Paper:

Clustering-Based Cloud Migration Strategies

Mubeen Aslam, Lukman bin AB Rahim, Junzo Watada, and Manzoor Hashmani

Department of Computer and Information Sciences, Universiti Teknologi PETRONAS
32610 Seri Iskandar, Perak Darul Ridzuan, Malaysia
E-mail: shim2sajid@gmail.com, {mubeen.aslam_g03587, lukmanrahim, junzo.watada, manzoor.hashmani}@utp.edu.my
[Received January 19, 2018; accepted February 8, 2018]

The k-means algorithm of the partitioning clustering method is used to analyze cloud migration strategies in this study. The extent of assistance required to be provided to organizations while working on migration strategies was investigated for each cloud service model and concrete clusters were formed. This investigation is intended to aid cloud consumers in selecting their required cloud migration strategy. It is not easy for businessmen to select the most appropriate cloud migration strategy, and therefore, we proposed a suitable model to solve this problem. This model comprises a web of migration strategies, which provides an unambiguous visualization of the selected migration strategy. The cloud migration strategy targets the technical aspects linked with cloud facilities and measures the critical realization factors for cloud acceptance. Based on similar features, a correlation among the migration strategies is suggested, and three main clusters are formed accordingly. This helps to link the cloud migration strategies across the cloud service models (software as a service, platform as a service, and infrastructure as a service). This correlation was justified using the digital logic approach. This study is useful for the academia and industry as the proposed migration strategy selection process aids cloud consumers in efficiently selecting a cloud migration strategy for their legacy applications.

Keywords: cloud migration, legacy application, migration strategies, migration strategy selection process

1. Introduction

High-power computer is used for solving complex scientific problems and it comes under High Performance Computing (HPC) technology. The power of the system can be increased by adding up computational processing power or by increasing the number of machines used such that the required processing can be realized. An economical method of achieving this is to synchronize the processing power of multiple computers, which is termed as parallel computing. Simulation software usually require such powerful computers. Complex scientific tasks can be performed on multiple processors [1]. With respect to advancement in information technology, there exist the following three computing models.

1. Cluster Computing: A cluster is an interconnected group of independent processors. Systems in a cluster may have different features such as the number of CPU cores or a similar processor structure or storing devices [2].

2. Grid Computing: A grid comprises heterogeneously scattered processing systems that include managing and allocating systems/machines, software, information, storage or network assets across dynamic and physically distributed industries [3]. The drawback of grid computing is that consumers of the grid cannot avail the services according to their requirements. There is a small number of software for consumers available on the grid [4].

3. Cloud Computing: Cloud computing is similar to grid computing but is slightly more advanced. It is considered to be a stack of scalable services available on the cloud. Consumers can be provided with these services according to their needs. In a cloud environment, resources are considered as services and can be used for as long as they are paid for. Additional payment is not required to be made if the service is not in use. This is explained by the US National Institute of Standards and Technology (NIST) as "Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction." This cloud model promotes availability of resources and has five essential characteristics, four deployment models (types of cloud), and three service models [5]. The five essential characteristics are on-demand self-service, broad network access, resource pooling, rapid elasticity, and measured service. The four deployment models/cloud types are public cloud, private cloud, community cloud, and hybrid cloud. The public cloud can be accessed by any user having an internet connection. The private cloud is created for a specific group or organization and its access is restricted to just that group. Organizations with the same requirements share a com-

Table 1. Research Questions.

Research questions		Motivation
RQ 1	How much research is carried out on cloud migration strategies for each cloud service model?	To investigate the specific extent of assistance provided to organizations with respect to cloud migration strategies.
RQ 2	What are the most commonly used and recommended cloud migration strategies on each cloud service model?	To help cloud consumers having knowledge about the endorsed cloud migration strategies at one glance.
RQ 3	What is the correlation between the cloud migration strategies?	To explore the linkages among the migration strategies based on similar features.
RQ 4	How should a migration strategy be selected?	To help cloud consumers in selecting a better migration strategy.

munity cloud. A hybrid cloud is a combination of at least two clouds (public, private, or community). The types of clouds [6] are selected according to the organization's needs [7] .

Cloud computing offers three basic service models: application/software, platform, and infrastructure [8]. Software as a service (SaaS) is a structure wherein multiple software are distributed and provided to consumers by sellers or service providers. Google's Gmail and Apps, IM on AOL, Yahoo, and Google are examples of SaaS. Platform as a service (PaaS) is a structure provided for the development and installation of internet-based application. PaaS is an extension of SaaS. PaaS provides the resources required for the development and distribution of services. Microsoft Azure and Sales-force's Force.com are some examples of PaaS. Infrastructure as a service (IaaS) is the delivery of systems like servers, memory, CPU, and network. Some related applications are Operating System (OS) and file systems. Examples of IaaS are Amazon Elastic Compute Cloud (EC2) and Secure Storage Service (S3) [9]. Organizations using legacy applications face issues related to maintainability and scalability [10]. Industries are interested in using the cloud because of its support [11]. Transferring legacy software, especially scientific software such as an open source simulation software, to the cloud requires some framework and technologies in order to make migration possible. Researchers the world over have introduced several approaches for the migration of applications and have successfully verified through experiments that cloud-based high-performance computing (HPC) technology is time and cost effective as compared to an in-house HPC cluster or the use of real physical machines [12, 13].

The objective of this research was to answer the research questions listed in **Table 1**.

The various sections of this paper provide the answers to the aforementioned research questions. Our aim is to provide guidance to academic and industry persons regarding cloud migration strategies for the migration of their legacy software. Software migration is the method of transferring a software from an older environment to a new and superior environment. Organizations do not stop using their legacy applications even if they have adopted new technology. This is because they have invested a great amount of time and money in it and the old application may hold a large amount of valuable data [14]. Therefore, partitioning clustering (as described in Section 3) was applied. We first analyzed the level of research performed on migration strategies for each cloud service model. More than twenty recently published articles were studied and thirteen sources were selected from them. After performing a deep investigation, four articles [15–18] were selected that helped us to determine the most recommended and related migration strategies proposed by researchers globally, till date. **Fig. 1** shows the number and year of publication of the selected studies. This allows an individual to obtain knowledge regarding migration strategies with respect to a specific cloud service model (SaaS, PaaS, or IaaS).

