

# Compléments sur le moteur

## NoSql CouchBase

Merci de développer les points ci-dessous théoriquement et pratiquement si possible :

- Modèles de données supportés
- Réévaluer la procédure d'installation du moteur et des utilitaires
- Architecture du moteur NoSql (avec des schémas expliqués)
- Méthode de partitionnement (avec des schémas expliqués)
- Méthode de réplication (avec des schémas expliqués)
- Montée en charge (avec des schémas expliqués)
- Gestion du ou des caches mémoire (avec des schémas expliqués)

# Table des matières

<a href="#">1.Modèles de données supportés.....</a>	<a href="#">3</a>
<a href="#">2.Réévaluer la procédure d'installation du moteur et des utilitaires.....</a>	<a href="#">3</a>
<a href="#">3.Architecture du moteur NoSQL (avec des schémas expliqués).....</a>	<a href="#">4</a>
<a href="#">4.Méthode de partitionnement (avec des schémas expliqués).....</a>	<a href="#">5</a>
<a href="#">5.Méthode de réplication (avec des schémas expliqués).....</a>	<a href="#">6</a>
<a href="#">6.Montée en charge (avec des schémas expliqués).....</a>	<a href="#">7</a>
<a href="#">7.Gestion du ou des caches mémoire (avec des schémas expliqués).....</a>	<a href="#">7</a>

## **1. Modèles de données supportés**

Couchbase est une base de données NoSQL de type document qui utilise le format JSON (JavaScript Object Notation) pour stocker les données. Les documents JSON sont flexibles et peuvent contenir différentes structures de données, allant des paires clé-valeur simples à des structures de données imbriquées plus complexes.

Cette flexibilité permet aux développeurs de modifier le schéma des données sans avoir à modifier la structure de la base de données, ce qui est un avantage majeur pour les applications évolutives et en rapide évolution.

En outre, Couchbase prend en charge des fonctionnalités telles que les requêtes ad hoc et l'indexation secondaire, ce qui permet une recherche et une récupération de données avancées.

## **2. Réévaluer la procédure d'installation du moteur et des utilitaires**

Vous pouvez installer Couchbase Server en téléchargeant le package approprié pour votre système d'exploitation (Windows, macOS, Linux) depuis le site officiel (<https://www.couchbase.com/downloads>). Une fois le package téléchargé, suivez les instructions d'installation fournies dans la documentation officielle : <https://docs.couchbase.com/server/current/install/install-intro.html>

Procédure d'installation de Couchbase en utilisant le lien fourni :

1. Allez sur le site officiel de Couchbase (<https://www.couchbase.com/downloads>) et téléchargez le package d'installation approprié pour votre système d'exploitation (Windows, macOS, Linux).
2. Une fois le package téléchargé, suivez les étapes d'installation spécifiques à votre système d'exploitation :
  - Pour Windows : Exécutez le fichier .exe téléchargé et suivez les instructions de l'assistant d'installation. Une fois l'installation terminée, démarrez le serveur Couchbase en utilisant le raccourci créé sur le bureau ou en cherchant "Couchbase Server" dans le menu Démarrer.
  - Pour macOS : Montez le fichier .dmg téléchargé en double-cliquant dessus, puis faites glisser l'icône Couchbase Server dans le dossier Applications. Pour démarrer Couchbase Server, ouvrez le dossier Applications et double-cliquez sur l'icône Couchbase Server.
  - Pour Linux : Téléchargez le package .deb ou .rpm approprié pour votre distribution Linux. Installez le package en utilisant les commandes d'installation standard (par exemple, `sudo dpkg -i` pour Debian/Ubuntu ou `sudo rpm -i` pour CentOS/RHEL). Une fois installé, démarrez Couchbase Server en utilisant la commande `sudo systemctl start couchbase-server`.
3. Après avoir démarré Couchbase Server, ouvrez un navigateur Web et accédez à l'adresse <http://localhost:8091> pour accéder à l'interface Web d'administration de Couchbase.
4. Sur l'interface Web d'administration, suivez les instructions pour configurer un nouveau cluster ou rejoindre un cluster existant. Configurez les paramètres du cluster, tels que la taille de la mémoire et les services disponibles (par exemple, Data, Index, Query, etc.).
5. Une fois la configuration du cluster terminée, vous pouvez créer des buckets pour stocker vos données et commencer à utiliser Couchbase Server.

