

MongoDB



Table des matières

Section 1 Installation.....	3
1.1 But.....	3
1.2 Enoncé.....	3
Section 2 C'est quoi mongoDB ?.....	6
Section 3 Premiers pas : 1ère base.....	7
3.1 But.....	7
3.2 Enoncé.....	7
Section 4 Utiliser MongoDB Compass.....	10
4.1 But.....	10
4.2 Utilisation.....	10
Section 5 Travail sur un jeu de données.....	11
5.1 But.....	11
5.2 Utilisation.....	11
Section 6 Retour d'expérience MongoDB.....	13
6.1 But.....	13
6.2 Résumé suite aux exercices.....	13
Section 7 Utilisation de Python.....	16
7.1 But.....	16
7.2 Mise en route.....	16
7.3 Un cas plus concret.....	16
Section 8 Se protéger des pannes.....	19
8.1 Utilisation de ReplicaSet.....	19
8.2 Mise en pratique : mise en place d'un Primary et 2 Secondary.....	20
8.3 La collection oPlog.....	22
Section 9 Le Sharding sous MongoDB.....	23

Section 1 Installation

1.1 But

- Installer MongoDB
- Découverte de l'environnement de travail

1.2 Enoncé

Le site <https://openclassrooms.com/fr/courses/4462426-maitrisez-les-bases-de-donnees-nosql/4474601-decouvrez-le-fonctionnement-de-mongodb> est utilisé comme source d'informations.

- Télécharger Mongo Community à partir de ce site : <https://www.mongodb.com/try/download/community>

- Installer.

Utiliser le bouton Custom

Changer le chemin d'installation vers [C:\mongoDB](#)

(le choix par défaut vers [c:\Program Files](#) pourrait par la suite poser des problèmes de droit d'accès)

Laisser le choix d'installation de MongoDB Compass

Pour la fenêtre suivante, si elle s'ouvre, faire ces choix (décocher)

Privacy Settings

To enhance the user experience, Compass can integrate with 3rd party services, which requires external network requests. Please choose from the settings below:

☐ **Enable Product Feedback Tool**
Enables a tool for sending feedback or talking to our Product and Development teams directly from Compass.

☒ **Enable Geographic Visualizations**
Allow Compass to make requests to a 3rd party mapping service.

☐ **Enable Crash Reports**
Allow Compass to send crash reports containing stack traces and unhandled exceptions.

☐ **Enable Usage Statistics**
Allow Compass to send anonymous usage statistics.

☐ **Enable Automatic Updates**
Allow Compass to periodically check for new updates.

With any of these options, none of your personal information or stored data will be submitted.
Learn more: [MongoDB Privacy Policy](#)

[Start Using Compass](#)

- Ensuite aller sur le site

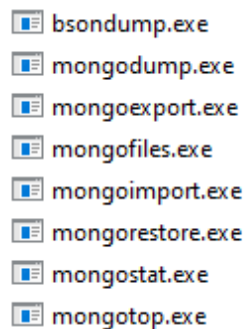
<https://www.mongodb.com/download-center/database-tools/releases/archive>

Récupérer l'archive zip (exemple pour Windows)

- Windows x64
 - Archive: [mongodb-database-tools-windows-x86_64-100.9.4.zip](#)

Dézipper le fichier obtenu.

- Récupérer le contenu de bin



et le placer dans C:\mongoDB\bin

Avec le task manager vérifier dans la partie Services que le service MongoDB est démarré.

