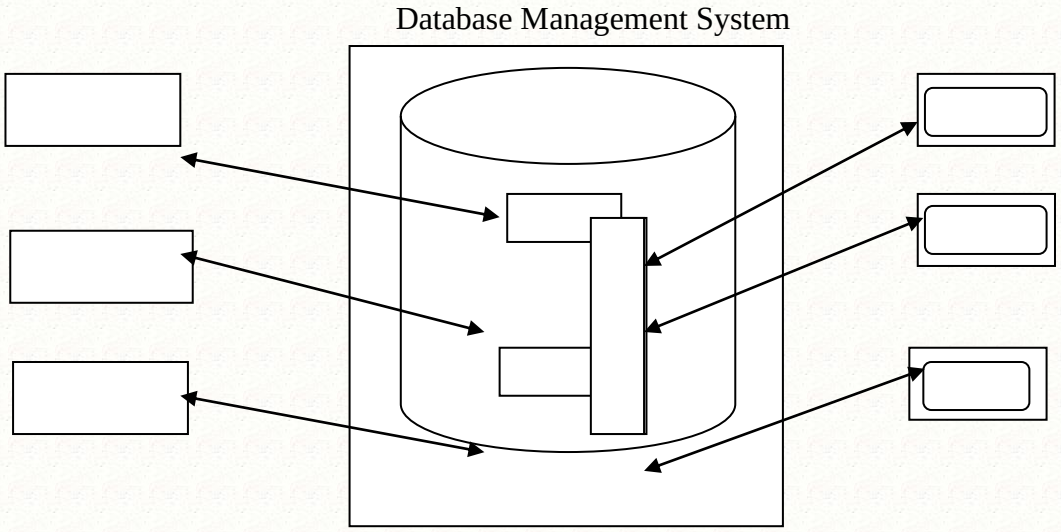
DEPARTMENT OF COMMERCE (CA)
DATABASE MANAGEMENT SYSTEM (Semester-III)
II B.COM(CA) Sub Code-18BCA32C
UNIT –I

Database System Architecture - Basic Concepts: Data System, Operational Data, Data Independence, Architecture for a Database System, Distributed Databases, Storage Structures: Representation of Data. Data Structures and Corresponding Operators: Approach, Hierarchical Approach, Network Approach.

1.Data base system

Database system is nothing more than a computer-based record keeping system (i.e.) a system whose overall purpose is to record and maintain information. The information concerned can be anything that is deemed to be of significance to the organization or the system which may serve the organization in decision-making processes involved in the management of that organization.

The database system involves four major components. They are data, hardware, software and users.



Application program End Users

Fig: Simplified picture of a database system

Data

The data stored in the system is partitioned into one or more databases. A database is a repository for stored data, it is of both integrated and shared.

Integrated: By integrated we mean that the database can be thought of as a unification of several distinct files, with the redundancy among those files eliminated.

Example: Combination of EMPLOYEE and ENROLLMENT data files.

Shared: By Shared we mean that individual pieces of data in the database can be shared among different users that is many users can have access to the same piece of data.

Example: The department information in EMPLOYEE file would be shared by users in the personal department, education department etc.

Hardware

The hardware consists of the secondary storage device disks, drums,etc… on which the database resides together with the associated devices, control units, channels and so forth.

Software

Between the physical database and the users of the system is a layer of software usually called the DBMS.All requests from users for access to the database are handled by the DBMS.One general function provided by the DBMS is thus the shielding of the database users from hardware level. The DBMS provides a view of the database that is elevated somewhat above the hardware level and supports user operation that are expressed in terms of that higher-level view.

Users

There are three broad categories of database users, they are
*application programmers
*end-users
*DBA

1. Application programmers
Application programmer is responsible for writing application programs that use the database. These application programs operate on the data in all the usual ways that is in retrieving information, creating new information, deleting or changing existing information.

2. End-users
End-users access the database from a terminal. An end-user may employ a query language provided as an integral part of the system or may invoke a user-written application program that accepts commands from the terminal and in turn issues requests to the DBMS on the end-user's behalf.

3. Database Administrator
DBMS have central control of both the data and to the programs that access those data. The person who has such control over the system is called DBA.The main functions of DBA are
*Schema definition
*Storage structure and access-method definition
*Granting and physical-organization modification
*Integrity-constraint specification

2. Operational data

A database is a collection of stored operational data used by the application systems of some particular enterprise. Where enterprise is a conventional generic term for any reasonably self-contained commercial, scientific, technical or other organization.

