TELKOMNIKA Telecommunication Computing Electronics and Control
Vol. 20, No. 2, April 2022, pp. 279~295
ISSN: 1693-6930, DOI: 10.12928/TELKOMNIKA.v20i2.20503

Monitoring and resource management taxonomy in interconnected cloud infrastructures: a survey

Vingi Patrick Nzanzu[1,3,4], Emmanuel Adetiba[1,2,3], Joke Atinuke Badejo[1,3], Mbasa Joaquim Molo[1,3], Claude Takenga[3,4], Etinosa Noma-Osaghae[1], Victoria Oguntosin[1], Sadeeq Suraju[3]
[1]Department of Electrical and Information Engineering, College of Engineering, Covenant University, Ota, Nigeria
[2]HRA, Institute for Systems Science, Durban University of Technology, Durban, South Africa
[3]Covenant Applied Informatics and Communication African Center of Excellence, Covenant University, Ota, Nigeria.
[4]Département de Génie Electrique et Informatique, Faculté des Sciences et Technologie Appliquées, Université Libre des Pays des Grands Lacs, Goma, RD Congo
[5]Infokom GmbH, Entreprise NTIC, Neubrandenburg, Germany

Article Info

Article history:

Received Apr 01, 2021
Revised Jan 12, 2022
Accepted Jan 20, 2022

Keywords:

Cloud computing
Cloud monitoring
Cloud resource management
Federated cloud

ABSTRACT

Cloud users have recently expanded dramatically. The cloud service providers (CSPs) have also increased and have therefore made their infrastructure more complex. The complex infrastructure needs to be distributed appropriately to various users. Also, the advances in cloud computing have led to the development of interconnected cloud computing environments (ICCEs). For instance, ICCEs include the cloud hybrid, intercloud, multi-cloud, and federated clouds. However, the sharing of resources is not facilitated by specific proprietary technologies and access interfaces used by CSPs. Several CSPs provide similar services but have different access patterns. Data from various CSPs must be obtained and processed by cloud users. To ensure that all ICCE tenants (users and CSPs) benefit from the best CSPs, efficient resource management was suggested. Besides, it is pertinent that cloud resources be monitored regularly. Cloud monitoring is a service that works as a third-party entity between customers and CSPs. This paper discusses a complete cloud monitoring survey in ICCE, focusing on cloud monitoring and its significance. Several current open-source monitoring solutions are discussed. A taxonomy is presented and analyzed for cloud resource management. This taxonomy includes resource pricing, assignment of resources, exploration of resources, collection of resources, and disaster management.

Corresponding Author:

Emmanuel Adetiba
Department of Electrical and Information Engineering, College of Engineering, Covenant University
Km 10, Idiroko road, Ota, Ogun-State, Nigeria
Email: Emmanuel.adetiba@covenantuniversity.edu.ng

1. INTRODUCTION

Cloud computing is an emerging paradigm that provides its customers with practically indefinite storage and computing power from anywhere [1]. It enables universal and straightforward access to different computer resources such as networks, storage, devices, services, and applications. Cloud services providers (CSPs) can dynamically utilize these resources with a minimum administrative effort and interaction [2], [3]. Cloud computing provides numerous service models, primarily infrastructure, platform, and software as a service (IaaS, PaaS, and SaaS, respectively), which facilitate its adaptation. Infrastructure as a service (IaaS) offers infrastructure-based services to cloud customers, including storage, computation, and network services. Platform as a service (PaaS) removes the need for companies to manage and develop conventional

platforms to construct applications. Software as a service (SaaS) provides software hosting [4]. Cloud computing implement four models of deployment: i) public cloud, a multi-contributor framework that enables numerous tenants to share various web server resources; ii) private cloud, a single-contributor framework that allows the usage of cloud services by consumers; iii) hybrid cloud for private and public CSPs to share resources; and iv) federated cloud, in which numerous CSPs operate together under one authority [5].

Being a new fundamental concept in cloud computing, the interconnected cloud computing environment (ICCE) paradigm has gained significant attention. It provides various CSPs a large variety of connections and partnerships. CSPs share their resources through the regulated federation in the ICCE paradigm, also known as the inter-cloud, federated cloud, multi-cloud, or interconnected cloud [6]. The most apparent explanation behind ICCE is the finite physical resources in a single provider's resource pool. One of the main features of ICCE is the interoperability of the framework. Both CSPs and customers benefit from various potential cloud situations as cloud interoperability occurs, this is highlighted by Figure 1. Interoperability in clouds necessitates CSPs to introduce and implement collaborative norms, interfaces, protocols, formats, and architectural components. CSPs and their customers have benefited from an interconnected cloud ecosystem in many ways. Consequently, cloud interoperability has important motivations for preventing vendor lock-in, scalability, availability, low latency, and energy efficiency [7], [8].

Resources and the cloud environment restriction, making it difficult for CSPs to install more resources in the scenario that services and data centers are extended and built. Therefore, keeping in mind resource constraints, it is likely that various CSPs work together to establish a federation to assign resources and expand their scope. Based on many administrative areas with many resource usage policies, an adaptive framework for handling all additional resources is needed to upgrade access rights privileges. The effective management of resources is crucial for both members and direct consumers of the federation [9]. For small organizations involved in the federation, this process is especially important as it gives them the ability to expand their business with minimal resources. Resource management: work scheduling and resource provision among locals are a few of the main problems for CSPs [10], [11]. Monitoring of resources may help system administrators monitor cloud computing platform resources. System administrators can understand the operating status of cloud computing components, and they can take adequate actions before cloud computing platforms fail to function. The question is how the main aspects of resource management can be handled and monitored. Resource management and monitoring in the ICCE include resource pricing, resource selection, resource allocation, resource discovery, and disaster management.

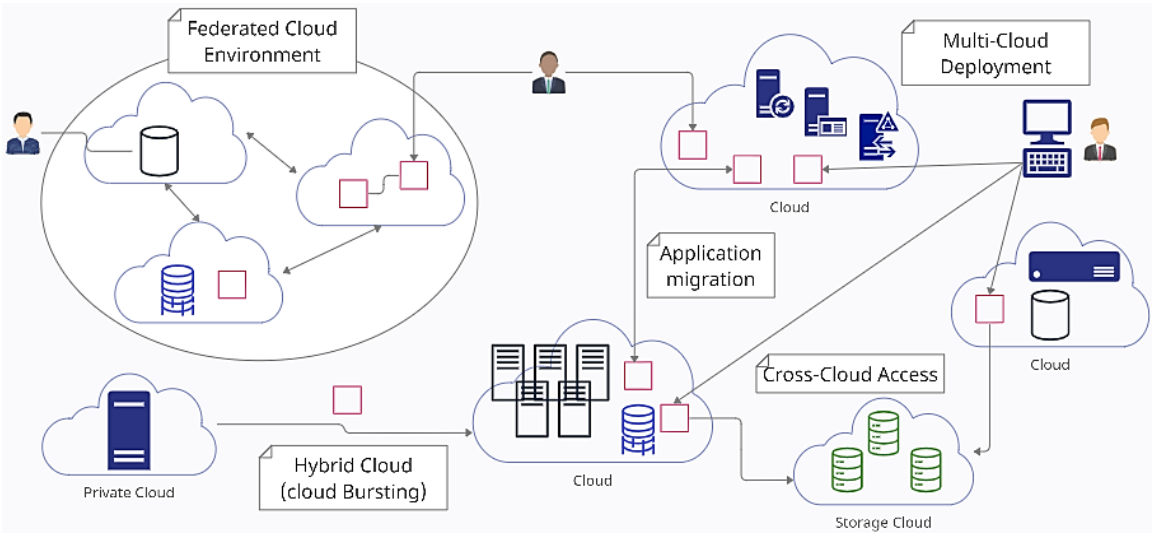


Figure 1. Scenarios and interoperability in the interconnected cloud computing environment

Surveys on cloud monitoring have been discussed in the literature [12]-[14]. However, these surveys focused on the taxonomies, terminologies, meanings, and challenges of cloud computing monitoring. In essence, this research differs in the sense that we concentrate on the monitoring aspects of ICCE. Our contributions are as follows: i) the monitoring mechanisms are analyzed according to the pricing of resources, selection of resources, resource allocation, the discovery of resources, and management of disasters;

ii) open challenges for resource monitoring are provided; and iii) researchers are motivated to address the problems that have been discovered.

The rest of the work is divided into four parts. Section 2 offers an overview of cloud computing, and the ICCE paradigm. Section 3 provides ICCE resource management, and monitoring. The classification of open-source monitoring functions and the discussion of state-of-the-art literature are given in section 4. The paper concludes with section 5.

2. INTERCONNECTED CLOUD COMPUTING ENVIRONMENT

This section provides a highlight on the background information regarding cloud computing environment, multi-cloud environment, and federated cloud environment. Cloud computing is a developmental technique that has arisen to more efficiently change the look of information technology (IT) strategies in organizations. Cloud computing's basic concept is to distribute computational resources over the Internet as services [15]. According to the US Department of Commerce (National Institute of Standards and Technology), cloud computing is a model for allowing omnipresent, easy, on-demand network access to a pool of customizable computing resources (e.g., networks, servers, storage, applications, and services). These computing resources can be quickly provisioned and delivered with negligible management involvement or contact with CSPs [16]. Cloud computing consists of five main features: self-service on-demand, broad network access, pooling of resources, rapid elasticity, and measured service. When compared to traditional IT models, cloud computing provides a variety of benefits [17]. For instance, faster data transfers, elasticity, resource sharing, pay-per-use, flexibility, ease of setup, low IT implementation costs, the need for data centers, and increased IT efficiency. In the cloud world, the user needs to be distinguished, i) the end user identifies the registered application as a resource and may not know that it runs on cloud computing; and ii) the user or consumer is a cloud resource administrator. To conduct business, customers do not have to invest in a large computer infrastructure; instead, they can purchase cloud computing services based on their requirements [5]. Besides, there is more than one service model for cloud computing. Hence the presence of an X as a service (XaaS), the X is used as a collective term for any other service [16]. Cloud computing resources are dispersed geographically in virtualized and distributed environments due to the great scalability rate offered. It makes cloud computing suitable for companies that can delegate some operations, which leads to a substantial reduction in physical infrastructure costs [18]. Although open-source clouds like OpenStack, Eucalyptus, OpenNebula, and CloudStack are more and more expanding, it is not easy to migrate to the cloud infrastructure, especially for critical services such as banking, and health. Cloud computing, however, is not without its challenges, these challenges include data protection and privacy issues, unequal availability of services, restricted compatibility with existing apps and systems, and some poor regulatory structures [4]. In respect of searching to mitigate these challenges, new architectures are proposed by researchers, hence the advent of ICCEs.

