

M02 – Base de dades

# Introducció als Sistemes Gestors de Bases de Dades (SGBD)

UF1 – Introducció a les bases de dades

Laura Villalba Guerrero  
18/09/2017

## Índex

1	Introducció a les bases de dades i als sistemes gestors de base de dades .....	3
2	Arquitectura dels sistemes de base de dades.....	5
3	Components del SGBD.....	8
3.1	Llenguatges dels SGBD .....	8
3.2	El diccionari de dades.....	9
3.3	Seguretat i integritat de dades .....	9
3.4	L'administrador de la BD .....	10
4	Model de dades .....	11
4.1.1	Models lògics basats en objectes .....	13
4.1.2	Models lògics basats en registres .....	14
4.1.3	Models físics de dades .....	14
5	El model de dades entitat-interrelació.....	14
6	Model de xarxa.....	15
6.1	Diagrames d'estructura de dades en un model en xarxa.....	15
7	Model Jeràrquic.....	15
8	Model Orientat o Objectes .....	16
9	Conceptes bàsics .....	18

# 1 Introducció a les bases de dades i als sistemes gestors de base de dades

Les dades que s'utilitzen de manera informatitzada s'emmagatzemen, habitualment, en bases de dades (BD).

El programari que serveix per gestionar-les, són els sistemes gestors de bases de dades (SGBD), és imprescindible entendre prèviament uns quants conceptes teòrics fonamentals, referents a les dades i a la seva representació. A banda, doncs, dels conceptes relatius a les dades i a les bases de dades, caldrà conèixer quines representacions de la informació s'utilitzen habitualment així com l'evolució del programari d'aquest àmbit.

Definim un **Sistema Gestor de Bases de dades o SGBD**, també anomenat **DBMS (Data Base Management System)** com una col·lecció de dades relacionades entre si, estructurats i organitzats, i un conjunt de programes que accedeixen i gestionen aquestes dades. La col·lecció d'aquestes dades es denomina Base de dades o BD, (DB Data Base).

Abans d'aparèixer els SGBD (dècada dels setanta), la informació es tractava i es gestionava utilitzant els típics sistemes de gestió d'arxius que anaven suportats sobre un sistema operatiu. Aquests consistien en un conjunt de programes que definien i treballaven les seves pròpies dades. Les dades s'emmagatzemen en arxius i els programes manegen aquests arxius per obtenir la informació. Si l'estructura de les dades dels arxius canvia, tots els programes que els manegen s'han de modificar; per exemple, un programa treballa amb un arxiu de dades d'alumnes, amb una estructura o registre ja definit; si s'incorporen elements o camps a l'estructura de l'arxiu, els programes que utilitzen aquest arxiu s'han de modificar per tractar aquests nous elements. En aquests sistemes de gestió d'arxius, la definició de les dades es troba codificada dins dels programes d'aplicació en lloc d'emmagatzemar-se de forma independent, i a més el control de l'accés i la manipulació de les dades ve impost pels programes d'aplicació.

Això suposa un gran inconvenient a l'hora de tractar grans volums d'informació. Sorgeix així la idea de separar les dades contingudes en els arxius dels programes que els manipulen, és a dir, que es pugui modificar l'estructura de les dades dels arxius sense que per això s'hagin de modificar els programes amb els quals treballen. Es tracta d'estructurar i organitzar les dades de manera que es pugui accedir a ells amb independència dels programes que els gestionen.

Inconvenients d'un sistema de gestió d'arxius:

- **Redundància i inconsistència de les dades**, es produeix perquè els arxius són creats per diferents programes i van canviant al llarg del temps, és a dir, poden tenir diferents formats i les dades poden estar duplicats en diversos llocs. Per exemple, el telèfon d'un alumne pot aparèixer en més d'un arxiu. La redundància augmenta els costos d'emmagatzematge i accés, i porta amb si la inconsistència de les dades: les còpies de les mateixes dades no coincideixen per aparèixer en diversos arxius.
- **Dependència de les dades física-lògica**, o el que és el mateix, l'estructura física de les dades (definició d'arxius i registres) es troba codificada als programes d'aplicació. Qualsevol canvi en aquesta estructura implica al programador identificar, modificar i provar tots els programes que manipulen aquests arxius.

- **Dificultat per tenir accés a les dades**, proliferació de programes, és a dir, cada vegada que es necessiti una consulta que no va ser prevista en l'inici implica la necessitat de codificar el programa d'aplicació necessari. El que es tracta de provar és que els entorns convencionals de processament d'arxius no permeten recuperar les dades necessàries d'una forma convenient i eficient.
- **Separació i aïllament de les dades**, és a dir, en estar repartits en diversos arxius, i tenir diferents formats, és difícil escriure nous programes que assegurin la manipulació de les dades correctes. Abans s'haurien de sincronitzar tots els arxius perquè les dades coincidissin.
- **Dificultat per a l'accés concurrent**, doncs en un sistema de gestió d'arxius és complicat que els usuaris actualitzin les dades simultàniament. Les actualitzacions concurrents poden donar per resultat dades inconsistents, ja que es pot accedir a les dades per mitjà de diversos programes d'aplicació.
- **Dependència de l'estructura de l'arxiu amb el llenguatge de programació**, doncs l'estructura es defineix dins dels programes. Això implica que els formats dels arxius siguin incompatibles. La incompatibilitat entre arxius generats per diferents llenguatges fa que les dades siguin difícils de processar.
- **Problemes en la seguretat de les dades**. Resulta difícil implantar restriccions de seguretat doncs les aplicacions es van afegint al sistema segons es van necessitant.
- **Problemes d'integritat de dades**, és a dir, els valors emmagatzemats en els arxius han de complir amb restriccions de consistència. Per exemple, no es pot inserir una nota d'un alumne en una assignatura si prèviament aquesta assignatura no està creada. Un altre exemple, les unitats en magatzem d'un producte determinat no han de ser inferiors a una quantitat. Això implica afegir gran nombre de línies de codi als programes. El problema es complica quan existeixen restriccions que impliquen diverses dades en diferents arxius.

