International Journal of Current Advanced Research
ISSN: O: 2319-6475, ISSN: P: 2319-6505, Impact Factor: 6.614
Available Online at www.journalijcar.org
Volume 7; Issue 7(J); July 2018; Page No. 14566-14569
DOI: http://dx.doi.org/10.24327/ijcar.2018.14569.2645

Research Article

A STUDY ON QUERY BASED OPTIMIZATION TECHNIQUES & ALGORITHMS OF DISTRIBUTED DATABASE MANAGEMENT SYSTEM (DDBMS)

Anurag Gupta*

Asian Business School, Noida

ARTICLE INFO

Article History:

Received 7th April, 2018
Received in revised form 16th May, 2018 Accepted 3rd June, 2018
Published online 28th July, 2018

ABSTRACT

Now-a-days data is distributed across the networks to make total world as a global Pool. Distributed database management systems (DDBMS) are amongst the most important and successful software developments in this decade. They are enabling computing power and data to be placed within the user environment close to the point of user activities. The performance efficiency of DDBMS is deeply related to the query processing and optimization strategies involving data transmission over different nodes through the network. Most real-world data is not well structured. Today's databases typically contain much non-structured data such as text, images, video, and audio, often distributed across computer networks. This situation demands new query processing, optimization techniques in distributed database environment. In this paper different techniques are elaborated which will provide efficient performance in optimization of query processing strategies in a distributed databases environment.

INTRODUCTION

Distributed database is a database that is under the control of a central database management system (DBMS) in which storage devices are not all attached to a common CPU. It may be stored in multiple computers located in the same physical location, or may be dispersed over a network of interconnected computers. Collections of data (e.g. in a database) can be distributed across multiple physical locations.

Query processing

Query processing is defined as the activities involved in parsing, validating, optimizing and executing a query. The main aim of query processing is Transform query written in high-level language (e.g. SQL), into correct and efficient execution strategy expressed in low-level language (implementing Relational Algebra) and to find information in one or more databases and deliver it to the user quickly and efficiently. High level user query -> Query Processor ->low-level data manipulation commands

Query optimization

Query optimization is defined as the activity of choosing an efficient execution strategy for processing a query. Query optimization is a part of query processing.

Objective

The main Objective of this Research study "A Study on Query based Optimization Techniques & Algorithms of Distributed Database Management System(DDBMS)" is to choose a transformation that minimizes resource usage, Reduce total execution time of query and also reduce response time of query and estimate the cost of different equivalent query expressions and chose the execution plan with the lowest cost and how it is beneficial for the service industry.

Review of literature

Fan Yuanyuan, Mi Xifeng (2010),

This research paper is based on query optimization technology, based on a number of optimization algorithms commonly used in distributed query, a new algorithm is designed, and experiments show that this algorithm can significantly reduce the amount of intermediate result data, effectively reduce the network communication cost, to improve the optimization efficiency.

Abdelkader Hameurlain and Franck Morvan (2009)

This research paper is based on the evolution of query optimization methods from uniprocessor relational database systems to data Grid systems through parallel, distributed and data integration systems. We point out a set of parameters to characterize and compare query optimization methods, mainly: (i) size of the search space, (ii) type of method (static or dynamic), (iii) modification types of execution plans (re-optimization or re-scheduling), (iv) level of modification (intra-operator and/or inter-operator), (v) type of event (estimation errors, delay, user preferences), and (vi) nature of decision making (centralized or decentralized control).

Corresponding author:* **Anurag Gupta
Asian Business School, Noida

The major contributions of this paper are

1. understanding the mechanisms of query optimization methods with respect to the considered environments and their constraints (e.g. parallelism, distribution, heterogeneity, large scale, dynamicity of nodes)
2. pointing out their main characteristics which allow comparing them, and
3. the reasons for which proposed methods become very sophisticated.

Deepak Sukheja, Umesh Kumar Singh (2011)

Query optimization in distributed databases explicitly needed in many aspects of the optimization process, often making it imperative for the optimizer to consult underlying data sources while doing cost based optimization. This is not only increases the cost of optimization, but also changes the trade-offs involved in the optimization process significantly. The leading cost in this optimization process is the "cost of costing" that traditionally has been considered insignificant. The optimizer can only afford a few rounds of messages to the under- lying data sources and hence the optimization techniques in this environment must be geared toward gathering all the required cost information with minimal communication.