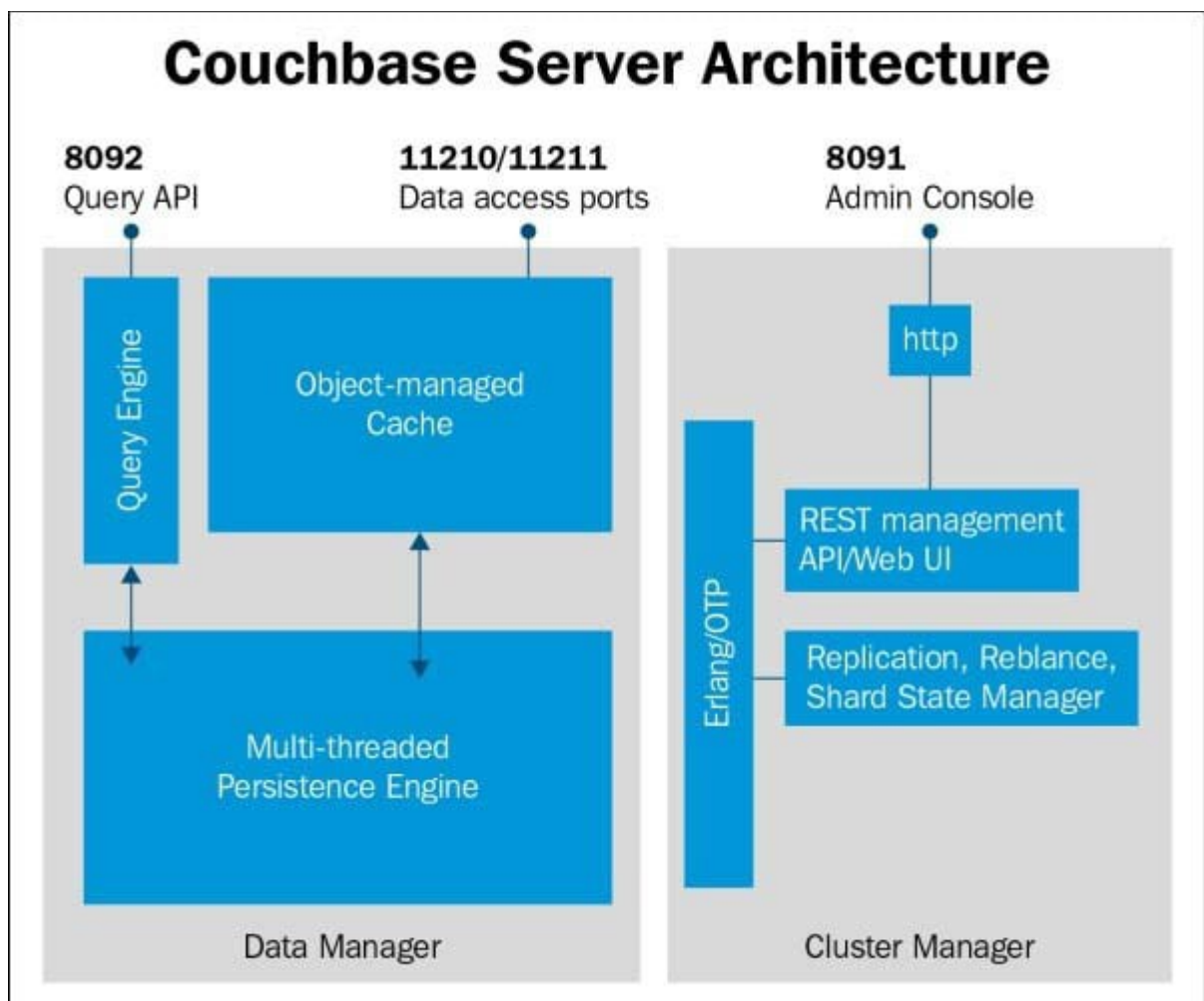
La documentation officielle de Couchbase propose des instructions détaillées pour chaque étape de l'installation et de la configuration :

<https://docs.couchbase.com/server/current/install/install-intro.html>

### **3. Architecture du moteur NoSQL (avec des schémas expliqués)**

Couchbase suit une architecture distribuée, ce qui signifie qu'elle est conçue pour répartir les données et les tâches sur plusieurs nœuds pour améliorer les performances et la fiabilité. Les composants clés de cette architecture comprennent les nœuds, les clusters, les buckets et les vBuckets. Un nœud est une instance individuelle de Couchbase Server, et un cluster est un groupe de nœuds qui partagent les mêmes configurations et fonctionnent ensemble. Les buckets sont des conteneurs de données qui sont répartis sur les nœuds du cluster, et chaque bucket est divisé en vBuckets, qui sont les unités fondamentales de partitionnement des données dans Couchbase. Cette architecture permet une distribution équilibrée des données et une répartition efficace de la charge de travail.

