ST. XAVIER’S COLLEGE

**Maitighar,Kathmandu**

**(Affiliated to Tribhuvan University)**



**Database Management System**

**Lab Assignment #2**

**Submitted By**

Ajita Khatiwada

B.Sc. CSIT

Year II/IV Semester

013BSCIT004

**Submitted To**

Er. Sanjay Kumar Yadav

Lecturer,

Department of Computer Science

St. Xavier’s College

Maitighar, Kathmandu

**Submitted On**

July 27, 2015

**STATEMENT :**

PREPARE A DETAILED REPORT ON :

1.SIMPLIFIED DATABASE SYSTEM ORGANIZATION.

2. APPROACHES TO MANAGEMENT OF DATA

* + DATA BASE APPROACH
  + FILE SYSTEM APPROACH

3. DATABASE VS FILE SYSTEM APPROACHES.

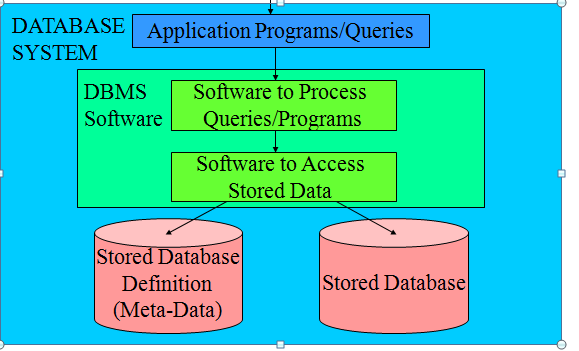
* + DATA ABSTRACTION
  + RELIABILITY
  + EFFICIENCY/PERFORMANCE

4.THREE LAYER ARCHITECTURE (ANSI ABARC ARCHITECTURE)

5.ADVANTAGES AND DISADVANTAGES OF DBMS.

6.DRAWBACKS OF USING THE FILE SYSTEMS TO STORE DATA.

**SIMPLIFIED DATABASE SYSTEM ORGANIZATION[1]**

****

**APPROACHES TO MANAGEMENT OF DATA**

**1.**File system approach

### Traditional Data Storage Model

1. In traditional approach, information is stored in flat files which are maintained by the file system under the operating system’s control.
2. Application programs go through the file system in order to access these flat files

### How data is stored in flat files

* Data is stored in flat files as records.
* Records consist of various fields which are delimited by a space, comma, pipe, any special character etc.
* End of records and end of files will be marked using any predetermined character set or special characters in order to identify them

Example:  Storing employee data in flat files

### 

### Fig: file system approach

### The Database Approach

### Compactness. Data is stored in a single logical “place.”

### Data can be shared and related between applications

### Data transfer between applications is easier

### Used for a wide range of applications

### 

**Traditional File-Based Approach vs Database Approach**

At the beginning, you should understand the rationale of replacing the traditional file-based system with the database system.

**File-based System**

File-based systems were an early attempt to computerize the manual filing system. File-based system is a collection of application programs that perform services for the end-users. Each program defines and manages its data.

However, five types of problem are occurred in using the file-based approach:

1. **Separation and isolation of data**

When data is isolated in separate files, it is more difficult for us to access data that should be available. The application programmer is required to synchronize the processing of two or more files to ensure the correct data is extracted.

1. **Duplication of data**

When employing the decentralized file-based approach, the uncontrolled duplication of data is occurred. Uncontrolled duplication of data is undesirable because:

i. Duplication is wasteful

ii. Duplication can lead to loss of data integrity

1. **Data dependence**

Using file-based system, the physical structure and storage of the data files and records are defined in the application program code. This characteristic is known as program-data dependence. Making changes to an existing structure are rather difficult and will lead to a modification of program. Such maintenance activities are time-consuming and subject to error.

1. **Incompatible file formats**

The structures of the file are dependent on the application programming language. However file structure provided in one programming language such as direct file, indexed-sequential file which is available in COBOL programming, may be different from the structure generated by other programming language such as C. The direct incompatibility makes them difficult to process jointly.

**Fixed queries / proliferation of application programs**

File-based systems are very dependent upon the application programmer. Any required queries or reports have to be written by the application programmer. Normally, a fixed format query or report can only be entertained and no facility for ad-hoc queries if offered.

**Database Approach:**

In order to overcome the limitations of the file-based approach, the concept of database and the Database Management System (DMS) was emerged in 60s.

**Advantages**

A number of advantages of applying database approach in application system are obtained including:

1. **Control of data redundancy**

The database approach attempts to eliminate the redundancy by integrating the file. Although the database approach does not eliminate redundancy entirely, it controls the amount of redundancy inherent in the database.

1. **Data consistency**

By eliminating or controlling redundancy, the database approach reduces the risk of inconsistencies occurring. It ensures all copies of the data are kept consistent.

1. **More information from the same amount of data**

With the integration of the operated data in the database approach, it may be possible to derive additional information for the same data.

