QueryProcessing for Relational Database by using University Students' Data

Yi Mon Win[1]

Abstract

Initially, databases are basically used for transaction oriented processing of operative data. Today, databases are also used to facilitate reporting and analysis on consolidated, historic data. Thus, the significance of database systems is increasing day by day. This significance results in complexity in data queries as well as increasing demand of efficiency in query processing. Query processing focus to the activities involved in extracting data from a data database. Query processing refers a process of translating a query written in a high-level language into low-level data manipulation operations. In query processing, one of the most important steps is query optimization. The objective of query optimization is to support minimum response time and maximum throughput. Query optimization is a fundamental part of any database management system (DBMS). In this paper, suggests about the various stages that a user submitted query has to pass through during the optimization process by using Dagon university students' data that start and end of graduated year.

Keywords: databases, processing, optimization.

Introduction

Database Management Systems (DBMS) obtains enormous amount of data with the wide spread application, users have to deal with large amount of data. Therefore, it is necessary to store this information in such a way that it can be retrieved from the database in the fastest possible manner to satisfy the request from a user. Databases are most useful in representing data in an organized manner. It provides the user with the ability to acquire accurate, reliable and timely data for effective decision making process. Thus, the significance of database systems is increasing day by day. At the same time, data queries are becoming more and more complex.This is first transformed into a standard high-level query language, such as SQL (Structured Query Language). This SQL query is read by syntax analyzer so that it can be check for correctness. At this step the syntax analyzer use the grammar of SQL as input and the parser portion of the query processor check the syntax and verify whether the relation and attributes of the requested query are defined in database. The relational algebraic expression now passes to the query optimizer. Here optimization is performed by substituting equivalent expression depends on the factors such that the existence of certain database structures, whether or not a given file is stored, the presence of different indexes & so on.

Query process

The query classifying is the first phase of the query processing that is transfer the high-level query into a relational algebra query and to check whether that query is syntactically and semantically correct. Thus the query classifying is start with a high-level query and transform into query graph of low-level operations, which satisfy the query. The SQL query is classified into query blocks (low-level operations), which form the basic unit. Hence nested queries within a query are identified as separate query blocks. The query decomposer goes through five stages of processing for decomposition into low-level operation and translation into algebraic expressions.

Query is used for accessing the database in an efficient manner. It is an art of obtaining desired information in a predictable, reliable and timely manner. Formally defines query

[1]Dr, Associate Professor, Department of Computer Studies, Dagon University

optimization as a process of transforming a query into an equivalent form which can be evaluated more efficiently.

During the optimization phase, the optimization engine performs various analyses on the query data. It applies various rules to the internal data structures of the query to transform these structures into equivalent and efficient representation. It then generates valid evaluation plans based upon the rules applied. From the generated evaluation plans, the best evaluation plan to be executed is determined and passed onto the query execution engine.The final phase in processing a query is the evaluation phase. During the evaluation phase, the best evaluation plan generated by the optimization engine is selected and then executed. Figure 1 shows the steps of query processing.

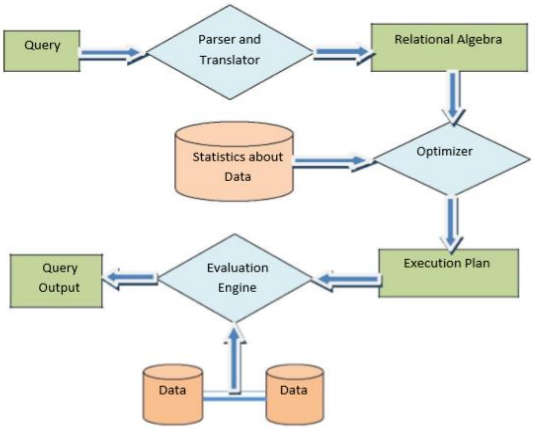

Figure 1. Steps in Query Processing

Query processing is concerned with execution of a query or refers to the activities involved in extracting data from a data database. On the other hand, query optimization process deals with the efficiency of the query. It defines the execution plans, the strategy of execution of the query and chooses the best execution plan. The next step is an optimization step that transforms the initial algebraic query using relational algebra transformation into other algebraic queries until the best one is found. The next step is called code generator, where we generate code for the selected query execution plan. This code is then executed by the run time database processor to produce the query result. The run time database processor has the task of running the query code, whether in compiled or interpreted mode, to produce the query result. If a run time error results, an error message is generated by the run time database processor. Figure 2 shows the different steps of query processing.