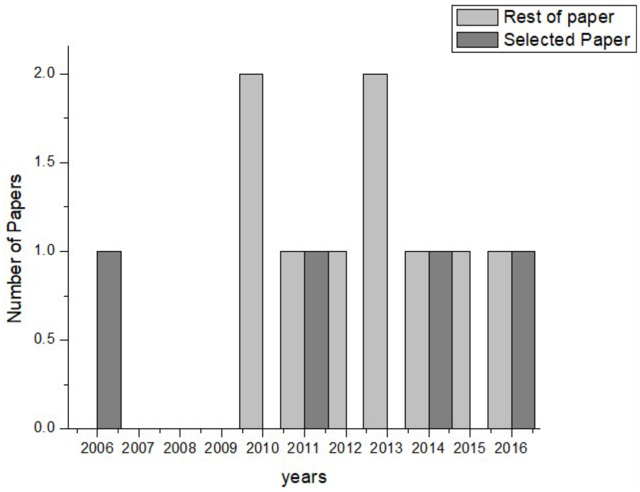

Fig. 1. Number of studies and year of publication of the studies under consideration.

Each organization can select the cloud service model according to the migration strategies available for their legacy application. This provides the answer to our first research question RQ 1.

For RQ 2, we present a web of the most recommended cloud migrations strategies. This web provides the organization with information regarding the migration types in just one glance. Thus, cloud consumers do not have to study lengthy research papers. Organizations can update themselves on the most relevant migration types for their requirements, which is specified under each cloud service model in our web. According to RQ 3, we explored the categorization of migration types and the correlation among the migration types. Migration strategies/types were clustered (as described in Section 3) into three basic sectors/clusters (complete, replacement, and component-based) and grounded on some similar features. This sorting is justified by the digital logic approach. Furthermore, the correlation among the migration strategies of each section was revealed. This correlation is especially helpful when dealing with complex scientific applications such as open-source simulation software, which sometimes require more than one migration strategy, i.e., the hybrid migration type. Having knowledge regarding the relationship among the migration strategies can help in selecting the most related migration strategies across the cloud service models. RQ 4 is about a method for helping cloud consumers to easily select a migration strategy. Our proposed migration strategy selection process (MSSP) helps cloud consumers in selecting the most relevant and appropriate migration strategy for their legacy application according to their organizational requirements.

The rest of the paper is organized as follows: Section 2 presents the methodology of the research conducted. Section 3 explains the analysis technique used in this paper. The level of the work done on migration strategies on each cloud layer is analyzed in Section 4. Section 5 presents the web of migration strategies. The correlation of the migration strategies and the corresponding justification is explained in Section 6. Section 7 explains our proposed MSSP, and Section 8 presents the conclusion of this study.

2. Methodology of Research

The methodology of our research was based on the five basic steps shown in **Fig. 2** and according to the research questions mentioned in Section 1. The main tasks performed at each stage are given below. The details of these tasks are provided in each section of the research paper.

The verification stage of our research methodology was completed by having peer reviews and by investigating personal experiences. **Table 2** lists the steps of the methodology.

3. Analysis Technique

Clustering techniques have been used to form groups of similar cloud migration strategies. The main objective of this was to discover a new set of categories based on various sets of distinctions and differences. Clusters can be created by having any similar and close relation among the objects of one cluster but that make them different from the objects of the other clusters. The most commonly used methods of clustering are hierarchical, partitioning, density-based, grid-based, and soft-computing [19]. While creating clusters, one should be very clear about the similarity and distance between the instances of the cluster, and this similarity and distance should be different from the instances of the other clusters.

Hierarchical clusters are created by dividing the pattern using the top-down or bottom-up approach. Partition clusters are the opposite of hierarchical clusters and are formed by assigning *k*-clusters without any hierarchical approach. Density-based clusters are based on local density conditions rather than the proximity between objects in a cluster [20]. Grid-based clusters are formed using a finite number of cells shaping a grid structure in which all the operations for the clustering are performed. Soft-computing includes fuzzy clustering and evolutionary approaches [19].

In this study, the partitioning clustering method is used as it allows the relocation of the objects by shifting them from one cluster to another. This brings change in the original cluster. Random formed clusters are called *k*-clusters. Furthermore, the number of all possible clusters/partitions $C(n,k)$ that can be derived by partitioning n patterns into *k*-clusters is given in Eq. (1) [20]:

$$C(n,k) = \frac{1}{k!}\sum_{i=1}^{k}(-1)^{k-1}\binom{k}{i}(i)^{n} \quad . \; . \; . \; . \; . \quad (1)$$

The *k*-means clustering algorithm of the partitioning clustering method is further used for forming concrete clusters. The work of this paper is done by following the steps of the *k*-means algorithm. This algorithm is one of the best known, bench-marked, and simplest clustering algorithms and is generally applied for solving clustering problems [21]. The data set of the cloud migration strategies in the current research is classified into *k*-clusters. The main objective is to define *k*-centroids – one for each cluster. Each set/cluster of cloud migration strategies has a specific point of interest. Section 4 of this paper presents an analysis of the extent of research work done on each layer of the cloud, i.e., SaaS, PaaS, and IaaS. The *k*-centroid for each cloud layer (cluster) is the extent of research work done for that specific layer. Furthermore, Section 6 of the paper presents three main clusters, namely, complete migration, replacement, and component-based migration. Each of these three clusters has a specific defined *k*-centroid, according to which the cloud migration strategies are correlated. The function F is given in Eq. (2) [21]:

$$F = \sum_{j=1}^{k}\sum_{i=1}^{n}\left\| x_i^{(j)} - C_j \right\|^2 \quad . \; . \; . \; . \; . \; . \; . \; . \; . \; . \quad (2)$$

where $\|x_i^{(j)} - C_j\|^2$ is a selected distance/feature measure between a data point/similar feature $xi^{(j)}$ and the cluster center/specific feature C_j.

The following are the steps for implementing the *k*-means algorithm [21].