Examples.
Manufacturing company, Bank, Hospital, University, Government department etc.
The enterprise should maintain a lot of data about its operation. The "operational data" for the enterprises quoted above are,
Product data, account data, patient data, student data, planning data.

Example for the illustration of operational data

Consider the manufacturing company where the enterprise will wish to retain information about the projects it has on hand; parts used in those projects; the suppliers who supply the parts; the warehouses in which the parts are stored; the employees who work on the projects etc..These are the basic entities about which data is recorded in the database. In general there will be associations or relationships linking the basic entities together(entity is any distinguishable object).

For example, there is an association between suppliers and parts that is each supplier supplies certain parts and conversely each part is supplied by certain suppliers etc..

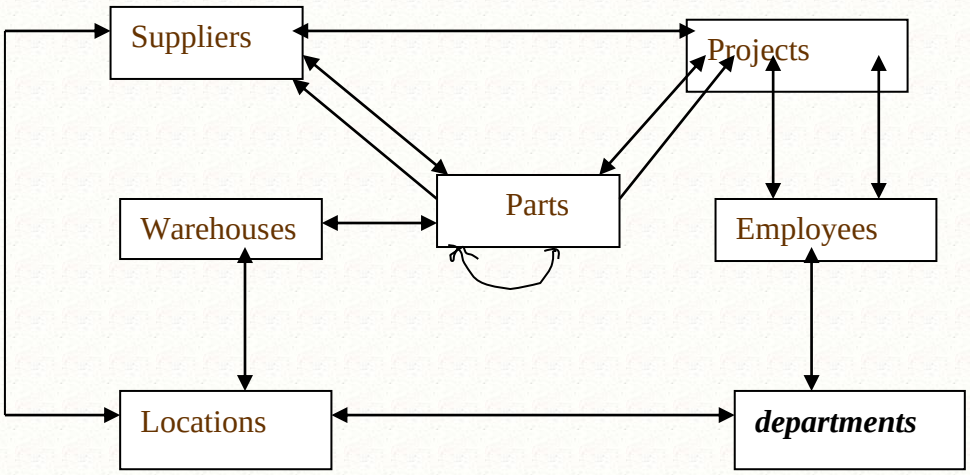


Fig: An example of operational data

The figure illustrates

1.Most of the associations are between two entities or more than that
ex., arrow connecting suppliers-parts-projects
Here supplier s2 supplies part p4 to project j3.

2.The example also shows one arrow involving only in one type of entity (parts)
ex., some parts are components of other parts (a screw is a component of a huge assembly or char etc..)

3.Some entities may be associated in more than one relationship
Ex., projects and employees are linked in two relationships
a. the employee works on the project
b .the employee is the manager of the project
This example clearly illustrates operational data and its functions.

3. Data Independence

The ability to modify a schema definition in one level without affecting a schema in the next higher level is called data independence.
Most present day applications are data-dependent. This means ,the way in which the data is organized in secondary storage and the way in which it is accessed are both dictated by the requirements of the application ,and moreover that knowledge of the data organization and access technique is built into the application logic.
For example, if a file is stored in indexed sequential form, and in order to modify the file the indexes defined should be known. Here the data is dependent, and the modification requires complete application program to be rewritten.
In database system, data resides independent and any modification done at physical level/conceptual level may not affect the database system.

Two types of data independence stated are

1.Physical data independence

Physical data independence is the ability to modify the physical schema without causing application programs to be rewritten. Modifications at the physical level are occasionally necessary to improve performance.
Example,
Modifying the structure of the database using ALTER command etc.

2.Logical data independence

Logical data independence is the ability to modify the logical schema without causing the application programs to be rewritten.
Example,
Modifications such as adding new columns or field to the database.
Most of the modifications are done by the DBA and the types of change that the

DBA wish to make may be explained with the help of the following definitions:

Stored field: Stored field is the smallest unit of data stored in the database.

Ex., database containing information about **parts** would probably include a stored field type called **part number** etc.

Stored record: Stored record is a named collection of associated stored fields.
Stored file: Stored file is the collection of all occurrences of one type of stored record.

Similarly if a data type of the stored field has to be changed is also done by Data. The data storage may be in any of the following form.

1. Representation of numeric data

Data may be stored in internal arithmetic form or as a character string.

2. Representation of character data

A character field may be stored in any of several character codes (eg.EBCDIC,ASCII..)

