

Tema 1: Introducció a les BDs

DAM1

Curs 2023-24

Conceptes previs

Base de dades:

Es un conjunt de dades organitzades de forma estructurada i ordenada dins un mateix àmbit per ser utilitzades en un posterior ús.

Un sistema gestor de bases de dades (SGBD) és el programari que permet la definició, manipulació, obtenció i gestió de les dades emmagatzemades en una DB.



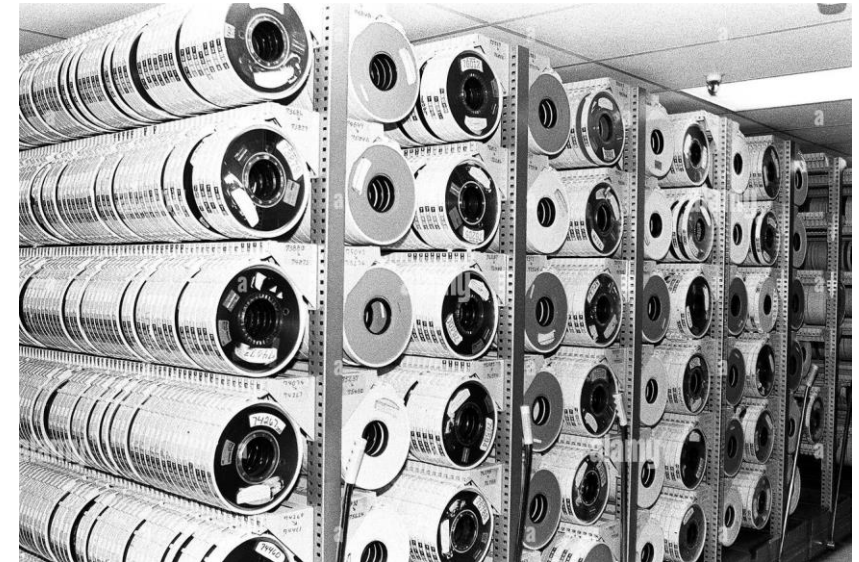
- Emmagatzematge en targetes perforades



Evolució Històrica de les base de dades

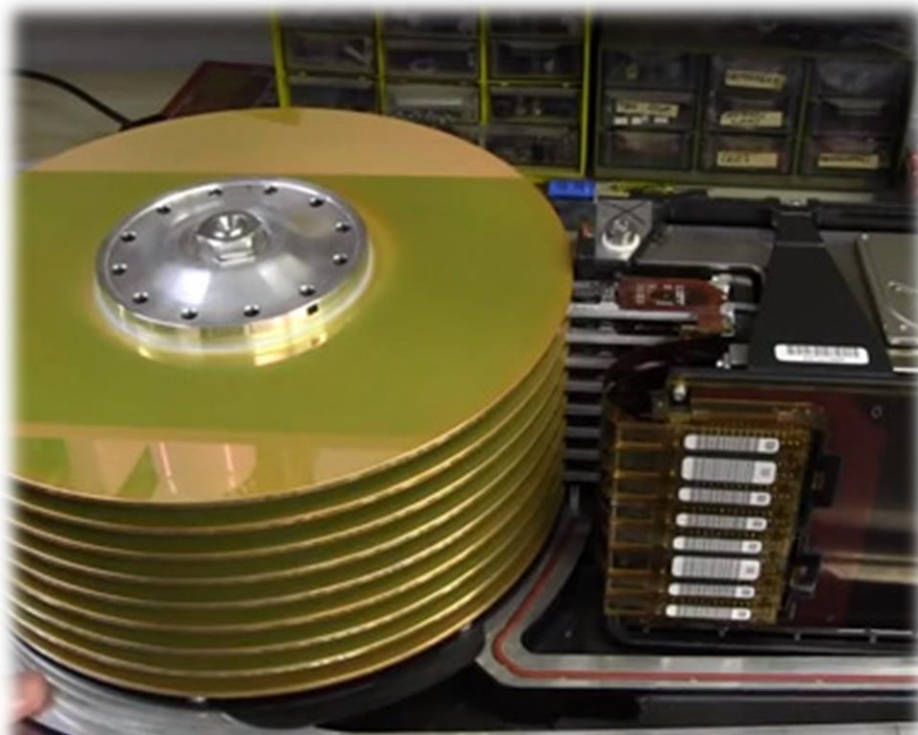
Anys 60: Processament Seqüencial

- Emmagatzematge en cintes magnètiques.
- Lectura seqüencial.
- Mida dades
 - Lectura dades (cinta/es)
 - Processat de dades



Evolució Històrica de les base de dades

Anys 80: SGBD relacionals



IBM HDD (1989) \$250K ~ 4GB ~ 38.5kg → Sistemes Bancaris

- La innovació en el hw, va comportar un cert abaratiment de la informàtica i la seva extensió a moltes més organitzacions, propiciant la necessitat de la innovació en el sw, que en els SGBDs va implicar la independència de les aplicacions dels aspectes físics de les dades, a fi de simplificar la feina dels programadors.
- A més, l'aparició del llenguatge de consulta estructurat **SQL** i la seva estandardització a partir de l'any **1986** van facilitar enormement l'ús dels **sistemes relacionals** i la seva implantació massiva.
- L'èxit de SQL i dels sistemes relacionals seria tan gran, que inclús avui dia la majoria dels SGBD es continuen basant en els mateixos principis que es van establir a finals dels 80.

Base de dades NOSQL

Not Only SQL (Structured Query Language)

- Apareixen degut a la WEB 2.0 ja que qualsevol usuari podia pujar contingut, provocant així un creixement exponencial de les dades. Aplicacions tals com Facebook, Twitter o Youtube.
- Bases dades relacionals tenen problemes per gestionar tota la informació. Molt poc rendiment en l'accessibilitat de les dades. Cal escalabilitat vertical. Això vol dir incrementar molt els costos.
- NOSQL no compleix res del que garantitza les BBDDRR.
- No segueixen un esquema, ni taules on emmagatzemar les dades i molt menys relacions.