1. Start: If we decide to form the *k*-clusters of the given data set, we randomly take k distinct similar features. These distinct features represent the initial

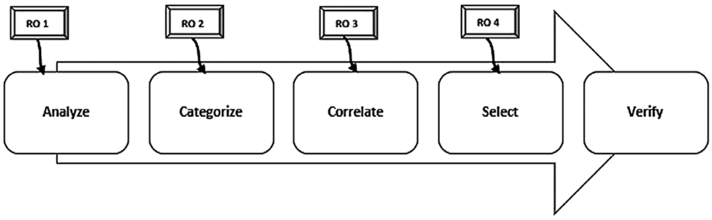

Fig. 2. Methodology of research.

Table 2. Strategies and tasks.

Sr.#	Stages	Tasks	Sections in this paper
1	Analyze:	Investigation of research work done on migration strategies for each cloud service model (using clustering method)	Analysis of Migration Strategies
2	Categorize	Classification of cloud migration strategies w.r.t. cloud service models	Web of Cloud Migration Strategies
3	Correlate:	Association discovery among migration strategies across the cloud service model and forming clusters	Correlation among Migration Strategies
4	Select	Proposing a process of selecting migration strategy	Migration Strategy Selection Process
5	Verify	Peer review / Independent review	

group centroids. As these centroids will change after each iteration before the clusters are fixed, there is no need to spend time in selecting the centroids.

2. Each cloud migration strategy is assigned to a group that has the closest similar centroid.

3. When all the cloud migration strategies have been as-signed, the positions of the *k*-centroids are recalculated.

4. Steps 2 and 3 are repeated until the centroids no longer move. This forces the instances to form groups from which the metric to be minimized can be calculated.

High-quality clusters are obtained on using good clustering methods and they represent high intra similarity and dissimilarity to the objects of other clusters. The quality of the clusters depends on the similarity measure used in the method and its implementation and on its ability to determine some of the similar hidden features [20].

4. Analysis of Cloud Migration Strategies

According to NIST, applications are migrated towards the three-basic layered cloud architecture, i.e., application/software layer (SaaS), platform layer (PaaS), and infrastructure layer (IaaS).

Based on an analysis, we determined that researchers the world over have introduced multiple migration types for handling various issues. Iterative, cloudify, horizontal, refactor, rebuild and holistic, etc., are some of the examples of migration types. The details of the migration strategies are explained in Section 6 of this paper. In **Fig. 3**, the specific extent of the research performed on each cloud layer is depicted in terms of migration strategies. The partitioning clustering technique (as described in Section 3) was applied to determine the research work done for each cloud layer, i.e., SaaS, PaaS, and IaaS. The majority of the research on migration type/strategy is performed for PaaS layer. As PaaS is the area for developers, researchers have helped developers in developing software programs for migration. The migration strategies for which the study has been conducted for PaaS are as follows: standard format from [16], redevelop from [17], [18] mentions refactor, partial, and rebuild, and [15] mentions iterative, vertical, and limited. Organizations that require their own cloud infrastructure are more interested to avail of the infrastructure services of the cloud. After PaaS, the infrastructure layer is the area for which the majority of research is conducted. The migration strategies for IaaS are rehost [18], reinstall, re-

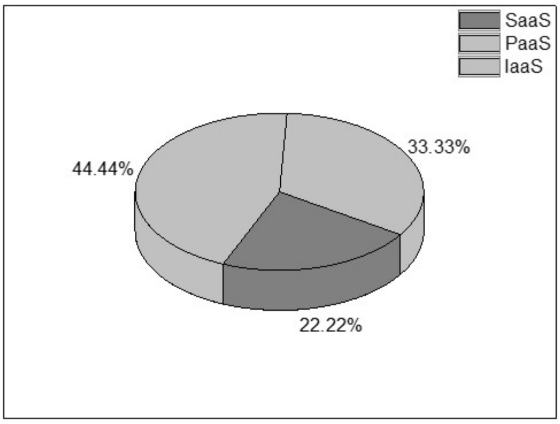

Fig. 3. Analysis of migration strategies.

locate and reconfigure (RR), and relocate and disguise environment (RD) [17], and horizontal [15]. Little research has been conducted on migration strategies for SaaS. The majority of the research conducted for SaaS is based on the applications that are directly used by users. The migration strategies for SaaS are cloudify, replace [18], and re-engineering [14]. The migration strategies used for all three layers of the cloud are holistic [16], complete [15], and revise [18]. The component format [16] migration strategy is used for PaaS and IaaS. Lastly, three main clusters SaaS, PaaS, and IaaS were organized based on the research conducted.

According to our investigation, a migration strategy is selected based on the requirements of an application and the environment. Migration is the adjustment of an application as a service over the cloud layer. If acceptable by the cloud layer, in accordance with the organizational requirements, any migration type may be used. The shifting of an application on the cloud depends on the internal architecture of the application. Developers must critically investigate the migration strategy and select an efficient migration strategy according to the requirements. If the internal architecture of an application actively fits into the architecture of a cloud service layer, only then should the migration be made; forced activity is not recommended. Migration to the cloud can be made using various methods: component by component [15], complete application [18], whole stack of an application [22], tier by tier [15], or layer by layer of an application [22]. Migration of applications for IaaS, PaaS, or SaaS is completely based on the organizations demands.

5. Migration Strategy Web

The migration of applications to the cloud raises the question of what layer of cloud should the application be shifted at and what migration type should be selected. Extant empirical studies help researchers to evaluate their current study and to categorize the future research areas for the practitioners to select the best option [23]. This section provides our empirical study on the state-of-the-art migration strategies. Two or more migration strategies can be applied to one service model, and they can be named as hybrid migration types/strategies. In **Fig. 4**, we present a web of migration strategies that provides a clear picture of the most commonly used migration types. One can easily learn of the migration types in just one glance. This model provides the most recommended and frequently used migration types introduced by researchers and entrepreneurs, as explained in Sections 4 and 6.

