

BAB I

ARSITEKTUR SISTEM BASIS DATA

Arsitektur system basis data memberikan kerangka kerja bagi pembangunan basis data.

1.1 LEVEL ARSITEKTUR BASIS DATA

Menurut ANSI/SPARC, arsitektur basis data terbagi atas tiga level yaitu :

Internal/Physical Level:

berhubungan dengan bagaimana data disimpan secara fisik (physical storage)

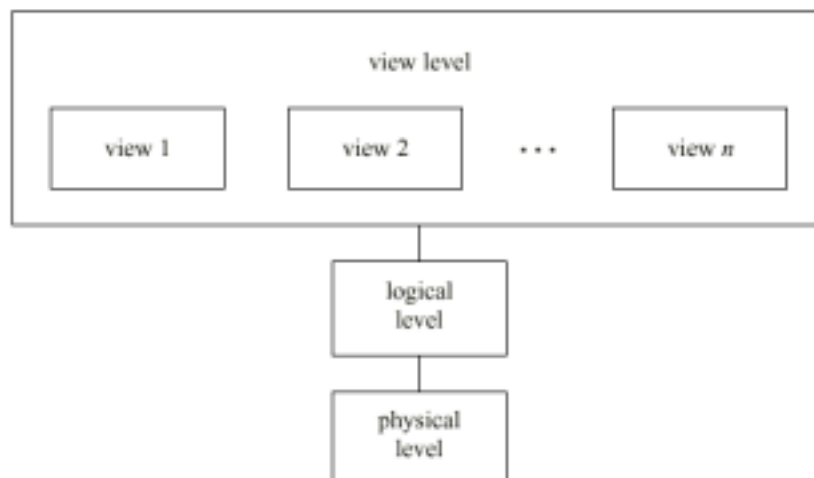
External /View Level

berhubungan dengan bagaimana data di representasikan dari sisi setiap user.

Conceptual/Logical Level

yang menghubungkan antara internal & external level

Berikut adalah gambar tiga level arsitektur system basis data :



Gambar I.1 Tiga level arsitektur system basis data

Contoh dalam sebuah basis data pegawai, ketiga level digambarkan sebagai berikut :

- Pada level **konseptual**, basis data mengandung informasi tentang sebuah entity yang disebut EMPLOYEE. Setiap EMPLOYEE mempunyai sebuah **EMP_NUMBER (6 karakter), DEP_NUMBER(4 karakter) & SALARY (5 digit desimal)**.
- Pada level **internal**, para pegawai direpresentasikan oleh **STORED_EMP, panjang 20 byte. STORED_EMP terdiri atas 4 field yaitu flag/pointer, 6 byte, tiga field untuk atribut pegawai.**
- User yang menggunakan **PL/I** mempunyai sisi pandang level **eksternal** dimana pegawai direpresentasikan dalam sebuah **record** yang ditulis dengan **sintaks PL/I**
- Demikianlah pula user yang menggunakan **COBOL** mempunyai sisi pandang level **eksternal** dimana pegawai direpresentasikan dalam **sebuah record** yang ditulis dengan **sintaks COBOL**.

1.1.1 *External Level /View Level*

Adalah level user. Yang dimaksud dengan user adalah programmer, end user atau DBA. Setiap user mempunyai 'bahasa' yang sesuai dengan kebutuhannya.

- **programmer** : bahasa yang digunakan adalah bahasa pemrograman seperti C, COBOL, atau PL/I
- **end user** : bahasa yang digunakan adalah bahasa query atau menggunakan fasilitas yang tersedia pada program aplikasi

pada level eksternal ini, user dibatasi pada kemampuan perangkat keras dan perangkat lunak yang digunakan aplikasi basis data. Didefinisikan sebagai sebuah Skema Eksternal.

