



IJIRCCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 12, Issue 3, March 2024

ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 8.379

 9940 572 462

 6381 907 438

 ijircce@gmail.com

 www.ijircce.com

Implementation and Analysis of FSQL to SQL Layer

Dr. Subita Kumari

Assistant Professor, Govt. PG College for Women, Rohtak, India

ABSTRACT: Databases are a very important component in computer systems. Because of their increasing number and volume, good and accurate accessibility to a database becomes even more important. When users work with usual software tools they have to change their many valued logical thinking (approximate reasoning) into the two-valued computer logic. Although the Structured Query Language (SQL) is a very powerful tool, it is unable to satisfy needs for data selection based on linguistic expressions and degrees of truth. The goal of the research whose results are presented in the paper is to capture these expressions and make them suitable for queries. For this purpose the fuzzy generalized logical condition for the WHERE part of SQL is developed. In this way, queries based on linguistic expressions are supported and are accessing relational databases in the same way as with the SQL. Fuzzy query is not only a querying tool; it improves the meaning of a query and extracts additional valuable information. In this paper, we are interested in flexible querying that is based on fuzzy set theory. To model the flexible queries and the concept of fuzzy attributes, an extension of the SQL language named fuzzy SQL has been defined. This architecture is based on the concept of weak coupling with the DBMS SQL Server.

KEYWORDS: SQL, FSQL, Membership Function, Fuzzy Querying

I. INTRODUCTION

This paper observes situations when database querying process by the two valued realization of Boolean algebra is not adequate and offers solution based on the fuzzy logic because the fuzzy logic is an approach to computing based on "degrees of truth" rather than the usual "true or false" logic. The fuzzy logic deals with reasoning that is approximate rather than precise to solve problems in a way that more resembles human logic.

Organizations work with very large data collections mainly stored in relational databases. Linguistic expressions are interesting for data extraction, analysis, dissemination and decision making. The research area of fuzziness in Data Base Management Systems (DBMS) has resulted in a number of models aimed at the representation of imperfect information in Data Bases, or at enabling non-precise queries (often called flexible queries) on conventional database schemas. However, few works have been done from a practical point of view.

In this paper, we are interested to the works of Medina *et al.* who introduced the GEFRED model in 1994 and its associated language named FSQL. This language presents new concepts such as fuzzy comparators, fuzzy attributes, fuzzy constants, *etc.* We propose to implement a new architecture of the Fuzzy Relational DBMS (FRDBMS) based on the GEFRED model. This architecture is based on the weak coupling principle with the RDBMS SQL Server.

II. BASIC CONCEPTS

A. General Definitions

- Database: A classical database is a structured collection of information (records or data) stored in a computer.
- Fuzzy Database: A fuzzy database is a database which is able to deal with uncertain or incomplete information using fuzzy logic.
- Fuzzy Logic: Fuzzy logic is derived from fuzzy set theory by Zadeh (1965) dealing with reasoning that is approximate rather than precisely deduced from classical predicate logic. It can be thought of as the application side of fuzzy set theory dealing with well thought out real world expert values for a complex problem.
- FRDB: It is an extension of the relational database. This extension introduces fuzzy predicates under shapes of linguistic expressions that, at the time of a flexible querying, permits to have a range of answers (each one with a membership degree) in order to offer to the user all intermediate variations between the completely satisfactory answers and those completely dissatisfactory.

