isrjournals

International Journal of Advanced Research in Computer Science Engineering and Information Technology

Volume: 4, Issue: 3,Special Issue: 2 ,Apr,2016 ,ISSN_NO: 2321-3337

TRANSLATING RELATIONAL QUERIES INTO SPREADSHEET

Kiruthika.R[1], Ms.Deepika.K[2]

Student, Dept. of Computer Science and Engineering, Agni College of Technology, India.[1]

Asst. Professor, Dept. of Computer Science and Engineering, Agni College of Technology, India. [2]

ABSTRACT:

Data mining the extraction of hidden predictive information from large databases, is a powerful new technology with great potential to help companies focus on the most important information in their data warehouses. Spreadsheets are the desktop counterpart of databases and OLAP in enterprise scale computing. They serve basically the same purpose data management and analysis but at the opposite extreme of the data quality scale. We demonstrate that a spreadsheet can implement all data transformation definable in SQL merely by utilizing spreadsheets formulas. We provide a query compiler, which translates any given SQL query into a worksheet of the same semantics, including NULL values. Thereby database operations become available to the users who do not want to migrate to a database. They can define their queries using a high level language and then get their execution plans in a plain spreadsheets.

Spreadsheets are among the most commonly used applications for data management and analysis. They combine data processing with very diverse supplementary features: statistics, visualization, reporting, linear programming solvers, Web queries periodically downloading data from external sources, etc. However, the spreadsheet paradigm of computation still lacks sufficient analysis.

Keywords— Relational databases, physical database design prototypes, spreadsheets, query languages.

1. INTRODUCTION

They combine data processing with very diverse supplementary features: statistics, visualization, reporting, linear programming solvers, Web queries periodically downloading data from external sources, etc. However, the spreadsheet paradigm of computation still lacks sufficient analysis. Computer applications in the form of formula-only spreadsheets are therefore highly portable, probably to the extent comparable with Java byte code. Spreadsheet systems can be regarded as virtual machines, offered by various vendors, on which spreadsheet applications can be run. It is therefore extremely surprising that those machines are predominantly programmed manually, with no compilers producing spreadsheet code from higher-level languages.

In our proposed system, we offer a fully automated method to construct spreadsheet implementations for a wide class of relational data transformations. We have re-implemented all operators of relational algebra to accept a variable number of input columns and to support NULL values. The full automation of the translation process reduces the number of human-introduced errors in the spreadsheet application. End users can still work in the vanilla spreadsheet, benefit from its features like data analysis and visualization, while the complex formulas are generated by a tool that allows to express them in a better suited high level language and avoids errors. One of them is an efficient sorting algorithm, implemented by spreadsheet formulas, improving on the quadratic sorting.

2. SCOPE OF THE PROJECT

2.1. OBJECTIVE:

The main aim of this project is to propose translating relational queries into spreadsheets and can define their queries using a high level language and then get their execution plan in spreadsheet.Spreadsheets are the desktop counterpart of databases and OLAP in enterprise scale computing. They serve basically the same purpose data management and analysis but at the opposite extreme of the data quality scale. We demonstrate that a spreadsheet can implement all data transformation definable in SQL merely by utilizing spreadsheets formulas.

2.2. SCOPE:

Spreadsheets are the desktop counterpart of databases and OLAP in enterprise scale computing. They serve basically the same purpose data management and analysis but at the opposite extreme of the data quality scale. We demonstrate that a spreadsheet can implement all data transformation definable in SQL merely by utilizing spreadsheets formulas. We provide a query compiler, which translates any given SQL query into a worksheet of the same semantics, including NULL values. Thereby database operations become available to the users who do not want to migrate to a database. They can define their queries using a high level language and then get their execution plans in a plain spreadsheets.

3. SYSTEM ANALYSIS

EXISTING SYSTEM

They combine data processing with very diverse supplementary features: statistics, visualization, reporting, linear programming solvers, Web queries periodically downloading data from external sources, etc. However, the spreadsheet paradigm of computation still lacks sufficient analysis. Computer applications in the form of formula-only spreadsheets are therefore highly portable, probably to the extent comparable with Java byte code. Spreadsheet systems can be regarded as virtual machines, offered by various vendors, on which spreadsheet applications can be run. It is therefore extremely surprising that those machines are predominantly programmed manually, with no compilers producing spreadsheet code from higher-level languages.

Limitations

- Retrieving Null values from spreadsheet is difficult.
- Implementation of sorting and applying join on spreadsheet is not efficient.
- Quality of maintaining spreadsheet is not feasible.

4. MAIN FEATURES

- In our proposed system, we offer a fully automated method to construct spreadsheet implementations for a wide class of relational data transformations.
- We have re-implemented all operators of relational algebra to accept a variable number of input columns and to support NULL values.
- The full automation of the translation process reduces the number of human-introduced errors in the spreadsheet application.

- End users can still work in the vanilla spreadsheet, benefit from its features like data analysis and visualization, while the complex formulas are generated by a tool that allows to express them in a better suited high level language and avoids errors.

5. IMPLEMENTATION

Application Creation & Inserting Details:

In this phase, we are going to develop an ecommerce application for mobile shop. In this application owner as to maintain the details of the each product (incoming and outgoing). To maintain all products details by using excel sheet. All the product information's are stored in excel sheet for easy access. Owner has to insert all the product details, customer details, workers details and bill details in their own excel database.

Implementing algebraic notation:

Once excel database has been created, we need to implement the algebraic notation on it. Here we are going to perform selection, union, difference and duplicate removal information. Translating all relational queries into excel sheet for performing notation. All the database will not allow duplicate records into their database. Every application has to maintain removing of duplicate values. In this application, algebraic notations are used when entering product details, viewing product details and entering bill details.

