Database System Architecture

External Level (Individual user view)

Level (Community user view)

Internal Level (Storage view)
### External (PL/I)
```
DCL 1 EMPP,
     2 EMP# CHAR(6)
     2 SAL FIXED BIN(31)
```

### External (COBOL)
```
01 EMPC,
   02 EMPNO PIC(6)
   02 DEPTNO PIC X(31)
```

### Conceptual
```
Employee

   Employee_number CHAR(6)
   Department_number CHAR(4)
   Salary NUMERIC(5)
```

### Internal
```
STORED_EMP BYTES = 20
PREFIX TYPE=BYTES(6), OFFSET = 0
EMP# TYPE=BYTES(6), OFFSET = 6, INDEX = EMPX
DEPT# TYPE=BYTES(4), OFFSET = 12
PAY TYPE=FULLWORD, OFFSET = 16
```
External Level

- **External view:**
  - Describes abstract representation of some portion of the total database seen by some particular user and hides the rest of the database from that user
  - It is defined by an external schema
  - Schema refers to how data is organized and relations among database storage structures (e.g. tables)

- Types of users
  - Application programmers
  - End users
  - DBAs

- Data Sub-languages (DSL)
  - DSL is embedded within host language
Conceptual Level & Internal Level

• **Conceptual view**
  – Describes abstract representation of the entire database for a community of users and hides physical storage from users
  – It is defined by a conceptual schema
  – A view of the data “as it really is”

• Library analogy
  – Catalog from computers -> Conceptual view
  – Books on shelf -> Physical storage
  – Patrons

• **Internal view** is the low-level representation of the entire database as physically stored
Mappings

• **Conceptual ↔ internal mapping**
  – Correspondence between the conceptual view and the stored database
  – Specify how the conceptual view is represented at the internal level

• **External ↔ conceptual mapping**
  – Correspondence between the external view and conceptual view
Data Independence

• **Definition**
  – The capacity to change the schema at one level of a database system without having to change the schema at the next higher level

• Logic data independence
  – The change to the conceptual schema does not need to make change to the internal schema or the stored database

• Physical data independence
  – The change to the internal schema does not need to make change to the conceptual schema

• Analogous to information hiding in OOP
Database Languages

- Languages
  - DDL (Data Definition Language)
    Definition or declaration of database objects such as creation of tables
  - DML (Data Manipulation Language)
    Manipulation or processing of such objects such as retrieval, updates, deletion of data

- Data sub-languages (DSL) and host languages
  - DSL: SQL (Structured Query Language)
  - Host Language: C/C++, Java, PHP, other 5GLs
Database Administrator (DBA)

- Define the conceptual schema (logical design)
- Define the internal schema (physical DB design)
- Liaise with users including user account management
- Define security and integrity constraints
- Define backup and recovery procedures
- Monitor performance and responding to changing requirements
Database Management System (DBMS)

- **DBMS** is the software that handle all access to database

**Processes**
- User request
- DBMS intercepts and analyze
- DBMS inspects different levels and mappings between the levels
- DBMS executes on the stored database
- Results return
Database Management System (DBMS)

• Functions DBMS supports
  – Data definition
  – Data manipulation
  – Optimization
  – Data security and integrity
  – Data recovery and concurrency
  – Data dictionary: ‘data about the data’
  – Performance
Client/Server Architecture

- **Server:** DBMS
- **Clients:** applications that run on top of the DBMS
  - User-written applications
  - Vendor-provided applications (tools)
    - Query language processor (SQL)
    - Report writers
    - Statistical packages
    - more ...
Client/Server Architecture

Communication Networks

Server Machine
Client/Server Architecture