

A Novel Architecture for Designing a Multimodal User Interface to Relational Databases

D. Mabuni¹, Dr. G.A. Ramachandra²

1 Research Scholar, Rayalaseema University, Kurnool, Andhra Pradesh, India.

2 Professor, Department of Computer Science & Technology, Sri Krishnadevaraya University, Anantapuramu, Andhra Pradesh, India

Corresponding Author: D. Mabuni

ABSTRACT: A user interface of a relational database acts like a link between the user and the relational database. It allows the users to retrieve data from the database. As the number of information workers is increasing day-by-day, there is a need to develop technologies to access information from databases easily and effectively. Structured Query Language (SQL) is a very standard interface to databases since many years. But, this formal query language interface forces the users to have cognitive skills that the typical users do not possess. Many alternative interfaces to databases were developed to access information from database such as Form-based interfaces, menu-based interfaces, graphical interfaces and natural language interfaces and so on. Each of these interfaces has its own advantages and disadvantages. Users need an interface to database that enables them to retrieve a variety of information they need from the database in a more natural way. This paper presents a novel architecture for designing a Multimodal user Interface to relational databases.

KEYWORDS: Forms, graphics ,menus, multimodal, natural language, relational database, user interface.

Date of Submission: 16-02-2018

Date of acceptance: 03-03-2018

I. Introduction

A user interface is an important part of a computer and its software that users can see, touch, talk to, hear, or otherwise direct and understand. It acts like a link between the user and the computer. It allows the user and the computer to communicate with each other. A user interface of any software system has mainly two components. They are Input and Output. Input is concerned with how an user informs his or her needs or desires to the computer. Output is concerned with how the computer shows the results of its computations to the user. A good user interface design will provide a mixture of well-designed input and output procedures that satisfy the users' needs, capabilities, and limitations in the most effective way.

The success of any software system depends heavily on its user interface. The user interface of any software system must have an important characteristic called usability. A usable user interface allows the users to access information and perform various tasks very easily and effectively. A user interface should provide a natural communication between a user and software system in the language that users know. It should not force the users to remember some thing and interact. The interface style must be uniform and most of the errors are to be prevented.

A user interface of a software system controls the way how data and instructions are entered (input) into the computer and how the computer presents information on the screen (output). Information is playing a major role in human lives since many years. In order to make a decision, a person needs a variety of information. Most of the organizations are storing their data in the databases to facilitate their users to easily retrieve and manipulate data. Databases contain a collection of related data, stored in an organized way to model some aspect of the real world, sometimes called the universe of discourse. Nowadays, databases have become one of the major sources of information for most of the users. With the advances in Information and Communications Technologies (ICT) such as Computers, Laptops, Smart phones, Internet, Personal Digital Assistants (PDAs), information can be accessed anywhere and at anytime. As the number of information workers is increasing day-by-day, there is a need to develop technologies to access information from databases easily and effectively.

II. User Interfaces To Databases

A user interface to a database provides facilities to the users to retrieve desired data from the database. Practically, every database needs a user interface to retrieve data from it. Nowadays, many people try to get information from databases for their needs. There is a need to have a good interface to submit requests and to get the information from the database. Most of the Database Management Systems (DBMSs) provide a query

language interface such as Structured Query Language (SQL) to retrieve information from the database. But, this formal query language interface forces the users to have cognitive skills that the typical users do not possess. The cognitive load demanded by existing query languages (such as SQL) during data retrieval includes the following.

- a. A good understanding of database schema design.
- b. A more or less extensive knowledge of database systems, database query language syntax and structure.
- c. The ability to manually or mentally create message graphs to join tables and to express desired database requests.

Furthermore, the complexity of conceptual structures i.e. schemas of the data that has been made available in the database is also increasing. This affects the users in formulating complex queries to retrieve or manipulate data in the database. Because of the increased proliferation of information and database usage by the many organizations, the potential user base of the database systems has increased. In turn, the importance of a good user interface to the database is increased.

Many alternative interfaces to databases were developed to access information from database such as Command-based interfaces, Form-based interfaces, menu-based interfaces, graphical interfaces and natural language interfaces and so on. Each of these interfaces has its own advantages and disadvantages. Users need an interface to database that enables them to retrieve a variety of information they need from the database in a more natural way.

III. Multimodal Interface To Relational Database

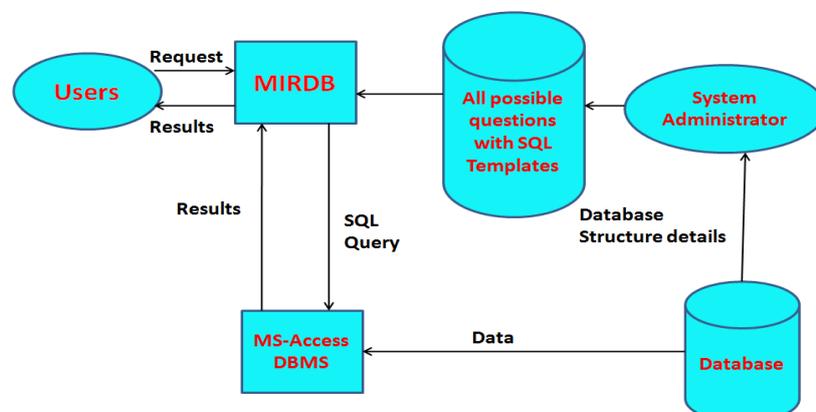
A Multimodal interface to Relational Database (MIRDB) is a user interface that allows the users to specify their requests for information from the Relational database more accurately using graphics, menus, forms and natural language. The interface combines the strengths of all modalities such as graphics, menus, forms and natural language.