Tots aquests inconvenients fan possible el foment i desenvolupament de SGBD. L'objectiu primordial d'un gestor és proporcionar eficiència i seguretat a l'hora d'extreure o emmagatzemar informació en les BD.

Els sistemes gestors de BBDD estan dissenyats per gestionar grans blocs d'informació, que implica tant la definició d'estructures per a l'emmagatzematge com de mecanismes per a la gestió de la informació.

Una BD és un gran magatzem de dades que es defineix una sola vegada; les dades poden ser accedits de forma simultània per diversos usuaris; estan relacionats i existeix un nombre mínim de duplicitat; a més en les BBDD s'emmagatzemaran les descripcions d'aquestes dades, la qual cosa es diu metadades en el diccionari de dades, que es veurà més endavant.

El SGBD és una aplicació que permet als usuaris definir, crear i mantenir la BD i proporciona un accés controlat a la mateixa. Ha de prestar els següents serveis:

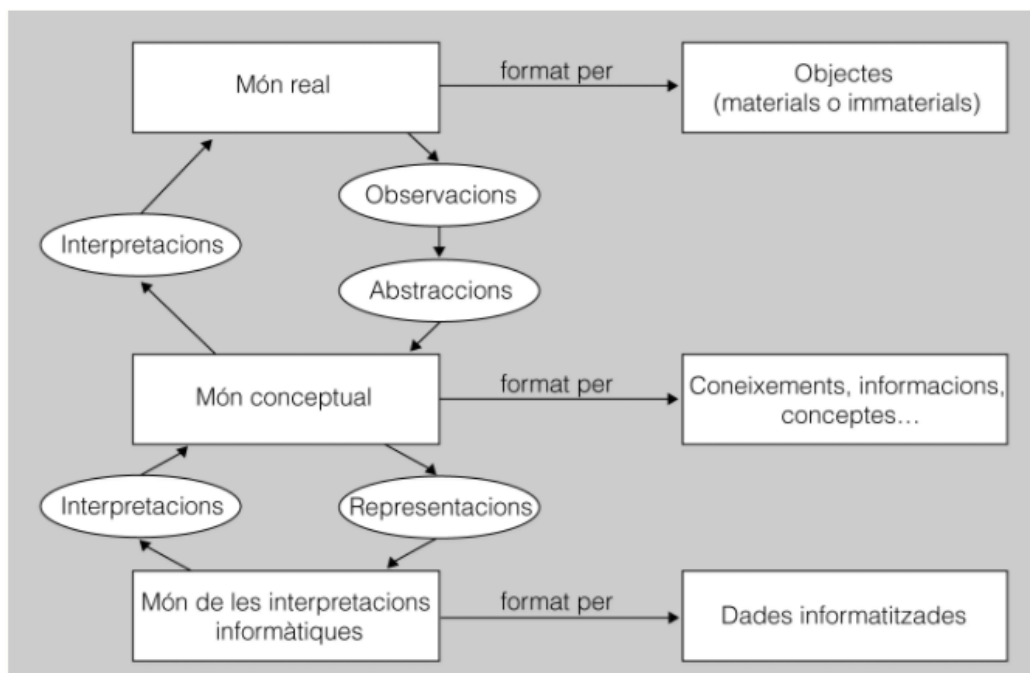
- **Creació i definició de la BD**: especificació de l'estructura, el tipus de les dades, les restriccions i relacions entre ells mitjançant llenguatges de definició de dades. Tota aquesta

informació s'emmagatzema en el diccionari de dades, el SGBD proporcionarà mecanismes per a la gestió del diccionari de dades.

- **Manipulació de les dades** realitzant consultes, insercions i actualitzacions dels mateixos utilitzant llenguatges de manipulació de dades.
- **Accés controlat** a les dades de la BD mitjançant mecanismes de seguretat d'accés als usuaris.
- **Mantenir la integritat i consistència de les dades** utilitzant mecanismes per evitar que les dades siguin perjudicades per canvis no autoritzats.
- **Accés compartit a la BD**, controlant la interacció entre usuaris concurrents.
- **Mecanismes de backup i recuperació** per restablir la informació en cas de fallades en el sistema.

## 2 Arquitectura dels sistemes de base de dades

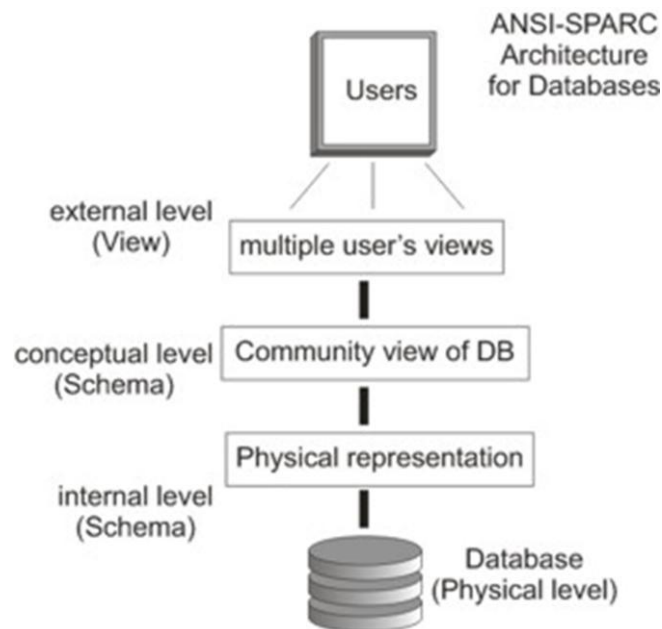
En 1975, el comitè **ANSI-SPARC** (*American National Standard Institute - Standards Planning and Requirements Committee*) va proposar una arquitectura de tres nivells pels SGBD que el seu objectiu principal era el de separar els programes d'aplicació de la BD física. En aquesta arquitectura l'esquema d'una BD es defineix en tres nivells d'abstracció diferents:



