Performance based Comparison between RDBMS and OODBMS

Tariq Aziz Rao
Department of Computer Science
Virtual University of Pakistan
Lahore, Pakistan

Ehsan-ul-Haq
Department of Computer Science
Virtual University of Pakistan
Lahore, Pakistan

Dost Muhammad Khan
Department of Computer Science
Islamia University of Bahawalpur,
Pakistan

ABSTRACT

Database is an obligatory part of any modern information system as mostly desktop and web-based applications utilize it for maintaining records in government and private sectors. Typically, Relational Database lead the database market but the evolution of Object-Oriented Database has started to substitute the Relational Database due to prerequisite of modern era's applications. Object-Oriented Database has several advantages as compare to Relational Database like better flexibility, reliability, reusability, extensibility, redundancy, response time, design & implementation and real-world modeling. Nevertheless, Object-Oriented Database has some limitations, such as, lack of standard query language, security issues, non-availability of basic features, no mathematical foundation and deficiency of qualified programmers & database administrators, due to which Relational Database may not eradicate entirely from database market. Despite some constraints of Object-Oriented Database, its future seems vivid.

Keywords

Relational Database, Object-Oriented Database, Performance Comparison, Advantages & Disadvantages of OODBMS, limitation of RDBMS

1. INTRODUCTION

Database is an important building block of any modern information system as it manages data proficiently and permits users to execute multiple tasks. Database Management System is a set of logically connected data and software package to retrieve, manipulate and manage a large amount of data efficiently. DBMS not only boost the efficiency of business process but also decrease the overall cost.

Edgar Codd is the inventor of RDBMS which based on Relational Data Model having data in tabular shape (rows & columns), whereas in Object-Oriented Database, everything including data is modeled as objects rather than as table which help to deal with complex data structure. Object-Oriented Database Management System (OODBMS) should assure the standard of DBMS as well as Object Oriented System.

Relational Database Management System based on very strong mathematical foundation which allows the consumer to query the tables to get the requisite data from one or more tables easily but inefficient in processing of large amount of complex data, whereas, Object-Oriented Database can easily handle the huge volume of complex data.

In 1985, Object-Oriented Database System comes into existence to remove the inadequacy of conventional Relational Database and to support the advance applications, such as CAD, CAM, CIM and CASE. Object-Oriented Database schema is a conceptual explanation of the real world data, which considered as future database to maintain high-level data abstraction.

This paper portrays the stream of two most popular technologies, Relational and Object-Oriented Databases. The basic theme of this study is to reflect the real picture of these technologies, so that, Database Administers could be well aware about pros and cons of Relational and Object-Oriented Database Systems before its implementation.

2. LIMITATION OF RELATIONAL DATABASE MANAGEMENT SYSTEM (RDBMS)

In fact, RDBMS is renowned in all over the world where there is need to store data but due to some constraints it does not support the requirements of modern applications, which are described below: -

- Relational Data Model cannot represent the real world entities in proper manner.

- The Relational Data Model has inadequate number of data types, which are insufficient to signify the real world objects. Therefore, after the development of the system, new data type cannot be added.

- In RDM, table is only single semantic structure for showing data as well as relationship, so, it is much more complex to trace out that which is available for relationship or model data [1].

- Normalization is essentially required to retain the consistency of database but sometime it is invaluable because a few broken relations are not relates to real world. The decomposition is affected the query processing as selection, project and join operations have to be used frequently for reconstruction of objects [2].

- Relational Data Model does not support the enterprise constraint, which are defined by industry principles. It supports inadequate number of constraints.

- Impedance mismatch between structure query language and advanced programming languages, such as, Java & C++.

- In Relational Database, every cell of a table includes only single/atomic value; this property makes it limited.

- Relational model operations cannot be extended because Relational Data Model has a fixed number of SQL operations.

- Relational Database Management System provides the poor schema evolution support.

- In RDBMS, difficulty faced in handling the recursive queries.

- RDBMS provide the meager support for long interval transactions because mostly transactions are small lived in RDBMS.

- RDBMS provides the poor navigational access.

- RDBMS does not provide the appropriate functional and structural support to advance applications.

