International Journal of Technology and Engineering System (IJTES)
Vol 7. No.5 2015 Pp. 454-459
©gopalax Journals, Singapore
available at : www.ijcns.com
ISSN: 0976-1345

Creation of Dynamic Query Form for Manipulating and Retrieving Results from the Database

Sathishkumar M
M.E., Software Engineering Department of Computer Science and Engineering
Anna University, College of Engineering, Guindy, Chennai, India
sathishkblr@gmail.com

ABSTRACT

Real-world databases contain many hundreds of relations and attributes. Traditional query forms are not able to satisfy various ad-hoc queries from users on those databases. With the proposed DQF, the Database Query Form interface, it will be able to dynamically generate query forms. A user can also fill the query form and submit queries to view the query results. In this way, a Query Form could be dynamically refined till the user satisfies with the query results at each iteration. The ranking of Form components are based on the captured user preferences. It focuses on the projection and selection components of the query form. Ad-hoc join is handled by our dynamic query form. Our dynamic query form is able to perform INSERT, UPDATE, DELETE, ALTER, SELECT and CREATE operations which are implemented using dropdown boxes and textboxes in the user interface of the query form. The F-measure is used for measuring the goodness of a query form.

Index Terms – Query form, user interaction, Dynamic Query form generation.

I. Introduction

Traditional query forms are designed and predefined by Database Administrators (DBA). In Healthcare, the databases have over hundreds of entities for chemical and biological data components. Healthcare Management System Database becomes very huge and difficult. Therefore, it is difficult to design a set of static query forms to satisfy various ad-hoc database queries on those complex databases. The creation of Dynamic queries totally depends on user's manual editing.

1.1 Our Approach

Dynamic Query Form (DQF), a query interface which is capable of dynamically generating query forms for users. The character of DQF is to capture user interests during user interactions and to adapt the query form frequently. It consists of two types of user influence: Query Form Improvement and Query Implement. It starts with a basic query form which contains very few main attributes of the database. The basic query form is then improved iteratively via the interactions between the user and our system until the user is satisfied with the query results. Fig 1 shows the work flow of DQF.

1.2 Contribution

- With the planned a dynamic query form system which generates the query forms according to the user's wishing for something to happen at run time. The system makes a solution for the query interface in large and complex databases.

- F-measure to calculate the beneficial of a query form. F-measure is a particular standard of measurement to calculate query results.