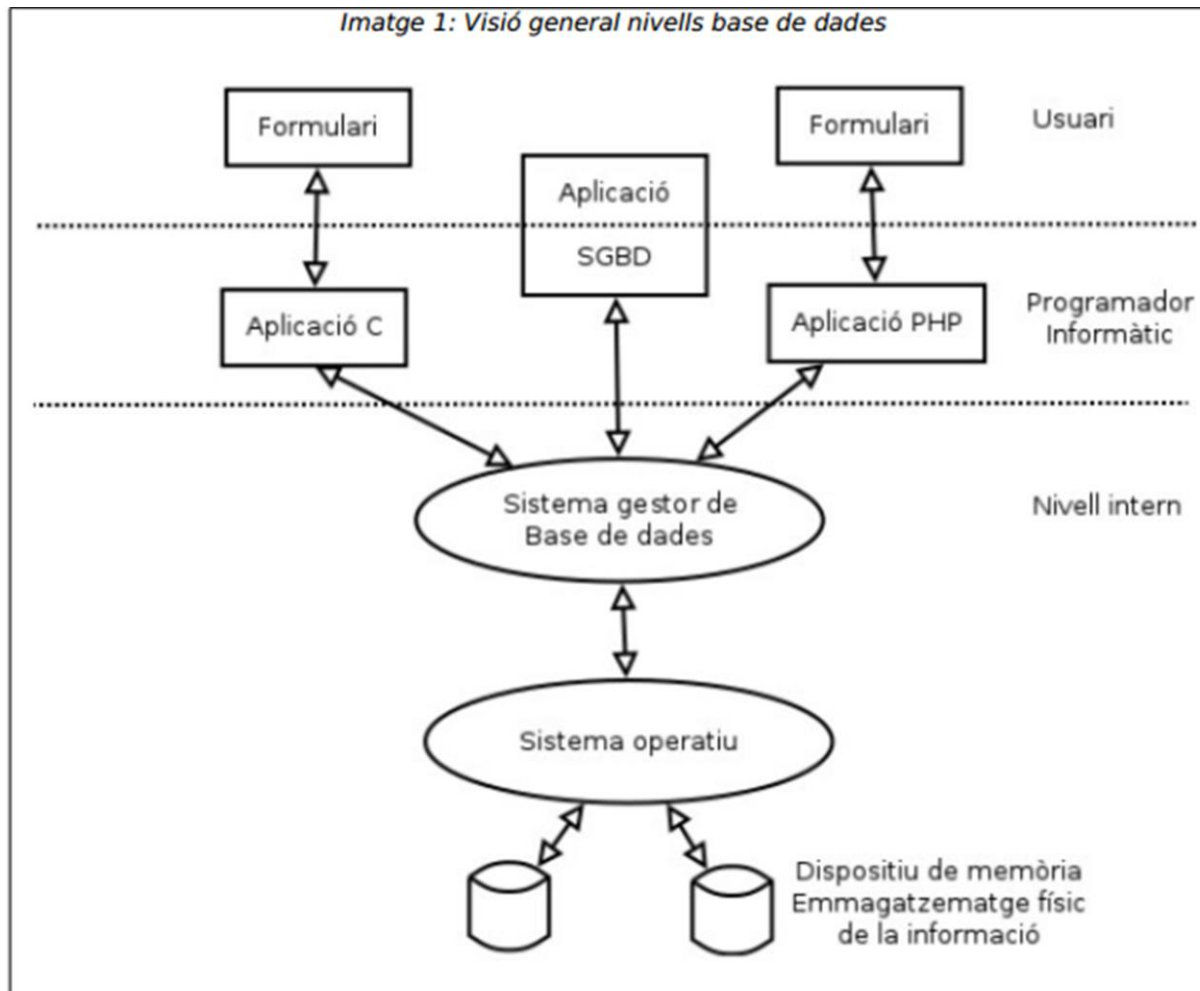
- **Nivell intern o físic:** el més proper a l'emmagatzematge físic, és a dir, tal com estan emmagatzemats en l'ordinador. Descriu l'estructura física de la BD mitjançant un esquema intern. Aquest esquema s'especifica amb un model físic i descriu els detalls de com s'emmagatzemen físicament les dades: els arxius que contenen la informació, la seva organització, els mètodes d'accés als registres, els tipus de registres, la longitud, els camps que els componen, etcètera.

- **Nivell extern o de visió:** és el més proper als usuaris, és a dir, és on es descriuen diversos esquemes externs o vistes d'usuaris. En aquest nivell es determina com s'han de mostrar les dades a l'usuari. La visió que tingui l'usuari de les dades no ha de coincidir a l'estructura de les dades en la base de dades. Així, per exemple, un usuari pot veure una part de les dades o pot veure dades de diferents orígens com a una unitat.
- **Nivell conceptual:** descriu l'estructura de tota la BD per a un grup d'usuaris mitjançant un esquema conceptual, resumint, és tal com estan emmagatzemades les dades en el SGBD. Aquest esquema descriu les entitats, atributs, relacions, operacions dels usuaris i restriccions, ocultant els detalls de les estructures físiques d'emmagatzematge. Representa la informació continguda en la BD.

