International Journal of Intelligent Computing and Information Sciences

https://ijicis.journals.ekb.eg/

Multi-Tenant RDBMS Migration in the Cloud Environment

Ahmed E. Abdel Raouf*, Alshaimaa Abo-alian, Nagwa L. Badr.

Faculty of Computer and Information Sciences, Ain Shams University, Cairo, Egypt

ahmed_ezzat991@yahoo.com, a_alian@ cis.asu.edu.eg,
nagwabadr@cis.asu.edu.eg

Abstract. In a multi-tenant business environment, tenants share the same applications and databases to store their data. Due to the widespread use of a multi-tenant environment, the service providers face difficult challenges daily. These challenges are condensed in how to guarantee the quality of service provided to tenants, which are documented in a formal document known as a Service Level Agreement (SLA). In addition, SLA should consider the irregular patterns of workload of tenants which may affect the level of guarantee. In this research, an Enhanced Multi-Tenant Database Management System (EMT DBMS) is proposed. In addition, an Enhanced Multi-tenant Migration Algorithm called EMT-M is presented, which aims to migrate the violated tenants depending on both the number of SLA violations and the variance rate. Experimental results prove that the proposed EMT-M algorithm is ideal for

* Corresponding author: Ahmed E. Abdel Raouf
Faculty of Computer and Information Sciences, Ain Shams University, Cairo, Egypt
E-mail address: ahmed_ezzat991@yahoo.com

migrating violated tenants, as it reduces the number of SLA violations compared to previous migration algorithms.

1   Introduction

Cloud computing has become one of the most important technologies in academia and business. The primary goal of cloud service providers is to provide inexpensive services to attract many customers by reducing the costs of operating services [14]. The use of a multi-tenant SaaS architecture is one of the most promising ways to achieve this goal. In a multi-tenant SaaS architecture, the companies, which are referred to as tenants, share a common application instance and a common relational database management system (RDBMS) instance to store their data.

In a multi-tenant SaaS architecture, different data layer designs have been proposed and used in different domains [2]. The only difference between these designs is the level of the data separation for all tenants. Regardless of the design used for the data layers, service providers face enormous challenges. The first challenge is to provide guarantees of the quality and performance of the rented services to the tenants which are known as performance service level agreements SLA [3,4,5,17,18]. The SLAs are agreements between a customer and a service provider to define the minimum requirements to ensure the quality of services provided.

The second challenge is that tenants have irregular workload patterns, where high SLA violation rates can lead to loss of tenants. Therefore, the main objective is to ensure SLA to a group of tenants according to irregular workload patterns. In this paper, we present an enhanced multi-tenant database management system (EMT DBMS). We also propose an enhanced multi-tenant migration algorithm called (EMT-M) which migrates violating tenants based on both the number of SLA violations of each client site and the variance rate.

The rest of the paper is organized as follows. Section 2 reviews the related work. Section 3 introduces the proposed Enhanced Multi-Tenant Database Management System (EMT DBMS) architecture and its components. Section 4 presents the Enhanced Multi-Tenant Migration (EMT-M) algorithm. The

experimental results are illustrated in section 5. Finally, the conclusions and future research are presented in Section 6.

2 Related Research Work

Designing an effective tenant migration strategy has many concerns. Firstly, the proposed strategy must meet quality and performance requirements, known as Service Level Agreements (SLA). So, the high number of SLA violations of tenants indicates the potential for losing the violating tenant. Secondly, the proposed strategy must consider that every tenant has unexpected changes in workload which requires migration and replication of violating tenants. Therefore, the main objective is deciding how to perform on-demand migration of the violated tenant databases to distribute the unexpected workload over a flexible set of sites.

To tackle these concerns, dynamic provisioning techniques [14] have been designed, which act based on workload observations. Provisioning techniques has two types: proactive and reactive. In the proactive approach, the forecasting model is used to trigger the migrations of the tenants to avoid SLA violations. On the one hand, the reactive approach does not use a forecasting model but rather detects SLA violation and uses appropriate technology to mitigate the SLA violation crisis.

