American Journal of Engineering Research (AJER) 2017
American Journal of Engineering Research (AJER)
e-ISSN: 2320-0847 p-ISSN : 2320-0936
Volume-6, Issue-2, pp-36-41
www.ajer.org
Research Paper Open Access

Query Processing and Interlinking of Fuzzy Object-Oriented Database

Shweta Dwivedi[1], Santosh Kumar[2]
[1](Department of Computer Science & Engineering, Maharishi University of Information Technology, Lucknow, India)
[2](Department of Computer Science & Engineering, Maharishi University of Information Technology, Lucknow, India)

ABSTRACT: *Due to the many limitation and poor data handling in the existing relational database, the software professional and researchers moves towards the object-oriented database which has much better capability to handling the real and complex real world data i.e. clear and crisp data and also have the capability to perform some huge and complex queries in an effective manner. On the other hand, a new approach in database is introduced named as Fuzzy Object-Oriented Database (FOOD); it has all the functionalities as an object-oriented database has including uncertain and vague data also. Therefore, this paper presents an approach of query processing and interlinking of Fuzzy Object-Oriented Database. An attempt is made to provide an improve query interface for fuzzy object-oriented database. A real case study of hospital-based-diagnostic system has taken for creating a fuzzy object-oriented database; a class and sequence diagram is also designed here for graphical representation through popular modeling language i.e. UML.*

Keywords: *Fuzzy Object-Oriented Database (FOOD), FSQL, Query Processing, Class Diagram and Sequence Diagram.*

I. INTRODUCTION

As the object-oriented database is widely used at theoretical and scientific level it hasn't support fuzzy information; for supporting those kinds of information the fuzzy object-oriented database has proposed. According to Zadeh [7] the fuzzy logic is used to represent the fuzziness in the database like vague and uncertain information. In the real world applications different kinds of imperfect and imprecise information or data is stored in the database. However, the existing relational database management system does not handle the complex and huge database that has the vague information. Therefore, the software professionals and researchers moved towards the fuzzy object-oriented database that can handle the huge and complex database with uncertain and vague information. There are two different ways to perform queries in the fuzzy object-oriented database i.e. FSQL and SQLF. These are the extension of SQL that has the several features of fuzzy sets that allows writing flexible conditions in the user defSined queries. It gives the freedom to make supple queries regarding fuzzy attributes. The FSQL inherits the SQL language's commands and queries like select command for expressing several supple queries. While a relational database is a collection of several tables that having data fields into predefined categories. These tables sometimes called the relational schema having one or more data categories in columns. Every row containing a unique instance of data for every category which is defined by the columns. As the object-oriented database is complemented with programming languages like C++, Java, C# etc. while the relational database systems (RDBs) have a poor connectivity and have not compatibility for these applications development languages because the RDBs are designed or applications with different performance requirements. In the comparison with the fuzzy object-oriented database (FOOD) the relational database systems represents the relation between two relations through the foreign key and primary key attributes.

Query processing is one of the main part of any kind of database that represent the process of compilation and execution of any query. A query has several specifications that are basically expressed in an opted database query language like Structured Query Language (SQL). There are two phases in the query processing i.e. compile-time and runtime phases, as the query is performed by the user the query compiler converts the query specification into an executable code, this process known as query compilation. There are several operators which are used for generating the code for the user performed queries. The performed query result is produced by the database engine after interpreting and executing the having the query specifications.

