



### Topic: 1.8.1 Database Management Systems (DBMS)

#### Flat files and relational databases

A database is an organized collection of data for one or more purposes, usually in digital form. The data are typically organized to model relevant aspects of reality (for example, the availability of rooms in hotels), in a way that supports processes requiring this information (for example, finding a hotel with vacancies). This definition is very general, and is independent of the technology used.

Originally all data were held in files. A typical file would consist of a large number of records each of which would consist of a number of fields. Each field would have its own data type and hold a single item of data. Typically a stock file would contain records describing stock. Each record may consist of the following fields.

Field Name	Data Type
Description	String
Cost Price	Currency
Selling Price	Currency
Number in Stock	Integer
Reorder Level	Integer
Supplier Name	String
Supplier Address	String

This led to very large files that were difficult to process. Suppose we want to know which items need to be reordered. This is fairly straightforward, as we only need to sequentially search the file and, if Number in Stock is less than the Reorder Level, make a note of the item and the supplier and output the details.

The problem is when we check the stock the next day, we will create a new order because the stock that has been ordered has not been delivered. To overcome this we could introduce a new field called On Order of type Boolean. This can be set to True when an order has been placed and reset to False when an order has been delivered. Unfortunately it is not that easy.

The original software is expecting the original seven fields not eight fields. This means that the software designed to manipulate the original file must be modified to read the new file layout.





### Topic: 1.8.1 Database Management Systems (DBMS)

Further ad hoc enquiries are virtually impossible. What happens if management ask for a list of best selling products? The file has not been set up for this and to change it so that such a request can be satisfied in the future involves modifying all existing software. Further, suppose we want to know which products are supplied by Food & Drink Ltd.. In some cases the company's name has been entered as Food & Drink Ltd., sometimes as Food and Drink Ltd. and sometimes the full stop after Ltd has been omitted. This means that a match is very difficult because the data is inconsistent. Another problem is that each time a new product is added to the database both the name and address of the supplier must be entered. This leads to redundant data or data duplication.

The following example, shown in Fig..a.1, shows how data can be proliferated when each department keeps its own files.