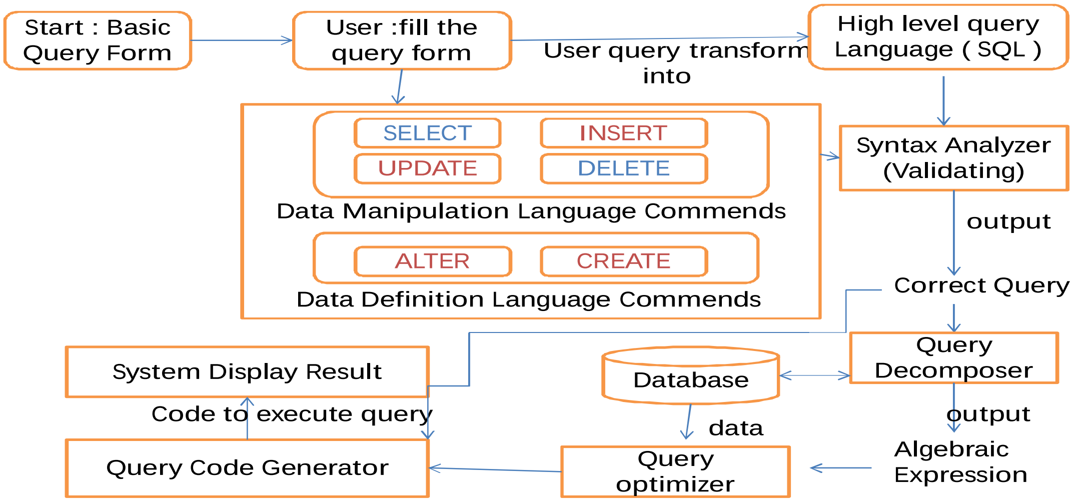


Fig 1 Flowchart of dynamic Query Form

II. Related Works

2.1. Query-By-Example

M. M. Zloof.[1] E. Chu, A. Baid, X. Chai, A. Doan, and J. F. Naughton [7] Liang Tang, Tao Li, Yexi Jiang, and Zhiyuan Chen [9] has presented an overview of information Retrieval, data manipulation, and data definition by the Query-by-Example(QBE) language. User has to conform to the structure of the Query table; the Query-By-Example user may enter any expression as an entry as long as it is syntactically correct. In this paper, described Query-By-Example and Query Form are two most widely used database querying interfaces. At present, query forms have been effective used in most real-world trade or based on science information systems. Present studies and works mainly activity on how to generate the query forms. Query-by-Example is the name of both a data-manipulation language and an early database system that included this language. Query By Example has a two dimensional syntax: Queries looks like tables. A query in a one-dimensional language (for example, SQL) can be written in one line. A two dimensional language depends on two dimensions for its expression. Instead of giving a procedure for obtaining the wanted answer, the user gives an example of what is want. The systems make more common this example to compute the answer to the query.

2.2. Customized Query Form (CQF)

M. Jayapandian and H. V. Jagadish[2] M. Jayapandian and H. V. Jagadish [3] Liang Tang, Tao Li, Yexi Jiang, and Zhiyuan Chen [9]drawed existing customized forms database clients and tools make great efforts to help developers design and generate the Query Form. The problem of those tools is that, they are provided for the worthy of developers who are well known with their databases, not for the person who uses a particular product.

2.3. Automatic Static Query Form (SQF)

M. Jayapandian and H. V. Jagadish [2], M. Jayapandian and H. V. Jagadish [4], Liang Tang, Tao Li, Yexi Jiang, and Zhiyuan Chen [9] in these paper proposed automatic approaches to generate the database query forms without user participation. data driven method. It first finds a set of data property, which are most likely queried based on the database structure and data objects. Then, the query forms are generated based on the selected attributes is a work based driven method. It applies clustering algorithm on historical queries to find the typical of a class of queries. The query forms are then created based on those representative queries. One problem of this proposal is that, if the database structure is large and complex, user queries could be quite widely varied. In that case, even if we create lots of query forms, there are still user queries that cannot be satisfied by any one of query forms structure. Another problem is that, when we create a large number of query forms, how to let users find an correct query forms.

2.4. Query Refinement

Meenu Joy Bhruguram T M [5] in this paper, they present Query refinement is a common practical technique used by most information retrieval Systems. Database query form, a database query is a structured relational query. In this paper, Dynamic faceted search engines are similar to dynamic query forms if only consider Row components in a query. Selections, a database query form has other important components, such as Projection components. Column components control the output of the query form and cannot be removed. Structured Query Language queries as items in the collaborative compressing approach, and recommend similar queries to related users. Dynamic Data Entry Form develops an adaptive forms system for information entry, which can be dynamically changed according to the previous data input by the user with the proposed database query forms instead of data-entry forms.

2.5. Active Feature Probing

S. Zhu, T. Li, Z. Chen, D. Wang, and Y. Gong [6] develop the active featuring probing technique for automatically generating clarification questions to provide appropriate recommendations to users in database search option. Different from their work which focuses on finding the appropriate questions to ask the user, Dynamic Query Form aims to select appropriate query components interface, which are usually generated by the system according to the database schema. In this paper, focus on the projection and selection components of a query form.