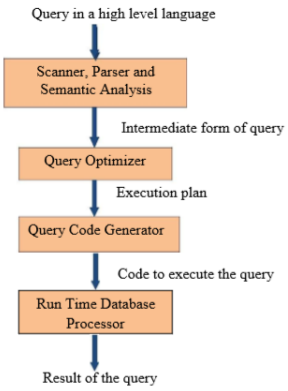

Figure 2. Step of Query Processing

Query optimization process becomes a complex task as query complexity increases with new application. Significant research work has been done in developing efficient query optimization techniques for processing complex queries in a cost effective manner. Some of the popular optimization techniques have been discussed in the following section.

Query Processing Methods

Query processing is the process of identifying an efficient way to execute the given query,so that with less time complexity we can obtain efficient results. In general, query optimization is performed by splitting the query into number of small query parts and executes them in different orders in such a way to reduce the time complexity. However,the query parts have different dependencies between them and the earlier methods do not handle this issue to reduce the time complexity and to improve the performance of query optimization. The various query processing techniques described are as follows: Heuristic optimization, Syntactical optimization, Cost based optimization and Semantic optimization. In this paper analyze the two query optimization techniques among these techniques.[1]

Cost Based Processing

The motivation behind cost based optimization is to come up with the cheapest plans available for each SQL statement. The cheapest plan is the one that will use the least amount of resources. The main objective of cost based optimization is to estimate the cost of different equivalent query expressions and choose the execution plan with the lowest cost. It mainly depends on two factors they are solution space and cost function. Solution space depends on the set of equivalent algebraic expressions and the cost function is equivalent to the summation of input/output cost, CPU cost and communication cost. The steps involved in cost based optimization are as follows: parsing, transformation, implementation and plan selection based on cost estimates. [4]

The cost of a query plan depends on the size of the basic tables referenced as well as the size of the intermediate results. To estimate the size of the intermediate results,we use the concept of selectivity factor. The selectivity factor roughly corresponds to the fraction of rows which are expected to satisfy a condition in the WHERE clause. The cost based optimization; an optimizer needs specific information about the stored data. This information is extremely system dependent and can include information such as file size, file structure types, available primary and secondary indices and attributes selectivity. A realistic goal of a cost based optimizer is not to produce the optimal execution plan for retrieving the required data, but to provide a reasonable execution plan. [2]

Semantic Processing

A semantic query is a query pertaining to knowledge or data that is expressed purely on the basis of a common business vocabulary, without any reference to how or where the data is stored. A semantic query attempts to help a user to obtain or manipulate data in a database without knowing its detailed syntactic structure. The term Semantic query optimization refers to the process of utilizing the integrity constraints in the optimization process. Semantic query optimization results in the transformation of an input query into a semantically equivalent query. Two queries are said to be semantically equivalent if, for every state of the database they produce the same result. [7]

The two major phases involved in semantic optimization. In the first phase, the optimizer locates applicable semantic knowledge and proposes a sequence of one or more reformulation operations that preserve the semantics of the query. It uses two forms of semantic knowledge which are semantic rules and range facts. During the second phase, it

evaluates the proposed reformulations and applies the best reformulation based on a cost model of query execution. [3]

Semantic query optimizer proposes reformulations depending on the applicable semantic rules. It operates on the premise that the optimizer has a basic understanding of the actual database schema. When a query is submitted, the optimizer uses its knowledge of system constraints to simplify or to ignore a particular query if it is guaranteed to return an empty result set. This technique holds great promise for providing even more improvements to query processing efficiency in future relational database systems. [7]

