International Journal of Modern Trends in Engineering and Research

Comparison of Relational Database and Object Oriented Database

Rutija S. Ghongade[1], Prof. P J. Pursani[2]

Final CSE Student, HVPM's COET Amravati[1]
Assistant Professor, CSE Dept., HVPM's COET Amravati[2]

Abstract—The object-oriented database (OODB) is the combination of object-oriented programming language (OOPL) systems and persistent systems. Object DBMSs add database functionality to object programming languages. They bring much more than persistent storage of programming language objects. A major benefit of this approach is the unification of the application and database development into a seamless data model and language environment. This report presents the comparison between object oriented database and relational database. It gives advantages of OODBMS over RDBMS. It gives applications of OODBMS.

I.   INTRODUCTION

A structured collection of records is known as Database. The way in which database is structured is defined as Database Model, Hierarchical model, Network model, Object model and Relational model are some of the commonly used models database in the structure of database. A database facilities an organized storage of data in the form of records. The record can be queried in different ways to retrieve information from database. Owning to an organized storage of data a database becomes a useful tool of data storage. Database form any basic tier of any multi-tier system. The information handled and manipulated by the other layers of an n-system basically stored in structured manner in database[6-12].

(RDBMS - relational database management system) A database based on the relational model developed by E.F. Codd. A relational database allows the definition of data structures, storage and retrieval operations and integrity constraints. In such a database the data and relations between them are organized in tables. A table is a collection of records and each record in a table contains the same fields[6-12].

Properties of Relational Tables:

- Values Are Atomic

- Each Row is Unique

- Column Values Are of the Same Kind

- The Sequence of Columns is Insignificant

- The Sequence of Rows is Insignificant

- Each Column Has a Unique Name

Object DBMSs add database functionality to object programming languages. They bring much more than persistent storage of programming language objects. Object DBMSs extend the semantics of the object oriented programming languages to provide full-featured database programming capability. According to Rao (1994), "The object-oriented database (OODB) paradigm is the combination of object-oriented programming language (OOPL) systems and persistent systems. The power of the OODB comes from the seamless treatment of both persistent data, as found in databases, and transient data, as found in executing program.[1-5]

II.   OBJECT ORIENTED DATABASE

Object oriented databases are also called Object Database Management Systems (ODBMS). Object databases store objects rather than data such as integers, strings or real numbers. Objects are used in object oriented languages such as Smalltalk, C++, Java, and others. Objects basically consist of the following:

Attributes - Attributes are data which defines the characteristics of an object. This data may be simple such as integers, strings, and real numbers or it may be a reference to a complex object.

Methods - Methods define the behavior of an object and are what was formally called procedures or functions.

In contrast to a relational DBMS where a complex data structure must be flattened out to fit into tables or joined together from those tables to form the in-memory structure, object DBMSs have no performance overhead to store or retrieve a web or hierarchy of interrelated objects. This one-to-one mapping of object programming language objects to database objects has two benefits over other storage approaches: it provides higher performance management of objects, and it enables better management of the complex interrelationships between objects. This makes object DBMSs better suited to support applications such as financial portfolio risk analysis systems, telecommunications service applications, World Wide Web document structures, design and manufacturing systems, and hospital patient record systems, which have complex relationships between data[1-5].

- **Object Persistence**

With traditional databases, data manipulated by the application is transient and data in the database is persisted (Stored on a permanent storage device). In object databases, the application can manipulate both transient and persisted data.

III.   ADVANTAGES OF OBJECT ORIENTED DATABASE

- **Capable to handle many different types of data types:**

An object oriented database can store any type of data including text, numbers, pictures, video and voice.

- **Combination of object oriented programming and database technology:**

The object oriented database technology combines object oriented programming with the database technology to provide an integrated application development system. There are many advantages of including the definition of operations with the definition of data. First the

operations which are defined are applied and are not depending on the specific database application running at the moment. Second the data types can be extended to support complex data.

