

## **BAB X**

### **ARSITEKTUR DATABASE**

#### **A. Abtraksi Data**

Kegunaan utama sistem database adalah agar pemakai/user mampu menyusun suatu pandangan abtraksi dari data. Bayangan mengenai data tidak lagi memperhatikan kondisi sesungguhnya bagaimana satu data masuk ke database, disimpan dalam disk disektor mana, tetapi menyangkut secara menyeluruh bagaimana data tersebut dapat di abtraksikan/digambarkan menyerupai kondisi yang dihadapi oleh pamakai.

Abstraksi data adalah bagaimana melihat data dalam sebuah sistem basis data. Salah satu tujuan dari DBMS adalah untuk menyediakan fasilitas dalam memperlihatkan data kepada pemakai/user. Untuk itu sistem akan menyembunyikan detail tentang bagaimana data itu disimpan dan dipelihara. Karena itu seringkali data yang terlihat oleh pemakai berbeda dengan yang tersimpan secara fisik.

#### **B. Bahasa Basis Data (*Database Language*)**

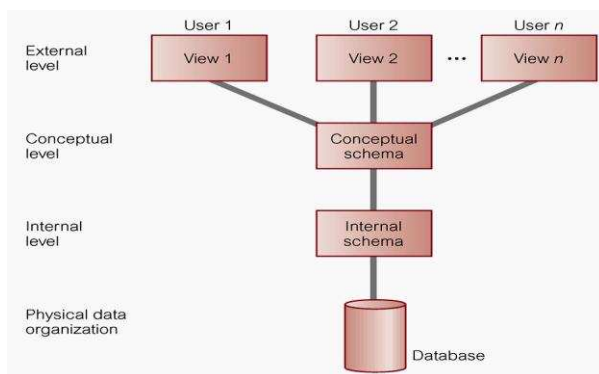
DBMS merupakan perantara bagi pemakai dengan basis data dalam disk, Cara berinteraksi/berkomunikasi tersebut menggunakan suatu bahasa khusus yang disebut **bahasa basis data** yang terdiri dari sejumlah perintah/statement.

Bahasa basis data dapat dipilah menjadi dua macam perintah yang digunakan untuk mengelola dan mengorganisasikan data, yaitu :

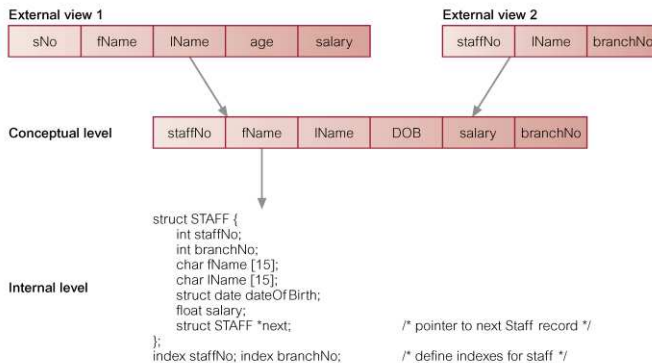
1. Data Definition Language (DDL)
2. Memberikan kemudahan kepada DBA atau USER untuk mendeskripsikan entitas, atribut dan relasi, yang dibutuhkan untuk aplikasi database. Hasil dari perintah DDL adalah kumpulan table yang disimpan dalam file khusus yang disebut Kamus Data(Data Dictionary)
3. Data Manipulation Language (DML)  
Berguna untuk melakukan manipulasi dan pengambilan data, yaitu berupa:
  - a. Penyisipan /penambahan data baru ke suatu basis data
  - b. Penghapusan data dari suatu basis data
  - c. Pengubahan data disuatu basis data

### C. Arsitektur Database

1. Menurut ANSI-SPARC



- a. Level Eksternal/View Level
  - Pandangan user terhadap database
  - Menggambarkan bagian dari database yang relevan untuk pemakai tertentu
- b. Level Konseptual/Logical Level
  - Merupakan pandangan komunitas terhadap database
  - Menggambarkan data apa yang disimpan dalam database dan bagaimana hubungan diantara data.
  - Level ini juga berisi struktur logic dari keseluruhan database.
- c. Level Internal/Physical Level
  - Merupakan gambaran phisik database pada computer
  - Menggambarkan bagaimana data tersimpan ke dalam database