The authors of [8] analyzed database workload using the proposed PredRep approach. However, the proposed approach faces some limitations: Firstly, high volume tenants with a busty workload may not be suitable for replication; Secondly, the forecasting model used cannot be accurate if tenants have a continuous shift in workload. An allocation strategy was proposed in [8,9] which aims to reduce the SLA's penalties. The aim of the strategy is to determine whether to migrate the violated tenant to an existing virtual machine or a new virtual machine.

In [9], an ARIMA model is used for forecasting based on short time windows. A flexible predictive load balancing service for multi-tenant databases called LABAREDA was proposed in [10] which expanded the existing work of [7]. The authors of [10,11] stated that forecasting using an exponential moving average (EMA) is more appropriate in a multi-tenant environment than using an ARIMA model, for making runtime migration decisions.

However, prediction using the ARIMA model is more appropriate in some cases. The first case is that the ARIMA model can be used offline after all

training data has been verified. The second case is that the ARIMA model can be used when there is a slight difference in multi-tenant workloads. Unfortunately, if the workload changes frequently, ARIMA model needs a frequent update to its parameters which results in more cost than EMA model. In [1] the authors did not predict performance but estimated the response time for queries as soon as they reach the site.

3   The Proposed Enhanced Multi-Tenant Database Management System (EMT DBMS)

To overcome the limitations of the current literature presented in Section 2, this paper proposes an enhanced multi-tenant database management system (EMT DBMS) illustrated in Fig. 1. The proposed system architecture consists of two layers: Multi-Tenant Management (MT-M) and the Multi-Tenant Storage (MT-S) Layer.
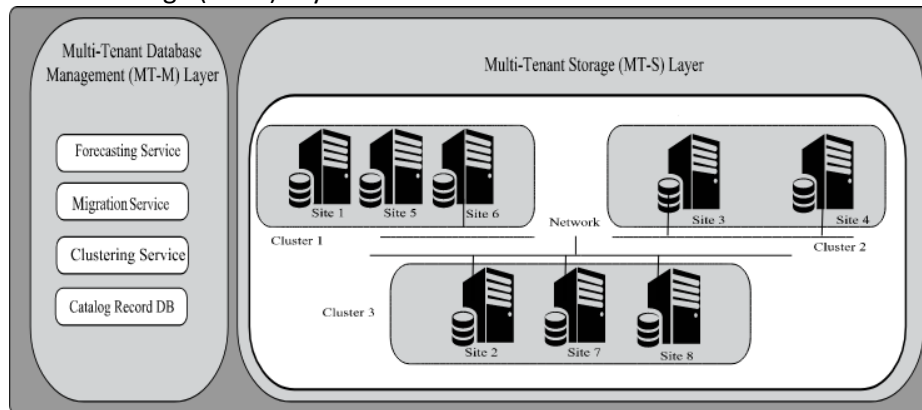

Fig. 1. The Proposed Enhanced Multi-Tenant Database Management System (EMT DBMS)

3.1   Multi-Tenant Management (MT-M) layer

This layer comprises three main services: The clustering service, the forecasting service, and the migration service.

A. The Clustering Service: Clustering service is usually accomplished by determining the similarity between the items depending on their characteristics [15][16]. This service is used to group the multi-tenant storage layer into separate groups using the lowest average connection cost between sites [6], to reduce the time required to perform the data allocation and execute the query transactions.

B. The Forecasting Service: Since any tenant requires strict guarantees of the performance and availability of the rental services, which are known as Performance Service Level Agreements (SLAs), the forecasting service uses the information in the catalog record database to pre-forecast the tenant's transaction behavior. The catalog record database is mainly used to store the complete information for each tenant transaction. The stored information consists of the tenant transactions' response time and the transaction types.

We developed the proposed forecasting service previously [12] using three different prediction models: The Recursive Window Forecasting Autoregressive Integrated Moving Average (ARIMA) model, the Exponential Moving Average (EMA) model and the proposed Recurrent Neural Network (RNN) with Long Short-Term Memory (LSTM) cells model to pre-forecast the tenant's transaction behavior. Our previous results showed that the proposed forecasting service using the proposed RNN forecasting model is the ideal choice for solving the forecasting problem in a multi-tenant business environment.