II. LITERATURE REVIEWS

As described earlier a fuzzy logic deals with imprecise or vague values in solving the problems in a way which is more resembles human logic; while a fuzzy query system is an interface to users to get information from database using query languages like SQL (Structured Query Language). There are several fuzzy query implementations have been proposed. Let us first describe the previous done work. Uzochukwu C. et al. [1] have develop yet another flexible query interface for relational databases that is user friendly and has the capability to adequately help users work with databases without a thorough knowledge of database programming. Li Y. [2] has published a handbook of research on innovative database query processing techniques. Sonia and Kumari S. [3] have illustrated the uncertainty relationship using fuzzy object-oriented database and fuzzy relational database. Kumari S. and Sonia [4] have applied classical as well as fuzzy queries on the database. A Relational Data Base Management System (RDBMS) is a software system that provides a convenient and effective method of defining, storing and retrieving precise information stored in the database. Raipurkar A. and Bamnote G.R. [5] have proposed a query processing in distributed database that handles to minimize the query response time and handling fuzziness in database by translating fuzzy queries into SQL. Touzi A. G. and Hassine M. A. B. [6] have extended the work of medina et al. to present a new architecture of fuzzy DBMS based on the GEFRED model. This architecture is based on the concept of weak coupling with the DBMS Oracle. Koyuncu M. and Yazici A.[7] have propose an intelligent object-oriented database architecture, IFOOD, which permits the flexible modeling and querying of complex data and knowledge including uncertainty with powerful retrieval capability. Tahani V. [8] has presented the techniques for fuzzy query processing in a database. Kacprzyk J. and Zadrozny S. [9] have proposed an interface that combines flexible (fuzzy) querying and data mining functionality. Marik V. et al. [10] have proposed a database expert system and its applications. Markl V. [11] has presented an encyclopedia of database systems for query processing (in relational databases). Zadeh [12] has introduced the theory of fuzzy sets and fuzzy logic, two concepts that laid the foundation of possibility theory in 1977. According to him "the theory of fuzzy sets is a step toward a rapprochement between the precision of classical mathematics and the pervasive imprecision of the real world… a rapprochement born of the incessant human quest for a better understanding of mental processes and cognition.

III. MODELING AND SIMULATION FOR QUERY PROCESSING

3.1 Class Diagram

As the UML is a very popular modeling language that deals with the modeling of huge and complex software systems. In the current scenario, the object-orientation is widely used by the software professionals. It is much more flexible towards the reusability of code and maintenance of the code in comparison of the structured design methodology.

A diagrammatic representation of query processing is represented though the popular modeling language named as Unified Modeling Language (UML), it shows the static behavior of the system, where the several properties and attributes of the system are used for complete system. There are several properties like association, aggregation; inheritances are designed in the form of sub classes. A complete process of query processing is represented here. The UML model contains the major classes like User, Parser & Translator, Query_Optimizer, Database_System and Query_Evalution_Engine. The UML class diagram is represented in Fig. 1.

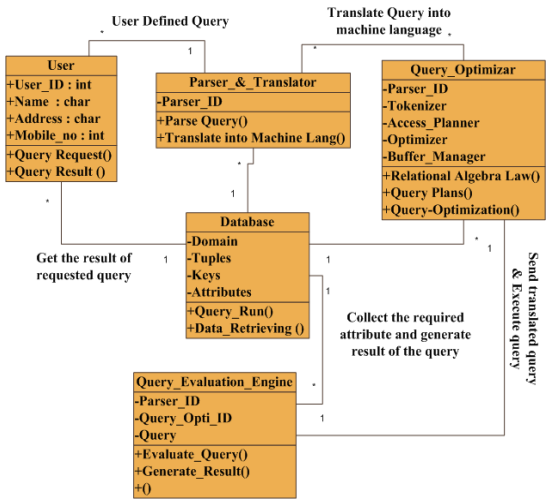

Figure 1. UML Class Diagram for Query Processing

The Database class has single associations with the classes User, Query_Evaluation_Engine, Parser_&_Translator and Query_Optimizer. The User class has single associations with the class Parser_&_Translator while the class Query_Optimizer has multiple associations with the Parser_&_Translator and single associations with the class Query_Evaluation_Engine. Therefore, according to the **Fig 1.** the user has requested any query to the database, the parser parsed and translate the user requested queries into the machine understandable form. The query optimizer gets the machine understandable language and sends to the query evaluation engine for executing the requested query through an appropriate execution plan and generates the result for the requested query and sends it to the user.

3.2 Architecture of Query Processing and Interlinking

The query processing architecture is presented in **Fig 2.** that shows the complete process of how a user requested queries are processed by the query evaluation engine and generate the result for it.