Result and Discussion of Proposed System for Student Intake Result Management

The proposed method generates the input query consists of number of objects or databases or data sets. Each data set has its own schema and number of tables or relational objects where the original information is stored. Each relational object has number of properties or attributes which constructs the rows of a table. For any simple execution of a small query, the query execution module has to possess the schema of the relational object and has to identify which object is necessary to perform the execution of input query. The problem domain of this system is query (searching) the total data of students through first year to final students within graduated year who are missing by resigned, transferred and resigned. User can search total student data of pass students, fail students, resigned student, pass away students, transferred students and graduated students within their graduated year.

User can query the detail data of total students by major by namely, student ID, father name, address, etc.This system has two sites, admin site and students' site. In student site, student can register who want to enroll the university. In admin site of this system, admin staff can enter user name and password correctly for student data security. Authority person or admin user can view, update, search and calculate by data of intake and academic year. The main effective of this system is finding the student data that missing students of register to first year to final year of their graduated year. Admin user can query thetransfer student, resign student, pass away etc.In this system has admin site and student site. Student can fill the register form only data input part. Firstly, student have to register to the university with register form, students clicks the submit button after filled the register form. Total register students data of Dagon University are shown in figure 3.

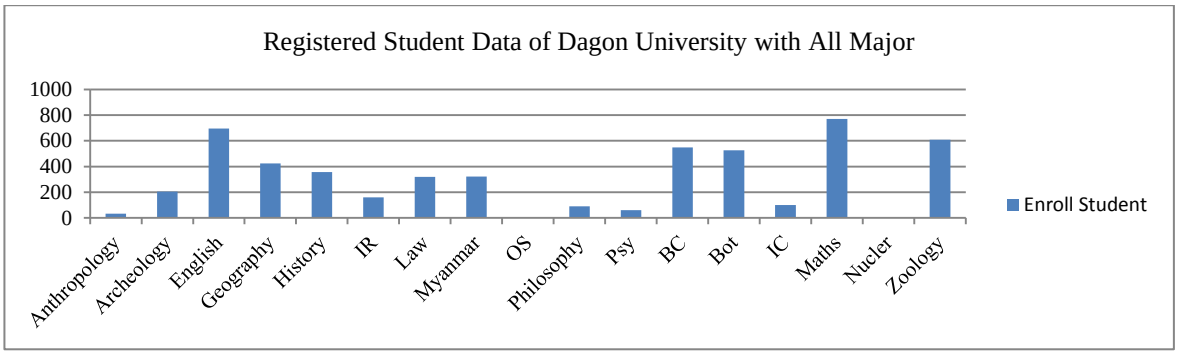

Figure 3. Register Student Data of Dagon University in 2012-2013

Its query languages are:

Registeration Query

```
string query = @"Insert into Academic_Year_Registration values( '" + txtStudentID.Text + "','" +
txtName.Text + "','" + txtFatherName.Text + "', '" + txtMotherName.Text + "','" + gender + "','" +
txtAddress.Text + "','" + txtEmail .Text+ "','" + txtPhoneNumber.Text + "','" +
cboMajor.Text+"','"+txtYear.Text+"','"+ txtAcdemicYear.Text+"','" +  txtIntakeYear.Text+"')";
```

In admin site, admin staff can update, search, view and calculationfrom admin choice form (Figure 4).

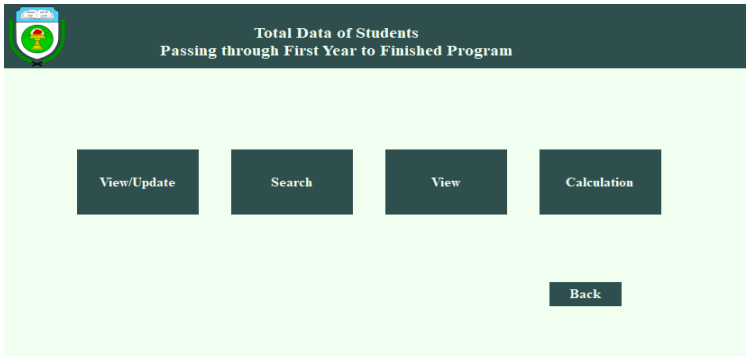

Figure 4. Admin Site View Form

When user clicks the view/update button, the view choice form (Figure 5) will appear.