MongoDB	13852	MongoDB Server (...)	En cours d'exécution
---------	-------	----------------------	----------------------

Pour information la commande suivante dans une fenêtre Cmd est équivalente :

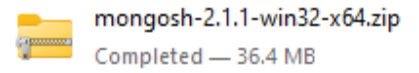
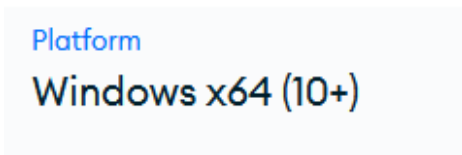
```
tasklist /FI "IMAGENAME eq mongod.exe"
```

S'il ne l'est pas le lancer par double clic sur c:\mongoDB\bin\mongod.exe

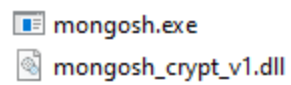
Installer ensuite mongo shell à partir du site suivant :

<https://www.mongodb.com/try/download/shell>

Exemple pour Windows, utiliser le zip :



Recopier les fichiers visibles dans bin



dans C:\mongoDB\bin

Section 2 C'est quoi mongoDB ?

- C'est un gestionnaire de Bases de données noSQL
- C'est un système de gestion orienté Documents
- La syntaxe des documents est au format JSON
- il gère la tolérance aux pannes : notion de replicaSet
- il permet le changement d'échelle horizontal : le sharding

Section 3 Premiers pas : 1ère base

3.1 But

- Créer une base
- Créer un utilisateur
- Créer une collection
- insérer quelques données
- le lien https://www.youtube.com/watch?v=0ad-8_3Gqx0 a été utilisé

3.2 Enoncé

Toutes les commandes MongoDB demandées dans cet exercice et qui ont abouti sont à placer dans un fichier .txt qui porte votre nom. Ex : dupond.txt

Si vous souhaitez y placer plus d'information un document word est aussi possible. Me rendre dans ce cas le document en .pdf

Vous pouvez y ajouter vos commentaires.

Ce document sera à envoyer en fin de journée à l'adresse contact@pragma-tec.fr

Ouvrir une fenêtre commande.

Se déplacer dans c:\mongodb\bin\

taper mongosh pour lancer un client mongoDB de type console

Nous allons utiliser la documentation à partir de ce lien

<https://docs.mongodb.com/manual/core/databases-and-collections/>

Dans ce qui suit la police Courier New signifie des choses tapées dans la fenêtre console en bleu) ou vues dans la fenêtre console (en rouge)

3.2.1 Créer une base : use

`use mesclients` crée la base mesclients

`db` montre les bases, ici `mesclients`

3.2.2 Créer un utilisateur

Regarder la documentation pour créer un utilisateur de la base

Céer l'utilisateur :

- votre nom ou pseudo
- un mot de passe
- le rôle ["readWrite", "dbAdmin"]

3.2.3 Créer une collection

Créer la Collection "clients" – équivalent de Table "clients" en SQL

Chaque « ligne » de la Collection se nomme un « document »

```
db.createCollection("clients")
```

3.2.4 Insérer les documents – équivalent de lignes en SQL

pour utiliser `db.clients.insert()` consulter la documentation.

Insérer :

- un seul doc avec ces données

Nom du champ	valeur
nom	Mac Cain
prenom	John
tel	12345678

- 2 documents à la fois avec ces données

Nom du champ	valeur
nom	<votre nom>
prenom	<votre prenom>
tel	1122334455

Nom du champ	valeur
nom	Dufour
prenom	Alain
tel	4455667788
societe	Altran
niveau	10

Les données sont au format JSON. Le site <https://jsoneditoronline.org> peut aider à la saisie et compréhension

Pour regarder le contenu actuel :


```
db.clients.find()  
ou  
db.clients.find().pretty()
```

3.2.5 Modifier un document

`db.clients.update()` est à utiliser

- 1) Ajouter à Mac Cain la société Microsoft
- 2) Enlever la donnée niveau à Dufour
- 3) Pour Dufour, renommer le champ `societe` en `company`

3.2.6 Chercher des informations

`db.clients.find()` va être utilisé

- 1) lister les documents de clients, uniquement pour les champs `nom` et `tel`
- 2) lister les nom et prenom des personnes dont le nom est Mac Cain OU la société est Altran

3.2.7 Trier des données

Trier les documents par ordre croissant des noms

Section 4 Utiliser MongoDB Compass

4.1 But

- Utiliser un outil d'administration MongoDB

4.2 Utilisation

Lancer l'outil MongoDB ComPass.

