Volume 6, Issue 3, March 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Query Optimisation in Distributed Databases

Karanvir Kaur, Ravi Kant Sahu
Computer Science Engineering & Lovely Professional University,
Punjab, India

Abstract— ***Data dispersed on different sites is processed using various strategies and various algorithms. To obtain the result or outcome for a particular query we have to make various types of joins and semi joins. We have various algorithms which efficiently obtain the solution search space with reduced cost. In order to obtain best solutions in a manner that can help to make decision very easily, in minimum amount of time with best solution and without bulky index files, we should analyse the algorithm performance on the base of desired factor efficiency. According to dispersed data fragments schema, different queries obtain same result with different cost of different factors like communication cost, access cost, operations cost etc.***

Keywords— ***Fragment Schema, Search Space, Search Strategy, Bidding, Query Execution Plan***

I. INTRODUCTION

Now a days, with the advancement or development of technology in database and computer network, distributed database is used widely by the enterprises; with the expand of application, queries of data are complex with the increase of time, the requirement for efficiency are high and its request is increasing high, a key or main issue of the distributed database system is processing query.

Distributed database system is physically dispersed but logically centralized system of database. It is a composition or combination of computer network and database system. Database is physically dispersed, referring to that the data which is distributed in different computers and composed of distributed database, and independent processing at every site is possible. Logically centralized, referring to that the site is a logical whole and managed by a distributed database management system, each node performs the overall application through the network communication subsystem.

A. Components of Distributed Systems

Distributed database system includes not only a distributed database management system and distributed database, but also contains more actual components. It can be run and stores, maintains the data by way of distributed database, as well as provides data and information to the applied network environment systems.

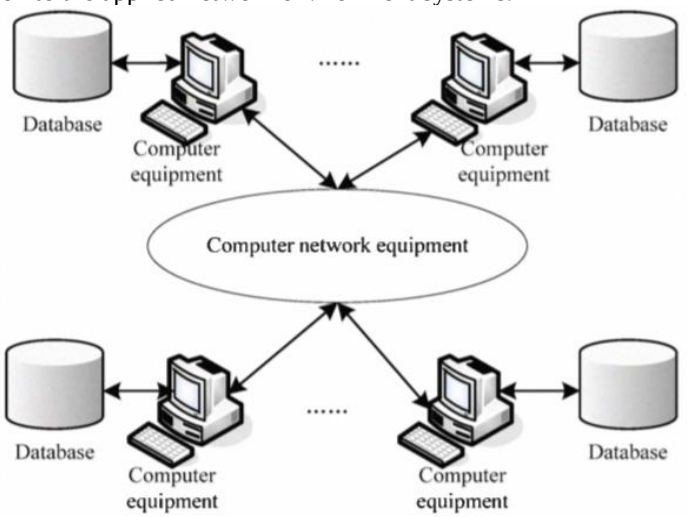

Figure 1 Component of Distributed Databases

Dispersed database framework comprises of the accompanying components:

1) *Multiple PCs hardware:* - PC associated through system.
2) *Computer system hardware:* - system correspondences program set.
3) *Distributed database administration framework*

4) Distributed database: - incorporates the worldwide database, neighbourhood database and self-governing site database.

5) Distributed database director:- It can be separated into two, first is worldwide database chief, and the second one for the nearby or independent site database administrators, which are by and large known as the neighbourhood database supervisor.

6) Distributed database framework programming: - Report, programming records set coordinated with the product, and also an assortment of framework directions and archives. The system architecture is shown as in figure 1.

Appropriated database administration framework ought to have the capacity to bolster the three essential elements of the disseminated database framework. The principal is the remote database operations of the application, the second is the full or halfway straightforwardness of it, and the third is the administration and control of the circulated databases. In this way, notwithstanding the elements of a brought together database administration framework, it should likewise give maps of every site database for concentrated administration and control.

Demand for data from a database is known as inquiry. Inquiry advancement is an element of numerous social database administration frameworks. The inquiry streamlining agent endeavours to decide the most effective approach to execute a given question by considering the conceivable question arranges.

B. Types of Distributed Systems

On the basis of software and platforms used at various sites to manage and process the dispersed data on various sites, classification is as follow

1) Homogenous Distributed Database: The dispersed sites uses same software that is same DBMS tool and implemented on same platform.

2) Heterogeneous Distributed Database: The dispersed sides have different DBMS product to manage the dispersed data.

