Simulation Of Diagonal Data Replication In Mesh

Rohaya Latip[a], Hamidah Ibrahim[b], Mohamed Othman[a], Md. Nasir Sulaiman[b], Azizol Abdullah[a]

[a]Department of Technology Communication and Network
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia, 43400 Serdang, Selangor
Tel: 03-89466536, Fax :03-89466577
E-mail: {rohaya, mothman., azizol}@fsktm.upm.edu.my

[b]Department of Computer Science
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia, 43400 Serdang, Selangor
Tel: 03-89466510, Fax :03-89466577
E-mail: {hamidah, nasir}@fsktm.upm.edu.my

ABSTRACT

In a large dynamic network, data can be copied anywhere to make it fault tolerant and easy accessed but there must be an efficient protocol to manage the replicas and make sure the data is consistent and high in availability with a low communication cost. In this paper, we introduced a new protocol, named Diagonal Replication in Mesh (DRM) for data replica control protocol for a large dynamic network by using quorum and voting techniques to improve the availability and the communication cost because quorum techniques reduce the number of copies involved in reading or writing data. The protocol of DRM replicates data for large dynamic network by putting the protocol in a logical mesh structure and access consistent data by ensuring the quorum not to have a nonempty intersection quorum. To evaluate our protocol, we developed a simulation model in Java. Our results proved that DRM improves the performance of the response time compare to Three Dimensional Grid structure Protocol (TDGS).

Keywords
Data Replication, Mesh Network, Response Time, Replica Control Protocol

1.0 INTRODUCTION

The emerging technologies of computer network and database make distributed database important therefore data can be shared and easily accessed. To allow data to be more accessible especially in large dynamic network, some techniques are introduced and implemented, among which is data replication. Data replication is an effective measure to access data in a geographically distributed environment (Venugopal, Buyya, & Ramamohanarao, 2005) because of identical copies of data are generated and stored at various globally distributed sites. Significantly this has reduced the data access latencies (Lamehamedi, & Szymanski, 2007). Data replication is also used in distributed system to increase data availability and achieve fault tolerance but having data replication affecting the consistency of data when the data needs to be updated. Also, having a number of replicas in the network increases the communication cost. Therefore, an efficient protocol must be proposed to cater these issues.

Previously a replica control protocol named Three Dimensional Grid Structure (TDGS) was introduced by M. Mat Deris (2008). TDGS is a similar technique of replica control protocol called Sense of Direction Approach (SODA) proposed by Abawajy (2005). It is a protocol replicating its data or copies in a logical 3D box shape structure with four planes. Figure 1 illustrates eight copies of a data object.

Read operations in TDGS are executed by acquiring a read quorum that consists of any hypotenuse copies. For the example shown in Figure 1, hypotenuse copies are {A, H}, {B, G}, {C, F}, and {D, E}. The read operations are executable by accessing one of these pairs of hypotenuse copies.

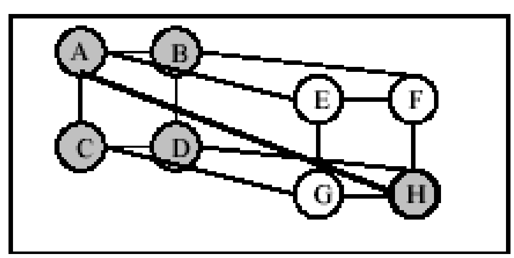

Figure 1: Eight Copies of a Data Object

Write operations are executable from any planes that consist of hypotenuse copy. Planes in Figure 1 are {H, A, B, C, D}, {C, F, E, G, H}, and etc. Example, to execute read operation, copies from {A, H} must be accessible and to execute write operation, copies from {H, A, B, C, D} must also be accessible.

Read operations are executed by acquiring a read quorum, which must come in pairs example {A, H} at the vertices of the box shape structure. If one of the copies of each pairs is unavailable, thus that hypotenuse copies are not accessible. Meanwhile, write operations are not executable at that plane. Read and write quorum must intersect; if the edge of read quorum is not accessible thus the write quorum is also not accessible to update the latest data. Therefore this has affected the consistency of the data.