The migration types are distributed with respect to the cloud service layers, i.e., SaaS, PaaS, and IaaS. According to our study, and as shown in **Fig. 4**, if we have a software stack containing an OS layer, system library layer, middleware layer, and application layer, and we want to shift everything to the cloud, then IaaS is a better option for migration. For IaaS, we can select migration strategies such as rehost, component format, revise, holistic, relocate and

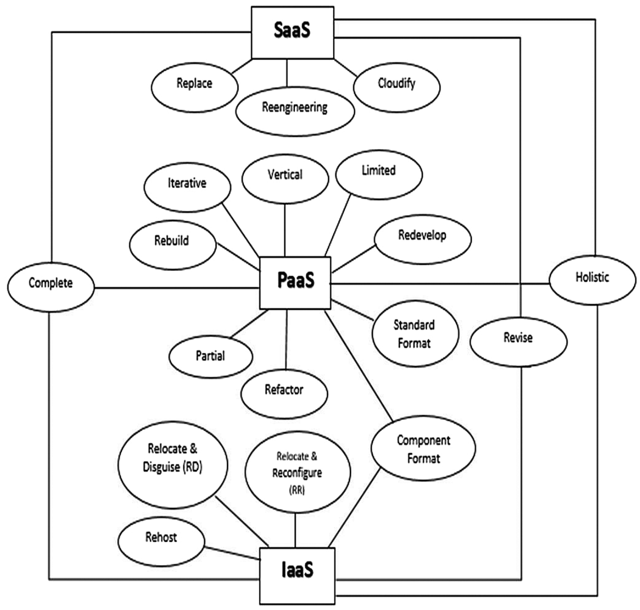

Fig. 4. Migration strategy web.

disguise, or relocate and reconfigure. These types of migrations deal with the direct infrastructure of the cloud service layer, and cloud consumers can typically deploy their own software on the infrastructure using these migration types [24]. If we intend to make any modification to the legacy source code, then PaaS is a better option, and we can select migration strategies from among complete, rebuild, iterative, vertical, limited, redevelop, partial, refactor, component format, standard format, or holistic. While using these migration types for PaaS, a cloud consumer is free to build and construct his own application. These migration strategies encapsulate the development environment and acts as a service upon which other advanced levels of offerings can be constructed [24]. If a simple replacement is required, then SaaS should be selected.

The migration strategies for SaaS are replace, reengineer, cloudify, complete, revise, and holistic, which deals with the complete migration of an application [24].

According to our investigation, the holistic, complete, and revise migration types can be used for all three, SaaS, PaaS, and IaaS, cloud service models. However, component format can be used for PaaS and IaaS only. It is very important to always remember that the migration strategies should be selected according to our requirements.

6. Correlation Among Migration Strategies

The selection of migration strategies varies from organization to organization. It depends entirely on the requirements of the organization. Some organizations need to migrate only a unit or a component of the legacy application while others need to replace the complete appli-

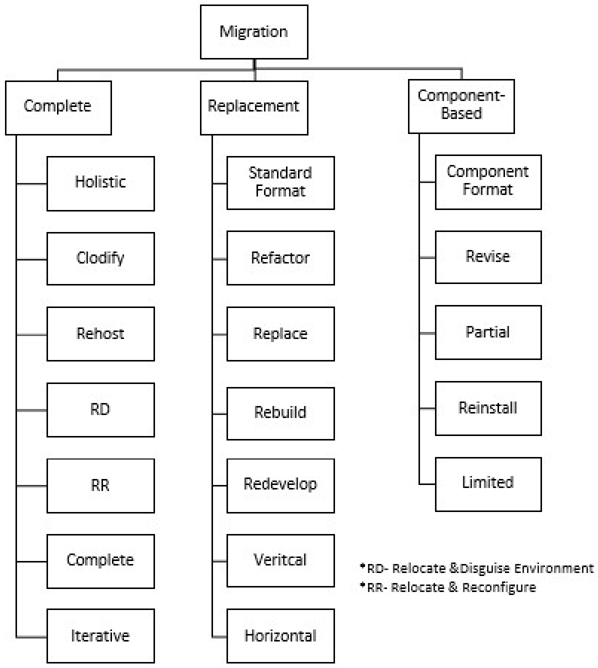

Fig. 5. Categorization of migration strategies.

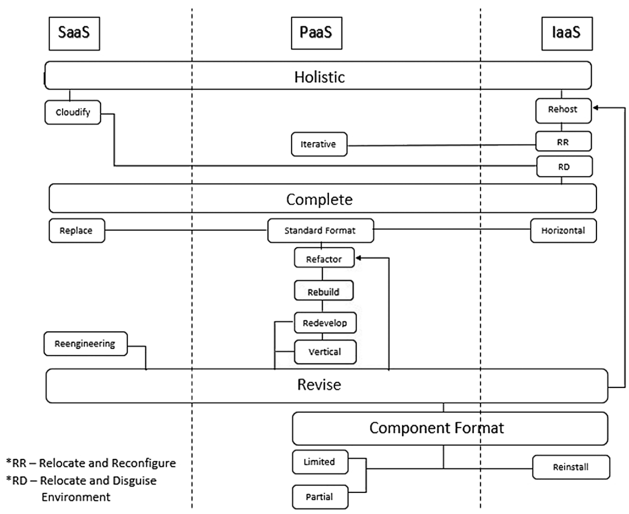

Fig. 6. Correlation among cloud migration strategies.

cation. Some organizations need to migrate a layer of an application (presentation/logic/data) while others need to replace an entire physical tier (client, application server, database server, etc.). Tiers are the positions wherein layers are deployed and executed. Some organizations need to migrate general legacy software, whereas some organizations need to migrate open-source simulation software. Any type of cloud migration can be done using the migration strategies shown in **Fig. 5**. The first part of this section is focused on the categorization/clustering of migration types. The linkages of the migration strategies across the cloud service models were then discovered, as shown in **Fig. 6**. **Table 3** presents the correlations of the migration strategies derived by us. Lastly, we have justified these correlations using the digital logic approach.

Cloud migration strategies are excavated from the selected articles after deep investigation and described in Sections 1 and 4. We have applied the clustering technique (as described in Section 3) to categorize the migration strategies into three sections/clusters, i.e., complete, replacement, and component-based. Although the migration strategies in each section do not have all the same properties, they do have a very close linkage and have some related characteristics as well and form clusters on applying the clustering rules. Each of these section is titled based on its analogous feature of migration activity that is performed by all the migration strategies of that section.

