ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ANALYTIC HIERARCHY PROCESS (AHP) FRAMEWORK FOR RANKING CLOUD DATABASES

[1]FATIMA ALZAHRA MOTASIM, [2]ADIL YOUSIF

[1]Faculty of Mathematical Science, University of Khartoum, Khartoum, Sudan

[2]Fculty of Arts and Science, Najran University, Sharourah, Saudi Arabia

E-mail: [1]fatimaalzharaa.1992@gmail.com, [2]ayalfaki@nu.edu.sa

ABSTRACT

With the emergence of cloud computing, many organizations started to offer cloud services to various costumer on the basis of their functional and non-functional demand. Cloud databases promise high performance, high availability and elastic scalability. In addition, consistency, efficiency cost and reliability multi tenancy are critical features of cloud databases. Although there are many ranking frameworks for cloud services, there is no distinct framework of ranking cloud databases. There is a need for framework for ranking cloud databases that automatically index cloud providers based on user's needs. To handle this problem, this paper proposed a framework for ranking cloud databases. To build this framework the research studied attributes and sub-attributes that are important for database. Then the research focuses on attributes and sub-attributes added by cloud databases. To calculate the ranking of the proposed framework the Analytic Hierarchy Process (AHP) model is employed. The advantage of AHP as it deals with any number of attributes and sub-attributes. The proposed framework automatically index cloud provider's databases based on user's needs. To evaluate the proposed framework a matlab experiments were developed. The results of the experiments and the evaluation process revealed that the proposed framework enhanced the ranking process of cloud databases.

Keywords: *Cloud computing; Database; Quality of service; Ranking.*

1. INTRODUCTION

Cloud computing can be defined as a new computing model that transfer the computing process from desktops computers into providers over the internet. Instead of buying actual physical devices server, storehouses, or any networking equipment, the client rents these resources from a cloud provider as an outsourced service [1-3].

There are three types for cloud services namely Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS)[4, 5]. In Software as a Service (SaaS) customers obtain the facility to access and use an application or service that is hosted in the cloud[6-8].

While in Platform as a Service (PaaS) customers obtain access to the platforms by enabling them to organize their own software and applications in the cloud[9, 10].

Infrastructure as a Service (IaaS) is based on the facility provided to the customer is to lease processing, storage, and other fundamental computing resources. A cloud database is hardening up on a cloud computing promise high carrying out, high availability, and elastic scalability[11-13].

Generally, ranking is sorting and assignment degree to some choices. This concept is applied in some cases, such as ranking of university and WWW table service etc. [14]. Ranking of cloud database is the organization of cloud database providers based on user's needs. Currently there is no framework for ranking cloud databases that can automatically index cloud providers based on customer's needs. To develop this framework an analysis of the current ranking frameworks for ranking cloud services is conducted. This paper proposed framework for ranking cloud database based on Analytic Hierarchy Process (AHP) model.

This paper contains seven sections. Section two illustrates the cloud multitenancy databases. Section three presents the related works. Section four describes the proposed cloud database ranking framework. Section five describes the details of the proposed framework attributes. Section six illustrates the experiments and the evaluation of the proposed framework. We concluded in section seven.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2. MULTI TENANCY OF CLOUD DATABASES

Cloud database multi-Tenancy is an emerging database architecture that try to realize the cloud financial profits by utilizing virtualization and permitting cloud database sharing[15, 16]. Generally, cloud Multi-Tenancy is identified as cloud resource sharing among multiple cloud clients. In cloud computing, three approaches are defined for multi-tenancy cloud databases based on the degree of data sharing. As shown in Figure 1. separate databases, shared databases separate schema and shared databases shared schema are the three approaches of managing cloud multitenant databases[17-19].

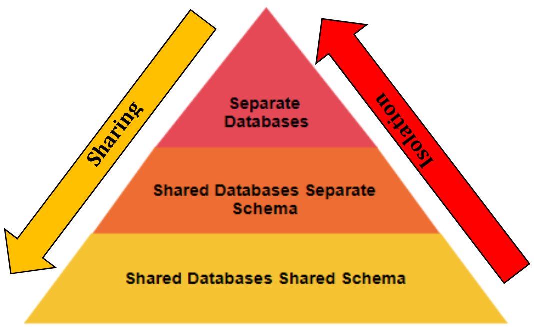

Figure 1: Cloud Database Multi-Tenancy Hierarchy

As might be expected, there is a trade-off among the three cloud database multi-tenancy approaches: namely, Separate Databases, Shared Databases Separate Schema, and Shared Databases Shared Schema[20].

The dilemma facing the cloud database provider is clear. The provider would like to use multi-tenancy approach that provide more database and cloud resource sharing, both because the sharing is needed and because resource sharing generates low database cost. However, to meet security requirements, the cloud provider needs to use isolated, relatively lower-sharing approaches with further data isolation[17, 21].

2.1 Separate Databases Multi-Tenancy

To implement tenant's data isolation, separate databases is the suitable option for database security and isolation. In separate multi-tenancy databases cloud resources and software are shared among all cloud tenants as shown in Figure 2. Nonetheless, each cloud tenant data is kept isolated from other tenant's data. Tenant identification number and other metadata protect clod tenants from unexpectedly or deliberately retrieving other tenants' data[17].