2.1. Multi-cloud

Multi-cloud, also called the cloud of clouds by some practitioners and experts. This term is used to demonstrate it is necessary to extend the cloud computing environment to several unified and interconnected CSPs [19]. It also indicates that a unique CSP should not be entrusted with sensitive data to avoid dependency. Multi-cloud is a cloud strategy that allows organizations to rely on two or more cloud computing platforms to perform several tasks. It enables the organization to select the service that best suits their needs, and commonly, companies in various parts of their business or for different use cases call different CSPs. Professionals often interchange concepts like Inter-cloud and cloud federation alongside hybrid cloud because all emanate from multi-cloud [20], [21]. However, some experts prefer staying inflexible and produce some key differences between the two concepts. The principal distinction between the federation and the inter-cloud is the fact that the inter-cloud focuses on potential standards and open interfaces while the federation considers the use of a supplied interface version. From the several discussions that took place, these experts agreed that federation is more of a prerequisite for the inter-cloud real target. Clouds tenants have to merge and interoperate their resources with the inter-cloud vision, and everybody will have a unique perception of how applications should be deployed. Moreover, the federation concept emphasizes the presence of the users of the resources [5].

2.2. Federated cloud

As earlier stated, one of the significant disadvantages of cloud computing is that small CSPs do not have sufficient capacity to cope with peak demand. Maintaining resources to meet high demand may serve as a palliative function and contributes to wasting energy and resources, thereby raising costs and likely decreasing resources over their life cycles [22]. With time, several CSPs started pooling resources to build the federated cloud to address these limitations.

The federated cloud infrastructure operated by a federated cloud broker provides a single mechanism for managing different clouds. In sharing their resources with the federated ecosystem, each cloud participant makes an agreement with the federated broker. This arrangement covers all technological and financial implications of the cloud federation. The cloud resources federation helps companies to share their workloads globally or migrate data through fragmented cloud networks. The benefits that the CSPs derive from the federated approach are significant: global demand is rising fast and easily outstripping any public cloud provider's capacity. Not to mention private clouds and their desire to burst into the public cloud scenario to achieve a hybrid cloud paradigm [23].

While several federation methods have been discussed, they are at risk of working with untrusted CSPs, leading to the deterioration of performance. Imagine the case of a CSP who does not have enough resources to meet the demanded virtual machines (VM) and wants to subcontract some of the needed VMs from other members of the federation. In the process of creating the federation, if the trust concern is not treated, many reasons will prevent specific CSPs from satisfying user requests. These include insufficient maintenance that results in repeated downsides, poor safety that causes compromised nodes or nodes to be slowed down by a distributed denial-of-service attack (DDoS), and a lack of agreed load capability. In addition, a given CSP may be self-centered, refusing to share sufficient resources. These kinds of CSPs contribute to the deterioration of results, decline in gains, low satisfaction of users, and breaches of the service level agreement (SLA) [24].

3. RESOURCE MANAGEMENT AND MONITORING IN ICCE

This section is about resource management and its taxonomy. Summaries of the taxonomy of resource management is presented. Resource management functions transcend the fact that various CSPs in ICCEs have different undelying technologies. This section also provides an overview of resource monitoring in ICCEs.

3.1. Resource management

Management of resources is the main feature of any man-made system, influencing the three fundamental system assessment criteria: performance, functionality, and cost [25]. Resource management (RM) is a broad area of research composed mainly of many interconnected aspects. RM is a technique used in the procurement and release of resources [26]. To provide performance isolation and efficient use of hardware resources, RM is regarded as one of the impactful aspects of cloud computing. Moreover, all the resources shared among the user are virtualized. Therefore, the consequence of the virtualization of the resources leads to the appearance of challenges due to resource management-related tasks [26], [27]. Also, the management of resources is a highly challenging task [28]. The heterogeneous, dynamic, and complex nature of users request that always fluctuates according to the need has made the management difficult.

Resource management involves complex decisions and policies due to the complex nature of the fracture. Therefore, transparency between all the stakeholders within an ICCE is a critical aspect of resource management. Everyone among the stakeholder, monitor the utilized resources for much various reason. The monitoring of resources may help the CSP to detect malfunctions in hardware or software within the environment or ideally optimize the distributed resources among users, while for a cloud client, the motivation behind the resource monitoring is to detect either the overspending of resources or resource shortcoming [29].

Almost every resource is virtualized and shared among several users in cloud environments. Virtualization involves a number of difficult tasks, especially for ICCE resource management. RM regularly deals with different aspects of the ICCE, including allocation, selection, discovery, and pricing of the resources alongside disaster management.

Various techniques are developed for efficient management of ICCE, such as awareness of energy efficiency, SLA awareness, load balancing, network load minimization, and profits management. In order to provide the best RM solutions, multi-criteria optimization techniques are also implemented. However, if the metrics under consideration are conflicting, multi-criteria optimization will introduce additional aspects and challenges to RM taxonomy. It is due to the fact that conflicting metrics depend on each other, and improving one results in degrading the performance of the others. Some examples of the conflict include energy efficiency and SLA violations, energy efficiency and network load, and SLA violations and maximizing profit [30]. The taxonomy of RM shown in Figure 2 is intrinsically linked to the aforementioned metrics such that any poor performance of a metric affects different aspects of RM.

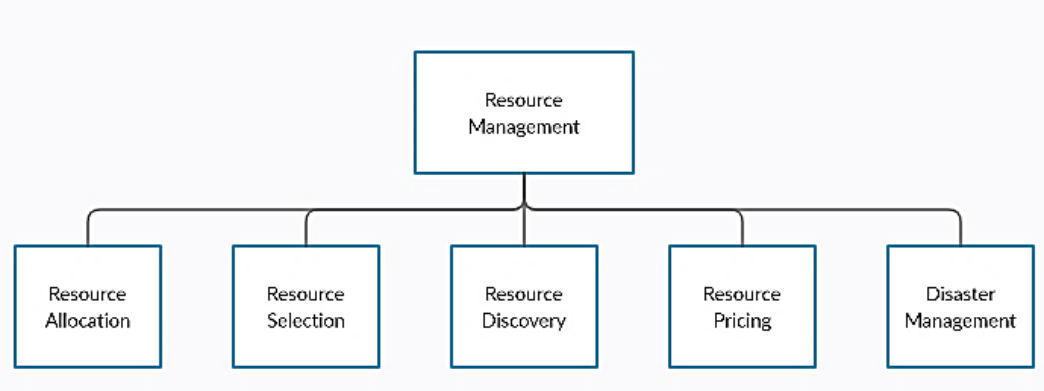


Figure 2. Resource management taxonomy

3.1.1. Resource allocation

The allocation of resources is integral to obligating CSPs for uncertain resource requirements and financial returns. In general, a singular CSP may not offer limitless services with finite physical resources. One potential solution could therefore be the federation of multiple clouds. Resource allocation helps CSPs to decide on the distribution of resources in order to overcome the problem of resource competition within the ICCE SLA [31]. For a variety of practical scenarios, the impact of ICCE over a single provider for allocating components of distributed applications raises concerns about the impact of different strategies on particular service workflow. Table 1 provides a summary of some schemes of allocation of resources.

Table 1. Summary of schemes of allocation of resources

Scheme	Objectives	Limitation
[32]	Managing unpredictable fluctuations in the workload while ensuring that SLA restrictions are coordinated in multiple geographical clouds.	The abilities to run instances were not studied during decision making and the network latency was also overlooked.
[33]	Network and computer resources are optimized and treated together so that virtual resources are dynamically allocated to physical resources within the cloud network.	Finds the dynamic infrastructures and environments that are heterogeneous.
[34]	Taking into account the possible contradiction between the CSPs interests and the cloud client.	Unforeseen situations and environmental deviations.
[35]	To satisfy end-user quality of service (QoS) and scale savings without increasing and improving certain physical resources.	The issue here is that the resources are modeled as a single type.
[36]	To examine the advantages of assigning distributed application components on several public clouds.	The time to search for an appropriate node in a very big ICCE with large workflows.
[37]	To handle cloud scaling while running CPU-intensive applications in a multi-cloud environment with unproportional cost-to-performance ratios.	The decision on the schedule does not take the fault tolerance into account.
[38]	To maximize energy efficiency by addressing consumer credibility awareness and minimal interaction resource allocation issues.	Spectrum efficiency is not handled.

The notion of sharing workload among different resources is referred to as load balancing. One important feature of ICCE is load balancing that defines the way workload is allocated to a specific resource. Therefore, after using the load balancing algorithm, tasks are migrated between physical and virtual machines. In addition, RM techniques to fit this kind of aspect need to be developed so that resources can share their workload.

