CONCURRENCY CONTROL MODEL FOR DISTRIBUTED DATABASE

Ranjana Jhala[1], Kamaljit I. Lakhtaria[2] and Sanjay Gaur[3]

[1]*Research Scholar, Faculty of Computer Science, Pacific University, Udaipur, India*

[2,] *Research Supervisor, Faculty of Computer Science, Pacific University, Udaipur, India*

***

Abstract --- ***Nowadays, distributed Databases have taken concentration in the database research. Data distribution and replication offer opportunities for improving performance through parallel query execution and load balancing as well as increasing the availability of data. These opportunities have played a major role in motivating the design of the current generation of database machines.***

The purpose of this study is to propose an approach that absolutely increases the performance of real-time distributed data service. The centralized real time database is based on critical closed loop control system. To overcome this problem we proposed a model/ system called user control distributed database model which try to stimulate the overload transaction during run time. Therefore the present system will be a step towards fulfill the needs of critical conditions like overload or run-time errors in context of distributed database system.

Keywords:D*istributed Databases, parallel, Performance, transaction,* control system, Real time

1. INTRODUCTION

In the today's world the real-time systems are wrapper ample range of technological work like networking, telecommunication, censor monitoring, traffic signals and e-governance etc. The centralized real-time databases are not only needs to read and process the data, but also to proceed in a timely manner without producing unexpected errors. In traditional databases mostly become failure due to work load or resource contention. In this study, we propose a user control system approach for distributed database which tries to reduce run time errors during overload with design quality service during transaction. The base for this study is taken by the model of M. Srinivasan and R. Manohar (2012) and Lam, Kamyiu et al. (1996) about performance of concurrency control Mechanisms in centralized database systems. In recent times, distributed databases have taken attention in the database research area due to demanding distributed data storage. Because data distribution and replication offer opportunities for improving performance through parallel query execution and load balancing as well as increasing the availability of data. These opportunities have played a major role in motivating the design of the current generation of database machines.

2. REAL-TIME DATABASE MODELING AND MANAGEMENT

According to the model of M. Srinivasan and R. Manohar (2012) about performance of concurrency control Mechanisms in centralized database systems. The operation of real-time database modeling and measure the performance of concurrency control system has three modules to manage real-time database functionalities are Temperature Identification (via Serial Communication Application Programming Interface –COMM API, Admin Control and User Control shown in the given diagram.

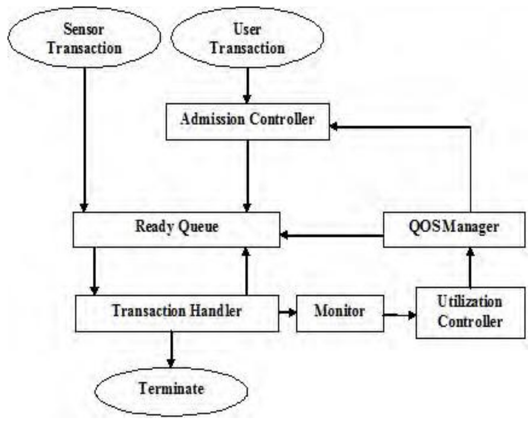

Fig-1 : Real Rime Database modeling

Input / Temperature Identification and Transmission: This module gets the temperature as an input for real-time database. Input is getting by the F2M (female-to-male) cable and transmitted to the admin control. For input transmission serial communication Java API i.e (COMM API) is used. The temperature input will store the current input into the database as well as update it into the admin controller.

Admin Control: The admin control module monitors the temperature as input, and control the request from users. The temperature input is updated into the database as well as the output is monitor into the admin control as Plotted graphs concurrently. When the users sends request for that temperature, first a queue is maintain for that requests. That the queue is altered based on the success ratio. Utilization controller is given for improving success ratio. Then the user gets the output temperature as plotted graphs.

User Control: The user control module user will receive the output plotted graphs for current temperature. The user send request to admin control for getting temperature request. Also the user control will get plotted graphs at specific period of timing interval.

Measurement of QoS

This simulation studies show that the proposed solution satisfies the requirements even during overloads and inaccurate run-time estimation errors. We show that our approach can expect the desire output and considerably enhancing the success ratio. The success ratio is as follows:

$SR = 100 \times (N\ timely / N\ submitted)\ (\%)$

N timely is the number of transactions committed before their deadlines and N submitted is the total number of transactions submitted to the RTDB.

