

# Object Oriented Query Response Time for UML Models

Vipin Saxena, Santosh Kumar

Department of Computer Science, Babasaheb Bhimrao Ambedkar University, Lucknow, India.  
Email: vsax1@rediffmail.com, sant7783@hotmail.com

Received April 17<sup>th</sup>, 2012; revised May 25<sup>th</sup>, 2012; accepted June 7<sup>th</sup>, 2012

## ABSTRACT

Nowadays, the size of database of any business organization is increasing and many of the companies are shifted the old structured database into the object oriented database. Due to increase of size of database complexity of database is increasing therefore, it is necessary to optimize the object oriented query response time from the complex object oriented database. In the present paper, a real case study of Life Insurance Corporation of India is taken and sample object oriented database is designed by the use of SQL Server 2008. A UML model is designed for computing the object oriented query response time. Table and graph are also represented for the computed records in five runs.

**Keywords:** UML; Object Orientation; Class Diagram; Sequence Diagram; SQL Server

## 1. Introduction

The Unified Modeling Language (UML) is a universal visual modeling language used to represent static and dynamic behavior of system. In 1997-OMG has approved the UML which is an open-industry standard visual modeling language. UML is standard language for writing software blueprints; it also used to visualize, specify, construct and document the artifacts of a software-intense system. it is a very popular language addressing all the views needed develop and then deployed for the system and also it is a process independent although optimally it should be used in a process *i.e.* use case driven architecture-centric, iterative and incremental. UML does not provide any kind of modeling methodology but some aspects of methodology are implied, it just provides a visual syntax that is used to construct the UML models.

The object oriented database was proposed to meet the need of more complex applications and this approach offers a flexibility to handle some of the complex requirements without being limited by the data types and query languages available in traditional database systems. The key feature of object oriented database is the power as they give the design to specify the structure of complex objects and the operations that can be applied to these objects. The queries related to object oriented databases support the object oriented features. Any query which returns the result in particular time is called as response time of the query. The response time of an object-oriented query is the time in which the requested output is deployed on computer screen.

## 2. Related Work

Let us briefly explain the literature related with the current work. Blaha and Rumbaugh [1] have defined the object-oriented modeling and design with UML. Booch *et al.* [2,3] have described all the tools of UML that help the user to understand and operate UML easily. Booch [4] has presented all the types of diagrams used for the visual UML modelling language and also described the purpose and advantages of UML for producing the stable software designs. Cranefiel and Purvis [5] have suggested that UML as an ontology modelling language. Dayal [6] has developed a unified approach to processing queries that contain nested sub queries, aggregates and quantifiers. Deshpande *et al.* [7] have explained the cache-and-query for wide area sensor databases. Ganski and Wong [8] have defined the optimization of nested SQL queries revisited. Gomaa [9] has described the concurrent object modelling and architectural design method for the development of concurrent applications, in particular for distributed and real time applications with the use of UML. Holz E. [10] investigated that how the new object-oriented modeling language UML is applied on the telecommunication domain. Huaiming S. *et al.* [11] has proposed a prediction model to forecast query predicates and then to choose them for speculative execution. Kim w. [12] has optimized the nested queries in SQL. The Object Management Group (OMG) [13-16] approved that UML is a visual modelling language; it also described UML profile for schedulability, performance and time; and also released the UML specifications versions. Pillana and Fahringer [17] have defined UML based modeling of performance oriented parallel and distributed Applica-