3.1.2. Resource selection

The resource selection process includes a configuration to fulfill all user needs and optimize infrastructure. The techniques implemented by different CSPs make it difficult to pick useful resources from the ICCE resource pool. Moreover, the ICCE resource selection is a traditional multi-attribute decision-making problem, with issues to solve: quantification of cloud service attributes, user preference weight coefficients, and cloud resource ranking [39]. Different CSP requirements, the high complexity of the algorithms, and dynamicity make it difficult to choose valuable resources in a federated resource pool. An optimization algorithm that considers all the variables influencing the allocation is the key to a resource or service selection in ICCE. In order to maintain a level of user satisfaction, the selection process should also

take into consideration all the behavioral aspects of the environment. Table 2 presents the summary of developed resource selection strategies. Energy efficiency is one of the key issues in the ICCE. One possible way to solve this problem is by consolidating the workload, which reduces energy use by consolidating more workload on fewer servers. This is only achieved by performing relevant resource selection.

Table 2. Summary of developed resource selection

Scheme	Objectives	Limitation
[40]	To simplify and speed up the CSP search for the best vendor selection.	Users have no possibility to negotiate certain SLA terms.
[41]	When there is insufficient information about CSPs and their services, automate service selection.	The maintenance of the agents may be a challenging concern in the case of service migration.
[42]	To choose the best provider in terms of user costs and needs.	To integrate the CSP database and its characteristics the systems need human interaction.
[43]	Selection of infrastructure services automatically for SaaS to capture CSPs SLA claims to their customers.	No detection of violations of SLA and penalty.
[44]	To sustain the dynamic load and identify the best spot to assign the request to improve performance.	Data centers are not involved in decision-making on the failure index and energy consumption index.
[45]	Selecting reliable cloud services for cloud users.	The granularity of past data for decision-making is not taken into account, ensuring that outdated information does not influence decision-making.
[46]	Incorporating a machine-learning classifier trained on practical past workloads into resource selection.	In the case of workloads that are predictable, well-known, and do not change, the system is useless.

3.1.3. Resource discovery

There are multiple machines/nodes in an ICCE, and each node has various resources available. Only a subset of all cloud nodes is known to any node. It is difficult to know all the nodes since information about the entire Internet is required. A user wishing to do specific processing that needs some resources (either processor power or amount of memory) must find enough nodes to fulfill their resource requirements in order to complete the computation task [47]. Resource discovery provides the ability to know which resource is available and the VM holding it. The Resource discovery feature describes how a CSP displays its resources and services so other members of the ICCE can find the resources and services needed to automate and easily utilize the process of selection, thereby fulfilling requests. Schemes such as adaptative resource discovery strategy, decentralized resource discovery by employing a meta-brokering component, constraint-based resource discovery model, scalable gossip-based hybrid multi-attribute overlay for the discovery of resources, set of natural-based and auto-organizational peer-to-peer (P2P) algorithms, and an improved collective discovery efficiency of basic cloud services for designing a workflow, and various novel framework for discovery of resources were proposed by researchers in order to efficiently manage the available resources in the ICCE. Table 3 provides a resource discovery schemes summary.

Table 3. Summary of ICCE resource discovery schemes

Scheme	Objectives	Limitation
[48]	A tolerant and fault-finding approach for discovering resources in ICCE.	Transparency in data exchange between meta-brokers can lead to the exposure of privacy of a CSP.
[49]	To design mechanisms to discover resources for the multi-cloud environment.	Interoperability is based on a higher amount of CSPs.
[50]	Resolves resource discovery queries based on network traffic with lower overhead and more efficiency.	The increase in the time of finding is linear.
[51]	To develop a distributed virtual network and resource detection framework on a semanticized basis over networking innovations over virtualized infrastructures (NOVI)-federated testbeds.	It is designed for large scientific testbeds and less practical for a commercial CSP federation.
[52]	To increase the efficiency of data centers during VM placing.	Scalability not taken into account.
[53]	to propose a scientific work scheduling hybrid resource discovery framework.	In distributed environments, scalability is difficult to handle.

Network load is the quantity of traffic flow within a network at a particular time. The sharing of information between resource managers, VM positioning, inter-VM communications, and VM migration can lead to an explosion of networking load in ICCE. High network loads lead to system performance degradation since the CSPs have to wait until VMs are re-placed and critical communication between VMs

is consequently delayed. Therefore, there is a need to effectively deal with the discovery of resources in order to minimize the traffic through a network.

3.1.4. Resource pricing

Within an ICCE, CSPs deal with user requests that change over time from one CSP to another. Considering the different information taken while monitoring the resources and the fact that every cloud provider wants to remain competitive, pricing is a crucial aspect that should be taken into consideration [54]. This is to ensure that the SLA between CSPs is not violated on the one hand and between CSP and users on the other hand. Thus, a dynamic resource pricing is to be promoted within the ICCE due to the fact that users' requests move from one VM to another in order to highly benefit from the quality of service offered [55], [56]. The supply and demand of resources fluctuate in an ICCE when consumers and suppliers join or leave the environment. The price function is then used to manage the consumer and provider's individual rationality. The workload oscillation and the availability of the resources in the federation are restricted to the requirement of dynamic ICCE pricing strategies based on demand and supply principles. Different authors have developed numerous strategies to handle the pricing functions in standalone single cloud systems and ICCEs. Table 4 provides a summary of several price schemes for resources.

One of the important questions to take in the management of resources is quality of service (QoS). Since CSPs aim to provide customers with the best possible performance, a service level agreement between the parties should be concluded and its terms followed. SLA includes details on the required service level and price. Hence SLA violations are worthy of attention.

Table 4. ICCE resource pricing schemes summary

Scheme	Objectives	Limitation
[57]	Dynamic price model with proactive evidence for the distribution of various kinds of common resources.	Single point of failure if the auctioneer fails.
[58]	Calculate the balance price to increase cooperation among cloud providers in an environment based on federation.	The balance price can sometimes not be achieved.
[59]	Taking pricing and capacity planning into consideration together.	Is not responsible for SaaS providers' optimal behaviors.
[60]	Impact on future demand and supply pricing elasticity.	Surplus supplies.
[61]	Examine the price effect on the number of rejections of the service.	In the audit, new contracts are not evaluated.
[62]	Pricing regulation governance for heterogeneous CSPs	Do not take into account SLA problems
[63]	The true mechanism of on-line auction-based on user assessment and cost.	Punish users who have a bad reputation by paying more.

3.1.5. Disaster management

Disaster management and tolerance for faults play a key role in recovering organizational data on natural disasters or caused by human beings. Disaster management functions allow for a system or a device, despite failures or corrupted hardware/software, to continue operating normally [29], [30]. For ICCE disaster situations, the management functions must be distributed and coordinated between each node of the federal set up to allow for a disaster-aware micro-level process that gives the level of QoS that members of the federations require. Table 5 provides a summary of some disaster management schemes.

Table 5. Summary of some disaster management schemes

Scheme	Objectives	Limitation
[64]	Plan the back-up of business-related critical data across several geographical locations.	In such a wide variety of sites, the security mechanism is really important.
[65]	Having a warm way to update and set up a standby cloud system.	In such regular updates, some computations/data may be lost.
[66]	Consider providing a tolerant resource in a cloud environment with a low-cost fault formulation.	It does not take into account factors such as makespan, system performance, and load balance.
[67]	Big data System disaster recovery.	As the mechanism for replicating various sources is too expensive, an intelligent mechanism should be used to reduce expenditure and automate certain tasks.
[68]	To ensure there is no cybersecurity violation of cloud data.	The main limitation of the research is the use of trojans and viruses as the unique known vulnerabilities.
[20]	Determining the minimum replica number to reduce storage space, storage costs, and recovery times.	Works in a single cloud environment

The scenario is that when a disaster occurs the primary site will be unavailable, and the secondary site should be activated. In this case, there is no synchronous or asynchronous replication capability in the backup site, but only the local system and data states can be saved. This situation leads the emergency system to be seriously threatened but temporary and will be removed after the primary site has been recovered.

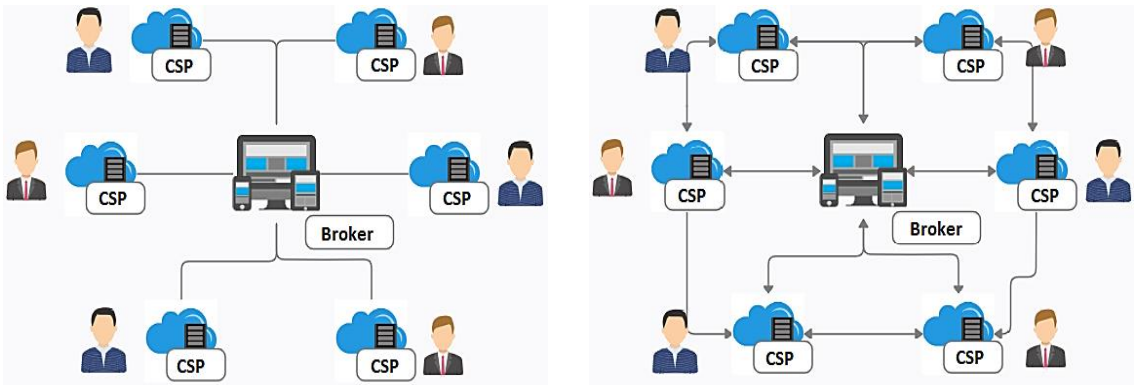
3.2. Resource monitoring

ICCEs are heterogeneous systems that combine several CSPs for providing unlimited resources to users. However, this task is not always easy to achieve. Therefore, ensuring that the quality of service (QoS) is at its high level is a very challenging task. CSPs should always be updated on the resource status of each other with the federation. Monitoring is at its simplest a three-phase process: the compilation of the respective state, the analysis of the aggregate state, and the decision-making as the conclusion [69]. Unix tools used for monitoring include, for example, df, uptime, or top. These tools are run by a user who analyzes the state of the system and determines the possibility of making a decision. The user, therefore, actually conducts the greater part of the monitoring and not software. With computer systems growing in size and complexity, automated tools are increasingly required for monitoring with reduced or eliminated human interaction requirements. These systems incorporate the 3-stage monitoring process in whole or in part. Each stage has its challenges, especially in the field of ICCE. There is many common monitoring motivating factors for basically all computing areas, including ICCE. These include scheduling capability, failure or detection of results, the discovery of redundancies, system evaluation, and detection of policy violations.

