A Novel Approach towards Translating SQL Queries into Spreadsheets

Tasnim T. Hajiwala
Department of Computer Engineering
VPKBIET
Baramati, India
E-mail: tasnim.kayamkhani@gmail.com

Prof. Santosh A. Shinde
Department of Computer Engineering
VPKBIET
Baramati,India
E-mail: shinde.meetsan@gmail.com

Abstract: SQL(Structured Query Language), in today's digitalized world, is present everywhere, since enormous amount of data is being collected every second and stored into the database. Also, SQL Queries can efficiently and very quickly retrieve large amounts of records from database. On the other hand, Database Management Systems require sophisticated hardware and software and highly skilled personnel. There are lot of government and private organizations whose data is in spreadsheet format. Spreadsheets also have diverse features like statistics, visualization, reporting, etc. To avail the RDBMS functionalities, they either need to log shift the data or a query compiler which will execute relational queries over this data and hence migration would no longer be required. And hence, the proposed system that provides a query compiler, which translates a given SQL queries into a worksheet of the same semantics. Queries could be defined using a high level language. We propose a system where the SQL Queries like simple SELECT, SELECT with Conditions, GROUP BY, ORDER BY, JOIN, VIEWS could be easily executed on spreadsheets. Our further contribution to the proposed approach is 'Log Shifting' of spreadsheet data into relational tables.

Keywords: RDBMS, Spreadsheets, SQL Queries

I. INTRODUCTION

SPREADSHEETS are desktop applications which kind of replace databases and OLAP in enterprise-scale computing. They actually serve the similar purpose of data management and data analysis, but it compromises on quantity of data. Being very popular, Spreadsheets are quite often described as the very first killer application for personal computers. Today they are widely used in small scale enterprises, to manage home budgets, to create, examine and manage extremely sophisticated models and data arising in business and research. Even Bill Gates in his keynote talk during SIGMOD 1998 spoke about the role and challenges of spreadsheets. He addressed about the discontinuity which is arising because people don't want to move towards database and want all the database type operation in spreadsheet. Despite this, relatively little research has been done to spreadsheets and hence they are quite less understood. After certain years, Excel users show up at community forums asking for help in performing database operations on their spreadsheet data. The second important fact about spreadsheet is that it is a language of formulas and has become a de facto standard. It is implemented in a large number of spreadsheet systems, available for all major operating systems and hardware platforms, starting from handhelds and ending in the cloud, from proprietary to open source. Computer applications which contain formula-only Spreadsheets are therefore highly portable, and it probably could be compared with Java byte code. Spreadsheet systems also could be regarded as virtual machines, offered by various vendors, on which spreadsheet applications can be run. It is therefore extremely surprising that those machines do not have compilers producing spreadsheet code. So, our proposed system does the task of Compiler which parses the SQL Queries and obtains the result from the Spreadsheet.

II. RELATED WORK

To the best of our literature survey, the problem of expressing relational algebra and SQL in spreadsheets has not been considered in the setting prior to [2]. Then J.Sroka, A.Panasuik, et al. proposed a system which can translate Queries into Spreadsheets using Relational Algebra. Relational Algebra we felt is little difficult to understand and implement.

The Google Visualization API Query Language lets us perform various data manipulations with the query to the data source. The Google Visualization API Query Language is subset of SQL with a few features of its own. It provides a quick and easy way to query a Google spreadsheet and return and display a selective set of data without actually converting a spreadsheet into a database. However, this function does not permit joining relations, and is incompatible with other spreadsheet systems.

MDSHEET, Model driven spreadsheets as proposed by J. Cunha et al.[8] is a framework for the embedding, evolution and inference of spreadsheet models. This framework offers a model-driven software development mechanism for spread- sheet users.

J. Cunha et l. proposed ES-SQL tool [9] which is an embedded tool for visually constructing queries over spreadsheets. This tool provides an expressive query environment which has knowledge on the business logic of spreadsheets. Query Language (ES-SQL) relies on model-driven spreadsheets where a model abstracts the structure and logic of a potentially large spreadsheet. This model allows queries to be expressed by names, instead of column letters, referencing entities.It uses Google's QUERY function for the execution

SQL Query Parser is an automated tool for translating Queries into Spreadsheets. Also it parses the statements into the parse tree and generates the syntax tree providing validation to the statements at an early stage.

