

Dynamic Environment and Snowflake Schema in Real Time Data Services

S. Aswini,
M.E., Computer Science & Engg. ,
S.K.P Engineering College
Thiruvannamalai,
Tamil Nadu, India

D. Murali, M.E.
Asst. professor, C.S.E.
S.K.P Engineering College
Thiruvannamalai,
Tamil Nadu, India

R. Selvam, Ph.D
Professor & Head, C.S.E.
Arignar Anna Institute of Science & Tech.
Pennalur, Sriperumbudur,
Tamil Nadu, India

ABSTRACT

The demand for real-time data service is increasing in many applications such as e-commerce, agile manufacturing, telecommunications, network management and transportation management. It is desirable but challenging, the workload may vary dynamically. In this paper we propose an approach for managing the dynamic environment using snowflake schema for real time database, that the dynamic environment operates an unpredictable environment such as they cannot predict accurate client request the stock quote at a moment of time. So they can implement dynamic workload adapting technique to improve the freshness of data in real-time and implement the technique called big bang adoption technique support the changes which are needed for a real changing process start .By Modeling database dynamic workload adoption in terms of the relation between data service delay and Conglomerate estimation. They also provide Quality Of Service to data guarantees for real-time data services. compare to most existing system feedback control representing , our dynamic snowflake schema significantly improves the average throughput.

Keywords

Big-bang adaption, snow flake schema, conglomerate.

1. INTRODUCTION

In recent year scrupulous for real time data service is increasing in many applications such as e-commerce, traffic monitoring, and military prosecution, need to process queries and transaction timely manner with the freshness of the data that muse on the current market or traffic status [3], [2]. In e-commerce, for example, stock quote queries and trade processing should be processed within the kosher response time bound using up-to-date stock prices. If most online retail customer wants the ritual time less than few second else most of customer be verge and go away from the system. When Online contract anchored in fusty data i.e.,out-of-date stash prices, It unfavorably to online-processing. Processing in a timely manner using data in freshness to representing the current traffic status. Data Stream Management Systems (DSMS) [1], [5], [4], [17], it produces high dispensation of torrent data . Generally , RTDBs occasionally update profane data, e.g., Online-stock prices, to support the proven reliability of data within their trusted intervals [2], [3]. DSMS [5], [6], [7], [8] that data are coming in a rupture comportment and implement them are connected in sequence of inquiry upon data influx.

Sustaining the preferred well-timed data is critical to achieve, because real-time data ritual requests may arrive in a spurt manner, i.e., Changing stock query status. So this kind of action

will be overcome further section. In this paper we proposed snowflake schema is multidimension database .It also provides many to one relation i.e., Many client to one database. Speed of data retrieval is very high , it resembles a snowflake in shape. **The drawback of star schema:** It will be a single dimension database. Also it does not posses data integrity.

They use snowflake schema in stock quote queries and trade transaction its very useful because it as many to one relationship i.e., Many client request saved in a single database. So that reduces the storage disk.

We also proposed a technique called big-bang adaptation technique .They consider the method called metadata its defined as data about data , metadata are traditionally found in the stock queries and trade transaction in turn has gone to be inflated clone, metadata are also used to describe clone data using metadata standard specific to a particular discipline. e. g. Allowing the browser to automatically improve the experience of users. Usage of metadata in statistics and census services . Thier aim to improve the experience of users. Usage of metadata in statistics and census services . Also improve the freshness of data, has been identified as one of the most important attributes of data quality for data consumers (Shin, 1996). Some surveys and empirical studies have proved that data freshness is linked to information system success (Wang et al., 1996) (Shin, 2003). Then, achieving required data freshness is a challenge for the Development of a large variety of applications . Furthermore, the increasing need to access to information that is available in several data sources introduces the problem of choosing between alternative data providers and of combining data with different freshness values. In our experiments, dynamic environment and snowflake schema reduce the average service delay, it will be shown in our performance results. Dynamic environment and snowflake approach for real-time data services minimum CPU utilization for example 3 percent in total.

