan relasi yang masing - masing dilengkapi dengan atribut - atribut yang merepresentasikan seluruh fakta dari “ dunia nyata “.

Nomor	Simbol	Nama Simbul
1.		<u>ENTITI</u> Digunakan untuk menggambarkan obyek yang dapat diidentifikasi dalam lingkungan pemakai.
2.		<u>ATRIBUT</u> Untuk menggambarkan elemen - elemen dari suatu entiti, yang menggambarkan karakter entiti.

3.		<u>HUBUNGAN</u> Menggambarkan relasi (hubungan) antar entiti
4.		<u>GARIS</u> Digunakan untuk menghubungkan entiti dengan entiti, atau entiti dengan atribut.

Gambar 8.11. Notasi – notasi dalam ERD

Berikut ini adalah contoh penerapan model ER yang dapat dituangkan kedalam Diagram – ER (ERD) :

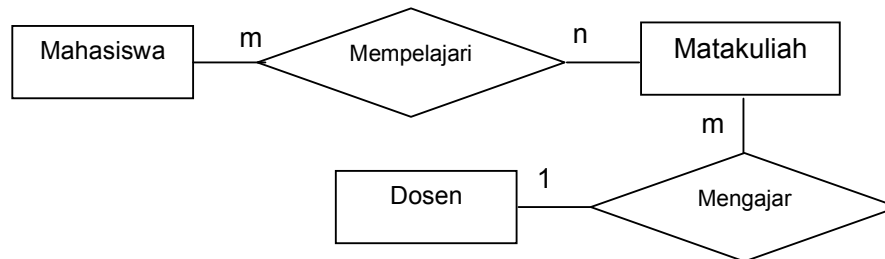


Gambar 8.12. Contoh ERD

Hubungan (relasi) antar tabel yang terjadi pada ERD tersebut diatas adalah sebagai berikut :

- Relasi Dosen dengan Matakuliah adalah one to many, dengan pengertian bahwa seorang dosen bisa mengajarkan satu atau lebih matakuliah.
- Relasi Mahasiswa dengan Matakuliah adalah many to many, dengan pengertian bahwa seorang mahasiswa bisa mempelajari satu atau lebih matakuliah, demikian juga untuk sebaliknya, bahwa satu matakuliah bisa dipelajari oleh satu atau lebih mahasiswa.

Diagram – ER (ERD / *entity relationship diagram*) tersebut diatas dapat dituangkan kedalam kamus data berikut ini :



Kamus Data :

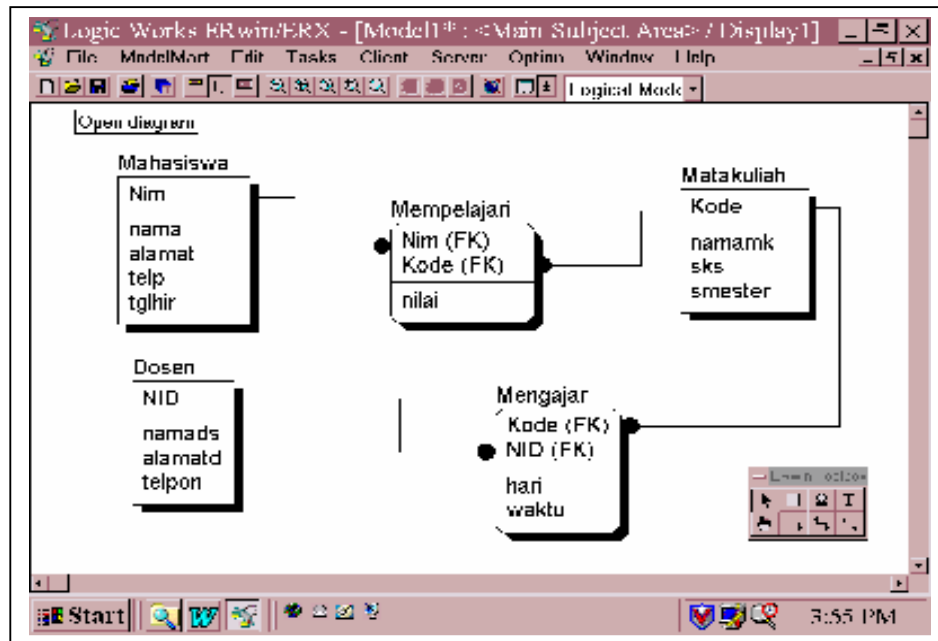
- ✓ Mahasiswa = { **Nim**, nama, alamat, telp, tglhir }
- ✓ Matakuliah = { **Kode**, namamk, sks, semester }
- ✓ Dosen = { **NID**, namads, alamatd, telpon }
- ✓ Mempelajari = { **Nim, Kode**, nilai }
- ✓ Mengajar = { **NID, Kode**, hari, waktu }

Gambar 8.13. Contoh ERD dengan kamus data

Primary key untuk masing – masing entiti tersebut diatas adalah sebagai berikut :

- Entiti mahasiswa = nim, karena pasti unik, tidak mungkin nim : mhs0001 misalkan dipakai oleh lebih dari 2 orang mahasiswa.
- Entiti matakuliah = kode, karena pasti unik, tidak mungkin kode matakuliah : mk001 misalkan dipakai oleh lebih dari 2 matakuliah.
- Entiti dosen = nid, karena pasti unik, tidak mungkin nid : ds0001 misalkan dipakai oleh lebih dari 2 orang dosen.
- Entiti mempelajari = nim dan kode (dimana kedua atribut tersebut adalah sebagai foreign key jadi dua atribut tersebut bisa dijadikan primary key pada entiti mempelajari, karena tidak ada satu atribut yang menyatakan unik pada entiti tersebut.
- Entiti mengajar = nid dan kode (dimana kedua atribut tersebut adalah sebagai foreign key jadi dua atribut tersebut bisa dijadikan primary key pada entiti mengajar, karena tidak ada satu atribut yang menyatakan unik pada entiti tersebut.

Berikut ini adalah implementasi pembentukan ERD (*entity relationship diagram*) tersebut dengan menggunakan software ERWIN (*Entity relationship for windows*) Erx.3.0



Gambar 8.14. Contoh ERD dengan ERWIN ERX.3.0