After considering the different aspects of the RM technique, it is relevant to bring out some challenges regarding the management of resources: first of all, users request in a cloud federation change overtime and migrate from time to time from one cloud provider to another, the management of resources in term of resource scheduling, load balancing and resource allocation are more complicated since the way that resource is distributed among CSPs is affecting SLA directly in terms of the economic interest. The heterogeneous aspect of the federation renders the management of resources to be more difficult.

4. MONITORING SYSTEMS AND THEIR ROLE IN THE RESOURCE MANAGEMENT TAXONOMY

This section covers two main areas. The first part discusses the review on the state-of-the-art in cloud monitoring technologies. The second part offers a thorough overview of the role of monitoring, solutions in the taxonomy of resource management. Moreover, this section provides open challenges for resource monitoring in ICCE. Figure 3 presents architectures of monitoring in ICCE, Figure 3(a) centralized and Figure 3(b) decentralized architectures of monitoring in ICCE.



(a) (b)

Figure 2. Architectures of monitoring in ICCE: (a) centralized and (b) decentralized

4.1. Survey of monitoring solutions

Cloud monitoring solutions are systems built from the ground for cloud implementation monitoring. Such systems are traditionally conscious of cloud principles such as elasticity, various availability zones, VM, and other cloud-related manifestations. These tools have a conceptual benefit over non-cloud-conscious tools because they can scale and adjust to cloud functionality while their infrastructure changes.

4.1.1. Monitoring platform as a service (MonPaaS)

The rapid growth in cloud usage poses some big challenges. The absence of information is one of the problems with the metric data setup. Another concern is the lack of virtualized protection resources. These issues were dealt within [70]. The study introduced an adaptive monitoring platform as a service (MonPaaS). MonPaaS is an open-source, accessible online, and implementable tool. The author also proposed two different ways of monitoring: CSPs monitoring and user monitoring. In addition, nagios was integrated within OpenStack [71]. The power of MonPaaS is its ability to intercept the message queue of OpenStack and to update VM information by using messages. Both cloud providers and consumers receive the MonPaaS module as an API. This introduces a different VM monitoring (MVM) feature for any new cloud user. MonPaas monitors physical and virtual resources and updates any improvements in virtual or physical infrastructure. It conducts agent-free controls that ensure a high degree of protection for customers. The downside of MonPaas lies in the way that it constructs separate MVM; additional physical resources are required. MonPaaS is an improved version of IaaSMon [72].

4.1.2. Nagios

Nagios is a well-known standard monitoring tool that monitors various deployments of servers. It is an open-source tool that, in its simplest configurations, is built upon a two-tier hierarchical architecture [73]. The server playing the role of monitoring is populated with a configuration file that details all the monitored servers with their services. Nagios, in the process of monitoring, generates a schedule then probes all the servers, in addition, examines each service according to the schedule. Nagios doesn't suit perfectly with cloud monitoring. A large amount of manual configuration is needed, including the need to adjust configuration when controlled VMs are instantiated and terminated.

4.1.3. Zabbix

Zabbix was firstly introduced in [74] work. The monitoring solution Zabbix is built for server/agent architecture. The Zabbix server operates on a separate machine so that data sent by Zabbix agents can be collected and aggregated to track. The Zabbix solution supports a warning system that activates when predefined events and conditions occur, such as when memory consumption reaches 80%. These warnings are helpful in that the stimuli activate preparations for adaptation, such as measures for elasticity. Structured query language (SQL) databases are used to store calculated measurements, and data is accessed via a web front-end and an application programming interface (API).

4.1.4. Private cloud monitoring system (PCMONS)

Private cloud monitoring is a concern as most company cloud solutions are highly costly. To solve this issue, a cloud monitoring open-source architecture has been proposed in [75]. The architecture proposed is split into three layers: i) infrastructure, ii) integration, and iii) view. The layer of infrastructure includes the corresponding hardware, software, and system. Alternatively, an abstraction layer addresses the virtualization method and hypervisors. The layer view is responsible for providing the monitoring graphical user interface (GUI) which is known as the dashboard. This GUI will provide different views for different users, depending on their needs. Based on this architecture, a private cloud monitoring scheme, called private cloud monitoring system (PCMONS), was introduced.

4.1.5. New-generation-monitoring (Ngmon)

In clouds, the multi-tenancy and complexity of a distributed virtualized system present new problems with cloud monitoring. In terms of data representation, storage, processing, and delivery, cloud monitoring faces challenges. A cloud monitoring method has been suggested in [76] to focus on data collected from cloud monitoring VMs. This approach underlines the fundamental concepts of data monitoring and issues relating to data representation, storage, transmission, and delivery monitoring. This study for a proof of concept incorporates a technical approach called new-generation-monitoring (Ngmon). The data collection feature of Ngmon is defined as logs, warnings, and business operations on an event-based framework and stored as a specified structure in typed data form. Data are encoded in JavaScript object notation (JSON) format in the next stage as an event entity. To ensure safety-based authentication, the access control list (ACL) stands. The query assessor is applied when there is a connection for querying/responding to track user data. A published/subscribed software is used for monitoring customer's basic needs, while a hybrid communication model was adopted in the proposed solution for presenting a transmission control protocol (TCP) system protocol. Ngmon also provides support for secure sockets layer (SSL) encryption.

4.1.6. Global monitoring system (GMonE)

Inefficient monitoring methods for broadly distributed infrastructures face problems due to the complexity of cloud systems. To overcome these problems, [77] proposed the cloud monitoring architecture called global monitoring system (GMonE). The architecture of GMonE consists of four main components:

global monitoring system monitor (GMonEMon), global monitoring system database (GMonEDB), global monitoring system access (GMonEAccess), and monitoring plug-ins. GMonEMon operates based on the required components to collect and send metric data to GMonEDB. Monitoring plug-ins in the form of applications for monitoring execution are part of GMonEMon. GMonEDB gathers and maintains GMonE tracking data as a database. The GMonEAccess GUI enables a user to have easy access to data. To prove their theory, [78] used OpenNebula to test GMonE on Grid'5000. The findings have shown that the total regulation of computer and communications resources is constrained in terms of consumption. The research proposed sharing messages using the published/subscribed model. Publish/subscribe is confronted with problems that affect the reliability factor, including rigid semantic relations and communication problems [79].

4.1.7. Distributed architecture for resource management and monitoring in clouds (DARGOS)

Cloud administrators establish policy provisions based on full awareness of the infrastructure's physical resources and facilities. A robust and up-to-date cloud infrastructure awareness is challenging if multi-tenant and complex cloud stacks were taken into account. To address this problem, Povedano-Molina *et al.* [80] developed the distributed architecture for resource management and monitoring in clouds (DARGOS). DARGOS enables the customer to monitor granularity in line with its requirements with flexibility. Povedano-Molina *et al.* [80] presented the proposed architecture, DARGOS, as an OpenStack project.

4.1.8. Monitoring service level agreement (MonSLAR)

In the cloud usage and CPS relationship, customers' satisfaction levels to the agreed service level agreement (SLA) is paramount. Most cloud monitoring solutions currently offer little thought for calculating the earlier stated parameter on the user side. Authors have addressed this problem in [81] with an architecture for residual resources management called a monitoring SLA (MonSLAR). This study is an ongoing project designed to develop MonSLAR as an efficient method. From the proposed architecture, the authors projected two control rates: the user and the CSP. MonSLAR provides users with information on the approved SLA-based service level that is or is not achieved. By collecting quality of experience (QoE), which the cloud service provider uses as a key performance management measure to meet SLA, MonSLar provides a benchmark for customer satisfaction for the service provider. The main aim of their study is to follow the SLA criterion. However, it doesn't explain why the SLA violation, and based on what metrics it should be calculated.

4.1.9. Near field monitoring (NFM)

Cloud monitoring applications find cloud environments in which VMs and containers are not primarily part of the cloud operating system. Containers and VMs are created and killed with incredible time precision in a cloud environment. Therefore, cloud monitoring calls for the validation of this claim in a specific way. Suneja *et al.* [82] suggested a brand-new near field monitoring (NFM) to address this problem. NFM uses a new tracking form where the monitoring agent will not interact with or mount in the host VM to provide monitoring and analytical operating services. In NFM, a user/host can opt-in and out, as there is no additional VM/container application or resource. NFM offers a broad view of tools and VMs through a cloud provider. To obtain measurements, the authors used kernel data. More than 1000 Linux flavors were also tested for NFM and worked well without alteration.

4.1.10. OpenNebula monitoring

OpenNebula is an open-source toolkit designed to create private, public, and even hybrid IaaS, PaaS, and SaaS cloud deployment models. OpenNebula includes a monitoring framework that works as a subsystem to collect host and VM information, including host status, basic performance indicators, VM status, and power consumption [54]. There are two configurations for OpenNebula: push and pull. The favored mode of functioning is the push model. Here, the monitoring agent gathers resources metrics and transfers them to the front end of OpenNebula via user datagram protocol (UDP). The web front end of OpenNebula will then produce visualizations and warnings. In the absence of the push mode, OpenNebula defaults to the pull mode, and in this mode, the monitoring agents focus on issuing SSH connections for every monitored VM [83]. The monitoring framework of OpenNebula is not especially innovative but is one of several examples of a cloud-based open-source monitoring solution.

