

SEMINARI TECNOLÒGIC: MongoDb

Xavier Sala Pujolar

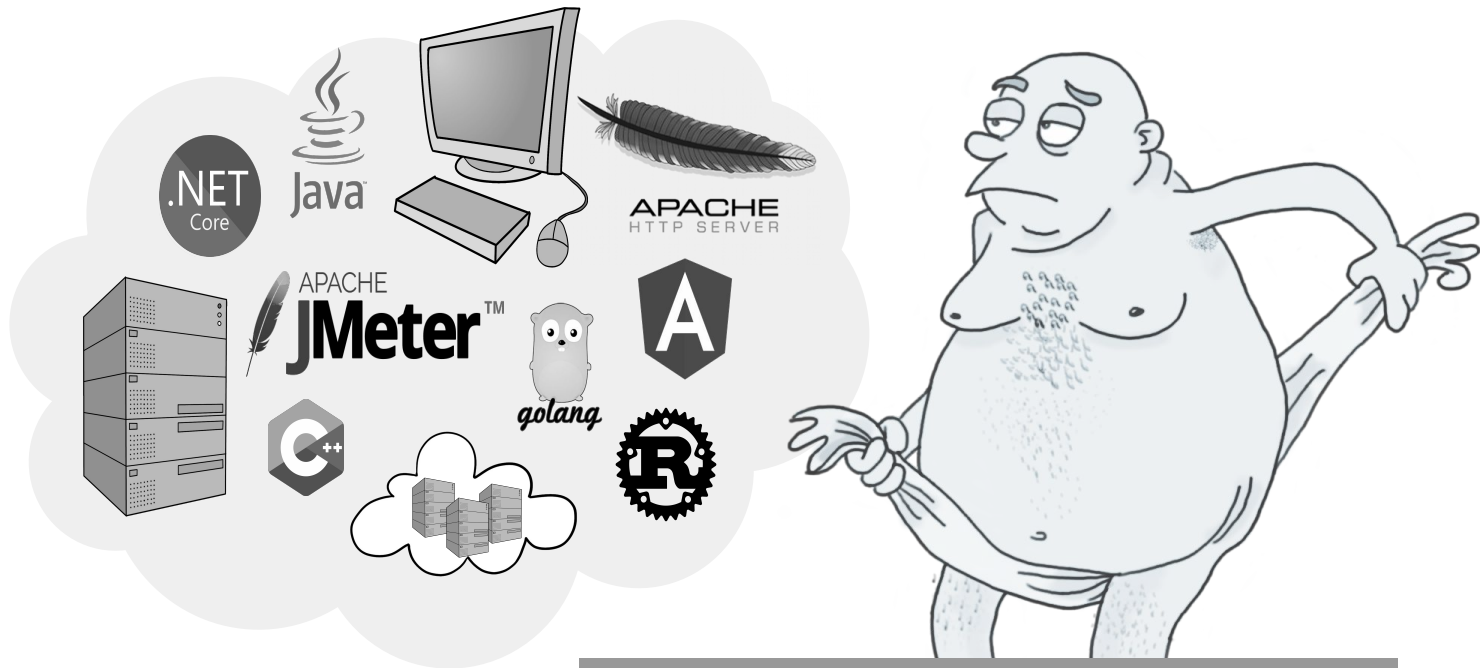


MongoDB

{
Universitat
de Girona
}

Abril 2021

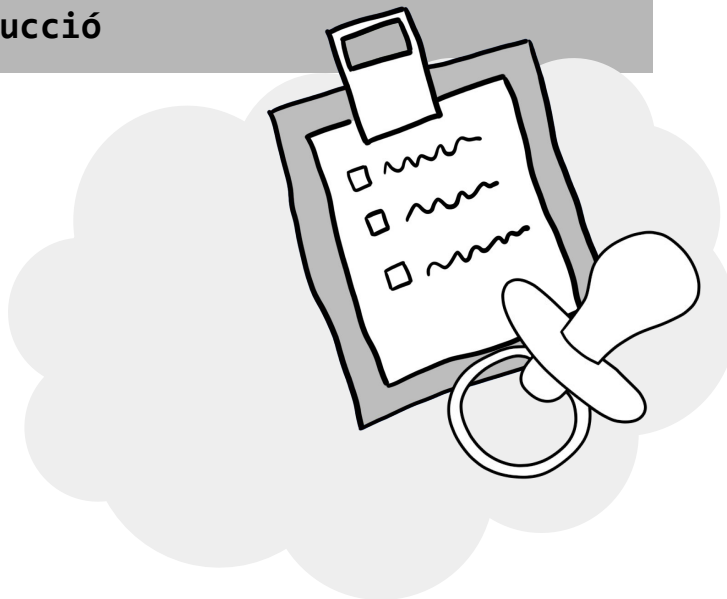
Qui sóc?



Xavier Sala

Objectiu 1 : Introducció a MongoDB

Introducció



Programa



Dia 1:

MongoDb i NoSQL

Dia 2:

Treball amb MongoDB

Dia 3:

Programació amb MongoDB (C#)

Dia 1: MongoDB i NoSQL

Xavier Sala Pujolar



MongoDB

{
Universitat
de Girona
}

Abril 2021

NoSQL



NoSQL
Not Only SQL

Les bases de dades NoSQL parteixen de la idea que les bases de dades relacionals no són la solució ideal per tots els casos