- FRDBMS: It is an extension of the relational DBMS in order to treat, store and interrogate imprecise data.
- Flexible Query: A query whose restrictions, or conditions, are weakly defined. Usually, the restrictions of this kind of queries are modeled as fuzzy sets. The results for this query are allowed to partially match the conditions of the query.
- FRDB Models: Two broad approaches are possibility model and the similarity relation based model. These models are considered in a very simple shape and consist in adding a degree, usually in the interval [0, 1], to every tuple. They allow maintaining the homogeneity of the data in DB. The main models under both approaches are Prade-Testemale, Umano-Fukami, Buckles-Petry, Zemankova-Kaendel and GEFRED of Medina *et al.*

B. SQL and its Limitations

Users search databases in order to obtain data needed for analysis, decision making or to satisfy their curiosity. The SQL is a standard query language for relational databases. The simple SQL query is as follows:

```
select attribute_1, ..., attribute_n from T
where attribute_p > P and attribute_r < R
```

The result of the query is shown in graphical mode in figure 1. Values P and R delimit the space of interesting data. Small squares in the graph show database records. In the graph it is obviously shown that three records are very close to meet the query criterion.

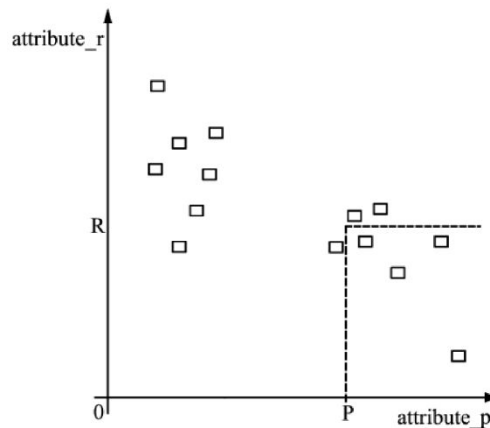


Figure 1: Result of Classical Query

The SQL uses the crisp logic in querying process that causes crisp selection. It means that the record would have not been selected even if it is extremely close to the intent of the query criterion. As the criterion becomes more and more complex, the set of records selected by the WHERE statement becomes more and more crisp. If the classical SQL is used for solving this problem, the SQL relaxation would have to be used in the following way:

```
select attribute_1, ..., attribute_n from T
where attribute_p > P-p and attribute_r < R+r
```

where p and r are used to expand the initial query criteria to select records that almost meet the query criteria. This approach has two disadvantages:

- First, the meaning of the initial query is diluted in order to capture adjacent records. The meaning of a query: “where attribute_p is more than P” is changed and adjacent records satisfy a query in the same way as initial ones.
- Secondly problem arises from the question: what about records that are very close to satisfy the new expanded query and it is useful to make another expanding of a query. In this way more data from the database is selected, but the user has lost the accuracy of his query.

Adding some flexibility to the SQL meets above mentioned requirements and increases effectiveness and comprehensibility of the whole querying process.

III. FUZZY QUERING PROCESS

The querying process consists of the two steps. In the first step lower and/or upper bounds of linguistic expressions (fuzzy sets) are used as parameters for database queries. It means that all records that have satisfaction degree greater

than zero are selected only. In the second step the chosen analytical form of the fuzzy set is used to calculate the membership degree of each selected record to appropriate fuzzy set.

To implement this, a fuzzy query interpreter, which transforms fuzzy queries to the classical SQL structure, was developed. In this way, queries based on linguistic expressions on client side are supported and are accessing relational databases in the same way as with the classical SQL. Figure 2 shows this model. The first step of querying is situated in parts 1, 2 and 3. Lower and/or upper limits of linguistic expressions are calculated and converted into SQL query in the part 1. Thus created SQL query selects data from database (part 2) and saves it into the temporary table (part 3). The second step uses data from part 3 for further calculations. Firstly, the chosen analytical form of the fuzzy set (from part 1) is used to determine the membership degree of each selected record to appropriate fuzzy set. These calculations are situated in the part 4. Finally the result of fuzzy query is displayed in part 5.