4.1.11. Lattice

Lattice is a monitoring framework that primarily aims to eliminate various constraints of some non-cloud-conscious monitoring tools [84]. These constraints are such difficulties in managing elasticity, regular changes, and dynamism. In comparison to other methods contained in this survey, lattice is not itself a monitoring solution. It is rather an environment for the development of monitoring solutions. Like other monitoring solutions, Lattice has the same abstraction rate for the CSPs, the users, and the samples. Lattice also includes a data source and distribution schema notion. Lattice does not fully implement the structures of

a full monitoring tool. However, it provides building blocks for developing a complete monitoring solution. This kind of implementation aims at enabling developers to build monitoring solutions that fit their specific needs. The benefit of this is that the segregation of issues between the different monitoring modules facilitates components to be designed and to be modified differently over time, without impacting the other elements [85]. Lattice provides an environment for the development of monitoring solutions and not being a monitoring tool itself. Also, it can deliver solutions that fit both the cloud-conscious and non-cloud-conscious.

4.1.12. Monitoring as-a-service OCCI API

A general cloud management system that can operate with all clouds is not usable. Many common clouds address several of the current paradigms. Therefore, it becomes necessary to have a general infrastructure for cloud computing, which can be implementable across all clouds. By proposing an extension to the open cloud computing interface application programming interface (OCCI API), Ciuffoletti [86] addresses this issue. It is an as-a-service on-demand control platform. The OCCI API is an IaaS open-source software API that supports some specifications and protocols. This research has focused on the creation of an OCCI monitoring platform and introduces a monitoring agent called the "Sensor". The sensor collects metric data, which users define in mixins. Mixins have three separate features that can be updated like ratings, aggregators, and publishers. In the end, the Docker-based prototype solution of the proposed work is introduced, and it is available as an open-source online. The focus of this work was on creating an extension for OCCI that could provide a solution for as-a-service monitoring. The results of this study were not a big concern. Another feature of this research is that it has concentrated mainly on control at the user level. The monitoring of the level of service providers was not included in this study, as well. Table 6 provides a summary of the studied resource monitoring schemes.

Table 6. Summary of the resource monitoring schemes

Scheme	Objectives	Limitation
[71]	Intra-site and inter-site tracking, both at the application and infrastructure level, of resources.	The VM descriptors increase as well as the network load.
[73]	To control the status of the services and to notify when an anomaly occurs.	Minimalism philosophy of the design concept.
[74]	Speed-up the monitoring of the availability and performance of all services within the infrastructure.	No resilience.
[75]	Scalable and adaptive business management platform based on multi-cloud alerting and virtual monitoring.	There is no question of interoperability with another CSP.
[76]	Simultaneous and transparent monitoring of public and private cloud systems.	The cost of the monitoring and monitoring process itself is not examined.
[77]	Offer a unified dashboard to monitor the SLAs of various CSPs.	No personalization is discussed over the monitoring events.
[80]	The cloud monitoring architecture for large-scale distributed systems is fully and highly customizable, interoperable, and efficient.	Multiple data controls can lead to system performance bottlenecks.
[81]	Collect monitoring data in an integrated private and public cloud environment.	It is only possible to monitor java-specific platforms.
[82]	Usage of a new tracking form where the monitoring agent will not interact with the host VM to provide monitoring and analytical operating services.	It is not without considering the interoperability with another CSP.
[84]	Eliminate constraints of various non-cloud-conscious monitoring tools.	The VM descriptor increase.
[86]	To introduce a monitoring agent called "sensor". The sensor collects metric data and compares them to the ideal state data.	The monitoring of the level of service providers is not considered.
[54]	Monitoring of resources across the heterogeneous domains from low to high-level services.	Unified display of data collected from different infrastructures.

4.2. Role of monitoring solutions in the taxonomy of resource management

Cloud resource monitoring provides detailed monitoring information on resource management and infrastructures, such as access control, service elasticity, service financial cost, and SLA management. Within the ICCE, monitoring plays the role of supervising everything that may occur to management taxonomy. Monitoring intervenes in the scaling of cloud resources as central processing unit (CPU)-intensive applications when non-proportional output costs are performed. This process helps allocate distributed application modules on many available nodes within the federation [22]. By monitoring various nodes, the management agents are aware of the required resources to meet the requirements of a specific component of a distributed application. In ICCEs, several federation formation techniques have been developed, and one of the most used is brokering. This technique requires a unique and autonomous unit, named broker, that works between cloud users and CSPs. The cloud broker meets customer's demands by selecting the best services with multiple CSPs [87]. The broker needs information related to all the resources in the pool, while the embedded monitoring agents provide the necessary information.

A detailed resource definition is required to encourage users to apply for the resources for efficient resource discovery within ICCE. Sim [88] suggested an algorithm based on gossip for resource discovery to propagate resource requests. It is a method used to determine correlations between user needs and resource profiles. The gossip-based resource discovery algorithm relies on data that monitoring agents gather throughout all the infrastructure. There are many resource discovery techniques. The majority of them need to collect information related to all the interconnected nodes. Monitoring is one way of collecting the required information.

Pricing in ICCE can be more than challenging. Considering the sharing of the resources of all the federation members constitutes the ICCE pool of resources. The question each CSP should ask is: how is one going to gain? both CSPs and users in ICCE are active actors when it comes to the usage of common resources and services. Moreover, they seek to optimize their benefits. Tracking the resources from one's pool is a regular activity that various tenants use. Monitoring activities and requests related to each member of the federation allows determining their gain according to the usage of the resources they put in the common pool [89] showed that cloud users could choose services, in terms of performance and prices, provided the price set by CSPs that provide them with the best return. Hence the emphasis on monitoring to help the federation tenants in handling the pricing-related issues.

Resource monitoring may also be a key element of the RM aspect, such as disaster management, to enable efficient management of disaster and loss of resources. Disaster recovery and fault tolerance play a significant role in restoring data [29], [30]. The resource recovery function needs to monitor the nodes affected by a disaster to perform the recovery actions.

4.3. Open research challenges

Monitoring has been a hot matter of research in ICCE recently. Numerous strategies have been suggested by researchers to deal with specific issues. But there is always a need to propose methods that can solve the RM challenges. On the basis of our study, we found few problems to be examined further.

a) CSPs lock-in

There is no standardization when logical or physical domain borders between CSPs are crossed. The CSPs lock-in is a critical obstacle to the sharing of ICCE resources. CSP lock-ins and heterogeneous infrastructure and architecture in ICCE make monitoring operations a challenge. Therefore, strategies to overcome this barrier must be developed.

b) Data acquisition

Work is needed to architecturally standardize APIs for collecting CSPs monitoring data. Gathering monitoring data is not supposed to increase the network traffic within the ICCE communication process. Network traffic optimization algorithms need to be developed by respecting the terms and conditions of development of the data acquisition mechanisms.

c) Energy consuming awareness

In order to facilitate efficient green cloud computing, energy monitoring should be carried out for ICCE CSPs through the planning and remodeling of the energy consumption index application tracker to reduce power consumption. Optimization is required to minimize the cost of energy consumption for instance the assignment of every virtual compontent to one and only one data center. Moreover, awaness alone can not reduice energy consumption within ICCEs, but subjective policies of energy usage should be shared among all the CSPs forming the environment.

d) Unified monitoring

To allow third-party monitoring and unified monitoring of ICCEs, cross-domain data leakage of services and internal settings of various CSPs must be addressed. Monitoring information should be easily spread across all the federation members. Moreover, an autonomous monitoring tool is needed to validate and measure heterogeneous applications used in ICCE.

e) Information management

There is no public data monitoring of single cloud or ICCE and there are no workload traces of the monitoring solutions in order to analyze the data using statistical methods to gain a greater understanding of the surveillance process. Information management strategies and mechanisms need to be developed for both single cloud infrastructure and ICCE. In addition, mutual information lifecycle management will proactively control data retention and disposal in accordance with business policy of ICCEs' members.

5. CONCLUSION

Monitoring plays a vital role in resource management taxonomy. In this review, we have surveyed different monitoring solutions. We have discussed resource monitoring in various aspects of resource management, including allocation, selection, discovery, pricing of the resources, and disaster management.

The future work focuses on different algorithms used in resource management taxonomy in ICCE with respect to QoS factors performing efficient monitoring of the infrastructure.

ACKNOWLEDGEMENTS

The authors acknowledge the sponsorship of the Covenant University Centre for Research, Innovation and Discovery (CUCRID), Ota, Ogun State, Nigeria.

