Cost Saving Replication Technique in Data Grid

A Shukri Mohd Noor¹, Auni Fauzi¹, Ainul Azila Che Fauzi¹
¹Faculty of Ocean Engineering Technology and Informatics, Universiti Malaysia Terengganu, 21030 Kuala Terengganu, Terengganu
ashukri@umt.edu.my

Abstract. Replication is a useful technique for distributed database systems. An existing popular technique that has been used before are Read One-write-All Monitoring Synchronization Transactions System (ROWA-MSTS) and Hierarchical Replication Scheme (HRS). Nevertheless, these techniques have their flaws in terms of replication time taken. Subsequently, ROWA and HRS take more executing time in order to complete a transaction. It is because they have to copy their data to all sites. This research, Binary Voting Fragmented Database Replication (BVFDR) model, the some-data-to-some-sites technique is proposed. In order to handle transactions in the systems, BVFDR considers the neighbouring servers binary vote assignment to its logical grid structure on fragmented data copies. Therefore, it reduces storage capacity. An experiment has been carried out in three replicated servers. The results have been compared with existing techniques such ROWA and HRS. The result shows that BVFDR is able to preserve data consistence and outperformed in terms of time taken for a complete transaction compare to existing techniques. Overall, BVFDR able to handle fragmented data replication as well as capable to manage transaction management in distributed database environment by maintaining consistency of the data through the synchronization approach.

1. Introduction

Data grid is one of the most popular grid computing implementations concerning data management system and data replication technologies. The aim of data replication is to increase availability, fault tolerance, load balancing and scalability while reducing bandwidth consumption, and job execution time [1] by having more than one copies of a data file into servers.

Replication has weaknesses that related with it. The cost of buying and maintaining these servers will be increased as the number of replicate data are keep increasing. In order to solve this costly problem and to provide data consistency is to combine data replication with database fragmentation. In order to fragment a file, the data will be chunk into smaller parts and it will be placed to sites during the allocation stage [2].

Fragmentation is very efficient in terms of how the system will works, practice and dependability in distributed database environment. The phase of fragmentation is the process of allocating a database table into a set of smaller tables. Every part of the fragmented distributed database may be copied. As soon as a data is edited at one server, the changes are captured. After that, they are committed to all the replica servers. Synchronous replication can be applied to ensure the consistency of the data. It can be divided into a few techniques, i.e., replicate some data to all sites, replicate all data to some sites and replicate all data to all sites. A right synchronization method in distributed database environment is needed to preserve the data integrity as well as data reliability.
In this paper, we able to save more cost in terms of buying and maintaining the servers as we replicate some data to some sites replication technique using a new proposed technique called Binary Voting Fragmented Database Replication model. This technique is the combination of replication and fragmentation. Combining these two techniques will increase data availability and data reliability as one server goes down, users can still query or update data by accessing the replica servers. The paper is structured as follows: in Section 2, we reviewed previous techniques on replication in distributed database. In Section 3, we explained the proposed technique. Next, we presented the results and discussions in Section 4. Finally, we concluded our work in Section 5.

2. Related Works
In this section, it reviews about replication, database fragmentation and the existing techniques that used the same technique with BVFDR which is replication in distributed database system.

2.1. Replication in data grid
Store a number of copies of the same data in different servers through the grid is the basic idea of data replication. This obviously increases the performance by decreasing remote access latency and failure [3]. Certainly, by using replication, data grid can accomplish high data accessibility and improve the consumption of the bandwidth [3].

There are three fundamental questions that must be answered in managing replica placement strategy in data grids [4]. The three questions are:
- When must the copies be produced?
- What data must be copied?
- Where the copies must be allocated?

Some replication approaches have been proposed to solve the problems that have been stated above [5,6,7,8,9].

In replicated systems, synchronization is a one of the problems that needs to be alarmed. There are two types of synchronization in replication methods which are synchronous and asynchronous replication. Asynchronous replication transfers the data using different transaction. This may affect the data consistencies. Conversely, synchronous replication guarantee data consistency because it works based on quorum to perform the processes. For each replica that has been changed, the changes are transferred instantly to all the replicas from the same transaction [10]. Hence, all the data in the replicas is the same and consistent.

2.2. Database Fragmentation
Generally, applications work with only some of database relations instead of the entire relations. Hence, database fragmentation gives a huge advantage for distributed database in terms of usage. In data distribution, it is better to work with subsets of relations as the unit of distribution [11]. Horizontal, vertical and hybrid are the three types of fragmentation [11]. Horizontal fragments are subsets of tuples and vertical fragments are subsets of attributes.