Se logger par :

```
mongodb://localhost:27017
```

Lire le contenu de la collection clients

Si souhaité, utiliser l'IHM pour modifier ou ajouter des valeurs.

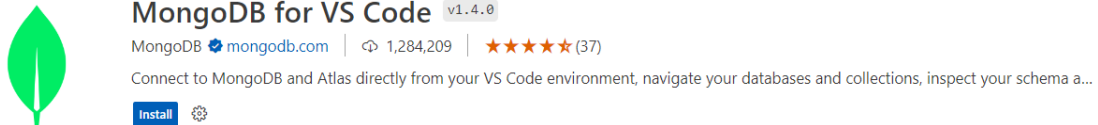
Section 5 MongoDB et VS Code

5.1 But

- Utiliser l'extension MongoDB pour VS Code

5.2 Utilisation

Dans VS code, onglet extension taper Mongoddb for VS Code



L'installer

L'utiliser.

Utiliser la chaîne de connexion

```
mongodb://localhost:27017
```

Section 6 Travail sur un jeu de données

6.1 But

- Utiliser un jeu de données pour y faire des recherches

6.2 Utilisation

De nombreux sites web sur le Big Data proposent des jeux de données pour entraînement, à destination des data Scientists.

Nous allons utiliser une de ces bases.

Base de restaurants de New York disponible ici :

<https://s3-eu-west-1.amazonaws.com/course.oc-static.com/courses/4462426/restaurants.json.zip>



Dé zipper le fichier.

Puis charger cette BD dans mongoDB .

Dans la fenêtre Cmde, sortir du client mongoDB par un exit

taper la commande

```
mongoimport --db new_york --collection restaurants <chemin vers>\restaurants.json
```

Relancer dans la fenêtre Cmd le client mongoDB par `mongosh`

Le lien suivant de openclassroom peut être utile pour la suite, prenez le temps de le parcourir et de faire des essais :

<https://openclassrooms.com/fr/courses/4462426-maitrisez-les-bases-de-donnees-nosql/4474606-interrogez-vos-donnees-avec-mongodb>

- 1) chercher les restaurants de Manhattan dont la cuisine est American dont le code postal est 10025. Se limiter aux 5 premiers restaurants
- 2) requête pour connaître leur nombre
- 3) reprendre le résultat du 1 pour n'afficher que le nom, le voisinage (borough) et le type de cuisine
- 4) quels sont les différents type de cuisine que l'on peut trouver à New york ?
- 5) Quel est le pourcentage de type de cuisine « American » vis à vis des autres types ?



Section 7 Retour d'expérience MongoDB

7.1 But

- Après cette série d'exercice, un résumé et présentations d'autres aspect de MongoDB

7.2 Résumé suite aux exercices

Le serveur mongoDB est implémenté sous forme d'un service dont le nom est mongod

Le client en mode console est l'exécutable mongosh.exe (sous Windows)