REFERENCES

[1] A. Ahmad, A. S. Alzahrani, N. Ahmed, and T. Ahsan, "A delegation model for SDN-driven federated cloud," *Alexandria Engineering Journal*, 2020, doi: 10.1016/j.aej.2020.06.018.
[2] R. Ahuja and S. K. Mohanty, "A Scalable Attribute-Based Access Control Scheme with Flexible Delegation cum Sharing of Access Privileges for Cloud Storage," *IEEE Transactions on Cloud Computing*, vol. 8, no. 1, pp. 32–44, 2020, doi: 10.1109/TCC.2017.2751471.
[3] S. Kamboj and N. S. Ghumman, "A survey on cloud computing and its types," in *Proc. of the 10th INDIACom: 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, 2016, pp. 2971–2974.
[4] S. Logesswari, S. Jayanthi, D. KalaiSelvi, S. Muthusundari, and V. Aswin, "A study on cloud computing challenges and its mitigations," in *Proc. Materials Today*, 2020, doi: 10.1016/j.matpr.2020.10.655.
[5] B. Varghese and R. Buyya, "Next generation cloud computing: New trends and research directions," *Future Generation Computer Systems*, vol. 79, pp. 849–861, 2018, doi: 10.1016/j.future.2017.09.020.
[6] H. Kurdi *et al.*, "A lightweight trust management algorithm based on subjective logic for interconnected cloud computing environments," *Journal of Supercomputing*, vol. 75, no. 7, pp. 3534–3554, 2019, doi: 10.1007/s11227-018-2669-y.
[7] Y. Demchenko, C. Ngo, C. De Laat, and C. Lee, "Federated access control in heterogeneous intercloud environment: Basic models and architecture patterns," in *Proc. 2014 IEEE International Conference on Cloud Engineering, IC2E 2014*, 2014, pp. 439–445, doi: 10.1109/IC2E.2014.84.
[8] A. N. Toosi, R. N. Calheiros, and R. Buyya, "Interconnected cloud computing environments: Challenges, taxonomy, and survey," *ACM Computing Surveys*, vol. 47, no. 1, pp. 1-47, 2014, doi: 10.1145/2593512.
[9] M. Liaqat *et al.*, "Federated cloud resource management: Review and discussion," *Journal of Network and Computer Applications*, vol. 77, pp. 87–105, 2017, doi: 10.1016/j.jnca.2016.10.008.
[10] A. Asghari, M. K. Sohrabi, and F. Yaghmaee, "A cloud resource management framework for multiple online scientific workflows using cooperative reinforcement learning agents," *Computer Networks*, vol. 179, pp. 107340, 2020, doi: 10.1016/j.comnet.2020.107340.
[11] M. A. Haghighi, M. Maeen, and M. Haghparast, "An Energy-Efficient Dynamic Resource Management Approach Based on Clustering and Meta-Heuristic Algorithms in Cloud Computing IaaS Platforms: Energy Efficient Dynamic Cloud Resource Management," *Wireless Personal Communications*, vol. 104, no. 4, pp. 1367–1391, 2019, doi: 10.1007/s11277-018-6089-3.
[12] M. N. Birje and C. Bulla, "Cloud monitoring system: Basics, phases and challenges," *International Journal of Recent Technology Engineering(IJRTE)*, vol. 8, no. 3, pp. 4732–4746, 2019, doi: 10.35940/ijrte.C6857.098319.
[13] H. J. Syed, A. Gani, R. W. Ahmad, M. K. Khan, and A. I. A. Ahmed, "Cloud monitoring: A review, taxonomy, and open research issues," *Journal of Network and Computer Applications*, vol. 98, pp. 11–26, 2017, doi: 10.1016/j.jnca.2017.08.021.
[14] J. S. Ward and A. Barker, "Observing the clouds: a survey and taxonomy of cloud monitoring," *Journal Cloud Computing*, vol. 3, no. 1, pp. 1–30, 2014, doi: 10.1186/s13677-014-0024-2.
[15] A. Abusitta, M. Bellaiche, and M. Dagenais, "On trustworthy federated clouds: A coalitional game approach," *Computer Networks*, vol. 145, pp. 52–63, 2018, doi: 10.1016/j.comnet.2018.08.009.
[16] Y. Liu, S. Dong, J. Wei, and Y. Tong, "Assessing cloud computing value in firms through socio-technical determinants," *Information and Management.*, vol. 57, no. 8, p. 103369, 2020, doi: 10.1016/j.im.2020.103369.
[17] F. A. M. Ibrahim and E. E. Hemayed, "Trusted Cloud Computing Architectures for infrastructure as a service: Survey and systematic literature review," *Computers and Security*, vol. 82, pp. 196–226, 2019, doi: 10.1016/j.cose.2018.12.014.
[18] N. Khorasani, S. Abrishami, M. Feizi, M. A. Esfahani, and F. Ramezani, "Resource management in the federated cloud environment using Cournot and Bertrand competitions," *Future Genereration Computer Systems*, vol. 113, pp. 391–406, 2020, doi: 10.1016/j.future.2020.07.010.
[19] R. Duncan, "A multi-cloud world requires a multi-cloud security approach," *Computer Fraud and Security*, no. 5, pp. 11–12, 2020, doi: 10.1016/S1361-3723(20)30052-X.
[20] M. M. Alshammari, A. A. Alwan, A. Nordin, and A. Z. Abualkishik, "Disaster Recovery with Minimum Replica Plan for Reliability Checking in Multi-Cloud," in *Procedia Computer Science*, 2018, vol. 130, pp. 247–254, doi: 10.1016/j.procs.2018.04.036.
[21] S. Sahmim and H. Gharsellaoui, "Privacy and Security in Internet-based Computing: Cloud Computing, Internet of Things, Cloud of Things: A review," in *Procedia Computer Science*, 2017, vol. 112, pp. 1516–1522, doi: 10.1016/j.procs.2017.08.050.
[22] B. S. Rajeshwari and M. Dakshayini, "Optimized Bit Matrix based Power Aware Load Distribution Policy for Federated Cloud," in *Procedia Computer Science*, 2020, vol. 167, pp. 1771–1790, doi: 10.1016/j.procs.2020.03.387.
[23] A. Levin, D. Lorenz, G. Merlino, A. Panarello, A. Puliafito, and G. Tricomi, "Hierarchical load balancing as a service for federated cloud networks," *Computer Communications.*, vol. 129, pp. 125–137, 2018, doi: 10.1016/j.comcom.2018.07.031.
[24] M. Chiregi and N. Jafari Navimipour, "Cloud computing and trust evaluation: A systematic literature review of the state-of-the-art mechanisms," *Journal of Electrical Systems and Information Technology*, vol. 5, no. 3, pp. 608–622, 2018, doi: 10.1016/j.jesit.2017.09.001.
[25] D. C. Marinescu, "Cloud Resource Management and Scheduling," in *Cloud Computing*, 2018, pp. 321–363, [Online]. Available: http://mallikarjunbangargi.yolasite.com/resources/Chapter6.pdf
[26] E. Arianyan, H. Taheri, and S. Sharifian, "Novel energy and SLA efficient resource management heuristics for consolidation of virtual machines in cloud data centers," *Computers and Electrical Engineering*, vol. 47, pp. 222–240, 2015, doi: 10.1016/j.compeleceng.2015.05.006.
[27] B. Jennings and R. Stadler, "Resource Management in Clouds: Survey and Research Challenges," *Journal of Network and Systems Management*, vol. 23, no. 3, pp. 567–619, 2015, doi: 10.1007/s10922-014-9307-7.
[28] S. Aslam, S. ul Islam, A. Khan, M. Ahmed, A. Akhundzada, and M. K. Khan, "Information collection centric techniques for cloud