1.1.2 **Conceptual Level / Logical level**

Conceptual level adalah sebuah representasi seluruh muatan informasi yang dikandung oleh basis data. Tidak seperti level eksternal maka pada level *conceptual*, keberadaannya tidak memperhitungkan kekurangan perangkat keras maupun perangkat lunak pembangun aplikasi basis data. Didefinisikan sebagai sebuah Skema Konseptual

1.1.3 **Internal/Physical Level**

Adalah level terendah untuk merepresentasikan basis data. Record disimpan dalam media penyimpanan dalam format byte. Didefinisikan sebagai sebuah Skema Internal.

1.2 PEMETAAN

1.2.1 *Conceptual/internal Mapping* :

Yaitu pemetaan yang mendefinisikan hubungan antara level konseptual dan level internal. Pemetaan ini menjelaskan bagaimana record & field pada level konseptual direpresentasikan pada level internal.

1.2.2 *External/Conceptual Mapping* :

Yaitu pemetaan yang mendefinisikan hubungan antara level eksternal dan level konseptual. Contoh : beberapa field dapat mempunyai tipe yang berbeda, field dan nama field dapat berubah, beberapa field pada level konseptual dapat dikombinasikan ke dalam satu field external.

1.3 MODEL-MODEL DATA

Yang dimaksud dengan model data adalah sekumpulan tool konseptual untuk mendeskripsikan data, relasi-relasi antar data, semantic data & konsistensi konstrain. Berbagai macam model data terbagi dalam tiga kelompok besar yaitu :

Object-based logical models, record-based logical models, & physical models.

1. *Object-based logical Models*

Yang termasuk dalam kelompok ini adalah :

- *Entity-relationship model*
- *Object-oriented model*
- *Semantic data model*
- *Functional data model*

Dua dari model di atas yaitu : *entity-relationship model & object-oriented model* akan dijelaskan lebih detail.

a. *Entity-relationship Model*

E-R model didasarkan atas persepsi terhadap dunia nyata yang terdiri dari sekumpulan objek, disebut entity dan hubungan antar objek tersebut, disebut relationship. Entity adalah objek di dunia yang bersifat unik. Setiap entity

mempunyai atribut yang membedakannya dengan entity lainnya. Contoh : entity Mahasiswa, mempunyai atribut nama, umur, alamat no. ktm.

Pemodelan data dengan model E-R menggunakan diagram E-R. Diagram E-R terdiri dari :

- **Kotak persegi panjang**, menggambarkan himpunan entity
- **Elip**, menggambarkan atribut-atribut entity
- **Diamon**, menggambarkan hubungan antara himpunan entity
- **Garis**, yang menghubungkan antar objek dalam diagram E-R

b. Object-Oriented Model

Model berorientasi objek berbasiskan kumpulan objek. Setiap objek berisi :

1. **Nilai yang disimpan dalam variable instant**, dimana variable tersebut “melekat” dengan objek itu sendiri.
2. **Metoda** : operasi yang berlaku pada objek yang bersangkutan
3. Objek-objek yang memiliki tipe nilai & metode yang sama dikelompokkan dalam satu **kelas**. Kelas disini mirip dengan tipe data abstrak pada bahasa pemrograman.
4. ***Sending a message*** : sebuah objek dapat mengakses data sebuah objek yang lain hanya dengan memanggil metode dari objek tersebut.

c. Model-model logik berbasis record

Terdapat beberapa model dalam kelompok ini :

c.1. Model Relasional (*Relational Model*)

model relasional menggunakan kumpulan tabel-tabel untuk merepresentasikan data & relasi antar data-data tersebut. Setiap tabel terdiri atas kolom-kolom, dan setiap kolom mempunyai nama yang unik.

Contoh :

NIM	NAMA	ALAMAT	DOSEN WALI
113940005	Ani Ambar	Jl. PGA No.1	WNS
113950098	Andi Ari	Jl. ABC No.6	TBH
113950100	Tuti	Jl. AA No.78	WNS

DOSEN WALI	ALAMAT
WNS	KOPO PERMAI
TBH	BALE ENDAH

c.2 Model Jaringan (*Network Model*)

Data dalam model jaringan direpresentasikan dengan sekumpulan record (Pascal), dan relasi antara data direpresentasikan oleh record & link. Link dipandang sebagai pointer. Record-record diorganisasikan sebagai graf. Contoh, model relasional di atas direpresentasikan dalam model jaringan.



c.3 Model Hirarki (*Hierarchical Model*)

Mirip dengan model jaringan. Data direpresentasikan dalam record dan link. Perbedaannya adalah, record-record diorganisasikan sebagai *tree* (pohon) daripada graf. Model relasional berbeda dengan model jaringan & hirarki, dalam hal penggunaan pointer atau link.