Our work is summarized as follows: The architecture real time data services, snowflake scheme, analysis of data freshness in real time environment are discussed in part 2. Analysis the data freshness in section 3.description of metadata in section 4. Conglomerate Estimation in section 5 . Performance evaluation and their results in section 6 . At the end conclusion of our work.

2. ARCHITECTURE REAL TIME DATA SERVICES AND SNOWFLAKE SCHEME

In this section, an overview of the architecture and timely data services is given. Also data freshness and their metrics with an

example of analysis of data freshness used for performance evaluation in this paper are explained.

Fig. 1 shows the architecture of real time data service with snowflake scheme. It consists of eight modules.

2.1 System overview

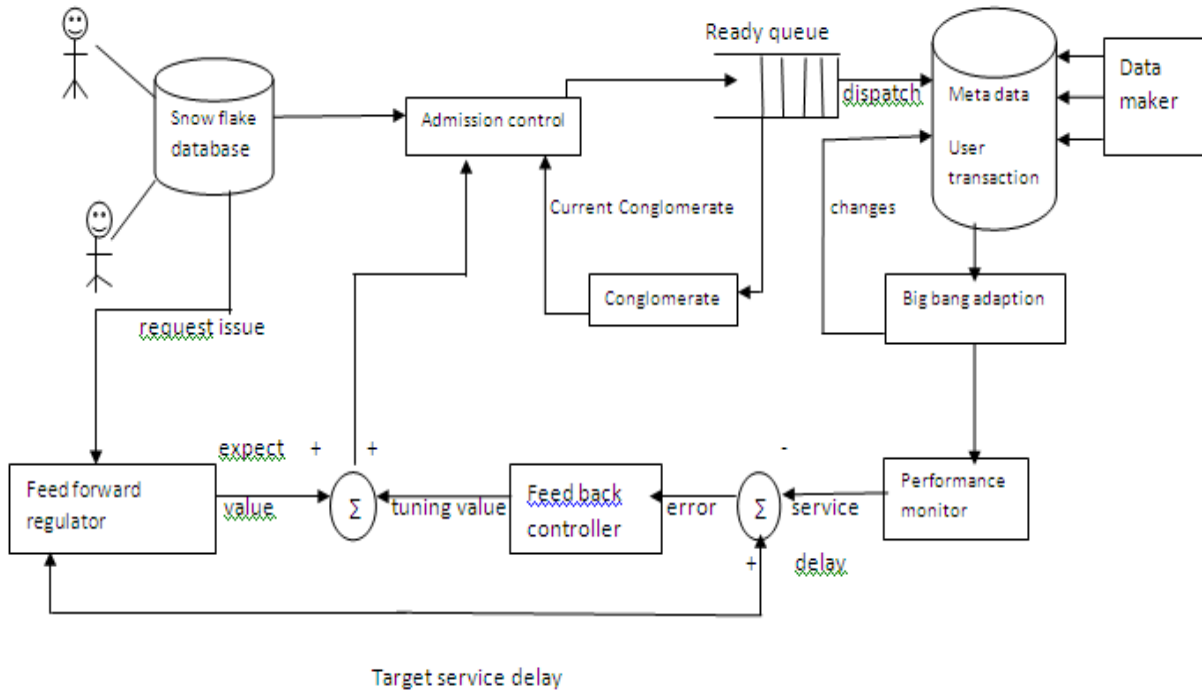


Fig 1. Snowflake scheme in real time database

2.1.1 Snowflake schema,

2.1.2 Admission controller,

2.1.3 Conglomerate

2.1.4 Metadata in database server monitor,

2.1.5 Big bang adaption ,

2.1.6 Performance monitor,

2.1.7 Feedback controller

2.1.8 Feed forwarded regulator

2.1.1 Snowflake schema

It estimates number client to be stored in a specific manner. The snowflake schema is a multi dimension database it as one to many relationship already explains that , i.e., Many client requests to stored in one database.

Normalizing attributes results in storage savings, the tradeoff additional in the source query joins [10].

