PAC 2

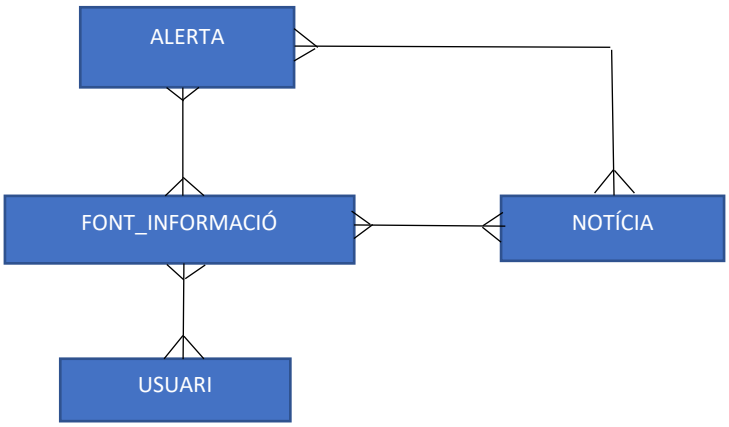
M2.952 - Arquitectures de bases de dades no tradicionals

Alumne: Carlos A. García Pérez

Exercici 1

El model de dades. En particular, estudiar la conveniència d'un model relacional o un de NoSQL i raonar quin model seria el més adequat.

Un possible model entitat-relació podria ser el següent:



En aquest cas, la conveniència de un model relacional o un NoSQL depèn de:

- **La necessitat de disponibilitat de lectura i escriptura.** El nombre d'usuaris és enorme, així que tindrem un nombre de lectures molt alt. El nombre de fonts d'informació és també enorme; això pot fer que el nombre d'escriptures (en particular notícies) sigui també molt alt.
- **Conflictes d'escriptura.** En aquest cas, no es preveuen gaires conflictes d'escriptura en les notícies. Si optem per replicar les dades, en cas d'escriptures simultànies de la mateixa notícia, és de suposar que seran coherents (la font d'informació i la alerta generada seran coherents). En cas d'escriptures simultànies d'usuaris, es pot aplicar una estratègia seqüencial: si un usuari es dóna d'alta a dues fonts de forma simultània a diferents nodes, el resultat final és que tindrà associades les dues.
- **Incoherències de lectura.** Si optem per un model NoSQL, un usuari donat d'alta a una font d'informació pot no tenir accés a les seves darreres alertes generades en el període de temps en el que s'actualitzen els nodes. No sembla un gran problema, però s'hauria de mesurar aquest temps.
- **Transaccionalitat.** No sembla necessari garantir la transaccionalitat donats els requeriments. Una operació d'escriptura pot no completar-se en una mateixa transacció i no ser cap problema (es pot reprendre posteriorment).

Tenint en compte aquests punts, crec que la millor estratègia és crear dos entitats a un model NoSQL: una d'usuaris en la que es puguin consultar les seves opcions de configuració, les fonts d'informació a les que està subscrit i les alertes generades. Una altra de fonts d'informació, notícies i alertes generades. Les consultes d'alertes per part dels usuaris es faran a la primera entitat; en cas de voler llegir les notícies, ja s'haurà de consultar la segona. El càlcul d'alertes es farà sobre la segona entitat. La primera entitat es particionarà seguint una estratègia de sharding per a facilitar la lectura simultània de molts d'usuaris. La segona es replicarà amb un model master-slave per a garantir la disponibilitat.

2. A partir de l'arquitectura de distribució escollida:

a. Indicar els avantatges i inconvenients que té l'estratègia de fragmentació (partició) escollida respecte a altres.

Com a avantatge de la fragmentació d'usuaris: els usuaris es poden distribuir segons una funció de hash. Això ens permetrà poder créixer horitzontalment. Com a inconvenient, quan es generi una alerta, s'haurà de propagar per tots els nodes, cercant els usuaris donats d'alta a la font d'informació.

b. Indicar els avantatges i inconvenients que té l'estratègia de replicació escollida respecte a altres.

La replicació master-slave garanteix la disponibilitat del servei, a més de facilitar la transaccionalitat. Afavoreix la velocitat de lectura, que també es pot fer sobre l'esclau. L'inconvenient és que no ens permet créixer horitzontalment. Per aconseguir això, s'hauria de combinar amb una estratègia de shard sobre les fonts.

3. El model transaccional més adequat per als requisits de l'aplicació descrita

És suficient amb garantir les transaccions a nivell de document. Cada vegada que es crea un nou element (notícia, alerta, usuari, font), basta que s'actualitzin els documents de forma individual. No és necessària una política ACID; un model BASE sembla molt més adient.

Exercici 2

A partir de la lectura del llibre NoSQL Distilled indiqueu què us semblen les següents afirmacions.

Afirmació 1

El model de replicació mestre-esclau és una solució molt útil per escalar un sistema si s'han de realitzar lectures intensives sobre les dades.