C. The Migration Service: This proposed enhanced migration service uses the forecasting service to detect the violated tenants in advance. After that, it uses the proposed enhanced multi-tenant migration algorithm (EMT-M) algorithm, which utilizes the number of SLA violations of each client site and the variance rate, in order to select the optimal site to migrate the violated tenant. The proposed EMT-M algorithm will be described in more detail in the following section.

3.2 Multi-Tenant Storage (MT-S) Layer

In the multi-tenant SaaS, different designs for data layer were proposed and used in different areas [2]. The data layer designs are Independent Databases and Independent Database Instances (IDII), Independent Tables and Shared Database Instances (ITSI), and Shared Tables and Shared Database Instances (STSI). The only difference between these designs is the separation level between the tenant data.

In IDII, each tenant has its own database system on the server. However, this design does not fulfill the main idea of a multi-tenant SaaS architecture, as each tenant must share the software and hardware.

In ITSI, all tenants share a common database with separate tables for each tenant, which reduce the maintenance costs as it uses a shared database system. In STSI, all tenants share the same database and the same tables, but it is more complicated in terms of customization and security terms when it is compared to other designs. This proposed layer consists of more than one site that have been distributed into clusters with the lowest average connection costs between them. Whereas the data layer design is built using Independent Tables and Shared Database Instances (ITSI).

4   The Proposed Enhanced Multi-Tenant Migration (EMT-M) algorithm

To overcome the limitations of existing literature highlighted by the above survey, this research proposes an enhanced multi-tenant migration algorithm called EMT-M algorithm, which is an extension of the work proposed in [13]. The main contribution in this extended research is adding a new parameter in the migration process to mitigate the SLA violations. In our previous work [13], we use the number of SLA violations to detect the source of the violation to mitigate the violation crisis.

Firstly, we migrate the violated tenant to the cluster with the maximum SLA violations to mitigate the SLA violation crisis for more than a violated site, not for just one site as the previous work. Secondly, inside the selected cluster, we migrate the violated tenant database to the site with the maximum violation rate, calculated using Eq. (1).

$$Violation\ Rate\ = MT_{IJ} - SLA_{IJ} \qquad (1)$$

The authors of [8,9] state that if the tenant's SLA cannot be met, the violated tenant database must be migrated to a site where its quality can be assured. As a result, they proposed an allocation algorithm to reduce the cost of the provider's penalties for SLA violations and improve performance. It works in the following way, given a violated tenant L in site K to be migrated.

They send the violated tenant to the site X with a max MD_J' as shown in Eq. (3), if and only if the site X has a free disk space and x ≠ k. Where MT_{IJ} is the mean execution time of the last M executed transactions, SLA_{IJ} is the service level agreement for tenant I located in site J, and the N_J is the number of tenant databases located in site J.

$$MD_J = \frac{\sum_{I=1}^{N_J}(MT_{IJ} - SLA_{IJ})}{N_J} \tag{2}$$

$$MD'_J = \begin{cases} \dfrac{\sum_{I=1}^{N_J}(MT_{IJ} - SLA_{IJ}) + (MT_{LK} - SLA_{LK})}{N_J + 1} & \text{If } J \neq 0 \\ MD_J & \text{Otherwise} \end{cases} \tag{3}$$

However, that allocation algorithm has a critical issue which rises when the site has more than one violating tenant, then the location chosen for migrating the tenant database will not be optimal. Consider the first example shown in Table 1, where tenant T3 should be migrated, if a site has more than one violated tenant databases ($MT_{IJ} > SLA_{IJ}$), then the allocation algorithm will migrate the violated tenant database to the site with the maximum average violation time, which does not meet the objective of the migration process. As a result, the allocation algorithm will send the violated tenant T3 to site 3 with maximum MD'. However, Site 3 has two tenants violate the SLA.