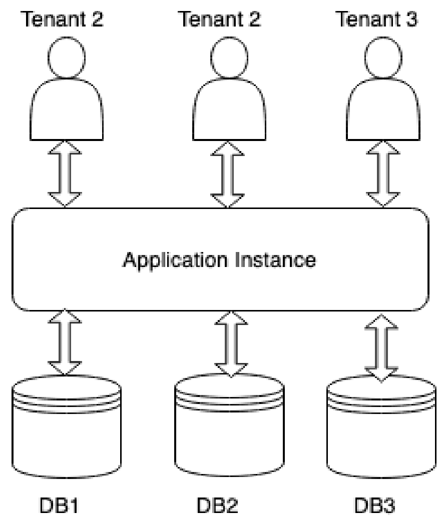

Figure 2: Separate Databases Multi-Tenancy Approach

2.2 Shared Database, Separate Schema

In shared database, separate schema several tenant's data are placed in the same database as visualized in Figure 3. However, each tenant having its private set of tables that are gathered into a specified schema generated for that tenant.

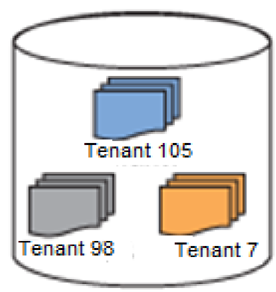

Figure 3: Shared Database, Separate Schemas

2.3 Shared Database, Shared Schema

The last multi-tenancy model utilizes the same database and the same set of tables for multiple tenants' data.

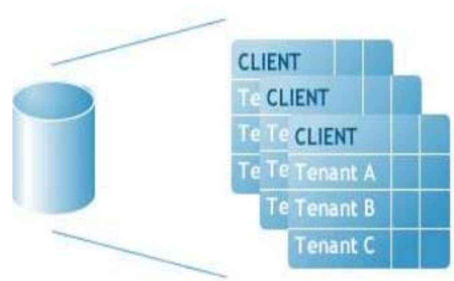

Figure 3: Shared Database, Shared Schema

Journal of Theoretical and Applied Information Technology
15th April 2020. Vol.98. No 07

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

In this approach one table may contains records from multiple tenants stored in the database. The tenant identification number field associates each tuple with the correct tenant.

TenantID	FirstName	BirthDate	C1	C2	C3
345	Ted	1970-07-02	null	"Paid"	null
777	Kay	1956-09-25	"66046"	null	null
1017	Mary	1962-12-21	null	null	null
345	Ned	1940-03-08	null	"Paid"	null
438	Pat	1952-11-04	null	"San Francisco"	"Yes"

Figure 4: Shared Database, Shared Schema Tenants Data

The shared database shared schema approach has the least resource and backup costs. The shared database shared schema approach permits cloud providers to serve the largest number of cloud tenants for each database server.

Nonetheless, several cloud tenants storing data in the same database tables needs additional efforts in data security and isolation[22, 23].

3. RELATED WORKS

Cloud computing considered as a leading. Cloud Computing has a number of ranking frameworks. Ranking frameworks help customers during the process of deciding which service provider to work with is already present on the scene.

Cloud computing creates a virtual network of various services to a numerous number of clients from all over the world. A payment is required in return and it is set based on the Quality of service the cloud is offering. These services have varying requirements depending on the services and resources involved [24, 25].

In the study presented in [16] the authors proposed a novel framework which provides both ranking and advanced attachment of cloud services using Quality of Service (QoS) features. Their focus was on the expansion number of requests problem which is continuously faced by the provider. the expansion number of requests problem makes it hard for the cloud to deliver or at least recognize the requests within requested time. Few number of the QoS characteristics are used to solve this critical problem using their proposed framework.

A more advance frameworks were offered in[26] and [27]. Again these frameworks measured the quality and prioritizes of cloud services. The differences between these frameworks and the framework in [16] is offering a healthy competition among cloud providers. Since the applications used to evaluate these providers are QoS requirements dependent.

QoS requirements dependent are used to developed an Analytical Hierarchical Process (AHP). A minor drawback is that it can only be used with quantifiable QoS attributes such as cost, confirmation of services, security and so on. Hence it fails to work with service response time, Transparency and Interoperability.

In[28] the authors took a different direction, to avoid the expensive real-world service invocation. Instead they input the past service usage experience into their QoS ranking prediction framework during the decision making phase. The collaborative method used to predict QoS services can only be used when it comes to cloud. For instance, Pearson Correlation Coefficient is used to calculate the similarity between users.

In [29] a generic QoS framework for cloud workflow systems was proposed. In this framework the process is divided into four main components, QoS requirement specification, QoS-aware service selection, QoS consistency monitoring and QoS violation handling. Nonetheless still this generic QoS framework fail to solve complex problems. Since generic QoS framework lack in terms of communication and knowledge sharing between the different components.

4. THE PROPOSED CLOUD DATABASE RANKING FRAMEWORK

The proposed ranking framework is a framework that arrange the attributes according to customer needs. The following subsections illustrates the details of the proposed cloud database ranking framework.

The main aim of this study is to adopt the proposed ranking framework for cloud databases to improve the use of the cloud computing as a result of choosing appropriate and suitable cloud provider effectively.

4.1 The Proposed Framework of Ranking Cloud Database

To create this framework, the research analyzed the critical attributes for databases. Subsequently, the research handled the emergent attributes added by cloud to databases. The important and crucial attributes are contained within the proposed model. The proposed framework shown in Figure 5 can be applied to various cloud databases.

