

# Recommandations Personnalisées de Requêtes MDX

Elsa Negre

Laboratoire d'Informatique, Université François Rabelais Tours, France  
elsa.negre@univ-tours.fr

**Résumé.** Une session d'analyse OLAP peut être définie comme une session interactive durant laquelle un utilisateur lance des requêtes pour naviguer dans un cube. Très souvent, choisir quelle partie du cube va être naviguée par la suite, et, de ce fait, concevoir la prochaine requête, est une tâche difficile. Dans cet article, nous proposons d'utiliser ce que tous les utilisateurs du système OLAP ont fait pendant leurs précédentes explorations du cube afin de recommander des requêtes MDX à l'utilisateur.

## 1 Introduction

Les utilisateurs de systèmes OLAP naviguent interactivement dans un cube en lançant une séquence de requêtes sur un entrepôt de données, ce que nous appelons une session d'analyse (ou session) dans la suite de l'article. Ce processus est souvent pénible puisque l'utilisateur peut ne pas avoir d'idée sur ce que pourrait être la prochaine requête [Sarawagi (2000)].

Le problème abordé dans Giacometti et al. (2008) et revu dans cet article est donc le suivant : Comment aider l'utilisateur à concevoir sa prochaine requête ?

Comme réponse, nous proposons d'exploiter ce que les utilisateurs (l'utilisateur courant et les autres utilisateurs du système OLAP) ont fait lors de leurs précédentes navigations dans le cube, et d'utiliser cette information comme base pour recommander ce que pourrait être la prochaine requête. À cet effet, nous présentons une instanciation particulière avec un lien vers la personnalisation d'un cadre générique existant. Celui-ci permet de recommander des requêtes MDX, en utilisant le log du serveur, c'est-à-dire, l'ensemble des sessions précédentes sur le cube, et la séquence de requêtes de la session courante.

Le cadre est fondé sur le processus suivant :

1. Partitionner le log pour regrouper des requêtes qui sont similaires, afin de faire face au problème de faible densité du log.
2. Générer des recommandations candidates en commençant par trouver quelles sessions du log coïncident avec la session courante et ensuite, prédire ce que peut être la prochaine requête.
3. Ordonner les requêtes candidates en présentant à l'utilisateur la requête la plus pertinente en premier.