Table 1. Case 1 Descriptions

Site ID	Site 1			Site 2		Site 3		
Tenant ID	T1	T2	T3	T4	T5	T6	T7	T8
MT	30	40	**50**	18	14	18	20	50
SLA	35	50	25	17	15	18	19	30
MD'				16.66666		*24*		

Consider the second case when ($MT_{IJ} < SLA_{IJ}$) at site 2 and site 3 shown in Table 2 whereas the tenant T3is violating and should be migrated, the allocation algorithm migrates the violated tenant to the site with the highest average free time after adding the violated tenant who was migrated. As a result, the allocation algorithm will send the violated tenant T3 to site 3 with maximum MD'. However, Site 2 has the maximum free space and the minimum number of tenants.

Table 2. Case 2 Descriptions

Site ID	Site 1			Site 2		Site 3		
Tenant ID	T1	T2	T3	T4	T5	T6	T7	T8
MT	30	40	**50**	12	5	14	16	25
SLA	35	50	25	17	15	18	19	30
Free Space				15		12		
MD'				11.66666		15.75		

To solve the limitation of existing literature and enhance our previous migration algorithm [13], we propose an enhanced multi-tenant migration algorithm called EMT-M. The proposed EMT-M algorithm uses the number of SLA violations of each client site and the variance rate to select the optimal site to migrate the violated tenant.

The proposed EMT-M algorithm work in the following way:
firstly, it detects the violated tenants using the information from the forecasting service.
Secondly, it uses the number of SLA violations to calculate the variance rate for each client site using Eq. (4).

$$VR_{Site\,(S)} = \frac{\sum_{I=1}^{\#\,Violations\,in\,Site\,(S)}(RT(I) - SLA)}{Total\,number\,of\,violations} \tag{4}$$

Where, RT(I) is the response time of the SLA violation (I). Thirdly, it uses the information from the clustering services to detect the cluster with the maximum number of sites exceeds the variance threshold, which is calculated using Eq. (5), to solve the problem for more than one violated site and not for just only one site as the previous work.

$$Variance\,Threshold = \frac{\sum_{S=1}^{\#\,of\,Sites} VR_{Site\,(S)}}{Total\,number\,of\,Sites} \tag{5}$$

Finally, inside the chosen cluster, it migrates the violated tenant database to the site with the maximum variance rate, which is calculated using Eq. (4).

5   The Experimental Results

The performance of the proposed EMT-M algorithm is studied in a simulated environment, as shown in Fig. 2, to verify its quality by comparing it with previously used algorithms [8,9,13]. The proposed simulation environment consists of 8 Fujitsu esprimo-P556 sites grouped into 3 groups using the inter-site communication cost [6] shown in Table 3. Each site contains a Core I7 processor - 3.40 GHz and 8 GB DDR3 memory using a SQL server as a DBMS.

Table 3. Inter-site communication cost in ms

Site #	S1	S2	S3	S4	S5	S6	S7	S8

S1	0	69	57	70	10	12	75	77
S2	69	0	85	69	80	75	8	14
S3	57	85	0	12	53	63	87	82
S4	70	69	12	0	75	72	72	80
S5	10	80	53	75	0	10	78	85
S6	12	75	63	72	10	0	70	79
S7	75	8	87	72	78	70	0	12
S8	77	14	82	80	85	79	12	0