Other limitation of this protocol is that, it must be formed in a perfect square box and if new copies are added, thus more copies are added at the other planes of the box to get a perfect square box. This some how has increased the read and write quorum size and increased the communication cost.

The paper is organized as follows. Section 2 introduces the Diagonal Replication in Mesh (DRM) protocol as a new replica control protocol. In Section 3, we present the latest protocol in replica control protocol, NRG including the formulation for communication cost and read/write availability. Section 4 discusses the simulation design and implementation of DRM. Section 5 describes the results comparing DRM with TDGS and ends with a conclusion and future work.

2.0 DIAGONAL REPLICATION IN MESH (DRG)

Diagonal Replication in Mesh (DRM) protocol organized all nodes logically into two dimensional mesh structured. By assuming that the replica copies are in the form of data file and all nodes are operational meaning that the copy at that node is available. DRG uses quorum technique and voting technique.

Quorum technique is grouping the nodes or sites. There are two types of quorum that are the read quorum and the write quorum (Kumar, 1991). Some research (Mat Deris et al., 2004; Ahmad et al., 2006; Ahmad et al., 2008) define quorums, Q as group, G which has the similarity of coterie. The definition of quorum is in Definition 5.2 can also be defined as coterie. The definition of coterie is as in Definition 1.

Definition 1: Coterie. Let U be set of groups that computes the system. A set of groups T is a coterie under U iff

i. $G \in T$ implies that $G \neq \emptyset$ and $G \subseteq U$

ii. If $G, H \in T$ then $G \cap H \neq \emptyset$ (intersection property)

iii. There are no $G, H \in T$ such that $G \cap H$ (minimality)

The data file is replicated to only one selected node from the diagonal sites of the quorum.

Firstly a number of nodes are grouped in a quorum and a selected replica, *R* is chosen in each quorum. The number of columns, *n* and rows, *n* must be odd to get the only middle node in the diagonal sites as shown in Figure 2. Based on the example, the selected node, *R* is the gray nodes where the data file is copied.

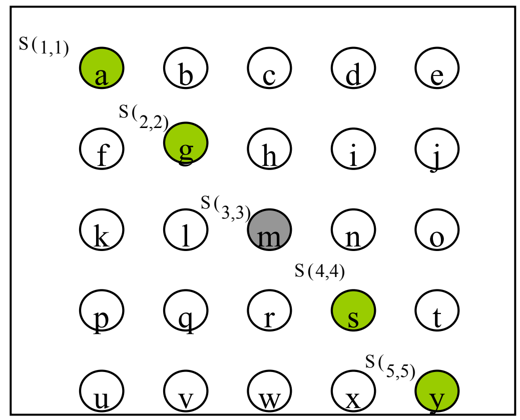

Figure 2 : Diagonal Sites of 5 x 5 Network Size

Definition 2: Assume that a database system consists of *n* x *n* nodes that are logically organized in the form of two dimensional grid structures. All sites are labeled $s(i,j)$, $1 \leq i \leq n$, $1 \leq j \leq n$. The diagonal site to $s(i,j)$ is $\{s(k,l) | k = i + 1, l = j + 1$, and $k, l \leq n\}$. A diagonal set, $D(s)$, is a set of diagonal sites, $D(s) = s(m,n)$, where $m = n$ and $m, n = 1, 2, \ldots, n$. In Figure 2, $n = 5$ in each quorum. This figure has 25 copies where the network size is 5 x 5 nodes.

The selected node in the diagonal site is assigned with vote one otherwise assign selected node as vote zero. A vote assigned on grid, *B*, is a function such that,

$$B(s(i,j)) \in \{0, 1\}, 1 \leq i \leq n, 1 \leq j \leq n$$

where $B(s(i,j))$ is the vote assigned to site $s(i,j)$. This assignment is treated as an allocation of replicated copies and a vote assigned to the site results in a copy allocated at the diagonal site, $D(s)$. That is,

1 vote ≡ 1 copy.

Let $L_B = \sum B(s(i,j)), \quad s(i,j) \in D(s)$

where L_B is the total number of votes assigned to the primary node. Thus L_B is equal to 1 in each quorum.

Theorem 1: DRM protocol accessed consistent data.