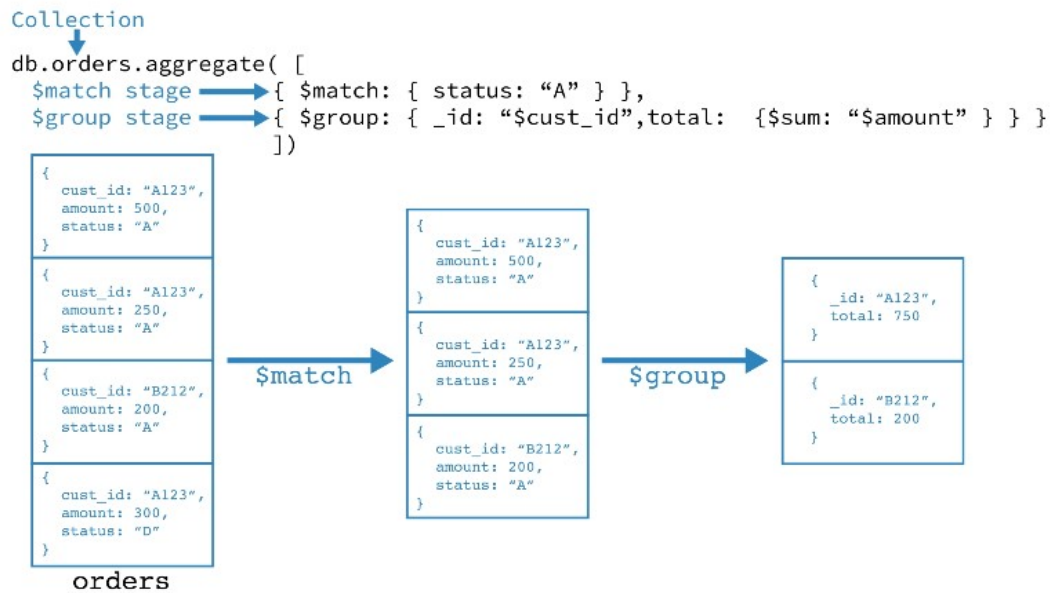
Les principales commandes qui ont été utilisées :

- use : à la fois création d'une base de données ou déplacement vers une base de données existante
- la syntaxe des paramètres des diverses commandes est globalement au format JSON :
 - chaîne simple lorsqu'il n'y a pas besoin de formuler un couple clé/valeur
exemple :
`db.createCollection("clients")`
`db.restaurants.distinct("cuisine")`
 - au format JSON avec { } lorsque des couples clé :valeur doivent être fournis
exemple :
`db.clients.insert({nom:"Mac cain",prenom:"John"})`
`db.clients.find({}, {_id:0,nom:1,tel:1})`
 - la syntaxe avec des [] permet de définir un tableau de clés :valeurs eux mêmes définis par des { }
 - certaines instructions nécessitent des mots clés commençant par '\$' : ce sont des opérateurs
<https://docs.mongodb.com/manual/reference/operator/query/>

Le lien <https://docs.mongodb.com/manual/reference/sql-comparison/> est intéressant car fournit, pour ceux qui maîtrisent SQL, des correspondances SQL vers MongoDB

Le site <https://www.site24x7.com/tools/sql-to-mongodb.html> convertit des requêtes SQL en MongoDB

7.2.1 L'agrégation



7.2.2 Les index

Dans une base SQL existe la notion d'index.

Sur le principe, analogue à un index dans un livre, l'index permet de retrouver rapidement une information dans une grosse BD.

Ex MySQL :

```
CREATE INDEX idx_pname
ON Persons (LastName, FirstName);
```

En contre partie l'écriture ou la mise à jour d'une information sera plus lente.

En NoSQL cette notion existe également.

<https://docs.mongodb.com/manual/indexes/>

Sans index, lors d'un find() MongoDB réalise un scan sur l'ensemble de la collection.

Pour créer un index il faut se poser la question de quels vont être les champs de recherche lors d'un find() couramment utilisé sur un lot très volumineux de données.

L'index est alors fait pour ces champs . Ex :

```
db.products.createIndex(  
  { item: 1, quantity: -1 } ,  
  { name: "query for inventory" }  
)
```

Les requêtes `find()` qui suivront cette commande utiliseront en interne l'index de façon transparente pour le développeur... et de façon efficace si l'index a été bien choisi.

