

TEMA 1

INTRODUCCIÓ

A LES BASES DE DADES

1. Introducció
2. Fitxers
3. Bases de Dades
4. Sistema Gestor de BD
5. Sistemes d'Informació

1. Introducció

Des de l'antiguitat, hem tingut la **necessitat d'emmagatzemar** la informació per a poder-la consultar posteriorment, traure conclusions, etc. Per exemple, fa milers d'anys ja es recollia la informació de collites o de censos, en els quals s'escrivien les dades en forma més o menys estructurada.

Hui vivim en una societat de la informació, on el gran volum d'informació necessita **tècniques per a emmagatzemar** eixes dades de forma electrònica i poder consultar-les després. Estes tècniques van des dels **fitxers** fins a les **Base de Dades (BD)**.

Una BD és un conjunt de dades organitzades segons una estructura coherent, de forma que un programa o aplicació pugui consultar o modificar eixes dades de forma eficient. Això s'aconsegueix amb el llenguatge SQL.

Exemples de BD: el catàleg de llibres i préstecs d'una biblioteca, la facturació d'una empresa, les estadístiques de la lliga de futbol, el sistema d'avaluació d'un institut, la informació d'hisenda, un lloc web d'apostes esportives...

Per a construir un edifici no es pot començar posant rajoles, sinó que cal primer que l'arquitecte faci uns plànols, etc. D'igual forma, **per a crear una Base de Dades** (d'una grandària considerable) caldrà fer primer unes **fases d'anàlisi i disseny**.

En esta assignatura vorem en què consisteixen eixes fases fins arribar a la creació pròpiament dita de l'estructura d'una base de dades per a, després, poder introduir-li dades, consultar-les i modificar-les amb el llenguatge SQL.

En este tema coneixerem primer altra forma de guardar les dades (els fitxers) i els seus inconvenients. A continuació vorem uns conceptes genèrics de les bases de dades. Després, les aplicacions per a tractar amb les BD: els Sistemes Gestors de Bases de Dades. I per a acabar, vorem les qualitats que ha de tindre la informació i les fases que cal seguir per a obtenir una BD de qualitat.

2. Fitxers

2.1 Definició de fitxer

Un fitxer (o arxiu) és una seqüència d'informació guardada en memòria secundària (disc dur, pen-usb, cd...) que s'identifica amb un nom únic dins d'un mateix directori.

Va ser la primera metodologia que es va usar per a emmagatzemar dades en suport informàtic.

2.2 Exemple de fitxer

Per exemple, suposem que tenim el fitxer clients.txt amb la següent informació:

```
Pep Garcia, 636476543, pepgar@gmail.com;  
Ximo Llopis, 619453323, ximet88@hotmail.com;  
Carmen Sisternes, 687654334, carmensita@gmail.com;  
Casimir Vendrell, 665456788, quasi1000@ono.com;  
Manolo Casanova, 676543456, manuksnv@gmail.com;  
Elisa Pastisa, 678886543, pastissereta88@hotmail.com;  
Pepita Marenrodes, 678765443, decabrioles@ono.com;  
Toni Giménez, 677765112, gimtoni@hotmail.es;  
Jaume Mimomu, 699986444, memimamolt@hotmail.com;  
Robert Tototó, 675432121, robertototototo@gmail.com;  
Maria Delaó, 654123877, esrascavaelfogo@hotmail.com;  
Conxa Queta, 676543212, ambjaqueta@ono.com;  
Aitor Menta, 666543212, faremcomencullera@hotmail.com;  
Alexis Oset, 676787655, alex6o7@gmail.com;  
Ferran Ferran, 678329812, perguelofici@hotmail.com;  
Felip Cinqué, 672111345, capperavall@hotmail.com;
```

La informació dels fitxers ha de tindre una certa estructura per a que pugui ser accedida des d'algun programa. En l'exemple anterior veiem que la informació està estructurada de la següent manera:

- Tota la informació d'un mateix client (registre) està en una mateixa línia. És a dir: els registres estan separats per un punt i coma i/o un caràcter especial EOL (End Of Line).
- De cada client volem saber les mateixes coses: nom, telèfon i e-mail (camps). Els camps estan separats per comes (també podrien estar separats per un tabulador, etc).
- El final del fitxer està marcat per un caràcter especial EOF (End Of File).