resource management: Taxonomy, analysis and challenges," *Journal of Network and Computer Applications*, vol. 100, pp. 80–94, 2017, doi: 10.1016/j.jnca.2017.10.021.
[29] C. B. Hauser and S. Wesner, "Reviewing Cloud Monitoring: Towards Cloud Resource Profiling," in *2018 IEEE 11th International Conference on Cloud Computing (CLOUD)*, 2018, pp. 678–685, doi: 10.1109/CLOUD.2018.00093.
[30] S. Mustafa, B. Nazir, A. Hayat, A. U. R. Khan, and S. A. Madani, "Resource management in cloud computing: Taxonomy, prospects, and challenges," *Computers and Electrical Engineering*, vol. 47, pp. 186–203, 2015, doi: 10.1016/j.compeleceng.2015.07.021.
[31] Y.-H. Lee, K.-C. Huang, M.-R. Shieh, and K.-C. Lai, "Distributed resource allocation in federated clouds," *Journal of Supercomputing*, vol. 73, no. 7, pp. 3196–3211, 2017, doi: 10.1007/s11227-016-1918-1.
[32] D. Ardagna, S. Casolari, M. Colajanni, and B. Panicucci, "Dual time-scale distributed capacity allocation and load redirect algorithms for cloud systems," *Journal of Parallel and Distributed Computing*, vol. 72, no. 6, pp. 796–808, 2012, doi: 10.1016/j.jpdc.2012.02.014.
[33] C. Papagianni, A. Leivadeas, S. Papavassiliou, V. Maglaris, C. Cervelló-Pastor, and Á. Monje, "On the optimal allocation of virtual resources in cloud computing networks," *IEEE Transactions on Computers*, vol. 62, no. 6, pp. 1060–1071, 2013, doi: 10.1109/TC.2013.31.
[34] F. Palmieri, L. Buonanno, S. Venticinque, R. Aversa, and B. Di Martino, "A distributed scheduling framework based on selfish autonomous agents for federated cloud environments," *Future Generation Computer Systems*, vol. 29, no. 6, pp. 1461–1472, 2013, doi: 10.1016/j.future.2013.01.012.
[35] M. M. Hassan, M. S. Hossain, A. M. J. Sarkar, and E. N. Huh, "Cooperative game-based distributed resource allocation in horizontal dynamic cloud federation platform," *Information Systems Frontiers*, vol. 16, no. 4, pp. 523–542, 2014, doi: 10.1007/s10796-012-9357-x.
[36] S. S. Woo and J. Mirkovic, "Optimal application allocation on multiple public clouds," *Computer Networks*, vol. 68, pp. 138–148, 2014, doi: 10.1016/j.comnet.2013.12.001.
[37] M. R. HoseinyFarahabady, Y. C. Lee, and A. Y. Zomaya, "Randomized approximation scheme for resource allocation in hybrid-cloud environment," *Journal of Supercomputing*, vol. 69, no. 2, pp. 576–592, 2014, doi: 10.1007/s11227-014-1094-0.
[38] X. Chen, Y. Zhou, L. Yang, and L. Lv, "Hybrid fog/cloud computing resource allocation: Joint consideration of limited communication resources and user credibility," *Computer Communications*, vol. 169, pp. 48–58, 2021, doi: 10.1016/j.comcom.2021.01.026.
[39] Y. Yang, R. Liu, Y. Chen, T. Li, and Y. Tang, "Normal cloud Model-Based algorithm for Multi-Attribute trusted cloud service selection," *IEEE Access*, vol. 6, pp. 37644–37652, 2018, doi: 10.1109/ACCESS.2018.2850050.
[40] S. Sundareswaran, A. Squicciarini, and D. Lin, "A brokerage-based approach for cloud service selection," in *Proc. 2012 IEEE 5th International Conference on Cloud Computing (CLOUD)*, 2012, pp. 558–565, doi: 10.1109/CLOUD.2012.119.
[41] J. O. Gutierrez-Garcia and K. M. Sim, "Agent-based cloud service composition," *Applied Intelligence*, vol. 38, no. 3, pp. 436–464, 2013, doi: 10.1007/s10489-012-0380-x.
[42] J. Son, "Automated decision system for efficient resource selection and allocation in inter-clouds," M.S. thesis, Department of Computing and Information Systems, University of Melbourne, Melbourne, AU, 2013. Accessed: Mar. 21, 2021. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.433.122&rep=rep1&type=pdf
[43] S. Farokhi, F. Jrad, I. Brandic, and A. Streit, "HS4MC - Hierarchical SLA-based Service Selection for Multi-Cloud Environments," in *Proc. 4th International Conference on Cloud Computing and Services Science*, 2014, pp. 722–734, doi: 10.5220/0004979707220734.
[44] W.-J. Fan, S.-L. Yang, H. Perros, and J. Pei, "A Multi-dimensional Trust-aware Cloud Service Selection Mechanism Based on Evidential Reasoning Approach," *International Journal of Automation and Computing.*, vol. 12, no. 2, pp. 208–219, 2015, doi: 10.1007/s11633-014-0840-3.
[45] A. Jaikar and S. Y. Noh, "Cost and performance effective data center selection system for scientific federated cloud," *Peer-to-Peer Networking and Applications*, vol. 8, no. 5, pp. 896–902, 2015, doi: 10.1007/s12083-014-0261-7.
[46] D. Fernández-Cerero, F. J. Ortega, A. Jakóbik, and A. Fernández-Montes, "DISCERNER: Dynamic selection of resource manager in hyper-scale cloud-computing data centres," *Future Generation Computer Systems*, vol. 116, pp. 190–199, 2021, doi: 10.1016/j.future.2020.10.031.
[47] P. Khethavath, J. Thomas, E. Chan-Tin, and H. Liu, "Introducing a distributed cloud architecture with efficient resource discovery and optimal resource allocation," in *Proc. 2013 IEEE 9th World Congress on Services*, 2013, pp. 386–392, doi: 10.1109/SERVICES.2013.68.
[48] S. Sotiriadis, N. Bessis, and P. Kuonen, "Advancing inter-cloud resource discovery based on past service experiences of transient resource clustering," in *Proc. 3rd International Conference on Emerging Intelligent Data and Web Technologies*, 2012, pp. 38–45, doi: 10.1109/EIDWT.2012.16.
[49] P. Wright, Y. L. Sun, T. Harmer, A. Keenan, A. Stewart, and R. Perrott, "A constraints-based resource discovery model for multi-provider cloud environments," *Journal of Cloud Computing*, vol. 1, no. 1, pp. 1–14, 2012, doi: 10.1186/2192-113X-1-6.
[50] W. C. Chung, C. J. Hsu, K. C. Lai, K. C. Li, and Y. C. Chung, "Direction-aware resource discovery in large-scale distributed computing environments," *Journal of Supercomputing*, vol. 66, no. 1, pp. 229–248, 2013, doi: 10.1007/s11227-013-0899-6.
[51] C. Pittaras, C. Papagianni, A. Leivadeas, P. Grosso, J. Van Der Ham, and S. Papavassiliou, "Resource discovery and allocation for federated virtualized infrastructures," *Future Generation Computer Systems*, vol. 42, pp. 55–63, 2015, doi: 10.1016/j.future.2014.01.003.
[52] C. Pham, N. H. Tran, M. N. H. Nguyen, S. Ren, W. Saad, and C. S. Hong, "Hosting virtual machines on a cloud datacenter: A matching theoretic approach," in *Proc. NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium*, 2016, pp. 659–664, doi: 10.1109/NOMS.2016.7502873.
[53] A. E. Ezugwu and A. O. Adewumi, "Soft sets based symbiotic organisms search algorithm for resource discovery in cloud computing environment," *Future Generation Computer Systems*, vol. 76, pp. 33–50, 2017, doi: 10.1016/j.future.2017.05.024.
[54] Y. Kessaci, N. Melab, and E. G. Talbi, "A multi-start local search heuristic for an energy efficient VMs assignment on top of the OpenNebula cloud manager," *Future Generation Computer Systems*, vol. 36, pp. 237–256, 2014, doi: 10.1016/j.future.2013.07.007.
[55] M. Aazam, E. N. Huh, M. St-Hilaire, C. H. Lung, and I. Lambadaris, "Cloud Customer's Historical Record Based Resource Pricing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 7, pp. 1929–1940, 2016, doi: 10.1109/TPDS.2015.2473850.
[56] L. Jin and L. Tang, "Prediction-based pricing for request outsourcing in cloud federation," in *2016 8th International Conference on Wireless Communications and Signal Processing*, 2016, doi: 10.1109/WCSP.2016.7752456.
[57] M. Mihailescu and Y. M. Teo, "Strategy-Proof Dynamic Resource Pricing of Multiple Resource Types on Federated Clouds," in *10th International Conference on Algorithms and Architectures for Parallel Processing*, 2010, pp. 337-350.

[Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.297.6641&rep=rep1&type=pdf
[58] E. R. Gomes, B. Q. Vo, and R. Kowalczyk, "Pure exchange markets for resource sharing in federated clouds," *Concurrency and Computation Practice and Experience*, vol. 24, no. 9, pp. 977–991, 2012, doi: 10.1002/cpe.1659.
[59] H. Roh, C. Jung, W. Lee, and D. Z. Du, "Resource pricing game in geo-distributed clouds," in *Proc. IEEE INFOCOM* , 2013, pp. 1519–1527, doi: 10.1109/INFCOM.2013.6566947.
[60] S. Qanbari, F. Li, S. Dustdar, and T. S. Dai, "Cloud Asset Pricing Tree (CAPT) - Elastic Economic Model for Cloud Service Providers," in *Proc. 4th International Conference on Cloud Computing and Services Science (CLOSER-2014)*, 2014, pp. 221–229, doi: 10.5220/0004849702210229.
[61] B. El Zant and M. Gagnaire, "New pricing policies for federated cloud," in *2014 6th International Conference on New Technologies, Mobility and Security (NTMS)*, 2014, doi: 10.1109/NTMS.2014.6814036.
[62] C. T. Do, N. H. Tran, E. N. Huh, C. S. Hong, D. Niyato, and Z. Han, "Dynamics of service selection and provider pricing game in heterogeneous cloud market," *Journal of Network and Computer Applications*, vol. 69, pp. 152–165, 2015, doi: 10.1016/j.jnca.2016.04.012.
[63] J. Zhang, N. Xie, X. Zhang, and W. Li, "An online auction mechanism for cloud computing resource allocation and pricing based on user evaluation and cost," *Future Generation Computer Systems*, vol. 89, pp. 286–299, 2018, doi: 10.1016/j.future.2018.06.034.
[64] S. Sengupta and K. M. Annervaz, "Multi-site data distribution for disaster recovery - A planning framework," *Future Generation Computer Systems*, vol. 41, pp. 53–64, 2014, doi: 10.1016/j.future.2014.07.007.
[65] A. Lenk and S. Tai, "Cloud standby: Disaster recovery of distributed systems in the cloud," in *Conference: European Conference on Service-Oriented and Cloud Computing*, 2014, vol. 8745, pp. 32–46, doi: 10.1007/978-3-662-44879-3_3.
[66] K. Liao, H. Shen, and L. Guo, "Improved approximation algorithms for constrained fault-tolerant resource allocation," *Theoretical Computer Science*, vol. 590, pp. 118–128, 2015, doi: 10.1016/j.tcs.2015.02.029.
[67] V. Chang, "Towards a Big Data system disaster recovery in a Private Cloud," *Ad Hoc Networks*, vol. 35, pp. 65–82, 2015, doi: 10.1016/j.adhoc.2015.07.012.
[68] V. Chang and M. Ramachandran, "Towards Achieving Data Security with the Cloud Computing Adoption Framework," *IEEE Transactions on Services Computing*, vol. 9, no. 1, pp. 138–151, 2016, doi: 10.1109/TSC.2015.2491281.
[69] S. Taherizadeh, A. C. Jones, I. Taylor, Z. Zhao, and V. Stankovski, "Monitoring self-adaptive applications within edge computing frameworks: A state-of-the-art review," *Journal of Systems and Software*, vol. 136, pp. 19–38, 2018, doi: 10.1016/j.jss.2017.10.033.
[70] J. M. Alcaraz Calero and J. G. Aguado, "MonPaaS: An adaptive monitoring platformas a service for cloud computing infrastructures and services," *IEEE Transactions on Services Computing*, vol. 8, no. 1, pp. 65–78, 2015, doi: 10.1109/TSC.2014.2302810.
[71] W. Barth, *Nagios : System and Network Monitoring*. San Fransisco, USA : No Starch Press, 2008.
[72] J. G.-Aguado, J. M. A. Calero, and W. D.-Villanueva, "IaaSMon: Monitoring Architecture for Public Cloud Computing Data Centers," *Journal of Grid Computing*, vol. 14, no. 2, pp. 283–297, 2016, doi: 10.1007/s10723-015-9357-4.
[73] S. Mongkolluksamee, P. Pongpaibool, and C. Issariyapat, "Strengths and limitations of Nagios as a network monitoring solution," in *Proc. 7th International Joint Conference on Computer Science and Software Engineering (JCSSE 2010)*, 2010, pp. 96–101, Accessed: Aug. 28, 2020. [Online]. Available: https://www.researchgate.net/profile/Sophon-Mongkolluksamee-2/publication/267366269_Strengths_and_Limitations_of_Nagios_as_a_Network_Monitoring_Solution/links/54eaabc10cf27a6de114ce97/Strengths-and-Limitations-of-Nagios-as-a-Network-Monitoring-Solution.pdf
[74] P. Tader, "Server monitoring with Zabbix," *Linux Journal*, no. 195, pp. 7, 2010. [Online]. Available: https://www.linuxjournal.com/article/10235.
[75] S. A. De Chaves, R. B. Uriarte, and C. B. Westphall, "Toward an architecture for monitoring private clouds," *IEEE Communications Magazine*, vol. 49, no. 12, pp. 130–137, 2011, doi: 10.1109/MCOM.2011.6094017.
[76] D. Tovarnňák and T. Pitner, "Towards multi-tenant and interoperable monitoring of virtual machines in cloud," in *Proc. 14th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, SYNASC 2012*, 2012, pp. 436–442, doi: 10.1109/SYNASC.2012.55.
[77] J. Montes, A. Sánchez, B. Memishi, M. S. Pérez, and G. Antoniu, "GMonE: A complete approach to cloud monitoring," *Future Generation Computer Systems*, vol. 29, no. 8, pp. 2026–2040, 2013, doi: 10.1016/j.future.2013.02.011.
[78] D. Milojičić, I. M. Llorente, and R. S. Montero, "OpenNebula: A Cloud Management Tool," *IEEE Internet Computing*, vol. 15, no. 2, pp. 11–14, 2011, doi: 10.1109/MIC.2011.44.
[79] S. Hasan, S. O'Riain, and E. Curry, "Approximate semantic matching of heterogeneous events," in *Proc. 6th ACM International Conference on Distributed Event-Based Systems, DEBS'12*, 2012, pp. 252–263, doi: 10.1145/2335484.2335512.
[80] J. Povedano-Molina, J. M. Lopez-Vega, J. M. Lopez-Soler, A. Corradi, and L. Foschini, "DARGOS: A highly adaptable and scalable monitoring architecture for multi-tenant Clouds," *Future Generation Computer Systems*, vol. 29, no. 8, pp. 2041–2056, 2013, doi: 10.1016/j.future.2013.04.022.
[81] S. Al-Shammari and A. Al-Yasiri, "MonSLAR: a middleware for monitoring SLA for RESTFUL services in cloud computing," in *2015 IEEE 9th International Symposium on the Maintenance and Evolution of Service-Oriented and Cloud-Based Environments (MESOCA)*, 2015, pp. 46–50, doi: 10.1109/MESOCA.2015.7328126.
[82] S. Suneja, C. Isci, R. Koller, and E. De Lara, "Touchless and always-on cloud analytics as a service," *Ibm Journal of Research and Development*, vol. 60, no. 2–3, 2016, doi: 10.1147/JRD.2016.2518438.
[83] D. Zou *et al.*, "Design and implementation of a trusted monitoring framework for cloud platforms," *Future Generation Computer Systems*, vol. 29, no. 8, pp. 2092–2102, 2013, doi: 10.1016/j.future.2012.12.020.
[84] S. Clayman, A. Galis, and L. Mamatas, "Monitoring virtual networks with lattice," in *2010 IEEE/IFIP Network Operations and Management Symposium Workshops, NOMS 2010*, 2010, pp. 239–246, doi: 10.1109/NOMSW.2010.5486569.
[85] M. R. M. Assis and L. F. Bittencourt, "A survey on cloud federation architectures: Identifying functional and non-functional properties," *Journal of Network and Computer Applications*, vol. 72, pp. 51–71, 2016, doi: 10.1016/j.jnca.2016.06.014.
[86] A. Ciuffoletti, "Application level interface for a cloud monitoring service," *Computer Standards and Interfaces*, vol. 46, pp. 15–22, 2016, doi: 10.1016/j.csi.2016.01.001.
[87] S. S. Chauhan, E. S. Pilli, R. C. Joshi, G. Singh, and M. C. Govil, "Brokering in interconnected cloud computing environments: A survey," *Journal of Parallel and Distributed Computing*, vol. 133, pp. 193–209, 2019, doi: 10.1016/j.jpdc.2018.08.001.
[88] K. M. Sim, "Agent-Based Fog Computing: Gossiping, Reasoning, and Bargaining," *IEEE Letters of the Computer Society*, vol. 1, no. 2, pp. 21–24, 2018, doi: 10.1109/locs.2018.2886828.
[89] D. Bernstein and D. Vij, "Intercloud directory and exchange protocol detail using XMPP and RDF," in *Proc. 2010 6th World Congress on Services, Services-1 2010*, 2010, pp. 431–438, doi: 10.1109/SERVICES.2010.131.