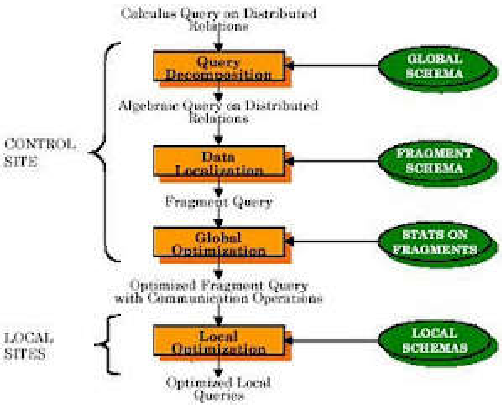
This paper, explores the design and search space for a query optimizer in distributed environment and demonstrate the need for this optimization approach in various aspects of the optimization process. This work present minimum-communication cost query cost variants of various query optimization techniques, and discuss trade-offs in their performance in the present development. Authors have implemented a novel optimization approach in the distributed database environment, somewhat unexpectedly, indicate that a simple two-phase optimization scheme performs fairly well as long as the physical database design is known to the optimizer, though more determined algorithms are required.

Yannis E. Ioannidis (2008)

This paper presented an abstraction of the architecture of a query optimizer and focused on the techniques currently used by most commercial systems for its various modules. In addition, it has provided a glimpse of advanced issues in query optimization, whose solutions have not yet found their way into practical systems.

Distributed Query Processing Methodology

Distributed query processing contains four stages which are query decomposition, data localization, global optimization and local optimization.



Query decomposition: in this stage we are giving Calculus Query as an input and we are getting output as Algebraic Query. This stage is again divided in four stages they are Normalization, Analysis, Simplification and Restructuring

Data localization: in this stage Algebraic query on distributed relations is input and fragment query is output. In this stage fragment involvement is determined.

Global optimization: in this stage Fragment Query is input and optimized fragment query is output. Finding best global schedule is done in this stage.

Local optimization: Best global execution schedule is input and localized optimization queries are output in this stage. it contain two sub stages they are Select the best access path ,Use the centralized optimization techniques.

Distributed Query Optimization

Distributed query optimization is defined as finding efficient execution strategy path in distributed networks.
Query optimization is difficult in distributed environment.
There are three components of distributed query optimization they are Access Method, Join Criteria, and Transmission Costs.

Access Method: The methods which are used to access data from distributed environment like hashing, indexing etc.

Join Criteria: In distributed database data is presented in different sites. Join criteria is used to join the different sites to get optimized result.

Transmission Costs: If data from multiple sites must be joined to satisfy a single query, then the cost of transmitting the results from intermediate steps needs to be factored into the equation. At times, it may be more cost effective simply to ship entire tables across the network to enable processing to occur at a single site, thereby reducing overall transmission costs. This component of query optimization is an issue only in a distributed environment.

There are many distributed query optimization issues some of them are types of optimizers, optimization granularity, network topologies and optimization timing.

Cost – based query optimization

- Objective of Cost-based query optimization is estimate the cost of different equivalent query expressions and chose the execution plan with the lowest cost.
- Cost based query optimization mainly depends on two factors they are solution space and cost function.
- Solution space: this is depends on the set of equivalent algebraic expressions.
- Cost function: cost function is equivalent to summation of i/o cost, CPU cost and communication cost it also depends on different distributed environments.
- By considering these factors cost based query optimization is processed in distributed environment.

Heuristic – based query optimization

Heuristic based query optimization process involve following steps

- Perform Selection operations as early as possible.
- Combine Cartesian product with subsequent selection whose predicate represents join condition into a Join operation.

- Use associatively of binary operations to rearrange leaf nodes so leaf nodes with most restrictive Selection operations executed first.
- Perform Projections operations as early as possible.
- Eliminate duplicate computations.

It is mainly used to minimize cost of selecting sites for multi join operations.

Rank-Aware Query Optimization

Ranking is an important property that needs to be fully supported by current relational query engines. Recently, several rank-join query operators have been proposed based on rank aggregation algorithms. Rank-join operators progressively rank the join results while performing the join operation. The new operators have a direct impact on traditional query processing and optimization. We introduce a rank-aware query optimization framework that fully integrates rank-join operators into relational query engines. The framework is based on extending the System R dynamic programming algorithm in both enumeration and pruning. We define ranking as an interesting property that triggers the generation of rank-aware query plans. Unlike traditional join operators, optimizing for rank-join operators depends on estimating the input cardinality of these operators. We introduce a probabilistic model for estimating the input cardinality, and hence the cost of a rank-join operator. To our knowledge, this paper is the first effort in estimating the needed input size for optimal rank aggregation algorithms. Costing ranking plans, although challenging, is key to the full integration of rank-join operators in real-world query processing engines. We experimentally evaluate our framework by modifying the query optimizer of an open-source database management system. .