#### D. Data Independence

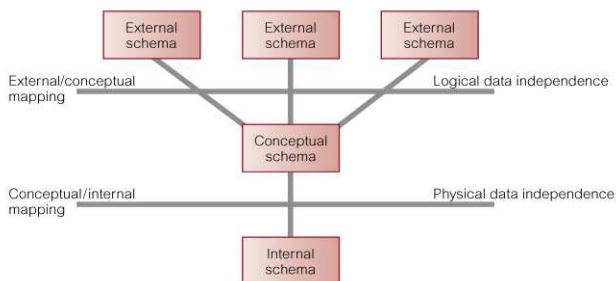
Tujuan utama dari arsitektur DBMS adalah untuk menyediakan data independence, dimana bahwa level yang lebih tinggi tidak dipengaruhi oleh pertukaran ke level yang lebih rendah.

## 1. Logical Data Independence

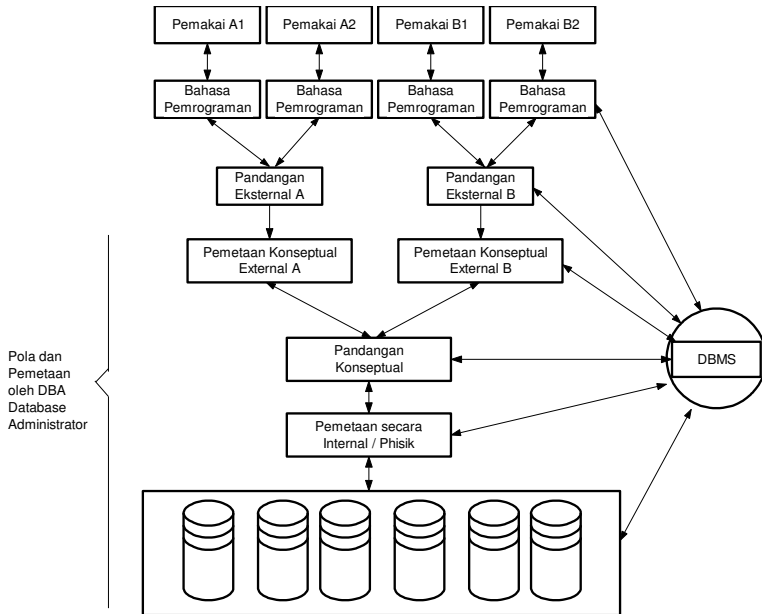
Menunjukkan kekebalan skema external untuk bertukar ke skema konseptual. Bertukarnya ke skema konseptual seperti penambahan atau penghapusan entity, atribut dan relation baru tanpa harus menukar skema external yang ada atau harus ditulisnya kembali program aplikasi

## 2. Physical Data Independence

Menunjukkan kekebalan skema konseptual untuk bertukarnya skema internal, Misalnya penggunaan organisasi file atau struktur penyimpanan yang berbeda penggunaan peralatan penyimpanan yang berbeda, pemodifikasian algoritma indek atau hashing tanpa harus menukar skeme konseptual (eksternal)



### Gambar Hirarki Data Indefence



Gambar Model Database Manajemen Sistem

Ada dua jenis DML

- Prosedural
- Nonprosedural

Ada dua cara untuk mengakses data pada basis data :

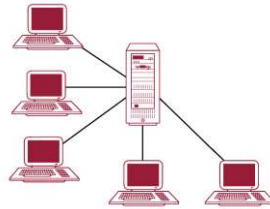
- Mengetikkan perintah-perintah yang ditujukan kepada DBMS untuk memanipulasi suatu data. Biasanya DML yang digunakan adalah bersifat nonprosedural.
- Melalui program aplikasi yang menerbitkan instruksi-instruksi internal ke DBMS untuk memanipulasi data dan memberikan hasil ke

program. Bahasa pemrograman yang digunakan dapat berupa bahasa pemrograman konvensional Pascal, C, Cobol, Fortran dll, yang menggunakan pendekatan prosedural.

## E. Arsitektur Database Multiuser

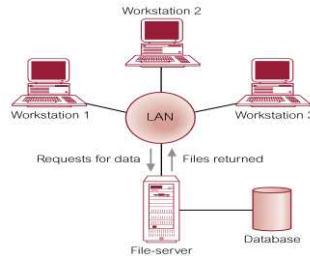
### 1. TELEPROCESSING

- Traditional architecture.
- Single mainframe with a number of terminals attached.
- Trend is now towards downsizing.



