International Journal of Research in Social Sciences
Vol. 9, Issue 6, June - 2019,
ISSN: 2249-2496 Impact Factor: 7.081
Journal Homepage: http://www.ijmra.us, Email: editorijmie@gmail.com
Double-Blind Peer Reviewed Refereed Open Access International Journal - Included in the International Serial Directories Indexed & Listed at: Ulrich's Periodicals Directory ©, U.S.A., Open J-Gate as well as in Cabell's Directories of Publishing Opportunities, U.S.A

Database System and Smart Usage Model: Critical Research

Amit Khare[1], Dr. K P Yadav[2]

Department of Computer Science and Engineering

[1,2]Shri Venkateshwara University, Gajraula (Amroha), U.P. India

Abstract- A database management system (DBMS) interface is a user interface that allows queries to be entered into a database without the use of the query language itself. A DBMS interface may be a Web client, a local application running on a desktop or even a smartphone device. Relationships may be of different forms, too. To find out more about E-R Diagrams, click the button. A database management system stores data and uses query language, such as SQL, to respond to queries. A DBMS interface allows data to be queried without the need to use the query language, which can be complicated. The typical way of doing this is to create some form showing what kinds of queries users can make. Web-based ways are becoming more prevalent with MySQL's growth, but local mobile applications have been the conventional way to do so. Mobile applications can be created, too. These interfaces provide a more friendly way to access data, rather than simply using the command line. To identify the Natural Language instead of SQL has prompted the development of new type of processing called Natural language Interface to DatabaseTo analysis the intelligent databases systems are the systems that manage information in a natural way, making that information easy to store, access and use.

Keywords: Web client, database management system, MySQL's, Relationships

1. Introduction

A user interface is a connection between a human being and a computer or system that allows people to communicate with the device or system (e.g., share information with it). An interface is

a common boundary or connection between two dissimilar objects, devices, or systems that move information through. The relationship can either be physical or logical.

For example, transportation vehicle user interfaces (e.g., vehicles, locomotives, and aircraft) usually consist of windows1, one or more instrument panels, and a range of controls that are controlled with the hands and feet. Increasingly, instrument panels contain one or more electronic display boards similar to those used for computers.

A computer user interface usually consists of a display screen, a keyboard, and a mouse. The display system has two specific types of user interfaces: the command line interface (CLI), which contains text only, and the graphical user interface (GUI), which also includes images (for example, windows, icons and menus). Most GUIs use a desktop metaphor that resembles a desktop that can be moved around, resized and otherwise manipulated with folders, files, pictures etc. Web browser-based user interfaces have become ever more popular, however. Sound also plays a role on personal computers within the user interfaces. For several years it has been common for a simple beeping sound to be produced in certain cases, for example in the event of a user error. Software is widely available that can convert text to spoken words, and a lot of improvements in speech recognition technology have been seen in recent years that allow computers to understand spoken commands.

There has also been considerable progress on tactile user interfaces which provide users with feedback. One example is devices that consist of a fine diameter rod matrix whose heights can be modified to reflect visual images. These devices are particularly useful to people with vision impairments.

A virtual reality user interface is one that simulates nature along with realistic sound, vibration, temperature, smells, etc., for example, a wrap-around computer screen or special goggles. This form of interface has become increasingly sophisticated and its use has evolved rapidly for a wide range of applications, from sports to technical training (e.g., surgeons and aircraft pilots).

2. DATABASE MANAGEMENT SYSTEM

A database management system is a software system designed to store and manages a large volume of data so that multiple users can access and manipulate data in a consistent and secure way. A database system along with one or more accounts is a information management system. The three key aspects of the DBMS are: continuity if a program modifies the data, the modifications will continue long after the program has ended; sharing – more than one system can access the data at the same time; and reliability the data will stay accurate despite the hardware and software failures.

A DBMS usually includes the different facilities such as Data Definition Language (DDL), Data Manipulation Language (DML), and other frameworks for managed database access. DDL enables users to describe the database, determine data types, data structures and data constraints to be stored in the database, and also to translate the schema written in a source language into the schema of the object. DML allows users to insert, update, delete and retrieve data from the database, and provides general query facilities through query language as well.

