SEMARAK ILMU
PUBLISHING
202103268166(003316878-P)

Journal of Advanced Research in Applied Sciences and Engineering Technology

Journal homepage:
https://semarakilmu.com.my/journals/index.php/applied_sciences_eng_tech/index
ISSN: 2462-1943

Comparison of Cloud Computing, SaaS, SOA and Microservices

Noor Athirah Muhammed Asri[1], Mohd Hamdi Irwan Hamzah[1,*], Rosziati Ibrahim[1], Mohd Zanes Sahid[1], Nur Ariffin Mohd Zin[1], Huang Shiling[2]

[1] Fakulti Sains Komputer dan Teknologi Maklumat, Universiti Tun Hussein Onn Malaysia, 86400 Batu Pahat, Johor, Malaysia
[2] College of Intelligent Manufacturing, Nanning University, Yongning District, Nanning, Guangxi, 530200 China

ARTICLE INFO

Article history:
Received
Received in revised form
Accepted
Available online

Keywords:

Network application; Cloud computing; Software as a service (SaaS); Service oriented architecture (SOA); Microservices

ABSTRACT

A network application is an application that runs on a single host and facilitates communication with another application running on a different computer or host. Numerous network applications have been introduced, including cloud computing, SaaS, SOA, and microservices. However, these technologies have often been used interchangeably, leading to confusion among developers and researchers. Therefore, the objective of this study is to review and compare the available network applications, specifically cloud computing, Software as a Service (SaaS), Service Oriented Architecture (SOA), and microservices. This study conducts a comparative analysis by examining several important criteria such as definition, type, language, interoperability, characteristics, and examples. By providing clear and precise definitions for each technology, the study aims to clarify their distinctions and appropriate usage contexts. The findings successfully establish the correct terminology for each technology and offer detailed discussions on the specific platforms and situations where each is most suitable. This comprehensive review serves as a valuable resource for developers and researchers seeking to navigate the complexities of modern network applications.

1. Introduction

The field of computer network has expanded at an unprecedented rate and the widespread use of networking has resulted in significant economic shifts. A computer network is defined as a collection of autonomous, independent computer systems that are connected in such a way that they can share resources interactively [1]. The Internet also has successfully connected us throughout the world and a variety of areas have thrived by this technology such as e-commerce, advertising, education, healthcare and many more. The existence of the internet which is defined as the largest computer network in the world, connecting millions of computers has introduced us to network applications allows doing everyday task such as sending emails, real-time video conferences, searching information on a web browser and many more.

* *Corresponding author.*
E-mail address: hamdi@uthm.edu.my

https://doi.org/10.37934/araset.63.2.89101

Apart from that, the technology has moved very fast and this computer networks have supported various applications and services in different areas such as business, home, mobile and social to ease our work [2]. Many systems and services have been distributed across the network which leads to difficulty in configuration management. Moreover, there have been so many network applications being introduced such as cloud computing, SaaS, SOA and microservices and these network applications have been popular and widely used. However, this study found that the terms used for each of these network applications have been used interchangeably, which has caused confusion among developers [3-6]. Therefore, this study is going to introduce and discuss several of the most popular network applications which are cloud computing, SaaS, SOA and microservices. By providing clear definitions and distinguishing the unique characteristics of each, the study aims to clarify these concepts and help developers understand the specific contexts in which they are best applied.

This comprehensive study conducted a thorough comparative analysis, meticulously examining various critical criteria. These included a detailed exploration of definitions, types, languages, interoperability of network applications, distinctive characteristics, and examples. Through this in-depth examination of these key elements, the research aimed to provide a nuanced and insightful perspective on the subject matter.

The structure of this paper is organized as follows: section 2 outlines the related works, section 3 provides an overview of network application followed by the introduction to cloud computing, SaaS, SOA and microservices. Section 4 presents the findings and discussions while section 5 concludes the study with a summary.

2. Related Work

Before the advent of network applications, businesses primarily relied on traditional software architecture to operate. This architecture comprised hardware and software components such as desktop computers, data centres, servers, networking hardware, and enterprise application software solutions [7]. However, the landscape has rapidly shifted from monolithic to microservices architecture. As legacy monolithic software systems grew in complexity, many organizations began embracing the microservices architectural style, which breaks down software into smaller, more manageable components [8].

In recent years, microservices architecture has gained significant popularity due to its numerous advantages, which include technological diversity, resilience, scalability, ease of deployment, enhanced productivity, component reusability, and replaceability [9]. Organizations are increasingly adopting microservices architecture for their business systems as a response to various limitations associated with monolithic approaches. These include high operational costs due to the requirement for extensive hardware, software, and physical space. Additionally, traditional architectures often store files on individual devices or local servers, limiting ubiquitous access to data [10]. Furthermore, traditional architectures lack elasticity, making it challenging to rapidly scale system resources up or down as needed. This often results in constant resource overprovisioning or inefficient underutilization throughout the operational lifecycle of systems.

Therefore, because of these disadvantages, enterprises are pursuing lower costs, greater availability, greater agility, and better risk management, all of which are accelerated by the adoption of cloud in their business architecture. Adoption of cloud has emerged as one of the major paradigm shifts, promising a cost-effective and robust computational architecture for large-scale and even overly complex enterprise application. With the existence of cloud, it has introduced us to a network application that able to runs on a single host facilitating communication with another application running on a different computer or host [6].