- **Improves productivity:**

Inheritance allows programmers to develop solutions to complex problems by defining the new objects in terms of defined objects in the past. Polymorphism and dynamic binding allows programmers to define operations for one object and then to share the specification of the operation with other object. These objects can further extend this operation to provide behaviors that are unique to those objects.

- **Data access:**

Object oriented databases represent relationships explicitly supporting both navigational and associative access to information.

- **Enriched modeling capabilities:**

The object oriented data mode allows the 'real world' to be modeled more closely. The objects which encapsulates both state and behavior is more natural and realistic representation of real world objects. An object can store all relationships that it has with other objects including many-to-many relationships and objects can be formed into complex objects that the traditional data model cannot cope with easily.

- **Extensibility:**

OODBMSs allow new data types to be built from existing data types. The ability to factor out common properties of several classes and form them into a super class that can be shared with subclass can greatly reduce redundancy within system is regarded that one of the main advantage of object orientation. Further, the reusability of classes promotes faster development and easier maintenance of database and its applications.

- **Removal of Impedance mismatch:**

A single language between Data Manipulation Language (DML) and the programming language overcomes the Impedance mismatch. This eliminates many of the efficiencies that occur in mapping a declarative language such as SOL to an imperative language 'C'.

- **More expressive query language:**

Navigational access is the most common form of data access in OODBMSs. This is in contrast to a associative access of SQL(that is, declarative statement with selection based on one or more predicates).Navigational explosion is suitable for handling parts explosion, recursive queries and so on.

- **Support for schema evolution:**

The tight coupling between data and applications in OODBMSs makes schema evolution more feasible.

- **Applicability to advanced database applications:**

There are many areas where traditional DBMSs have not been used, such as Computer Aided Design (CAD), Computer Aided Software Engineering (CASE), Office Information System (OIS), and Multimedia systems. The enriched modeling capabilities of OODBMSs have made them suitable for these applications.

- **No Primary Keys:**

The user of an RDBMS has to worry about uniquely identifying tuples by their values and making sure that no two tuples have the same primary key values to avoid error conditions. In an OODBMS, the unique identification of objects is done behind the scenes via OIDs and is completely invisible to the user. Thus there is no limitation on the values that can be stored in an object.

IV. COMPARISON WITH RELATIONAL DATABASE

- **Comparison of OODBMS and RDBMS**

Object Oriented model	Relational model	Differences
Object	Entity	The object specifies behavior too
Class of Objects	Types of Entities	The class of objects includes the common behavior of objects in that class
Class hierarchy	The database schema	The class hierarchy includes inheritance, while the scheme use the external keys
Class instance	Entity, tuple or record	The instance may have a more restrictive activity
Object identifier(OID)	Primary key	In the relational model primary key is not identified by the system

Table 1. Comparing OODBMS and RDBMS as far as data modeling is concerned.

OODBMS	RDBMS
Main Objectives: data encapsulations and independence.	Main Objective: ensuring data independence from application programs.
Independence of classes: classes can be recognized without affecting the mode of using it.	Data independence: Data can be recognized without affecting the mode of using it.
OODBMS store data and methods.	RDBMS store only data.
Encapsulations: the data can be used only through their class methods.	Data partitioning: data can be partitioned depending on the requirements of the users and on the specific user's applications.
Complexity: the structure of data may be complex, involving types of data.	Simplicity: users perceive data as columns, rows/tuples and tables.

Table 2. Comparing OODBMS with RDBMS considering their objectives.

V. LITERATURE REVIEW

1. Comparison of RDBMS, OODBMS and ORDBMS[1]:

In this paper author has presented the similarities and differences between relational modeling of data and the object oriented modeling of data is of great importance both for database designers and for users. By being well acquainted with the relational model and by noting the similarities and differences between the two approaches to data modeling, designers will be able to turn into account and to make use of the already acquired experience as an important basis for under-standing and learning the methodology of designing object oriented databases.

2. Comparison Between Object-Relational and Object-Oriented Database[2]:

The author of this paper discuss some concepts related to the object-relational and object-oriented database system such as object identity, user defined types(UDTs), user-defined routines, polymorphism, subtypes and super types, persistent stored modules, and large objects. At the end of this paper exist comparison between ORDBMS and OODBMS.