3. Units for numeric data

The units in a numeric field may change.Ex.,from inches to centimeters

4. Data coding

In some situations it may be desirable to represent data in storage by coded values.
Ex., the value for part color=RED can be interpreted as 1='RED'.

5. Structure of stored records

Two existing types of stored record may be combined into one. For ex., the record types(part number, color) and (part number, weight) may be integrated to give (part number,color,weight).
Also a single type of stored record may be split into two. For ex.,(part number,color,weight) may be broken down into (part number, color) and (part number, weight).

6. Structure of stored fields

A given stored file may be physically implemented in storage in a wide variety of ways.
For ex., storing the file in single storage volume or spread across several volumes.
The above fact implies that the database is able to grow without affecting existing applications.

4.Architecture for a Database system

The architecture is divided into three general levels, they are internal,conceptual,external levels,

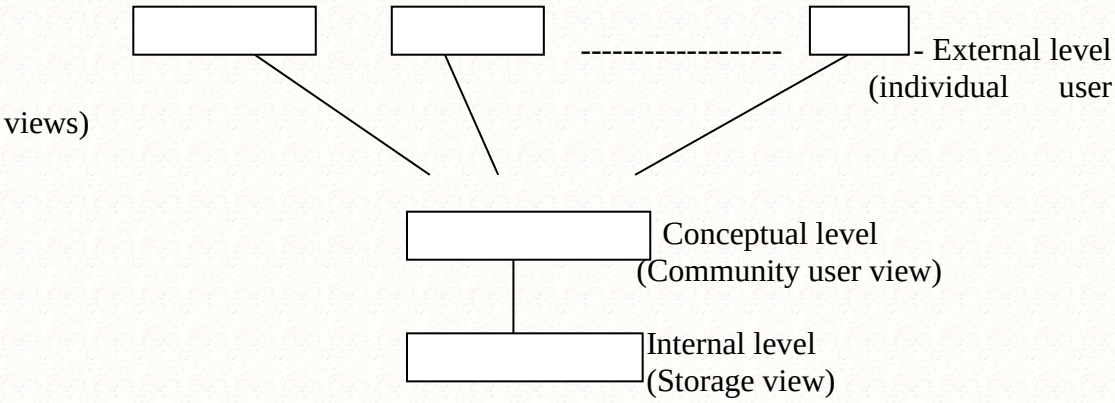


Fig:Three levels of architecture

***Internal level(Physical level)**

This level is the one closest to the physical storage .This is a low-level representation of the entire database; it consists of many occurrences of each of many types of internal record .The storage view is described by means of the internal schema which not only defines the various stored record types but also specifies what indexes exist, how stored files are represented ,what physical sequence the stored records are in and so on.

***Conceptual level (Community logical level)**

This level is the representation of the entire information content of the database. It consists of many occurrences of each of many types of conceptual record. Also this is a level of indirection between the other two levels.

***External level(user logical level)**

This level is closest to the users and is concerned with the way the data is seen by the individual users. The users may be application programmers,end-users,DBA etc.Each user has a language at his/her disposal to interact with the database.
For the application programmer the language will be either a conventional programming like c++,JAVA etc.
For end users the language will be either a query language or some special-purpose language and that language is data sub language (DSL) which is a subset of the total language that is concerned with database objects and operations. The DSL is embedded within the corresponding host language . A given system might support any number of host languages and any number of data sub languages; however, one particular data sub language that is supported by almost all current systems is the language SQL.
Any given data sub language is a combination of at least two subordinate languages-a Data definition language(DDL) and data manipulation language(DML).Where the DDL portion consists of declarative constructs and the DML portion consists of executable statements.
The individual user will generally be interested only in some portion of the total database; moreover ,that user's view of that portion will generally be somewhat abstract when compared with the way the data is physically stored. The term for an

individual user's view is an external view. An external view is thus the content of the database as seen by some particular user.

For example,
A user from the Personnel Department might view the details of employee and department and nothing else.