Panorama SQL vs NoSQL



SQL: Structured Query Language

SQL

DDL:
Definició

DML:
Manipulació

DCL:
Control

TCL:
Transacció

CREATE

ALTER

DROP

SELECT

INSERT

UPDATE

DELETE

GRANT

REVOKE

COMMIT

ROLLBACK

SAVEPOINT

SQL: Llenguatges Suportats

Data Definition Language (DDL)

- Permet la definició, alteració i eliminació dels elements que conformen l'estructura en sí de la base de dades → Metadades o esquema.

Data Manipulation Language (DML)

- Permet la inserció, actualització, esborrat i consulta de les dades emmagatzemades a la BD. → Dades

Data Control Language (DCL)

- Permet el control d'accés a la base de dades i la gestió de permisos en base a rols i usuaris.

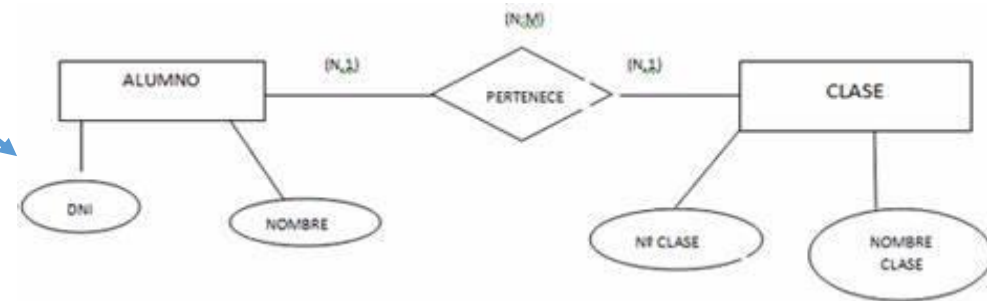
Transaction Control Language (TCL)

- Permet el control transaccional de les operacions realitzades.

Disseny de BDs

1. Presa de requeriments
2. Requeriments
3. Model Entitat-Relació
4. Model Relacional (taules)
5. Model físic

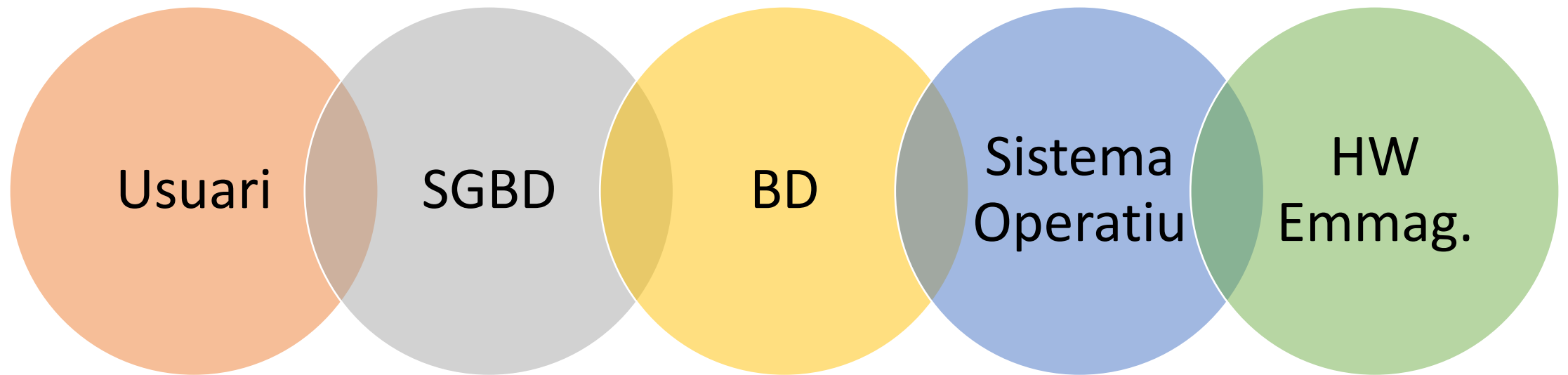
Es vol tenir el control de la ubicació assignada a cada alumne per classe.
.....



Create table alumno (
 DNI char(9) primary key,
 nombre varchar(20)
) engine = innnoDB;
....

Alumno: DNI(pk), nombre
Clase: NumClase(pk), nombre
Pertenece: alumnoDNI(pk,fk), clase(pk,fk)

Actors que intervenen en els SGBD



Introducció a les BDs: Principis

- **Independència de Dades**

- Esquema Físic: Fitxers a disc
- Esquema Lògic: Metadades
- Esquema Extern: Vistes



- **Accés a Dades**

- Depenent del tipus de model: Relacional, Multidimensional, Documental, ...
- Llenguatges declaratius: SQL99, PL/SQL, TransactSQL, pgSQL, MySQL, ...
- Sense Llenguatges declaratius (NoSQL): MongoDB, ArangoDB, ...

Introducció a les BDs: Principis

- **Consistència de Dades**

- Restriccions d'Integritat: restringeixen les dades a emmagatzemar
 - Referencial: Els valors de les FKs existeixen a la relació referenciada com PKs o són NULL.
 - Domini: Els valors de les dades s'ajusten als tipus definits.
 - Entitat: Cap atribut d'una PK pot ser NULL.
 - El DBMS les garanteix refusant l'operació o aplicant propagació, restricció o nullificació.
- Control Transaccional : paradigma **ACID**
 - Atomicitat: una Tx es fa sencera o no es fa. → Gestor de Recuperacions
 - Consistència: si es pren una BDs consistent, quan s'acaba també.
 - Isolament: Txs concurrents = Txs en sèrie. → Gestor de Concurrència
 - Durabilitat: Un cop acabada la Tx els canvis persisteixen. → Gestor de Recuperacions

Introducció a les BDs: Principis

• Seguretat i Administració Centralitzada

- Seguretat en les dades: Paradigma CIA
 - Confidentiality: No permetre la consulta de les dades a usuaris no autoritzats.
 - Integrity: Només els usuaris autoritzats poden modificar les dades.
 - Availability: Als usuaris autoritzats no se'ls ha de denegar l'accés.
- Control d'usuaris: Identificació, autenticació i autorització.
- En SQL el control d'accés és discrecional a nivell d'objectes
 - Els usuaris/rols tenen assignats privilegis sobre taules, vistes, ...
 - Existeixen mecanismes per assignar i revocar aquests privilegis.
- El DBA és el responsable de la seguretat, de la recuperació davant de fallides, ajustos de rendiment, ...



