IDEA GROUP PUBLISHING
701 E. Chocolate Avenue, Suite 200, Hershey PA 17033-1240, USA
ITJ2607

Mobile Agent Based Self-Adaptive Join for Wide-Area Distributed Query Processing

J. -P. Arcangeli, IRIT, Université Paul Sabatier, France
A. Hameurlain, IRIT, Université Paul Sabatier, France
F. Migeon, IRIT, Université Paul Sabatier, France
F. Morvan, IRIT, Université Paul Sabatier, France

ABSTRACT

In this article, optimization of decision support queries is considered in the context of wide-area distributed databases. An original approach based on the "mobile agent" paradigm is proposed and evaluated. Agents' autonomy and reactivity allow operators of the execution plan to adapt dynamically to estimation errors on relations and to evolutions in the state of the execution system, avoiding time overheads commonly associated with centralized monitoring. We present decentralized self-adaptive algorithms for dynamic optimization of join operators, and their implementations in Java using mobile agents. Then, we evaluate performance depending on error rate on statistical information on database, and on communication bandwidth and CPU frequency. The results show that the agent-based approach can lead to a significant reduction of response time and provide decision criteria for developing an effective migration policy.

Keywords: dynamic optimization; mobile agents; mobile join; performance evalution; wide-area distributed databases.

INTRODUCTION

Computer networks have been evolving and growing rapidly in the last years. As a consequence, new distributed applications have been emerging in different domains: e-business and e-commerce, Web mining, data warehousing, telecommunication services, mobile and wireless computing, etc. Presently, large-scale distributed networks and grids link a large amount of distributed data, and make them available and accessible from anywhere in the network. Data repositories of organizations and enterprises are becoming increasingly decentralized, and many applications are constituted from geographically distributed and concurrent components. However, even if data are stored in structured databases, a major problem resides in volumes and networking: moving large amounts of data over the network leads to high communication costs and consequently to performance limitation.

In this work, we investigate optimization of decision support queries when relations are physically distributed over a wide-area network. Optimization of decision support queries is a complex problem that has been widely studied (Kossmann, 2000; Ozsu & Valduriez, 1999). Many authors have shown that dynamic load balancing and correction of sub-optimal execution plans are necessary (Hameurlain & Morvan, 2002; Kabra & DeWitt, 1998). In order to reduce response time, the initial execution plan (built at compile time from a cost model capturing both the statistics on relations and the characteristics of the execution system) must be changed or adapted at runtime because of

- Estimation errors on relations: Statistics on base relations are stored in the system catalog (Bernstein et al., 1981; Mannino et al., 1988; Selinger et al., 1979) in order to estimate the response time of query execution plans. For example, the system catalog contains the size of relations and the average size of tuples, and for each attribute, the number of possible values, and the maximal and the minimal values. Statistics on base relations can be imprecise or obsolete, leading to loss of accuracy and to errors in estimations, for example concerning sizes of temporary relations. Ioannidis and Christodoulakis (1991) have shown the impact of estimation errors at the operator level: the absolute value of the error propagates and grows exponentially with the number of joins.
- Resource and data unavailability: Processor load, memory capacity, input/output or network bandwidth can differ at runtime from the expected values, thus preventing the execution plan from running as scheduled.

Dynamic optimization corrects at runtime the initial execution plan. In the context of parallel or distributed database systems, solutions mainly rely on a centralized optimizer (Avnur & Hellerstein, 2000; Bouganim et al., 2000; Deshpande & Hellerstein, 2002; Hellerstein & Franklin, 2000; Ives et al., 1999). The optimizer collects statistical information on relations and availability of resources and data. Then, depending on the information collected, it decides to modify the execution plan and monitors the relocation of operations.

In a wide-area network, it is difficult to get up-to-date statistical information because of physical distribution of numerous sources and of distances between sources and collectors of information. Hence, inaccuracy of estimations and unawareness of resource availability limit efficiency of execution plans. Thus, in such a context, dynamic adaptation is even more urgently called for. However, centralized control is limited by bottleneck and involves a lot of remote interactions when the system is distributed. Actually, centralized optimization cannot scale up with the size of the network because it involves prohibitive overheads. In the wide and decentralized environment considered here, it is unrealistic to base reactive software architecture on a centralized process, no matter where this one is located. We argue that control and optimization must be decentralized among the components of the execution plan in order to benefit from proximity between the decision process and significant data on relations and physical resources. Therefore, we propose to deal with optimization at the join operator level, and to turn operators into software entities able to react to estimation errors on relations and to evolutions in the state of the execution system. Consequently, we propose a decentralized join execution model based on mobile agents, which can decide on their location and move autonomously. In this model, agents are first placed according to the initial execution plan; at runtime, they

Big Data in Massive Parallel Processing: A Multi-Core Processors Perspective
Vijayalakshmi Saravanan, Anpalagan Alagan and Isaac Woungang (2018).
Handbook of Research on Big Data Storage and Visualization Techniques (pp. 276-302).
www.igi-global.com/chapter/big-data-in-massive-parallel-processing/198767?camid=4v1a

On the Adaptation of an Agile Information Systems Development Method
Mehmet N. Aydin, Frank Harmsen, Kees van Slooten and Robert A. Stagwee (2005).
Journal of Database Management (pp. 25-40).
www.igi-global.com/article/adaptation-agile-information-systems-development/3340?camid=4v1a