INTERNATIONAL JOURNAL OF SOFTWARE ENGINEERING & COMPUTER SYSTEMS (IJSECS)
ISSN: 2289-8522 e-ISSN: 2180-0650
VOL. 6, ISSUE 1, 18 – 24
DOI: https://doi.org/10.15282/ijsecs.6.1.2020.3.0066

ORIGINAL ARTICLE

Fragmentation Techniques with Ideal Performance in Distributed Database – A Review

S.A.U. Sharifah Hafizah[1], A. Noraziah[1,2], Julius Beneoluchi Odili[3]

[1]Faculty of Computing, Universiti Malaysia Pahang, 26300 Gambang, Kuantan, Pahang, Malaysia.
[2]IBM Center of Excellence, Universiti Malaysia Pahang, 26300 Gambang, Kuantan, Pahang, Malaysia.
[3]Department of Mathematical Sciences, Anchor University Lagos, Lagos, Nigeria.

ABSTRACT – An operative way to increase the performance of the distributed database systems and thus showing the effectiveness of computational intelligence is by implementing distributed processing. A unique distributed processing arrangement by distributed database, in which a logically interconnected database is physically dispersed across several distinct but connected by the computing facilities. Hence, proper fragmentation of data across numerous sites of the network is deliberated as a significant exploration area in the distributed database environment. In distinct servers, data can be allocated by fragmenting the entire or partial database into multiple pieces using various techniques. Each piece is located on the numerous locations. A review of fragmentation techniques applied in distributed database is conducted in this paper. All of the related techniques are evaluated in terms of performance, transfer cost/communication cost and access time/response time. The fragmentation technique is said to be good if it can increase the performance of the distributed database while reducing the transfer cost/communication cost and access time/response time. Based on the reviewed that had been done in this paper, all of the fragmentation techniques demonstrated ideal performance of distributed database and decreasing the cost of transfer/communication and time of access/response.

ARTICLE HISTORY
Received: 24 Dec 2019
Accepted: 4 Mac 2020

KEYWORDS
Distributed Database, Data Fragmentation, Computational Intelligence, Transfer Cost, Access Time

INTRODUCTION

Data is growing rapidly at the moment (Raza et. al, 2019). Centralized databases are commonly used for the data retention and management. They offer the benefits of immense level of security, qualifications, and control over backup and retrieval (Verma, 2016). Several researchers had applied centralized database environment in their studies, for example in the creation of graphical user interface for tracking data access and data changes in a centralized database discovered by Alexander et. al (2018), automated storage provisioning and management using a centralized database done by Smith et. al (2016), web based admission analysis system (Bhuvaneshwari et. al., 2019), Centralized Alumni Management System (CAMS), centralized alumni management system proposed by Mukherjee et. al (2019) and a centralized Pakistani mutome data source produced by Qasim et. al (2018).

There has been a recent trend away from centralized database systems for a number of reasons, researcher moves to distributed database due to some drawbacks of centralize database such as expensive communication costs, inaccessibility in situation of system breakdown and a lone source obstacle (Kaundal et. al, 2014 and Verma, 2016). These matters rise the necessity of spreading the databases to numerous systems or locations. Due to technology development, many implementations will be distributed in the future. So the databases will be distributed as well. The mobile communications industry also wants to be a part of this data distribution technology (Verma, 2016). There are nowadays many studies in distributed database setting for many applications, such as Ahsan et. Al (2019) developed a methodology for ontology of the domain using the Entity Relationship Model (ERM), scoring with encrypted credit communications through distributed database-grant and verification of recognition done by Chan (2019) and Phan et. al (2019) established a technique and a device for reproducing contentions in a distributed database of a mobile telecommunication network and for personalizing internet - of - things devices.