1.4 BAHASA YANG DISEDIAKAN OLEH SISTEM BASIS DATA

Sebuah system basis data menyediakan dua tipe bahasa yaitu : tipe untuk menspesifikasikan skema basis data, disebut DDL, dan tipe untuk mengekspresikan queri atau update basis data, disebut MDL.

1.4.1 *Data-definition Language*

Skema basis data dispesifikasikan oleh sekumpulan definisi dengan sebuah bahasa khusus yang disebut data-definition language (DDL). Hasil kompilasi DDL berupa tabel-tabel yang disimpan dalam sebuah file, disebut data dictionary (kamus data) atau data directory. Kamus data adalah sebuah file yang berisi metadata. File ini yang dikonsultasi sebelum data yang sebenarnya dibaca atau dimodifikasi oleh system basis data.

1.4.2 *Data-manipulation Language*

Adalah bahasa untuk memanipulasi data yaitu :

- Pengambilan informasi yang disimpan dalam basis data
- Penyisipan informasi baru ke basis data
- Penghapusan informasi dari basis data
- Modifikasi informasi yang disimpan dalam basis data

Query adalah statemen yang ditulis untuk mengambil informasi. Bagian dari DML yang menangani pengambilan informasi ini disebut bahasa query.

1.5 DATABASE ADMINISTRATOR (ADMINISTRATOR BASIS DATA)

Administrator basis data adalah orang yang bertanggungjawab terhadap strategi yang berhubungan dengan pengolahan data di suatu perusahaan. Selain hal tersebut, DBA adalah orang yang bertanggungjawab terhadap implementasi strategi tersebut secara teknis. Berikut dijelaskan bebrapa fungsi DBA secara detil :

- **mendefinisikan skema konseptual**

yaitu memutuskan informasi apa saja yang akan dihasilkan oleh satu basis data. Dalam hal ini DBA bertugas mendefinisikan entity-entiti yang terlibat pada perancangan basis data. Proses ini disebut dengan Logical Database Design (kadang-kadang disebut Conceptual Database Design). Setelah DBA menentukan muatan dari basis data maka DBA akan menyusun skema konseptual menggunakan DDL (data definition language). Kemudian dengan DBMS, skema tersebut diimplementasikan.

- **mendefinisikan Skema Internal**

selain menentukan skema konseptual dari basis data, DBA juga bertanggung jawab terhadap desain penyimpanan data secara fisik.

- **mensosialisasikan hasil rancangan baik konseptual & internal ke user**

Dalam hal ini tugas DBA adalah menyakinkan user bahwa data-data yang mereka perlukan semua telah tersedia & membantu user untuk mengimplementasikan rancangan tersebut dalam DDL.

- **mendefinisikan aturan-aturan sekuriti & integritas data**

- **mendefinisikan prosedur-prosedur *back-up & recovery***

- **memonitor performansi system & menangani perubahan-perubahan yang ada.**

1.6 DATABASE MANAGEMENT SYSTEM (SISTEM MANAJEMEN BASIS DATA)

DBMS adalah software yang menangani semua akses ke basis data. Secara konsep apa yang terjadi adalah sebagai berikut :

1. user melakukan pengaksesan basis data untuk informasi yang diperlukannya menggunakan suatu bahasa manipulasi data, biasanya disebut SQL.
2. DBMS menerima *request* dari user & menganalisa request tersebut
3. DBMS memeriksa skema eksternal user, pemetaan eksternal/konseptual, skema konseptual, pemetaan konseptual/internal, & struktur penyimpanan.
4. DBMS mengeksekusi operasi-operasi yang diperlukan untuk memnuhi permintaan user.