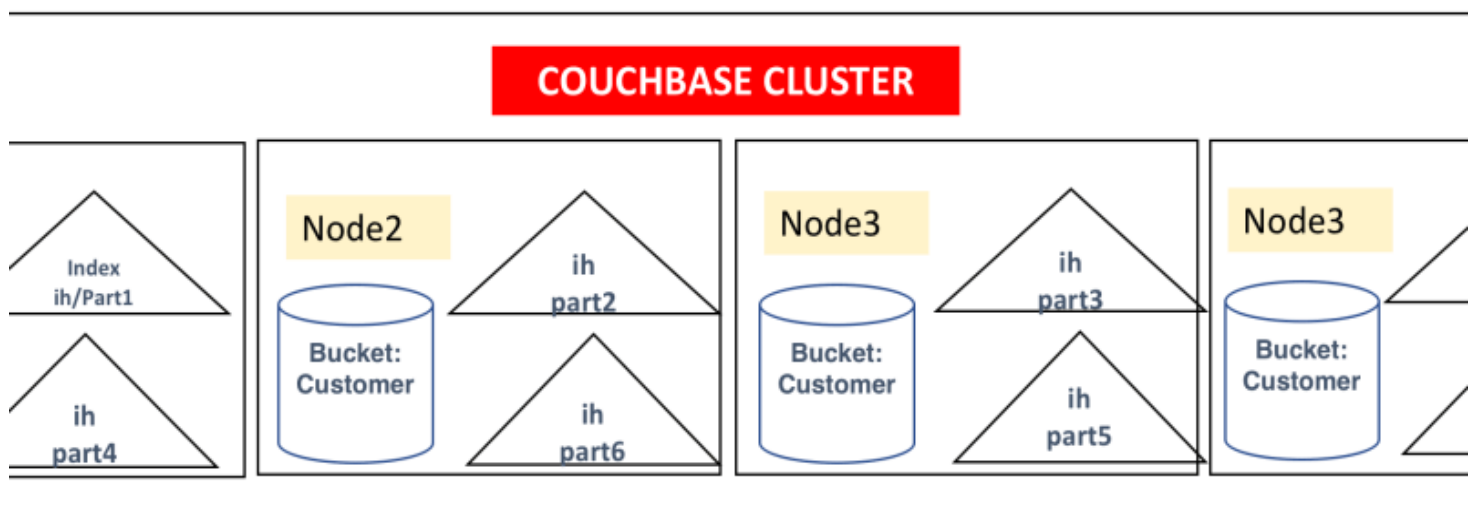
Architecture du moteur Couchbase



## 4. Méthode de partitionnement (avec des schémas expliqués)

Couchbase utilise un partitionnement basé sur vBucket pour équilibrer les données et la charge de travail entre les nœuds d'un cluster. Un bucket est divisé en plusieurs vBuckets, qui sont ensuite répartis équitablement entre les nœuds du cluster. Lorsque vous ajoutez ou supprimez un nœud du cluster, Couchbase effectue un rééquilibrage pour redistribuer les vBuckets entre les nœuds disponibles. Cette méthode de partitionnement permet une distribution uniforme des données et des tâches, ce qui améliore les performances et l'efficacité.