C. Query Optimization

It has been one of the examination centres of database territory, albeit numerous analysts have done a considerable measure of work, yet the not similar with the effective utilization of social database innovation in information handling is multi-join inquiry enhancement has been an issue not all around determined in social database frameworks. Query processing is done by divided into mainly four phases which are decomposition, globalization, localization and execution. Each phase has its own module of function which influence the processing time of query for a particular time. query optimization has key role in the performance of distributed database systems. Dispersed data need to be held in the way that it will affect the performance in such a manner that will totally influence the time of its execution and the level of processing the data. Simply the flow chart can explain the process of query execution. Flow chart system helps to understand the key concept very easily and effectively.

Dispersed framework first checks whether there is such a neighbourhood database after the landing of the client questions, if there is, for a nearby execution; if there is no, then the worldwide inquiry preparing module is to choose an ideal hub to handle this question as indicated by the table data, that is, select a database hub which has the database and has the base inquiry expense of the controlled table, and to build up an association with the improvement hub to send the inquiry summon to it for usage, while giving back its IP to the customer. The customer will promptly re-build up the association with the new IP in the wake of getting the criticism data. At the point when the new server hub finishes the question, it will give back the outcomes to the customer. The structure in figure 2 demonstrates the different phases of the conveyed inquiry handling, that is, the question starts from the client data of SQL proclamation, then the parser makes an interpretation of and enhances to make it turn into an inquiry arrangement which can be performed, and afterward the line asset generator will execute this arrangement, at last acquires the question results.

D. Phases of Query Optimisation

To do a certain work it is good to divide them and solve it to do it appropriately and speedily. In distributed database environment we have to search data from different sites. So we have different phases: Initial processing phase, reduction phase, final processing phase.

1) *Initial processing phase*: It involves to specifying queries on global relations and to convert into fragments and made available to the respective sites for processing.
2) *Reduction phase*: It is consist of arranging joins and semi joins to reduce the amount of data to process.
3) *Final processing*: It involves generating final outcome at main site by assembling the data came from various sites.

E. Principles of Query Optimization

We can call query optimization as program translation problem so according to this we can define three major independent aspects:

1) *The QEP Generation*: It involves finding query which can obtain result optimally and speedily.
2) *The Search Strategy:* It involves selecting appropriate search strategy which can search data in the way that can reduce the amount of transmit data.
3) *The cost function*: It involves reducing the function of cost to get result which is cost effective.

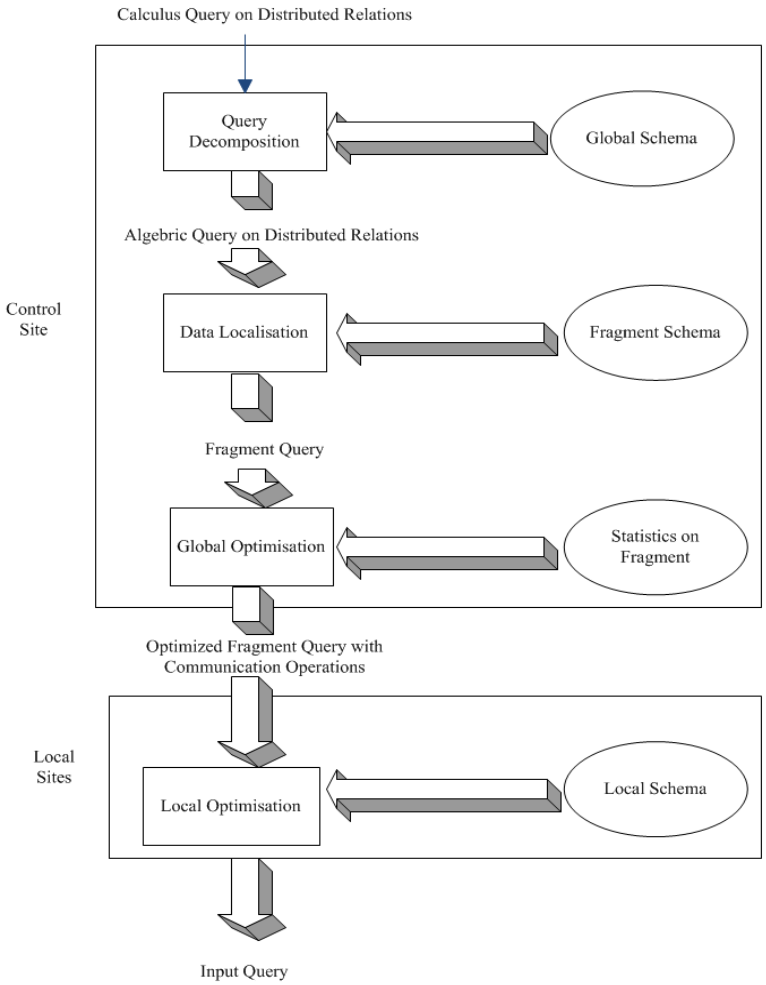

Figure 2 Phases of Query Optimisation

F. Optimization Model

Optimization model composed of three major components which are called as component of creation, searching strategy and costing.