Emmagatzemen les dades d'una forma diferent a les taules relacionals

NOSql neix per solucionar problemes concrets



A vegades fan falta
esquemes que no siguin
rígids

Incrementar la
velocitat de
lectura

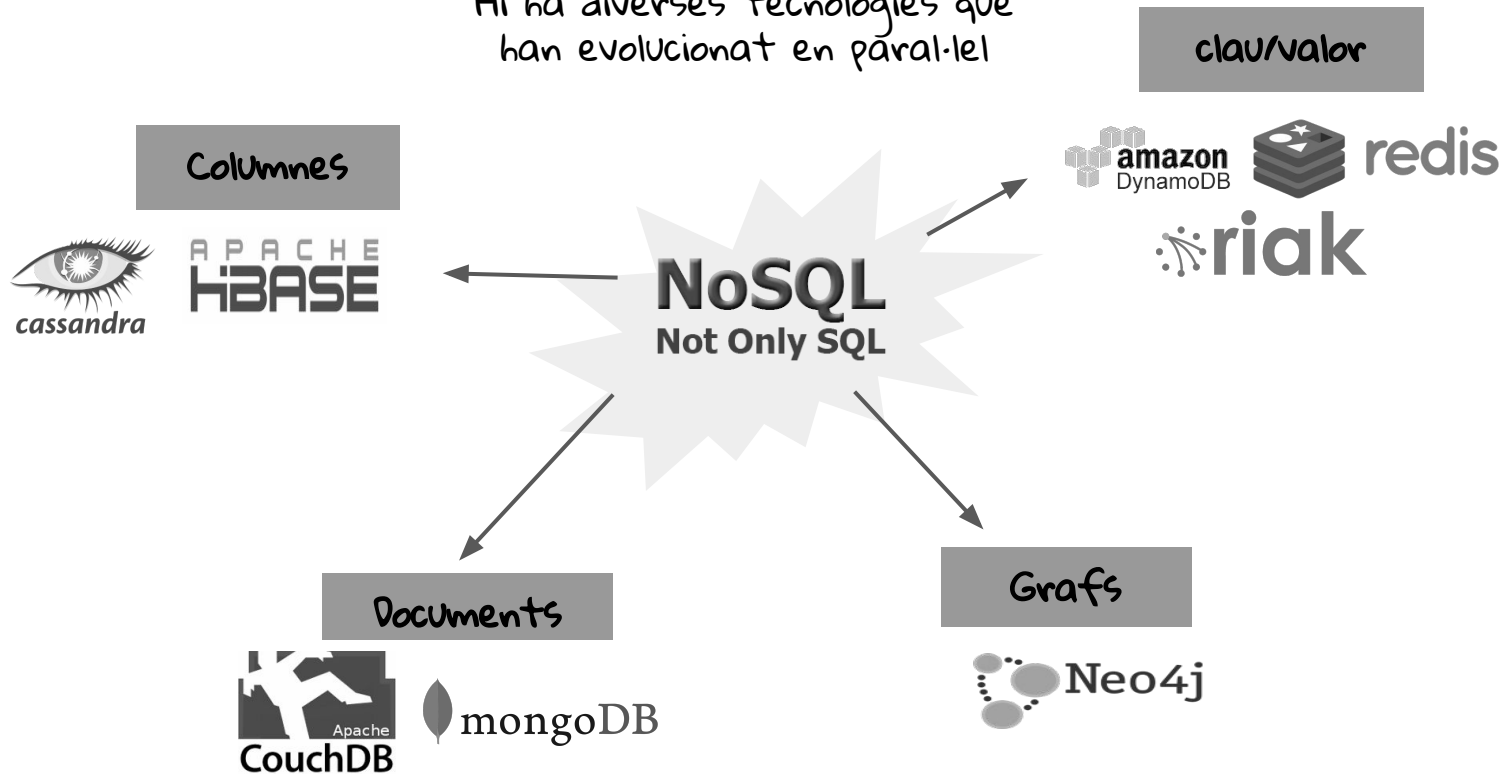
Facilitar la
tolerància a
fallades

Tractament de
grans volums de
dades

L'escalabilitat en bases
de dades relacionals no
sempre és senzilla

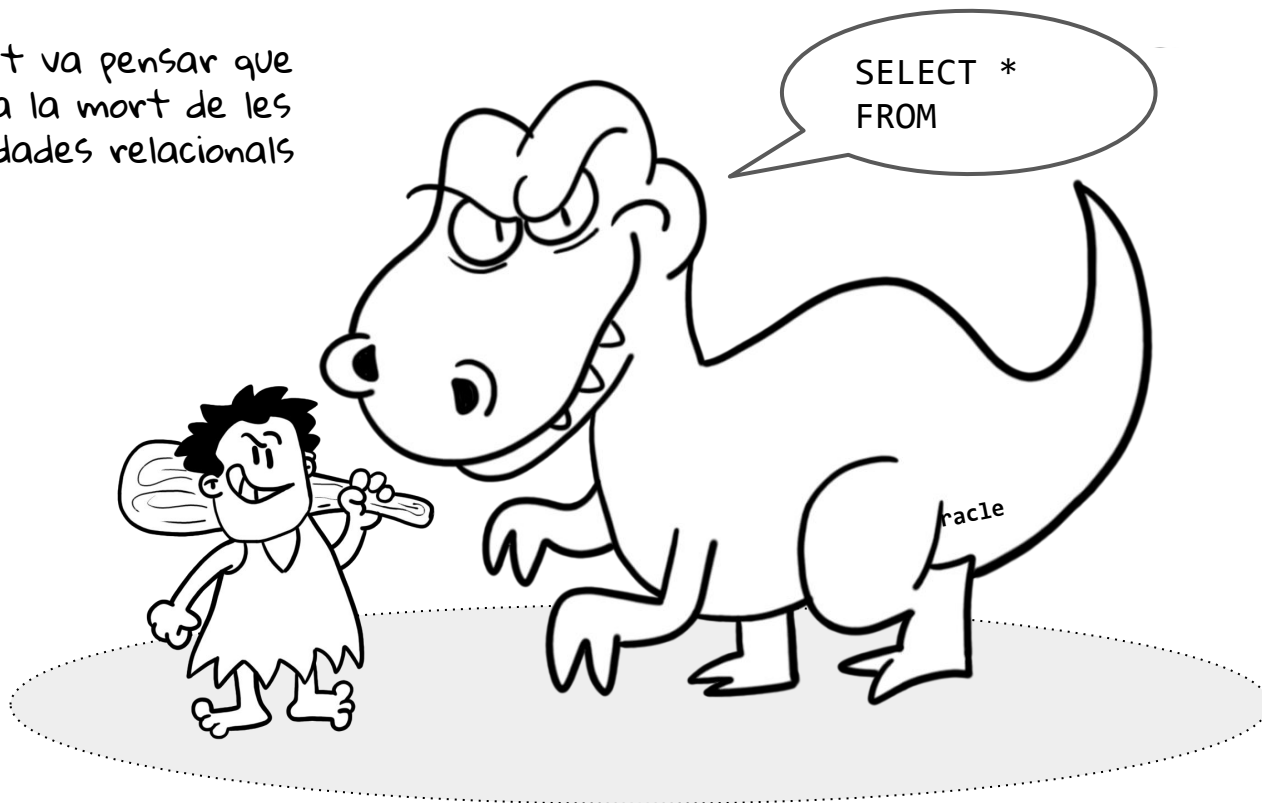
Tipus de NoSQL

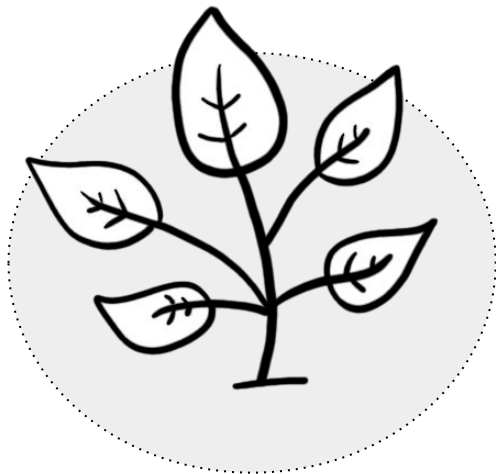
Hi ha diverses tecnologies que han evolucionat en paral·lel