Detailed System architecture

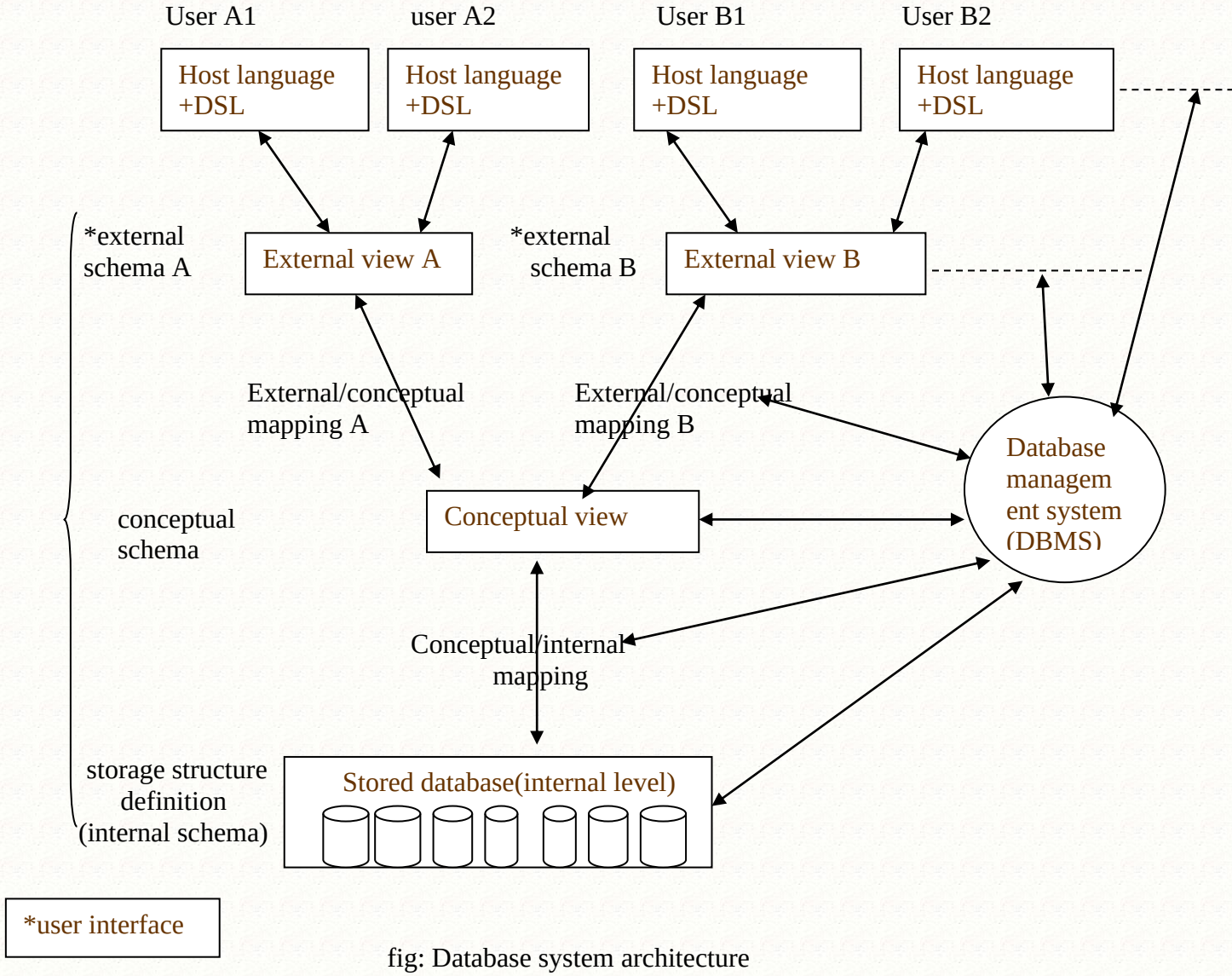


fig: Database system architecture

Mappings

The mappings involved in the architecture are conceptual/internal mapping and external/conceptual mappings.
The conceptual/internal mapping defines the correspondence between the conceptual view and stored database, it specifies how conceptual records and fields are represented at the internal level. If the structure of the stored database is changed then the conceptual/internal mapping must be changed accordingly, so that the conceptual schema can remain invariant. The effects of such changes must be isolated below the conceptual level, in order to preserve physical data independence.

The external/conceptual mapping defines the correspondence between a particular external view and the conceptual view.

Database administrator(DBA)

The Data Administrator(DA) is the person who makes the strategic and policy decisions regarding the data of the enterprise and the DBA is the person who provides the necessary technical support fro implementing those decisions. Thus the DBA is responsible for the overall control of the system in technical level. The major tasks of DBA are