Sistemes Gestors BDs vs Sistemes de Fitxers

Els SGBDs incorporen una sèrie de controls i verificacions de sèrie que els sistemes de fitxers no incorporen ni requereixen:

- Atomicitat
 - En una SGBD transaccional o es completa la transacció o es reverteix completament.
 - En un sistema de fitxers les operacions no són atòmiques i es poden tallar a mig fer.
- Consistència
 - En un SGBD amb control d'integritat referencial, no es poden associar dades no vinculades per les claus definides.
 - En un sistema de fitxers les dades es poden associar de qualsevol manera sense restriccions.
- Integritat
 - En un SGBD es garanteix la integritat de dades amb controls basats en signatures de valors de hash.
 - En un sistema de fitxers, un fitxer es pot corrompre a nivell de dades i no adonar-nos 'en si no afecta la firma.
- Redundància
 - En un SGBD que implementi claus primàries no es poden inserir dades duplicades.
 - En un sistema de fitxers, les dades es poden duplicar tants cops com sigui necessari.

Sistemes Gestors BDs vs Sistemes de Fitxers

- Aïllament
 - En un SGBDs les dades són independents del suport físic i lògic, podent-se treballar amb elles des de sistemes heterogenis: MacOS, Windows, Linux, iOS, Android, ...
 - En un sistema de fitxers les dades estan vinculades al sistema lògic/físic, i poden implicar restriccions d'accés. Per exemple, Windows és incapaç de llegir per si sol EXT4 de Linux.
- Persistència
 - En un SGBDs les dades una vegada es confirmen, persisteixen i estan volcades a disc i assegurades per el dietari del motor.
 - En un sistema de fitxers pot donar-se la situació que les dades aparentment s'hagin copiat/modificat mentre que encara es troben en caché/búffer → Ex: Extracció USBs.
- Concurrencia
 - En un SGBDs l'accés a dades està dissenyat per permetre i controlar la concurrència, tant en lectura com en escriptura.
 - En un sistema de fitxers, l'accés concurrent en escriptura no està suportat.
- Accés Segur
 - El control d'accessos en un SGBDs és granular i parametritzable en base a rols
 - En un sistema de fitxers el control d'accessos sol ser força més bàsic i menys granular.

Sistemes Gestors BDs vs Sistemes de Fitxers

En tot cas, un SGBD sempre farà ús del sistema de fitxers del sistema i acabarà desant la informació en forma de fitxers, ja sigui traduint cada taula en un fitxer, en múltiples fitxers o amb un fitxer implementant diverses taules alhora.

Aquests fitxers poden organitzar-se de diferents formes, facilitant l'accés directe o el seqüencial.

Models de Dades

Els SGBDs implementen diferents models de dades per tal d'organitzar la informació. D'acord al model que implementin, en variarà el disseny de la BDs i la forma de comunicar-nos amb el sistema.

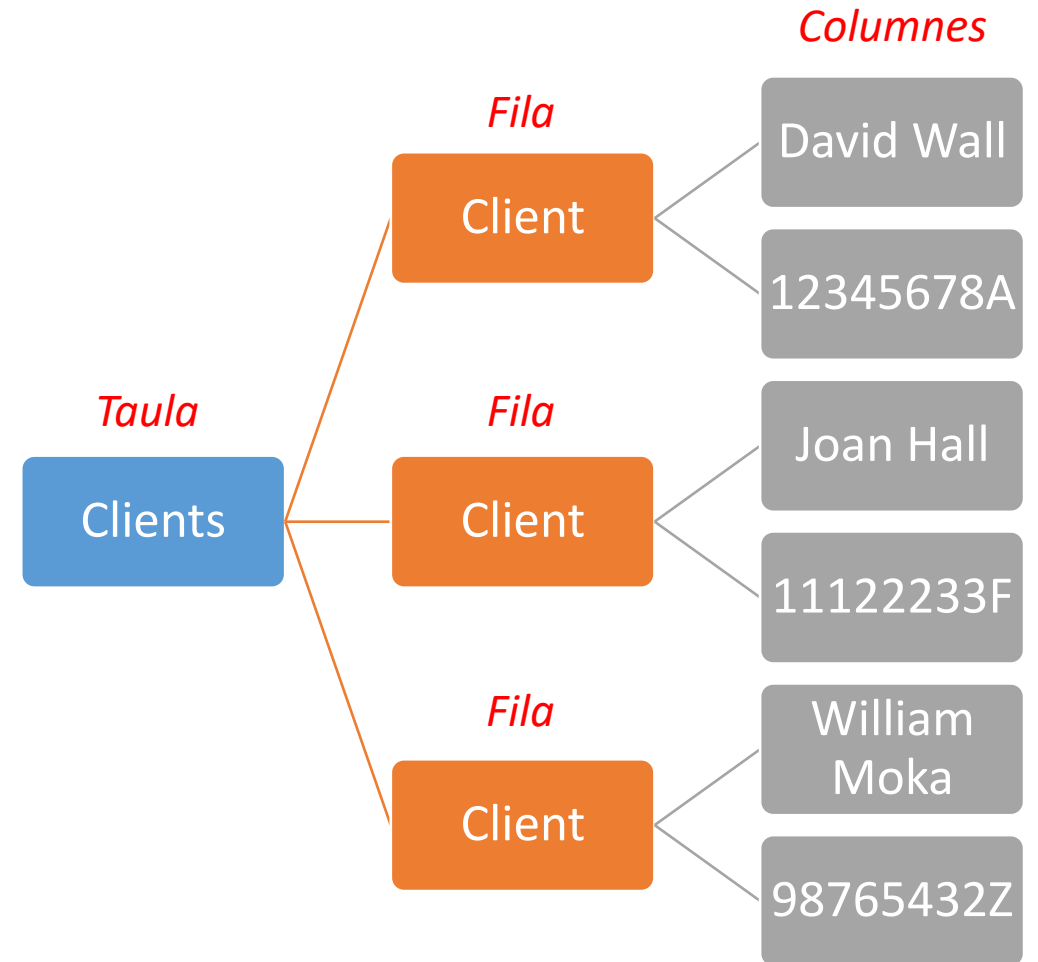
El model de dades més estès a l'actualitat és el model relacional ja sigui dins d'arquitectures centralitzades o distribuïdes.