We used 6 different primary tenants with secondary replicas randomly distributed over the multi-tenant system sites as shown in Fig. 2. To verify the performance of the proposed EMT-M algorithm, we committed a violation in one specific tenant T-D1 located at Site 3. To cause the tenant violation T-D1, we increased the number of users who simultaneously accessed the tenant and the proportion of transactions. Table 4 contains all transactions and inquiries submitted by all locations for tenant (T-D1).
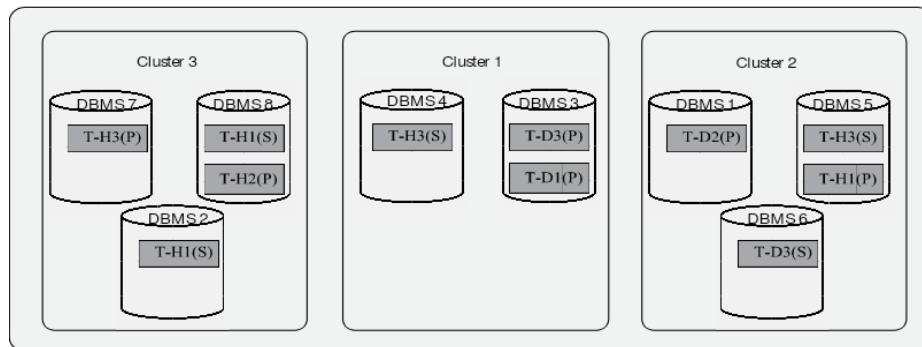

Fig. 2. The Proposed Multi-Tenant Simulated Environment

Table 4. The queries submitted to the tenant (T-D1)

Sites	Virtual Users	Query Type	# Queries
Site 1	1	Insert - Update - Delete	30000
Site 7	1	Insert - Update - Delete	30000
	20		5000
Site 5	1	Insert - Update - Delete	30000
	20		5000
Site 6	1	Insert - Update - Delete	30000
	20		5000

According to [8,9], the violated tenant T-D1 will be migrated to site 4, which has the max average free time. In addition, according to our previous

migration algorithm [13], the violated tenant T-D1 is migrated to site 7 in cluster 3, which has the maximum number of SLA violation. According to our proposed EMT-M algorithm, we use both the number of SLA violations and the variance rate to select the optimal site to migrate the violated tenant. Table 5 presents the number of the SLA violation and the calculated variance rate for the violated tenant T-D1 located in site 3, which calculated using Eq. (4).

Table 5. the SLA Violations and calculated Variance Rate for the tenant (T-D1)

Sites	SLA Violations	Variance Rate
Site 1	18	9.422262
Site 7	106	4.892512
Site 5	25	2.789669
Site 6	51	5.912793

According to table 5, using Eq. (5) the variance rate threshold value is 5.754309. According to the proposed EMT-M algorithm, the violated tenant T-D1 will migrated to the cluster with the maximum number sites that exceeds the variance threshold. As a result, the violated tenant T-D1 will be migrated to cluster 2. Inside the selected cluster, the violated tenant is migrated to the site with maximum variance rate, which is calculated using Eq. (4), if the selected site has a free disk space, as a result T-D1 will be migrated to site 1. If the specified site does not have the required capacity to accommodate the relayed tenant, then the tenant will be migrated to the site with the second maximum variance rate within the previously selected cluster.
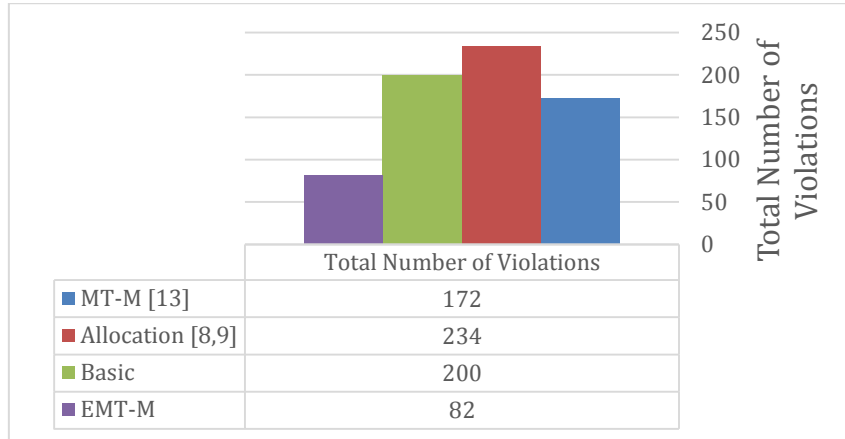
	Total Number of Violations
■ MT-M [13]	172
■ Allocation [8,9]	234
■ Basic	200
■ EMT-M	82