En la següent figura es representen els nivells d'abstracció de l'arquitectura ANSI.



Aquesta arquitectura descriu les dades a tres nivells d'abstracció. En realitat les úniques dades que existeixen estan a nivell físic emmagatzemats en discos o altres dispositius. Els SGBD basats en aquesta arquitectura permeten que cada grup d'usuaris faci referència al seu propi esquema extern. El SGBD deu transformar qualsevol petició d'usuari (esquema extern) a una petició expressada en termes d'esquema conceptual, per finalment ser una petició expressada en l'esquema intern que es processarà sobre la BD emmagatzemada. El procés de transformar peticions i resultats d'un nivell a un altre es denomina correspondència o transformació. La següent imatge mostra el correut que ha de realitzar una petició dins els nivells de la base de dades.



Per una **BD específica només hi ha un esquema intern i un de conceptual**, però pot haver-hi diversos esquemes externs definits per a un o per a diversos usuaris.

Amb l'arquitectura a tres nivells s'introdueix el concepte d'independència de dades, es defineixen dos tipus d'independència:

- **Independència lògica**, la capacitat de modificar l'esquema conceptual sense haver d'alterar els esquemes externs ni els programes d'aplicació. Es podrà modificar l'esquema conceptual per ampliar la BD o per reduir-la, per exemple, si s'elimina una entitat, els esquemes externs que no es refereixin a ella no es veuran afectats.
- **Independència física**, la capacitat de modificar l'esquema intern sense haver d'alterar ni l'esquema conceptual, ni els externs. Per exemple, es poden reorganitzar els arxius físics amb la finalitat de millorar el rendiment de les operacions de consulta o d'actualització, o es poden afegir nous arxius de dades perquè els que hi havia s'han omplert. La independència física és més fàcil d'aconseguir que la lògica, doncs es refereix a la separació entre les aplicacions i les estructures físiques d'emmagatzematge.

En els SGBD basats en arquitectures de diversos nivells, es fa necessari ampliar el catàleg o el diccionari de dades per incloure la informació sobre com establir les correspondències entre les peticions dels usuaris i les dades, entre els diversos nivells.

La independència de les dades s'aconsegueix perquè en modificar-se l'esquema en algun nivell, l'esquema del nivell immediat superior roman sense canvis. Només es modifica la correspondència entre els dos nivells. No cal modificar els programes d'aplicació que fan referència a l'esquema del nivell superior.

No obstant això, els dos nivells de correspondència impliquen una despesa de recursos durant l'execució d'una consulta o d'un programa, la qual cosa redueix l'eficiència del SGBD. Per aquesta raó pocs SGBD han implementat l'arquitectura completa.

## 3 Components del SGBD

Tot i que molts SGBD incorporen eines que ens permeten realitzar tasques més o menys complexes amb les bases de dades, tots ells incorporen una sèrie de llenguatges estàndards que ens permeten interactuar amb el sistema. El coneixement d'aquests llenguatges ens permeten interactuar amb qualsevol SGBD i no dependre de quines utilitats gràfiques hi ha instal·lades o disposa el SGBD.

### 3.1 Llenguatges dels SGBD

Els SGBD incorporen tres llenguatges diferents, cada un d'ells desenvolupat amb una finalitat determinada.

Els llenguatges van a permetre a l'administrador de la BD especificar les dades que componen la BD, la seva estructura, les relacions que existeixen entre ells, les regles d'integritat, els controls d'accés, les característiques de tipus físic i les vistes externes dels usuaris. Els llenguatges del SGBD es classifiquen en:

- **Llenguatge de definició de dades (*Data Definition Language (DDL)*)**, és el llenguatge que permet crear la pròpia base de dades, amb les estructures necessàries per a poder emmagatzemar la informació de forma òptima. Tots els elements que formen part de l'estructura de dades de la base de dades, així com la definició d'aquests elements, són creats i mantinguts mitjançant les instruccions disponibles en el llenguatge DDL.
- **Llenguatge de control de dades (*Data control Language (DCL)*)**, és un llenguatge de control que permet:
  - Crear i gestionar els usuaris: crear, bloquejar, canviar paraules claus...
  - Gestionar els seus accessos.
  - Gestionar els seus privilegis.
- **Llenguatge de manipulació de dades (*Data manipulation language (DML)*)**, el llenguatge de manipulació de dades permet la introducció d'informació a la base de dades, la consulta d'aquesta informació i, finalment, la seva modificació o eliminació. Els hi ha procedurals, en els quals l'usuari serà normalment un programador i especifica les operacions d'accés a les dades cridant als procediments necessaris.
- **Llenguatges de 4a generació (4GL)**, que permeten a l'usuari desenvolupar aplicacions de forma fàcil i ràpida, també se'ls crida eines de desenvolupament. Exemples d'això són les eines del SGBD ORACLE:
  - SQL Forms per a la generació de formularis de pantalla i per interactuar amb les dades;



- SQL Reports per generar informes de les dades contingudes en la BD;
- PL/SQL llenguatge per crear procediments que interactuen amb les dades de la BD.

### 3.2 El diccionari de dades

El diccionari de dades és el lloc on es diposita informació sobre totes les dades que formen la BD. És una guia en la qual es descriu la BD i els objectes que la formen.