tions. P. Mohankumar [18] has optimized and analyzed the parallel queries in databases through multicores. Vipin *et al.* [19,20] have proposed an object oriented distributed architecture system and protection of domain based system through UML. Wang *et al.* [21] suggested that the dynamic data migration policies for query-intensive distributed data environments. Liang *et al.* [22] addressed the issues explicitly by studying the top-k query problem in sensor networks with the response time constraint. Muruganathan *et al.* [23] have analytically evaluated the average query response time of the two-level Hierarchical Clustering based Hybrid-routing Protocol (THCHP) proposed recently for wireless sensor networks (WSNs). Wei and Ming [24] presented a preliminary design for a P2P object-oriented database on the open Chord which supports multi-attribute and range queries. Yfoulis *et al.* [25] have presented an efficient and effective MPC-based solution, which improves upon previous work. Tavenard *et al.* [26] have proposed to modify k-means centroids to produce clusters with more comparable sizes without sacrificing the desirable properties. Zhenyou *et al.* [27] have described the structure of the distributed heterogeneous database system of query optimization system based on the Hibernate technical for the B/S structure, and analysis the calculation of the response time. Liang *et al.* [28] have addressed the query response time and its effect on the network lifetime through the study of the top-k query problem in sensor networks with the response time constraint. Murray and Rosenthal [29] have structured dual to ri-tries, the reduced implicant tries are investigated, and the dual problem—determining the implicants of a formula—is considered.

### 3. UML Modeling for Query Response Time

#### 3.1. UML Class Diagram

The Query Response Time (QRT) system is modeled with the help of UML class concept. The QRT system contains the six major classes with their major attributes as shown in **Figure 1** below. In the present work, the complete process of computing the object-oriented query response time for UML models is explained in the form of UML class diagram. The Agent class has multiple associations with the Customer class, the Agent class is used to describe about the policy plan with premium amount and sum assured for Customer who has to fill the policy form. The Agent class has also multiple associations with Branch\_Office class; the Agent class sends filled form to the Branch\_Office class. The Branch\_Office class has single association with the Main\_Branch class, the Main\_Branch class issued the requested policy plan to the Customer; the Main\_Branch has multiple association with the Customer class. The Main\_Branch class has also multiple associations with the Query class, the Main\_Branch class executes a query to

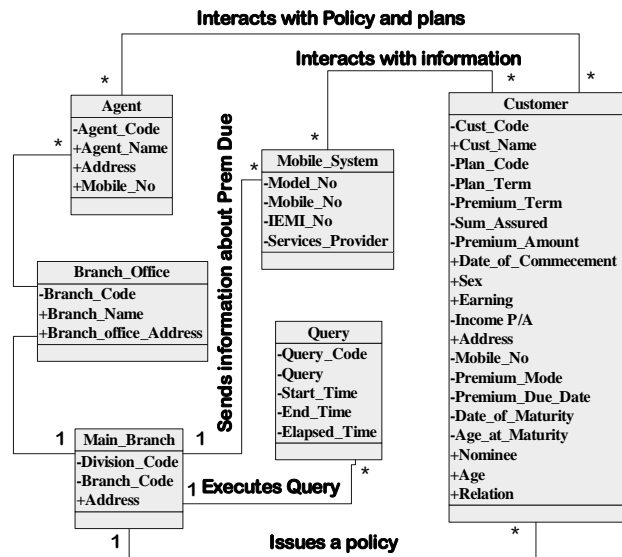


Figure 1. A UML class diagram.

find the customers whose premium due date of issued policy plan is scheduled and sends an information about premium due to the Mobile\_System class. This information is automatically sent to the Mobile\_System within fifteen days before the due date; the Mobile\_System class has the single association with the Main\_Branch class. The Mobile\_System class associates with the Customer class and the customer interacts with the Mobile\_System to know about the premium due date.

#### 3.2. UML Sequence Diagram

A UML sequence diagram represents the dynamic behavior of system which is shown in **Figure 2**. It shows the sequence of executions represented through object life line. In the current work a UML sequence diagram is designed which shows that the complete process of sending information about the premium due of the issued policy plan through executing a query whose premium is due. The four main objects are Main\_Branch, Query, Mobile\_System and Customer. Arrow along the communication message shows the communication between two objects. The vertical line alongwith the pipe shows that the life line of an object. The Main\_Branch executes a query to get the list of customers whose premium of the issued policy is due, the Main\_Branch send information on the Customer's Mobile\_System; the Customer interacts with that information and deposit the due premium amount. The main purpose of this diagram is to check whether the designed model is functioning or not.

