

DATABASE SYSTEMS

(Sistem Basis Data)

Pertemuan 2

Politeknik Piksi Ganesha
Esson Sihombing, S.Kom.
2012

Database Management System (DBMS)



- DBMS contains information about a particular enterprise
 - Collection of interrelated data
 - Set of programs to access the data
 - An environment that is both *convenient* and *efficient* to use
- Database Applications:
 - Banking: all transactions
 - Airlines: reservations, schedules
 - Universities: registration, grades
 - Sales: customers, products, purchases
 - Online retailers: order tracking, customized recommendations
 - Manufacturing: production, inventory, orders, supply chain
 - Human resources: employee records, salaries, tax deductions

DBMS



- Definisi:

Merupakan sistem yang terdiri atas kumpulan file (tabel) yang saling berhubungan (dalam sebuah basis data di sebuah sistem komputer) dan sekumpulan program lain untuk mengakses dan memanipulasi file (tabel) tersebut.

- **DBMS** adalah paket perangkat lunak yang didesain untuk melakukan penyimpanan dan pengaturan basis data.

Manfaat DBMS



- untuk mengorganisasi dan mengelola data dalam jumlah besar
- untuk membantu dalam melindungi data dari kerusakan yang disebabkan pengaksesan yang tidak sah
- memudahkan dalam pengambilan data
- untuk memudahkan dalam pengaksesan data secara bersamaan dalam suatu jaringan

Komponen Sistem Basis data



- Komponen:

- Hardware

- Sistem Operasi

- Basis Data

- Software DBMS :

MySQL, Ms. SQL Server, Interbase, Paradox, Ms. Access,
Oracle, DB2

- User

- Aplikasi Lain

Komponen DBMS



- ❑ Data: yang disimpan dalam basis data
- ❑ Operasi standart: untuk memanipulasi data
 - DDL: untuk mendeskripsikan nama” atribut, tipe data atau struktur basis data
 - DML : bahasa query
- ❑ Bahasa pemrograman
- ❑ Struktur files: untuk mengorganisasikan data

Komponen Fungsional Sistem Basis data

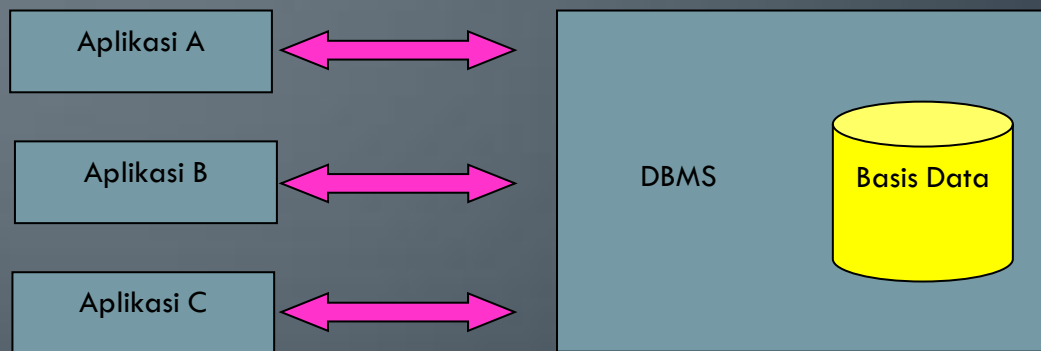
- File manager: mengelola alokasi kebutuhan ruang penyimpanan (storage) basisdata beserta struktur-struktur yang digunakan untuk merepresentasikan informasi yang disimpan di dalam disk.
- Database Manager: menyediakan interface antara data low-level yang disimpan didalam basisdata dengan program-program aplikasi dan queries yang dikirimkan ke system.
- Query processor: menterjemahkan pernyataan-pernyataan bahasa query ke dalam instruksi-instruksi low-level yang dimengerti oleh database manager.
- DML precompiler : mengkonversi pernyataan-pernyataan DML yang dimasukkan di dalam program aplikasi ke dalam pemanggilan prosedur normal di dalam bahasa induknya. Procompiler harus berinteraksi dengan query processor untuk membuat kode-kode yang diperlukan.
- DDL compiler : mengkonversi pernyataan DDL ke dalam sekumpulan table yang mengandung metadata atau “data mengenai data”

Hubungan DBMS dan Aplikasi Basis Data

1. DBMS dan Aplikasi Basis Data terpisah

Biasanya digunakan pada system DBMS yang besar dan digunakan untuk multi user. Pada system DBMS ini mempunyai banyak feature/ sarana yang bisa digunakan untuk user. Misalnya: menyediakan sarana untuk perancangan basis data.