From previous studies, Familiar Bob noted that cloud computing involves delivering computing resources over the internet, offering flexibility and cost-effectiveness, while, SaaS represents a specific type of cloud computing service model that provides internet-based software applications [11]. Additionally, Service-Oriented Architecture (SOA) plays a crucial role in cloud computing by facilitating communication and interaction between services within the cloud network. It enables seamless data and information exchange between clients and servers, supporting the delivery of diverse services to cloud users. SOA organizes software and infrastructure into interconnected services, promoting interoperability across different technologies through web services (SOAP or REST). It emphasizes integration, application integration, and service discovery [12]. In contrast, microservices architecture features small, autonomous components known as microservices that communicate via lightweight mechanisms such as HTTP-based APIs. This architectural style focuses on modularity, scalability, continuous delivery, and the independent deployment, testing, and scaling of individual components [12].

Previously, several studies have delved into specific aspects such as comparing Cloud Computing with SaaS [11], exploring the integration of SaaS within Cloud Computing [13], and examining the relationships between SOA and both Cloud Computing and Microservices [12,14]. However, there remains a gap in the literature as none of these studies have comprehensively tackled all four of these prominent network applications together. This study aims to fill this gap by providing a comprehensive analysis that examines and compares Cloud Computing, SaaS, SOA, and Microservices in a unified framework.

To provide a better understanding, this study plans to compare them using important criteria such as definition, type, language, interoperability, characteristics, and examples. This study has successfully provided the actual term for each technology and clearly discussed where this individual technology is suitable to be applied on which platform and situation.

3. Network Applications

This study will compare several network applications such as cloud computing, SaaS, SOA and microservices. These network applications have been popular and widely used and have raised and issues where the term is mixed up by the developers.

3.1 Cloud Computing

As an answer to this problem that has been encountered with traditional software, cloud computing has been proposed as a possible solution. In late 2007, cloud computing emerged as one of the newest IT industry buzzwords, referring to the process of moving data and applications to a distant 'cloud' that can be accessed simply and ubiquitously [15]. Cloud computing is a business model that offer on-demand network access to a shared pool of programmable computing resources that can be instantly deployed and released with no administration effort or contact from service [16,17]. Furthermore, computing services such as servers, databases, networking, software and data analytics are delivered via the internet using cloud computing to allow for speedier deployment, more flexible resource allocation, and greater economies of scale [18]. In Industry 4.0 and IoT, developments in Cloud Computing enable the utilization of vast datasets and networks of distant processors to filter, model, optimize, and facilitate data sharing across systems, ultimately enhancing performance over time [19].

The cloud computing model was developed by internet service providers to accommodate the greatest number of users and deliver elastic services while using the least number of resources. Developing cloud computing has quickly risen to become the most talked-about technology within a few years. Cloud computing defined as delivering IT services, programs, and data through the use of dynamically scalable pool(s), which may be located remotely, so that customers do not have to worry about the actual location of the server or storage that meets their demands. Users have the flexibility to select from public, private, or hybrid storage solutions based on their business requirements or specific needs, and their desired level of control such as Software as a Service (SaaS), Platform as a Service (PaaS), or Infrastructure as a Service (IaaS) [20]. Cloud facilitates data and service exchanges between various cloud infrastructure offerings and the use of the data and services exchanged to make them work together effectively [21].

As shown in Table 1, there are four types of cloud computing models which are public cloud, private cloud, hybrid cloud and community cloud. Public cloud infrastructure is located on the premises of the cloud service provider. On a pay-per-use basis, either the general public or a major industrial group can utilize cloud services for their own personal or business use. Private cloud is administered entirely for the benefit of a single enterprise. Either the organization or a third party might be in charge of managing it that located on or off premises. A hybrid cloud is a cloud formation made up of two or more clouds (private, community, or public) that remain distinct companies, connected via the use of standardized or proprietary technologies. Community cloud is a cloud that is shared by several companies that have common issues such as security requirements, policy, and compliance issues. Community clouds are often maintained by groups in the community or by a third party, and it can be located on or off-premises depending on the circumstances [22].

Table 1
Types of Cloud Computing [23]

Type of Cloud Computing	Description
Public Cloud	Users: general public Hosting Cloud services providers
Private Cloud	Users: a single organization Hosting: internally and externally
Community Cloud	Users: organizations that have similar concerns Hosting: externally or internally by one of the organizations
Hybrid Cloud	Users: from two or more clouds bound together but still hold distinct entities Hosting: internally and externally

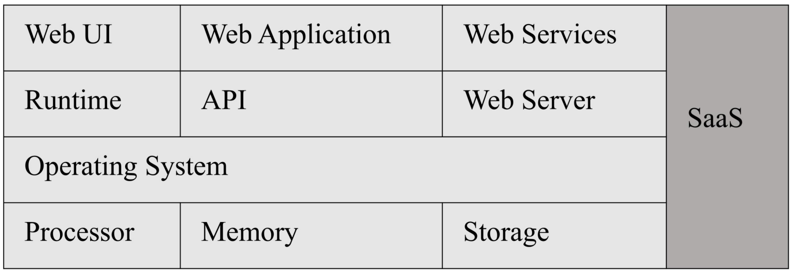
Cloud computing usually uses high-level programming languages such as Java and Python. Cloud computing also enables customers to choose from a variety of software platforms, including Windows Server, OpenSolaris, and seven Linux distributions, as well as MySQL and SQL Server, as well as some development environments such as Oracle 11, Google database, Java, JBoss, and Ruby on Rails [24]. One of the key characteristics of cloud-based services is the use of data centres in multiple locations to optimize network performance, service utility, and localization efficiency. The ability of cloud-based services to dynamically provision and decommission resources in response to current demand is make it capable to reduce operational costs is one of cloud computing key specification [25]. Recently, cloud computing is evolving in the direction of embracing sustainability and environmental consciousness. There have been discussions regarding emerging developments in green cloud computing, along with the categorization of various energy-efficient techniques utilized within data centres [26].