Bases de dades

Molta gent va pensar que
NoSQL era la mort de les
bases de dades relacionals





MongoDB és una base de dades
NOSQL de documents

Pensada per oferir alt
rendiment, escalabilitat,
flexibilitat i alta
disponibilitat

Els documents són en format BSON

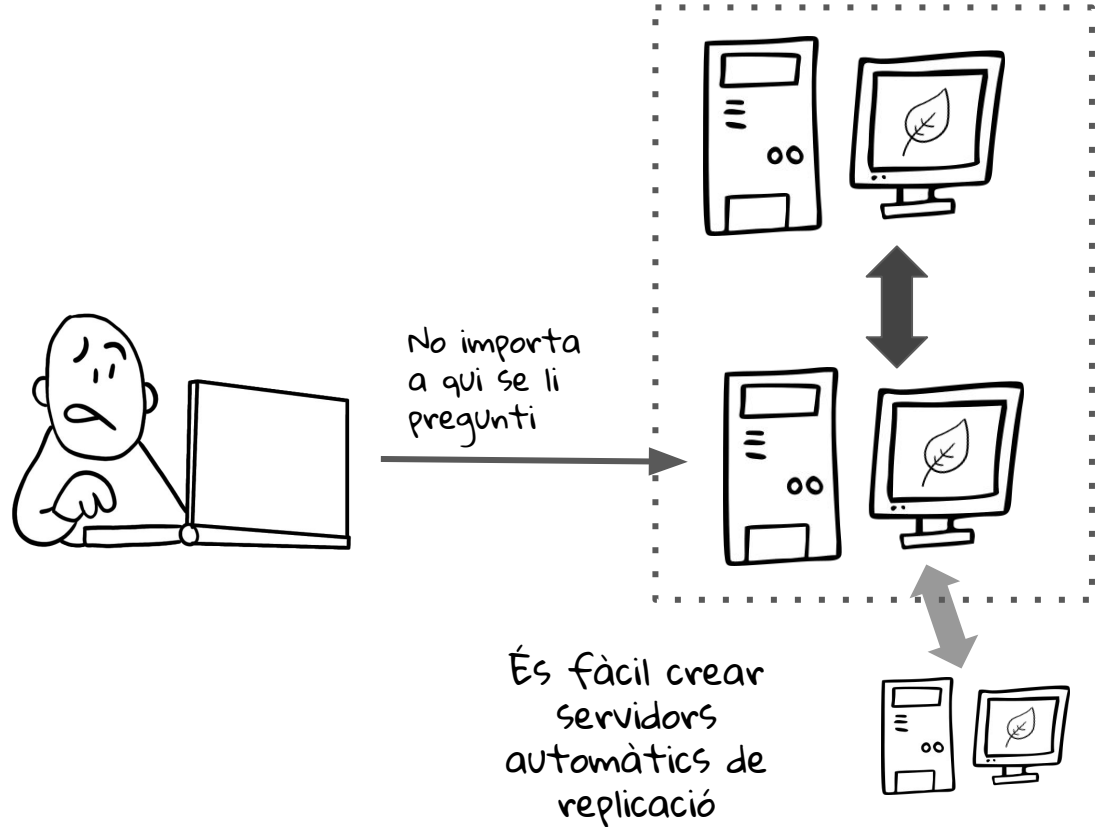
Semblant a
JSON

De 16 MB com a màxim
(ampliable amb GridFS)



```
{  
  "nom": "xavier",  
  "edat": 44,  
  "lletres": [ "A", "B", "C" ],  
  "correus": {  
    "personal": "jo@un.cat",  
    "feina": "xa@udg.edu"  
  }  
}
```

