

M02. BASES DE DADES

UF4. Bases de dades NoSQL

BASES DE DADES

NOSQL

Com treballam amb dades no estructurades?

BBDD RELACIONALS



Escalament vertical:

- Més emmagatzematge
- Processament més ràpid



BBDD NoSQL



BBDD NoSQL

Models de dades: Una BBDD NoSQL permet construir una aplicació sense haver de definir primer el schema, al contrari que en BBDD relacionals on cal definir primer el teu schema abans de poder afegir-hi dades.

Estructura de les dades: Les BBDD NoSQL estan dissenyades per gestionar dades desestructurades (texts, dades de xarxes socials, vídeos, emails, ...).

Model desenvolupament: Les BBDD NoSQL són open source, mentre que les BBDD relacionals són típicament de software propietari.

BBDD NoSQL

- Als anys 90 es parlava de la idea de bases de dades d'objectes.
 - Desar objectes directament a memòria.
- No va tenir èxit
 - Les empreses havien invertit molt en integracions amb BBDD SQL.
 - No eren únicament per desar dades, sinó per integrar aplicacions.
- Empreses amb necessitats específiques van començar a treballar-hi:
 - Google → Bigtable (2005)
 - Amazon → DynamoDB (2012)

BBDD NoSQL

El terme **NoSQL** es va popularitzar per fer referència a les bases de dades que no suporten el model relacional i que no utilitzen SQL.

Aquests sistemes de bases de dades no relacionals estan dissenyats per solventar problemes en què les BBDD relacionals no són adequades.

NoSQL fa referència a totes aquelles BBDD que són no-relacionals, distribuïdes, open source, sense schema i horitzontalment escalables.

BBDD NoSQL. AVANTATGES

Alt rendiment: La velocitat de lectura i escriptura en un sistema NoSQL és normalment molt més alta que en sistemes relacionals tradicionals. Per exemple, Hypertable pot emmagatzemar un trilió de dades per dia. Google BigTable pot processar 20 petabytes de dades al dia.

Escala horitzontalment: Si la base de dades creix, és possible afegir nodes en un sistema distribuït de manera que es pot ampliar la capacitat d'emmagatzematge i processament amb facilitat.

BBDD NoSQL. AVANTATGES

Mecanisme d'emmagatzematge: La forma d'emmagatzematge és més senzilla que el esquema de taules d'un model relacional. Això afecta positivament el rendiment, ja que no hi ha un esquema rígid per desar dades.

Emmagatzematge flexible: Els esquemes de les bases de dades no relacionals són molt més flexibles que l'esquema rígid de les taules. De fet, no tenen cap esquema fixe. Quan les dades que cal desar no es poden traduir en taules d'un model relacional, és convenient considerar una solució NoSQL.

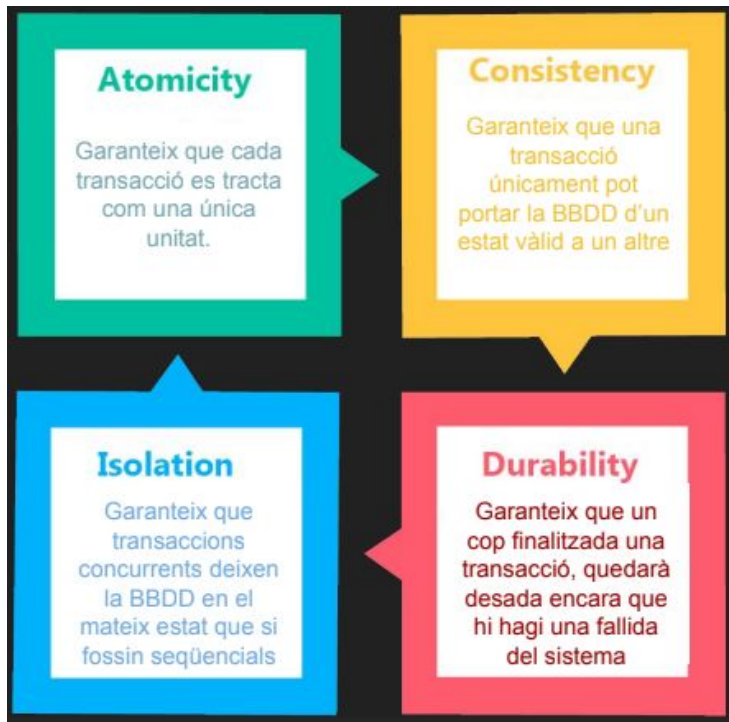
BBDD NoSQL. INCONVENIENTS

Llenguatge de consulta no estàndard: No hi ha un llenguatge de consultes (query language) estandarditzat per les bases de dades NoSQL. Cal aprendre el llenguatge propi de cadascun dels sistemes NoSQL.

Transaccions: Com a conseqüència de no complir les propietats ACID, hi ha un control feble de la consistència, durabilitat i aïllament de les transaccions.

Integritat: Assegurar la integritat de les dades requereix programació manual addicional i no és una funció del sistema NoSQL.