Contoh : MS-SQL Server, Oracle, Sybase, Informix, dsb.



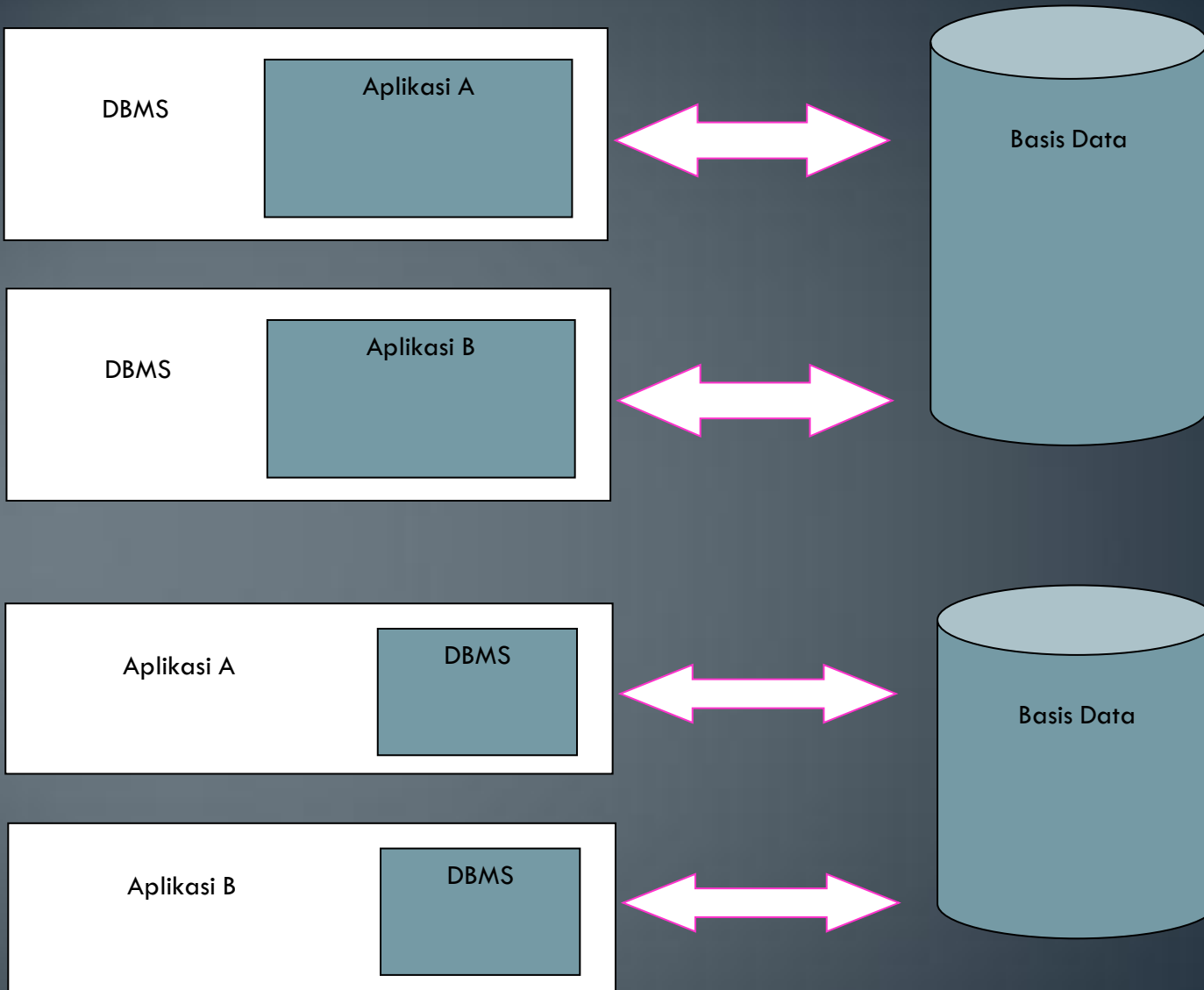
Hubungan DBMS dan Aplikasi Basis Data



2. DBMS dan Aplikasi Basis Data menyatu

Biasanya digunakan pada system basis data untuk perorangan. Feature/ Sarana tidak begitu banyak dibandingkan pada system dimana DBMS dan Aplikasi Basis Data terpisah.