Els models de dades que podem trobar són:

- Jeràrquic (Sistemes Legacy) → Principis de les base de dades.
- Relacional → El més habitual i propi dels dialectes SQL.
- En Xarxa → Complex i basat en l'ús de punters.
- Orientat a Objectes (NoSQL) → Força comú en APPs i utilitzat amb XML i JSON

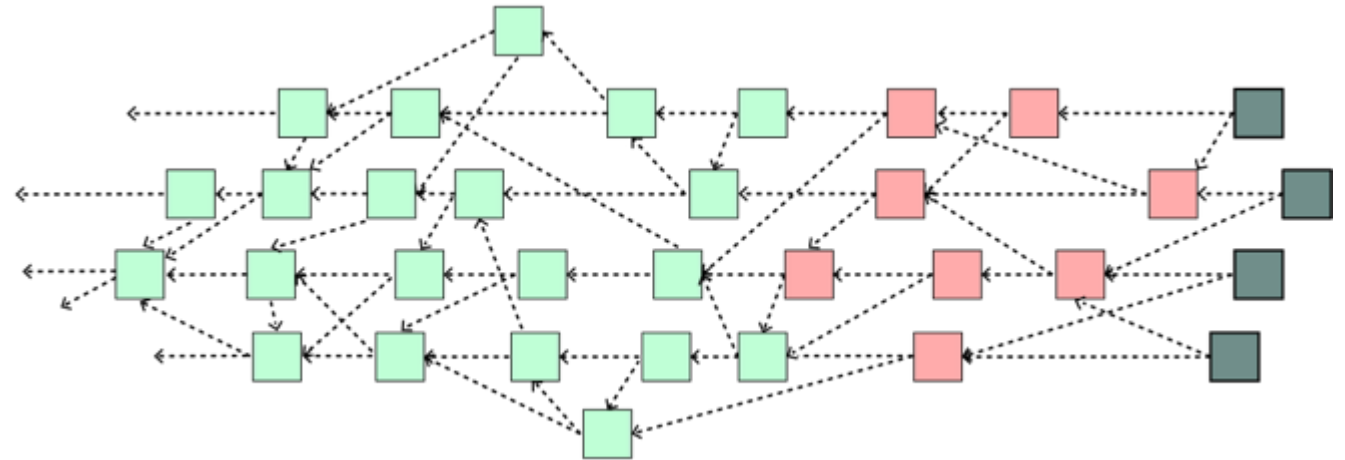
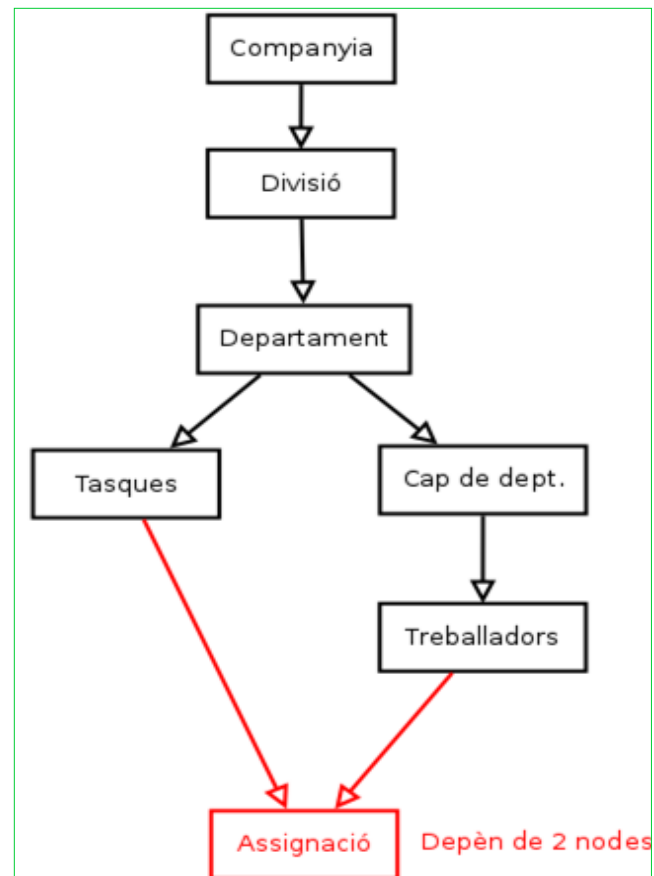
Models de Dades: Jeràrquic

```
<?xml version="1.0" encoding="UTF-8"?>
<clients>
  <client>
    <nom>David Wall</nom>
    <nif>12345678A</nif>
  </client>
  <client>
    <nom>Joan Hall</nom>
    <nif>11122233F</nif>
  </client>
  <client>
    <nom>William Moka</nom>
    <nif>98765432Z</nif>
  </client>
</clients>
```