Sharding automàtic i transparent

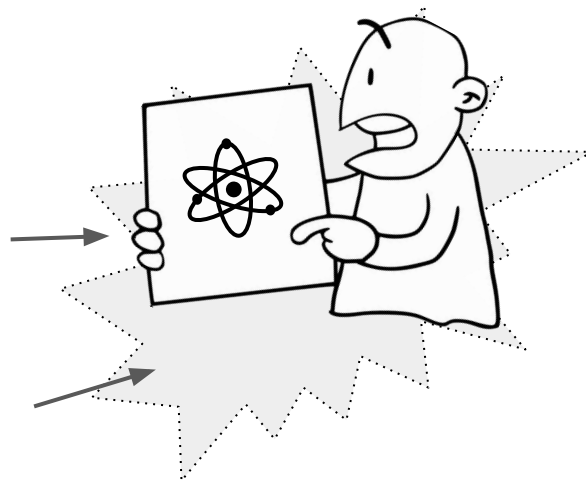


Transaccions

Fa operacions atòmiques en documents

A efectes pràctics
sovint no fan falta
transaccions

Però des de la versió
4.2 n'hi ha



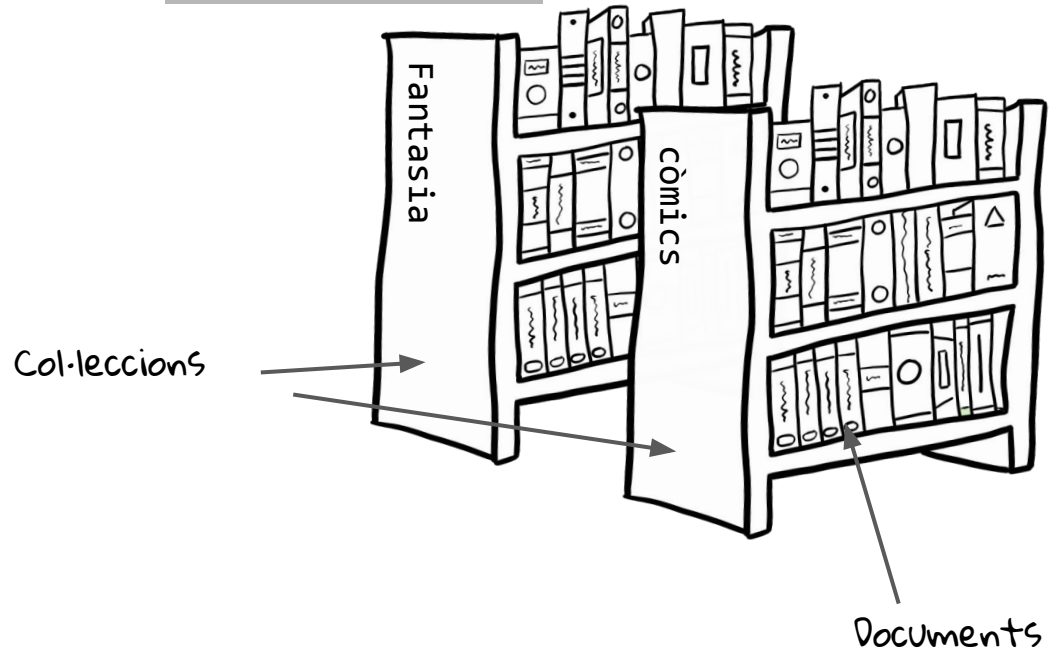
Les Dades



A diferència del que fan les bases de dades relacionals, les dades en Mongo s'emmagatzemen en forma de documents