Spreadsheet as a relational database engine [2] can play the role of a relational database engine, without any use of macros or built-in programming languages, merely by utilizing spreadsheet formulas. Given a definition of a database in SQL, it is therefore possible to construct a spreadsheet workbook with empty worksheets for data tables and worksheets filled with formulas for queries. From then on, when the user enters, alters or deletes data in the data worksheets, the formulas in query worksheets automatically compute the actual results of the queries. Thus, the spreadsheet serves as data storage and executes SQL queries, and therefore acts as a relational database engine.

III. THE PROPOSED SYSTEM

The functioning of the system is represented as in Fig 1.

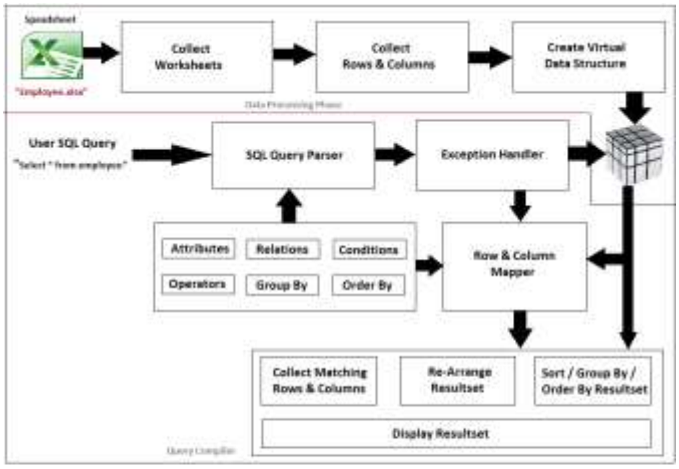

Figure I. System Architecture

The operation of the system is divided into four phases:

- Date Preprocessing phase
- Query Compiler
- Generating Resultset
- Displaying Resultset.

A. *Data Preprocessing phase*

1) Collecting Worksheet: This is the initial phase of the system where the user is asked to Browse the spreadsheet on which he is intended to work upon. The spreadsheet could contain n number of worksheets in it. Those worksheets could be collected by indices, by Name and then the worksheets are arranged in list.

2) Collecting Rows and Columns: The next task is to read the data from the collected Worksheets. The Data is extracted from the Rows with respect to column and it is read vertically.

3) Creating Virtual Data Structure: After fetching rows and columns from the worksheets, corresponding virtual data structure is created. The following entire processes would be done on this virtual data structure. The Relations inside the Virtual data structure would have the same name as those of worksheets. And Attributes would have the same name as of column name of the corresponding worksheet

B. *Query Compiler*

1) Query Parser: The first phase of collection of worksheet and creation of virtual data structure is achieved. In the next phase, the user is asked to execute an SQL Query as per the desired outcome. As soon as the SQL query is initiated, SQL Query Parser starts parsing the query. The following are the tasks of Query Parser:

- To obtain Attributes
- To obtain Relations
- To obtain Conditions
- To obtain Group By Clause
- To obtain Order By Clause
- To obtain Operators
- To obtain Constants

After obtaining the above list of items, the result is passed on to the exception handler.

2) Exception Handler: The parsed SQL Query is checked by Exception Handler. It would throw an exception if any of the following things occur:

- Missing/Wrong Attributes mentioned.
- Missing/Wrong Relations mentioned.
- Misplacement of Clauses.
- Syntactical error.
- Wrong Operator used.
- Nested Query found.

4) Row Mapper: : After the SQL Query has been parsed and there are no exceptions, Row Mapper does the task of collecting matching rows and columns from the data structure as per the query.

C. *Generating Resultset*

1) Rearranging Resultset: The Row Mapper collects matching rows. The Conditions if any are applied to them. The Resultset is rearranged if sorting was required and the resultset is generated.

D. *Displaying Resultset*

The generated Resultset is displayed to the user.

IV. RESULT ANALYSIS

E. *Data Table Discussion*

Sr. No	#Queries	#Attributes	#Relations	#Predicates	#Constants	#Conditions
1.	20	210	20	-	-	-
2.	15	50	30	-	-	15
3.	20	35	20	20	-	-
4.	20	170	20	15	32	30
5.	15	120	30	-	10	22
6.	15	180	25	7	20	35

The above table refers to the data input given to the system. The input consists of the set of different types of Queries which are executed on the WorkSheet that is collected. First set of queries is Simple SELECT Queries. From each query, the number of Relations, Attributes, Conditions, Predicates, and Constants are obtained. Second set of queries is based on INNER JOIN and SELF JOIN. Third set consists of GROUP BY queries whereas Fourth contains SELECT queries with WHERE condition including ORDER BY. Fifth set includes NESTED SELECT queries and sixth is based upon VIEWS.