### 4. Experimental Results

#### 4.1. A Sample Object-Oriented Database

The object-oriented database is designed by the use of

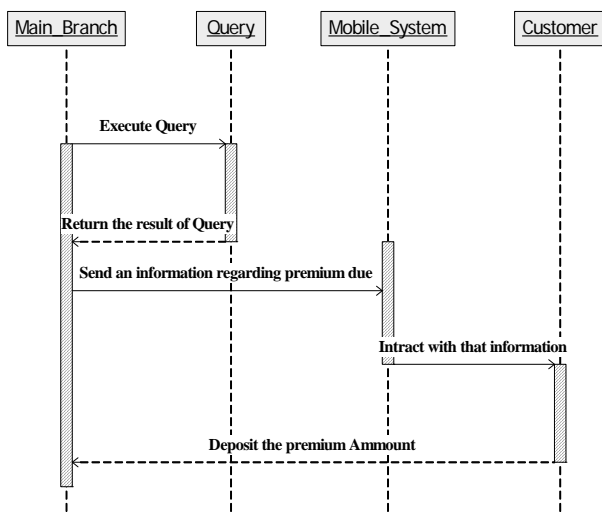


Figure 2. A UML sequence diagram.

SQL Server Management utility of SQL Server 2008. The main database table contains the overall database fields related to customers with their issued policy plan. The purpose of this table is to perform a query and to get quickly the list of customers whose premium of issued policy is due. A list of the customers is shown in the following Figure 3.

#### 4.2. Computation of Query Response Time

In an object oriented database the query processor processed the requested query; the query processor is a procedure that selects the suitable strategy which executes and generates the result. The query processing is the stepwise process, it transformed the requested query into the SQL standard form, while the transformation is performing the parser checks the syntax and verify the relational attributes used in the query that are defined in the data-base. When the query translated into the relational algebraic expression, the query optimization is performed that substitute the equivalent expression and the number of strategies called access plans are generated for evaluating the query, the optimal access plan and executed.

The query response time is a time in which the query processor executes and generates a result in the response of requested data-base by the user. The query response time (QRT) is the part of the query processing strategy (QPS); it is treated as an elapsed time of a query from the execution to end. The given Figure 3 shows the complete process for executing the query requested by the user. The application forwards the requested query to the database server, then it optimize query and retrieve any user defined routines, add or updates the appropriate records and performs any back-ground input/output operations; then the database server return the result to the application which display on the user device.

	Cust_Name	Date_of_Commt	Prem_Due_Date
1	Maya Devi	2010-07-28	2011-01-27
2	Kaushendra Sengar	2010-07-28	2010-10-27
3	Pankaj Srivastava	2010-07-28	2010-10-27
4	Sajeev Kumar	2010-06-28	2011-06-27
5	Sirajuddeen Khan	2010-07-28	2010-10-27
6	Amit Kumar Singh	2010-06-28	2011-06-27
7	Siddhant Thakur	2011-01-28	2011-04-27
8	Upendra Kumar	2011-02-15	2011-03-14
9	Shridhar	2011-02-15	2011-03-14
10	DhanDevi	2010-12-28	2011-06-27
11	Ram Avtar	2011-02-15	2011-07-14
12	Mahesh Prasad	2011-02-14	2011-08-13
13	Rumila	2011-01-28	2012-01-27
14	Ram Das	2011-02-15	2012-02-14
15	Dharmendra Mishra	2011-01-28	2011-07-27
16	Dori lal	2011-02-15	2012-02-14
17	Islam Ali	2011-01-28	2011-07-27
18	Mithlesh Kumar	2011-02-15	2011-05-14
19	Nitesh Kumar	2011-02-15	2011-08-14
20	Tavhid	2011-02-15	2011-05-14
21	Saiyyada Khatoon	2011-02-15	2011-05-14

Figure 3. A sample object-oriented database.