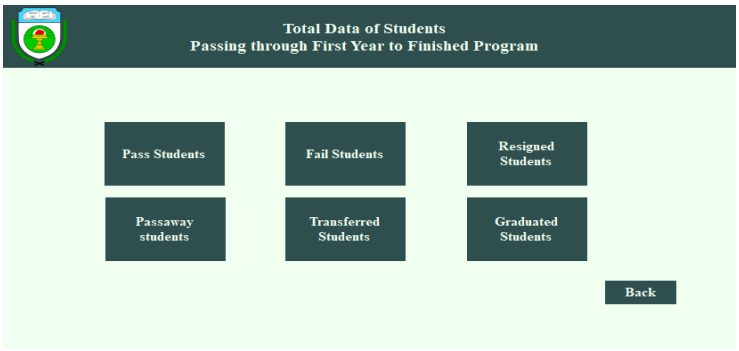

Figure 5. View Choice Form

In view choice form, admin staff can view pass student, fail student, resigned student, pass away student, transferred and graduated student. Query languages processes for these functions are:

Query for Passview

```
privatevoidLoadDataIntoDGV(string cbo1)

        {            string query = @"select StudentID,StudentName, FatherName, Major, Year,
AcademicYear,IntakeYear,Sync from StudentPass where AcademicYear='" + cbo1 + "'";

DataTabledt = da.RetrieveDataTable(query);
```

Query for resign view

```
privatevoidLoadDataIntoDGV(string cbo1)

        {            string query = @"select StudentID, StudentName, FatherName,Major, Year,
AcademicYear, IntakeYear, Sync from StudentResign where AcademicYear='" + cbo1 + "'";

DataTabledt = da.RetrieveDataTable(query);
```

Query for Transfer Student

```
privatevoidtxtsearch_TextChanged(object sender, EventArgs e)

        {     adapt = newSqlDataAdapter("select*from StudentTransfer where StudentID like '" +
txtsearch.Text + "%'", con);
```

When user clicks the graduated student button, the graduated students form will appear. In this form, admin user can query by academic year see in figure 6.

Figure 6. Graduated Students Form

For Graduated Students Query

```
privatevoidLoadDataIntoDGV(string cbo1)

        {  string query = @"select
StudentID,StudentName,FatherName,Major,Year,AcademicYear,IntakeYear,Sync from StudentGraduate where
AcademicYear='" + cbo1 + "'";
```

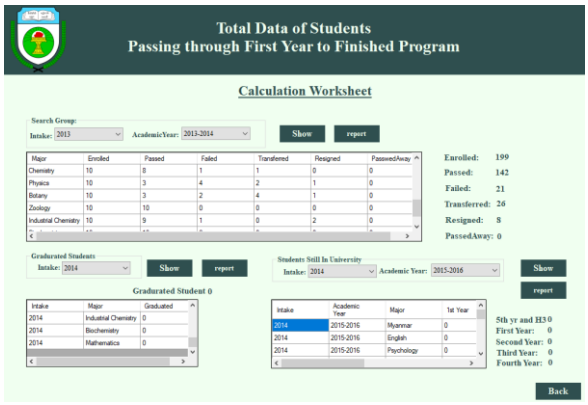

Figure 7. Calculation Work Sheet

When admin user clicks the calculation button from view choice from, Calculation work sheet form will appear shown in figure 7. This form summarized the Tracing System for Missing University's Students. Admin user can query total data of all major students such as enrolled, pass(graduated), failed, transferred, resigned and pass away students by intake year and academic year. As the result of this figure, largest amount is pass student as 64 % of total register of that academic year and fail student is the 19% of total register student (figure 8).