El diccionari conté les característiques lògiques dels llocs on s'emmagatzemen les dades del sistema, incloent nom, descripció, àlies, contingut i organització. Identifica els processos on s'empren les dades i els llocs on es necessita l'accés immediat a la informació.

Dins una BD relacional, el diccionari proporciona la següent informació:

- L'estructura lògica i física de la BD.
- La definició de tots els objectes de la BD: vistes, taules, índex, disparadors, procediments, funcions, etc.
- L'espai assignat i utilitzat pels objectes.
- Els valors per defecte de les columnes de les taules.
- Informació sobre les restriccions d'integritat.
- Els privilegis i rols atorgats als usuaris.
- Auditoria d'informació, com els accessos als objectes.
- Un diccionari de dades ha de complir les següents característiques:
- Ha de suportar les descripcions dels models conceptual, lògic, intern i extern de la BD.
- Ha d'estar integrat dins del SGBD.
- Ha de recolzar la transferència eficient d'informació al SGDB. La connexió entre els models intern i extern ha de ser realitzada en temps d'execució.
- Ha de començar amb la reorganització de versions de producció de la BD. A més ha de reflectir els canvis en la descripció de la BD. Qualsevol canvi a la descripció de programes ha de ser reflectit automàticament en la llibreria de descripció de programes amb l'ajuda del diccionari de dades.
- Ha d'estar emmagatzemat en un mitjà d'emmagatzematge amb accés directe per a la fàcil recuperació d'informació.

### 3.3 Seguretat i integritat de dades

Un SGBD proporciona els següents mecanismes per garantir la seguretat i integritat de les dades:

- Ha de garantir la protecció de les dades contra accessos no autoritzats, tant intencionats com a accidentals. Ha de controlar que només els usuaris autoritzats accedeixin a la BD.
- Els SGBD ofereixen mecanismes per implantar restriccions d'integritat en la BD. Aquestes restriccions van a protegir la BD contra danys accidentals. Els valors de les dades que s'emmagatzemen han de satisfer certs tipus de restriccions de consistència i regles d'integritat, que especificarà l'administrador de la BD. El SGBD pot determinar si es produeix una violació de la restricció.
- Proporciona eines i mecanismes per a la planificació i realització de còpies de seguretat i restauració.

- Ha de ser capaç de recuperar la BD portant-la a un estat consistent en cas d'ocórrer algun succés que la danyi.
- Ha d'assegurar l'accés concurrent i oferir mecanismes per conservar la consistència de les dades en el cas que diversos usuaris actualitzin la BD de forma concurrent.

### 3.4 L'administrador de la BD

En els sistemes de gestió de BBDD actuals existeixen diferents categories d'usuaris. Aquestes categories es caracteritzen perquè cadascuna d'elles té una sèrie de privilegis o permisos sobre els objectes que formen la BD.

En els sistemes Oracle les categories més importants són:

- Els **usuaris de la categoria DBA (*Database Administrator*)**, la funció de la qual és precisament administrar la base i que tenen, el nivell més alt de privilegis.
- Els **usuaris de la categoria RESOURCE**, que poden crear els seus propis objectes i tenen accés als objectes pels quals se'ls ha concedit permís.
- Els **usuaris del tipus CONNECT**, que solament poden utilitzar aquells objectes pels quals se'ls ha concedit permís d'accés.

El DBA té una gran responsabilitat ja que posseeix el màxim nivell de privilegis. Serà l'encarregat de crear els usuaris que es connectaran a la BD. En l'administració d'una BD sempre cal procurar que hi hagi el menor nombre d'administradors, a ser possible una sola persona.

L'objectiu principal d'un DBA és garantir que la BD compleix les finalitats previstes per l'organització, la qual cosa inclou una sèrie de tasques com:

- Instal·lar SGBD en el sistema informàtic.
- Crear les BBDD que es vagin a gestionar.
- Crear i mantenir l'esquema de la BD.
- Crear i mantenir els comptes d'usuari de la BD.
- Arrencar i parar SGBD, i carregar les BBDD amb les quals s'ha de treballar.
- Col·laborar amb l'administrador del S.O. en les tasques d'ubicació, dimensionament i control dels arxius i espais de disc ocupats pel SGBD.
- Col·laborar en les tasques de formació d'usuaris.
- Establir estàndards d'ús, polítiques d'accés i protocols de treball diari per als usuaris de la BD.

- Subministrar la informació necessària sobre la BD als equips d'anàlisis i programació d'aplicacions.
- Efectuar tasques d'explotació com:
  - Vigilar el treball diari col·laborant en la informació i resolució dels dubtes dels usuaris de la BD.
  - Controlar en temps real els accessos, taxes d'ús, càrregues en els servidors, anomalies, etcètera.
  - Arribat el cas, reorganitzar la BD.
  - Efectuar les còpies de seguretat periòdiques de la BD.
  - Restaurar la BD després d'un incident material a partir de les còpies de seguretat.
  - Estudiar les auditories del sistema per detectar anomalies, intents de violació de la seguretat, etcètera.
  - Ajustar i optimitzar la BD mitjançant l'ajust dels seus paràmetres, i amb ajuda de les eines de monitoratge i de les estadístiques del sistema.

En la seva gestió diària, el DBA sol utilitzar una sèrie d'eines d'administració de la BD.

Amb el pas del temps, aquestes eines han adquirit sofisticades prestacions i faciliten en gran manera la realització de treballs que, fins a no fa massa, requerien d'ardus esforços per part dels administradors.

## 4 Model de dades

Un dels objectius més importants d'un SGBD és proporcionar als usuaris una visió abstracta de les dades, és a dir, l'usuari va a utilitzar aquestes dades però no tindrà idea de com estan emmagatzemats físicament.