- Aplikasi BD menjadi subordinate DBMS
Contoh : dBase III+, Foxbase, Ms Access
- DBMS menjadi subordinate Aplikasi BD
Contoh : Clipper, Delphi



Klasifikasi DBMS



- Model Data yang disupport:
Terdiri dari Model Data Relasional, Hirarki, Jaringan, Object Oriented
- Jumlah user yang bisa ditangani oleh DBMS:
User tunggal dan Multi User
- Jumlah sites (Lokasi):
Centralized, Distributed, Client-Server
- Biaya/harga DBMS

Level Abstraksi Data



- ❑ Abstraksi data merupakan arsitektur/tingkatan/level untuk melihat data dalam sebuah sistem basis data.
- ❑ Karena tidak semua pengguna basisdata terlatih dengan baik dan penggunaanya terbagi dalam beberapa tingkatan, maka kompleksitas basisdata akan tersembunyi dari para penggunaanya melalui beberapa level abstraksi data.
- ❑ Fungsinya untuk menyederhanakan interaksi antara pemngguna dengan sistemnya dengan basisdata dapat mempresentasikan view yang berbeda kepada para pengguna, programmer dan administratornya

Level Abstraksi Data (2)

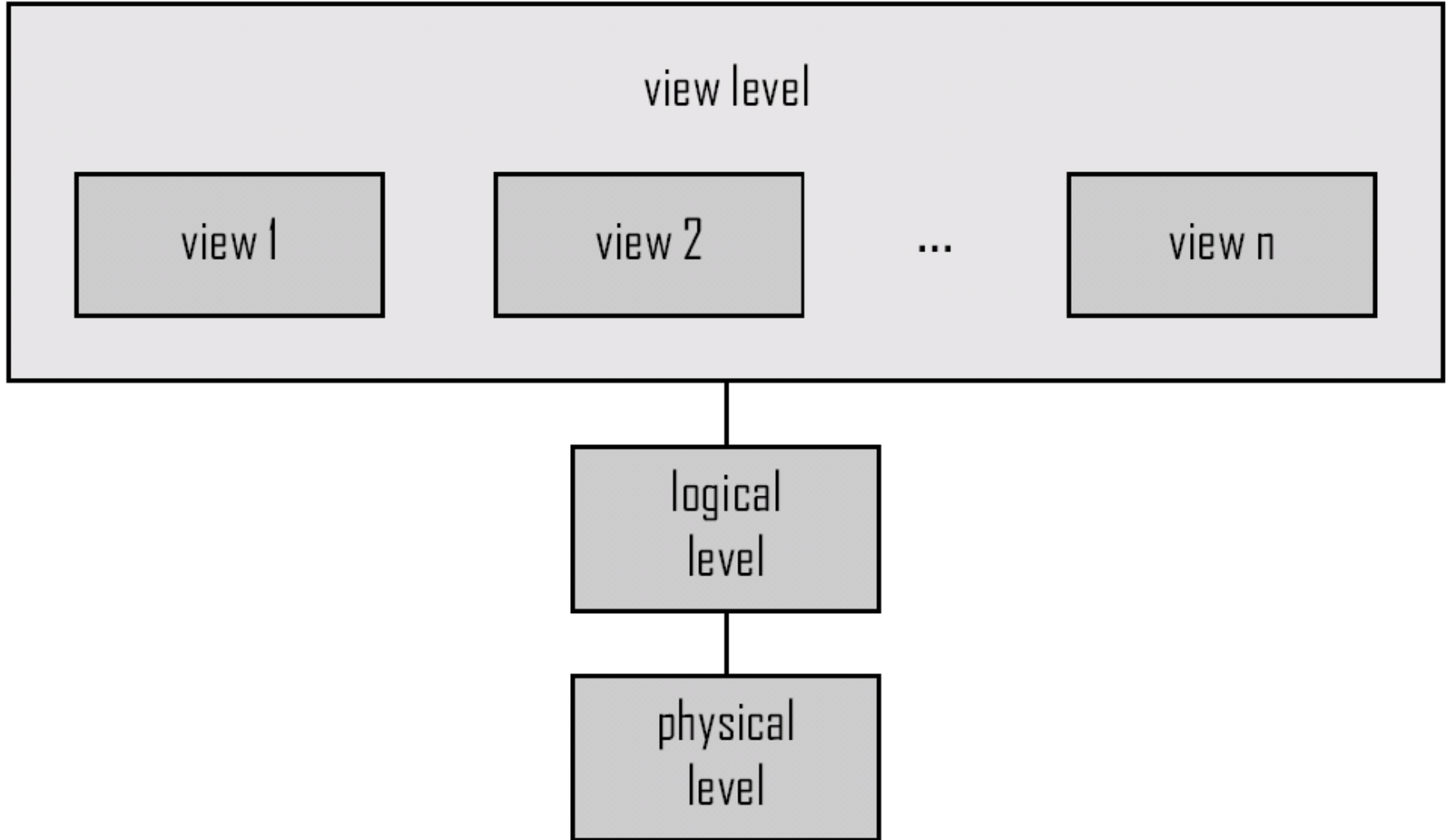
- **Physical level:** describes how a record (e.g., customer) is stored.
- **Logical level:** describes data stored in database, and the relationships among the data.