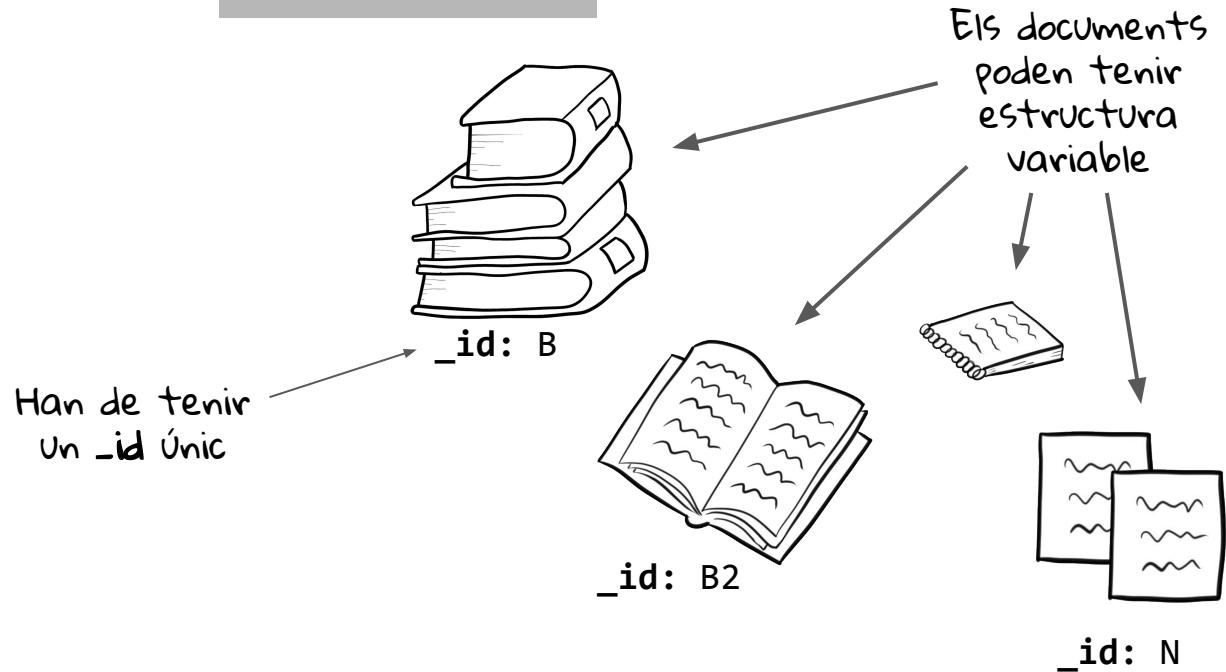
Bases de dades

Bases de dades



Estructura dels documents

Collections



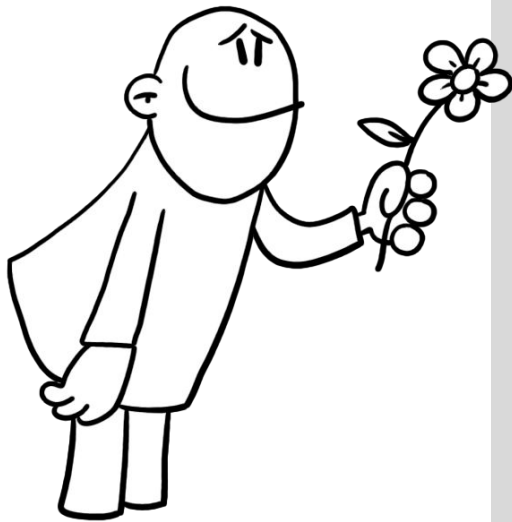
Format BSON

Han de tenir
un `_id` únic

```
{  
  _id: 1233444451,  
  nom: "xavier",  
  edat: 44,  
}  
{  
  _id: 1233444452,  
  operacio: "divisió",  
  numerador: 36,  
  denominador: 12,  
  resultat: 3  
}
```

Relacions

Normalment es fan
explícitament
(repetint les dades)

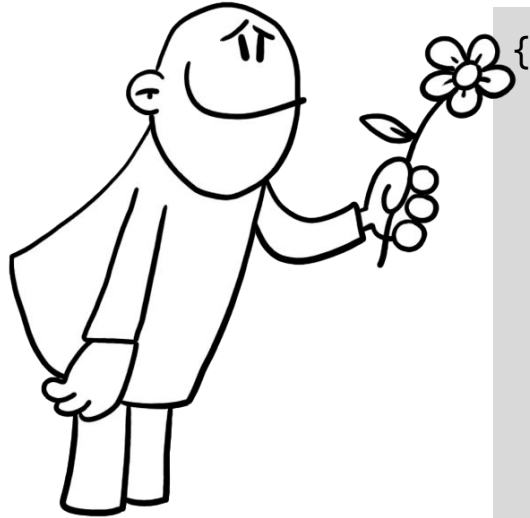


```
{  
  _id:23451667 ,  
  data: 23-05-2018,  
  jugadors: [  
    { nom: Pere, rànk: 2 },  
    { nom: Iu, alies: 'pito',  
      rànk: 3 },  
  ]  
  jugades [  
    { 1, "pedra", "pedra" },  
    { 2, "tiso", "tiso" },  
    { 3, "tiso", "pedra" }  
  ]  
}
```



No és una base de
dades relacional!!

No importa que hi
hagi dades repetides
en la base de dades

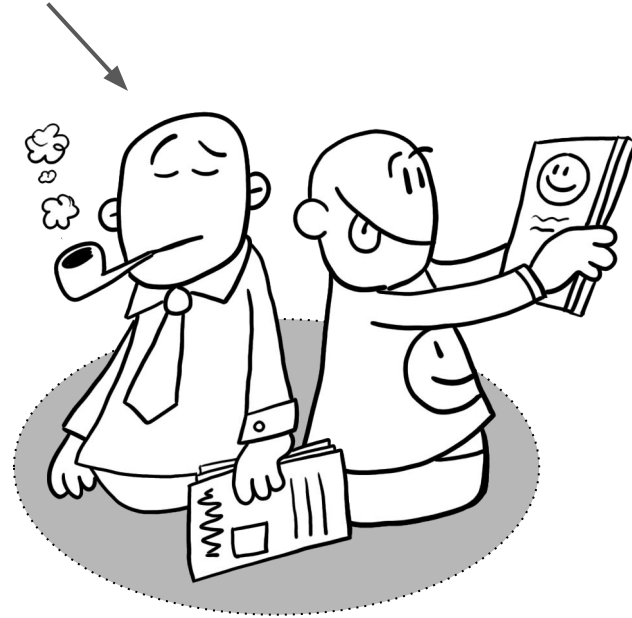


Es poden fer amb
referències als `_id`

```
{  
  _id: 23451667,  
  data: 23-05-2018,  
  jugadors: [  
    ObjectId("52ffc4a5d8520000"),  
    ObjectId("52ffc4a5d8d00001")  
  ],  
  jugades [  
    ObjectId("52ffc41116d00027"),  
    ObjectId("52ffc4a348d06043"),  
    ObjectId("52ffc4a348d06043")  
  ]  
}
```