2.3. Existing Techniques
In this section the existing replication techniques are explained briefly.

2.3.1. Read-One-Write-All Monitoring Synchronization Transactions System (ROWA-MSTS). ROWA-MSTS has been established based on previous technique which is ROWA. The ROWA-MSTS technique handles each site either it is operational or down. The researcher used VSFTPD (GPL licensed FTP server for UNIX systems) as an agent communication between replicated servers [10]. In ROWA-MSTS techniques, the consistencies of replicate servers are guaranteed by the consistency of execution on one replica, but the client replicas are only updated and cannot provide accurate responses to queries. Synchronous replication methods preserve that all replicas are maintained
consistently at all times by executing each transaction locally only after all replicas have agreed on the execution order. Hence, a very strict level of consistency is maintained.

However, this technique practices all-data-to-all-sites replication protocol which means, same data will be in all servers. Moreover, the time taken in order to complete a transaction also will be high as the main server needs to wait for all other replica servers to continue with the transaction. Figure 1 shows the ROWA-MSTS framework.

![Figure 1. The framework of ROWA-MSTS](image)

### 2.3.2. Hierarchical Replication Scheme (HRS) Protocol.

HRS contains of a root database server and at least one database server. These servers are organized into a hierarchy topology [12]. Figure 2 illustrates all replicas in HRS.

![Figure 2. Hierarchical Replication Scheme](image)
The architecture of HRS is shown in Figure 2. Based on that figure, replication process starts when a transaction initiates at any block at site 1. In HRS, all update operations are conducted on a master replica, and then the modifications are propagated to all replicas. Once the changes have been made, all the data will be replicated into all sites. At last, all sites will have all the same data.

The drawback in HRS is it requires many replica servers. Consequently, it will take more executing time to compare to BVFDR because BVFDR only requires minimum 3 servers. HRS also will replicate whole file to its replica servers while in BVFDR, the file will be fragmented before it is replicated to replica servers.

2.3.3 Dynamic Algorithms Replication Using Grid Computing. Due to high numbers of users, bottleneck problem always occurs in the internal network traffic at the University of Khartoum. This paper studies data grid as a solution to the bottleneck problem by copying the internal systems of the University nearer to the users. In order to study the behavior of the various dynamic replication algorithms as well as to evaluate the performance metrics, OptorSim Grid simulator was used. In addition, other metrics such as the thread numbers at each computing element and the bandwidth range to particular site in comparison to the number of executed jobs were also evaluated. OptorSim simulator was also used to develop a simulation model for the University of Khartoum network.

There are numerous number grid simulators that have been developed, for instance, Chicago, EDGSim, GridSim, GridNet, and OptorSim. All these simulators concentrate on optimizing job scheduling in a grid environment. Nevertheless, OptorSim is known for combining job optimization with replication strategies optimization to allow the best performance from all grid resources. OptorSim is written in Java and established by work package of the European Data Grid Project in order to test file access optimization and dynamic replication strategies. OptorSim allow users to visualize the performance of the algorithm. The simulator was installed on a machine running windows 7 64 bits with 8.00 GB memory and Intel (R) Core (TM) i7-4790 CPU @ 3.6 GHz 3.6 GHz processor.

The purpose of replication is to place the files near to users so any transactions initiate by users will not occupy bandwidth on the overall network but on the local networks. The other purposes are to avoid bottlenecks and server’s timeout.

3. BVFDR Model

In this section, we proposed Binary Vote Fragmented Database Replication (BVFDR) model by considering the distributed database fragmentation.

![Figure 3. Fragmented data replication placement at site A.](image-url)
B, C, D, E, F, G, H and I.

In BVFDR technique, all servers are logically organized in the form of two-dimensional grid structure. Refer to Figure 3, data a from server A is copied to server B and D which are its adjacent replicas. Data b at server B is copied to its adjacent replicas servers A, C, and E; data c at server C is copied to its adjacent replicas servers B, and F; data d at server D is copied to its adjacent replicas server A, E, and G; data e at server E is copied to its adjacent replicas server B, D, F and H, data f at server F is copied to its adjacent replicas servers C, E and I; data g at server G is copied to its adjacent replicas servers D and H; data h at server H is copied to its adjacent replicas servers E, G and I. Meanwhile, data i at server I is copied to servers F and H.

3.1 Implementation of BVFDR
To demonstrate how BVFDR manage the transaction, nine servers that logically organized in 3 × 3 based on BVFDR two-dimensional logical design. Total number of replicated data, d, can be minimum 3 servers and maximum 5 servers. Figure 3 shows the arrangement of three replication servers. Every server is link to each other through Local Area Network (LAN). Every server is assigned with vote 1 or 0. Vote 1 means the server is not free or busy. The server is busy because it has been locked by other transaction that came earlier. Thus, new transaction cannot be proceeding on that server. Meanwhile, vote 0 means the server is not busy and ready to receive a new transaction.

![Figure 4. Three replication servers connect each other.](image)

Using Binary Voting Fragmented Database Replication (BVFDR) model, all data in main replica server will be copied to its neighbour replicas only. BVFDR applied the concept of copy-some-data-to-some-sites. By replicate their data to the neighbour servers, it will increase the availability of data. It is because, if the main server goes down, user still can access the data at any server that has its replica at any time. Based on Figure 3, we assume that data x is located in primary Server A while Server B and Server D are the neighbour replicas. As we can see form the Figure 3, the data x has been replicated to the Server B and Server D also.

