

Bases de dades NoSQL



Andreu Moll Doménech

NoSQL NOT Only SQL

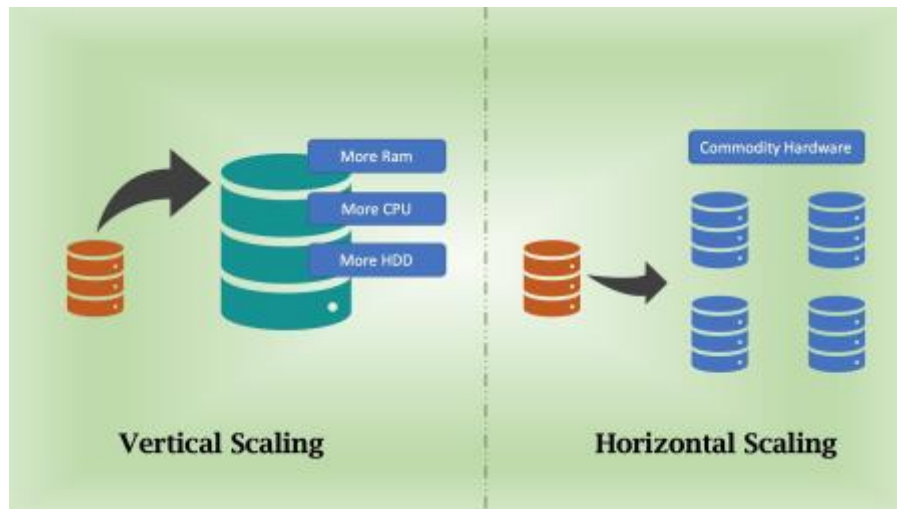
Motivació

Creixement exponencial de les dades provocat per l'aparició de les xarxes socials *Facebook, Twitter, Youtube*, etc.

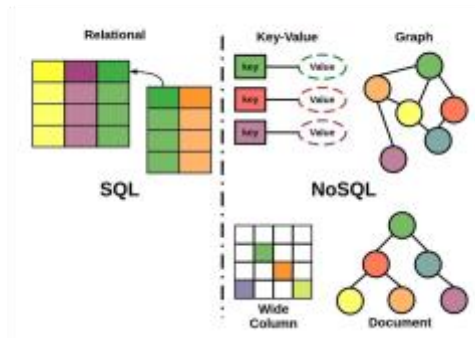
Que no arreglaren incorporant més màquines i més potents...



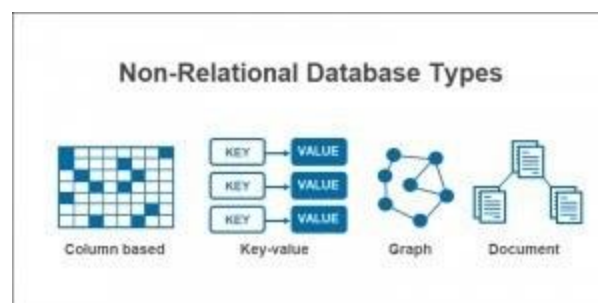
Escalabilitat *horitzontal* afegint més nodes (i no més potència en cada node preexistent que seria la modalitat vertical)



Model de dades

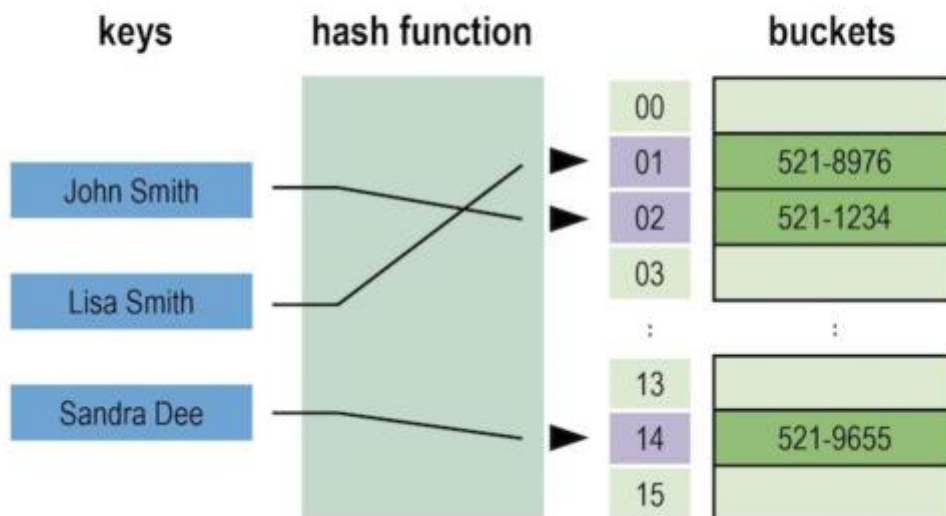


Model de dades (II)



