UT1. Data Storage

“Bases de datos”

DAM-DAW

Index

- File concepts
- File operations
- File organization methods
- Hash Table
- Database:
  - Integrity, Concurrency, Recovery, Data Dictionary, Confidentiality/ Security
- Three Level Database Architecture Abstraction levels
- Data Dictionary
- DBMS, RDBMS
- RDBMS functions: languages
- Structure of DBMS
- DBA
- MySQL

File Concepts

- File
  - In data processing, using an office metaphor, a file is a related collection of records.
    - **Records:** logical divisions of one file
    - **Fields:** logical divisions of the records
- Classification according to file's content
  - ASCII: (line feed + carriage return)
  - Binary
  - Consider that information is stored using binary code
  - Difference between ASCII, ANSI, UTF-8
    - https://stackoverflow.com/questions/700187/unicode-utf-ascii-ansi-format-differences?noredirect=1
    - http://ascii-table.com/ansi-codes.php

Types of file operations

- **Creation:** In some OS you can allocate in the disk for the file
- **Read**: Meant To Read the information which is Stored into the Files
- **Write**: For inserting or updating contents into a File
- **Delete:**
  - release space and delete contents
  - Is it deleted completely or marked for deleting? Possibility of recover deleted files
- **Sorting** or Arrange the Contents of File
- …

File organization methods

- Serial files
  - Read UT1_GeneralConcepts.pdf
  - Features
    - Sequential reading: to read record n it's necessary to read the previous n-1
    - It doesn't allow reading backwards
    - Update data in the middle of the file requires a complete overwrite of the whole content:
    - New records are inserted at the end (Append mode)
    - No concurrent access in writing mode
    - EOF. What is it?. Look for it in ASCII code

File organization methods

- **Random or direct file organization**
  - **Read UT1_Ingles.doc**
- Aparecen con el disco magnético, disco duro…
  - Permiten posicionamiento directo en un registro concreto, hacia delante o hacia atrás
  - Características
    - Posicionamiento inmediato
    - Registros de longitud fija
    - Permite acceso concurrente en escritura: varios usuarios pueden escribir en el mismo fichero al mismo tiempo siempre que sean posiciones distintas
      - Si es la misma posición se necesita mecanismo de bloqueo
    - Borrado lógico
      - Se requiere algoritmo de reutilización de "huecos" → requiere regenerar las claves de acceso
      - Ejemplo: Compactación de un disco duro
  - Ejemplo:
    - Adecuados cuando se tiene un caso en que p.ej. el código de registro (cliente, proveedor,…)pueda ser directamente el número de registro donde se guardan los datos

File organization methods

- **Indexed-sequential File**
  - **Read UT1_Ingles.docx**
  - It can be accessed sequentially or by direct access using indexes
  - The index is an structure that contains information about where data if we know the index
  - One file can have several indexes defined
  - The index requires space in disk (by the moment consider than we'll have one file for data and other file for each index)
  - B-tree :
    - composed by nodes where information about the key value and position of the record is stored.
    - Extension of binary trees. (we can have more than 2 nodes below another node) . That is a node can have more than two children.
      - By the way what does siblings mean?
  - Example:
    - Book. How many indexes do you find in a book?

File organization methods

- **Indexed Files**
  - An **indexed file** is a computer file with an index that allows easy random access to any record given its file key.
  - The key must be such that it uniquely identifies a record. If more than one index is present the other ones are called *alternate indexes*. The indexes are created with the file and maintained by the system.
  - **Index Tree**
    - a B-tree is a self-balancing tree data structure that keeps data sorted and allows searches, sequential access, insertions, and deletions in logarithmic time. (
    - The B-tree is a generalization of a binary search tree in that a node can have more than two children.-
    - Discussion: Does index access is always better than sequential access?:
      - Rule of thumb : Answer for relational Data Bases
        - » If a query needs to access more than 10% of the data, normally a full scan is better than an index

Hash Table

- Indexed
  - **hash table** (**hash map**) is a data structure that implements an associative array abstract data type, a structure that can map keys to values.
  - A hash table uses a hash function to compute an *index* into an array of *buckets* or *slots*, from which the desired value can be found.
  - In many situations, hash tables turn out to be on average more efficient than search trees or any other table lookup structure.
  - Collisions:
    - Hash collisions are practically unavoidable when hashing a random subset of a large set of possible keys.
    - Almost all hash table implementations have some collision resolution strategy

Exercises

- Student task
  - ASCII vs. ANSI.
  - Unicode
    - Why Unicode
      - http://www.unicode.org/charts
  - Collection of data and file design
    - A practical exercise will be given to the students for the Analysis and Design of the structure of different files that will complement the concepts worked in this unit.