Els models de dades són l'instrument principal per oferir aquesta abstracció. Són utilitzats per a la representació i el tractament dels problemes. Formen el problema a tres nivells d'abstracció, relacionats amb l'arquitectura ANSI-SPARC de tres nivells pels SGBD:

- **Nivell físic:** el nivell més baix d'abstracció; descriu com s'emmagatzemen realment les dades.
- **Nivell lògic o conceptual:** descriu les dades que s'emmagatzemen en la BD i les seves relacions, és a dir, els objectes del món real, els seus atributs i les seves propietats, i les relacions entre ells.
- **Nivell extern o de vistes:** descriu la part de la BD a la qual els usuaris poden accedir.

Per fer-nos una idea dels tres nivells d'abstracció, ens imaginem un arxiu d'articles amb el següent registre:

```
struct ARTICULOS
{
    int Cod;
    char Deno[15];
    int cant_almacen;
    int cant_minima;
    int uni_vendidas;
    float PVP;
    char reponer;
    struct VENTAS Tventas[12];
};
```

El **nivell físic** és el conjunt de bytes que es troben emmagatzemats en l'arxiu en un dispositiu magnètic, que pot ser un disc, una pista a un sector determinat.

El **nivell lògic** comprèn la descripció i la relació amb altres registres que es fa del registre dins d'un programa, en un llenguatge de programació.

L'últim nivell d'abstracció, l'**extern**, és la visió d'aquestes dades que té un usuari quan executa aplicacions que operen amb ells, l'usuari no sap el detall de les dades, unes vegades operarà amb uns i unes altres amb uns altres, depenent de l'aplicació.

Si traslладem l'exemple a una BD relacional específica hi haurà, com en el cas anterior, un únic nivell intern i un únic nivell lògic o conceptual, però pot haver-hi diversos nivells externs, cadascun definit per a un o per a diversos usuaris. Podria ser el següent:

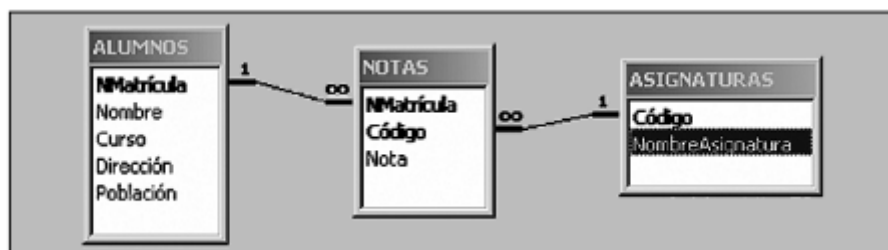
Curso	Nombre	Nombre de asignatura	Nota
1	Ana	Programación en lenguajes estructurados	6
1	Ana	Sistemas informáticos multiusuario y en red	8
2	Rosa	Desa. de aplic. en entornos de 4.ª Generación y H. Case	5
2	Juan	Desa. de aplic. en entornos de 4.ª Generación y H. Case	7
1	Alicia	Programación en lenguajes estructurados	5
1	Alicia	Sistemas informáticos multiusuario y en red	4

La imatge anterior mostra la vista d'una BD per un usuari.

- **Nivell extern:** Visió parcial de les taules de la BD segons l'usuari. Per exemple, la vista que es mostra en la següent taula obté el llistat de notes d'alumnes amb les següents dades: Curs, Nom, Nom d'assignatura i Nota.
- **Nivell lògic i conceptual:** Definició de totes les taules, columnes, restriccions, claus i relacions. En aquest exemple, disposem de tres taules que estan relacionades:

- Taula ALUMNES. Columnes: NMatrícula, Nom, Curs, Adreça, Població. Clau: NMatrícula. A més té una relació amb NOTES, doncs un alumne pot tenir notes en diverses assignatures.
- Taula ASSIGNATURES. Columnes: Codi, Nom d'assignatura. Clau: Codi. Està relacionada amb NOTES, doncs per a una assignatura hi ha diverses notes, tantes com a alumnes la cursin.
- Taula NOTES. Columnes: NMatrícula, Codi, Nota. Està relacionada amb ALUMNES i ASSIGNATURES, doncs un alumne té notes en diverses assignatures, i d'una assignatura existeixen diverses notes, tantes com a alumnes.

Podem representar les relacions de les taules en el nivell lògic com es mostra en la següent figura, que representa les relacions entre les taules a nivell lògic:



- **Nivell intern:** En una BD les taules s'emmagatzemen en arxius de dades de la BD. Si hi ha claus, es creen índexs per accedir a les dades, tot això contingut en el disc dur, en una pista i en un sector, que només el SGBD coneix. Davant una petició, sap a quina pista, a quin sector, a quin arxiu de dades i a quins índexs accedir.

Per a la representació d'aquests nivells s'utilitzen els models de dades. Es defineixen com el conjunt de conceptes o eines conceptuals que serveixen per descriure la estructura d'una BD: les dades, les relacions i les restriccions que s'han de complir sobre les dades. Es denomina **esquema de la BD** a la descripció d'una BD mitjançant un model de dades. Aquest esquema s'especifica durant el disseny de la mateixa.

Podem dividir els models en tres grups: models lògics basats en objectes, models lògics basats en registres i models físics de dades. Cada SGBD suporta un model lògic.

#### 4.1.1 Models lògics basats en objectes

Els models lògics basats en objectes s'usen per descriure dades en el nivell conceptual i l'extern. Es caracteritzen perquè proporcionen capacitat d'estructuració bastant flexible i permeten especificar restriccions de dades. Els models més coneguts són el model entitat-relació i l'orientat a objectes.