The DBMS provides controlled access mechanism (software) to the database such as security system to prevent unauthorized users from attempting to access the database, competitiveness control system to allow shared access to the database and recovery system to a database to restore a previous consistent state due to hardware or software failure. DBMSs are highly complex and sophisticated pieces of software designed to provide the types of functions and services, including data storage , retrieval and updating, user-accessible catalogues, transaction support, competitiveness control services, recovery services , data communication authorization services, integrity services , data independence and utility services.

The DBMS should be configured with the various interrelated components to provide these services including query processor, database manager, file manager, DML processor, DDL processor, and catalog manager.

International Journal of Research in Social Sciences
Vol. 9, Issue 6, June - 2019,
ISSN: 2249-2496 Impact Factor: 7.081
Journal Homepage: http://www.ijmra.us, Email: editorijmie@gmail.com
Double-Blind Peer Reviewed Refereed Open Access International Journal - Included in the International Serial Directories Indexed & Listed at: Ulrich's Periodicals Directory ©, U.S.A., Open J-Gate as well as in Cabell's Directories of Publishing Opportunities, U.S.A

Database manager is the DBMS 'central software component which interfaces applications and queries submitted by the user. The Server Manager is also referred to as the Server Control System and manages Database Access at runtime with subcomponents such as Authorization Control, Command Processor, Integrity Checker, Request Optimizer, Transaction Manager, Scheduler, Recovery Manager and Buffer Manager.

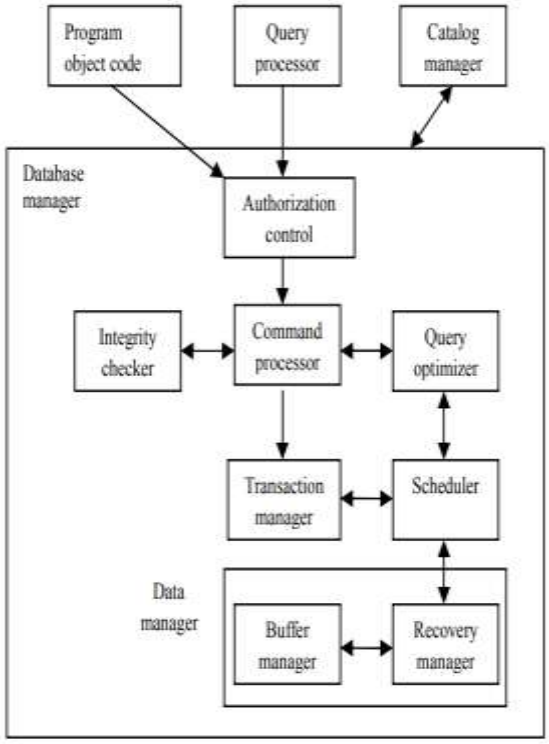

Figure 4.1: Components of a database manager

In general, database systems are categorized on the basis of specific factors such as the number of users using the system, the nature and degree of usage and the locations of the database site. Single user DBMS and multiuser DBMS identify a database system based on number of users. Single user DBMS is limited to personal computer systems with no data sharing.

3. INTERFACE DATA MODELING TO DETECT AND DIAGNOSE INTERSYSTEM FAULTS FOR DESIGNING AND INTEGRATING SYSTEM OF SYSTEMS

A complex system like automotive, marine, or aerospace system (SoS) comprises a number of subsystems that need to be planned and configured to work together. For example, in an underwater vessel an inertial navigation system (INS) receives speed over water and location information from an electromagnetic log (EM log) and GPS, respectively. They allow the INS to improve its orientation and velocity in computational accuracy.

In this case, the INS input data were used precisely to calculate the outputs of the INS; in addition, the outputs are also a basis for estimating the vessel's geographic position in water. Therefore, understandable and consistent interfaces between the subsystems are the key requirement for corporate-level management of the subsystems as an integrated framework. For the following reasons, a device integrator has difficulty finding out interface faults when designing and implementing the complex SoS.

Secondly, both interface subsystems are typically built by various manufacturers, resulting in inconsistent implementation of the same interface protocols. In addition, the manufacturers also prefer custom protocols to standardized ones for fast modificability and scalability. In this respect, both the protocols and the integration process are periodically updated during the device design phase.