6.1. Complete Migration

Holistic migration [16] is the shifting of a complete application comprising several components, wherein each component migrates separately. It can be used for all three cloud deployment models, i.e., SaaS, PaaS, and IaaS. It is also closely related to rehost of IaaS and cloudify of SaaS owing to the similar activity of complete application migration, as explained in **Table 3**. Rehost is the redeployment of the application and middleware software in the new environment of IaaS. Holistic is the migration of a complete application but in units or components. The terms rehost and redeployment can be used interchangeably. However, cloudify [18] is used to migrate a complete application. Cloudify converts an application into a complete cloud service application, in a manner similar to holistic. Component-by-component shifting of an application is done using iterative migration [15] of PaaS. The rehost/redeploy [18] migration type of IaaS dispatches the entire application to a new environment and changes the internal structure of the application. The RD and RR [17] migration types of IaaS are very similar to each other. In RR, all the related files are copied and shifted to the cloud and the necessary configurations are then made. Therefore, rehost and RR are related to each other because both migrations deal with a change in the internal construction of an application. However, in RD, the related file is copied without taking into consideration the environment and installation. The complete [15] migration type also shifts all the components of an application as a whole without going into the details of the environment and can be used with all three cloud service models. **Fig. 6** demonstrates the linkage of RR with iterative, cloudify with RD, RD with complete, and rehost with RR. Therefore, all the migration strategies in this section are correlated with each other across the cloud service models and comprise the ultimate complete migration of an application.

6.2. Replacement

The standardized format migration (SFM) [16] is specifically applicable to the PaaS layer. Standard format shifts an Independent module. Modules/components

Table 3. Correlation among cloud migration strategies.

Complete Migration	**Complete** SaaS, PaaS, IaaS (All the components as a whole)	**Relocate and Disguise Environment (RD)** IaaS (Copy the legacy app without worrying about the environment and installation)	**Cloudify** SaaS (Complete migration of the application)	**Holistic** SaaS, PaaS, IaaS (Complete app. based on components is migrated. Components are migrated separately)
	Iterative PaaS (Component by Component)	**Relocate and Reconfigure (RR)** IaaS (Copy all the related files and put them into the cloud. Then make necessary configuration updates)	**Rehost** - on IaaS (App. & middle-ware s/w are redeployed)	
Replacement	**Horizontal** IaaS (Replace an entiretire first)	**Replace (RP) - SaaS** (Discard legacy s/w and use pay-as-you-go s/w services)		
			Refactor using PaaS (Application layer remains the same and cloud vendor provides the Middleware)	**Standard Format** - PaaS (A component with acceptable format)
	Vertical - PaaS (Tier by tier. Components from diff. tiers are migrated)	**Redevelop (RW)** PaaS (Redevelop or enhance the existing code base)	**Rebuild on** PaaS (Discards the legacy code. Applications are completely rewritten on top of PaaS)	
			Revise for IaaS or PaaS (The First improvement is made in the code of legacy app to fit into a cloud environment. Then Rehost or Refactor is applied)	**Component Format** PaaS, IaaS (Change of the format of a component)
Component-based Migration	**Limited** PaaS (Only required components are migrated)	**Reinstall (RI)** - IaaS (New version of required s/w packages are reinstalled)	**Partial** PaaS (Only a subset of the components of the applications are moved to cloud)	

are repositioned among the units/parts of the same applications or among the applications that can digest the migrating unit/component. A component with the acceptable format is replaced. The SFM of PaaS can be related to replace [17], [18] of the SaaS migration type as it replaces an old application/component with a new application/component. It is more preferably used for SaaS and requires that data or business tiers be shifted to the cloud. Similar to the replace migration type, the horizontal [15] migration type of IaaS comprises tier-by-tier replacement as well. However, the vertical [15] migration type of PaaS replaces vertically, i.e., one portion/component of an application is replaced from all the tires. The refactor [18] migration type runs over the PaaS. The application layer remains the same and the cloud vendor provides the Middleware. Therefore, one layer works at a time in a manner similar to vertical. The rebuild [18] migration strategy discards the legacy code and application is completely rewritten on top of the PaaS while the old application is replaced. Redevelop [17] enhances the existing code base and replaces the architecture of an application to suit the new environment. Redevelop is also termed as re-architecting or re-engineering. Redevelop or redesign [17] replace the legacy look and gives a new look to the application. Redevelop and rebuild are very closely related to each other. Therefore, all the migration strategies of this section are correlated to each other, as shown in **Fig. 6**, and provide a common migrating activity of replacement, as illustrated in **Table 3** as well.

6.3. Component-Based Migration

This section explains the component format migration type (CFM) [16] that is used for PaaS and IaaS as shown in **Fig. 6**. In this migration type, the format of a component is changed into another format and not all the components are migrated. Only the required components are migrated. This is also done in the limited migration type [15] of PaaS. Similarly, the partial migration type [18] of PaaS migrates only a required subset of the components of an application. Reinstall [17] of IaaS suggests the reinstallation of a new version of the required software packages. Therefore, CFM, limited, reinstall, and partial migrate only the required components. Revise [18] can be used for all three cloud service models and can be related to redevelop and vertical (as explained in the replacement section). These deal with the enhancement of the code of a part of an application before migration. Revise suggests making some improvements in the code of

Table 4. Digital logic approach for cloud migration strategies.