Define dimension order as (order no, order date, brand, type, supplier type)

Define dimension Sales person as (sales person ID, Sales person's name, order type)

Define a dimension City as (City name, street, province or state, country)

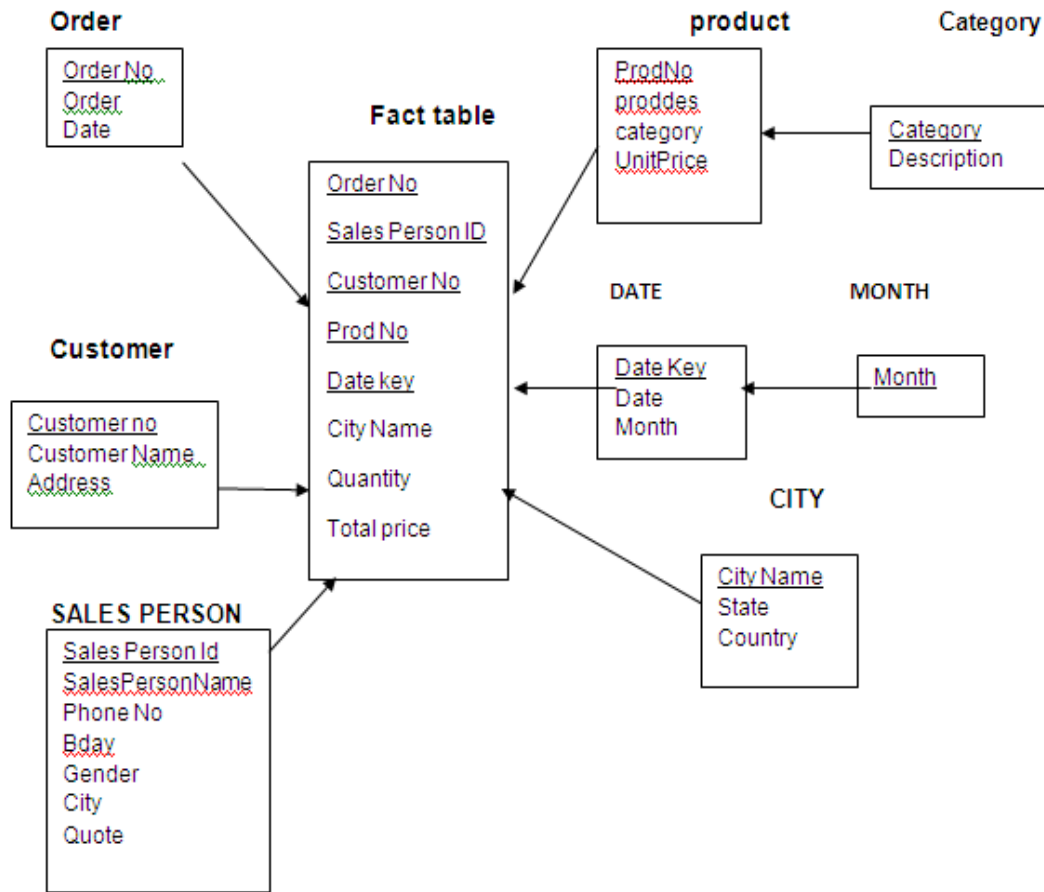


Fig 2. Snow flake database

Define cube product [time, category, product, from the city, to city]:

Dollars cost = sum (cost in dollars), units price = count (*)

Define dimension time as time in cube sales

Define dimension product as a product in cube sales

Define dimension product as (product no, product name, city as a city in cube sales, production des)

A defining cube statement is used to define data cubes for product and category corresponding to the fact tables of the schema.

Example1.1

An online order Software company requires the designing of a record the quantity and sales of its wines to its customers. Part of the original database is composed of the following tables:

CUSTOMER (Customer no, Customer Name, Address, Phone no, BDay, Gender)

Software (prodno, prodName, category, softwarePrice, Class)

CLASS (Code, Name, Region)

TIME (TimeStamp, Date, Year)

ORDER (Customer no, Software, Time, order date)

Note: that the tables represent the main entities of the ER schema,

Thus it is necessary to derive the significant relationships among them in order to correctly design .

