

UN MODÈLE DE COÛT POUR LE NoSQL

Hao ZHANG

RECHERCHE COMPARATIVE DU MODÈLE DE COÛT POUR LA NORMALISATION ET LA
DÉNORMALISATION DE BD NOSQL

28 janvier 2019

Sommaire

① Sujet d'Étude

Contexte de la Recherche
Objectif de Projet

② État de l'art

③ Hypothèse

Normalisation et Dénormalisation
Objectif de recherche

④ Démarche Expérimentale

Modélisation
Quatre requêtes de CRUD

⑤ Résultats et conclusion

Contexte de la Recherche

Dans le BigData, les bases NoSQL répondent au problème de stockage et de distribution des données à large échelle pour souvent répondre à des problèmes de performances en centralisé.

De nombreuses solutions existent, et faire le bon choix lors de la définition des besoins d'un Système d'Information devient un choix crucial, car il est difficile et risqué de faire un changement technologique.

Objectif de Projet

Le but de ce projet de recherche est de définir un modèle de coût d'évaluation de requêtes sur des solutions NoSQL.

Ainsi, ce modèle de coût (accès réseaux et accès locaux) sera capable d'estimer pour chaque type de requête ce qu'il en coûtera, et ainsi de choisir la solution NoSQL qui minimise ce coût.

L'objectif de ce projet est donc de raffiner les modèles de coût existant dans la littérature pour avoir un modèle plus global, le plus précis possible.

Sommaire

① Sujet d'Étude

- Contexte de la Recherche
- Objectif de Projet

② État de l'art

③ Hypothèse

- Normalisation et Dénormalisation
- Objectif de recherche

④ Démarche Expérimentale

- Modélisation
- Quatre requêtes de CRUD

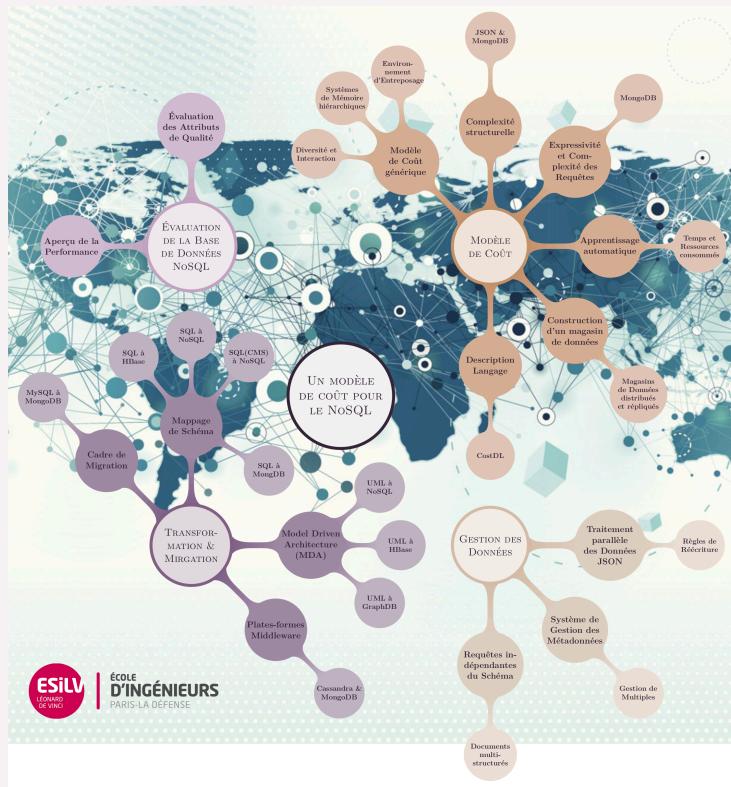
⑤ Résultats et conclusion

Classification des articles scientifiques

Ces articles peuvent être divisés en quatre catégories :

- 1 Évaluation de la Base de Données NoSQL
- 2 Transformation et Migration vers NoSQL
- 3 Modèle de coût d'une base de données
- 4 Gestion de la base de données NoSQL

Classification des articles scientifiques



Sommaire

① Sujet d'Étude

- Contexte de la Recherche
- Objectif de Projet

② État de l'art

③ Hypothèse

- Normalisation et Dénormalisation
- Objectif de recherche

④ Démarche Expérimentale

- Modélisation
- Quatre requêtes de CRUD

⑤ Résultats et conclusion

Normalisation

Normalisation

- Dans une base de données relationnelle, une forme normale désigne un type de relation particulier entre les entités.
- Le but essentiel de la normalisation est d'éviter les anomalies transactionnelles pouvant découler d'une mauvaise modélisation des données et ainsi éviter un certain nombre de problèmes potentiels tels que les anomalies de lecture, les anomalies d'écriture, la redondance des données et la contre-performance.

Dénormalisation

Dénormalisation

- La dénormalisation peut être définie comme la copie des mêmes données dans plusieurs documents ou tables afin de simplifier / optimiser le traitement des requêtes ou d'adapter les données de l'utilisateur à un modèle de données particulier. Elle permet de stocker des données dans une structure adaptée pour simplifier le traitement des requêtes. L'objectif pour les systèmes NoSQL est de proposer un schéma de stockage permettant d'éviter l'utilisation des jointures pour lesquelles les systèmes NoSQL sont peu adaptés en raison de l'architecture distribuée sous-jacente.
- Systèmes applicables : dépôts clé-valeur, bases de données document, bases de données colonnes.