Section 8 Utilisation de Python

8.1 But

- Utiliser le connecteur Python pour créer un client MongoDB
- Réaliser diverses opérations
- s'approprier le contenu d'une BD mongoDB et faire des modifications

8.2 Mise en route

Si votre poste de travail ne dispose pas de Python ou dispose d'une ancienne version Python 3.x ou 2.x télécharger la fourniture Anaconda à l'adresse

Utiliser le site

<https://www.anaconda.com/products/individual>

Utiliser ensuite le site

https://www.w3schools.com/python/python_mongodb_getstarted.asp

pour réaliser les expériences :

- de création de base
- de création de collection
- opérations CRUD

8.3 Un cas plus concret

L'office du tourisme de la mairie de Paris dispose d'une base de données de type MongoDB dont la structure date un peu.

Le fichier source se trouve à l'adresse

https://s3-eu-west-1.amazonaws.com/course.oc-static.com/courses/4462426/tour-Pedia_paris.json.zip

Il nous est demandé de produire une autre base contenant les améliorations suivantes :

8.3.1 Définir le code postal

La plupart des documents dispose d'un champ address mais la plupart ne contiennent pas le code postal.

Exemple : "address" : "Route de Suresnes"

Nous sommes interrogés sur la faisabilité d'ajouter un champ séparé zipcode contenant cette valeur.

Il faut donc étudier si ce code peut être supprimé de address pour être déplacé vers zipcode lorsqu'il existe dans address.

Et lorsqu'il n'existe pas, le trouver et l'ajouter dans le nouveau champ.

Notre client nous mentionne ce lien pour peut être trouver l'information qu'il nous manque :

https://opendata.paris.fr/explore/dataset/adresse_paris/export/?location=12.48.85902,2.34899&basemap=jawg.streets&dataChart=eyJxdWVyaWVzIjpbeyJjb25maWciOnsiZGF0YXNldCI6ImFkcmVzc2VfcGFyaXMiLCJvcHRpb25zIjpb7fX0sImNoYXJ0cyI6W3siYWxpZ25Nb250aCI6dHJ1ZSwidHlwZSI6ImNvbHVtbilslmZ1bmMiOiJBVkcilCJ5QXhpcyl6Im5fc3FfYWQiLCJzY2llbnRpZmljRGlzcGxheSI6dHJ1ZSwiY29sb3IiOiIjMDAzMzY2In1dLCJ4QXhpcyl6ImJfYW5nbGUiLCJtYXhwb2ludHMiOjUwLjZjb3J0Ijoiln1dLCJ0aW1lc2NhbGUiOiIiLCJkaXNwbGF5TG5nZW5kljp0cnVlLCJhbGlnbk1vbnR0ljp0cnVlQ%3D%3D

ou plus précisément

https://opendata.paris.fr/explore/dataset/adresse_paris/download/?format=json&timezone=Europe/Berlin&lang=fr

Si vous récupérez un fichier au format Json pour MongoDB, regarder les options de mongoimport, notamment /jsonArray

Le sujet va peut être être coriace de temps en temps : il faudra que la solution face un lien entre « Square de Montsouris » vu d'un côté et « SQ DE MONTSOURIS » vu de l'autre. ... Mais nous sommes tenaces !

Faire le ménage dans la BD résultante : on ne souhaite que les lieux sur Paris ...

8.3.2 Restaurants autour de Notre Dame

On nous demande encore autre chose !

On souhaite placer dans un document Excel la liste des restaurants situés sur une carte dans un carré de côté de 2 km dont le centre est notre dame de Paris.

Les colonnes souhaitées sont :

- le nom du restaurant
- l'adresse
- le code postal

8.3.3 Livrables de cet exercice

Le livrable est un fichier compressé de type .zip

Dans un document pdf vous indiquerez :

- vos idées,

- votre démarche de recherche
- les essais
- si réalisable, un petit schéma et une explication de ce que vous allez produire

Au pdf vous ajouterez :

- votre/vos codes source Python
- Un export des 40 premiers documents de la nouvelle base MongoDB.