3. DISTRIBUTED REAL-TIME DATABASE MODEL

In the distributed network at each site, the transaction generator generates transactions independently under controlled monitoring. Basically two types of transactions are take place in the network ate local and global. The local transaction accesses local data items, where a global transaction consists of a chain of sub-transactions

If a sub-transaction requests for remote data item, it will be transmitted to a remote site through the communication network, and be processed there. In distributed network both local transactions and sub-transactions have similar CPU and data also they access the same number of data items, and the execution times of the database operation are the same. In view of the fact that main memory database systems can better support real-time applications, it is assumed that the databases are residing in the main memory. By use of main memory database, the impact of different I/O scheduling on the system performance can be eliminated.

At each site, transactions and sub-transactions are scheduled to the CPU by the scheduler based on their priorities. Transactions and sub-transactions that are ready for execution are en-queued into the ready queue as per their priorities. After processing of all the operations of a transaction, the transaction enters in the validation phase. Circular validation is started at the site with the highest site order.

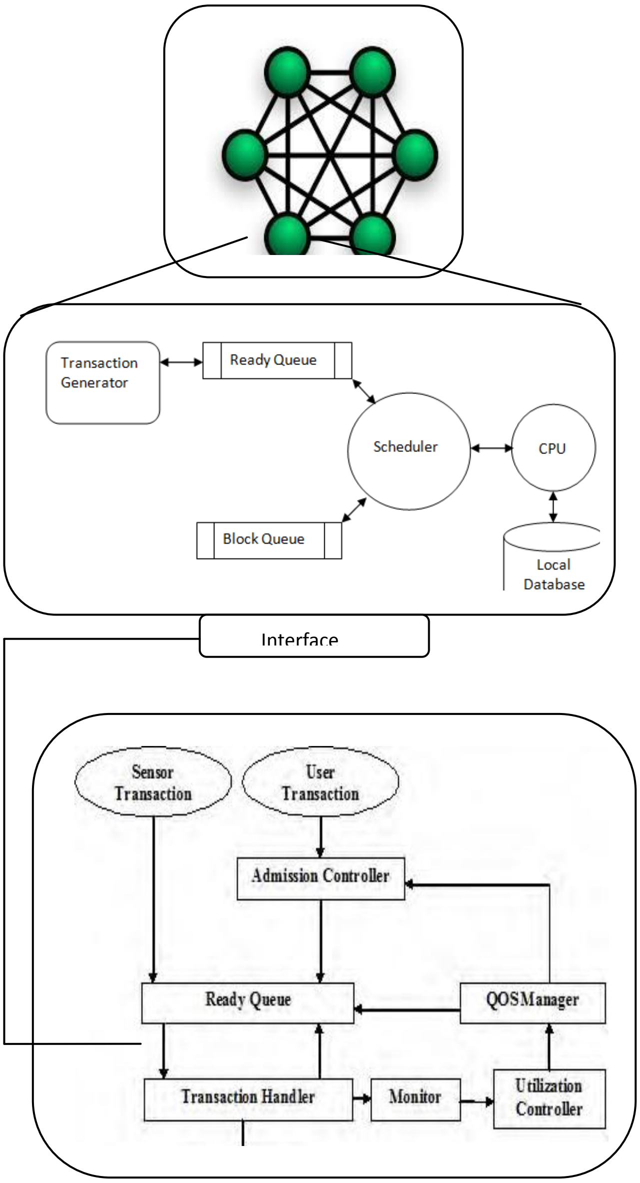

Fig-2: Model for Real Rime Distribited Database Concurrency Control System

At each site, transactions and sub-transactions are scheduled to the CPU by the scheduler based on their priorities. Transactions and sub-transactions that are ready for execution are en-queued into the ready queue as per their priorities. After processing of all the operations of a transaction, the transaction enters in the validation phase. Circular validation is started at the site with the highest site order.

In the online concurrency control, when a transaction accesses a data item in the read phase, the data item will be marked by setting a lock in the lock table to indicate that it is being used by the transaction. Data conflicts are checked by looking at the lock table in each site.

The broadcast commit method is used here to solve the conflict. After the completion of the validation phase, the transaction enters the write phase in which two phase commit protocol and permanent updates of the write operations will be performed. If all the sub-transactions (for global transactions only) are about to commit, the parent transaction will decide to commit. After all the sub-transactions and the parent transaction have committed, the transaction is completed.