FACT Sales

MEASURES Quantity, Cost

DIMENSIONS Customer, Area, Time, Software→Class

Benefits of snowflake schema:

- 1) Sequence of query processing is very high i.e.,Process the query with few second.
- 2) Zero repetition and maintain cost will to low.
- 3) Increasing the system performance since it requires a small table to join together

2.1.2 Admission control

The admission control allows or cancel an arriving data ritual request, if the overall sum of to calculate total amount of data to be accessed by present database conglomerate could not beyond the conglomerate bound computed by the feedback and feedforward (F3 B) controller [9]. Admission control uses the sampling period be the time interval $[(n-1)sp, nsp]$ sp -sampling period. The n^{th} sampling period is equal to nsp . Let $k(n)$ be the number of the data service request i.e., Queries and requests, processed in the n^{th} sampling period.

2.1.3 Conglomerate estimator

The Conglomerate estimator calculates the cumulative value of data and conglomerate has to process the data ritual requests. For example stock queries, in the online system. Detail is explained in section 4.

2.1.4 Metadata in database server

Metadata in database server provides an action such as processes the ritual request on an online stock value query for the transaction. Also it takes an action such as periodically refresh the data freshness on the stock value.

2.1.5 Big-Bang adaption

Once the management has decided to apply the big bang adoption method and maintain the modification

It will be necessary for the real changing process can begin. This changing process was followed stage:

Renovate the logic from one phase to next phase,

Issues function of the system in logic manner and

Teaching the upcoming client (Eason, 1988). I.e., Any update is required for stock queries

2.1.6 Performance monitor

It contains the following fields:

1. Unique id of the stock queries.
2. The id of the stock queries on which the I/O operation occurred
3. Duration of the operation in seconds
4. The id of the stock operation refers to stock queries
5. Number of bytes written by the operation
6. Number of variables passed to the I/O stock queries
7. Store value used by the bib-bang adaption that recorded this data.

2.1.7 Feedback control

It provides for timely data services in following manner [1] [16].

- 1 Its relation between Conglomerate and data ritual delay.
- 2 Plan and tweak the feedback controller.
- 3 Evaluate the occurrence. If the preferred occurrence is not maintained, go back the beginning steps.

2.1.8 Feed forward regulator

Feed-forward is a constituent. Which allow stock value from the external environment to database server. The request will be load to feed forward system that will be controlling the value.

Advantages of Feed forward regulator

- 1 Feedforward regulator will be reduce the extra amount to be process the stock Value.
- 2 Accuracy of the system can be increased.
- 3 Power utilization of feed-forward system compare to other regulator that will be consumed very low power. It also increases the overall performance.

2. ANALYSIS THE DATA FRESHNESS

Performance to use of statistics to estimate the overall Quality of Data after adapting the materialization plan. They will use the term *materialization plan* to refer to any solution to the Online View Selection problem. [10]. Let us assume that

$$N \text{ fresh } (W_j | P_i)$$

Is the number of fresh accesses to stock quotes and transaction processing W_i why

Originated from requests to page P_j . The overall QoD Quality of Data definition can be written as follows:

$$QoD = \frac{1}{b} \times \sum_{n=y}^{y+b-1} f(xk)$$

b -number of access request from client side

$f(xk)$ - the freshness value of data base server

$$f(xk) = f(pi, time) = \sum_{i=1}^{bj} a_{i,j} \times f(W_j, time)$$

Where $pi, a_i W_j$ weight factor.

3. META DATA

3.1 Data description

Meta data is defined as data about data in a database server. In stock trading center when online transaction will be acquired, then the data will be stored in a database. So the data will be fresh to the client. A client could be see the up-to-date data note stale data.

3.1.1 Main components

Stock price in trade center will be recorded in every month base of arrivals and outgoing of transaction i.e., Trade in and trade out between the database. Then the transaction will be placed in database server [9].

Metadata provides fundamental information to the user / client:

Concentration of metadata will be Quality assessment in that they provide reverence of a stock quote, Timeliness and punctuality of data.

The following basic information : provided by the metadata

Reporting country, reference period, trade flow, product, trading partner , mode of transport.

4. CONGLOMERATE ESTIMATION

Conglomerate estimation evaluate the stock query and transaction processing based on the online trade in visit stock price, visit the range, procure goods based on the stock quotes .

