Fonctions d'oubli et Conservation de détail dans les entrepôts de données

Aliou Boly*, **, Georges Hébrail*, Marie-Luce Picard**

*Ecole Nationale Supérieure des Télécommunications de Paris
46, Rue Barrault 75634 Paris cedex 13 France
boly@enst.fr, hebrail@enst.fr
**Electricité de France Recherche et Développement
marie-luce.picard@edf.fr

Face au problème de saturation des volumes d'entrepôts de données, nous proposons ici une solution pour éviter leur engorgement par la définition de spécifications de fonctions d'oubli à travers un langage. L'objectif est de déterminer les données qui doivent être présentes dans l'entrepôt de données à chaque instant. Les spécifications de ces fonctions se traduisent par un ensemble de vues définies par l'utilisateur ; elles permettent de spécifier la conservation de n-uplets spécifiques, de résumés par agrégation. La suppression de données inutiles est réalisée de façon mécanique à chaque pas de mise à jour.

Dans ce papier, nous étudions l'oubli des données pour la conservation de n-uplets spécifiques jugés utiles dans le cadre de l'utilisation de données, et pour définir les spécifications de fonctions d'oubli, nous considérons l'utilité des données qui est définie à travers la clause WHERE d'une expression de vue.

Dans un entrepôt de données, certaines données ont une utilité significative et il est intéressant de conserver leur détail dans les stratégies d'oubli des données. Par exemple, dans un entrepôt de données de facturation de clients, il est intéressant de garder le détail des clients qui ont eu les montants de factures les plus élevés pendant les cinq dernières années.

Dans notre démarche, nous prenons en compte les contraintes d'intégrité référentielles dans le cadre du modèle relationnel pour déterminer, à partir des spécifications de fonctions d'oubli définies par l'utilisateur, les n-uplets à conserver ou à supprimer de la base de données. Un algorithme est défini : il prend en entrée l'ensemble des tables de la base de données, les vues correspondant aux spécifications de conservation de détail de n-uplets définies par l'utilisateur, pour produire en sortie un ensemble de n-uplets à supprimer tout en maintenant cohérente la base de données. Une expérimentation sous Oracle est faite.

Références

Boehm H.J., Space efficient conservative garbage collection. In ACM SIGPLAN Conference on Programming Languages Design and Implementation, 1993.

Boly A., Hébrail G., Picard M.L, EGC « Fonctions d'oubli dans les entrepôts de données », 2004, Clermont Ferrand, Janvier 2004.

Chaudhuri S., Das G., Narasayya V., A robust, optimisation-based approach for approximate answering of aggregate queries, Proceedings of SIGMOD Conference 2001.

Dumas M., Fauvet C., Scholl P., Modèles et langages pour données temporelles, Chapitre du livre « Bases de Données et Internet » A. Doucet et G. Jomier éditeurs, Hermes 2001.

Ullman J.D., Principles of Databases and Knowledge Base Systems, volume 1 and 2. Computer Science Press, 1989.