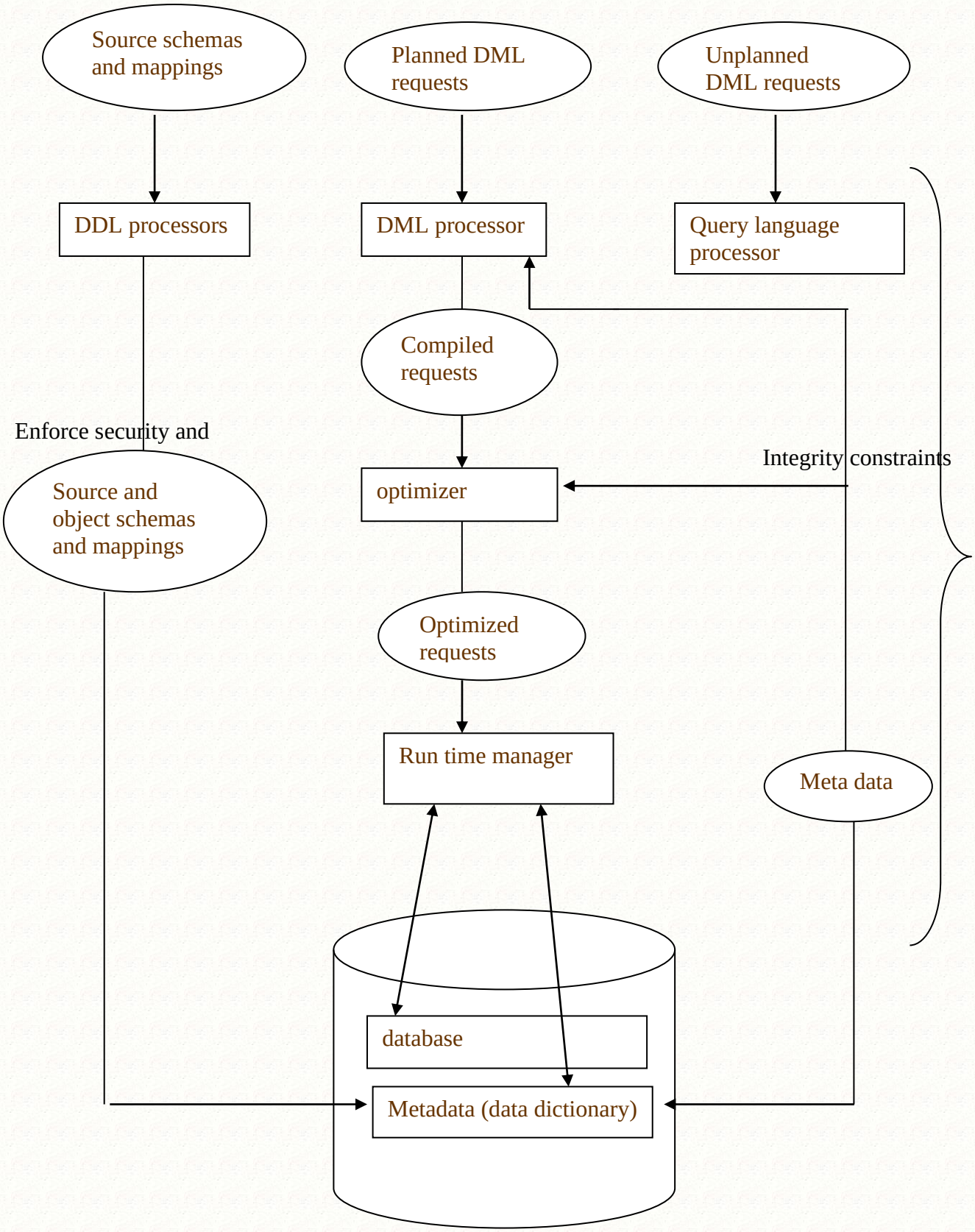
*defining the conceptual schema or schema definition
*storage structures and access-method definition
*schema and physical organization modification
*granting of authorization for data access
*integrity constraint specification

DBMS

The DBMS is the software that handles all access to the database. Its functions are as follows