Query Optimization problems and some solutions in distributed database

Stochastic query optimization problem for multiple join

The model of three joins stored at two sites leads to a nonlinear programming problem, which has an analytical solution. The model with four sites leads to a special kind of nonlinear optimization problem (P).This problem is known as stochastic query optimization problem for multiple join. This problem can not be solved analytically. It is proved that problem (P) has at least one solution and two new methods are presented for solving the problem. An ad hoc constructive model and a new evolutionary technique is used for solving problem (P). Results obtained by the two considered optimization approaches are compared.

Problem of optimizing queries that involve set operations

The problem of optimizing queries that involves set operations (set queries) in a distributed relational database system. A particular emphasis is put on the optimization of such queries in horizontally partitioned database systems. A mathematical programming model of the set query problem is developed and its NP-completeness is proved. Solution procedures are proposed and computational results presented. One of the main results of the computational experiments is that, for many queries, the solution procedures are not sensitive to errors in estimating the size of results of set operations.

Stochastic optimization problem for multiple queries

Many algorithms have been devised for minimizing the costs associated with obtaining the answer to a single, isolated query in a distributed database system. However, if more than one query may be processed by the system at the same time and if the arrival times of the queries are unknown, the determination of optimal query-processing strategies becomes a stochastic optimization problem. In order to cope with such problems, a theoretical state-transition model is presented that treats the system as one operating under a stochastic load. Query-processing strategies may then be distributed over the processors of a network as probability distributions, in a manner which accommodates many queries over time. It is then shown that the model leads to the determination of optimal query-processing strategies as the solution of mathematical programming problems, and analytical results for several examples are presented. Furthermore, a divide-and-conquer approach is introduced for decomposing stochastic query optimization problems into distinct sub problems for processing queries sequentially and in parallel.

Sum product optimization problem

Most distributed query optimization problems can be transformed into an optimization problem comprising a set of binary decisions, termed Sum Product Optimization (SPO) problem. We first prove SPO is NP-hard in light of the NP-completeness of a well-known problem, Knapsack (KNAP). Then, using this result as a basis, we prove that five classes of distributed query optimization problems, which cover the majority of distributed query optimization problems previously studied in the literature, are NP-hard by polynomials reducing SPO to each of them. The detail for each problem transformation is derived. We not only prove the conjecture that many prior studies relied upon, but also provide a frame work for future related studies.

Advantages of distributed database

Reflects organizational structure-database fragments are located in the departments they relate to.
Local autonomy - a department can control the data about them (as they are the ones familiar with it.)
Improved availability - a fault in one database system will only affect one fragment, instead of the entire database.
Improved performance - data is located near the site of greatest demand, and the database systems themselves are parallelized, allowing load on the databases to be balanced among servers. (A high load on one module of the database won't affect other modules of the database in a distributed database.)
Economics - it costs less to create a network of smaller computers with the power of a single large computer.
Modularity - systems can be modified, added and removed from the distributed database without affecting other modules (systems).

Advantages of Distributed query optimization

Distributed Query optimization techniques provide exact results in distributed environment. These techniques provide efficient performance in different distributed networks. In internet these techniques helps to search exact information and extract the required one

Bibliography (References)

1. Elmsari and Navathe, “Fundamentals of Database Systems”, Pearson Education, 5th Ed. 2006.
2. Korth, Silberschatz, “Fundamentals of Database System Concepts”, TMH, 6th Ed., 2010.
3. Desai, B., “An Introduction to Database Concepts”, Galgotia.
4. Sham Tickoo and Sunil Raina, “Oracle 11g with PL/SQL Approach”, Pearson, 2010.
5. Ivan Bayross, SQL, PL/SQL- The Programming Language of Oracle, BPB Publication, New Delhi.
6. Date C. J., “An Introduction to Database Systems”, Narosa Publishing, 7th Ed., 2005.
7. S. K. Singh, “Database Systems: Concept, Design, and Applications”, Pearson’s Education, 1st Ed., 2008.
8. Kiffer, “Database Systems: An Application oriented Approach”, Pearson.
9. Ullman J. D., “Principals of database systems”, Galgotia.
10. Shio Kumar Singh, “Databases Systems Concepts, Design and Applications,” Pearson, 2006.

URL’s

11. http://dl.acm.org/citation.cfm?id=75474
12. citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.37...rep...
13. citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.56.356...
14. ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=390407

How to cite this article:

Anurag Gupta (2018) 'A Study on Query Based Optimization Techniques & Algorithms of Distributed Database Management System (Ddbms)', *International Journal of Current Advanced Research,* 07(7), pp. 14566-14569.
DOI: http://dx.doi.org/10.24327/ijcar.2018.14569.2645

****