The framework contains security, cost, performance, consistency, dependability and design as basic attributes in the first level. In the second level throughput, response time and scalability as sub-attributes of performance and availability and reliability as sub-attribute of dependability.

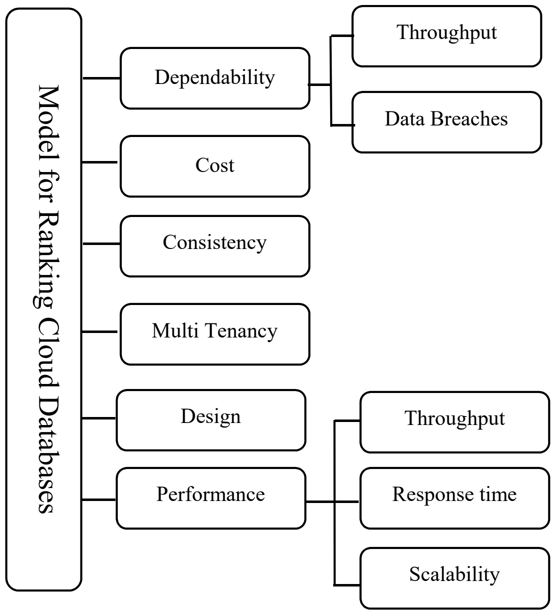

Figure 5: AHP Framework for Ranking Cloud Database

4.2 Techniques for Calculating Ranking Measurements

The proposed framework calculates the ranking measurements based on Analytic Hierarchy Process (AHP). AHP is created by the Future Generation Computer Systems in 2013 and published by ESELVIER [26]. One of the main advantages of AHP is working with any number of attributes. There are three phases in this appendage: trouble decomposition, judgment of precedence, and aggregation of this precedence. In the first phase, the ranking of a complex problem is modeled in an influence structure that specifies the interrelation among three sorts of elements. These elements include the overall goal basic attributes, their sub-attributes, and option provider. The second phase consists of two parts: pairwise comparability of attributes is done to specify their relative priority. Besides pairwise compares of Cloud provider based on their attributes to calculate their local rank. The final testing phase aggregates all local ranks of all attribute to generate the global ranking. There are four phases in the process of calculating the ranking as described in the following subsections:

4.3 Calculating of the Relative Weights of each attribute

Comparing two Cloud providers, the proposed framework needs to assign free weight to each attribute to take into account their relative grandness. This achieved by using assigned weights based on AHP standard method. The user of Cloud database can assign weights to each of the frameworks dimensions using value in some importance exfoliation, for example [1…9] as suggested in the AHP method. These values are used to indicate the importance of one attribute over another. The next step is to divide the scale of attribute between the sub-attributes that make the attribute.

4.4 Calculating the Relative Service Ranking Vector (RSRV)

This stage, find the ratio of all cloud services to each attribute. The output of this stage is the relative service ranking Vector (RSRV) for each attribute. If an attribute contains sub-attributes, then the framework calculates RSRV for each sub-attribute and aggregates the sub-attributes in matrix (M). The next step is to multiply the weight of the sub-attributes by M to produce RSRV for the attribute contains sub-attributes. The output of this phase is the RSRV of each attribute.

4.5 Aggregation of RSRV to produce the Ranking

In the final phase, the relative ranking vectors of each attribute in phase three are aggregated in matrix and this matrix is multiplied by user weights to produce the ranking results.

5. ATTRIBUTES OF RANKING FRAMEWORK

This section describes the attributes and explains the method of calculation. Then describes the weight of each of attributes, as the user input these weights into the framework. Although for some attributes the weight calculation is difficult or complex, nonetheless each attribute has value with time and has a degree or level that is considered the weight of the attribute. In other words, the weight does not necessarily mean the quantity.

5.1 Multi-Tenancy Security Attribute

Security in the proposed framework (range from 0 to 10 levels) is equivalent to multi tenancy. This is because multi- tenancy has three types. As the user input the designated level of protection they need, permission level is considered the weight. Each type equivalent level of three and a fraction of the level of security as shown in Table 1.

Table 1. Relationship between Security and Multi-tenancy

Multi Tenacity	Degree of security	Level of security/weight
Separate Databases	High	6.69 – 10
Shared Database, Separate Schemas	Mid	3.33-6.67

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Shared Database, Shared Schema	Low	0 - 3.33

5.2 Cost Attribute

The cost is one of the most important attributes. Generally, the service cost as attribute determines the quality of the other quality attributes. In other words, each cloud service determines three variables to calculate the cost of their services. The three variables are virtual machine(VM), data and storage.

5.3 Performance Attribute

The performance is divided into three sub-attributes throughput, response time and scalability. The weights of these sub-attributes are calculated according to the following equations.

5.3.1 Average Throughput (AT) Sub-Attribute

A throughput measurement mi is the number of operations over a given time period Ti. (Number of operations per second ops)

$$AT = \sum mi / \sum Ti$$

5.3.2 Average Response Time (AR) Sub-Attribute

A response is defined as the time lag between issuing a service request and receiving a successful response.

$$AR = Treq - Tresp$$

(Where Tresq= time of request, Tresp = time of response)

5.3.3 Elastic Scalability Sub-Attribute

Elasticity scalability is defined via both speed and performance. Elasticity scalability is not affected by adding new resources into a cloud database service system and removing resources from a system. Scalability is the scale-up and speedup. Scale-up defines the increase in throughput and speed up is the decrease in response time.