Database

- A database is a collection of information that is organized so that it can be easily accessed, managed and updated.
  - When information is entered into and stored in a computer, it is generally referred to as data. After processing (such as formatting and printing), output data can again be perceived as information.
  - We have to think about
    - Relationship between data
    - Redundancy
    - Integrity
    - Consistency
    - Confidentiality /Security
    - Concurrency
    - …

Database. Think about…

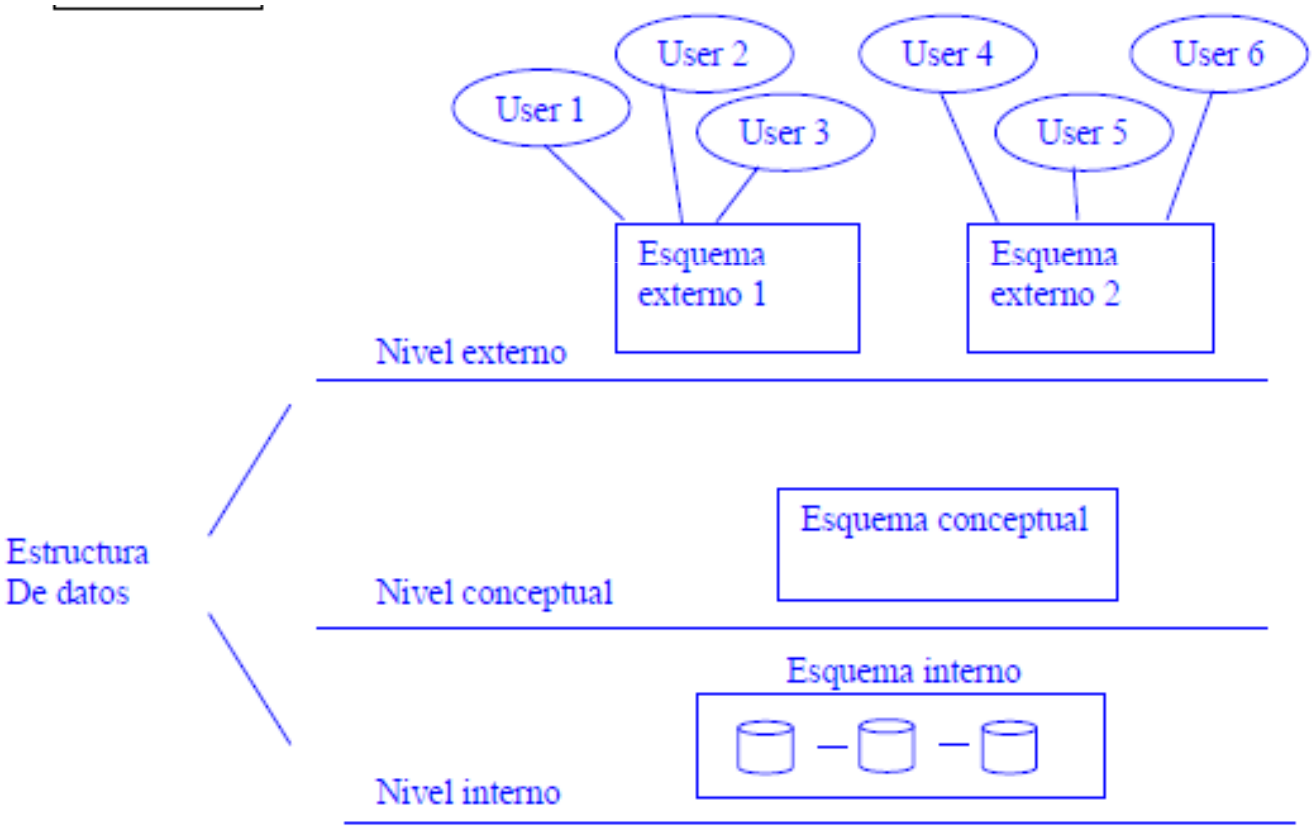
- **Integrity :** consistency of data
  - Give an example of inconsistency
  - Data cannot be “corrupted”, logically of physically
- **Concurrency:** several users can see or trying to update the same data
- **Recovery:** how can we recover at a precise moment in the past?
- **Definition and content:**
  - Definition of containers (tables) and data are inside de database
    - Data Dictionary:
- **Confidentiality / security.** Legal requirements (LOPD)