Estructura segons l'ús previst

Els documents
s'organitzen segons l'
ús que se'ls hi donarà



L'accés serà majoritàriament per
mòdul o per nom?

```
{ mòdul: Programació,  
  notes: [  
    { nom: Pere, nota: 5 }  
    { nom: Pau, nota: 8 }  
    { nom: Maria, nota: 10 } ]  
}  
{ mòdul: Base de dades  
  notes: [  
    { nom: Pere, nota: 6 },  
    { nom: Maria, nota: 7 } ]  
}
```

```
{ nom: Pere,  
  notes: [  
    {mòdul: Programació, nota: 5 },  
    {mòdul: Base de dades, nota: 6}]  
}  
{ nom: Pau,  
  notes: [  
    {mòdul: Programació, nota: 8}]  
}  
{ nom: Maria,  
  notes: [  
    {mòdul: Programació, nota: 10},  
    {mòdul: Base de dades, nota: 7}]  
}
```

Es pot carregar tot
en una sola operació
de lectura

Es pot desar tot en
una sola operació
d'escriptura



```
{  
  _id:23451667 ,  
  data: 23-05-2018,  
  jugadors: [  
    { nom: Pere, rànding: 2 },  
    { nom: Iu, alies: 'pito',  
      rànding: 3 },  
  ]  
  jugades [  
    { 1, "pedra", "pedra" },  
    { 2, "tisoires", "tisoires"},  
    { 3, "tisoires", "pedra"}  
  ]  
}
```

Llenguatge per fer CRUD

No està
basat en SQL

```
SELECT Direccio  
FROM mapa  
WHERE Direccio='Nord'
```

```
db.Mapa  
.Find({direccio:"Nord"})
```

cap a
l'esquerra

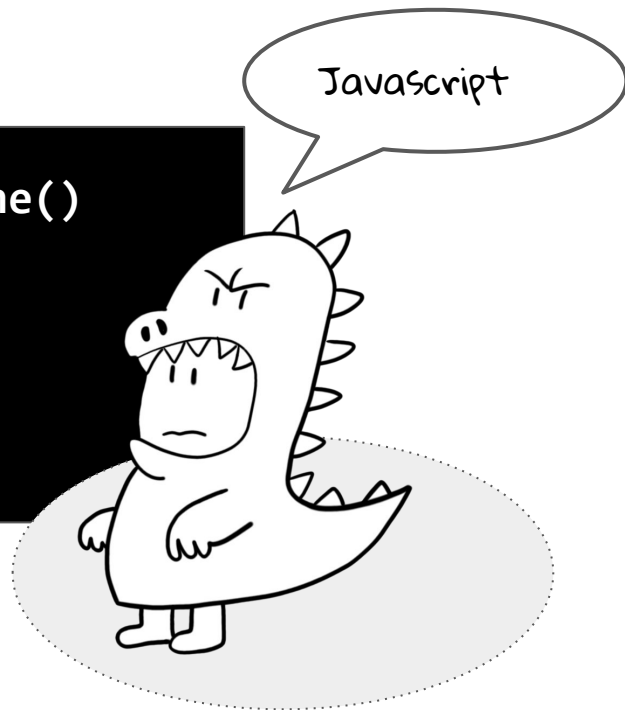
?



Consola

Com els llenguatges
SQL hi ha una
consola per
interactuar-hi

```
> db.colors.findOne()
```



Afegir documents



Si no hi ha `_id`, es crea
automàticament

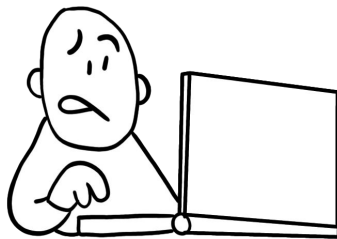


```
> db.usuaris.insertOne({nom:"Pere"})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("607023c22a")
}
> db.usuaris.insert({_id:3,
                      nom:"Iu",
                      cognoms: "Pi"})
WriteResult({ "nInserted" : 1 })
```

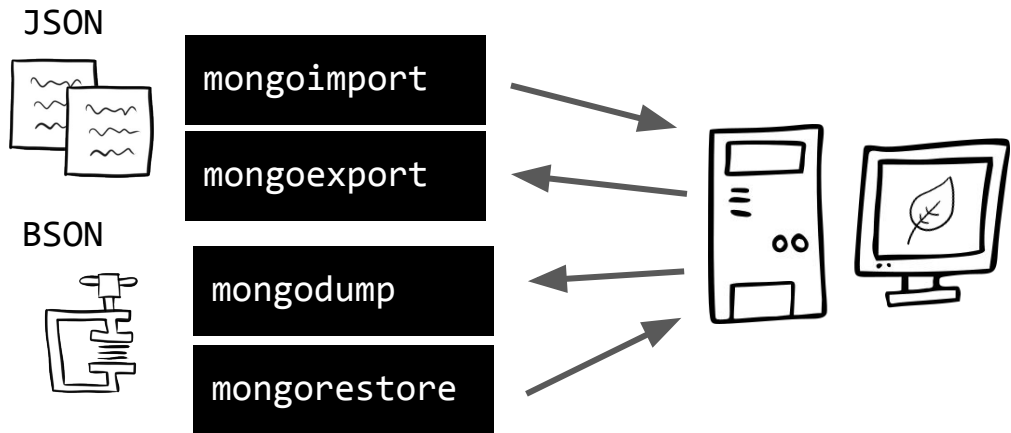


```
> db.usuaris.insertMany(  
  [ {nom:"Frederic"},  
    {nom:"Tomeu", cognom: "Pi" },  
    {cognom: "Ferrer", edat: 42 }  
  ]  
)
```