From the above, it can be concluded that scalability is defined as the increase in throughput, and the decrease in response time ➔

$$\text{Scalability} = \text{Throughput}/\text{Response time}$$

$$(\text{If result } > 1 \; then\ scalability = 1\ else\ scalability = 0)$$

In the event that the result was greater than 1 this means that the throughput is greater than response time. This is equivalent to the increase in throughput and the decrease in response time. Either case, the result was less than 1 this means that response time is greater than throughput. This is equivalent to the decrease in throughput and increase in the response time.

5.4 Consistency Attribute

Database is considered in consistent state if transactions maps the database from one consistent state to another. Consistency represents in four degrees from degree3 (high degree) to degree0 (lower degree).

It is true that a higher degree of consistency encompasses all the lower degrees. Table 2 describes the weight of consistency.

Table 2. Relationship between Degree of Consistency and Weight

Degree of consistency	Weight
Degree 3	0.76 – 1.0
Degree 2	0.51 - 0.75
Degree 1	0.26 - 0.50
Degree 0	0 - 0.25

5.5 Dependability Attribute

Dependability encompasses both reliability and availability. Their weights are calculated based on availability and reliability.

5.5.1 Availability Sub-Attribute

Availability can be measured as the ratio between successful requests due to unavailability of service and number of all requests.

$$\frac{\text{number af all requests} - \text{number of unavilable requsts}}{\text{number of all requsts}} \%$$

5.5.2 Reliability Sub-Attribute

Reliability reflects how a DB operates without failure during a given time.

Reliability is defined as the ratio

$$\frac{\text{MTTF}}{\text{MTTF} + \text{MTTR}}$$

(Where MTTF denotes the mean time to failure and MTTR the mean time to recovery).

Or

$$\text{Reliability} = \frac{\text{full time of system} - \text{down time of system(unavilable time)}}{100} \%$$

5.6 Design Attribute

Goals of database design are including BCNF, lossless join, dependency preservation and no NULL values in records. The proposed framework tests only if the database in BCNF or not and number of null values in the tuple.

6. CASE STUDY: RANKING CLOUD DATABASE BASED ON THE USER NEEDS

In this case study, the data were taken from three real cloud providers. Moreover, the attributes weights were collected from various evaluation studies for the three cloud providers: Amazon EC2, Windows Azure, and Rackspace. The attributes are security level, cost, and performance and also dependability [26]. Weight of throughput for Amazon EC2, Windows Azure, and Rackspace [30], [31] and [32] for three providers. , The reliability attributes dare were collected from [33]. Weights security, VM cost, data cost and storage cost are collected from [1]. Other sub-attributes are divided according to its importance for the cloud database.

The unavailable data, such as the consistency, design are randomly assigned to each cloud providers. Users weights are also randomly assigned to each attribute depend on needs. The top level of attributes is security, cost, performance, consistency and dependability[34-36].

In the following, we show in step by step manner the ranking computation process for Cloud services. The relative weighting method is used to calculate the relative ranking of cloud services for each attribute. The collected data is described in Table 3 and a numeric weight for user's request is in Table 4.

In Table 3 weights at the top level are the ratio of the user requirements for the attribute. The first level weights represent the ratio of sub- attributes contribution to the first level attribute.

Table 3: User Request Weight

Weight / User request	W1	W2	W3	W4	W5	W6
Request 1	0.05	0.1	0.2	0.3	0.3	0.05
Request 2	0.35	0.01	0.04	0.51	0.08	0.01

As shown in Table 3. and Table 4. Scalability for the three providers: P1= 2700 MB/100= 27 ➔ 27>1 ➔ scalability (p1) =1

P2 = 60 MB/600 =0.1 ➔ 0.1 <1 ➔ scalability (p2) =0, P3=10000 MB/1024 =9.76 >1

Scalability (p3) =1.

Top level attributes (Weights)	First level sub-attribute(Weights)	Provider 1	Provider 2	Provider 3
Multi Tenancy(w1)	Level: 0 – 10 (1)	4	8	4
Cost (w2)	VM cost (0.6)	0.68	0.96	0.96
	Data cost (0.2)	10	10	8
	Storage cost(0.2)	12	15	15
Performance (w3)	Throughput(0.3)	2.7	0.06	0.01
	Response time(0.3)	100	600	30
	Scalability(0.4)	1	0	1
Consistency(w4)	(1)	0.50	0.65	0.75
Dependability (w5)	Availability(0.5)	99	99	100
	Reliability(0.5)	99.	99.	100
Design (w6)	BCNF / 3NF(0.7)	0	0	1
	# of Null value (0.3)	3	7	4

Table 4: The Case Study

To find the rank of the users according to their needs the study performs some mathematical calculations. Therefore, finding RSRV which is a ratio of (provider one: provider two: provider three) for each attribute.

The calculations of the ranking system start by finding the RSRV for each attribute and multiply by user weight as follows:

The security of providers is described in the following equations:

P1 = 4, P2 =8, P3 = 4 Sum of P1, P2, P3 = 4+8+4 =16

RSRV of security = (4/16 8/16 4/16) = $(0.25 \quad 0.5 \quad 0.25) \ldots \ldots \ldots \ldots (v1)$

The cost attribute has many steps begin by calculating the RSRV for each sub-attribute and then multiply the result by their weights to produce RSRV cost.