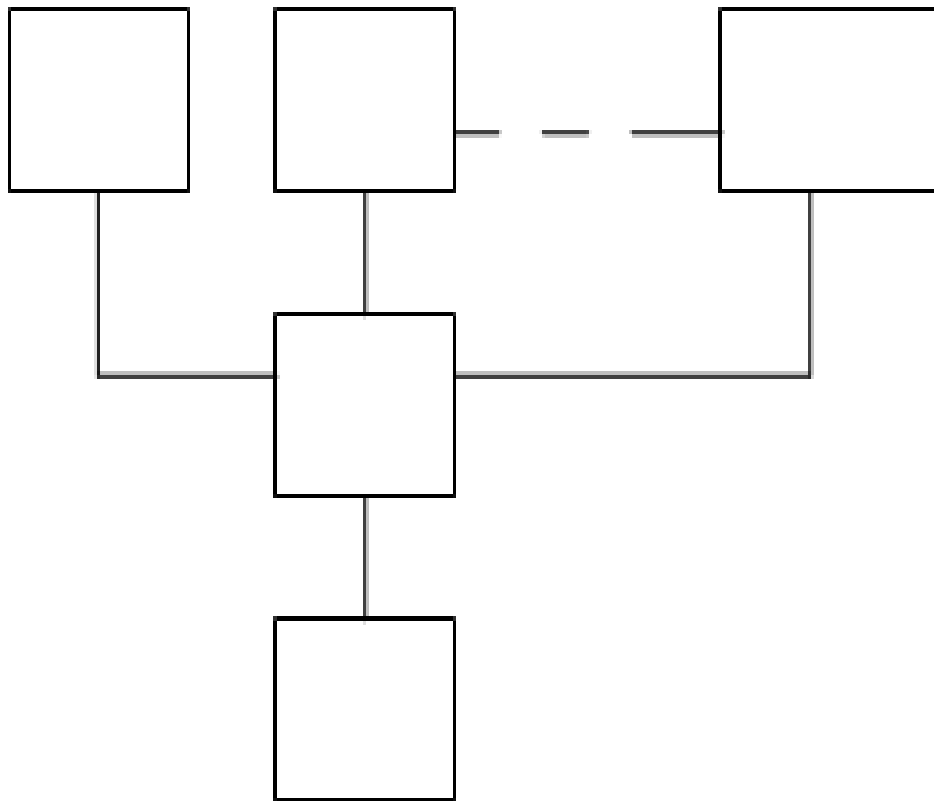
```
type customer = record
```

```
    customer_id : string; customer_name : string;  
    customer_street : string; customer_city : integer;  
end;
```

- **View level:** application programs hide details of data types. Views can also hide information (such as an employee's salary) for security purposes.