El Model Relacional no permet més nivells de profunditat

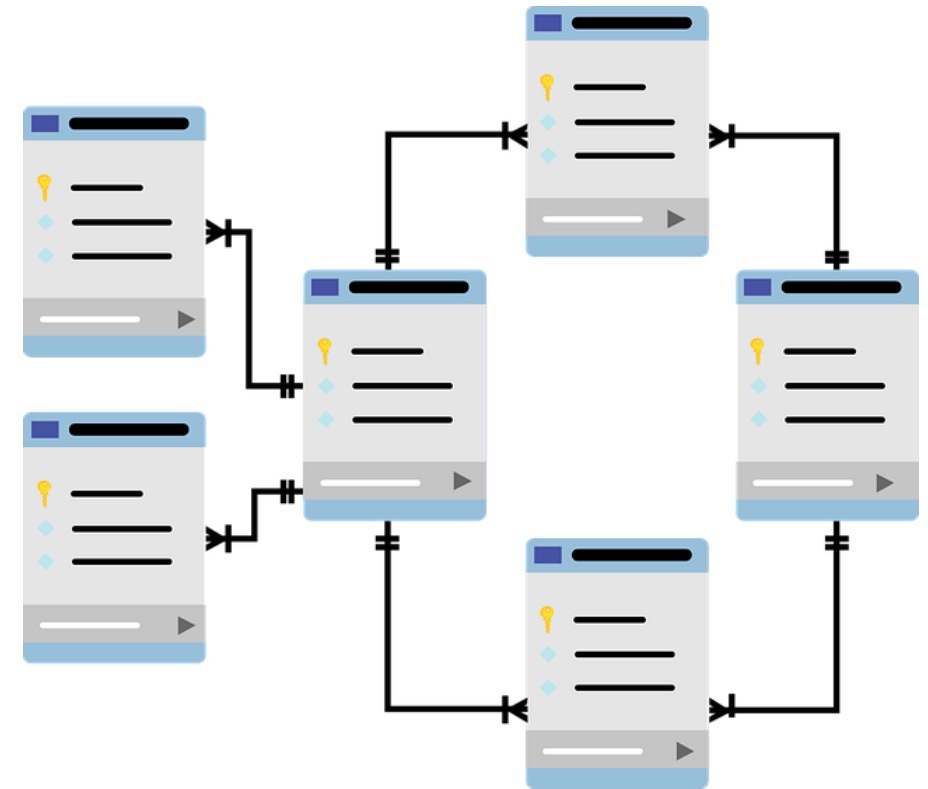
Models de Dades: En Xarxa



El Model Jeràrquic no permet dependències de més d'un node

Models de Dades: Relacional

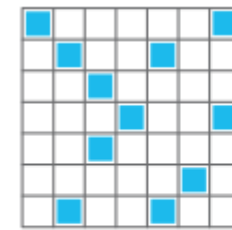
- Simplifiquen les relacions entre les entitats de la BDs
- Llenguatge estàndard SQL
- Disseny Simple
- Problemes per desar-hi certes dades
 - Imatges
 - Contingut Multimèdia
 - Binaris
 - Objectes



Models de Dades: Orientat a Objectes

- Prioritzen els objectes per sobre de les relacions.
- Integritat Referencial passa a segon terme.
- Molt estesos en l'àmbit IoT i de les APPs.
- Molt eficients per adquisició de dades.
- Adaptabilitat a diferents estructures de dades.
- Traducció directa a objectes.
- JSON és la codificació d'objectes més usada.

NoSQL
Database



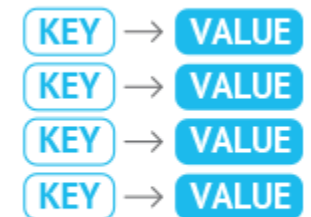
Column-Family



Graph



Document



Key-Value

Tipus d'usuaris del SGBDs

Podem diferenciar tres categories d'usuaris del SGBDs en funció de la manera en què interactuen amb el sistema:

- **Usuaris externs:** Són usuaris no sofisticats, que no interactuen directament amb el sistema, sinó mitjançant alguna aplicació informàtica desenvolupada prèviament per altres persones amb aquesta finalitat.
- **Administradors:** Interactuen directament amb el sistema, sense utilitzar les interfícies proporcionades per programes intermediaris. Formulen les consultes en un llenguatge de BD (normalment, SQL), des de dins de l'entorn que el SGBD posa a la seva disposició.

Tipus d'usuaris del SGBDs

- **Programadors d'aplicacions:** Són professionals informàtics que creen els programes que accedeixen als SGBD i que, posteriorment, són utilitzats pels usuaris que hem anomenat externs. Aquestes aplicacions es poden desenvolupar mitjançant diferents llenguatges de programació i eines externes al SGBD.