Section 9 Se protéger des pannes

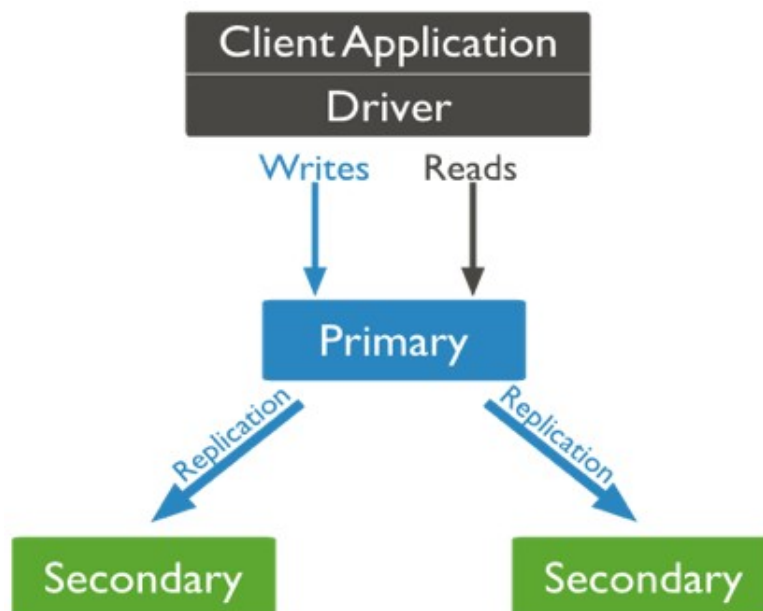
9.1 Utilisation de ReplicaSet

Ce sujet est traité ici <https://openclassrooms.com/fr/courses/4462426-maitrisez-les-bases-de-donnees-nosql/4474611-protégez-vous-des-pannes-avec-les-replicaset>

MongoDB offre une solution de tolérance aux pannes en permettant d'implémenter des serveurs MongoDB sur plusieurs machines.

Le principe utilisé est le principe maître/ esclaves (Primary / Secondary)

Cette architecture est appelée le ReplicaSet



Il faut y ajouter un système d'arbitrage « arbiter » qui va vérifier l'état du réseau et l'état du Primary. S'il y a une défaillance, un système de vote va élire un des Secondary en tant que Primary.

Un ReplicaSet doit contenir au minimum 3 serveurs, un Primary et 2 Secondary pour garantir une tolérance aux pannes.

Un ReplicaSet peut contenir jusqu'à 50 serveurs, avec 7 serveurs max inscrits pour pouvoir passer en Primary en cas de panne.

L'emplacement géographique des serveurs est à concevoir par le Data Architect.

Par exemple un serveur Primary et secondary dans le même rack pour un basculement rapide, des serveurs dispatchés dans plusieurs data centers pour les risques d'incendie, proches des centres d'analyse ...

Ceci est configurable.

C'est le serveur Primary qui reçoit les demandes d'écriture et les demandes de lecture – c'est le traitement par défaut.

Il est cependant possible par configuration de modifier le flux pour la lecture : un serveur Secondary peut être désigné pour répondre à la lecture, car par exemple plus près géographiquement des utilisateurs (mots Primary, primaryPreferred, Secondary, Nearest dans des fichiers de conf).

9.2 Mise en pratique : mise en place d'un Primary et 2 Secondary

Si un serveur mongod tourne déjà 'depuis les exercices précédents, nous allons l'arrêter.

Pour s'en assurer taper dans une fenêtre Cmd

```
tasklist /FI "IMAGENAME eq mongod.exe"
```

S'il existe le tuer par l'explorateur de tâche

Le fichier joint ReplicaSet.txt contient les commandes DOS décrites ci-après

Puisque nous allons lancer des serveurs mongod à la main, il ne faut plus qu'il soit considéré comme un service. Dans une Cmd sur mongoDB\bin

```
mongod --remove
```

Dans le répertoire mongoDB créer les sous répertoires data\RS1, data\RS2 et data\RS3

Lancer 3 serveurs mongod utilisant le même identifiant replicaSet myrepl avec 3 ports différents. (Voir fichier joint)

Pour information, quand l'on souhaite voir les ports d'écoute, sur Windows, 2 manières :

la Cmde netsat -a

ou ouvrir le « moniteur de ressources », onglet Ports d'écoute

Lancer ensuite dans une autre cmd un client mongo sur le port 27018 (cf fichier joint).