For computing the query response time, the authors have used query analyzer in SQL Server Management Studio. The code segment for computing the query response time is developed in SQL Server 2008 query analyzer. The code segment is given below:

```

DECLARE @EndTime DATETIME
DECLARE @StartTime DATETIME
DECLARE @I INT
SET @I = 0
SET @StartTime = GETDATE()
SET @EndTime = GETDATE()
While (@I < 10)
Begin
PRINT 'StartTime = ' + CONVERT(VARCHAR(30),
@StartTime, 121)
Select Cust_Name, Date_of_Commt, Prem_Due_Date from
tblCustomers
SET @I = @I + 1
PRINT 'EndTime = ' + CONVERT(VARCHAR(30),
@EndTime, 121)
PRINT 'RESPONSE TIME = ' + CONVERT(VARCHAR(30),
@ENDTIME - @STARTTIME, 114) + ''
End
  
```

By the use of above code, a query response time is

compute on the basis of five runs. It is computed in millisecond and on the basis of five runs, average response time is evaluated. In query, lines of codes are increase from 10 to 10<sup>5</sup>. As depicted in the graph, it is increasing as the lines of code increasing, which is shown as in **Table 1**.

The graphical representation of the average query response time and query run is given in **Figure 4**.

Thus from the above analysis the query response time is directly proportional to the line of codes which are executed.

$$QRT \propto LOC \text{ (Line of codes)}$$

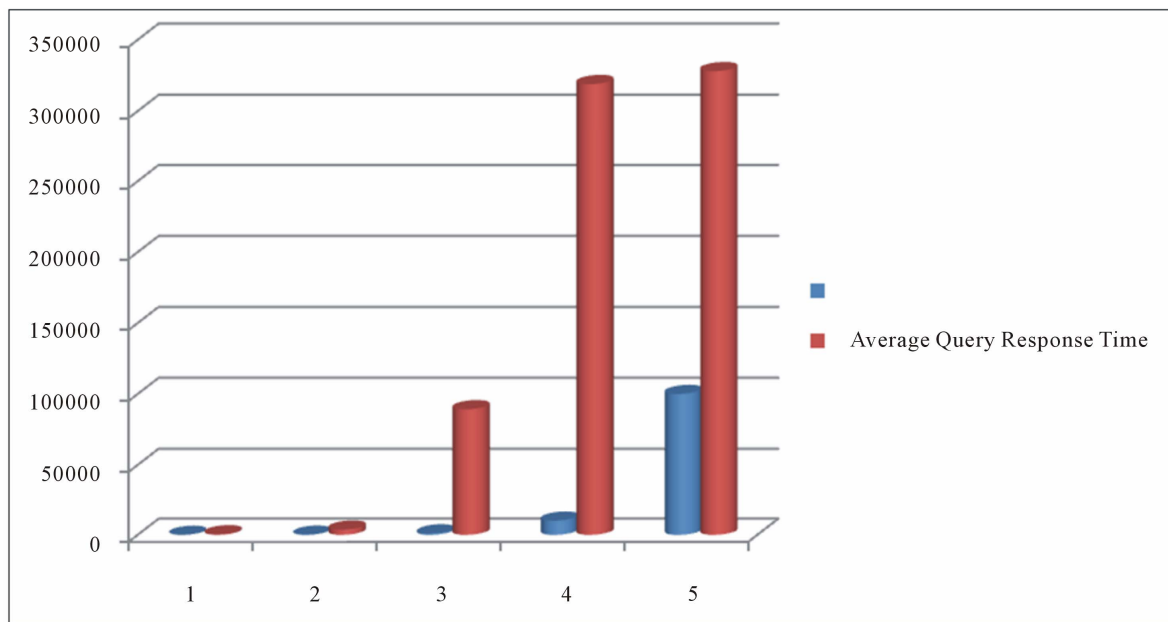
### 5. Conclusion

Form the presented work, it is observed that the UML is a powerful modeling language for presentation of object

oriented database. It is also noted from the literature that UML is not much applied for creating the object oriented database; therefore, it is necessary to construct more building blocks for the object oriented database as software professionals are converting the old structured based technology towards object oriented technology. In this paper, query response time is computed after designing of the UML class and the steps of executions in the form of sequence diagram, the work further can be extended for comparison of search time for various hand move system supported by the SQL Server 2008 and can be optimized the search time for complex query to be executed on the object-oriented databse.