2.3 Inconvenients dels fitxers

Esta forma de guardar la informació presenta molts inconvenients:

- ✓ **Redundància de dades.** Per exemple, en un fitxer de factures apareixeria moltes vegades la informació del client (dni, nom, telèfon...) i pot dur a incoherències, més ocupació de memòria, etc.
- ✓ **Dependència del programa respecte a les dades.** Suposem que un programa consulta el telèfon d'un fitxer de clients (segon camp de cada registre). Si afegírem el domicili entre el nom i el telèfon, el programa deixaria de funcionar.
- ✓ **Difícil accés a les dades relacionades.** Si la informació d'un institut (assignatures, alumnes, matrícules...) està en diferents fitxers, si per exemple volem obtindre un llistat amb les dades dels alumnes i de les notes que han tret en cada assignatura, caldria recórrer diversos fitxers per a obtindre eixa informació.
- ✓ **Anomalies en accessos concurrents.** Si dos o més programes intenten modificar al mateix temps un fitxer dels comptes d'un banc per a actualitzar un saldo, pot ocasionar incorreccions.
- ✓ **Costosa recuperació de fitxers.** Si mentre es modifica un fitxer se'n va la llum, pot provocar una actualització incompleta i, per tant, incorrecta.
- ✓ **Problemes de seguretat.** No tots els usuaris haurien d'accedir a tota la informació d'un fitxer o de la mateixa manera. Per exemple, els alumnes haurien de poder vore les notes, però no modificar-les.

Per a solucionar estos problemes es desenvoluparen les bases de dades: llocs on poder emmagatzemar tota la informació de forma estructurada, sense redundàncies, independents dels programes que les usen, de fàcil accés, etc.

3. Bases de Dades

3.1. Definicions

Base de dades: conjunt de dades estructurades, amb redundància controlada que capten les interrelacions i restriccions del món real i emmagatzemades en suport secundari, de forma que els usuaris o aplicacions puguin consultar o modificar eixes dades de forma eficient. Això s'aconsegueix amb el llenguatge SQL.

Exemple: la BD "institut", on estaran les dades de tot un institut.

Taula: Una base de dades està formada per diverses taules. Una taula guarda informació d'objectes del mateix tipus.

Exemple: la taula "alumnes" de la base de dades "institut".

ALUMNES			
DNI	NOM	DATA_NAIX	TEL
11223344A	Joan Antoni Ciscar Peris	1995-10-01	555 962 123
22334455F	Maria Ciscar Peris	1995-10-01	555 962 123
33445566A	M. Josep Mas Ortola	1996-02-16	555 635 287

Atributs: cadascuna de les dades que es vol saber de cada objecte

Exemples: dni, nom, data_naix...

Registres: un registre és el conjunt dels valors de cadascun dels atributs d'una taula referents a un mateix objecte.

Exemple: la informació d'un alumne de la taula "alumnes":

22334455F	Maria Ciscar Peris	1995-10-01	555 962 123
-----------	--------------------	------------	-------------

SGBD: programa que permet als usuaris i programes interactuar amb les BD.

Exemples: MySQL, PostgreSQL, Oracle, Informix...

3.2. Avantatges de les BD respecte als fitxers

- ✓ Fàcil accés a dades interrelacionades.
- ✓ Fàcil modificació, inserció i esborrat de les dades.
- ✓ Evita la redundància de la informació.
- ✓ Garanteix la integritat de la informació (evita contradiccions, etc).
- ✓ Permet el control de permisos als usuaris sobre les dades.
- ✓ Permet la concurrència per a diversos usuaris simultanis.
- ✓ S'aconsegueix independitzar dades amb les aplicacions que hi accedeixen.

3.3. Tipus de BD

Ja veurem que, per a dissenyar una BD, cal seguir un model de dades determinat. Depenent del model de dades obtindrem un tipus de bases de dades diferent.

- ***Tipus de BD clàssics***
 - BD jeràrquiques
 - BD en xarxa
 - BD relacional
- ***Tipus de BD OO (Orientades a Objectes)***
 - BD orientades a objectes pur
 - BD objecte-relacionals

Les BD més utilitzades amb molta diferència són les **BD relacionals**, que són les que dissenyarem en este mòdul, usant el model relacional. El nom es deu a les relacions que hi haurà entre les diferents taules de la BD.