Actualment, el més utilitzat és el model entitat-relació, encara que el model orientat a objectes inclou molts conceptes de l'anterior, i a poc a poc està guanyant mercat.

La majoria de les BBDD relacionals afegeixen extensions per poder ser relacionals-orientades a objectes.

### 4.1.2 Models lògics basats en registres

Els models lògics basats en registres s'utilitzen per descriure les dades en els models conceptual i físic. A diferència dels models lògics basats en objectes, s'usen per especificar l'estructura lògica global de la BD i per proporcionar una descripció a nivell més alt de la implementació.

Els models basats en registres es diuen així perquè la BD està estructurada en registres de format fix de diversos tipus. Cada tipus de registre defineix un nombre fix de camps, o atributs, i cada camp normalment és de longitud fixa. L'estructura més rica d'aquestes BBDD sovint porta a registres de longitud variable en el nivell físic.

Els models basats en registres no inclouen un mecanisme per a la representació directa de codi de la BD, en canvi, hi ha llenguatges separats que s'associen amb el model per expressar consultes i actualitzacions. Els tres models de dades més acceptats són els models relacional, de xarxa i jeràrquic. El model relacional ha guanyat acceptació per sobre dels altres; representa les dades i les relacions entre les dades mitjançant una col·lecció de taules, les columnes de les quals tenen noms únics, les files (tuples) representen als registres i les columnes representen les característiques (atributs) de cada registre. Aquest model s'estudiarà en la següent Unitat.

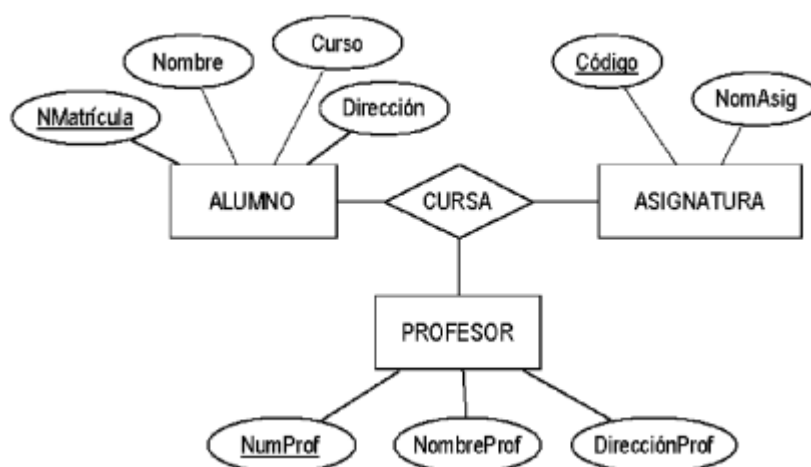
### 4.1.3 Models físics de dades

Els models físics de dades s'usen per descriure com s'emmagatzemen les dades en l'ordinador: format de registres, estructures dels arxius, mètodes d'accés, etcètera. Hi ha molt pocs models físics de dades en ús, sent els més coneguts el model unificador i de memòria d'elements.

## 5 El model de dades entitat-interrelació

El model de dades, també entitat-relació (E-R), va ser proposat per Peter Chen en 1976 per a la representació conceptual dels problemes del món real. En 1988, el ANSI ho va seleccionar com a model estàndard per als sistemes de diccionaris de recursos d'informació. És un model molt estès i potent per a la representació de les dades. Se simbolitza fent ús de grafs i de taules. Proposa l'ús de taules bidimensionals per a la representació de les dades i les seves relacions.

La següent imatge és una representació del model entitat-relació:



## 6 Model de xarxa

Aquest model utilitza estructures de dades en xarxa, també conegudes com a estructures plex.

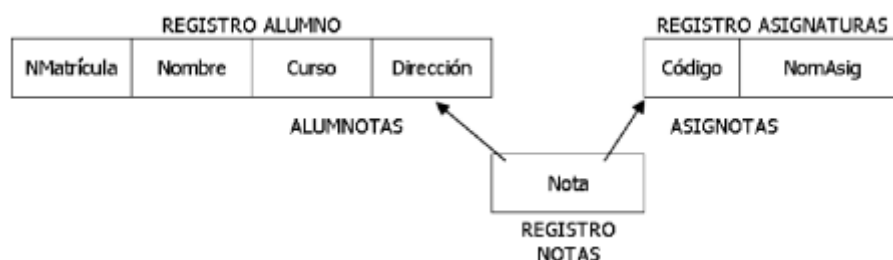
Les entitats es representen com a registres o nodes, i les relacions com a enllaços o punters. En una estructura xarxa qualsevol component pot vincular-se amb qualsevol un altre. És possible descriure-la en termes de pares i fills, però, a diferència del model jeràrquic, un fill pot tenir diversos pares.

Les representacions lògiques basades en arbres o en estructures plex, sovint, limiten el canvi que el creixement de la BD exigeix, fins a tal punt que les representacions lògiques de les dades poden variar afectant als programes d'aplicació que usen aquestes dades.

### 6.1 Diagrames d'estructura de dades en un model en xarxa

El diagrama és l'esquema que representa el disseny d'una BD de xarxa. Es basa en representacions entre registres per mitjà d'enllaços. Existeixen relacions en les quals participen solament dues entitats (binàries) i relacions en les quals participen més de dues entitats (generals) ja sigui amb o sense atribut descriptiu en la relació. S'utilitzen quadres o cel·les per representar els registres i línies per representar els enllaços.