Proof: To ensure that read operation always gets the updated data, $r + w$ must be greater than the total numbers of copies (votes) assigned to all sites. To make sure the consistency is

Subquorum 3

obtained, the following calculation shows the DRM fulfill the conditions of quorum intersection property.
Example, if there are four quorums for read and four quorums for write, thus L_B is 7 as shown below in the first condition:

$$r + w = L_B + 1$$
$$L_B = r + w - 1$$
$$= 4 + 4 - 1$$
$$L_B = 7$$

The second condition is fulfill where r and w is equal to 4 and L_B is equal to 7 as shown below,

$$1 \leq 4 \leq 7,\ 1 \leq 4 \leq 7$$

Let $S(B) = \{s(i,j) | B(s(i,j)) = 1, 1 \leq i \leq n, 1 \leq j \leq n, i = j\}$

Definition 3: For a quorum q, a quorum group is any subset of $S(B)$ where the size is greater than or equal to q. The collection of quorum group is defined as the quorum set. Let $Q(B,q)$ be the quorum set with respect to assignment B and quorum q, then

$$Q(B,q) = \{G | G \subseteq S(B) \text{ and } |G| \geq q\}$$

For example, from Figure 2, let site $s(3,3)$ be the primary site of the master data file m. Its diagonal sites are $s(1,1)$, $s(2,2)$, $s(4,4)$, and $s(5,5)$. Consider an assignment B for the data file m, such that

$$B(s(1,1)) = B(s(2,2)) = B(s(3,3)) = B(s(4,4)) = B(s(5,5)) = 1$$

and $L_B = B(s(3,3))$. Therefore, $S(B) = \{s(3,3)\}$.

For DRM protocol, a read quorum for data file m, is equal to write quorum. The quorum sets for read and write operations are $Q(B,q1)$, $Q(B,q2)$, $Q(B,q3)$ and $Q(B,q4)$, where, $Q(B,q1) = \{s(3,3)\}$, $Q(B,q2) = \{s(3,3)\}$, $Q(B,q3) = \{s(3,3)\}$, and $Q(B,q4) = \{s(3,3)\}$.

If there is four quorums in the entire network therefore replicated data file m is equal to 4.

Definition 4: Quorum of operation x is define as q_x where q_x *is* $\{q_1, q_2, \ldots, q_i\}$, $i = 1, 2, \ldots, n$.

Theorem 2: Read quorum is equal to write quorum.

Proof: From Definition 3, q_r is $\{q_1, q_2, \ldots, q_i\}$, $i = 1, 2, \ldots, n$. and q_w is $\{q_1, q_2, \ldots, q_i\}$, $i = 1, 2, \ldots, n$. Since write quorum, q_w has the same member of the quorum q_r thus q_w is equal to read quorum, q_r.

3.0 SIMULATION DESIGN

The model of DRM is developed in Java. There are three major functions that have been developed as discussed below:

i. Identifying the Number of Quorums

This function is to identify the number of quorums, Q required in the large network using the DRM algorithm. The number of quorums for read and write operations are dynamic depending on the number of nodes, n in the whole network. The q is for selected number of nodes in a row and since it is a mesh topology therefore Q is q^2. The calculation for counting the number of quorums, Q in a mesh for the entire network is as in Eq. (1).

$$q = \left\lfloor \sqrt{n} - \frac{n}{10} \right\rfloor \qquad (1)$$

$$Q = q^2$$

ii. Identifying the Number of Replicas

This function is to identify the number of replicas, R in each quorum and the number of groups that will be formed from the n number of nodes. To calculate the number of replicas in each quorum is as in Eq. (2).

$$R = {}^{n}\!/_{q} \qquad (2)$$

iii. Selecting the replica as the primary database

Replica is selected based on the location and the number of connected nodes. This function deals with virtual column and row if new column and row are needed to perform a perfect 2D mesh. The selected replica has the replicated data file copies. If a requested node wants a copy of a file from the primary database it will go to the primary database located locally at its own quorum. If the primary database is down then the requester accessed other quorums. Accessing local primary database is important to reduce the response time of accessing the primary database.

The simulation model initializes all the variable parameters as in Table 1. Every time a simulation ends, the initialization called for the parameters to be reset.