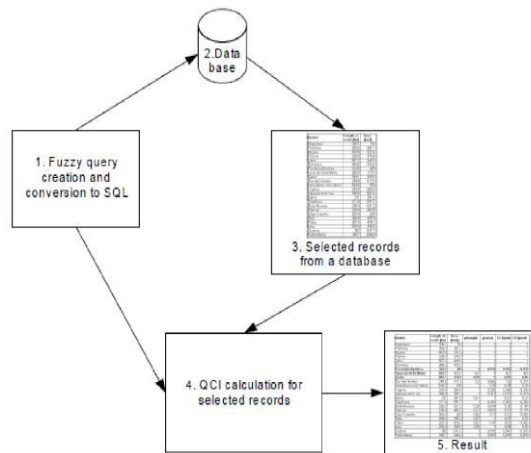


Figure 2: Structure of Fuzzy SQL

IV. FUZZY MEMBERSHIP FUNCTION

We are using fuzzy membership function to express fuzziness in the query. Zadeh proposed a series of membership functions that could be classified into two groups: those made up of straight lines, or “linear,” and Gaussian forms, or “curved.” In our work we are using Trapezoid function.

Trapezoid Function (Figure 3): Defined by its lower limit a , its upper limit d , and the lower and upper limits of its nucleus or kernel, b and c , respectively:

$$T(x) = \begin{cases} 0 & \text{if } (x \leq a) \text{ or } (x \geq d) \\ (x-a)/(b-a) & \text{if } x \in (a,b) \\ 1 & \text{if } x \in [b,c] \\ (d-x)/(d-c) & \text{if } x \in (c,d) \end{cases}$$

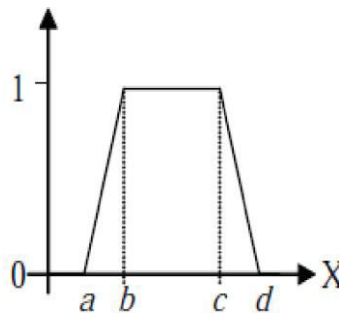


Figure 3: Trapezoidal fuzzy set

V. FRDBMS ARCHITECTURE

We propose the weak coupling approach with DBMS. The concept of weak coupling is shown in Figure 4. The FRDBMS proposed respects the GEFERD model. The language of description and manipulation of the data is therefore FSQL. Seen that the FSQL language is an extension of the SQL language, a FRDBMS can model a RDB (described in SQL language) or a FRDB (described in FSQL language). The principle of this coupling is the definition of a software layer that allows the transformation of the command written by the user in FSQL language in their equivalent written in SQL. In order to implement a system which represent and manipulate “imprecise” information, Medina *et al.* have developed FIRST architecture (a fuzzy Interface for relational systems). It is built on RDBMS Client-Server architecture provided by Oracle. It extends the existing structure and adds new components to handle fuzzy information. The main important component added to this architecture is the FSQL Server which assures the translation of flexible queries written in FSQL in a comprehensible language by the DBMS (SQL).

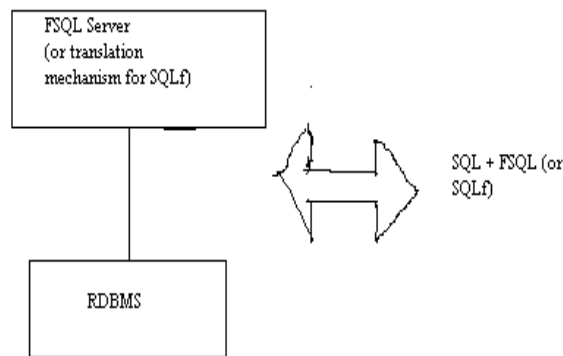


Figure 4: Weak Coupling Concept

VI. RESULT OF FUZZY QUERY

We are using the database of a school to implement the concept of flexible querying. We have three tables namely Teacher, Student and Class. On certain fields (age, salary etc.) of Teacher and Student table we have applied fuzzy conditions. Figure 5 below shows the result of one such query. We have used the DBMS SQL Server to implement the same.