### 6. Acknowledgements

Thanks are due to University Grant Commission New



**Figure 4. Line of code v/s query response time.**

**Table 1. Query response time and average response time (in milliseconds).**

Line of code	Number of runs	10	10 <sup>2</sup>	10 <sup>3</sup>	10 <sup>4</sup>	10 <sup>5</sup>
		1	430	3205	89,847	318,504
2	349	3843	89,066	318,770	340,130	
3	385	3783	89,301	317,921	322,039	
4	546	3740	89,494	319,942	323,202	
5	382	3911	88,525	319,679	323,905	
Average query response time		418	3696	89,246	318,963	328,013

Delhi, for financial support to carry out the above research work.

## REFERENCES

- [1] M. Blaha and J. Rumbaugh, "Object Oriented Modeling," 2nd Edition, Prentice Hall, Upper Saddle River, 2005.
- [2] G. Booch, J. Rumbaugh and I. Jacobson, "The Unified Modeling Language User Guide," 12th Indian Reprint, Pearson Education, Upper Saddle River, 2004.
- [3] G. Booch, J. Rumbaugh and I. Jacobson, "The Unified Modeling Language User Guide," Addison Wesley, Reading, 1999.
- [4] G. Booch, "Object-Oriented Analysis and Design with Applications," 2nd Edition, Addison Wesley, Reading, 1994.
- [5] S. Cranefiel and M. Purvis, "UML as an Ontology Modeling Language," *Proceedings of the Workshop on Intelligent Information Integration, 16th International Joint Conference on Artificial Intelligence*, Barcelona, 16-22 July 1999.
- [6] U. Dayal, "Of Nests and Trees: A Unified Approach to Processing Queries That Contain the Nested Sub Queries, Aggregates and Quantifiers," *Proceedings of 13th VLDB Conference*, Brighton, 1-4 September 1987, pp. 197-208.
- [7] A. Deshpande, S. Nath, B. P. Gibbons and S. Seshan, "Cache-and-Query for Wide Area Sensor Databases," *SIGMOD'03*, San Diego, 8 June 2003, pp. 503-514.
- [8] R. Ganski and H. K. T. Wong, "Optimization of Nested SQL Queries Revisited," *Proceedings of SIGMOD Conference*, Vol. 16, No. 3, 1987, pp. 23-33. [doi:10.1145/38714.38723](https://doi.org/10.1145/38714.38723)
- [9] H. Gomaa, "Designing Concurrent, Distributed, and Real-Time Applications with UML," *Proceedings of the 23rd International Conference on Software Engineering*, Toronto, 12-19 May 2001, p. 829.
- [10] E. Holz, "Application of UML within the Scope of New Telecommunication Architectures," *GROOM Workshop on UML*, Physicaverlag, Mannheim, 1997.
- [11] S. Huaiming, W. Yang, A. Mingyuan, W. Weiping and S. Ninghui, "Query Prediction in Large Scale Data Intensive Event Stream Analysis Systems," *7th International Conference on Grid and Cooperative Computing*, Shenzhen, 24-26 October 2008.
- [12] W. Kim, "On Optimizing a SQL-like Nested Query," *ACMTODS*, Vol. 7, No. 3, 1982, pp. 443-469.
- [13] OMG, "UML Profile for Schedulability, Performance and Time," OMG Document Ptc/03-02-03, Needham, 2002.
- [14] OMG, "UML Superstructure Specification v 2.0," 2005. [http://www.omg.org/cgi-bin/doc? Formal/](http://www.omg.org/cgi-bin/doc?Formal/)
- [15] OMG, "Unified Modeling Language (UML)—Version 1.5," OMG Document Formal/2003-03-01, Needham, 2003.
- [16] OMG, "Unified Modeling Language Specification," 2011. <http://www.omg.org>
- [17] S. Pllana and T. Fahringer, "UML Based Modeling of Performance Oriented Parallel and Distributed Applications," *Winter Simulation Conference USA*, Vol. 1, 2002, pp. 497-505.