Table 1: *Parameters Initialization*

Global parameters	Value	Each node's parameters	Value
Current number of nodes	0	Packets departed	0
		Delay per packet	
		Data file	0.0
Requested nodes	0		
		Interval time	64 bits
Number of quorums	0		
			5s

The number of columns and rows, n in each quorum must be odd, to get the middle replica. By selecting only one middle replica in each quorum has minimizes the communication cost. Figure 3 is the framework of DRM where each quorum consists of basic components such as primary database, replica management, routing table, data monitoring and replica selector. The linked between the quorums are visualized in the simulation to always be connected and active.

The storage element is where the data file is copied. For this simulation only 8 bytes are allocated because the data transfer is just an example to prove the functionality of the DRM protocol. The routing table is to identify the neighbors and links to route to the shortest path to the primary database from the requested node.

The replica management is the core algorithm that runs to select the best node to be the primary database and to allocate the copies of files by implementing the quorum approach. The data monitoring is to save the current data so at certain time the updated data are compared to the other primary database and if it is not the latest data then it will replicate to that primary database which is located at the other quorums.
After the number of quorums is identified in such network size as in Eq. (1), the number of replicas, R in each quorum is compute to make sure the quorums intersect by having odd number of nodes in each quorum. Eq. (2) is to identify the number of nodes needed which is if R is even then takes the next odd value after R.

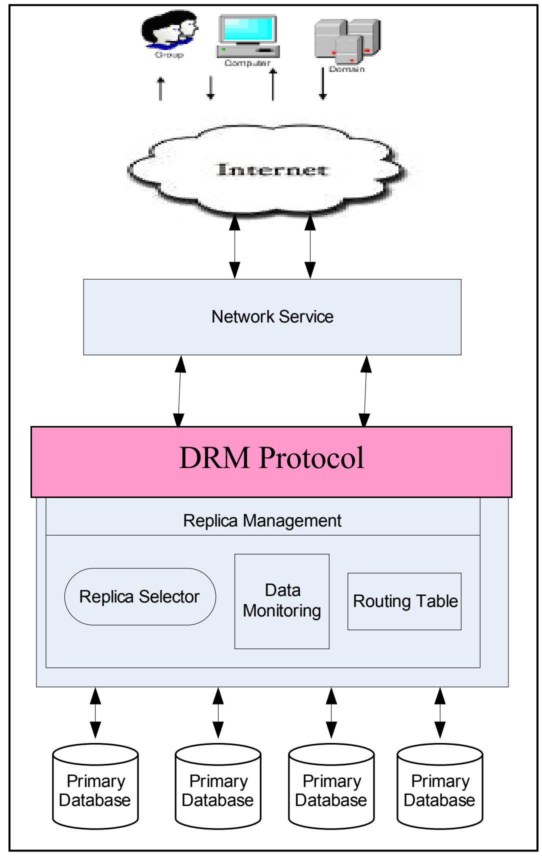

Figure 3: DRM Framework

Figure 3 is the framework for DRM protocol. It illustrates how the algorithm is designed and implemented.

4.0 RESULTS

DRM evaluates response time, t to access the data replication at the primary database. The response time is calculated based on the number of hops to the nearest primary database of a quorum where 1 hop is assumed to be 1 second. Meanwhile average of response time is summarized parameter values of response time observed. If $\{t_1, t_2, \ldots, t_n\}$ are n observed values of requested numbers, the average of response time is given as in Eq. (3) as referred in (Jain, 1991).

$$\bar{t} = \frac{1}{n}\sum_{i=1}^{n} t_i \quad (3)$$

Figure 4 shows the results of response time of TDGS and DRM. The results are for 81 nodes where the DRM has reduced the response time even though the number of nodes has increased because of the multi quorums which are intersected with other quorums in the entire network.

Response time means the time taken for a requested node to access a copy of a data file at a primary database starting from the time requested acknowledgment is sent to read or write the data file at the primary database. Meanwhile average response time means the total of response times divided by the number of response times taken.

Figure 4 shows that DRM has 36.36% lower than TDGS. The location of the primary database for TDGS is at the edges of the box shape topology making it difficult for the requested nodes to access the primary database.