VM cost:

P1=0.68, P2=0.96, P3= 0.96 Sum of P1, P2, P3 =2.6

RSRV of VM cost = $(0.2615 \quad 0.3692 \quad 0.3692) \ldots \ldots \ldots \ldots . (1)$

Data cost:

P1=10, P2=10, P3=8 Sum of P1, P2, P3 = 28

RSRV of data cost $=(0.3571 \quad 0.3571 \quad 0.2857) \ldots \ldots \ldots \ldots \ldots (2)$

Storage cost:

P1=12, P2=15, P3=15 Sum= 42

RSRV of storage $= (0.2857 \quad 0.3571 \quad 0.3571) \ldots \ldots \ldots \ldots \ldots \ldots (3)$

From (1), (2) and (3) RSRV of cost = [(1) (2) (3)] [weight of sub-attribute of cost]

$$\Rightarrow \begin{pmatrix} 0.2615 & 0.3571 & 0.2887 \\ 0.3692 & 0.3571 & 0.3571 \\ 0.3691 & 0.2857 & 0.3571 \end{pmatrix} * \begin{pmatrix} 0.6 \\ 0.2 \\ 0.2 \end{pmatrix} =$$

$$(0.2860 \quad 0.3643 \quad 0.3500) \ldots (v2)$$

Performance attribute according to the proposed analytic hierarchy process framework has three sub attributes, throughput, response time and scalability.

The calculations of performance attribute based on the proposed analytic hierarchy process framework for ranking cloud databases as follows:

For the throughput:

P1=2.7, P2=0.06, P3=0.01 Sum=2.77

⇨ RSRV of throughput

$=(0.9747 \quad 0.0216 \quad 0.0003) \ldots \ldots \ldots (4)$

- P1=100, P2=600, P3=30 Sum 730

⇨ RSRV of response time

$=(0.1369 \quad 0.8219 \quad 0.0410) \ldots \ldots \ldots \ldots (5)$

P1=1, P2=0, P3=0 Sum=2

⇨ RSRV of scalability =

$(0.5 \quad 0 \quad 0.5) \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots (6)$

From (1), (2) and (3) RSRV of performance =

$$\Rightarrow \begin{pmatrix} 0.9747 & 0.1369 & 0.5 \\ 0.0216 & 0.8219 & 0 \\ 0.0003 & 0.0410 & 0.5 \end{pmatrix} * \begin{pmatrix} 0.3 \\ 0.3 \\ 0.4 \end{pmatrix} =$$

$$(0.5334 \quad 0.2530 \quad 0.2123) \ldots \ldots \ldots (v3)$$

For consistency sub attribute:

P1=0.50, P2=0.65, P3=0.75 Sum=1.9

⇨ RSRV of consistency

$=(0.2631 \quad 0.3421 \quad 0.3947) \ldots \ldots \ldots (v4)$

For Dependability attribute we need to consider data availability and data breaches:

- Availability:

P1= 99.99, P2=99.95, P3=100 Sum = 299.94

⇨ RSRV of availability =

$(0.3333 \quad 0.3332 \quad 0.3334) \ldots \ldots \ldots (7)$

- Reliability:

P1= 99.99, P2=99.95, P3=100 Sum = 299.94

⇨ RSRV of reliability

$=(0.3333 \quad 0.3332 \quad 0.3334) \ldots \ldots \ldots (8)$

From (1) and (2) RSRV =

$$\Rightarrow \begin{pmatrix} 0.3333 & 0.3333 \\ 0.3332 & 0.3332 \\ 0.3334 & 0.3334 \end{pmatrix} * \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix} =$$

$$(0.3333 \quad 0.3332 \quad 0.3334) \ldots \ldots \ldots \ldots (v5)$$

For design attribute the proposed analytic hierarchy process framework consider the following:

BCNF/3NF:

P1= 0, P2=0, P3=1 Sum = 1

⇨ RSRV of BCNF/3NF =

$(0 \quad 0 \quad 1) \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots (9)$

- # of null value:

P1=3, P2=7, P3=4 Sum = 14

⇨ RSRV of # of null value = $(0.2142 \quad 0.5 \quad 0.2857) \ldots \ldots (10)$

From (1) and (2) RSRV of design =

$$\Rightarrow \begin{pmatrix} 0 & 0.2142 \\ 0 & 0.5 \\ 1 & 0.2857 \end{pmatrix} * \begin{pmatrix} 0.7 \\ 0.3 \end{pmatrix} =$$

$$(0.0642 \quad 0.15 \quad 0.7857) \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots (v6)$$

The ranking result = RSRV

$$\Rightarrow (v1 \quad v2 \quad v3 v4 \quad v5 \quad v6) * \begin{pmatrix} w1 \\ w2 \\ w3 \\ w4 \\ w5 \\ w6 \end{pmatrix} =$$

$$(x1 \quad x2 \quad x3)$$

For this case study we found the value $(v1, v2, v3, v4, v5, v6)$ from the (Table 4.3) and $(w1, w2, w3, w4, w5, w6)$ from the (Table 4)

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

⇨

$$= \begin{pmatrix} 0.25 & 0.2860 & 0.5334 & 0.2631 & 0.3333 & 0.0642 \\ 0.5 & 0.3643 & 0.2530 & 0.3421 & 0.3332 & 0.1500 \\ 0.25 & 0.3500 & 0.2123 & 0.3947 & 0.3334 & 0.7857 \end{pmatrix} * \begin{pmatrix} w1 \\ w2 \\ w3 \\ w4 \\ w5 \\ w6 \end{pmatrix}$$

⇨ RSRV= $(x1 \quad x2 \quad x3)$

6.1 The Result of Request 1 based on the proposed ranking framework:

According to the proposed framework calculations, this user is interested in consistency (dgree1) and dependability followed by performance, then medium cost and does not care about security (shared database, shared schema) and design.