Nonetheless, the key enthusiasm underneath the idea of distributed databases is the well-organized administration of massive quantity of data with bigger accessibility and decreased communication cost (Kaundal et. al, 2014 and Verma, 2016). The distribution of data in the distributed database system therefore desires to be done properly. The distributed processing of databases is the appropriate way of boosting data efficiency. But perhaps the effectiveness of query processing is a crucial problem in the database system, because applications for decision support need limited response times (Kaundal et. al, 2014). The fragmentation of data is an essential feature of the distributed database system (Fodor and Lungu, 2016). Fragmentation in the distributed databases is very beneficial in terms usage, as typically applications are studying with only some of the relationships rather than the whole relationship (Fauzi et. al, 2011). In data distribution, evaluating with subsets of relationship as the distribution portion is better. Fragmentation of data allows the user to split

***CORRESPONDING AUTHOR** | S.A.U. Sharifah Hafizah | ✉ sh.hafizah1989@gmail.com

up a single element into two or more parts or fragments. The element may be database of a person, database of a program, or table. It can be preserved over a computer network at any location for each fragment.

Fragmentation is designed to enhance the reliability and performance of the distributed database. It is also used to balance the storage capacity and connectivity of the distributed database (Al-Sanhani et. al, 2017). In addition, fragmentation is also used to lessen the costs of network transmission and communication, besides reduced the response time. The further advantage of fragmentation is its effectiveness. Data is stored near where it is most widely used, and is not stored for unnecessary data. Of using fragmentation, it is possible to split a transaction into a few sub-queries that work on fragments. So the degrees of parallelism are rising. This will also be beneficial for security as data is not needed for local non-stored applications. Thus, illegal users would not be able to access this (Fauzi et. al, 2011).

Throughout this paper, we present variety of techniques of fragmentation with ideal and robust performance in distributed databases, including Hybrid Ant Clustering Algorithm (HACA), Vertical Fragmentation and Response Time Allocation (VFA-RT), Bond Energy Algorithm (BEA), Full Vertical Fragmentation, Allocation and Replication (FVFAR) and Binary Vote Assignment Grid Quorum with Association Rule (BVAGQ-AR). This paper examines the output of each fragmentation technique which had been implemented in terms of various performance metrics such as performance, transfer cost/communication cost and access time/response time.

This paper is organized according to the following. The nature of a distributed database system is discussed in Section 2. Section 3 addresses fragmentation on the distributed database. Section 4 explains how fragmentation methods are applied to the distributed databases. Finally, Section 5 summarizes our research from this article.

DISTRIBUTED DATABASE SYSTEM

A distributed database is a set of independent, technologically reticulated, data bases spread across a system. Distributed Management System (DDMS) is the program that controls the distributed database by creating an authentication technique that would appear to the consumer to generate this distribution (Rababaah and Hakimzadeh, 2005). Distributed database and DBMS incorporation is known as Distributed Database System (DDS). Such convergence is attained by bringing together networking technologies and database. But it could even be said as a system running on a series of devices which have no shared memory, but still act like a single device for the user. Figure 1 provides an image of a DDS.

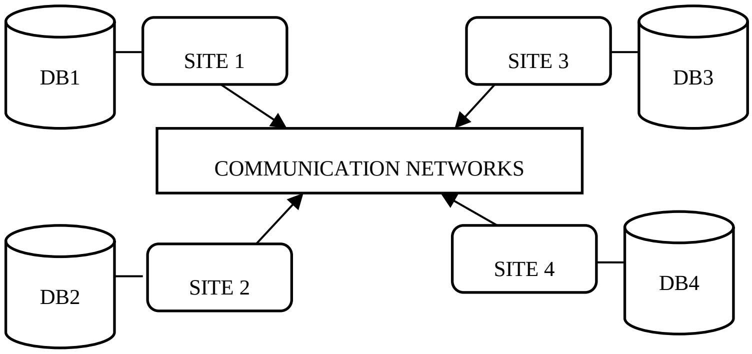

Figure 1. Distributed Database System (DDS).