La següent imatge és una representació del model en xarxa:

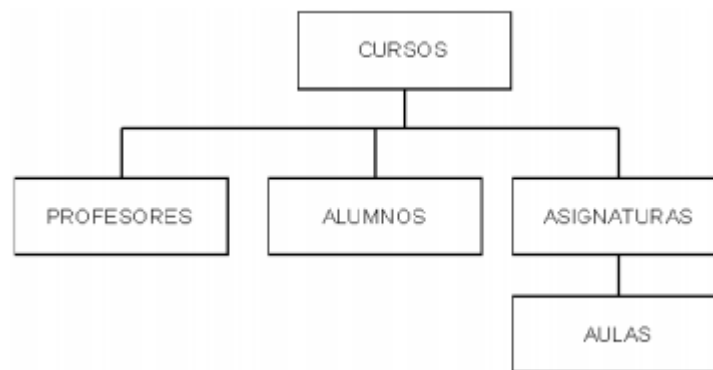


## 7 Model Jeràrquic

El model jeràrquic és similar al model de xarxa. Les dades i les relacions es representen mitjançant registres i enllaços. Es diferencia del model de xarxa en què els registres estan organitzats com a col·leccions d'arbres. El model jeràrquic se serveix d'arbres per a la representació lògica de les dades, la seva implementació es duu a terme mitjançant arbres i punters.

La representació gràfica del model jeràrquic es realitza mitjançant un arbre invertit, en el qual el nivell superior està format per una única entitat o segment sota el qual pengen la resta d'entitats o segments en nivells que es van ramificant. Els diferents nivells queden units per mitjà de les relacions. El nivell més alt de la jerarquia té un node que es diu arrel. Cada node representa un tipus de registre anomenat segment amb els seus corresponents camps. Els segments s'organitzen de manera que en un mateix nivell estan tots aquells que depenen d'un segment de nivell immediatament superior.

La següent imatge és una representació del model en xarxa:



## 8 Model Orientat o Objectes

El model de dades orientat a objectes sorgeix per les limitacions del model relacional, sobretot a l'hora d'abordar tipus de dades més complexes, i per la falta de capacitat semàntica del model relacional per desenvolupar aplicacions en àrees com el disseny assistit per ordinador, l'enginyeria del programari, els sistemes basats en el coneixement i el tractament de documents, multimèdia i gestió de xarxes, que requereixen modelar objectes i interrelacions més complexes.

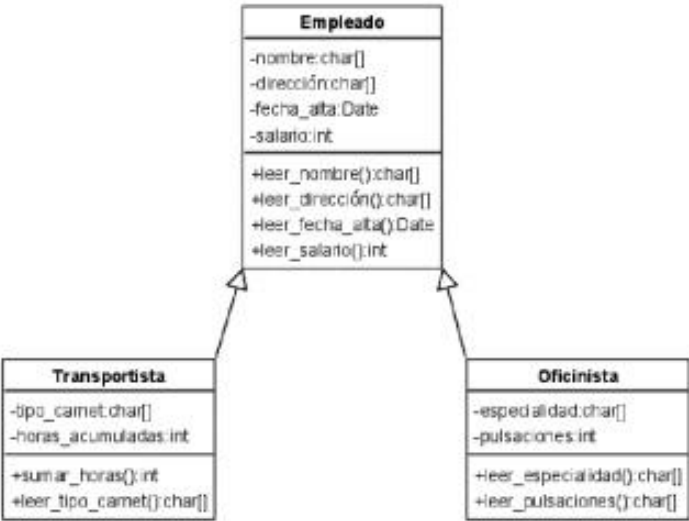
Aquest model està basat en el paradigma de la programació orientada a objectes (POO).

Els SGBD (Sistemes de Gestió de Bases d'Objectes) o SDBDOO (Sistemes de Gestió de Bases de dades Orientats a Objectes) gestionen objectes en els quals estan encapsulats les dades i les operacions que interactuen amb ells. A més proporcionen un model únic de dades. No hi ha diferència entre el model conceptual (el model E-R) i el model lògic (el relacional), i les aplicacions poden accedir directament al model.

Les SGBDOO adopten una arquitectura que consta d'un sistema de gestió que suporta un llenguatge de BBDD orientat a objectes, amb una sintaxi similar a un llenguatge de programació també orientat a objectes, com pot ser C++ o Java. El llenguatge de BBDD és especificat mitjançant un llenguatge de definició de dades (ODL), un llenguatge de manipulació de dades (OML), i un llenguatge de consulta (OQL), sent tots ells portables a altres sistemes.

La següent imatge representa la generalització d'una classe "empleat":





## 9 Conceptes bàsics

- **SGBD o DBMS (Sistema Gestor de Bases de dades):** col·lecció de dades estructurades, organitzats i relacionats entre si, i el conjunt de programes que accedeixen i gestionen aquestes dades.
- **ANSI-SPARC (American National Standard Institute – Standards Planning and Requirements ommittee):** comitè que va proposar una arquitectura de tres nivells pels SGBD que el seu objectiu principal és el de separar els programes d'aplicació de la BD física.
- **Diccionari de dades:** és el lloc on es disposa informació sobre totes les dades que formen la BD. Conté les característiques lògiques dels llocs on s'emmagatzemen les dades del sistema, incloent nom, descripció, àlies, contingut i organització. Identifica també els processos on s'empren les dades i els llocs on es necessita l'accés immediat a la informació.
- **Models de dades:** l'instrument principal per oferir l'abstracció de les dades, s'utilitzen per a la representació i el tractament dels problemes i els representen a tres nivells d'abstracció. En la següent figura podem veure la relació entre els models lògics i conceptuals de dades