As shown in Figure 6. provider 3 meets the demands of this user followed by provider one and finally provider two.

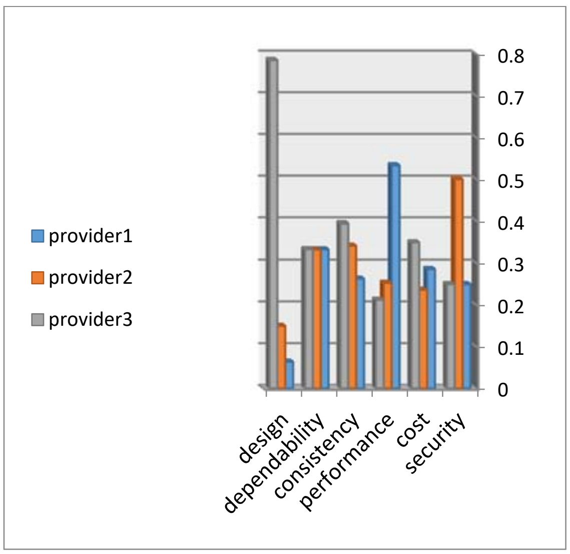

Figure 6. Cloud Provider Comparison for First Request

6.2 The Result of Request 2 based on the proposed ranking framework:

For the second request the result of ranking system is described in Figure 7. This user is interested in consistency (dgree2) followed by security (shared database, separate schema) followed by dependability then performance and lower cost does and not care about design.

RSRV2

$$= \begin{pmatrix} 0.25 & 0.2860 & 0.5334 & 0.2631 & 0.3333 & 0.0642 \\ 0.5 & 0.3643 & 0.2530 & 0.3421 & 0.3332 & 0.1500 \\ 0.25 & 0.3500 & 0.2123 & 0.3947 & 0.3334 & 0.7857 \end{pmatrix} * \begin{pmatrix} 0.35 \\ 0.01 \\ 0.04 \\ 0.51 \\ 0.08 \\ 0.01 \end{pmatrix} =$$

sRSRV2 = $(0.1830 \quad 0.3913 \quad 0.3353)$

Result of the second request ➔ P2 > P3 >P1

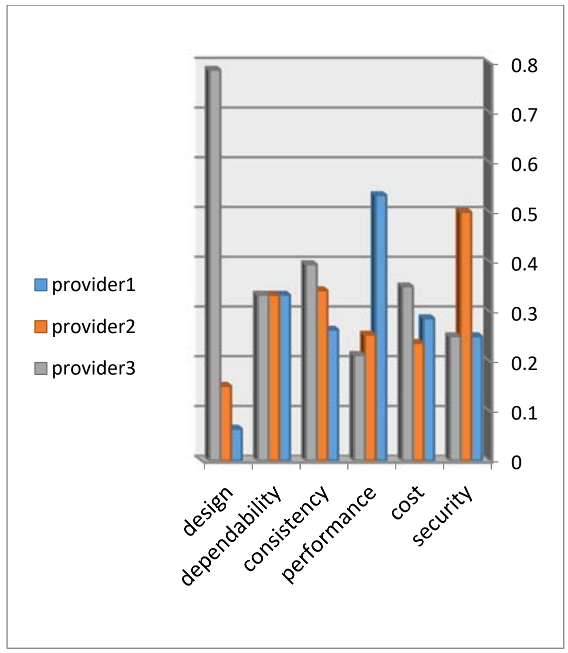

Figure 7. Cloud provider Comparison for Second Request

6.3 The proposed AHP framework using matlab simulation:

This study developed matlab simulation to evaluate the proposed AHP framework for ranking cloud databases.

In the matlab simulation cloud clients are requested to key in any three cloud providers features as an attribute vectors. Formerly, the matlab simulation applies the proposed AHP framework calculations. To end, the simulation produces ranking and arrangement of cloud provider based on clients need. The output is the ranking cloud databases based on the proposed AHP framework.

In order to confirm the validity of the simulation the study tested different consistency and assumed that clients are interested in provider's consistency. The experiments implemented the simulation three

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

times for the same the three Providers. However, each simulation the experiments places weights with largest consistency in a different position.