DDS is broadly distributed under two categories which are Heterogeneous Distributed Database System and Homogeneous Distributed Database System and (Rababaah and Hakimzadeh, 2005). In Homogenous distributed database system (Figure 2), all servers run persistent DDMS software while the information is dispersed. Whereas the distributed heterogeneous database system (Figure 3) has several sites run under the control of significantly different DDMSs. These databases are linked anyhow to allow access to data from various sites.

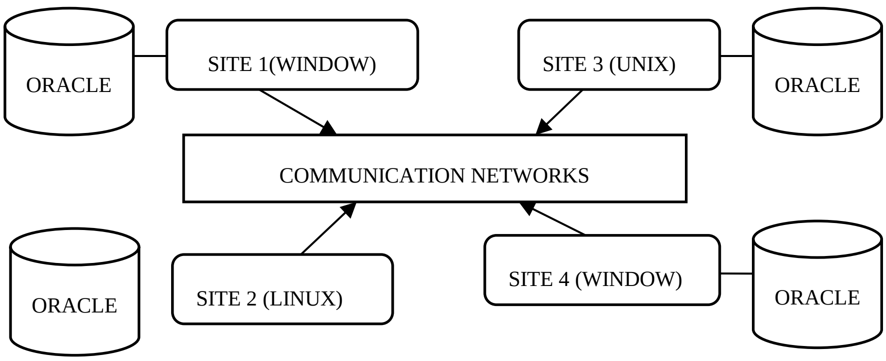

Figure 2. Homogeneous Distributed Database System.

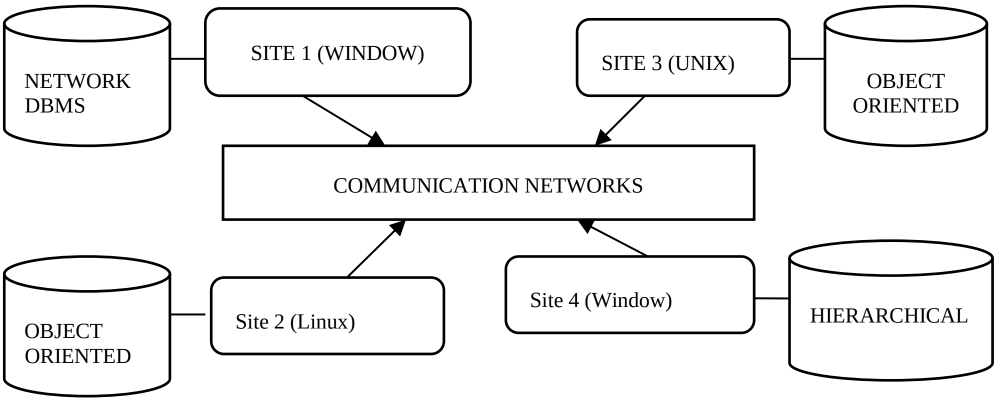

Figure 3. Heterogeneous Distributed Database System.

DDS has several benefits including being a robust system, ensuring strong-security handling, reducing bandwidth costs attributable to decreased network traffic and good performance in queries and changes due to a stable local database. Yet, DDS also have some drawbacks. By contrast with a central database system, a distributed database seems to be more difficult to configure and taking care of. It's really not that helpful if there is too much contact between sites.

FRAGMENTATION ON DISTRIBUTED DATABASE

The separation of data has been one of the main strategies used to segment distributed database systems. Separation in a distributed database is rather beneficial in terms of its management because programs work with only a few of the relationships on a regular basis in addition to the whole. Works with groups of relations as the distribution unit is better in data distribution (Coronel and Morris, 2016). Fragmentation is a model technique that divides one relationship or division of a database into two or many segments defined the segment combination provides the original database with no loss of data. It decreases the number of extraneous information accessed by database users, thereby reducing disk access capability (Bhuyar et. al, 2012).