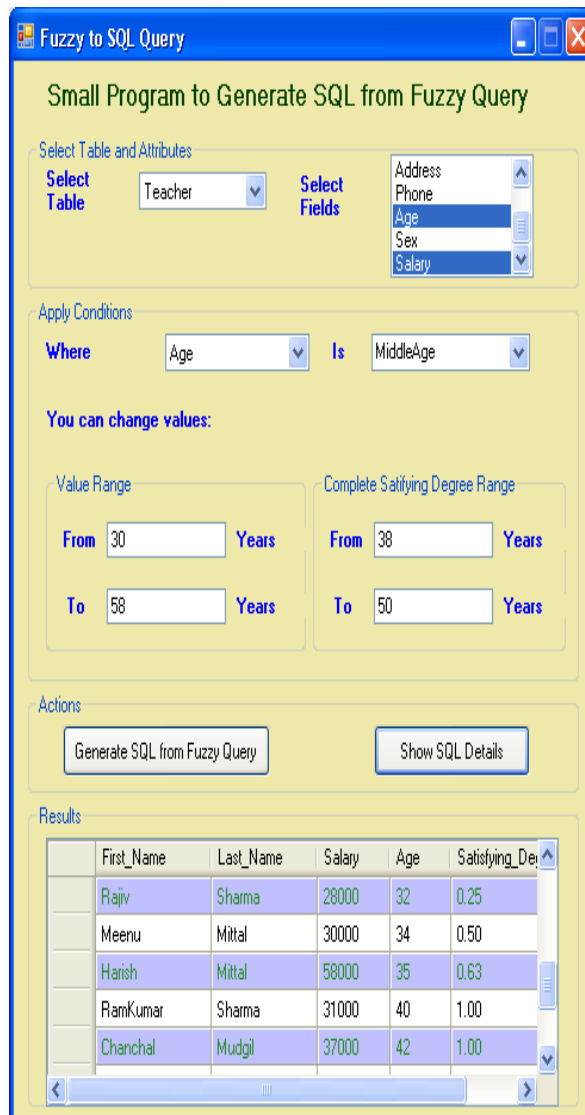


Figure 5: Result of selecting Table–Teacher, Condition Field = Age, Fuzzy Value = Middle Age

In the snapshot shown in figure 5, we can see the resulting rows of table. After selecting required table, field and fuzzy value, we have to click “Generate SQL from Fuzzy query” button. Then we get the resultant table with desired fields and also the calculated attribute “Satisfying Degree” which shows the satisfaction degree of correspondence row as per fuzzy value selected. If we click “Show SQL Details” button, we get the Query in the form of SQL as shown in figure 6.

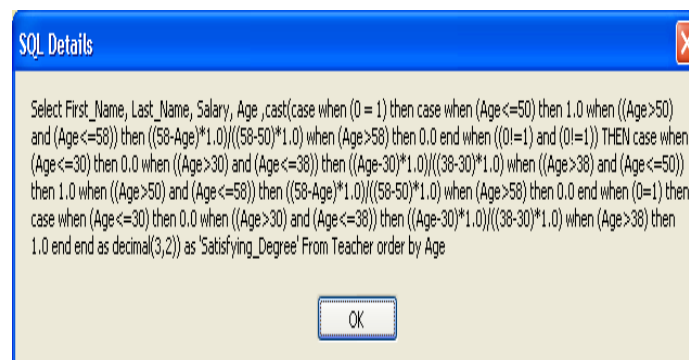


Figure 6: SQL corresponding to Fuzzy Query

VII. CONCLUSION

Fuzzy relational data bases have been extensively studied in a theoretical level. The majority of these works used the fuzzy sets formalism to model the linguistic terms as “moderate”, “means” and to value the predicates including such terms. Medina et al. have developed a server named fuzzy SQL, supporting flexible queries and based on a theoretic model called GEFRED. This server has been programmed in PL/SQL language under Oracle database management systems. To model the flexible queries and the concept of fuzzy attributes, an extension of the SQL language named fuzzy SQL has been defined. The FSQ language extends the SQL language, to support the flexible queries, with many fuzzy concepts. The FRDB is supposed has already been defined by the user.