Cloud computing evolved as a successful reuse paradigm, in which software functionality, hardware processing power, and other computer resources are supplied as a service and made broadly available to users [23]. In order to serve the service to users, a Software as a Service (SaaS) is designed to allow clients manage their applications over the Internet. SaaS has supported millions of users with its scalable performance in variety of environments and platforms [27].

3.2 Software as a Service (SaaS)

Fundamentally, Software as a Service (SaaS) is a subset of cloud computing. Cloud computing use SaaS primarily concerned with providing application access to cloud end users, such that numerous users can run the same application binary in their own virtual machine or server instance. Although these application sessions may be operating on the same or different underlying hardware, the SaaS model allows application vendors to seamlessly update and repair their binary distributions [27]. Example of SaaS is Human Resource Management (HRM) service from Employease.com and Customer Relationship Management (CRM) service from Salesforce.com [28]. There are two types of SaaS: horizontal SaaS and vertical SaaS [29]. Vertical SaaS have been developed to meet the needs of various companies in terms of organizing project management and control while horizontal SaaS is products that are specialized in a particular type of software [29].

Software as a service (SaaS) is a cloud computing service that facilitates software delivery and becomes a viable choice for businesses of all sizes and types [30]. The structure of SaaS is shown in Figure 1 and by using SaaS, customers are not required to install or execute the program on their local computers, thus, decrease a load of software maintenance for customers while also lowering the cost of software acquisitions [15]. SaaS is defined as a software application or service that is deployed across a network from a centralized data centre, providing access and use on a recurrent charge basis, with users often renting the applications or services from a central provider [31]. It gains popularity as customer does not need to build their own IT infrastructure and just simply used on the hosting's company [32]. SaaS can be implemented by purchasing a software license with SaaS and cloud provider and paying fees based on usage.



Web UI	Web Application	Web Services	SaaS
Runtime	API	Web Server	
Operating System			
Processor	Memory	Storage	

Fig. 1. SaaS Structure [33]

In 2010s, migration from SaaS towards Service Oriented Architecture (SOA) has emerged which enables software functionality to be made accessible in the form of services to anybody who has been granted permission to use the service [34]. SaaS certain limitations as it is bound to cloud provider's as they are the one who runs the infrastructures. There is a limitation in changing business rules and the exchange of business information through a third party.

3.3 Service Oriented Architecture (SOA)

From SaaS, SOA has been the talk of the town as SOA achieves specifically technical and operational characteristics, such as run-time identification of new business processes, interoperability challenges, information connectivity and collaboration with other parties [32].

Service-Oriented-Architecture (SOA) is a conceptual business architecture in which business functionality or application logic is made available on an IT network to SOA users, or consumers [35]. SOA emerges as an architectural style that improves current traditional systems' service delivery performance while keeping their most significant aspects [36]. SOA allows for information transfer across enterprise boundaries and enables the implementation of diverse applications via service components that increase the ease and flexibility with which business processes can be reconfigured [35].

SaaS takes advantage of Service Oriented Architecture (SOA) that allows software programs to connect with one another and with the outside world. SaaS allows both the service provider and the service requester to incorporate data and functionality from other services. As a result, SaaS requires SOA to meet peak use demands while also having the capacity to execute huge quantities of transactions in a safe and dependable environment [33]. SOA allows software systems to be formed of a number of loosely linked software components known as services which are used to compose software systems [37]. SOA is loosely coupled which means solutions developed using this approach are more flexible and scalable when compared to solutions developed using a tight coupling between software modules [38]. SOA can be written in any programming language, including Java, Microsoft.Net, Cobol, or any other computer language [7]. But since SOA is highly dependent, if an application is written in Java, thus any added modules or service need to be written in Java.

When interacting with SOA, a service requestor will either request a service from a service provider to perform a task or search the service registry for the services that are required to complete the task (Figure 2). The service provider is responsible for registering new services and providing all the information about these new services in the registry so others can use it. This service provider also publishes and implements web services to make them available on the network. An example of an application that successfully adopted SOA is Vanilla Sales Application that aims to ease sales process with powerful performance which benefits companies in efficiently conducting business. In recent years, the SOA practices have shifted towards microservice where it is influenced by Domain Driven Design (DDD) that assists business analysts and enterprise architects in the development of scalable applications that have the flexibility for the addition of new functions as the organization grows [39].

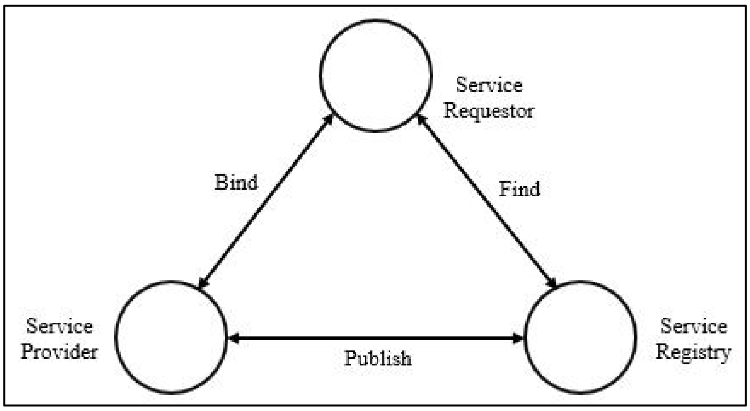


Fig. 2. Service Oriented Architecture [40]

3.4 Microservices

Motivated from SOA, microservices has gained attentions for these past years. In recent years, microservice has become a boom where it is called to lessen the complexity of monolith application.