Database Objectives

- Avoid data inconsistency
- Logical data and physical data independence
  - Immunity of external models to changes in the logical model
  - Immunity of logical model to changes in internal model
- Allow concurrent access
- Provide security mechanisms to avoid access to certain data
- Integrity problems :
  - Only ‘Y’, ‘N’
  - Maximum value is 100….

Three Level Database Architecture
Abstraction levels

- Data are actually stored as bits, or numbers and strings, but it is extremely difficult to work with the variety and complexity of data at this level.
- It is helpful to view data with different levels of abstraction



Three Level Database Architecture

- **External level**
  - The most abstract level and closest to users.
  - An external schema specifies a **view** of the data in terms of the conceptual level.
  - It is tailored to the needs of a particular category of users.
  - Portions of stored data should not be seen by some users and begins to implement a level of security and simplifies the view for these users.
  - LOPD
  - Examples:
    - Students should not see faculty salaries.
    - Some users could read some data but not update them
- **Conceptual data level**
  - Also referred to as the Logical level when the conceptual level is implemented to a particular database architecture.
  - It describes which data are stored in the DB, the relationships among them, integrity restrictions, ….
  - It neither consider physical organization not access methods
  - We will abstract the logical view as a conceptual view using **Entity-Relationship Modelling,** which is database architecture independent.
  - Hides storage details of the internal/physical level
- **Internal data level (physical scheme)**
  - It's managed by the DBA (Data Base Administrator).
  - Lowest abstraction level and the closes to the physical storage of data
  - The physical schema of the internal level describes details of how data is stored: files, indices, etc. on the random access disk system.

Abstraction Levels: Example

EJEMPLO que muestra los 3 niveles de abstracción para una base de datos relacional correspondiente a un centro de estudios. En dicha D.B. se almacenan los datos relativos a los alumnos, las asignaturas en las que se están matriculados y las calificaciones que obtienen en las mismas.

ESQUEMA CONCEPTUAL: Definición de todas las Tablas, Columnas y R estricciones.

Tabla Alumnos. Columnas: Nº Matrícula, Nombre, Curso, Dirección, Población.
Clave: Nº Matrícula

Nº Matrícula	Nombre	Curso	Dirección	Población
11111	Ana	1	C/ Pilón, 10	Oropesa
11110	Rosa	2	C/ Las viñas, 26	Lagartera
11122	Juan	2	C/ Amapolas, 24, 3F	Berrocalejo
23445	Alicia	1	C/Lamina, 34	Celeruela

Tabla Asignaturas. Columnas: Código Asignatura, Nombre de Asignatura. Clave: Código.

Código Asignatura	Nombre deAsignatura
1	Desa. de Aplic. en Entornos de 4ª Genera.y H. Case
2	Programación en Lenguajes Estructurados
3	Sistemas Informáticos Multiusuario y en Red

Tabla Notas. Columnas: Nº Matrícula, Código Asignatura. Nota

Nº Matrícula	Código Asignatua	Nota
11111	2	6
11111	3	8
11110	1	5
11122	1	7
23445	2	5
23445	3	4

ESQUEMA EXTERNO: Visión parcial de las tablas según el usuario. Vista para el programa de listado de notas de alumnos con los siguientes datos: Curso, Nombre, Nombre de Asignatura y Nota.

Curso	Nombre	Nombre Asignatura	Nota
1	Ana	Programación en Lenguajes Estructurados	6
1	Ana	Sistemas Informáticos Multiusuario y en Red	8
2	Rosa	Desa. de Aplic. en Entornos de 4ª Genera.y H. Case	5
2	Juan	Desa. de Aplic. en Entornos de 4ª Genera.y H. Case	7
1	Alicia	Programación en Lenguajes Estructurados	5
1	Alicia	Sistemas Informáticos Multiusuario y en Red	4

ESQUEMA INTERNO: Almacenamiento físico de datos.

