Infrastructure de données

M2 - DataScience - Polytechnique

Projet

ESILV
nicolas.travers (at) devinci.fr

Chapitre

Table des matières

Chapitre 1

Modèle de données NoSQL

Le but de ce projet est définir le cadre de conception d'une application Big Data pour une base de données NoSQL orientée document. Pour des raisons d'homogénéisation, nous prendrons comme support MongoDB.

Chaque projet reposera sur un dataset réel trouvé sur le Web qui sera fusionné, partiellement ou intégralement, pour le faire passer à l'échelle sur une infrastructure distribuée.

Un rapport et une présentation finale doit être fournie.

1.1 Choix du jeux de données

Dans le cadre de ce projet, vous devrez choisir votre propre jeu de données. L'idée est de partir d'un jeu de données provenant d'une base de données relationnelle que vous pourrez trouver sur internet, comme celles présentes aux adresses suivantes :

- `https://relational.fit.cvut.cz/search` ;
- `https://toolbox.google.com/datasetsearch` ;
- `https://www.kaggle.com/datasets` ;
- `https://registry.opendata.aws/` ;
- `https://github.com/awesomedata/awesome-public-datasets#publicdomains` ;
- `http://millionsongdataset.com/` ;
- `https://www.google.com/publicdata/directory` ;
- `https://www.ncdc.noaa.gov/cdo-web/datasets`.

⚠Vous devez respecter les conditions suivantes :

1.1.1 Un schéma à plusieurs tables est nécessaire (minimum 4), interconnectées (jointures à faire) et si possible un volume de données conséquent (plus de 400Mo serait bien, sinon minimum 100Mo).

1.1.2 ⚠Certains dataset contiennent plusieurs tables, mais sans jointures. Etudiez bien le contenu du dataset pour faire votre choix.

Chaque jeu de données correspond à un cas d'usage particulier que vous aurez à spécifier pour définir l'infrastructure de données associée.

⚠Pour valider votre choix, vous devrez pour chaque groupe vous inscrire sur Moodle/groupes - dataset correspondant (si le dataset manque, demandez-le)

1.2 Cas d'usage

Pour pouvoir orienter les choix des étapes suivantes, il va falloir étudier les cas d'usage sur votre jeu de données. Nous prendrons deux vues distinctes :

- *End-User view* : Définir, en langage courant, 4 types d'interrogations sur votre jeu de données. On estimera que celles-ci sont effectuées très fréquemment.
  Positionnez-vous comme un utilisateur standard de l'application. 1 jointure (ou plus) et 1 filtre (ou plus) **minimum** pour chaque requête.
- *Data Analyst View* : Définir, en langage courant, 4 types d'interrogations lourdes sur votre jeu de données (agrégation, jointures, transformations, calcul complexe).
  Positionnez-vous comme un analyste ou un décisionnaire de l'application.

La complexité de votre cas d'usage aura un fort impact sur la complexité de votre projet, et de fait sur la note que vous obtiendrez. La variété des jointures est également un critère d'évaluation (éviter de faire toujours la même jointure).

Ces requêtes seront par la suite traduites en MQL (MongoDB Query Language) sur le modèle de données que vous aurez proposé. À cette étape du projet, les requêtes sont simplement un cas d'usage ne dépendant pas de la structure de données que vous produirez.

1.3 Relier les requêtes au schéma de base de données

Pour chaque requête du cas d'usage :

— Identifier les tables et attributs ciblés ;

— Lister les jointures à effectuer ;

— Lister les filtres sur attributs ;

— Lister les projections ;

— Lister les agrégats, ainsi que les informations nécessaires pour leur application.

Un schéma récapitutalif appliqué sur le schéma de base de données serait appréciable.

1.4 Statistiques

Sur chaque table, donner l'estimation de :

— Nombre de documents ;

— Cardinalité/Distribution des attributs associé à un filtre (requête) ;

— Cardinalité des jointures ;

1.5 Rendu

Pour cette partie, une description du modèle de données d'origine et du cas d'usage pour présenter la problématique doit être présenté pour comprendre ce qui est attendu.

⚠**Le rapport doit être rendu pour la 2° séance de cours.**

Chapitre 2

Dénormalisation

2.1 Dénormalisation du schéma

Sur ce jeu de données, vous devrez faire des choix de dénormalisation pour intégrer les données dans une base de données de type MongoDB. Pour cela, reposez-vous sur les interrogations produites dans la section précédente pour orienter vos choix.