Adoption of microservice is progressing. Microservice is the current trend, and it is called ‘micro’ because it is relatively small works in application. This technology is gaining traction by the industries as it allows agile delivery methods in service-oriented designs, as well as shifting away from function-oriented traditional systems and toward highly flexible service-oriented architectures. The microservice term is first discussed in 2014 by James Lewis and Martin Fowler [41]. Lewis and Fowler defined microservice as a method of building a program as a collection of small services, each executing in its process and interacting through lightweight mechanisms, most typically an HTTP resource API, rather than as a single large application.

Before the existence of microservice, most web-based systems were designed using a monolithic architectural style until the concept of microservices emerged [42]. Microservices have gained tremendous popularity in recent years. Enterprise IT is increasingly embracing microservices. Arguably, microservice is a part of Service Oriented Architecture (SOA) but SOA depends on heavyweight middleware such as Simple Object Access Protocol (SOAP) and Web Services Description Language (WSDL) while microservice rely on lightweight technologies such as Representational State Transfer (REST) and HTTP [41]. A few years ago, microservices began to arise and have been successfully adopted by large companies such as Google, Amazon, Netflix, Spotify, LinkedIn, SoundCloud, The Guardian, Uber and Verizon [42]. There are two types of microservices which is stateless microservice and statefulness microservice. Stateless microservice do not maintain session state between requests, so removing a service instance does not affect the service's overall processing logic. Statefulness microservices store session information in the code that runs on the client side.

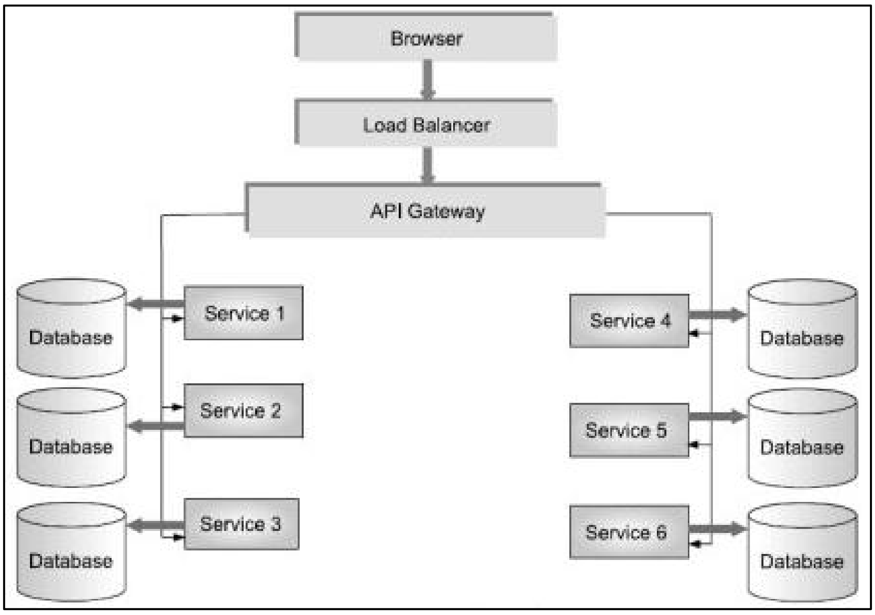


Fig. 3. Microservice Architecture [43]

Compared to SOA, microservice offers a fine-grained service granularity in which each service can have independent data storage (shown in Figure 3) compared to SOA here SOA is more modular service that involves sharing data storage between services [44]. Microservice is a more adaptable software systems due to independent service deployment and easier replacement, isolated testability which increases in service quality and safety, resilience, and runtime scalability by Rademacher [45]. Microservice can be written in any language same as SOA but microservice can use any language on each service as it is highly independent. Microservices are typically built in

Java, which is a popular programming language. Other programming languages, such as Golang and Python, may also be [7].

Microservice is a small and lightweight services that are specifically designed to perform on a very cohesive business function and is an evolution of the traditional service-oriented architecture style of architecture that emphasises division of the system [41]. Each microservice is centred on a business capability, runs in its own process, and communicates with the other microservices in an application via lightweight mechanisms, allowing for better reliability and adaptability to changing workloads via fault detection and autoscaling policies at the service level. Microservices allows disjoint teams to develop independently part of an application without complete system knowledge. Microservice is the current trend of network application. Microservices also supported the current trend in this Industry 4.0 era where Internet-of-Things (IoT) has been made popular and useful to society [47,48]. Microservices provided the IoT application with better flexibility, scalability and platform dependency. This is because microservice supported the possibility of upgrading the plugin without affecting the overall system, or even replacing the entire module of a microservice by interacting with the core service only and this, core service will then interact with the device by using the related plugins. Furthermore, the device plugins can be mounted or unmounted dynamically, and the plugins themselves can be upgraded or replaced freely [28].