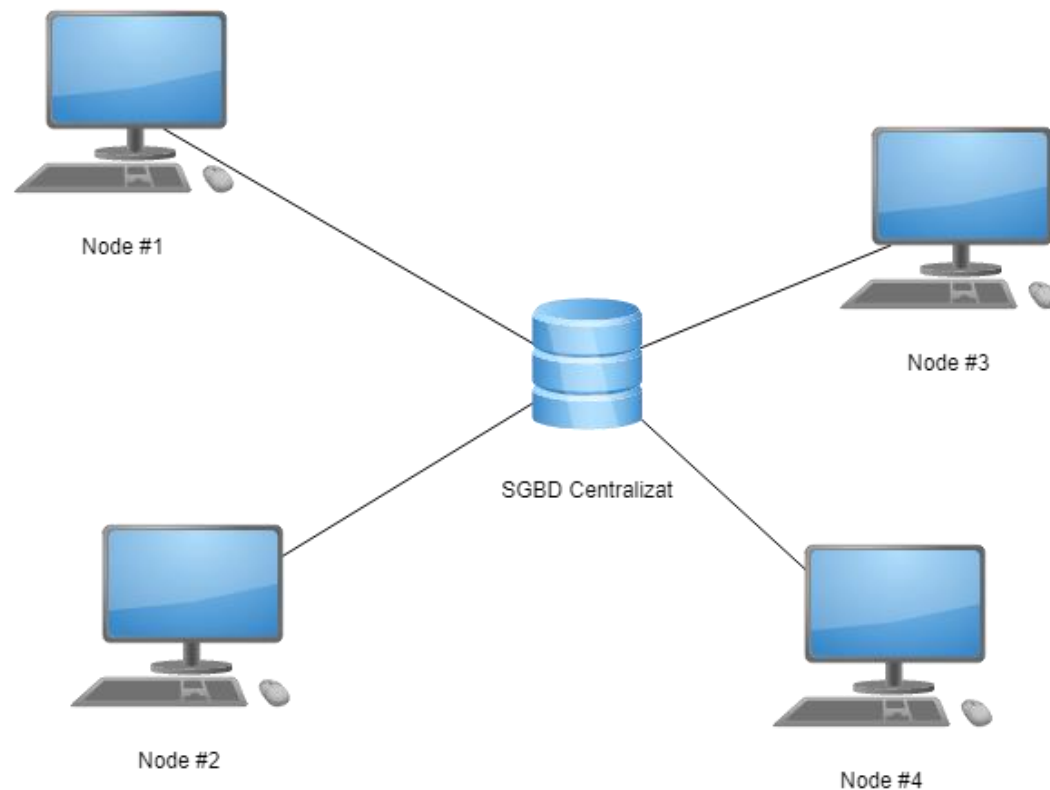
Creen funcions, procediments, triggers, vistes.

Administradors del SGBD

Una llista no exhaustiva de les tasques dels administradors podria ser la següent:

- Crear i administrar els esquemes de la BD.
- Administrar la seguretat: autoritzacions d'accés, restriccions, etc.
- Realitzar còpies de seguretat periòdiques.
- Controlar l'espai de disc disponible.
- Vigilar la integritat de les dades.
- Observar l'evolució del rendiment del sistema i determinar quins processos consumeixen més recursos.
- Assessorar els programadors i els usuaris sobre la utilització de la BD.
- Fer canvis en el disseny físic per millorar el rendiment.
- Resoldre emergències.

Arquitectura SGBD Centralitzat Client-Servidor



Avantatges

Gestió i Manteniment
Control d'integritat i consistència
Seguretat Simple
Disseny Simple
Backups

Requeriments Hw
Escalabilitat
Ample de banda
Disponibilitat: Únic punt de Fallida

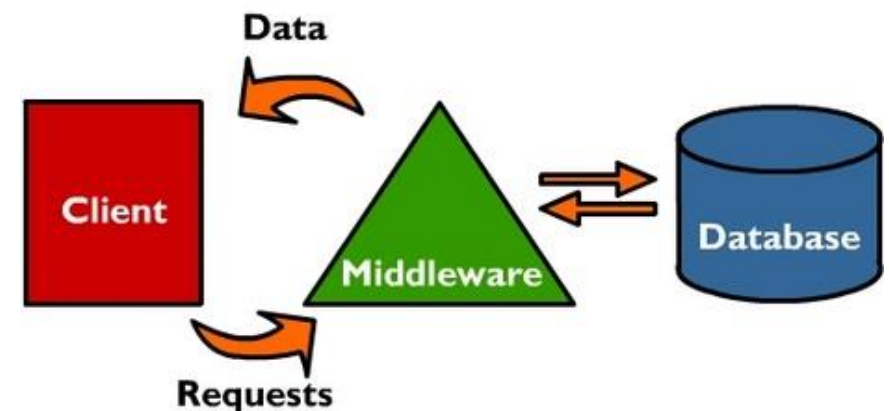
Inconvenients

Arquitectura SGBD Centralitzat Client-Servidor

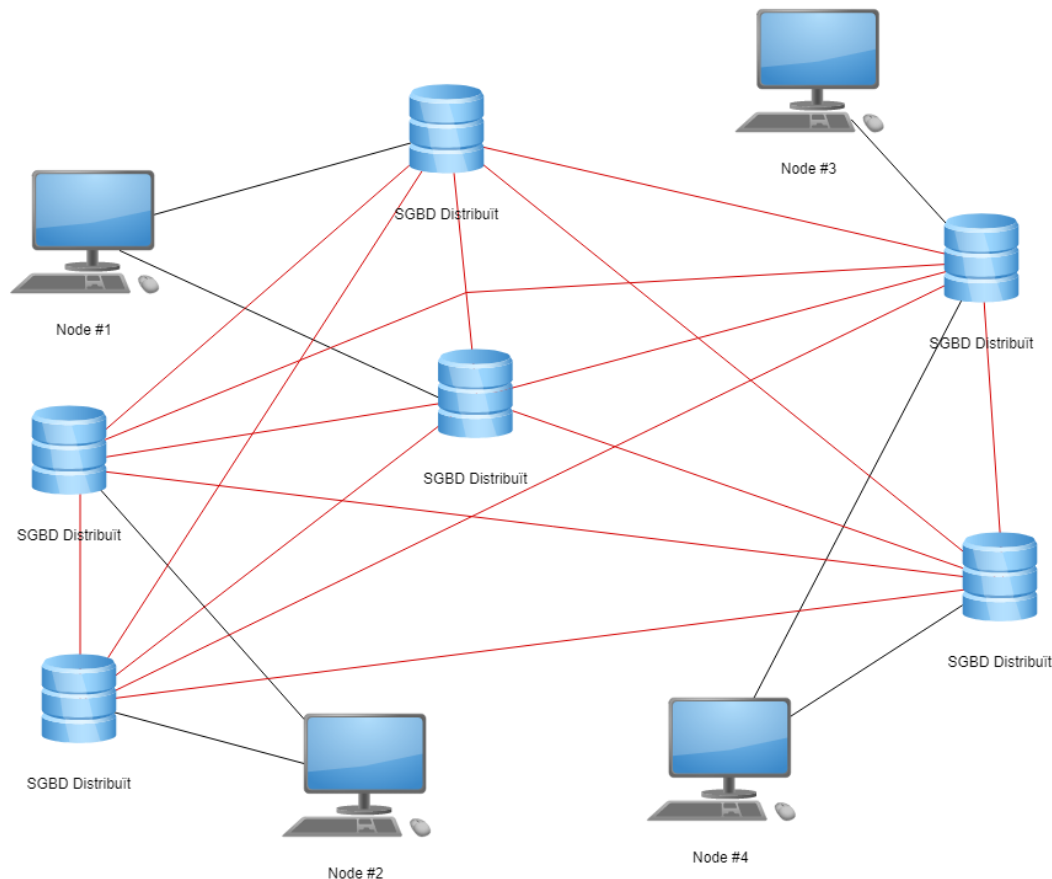
Aquest sistema inclou 3 components principals: clients, servidors i programari intermediari de comunicacions.