Més endavant també vorem les **BD objecte-relacionals**, que vénen a ser una ampliació de les BD relacionals per a usar en les taules l'orientació a objectes.

4. Sistema Gestor de Bases de Dades (SGBD)

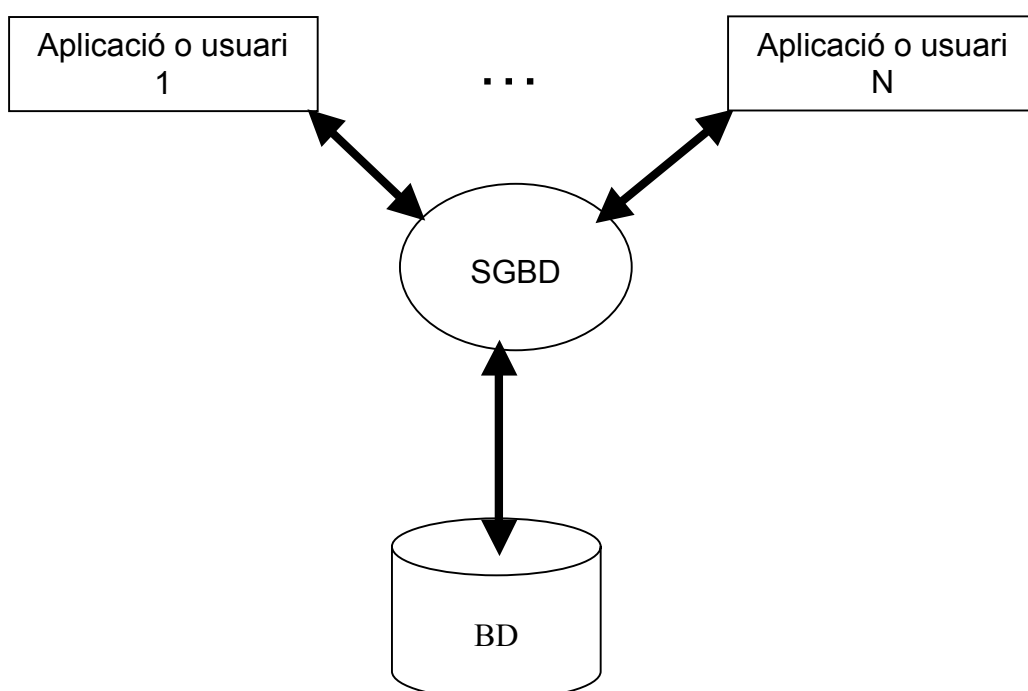
4.1 Definició

Un SGBD és un conjunt de programes que proporcionen l'accés a la informació d'una base de dades. És la interfície entre la base de dades i els programes d'aplicació que accedeixen a aquesta.

L'SGBD que utilitzarem nosaltres serà MySQL, que, com molts altres, utilitza el llenguatge SQL per a interactuar amb les bases de dades.

4.2 Funcionament

Quan un usuari o aplicació vol fer alguna operació sobre la BD (consulta, inserció, modificació de dades...) envia a l'SGBD la petició (instrucció en SQL). El SGBD processa la petició SQL i accedeix a la BD (a través del sistema operatiu) per a consultar o fer els canvis corresponents en les dades. A continuació el SGBD torna a l'aplicació o usuari les dades sol·licitades (o bé informació de com han anat les accions d'inserció, modificació, etc).



4.3 Funcions

Un SGBD actua d'interfície entre les bases de dades i els usuaris per a realitzar les següents funcions:

- **Definir les dades:** el dissenyador de la BD pot crear les estructures de dades, les relacions entre elles, les restriccions d'integritat, etc. És a dir: crear les taules, vistes, etc., així com la posterior modificació i/o esborrat d'aquestes estructures.
- **Manipular i consultar les dades:** un usuari o aplicació pot inserir dades (en les taules creades), modificar-les, esborrar-les i, sobretot, consultar-les d'una manera senzilla i eficient.
- **Garantir la privacitat de les dades:** un administrador o usuari pot crear usuaris i assignar-los permisos per a consultar, modificar, esborrar... certes dades de la BD.
- **Permetre la concurrència segura:** amb les transaccions s'impedeix que, per exemple, dos usuaris actualitzen alhora un saldo bancari.
- **Disposar de mecanismes de recuperació:** còpies de seguretat.