4. Discussion

This study has identified that cloud computing primarily focuses on delivering services, while SaaS architecture has emerged as a preferable option for facilitating software delivery. Additionally, SOA has been implemented to manage business functionality. In the 2010s, with increasing business complexity, microservices architecture gained prominence and became the preferred choice for many large companies. Nevertheless, confusion persists as these terms are often used interchangeably, making it challenging to discern their distinct philosophies. Table 2 in this study provides a comparison of these network applications across various dimensions, including their definitions, types, programming languages, interoperability, characteristics, and illustrative examples. This comparative analysis aims to clarify these technologies' nuances and assist stakeholders in making informed decisions about their adoption and implementation strategies.

In the beginning, cloud computing has been used to perform business by storing data on 'cloud'. By using cloud computing, users do not need to worry about the construction, how it works, who operates it or where it is located. It also can save their data on cloud rather than on local infrastructure which required them to pay on the storage that they use. On the other hand, there is limitation storing data in cloud computing. For big business, using cloud computing is not flexible as there will be issues on security. Company's sensitive information will be jeopardized as we share the data to cloud service provider. Furthermore, the cloud is vulnerable of attacks and threats. This could potentially impose a great risk to the company. Furthermore, there is possible downtime which using cloud computing means fully rely on internet connection. In order to serve the service to users, a Software as a Service (SaaS) is designed to allow clients manage their applications over the Internet.

Software as a service (SaaS) is a type of business model that allows companies to deliver software as a service rather than owning and maintaining software locally. SaaS is primarily concerned with the delivery of services, rather than the transfer of ownership of software to the end users. By using SaaS, IT control is easier, there are no installation and development and cost [3]. SaaS enables the implementation of software changes to be completed more quickly such as IT installation, control, and maintenance. The user is required to pay a fee for the use of the software or for the duration of time that the software is in use by the user that provides by provider host. However, security is the main distress of SaaS. Customer data security, as well as potential contractual loopholes that could be exploited to the detriment of customers, are the primary concerns for SaaS providers [49]. Similar to cloud computing, both software and information are stored on the systems of the SaaS vendor, whereas the ownership of the information is held by the user of the service [3]. Thus, it is required for insource software development, thus, SOA has been introduced to solve this problem.

Table 2
Comparison of Network Applications

	Cloud Computing	SaaS	SOA	Microservice
Definition	Cloud computing is a business model that offer on-demand network access to a shared pool of programmable computing resources to deliver IT services, program and data which can be located remotely [16,17,27].	Software as a service (SaaS) is a cloud computing service that facilitates software delivery that is deployed across a network from a centralized data centre and becomes a viable choice for businesses of all sizes and types [29].	SOA is a conceptual business architecture in which business functionality, applications, and services that can be to integrate across operating systems, platforms, languages, and networks [40].	Microservices is an approach that represent business capabilities that have been defined through domain-driven design for developing a single application as a collection of small and autonomous services, each of which runs in its own process and communicates via lightweight mechanisms [41,50].
Type	Public cloud, Private Cloud, Hybrid Cloud, Community cloud	Vertical SaaS, Horizontal SaaS	Service provider, Service consumer, Service registry	Stateless microservice and statefulness microservice
Language	Use high-level programming language such as Java, Python	Use high-level programming language such as Java, Python	can be written in any programming language. One application can use only one language.	can be written in any programming language. different services able to use different language.
Interoperability	Facilitates data and service exchanges between various cloud infrastructure	Software applications running on cloud infrastructure are made available as a service to end users	Use heterogenous messaging protocols such as SOAP, AMQP and MSMQ.	Use lightweight messaging protocols like HTTP/REST
Characteristics	Resource sharing, Multi-Located Data Centres Resource Provisioning, On Demand, Self Service	Subscription based billing Single sign on Automated provisioning	Coarse-grained interface, highly dependent, Loosely coupled, Self-contained	Highly independent API gateways, Small and lightweight services, Resilience, lightweight containers, Domain Driven Design (DDD)
Examples	Amazon’s Elastic Cloud Computing (EC2), Nirvanix, Salesforce.com	Human Resource Management (HRM) service, Customer Relationship Management (CRM)	Wachovia Bank’s corporate, Investment Banking division	Google, Amazon, Netflix, Spotify, LinkedIn, SoundCloud, The Guardian, Uber, Verizon

SOA is an approach that can be used to integrate different services across operating systems, platforms, languages, and networks, and which has several benefits [40]. SOA promises many other benefits, including reduced application development time and maintenance costs, real-time information, and better alignment between IT capabilities and business requirements [51].

However, SOA is not suitable in an environment of application that based on heavy data exchange such as Maps. This is because it can increase the server's workload and management overhead due to transmission control protocol (TCP). SOA is dependent on each other, thus, if one service failed to run, it can affect both client and server side. In term of scalability, it is hard to extend an existing application, and it requires additional development to do so. To overcome the drawback of SOA, microservice has evolved and is still evolving to solve the modern business demands.

Microservice is a method of building a program as a collection of small services, each executing in its process and interacting through lightweight mechanisms [41]. Microservice architecture is used to create loosely coupled systems, which means that the system's components can work independently of one another. Every service is created with the goal of completing the task, and the most appropriate tool is chosen for each. Hence, many teams and developers can work independently of one another while contributing to the overall system's cohesiveness. For that reason, a faster deployment can be achieved. As a result, microservices began to arise and have been successfully adopted by large companies such as Google, Amazon, Netflix, Spotify, LinkedIn, SoundCloud, The Guardian, Uber and Verizon [42,51].