There are some major benefits of data fragmentation in distributed database (Kaundal et. al, 2014). First of all, data is kept close to wherever it is required and is independent of similar data generated by certain users or applications. Thereafter, data is kept to better perform for native access. There have been 3 main fragmentation forms that are vertical, horizontal and hybrid / mixed fragmentation. Vertical fragments are attribute subsets while horizontal fragments are tuple subsets (Kaundal et. al, 2014). The hybrid fragmentation is that it blends horizontal fragmentation with vertical fragmentation.

Horizontal fragmentation sets the tuples as one in a partnership employed by the necessary transactions (Coronel and Morris, 2016). Horizontal fragmentation enables the partitioning of a relationship or group to separate tuples or instances. Here, the principle of horizontal fragmentation is that any site will contain all the data required to query the database, and therefore the information on the site must be distributed just so the queries on the site run faster (Bhuyar et. al, 2012). Figure 4 illustrates the horizontal fragmentation.

Name	**Matric No**	**Subject**	**Grade**
Horizontal Fragmented Data A			
Horizontal Fragmented Data B			
Horizontal Fragmented Data C			
Horizontal Fragmented Data D			

Figure 4. Horizontal Fragmentation.

Vertical fragmentation teams along the attributes in an exceedingly relation which is used conjointly via vital transactions (Coronel and Morris, 2016). Vertical fragmentation (VF) makes the partitioning of a relation or division to various sets of columns or attributes other than the primary element. The primary key attributes should be enclosed within the table for each partition. This composition will add up once numerous sites are responsible of dealing with completely different functions regarding an entity (Bhuyar et. al, 2012). Figure 5 shows an example of vertical fragmentation.

Name	**Matric No**	**Subject**	**Grade**
Vertical Fragmented Data A	Vertical Fragmented Data B	Vertical Fragmented Data C	Vertical Fragmented Data D

Figure 5. Vertical Fragmentation.

Hybrid or mixed fragmentation is that the set of relationships is comprised of attribute subsets which are still as tuples which can be seen in Figure 6. The relative table is separated into arbitrary sections, facilitating the queries ' use of the attribute and tuple use. One specific site may be allocated to each fragmentation (Harikumar and Ramachandran, 2015).

Name	**Matric No**	**Subject**	**Grade**
Hybridized Fragmented Data A		Hybridized Fragmented Data B	
Hybridized Fragmented Data C	Hybridized Fragmented Data D		

Figure 6. Hybrid/Mixed Fragmentation.

AN ANALYSIS ON THE APPLICATION OF FRAGMENTATION TECHNIQUES ON DISTRIBUTED DATABASE

These days, there have been a lot of fragmentation approaches recommended in the works to be explored in a distributed database. Nearly all of the latest research coincides with the development of a new set of fragmentation and allocation algorithms. In this paper, numerous research papers that studied data fragmentation in the distributed database with good result are reviewed based on various performance metrics such as performance, transfer cost/communication cost and access time/response time. The summary of each paper is explained as below.

An innovative Binary Vote Assignment on Grid Quorum method has been offered to handle and put up with the replication of fragmented database in Fauzi et. al (2011). Conducting fragmented database replication has higher granularity than replicated flat data file, thus increased the complexity of managing data in Noraziah et. al (2019). Managing fragmented database replication is very crucial so the data accessibility, reliability and dependability of the systems can be well-kept-up. From the research outcome, it demonstrates that the system conserves the data reliability via the harmonization method for all replicate sites. Also, it promises the reliability as the transaction implementation is followed the one-copy-serializability.

In (Bhuyar et. al, 2012), a horizontal fragmentation technique is recommended for the distributed database. The aim of this study is to implement distributed databases which are now somewhat familiar with the concept of the distributed database system and horizontal fragmentation technique. In this work, a fragmentation technique has been introduced which can be applied not only at the initial stages but also at the final stages of a distributed database system for partition relations. Results of the experiment indicate that somehow the new approach in relation to relational databases can properly address the initial fragmentation problem for the distributed systems.