Arsitektur Database System





External Level
(Individual User View)

Conceptual Level
(Community Users View)

Internal Level
(Storage View)

- View Level – External Level
- Logical Level – Conceptual Level
- Physical Level – Internal Level (Storage)

Level Fisik/Internal



- Level fisik merupakan level paling rendah yang menggambarkan bagaimana sesungguhnya suatu data disimpan, yang pada umumnya tidak terlihat oleh pengguna atau programmer aplikasinya
- Level yang terdekat dengan penyimpanan fisik, yaitu bagaimana pengorganisasian data yang tersimpan dalam peralatan penyimpan sekunder (harddisk).
- Menjelaskan struktur penyimpanan fisik dari basis data. Model yang digunakan adalah model data fisik yaitu (berisi struktur penyimpanan atau bahkan notasi bit-bit (0 dan 1).
- Pada level ini representasi dari organisasi data disimpan sebagai teks, angka, atau himpunan bit data. Mengetahui bahwa data disimpan dalam beberapa file/tabel, seperti file pegawai, file keluarga, dll

Level Konseptual/Logika

- Menggambarkan data apa yang sebenarnya disimpan dalam basis data dan hubungannya/relasi yang terjadi antara data dengan data lain. Model yang digunakan adalah Model data tingkat tinggi (bisa langsung dipahami manusia).
- Pemakai tidak perlu memperdulikan kerumitan dalam struktur level fisik lagi, penggambaran cukup dengan memakai kotak, garis, dan hubungan secukupnya.
- Dimana administrator basis data (DBA) membangun dan mengolah basis data, contohnya: pengguna akan mengetahui bahwa data penjualan disimpan didalam tabel-tabel barang, produksi, keuangan, marketing

Level User View/Eksternal



- Level abstraksi data tertinggi yang hanya menunjukkan sebagian saja yang dilihat dan dipakai dari keseluruhan database, sesuai dengan kebutuhan pemakai.
- Menjelaskan sebagian basis data yang menjadi “interest” dari sekelompok user tertentu. Model yang digunakan adalah Model data tingkat tinggi (bisa langsung dipahami manusia)
- Level yang terdekat dengan pengguna, dalam arti bagaimana cara dapat disajikan/dipandang oleh pengguna.

Implementasi 3 Level

EXTERNAL (PL/I)

DCL 1 EMPP,
2 EMP# CHAR(6)
2 SAL FIXED BIN(31) ;