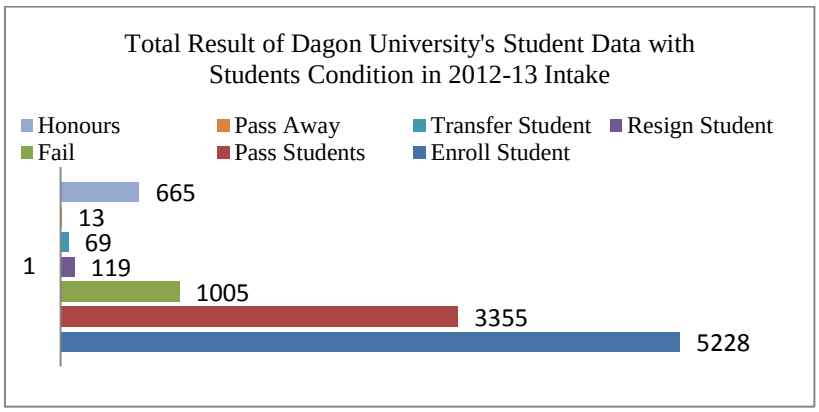

Figure 8.Total Results of Dagon University's Student Data with Students Condition in 2012-13 Intakes

Conclusion

In this research, analyze the query processing by using the university student data with intake result management system. This system has been developed by using Microsoft visual studio 2015 and SQL server database are used to implement this proposed system.

This paper mainly focuses on facilitating the enhancement of query performance, which has being the increase in demand for many database applications. In this paper studied the role of query processing in the relational database. The primary task of the query processing is to transform a declarative SQL statement into a query execution plan by constructing alternative query execution plans.Thus query optimizer is widely considered to be the most important component of a database management system. In this research learned the various related research work carried out in the field of query processing. The increase in the number of relations in the database has emerged the need for creating new SQL queries that can get required data from the database.

In this paper, discuss about the factors that play a vital role in the performance tuning of the user query. The various factors identified are query optimizer, equivalence rules, indexing, cost estimation and dependencies. Another important factor that increases complexity in retrieving data from the databasesis the dependencies that exist among the attributes in a database. Finally, implement the proposed system with the proposed architecture and optimization algorithm using the student data of university. In this system have admin site and student site. Admin can update, search, view and calculation of pass, fail, resigns, pass away and graduated student timely. So, this paper can apply not only effective query processing but also admin user of university can easily and effectively use for total student data of university timely and accurately.

Acknowledgements

I would like to thank Dr Win Naing, (Rector), Dr Nu Nu Yi(Proretor) and Dr Nay ThweKyi (Prorector) of Dagon University, for their kind permission to carry out this paper.

References:

1. HweeHwaPANG ,Hongjun LU, Beng Chin LU An Efficient Semantic Query Optimization Algorithm,7th IEEE International Conference on Data Engineering (ICDE): April 8-12, 1991, Kobe, Japan.
2. KapoorRidhi, R.S.Virk, Selectivity & Cost Estimates in Query Optimization in Distributed Databases.International Journal of Enhanced Research in Management & Computer Applications, ISSN: Vol. 2 Issue 6, 2013
3. Kumar P Mohan and Vaideeswaran (2012),Implementing Semantic Query Optimatization in RelationalVolume 52– No.9, August 2012. 41.
4. Jurgenkoch, Matthias Jarke, Query Optimization in Database Systems, ACM Computing Surveys **(CSUR)** Surveys Homepage archive,Volume 16 Issue 2, June 1984
5. SayliAyla and Lowden Barry, A Fast Transformation Method to Semantic Query Optimisation, International conference on International database engineering and applications symposium, 1997
6. Sheetal S. Dhande, Dr. G. R. Bamnote,Query Optimization in Oodbms: Identifying Subquery for Complex Query Management,Second International Conference on Computational Science and Engineering, **2014**
7. Sreekumar T. shenoy and ZehraMeral Ozsoyoglu,SemanticOptimization**:**IEEE Transactions on Knowledge and Data Engineering, Vol. 1, 1989.