### 2. FILE-SERVER

- File-server is connected to several workstations across a network.
- Database resides on file-server.
- DBMS and applications run on each workstation.

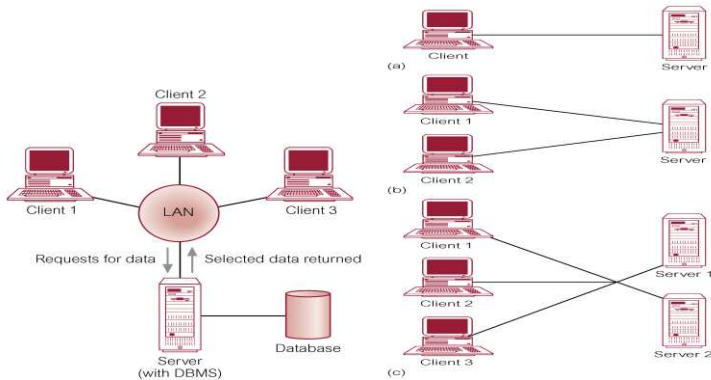


#### Kelemahan File-Server

- Significant network traffic.
- Setiap workstation harus memiliki copy DBMS
- Concurrency, recovery and integrity control more complex.

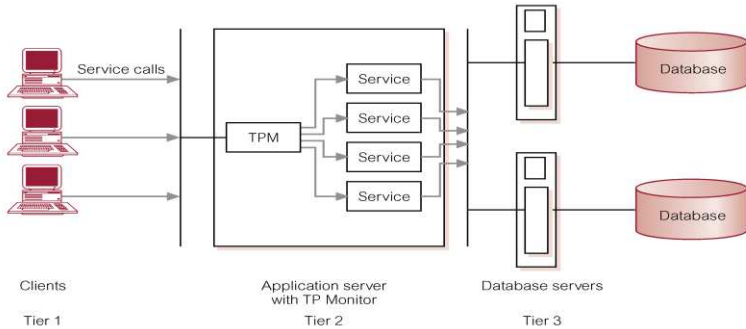
### 3. CLIENT-SERVER

- a. Database dan DBMS terletak pada Server
- b. Client mengatur sendiri hubungan dengan database, DBMS dan menjalankannya



#### Transaction Processing Monitors

- a. Merupakan program yang mengontrol pertransferan data antara Clients dan Servers saat terjadinya transaksi dan merupakan bagian dari OLTP (Online Transaction Processing)
- b. Transaction Processing Monitor merupakan middle tier dari arsitektur three-tier client-server



Gambar Arsitektur Database Multiuser

**Materi Latihan :**

1. Jelaskan kegunaan dari abstraksi data
2. Apa yang anda ketahui tentang bahasa basis data
3. Jelaskan cara mengakses data pada basis data



## **BAB XI**

### **MODEL DATA**

Model basis data menunjukkan suatu cara/ mekanisme yang digunakan untuk mengelola/ mengorganisasikan data secara fisik dalam memori sekunder yang akan berdampak pada bagaimana kita mengelompokkan dan membentuk keseluruhan data yang terkait dalam sistem yang sedang kita tinjau. Suatu model adalah suatu representasi terhadap objek yang ada “real world” dan kejadian serta hubungan diantara mereka. Hingga saat ini, model basis data yang paling umum ada 3 macam, yaitu : hirarkis, jaringan dan relational.

#### **A. Model Data Hirarkis**

Model hirarkis biasa disebut model pohon, karena menyerupai pohon yang dibalik. Model ini menggunakan pola hubungan orang tua-anak. Setiap simpul (biasanya dinyatakan dengan lingkaran atau kotak) menyatakan sekumpulan medan. Simpul yang terhubung ke simpul pada level dibawahnya disebut orang tua sebaliknya simpul yang berada dibawah orang tua disebut anak. Setiap orang tua dapat memiliki satu (hubungan 1:1) atau beberapa anak (hubungan 1:M), tetapi setiap anak hanya memiliki satu orang tua. Contoh gambar yang menunjukkan model hirarkis :