In this paper, we have extended the work of medina et al. to implement a software layer which will convert FSQ queries to SQL queries. This architecture is based on the concept of weak coupling with the DBMS. This will facilitate the user a powerful and easy to use data mining tool which allows him to query data from databases by using linguistic expressions in order to improve the quality of selection process.

VIII. FUTURE SCOPE

The proposed architecture of FRDBMS based on the GEFRED model makes use of weak coupling concept with the DBMS. As a future work a new architecture supporting the concept of strong coupling with DBMS can be developed. As futures perspectives of this work, we also mention the automatic mapping of existing relational DB to FRDB. This point is theoretically done but not implemented yet, so we think that it will contribute to make easier the use of the FRDB in real applications.

REFERENCES

1. Touzi, A. G., & Hassine, M. A. B. (2009). New architecture of fuzzy database management systems. *Int. Arab J. Inf. Technol.*, 6(3), 213-220.
2. Bosc, P., & Pivert, O. (1995). SQLf: A Relational Database Language for Fuzzy Querying. *Computer Journal of IEEE Transactions on Fuzzy Systems*, 3(1), 80-89.
3. Bosc, P., & Pivert, O. (2000). SQLf query functionality on top of a regular relational database management system. *Knowledge Management in Fuzzy Databases*, 171-190.
4. Galindo, J., Medina, J. M., Pons, O., & Cubero, J. C. (1998). A server for fuzzy SQL queries. In *Flexible Query Answering Systems: Third International Conference, FQAS'98 Roskilde, Denmark, May 13–15, 1998 Proceedings 3* (pp. 164-174). Springer Berlin Heidelberg.
5. Galindo, J., Aranda, M. C., Caro, J. L., Guevara, A., & Aguayo, A. (2002). Applying fuzzy databases and FSQ to the management of rural accommodation. *Tourism Management*, 23(6), 623-629.
6. Galindo, J. (2005). New characteristics in FSQ, a fuzzy SQL for fuzzy databases. *WSEAS Transactions on Information Science and Applications*, 2(2), 161-169.
7. Galindo, J., Urrutia, A., & Piattini, M. (Eds.). (2006). *Fuzzy databases: Modeling, design, and implementation*. IGI Global.
8. Grissa, A., Ben Hassine, A., & Ounelli, H. (2006). Extended_FSQ_Server: A Server for the Description and the Manipulation of FRDB. In *Proceedings of the 4th International Multi Conference on Computer Science and Information Technology (CSIT)* (pp. 454-464).
9. Mama, R., & Machkour, M. (2021). Fuzzy querying with SQL: Fuzzy view-based approach. *Journal of Intelligent & Fuzzy Systems*, 40(5), 9937-9948.
10. Rachid, M., & Mustapha, M. (2022). A Comparative Study of Existing Fuzzy Query Systems of Database. In *Distributed Sensing and Intelligent Systems: Proceedings of ICDSIS 2020* (pp. 465-476). Cham: Springer International Publishing.
11. Mama, R., & Machkour, M. (2019, July). A study on fuzzy interrogation systems of database. In *2019 International Conference of Computer Science and Renewable Energies (ICCSRE)* (pp. 1-6). IEEE.
12. Mama, R., & Machkour, M. (2019, October). A study of fuzzy query systems for relational databases. In *Proceedings of the 4th International Conference on Smart City Applications* (pp. 1-5).
13. Mama, R., & Machkour, M. (2020). Fuzzy questions for relational systems. In *Innovations in Smart Cities Applications Edition 3: The Proceedings of the 4th International Conference on Smart City Applications 4* (pp. 104-114). Springer International Publishing.



INNO  **SPACE**
SJIF Scientific Journal Impact Factor
Impact Factor: 8.379



ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

 **9940 572 462**  **6381 907 438**  **ijircce@gmail.com**



www.ijircce.com

Scan to save the contact details