4. Results and Discussion
In this section we will show how BVFDR will make improvement in the terms of cost in communication compare to existing techniques.

4.1. Communication Cost Comparison
Table 4.7 shows the communication costs for ROWA, HRS and BVFDR with the total number of sites, $n = 9, 25, 36, 49$ and $64$.

| Replication technique | Number of sites ($n$) | 9  | 25  | 36  | 49  | 64  |
|-----------------------|-----------------------|----|-----|-----|-----|-----|
| ROWA                  | 9                     | 25 | 36  | 49  | 64  |
| HRS                   | 9                     | 25 | 36  | 49  | 64  |
| BVFDR                 | $3 \leq d \leq 5$     | $3 \leq d \leq 5$ | $3 \leq d \leq 5$ | $3 \leq d \leq 5$ | $3 \leq d \leq 5$ |

| BVFDR Improvement     | ROWA                  | 44.44% | 80% | 86% | 89.8% | 92.19% |
|                       | HRS                   | 44.44% | 80% | 86% | 89.8% | 92.19% |

In Table 1, it is shown that BVFDR has the lowest communication cost, $3 \leq d \leq 5$ where $d$ is equal to the number of servers. This means BVFDR minimum copies are 3 and maximum copies are only 5. The servers replicated in BVFDR can be 9, 25, 36, 49 or 64, but the copies of each data only available in 5 servers. From Table 1, it is shown that, in 9 servers environment, ROWA and HRS has the same number of communication cost because both of the techniques apply all-data-to-all-site replication scheme.

From Table 1, it is shown that, in 9 servers environment, BVFDR has 44.44% improvement from ROWA and HRS techniques, 80% improvement in 25 servers environment, 86% improvement in 36 servers environment, 89.8% improvement in 49 servers environment and 92.19% improvement in 64 servers environment. In conclusion, BVFDR has the lowest communication cost compare to ROWA and HRS.

5. Results and Discussion

It is very important to be able managing the transaction in order to ensure the consistency and availability of data. With the objective of managing transaction and handling fragmented database in distributed database replication environment in an efficient way while saving the cost we propose a new model called Binary Voting Fragmented Database Replication (BVFDR). Based on the result of experiment, it is clearly proved that BVFDR is able to save cost even though by using replication technique. Hence, it is very useful for any organization that wants to make sure their data is secured while still in budget.

References

[1] Omer K and Abdalla G M T 2018 Dynamic Algorithms Replication Using Grid Computing International Conference on Computer, Control, Electrical, and Electronics Engineering
[2] Shaoming P, Lian X, Zhengquan X, Yanwey C and Qingxiang M 2018 A dynamic replication management strategy in distributed GIS Computer and GeoSciences 112 1-8
[3] Hossein R, Fereshteh-Azadi P and Davoud R 2018 Hierarchical simultaneous vertical fragmentation and allocation using modified Bond Energy Algorithm in distributed databases Applied Computing and Informatics 114 127-133
[4] Ranganathan K and Foster I 2001 Identifying Dynamic Replication Strategies for a High-Performance Data Grid Second International Workshop on Grid Computing 75–86
[5] Charrada B F Ounelli H and Chettaoui H 2010 Dynamic Period vs Static Period in Data Grid Replication Proceeding of IEEE International Conference on P2P, Parallel, Grid, Cloud and Internet Computing 565-568
[6] Zhao W Xu X Xiong N and Wang Z 2008 A Weight-Based Dynamic Replica Replacement Strategy in Data Grids Proceeding of IEEE on Asia-Pacific Services Computing Conference 1544-1549
[7] Zhao W Xu X Wang Z Zhang Y and He H 2010 A Dynamic Optimal Replication Strategy in Data Grid Environment International Conference on Internet Technology and Applications 1-4
[8] Ranganathan K and Foster I 2001 Identifying Dynamic Replication Strategies for a High-Performance Data Grid Second International Workshop on Grid Computing 75–86
[9] Mistarihi H E and Yong C 2008 Replica Management in Data Grid International Journal Computer Science Network Security 8(6) 22–32
[10] Noraziah A Abdalla AN and Roslina MS 2010 Data Replication Using Read-One-Write-All Monitoring Synchronization Transaction Systems in Distributed Environment Journal of Computer Science 6(10) 1033-1036
[11] Connolly T and Begg C 2015 Database System a Practical Approach to Design, Implementation and Management International Edition, Fifth Edition, Pearson Education
[12] Pérez JM Carballeira FG Carretero J Calderón A and Fernández J 2010 Branch Replication Scheme: A New Model for Data Replication in Large Scale Data Grids Future Generation Computer Systems 26 12-20