In the first position, the experiments placed the weight with the largest consistency in provider 1. The three providers and user weights are:

$p1 =$
$(8 \quad 0.95 \quad 158 \quad 3.1 \quad 840 \quad 0 \quad 0.85100 \quad 100 \quad 1 \quad 5)$

$p2 =$
$(4 \quad 0.96 \quad 815 \quad 0.01 \quad 301 \quad 0.75 \quad 100100 \quad 0 \quad 4)$

$p3 =$
$(2 \quad 0.5 \quad 92 \quad 5.7 \quad 2000 \quad 0.55 \quad 8888 \quad 0 \quad 9)$

$w = (0.01 \quad 0.01 \quad 0.010.94 \quad 0.01 \quad 0.01)$

The results of first positon is shown in Figure 8.

```
enter three providers p1,p2,p3  and wieght
p1=[8 0.95 15 8 3.1 40 0 0.85 100 100 1 5]
p2=[4 0.96 8 15 0.01 30 1 0.75 100 100 0 4]
p3=[2 0.5 9 2 5.7 200 0 0.55 88.00 88.00 0 9]
w=[0.01 0.01 0.01 0.94 0.01 0.01]

Providers_Indecies =

    0.3941    0.3453    0.2526
```

Figure 8: First position Ranking Results

Note that, as shown in Figure 8, provider 1 is considered the best provider in the ranking process as it has the largest consistency.

In the second situation, the experiments placed the weights with the largest consistency in provider 2. The three providers and user weights are:

$p1 =$
$(2 \quad 0.5 \quad 92 \quad 5.7 \quad 2000 \quad 0.55 \quad 8888 \quad 0 \quad 9)$

$p2 =$
$(8 \quad 0.95 \quad 158 \quad 3.1 \quad 840 \quad 0 \quad 0.85100 \quad 100 \quad 1 \quad 5)$

$p3 =$
$(4 \quad 0.96 \quad 815 \quad 0.01 \quad 301 \quad 0.75 \quad 100100 \quad 0 \quad 4)$

$w = (0.01 \quad 0.01 \quad 0.010.94 \quad 0.01 \quad 0.01)$

```
enter three providers p1,p2,p3  and wieght
p1=[2 0.5 9 2 5.7 200 0 0.55 88.00 88.00 0 9]
p2=[8 0.95 15 8 3.1 40 0 0.85 100 100 1 5]
p3=[4 0.96 8 15 0.01 30 1 0.75 100 100 0 4]
w=[0.01 0.01 0.01 0.94 0.01 0.01]

Providers_Indecies =

    0.2526    0.3941    0.3453
```

Figure 9: Second position Ranking Results

As shown in Figure 9 provider 2 is the best cloud provider since it has the largest consistency weight.

In the third situation, the largest weight is assigned to provider 3 attributes. The three providers and user weights as follows:

$p1 =$
$(4 \quad 0.96 \quad 815 \quad 0.01 \quad 301 \quad 0.75 \quad 100100 \quad 0 \quad 4)$

$p2 =$
$(2 \quad 0.5 \quad 92 \quad 5.7 \quad 2000 \quad 0.55 \quad 8888 \quad 0 \quad 9)$

$p3 =$
$(8 \quad 0.95 \quad 158 \quad 3.1 \quad 840 \quad 0 \quad 0.85100 \quad 100 \quad 1 \quad 5)$

$w = (0.01 \quad 0.01 \quad 0.010.94 \quad 0.01 \quad 0.01)$

```
enter three providers p1,p2,p3  and wieght
p1=[4 0.96 8 15 0.01 30 1 0.75 100 100 0 4]
p2=[2 0.5 9 2 5.7 200 0 0.55 88.00 88.00 0 9]
p3=[8 0.95 15 8 3.1 40 0 0.85 100 100 1 5]
w=[0.01 0.01 0.01 0.94 0.01 0.01]

Providers_Indecies =

    0.3453    0.2526    0.3941
```

Figure 10: Third position Ranking Results

Provider 3 has the largest consistency weight and hence it's the best provider as shown in Figure 10.

The previous scenario shows that the matlab simulation results of the proposed AHP framework for ranking cloud databases select the provider that has the greatest consistency as the best provider regardless of its position.

7. CONCLUSION

This paper proposed an AHP framework for ranking cloud databases. Cloud database selection is a challenging issues the clients find difficulties in the searching for appropriate provider. The proposed an AHP framework established ranking system contains framework of ranking cloud database to helps users to select the appropriate provider based on their need. The ranking cloud database framework improves the cloud database use as it saves user service selection time. The ranking system contains framework of cloud database that has attributes and sub-attributes. This research developed a case-study as a research methodology. To evaluate the proposed framework a mat-lab software was developed. The results of the developed software and the evaluation process revealed that the proposed

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

framework enhanced the ranking process for cloud database.

REFERENCES:

[1] A. Malik and M. M. Nazir, "Security framework for cloud computing environment: A review," Journal of Emerging Trends in Computing and Information Sciences, vol. 3, pp. 390-394, 2012.

[2] T. Velte, A. Velte, and R. Elsenpeter, Cloud computing, a practical approach: McGraw-Hill, Inc., 2009.

[3] B. Furht and A. Escalante, Handbook of cloud computing vol. 3: Springer, 2010.

[4] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, et al., "A view of cloud computing," Communications of the ACM, vol. 53, pp. 50-58, 2010.

[5] A. T. Velte, T. J. Velte, R. C. Elsenpeter, and R. C. Elsenpeter, Cloud computing: a practical approach: McGraw-Hill New York, 2010.

[6] M. Elsayed and M. Zulkernine, "Offering security diagnosis as a service for cloud SaaS applications," Journal of information security and applications, vol. 44, pp. 32-48, 2019.

[7] Y. Cao, C.-H. Lung, S. A. Ajila, and X. Li, "Support mechanisms for cloud configuration using XML filtering techniques: A case study in SaaS," Future Generation Computer Systems, vol. 95, pp. 52-67, 2019.