	Migration Techniques / Migration Strategies	Complete	CBC	Change	NWNE	TBT	AT	LLC	Replace	FC	MCS	CET	Reinstall	Legacy s/w
Complete	Holistic	1	1	0	0	0	0	0	0	0	0	0	0	0
	Rehost	1	0	1	0	0	0	0	0	0	0	0	0	0
	Cloudify	1	0	0	1	0	0	0	0	0	0	0	0	0
	RD	1	0	0	1	0	0	0	0	0	0	0	0	0
	RR	1	0	1	0	0	0	0	0	0	0	0	0	0
	Iterative	1	1	0	0	0	0	0	0	0	0	0	0	0
	Complete	1	0	0	0	0	0	0	0	0	0	0	0	0
		1	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Replace	Replace	0	0	0	0	1	0	0	1	0	0	0	0	1
	Horizontal	0	0	0	0	1	0	0	1	0	0	0	0	0
	Standard Format	0	0	0	0	0	0	0	1	1	1	0	0	0
	Refactor	0	0	0	0	0	1	0	1	0	0	0	0	0
	Vertical	0	0	0	0	0	0	0	1	0	0	1	0	0
	Rebuild	0	0	0	0	0	0	1	1	0	0	0	0	0
	Redevelop	0	0	0	0	0	0	1	1	0	0	0	0	0
		0	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**0**	**0**
Component by Component	Revise	0	1	1	0	0	0	0	0	0	1	0	0	0
	Component Format	0	1	1	0	0	0	0	0	1	0	0	0	0
	Limited	0	1	0	0	0	0	0	0	0	0	0	0	0
	Partial	0	1	0	0	0	0	0	0	0	1	0	0	0
	Reinstall	0	1	0	0	0	0	0	0	0	1	0	1	0
		0	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

the legacy application to allow it to fit into the cloud environment and then rehost or refactor is applied, as shown in **Fig. 6**. All the migration strategies in this section migrate a unit or component or a part of an application to the cloud instead of a complete application or any replacements. This reflects the correlation among the migration strategies of this section across the cloud service models. **Fig. 6** presents a deeper and clear vision of consociation among the migration strategies. According to our study, the holistic, complete, and revise migration types can be used for all three cloud service models, i.e., SaaS, PaaS, and IaaS. However, the CFM type can be used only for PaaS and IaaS.

6.4. Digital Logic Approach

The digital logic approach used in **Table 4** justifies the similar migration activities performed by migration strategies of each cluster, as explained in Section 6. Binary digits 1 and 0 respectively are allocated according to the presence and absence of the migration activities in each migration strategy. Migration techniques are named as complete (whole application migration), CBC (component by component), change (any type of change), NWNE (no worry about new environment), TBT (tier by tier), AT (acceptable tiers), LLC (legacy look changed (code)), replace, FC (format of a component), MCS (module/component shifted), CET (component from each tier), RCM (required component migration), reinstall, and legacy software. The cloud migration strategies explained in Section 6 are represented in three clusters: complete, replace, and component-based.

Table 4 acts as a truth table. The AND operator is applied vertically to each migration activity and the OR operator is applied horizontally to the result of each section. By applying the AND operator to each section, we obtain 1 as a result for the migration activities under complete, replace, and CBC (component based). For the rest, we obtain 0 as a result. The output 1 indicates that all the migration types of each section perform a common migration activity. Furthermore, on applying the OR operator to the results of each section, we obtain the output 1, which meets our requirement. We thus obtain the output 1 for each section under the migration techniques of complete, replace, and CBC. The above truth table can be verified electronically using a digital circuit.

7. Migration Strategy Selection Process

Moving an application to the cloud has always been very risky. It is a challenging task to select an appropriate and successful migration type. Selecting an accurate cloud migration strategy should be a very simple process with clear milestones. One should not get mired in discussions over which strategy to adopt but should instead keep moving on a fixed course. One should have a clear and complete understanding of all the stages in selecting an accurate cloud migration strategy. The MSSP in this study provides this understanding and a series of steps to follow. The MSSP suggests the adjustment of the application for which the cloud migration strategy would be selected. The MSSP is quite scalable as it provides options at every stage, which makes it convenient for cloud consumers to make decisions and select a suitable set of tools. The MSSP provides the option of pilot testing, which can help to save money, time, and resources. Once the pilot testing results are verified, the selected cloud migration strategy is implemented on the target source. Research shows that the selection of a wrong migration type increases the cost and time required to an unexpected limit in several industries [25]. To reduce these types of risks, we proposed the use of a refined MSSP as shown in **Fig. 7**.

The details of the cloud terms are explained in Section 1 of this paper. The MSSP starts with a deep study of the goals, feasibility (e.g., business and technical), and requirements. In addition to this, the type of migration is also investigating in this phase. For example, we investi-

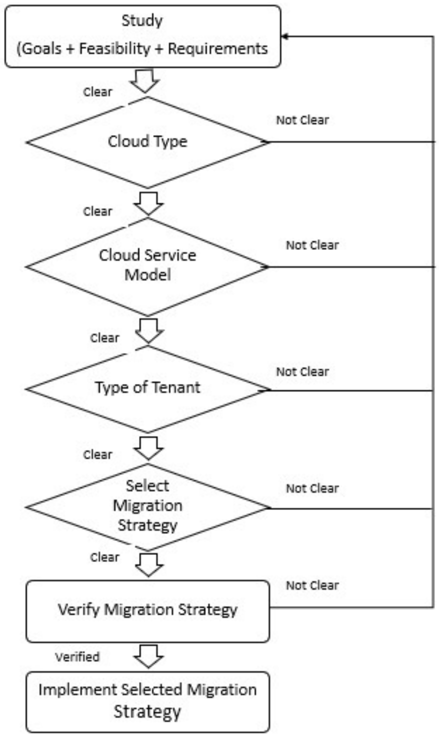

Fig. 7. Migration strategy selection process (MSSP).

gate whether a complete application or just some components of an application is required to be replaced/migrated to the cloud, whether the development of a brand-new application is required, or whether the re-engineering of a part of an application is required. Multiple questions should be investigated in the first phase to obtain clear objectives such that the entire process of selection of the migration type is completely based on the objectives and goals. If this phase is not defined and clearly understood, it is possible that the objective will not be met. After defining the goals, the next step of selecting a cloud type may be undertaken. The most commonly used cloud types are the public, private, or hybrid type. The community cloud type can also be selected based on the requirements. After selecting the most suitable cloud type according to the defined requirements, it is necessary to determine what type of cloud service is required. For this purpose, a cloud service model must be selected, i.e., SaaS, PaaS, or IaaS, in the next stage. Subsequent to selecting the cloud service model, the next step of determining the type of tenant must be undertaken. Else, the Study phase can be reviewed to ensure that the steps have been properly followed. The type of tenant, i.e., single or multi-tenant, must be determined in the next stage. In a single-tenant architecture, there is only one system hosted in the cloud, which means that one instance of an application is served by one customer. This tenant has its own copy of the software and database. No two customers can interfere with each other. Furthermore, customers can make modifications according to their needs. A customer can usually customize the application, maintain the schedule, make backups of the data, etc. without any restriction. However, in a multi-tenant architecture, a single instance of an application is run on the infrastructure of the service provider and multiple customers access the same instance and share the same database. Moreover, customers may interfere with each other. Data is logically separated and secured completely. In multi-tenancy, customers are not allowed to make any type of modification [26]. Single tenancy is generally recommended for use with SaaS in a particular cloud whereas multi-tenancy can be used with all three service models of the cloud [26, 27]. For migration, it is important to have the knowledge of the type of tenancy being handles. Based on the selection of single or multi-tenancy, the type of migration that is required and whether the application can be used for single or multiple users would become clear.