- El **client** és qualsevol ordinador que sol·licita serveis al servidor. → Frontend.
- El **servidor** és qualsevol procés de computadora que proporciona serveis als clients. → Backend.
- El **programari intermediari de comunicacions** és qualsevol procés o conjunt d'aquests mitjançant els quals el procés client i el procés servidor es comuniquen → Middleware.

Exemple: Connectors ODBC, JDBC, etc.



Arquitectura SGBD Distribuït



Avantatges

Hw Requerit
Escalabilitat
Ample de banda
Alta disponibilitat

Gestió i Manteniment
Control d'integritat i consistència
Seguretat Complexa
Disseny Complexe
Backups

Inconvenients

SGBD Distribuïts: Avantatges

- Les dades es localitzen prop del lloc on hi ha més demanda.
- Permeten un accés més ràpid a les dades.
- Permeten un processament més ràpid de les dades.
- Faciliten el creixement, ja que permeten afegir nous llocs a la xarxa.
- Hi ha una millora en les comunicacions, ja que els llocs són més petits i estan més a prop dels clients. → Latències mínimes
- Reducció de costos, ja que és més barat afegir una nova estació de treball que actualitzar un servidor central. → Flexibilitat

SGBD Distribuïts: Avantatges

- Menys perill de fallida en un sol punt, ja que en cas que una estació de treball es col·lapsi la seva càrrega de treball pot ser absorbida per altres. → Alta disponibilitat
- Independència del processador, ja que una sol·licitud d'un usuari pot ser processada per qualsevol processador disponible. → Cloud computing



SGBD Distribuïts: Desavantatges

- Increment en el cost.
- Control d'Integritat més difícil.
- Manca de normes estàndard.
- Disseny de base de dades més complex.
- Dificultat en la gestió i control. Les sol·licituds han de reconèixer les dades d'ubicació i han de ser capaces d'unir les dades de diversos llocs.
- Tecnològicament complex: integritat de dades, gestió de transaccions, control de concurrència, seguretat, còpia de seguretat, recuperació, optimització de consultes, selecció de ruta d'accés, ... són tan sols algunes de les qüestions que s'han d'abordar.

SGBD Distribuïts: Desavantatges

- Els errors de seguretat augmenten quan les dades es troben en diverses ubicacions.
- Augment d'emmagatzematge i requisits d'infraestructura, a causa de la redundància de dades i els requeriments de seguretat a nivell de backup.
- Dependència de proveïdors externs (cloud).
- Es requereix duplicar la infraestructura (personal, programari i llicències, la ubicació física o del medi ambient) i això a vegades no pot compensar els estalvis operatius.

SGBD Distribuïts: Fragmentació

*La **fragmentació** consisteix a dividir les relacions de la BD en diferents fragments. Aquests fragments han de contenir tota la informació necessària per tal de reconstruir les relacions originàries corresponents, en cas necessari.*



El problema fonamental de la fragmentació inherent a les BD distribuïdes consisteix a trobar la unitat ideal de distribució de les dades. Normalment les relacions no són la millor opció de distribució per moltes raons.

SGBD Distribuïts: Fragmentació Horitzontal

La **fragmentació horitzontal** consisteix a dividir les tuples d'una relació (és a dir, les files) en dos o més subconjunts en funció dels valors que aquelles tinguin en un o més atributs.

Aquests valors han de ser indicatius dels nodes que més consultes realitzaran sobre les respectives tuples, per tal d'acostar aquestes als seus usuaris més habituals. Les tuples poden estar presents en més d'un fragment, però han d'estar com a mínim en un d'ells per tal que la fragmentació sigui correcta.

SGBD Distribuïts: Fragmentació Horitzontal

Relació original PROVEÏDOR:

NIF*	Nom	Telefon	Adreça	Localitat
33333333K	L'abastadora, SL	902456456	Pol. Ind. Polièdric, s/n	Lleida
44444444L	Proveïdora Ibèrica, SA	906789789	C/ del pi, 3	Lleida
55555555M	Supplies & Co. Ltd.	900123123	C/ del call, 4	Girona
66666666N	Assortiments de l'Onyar, SCP	908852852	Pg. De la ribera, s/n	Girona

SGBD Distribuïts: Fragmentació Horitzontal

Primer fragment horitzontal de la relació PROVEIDOR

NIF*	Nom	Telefon	Adreça	Localitat
33333333K	L'abastadora, SL	902456456	Pol. Ind. Polièdric, s/n	Lleida
44444444L	Proveïdora Ibèrica, SA	906789789	C/ del pi, 3	Lleida

Segon fragment horitzontal de la relació PROVEIDOR

NIF*	Nom	Telefon	Adreça	Localitat
55555555M	Supplies & Co. Ltd.	900123123	C/ del call, 4	Girona
66666666N	Assortiments de l'Onyar, SCP	908852852	Pg. De la ribera, s/n	Girona

SGBD Distribuïts: Fragmentació Vertical

La **fragmentació vertical** consisteix a dividir els atributs de la relació (és a dir, les columnes) en diferents fragments. Els fragments resultants han de contenir els atributs que utilitzaran més freqüentment els usuaris del node on seran respectivament emmagatzemats.