2.6. Problem statement

Existing customized forms database clients and tools make great efforts to help developers design and generate the Query Form. The common problem of that tool is that, they are provided for the well known developers who are familiar with their databases, not for normal users. The effectiveness of a manually designed forms-based interface largely depends on the developer’s understanding and estimation of its users needs. In this paper, introduce dynamic query form end user familiar with their database. It first finds a set of data attributes, which are common likely queried based on the database structure and data objects. Then, the query forms are created based on the selected components is a work based-driven method. The queries forms are then generated based on those represent queries. One problem of these approaches is that, if the database structure is complex, user queries could be quite varied. In that case, even if we generate lots of query forms means, there are still user queries that cannot be satisfied by any one of query forms. Second problem is that, when we generate a large number of query forms, how to let users find a write and appetite query form would be challenging. In this paper, proposed combines keyword search with query form generation.

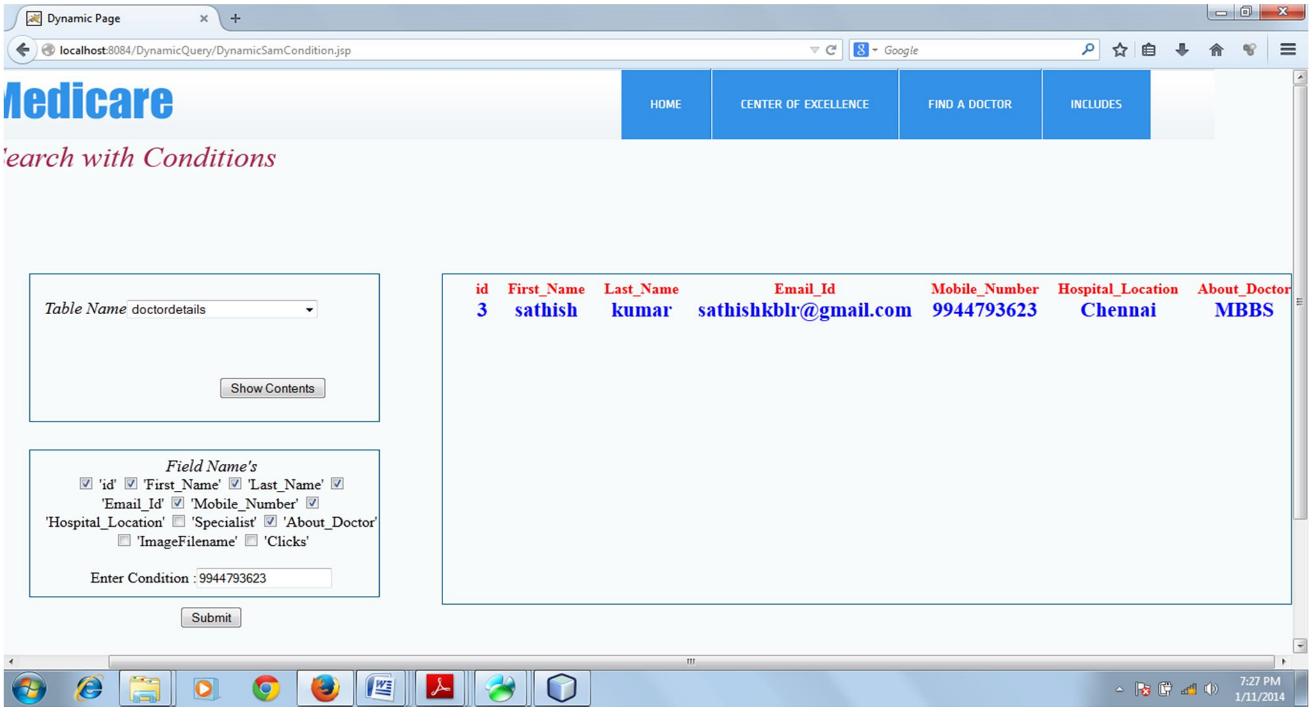
III. Evaluations

3.1 System Implemented and Experimental Setup

We implemented the dynamic query forms as a web based system using Java Development Kit 1.6 with JSP. The dynamic web interface for the query forms used open-source java script library jQuery 1.4. We used Navigate8.0 Litefor MySQL as the database engine. All experiments were run using a machine with Intel(R) Core(TM) i3 CPU @2.13GHz, 3.00GB main memory, and running on Windows 7.Fig 2 shows a system prototype.