Throughout (Abdalla and Amer, 2012), the distributed database system incorporates a complex horizontal fragmentation, duplication, and allocation model. A horizontal fragmentation technique is being implemented throughout this paper to break the relationships of a distributed database synchronously by combining a direct knowledge of query volumes and pattern data transfer information. It deliberates the overhead of connectivity within sites and site restrictions. By utilizing this approach, due to the synchronization, no additional complexity is added to the distribution of fragments at the various sites of a distributed database system. Consequently, the efficiency of the DDSs is significantly improved by evading regular remote accesses and expensive data transfer costs between sites.

A new Hybrid Ant Clustering Algorithm (HACA) based vertical fragmentation algorithm is proposed in (Goli and Rankoohi, 2012). Throughout this paper, the common problem of vertical fragmentation challenge in distributed database systems was discussed and the introduction of the Ant Clustering Algorithm (ACA) resulted in a new heuristic approach-based vertical fragmentation algorithm and an efficient number of fragment functions was implemented. The theoretical capacity and the precision of the results obtained from the suggested algorithm comparable to the algorithm of the Group Oriented Restricted Growth String-Genetic Algorithm (GRGS-GA). These tests revealed that the recommended algorithm has a better precision compared to GRGS-GA, which makes it an inexpensive algorithm in considerations of both computational cost and implacability. It can also produce a much more accurate result mostly for wide-scale challenges than GRGS-GA.

The vertical fragmentation technique called Vertical Fragmentation and Allocation (VFA) was used in (Pazos et. al, 2014) to diminish the response time of the round trip in the distributed database. The main purpose of this paper is to demonstrate the advantage of using the proposed method, which optimizes round trip response time against traditional models that minimize query transmission and processing costs for implementing a distributed database that is vertically fragmented. An experiment was conducted in this research to evaluate the round trip response time of the ideal solution obtained using the model suggested, rather than one of the optimal solution obtained using a traditional model. The research results showed that, for most situations, the best solution from a traditional model generates a response time that is greater than the response time of the optimal solution achieved from the proposed model, and may sometimes be three times as large.

A mixed fragmentation of clustered, huge databases was proposed in (Harikumar and Ramachandran, 2015). This research provides an exclusive method of hybridized fragmentation thru implementing the subspace clustering algorithm to create a set of fragments that partition the data apart from tuple attributes. Projected Clustering (PC) is that which controls the clusters in high-dimensional subspaces of data. This hypothesis contributes to the discovery of the intensively correlated attributes for various instance sets, thereby providing better hybridized fragments for distributed databases.

Findings demonstrate that clustering-based fragmentation of the database results in a decrease in the access time of the database particularly in comparison to the fragments present at the time of design via some statistics.

In the distributed database (Rahimi et. al, 2015), a hierarchical simultaneous vertical fragmentation and allocation using the Modified Bond Energy Algorithm (BEA) was proposed. The proposed method fragments data vertically at the same time and assigns the fragments across the network to appropriate sites. The modified BEA is being used with a reliable indicator of affinity which improved the attribute clusters produced. At the same time, the algorithm creates clusters of attributes, calculates the costs of allocating each cluster to each site, and allocates each cluster to the most appropriate location. The trial findings indicate a more reliable clustering and allocation which has led to improved performance.

An improved Creating, Reading, Updating and Deleting (CRUD) matrix has been implemented for vertical fragmentation allocation and replication throughout the cloud atmosphere (Raouf et. al, 2015). A Full Vertical Fragmentation, Allocation, and Replication (FVFAR) framework across the cloud environment is provided in this study, which emphasizes on the restriction of existing vertical fragmentation approaches as well as providing vertical fragmentation, allocation, and replication as a cloud-based infrastructure. Findings showed that the FVFAR framework focuses on restricting the existing approaches to vertical fragmentation. In addition; the findings showed a significant reduction in overall communication costs to conduct distributed database system queries compared to earlier approaches. It also demonstrated the ability of the FVFAR scheme at the primary phase of the distributed database architecture to vertically partition distributed database connections.