Implementation of decision when to terminate and where to continue the search, we have further subcomponents which are tester, decider and updater. These all subcomponents will perform their individual functions in the model. Firstly initial state will be generated then cost evaluator will find its cost and then updater updates cost function value. Decider will decide about the efficient query plan and outcome will be that particular QEP. This process will be repeated till minimum cost function is searched for that query.

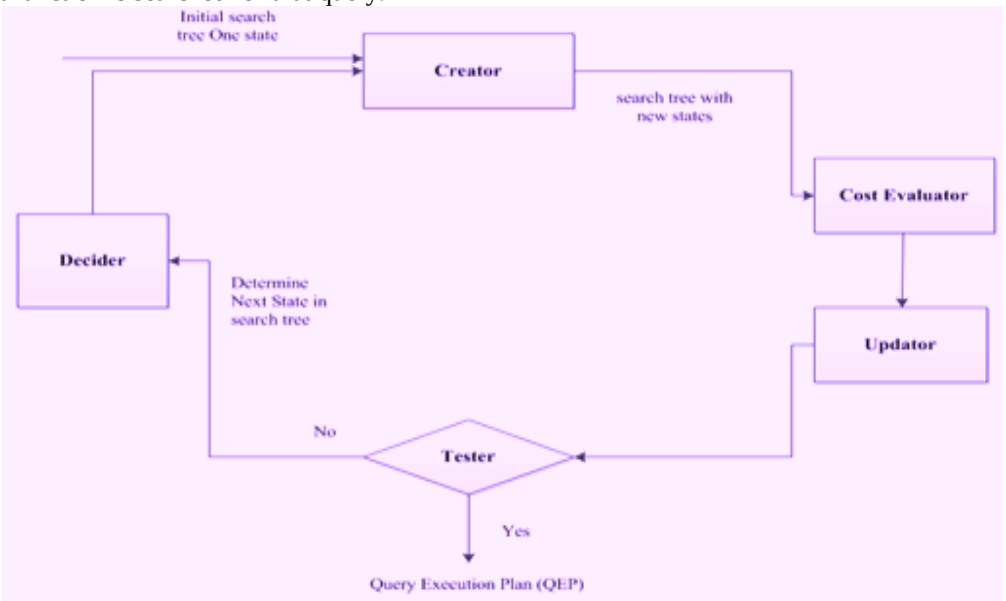

Figure 3 Model for query optimization

II. RELATED WORK

In today's era, Most of applications had dispersed data sites. Various business companies have their data dispersed over different places and they implement the system in such a way that they get outcomes with efficiency and can satisfy their customers from their services. a business with good capability and A grade service can make profit in today's time. Customers are now well aware of all the things and stuff happening in the world and technologies.

As advancement data is increasing very rapidly we need to store at the location where it can be processed and can be retrieving easily. Data can be centralised and distributed. So research in infrastructure and framework of query optimisation has vast scope. Late proposition, for example, Scope, raise the level of deliberation by giving an explanatory dialect that builds programming efficiency as well as amiable to modern enhancement. Like in customary database frameworks, such advancement depends on point by point information insights to pick the best execution arrangement in an expense based design. Nonetheless, as opposed to database frameworks, it is exceptionally difficult to acquire and keep up great quality insights in a much disseminated environment that contains many thou-sands of machines. In the first place, it is exceptionally testing to efficiently join an extensive number of separately gathered nearby complex factual data (e.g., histograms, particular qualities) in a measurably significant manner. Second, computing insights regularly requires look over the full dataset. Such operation can be overwhelmingly costly for terabytes of information.

We influence the way that a substantial extent of scripts in this environment are parametric and repeating over a period arrangement of information. The information datasets more often than not come in routinely, say, hourly or day by day. The same business rationale is connected to distinctive datasets in an hourly or day by day design. Despite the fact that the information datasets touch base in a period arrangement, they share a comparable information appropriation and attributes. We accomplish this objective by incrementing different employment organizes and piggybacking insights accumulation with the ordinary execution of work. Subsequent to gathering such measurements, we demonstrate to bolster them back to the inquiry analyzer so that future summons of the same (or comparable) employments exploit exact insights. It actualized this methodology in the Scope framework, which keeps running over a huge number of machines and procedures more than 30 thousand occupations day by day, 40% of which have a repeating example. The multifaceted nature of the enhancer increments as the quantity of relations and number of joins in an inquiry increases. Study is being did to locate a proper calculation to look for an ideal arrangement particularly when the extent of the database increments.