5. Conclusion

Conclusively, this study has compared several of the network application that has been successfully adopted by many organizations. The study starts with introducing the network application and the related works. Then several of the network application which are cloud computing, SaaS, SOA, and microservices are discussed. This study is motivated by the misuse terms of cloud computing, SaaS, SOA, and microservices where this problem has raised issues where sometimes developers are confused on which technology they are using, and which ones are appropriate for them to adopt based on their situation.

Therefore, this study has made a comparative study where cloud computing, SaaS, SOA, and microservices are being compared based on their definition, type, language, interoperability, characteristics, and examples. Based on this comparison, this study found that cloud computing is a way to store and manage data through the internet while SaaS is providing the application with access to cloud end users that run on the same application binary in their virtual machine or server. This study also has found that SOA is a software architecture that allows for cross-company data sharing and the deployment of diverse applications via service components, increasing the ease and flexibility of business process reconfiguration. Moreover, microservice is similar to SOA but the way microservice presented the service is different. Each service in microservice is on its own, unlike SOA that depends on each other in terms of data storage, communication, and deployment. Microservice is the recent trend. It has gained prominence due to its ability to address the evolving demands of modern applications, enabling organizations to achieve greater agility and efficiency in their software development processes. As a result, microservices have become a pivotal and influential trend shaping the future of software engineering and application development.

In the future, this study will be instrumental in guiding researchers and practitioners towards a deeper understanding of the challenges and the adoption landscape of microservices. It will actively contribute to this knowledge domain by undertaking a comprehensive survey to identify and elucidate the critical success factors associated with the implementation and utilization of microservices.

Acknowledgement

This research was supported by Ministry of Higher Education (MOHE) through Fundamental

Research Grant Session (FRGS/1/2020/ICT03/UTHM/03/1).

References

[1] Roberts, Lawrence G., and Barry D. Wessler. "Computer network development to achieve resource sharing." In *Proceedings of the May 5-7, 1970, spring joint computer conference*, pp. 543-549. 1970. https://doi.org/10.1145/1476936.1477020

[2] Kizza, Joseph Migga, Wheeler Kizza, and Wheeler. *Guide to computer network security*. Vol. 8. Berlin: Springer, 2013. https://doi.org/10.1007/978-1-4471-4543-1

[3] Janssen, Marijn, and Anton Joha. "Challenges for adopting cloud-based software as a service (saas) in the public sector." (2011).

[4] Nassif, Ali Bou, and Miriam AM Capretz. "Offering SaaS as SOA services." In *Innovations and advances in computer, information, systems sciences, and engineering*, pp. 405-414. Springer New York, 2013. https://doi.org/10.1007/978-1-4614-3535-8_35

[5] Manduca, Alexandre Michetti, Ethan V. Munson, Renata PM Fortes, and Mariada Graça C. Pimentel. "A nonintrusive approach for implementing single database, multitenant services from web applications." In *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, pp. 751-756. 2014. https://doi.org/10.1145/2554850.2555072

[6] Zhang, Qi, Lu Cheng, and Raouf Boutaba. "Cloud computing: state-of-the-art and research challenges." *Journal of internet services and applications* 1 (2010): 7-18. https://doi.org/10.1007/s13174-010-0007-6

[7] Team, IBM Cloud. "SOA vs. Microservices: What's the Difference." *IBM. Retrieved March* 30 (2021): 2022.

[8] Kazanavičius, Justas, and Dalius Mažeika. "Migrating legacy software to microservices architecture." In *2019 Open Conference of Electrical, Electronic and Information Sciences (eStream)*, pp. 1-5. IEEE, 2019. https://doi.org/10.1109/eStream.2019.8732170

[9] Carrasco, Andrés, Brent van Bladel, and Serge Demeyer. "Migrating towards microservices: migration and architecture smells." In *Proceedings of the 2nd International Workshop on Refactoring*, pp. 1-6. 2018. https://doi.org/10.1145/3242163.3242164

[10] Kacha, Lynda, and Abdelhafid Zitouni. "An overview on data security in cloud computing." *Cybernetics Approaches in Intelligent Systems: Computational Methods in Systems and Software 2017, vol. 1* (2018): 250-261. https://doi.org/10.1007/978-3-319-67618-0_23

[11] Familiar, Bob. *Microservices, IoT, and Azure*. Berkeley, CA: Apress, 2015. https://doi.org/10.1007/978-1-4842-1275-2

[12] Andriyanto, Agustinus, Robin Doss, and Purnomo Yustianto. "Adopting SOA and microservices for inter-enterprise architecture in SME communities." In *2019 International Conference on Electrical, Electronics and Information Engineering (ICEEIE)*, vol. 6, pp. 282-287. IEEE, 2019. https://doi.org/10.1109/ICEEIE47180.2019.8981437

[13] Fithri, Diana Laily, Andy Prasetyo Utomo, and Fajar Nugraha. "Implementation of SaaS cloud computing services on E-learning applications (case study: PGRI foundation school)." In *Journal of Physics: Conference Series*, vol. 1430, no. 1, p. 012049. IOP Publishing, 2020. https://doi.org/10.1088/1742-6596/1430/1/012049

[14] Raj, Vinay, and Ravichandra Sadam. "Performance and complexity comparison of service oriented architecture and microservices architecture." *International Journal of Communication Networks and Distributed Systems* 27, no. 1 (2021): 100-117. https://doi.org/10.1504/IJCNDS.2021.116463