BIOGRAPHIES OF AUTHORS

Vingi Patrick Nzanzu received his BSc in Electrical and Computer Engineering in 2017 at Université Libre des Pays des Grands in D.R. Congo. He is currently a master student at Covenant University Ota Ogun State Nigeria in the department of Electrical and Information Engineering, Information and Communication Engineering program. He works with the Covenant Applied Informatics and Communication-Centre of Excellence (CApIC-ACE) as a research assistant. He is presently studying Federated Cloud Computing Management. IoT, Software and System development, Real time Systems are his other areas of interest. He can be contacted at email: parickvingi@gmail.com.

Emmanuel Adetiba (Member, IEEE) received the Ph. D degree in information and communication engineering from Covenant University, Ota, Nigeria. He is a registered engineer (REngr.) with the Council for the Regulation of Engineering in Nigeria (COREN) and a member of the Institute of Information Technology Professional (IITP), South Africa as well as the Institute of Electrical & Electronics Engineering (IEEE), USA. He has received several scholarly grants and research funds from reputable bodies such as World Bank, France Development Agency (AFD), US National Science Foundation, Durban University of Technology in South Africa, Nigeria Communications Commission, Rockefeller Foundation, International Medical Informatics Association (IMIA) and hosts of others. He has authored/co-authored about 100 scholarly publications in journals and conference proceedings some of which are indexed in Scopus/ISI/CPCI. His research interests include Machine Intelligence, Software Defined Radio, Cognitive Radio, Biomedical Signal Processing, Cloud Federation and EduTech. He was the Director at the Center for Systems and Information Services (ICT Center), Covenant University from 2017 to 2019. He is a Deputy Director at the Covenant Applied Informatics and Communication Africa Centre of Excellence (CApIC-ACE) and Co-PI for the FEDGEN cloud infrastructure research project at the center (World Bank funded). He is the founder and Principal Investigator of the Advanced Signal Processing and Machine Intelligence Research (ASPMIR) group. He is a full Professor and the incumbent Head of Department, Electrical & Information Engineering, Covenant University. He is also an honorary research associate at the Institute for System Sciences, Durban University of Technology, Durban, South Africa. He can be contacted at email: emmanuel.adetiba@covenantuniversity.edu.ng.

Joke Atinuke Badejo received the B.Tech. degree from Ladoke Akintola University of Technology in 2003, the M.Eng. and Ph.D. degree from Covenant University, Nigeria in 2007 and 2015 respectively; all in Computer Engineering. During her Ph.D., she received the International Association for Pattern Recognition and MORPHO grant in 2012 for biometrics training and was also a visiting student at Biomedical Signal Analysis Laboratory in Clarkson University, in 2015. She currently leads the Data Analytics team which support data-driven decision making in Covenant University. She is a member of a number of professional bodies such as the Council for the Regulation of Engineering in Nigeria (COREN) and Institute of Electrical and Electronics Engineers (IEEE). Joke enjoys being an academic and loves contributing impactful and cost-effective solutions to prevailing societal engineering problems in Africa. She can be contacted at email: joke.badejo@covenantuniversity.edu.ng.

Mbasa Joaquim Molo received his BSc in Electrical and Computer Engineering in 2019 at Université Libre des Pays des Grands in D.R. Congo. He is currently a master student at Covenant University in Nigeria in Information and communication engineering. He works as a research assistant in the Covenant Applied Informatics and Communication-Centre of Excellence. His research interest is focused on machine learning and cloud computing. Also, he collaborates with the R&D department of the Infokom GmbH where the focus is on integrating IT technologies in the health sector. He can be contacted at email: mbasakhim@gmail.com.

Claude Takenga received the BSc. and MSc. Degrees in Radio engineering and Telecommunication from the Saint Petersburg State Technical University in 2001 and 2003 respectively. He received his Ph. D. Degree in electrical engineering from the Leibniz University of Hannover with the Institute for Communication Technology in 2007 in Germany. He is currently working at the Industry with the R&D department of the Infokom GmbH in Germany, where he collaborates in several international joint-projects focusing on integrating IT-technologies in the health sectors. His research interests focus on eHealth systems, mobile applications, communication technologies. He serves as university professor in some countries and has published numerous papers in conference proceedings and journals. He can be contacted at email: takenga@yahoo.fr.

Etinosa Noma-Osaghae is with the Department of Electrical and Information Engineering, Covenant University. He has obtained industry certifications in Application Security, Security Intelligence, Blockchain and Artificial Intelligence. The certifications were awarded by the International Business Machines (IBM). He is a registered Engineer with the Council for the Regulation of Engineering in Nigeria (COREN) and a corporate member of the Nigeria Society of Engineers (NSE). He is presently studying the spectral and energy efficiency of non-orthogonal multiple access techniques in heterogeneous cellular networks. Biometrics, cybersecurity, and alternative energy are his other areas of interest. He can be contacted at email: etinosa.noma-osaghae@covenantuniversity.edu.ng.

Victoria Oguntosin received the BEng in Electrical Engineering from the University of Ilorin, Nigeria, an MSc in Electrical and Electronic Engineering from the University of Greenwich, UK, and a Ph.D. from the University of Reading, UK. Her Ph.D. work at the University of Reading was on the development of a soft modular robotic arm. The research work involved building a soft robotic assistive arm that is actuated by pneumatics. She can be contacted at email: victoria.oguntosin@covenantuniversity.edu.ng.

Sadeeq Suraju received the BSC and, an MSc in Statistics from the University of Ilorin, Nigeria, He is presently undergoing a Ph. D at Covenant University Ota Ogun State Nigeria in the department of Computer Science, Bioinformatic programm. His research focuses on OMICS and earlier detection of Cancer. He works with the Covenant Applied Informatics and Communication-Centre of Excellence (CApIC-ACE) as a research assistant. He can be contacted at email: suraj.adewale1@gmail.com.