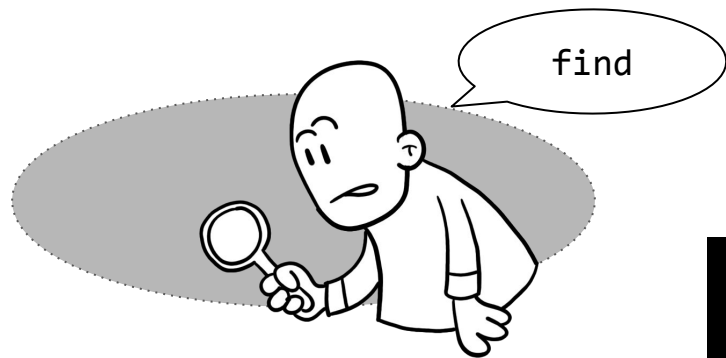
Importar i exportar



```
$ mongoimport -d database -c usuaris --file dades.json --jsonArray
```



Cerca bàsica



```
SELECT *  
FROM usuaris  
WHERE nom=Pere
```

Condicció



```
> db.usuaris.find({nom:"Pere"})  
{  
  "_id": ObjectId("123456789"),  
  "nom": "Pere",  
  "cognoms": "Pi"  
}  
{  
  "_id": ObjectId("2345645667"),  
  "nom": "Pere",  
}
```

```
{  
  nom: "Pere",  
  cognoms: "Pi",  
  edat: 40,  
  adreça: {  
    carrer: "Nou",  
    municipi: "Girona"  
  },  
  idiomes: [ "català", "anglès", "francès" ]  
}
```

Les condicions són
documents JSON

Per defecte es
relaciona amb
"and"

```
{ nom:"Pere" }
```

```
{ nom:"Pere", cognoms:"Pi" }
```

```
{ "adreça.municipi":"Girona" }
```

```
{ "idiomes.2" : "anglès" }
```



```
{  
  nom: "Pere",  
  cognoms: "Pi",  
  edat: 40,  
  adreça: {  
    carrer: "Nou",  
    municipi: "Girona"  
  },  
  idiomes: [ "català", "anglès", "francès" ]  
}
```

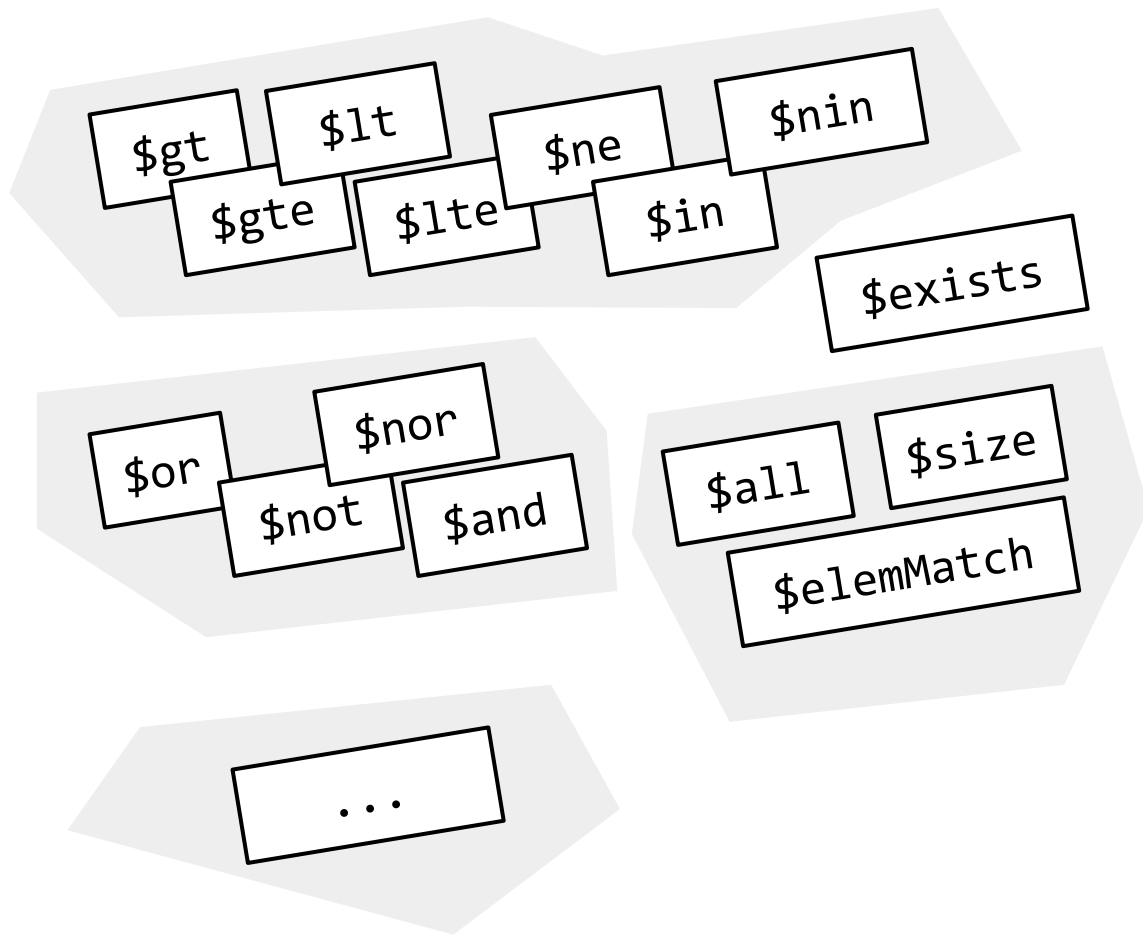
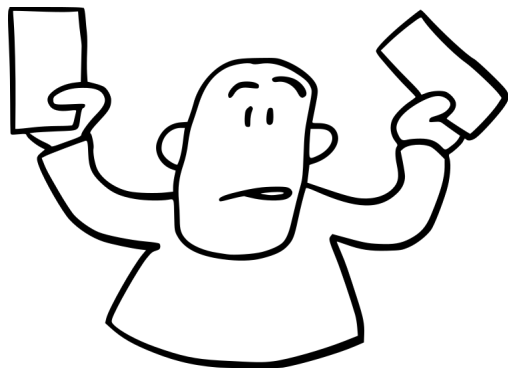
Hi poden haver
operadors

```
{ edat: { $gt: 30 } }
```