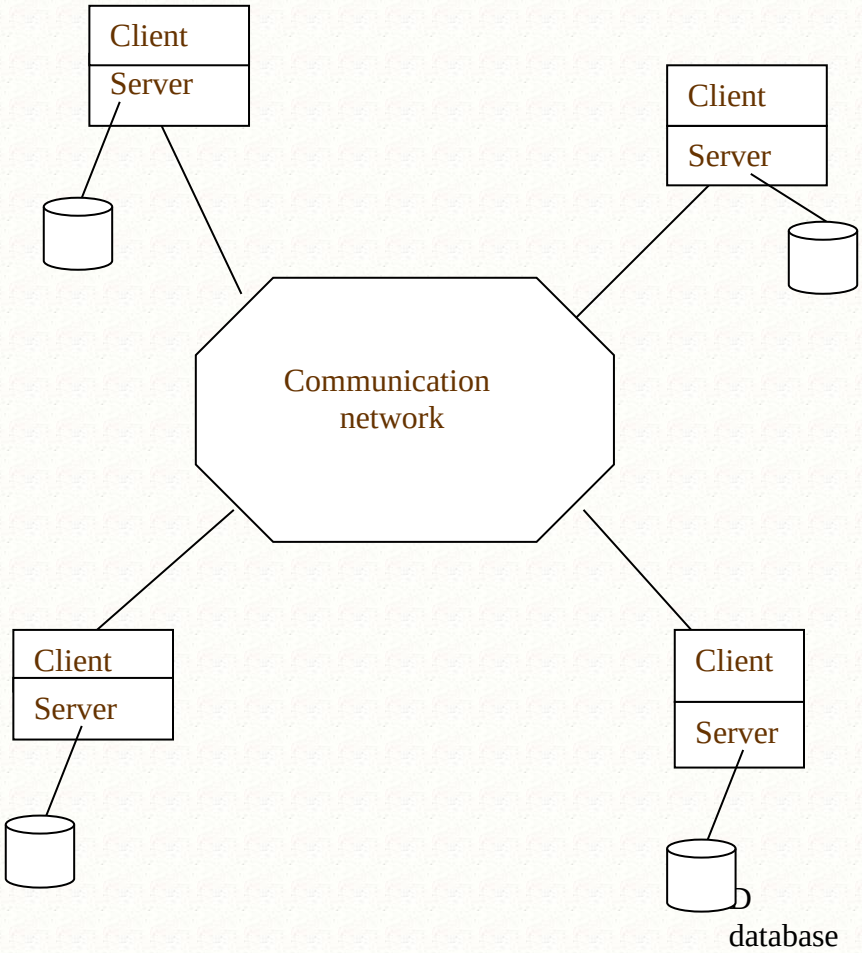
- A user issues an access request using some particular data sub language
- The DBMS intercepts that request and analyses it.
- The DBMS inturn,intercepts the external schema for that user, the corresponding external/conceptual mapping, the conceptual schema, the conceptual/internal mapping, the storage structure definition.
- The DBMS executes the necessary operations on the stored database

The diagrammatic representation of the major functions of DBMS and its components.



5. Distributed databases

The key objective of distributed system is that it should look like a centralized system to the users. Distributed processing means that distinct machines can be connected together into communication network such as the Internet, so that the single data-processing task can span several machines in the network.
A distributed database is typically a database that is not stored in its entirety at a single physical location, but rather is spread across a network of computers that are geographically dispersed and connected through communication links.
For example, consider a banking system in which the customer accounts database is distributed across the bank branch offices, such that each individual customer account record is stored at the customer's local branch. It other words the data is stored at the location at which it is frequently used, but is still available through communication network to users at other locations for example, users at the bank's central office.



Advantages

- Efficiency of local processing
- Data sharing

Disadvantages

- Overhead may be quite high
- Technical difficulties

6. Storage structures and its purposes.

The main idea behind data maintenance is for future reference and it has to be stored for the storage and access of data ,various techniques like sequential ,direct access etc. exists. Once the data is stored in the memory in internal level(physical storage) then it is accessed through DML operations in terms of external records and must be converted in turn to operations at the actual hardware level that is to operations on physical records or blocks. The component responsible for this internal/physical conversion is called an access method. The access method consists of a set of routines whose function is to conceal all device-dependent details from the DBMS and to present the DBMS with a stored record interface.