A new horizontal fragmentation scheme based upon the clustering technique was proposed in (Ramachandran et. al, 2016). This research takes data relationships into consideration during fragmentation and emphasizes on horizontal fragmentation. A data mining technique called a K-prototype clustering algorithm which can handle multivariate data is used in this research. Integrating the clustering technique with fragmentation will benefit from discovering the relationships of the data. The results of this study verify that the proposed algorithm will boost distributed database system performance and minimize access time via clustered fragments.

Lwin and Naing (2018) suggest a framework for non-redundant allocatio of complex fragments in distributed database system.Also the proposed method enhanced the dimension of reading and writing data size to the Threshold Time Volu-me and Distance Constraints Algorithm. In the creation of fragmentation, horizontal technique is applied to break global relation into small fragments to cut down on the volume of unrelated data recovered by the database transactions, thereby decreasing the amount of disk accesses. In the proposed algorithm, the size of read and write data volume is calculated to be determined for fragment re-allocation process that re-allocates the fragment to the position that has the normal write entry mean to lower overall transfer costs from various locations and total access times.

In Noraziah et al (2019), a new algorithm named as Binary Vote Assignment on Grid Quorum with Association Rule (BVAGQ-AR) is presented. This proposed method manages fragmented database replication using BVAGQ with a clustering data mining technique called as Association Rule. BVAGQ-AR algorithm is able of dividing the database into split fragments which contained similar patterns. The result illustrated that handling fragmented database replication using proposed BVAGQ-AR algorithm capable to sustain the data reliability, increase the performance and decrease the response time.

Table 1 shows a review of the different fragmentation techniques developed in the distributed database system. There are some hypothesizes that can be made based on the review of each research paper as mention earlier. As the result of each paper cannot be compared due to different type of data used, thus, the papers are summarize based on other issues. Firstly, any type of fragmentation (horizontal, vertical or hybrid) can be applied at the initial stage or the last stage of the distributed database system queries like Bhuyar et. al. (2012) and Raouf et. al. (2015). Next, some fragmentation techniques are also being combined with allocation and replication techniques for distributed database systems in order to improve the performance of DDS as in Fauzi et al (2011), Abdalla and Amer (2012), Raouf et. al. (2015) and Lwin and Naing (2018). The fragmentation techniques can also be applied based on the clustering approach as in Harikumar and Ramachandran (2015), Rahimi et. al. (2015), Ramachandran et. al. (2016) and Noraziah et. al. (2019). Besides, it is also can be used in a large-scale problem of DDS based on the performance in (Goli and Rankoohi, 2012) which showed the precise results with a cheaper method. As for the results obtained by all papers based on the performance metrics used, it can be said that various fragmentation techniques have been explored and proposed in order to get ideal performance of DDS. By applying fragmentation techniques in DDS, the transfer cost/communication cost and access time/response time of the DDS can be decreased. Those fragmentation techniques can be said to be efficient as a tool used in DSS.