3. ADVANTAGES OF OODBMS OVER RDBMS

Due to large volume and complex data, the Relational Database Management System suffers several problems. In order to handle the complex graphical & hypermedia data and advance applications, such as, Computer Aided Design (CAD), Computer Aided Manufacturing (CAM), Computer Integrated Manufacturing (CIM), Computer Aided Software Engineering (CASE) and Advance Office Automation Systems are used Object-Oriented Database System. The major aspects of using Object-Oriented Database Management System are high performance and handling complex data.

Object Oriented Database consists of abstract data types which permit the users to explain the features of the data to be store during development of an application [3]. Mahesh S. Raisinghani and Gabriel Custodio [4] described that Object-Oriented Database Management System carry out schema evolution, direct navigation, clustering and further helpful functions that Relational Database Managements System cannot contest. Object-Oriented Databases are suitable for client/server and distributed architectures and better for applications where the relationship between objects in the database holds the important data [5].

Relational technology permit attributes to be added into existing tables but plunging or moving attributes to other tables are not often permitted [6], however, in Object-Oriented Technology, there are two kinds of schema evolution:

   i. Changes to the class definition
   ii. Changes to the structure of a class hierarchy

Object-Oriented Technology has several other features, which are not available in the relational technology like concepts of temporal evolution of data, dynamic binding and polymorphism, etc. Some benefits of OODBMS over RDBMS are: -

3.1 Reduced maintenance cost

The basic purpose of OO development is to design such a system which have longer life and smaller maintenance cost. In Object-Oriented paradigm, mostly processes encapsulated within the system and performance may be reutilized and integrated into new behaviors.

3.2 Better Flexibility and reliability

Object-Oriented are more flexible and reliable as compare to the conventional systems as new activities can easily constructed from exiting objects. Fresh objects can be formed at anytime because objects are accessed and called dynamically. Furthermore, behavior may also be hereditary from super classes.

3.3 More Reusability and extensibility

Object-Oriented databases provide more reusability as newly formed objects will automatically inherit the data attributes and characteristics of the class from which it formed. Object-Oriented Databases also provide the ability to produce new data types from existing ones by using its core concepts of inheritance and polymorphism, whereas, these features are not available in relational databases.

3.4 Easy managing of very large data

Very large volume data can easily handle by utilizing the Object-Oriented Databases. According to Hibatullah Alzahrani [7] Object-Oriented Database Management System is the world largest database having volume approximately 900 TB.

3.5 Better response time and faster data access

Response time of Object-Oriented Databases is much better than Relational Databases. Similarly, Object-Oriented Database Management System is typically faster as they have many-to-many relationship and objects can be accessed by using pointers only and there is no requirement of 'join' because objects are linked through pointers.

3.6 Requirement for advance application

Object-Oriented Databases are the dire need of advance applications of modern era because some features are not provided by the Relational Databases.

3.6.1 Computer Aided Design

Data regarding integrated circuit chips, buildings and airplanes is very large which is stored and managed in such kinds of applications. Configuration management and version control of these applications is essential which is only fulfilling by Object-Oriented Databases. Mostly many people working on same design, so, it is easily support for cooperative engineering.

3.6.2 Computer Aided Software Engineering:

Data regarding different phases of software development life cycle is managed by these modern applications. Design of such applications may be very large which involves cooperative engineering.

3.6.3 Computer Aided Manufacturing:

Data of these applications is very similar to Computer Aided Design (CAD) but requires separate production. Furthermore, in order to response to a specific situation common algorithms and customs rules are used.

3.7 Real-world modeling

Object-Oriented System has the ability to model the real-world objects in natural way instead of traditional methods, which is very helpful in maintaining data, such as, multimedia content. On the other hand, it is much more difficult to store multimedia content in form of table in relational database, whereas, Object-Oriented Database can model the multimedia content like an object and store it effortlessly as object oriented model is based on objects instead of data.

3.8 Better Design and implementation

Mostly, preference gives to Object-Oriented Databases due to its better design and implementation. In Object-Oriented Database, data is directly design as objects and then implemented it as such that makes the implementation fluently.

3.9 Redundancy

In Object-Oriented Database Management System, non-redundancy is achieved by using the concept of encapsulation and inheritance but the Relational Database Management system is not effective in reducing the data redundancy.