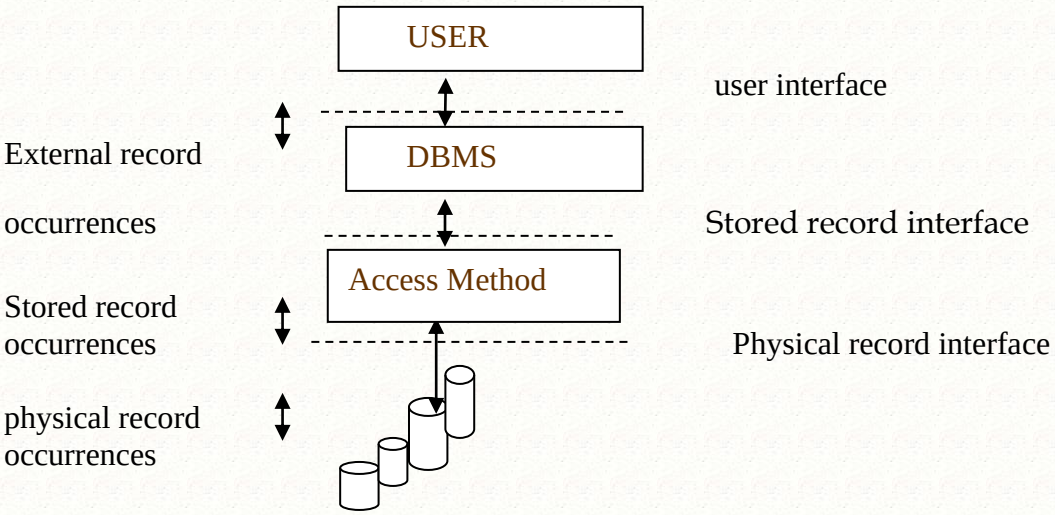


Fig: The stored record interface

The stored record interface thus corresponds to the internal level, just as the user interface corresponds to the external level. Also the stored record interface allows DBMS to view the storage structure as a collection of stored files each consisting of all occurrences of one type of stored record. The DBMS knows

*What stored files exist
*The structure of the corresponding stored record
*The stored fields on which it is sequenced
*The stored field which can be used for direct access etc.

These information will be specified as part of the storage structure definition.
The DBMS does not know

a)anything about physical records
b)how sequencing is performed
c)how direct access is performed
These information are specified to the access method not to the DBMS.

Also ,when a new stored record occurrence is first created and entered into the database, the access method is responsible for assigning it a unique stored record address(SRA).This value distinguishes each stored records from other records, the SRA for a particular occurrence is returned to the DBMS by the access method when the occurrence is first created and may be used by the DBMS for subsequent direct access to the occurrence concerned. The SRA for a given occurrence does not change until the occurrence is physically moved as part of a database reorganization.

7.How data are stored in the physical storage?

There are various possible representations of data within the memory and some of them are explained here. Consider the following example.

S#	Sname	Status	City
S1	Smith	20	London
S2	Jones	10	Paris
S3	Blake	30	Paris
S4	Clark	20	London
S5	Adams	30	Athens

The table consists of information about five suppliers for each supplier a record number ,a supplier name, a status value and a location is recorded. Also the supplier number for each supplier is unique, that is each record is sequenced on the basis of its **primary key**.
The above example is the simplest from of data representation containing only five record occurrences with unique supplier number. If the suppliers are 10000 rather than five and located in **only 10 different cities** then the storage will be wasted specifying the 10 cities among 10000 suppliers. Then the pointer is specified from the supplier file to the city file by separating the city attribute alone to a file.

The following is another form of data the representation

Supplier file

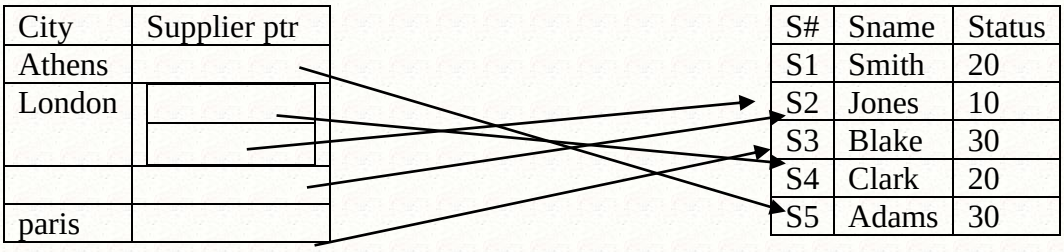
city file

S#	Sname	Status	City-ptr
S1	Smith	20	
S2	Jones	10	
S3	Blake	30	
S4	Clark	20	
S5	Adams	30	

City
Athens
London
Paris

In the above figure the pointers exists from supplier file to the city file and they are SRAs(Storage record address).Advantage of this form of representation over the previous one is, in the later memory space is saved.

The third form of data representation is indexing. If a file is indexed on any of its attributes(more frequently occurring) then accessing such file is quite easier. The representation can be



City	Supplier ptr
Athens	
London	
paris	

S#	Sname	Status
S1	Smith	20
S2	Jones	10
S3	Blake	30
S4	Clark	20
S5	Adams	30

indexed on city

An example,"Find all suppliers in a given city",when this query is placed then the result is retrieved quite easily from the database if represented as above that is in indexed form.