4.1 Visit-stock quote

Stock values based on the information provided by the software company and that value connected to stock quote prices/current stock value. To manage this request, the database server require to access CURRENT STOCKS and ESTIMATE tables that will be embraced <current stock symbol, full soft/hardware company name, soft/hardware company ID> and <soft/hardware company ID, recent stock price> for each company.

$$c_j = no.a.\{a(current\ Stocks) + a(Estimates)\}$$

4.2 Visit-range

In this section clients wants to see about certain stock price in that range.

$dic_j = |range(id)|.\{a(range) + a(Estimates)\}$
i.e., $|range(id)|$ Is to represent the number of stockpile available in the range hold by client for example whose ID is the id .

4.3 Procure

When a customer wants to buy a current item, then the database server acquires current stockpile from ESTIMATE table. If the buy a goods was not in the range before they acquire, the stockpile and Its range of the price will be added to the range of the current stockpile i.e., The update will be done in the database server using our big-bang adaptation technique. Then the range processing will be shown below

$$c_j = a(Estimates) + (|range(id)| + 1).a(range)$$

Trade. A client wants to be put for sale of his stockpile in online sale processing. The client just identifies the total number of stockpile $w_{j,sell}$ That the client will be selling. $w_{j,sell} \leq |range(id)|$, then the sale transaction will be shown below,

$$c_j = |range(id)|.a(ranges) + w_{sell}.a(Estimates) + w_{sell}.a(ranges)$$

5. PERFORMANCE EVALUATION

In this part system experimental will be shown then the F3B performance will be compared to open, CLT+FB.

5.1 Experimental Settings

5.1.1 Open

Open will be referred to Berkeley database [15]. It will accept all arriving request not included in feedback and feed forward regulator.

5.1.2 FB

FeedBack- It will be based on the admission control scheme .FB provides timely data services [11], [13], [12], [14].

5.1.3

CLT+FBControl +FeedBack. It will combine the statistical feedforward [7] approach it will refer the center limit theorem [8]. It will be alter the database server in a timely manner to the freshness of the data.

5.1.4

WorkloadsIt will contain the number of clients, number request per second. The database servers to be receiving the client request. They will consider three workloads namely W1, W2, W3.

Table 1. Workload for client request per second for testing approach

Workloadad	Number of clients	Numberofrequest/Sec $0_s \sim 300_s \rightarrow 301_s$ $\rightarrow 1000_s$
W1	1200	[300,360] \rightarrow [2,400, 12100] ($5L_H$)
W2	1600	[380,430] \rightarrow [3500,1600] ($7.35L_H$)
W3	2100	[530,680] \rightarrow [4,800, 22000] ($9L_H$)

5.2 Performance Results for Workload 1

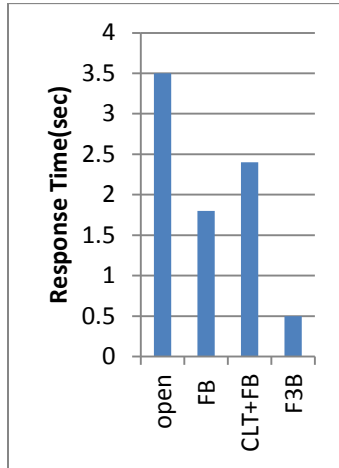


Fig 3. Average service delay for W1

In this paper measure the average service delay between 301 to 1000 second for W1.

Step 1: consider the Fig 3. Open provide service delay 3.5 second. This because it accepts all incoming request then the result produce , the system will be overloaded.

Step 2:Feed back provide service delay 1.8 second. It will be fluctuates between 1.5 and 2 second.

Step 3: CLT+FB [7] it provides service delay greater than FB at 2.2second. But we will want a lower service delay.

Step 4: F³B (feedforward and feed regulator) provides lower service delay that is 0.5second this will be preferable compared to others.