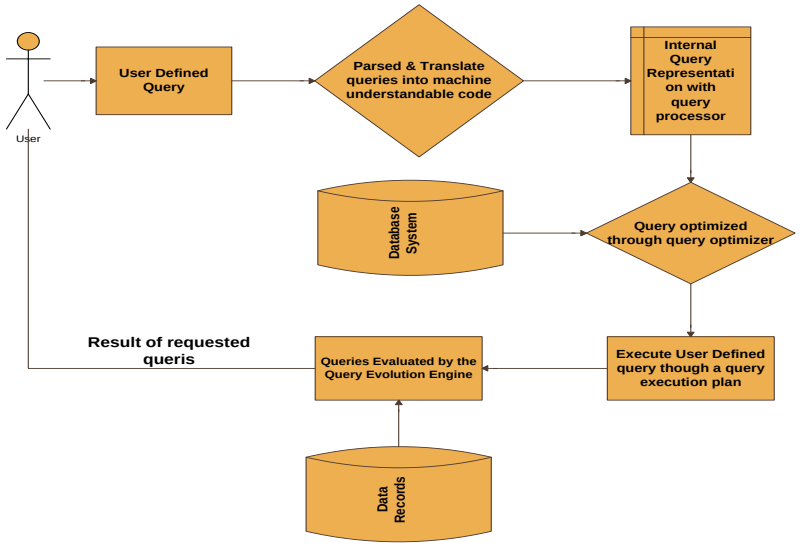

Figure 2. Query Processing Architecture

A new architecture of query processing for the fuzzy object-oriented database is presented in the **Fig 3** in which A fuzzy object-oriented database considered where the system catalog contains both the DDL and DML that assured the data description and data manipulation respectively. While the fuzzy object-oriented database has the ability to represent the vague or imprecise information in every aspect and process it for storing and fetching the desired information in an easy and faster manner. There is a tool called FSQL-to-SQL that allows the implementation of a Fuzzy Object-Oriented Database into FSQL which automatically convert the FSQL script or query in to an equivalent SQL script. Therefore, the FSQL server translates the fuzzy queries that written in the FSQL language and looking the desired information stored in the FMB. As the query translation is finished, the database management system manages the crisp data which is translated by the FSQL server and represents the resultant information for the user interface.

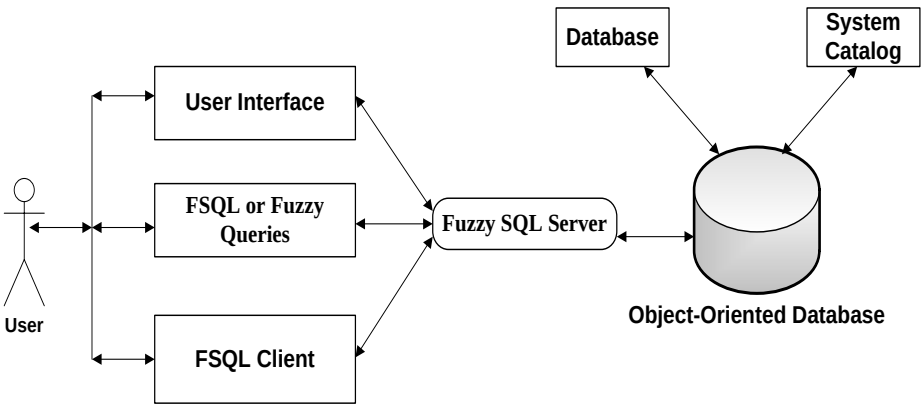

Figure 3. Query Processing Architecture of Fuzzy Queries

IV. EXPERIMENTAL STUDY & RESULTS

4.1 A Sample Relational Database

A classical or relational database is a collection of data items called information (record or data) which stored in tabular form which is managed through the database language such a SQL (structured query language) this information is also called relational schema. A relational database is designed and represented in Table 1 by the use of structured query language i.e. SQL that has several tables in with several fields that represent the brief look of all records. Some queries are performed for evaluating the performance of the designed database.