This research suggests a systemic approach on how a subsystem communicates effectively with a counterpart when built and implemented for the entire device. Specifically, during node-to - node distribution over the digital network, we concentrated on solving device faults (i.e., anomalies). In this analysis, our goal is to ensure consistent and stable interfaces by testing the consistency between the interface protocols and the interface data obtained.

4. DBMS implementation

The stage of constructing the database comes after the design of a database for an application. A suitable general purpose DBMS may usually be selected for use for this purpose. A DBMS provides the user interfaces that database administrators need to use to identify the data structures of the specified application within the respective DBMS model. Certain user interfaces are used to select appropriate DBMS parameters (such as security related parameters, storage allocation, etc.).

When the database is ready (defining all of its data structures and other necessary components), it is usually filled with data from the initial application (database initialization, generally a distinct project; in many cases, utilizing specialized DBMS interfaces that allow bulk insertion) before it is operational. For certain instances, though empty from the data of the application, the database is active, and data is collected throughout its operation. The database maintenance stage occurs after completion, constructing the database and making it operational: different database parameters which need adjustments and tuning for improved performance, the data structures of the application may be modified or added, new specific computer programs may be written to add to the functionality of the application, etc.

The reasons for this are mainly economic (different DBMSs may have different total ownership costs or TCOs), functional, and operational (different DBMSs may have different capabilities). The conversion requires the transformation of the database from one sort of DBMS into another. The transformation will keep the framework related to the database (i.e., all associated application programs) intact (where possible).

Thus, in transformation the conceptual and external architectural levels of the database should be retained. It may be beneficial to retain certain elements of the internal architecture standard, too. A complex or significant migration of databases can be a complicated and costly (one-time) project by itself, which should be factored into the migration decision. This while there may be resources to help migrate between different DBMS. A DBMS vendor usually offers tools to help import the databases from other common DBMSs.

5. Conclusion

A database is a set of linked data that reflects some real world aspect. For a specific function, a database system is built to be constructed and filled with data. Database Management System (DBMS) is a program for storing and restoring data from users while taking necessary security measures into consideration. This is composed of a collection of programs that control the database. The DBMS acknowledges an application 's request for data, and instructs the operating system to provide similar data. A DBMS allows users and other third-party applications to store and retrieve data in large systems. DBMS enables users to build their own databases, as needed. The word "DBMS" covers the database owner as well as other computer programmes. It offers an interface between the software application and the data.

Database management system (DBMS) is the essential part of current IS technology and provides a robust and efficient framework for storing and managing large quantities of information in a multiuser environment. DBMS architecture is necessary to handle massive quantities of data to meet every user's information needs. Database work recently aims to improve DBMS flexibility and performance to meet the requirements of modern database applications.

References

1. Bertino, B. Catania, G.P. Zarri, "Intelligent database systems", Reading, Addsion Wesley Professional, 2001.

2. M. Brodie, "Future Intelligent Information Systems: AI and Database Technologies Working Together" in Readings in Artificial Intelligence and Databases, Morgan Kaufman, San Mateo, CA, 1988.

3. Johnson T. (1985), Natural Language Computing : The commercial Applications. Ovum Limited , London

4. Weizenbaum J, (1966).“ELIZA- A computer program for the study of natural language communication between man and machine”, Communications of the ACM, 9(1), pp.36-45.

5. W.A. Woods, R.M. Kaplan, B.N. Webber, The lunar sciences natural language information system: Final report, BBN Report 2378, Bolt Beranek and Newman Inc., Cambridge, Massachussetts, 1972.

6. Rangel, Rodolfo and Joaquín Pérez, O. and Juan Javier González, B. and Gelbukh, Alexander and Sidorov, Grigori and Rodríguez, M.(2005), “A Domain Independent Natural Language Interface to Databases Capable of Processing Complex Queries”,MICAI 2005: Advances in Artificial Intelligence LNCS 3789 pp 833-842.

7. Buchanan, B., Mitchell, T.M., Smith, R.G., and Johnson, C.R. Models of learning systems. In Encyclopedia of Computer Science and Technology, Belzer, J., Holzman, A.G., and Kent, A., editors. Marcel Dekker, New York, 1977, Volume 11, pp. 24– 51.

8. Bertino, B. Catania, G.P. Zarri, “Intelligent database systems”, Reading, Addsion Wesley Professional, 2001.