Les points clés de la dénormalisation :

— Reposez vos choix sur les cardinalités entre les tables de votre schéma ;

— Tenez compte les données fréquemment accédées par vos requêtes utilisateurs et le coût élevé de vos requêtes Data Analystes ;

— Ne dénormalisez que ce qui est nécessaire ;

— Tenez compte d'une éventuelle évolution de la volumétrie de votre jeu de données (elle n'est pas statique).

Il est demandé de produire **2 schémas de base de données différents** (au moins) pour se permettre de faire un choix d'implémentation efficace. Les deux schémas de base de données doivent avoir une structure TRES différentes afin de coller aux différents types de requêtes du cas d'usage. Chaque schéma de bases de données doit lui-même détailler les schémas de chaque collection produite.

Les schémas de collection dénormalisées seront à présenter au format **JSON Schema**. Des jeux de couleurs faciliteront la visualisation des dénormalisations appliquées.

2.2 Types de dénormalisations

Les types de dénormalisation possibles :

— `Fusion` : fusionner deux tables (imbrication, liste)

— `Eclatement` : une table est séparée en deux spécialisations

— `Surcharge` : Une information (attribut) est dupliquée dans une autre table pour éviter un accès inutile

— `Matérialisation` : Le résultat d'un calcul (agrégation) est matérialisé dans un attribut

Vous pouvez user de n'importent quelles étapes de dénormalisation du moment qu'elles soient justifiée.

2.3 Requêtes et statistiques

À partir du rapport précédent, modifier la liste des caractéristiques des requêtes (section 1.3) en fonction de vos dénormalisations.

De même pour les statistiques, mettre à jour le nombre de documents par collection et filtres le cas échéant. De plus, donner le nombre de documents imbriqués dans des listes, le cas échéant.

2.4 Rendu

Les étapes de dénormalisation du schéma et l'argumentaire est nécessaire pour la compréhension des choix effectués.

Une présentation des points clés de la dénormalisation et le schéma obtenu en sortie est attendu. Un exemple de document JSON produit en sortie serait appréciable.

⚠**Le rapport doit être rendu pour la 4° séance de cours.**

Chapitre 3
Optimisation de l'infrastructure de données

3.1 Requêtes MQL

Reprenez les cas d'usages définies en section 1.2 et traduisez chacun sous forme de requêtes MQL (MongoDB Query Language). Vous vous aiderez de l'exemple de document JSON produit précédemment pour faciliter l'écriture.

3.2 Choix de clé de sharding

Afin d'optimiser l'infrastructure de données envisagée, il est nécessaire de définir différentes solutions de clé de partitionnement (ou sharding) adaptées aux requêtes (clés de filtres ou clés d'agrégations).

Pour cela, reposez vos choix sur les requêtes produits dans la section 3.1.

Vous devrez proposer deux choix de sharding pour pouvoir comparer les coûts.

Les clés filtrées mais non utilisées pour le partitionnement seront associées à un index secondaire.

3.3 Calcul de coût - Modèle de coût réseau

Afin de choisir la meilleure combinaison de partitionnement et de dénormalisation, produisez un tableau donnant le calcul entre communications effectuées sur le réseau. Nous estimerons que les données sont distribuées sur 100 *shard*.

Exemple :

Requête	Schéma de base de données 1		Schéma de base de données 2	
	Coût Sharding 1	Coût Sharding 2	Coût Sharding 1	Coût Sharding 2
R_{u1}				
R_{u2}				
R_{u3}				
R_{u4}				
R_{da1}				
R_{da2}				
R_{da3}				
R_{da4}				
Total pondéré				

Pour la pondération des requêtes, nous estimerons que :

- R_{u1} : 10 000x par jour
- R_{u2} : 1 000x par jour
- R_{u3} : 500x par jour
- R_{u4} : 100x par jour
- R_{da1} : 50x par jour
- R_{da2} : 25x par jour
- R_{da3} : 10x par jour
- R_{da4} : 2x par jour

3.4 Rendu

Les requêtes appliquées sur le modèle de données avec rappel du cas d'usage.

Le calcul sous forme de tableau avec le détail des communications réseaux effectuées par chaque requête.

⚠**Le rapport doit être rendu pour la 6° séance de cours, soit la dernière séance avant la session de soutenance des projets.**

Chapitre 4

Présentation de votre projet

Lors de la 7° séance (semaine des examens), vous devrez présenter votre projet. Une présentation de X min (X entre 15 et 20 minutes - dépendant du nombre de groupes) est prévue.

Vous devrez présenter le contenu des 3 rapports, mettant en valeur le dataset, le cas d'usage, la dénormalisation et le modèle de coût. Vous pouvez présenter un sous-ensemble significatif.