BBDD NOSQL. INCONVENIENTS



- **Atomicitat:** una transacció és indivisible, o s'executen totes la sentències o no s'executa cap.
- **Consistència:** assegura que després d'una transacció la base de dades estarà en un estat vàlid i consistent.
- **Aïllament:** garanteix que cada transacció està aïllada de la resta de transaccions i que l'accés a les dades es farà de manera exclusiva.
- **Durabilitat:** els canvis que realitza una transacció sobre la base de dades són permanents.

BBDD NOSQL. RELACIONAL O NOSQL?

El millor sistema de bases de dades a utilitzar dependrà d'un conjunt de factors, entre ells:

- El volum de dades a emmagatzemar.
- El nivell de concurrència esperat.
- Nombre d'operacions a realitzar en la BBDD per unitat de temps.
- Tipus d'escalabilitat desitjada en la BBDD.
- El nivell d'integritat i consistència desitjat.
- La naturalesa de les dades a emmagatzemar.
- El tipus més freqüent d'operacions a realitzar amb les dades.

BBDD NOSQL. TIPUS

Hi ha 4 categories principals de sistemes de bases de dades no-relacionals:

- Bases de dades clau - valor
- Bases de dades orientades a columna
- Bases de dades orientades a documents
- Bases de dades orientades a grafs

BBDD NOSQL. BASES DE DADES CLAU-VALOR

- És un sistema que emmagatzema valors indexats per claus, i pot emmagatzemar dades estructurades i desestructurades.
 - Els valors s'emmagatzemen com arrays de bytes.
 - El contingut no és important per a la base de dades.
 - Alt rendiment, molt escalable, molt flexible, i poca complexitat.
- El principal inconvenient d'aquest tipus de bases de dades és que no permeten consultes complexes, ja que únicament cerquen per una clau.
- Exemples: Amazon DynamoDB, Redis, Cassandra, Voldemort, RAMCloud, Flare, ...

BBDD NOSQL. BASES DE DADES ORIENTADES A DOCUMENT

- Les dades s'emmagatzemen típicament en format de documents JSON (JavaScript Object Notation), però també es pot utilitzar altres formats com XML.
 - Permeten consultar les dades amb queries molt avançades.
 - Permeten relacions entre les dades.
 - No permeten operacions de join (per evitar problemes de rendiment).
- Exemples: [MongoDB](#), CouchDB, Cloudkit, DB2, pureXML, ...

BBDD NoSQL. MONGODB

MongoDB és una de les bases de dades NoSQL més populars. És una base de dades multiplataforma, orientada a objectes. Bàsicament, NoSQL significa que MongoDB no emmagatzema dades en la taula ni en el format relacional, sinó que proporciona un mecanisme diferent per a l'emmagatzematge i la recuperació de dades. Aquest mecanisme s'anomena BSON (Binary JSON), similar al JSON. Ofereix alta velocitat, alta disponibilitat i alta escalabilitat.

MongoDB emmagatzema registres de dades com a documents (específicament documents BSON) que es reuneixen en col·leccions. Una base de dades emmagatzema una o més col·leccions de documents.

BBDD NOSQL. MONGODB

Sentències bàsiques

Sentència	Descripció	Exemple
<code>MongoClient()</code>	Connexió al cluster de MongoDB	<code>client = MongoClient(MONGODB_URI)</code>
<code>list_database_names()</code>	Llista totes les DB al cluster	<code>client.list_database_names()</code>
<code>list_collection_names()</code>	Llista totes les col·leccions de la DB	<code>db.list_collection_names()</code>
<code>insert_one()</code>	Insereix un document a la col·lecció	<pre>col.insert_one({ 'nom': 'Miquel', 'cognoms': 'Garcia', 'especialitats': ['ASIX', 'CIBER'] })</pre>

BBDD NOSQL. MONGODB

Sentències bàsiques

Sentència	Descripció	Exemple
<code>insert_many()</code>	Insereix diversos documents a la col·lecció	<pre>col.insert_many([{ 'nom': 'Laura', 'cognoms': 'Fernández', 'especialitats': ['DAM', 'Android'] }, { 'nom': 'Joana', 'cognoms': 'Ivars', 'especialitats': ['DAW', 'Python'] }])</pre>

BBDD NOSQL. MONGODB

Sentències bàsiques

Sentència	Descripció	Exemple
<code>count_documents()</code>	Retorna el total de documents dins una col·lecció	<code>col.count_documents({})</code>
<code>find_one()</code>	Cerca un document de la col·lecció. Es poden aplicar filtres	<code>col.find_one({"age":{"\$lt":33}})</code>
<code>find()</code>	Cerca tots els documents de la col·lecció. Es poden aplicar filtres	<pre>for x in col.find({}): print(x)</pre>

BBDD NOSQL. MONGODB

Sentències bàsiques

Sentència	Descripció	Exemple
<code>delete_one()</code>	Elimina el primer document d'una col·lecció que compleix la condició	<pre>col.delete_one({ 'nom': 'Miquel' })</pre>
<code>delete_many()</code>	Elimina tots els documents d'una col·lecció que compleixen la condició	<pre>col.delete_many({"age":{"\$lt":33}})</pre>