Objectif de recherche

- Comparer la durée pendant laquelle les bases de données (normalisées ou dénormalisées) de différentes tailles répondent aux quatre principaux types de commandes(CRUD), c'est-à-dire comparer leurs modèles de coût sur temps.
- À l'avenir, sur la base des résultats obtenus, comparez les théories existantes et découvrez les lois mathématiques.

Sommaire

① Sujet d'Étude

Contexte de la Recherche
Objectif de Projet

② État de l'art

③ Hypothèse

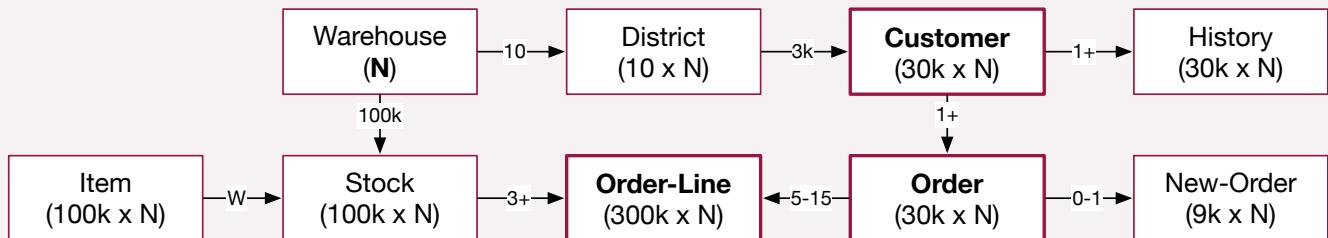
Normalisation et Dénormalisation
Objectif de recherche

④ Démarche Expérimentale

Modélisation
Quatre requêtes de CRUD

⑤ Résultats et conclusion

Schéma



Modélisation

Quatre requêtes de CRUD

Modélisation

1 db.getCollection("Semi_30").find({"o_c_id" : "30"})
2

Find Find 50 Documents 1 to 50

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 50 Documents 1 to 50

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 50 Documents 1 to 50

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 50 Documents 1 to 50

Semi-dénormalisation

Quatre requêtes de CRUD

- La requête C, créer une nouvelle commande avec 10 articles différents
- La requête R, chercher aléatoirement l'information d'un client par son nom et prénom en montrant l'id de la plus nouvelle commande.
- La requête U, modifier le temps de livraison des dix articles les plus anciens qui ne sont pas encore délivrés
- La requête D, supprimer aléatoirement d'une commande pas encore livrée.

Sommaire

① Sujet d'Étude

- Contexte de la Recherche
- Objectif de Projet

② État de l'art

③ Hypothèse

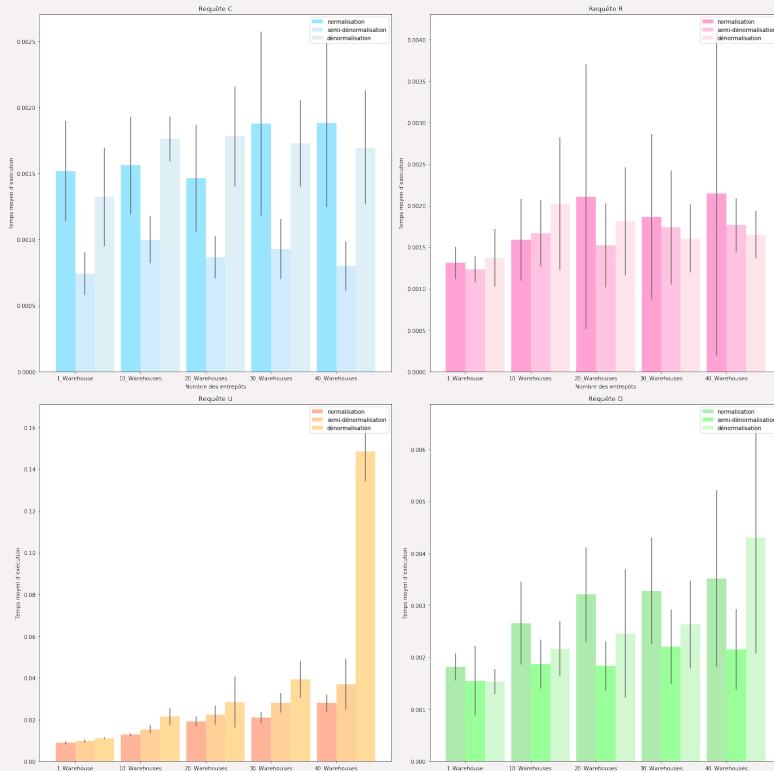
- Normalisation et Dénormalisation
- Objectif de recherche

④ Démarche Expérimentale

- Modélisation
- Quatre requêtes de CRUD

⑤ Résultats et conclusion

Résultats



Résultats



Conclusion

À cause de temps limité, la recherche actuelle sur le modèle de coût de la base de données NoSQL est encore au stade initial de la modélisation. Au cours du processus, j'ai rencontré de nombreux problèmes, tels que le temps de fusion de la base de données est trop long ou le langage de requête n'est pas assez précis. Une fois la modélisation actuelle mise au point, il est nécessaire de répéter les pratiques pour obtenir un résultat de test plus crédible. Quant à savoir si nous pouvons enfin trouver une loi crédible, vous devez la tester tout le temps.

Perspective

Pour un sujet axé sur la recherche, sa tâche ne doit pas se limiter à des tests, mais en être déduite. Par conséquent, pour ce sujet, lorsque les résultats des tests de tous les cent entrepôts de données sont établis, il est naturel de combiner la littérature scientifique pour en déduire des lois et même des formules de données. Les facteurs affectant cette formule peuvent être nombreux, tels que les performances de la machine ou la complexité de la structure.