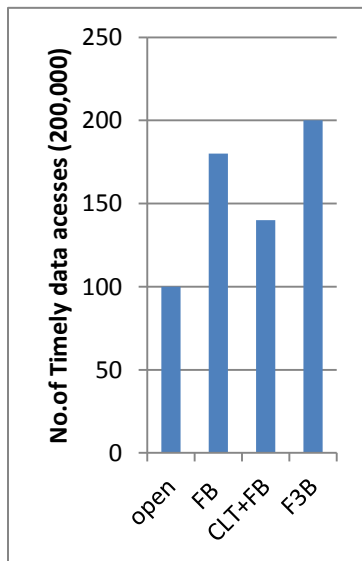


Fig 4. Timely throughput for W1

Step 1: consider Fig 4. open produce throughput 1 lakh clients per second.

Step 2: FB produce throughput 1,80,000 lakh clients per second.

Step 3: CLT+FB produce 1,40,000 lakh clients per second.

Step 4: F³B produce 2,00,000 lakh clients per second this our work. It is benefit than most existing systems.

6. CONCLUSION

In this paper, their present a new dynamic and a snowflake scheme to support timely data services even in the presence of dynamic workload .The system consists of eight modules each of with own activity and produce work is different from most existing work on real-time data management in that it applies snowflake to enhance the freshness of data .Analysis data freshness produces better throughput than most existing systems.Conglomerate estimation provides better output i.e The information about the schema and query types in meta data in data base benefits is always available.

7. REFERENCES

- [1] " Exploratory Stream Processing System," http://domino.research.ibm.com/comm/research_projects.nsf/pages/esps.index.html2012
- [2] K. Ramamiritham,S.H.Son, and L.C.Dipippo,"Real-Time Databases and data Services"
- [3] Real-time Database Systems, K. Y. Lam and T. W. Kuo, Eds . Kluwer Academic Publishers, 2006.
- [4] "StreamBase ." <http://www.streambase.com/>,2012.
- [5] "Microsoft SQL Server 2008 R2 - StreamInsight," <http://www.microsoft.com/sqlserver/2008/en/us/r2-complex-event.aspx>,2012.
- [6] G.F. Franklin, J.D. Powell, and M.L. Workman, Digital Control of Dynamic Systems, third Ed. Addison-Wesley, 1998.
- [7] Y. Zhou and K.-D. Ken, "Integrating Proactive and Reactive Approaches for Robust Real-Time Data Services," Proc. 30th IEEE Real- Time Systems Symp., Pp. 105-114, 2009.
- [8] J.L. Devore, Probability and Statistics for Engineering and the Sciences sixth ed. Thomson Learning, Inc., 2004.
- [9] <file://g.journal/metadata.html>.
- [10] Exploring the Trade-off Between Performance and Data Freshness in Database-DrivenWeb Servers, Alexandros Labrinidis, Nick Roussopoulos.
- [11] K.-D. Kang, J. Oh, and Y. Zhou, "Backlog Estimation and Management for Real-Time Data Services," Proc. 20th Euromicro Conf. Real-Time Systems, pp. 289-298, 2008.
- [12] K.D. Kang, S.H. Son, and J.A. Stankovic, "Managing Deadline Miss Ratio and Sensor Data Freshness in Real-Time Databases,"IEEE Trans. Knowledge and Data Eng., vol. 16, no. 10, pp. 1200-1216,Oct. 2004.

- [13] M. Amirijoo, N. Chaufette, J. Hansson, S.H. Son, and S.Gunnarsson, "Generalized Performance Management of Multi-Class Real-Time Imprecise Data Services," Proc. 26th IEEE Int'l Real-Time Systems Symp., pp. 38-49, 2005.
- [14] 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.
- [15] "Oracle Berkeley DB Product Family, High Performance, Embeddable Database Engines," <http://www.oracle.com/database/berkeley-db/index.html>, 2012.
- [16] J.L. Hellerstein, Y. Diao, S. Parekh, and D.M. Tilbury, Feedback Control of Computing Systems. John Wiley & Sons, 2004.
- [17] D.J. Abadi, Y. Ahmad, M. Balazinska, U. Cetintemel, M.Cherniack, J.-H. Hwang, W. Lindner, A.S. Maskey, A. Rasin, E.Ryvkina, N. Tatbul, Y. Xing, and S.B. Zdonik, "The Design of the Borealis Stream Processing Engine," Proc. Conf. Innovative DataSystems Research (CIDR), 2005.