DBMS Architecture

Three-Level DBMS Architecture:

1. Internal (Physical) level
2. Conceptual (Logical) level
3. External (View) level

Each level consists of one or more **views** of the underlying data.

Views are described by **schemas**.

Schemas are stored in the **system catalogue**.

|  |  |
| --- | --- |
| Internal  Level | Lowest level of data abstraction.  Internal schema describes how data is physically stored.   * Storage space allocation techniques. * Access paths e.g. indexes. * Data compression & encryption techniques. |
| Conceptual Level | Describes what data is stored, and the relationships among the data.  Includes all information represented in the database. |
| External  View | User’s view of database.  External schemas describe a part of the database for a **particular** group of users or applications. |

### DBMS Components

A **stored data manager** (SDM) controls access to DBMS information on disk.

A **DDL compiler** processes **schema definitions** and stores them in the **catalogue**.

A **catalogue** contains information such as:

* File names & sizes
* Data types of items
* Storage details
* Mapping information among schemas
* Constraints

A **query compiler** parses & validates the submitted query.

It consults the DBMS catalogue, and generates executable code.

A **precompiler** extracts DML commands from a program’s host language, and sends them to the **DML compiler**. Object code for DML commands and the rest of the program are linked to create a **canned transaction**.

A **runtime database processor** handles database accesses at runtime:

* Privileged commands
* Executable queries
* Canned transactions

It utilises and updates the catalogue. It also manages concurrency control and backup & recovery as part of **transaction management**.

A **data dictionary** augments the DBMS catalogue with semantic support.

A **data dictionary system** (DDS) should store and manage:

* Descriptions of the DB schemas
* Detailed information on physical DB design.
* Descriptions on the types of DB users, responsibilities and access rights.
* Descriptions of transactions, applications and relationships of users to transactions.

|  |  |
| --- | --- |
| Integrated Data Dictionary | Independent Data Dictionary |
| Documents the computerised data that is managed by the DBMS.  Accessed at runtime. | Free-standing system that performs its own data management functions.  Normally **passive** - no runtime link between dictionary and DBMS. |