F. Result Table

Sr. No.	#Queries	#CorrectResult	Precision
1.	20	20	1.0
2.	15	12	0.8
3.	20	15	0.75
4.	20	17	0.85
5.	15	13	0.86
6.	15	14	0.93

The above set of queries is executed by our system on the worksheets and it gave 86% of the accuracy. The efficiency of the system can be measured in terms of precision. It can be derived as:

$$\text{Precision} = \frac{\text{No. of correct queries executed}}{\text{Total No. of queries executed}}$$

So, when the first set of queries were executed which were simple SELECT queries, we obtained 100% accuracy. i.e. precision is 1.0. Similarly, when all the sets of queries were executed, the average precision calculated was 0.865.

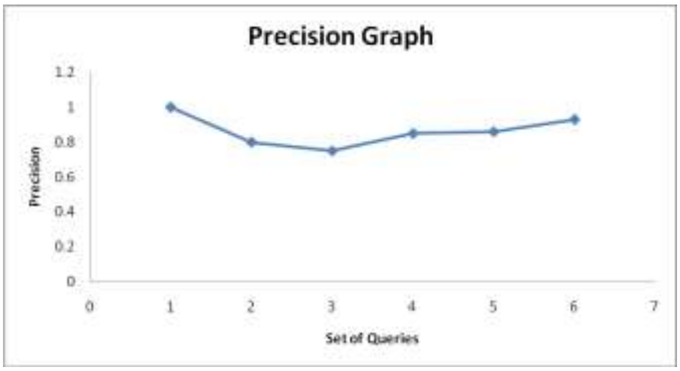

Figure II: Precsion graph

V. CONCLUSION

Thus, the proposed system can easily Query a spreadsheet using myriad SQL statements and obtain desired result. And, hence we can easily work upon spreadsheets. The various SQL statements include simple SELECT, SELECT using WHERE conditions, JOIN, GROUP BY, ORDER BY, VIEWS. Bubble sort has been implemented for sorting the mapped rows and columns. Thus, we have shown the power of the spread-sheet paradigm, which subsumes the paradigm of relational databases. As the next time we plan to develop optimizations for SQL Queries translated into spreadsheets. The future work also includes log shifting of spreadsheets to relational database.

REFERENCES

[1] J.Sroka, A.Panasuik,K.Stencel and J.Tyszkiewicz,"Translating Relational Queries into Spreadsheets," IEEE Transactions on Knowledge and Data Engineering,Aug,2015,pp. 2291-2303.

[2] J. Tyszkiewicz, "Spreadsheet as a relational database engine," Proceedings of the 2010 ACM SIGMOD International Conference on Management of data, New York, NY, USA: ACM, 2010, pp. 195206.

[3] B. Liu and H. V. Jagadish,"A spreadsheet algebra for a direct data manipulation query interface," ICDE 09: Proceedings of the 2009 IEEE International Conference on Data Engineering. Washington, DC, USA:IEEE Computer Society, 2009, pp.417428.

[4] A. Witkowski, S. Bellamkonda, T. Bozkaya, G. Dorman, N. Folkert,A.Gupta, L. Shen, and S.Subramanian,"Spreadsheets in RDBMS for OLAP," SIGMOD 03: Proceedings of the 2003 ACM SIGMOD international conference on Management of data. New York, NY, USA: ACM, 2003,pp. 5263.

[5] A. Witkowski, S. Bellamkonda, T. Bozkaya, N. Folkert, A. Gupta, L.Sheng, and S. Subramanian, "Business modeling using SQL spreadsheets," VLDB 2003: Proceedings of the 29th international conference on Very large data bases. VLDB Endowment, 2003, pp. 11171120.

[6] A. Witkowski, S. Bellamkonda, T. Bozkaya, A. Naimat, L. Sheng, S. Subramanian, and A.Waingold,"Query by Excel," VLDB 05: Proceedings of the 31st international conference on Very large data bases. VLDB Endowment, 2005, pp.12041215.

[7] L. V. S. Lakshmanan, S. N. Subramanian, N. Goyal, and R. Krishnamurthy, "On query spreadsheets," ICDE. IEEE Computer Society, 1998, pp. 134141.

[8] J. Cunha, J. Fernandes, J. Mendes, J. Saraiva, "MDSheet : A framework formodel driven spreadsheet engineering," ICSE-12,Proceeding of 34th National Conference on Software Engineering,1395-1398.