Table 1. Sample Relational Database for Patient Diagnosis

WIN-S1R08VIK25P....w - dbo.Patient | WIN-S1R08VIK25P.....Patient_History1

P_ID	Name	Doctor_ID	Doctor_Name	Gender	Age	Weight	Blood Group	Disease
1001	MR. ANURAG SH...	D0001	Dr. A.K.SACHDEV	Male	45	76	O+ve	Virul Fever
1002	MR. JAGDISH	D0002	DR. PRACHI SRI...	Male	67	67	AB+ve	Virul Fever
1003	MRS. CHANDAW...	D0003	Dr. A.K.SACHDEV	Female	78	57	AB+ve	Maleria
1004	MASTER JAINUL	D0004	DR. PRACHI SRI...	Male	23	60	A	Dangue
1005	MR. SANTOSH K...	D0005	DR. GAURAV KU...	Male	35	56	B+ve	Virul Fever
1006	BABY RADHA	D0006	Dr.ROHIT	Male	12	15	O+ve	Dangue
1007	MR. JITIN WAD...	D0007	DR. RICHA SETH	Male	69	55	AB	Dangue
1008	MRS. KIRAN	D0008	DR. HEMA VERMA	Female	25	45	AB+	Dangue
1009	MRS. USHA MIS...	D0009	Dr. ANIMA PRAS...	Female	56	78	O+	Virul Fever
1010	MISS. SUHASI	D0010	Dr. A.K.SRIVAS...	Female	32	55	O+	Typhoid fever
1011	MRS. NIRMALA	D0011	Dr.ROHIT	Female	35	46	AB+	Dangue
1012	MISS. SHAHEEN ...	D0012	Dr.AMIT KUMAR	Female	25	45	B+	Dangue
1013	MISS. ROHINI	D0013	Dr. ABHAY KRIS...	Female	23	50	B+	Typhoid fever
1014	MR. RAMESH	D0014	Dr. M.S. ZUSTI	Male	37	66	O+	Dangue
1015	MASTER ANKIT	D0015	DR. GAURAV KU...	Male	26	56	AB+	Virul Fever
1016	MR. RAM SWAR...	D0016	DR. NEETU SHU...	Male	30	63	AB+	Dangue
1017	MRS. PYARA	D0017	DR. REENA SHA...	Female	17	36	O+	Dangue
1018	MR. TULAI GAU...	D0018	DR. AYESHA	Male	18	38	B+	Dangue
1019	MRS. SHAYRA	D19	DR. PRACHI SRI...	Female	12	20	AB+	Dangue
1020	MR. SHIV DULARE	D0020	DR. ANCHAL KE...	Male	39	65	O+	Virul Fever

4.2 A Sample Fuzzy Object-Oriented Database

As discussed in the above section a fuzzy database is a database which deals with the vague or imprecise information using fuzzy logic. A fuzzy object-oriented database is an extension of a database that deals the uncertain or incomplete information with object-oriented techniques that helps to implement some object-oriented programming concepts in it. It stores and interrogates imprecise information. Therefore, a fuzzy object-oriented database is designed for the patient diagnostic system (PDS) of **"dengue fever"** with its range value and is represented in the Table 2. Some fuzzy queries are performed, for that the fuzzy query approach is based on the fuzzy logic.

Table 2. Sample Fuzzy Object-Oriented Database for Patient Diagnostic System (PDS)

	P_ID	Name	Gender	Age	Weight	Blood Group	Disease
1	1004	MASTER JAINUL	Male	23	60	A	Dangue
2	1006	BABY RADHA	Male	12	15	O+ve	Dangue
3	1007	MR. JITIN WADHWANI	Male	69	55	AB	Dangue
4	1008	MRS. KIRAN	Female	25	45	AB+	Dangue
5	1011	MRS. NIRMALA	Female	35	46	AB+	Dangue
6	1012	MISS. SHAHEEN BANO	Female	25	45	B+	Dangue
7	1014	MR. RAMESH	Male	37	66	O+	Dangue
8	1016	MR. RAM SWAROOP	Male	30	63	AB+	Dangue
9	1017	MRS. PYARA	Female	17	36	O+	Dangue
10	1018	MR. TULAI GAUTAM	Male	18	38	B+	Dangue
11	1019	MRS. SHAYRA	Female	12	20	AB+	Dangue

Therefore, in the fuzzy logic there is a membership function that is a tool to express fuzzy attributes. There are some membership functions for checking whether the patient is infected by the dengue fever or not. The Table 3 represents the member function for the Dengue Antibody lgG along with the linguistic terms and

their associated range. The input value has the three linguistic terms: Negative (<18.0), Equivocal (18.0-22.0), and Positive (>22.0). One patient can have only one type linguistic term at a time. Therefore, if the Dengue Antibody–lgG count is greater than the 22.0 then it makes sure that the patient is suffer from dengue.

Table 3. Memberships function for Dengue Antibody lgGand linguistic terms and associated ranges

Input Field	Linguistic Terms	Range Value
Dengue Antibody lgG	Negative	<18.0
	Equivocal	18.0-22.0
	Positive	>22.0

As getting the positive result for the dengue infection in the patient, a test is conducted for calculating the RBC's and WBC's count in the blood. Therefore, the membership function for the RBC/WBC count is represented in the Table 4 along with the linguistic variables and associated range.

