

Construction de cube OLAP à partir d'un entrepôt de données orienté colonnes

Khaled Dehdouh, Fadila Bentayeb, Nadia Kabachi, Omar Boussaid

Université de Lyon, ERIC (Lyon 2)
5 avenue Pierre Mendès-France, 69676 Bron Cedex, France
{Khaled.Dehdouh,Fadila.Bentayeb,Nadia.Kabachi,Omar.Boussaid}@univ-lyon2.fr

Résumé. L'optimisation de la construction de cubes OLAP¹ a été jusqu'à présent axée sur le développement d'algorithmes de calcul performants. Ces derniers opèrent sur des données extraites de l'entrepôt de données qui est généralement implémenté selon le modèle relationnel qui adopte l'architecture orientée lignes. Or, pour les requêtes décisionnelles, l'architecture orientée colonnes offre de meilleures performances. Cependant, les SGBDR² selon cette architecture ne disposent pas d'opérateurs appropriés pour le calcul de cube OLAP. Nous proposons dans cet article une nouvelle méthode de calcul de cube OLAP. Les résultats obtenus à partir des expérimentations que nous avons menées démontrent que notre approche optimise considérablement le temps de construction de cube OLAP et réduit le temps de réponse relatif à l'exploitation du cube comparé à l'approche orientée lignes.

1 Introduction

Un entrepôt de données est une base de données dédiée à l'analyse en ligne pour l'aide à la prise de décision. Grâce aux opérateurs OLAP, l'utilisateur peut extraire des cubes de données correspondants à des contextes d'analyse (Inmon, 1992). Dans un SGBDR, la construction d'un cube OLAP nécessite le calcul d'agrégats à partir des n-uplets stockés en lignes. Ce type de stockage est pénalisé par un coût de jointure important, ce qui implique un temps élevé pour l'extraction et le traitement des données. Par ailleurs, l'architecture orientée colonnes offre un mode de stockage et une technique de traitement plus adéquats au processus analytique en réduisant considérablement les accès au disque (Matei, 2010). Cependant, les SGBD orientés colonnes ne disposent pas d'opérateurs de construction de cube OLAP. L'objectif de ce travail est de proposer une nouvelle approche de construction de cube OLAP. Cette approche utilise un système de stockage et de traitement orienté colonnes pour réduire le temps d'extraction de données et une nouvelle méthode de calcul de cube OLAP. Nous avons implémenté cette approche sous l'SGBDR orienté colonnes *MonetDB*³.

1. On-Line Analytical Processing

2. Système de Gestion de Base de Données Relationnel

3. <http://www.monetdb.org/>

2 Construction de cube OLAP

L'idée sous-jacente est d'extraire les données constituant le cube une seule fois de l'entrepôt de données stocké en colonnes, et d'appliquer ensuite un ensemble de traitements pour calculer tous les agrégats possibles, à différents niveaux de granularité. Une fois le cube est calculé, il est matérialisé suivant l'architecture orientée colonnes pour permettre à l'analyse OLAP de bénéficier des avantages de cette architecture quant à la manipulation du cube (Slice, Dice, Drill-Down, Drill-Up, ..).

L'approche que nous proposons s'appuie sur une architecture orientée colonnes pour l'entrepôt des données et un nouveau algorithme de calcul de cube OLAP qui parcourt le treillis de cuboïdes et exploite la relation originelle. A la différence des algorithmes de la deuxième approche de construction de cube OLAP dans les bases de données orientées lignes, où le treillis de cuboïdes est créé au fur et à mesure suivant un plan d'exécution, notre méthode crée le treillis de cuboïdes global par un pré-traitement et le croise avec les données extraites de l'entrepôt (relation originelle). Cela permet de diminuer le nombre d'opération de calcul. Plus précisément, cette approche est exécutée en trois phases :

Phase 1 : Extraction des données

Cette phase consiste à extraire les données à partir de l'entrepôt de données. Le résultat de cette phase est une relation R composée de valeurs de dimensions et de la mesure qui satisfont les prédicats de la requête. Avec un entrepôt de données implémenté suivant l'approche orientée colonnes, cette phase est plus performante qu'avec un entrepôt de données implémenté suivant l'approche orientée lignes. En effet, le mécanisme de stockage de l'architecture orientée colonnes permet un accès rapide aux données des différentes colonnes et diminue considérablement le coût de jointure comparé à celui de l'architecture orientée lignes.

Phase 2 : Construction du treillis de cuboïdes

Cette phase consiste à construire le treillis de cuboïdes et définir les combinaisons possibles du cube à calculer.

Phase 3 : Calcul et matérialisation du cube

Cette phase consiste à parcourir le treillis de cuboïdes pour calculer le cube en fonction de la relation obtenue de la première phase, ainsi, le cube est matérialisé colonne par colonne

3 Expérimentations

Nous avons implémenté le modèle benchmark en étoile (SSBM)⁴ sous le SGBDR orientés colonnes *MonetDB*, nous avons utilisé un échantillon de données d'une taille de 1Go. L'environnement de tests mis en place consiste en une machine intel-Core TMi5-3550 avec une mémoire RAM de 16Go. Cette machine fonctionne avec le système d'exploitation Microsoft Windows 7 de 64bits.