Fig. 3. Total Number of SLA Violation

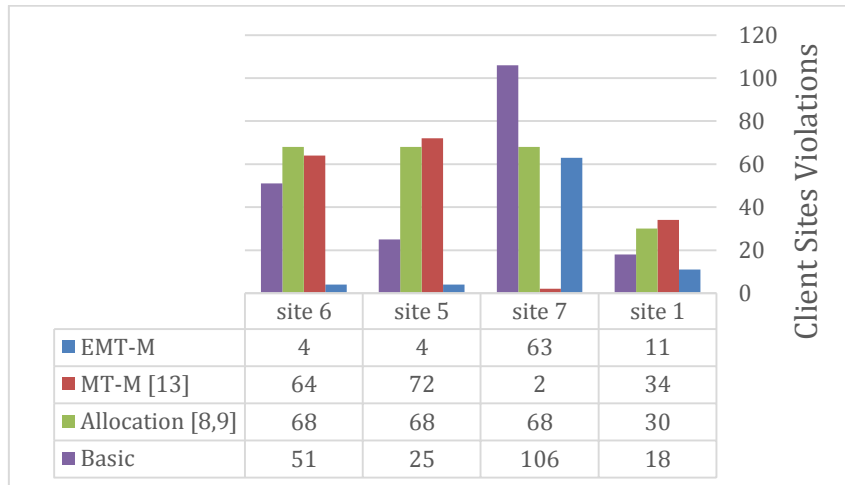|	site 6	site 5	site 7	site 1
■ EMT-M	4	4	63	11
■ MT-M [13]	64	72	2	34
■ Allocation [8,9]	68	68	68	30
■ Basic	51	25	106	18

Fig. 4. Client Sites SLA Violations

We studied the impact of implementing the proposed EMT-M algorithm and previous algorithms [8,9,13] in relation to the total number of SLA violations and the number of client sites SLA violations.

Fig. 3 and Fig. 4, show that the proposed EMT-M algorithm reduces total counts of SLA violations by 64.957% and the number of the client sites SLA violations counts by average 64.73% than the allocation algorithm [8,9]. It also shows that the proposed EMT-M algorithm reduces total counts of SLA violations by 52.3255% and the number of the client sites SLA violations counts by average 88.1655% than the previous MT-M algorithm [13].

6   **Conclusions and future work**

The irregular pattern of tenants' workload can lead to many SLA violations. Therefore, the main task of the migration algorithm is to ensure SLA agreements according to the irregular workload patterns of the tenants. In this paper, we present an enhanced multi-tenant database management system (EMT DBMS). We also present an enhanced multi-tenant migration algorithm called (EMT-M), which migrates the violating tenants' databases on a flexible set of clusters to mitigate tenant SLA violations. Experimental results prove that the proposed EMT-M algorithm results in a significant reduction in the total number of SLA violations by an average 58.64125 % than the previous migration algorithms. It also proves that the proposed EMT-M algorithm reduce the number of client sites violations by average 76.44775% than the previous algorithms. In this research, the proposed EMT-M algorithm considers both the number of SLA violations and the variance rate as the main factors to select the optimal site to allocate the migrated tenant databases. However, we believe that the proposed EMT-M algorithm still has additional important factors to consider, which are firstly when the violated tenant is replicated and secondly how the number of replicas to be created is controlled. As future work, we also plan to add a replication service which uses a forecasting service to make replication decisions in advance to avoid tenant SLA violations.