CONCEPTUAL

EMPLOYEE

EMPLOYEE_NUMBER
DEPARTMENT_NUMBER
SALARY

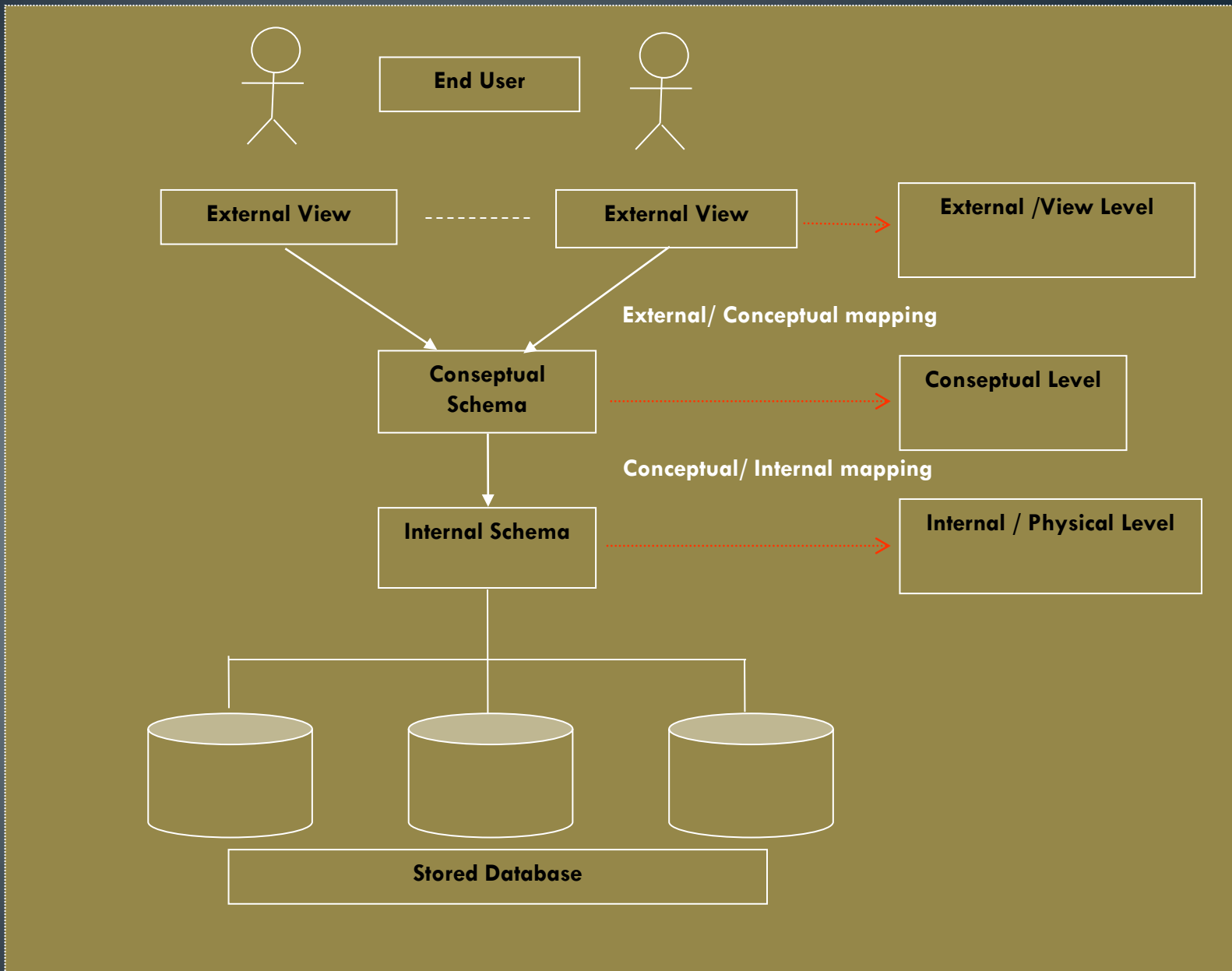
EXTERNAL (COBOL)