[15] Wang, Lizhe, Gregor Von Laszewski, Andrew Younge, Xi He, Marcel Kunze, Jie Tao, and Cheng Fu. "Cloud computing: a perspective study." *New generation computing* 28 (2010): 137-146. https://doi.org/10.1007/s00354-008-0081-5

[16] Seda, Pavel, Pavel Masek, Jindriska Sedova, Milos Seda, Jan Krejci, and Jiri Hosek. "Efficient architecture design for software as a service in cloud environments." In *2018 10th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, pp. 1-6. IEEE, 2018. https://doi.org/10.1109/ICUMT.2018.8631237

[17] Alam, Tanweer. "Cloud Computing and its role in the Information Technology." *IAIC Transactions on Sustainable Digital Innovation (ITSDI)* 1, no. 2 (2020): 108-115. https://doi.org/10.34306/itsdi.v1i2.103

[18] Dang, L. Minh, Md Jalil Piran, Dongil Han, Kyungbok Min, and Hyeonjoon Moon. "A survey on internet of things and cloud computing for healthcare." *Electronics* 8, no. 7 (2019): 768. https://doi.org/10.3390/electronics8070768

[19] Oztemel, Ercan, and Samet Gursev. "Literature review of Industry 4.0 and related technologies." *Journal of intelligent manufacturing* 31, no. 1 (2020): 127-182. https://doi.org/10.1007/s10845-018-1433-8

[20] Ismail, Nuur Ezaini Akmar, Noraida Haji Ali, Masita Abdul Jalil, Farizah Yunus, and Ahmad Dahari Jarno. "A Proposed Framework of Vulnerability Assessment and Penetration Testing (VAPT) in Cloud Computing Environments from Penetration Tester Perspective." *Journal of Advanced Research in Applied Sciences and Engineering Technology* 39, no. 1 (2024): 1-14. https://doi.org/10.37934/araset.39.1.114

[21] Kostoska, Magdalena, Marjan Gusev, and Sasko Ristov. "An overview of cloud interoperability." In *2016 Federated Conference on Computer Science and Information Systems (FedCSIS)*, pp. 873-876. IEEE, 2016. https://doi.org/10.15439/2016F463

[22] Rani, B. Kezia, B. Padmaja Rani, and A. Vinaya Babu. "Cloud computing and inter-clouds–types, topologies and research issues." *Procedia Computer Science* 50 (2015): 24-29. https://doi.org/10.1016/j.procs.2015.04.006

[23] Lee, Jae Yoo, Jung Woo Lee, and Soo Dong Kim. "A quality model for evaluating software-as-a-service in cloud computing." In *2009 seventh ACIS international conference on software engineering research, management and applications*, pp. 261-266. IEEE, 2009. https://doi.org/10.1109/SERA.2009.43

[24] Zeng, Q. P., and S. X. Wu. "A Saa S Development Platform Based on Cloud Computing." *Indonesian Journal of Electrical Engineering and Computer Science* 11, no. 3 (2013). https://doi.org/10.11591/telkomnika.v11i3.2320

[25] Moghaddam, Faraz Fatemi, Mahsa Baradaran Rohani, Mohammad Ahmadi, Touraj Khodadadi, and Kasra Madadipouya. "Cloud computing: Vision, architecture and Characteristics." In *2015 IEEE 6th control and system graduate research colloquium (ICSGRC)*, pp. 1-6. IEEE, 2015. https://doi.org/10.1109/ICSGRC.2015.7412454

[26] Bharany, Salil, Sandeep Sharma, Osamah Ibrahim Khalaf, Ghaida Muttashar Abdulsahib, Abeer S. Al Humaimeedy, Theyazn HH Aldhyani, Mashael Maashi, and Hasan Alkahtani. "A systematic survey on energy-efficient techniques in sustainable cloud computing." *Sustainability* 14, no. 10 (2022): 6256. https://doi.org/10.3390/su14106256

[27] Tsai, WeiTek, XiaoYing Bai, and Yu Huang. "Software-as-a-service (SaaS): perspectives and challenges." *Science China Information Sciences* 57 (2014): 1-15. https://doi.org/10.1007/s11432-013-5050-z

[28] Sun, Long, Yan Li, and Raheel Ahmed Memon. "An open IoT framework based on microservices architecture." *China Communications* 14, no. 2 (2017): 154-162. https://doi.org/10.1109/CC.2017.7868163

[29] Naidu, Haripriya. "What makes a SaaS company cross the chasm?–Analysing factors that make customers purchase a freemium service." PhD diss., Dublin Business School, 2019.

[30] Seethamraju, Ravi. "Adoption of software as a service (SaaS) enterprise resource planning (ERP) systems in small and medium sized enterprises (SMEs)." *Information systems frontiers* 17 (2015): 475-492. https://doi.org/10.1007/s10796-014-9506-5

[31] Loukis, Euripidis, Marijn Janssen, and Ianislav Mintchev. "Determinants of software-as-a-service benefits and impact on firm performance." *Decision Support Systems* 117 (2019): 38-47. https://doi.org/10.1016/j.dss.2018.12.005

[32] Jing, Junjie, and Jian Zhang. "Research on open SaaS software architecture based on SOA." In *2010 International Symposium on Computational Intelligence and Design*, vol. 2, pp. 144-147. IEEE, 2010. https://doi.org/10.1109/ISCID.2010.125

[33] Kulkarni, Gurudatt, Jayant Gambhir, and Rajnikant Palwe. "Cloud computing-software as service." *International Journal of Cloud Computing and Services Science* 1, no. 1 (2012): 11-16. https://doi.org/10.11591/closer.v1i1.218