3.10 No Primary Key

In Relational Database Management System, the user is mostly concern regarding recognizing tuples by their values and it is also make confident that no two tuples have the identical primary key to avoid any unwanted error. However, in OODBMS, the unique reorganization of objects is performed via OIDs [8], which is not visible to users. The Object ID (OID) table gives a record for searching and getting data in the OODB [9].

4. DISADVANTAGES OF OODBMS:

Regardless of several advantages of OODBMS, RDBMS is still popular in all over the world. Some disadvantages of OODBMS are:

4.1 No standard query language

No standard query language is available for OO databases like SQL, which is being used in relational databases. However, Object Data Management Group (ODMG) standardized object query language (OQL) but it is not widely used. Furthermore, approach is being changed towards implementing local queries in programming languages like LINK in C#.

4.2 No proper implementation of consistency constraints mechanisms

Object-Oriented Databases are not fully implementing the consistency constraints. OODMBS provides only limited number of features for integrity constraints, rareness of constraints and other enterprise constraints.

4.3 Non-availability of Basic Features

Few common features such as Meta Data Management, Triggers and constraints like, NULL and UNIQUE are found missing in Object-Oriented Databases but these features are available in Relational Databases.

4.4 Absence of Ad hoc queries & closure

Closure property is unavailable in Object-Oriented Databases because it is a property of relational databases that enables the nested queries where new tables are created by joining existing ones and then querying the new table [7]. No nested query in OODBMS as there is no 'join'. All Database Management Systems must be required to support ad hoc queries; however, few Object-Oriented Database Managements Systems violate this criterion.

4.5 No capability for view

Object Databases have no capability for view but Relational Databases have the capability for view (tables). Due to some characteristics of object model like object identity, it is much more difficult to implement the view capability. Moreover, the core concept of OO programming, such as encapsulation and inheritance make it more complex to execute views in Object Databases.

4.6 Difficult to optimize queries

Queries optimization is crucial in order to boost the overall worth of the system, however, due to the subsequent causes; it is complicated to optimize queries in Object Databases [1]:

- The users define data types.
- OQL (Object Query Language) has nested structure.
- Encapsulation, complex objects and methods.
- Object identity and changing selection of types.

4.7 No mathematical foundation

No mathematical foundation is exist for Object-Oriented Databases, whereas, Relational Databases are based on Relational Algebra and Relational Calculus which increase its popularity worldwide.

4.8 Deficiency in availability of qualified programmers & Database Administrators

Deficiency in availability of qualified programmers and database administrators is another cause of unfamiliar with Object-Oriented Database Management System because mostly mangers of software companies do not give proper time in training their staff in Object-Oriented methods. A considerable time has elapsed but the number of qualified and well-versed programmers are still limited.

4.9 Non-acceptance by major vendors

Another reason of limited use of OO Databases is the non-acceptance of OO paradigm by major vendors. For instance, when the American Standards Committee proposed ASCII, at that time, IBM ignored the standard and carry on with their own character set i.e. EBCDIC. Even some decades later, no resolution between ASCII and EBCDIC was exit. Therefore, there is a dire need to convince major vendors to shift from RDBMS to OODBMS.

4.10 Security Issues

Relational Databases support the authorization but most of the Object-Oriented Databases do not support the authorization. In Object-Oriented Databases, users require explicitly setting and releasing locks, whereas, Relational Databases mechanically set and free locks in user processing query and update reports [10].

5. COMPARISON BETWEEN RDBMS AND OODBMS

The key objective of Object-Oriented Database System is to propose class independence features for the objects and encapsulation, whereas, the main goal of Relational Database System is to ensure data independence from application programs. Comprehensive comparison between RDBMS and OODBMS is appended in Table 1:-

Table 1. Comparison between RDBMS & OODBMS

Features	RDBMS	OODBMS
Model	Relational Database Model	Object-Oriented Database Model
Standard	It is fully standardized.	Absence of standard