01 EMPC
02 EMPNO PIC X(6).
02 DEPTNO PIC X(4).

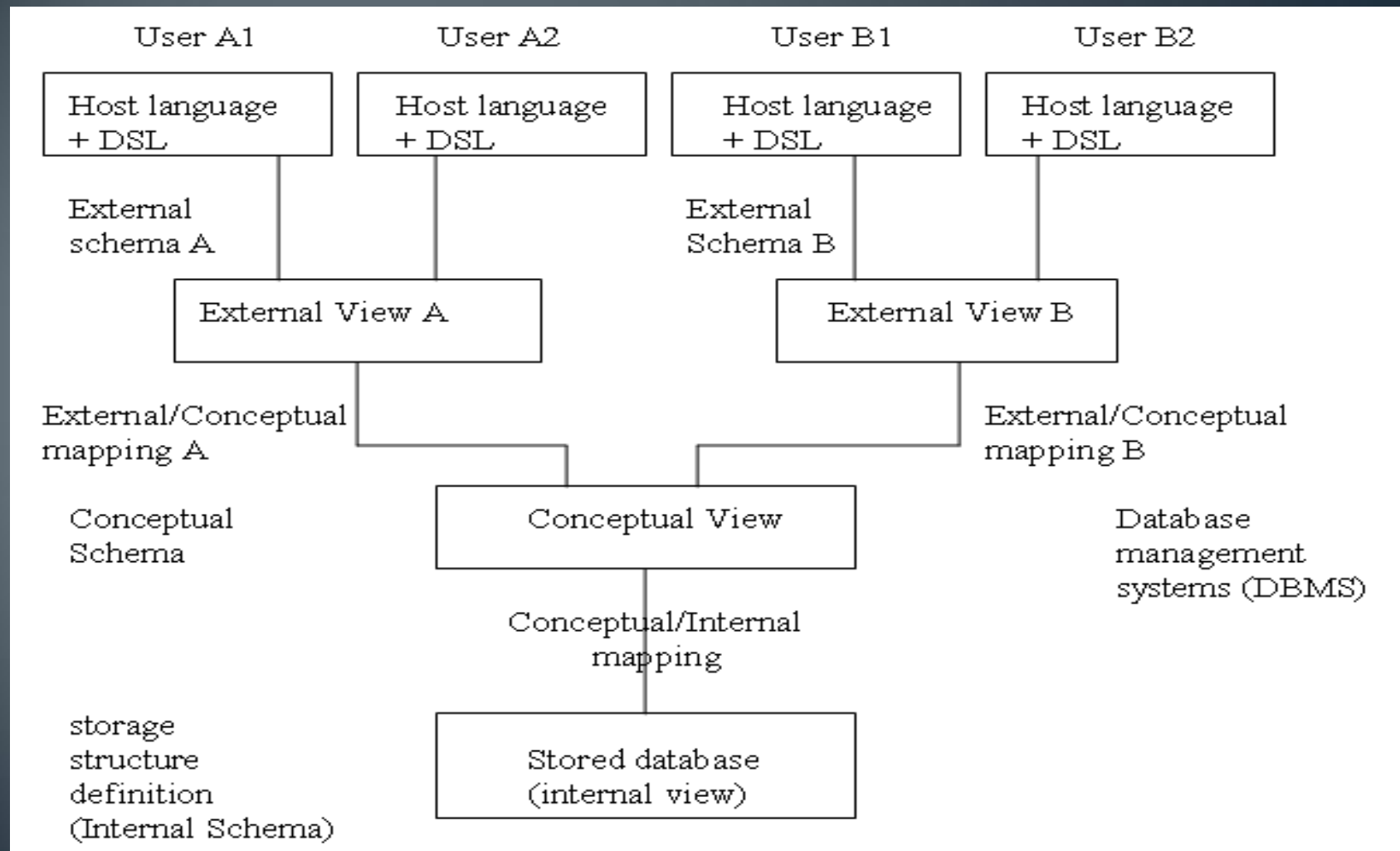
INTERNAL

STORED_EMP
PREFIX
EMP#
DEPT#
PAY

LENGTH=18
TYPE=BYTE (6), OFFSET=0
TYPE=BYTE (6), INDEX=EMPX
TYPE=BYTE (4), OFFSET=12
TYPE=FULLWORD, OFFSET=16

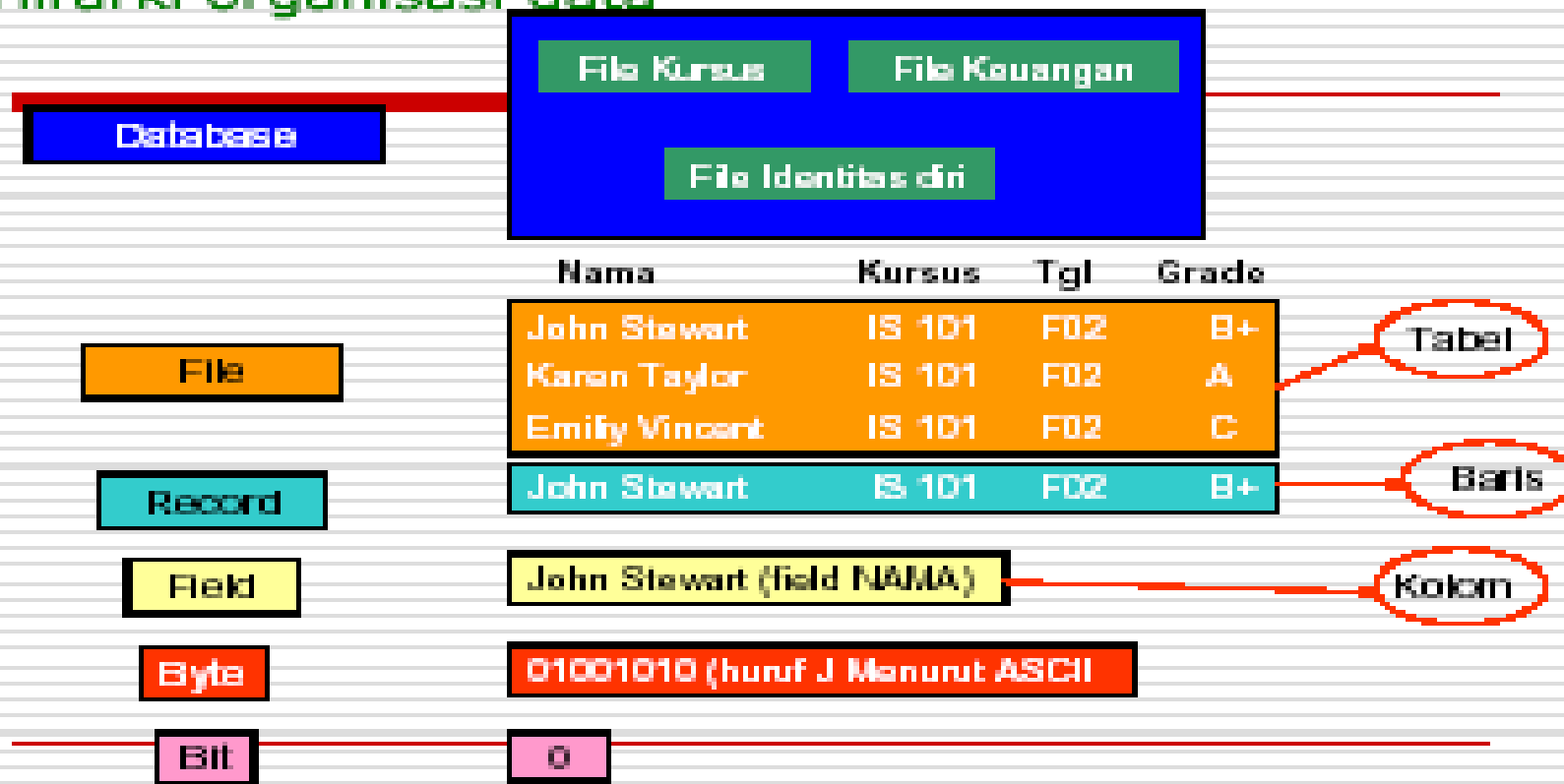


Detail Arsitektur Database



Hirarki Organisasi Data

Hirarki organisasi data



Tipe File



□ File Induk (*master File*)

1. file induk acuan (*reference master file*): file induk yang recordnya relatif statis, jarang berubah nilainya. Misalnya file daftar gaji, file mata pelajaran.
2. file induk dinamik (*dynamic master file*): file induk yang nilai dari record-recordnya sering berubah atau sering dimutakhirkan (*update*) sebagai hasil dari suatu transaksi. Misalnya file induk data barang, yang setiap saat harus di *up-date* bila terjadi transaksi.

□ File Transaksi (*transaction file*)

File ini bisa disebut *file input*, digunakan untuk merekam data hasil dari transaksi yang terjadi. Misalnya file penjualan yang berisi data hasil transaksi penjualan.

Tipe File (2)



❑ File Laporan (*Report file*)

File ini bisa disebut *output file*, yaitu file yang berisi informasi yang akan ditampilkan.

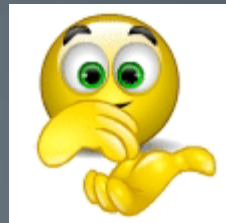
❑ File Sejarah (*history file*)

File ini bisa disebut file arsip (*archival file*), merupakan file yang berisi data masa lalu yang sudah tidak aktif lagi, tetapi masih disimpan sebagai arsip.

❑ File Pelindung (*backup file*)

File ini merupakan salinan dari file-file yang masih aktif di dalam database pada suatu saat tertentu. File ini digunakan sebagai pelindung atau cadangan bila file database yang aktif mengalami kerusakan atau hilang.

GAMSAHAPNIDA



See U Next Week

