International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-3 Issue-3, February 2014

An Efficient Way to Reduce Data Service Delay Using Map Reduce Framework

S.Thirumagal, A.Safiya Parvin

Abstract— ***The demand of real time data services has increased in various data intensive real time applications such as e-commerce, traffic control, manufacturing etc. Supporting timely data services using such applications is a challenging issue, since the workload may dynamically change based on the current market status. Real time data services are services which has to provide better quality of service to the end users . Most e-commerce clients may tend to leave if the service delay is longer than a few seconds. To overcome this problem, we use the map reduce framework to reduce the real time data service delays. With the existing chronos architecture map reduce framework is used to minimize the delays. The Map Reduce framework processes large amount of data in a parallel way.***

Index Terms—***Map Reduce, Quality of Service, Real time data service***

I. INTRODUCTION

The real time applications such as e-commerce, traffic control, and military applications, need to process queries and transactions in a timely fashion using fresh data that reflects the current market or traffic status. If the service delay is longer than a few seconds, many e-commerce clients may tend to leave the system. Transaction processing with the stale or outdated data may adversely affect the decision making.

Supporting the desired timeliness and data freshness is challenging, since data service requests may arrive in a bursty manner, for example , due to the varying market or traffic status. Our goal is to reduce the data service delay to be below the specific threshold even in the presence of dynamic workloads, while supporting the freshness of data.

In this paper, we propose map reduce framework for reducing real time data service delays. Chronos2 architecture[5], has backlog estimator, admission controller, feedback controller.

The feedback controller however reacts only to performance errors. It takes action control after a performance problem is detected. For example, when an excessive data service delay occurs. To overcome this issue, a predictive and reactive scheme with map reduce function is proposed in this paper.

Manuscript published on 28 February 2014.
***** Correspondence Author (s)
A. Safiya Parvin, Department of Computer Science and Engineering, Sathyabama University, Chennai, India.
S. Thirumagal, Department of Computer Science and Engineering, Sathyabama University, Chennai, India.

II. RELATED WORK

Sha et al [6] have proposed a method to enhance the web service performance by investigating the benefits of integrating the feedback control loop with a queuing theoretic predictor.

The queuing theoretic predictor derives the service delay that is expected is then used to change the service rate dynamically in order to fulfill the delay specifications. In [7], the authors used the M/M/1 queuing model to predict the web server delay. The feedback control is integrated with the queuing model to manage the web server delay. However, queuing theory is not very effective to model real-time database workloads which arrives in a bursty manner.

Henriksson et al [8], developed an improved feed-forward scheme. Without an assumption about the statistics of the incoming workload, this scheme replaces the queuing model with a predictor which uses a novel method to predict about the future delays. Based on the predicted delay in the closed loop, the web server adapts its resource allocation. To manage the CPU utilization in multiprocessor environment model predictive control technique is applied. However, database specific issues such as integrated data service backlog adaptation and fine-grained data access delay prediction are not considered by these approaches.

The Backlog Estimation[1] deals with the estimation of the backlog. That is the amount of the data need to be processed by looking up the data service requests in the queue and by considering not the length of the queue rather the amount of data. The feed forward predictor predicts the database backlog bound at (k-1)th sampling period for kth sampling period.

Y.Zhou et al [3] have proposed the method of integrating proactive and reactive approaches for robust real time data services. They have developed a feed forward approach that proactively adapts the incoming load and if necessary to support the delay threshold. Further, they have integrated it with a feedback controller to compensate potential prediction errors and adjust the system behavior for timely data services.

Various approaches have been made to reduce the delays in real time data service applications, but the integrated predictive and reactive method [11] seems to be efficient in reducing the delays. The predictive method predicts the backlog bound used for the next sampling period in the current sampling point. The sampling period is a fixed time, for example it may be 1s. This approach says that, whatever the situation may be, the server has to provide the services at a stretch. That is, even if there is any overload conditions occur the server should have the capability of providing quality of service to the end users.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

III. PROPOSED SYSTEM

Map Reduce framework along with the chronos architecture is proposed to reduce the data service delay effectively. It has two functions called map and reduce. The map phase is also called as data collection phase. It collects the data and breaks the large chunks in to smaller ones and takes action on each chunk. The reduce phase is also called as processing phase or data collation phase. It combines many results from the map phase to form a single output.

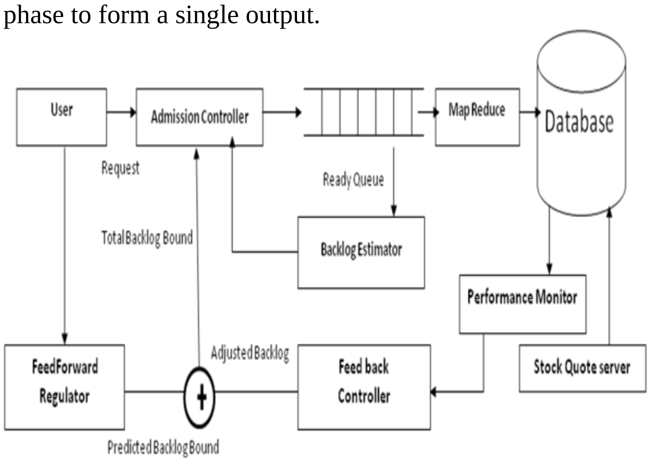

Fig 1: Proposed Architecture

Figure 1 shows the proposed architecture of the paper. In this architecture we added the Map Reduce to the Chronos2 architecture [11] to provide better QoS in real time data services. It has database backlog estimator, admission controller, feedback controller, feed forward regulator. The backlog estimator estimates the amount of data to be processed by the database server by looking up into the queue rather than the queue length. The feedback controller is designed based on either linear control theory or fuzzy logic control.