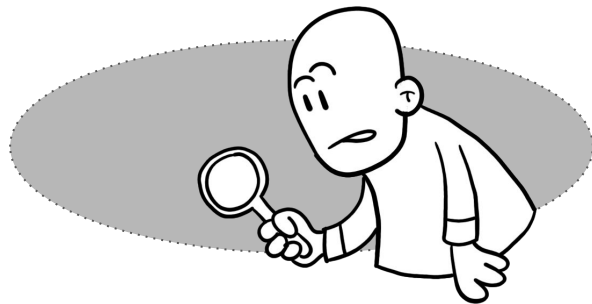
```
{ edat: { $or: [ $lt: 10, $gt: 20 ] } }
```

```
{ email: { $exists: false } }
```





Projeccions de resultats

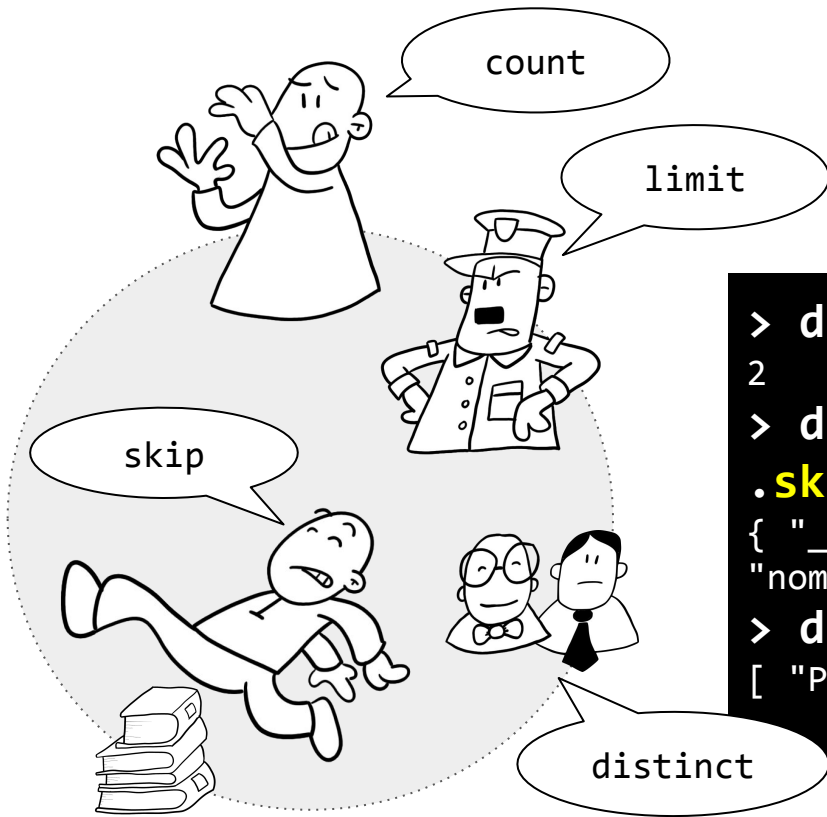


```
> db.usuaris.find({},  
                    {cognoms: true})  
{ "_id" : ObjectId("5fc63ccee3b18ed8940497a"),  
  "cognoms" : "Pi" }  
{ "_id" : ObjectId("5fc63db7ef3b18ed8940497c"),  
  "cognoms" : "Perez" }  
> db.usuaris.find({"nom":"Pere",  
                    {cognoms: true  
                      _id: false})
```

El valor de l'_id
sempre surt si no
li diem el contrari

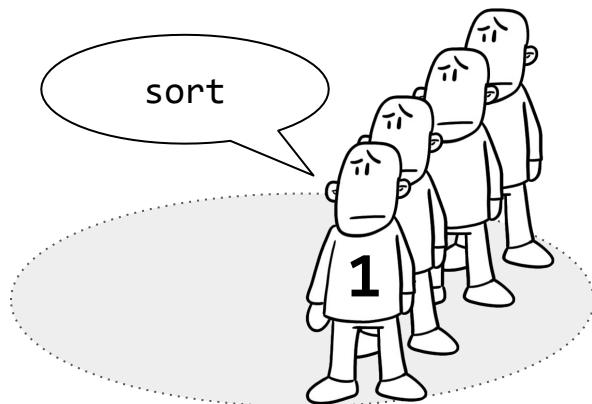
SELECT cognoms FROM usuaris

Comptar, saltar, limitar ...



```
> db.usuaris.count({nom:"Pere"})
2
> db.usuaris.find({nom:"Pere"})
.skip(1).limit(1)
{ "_id" : ObjectId("5fc63cf2ef3b18ed8940497b"),
  "nom" : "Pere", "cognoms" : "Puig" }
> db.usuaris.distinct("nom")
[ "Pere", "Joan", "Frederic" ]
```

Ordenar



+l: ascendent
-l: descendent



```
> db.usuaris.({nom:"Pere"})  
      .sort({cognoms: 1})  
{ "_id" : ObjectId("5fc63db7ef3b18ed8940497c"),  
  "nom" : "Pere", "cognoms" : "Perez", "telèfon" :  
  "505050" }  
{ "_id" : ObjectId("5fc63cceef3b18ed8940497a"),  
  "nom" : "Pere", "cognoms" : "Pi" }
```



2021