Every user interface should inform the user about the possibilities, functionalities and limits of the system. The degree in which this purpose is fulfilled is called interface's usability. An interface having high usability makes it easy for users to access most or all of the functionality provided by the system. The following are the some important usability principles.

- i) A simple and natural dialogue between the user and system.
- ii) Informs in the users' language
- iii) Users' memory load must be reduced
- iv) Maintain consistency
- v) Feedback must be provided to the user
- vi) Exits should be marked clearly
- vii) Good error messages must be provided
- viii) Prevent most of the errors

IV. Proposed Architecture For Designing A Multimodal Interface To A Relational Database

The following figure shows the proposed Architecture for Multimodal interface to a Relational Database (MIRDB).

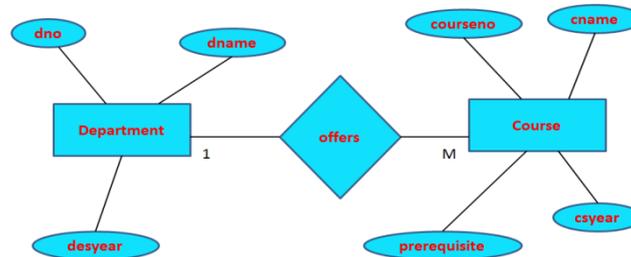


In this architecture for Multimodal interface to a Relational Database (MIRDB), the following activities are performed.

- i. The system administrator will analyze the Database Structure i.e., Tables and their relationships in the Database and identifies all the possible questions for each table and relationship between tables.
- ii. For each possible question, a SQL template is attached
- iii. All the table names of all the tables in the database are displayed as graphic objects.
- iv. The user can perform the following steps to get information from the database.
 - a) Select the Table by clicking on the corresponding graphic object. The system displays all the possible questions related to the table as the menu of questions.
 - b) The user selects the appropriate question by clicking on it. The system displays the selected question in a Form so that the user can modify the value(s).
 - c) The user can modify the value(s) in the question.
 - d) The user clicks on submit button. The system displays the result to the user.

4.1. Example Database:

The following is the ER Diagram showing the structure of a portion of a University database.



The above portion of a University database example has been taken for the purpose of describing how the proposed architecture can be used to design a multimodal interface to a relational database.

As per the ER diagram shown above, the University database has two tables, Department and Course. Each department has unique number (dno), unique name (dname) and year of establishment (desyear). A department offers a number of courses, each of which has unique number (courseno), unique name (cname), starting year (cyear) and prerequisite.

The following are the resulting tables after mapping the above ER schema into relational database schema.

Tables:

- a. department(dno, dname, desyear)
- b. course(courseno, cname, prerequisite, cyear, dno)

keys:

- The primary key of department table is dno. No foreign keys in department table.
- The primary key of course table is courseno. The foreign key is dno and refers to dno of department table.

4.2. Screen Designs:

Screen 1:

Displays all the table names as Graphic Objects and enables the users to click on any one of them. (Graphics Interface)

Data Retrieval System for University Database

Please click on any one of the boxes given below to retrieve data from the University Database. University Database has data related to the Departments and Courses.

**Data related to the
Departments
only**

**Data related to the
Courses only**

**Data related to
both Departments
and Courses**

If the user clicks on the first box i.e., "Data related to the Departments only", then the following screen will appear.

Screen 2:

Displays all the possible questions related to the selected graphic object in the form of a menu and enables the users to select any one of them. (menu concept is used and questions are in natural language)

Please select one of the questions given below by clicking on it.

Possible Questions that can be answered by the system
What are the departments established in the University?
What are the names of the departments established in the University?
What are the numbers of the departments established in the University?
How many departments were established in the University?
What is the name of the department whose number is 1? (dept. no. can be edited)
What is the number of the department whose name is "history"? (dept. name can be edited)
What is the year of establishment of the department whose number is 1? (dept. no. can be edited)
What is the year of establishment of the department whose name is "history"? (dept. name can be edited)
What are the departments established in the year 1998? (year can be edited)
What are the departments established between the years 1998 and 2001? (years can be edited)

Home

If the user clicks on the first question, then the following screen will appear.

Screen 3:

Displays the selected question in a form and allows the users to edit values if desired. (forms concept is used)

Selected Question

What are the departments established in the University?

Submit

Back**Home**

If user presses Back button, then screen 2 appears. If the user presses Home button, the screen 1 appears. If the user presses Submit button, then the following screen 4 will appear.