The database server processes service requests and updates stock prices periodically received from the stock quote server to provide the updated stock prices. We consider periodic temporal updates that are commonly used in RTDBs for data temporal consistency and a fixed time interval is selected for updating each stock price in a range [0.2 s, 5 s]. A random data generator is used for updating the stock prices in a fixed time interval. The prices are updated in random manner. Real prices are not important here, as we are not developing a stock market application. To support the data freshness, prices are updated. The stock prices and names can be downloaded from yahoo finance[4]. It provides the historical data for all stocks.

The backlog estimator estimates the amount of data to be processed for the requests in the ready queue. The backlog bound can be calculated by subtracting the number of data accesses that the database is currently processing from the total number of data accesses that the database should actually process. By deriving this[11], the feed forward regulator can predict the backlog bound at kth sampling period itself to use it for (k+1)th sampling point. The feedback controller [5] based on control theory continuously adjusts the database backlog bound due to performance errors.

The summation of backlog bound predicted by feedforward regulator and adjusted by feedback controller should be greater than the current backlog bound after admitting the new request. If the total backlog bound is less than the current backlog bound after admitting the request, the request is rejected and a busy message is sent to the user. There by we can reduce the overload condition before they occur.

The feedforward regulator takes the value of the backlog from the previous sampling period. The current backlog will be used for the next sampling period. Thus in terms it is called as feedforward prediction, which is predicting priorly. Whenever a request arrives, the map function takes the value and key pair dynamically and produces the intermediate result. The reduce function takes the intermediate result produced by map function and produces a single output.

This can be implemented using three identical computers. One is a server, another one is the client machine, and another one is the stock quote server. The stock quote server updates the stock prices at a fixed interval continuously whenever the server is started. This gives the fresh data to the users as it is mandatory in real time applications. The client machine which has the application will send the requests to server. The requests such as buy share, sell share, view portfolio, view share prices as we are using the stock market database. The server updates the share prices and responses the requests in parallel.

Compared to the existing system, the proposed system technique can minimize the service delays under overload conditions by supporting the freshness of data. It improves the quality of service compared to the existing system. If the quality of service is not guaranteed in the real time data services, the end users may ignore the system. Fresh data is also very important. Stale or outdated data may degrade the quality of service. Quality plays a major role in real time data services.

IV. CONCLUSION

An efficient way to reduce data service delay is proposed. In future, we will further investigate more effective approaches for real-time data services in dynamic environments and by improving the request load time using Minification and Bundling techniques.

REFERENCES

1. K.-D. Kang, J. Oh, and Y. Zhou, "Backlog Estimation andManagement for Real-Time Data Services," Proc. 20th EuromicroConf. Real-Time Systems, pp. 289-298,2008.
2. N.Y.Times, "Stock Traders Find Speed Pays,in Milliseconds,"http://www.nytimes.com/2009/07/24/business/24trading.html,2012.
3. Y. Zhou and K.-D. Kang, "Integrating Proactive and Reactive Approaches for Robust Real-Time Data Services," Proc. 30th IEEE Real-Time Systems Symp., pp. 105-114, 2009.
4. "Yahoo! Finance," http://finance.yahoo.com, 2012.
5. K.D. Kang, J. Oh, and S.H. Son, "Chronos: Feedback Control of a Real Database System Performance," Proc. 28th IEEE Real-Time Systems Symp., pp. 267-276, 2007.
6. L. Sha, X. Liu, Y. Lu, and T. Abdelzaher, "Queueing Model BasedNetwork Server Performance Control," in IEEE Real-Time SystemsSymporisum, December 2002.
7. Y. Lu, T. F. Abdelzaher, C. Lu, L. Sha, and X. Liu, "Feedback Control with Queueing-Theoretic Prediction for Relative Delay Guarantees in Web Servers," in IEEE Real-Time and Embedded Technology and Applications Symposium, 2003.
8. D. Henriksson, Y. Lu, and T. Abdelzaher, "Improved Prediction for Web Server Delay Control," in Euromicro Conference on Real-Time Systems, 2004.
9. J. L. Hellerstein, Y. Diao, S. Parekh, and D. M. Tilbury," Feedback Control of Computing Systems," A John Wiley and Sons, Inc., Publication,2004.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

10. K. Ramamritham, S. H. Son, and L. C. Dipippo, “Real-Time Databases and Data Services,” Real-Time Systems Journal, vol. 28, Nov.-Dec. 2004.
11.] J. Oh and K.D. Kang, “A Predictive-Reactive Method for Improving the Robustness of Real-Time Data Services,” IEEE Transactions on Knowledge and Data Engineering, vol. 25, no. 5, pp. 974-986, 2013.