The average of the response time of DRM increases when the number of nodes in the network increases. This is because the distance from the requested node to the primary database has also increased when the quorum size increased. The result of the average response time is also influenced by the shortest path from the requested node to the primary database. The requested node is randomly selected and if the requested node is in a far distance from the primary database then this has affected the response time. The result in Figure 4 is the average of response time which is calculated by dividing the total of response time with the number of response time taken.

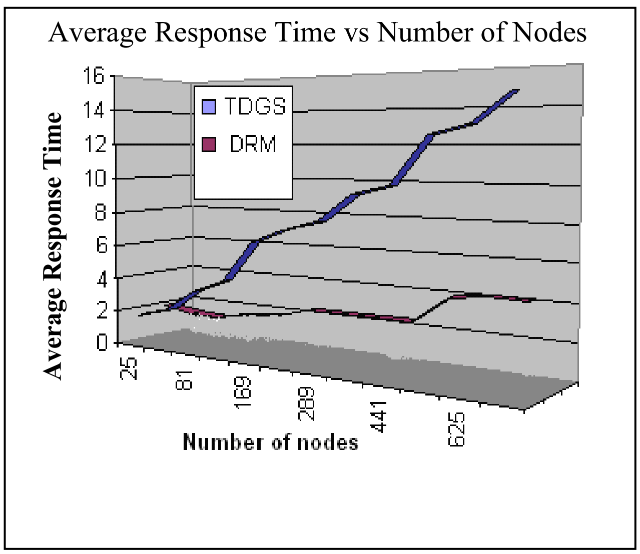

*Figure 4:*Results of Average Response Time

5.0 CONCLUSIONS AND FUTURE WORKS

By using quorum approach and selecting the middle node of the diagonal site in a quorum has improved the response time and data consistency for DRM protocol. The calculation on how data consistency in DRM is proven by fulfilling the conditions which was introduced by previous papers. The DRM shows the protocol is suitable for a larger dynamic network but yet has the lowest response time to access the data at the selected primary database. For future work, we would like to investigate the throughput of DRM protocol by having different dataset.

REFERENCES

Abawajy, J.H. (2005). An Efficient Replicated Management Approach for Peer to Peer System. *Lecture Notes of Computer Science Springer Verlag Berlin Heidelberg.* 3516: 457-463.

Ahmad, N., Mat Deris, M., Saman, M.Y.M., Norhayati, R., Rabiei, M., and Shuhadah, W.N., (2008). Managing Transactions on Grid-Neighbor Replication in Distributed Systems, *International Journal of Computer Mathematics, Taylor & Francis,* UK.

Ahmad, N., Mat Deris, M., Saman, M.Y., Norhayati, R., Rabiei, M., and Shuhadah, W.N.W. *Managing Neighbor Replication Transactions in Distribute System*, Proceeding of International Symposium on Distributed Computing and Applications Business Engineering and Science, Hangzhou,China. October 12-15, 2006. ISTP, 2006

Jain, R. (1991). *The art of Computer Systems performance Analysis : Techniques for Experiment design, Measurement, Simulation, and Modeling.* New York: John Wiley & Sons, Inc.

Kumar, A. (1991). Hierarchical Quorum consensus: A New Algorithm for Managing Replicated Data. *IEEE Trans. on Computers*. 40(9): 996-1004.

Lamehamedi, H., and Szymanski, B.K. (2007). Decentralized Data Management Framework For Data Grids, *Future Generation Computer Systems.* 23(1): 109-115.

Mat Deris, M., Abawajy, J.H., Mamat, A. (2008). An Efficient Replicated Data Access Approach for Large-Scale Distributed Systems. *Journal of Future Generation of Computer Science*, Elsevier. 24(1):1-9.

Mat Deris, M., Bakar, N., Rabiei, M., and Suzuri, H.M. (2004). Diagonal Replication On Grid For Efficient Access Of Data In Distributed Database Systems. *ICCS 2004, Lecture Notes Computer Science.* 3038: 379-387.

Venugopal, S., Buyya, R., and Ramamohanarao, K. *A Taxonomy of Data Grids for Distributed Data Sharing, Management and Processing.* Technical Report GRIDS-TR-2005-3: Grid Computing and Distributed systems Laboratory, University of Melbourne, Australia, 2005.