The transactions are associated with firm deadlines. Earlier than a transaction is allocated the CPU; the scheduler checks its deadline. If it has already missed the deadline, the transaction is aborted immediately

4. REAL-TIME DISTRIBUTED DATABASE MODELING AND MANAGEMENT

The operation of real-time distributed database modeling and measure the performance of concurrency control system has four modules to manage real-time database functionalities are Global Input Identification, Local Input Identification, Admin Control and User Control shown in the given diagram.

Input (Global and Local) / Temperature Identification and Transmission: This module gets the temperature as an input for real-time distributed database. Input is getting by the F2M (female-to-male) cable/ Network cable and transmitted to the admin control. There is interface interaction to put sub-transaction (global) with the queue of transaction (local). Here priority is decided for each transaction than ready queue get formed. The temperature input will store the current input into the database as well as update it into the admin controller.

Admin Control: The admin control module monitors the input, and control the request from users. The input is updated into the database as well as the output is monitor into the admin control. When the users sends request for that temperature/input, first a queue is maintain for that requests. That the queue is altered based on the success ratio. Utilization controller is given for improving success ratio. Then the user gets the output temperature as plotted graphs.

User Control: The user control module user will receive the output plotted graphs for current input. The user send request to admin control for getting input request. Also the user control will get plotted graphs at specific period of timing interval.

5. TRANSACTION MANAGEMENT

Assuming that, the arrival rates of global and local transactions in a site are A_{glb} and $A_{lcl.}$ There are m sub-transactions in each global transaction. Similar kind of model is used to operate both local transaction and sub-transaction, which requires N_{oprn} number of database operations. Every operation has locking of a data item which takes T_{lock} amount of time and processing of the data which takes T_{prcs} amount of time.

Thus a local or sub-transaction acquirers total processing time is $(T_{lock} + T_{prcs})\ N_{oprn}$ and for a global transaction, m times that amount. Now load of total system is $(A_{lcl} + A_{glb} \times m) \times N_{oprn} \times (T_{lock} + T_{prcs})$ of which a fraction of $A_{lcl} / (A_{lcl.} + A_{glb} \times m)$ is contributed by local transactions. We denote this latter fraction, frac-local. A sub-transaction may access data items in a remote site, in which case, a communication delay of T_{comm} amount of time is incurred before its execution.

The deadline of a local transaction, X_{lcl}, is:

$$\text{Deadline} = ar(x_{lcl}) + (T_{lock} + T_{prcs}) \times N_{oprn} \times (1 + sf)$$

Here SF is the slack factor which is a random variable uniformly chosen from a range

For global transactions, X_{glb} the deadline formula include the network delay:

$$\text{Deadline} = ar(X_{glb}) + ((T_{lock} + T_{prcs}) \times N_{oprn} \times m + T_{comm} \times N_{trnst} \times (1 + sf)$$

where N_{trnst} the number of transit across the network required to access all the remote data.

6. CONCLUSION:

In general the performance of real-time concurrency control system is affected by the method used for assigning the priorities of the transactions. The present study gives a distributed control with various sub-transaction and transaction on network which are based on priority assignment as optimistic concurrency control protocol. The deadline-driven approach is helpful to manage the transaction in distributed real time database concurrency control system. For a little distributed network or private distributed network the system is efficient further scalability factor required to implement it for worldwide implementation.

REFERENCES:

1. Arun Kumar Yadav and Ajay Agarwal , An Approach for Concurrency Control in Distributed Database System, International Journal of Computer Science & Communication, Vol. 1, No. 1, pp. 137-141, 2010.

2. Abbott, R., and Garcia-Molina, H., "Scheduling Real-time Transactions: A Performance Evaluation," *ACM Transactions on Database Systems,* vol. 17, no. 3, pp. 513-60, 1992.

3. Bernstein, P.A., Hadzilacos, V. and Goddman, N. *Concurrency Control and Recovery in Database Systems,* Addison-Wesley, Reading, Mass., U.S.A. 1987.

4. Lam, Kamyiu; Lee, Victor CS; Hung, Sheunglun; Kao, Benjamin CM, Impact of priority assignment on optimistic concurrency controlin distributed real-time databases, Proceedings Of The International Workshop On Real-Time Computing Systems And Applications/Rtcsa, 1996, p. 128-134, 1996.

5. Manohar Srinivasan and Revathi Manohar (2012), Performance of Concurrency Control Mechanisms in Centralised Database Systems, Indian Journal of Computer Science and Engineering, Vol. 3 No.3, pp-492-493, 2012.