References

1. Tos, U., Mokadem, R., Hameurlain, A., Ayav, T., Bora, S.: A performance and profit oriented data replication strategy for cloud systems. In: 2016 Intl IEEE Conferences on Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld), pp. 780-787. IEEE, Toulouse (2016)

2. Ni, J., Li, G., Wang, L., Feng, J., Zhang, J., Li, L.: Adaptive database schema design for multi-tenant data management. IEEE Transactions on Knowledge and Data Engineering 26, 2079-2093 (2013)

3. Floratou, A., Patel, J. M.: Replica placement in multi-tenant database environments. In: 2015, IEEE International Congress on Big Data, pp. 246-253. IEEE, New York (2015)

4. Ji, Y., Lin, Z., Rong, T.: AdaptiveSLA: A Two-Stage Scheduling Framework for SLA Profit Maximization in Multi-tenant Database. Journal of Physics: Conference Series 1187, 052002 (2019)

5. Sakr, S., Liu, A.: Sla-based and consumer-centric dynamic provisioning for cloud databases. In: 2012 IEEE Fifth International Conference on Cloud Computing, pp. 360-367. IEEE, Honolulu (2012)

6. Abdel Raouf, A. E., Badr, N. L., Tolba, M. F.: Dynamic data reallocation and replication over a cloud environment. Concurrency and Computation: Practice and Experience 30, e4416 (2018)

7. Marinho, C. S., Coutinho, E. F., Costa Filho, J. S., Moreira, L. O., Sousa, F. R., Machado, J. C.: A Predictive Load Balancing Service for Cloud-Replicated Databases. In: SBBD (Short Papers), pp. 210-215, Brazil (2017)

8. Sousa, F. R., Moreira, L. O., Costa Filho, J. S., Machado, J. C.: Predictive elastic replication for multi-tenant databases in the cloud. Concurrency and Computation: Practice and Experience 30, e4437 (2018)

9. Moreira, L. O., Farias, V. A., Sousa, F. R., Santos, G. A., Maia, J. G., Machado, J. C.: Towards improvements on the quality of service for multi-tenant RDBMS in the cloud. In: 2014 IEEE 30th International Conference on Data Engineering Workshops, pp. 162-169. IEEE, Chicago (2014)

10. Marinho, C. S., Moreira, L. O., Coutinho, E. F., Costa Filho, J. S., Sousa, F. R., Machado, J. C.: LABAREDA: A Predictive and Elastic Load Balancing Service for Cloud-Replicated Databases. Journal of Information and Data Management 9, 94-94 (2018)

11. Andreolini, M., Casolari, S.: Load prediction models in web-based systems. In: Proceedings of the 1st international conference on Performance evaluation methodolgies and tools, pp. 27-es. ACM, New York (2006)

12. Abdel Raouf, A. E., Abo-alian, A., & Badr, N. L. (2021). A predictive replication for multi-tenant databases using deep learning. Concurrency and Computation: Practice and Experience, e6226.

13. Raouf, A. E. A., Abo-alian, A., & Badr, N. L. (2021). Towards Improvements on Multi-tenant RDBMS Migration in the Cloud Environment. In Advanced Machine Learning Technologies and Applications: Proceedings of AMLTA 2021 (pp. 593-601). Springer International Publishing.

14. Karam Ibrahim,Mohamed Aborizka, and Fahima Maghraby, "Prediction of users charging time in cloud environment using machine learning," International Journal of Intelligent Computing and Information Sciences, vol. 18,no.2 pp.39-57, 2018.

15. Arwa Awad, et al. "AN INTERACTIVE TOOL FOR EXTRACTING LOW-QUALITY SPREADSHEET TABLES AND CONVERTING INTO RELATIONAL DATABASE," International Journal of Intelligent Computing and Information Sciences, vol. 21,no.1 pp.1-18, 2021.

16. Mohamed Ismail Roushdy, and Abdel-Badeeh M. Salem, "INTELLIGENT CLUSTERING TECHNIQUE BASED ON GENETIC ALGORITHM," International Journal of Intelligent Computing and Information Sciences, vol. 21,no.1 pp.19-32,2021.

17. Rady, M., Mostafa, T., Ismail, R. (2021). Securing Query results for Cloud Databases. International Journal of Intelligent Computing and Information Sciences, 21(1), 104-118. doi: 10.21608/ijicis.2021.71016.1081

18. Saeed, Ahmed, Sally Saad, and Mostafa Aref. "Mining publication papers via text mining Evaluation and Results." International Journal of Intelligent Computing and Information Sciences 21, no. 1 (2021): 68-83.