After selecting the type of tenant, the next step is to select the migration strategy (holistic, iterative, limited, etc.). If there exist any doubts, the first phase should be investigated again. Once the migration strategy is finalized, tests are performed to confirm whether the right strategy has been selected. This can be achieved by performing pilot tests such as the functional test, unit test, model test, integration test, or performance test according to your requirements. The pilot test [28] comprises the verification of a component of a system or the entire system under real-time operating conditions. It verifies the main working of a system before implementing the entire project. The verification in the MSSP is performed on a small part/model/blueprint/component of the legacy system. If satisfactory results are not obtained, the requirements may be revisited. Subsequent to the verification of the selected migration strategy, the component/application may be migrated accordingly. Subsequent to the implementation, the element may be monitored for successful working. Therefore, finally, after following the MSSP, the migration can be performed without any worry because it gives the closest and precise results.

8. Conclusion

The applicability of migration strategies for migrating a legacy application to the cloud has been a contentious issue in the practitioner literature. Our paper describes the selection and correlation among the cloud migration strategies for each cloud service model.

Our paper describes the extent of research on migration strategies for cloud service models, what are the most usable migration strategies, how the cloud migration strategies across the cloud service models are correlated and how a migration strategy can be selected.

Legacy applications in traditional business environments are pending migration to the cloud to exploit its benefits. The *k*-means algorithm of the partitioning clustering method has been used to analyze the cloud migration strategies in depth in various ways. The majority of the research conducted by researchers on migration strategies was found to be for PaaS, then for IaaS, and then for SaaS. As a result, the developers obtain the maximum support from researchers on the subject of migration strategies while programming. We presented a set of migration strategies for organizations that could be the most usable in the form of a web. Based on the cloud service models (SaaS, PaaS, and IaaS), this web provides an unambiguous visualization of the selected migration strategy. Therefore, organizations can now make quick decisions regarding migration strategies without going through the literature.

We presented the correlation among the migration strategies of the web based on similar features and formed three main clusters. This correlation among the migration strategies is across the cloud service models and was justified using the digital logic approach. Cloud consumers can easily connect the defined migration strategies across the cloud service models while working under hybrid conditions. The selection of a migration technique for applications to the cloud has always been very challenging. We presented a rapid MSSP that could aid in the selection of a suitable migration strategy according to the requirements. This work will prove to be a valuable contribution to the growing scientific knowledge base of cloud migration strategies. It also provides a foundation for the description of the methods of migrating applications to the cloud.

References:

[1] M. Bakery and R. Buyyaz, "Cluster computing at a glance," High Performance Cluster Computing: Architectures and Systems, Vol.1, pp. 3-47, 1999.

[2] C. S. Yeo, R. Buyya, H. Pourreza, R. Eskicioglu, P. Graham, and F. Sommers, "Cluster computing: High-performance, high-availability, and high-throughput processing on a network of computers," Handbook of Nature-Inspired and Innovative Computing, pp. 521-551, 2006.

[3] S. M. Hashemi and A. K. Bardsiri, "Cloud computing Vs. grid computing," ARPN J. of Systems and Software, Vol.2, pp. 188-194, 2012.

[4] H. Jin, "Challenges of grid computing," Int. Conf. on Web-Age Information Management, pp. 25-31, 2005.

[5] P. Mell and T. Grance, "The NIST definition of cloud computing," National Institute of Standards and Technology Special Publication 800-145, 2011.

[6] A. Huth and J. Cebula, "The basics of cloud computing," United States Computer Emergency Readiness Team, 2011.

[7] C. C. Rao and M. L. Y. R. Kumar, "Cloud: computing services and deployment models," Int. J. of Engineering and Computer Science, Vol.2, No.12, pp. 3389-3392, 2013.

[8] G. Polančič, G. Jošt, and M. Heričko, "An experimental investigation comparing individual and collaborative work productivity when using desktop and cloud modeling tools," Empirical Software Engineering, Vol.20, pp. 142-175, 2015.

[9] S. Bhardwaj, L. Jain, and S. Jain, "Cloud computing: A study of infrastructure as a service (IAAS)," Int. J. of Information Technology and Web Engineering, No.2, Vol.1, pp. 60-63, 2010.

[10] W. K. Assunção, R. E. Lopez-Herrejon, L. Linsbauer, S. R. Vergilio, and A. Egyed, "Reengineering legacy applications into software product lines: a systematic mapping," Empirical Software Engineering, Vol.22, No.6, pp. 1-45, 2017.

[11] R. Rai, G. Sahoo, and S. Mehfuz, "Exploring the factors influencing the cloud computing adoption: a systematic study on cloud migration," SpringerPlus, Vol.4, p. 197, 2015.

[12] W. Wu, W. Gentzsch, and J. Kern, "Dry-type transformer optimization using high performance cloud computing: Performance evaluation," IEEE SoutheastCon, pp. 1-2, 2016.

[13] W. Gentzsch, "How Cost Efficient is HPC in the Cloud? – A Cost Model for In-House Versus In-Cloud High Performance Computing," https://community.theubercloud.com/cost-hpc-cloud/ [accessed February 2, 2014]

[14] J. F. Zhao and J. T. Zhou, "Strategies and Methods for Cloud Migration," Int. J. of Automation and Computing, Vol.11, pp. 143-152, 2015.

[15] S. When, "Six Steps to Migration Project Success," Applications: A White Paper Series, Syntel Inc., 2006.