1. **Sharing of data**

Database belongs to the entire organization and can be shared by all authorized users.

1. **Improved data integrity**

Database integrity provides the validity and consistency of stored data. Integrity is usually expressed in terms of constraints, which are consistency rules that the database is not permitted to violate.

1. **Improved security**

Database approach provides a protection of the data from the unauthorized users. It may take the term of user names and passwords to identify user type and their access right in the operation including retrieval, insertion, updating and deletion.

1. **Enforcement of standards**

The integration of the database enforces the necessary standards including data formats, naming conventions, documentation standards, update procedures and access rules.

1. **Economy of scale**

Cost savings can be obtained by combining all organization's operational data into one database with applications to work on one source of data.

1. **Balance of conflicting requirements**

By having a structural design in the database, the conflicts between users or departments can be resolved. Decisions will be based on the base use of resources for the organization as a whole rather that for an individual entity.

1. **Improved data accessibility and responsiveness**

By having an integration in the database approach, data accessing can be crossed departmental boundaries. This feature provides more functionality and better services to the users.

1. **Increased productivity**

The database approach provides all the low-level file-handling routines. The provision of these functions allows the programmer to concentrate more on the specific functionality required by the users. The fourth-generation environment provided by the database can simplify the database application development.

1. **Improved maintenance**

Database approach provides a data independence. As a change of data structure in the database will be affect the application program, it simplifies database application maintenance.

1. **Increased concurrency**

Database can manage concurrent data access effectively. It ensures no interference between users that would not result any loss of information nor loss of integrity.

1. **Improved backing and recovery services**

Modern database management system provides facilities to minimize the amount of processing that can be lost following a failure by using the transaction approach.

**Disadvantages**

In split of a large number of advantages can be found in the database approach, it is not without any challenge. The following disadvantages can be found including:

1. **Complexity**

Database management system is an extremely complex piece of software. All parties must be familiar with its functionality and take full advantage of it. Therefore, training for the administrators, designers and users is required.

1. **Size**

The database management system consumes a substantial amount of main memory as well as a large number amount of disk space in order to make it run efficiently.

**THREE LAYER ARCHITECTURE (ANSI ABARC ARCHITECTURE)**

**ANSI/SPARC 3-Tier Architecture[2]**

* Proposal for standard terminology & general architecture for DBSs produced in 1971 by DBTG (Data Base Task Group) appointed by Conference on DBSs & Languages (CODASYL)
* DBTG recognized the need for a 2-tier architecture with system view (schema) & user view (subschema)
* ANSI (American National Standards Institute)-SPARC (Standards Planning & Requirements Committee) produced similar terminology & architecture in 1975(ANSI/X3/SPARC)\* in 1975
* ANSI-SPARC recognized the need for a 3-tier architecture

**Objectives of Three-Level Architecture**

* All users should be able to access same data.
* A user’s view is immune to changes made in other views.
* Users should not need to know physical database storage details.
* DBA should be able to change database storage structures without affecting the users’ views.
* Internal structure of database should be unaffected by changes to physical aspects of storage.
* DBA should be able to change conceptual structure of database without affecting all users**.**

The three level architecture defines DBMS schemas at threelevels:

* + **Internal schema** at the internal level to describe physical storage structures and access paths. Typically uses a physical data model.
  + **Conceptual schema** at the conceptual level to describe the structure and constraints for the *whole* database for a community of users. Uses a *conceptual* or an *implementation* data model.
  + **External schemas** at the external level to describe the various user views. Usually uses the same data model as the conceptual level.

