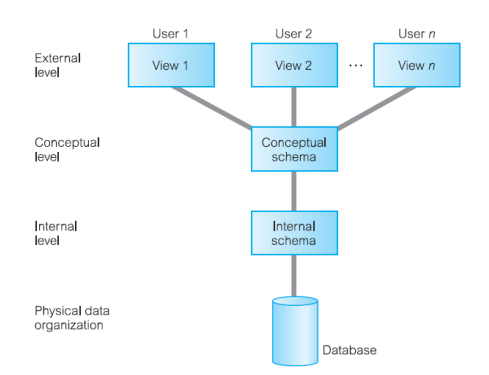
Topic 2: The 3- Level Database Architecture

-architecture of most commercial DBMSs available today is based to some extent on the so-called ANSI-SPARC architecture.

**The 3-level DB Architecture are:-**

1. **External**: The way users perceive data.
2. **Conceptual:** provides both mapping and independence between external and internal levels.
3. **Internal:** The way DBMS and OS perceive data where data is stored using data structures and file.

****

**Reason behind 3-level architecture:   
Main:  
-**separate each other’s view of db from the way db is physically presented.

Other:

1. Each user should be able to access the same data, but have a different customized view of the data.
2. a user’s interaction with the database should be independent of storage considerations.
3. The Database Administrator (DBA) should be able to change the database storage structures without affecting the users’ views.
4. The internal structure of the database should be unaffected by changes to the physical aspects of storage, such as the changeover to a new storage device.
5. The DBA should be able to change the conceptual structure of the database without affecting all users.

**External Level**

The users’ view of the database. This level describes that part of the database that is relevant to each user.   
• The external level consists of a number of different external views of the database.   
• Each user has a view of the ‘real world’ represented in a form that is familiar for that user.   
• The external view includes only those entities, attributes, and relationships in the ‘real world’ that the user is interested in.  
• Other entities, attributes, or relationships that are not of interest may be represented in the database, but the user will be unaware of them.   
• In addition, different views may have different representations of the same data.   
• For example, one user may view dates in the form (day, month, year), while another may view dates as (year, month, day).   
• Some views might include derived or calculated data: data not actually stored in the database as such, but created when needed.

Conceptual Level

• The community view of the database. This level describes what data is stored in the database and the relationships among the data.   
• The middle level in the three-level architecture is the conceptual level. This level contains the logical structure of the entire database as seen by the DBA.   
• It is a complete view of the data requirements of the organization that is independent of any storage considerations. •  
 The conceptual level represents:   
❖ all entities, their attributes, and their relationships;   
❖ the constraints on the data;   
❖ semantic information about the data;   
❖ security and integrity information.   
• The conceptual level supports each external view, in that any data available to a user must be contained in, or derivable from, the conceptual level.   
• However, this level must not contain any storage-dependent details.   
• For instance, the description of an entity should contain only data types of attributes (for example, integer, real, character) and their length (such as the maximum number of digits), but not any storage considerations, such as the number of bytes occupied.

Internal Level

• The physical representation of the database on the computer. This level describes how the data is stored in the database.   
• The internal level covers the physical implementation of the database to achieve optimal runtime performance and storage space utilization.   
• It covers the data structures and file organizations used to store data on storage devices.   
• It interfaces with the operating system access methods (file management techniques for storing and retrieving data records) to place the data on the storage devices, build the indexes, retrieve the data, and so on.   
• The internal level is concerned with such things as:   
❖ storage space allocation for data and indexes;   
❖ record descriptions for storage (with stored sizes for data items)   
❖ record placement;   
❖ data compression and data encryption techniques.   
• Below the internal level there is a physical level that may be managed by the operating system under the direction of the DBMS.   
• However, the functions of the DBMS and the operating system at the physical level are not clear-cut and vary from system to system.   
• Some DBMSs take advantage of many of the operating system access methods, while others use only the most basic ones and create their own file organizations.   
• The physical level below the DBMS consists of items only the operating system knows, such as exactly how the sequencing is implemented and whether the fields of internal records are stored as contiguous bytes on the disk.

**Schemas, Mappings, and Instances**

Database schema: description of the database  
3 types of schema:  
a) external (highest level)   
b) conceptual (entities, attributes and relationships together with integrity)  
c) internal (stored records, methods of representation, data field and indexes/storage structure used)

There is only one conceptual schema and one internal schema per database.

**Mapping**

- The DBMS is responsible for mapping between these three types of schema.  
-Through conceptual/internal mapping conceptual schema is related to internal schema.  
-Uses info in conceptual schema to map between external and internal schema.  
- This enables the DBMS to find the actual record or combination of records in physical storage that constitute a logical record in the conceptual schema.  
- It also allows any differences in entity names, attribute names, attribute order, data types, and so on, to be resolved.  
  
- Finally, each external schema is related to the conceptual schema by the external/conceptual mapping.  
- This enables the DBMS to map names in the user’s view on to the relevant part of the conceptual schema.   
- The conceptual level is then mapped to the internal level, which contains a physical description of the structure for the conceptual record.   
- Note that the order of fields at the internal level is different from that at the conceptual level. Again, the DBMS maintains the conceptual/internal mapping.

Schema/Intension

-Database schema is the description of db and not db itself.  
-Schema is specified during the db design process.(doesn’t really change)  
-Data in db changes frequently.

Instance/extension/state

-DB instance is the data in db  
- many database instances can correspond to the same database schema.

Data Independence

-Meaning upper levels aren’t affected by lower level changes

>Two types:

1. Logical

* It’s immunity of external schema to changes in conceptual schema
* Changes to conceptual shouldn’t affect external or having to rewrite programs

1. Physical

* Immunity of conceptual schema to changes in the internal schema.
* Changes to internal schema shouldn’t affect conceptual or external schemas.
* From user’s view, only performance change can be noticed (deterioration in performance is main reason for internal schema changes)