[8] S. A. Elmubarak, A. Yousif, and M. B. Bashir, "Performance based Ranking Model for Cloud SaaS Services," 2017.

[9] E. Hossny, S. Khattab, F. Omara, and H. Hassan, "A Case Study for Deploying Applications on Heterogeneous PaaS Platforms," in Cloud Computing and Big Data (CloudCom-Asia), 2013 International Conference on, 2013, pp. 246-253.

[10] A. Yousif, M. Farouk, and M. B. Bashir, "A Cloud Based Framework for Platform as a Service," in Cloud Computing (ICCC), 2015 International Conference on, 2015, pp. 1-5.

[11] M. Tamer èOzsu and P. Valduriez, Principles of distributed database systems: Springer, 2011.

[12] A. Alsirhani, P. Bodorik, and S. Sampalli, "Improving database security in cloud computing by fragmentation of data," in 2017 International Conference on Computer and Applications (ICCA), 2017, pp. 43-49.

[13] R. Elmasri and S. Navathe, Fundamentals of database systems: Pearson, 2017.

[14] K. Sridevi and R. Umarani, "A novel and hybrid ontology ranking framework using semantic closeness measure," International Journal of Computer Applications, vol. 87, 2014.

[15] W. A. Jansen, "Cloud hooks: Security and privacy issues in cloud computing," in 2011 44th Hawaii International Conference on System Sciences, 2011, pp. 1-10.

[16] P. Saripalli and B. Walters, "Quirc: A quantitative impact and risk assessment framework for cloud security," in 2010 IEEE 3rd international conference on cloud computing, 2010, pp. 280-288.

[17] H. AlJahdali, A. Albatli, P. Garraghan, P. Townend, L. Lau, and J. Xu, "Multi-tenancy in cloud computing," in Service Oriented System Engineering (SOSE), 2014 IEEE 8th International Symposium on, 2014, pp. 344-351.

[18] S. Mirzargar, "Elastic In-Memory Transaction Processing for Multi-Tenant Database Systems," in CIDR, 2019.

[19] E. Truyen, D. Van Landuyt, B. Lagaisse, and W. Joosen, "Performance overhead of container orchestration frameworks for management of multi-tenant database deployments," in Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing, 2019, pp. 156-159.

[20] Y. H. Zhou, Q. R. Wang, Z. H. Wang, and N. Wang, "DB2MMT: A massive multi-tenant database platform for cloud computing," in 2011 IEEE 8th International Conference on e-Business Engineering, 2011, pp. 335-340.

[21] I. U. Bulumulla and R. Guest, "Systems and methods for automatically collection of performance data in a multi-tenant database system environment," ed: Google Patents, 2019.

[22] V. Joshi, P. Priyadarshini, P. R. Tatini, and M. E. Weihrauch, "Supporting multi-tenant applications on a shared database using pre-defined attributes," ed: Google Patents, 2020.

[23] P. GB and P. Jayarekha, "An efficient resource sharing technique for multi-tenant databases," International Journal of Electrical & Computer Engineering (2088-8708), vol. 10, 2020.

[24] Z. Zheng, X. Wu, Y. Zhang, M. R. Lyu, and J. Wang, "QoS ranking prediction for cloud services," IEEE transactions on parallel and distributed systems, vol. 24, pp. 1213-1222, 2012.

[25] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," Future

Generation computer systems, vol. 25, pp. 599-616, 2009.

[26] S. K. Garg, S. Versteeg, and R. Buyya, "A framework for ranking of cloud computing services," Future Generation Computer Systems, vol. 29, pp. 1012-1023, 2013.

[27] K. Saravanan and M. L. Kantham, "An enhanced QoS Architecture based framework for ranking of cloud services," International Journal of Engineering Trends and Technology (IJETT), vol. 4, pp. 1022-1031, 2013.

[28] S. K. Garg, S. Versteeg, and R. Buyya, "Smicloud: A framework for comparing and ranking cloud services," in 2011 Fourth IEEE International Conference on Utility and Cloud Computing, 2011, pp. 210-218.

[29] X. Liu, Y. Yang, D. Yuan, G. Zhang, W. Li, and D. Cao, "A generic QoS framework for cloud workflow systems," in 2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing, 2011, pp. 713-720.

[30] (2019). Amazon AWS. Cloud servers. Available: https://aws.amazon.com/ec2/

[31] (2019). Azure, Microsoft. Cloud servers. Available: https://azure.microsoft.com/en-us/documentation/articles/azure-subscription-service-limits

[32] (2019). Rackspace. Cloud servers. . Available: https://cart.rackspace.com/cloud?transaction_id=102289ea252d47b974788b7c57ed03

[33] Cloud Community. Available: http://cloudharmony.com

[34] C. Li, C. Wang, and Y. Luo, "An efficient scheduling optimization strategy for improving consistency maintenance in edge cloud environment," The Journal of Supercomputing, pp. 1-28, 2020.

[35] R. G. Bacalzo, R. M. Kreuch, and M. R. Seaman, "Maintaining consistency between a transactional database system and a non-transactional content repository for document objects," ed: Google Patents, 2020.

[36] Z. Mahfoud and N. Nouali-Taboudjemat, "Consistency in Cloud-based database systems," Informatica, vol. 43, 2019.