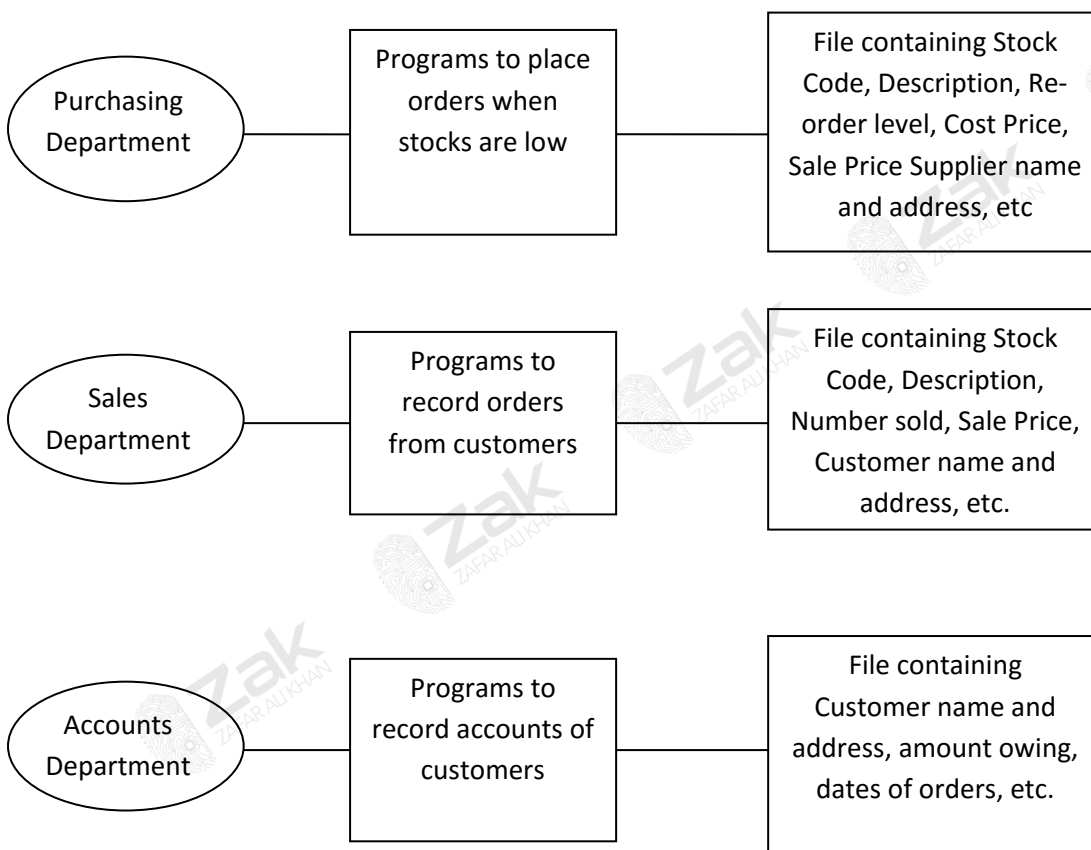


Fig. a.1





### Topic: 1.8.1 Database Management Systems (DBMS)

This method of keeping data uses *flat files*. Flat files have the following limitations.

- Separation and isolation of data  
Suppose we wish to know which customers have bought parts produced by a particular supplier. We first need to find the parts supplied by a particular supplier from one file and then use a second file to find which customers have bought those parts. This difficulty can be compounded if data is needed from more than two files.
- Duplication of data  
Details of suppliers have to be duplicated if a supplier supplies more than one part. Details of customers are held in two different files.
  - Duplication is wasteful as it costs time and money. Data has to be entered more than once, therefore it takes up time and more space.
  - Duplication leads to loss of data integrity. What happens if a customer changes his address? The Sales Department may update their files but the Accounts Department may not do this at the same time. Worse still, suppose the Order Department order some parts and there is an increase in price. The Order Department increases the Cost and Sale prices but the Accounts Department do not, there is now a discrepancy.
- Data dependence  
Data formats are defined in the application programs. If there is a need to change any of these formats, whole programs have to be changed. Different applications may hold the data in different forms, again causing a problem. Suppose an extra field is needed in a file, again all applications using that file have to be changed, even if they do not use that new item of data.
- Incompatibility of files  
Suppose one department writes its applications in COBOL and another in C. Then COBOL files have a completely different structure to C files. C programs cannot read files created by a COBOL program.
- Fixed queries and the proliferation of application programs  
File processing was a huge advance on manual processing of queries. This led to end-users wanting more and more information. This means that each time a new query was asked for, a new program had to be written. Often, the data needed to answer the query were in more than one file, some of which were incompatible.

To try to overcome the search problems of sequential files, relational database management systems were introduced.

A database management system (DBMS) is a software package with computer programs that control the creation, maintenance, and the use of a database. It allows organizations to conveniently develop databases for various applications. A database is an integrated collection of data records, files, and other database objects. A DBMS allows different user application programs to concurrently access the same database. DBMSs may use a variety of database models, such as the relational model, to conveniently describe and support applications.

The relational model for database management is a database model that was first formulated and proposed in 1969 by Edgar F. Codd. The purpose of the relational model is to provide a declarative method for specifying data and queries: users directly state in simple sentences that what information the database contains and what information they want from it, and let the database management system





### Topic: 1.8.1 Database Management Systems (DBMS)

software take care of describing data structures for storing the data and retrieval procedures for answering queries.

#### Database Management System (DBMS)

A DBMS can be defined as a collection of related records and a set of program that access and manipulate these records. A DBMS enables the user to enter, store, and manage data. The main problem with the earlier DBMS packages was that the data was stored in the flat file format. So, the information about different objects was maintained separately in different physical files. Hence, the relations between these objects, if any, had to be maintained in a separate physical file. Thus, a single package would consist of too many files and vast functionalities to integrate them into a single system.

A solution to these problems came in the form of a centralized database system. In a centralized database system, the database is stored in the central location. Everybody can have access to the data stored in a central location from their machine. For example, a large central database system would contain all the data pertaining to the employees. The Accounts and the HR department would access the data required using suitable programs. These programs or the entire application would reside on individual computer terminals.