A. Search Space

It alludes to the era of sets of option and comparable QEPs of an information question by applying Transformational Rules such that they contrast in the execution request of the administrators. The QEPs are usually alluded to as Operator Trees or Join Trees whose administrators are different sorts of Joins or Cartesian Products. This can be spoken to as an inquiry diagram (clarified tree) signified as G = (N, A) where N is the arrangement of nodes (vertices) in the Query Graph and An is the arrangement of circular segments (edges). Every hub speaks to an arrangement of Base File (BF) in the join particular of the inquiry. Two sites and hubs are associated by a circular segment if the question joins the two comparing records. Every site in the question chart has a related site set. These leaf hubs speak to document emergences coming about because of nearby handling and it is a decreased record. The root site speaks to the last step where the inquiry result is produced. Every guardian site utilizes the outcome records of its youngsters as its inputs.

B. Search Strategy

There are fundamentally two classes of techniques that take care of the issue of Join Scheduling for Query Optimization.

1) *Deterministic Strategy* : Deterministic Strategy that returns by building arrangements, beginning from base relations, joining one or more relations at every progression till complete arrangements are acquired. To diminish the streamlining expense, the arrangements that don't prompt ideal arrangements are pruned. The Dynamic Programming fabricates every such arrangement utilizing Breadth-first pursuit while Greedy Algorithms utilizes profundity first hunt.
2) *Randomized Strategy:* Randomized Strategies that hunt the ideal arrangement around some specific focuses. These techniques don't promise ideal arrangement yet they evade high cost of enhancement as far as memory and time utilization. Iterative Improvement and Simulated Annealing are regular arrangement calculations under these systems.

C. Cost Model

The Objective of Query Optimization in Distributed Database Environment is to minimize the aggregate expense of PC assets. A streamlining agent expense model incorporates expense capacities to anticipate the expense of administrators and recipes to assess the sizes of the outcomes. The expense capacity can be communicated as for either aggregate time or reaction time. The aggregate time is comprehensive of Local Processing Cost (CPU Time + I/O Cost), Communication Cost (Fixed time to start a message + Time to transmit an information). Minimizing the aggregate time infers that the use of assets builds in this way expanding the framework throughput. The Response Time is assessed as the time slipped by in the middle of start and fulfilment of an inquiry including parallelism. In parallel exchanging, the reaction time is minimized by expanding the level of parallel execution.

D. Various Algorithms

Genetic Algorithms are a group of Computational models propelled by nature or Biological Evolutions. The idea of GA was proposed by John Holland where arbitrarily created answers for an issue are assessed as chromosomes and these

chromosomes are permitted to deliver new arrangement of people with better qualities by means of hybrid and changes administrators in view of wellness capacity. The calculation was likewise ready to diminish the expense of the conveyed inquiry tree. The flow chart explains the general idea of the Genetic or heuristic Algorithm. It is the most basic system of execution of the query by which can simply configure the data from dispersed sites. A SQL query to a modern relational DBMS does more than just selections and joins. In particular, SQL queries often nest several layers of SPJ blocks (Select-Project-Join), by means of group by, exists, and not exists operators. In some cases such nested SQL queries can be flattened into a select-project-join query, but not always. Query plans for nested SQL queries can also be chosen using the same dynamic programming algorithm as used for join ordering, but this can lead to an enormous escalation in query optimization time. So some database management systems use an alternative rule-based approach that uses a query graph model.

ANT Colony Optimization Algorithm (ACO) is a novel meta-heuristic calculation which is suitable for issues identified with Combinatorial Optimization. Like inquiry advancement in dispersed database as a result of its qualities like canny hunt systems, worldwide streamlining, strong, circulated processing and capacity to join with different heuristics.

Hybrid GA-ACO is a hybrid system of both algorithms. The ability of Hybrid GA-ACO to hunt broad abundance to replies down join questions in social Database can be reached out to improve the join inquiries in disseminated database where the most imperative test is to create the best QEP for ideal results. it was earlier proposed a Genetic Algorithm based arrangement technique to rapidly decide ideal QEP. This model incorporates Copy Identification (repetition of information), Beneficial Semi joins ID, Join Site Selection, Join Order Execution, and Local Processing Cost and Communication Cost.