Datasets: Hospital management system database used in our experiments.

User Study Setup: In the phase took 10 tasks listed in the table 1.Each participants used all three form generation approaches to form queries.



earch with Conditions

Table Name doctordetails

Show Contents

Field Name's

☑ 'id' ☑ 'First_Name' ☑ 'Last_Name' ☑ 'Email_Id' ☑ 'Mobile_Number' ☑ 'Hospital_Location' □ 'Specialist' ☑ 'About_Doctor' □ 'ImageFilename' □ 'Clicks'

Enter Condition : 9944793623

Submit

id	First_Name	Last_Name	Email_Id	Mobile_Number	Hospital_Location	About_Doctor
3	sathish	kumar	sathishkblr@gmail.com	9944793623	Chennai	MBBS

Fig 2 Screenshots of web based Dynamic Query Form

Table 1 Query Tasks

Task No	**Task Query**	**Static Query Form** (No of actions needed to getting results)	**Dynamic Query Form** (No of actions needed to getting results)
1.	SELECT first name, last name FROM doctor details	18	7

2.	SELECT first name, last name
	FROM doctor details
	WHERE first name='Dr. Sathish'
		41	16

3.	SELECT d.first name, d.last name,l.location
	FROM doctor details INNER JOIN location
		58	4

4.	SELECT d.first name, l.location
	FROM doctor details INNER JOIN location
	WHERE d.first name = 'Dr.Sathish'
		77	15

5.	SELECT Avg(Salary) from salary	38	13
6.	SELECT MAX(salary) , MIN(salary) from salary	38	13
7.	SELECT ucase(First_Name), lcase(First_Name) FROM doctordetails	49	14
8.	SELECT * FROM doctordetails where First_Name='Arun' AND Last_Name='M'	71	20
9.	SELECT * FROM doctordetails where First_Name='Arun' OR	70	20

	Last_Name='M'		
10.	SELECT * FROM doctordetails where id in(1,2)	40	15
11.	SELECT * FROM doctordetails where id Not in(1,2)	41	15
12.	CREATE DATABASE DynamicQuery	20	4
13.	CREATE table hospital_bill	32	13
14.	INSERT into location values(6,'chennai')	45	10
15.	DELETE from doctor details WHERE first name = 'Dr.Sathish'	56	15
16.	ALTER table bill ADD age(10)	37	19
17.	ALTER table bill DROP age	34	17
18.	ALTER table bill MODIFY name varchar(10)	48	20