[16] T. Binz, F. Leymann, and D. Schumm, "CMotion: A Framework for Migration of Applications into and between Clouds," 2011 IEEE Int. Conf. on Service-Oriented Computing and Applications (SOCA), pp. 1-4, 2011.

[17] B. C. Tak and C. Tang, "Appcloak: Rapid migration of legacy applications into cloud," 2014 IEEE 7th Int. Conf. onCloud Computing (CLOUD), pp. 810-817, 2014.

[18] A. S. Ganesan and T. Chithralekha, "A Survey on Survey of Migration of Legacy Systems," Proc. of the Int. Conf. on Informatics and Analytics, p. 72, 2016.

[19] L. Rokach and O. Maimon, "Clustering methods," Data Mining and Knowledge Discovery Handbook, pp. 321-352, 2005.

[20] D. Sisodia, L. Singh, S. Sisodia, and K. Saxena, "Clustering Techniques: A Brief Survey of Different Clustering Algorithms," Int. J. of Latest Trends in Engineering and Technology (IJLTET), Vol.1, pp. 82-87, 2012.

[21] A. Saxena, M. Prasad, A. Gupta, N. Bharill, O. P. Patel, A. Tiwari, M. J. Er, W. Ding, and C.-T. Lin, "A review of clustering techniques and developments," Neurocomputing, Vol.267, pp. 664-681, 2017.

[22] V. Andrikopoulos, T. Binz, F. Leymann, and S. Strauch, "How to adapt applications for the Cloud environment," Computing, Vol.95, pp. 493-535, 2013.

[23] M. A. Babar, P. Lago, and A. V. Deursen, "Empirical research in software architecture: opportunities, challenges, and approaches," Empirical Software Engineering, Vol.16, p. 539, 2011.

[24] G. K. Juneja, "Use of Modeling Language to deploy applications in clouds," Int. Workshop on Maintenance and Evolution of Service-Oriented and Cloud-Based Systems (MESOCA), pp. 58-59, 2012.

[25] R. Harms and M. Yamartino, "The Economics of the Cloud White Paper," Microsoft Whitepaper, Microsoft Corporation, 2010.

[26] T. Garg, R. Kumar, and J. Singh, "A way to cloud computing basic to multitenant environment," Int. J. of Advanced Research in Computer and Communication Engineering, Vol.2, pp. 2394-2399, 2013.

[27] G. Kulkarni, R. Shelke, R. Palwe, P. Khatawkar, S. Bhuse, and H. Bankar, "Multi-tenant SaaS cloud," 2013 4th Int. Conf. on Computing, Communications and Networking Technologies (ICCCNT), pp. 1-4, 2013.

[28] S. Vogel and J. Draper-Rodi, "The importance of pilot studies, how to write them and what they mean," Int. J. of Osteopathic Medicine, Vol.23, pp. 2-3, 2017.

Name:
Mubeen Aslam

Affiliation:
Assistant Professor, National Textile University

Address:
32610 Seri Iskandar, Perak Darul Ridzuan, Malaysia

Brief Biographical History:
2005- Teacher Assistant, Government College
2006- Lecturer, Government Commerce College
2009- Lecturer, Government College University
2011- Lecturer, National Textile University
2018- Assistant Professor, National Textile University

Name:
Lukman bin AB Rahim

Affiliation:
Senior Lecturer, Department of Computer and Information Sciences, Universiti Teknologi PETRONAS

Address:
32610 Seri Iskandar, Perak Darul Ridzuan, Malaysia

Brief Biographical History:
2005- Assistant Software Engineer, Xybase Sdn.Bhd
2008- Lecturer, University Technologi PETRONAS
2014- Senior Lecturer, University Technologi PETRONAS

Main Works:
- Software and System Modelling
- Software Formal Verification

Membership in Academic Societies:
- The International Association of Engineers (IAENG)

Name:
Junzo Watada

Affiliation:
Professor, Department of Computer and Information Sciences, Universiti Teknologi PETRONAS

Address:
32610 Seri Iskander, Perak Darul Ridzuan, Malaysia

Brief Biographical History:
1982- Ryukoku University
1990- Osaka Institute of Technology
2003- Waseda University
2017- Emeritus Professor of Waseda University
2017- Professor, Universiti Teknologi PETRONAS

Main Works:
- Recipient of Henri Coanda Gold Medal Award from Inventico in Romania in 2002.
- "Decision making with an interpretive structural modeling method using a DNA-based algorithm," IEEE Trans. on Nanobioscience, Vol.8, No.2, pp. 181-191, 2009.
- "Value-at-Risk-based two-stage fuzzy facility location problems," IEEE Trans. on Industrial Informatics, Vol.5, No.4, pp. 465-482, 2009.
- "Dual Scaling Approach to Data Mining from Text Data Base," J. Adv. Compt. Intell. Intell. Inform., Vol.10, No.4, pp. 441-447, 2006.
- "Building Confidence-Interval-Based Fuzzy Random Regression Models," IEEE Trans. on Fuzzy Systems, Vol.17, Issue 6, pp. 1273-1283, 2009.

Membership in Academic Societies:
- The Japan Society of Fuzzy Theory and Intelligent Informatics (SOFT), Fellow
- The Institute of Electrical and Electronics Engineers (IEEE), Senioir Member
- Principal Editor, Co-Editor, and Associate Editor of sevral international journals, including ICIC Express Letters, Information Sciences, Int. J. of Systems and Control Engineering, and J. Adv. Compt. Intell. Intell. Inform.

Name:
Manzoor Ahmed Hashmani

Affiliation:
Associate Professor, Department of Computer and Information Sciences, Universiti Teknologi PETRONAS

Address:
32610 Seri Iskandar, Perak, Malaysia

Brief Biographical History:
1990- Senior Engineer, NS Solutions Corporation
2004- Professor, Mehran University of Engineering & Technology
2011- Professor, Iqra University
2016- Associate Professor, Universiti Teknologi PETRONAS

Main Works:
- Computer Science
- Communication Networks

Membership in Academic Societies:
- The Institute of Electrical and Electronics Engineers (IEEE)
- The IEEE Computational Intelligence Society
- The Pakistan Engineering Council (PEC)