5. Sistemes d'Informació

5.1 La informació

La informació és un recurs fonamental. La investigació, la planificació i la presa de decisions exigeixen una informació de qualitat.

Criteris de qualitat de la informació:

- ✓ **Precisa:** que no tinga inexactituds.
- ✓ **Oportuna:** que es pose a disposició de l'usuari com més prompte millor.
- ✓ **Completa:** que no falte res important.
- ✓ **Significativa:** que siga útil i comprensible.
- ✓ **Coherent:** que no tinga contradiccions. Que tinga *integritat*.
- ✓ **Segura:** ha d'estar protegida pel que fa a la seua pèrdua o a accessos no autoritzats.

5.2 Els Sistemes d'Informació

És un conjunt d'elements relacionats entre sí que aporten a l'organització la informació necessària per a complir els seus fins.

El SI d'una organització ha de prendre les dades de l'entorn (la pròpia organització així com fonts externes) i ha de produir com a resultat la informació que esta organització necessita per a la seua gestió i presa de decisions.

5.3 Els Sistemes d'Informació Informàtics

Un sistema d'informació informàtic és un SI en suport informàtic.

Components d'un SI informàtic:

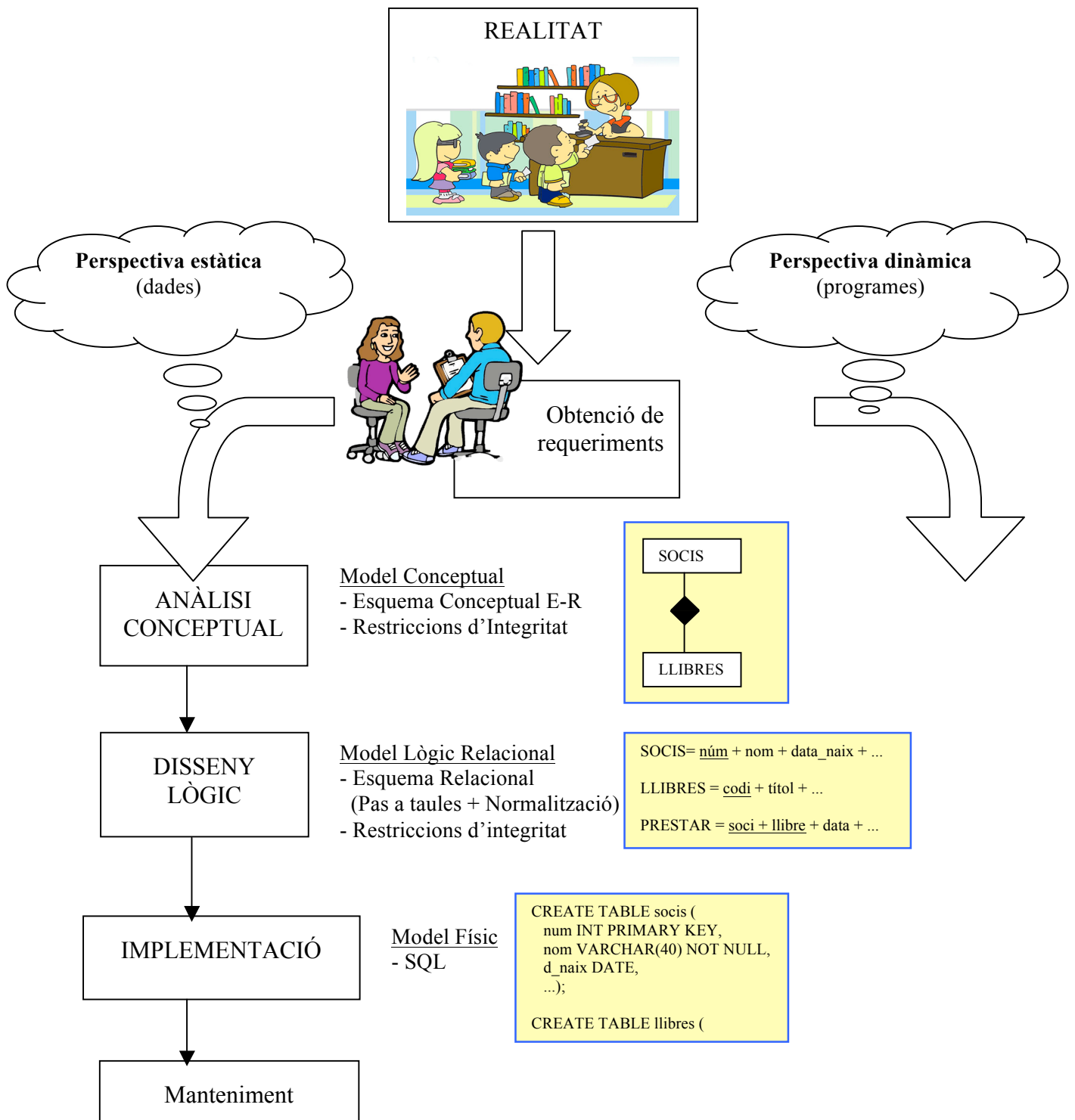
- ✓ **Components lògics:** programari o software (SW).
 - Programes: tant aplicacions finals com el propi Sistema Gestor de Bases de Dades (SGBD).
 - Informació: les dades i la seua descripció
- ✓ **Components físics:** maquinari o hardware (HW) per al suport de la informació (discos, cables, routers...)
- ✓ **Components humans:**
 - L'administrador: ha d'assegurar la qualitat i permetre l'ús correcte i permanent de les dades.
 - Els usuaris finals: introdueixen les dades, les modifiquen i les consulten.