Table 1. Summary of fragmentation techniques in distributed database system

Author (Year)	Type of Fragmentation	Combination with Allocation and Replication Techniques	Clustering Technique Involved	Improve Performance	Reduce Transfer Cost/ Communication Cost	Reduce Access Time/ Response Time
Fauzi et al (2011)	Hybrid	Yes	No	Yes	Yes	Yes
Bhuyar et al (2012)	Horizontal	No	No	Yes	Yes	Yes
Abdalla & Amer (2012)	Horizontal	Yes	No	Yes	Yes	Yes
Goli, & Rankoohi,(2012)	Vertical	No	Yes (ACA)	Yes	Yes	Yes
Pazos et al (2014)	Vertical	No	No	Yes	Yes	Yes
Harikumar & Ramachandran (2015)	Hybrid	No	Yes (Projected)	Yes	Yes	Yes
Rahimi et al (2016)	Vertical	No	No	Yes	Yes	Yes
Raouf et al (2016)	Vertical	Yes	No	Yes	Yes	Yes
Ramachandran et al (2016)	Horizontal	No	Yes (K-prototype)	Yes	-	Yes
Lwin and Naing (2018)	Horizontal	Yes	No	Yes	Yes	Yes
Noraziah et al (2019)	Hybrid	No	AR	Yes	Yes	Yes

CONCLUSION

The paper presents briefly on various fragmentation techniques of the distributed database systems. All of the fragmentation techniques that have been reviewed in this paper are able to increase the efficiency of the performance of the distributed database systems, lower the transfer cost/communication cost and minimize access time/response time. Based on the researchers ' work, these can be concluded that fragmentation strategies are an important method that can be used to enhance distributed database systems and validity of data. In order to leverage the resources, it is crucial to maintain an adequate data fragmentation framework and therefore it is necessary to choose a reliable and effective fragmentation framework to maximize the efficiency of the distributed database system. Research on the implementation of techniques for data fragmentation in a distributed database system is however still limited. In hopes of improving the efficiency of the applications, more data fragmentation techniques should be studied and introduced in the distributed database systems for continued research.

REFERENCES

Abdalla, H.I. and Amer, A.A. (2012). Dynamic horizontal fragmentation, replication and allocation model in DDBSs. International Conference on in Information Technology and e-Services, 1-7.

Ahsan M. R., M. Rahmah, Sehrish Raza, A. Noraziah and Roslina Abd. Hamid. (2019). A Methodology for Engineering Domain Ontology using Entity Relationship Model. *International Journal of Advanced Computer Science and Applications*, 10(8). doi: 10.14569/IJACSA.2019.0100842.

Alexander CR, Schantz JM, inventors; Hartford Fire Insurance Co, assignee. (2018). Graphical user interface for tracking data access and data changes in a centralized database. United States patent application US 10/083,175.

Al-Sanhani AH, Hamdan A, Al-Thaher AB, Al-Dahoud A. (2017). A comparative analysis of data fragmentation in distributed database. IEEE 8th International Conference on Information Technology, pp. 724-729.

Bhuyar, P.R., Gawande, A.D. and Deshmukh, A.B. (2012). Horizontal Fragmentation Technique in Distributed Database. *International Journal of Scientific and Research Publications*, 2(5), 1-7.

Bhuvaneshwari R, Prasath P, Manikandan G, Ponraj V. (2019). Web Based Admission Analysis System.

Chan LM, inventor; www TrustScience com Inc, assignee. (2019). Scoring using distributed database with encrypted communications for credit-granting and identification verification. United States patent application US 10/187,277.

Coronel, C. and Morris, S. (2016). Database Systems: Design, Implementation & Management. *Cengage Learning*.

Fodor, A.G. and Lungu, I. (2016). Implementation of fragmentation and replication methods in distributed systems. *Journal of Information Systems & Operations Management*, 373.

Fauzi, A.A., Noraziah, A., Zain, N.M., Beg, A.H., Khan, N. and Sultan, E.I. (2011). Handling fragmented database replication through binary vote assignment grid quorum. *Journal of Computer Science*, 7(9), 1338.

Gao J, Liu W, Li Z, Zhang J, Shen L. (2019). A general fragments allocation method for join query in distributed database. *Information Sciences*, 512, 1249-1263.

Goli, M. and Rankoohi, S.M.T.R. (2012). A new vertical fragmentation algorithm based on ant collective behavior in distributed database systems. *Knowledge and Information Systems*, 30(2), 435-455.