Els atributs poden estar presents en més d'un fragment, però han d'estar com a mínim en un d'ells per tal que la fragmentació sigui correcta.

SGBD Distribuïts: Fragmentació Vertical

Els fragments resulten de seleccionar només els atributs de cada proveïdor que utilitzaran amb més freqüència els nodes que respectivament els emmagatzemin.

Primer fragment vertical de la relació PROVEIDOR

NIF*	Nom	Telefon
33333333K	L'abastadora, SL	902456456
44444444L	Proveïdora Ibèrica, SA	906789789
55555555M	Supplies & Co. Ltd.	900123123
66666666N	Assortiments de l'Onyar, SCP	908852852

Segon fragment vertical de la relació PROVEIDOR

NIF*	Nom	Adreça	Localitat
33333333K	L'abastadora, SL	Pol. Ind. Polièdric, s/n	Lleida
44444444L	Proveïdora Ibèrica, SA	C/ del pi, 3	Lleida
55555555M	Supplies & Co. Ltd.	C/ del call, 4	Girona
66666666N	Assortiments de l'Onyar, SCP	Pg. De la ribera, s/n	Girona

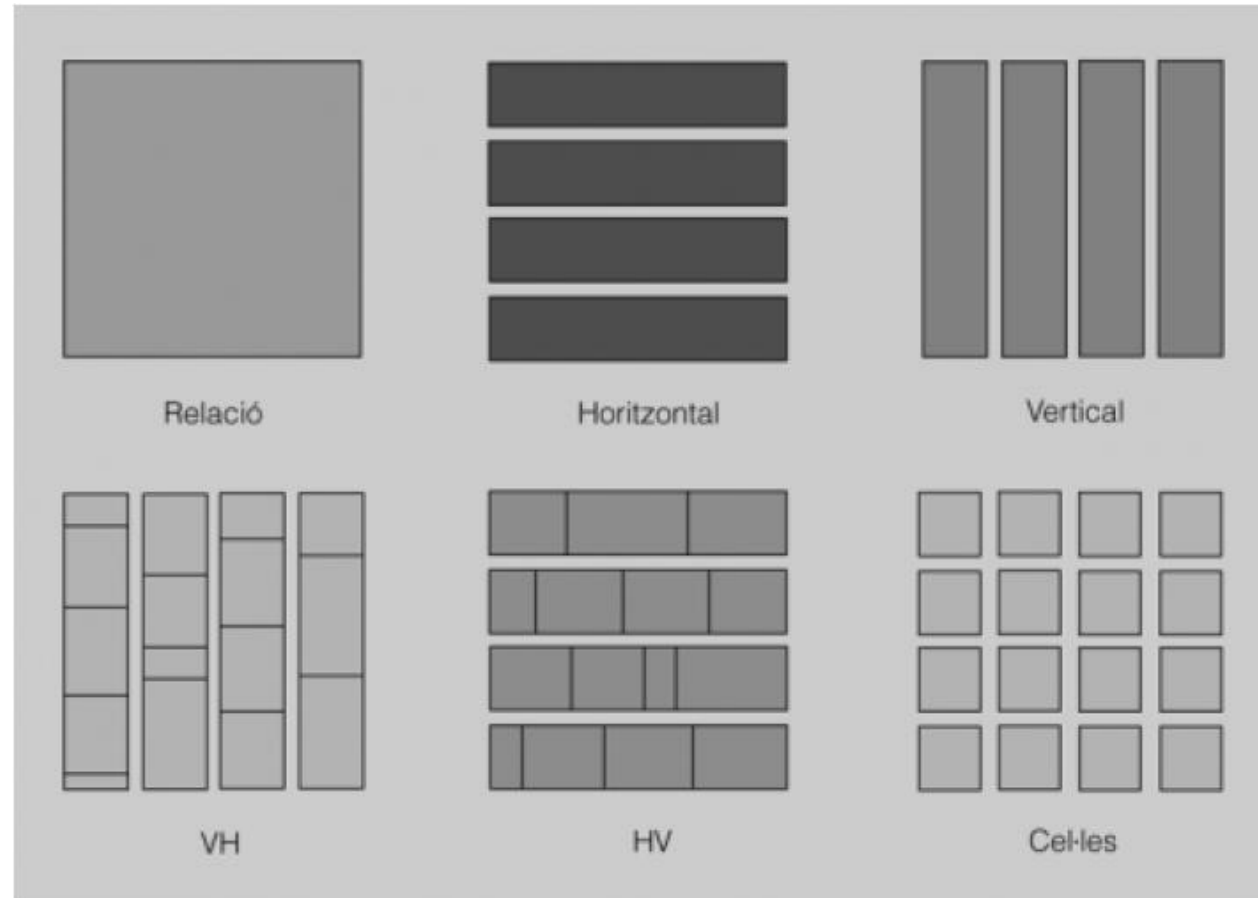
SGBD Distribuïts: Fragmentació Mixta

Les **fragmentacions mixtes** consisteixen a aplicar tant la fragmentació horitzontal com la vertical.

En funció de com es combinen les fragmentacions horitzontal i vertical:

- 1. Fragmentació VH.** Es desenvolupa en primer lloc la fragmentació vertical, i a continuació l'horitzontal.
- 2. Fragmentació HV.** S'aplica primer una divisió horitzontal i tot seguit es desenvolupa una altra de vertical sobre els fragments prèviament generats.

SGBD Distribuïts: Fragmentació Mixta



SGBD Distribuïts: Disseny de BDs distribuïdes

En el cas d'optar per una BD distribuïda, cal prendre altres decisions no menys importats sobre quina és la millor manera d'ubicar en els diferents nodes del sistema tant les dades com les aplicacions que hagin d'accedir a aquestes dades.