Els SI informàtics tradicionals estaven basats en l'ús de fitxers, amb els inconvenients ja explicats. La solució a aquests problemes va consistir en els nous SI, basats en bases de dades.

5.4 Cicle de vida dels S.I. Informàtics

La vida de tota aplicació informàtica des de que es crea s'anomena "Cicle de vida", el qual consta de diferents fases que cal seguir, des dos perspectives: l'estàtica (dades) i la dinàmica (programes).

En esta assignatura només vorem la perspectiva estàtica: la de les dades. Vegem gràficament esta part del cicle de vida amb un exemple sobre la gestió d'una biblioteca:



Fase 1: Obtenció de requeriments / Anàlisi de requeriments

Consisteix en investigar quines són les dades que necessita una organització per a poder funcionar i les relacions entre elles. Això s'ha de fer amb entrevistes amb els possibles usuaris, etc.

Fase 2: Anàlisi Conceptual / Disseny Conceptual / Modelització

Consisteix en obtenir un esquema que represente les dades i les seues relacions que s'han obtés en la fase anterior. Este esquema s'anomena Esquema Conceptual. Hi ha moltes formes de fer-lo, però usarem la més coneguda, que és l'esquema Entitat-Relació (E-R). Aquells requeriments que no s'han pogut captar en eixe esquema, els anotarem apart, en el que s'anomena: Restriccions d'Integritat.

Fase 3: Disseny / Disseny lògic / Disseny relacional

Una vegada ja sabem quin SGBD usarem (PostgreSQL, MySQL...), cal traduir l'esquema de la fase anterior en un altre que s'acoste més a la implementació. Este esquema s'anomena Esquema Lògic. Hi ha diverses formes (models) d'obtenir estos esquemes, però usarem el més conegut, que és el Model Relacional (també hi ha el model jeràrquic i model en xarxa). A partir de l'esquema E-R i les Restriccions d'Integritat de la fase anterior obtindrem un Esquema Relacional. Allò que no es puga expressar en eixe esquema ho anotarem en altres Restriccions d'Integritat.

Fase 4: Implementació / Disseny físic

A partir de l'Esquema Relacional i les Restriccions d'Integritat de la fase anterior obtenim l'esquema físic de la BD: les instruccions SQL que permetran crear les estructures on es guardaran les dades que necessita l'organització. Aquelles RI que no es puguin expressar en SQL s'hauran de controlar en la part dels programes.

Fase 5: Manteniment

Una vegada ja està en funcionament la BD en l'organització, al llarg del temps eixiran aspectes a millorar o a adaptar a noves circumstàncies. Este fet farà tornar a alguna de les fases anteriors i continuar amb el cicle de vida.

En este mòdul tractarem l'anàlisi, disseny i implementació de BD (fases 2 a 4).