A Database is a collection of interrelated data, and a DBMS is a set of programs used to add or modify this data. Thus, a DBMS is a set of software programs that allow databases to be defined, constructed, and manipulated.

A DBMS provides an environment that is both convenient and efficient to use when there is a large volume of data and many transactions to be processed. Different categories of DBMS can be used, ranging from small systems that run on personal computers to huge systems that run on mainframes.





### Topic: 1.8.1 Database Management Systems (DBMS)

#### Advantages of Using a Relational Database Management System (RDB)

Advantage	Notes
Control of data redundancy	Flat files have a great deal of data redundancy that is removed using a RDB.
Consistency of data	There is only one copy of the data so there is less chance of data inconsistencies occurring.
Data sharing	The data belongs to the whole organisation, not to individual departments.
More information	Data sharing by departments means that departments can use other department's data to find information.
Improved data integrity	Data is consistent and valid.
Improved security	The database administrator (DBA) can define data security – who has access to what. This is enforced by the Database Management System (DBMS).
Enforcement of standards	The DBA can set and enforce standards. These may be departmental, organisational, national or international.
Economy of scale	Centralisation means that it is possible to economise on size. One very large computer can be used with dumb terminals or a network of computers can be used.
Improved data accessibility	This is because data is shared.
Increased productivity	The DBMS provides file handling processes instead of each application having to have its own procedures.
Improved maintenance	Changes to the database do not cause applications to be re-written.
Improved back-up and recovery	DBMSs automatically handle back-up and recovery. There is no need for somebody to remember to back-up the database each day, week or month.





### Topic: 1.8.1 Database Management Systems (DBMS)

#### Benefits of DBMS

A DBMS is responsible for processing data and converting it into information. For this purpose, the database has to be manipulated, which includes querying the database to retrieve specific data, updating the database, and finally, generating reports.

These reports are the source of information, which is, processed data. A DBMS is also responsible for data security and integrity.

The benefits of a typical DBMS are as follows:

#### Data storage

The programs required for physically storing data, handled by a DBMS, is done by creating complex data structures, and the process is called data storage management.

#### Data definition

A DBMS provides functions to define the structure of the data in the application. These include defining and modifying the record structure, the type and size of fields, and the various constraints/conditions to be satisfied by the data in each field.

#### Data manipulation

Once the data structure is defined, data needs to be inserted, modified, or deleted. The functions, which perform these operations, are also part of a DBMS. These functions can handle planned and unplanned data manipulation needs. Planned queries are those, which form part of the application. Unplanned queries are ad-hoc queries, which are performed on a need basis.

#### Data security and integrity

Data security is of utmost importance when there are multiple users accessing the database. It is required for keeping a check over data access by users. The security rules specify, which user has access to the database, what data elements the user has access to, and the data operations that the user can perform.

Data in the database should contain as few errors as possible. For example, the employee number for adding a new employee should not be left blank. Telephone number should contain only numbers. Such checks are taken care of by a DBMS.

Thus, the DBMS contains functions, which handle the security and integrity of data in the application. These can be easily invoked by the application and hence, the application programmer need not code these functions in the programs.







### Topic: 1.8.1 Database Management Systems (DBMS)

#### Data recovery and concurrency

Recovery of data after a system failure and concurrent access of records by multiple users are also handled by a DBMS.

#### Performance

Optimizing the performance of the queries is one of the important functions of a DBMS. Hence, the DBMS has a set of programs forming the Query Optimizer, which evaluates the different implementations of a query and chooses the best among them.

#### Multi-user access control

At any point of time, more than one user can access the same data. A DBMS takes care of the sharing of data among multiple users, and maintains data integrity.