Archivo de índices para ALUMNOS: Clave Nº Matrícula, dirección de la fila.
Archivo de índices para ASIGNATURAS: Clave Código Asignatura, dirección de la fila.
Archivo de ALUMNOS: Nº matricula, nombre, curso, dirección, población.
Archivo de ASIGNATURAS: Código, nombre de asignaturas.

Exercise

- Create the conceptual schema in MS Access
- Create the file structure in MS Access

Data Dictionary

- For a specific DB we'll have only one internal schema, one logical and several externals or user
- A **data dictionary contains**:
  - The definitions of all schema objects in the database (tables, views, indexes, clusters, synonyms, sequences, procedures, functions, packages, triggers, and so on)
  - How much space has been allocated for, and is currently used by, the schema objects
  - Default values for columns
  - Integrity constraint information
  - The names of Oracle users
  - Privileges and roles each user has been granted
  - Auditing information, such as who has accessed or updated various schema objects
  - Other general database information

Database Management Systems

- https://searchdatamanagement.techtarget.com/definition/RDBMS-relational-database-management-system
- Relational database management system (RDBMS) is **a collection of programs and capabilities that enable IT teams and others to create, update, administer and otherwise interact with a relational database**. Most commercial RDBMS use Structured Query Language (SQL) to access the database, although SQL was invented after the initial development of the relational model and is not necessary for its use.

- **RDBMS vs. DBMS**
  - In general, databases store sets of data that can be queried for use in other applications. A database management system supports the development, administration and use of database platforms.
  - An RDBMS is a type of DBMS with a row-based table structure that connects related data elements and includes functions that maintain the security, accuracy, integrity and consistency of the data.

RDBMS

- Is a database management system (DBMS) based on the relational model invented by Edgar F. Codd at IBM's San Jose Research Laboratory.
- Most databases in widespread use today are based on his relational database model.
- Relational databases have often replaced legacy **hierarchical databases** and **network databases** because they were easier to implement and administer
- RDBMS received continued, unsuccessful challenges by **object database management** systems in the 1980s and 1990s, (which were introduced in an attempt to address the so-called object-relational impedance mismatch between relational databases and object-oriented application programs), as well as by **XML database** management systems in the 1990s
- Due to the expanse of technologies, such as horizontal scaling of computer clusters, NoSQL databases have recently become popular as an alternative to RDBMS databases.
- See Garceta book

RDMBS functions: Languages

- The most basic RDBMS functions are related to create, read, update and delete operations, collectively known as CRUD
- https://www.w3schools.in/mysql/ddl-dml-dcl/
- **DDL** is short name of **Data Definition Language,** which deals with database schemas and descriptions, of how the data should reside in the database.

- **DML** is short name of **Data Manipulation Language** which deals with data manipulation and includes most common SQL statements such SELECT, INSERT, UPDATE, DELETE, etc., and it is used to store, modify, retrieve, delete and update data in a database.

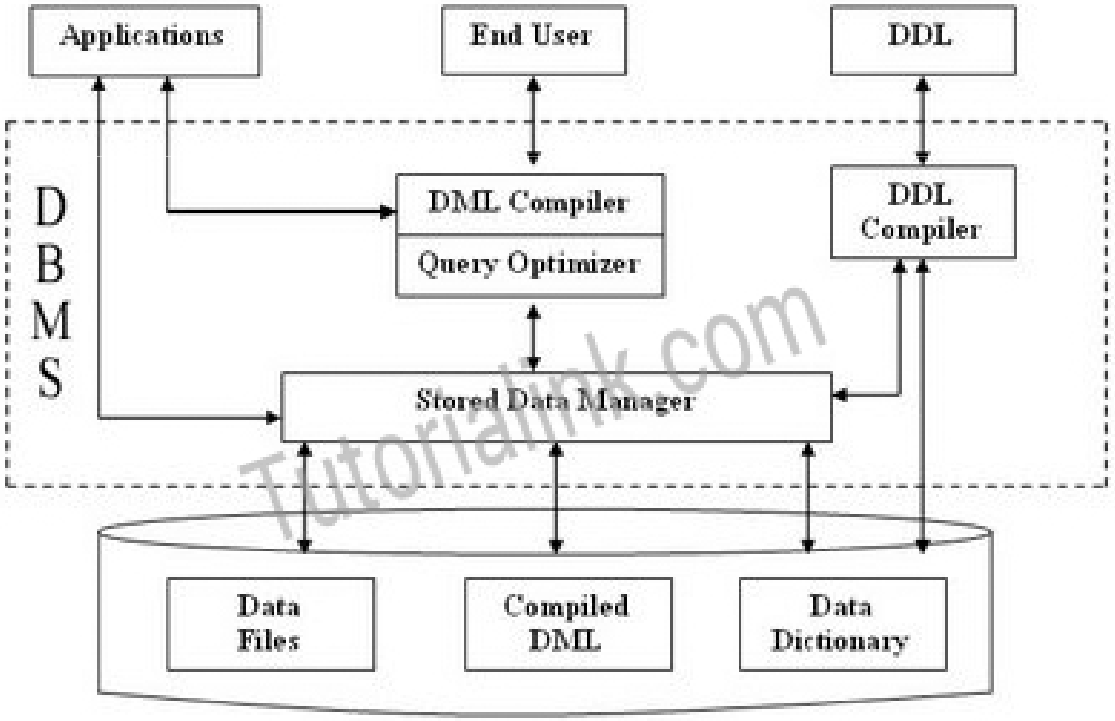
- **DCL** is short name of **Data Control Language** which includes commands such as GRANT and mostly concerned with rights, permissions and other controls of the database system.

