Chapter 6

RCUBE: Parallel Multi-Dimensional ROLAP Indexing

Frank Dehne
Carleton University, Canada

Todd Eavis
Concordia University, Canada

Andrew Rau-Chaplin
Dalhousie University, Canada

ABSTRACT

This chapter addresses the query performance issue for Relational OLAP (ROLAP) data cubes. The authors present RCUBE, a distributed multi-dimensional ROLAP indexing scheme which is practical to implement, requires only a small communication volume, and is fully adapted to distributed disks. Their solution is efficient for spatial searches in high dimensions and scalable in terms of data sizes, dimensions, and number of processors. Their method is also incrementally maintainable. Using “surrogate” group-bys, it allows for the efficient processing of arbitrary OLAP queries on partial cubes, where not all of the group-bys have been materialized. Experiments with RCUBE show that the ROLAP advantage of better scalability, in comparison to MOLAP, can be maintained while providing, at the same time, a fast and flexible index for OLAP queries.

INTRODUCTION

Online Analytical Processing (OLAP) has become a fundamental component of contemporary decision support systems. Gray et al. (1995) introduced the data cube, a relational operator used to compute summary views of data that can, in turn, significantly enhance the response time of core OLAP operations such as roll-up, drill down, and slice and dice. Typically constructed on top of relational data warehouses, these summary views (called group-bys) are formed by aggregating values across attribute combinations. For a d-dimensional input set R, there are \(2^d\) possible group-bys. Figure 1 illustrates a data cube as well as a lattice which is often used to represent the inherent relationships between group-bys (Harinarayan, 1996).
There are two standard data cube representations: MOLAP (multi-dimensional array) and ROLAP (set of relational tables). The array-based model, MOLAP (Multi-dimensional OLAP), has the advantage that native arrays provide an immediate form of indexing for cube queries. Interesting MOLAP based systems have been described and implemented in both the sequential and parallel settings; e.g. (Goil, 1997). However there is some evidence, that MOLAP based systems may encounter significant scalability problems (Pendse, 2002). For example, high-dimension data cubes represent extremely sparse spaces that are not easily adapted to the MOLAP paradigm. Hybrid indexing schemes are often used, significantly diminishing the power of the model. Moreover, since MOLAP needs to be integrated with standard relational databases, middleware of some form must be employed to handle the conversion between relational and array-based data representations. The relational model, ROLAP (Relational OLAP), does not suffer from such restrictions. Its summary records are stored directly in standard relational tables without any need for data conversion. Its table based data representation does not pose scalability problems. Yet, many current commercial systems use the MOLAP approach (Pendse, 2002). The main reason, as outlined in (Pendse, 2002), is the indexing problem for the fast execution of OLAP queries. The problem for ROLAP is that it does not provide an immediate and fast index for OLAP queries. Many vendors have chosen to sacrifice scalability for performance. However, this path is becoming increasingly unsustainable. As outlined in the 2005 Winter Report (Winter Corp., 2005), the size of data warehouses grew exponentially during recent years. More precisely, between 2001 and 2005 the average size of data warehouses grew by 243%, and the size of the largest data warehouses grew by an astounding 578% (Winter Corp., 2005). Hence, there is an urgent need for scalable (i.e. ROLAP) and high performance data cube indexing methods.

This paper addresses the query performance issue for ROLAP and proposes a novel, distributed

---

**Figure 1.** (a) A three dimensional data cube for automobile sales data. (b) The lattice corresponding to a four dimensional data cube with dimensions A, B, C and D.
Related Content

Efficient Top-k Keyword Search Over Multidimensional Databases
www.igi-global.com/article/efficient-top-keyword-search-over/78373?camid=4v1a

What-if Simulation Modeling in Business Intelligence
www.igi-global.com/article/simulation-modeling-business-intelligence/37403?camid=4v1a

Retailer Case Study
www.igi-global.com/chapter/retailer-case-study/27343?camid=4v1a

A Case Study in Data Mining for Automated Building of Teams
www.igi-global.com/chapter/case-study-data-mining-automated/29154?camid=4v1a