- [18] P. M. Kumar, P. Kumarsen and J. Vaideeswasam, "Optimism Analysis of Parallel Queries in Databases through Multicores," *International Journal of Database Management Systems*, Vol. 3, No. 1, 2011, 156 p.
- [19] V. Saxena, D. Arora and S. Ahmad, "Object Oriented Distributed Architecture System through UML," *IEEE International Conference on Advanced in Computer Vision and Information Technology*, Aurangabad, 28-30 November 2007, pp. 305-310.
- [20] V. Saxena and G. A. Ansari, "UML Modeling & Protection of Domain Based System," *International Journal of Computer Science and Network Security*, Vol. 8, No. 7, 2008, pp. 338-344.
- [21] T. Wang, B. Yang, A. Huang, Q. Zhang, J. Gao, D. Yang, S. Tang and J. Niu, "Dynamic Data Migration Policies for Query-Intensive Distributed Data Environments," *AP-Web/WAIM 2009*, LNCS 5446, Springer-Verlag, Berlin, 2009, pp. 63-75.
- [22] W. Liang, B. Chen and J. X. Yu, "Response Time Constrained Top-k Query Evaluation in Sensor Networks," *14th IEEE International Conference on Parallel and Distributed Systems*, Melbourne, 8-10 December 2008, pp. 575-582. [doi:10.1109/ICPADS.2008.65](https://doi.org/10.1109/ICPADS.2008.65)
- [23] S. D. Muruganathan, A. B. Sesay and W. A. Krzymien, "Analytical Query Response Time Evaluation for a Two-Level Clustering Hierarchy Based Wireless Sensor Network Routing Protocol," *Communications Letters, IEEE*, Vol. 14, No. 5, 2010, pp. 486-488. [doi:10.1109/LCOMM.2010.05.091473](https://doi.org/10.1109/LCOMM.2010.05.091473)
- [24] G. C. Wei and L. T. Ming, "A P2P Object-Oriented Database System That Supports Multi-Attribute and Range Queries with Improved Query Response Time," *Information Technology (ITSim)*, 2010 *International Symposium*, Kuala Lumpur, 15-17 June 2010, pp. 1250-1255.
- [25] C. A. Yfoulis, A. Gounaris and D. Tzolas, "Minimization of the Response Time in Parallel Database Queries: An Adaptive Cost-Aware MPC-Based Solution," *19th Mediterranean Conference on Control & Automation (MED)*, Corfu, 20-23 June 2011, pp. 813-818.
- [26] R. Tavenard, H. Jegou and L. Amsaleg, "Balancing Clusters to Reduce Response Time Variability in Large Scale Image Search," *9th International Workshop on Content-Based Multimedia Indexing (CBMI)*, Madrid, 13-15 June 2011, pp. 19-24. [doi:10.1109/CBML.2011.5972514](https://doi.org/10.1109/CBML.2011.5972514)
- [27] Z. Y. Zhang, L. Bin and C. Zhi, "The Research on the Query Optimization on the Distributed Heterogeneous Database Based on the Response Time," *Computer Science and Network Technology, International Conference*, Harbin, 24-26 December 2011, pp. 1541-1544.
- [28] W. Liang, B. Chen and J. X. Yu, "Top-k Query Evaluation in Sensor Networks under Query Response Time Constraint," *Information Sciences: An International Journal*, Vol. 181, No. 4, 2011, pp. 869-882. [doi:10.1016/j.ins.2010.10.006](https://doi.org/10.1016/j.ins.2010.10.006)
- [29] N. V. Murray and E. Rosenthal, "Linear Response Time for Implicate and Implicant Queries," *Knowledge and Information Systems*, Vol. 22, No. 3, 2010, pp. 287-317. [doi:10.1007/s10115-009-0199-x](https://doi.org/10.1007/s10115-009-0199-x)