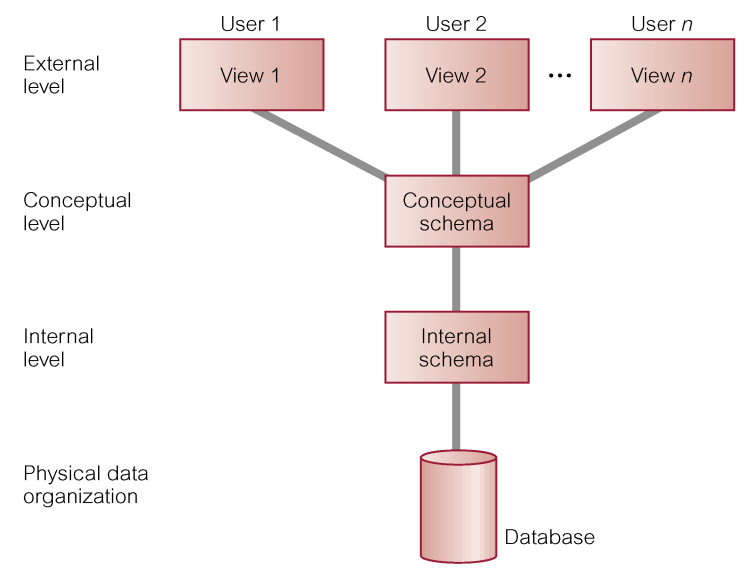
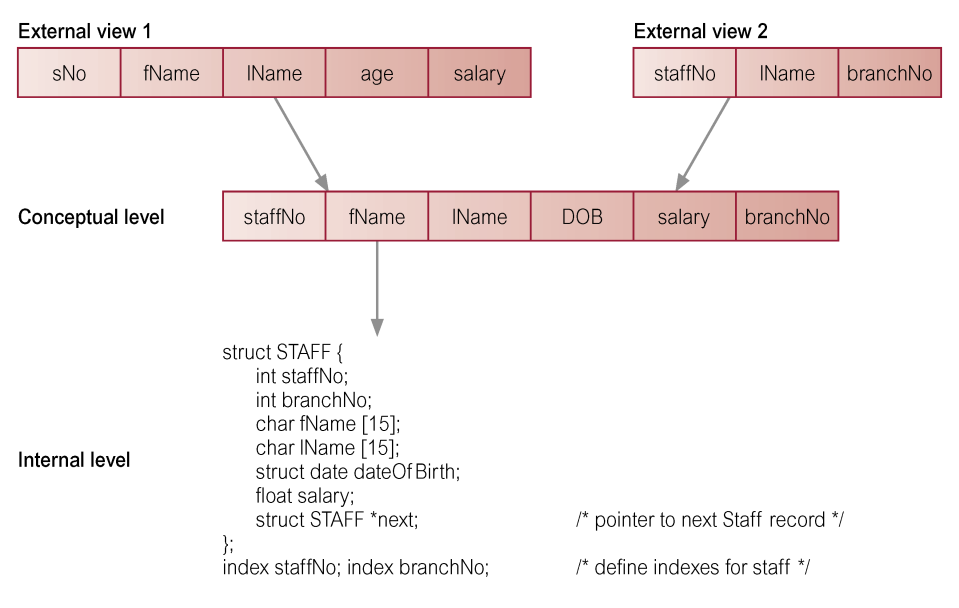


Fig 1: 3 tier architecture

**Differences between Three Levels of ANSI-SPARC Architecture**



**ADVANTAGES AND DISADVANTAGES OF DBMS**

**Advantages of DBMS:**

**Data independence**: DBMS provides abstract view of data. Application programs are independent from details of data representation and storage.

**Efficient data access**: DBMS provides verity of sophisticated techniques to store and retrieve data efficiently. Data integrity and security: DBMS allow to enforce integrity constraints on data. For example before inserting salary information for an employee, DBMS can enforce integrity constraint to check salary is not exceeded department budget. DBMS can also enforce access controls, what data is visible to what class of users.

**Data administration**: DBMS provides centralized administration of data. It is appropriate when several no. of database user shares data. It improves the overall performance of database system.

**Concurrent access and crash recovery**: DBMS has a capability manage concurrent access. It schedules concurrent access to the data in such a manner that user fell data is being accessed by only one user at a time. Moreover, DBMS protects users from the effects of system failures.

**Reduced application development time**: since DBMS supports many important functions that are common to many applications accessing data stored in database. It provides high level interface to data and facilitates quick development of applications.

**Disadvantage of DBMS**

• Complex architecture of DBMS software

• DBMS software cost

• Since DBMS is optimized certain kind of workloads (e.g. answering complex queries or handling many concurrent requests) its performance may not appropriate for certain specialized applications.

• Abstract view of data presented by DBMS may not match for certain applications. For example, relational databases does not supports flexible analysis of text data

• If specialized performance or data manipulation requirements are central to an application, DBMS is not appropriate for such application. The added benefits of a DBMS (e.g. flexible querying, security, concurrent access and crash recovery) may not require for applications.

**DRAWBACKS OF USING THE FILE SYSTEMS TO STORE DATA.**

**Data Redundancy and Inconsistency:**

* Data redundancy: The presence of duplicate data in multiple data files so that the same data are stored in more than one place or location
* Data inconsistency: The same attribute may have different values**.**

**Program-data dependence**

* The coupling of data stored in files and the specific programs required to update and maintain those files such that changes in programs require changes to the data

**Lack of flexibility:**

A traditional file system can deliver routine scheduled reports after extensive programming efforts, but it cannot deliver ad-hoc reports or respond to unanticipated information requirements in a timely fashion.

**Poor security:**

* Because there is little control or management of data, management will have no knowledge of who is accessing or even making changes to the organization’s data**.**

**Lack of data sharing and availability:**

* Information cannot flow freely across different functional areas or different parts of the organization. Users find different values of the same piece of information in two different systems, and hence they may not use these systems because they cannot trust the accuracy of the data.

Reference :

[1] <https://ifm.ac.tz/staff/msaleh/courses/ulevel/DT228/WEB/architecture.ppt>

[2]http://cci.drexel.edu/faculty/thu/Teaching/INSYS210/ENCh02.ppt