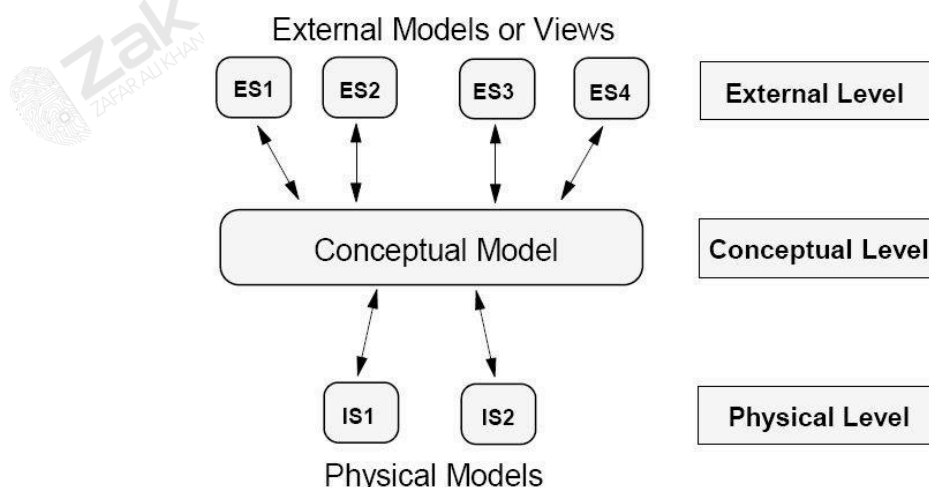
#### Database access languages and Application Programming Interfaces (APIs)

The query language of a DBMS implements data access. SQL is the most commonly used query language. A query language is a non-procedural language, where the user needs to request what is required and need not specify how it is to be done. Some procedural languages such as C, Visual Basic, Pascal, and others provide data access to programmers through these APIs and other tools.

#### Data Modelling

Conceptual, logical and physical schema

This shows that a data model can be an external model (or view), a conceptual model, or a physical model. This is not the only way to look at data models, but it is a useful way, particularly when comparing models.





### Topic: 1.8.1 Database Management Systems (DBMS)

**Conceptual schema:** describes the semantics of a domain (the scope of the model). For example, it may be a model of the interest area of an organization or of an industry. This consists of entity classes, representing kinds of things of significance in the domain, and relationships assertions about associations between pairs of entity classes. A conceptual schema specifies the kinds of facts or propositions that can be expressed using the model. In that sense, it defines the allowed expressions in an artificial "language" with a scope that is limited by the scope of the model. Simply described, a conceptual schema is the first step in organizing the data requirements.

**Logical schema:** describes the structure of some domain of information. This consists of descriptions of (for example) tables, columns, object-oriented classes, and XML tags. The logical schema and conceptual schema are sometimes implemented as one and the same.

**Physical schema:** describes the physical means used to store data. This is concerned with partitions, CPUs, table spaces, and the like.

According to ANSI, this approach allows the three perspectives to be relatively independent of each other. Storage technology can change without affecting either the logical or the conceptual schema. The table/column structure can change without (necessarily) affecting the conceptual schema. In each case, of course, the structures must remain consistent across all schemas of the same data model.

#### Database Schema:

Database schema is the skeleton of database. It is designed when database doesn't exist at all and very hard to do any changes once the database is operational. Database schema does not contain any data or information.

A database schema defines its entities and the relationship among them. Database schema is a descriptive detail of the database, which can be depicted by means of schema diagrams. All these activities are done by database designer to help programmers in order to give some ease of understanding all aspect of database.

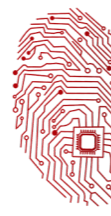
Database schema can be divided broadly in two categories:

**Physical Database Schema:** This schema pertains to the actual storage of data and its form of storage like files, indices etc. It defines the how data will be stored in secondary storage etc.

**Logical Database Schema:** This defines all logical constraints that need to be applied on data stored. It defines tables, views and integrity constraints etc







### Topic: 1.8.1 Database Management Systems (DBMS)

#### Data Dictionary

In database management systems, a file that defines the basic organization of a database. A data dictionary contains a list of all files in the database, the number of records in each file, and the names and types of each field. Most database management systems keep the data dictionary hidden from users to prevent them from accidentally destroying its contents.

Data dictionaries do not contain any actual data from the database, only book-keeping information for managing it. Without a data dictionary, however, a database management system cannot access data from the database.

Developer environment and query processor:

Every database software provides the interface to design schemas and manipulate data through the query processors.