Model key-value

Cassandra



En xarxes socials els “buckets” solen ser BLOB’s

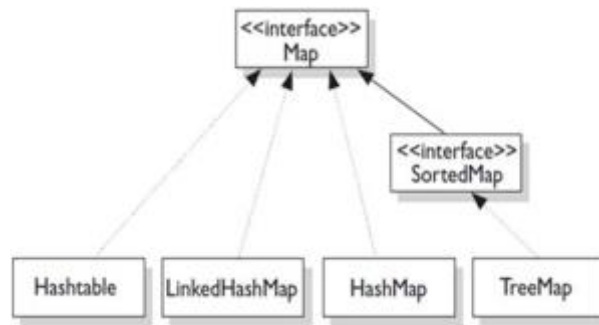


Maps en JAVA

No bidireccionalitat

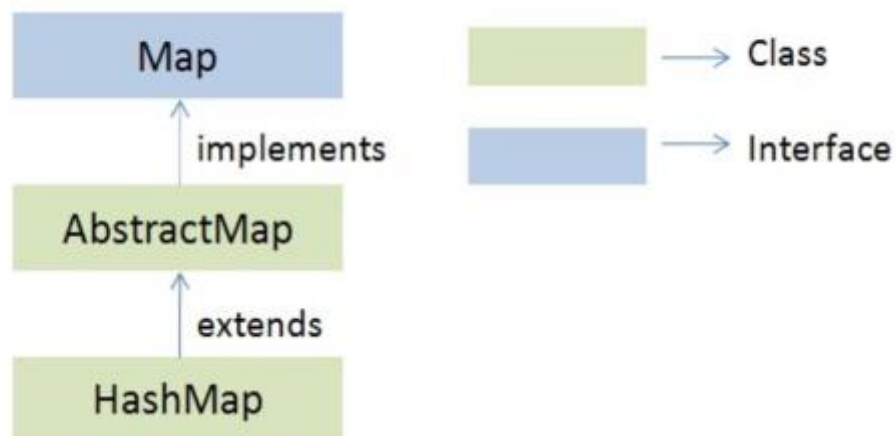
KEYS	VALUES
Jan	327.2
Feb	368.2
Mar	197.6
Apr	178.4
May	100.0
Jun	69.9
Jul	32.3
Aug	37.3
Sep	19.0
Oct	37.0
Nov	73.2
Dec	110.9
Annual	1551.0

Herència Implementacions



Herència

Map i HashMap



Definició

Un map....

Sense ordre

Ni duplicats

Ni nulls

Exemple

```

HashMap<Integer, String> Equip = new HashMap<Integer, String>();
Equip.put(1, "Casillas"); Equip.put(15, "Ramos");
Equip.put(3, "Pique"); Equip.put(5, "Puyol");
Equip.put(11, "Capdevila"); Equip.put(14, "Xabi Alonso");
Equip.put(16, "Busquets"); Equip.put(8, "Xavi Hernandez");
Equip.put(18, "Pedrito"); Equip.put(6, "Iniesta");
Equip.put(7, "Villa");

```

Recurrent amb iterator

```

.....!
Iterator it = Equip.keySet().iterator(); while(it.hasNext()) { Integer key = it.next();
System.out.println("Clave: " + key + " -> Valor: " + map.get(key)); }
Un resultat seria..

```

Sense ordre lògic...

```

Clave: 16 -> Valor: Busquets
Clave: 1 -> Valor: Casillas
Clave: 18 -> Valor: Pedrito
Clave: 3 -> Valor: Pique
Clave: 5 -> Valor: Puyol
Clave: 6 -> Valor: Iniesta
Clave: 7 -> Valor: Villa
Clave: 8 -> Valor: Xavi Hernandez
Clave: 11 -> Valor: Capdevila
Clave: 14 -> Valor: Xabi Alonso
Clave: 15 -> Valor: Ramos

```

Model jeràrquic documents MongoDB

```

.....!

```

Col1	Col2	Col3	Col4
Data	Data	Data	Data
Data	Data	Data	Data
Data	Data	Data	Data

Document 1
{ "prop1": data, "prop2": data, "prop3": data, "prop4": data }

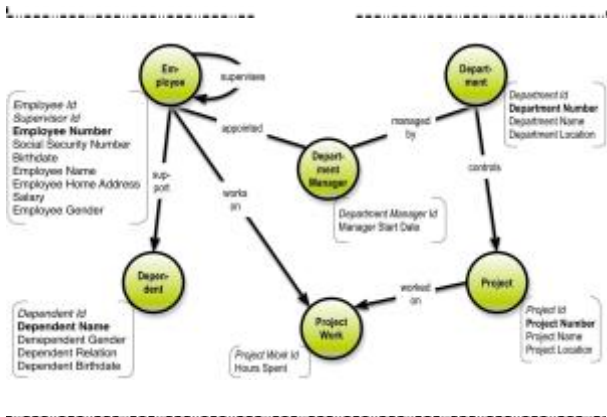
Document 2
{ "prop1": data, "prop2": data, "prop3": data, "prop4": data }

Document 3
{ "prop1": data, "prop2": data, "prop3": data, "prop4": data }

Document en format JSON, XML

Model graf

Neo4J



Casos

Sistema	Empresa
Cassandra	Twitter, Facebook...
RedIs	Flickr, Instagram, Github
Neo4J	Infojobs...
MongoDB	Sorceforge, CERN,...