Performing Aggregation function

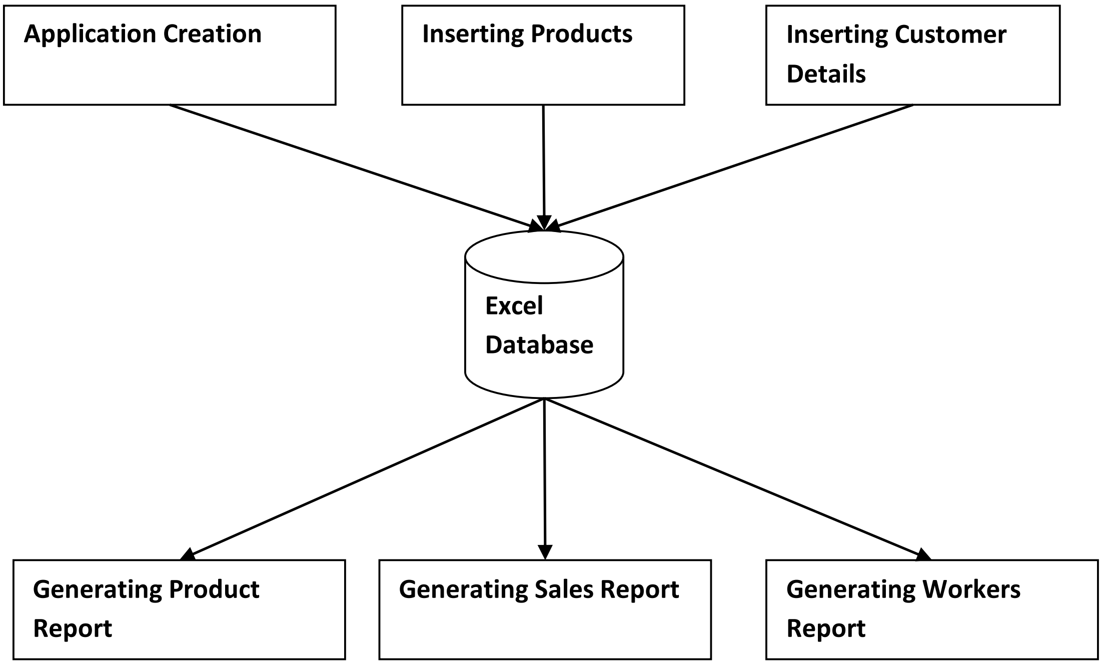
In excel sheet we need to implement aggregation functions. To perform aggregation we need to transform relational queries for excel sheet. For quick searching on product we are going to apply aggregation function. Aggregation functions are max, min, count, sum, less than, greater than etc,

Sorting & Searching details:

In this phase, we are going to implement sorting and searching operations on excel sheet. Sorting operations will be apply on showing workers performance of every month. For sorting we are implementing both ascending and descending order. For searching details, we are implementing BFS and DFS searching algorithm. To implementing searching algorithms, we are using join operations on excel sheet. Using join query, searching will be performing on excel sheet.

6. QUERY

The MySQL server provides a database management system with querying and connectivity capabilities, as well as the ability to have excellent data structure and integration with many different platforms. It can handle large databases reliably and quickly in high-demanding production environments. The MySQL server also provides rich function such as its connectivity, speed, and security that make it suitable for accessing databases.The MySQL server works in a client and server system. This system includes a multiple-threaded SQL server that supports varied backends, different client programs and libraries, administrative tools, and many application programming interfaces (API).


Application Creation
Inserting Products
Inserting Customer Details
Excel Database
Generating Product Report
Generating Sales Report
Generating Workers Report

System Architecture

7. CONCULSION AND FUTUREWORK

CONCLUSION

Thus we designed and developed translating relational queries into spreadsheets and can define their queries using a high level language and then get their execution plan in spreadsheet. We have demonstrated that SQL can be automatically translatedinto spreadsheet code. Thus, we have shown the power of the spreadsheet paradigm, which subsumes the paradigm of relational data bases.A part from SQL, we have also implemented a few specific algorithms: a line arithmetic sorting procedure and two graph traversing algorithms: BFS and DFS.As the next step plan to develop optimizations for SQL queries translated into spreadsheets. We are also interested whether spreadsheets can naturally implement other models of databases, like semi-structured or object-relational ones.

FUTURE WORK

On the practical side, we consider using our implementationsof algebra operators to construct a translator capable ofproducing spreadsheet files compatible with other spreadsheetsystems from those downloaded from Google docs.This requires translating Google specific functions QUERY,SORT, FILTER and UNIQUE which are not recognized byother spreadsheet systems.

REFERENCE

[1] B. Gates, “Investing in research, SIGMOD conference 1998 keynote speech, video,” ACM SIGMOD Digital Symp. Collection, vol. 1, no. 2, 1999

[2] SirMille. (2012, Feb.). inner/outer/full join tables? [Online]. Available: http://www.mrexcel.com/forum/excel-questions/612236- inner-outer-full-join-tables.html

[3] xil. (2012, May). Connecting list of users with list of companies— please help [Online]. Available: http://www.mrexcel.com/ forum/excel-questions/642466-connecting-list-users-list-companies- please-help.html

[4] stefanaalten. (2013, May). Combining two lists into one big list [Online]. Available: http://www.mrexcel.com/forum/excelquestions/ 704217-combining-two-lists-into-one-big-list.html

[5] mzalikhan. (2012, Apr.). Double V-look up [Online]. Available: http://www.mrexcel.com/forum/excel-questions/625750- double-v-look-up.html

[6] G. Inc. Query Language Reference (Version 0.7) [Online]. Available: https://developers.google.com/chart/interactive/ docs/querylanguage, accessed 2014.