Utiliser la commande `rs.initiate()` pour faire connaître la liste des serveurs du replicaSet `myrepl`

Regarder l'état du système par `rs.status()`

Quitter ce client mongo par `exit` puis le relancer avec la commande
`mongo --host myrepl/localhost:27018,localhost:27019,localhost:27020`

9.2.1 Implémenter l'arbitre

Certaines documentations mentionnent le lancement de l'arbitre par la commande `rs.addArb()`

Le test fait ici ne l'a pas nécessité. Nous n'utiliserons pas cette commande.

9.2.2 Utiliser le système

Nous allons créer une BD et collection de test et vérifier qu'en cas de panne la BD continue à être disponible

Avec MongoDB Compass, se connecter sur le serveur 27018

Créer une BD de nom `testReplica` avec une collection de nom `clients`

Créer un ou deux documents contenant un nom et prénom

Si vous préférez, ceci peut être fait depuis un client console.

Se déconnecter de Compass

Si pas déjà fait, lancer un client mongo en donnant la liste des serveurs du replicaSet :

```
mongo --host myrepl/localhost:27018,localhost:27019,localhost:27020
```

faire à nouveau

`rs.status()` pour repérer le Primary

Nous allons provoquer l'arrêt du serveur 27018 : dans la fenêtre cmd de ce serveur, faire `Ctrl C` et attendre (peut prendre une minute)

Dans le client mongo faire à nouveau

`rs.status()` pour repérer le Primary et constater le changement

Regarder la BD créée précédemment.

9.3 La collection oPlog

Pour son fonctionnement, la fonction ReplicaSet entretient une collection qui stocke les opérations de mise à jour, pour être utilisées par les serveurs Secondary.

Le Primary écrit dans cette collection et les secondary vont y puiser, en mode pull, le contenu pour mettre à jour leur réplication.

Section 10 Le Sharding sous MongoDB

Lien utile :

<https://openclassrooms.com/fr/courses/4462426-maitrisez-les-bases-de-donnees-nosql/4474616-distribuez-vos-donnees-avec-mongodb>

Le replicaSet vu ci-avant permet d'assurer la tolérance aux pannes avec l'architecture Primary / Secondaire.

Pour passer à l'échelle horizontale il faut implémenter le Sharding.

Le Sharding consiste à partitionner la BD en plusieurs Bases appelées Shard, et les distribuer sur plusieurs machines – un cluster de machines.

Les blocs de données sont des Chunks et le Sharding est le process pour administrer/gérer ces Chunks.

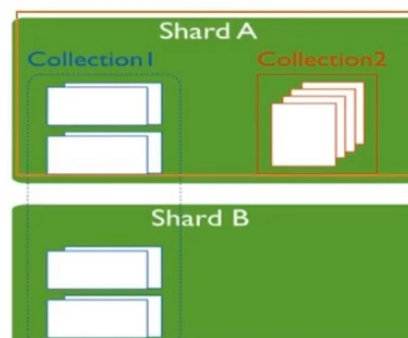
Mettre en place un sharding est complexe et coûteux, il nécessite des ressources matérielles et humaines.