3. Performance Comparison of Relational Database with Object Database DB4O[3]:

The author of this paper discusses object oriented approach for solving queries. The queries executed and specifically on an object oriented database DB4o. The paper also discusses two factors namely querying methodology and performance analysis in comparison with traditional DBMS like MYSQL. The proposed approach proves that OODBMS is more efficient compare to RDBMS.

VI. APPLICATIONS OF OBJECT ORIENTED DATABASE

Object databases work well with:
- CAS Applications (CASE-computer aided software engineering, CAD-computer aided design, CAM-computer aided manufacture)
- Multimedia Applications
- E-Commerce

VI. REASONS OF NOT USING OBJECT ORIENTED DATABASE PUBLICALLY

- **Lack of Universal Data Model:**

There is no universally agreed Data Model for OODBMS and most models lack of theoretical foundation. This disadvantage is seen as significant drawback and is compared to pre-relational system.

- **Lack of Experience:**

In comparison to RDBMSs the use of OODBMSs is relatively limited. This means that we don't have level of experience that we have with traditional system. OODBMSs are still very much geared towards the programmer rather than native users. While the OODBMS is limited to small niche market this problem will continue to use.

- **Lack of Standards:**

There is a general lack of standards of OODBMS. There is no standard Object Oriented Query Language.

- **People awareness about OODBMS:**

People are not aware about the Object Oriented Database and therefore OOD had not been expanded into the business field.

VII. CONCLUSION

It is difficult to represent and implement a variety of applications using relational database technology because of the shortcomings .These applications include computer aided design, engineering, software engineering and manufacturing systems (CAD, CAE, CASE, and CAM); knowledge based systems multimedia systems that manage images, graphics, voice, and textual documents; statistical and scientific modeling and analysis programs; information systems; and so on. However, objects oriented technology, which is the fundamental advancement in database technology, satisfies the objective of reducing the difficulty of designing and implementing very large and complex database requirements.

REFERENCES

[1][Georghe& Bucharest, 2007], Georghe and Bucharest " Comparison of RDBMS, OODBMS and ORDBMS"Journal of RevistaInformaticaEconomică,Vol.4 , No.44, 2007

[2] [Daniela & Mehmud , 2012], Daniela and Loie Naser Mehmud"Technical Institute of Sofia, Bulgaria" , VOL.13,no 3(36), November 2012

[3] [Roopak et al., 2013], K. E. Roopak, K.S.Rao, S. Ritesh and S. Chickerur, "Performance Comparison of Relational Database with Object Database DB4O", IEEE 5thComputational Intelligence Conference and Communication Networks(CICN), 2013.

 [4] [Codd, 1970] Codd, E.F., 1970. A Relational Model for Large Shared Data Bank, CACM.

[5]Date, C.J., 1985. An Introduction to Database System, Addison Wesley.

[6]Lamb, C., G. Landis, J. Orenstein and D. Weinreb, 1991. The object store database system. Commun.ACM, 34: 50-63.

[7]Mary, E.S.L., 1995. Object Databases: The Essentials, Reading, Mass. Addison-Wesley.

[8]Mary, E.S.L., 1992. ODBMS vs. Relational. J. Object-Oriented Programming Focus on ODBMS, pp: 35.

[9]Brodie, M., J. Mylopoulos and J. Schmidt, 1985.On Conceptual Modeling. Springer-Verlag.

[10]Atkinson, M., F. Bancilhon, D. DeWitt, K.Dittrich,D. Maier and S. Zdonik, 1989. The Object-Oriented Database System Manifesto. Proc. First Intl. Conf. Deductive and Object-Oriented Databases, Kyoto, Japan, pp: 223-40.

[11]Michael, S. and G. Kemnitz, 1991. The postgres next generation database management system.Commun. ACM, 34: 78-92.

[12]Arthur, M.K., R. Jensen and S. Agarwal, 1993.Persistence software: Bridging object-oriented programming and relational databases. Proc. ACM SIGMOD Intl. Conf. Management of Data, Washington DC, pp: 523-528.