[9] J.Cunha, J.Fernandes, J.Mendes, J.Saraiva,"ES-SQL: Visually Querying spreadsheets, "Visual Languages and Human-Centric Computing(VL/HCC) IEEE symposium,2014.

[10] C.Pei,Y.Cai and Z.Ma,An Indoor Positioning Algorithm Based on Received Signal Strength of WLAN, Pacific- Asia Conference on Circuits, Communications and System, 2009: 516-519.

[11] M. M. Burnett, J. W. Atwood, R. W. Djang, J. Reichwein, H. J. Gottfried,and S. Yang, "Forms/3: A firrst-order visual language to explore the boundaries of the spreadsheet paradigm," J. Funct.Program., vol. 11, no.2, pp. 155206, 2001.

[12] M. Kassoff, L.-M. Zen, A. Garg, and M. Genesereth, "PrediCalc: a logical spreadsheet management system," VLDB 05: Proceedings of the 31st international conference on Very large data bases. VLDB Endowment, 2005, pp. 12471250.

[13] Real spreadsheets for real programmers, in ICCL, H. E. Bal, Ed. IEEE Computer Society, 1994, pp. 2030.

[14] J. V. den Bussche and S. Vansummeren, "Translating SQL into the relational algebra," Lecture material, Universiteit Limburg, lecture INFOH- 417: Database Systems Architecture.

[15] R. Mittermeir and M. Clermont, "Finding high-level structures in spreadsheet programs," in WCRE '02: Proceedings of the Ninth Working Conference on Reverse Engineering (WCRE'02).Washington, DC, USA: IEEE Computer Society, 2002, p. 221.

[16] B. Ronen, M. A. Palley, and J. Henry C. Lucas, "Spreadsheetanalysis and design," Commun. ACM, vol. 32, no.1, pp. 84–93, 1989.

[17] S. K. Shin and G. L. Sanders, "Denormalization strategies for data retrieval from data warehouses," Decis. Support Syst., vol. 42, no. 1, pp. 267–282, Oct. 2006.

[18] P. W. P. J. Grefen and R. A. de By, "A multi-set extendedrelational algebra - a formal approach to a practical issue," in ICDE. IEEE Computer Society, 1994, pp. 80–88.

[19] G. Engels and M. Erwig, "ClassSheets: automatic generation of spreadsheet applications from object-oriented specifications," in ASE'2005. ACM, 2005, pp. 124–133.

[20] J. Cunha, M. Erwig, and J. Saraiva, "Automatically inferring classsheet models from spreadsheets," in VL/HCC'10: IEEE Symp. on Visual Languages and Human-Centric Computing. IEEE Computer Society, 2010, pp. 93–100.

[21] D. Chamberlin, "Xquery: An xml query language," IBM Syst. J., vol. 41, no. 4, pp. 597–615, Oct. 2002.

[22] O. de Moor, D. Sereni, M. Verbaere, E. Hajiyev, P. Avgustinov, T. Ekman, N. Ongkingco, and J. Tibble, ".ql: Object-oriented queries made easy," in GTTSE, ser. Lecture Notes in Computer Science, R. L¨ammel, J. Visser, and J. Saraiva, Eds., vol. 5235. Springer, 2007, pp. 78–133.

[23] S. P. Jones, A. Blackwell, and M. Burnett, "A user-centred approach to functions in Excel," in ICFP '03: Proceedings of the Eighth ACM SIGPLAN International Conference on Functional Programming. New York, NY, USA: ACM, 2003, pp. 165–176.

AUTHOR'S BIOGRAPHIES

1. Ms. Tasnim T. Hajiwala:

Received B.E. degree in Information and Technology from VNSGU, Surat. She has 3 years of teaching experience and now pursuing M.E. degree in Computer Engineering at VPKBIET, Baramati, Savitribai Phule Pune University after a sabbatical period of 8 years.

2. Prof. Santosh A. Shinde

Received his B.E. degree in computer engineering (First class with Distinction) in year 2003 from Pune University and M.E. degree in computer engineering (First class with Distinction) in year 2010 from Pune University. He has 14 years of teaching experience at undergraduate and postgraduate level. Currently, he is working as Assistant Professor in Department of Computer Engineering of VPKBIET, Baramati, Savitribai Phule Pune University. His autobiography has been published in Marquis Who's Who, an International Magazine of Prominent Personalities of the World in the year 2012. He is also a Life Member of IACSIT and ISTE professional bodies. His research interests are Digital Image Processing and Web Services.