Quand utiliser le sharding :

- la capacité de stockage dans une seule BD mongoDb devient critique
- les temps d'écriture et d'accès se font ressentir
- Ou dès la conception on détecte une capacité/performance exigée importante

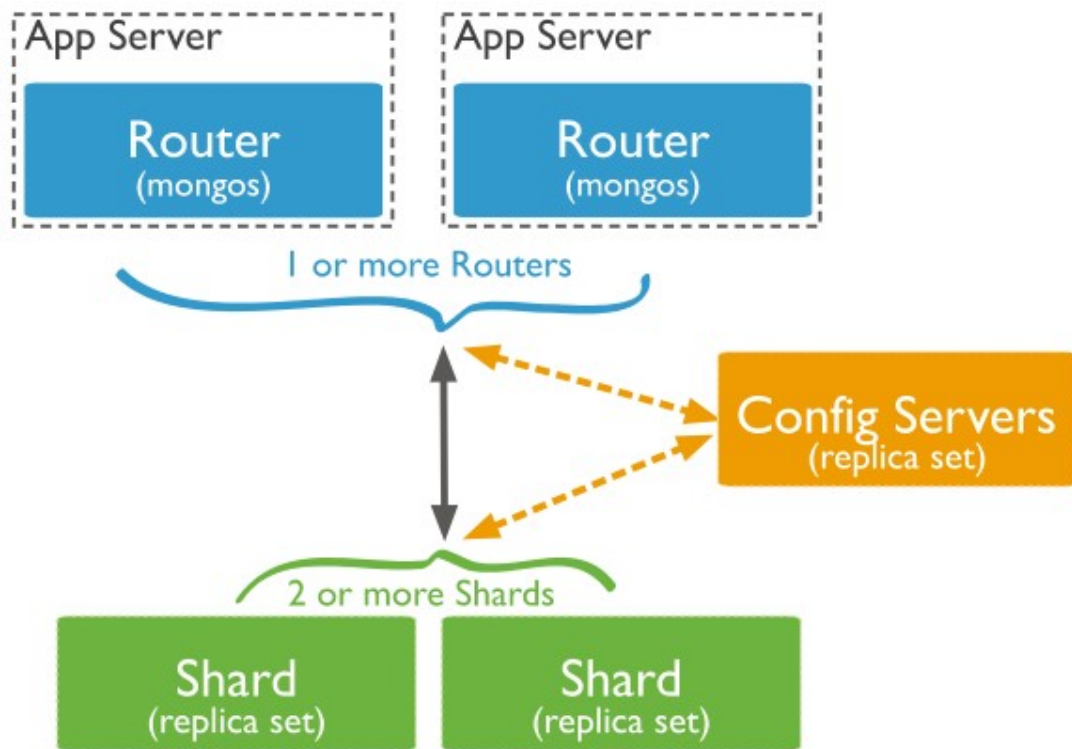
Qu'est ce qu'un Shard ?

- Un Shard est un serveur simple ou plutôt un ReplicaSet qui détient un bloc de données.



Comment parviennent les requêtes sur un Shard donné ?

- Un ensemble d'au minimum 2 routeurs dispatchent les requêtes vers le bon Shard
- Des serveurs de configuration , config Servers s'occupent de la gestion du réseau, la répartition des charges vers les Shards ...
Ils sont au nombre de 3 et organisés en replicaSet.



Architecture du sharding sous MongoDB

Nous n'allons pas implémenter cette architecture.

Elle consisterait à implémenter :

- 2 config server en replicaSet avec la commande

```
mongod --configsvr --replSet configReplSet --port 27019 --dbpath /data/config1
```

```
mongod --configsvr --replSet configReplSet --port 27020 --dbpath /data/config2
```
- à créer des shard par la commande `mongod --shardsvr`
- à lancer des routeurs mongos en replicaSet
-
- Puis à indiquer quelles collections ont souhaite mettre en Shard par
`sh.shardCollection("testDB.test",{"_id":1});`

Il faut avoir une stratégie sur le critère de placement d'information d'un document dans un Shard plutôt qu'un autre : le Shard Key

Le choix du Shard Key doit être fait selon :

- le schéma des données
- la façon dont les données vont être utilisées : tenter d'éviter que des requêtes récurrentes sollicitent trop de Shards.
Par exemple plutôt que de laisser `_id`, choisir `codePostal`, `ville` ...