By utilization of the properties of Ant Colony Algorithm and Particle Swarm Optimization, a mixture calculation is proposed to take care of the voyaging sales representative issues. The calculation first embraces insights system to show signs of improvement arrangements and as per them, gives data pheromone to convey. At that point it makes utilization of the insect state calculation to get a few arrangements through data pheromone collection and reestablishment. At long last, by utilizing crosswise over and transformation operation of molecule swarm enhancement, the successful arrangements are acquired. The Hybrid Algorithm of ACO-PSO has turned out to be viable.

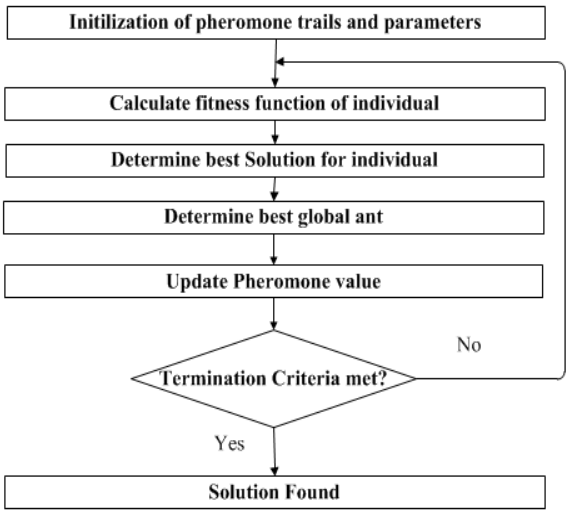

Figure 4 Ant Colony Optimisation Algorithm Flow Chart

Bee colony Optimisation (BCO) algorithm is act as ant colony optimization algorithm but it is based on Group decision. Bee do walve dance to tell other bees about the food and nest place. In this Algorithm, natural process of making decision by bee to select the site for nest or for food is implemented artificially.

E. Indexing files

Files to have the records of data values so that search can be easily done with less time and cost is called indexing files. We have various categories for making index and various complexity of obtaining the search key. The search key will reduce the seek time of finding the data location. It will add more time if search key of data's location is not sought in index file then it will increase the time and also complexity to find the data.

F. Bidding process

In Dispersed Data fashion, we have algorithms which analysis the cost at various sites. Bidding Process involves the steps in which each site gets the query part which is to be execute, each site will make plan and calculate cost function and then send to the main site which process is bidding for executing the plan according to lowest value.

III. CONCLUSION

Distributed database system has various strategies to search the result from dispersed data according to a well defined global schema. Dispersed data has logical correlation which helps to access the result of desired query. In this paper we discussed various strategies and algorithms in brief. These algorithms define the fashion to access the data from

distributed environment. These strategies may have bulky access and high communication costs depend on architecture and schema of dispersed data. Index files provide efficient fashion to search the data from dispersed data framework but it add cost to access function of query. The strategies must be analysed and implement according to desired efficiency in query processing optimisation. A query can be executed in different fashion and at different cost depend on operation and strategies. The selection of algorithm is done according to required range of efficiency in query processing.

REFERENCES

[1] T. e. a. C. Alp, "Optimizing queries with foreign function in a distributed environment," IEEE Trans on Knowledge and Data Eng, pp. 809-824, 2002.

[2] a. M. Fan Yuanyuan et, "Distributed Database System Query Optimization Algorithm Research," International Journal of Advanced Research in Computer Science and Software Engineering, pp. 657-660, 2000.

[3] M. P. T. e. a. S. V. Chande, "Query Optimization In Distributed Environment," in Proceedings of 2nd International Conference on Emerging Trends in Engineering and Management, 2013.

[4] D.Kossmann, "T he State of the Art in Distributed Query Processing," ACM Computing Surveys, pp. 422-469, 2000.

[5] M. Gregory, "Genetiic Algorithm Optimization of Distributed Database Queries," International Journal of Advanced Research in Computer Science and Software Engineering, pp. 290-298, 2012.

[6] M. P. T. e. .. D. S. V. Chande, "Optimization of Distributed Database Queries Using Hybrids of Ant Colony Optimization Algorithm," International Journal of Advanced Research in Computer Science and Software Engineering, pp. 609-614, 2013.

[7] E. Z. e. a. M. R. F. Derakhshi, "Presenting a New Method for Optimizing Join Queries Processing," in Third International Conference on Knowledge Discovery and Data Mining, 2010 .

[8] S. B. e. a. A. Sengupta, "A cyclic multi-relation semijoin operation for query optimization," International Journal of Advances in Engineering Sciences, pp. 101-107, 2007. [10] Hsiao-Fei Liu, Ya-Hui Chang and Kun-Mao Chao. An Optimal Algorithm for Querying Tree Structures and its Applications in Bioinformatics. ACM SIGMOD Record Vol. 33, No. 2, June 2004.