[34] Nassif, Ali Bou, and Miriam AM Capretz. "Moving from SaaS applications towards SOA Services." In *2010 6th World Congress on Services*, pp. 187-188. IEEE, 2010. https://doi.org/10.1109/SERVICES.2010.63

[35] Mohammadi, Mohsen, and Muriati Mukhtar. "Service-oriented architecture and process modeling." In *2018 International Conference on Information Technologies (InfoTech)*, pp. 1-4. IEEE, 2018. https://doi.org/10.1109/InfoTech.2018.8510730

[36] Niknejad, Naghmeh, Waidah Ismail, Imran Ghani, Behzad Nazari, and Mahadi Bahari. "Understanding Service-Oriented Architecture (SOA): A systematic literature review and directions for further investigation." *Information Systems* 91 (2020): 101491. https://doi.org/10.1016/j.is.2020.101491

[37] Elkholy, Mohamed, Youcef Baghdadi, and Marwa Marzouk. "Snowball framework for web service composition in SOA applications." *International Journal of Advanced Computer Science and Applications* 13, no. 1 (2022). https://doi.org/10.14569/IJACSA.2022.0130143

[38] Ahamad, Shahanawaj. "Contemporary Research Challenges and Applications of Service Oriented Architecture." *Available at SSRN 3948566* (2021). https://doi.org/10.2139/ssrn.3948566

[39] Shadija, Dharmendra, Mo Rezai, and Richard Hill. "Towards an understanding of microservices." In *2017 23rd International Conference on Automation and Computing (ICAC)*, pp. 1-6. IEEE, 2017. https://doi.org/10.23919/IConAC.2017.8082018

[40] Hamzah, Mohd Hamdi Irwan, Fauziah Baharom, and Haslina Mohd. "An exploratory study for investigating the issues and current practices of service-oriented architecture adoption." *Journal of Information and Communication Technology* 18, no. 3 (2019): 273-304. https://doi.org/10.32890/jict2019.18.3.8291

[41] Jamshidi, Pooyan, Claus Pahl, Nabor C. Mendonça, James Lewis, and Stefan Tilkov. "Microservices: The journey so far and challenges ahead." *IEEE Software* 35, no. 3 (2018): 24-35. https://doi.org/10.1109/MS.2018.2141039

[42] Cerny, Tomas, Michael J. Donahoo, and Michal Trnka. "Contextual understanding of microservice architecture: current and future directions." *ACM SIGAPP Applied Computing Review* 17, no. 4 (2018): 29-45. https://doi.org/10.1145/3183628.3183631

[43] Pachghare, Vinod Keshaorao. "Microservices architecture for cloud computing." *architecture* 3 (2016): 4.

[44] Pautasso, Cesare, Olaf Zimmermann, Mike Amundsen, James Lewis, and Nicolai Josuttis. "Microservices in practice, part 2: Service integration and sustainability." *IEEE Software* 34, no. 02 (2017): 97-104. https://doi.org/10.1109/MS.2017.56

[45] Rademacher, Florian, Sabine Sachweh, and Albert Zündorf. "Differences between model-driven development of service-oriented and microservice architecture." In *2017 IEEE International Conference on Software Architecture Workshops (ICSAW)*, pp. 38-45. IEEE, 2017. https://doi.org/10.1109/ICSAW.2017.32

[46] Buchgeher, Georg, Mario Winterer, Rainer Weinreich, Johannes Luger, Roland Wingelhofer, and Mario Aistleitner. "Microservices in a small development organization: An industrial experience report." In *Software Architecture: 11th European Conference, ECSA 2017, Canterbury, UK, September 11-15, 2017, Proceedings 11*, pp. 208-215. Springer International Publishing, 2017. https://doi.org/10.1007/978-3-319-65831-5_15

[47] Bigheti, Jeferson A., Michel M. Fernandes, and Eduardo P. Godoy. "Control as a service: a microservice approach to Industry 4.0." In *2019 II Workshop on Metrology for Industry 4.0 and IoT (MetroInd4. 0&IoT)*, pp. 438-443. Ieee, 2019. https://doi.org/10.1109/METROI4.2019.8792918

[48] Zainal, Salbiah, Rasimah Che Mohd Yusoff, Hafiza Abas, Suraya Yaacub, and Norziha Megat Zainuddin. "Review of design thinking approach in learning IoT programming." *International Journal of Advanced Research in Future Ready Learning and Education* 24, no. 1 (2021): 28-38.

[49] Benlian, Alexander, and Thomas Hess. "Opportunities and risks of software-as-a-service: Findings from a survey of IT executives." *Decision support systems* 52, no. 1 (2011): 232-246. https://doi.org/10.1016/j.dss.2011.07.007

[50] Dragoni, Nicola, Saverio Giallorenzo, Alberto Lluch Lafuente, Manuel Mazzara, Fabrizio Montesi, Ruslan Mustafin, and Larisa Safina. "Microservices: yesterday, today, and tomorrow." *Present and ulterior software engineering* (2017): 195-216. https://doi.org/10.1007/978-3-319-67425-4_12

[51] Yoon, Tom, and Bong-Keun Jeong. "Service oriented architecture (SOA) implementation: Success factors and realized benefits." *International Journal of Information Systems in the Service Sector (IJISSS)* 10, no. 2 (2018): 1-21. https://doi.org/10.4018/IJISSS.2018040101