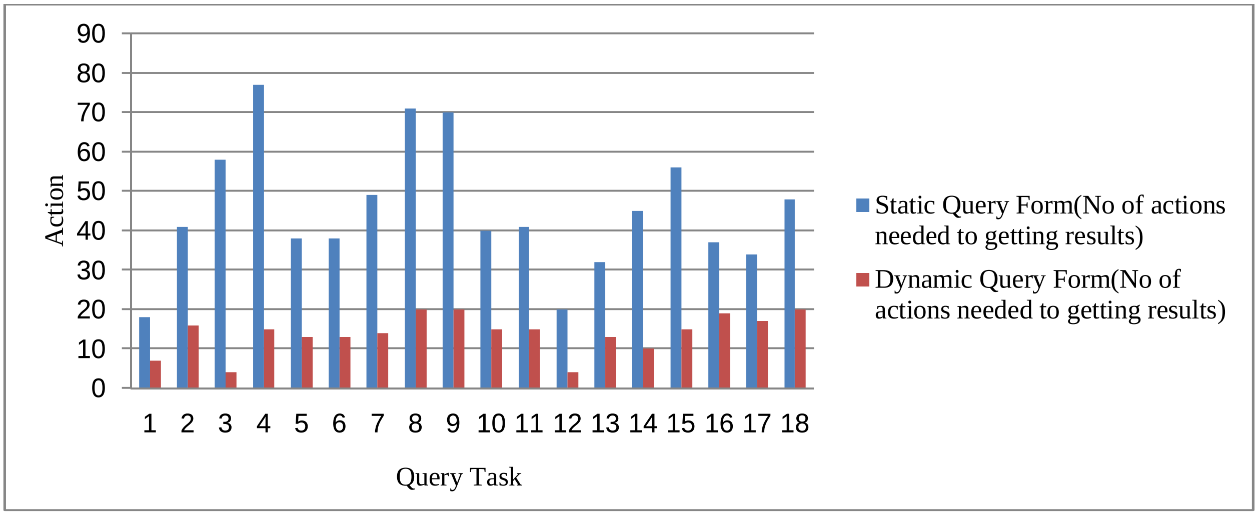


Figure : 1 Compare SQF and DQF based on actions

Algorithm

Key Value Pair Algorithm [7] .To implements this strategy, we use two inverted indexes, one on the data set and the other on the set of query forms. The first index, called data key and data value Index, takes in a term and returns a set of <tuple-id, table> pairs. tuple id is the constraints of the tuple, and table is the name of the table hold with the tuple. The second index, called Query Form Index, takes in a term and returns a set of query form-ids, or identifiers of the forms containing the term.

Search with Condition

Input: A user search quer

Output: Dynamic Query Form shown results based on user keyword query.

Algorithm

- ✓ Add each table to form terms (database structure)
- ✓ For each qi € Q
- ✓ If user query (qi) returns <table,table-id>pairs
- ✓ Show results based on qi
- ✓ Add qi to form terms

- ✓ Return form terms

IV. Conclusion

Dynamic Query form is widely regarded as the most easy to use or understand method. In this paper, we implement an query form generation approach which helps users to dynamically generate query forms. As future work, we will study how our approach can be used NoSQL(Not Only SQL). we plan to develop capture the user's want for the queries apart from click feedback. We can include a text box for users to input some keyword queries.

V. References

1. M. M. Zloof. Query-by-example: the invocation and definition of tables and forms. In *Proceedings of VLDB*, pages 1–14, Framingham, Massachusetts, USA, September 1975.
2. [MJayapandian and H. V. Jagadish. Automated creation of a forms-based database query interface. In *Proceedings of the VLDB Endowment*, pages 695–709, August 2008.
3. M. Jayapandian and H. V. Jagadish. Expressive query specification through form customization. In *Proceedings of International Conference on Extending Database Technology (EDBT)*, pages 416–427, Nantes, France, March 2008.
4. M. Jayapandian and H. V. Jagadish. Automating the design and construction of query forms. *IEEE TKDE*, 21(10):1389– 1402, 2009.
5. Meenu Joy Bhruguram T M. Dynamic Query Form with query Refinement and Database encryption.*In Proceeding of IOSR Journal of Computer Engineering (IOSRJCE),PP 154-159:* Sep-Oct. 2014.
6. S. Zhu, T. Li, Z. Chen, D. Wang, and Y. Gong. Dynamic active probing of helpdesk databases. *Proc. VLDB Endow.*, 1(1):748–760, Aug. 2008.
7. E. Chu, A. Baid, X. Chai, A. Doan, and J. F. Naughton. Combining keyword search and forms for ad hoc querying of databases. In *Proceedings of ACM SIGMOD Conference*, pages 349–360, Providence, Rhode Island, USA, June 2009.
8. Rakesh Agrawal,Ralf Rantzau,Evimaria Terzi.Contextsensitive ranking. *SIGMOD 2006,* June 27–29, 2006, Chicago, Illinois, USA. Copyright 2006 ACM 1595932569/06/0006.