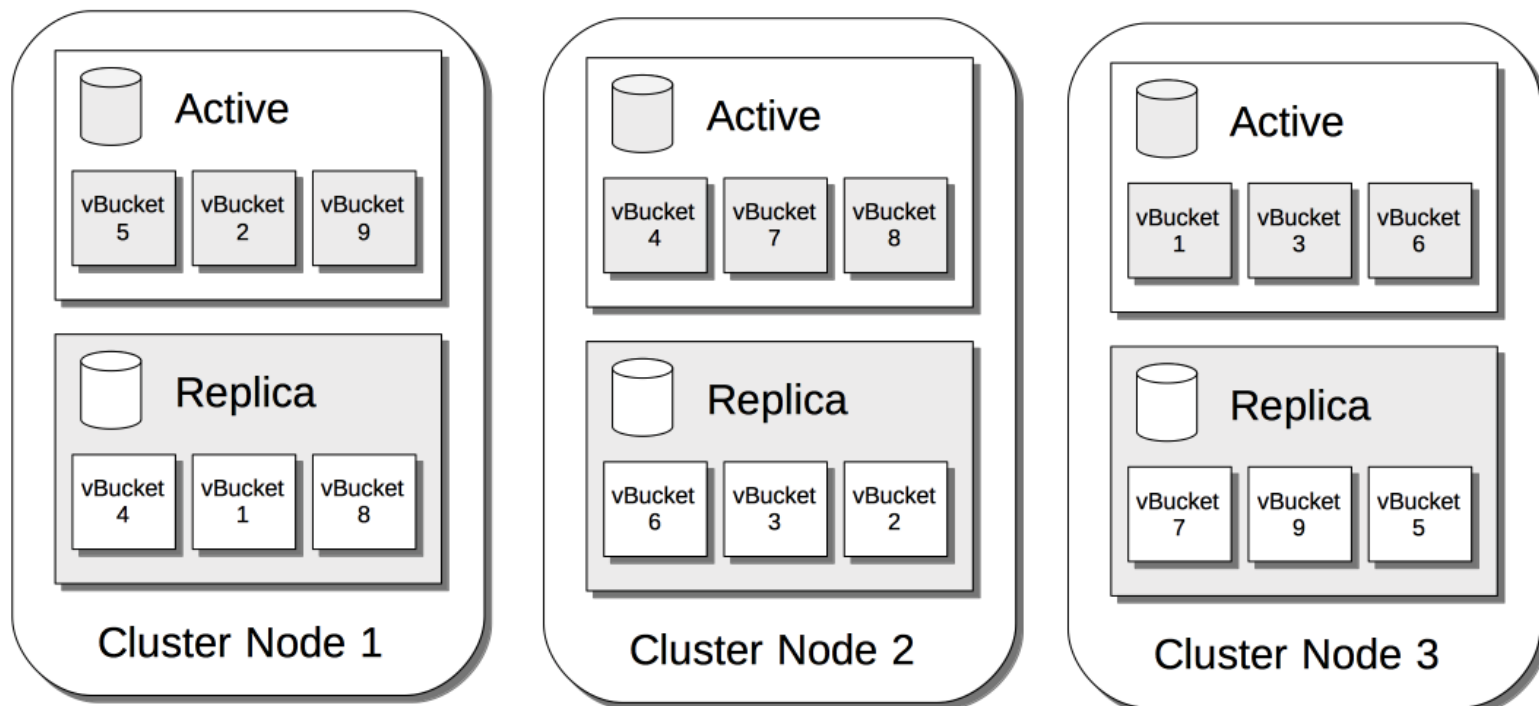
### Schéma de partitionnement



## 5. Méthode de réplication (avec des schémas expliqués)

Couchbase utilise une méthode de réplication active-active pour assurer la disponibilité et la résilience des données. Chaque vBucket dans le système a un réplica sur un autre nœud. Ces réplicas servent de sauvegardes en cas de défaillance d'un nœud. Si un nœud tombe en panne, le système peut rapidement promouvoir un réplica en vBucket actif, garantissant ainsi une disponibilité continue des données. En outre, les modifications apportées à un vBucket sont automatiquement répliquées sur ses réplicas, assurant ainsi la cohérence des données.