4. <http://www.cs.umb.edu/~poneil/StarSchemaB.PDF>

Calcul du cube OLAP : Cette expérimentation consiste à comparer le temps de construction du cube OLAP en utilisant les deux fonctions ; Group by *CUBE* et *ROLLUP*, selon notre approche avec celui de l’approche classique (orientée lignes). Nous exécutons quatre requêtes décisionnelles de construction de cube OLAP avec un nombre de dimensions qui augmente progressivement. Les résultats obtenus sont présentés dans la figure 1.

Manipulation du cube OLAP : Cette expérimentation est consacrée à l’évaluation du temps relatif à la manipulation du cube OLAP matérialisé selon notre approche avec celui de l’approche classique. Pour cela, nous avons construit, selon les deux approches, un cube OLAP à trois dimensions et qui répond à la requête décisionnelle suivante : *Quelle est la somme des revenus des ventes par année, par marque de produit et par région des clients*. Nous avons ensuite exécuté des opérations de forage (*slice et dice*) sur le cube. Les résultats que nous avons obtenus sont présentés dans la figure 2.

4 Résultats des expérimentations

La figure 1 représente les temps d’executions des requêtes relatif à la construction de cube OLAP (*CUBE et ROLLUP*) à 2, 3, 4 et 5 dimensions selon l’approche classique comparée à celle que nous avons proposée.

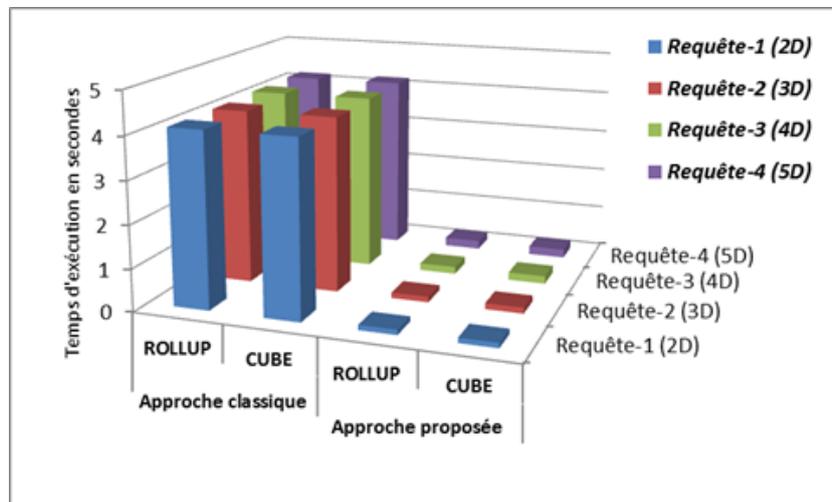


FIG. 1 – Résultat de la construction de cube OLAP (*CUBE et ROLLUP*) selon l’approche classique et notre approche

S’agissant de la manipulation du cube OLAP, la figure 2 représente les temps d’executions de la requête relatif à l’exploration du cube OLAP via les opérateurs de forge (*Slice et Dice*).

Construction de cube OLAP à partir d'un ED orienté colonnes

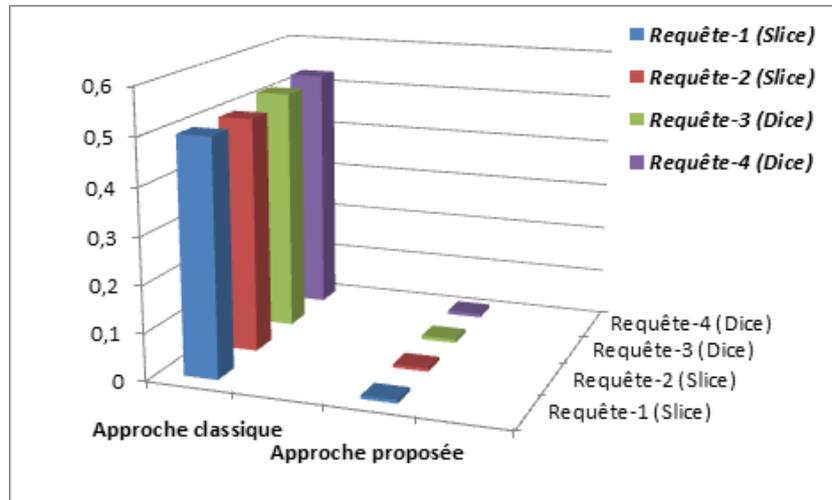


FIG. 2 – Résultat des opérations de forage (Slice et Dice) sur le cube OLAP matérialisé selon l'approche classique et notre approche

5 Conclusion

Ce travail propose une nouvelle approche de construction de cube OLAP basée à la fois sur une architecture relationnelle orientée colonnes et sur une nouvelle méthode de calcul des agrégats. L'intérêt de ce travail est de créer et d'intégrer un opérateur de construction de cube OLAP au sein des SGBDR orientés colonnes. L'implémentation de notre méthode et les expérimentations que nous avons menées ont montré clairement la performance de notre opérateur par rapport à celui de l'approche relationnelle orientée lignes.

Références

- Inmon, W. (1992). *Building the Data Warehouse*. Information Sciences, Inc Wellesley, MA, USA.
- Matei, G. (2010). Column-oriented databases, an alternative for analytical environment. *Database Systems Journal*, 3–16.

Summary

This paper presents a new method of calculating the OLAP cube. The results obtained from experiments carried out has shown that our approach significantly optimizes the OLAP cube building time and reduced the response time of cube manipulation, compared to the traditional approach.