Object-oriented Programming	It is not directly supported OO programming.	It is direct and extensive support to OO programming
Mathematical foundation	It based on very strong mathematical foundation (Relational Algebra).	No mathematical found is exists for object-oriented databases.
Query language	No standard query language is available for object-oriented databases.	Standard query language is available for relational databases like SQL.
User defined data types	Meager support for user defined data types	Fully support for user defined data types
Relationship support	It provide support only for simple relationship	It provide support for simple as well as complex relationships
Advance applications support	It does not provide the appropriate functional and structural support to advance applications.	It provides fully support to advance applications.
Support for complex objects	It provides fewer supports for complex objects.	It provides better support for complex objects.
Transactions	It provides deprived support for wide duration transactions because mostly transactions are small lived.	It provides better support for long duration transactions.
Navigational access	It provides poor navigational access.	It provides better navigational access.
Schema evolution support	It provides poor schema evolution support	It provides better schema evolution support.
Normalization	Normalization is strongly recommended except few situations in which broken relations are not relates to real world.	No need of normalization
Implementation of integrity constraints and enterprise constraints	Partial	Full
Support for multimedia data	It provides less support for multimedia data.	It provides full support for multimedia data.
Data storage	It can only store data in table, which is passive in nature.	It can store both data and methods as active object
Semantic nature	It is semantically very weak.	It is semantically very strong.
Representation of real world entity	Poor	Strong
Set of operations	It has fixed set of processes due to relational algebra.	Set of processes can be extended easily
Availability of Programmers & DBAs	Sufficient number of Programmers and Database Administrators are available for RDBMS.	No sufficient number of Programmers and Database Administrators are available for OODBMS.
Acceptance ratio	Mostly vendors are interested to use RDBMS.	Non-acceptance of Object-Oriented paradigm by major vendors
Data structure	Simple	Complex
Redundancy	Non-redundancy is achieved by data normalization.	Non-redundancy is achieved by using the concept of encapsulation and inheritance.

6. CONCLUSION

Complex modern applications lead to poorer maintainability and extensibility for which elastic and dynamic concepts can help to diminish this complexity. Despite of some limitations, Object-Oriented Databases offers superior performance and various valuable features, such as, faster data access, custom data types, support for modeling real-world complex data structures and much more which are missing in Relational Databases, so, it is expected that the trend would move from RDBMS to OODBMS in future. This paper will prove supportive for Database Administrators in performance estimation of OODBMS and RDBMS before its implementation. However, the production of most dominant Database Management System is still a great challenge for researchers.

7. REFERENCES

[1] Damesha, H. S., 2015. Object Oriented Database Management Systems-Concepts, Advantages, Limitations and Comparative Study with Relational Database Management Systems. Global Journal of Computer Science and Technology: C Software & Data Engineering, 15(3), pp. 11-18.

[2] Saxena, V. and Pratap, A. 2013. Performance Comparison between Relational and Object-Oriented Databases. International Journal of Computer Applications, 71(22), pp. 6-9.

[3] Suri, P. and Sharma, M. 2011, May. A comparative study between the performance of relational & object oriented database in data warehousing. International Journal of

Database Management Systems (IJDMS), 3(2), pp. 116-127.

[4] Mahesh, S. and Raisinghani, G.C. 2000. Object-Oriented Database Management Systems: Architecture and Application. Journal of Information Technology Theory and Application (JITTA), 2(1), pp. 11-18.

[5] Babu, R. 2014, June. Object Oriented Database Management System for Decision Support System. International Refereed Journal of Engineering and Science (IRJES), 3(6), pp. 55-59.

[6] Butuner, H. 2012. Advantages of object-oriented over relational databases on real-life applications. Research Journal of Economics, Business and ICT, Vol 5(2012).

[7] Alzahrani, H. 2016. Evolution of Object-Oriented Database Systems. Global Journal of Computer Science and Technology: C Software & Data Engineering, 16(3), pp. 32-36.

[8] Ghongade, R. S. and Pursani, P. J. 2014, November. Comparison of Relational Database and Object Oriented Database. International Journal of Modern Trends in Engineering and Research (IJMTER), 1(5), pp. 27-33.

[9] Kumar, C. S., Seetha, J. and Vinotha, S. R. 2012, November. Security Implications of Distributed Database Management System Models. International Journal of Soft Computing and Software Engineering (JSCSE), 2(11), pp. 20-28.

[10] Bagui, S. 2003. Achievements and Weaknesses of Object-Oriented Databases. Journal of Object Technology (JOT), 2(4), pp. 29-41.