Screen 4:

Displays the results (answer) for the selected question.

dno	dname	desyear
1	history	1998
2	computer science	2000
3	english	2002
4	philosophy	2002
5	chemistry	2005

Back Home

If the user clicks on the fifth question in the screen 2, then the following screen 3 will appear.

Screen 3:

Displays the selected question and allows the users to edit values if desired. (forms concept is used)

Selected Question

What is the name of the department whose number is 1?

Select dept number 1

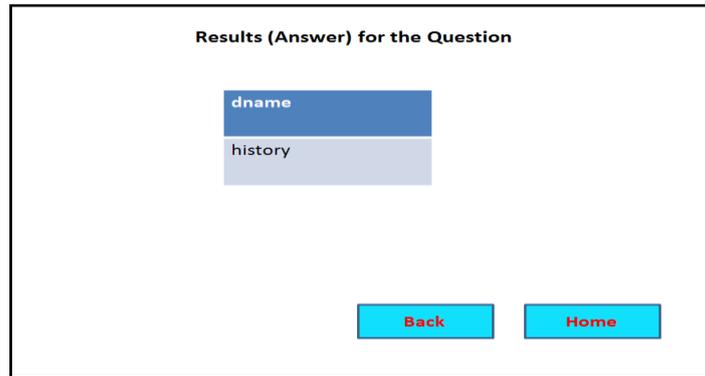
Submit

Back Home

If user presses Back button, then screen 2 appears. If the user presses Home button, the screen 1 appears. If the user presses Submit button, then the following screen 4 will appear.

Screen 4:

Displays the results (answer) for the selected question.



The above multimodal interface to a relational database combined the strengths of all modalities such as graphics, menus, forms and natural language.

V. Implementation And Results

The system administrator will analyze the relational database for which a Multimodal Interface is to be designed and identify all the possible questions that can be asked by the user for the data from the database. The system administrator will create SQL template for each possible question and store in the database in the table "sqltemplates".

The following is the Algorithm for processing the question asked by the user.

Algorithm Process_User_Question:

1. Accept the question selected by the user, say Q
2. Accept any values related to the question that are specified by the user
3. Retrieve the corresponding SQL Template from the table "sqltemplates", say T
4. Populate the SQL Template T with suitable values to create executable SQL query, say R
5. Send SQL query R to the DBMS (e.g. MS-Access) for execution
6. Receive data from the DBMS
7. Display the results on the screen

The above algorithm has been implemented using Visual Basic 6.0 as front-end and MS-Access as back-end and tested.

Results:

The MIRDB allowed the users to retrieve all the data that is available in a relational database. The users could retrieve desired data 100% accurately without any errors.

6. CONCLUSION

The following are the Achievements and Advantages of Multimodal interface to a Relational Database (MIRDB).

- Multimodal Interface to a Relational Database (MIRDB) combines all the modalities such as graphics, forms, menus and natural language to develop effective and more reliable interface to a relational database.
- The user need not learn the syntax of any command language.
- The user need not know the structure of the database from which he/she wants to retrieve information.
- The user need not perform complicated actions on the graphics.
- The user need not fill many forms.
- The user need not select many menu options.
- The user need not worry about the linguistic capabilities of the system.
- The MIRDB allows the users to retrieve all the data that is available in a relational database. The users can retrieve desired data 100% accurately without any errors.
- The MIRDB satisfies most of the usability principles such as A simple and natural dialogue between the user and system, Informs in the users' language, Users' memory load must be reduced, Maintain consistency, Exits should be marked clearly, Prevent most of the errors etc.

References

- [1]. Akmal B. Chaudhri, "User Interfaces for Database Systems," <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.29.4949&rep=rep1&type=pdf>
- [2]. Jon Olav Hauglid, "User Interfaces for Accessing Information in Digital Repositories," https://brage.bibsys.no/xmlui/bitstream/handle/11250/249971/125230_FULLTEXT01.pdf?isAllowed=y&sequence=1
- [3]. I. Androutsopoulos, G.D. Ritchie, P. Thanisch, "Natural Language Interfaces to Databases – An Introduction," https://arxiv.org/PS_cache/cmp-lg/pdf/9503/9503016v2.pdf
- [4]. D. Mabuni, Dr. G.A. Ramachandra, "User Interfaces to Databases - A Survey," Volume 8, No. 9, November-December 2017, International Journal of Advanced Research in Computer Science (IJARCS), pp. 710-713.
- [5]. Shamika S. Mule, Rajendra D. Gawali, "Runtime Database Query Recommender For Dynamic Query Forms," Vol. 4, Issue 12, December 2015, International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET), pp. 12127-12134.

International Journal of Engineering Science Invention (IJESI) is UGC approved Journal with Sl. No. 3822. Journal no. 43302.

D. Mabuni "A Novel Architecture for Designing a Multimodal User Interface to Relational Databases" International Journal of Engineering Science Invention (IJESI), vol. 07, no. 03, 2018, pp 01-07.