Table 4. Membership function for RBC/WBC count and linguistic terms and associated ranges

Input Field	Linguistic Terms	Range Value
Red Cell Count/White Cell Count	Normal	4.0-6.0
		5.0-10.0
	Mid	3.0-5.0
		4.0-9.0
	Low	2.0-4.0
		3.0-8.0
	Very Low	1.8-3.8
		2.0-7.0

V. FUZZY QUERIES

There are some fuzzy queries are performed below for testing the performance of the designed fuzzy object-oriented database.

5.1 Sample Query 1:

Show all the persons who is young (Age between 16-40) AND dengue positive (with Dengue Antibody lgG>22.0). The query is mentioned FSQL language is as follows:

SELECT *from [Patient new].dbo.Patient_History1 WHERE young FEQ $ Age THOLD 16-40 AND Disease FGT $ Dengue THOD $ >22.0 .

5.2 Sample Query 2:

Show all the persons who is old (Age >40) and having Red Cell Count is very Low (With Red Cell Count between 1.8-3.8). This Query is mentioned in FSQL Language is as follows:

SELECT *from [Patient new].dbo.Patient_History1 WHERE Old FEQ $ Age THOLD >40 AND Disease FGT $ Dengue THOD $ >22.0 AND RBC_COUNT FEQ $ RedCellCount THOLD 1.8-3.8 .

VI. CONCLUSION

From the above work, it is concluded that the query processing is one of the powerful way to query the desired information from the designed fuzzy object-oriented database by using the linguistic expressions to improve the quality of the select command. A fuzzy interpreter transform the user defined fuzzy query into the SQL one and represent the resultant information by linking the FSQL-to-SQL. The present work further can be extended in different fields of medical sciences, train tracking system (train arrival and departure), Web based applications with fuzzy data etc.

ACKNOWLEDGEMENTS

Authors are grateful to the Vice-Chancellor (Prof. P.K. Bharti), Maharishi University of Information Technology Lucknow for providing the excellent facility in the computing lab of Maharishi university of Information Technology, Lucknow, India. Thanks are also due to University Grant Commission, India for support to the University.

REFERENCES

[1] Uzochukwu C, Oghenekaro and Linda U (2015): An application of flexible query interface to relational databases, IOSR Journal of Computer Engineering (IOSR-JCE), Vol. 17, Issue 5, pp. 1-07.

[2] Li Y. Handbook of Research on Innovative Database Query Processing Techniques**, (**IGI Global 2015).

[3] Sonia and KumariS. (2013): Fuzzy Object-Oriented Database versus FRDB for Uncertainty Management, International Journal of Computer Application, Ney York, Vol. 74, No. 17.

[4] Kumari S. and Sonia (2013): Fuzzy RDBMS Design: SQL Add-on, International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 2, No 5, pp. 1862-1865.
[5] Raipurkar A. and Bamnote G.R. (2013): Query Processing in Distributed Database through Data Distribution, International Journal of advanced Research in computer and communication Engineering, Vol. 2, Issue 2, pp. 1135-1139.
[6] Touzi A. G. and Hassine M. A. B. (2009): New Architecture of Fuzzy Database Management System, The International Arab Journal of Information Technology, Vol. 6, No. 3, pp. 213-219.
[7] Koyuncu M. and Yazici A. (2003): IFOOD: An Intelligent Fuzzy Object-Oriented Database Architecture, IEEE Transaction on Knowledge and Data Engineering, Vol. 15, No. 5.
[8] Tahani V. (2002): A Conceptual Framework for Fuzzy Query Processing-A Step Toward Very Intelligent Database Systems, Information Processing & Management, Vol. 13, Issue 5, pp. 289-303.
[9] Kacprzyk J. andZadroznyS. (2000): On a Fuzzy Querying and Data Mining Interface", KYBERNETIKA, Vol. 36, No. 6, pp. 657-670.
[10] Marik V., Lazansky J. and Wagner R. R. (1993): Database and Expert Systems Applications, Proceedings in 4th International Conference, DEXA 93 Prague, Czech Republic, September 6-8.
[11] Markl V.: Query Processing (in Relational Databases), Encyclopedia of Database Systems, Springer US, pp. 2288-2293.
[12] Zadeh L. A. "Fuzzy Sets, Information and control", (1965), Vol. 8, pp. 338-353.