The purpose of indexing is to provide an access path to the file.An index is a file in which each entry(record) consists of a data value together with one or more pointers.The data value is a value for some field of the indexed file and the pointers identify records in the indexed file having that value for that field.An index can be used in two ways first it is used for sequential access to the indexed file and another is used for direct access to individual records in the indexed file on the basis of a given value for that same field.
The another form of dat representation is multilist organisation.

8.DATA STRUCTURES AND CORRESPONDING OPERATORS

The range of data structures supported at the user level is a factor that critically affects many componenets of the system .It dictates the design of the corresponding data manipulation languages,since DML operation must be defined in terms of its effect on those datastructures.We may categorize database systems according to the approach and the best known approaches are

Relational approach
Hierarchical approach
Network approach

The relational approach

The relational approach uses a collection of tables to represent both data and the relationships among those data. Each table has multiple columns and each column has a unique name.

Sample relational database

Bank customer

Customer name	Snsocial-security-no.	customer-street	customer-city	account-no.
Johnson	92-83-7465	Alma	Palo Alto	A-101
Smith	019-28-3746	North	Rye	A-215
Hayes	677-28-9011	Main	Harrison	A-102
Turner	182-73-6091	Putnam	Stamford	A-305
Johnson	192-83-7465	Alma	Palo Alto	A-201
Jones	321-12-3123	Main	Harrison	A-217
Lindsay	336-66-9999	Park	Pits field	A-222
Smith	019-28-3746	North	Rye	A-201

Accounts

account-no	balance
A-101	500
A-215	700
A-102	400
A-305	350
A-201	900
A-217	750
A-222	700

For example, customer Johnson whose social-security-no. is 192-83-7465 lives on Alma in Palo Alto and has 2 accounts A-101 with balance 500,a-201 with balance 900.Also smith and Jhonson shares A-201 account.

Network model

Data in the network model are represented by collections of records and relationships among data .The relationships among data can be represented by links, which can be viewed as pointers

Sample network databases

Johnson	192-83-7465	Alma	Palo Alto

A-101	500

Smith	019-28-3746	North	Rye

A-215	700

Hierarchical Model

This form of data representation is similar to network model in the sense that records represent data and relationships among data and links .It differs from the network model in that the records are organized as collection of trees rather than graphs.

9.Advantages of using DBM
Many enterprises choose to store its operational data in an integrated database because it provides the enterprise with centralized control of its operational data, which is most valuable.

DBA has the central responsibility over operational data.
Advantages if data is stored under centralized control.

1.Redundancy can be reduced
In non-database system each application has its own private files-which may cause redundancy in stored data. By means of integration this can be avoided.

2.Inconsistency can be avoided (to some extent)
Suppose the fact, Employee E3 works in department D8 is represented by two distinct entries in the database and the system is not aware of this duplication. And if any one alone is updated in some occasions they will not agree and comes inconsistent state.
So if the redundancy is controlled then the system could guarantee that the database is never inconsistent as seen by the user, by ensuring that any change made to either of two entries is automatically made to each other. This process is known as propagating updates.

3.The data can be shared
New applications can access the stored databases.

4.Security restrictions can be applied.
Only if permissions are available all users could access the database. The permissions are given by the DBA, so the data ensures security.

5.Integrity can be maintained
Data in the database is accurate or not is mostly validated.

10.Database Administrator

One of the main reasons for using DBMS is to have central control of both the data and the programs that access those data. The person who has such central control over the system is called the database administrator (DBA). The functions of the DBA include the following.

Schema definition: The DBA creates the original database schema by writing a set of definitions that is translated by DDL compiler to a set of tables that is stored permanently in the data dictionary.

Storage structure and access-method definition: The DBA creates appropriate storage structures and access methods by writing a set of definitions, which is translated by the data-storage and data-definition-language compiler.

Schema and physical-organization modification: Programmers accomplish the relatively rare modifications either to the database schema or to the description of the

physical storage organization by writing a set of definitions that is used by either the DDL compiler or the data-storage and data-definition language.

Granting of authorization for data acess: Granting of different types of authorization allows the DBA to regulate which parts of the database various users can access.

Integrity – constraint specification: Setting constraints (conditions) while entering data to the database. For ex, the minimum balance in the account should be at least 500 etc.

BOOKS FOR REFERENCE:

1. Database systems concepts by Abraham Silberschatz, Henry F. Korth.
2. An Introduction to Database System – C. Dsai.
3. An introduction to Database Systems (Seventh Edition) – C.J. Date

Prepared by Dr.N.SHANMUGAVADIVU