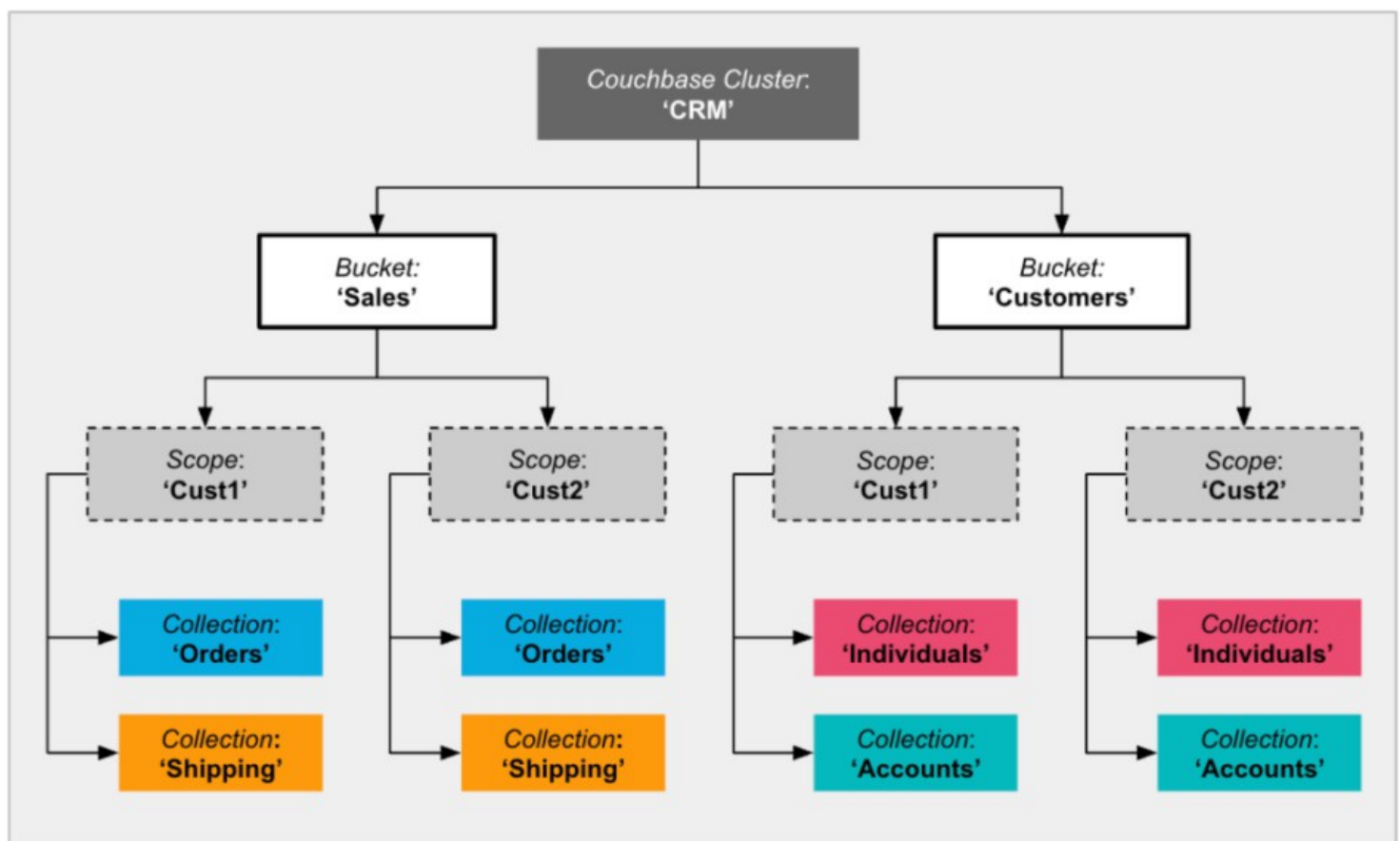
Schéma de réplication



## 6. Montée en charge (avec des schémas expliqués)

Couchbase est conçu pour l'évolutivité horizontale, ce qui signifie qu'il peut augmenter sa capacité en ajoutant simplement plus de nœuds au cluster. Lorsque vous ajoutez un nouveau nœud, Couchbase effectue un rééquilibrage pour répartir les vBuckets (et donc les données et les tâches) entre tous les nœuds disponibles. Cela permet à Couchbase de gérer efficacement l'augmentation de la charge de travail et de l'espace de stockage sans sacrifier les performances ou la disponibilité. L'évolutivité horizontale est un avantage majeur pour les applications qui connaissent une croissance rapide ou qui ont des besoins de performance élevés.

Schéma montée en charge





## 7. Gestion du ou des caches mémoire (avec des schémas expliqués)

Couchbase utilise un système de cache en mémoire appelé "Managed Cache" pour stocker les données les plus fréquemment utilisées en mémoire. Les données qui sont régulièrement accédées sont conservées en mémoire, ce qui permet d'éviter les accès coûteux au disque et d'améliorer les performances. En plus de cela, Couchbase utilise un mécanisme appelé "ejection" pour gérer le remplissage du cache. Lorsque le cache est presque plein, Couchbase commence à "éjecter" les éléments les moins récemment utilisés vers le disque pour faire de la place pour les nouveaux éléments. Cette stratégie d'éjection assure que les éléments les plus pertinents sont toujours conservés en mémoire pour un accès rapide.

Schéma caché mémoire