Harikumar, S. and Ramachandran, R. (2015). Hybridized fragmentation of very large databases using clustering. IEEE International Conference on in Signal Processing, Informatics, Communication and Energy Systems, 1-5.

Kaundal, G., Kaur, S. and Vashisht, S. (2014). Review on Fragmentation in Distributed Database Environment, *IOSR Journal of Engineering*, 4(3), 28-32.

Lwin, N. K., Naing, T. M. (2018). Non-Redundant Dynamic Fragment Allocation with Horizontal Partition in Distributed Database System. IEEE International Conference on Intelligent Informatics and Biomedical Sciences, 3, 300-305.

Mukherjee A, Roy A, Lath MK, Ghosal A, Sengupta D. (2019). Centralized Alumni Management System (CAMS)-A Prototype Proposal. IEEE International Conference on Artificial Intelligence, 967-971.

Noraziah, A, Mat Deris, Mohd Saman, M.Y., Norhayati, R, Rabiei, M., Shuhadah, W.N.W., (2009). Managing Transaction on Grid-Neighbour Replication in Distributed System. *International Journal of Computer Mathematics*, Taylor and Francis, 86(9), 1624-1633.

Noraziah A, Fauzi AA, Mohd WM, Herawan T, Abdullah Z. (2019). Managing MyGRANTS Fragmented Database Using Binary Vote Assignment Grid Quorum with Association Rule (BVAGQ-AR) Replication Model. International Conference on Data Engineering, Lecture Notes in Electrical Engineering, Springer, pp. 41-50.

Pazos, R.A., Vázquez, G., Martínez, J.A., Pérez-Ortega, J. and Martínez-Luna, G. (2014). Minimizing roundtrip response time in distributed databases with vertical fragmentation. *Journal of Computational and Applied Mathematics*, 259, 905-913.

Phan LT, Mahalal I, inventors; Gemalto SA, assignee. (2019). A method and an apparatus for publishing assertions in a distributed database of a mobile telecommunication network and for personalizing internet-of-things devices. United States patent application US 16/099,344.

Qasim I, Ahmad B, Khan MA, Khan N, Muhammad N, Basit S, Khan S. (2018). Pakistan genetic mutation database (PGMD); a centralized Pakistani mutome data source. *European Journal of Medical Genetics*. 61(4), 204-8.

Rababaah, H. and Hakimzadeh, D.H. (2005). Distributed Databases Fundamentals and Research. *Advanced Database*, B561. Spring.

Rahimi, H., Parand, F.A. and Riahi, D. (2015). Hierarchical simultaneous vertical fragmentation and allocation using modified Bond Energy Algorithm in distributed databases. *Applied Computing and Informatic*.

Raouf, A.E.A., Badr, N.L. and Tolba, M.F. (2016). An Enhanced CRUD for Vertical Fragmentation Allocation and Replication Over the Cloud Environment. 10th International Conference on Informatics and Systems, 146-152.

Ramachandran R, Nair DP, Jasmi J. (2016). A horizontal fragmentation method based on data semantics. IEEE International Conference on Computational Intelligence and Computing Research, pp. 1-5.

Sharma, T. and Sharma, D. (2014). Data Fragmentation in Multi-Cloud. *International Journal of Emerging Research in Management and Technology*, 3(5), 69-73.

Smith AT, Brennan E, Meyer K, Sahin A, Nadler M, Williams IJ, Keefe JD (2016). inventors; EMC IP Holding Co LLC, assignee. Automated storage provisioning and management using a centralized database. United States patent US 9,519,441.

Verma, S.K. (2016). Fragmentation Techniques for Distribution Database: A Review. *International Journal of Innovative Computer Science & Engineering*, 3(2):47-50.

ACKNOWLEDGEMENT

Recognition given to the Ministry of Higher Education Malaysia for Fundamental Research Grant Scheme RDU190185, University of Malaysia, Pahang grants RDU170398 and PGRS170329 for financing this research.