Cert. Les escriptures es realitzen sobre el Mestre, però es poden fer lectures simultànies sobre els esclaus (pàgina 40 del llibre).

Afirmació 2

El model de replicació peer-to-peer té com a principal avantatge assegurar la consistència de la informació.

Fals. És molt complex garantir la consistència d'escriptura si cada dada no té un únic mestre. La principal avantatge és que es pot escriure de forma simultània a diferents nodes (pàgina 42 del llibre).

Afirmació 3

La consistència de replicació consisteix en que dins d'una sessió d'usuari hi hagi una consistència de tipus "read-your-writes".

Fals. Això és consistència de modificació. La consistència de replicació és que la dada és la mateixa independentment de la partició accedida. Normalment es garanteix mitjançant quòrum (la dada és igual al menys a la meitat més ú dels nodes). Pàgines 57 i 58 del llibre.

Afirmació 4

Una funció reductor combinable és una funció de reducció que té com a peculiaritat que la seva sortida coincideix amb la seva entrada.

Cert. Una funció de combinació és, en essència, una funció de reducció. En aquest cas, pot utilitzar-se la mateixa funció de combinació com a reducció final. En aquest cas, coincideix la sortida amb l'entrada (pàgina 70 del llibre).

Exercici 3

Punt 1

Hi ha vàries solucions que compleixen amb el requisit de garantir la transaccionalitat (una modificació només afecta a un agregat). Una és crear un document per usuari i mes; la key seria la concatenació de la clau d'usuari i mes, mentre que les dades serien el que es sol·licita: nombre de missatges, nombre de descàrregues i nombre de visualitzacions. Aquesta solució generaria molts de documents, el que no sembla necessari.

Una altra solució és un document per usuari i un array de mesos on s'inclogui la resta d'informació. D'aquesta forma comprimim més l'informació.

```
{
	"key": "joan_coll_pons_jcoll@universitat.bal",
	"months_contribution": [{
		"201901": {
			"messages": 1,
			"downloads": 5,
			"visualizations": 3
		},
		"201902": {
			"messages": 3,
			"downloads": 0,
			"visualizations": 3
		}
	}]
}
```

Punt 2

Crearem un agregat per a cada fòrum, que sembla la unitat bàsica que se'ns demana. La clau serà l'especificada: nom del fòrum + alfanumèric. Dedins agruparem la informació a un array de dies, que inclourà la informació demanada: nombre de fils oberts, nombre de rèpliques, nombre d'estudiants que han contribuït i nom de l'estudiant que més ha contribuït.

```
{
	"key": "forum_prac_2_nosql_167ABBS654A",
	"daily_contribution": [{
		"20190101": {
			"openTh": 23,
			"replies": 65,
			"students": 3,
			"name": "John Doe"
		},
		"20190102": {
			"openTh": 24,
			"replies": 72,
			"students": 4,
			"name": "John Doe"
		}
	}]
}
```

Punt 3

Crearem un agregat per a cada recurs. És la unitat bàsica que se'ns demana. La clau serà l'especificada: nom del recurs + id alfanumèric. Dedins agruparem la informació a un array de mesos, que inclourà la informació demanada: nombre de descàrregues, nombre d'estudiants que se l'han descarregat, data de la darrera descàrrega (basta el dia, ja que tenim el mes a la clau del registre) i data amb més descàrregues (basta el dia, ja que tenim el mes a la clau del registre).

```
{
	"key": "resource_key_value_AQSD6766",
	"monthly_contribution": [{
		"201901": {
			"downloads": 102,
			"students": 65,
			"lastDate": 31,
			"popularDate": 18
		},
		"201902": {
			"downloads": 23,
			"students": 12,
			"lastDate": 25,
			"popularDate": 1
```

```
                    }
                }]
    }
```

Exercici 4

L'operació de Map és l'encarregada de tractar cada agregat per tal de recopilar les dades. Si hi ha més d'un node, recopilaria la informació de tots els nodes involucrats. En el nostre cas, es recolliran les dades escriptor, editorial, exemplars i guanys. Després d'aplicar la funció de Map quedaria de la forma:

Autor	Editorial	Exemplars	Guanys
Miguel de Cervantes	Anaya	45	546
Miguel de Cervantes	Anaya	34	234
William Shakespeare	Santillana	23	256
William Shakespeare	Alianza	15	345
Bram Stocker	Alianza	34	456
Anónimo	Cátedra	35	245
Don Juan Manuel	Cátedra	27	346

La funció de Reduce es l'encarregada d'agrupar les dades tal i com es demana. Seria l'equivalent dels SUM de la query i del GROUP BY. Quedaria de la forma:

Autor	Editorial	Exemplars	Guanys
Miguel de Cervantes	Anaya	79	780
William Shakespeare	Santillana	23	256
William Shakespeare	Alianza	15	345
Bram Stocker	Alianza	34	456
Anónimo	Cátedra	35	245
Don Juan Manuel	Cátedra	27	346

A aquest exemple, només "reduïm" (les dades que podem agrupar) els registres de Miguel de Cervantes de Anaya.