- **TCL** is short name of Transaction Control Language which deals with a transaction within a database.

Structure of DBMS

- DBMS (Database Management System) acts as an interface between the user and the database. The user requests the DBMS to perform various operations such as insert, delete, update and retrieval on the database.

- The components of DBMS perform these requested operations on the database and provide necessary data to the users.



Structure of DBMS

- Modules for query management
  - DDL Compiler:
    - Data Description Language compiler processes schema definitions specified in the DDL.
    - It includes metadata information such as the name of the files, data items, storage details of each file, mapping information and constraints etc.
  - DML Compiler and Query optimizer:
    - The DML commands such as insert, update, delete, retrieve from the application program are sent to the DML compiler for compilation into object code for database access.
    - The object code is then optimized in the best way to execute a query by the query optimizer and then send to the data manager.
    - Compiled DML: The DML complier converts the high level Queries into low level file access commands known as compiled DML
  - DDL interpreter
    - Interprets DDL instructions and register then in the Dictionary (metadata)

Structure of DBMS

- Data Manager:
  - The Data Manager is the central software component of the DBMS also knows as Database Control System.
- The Main Functions Of Data Manager Are:
    - Convert operations in user's Queries coming from the application programs or combination of DML Compiler and Query optimizer which is known as Query Processor from user's logical view to physical file system.
    - Controls DBMS information access that is stored on disk.
    - It also controls handling buffers in main memory.
    - It also enforces constraints to maintain consistency and integrity of the data.
    - It also synchronizes the simultaneous operations performed by the concurrent users.
    - It also controls the backup and recovery operations.

Structure of DBMS

- Data Dictionary:
  - which stores **metadata** about the database, in particular the schema of the database.
  - names of the tables, names of attributes of each table, length of attributes, and number of rows in each table.
  - Detailed information on physical database design such as **storage structure, access paths, files and record sizes**.
  - **Usage statistics such as frequency of query and transactions.**
  - Data dictionary is used to actually control the data integrity, database operation and accuracy. It may be used as a important part of the DBMS

DBA

- http://pkirs.utep.edu/cis4365/Tutorials/Database%20Administration/8.00700/1_multipart_xF8FF_2_tutorial.htm
- DBA. **Database administration** is more of an operational or technical level function responsible for physical database design, security enforcement, and database performance. Tasks include maintaining the data dictionary, monitoring performance, and enforcing organizational standards and security.
  - Definition of the internal scheme.
  - Selection of hardware and software
  - Managing data security and privacy
  - Protection of data against accidental or intentional loss, destruction, or misuse
    - Backup/Recover policies
  - Data Disaster Recovery plan
  - Establishment of user privileges
  - Complicated by use of distributed systems such as internet access and client/ server technology.
  - …

RDBMS . MySQL

- Activity
  - MySQL installation
  - Create in MySQL the databases defined in Access

References

- MySQL
  - http://www.mysql.com
  - https://dev.mysql.com/doc/refman/8.0/en/

- Tutorial
  - https://www.w3schools.com/sql/default.asp