

A Comprehensive Comparative Study of MOM for Adaptive Interoperability Communications in Service Oriented Architecture

Najhan Muhamad Ibrahim¹, Mohd Fadzil Hassan²

¹Post Doctoral Fellow, ²Associate Professor

¹Department of Computer Science and Information Systems (CSIS), University of Limerick, Limerick, Ireland

²Department of Computer and Information Sciences, Universiti Teknologi Petronas, Perak, Malaysia

How to cite this paper: Najhan Muhamad Ibrahim | Mohd Fadzil Hassan "A Comprehensive Comparative Study of MOM for Adaptive Interoperability Communications in Service Oriented Architecture" Published in International Journal of Trend in Scientific Research and Development (ijtsrd), ISSN: 2456-6470, Volume-3 | Issue-3, April 2019, pp. 23-30. <http://www.ijtsrd.com/papers/ijtsrd21595.pdf>



IJTSRD21595

Copyright © 2019 by author(s) and International Journal of Trend in Scientific Research and Development Journal. This is an Open Access article distributed under the terms of the Creative Commons Attribution License (CC BY 4.0) (<http://creativecommons.org/licenses/by/4.0>)



ABSTRACT

Numerous software platforms have developed to support several application systems with a variety of techniques and solution. The communications between those platforms are required to ensure tasks distributed across different application systems. The collaboration between distinct application systems to grow, in tandem with the growth of SOA in which the resources of each SOA application might be developed and implemented using different standard and specification. Within this context, interoperability communications is significant to consolidate the optimization of these resources. Additionally, identifying the requirements attribute related to interoperability communications are the first step required in the development of interoperability communications. MOM can be considered as a tool for enabling the communications between distributed applications where the transaction or event notifications are delivered between distributed applications through the message. MOM provides a program-to-program connection by message passing. Most of the MOM environments are implemented using queued message store-and-forward capability, which is Message Queuing Middleware (MQM). In addition, an adaptive interoperability communications can be defined as the computer system that enables self-coordination to distribute computing resources and an adaptive to the new environment while hiding fundamental complexity from the operators and users. The intelligent system has the capability of self-management to overcome the rapidly growing complexity in computing systems. The system makes decisions on its own by using the rules that have been identified. Intelligent system also provides an adaptive and dynamic environment for the process of communication. The framework of an intelligent system, fundamentally consists of intelligent elements. Each element performs a fixed function and interacting with other elements in a very dynamic environment.

KEYWORDS: Service Oriented Architecture (SOA), Message Oriented Middleware (MOM), Web Services, Adaptive interoperability

INTRODUCTION

According to Rob van der Meulen from Gartner (2015), the worldwide market revenue for middleware application has increased 8.8 percent from \$21.8 billion in 2013 to \$23.8 billion in 2014. A growing number of organizations transforming into digital businesses is demanding new strategies for middleware application infrastructure. It is realized that the application infrastructures, designed in the last ten years, would not support the need for real-time analytics, development agility, deployment flexibility and fast reconfiguration of business networks required in the digital era [1]- [5].

There are several different types of middleware for connecting distributed applications, such as Message Oriented Middleware (MOM), Message Broker, Enterprise Service Bus (ESB) and Content-Centric Middleware [6]. MOM

is generally accepted as a tool for enabling the communications between distributed applications where the transactions or event notifications are delivered between distributed application through messages [7]. However, some weaknesses of MOM include its inability to trace the execution and rollback tasks at any level. The MOM's based execution models are unable to record any level of execution in applications and maintain the states of those executing tasks. It also has a limited ability to connect multiple types of applications with very low adaptive and flexibility [8]. The weaknesses of MOM are critical and need to be addressed to enhance MOM for supporting one-to-many interoperability communications by using agent technology. Research work is therefore needed to consolidate these issues. It is aimed to find a common ground toward the establishment of interoperability communications framework that enable

communications between different type of SOA-based applications.

Interoperability communications application has grown tremendously and the needs for different applications to communicate and collaborate with each other are in critical demand. The development of adaptive software application to support interoperability communications is now necessary [9]. An agent-based system has the ability to assess its environment and takes actions to achieve some specified goals [10]. Without the adaptive system, each application is required to wait for a respond from a partner application to complete the communication process [11].

BACKGROUND

A comprehensive literature review in the area of interoperability communications revealed that the interoperability communications framework for multiple types of SOA applications in runtime is not available. It was also found that the exiting similar framework did not incorporate the requirement attributes for interoperability communications. There are several specifications and standards of web services currently available to implement SOA. Different SOA applications may be developed or implemented using different web services and specifications to acquire the services of others in achieving the specified goals. In this context, interoperability communications is required to communicate between the different types of SOA-based applications. Without the interoperability communications, different SOA applications are not able to connect and communicate for the purpose of data

transmission [12]. However, the main challenge for interoperability communications is the standard requirements to construct interoperability communications framework in SOA. Most of the existing solutions do not support multiple types and independent interoperability communications in which the communications have to be specified manually by the user.

One of significant research works that focuses on interoperability communications framework, which models and discusses the requirement attributes for the collaboration between different application systems. Chituc, C.-M. Azevedo, A.r. and Toscano, C. [13] strongly argued that the development of interoperability communications framework to support communications among diverse and geographically distributed system are significantly important. They have proposed six main requirement attribute with twenty-seven sub-attributes, which is a general requirement for distributed and collaboration application as the following as shown in Table 1. In the sub-requirements, several significant requirements for interoperability communications have been highlighted. Nevertheless, the main proposed requirements for interoperability communications in this research work are too general to be incorporated in the concrete interoperability communications framework. This research work is most likely focus on non-functional requirement attributes. Therefore, the comprehensive literature study need to be conducted to evaluate and consolidate others requirement attribute to be included in the proposed interoperability communications.

Table 1: General Requirements for Interoperability communications [13]

Main requirements		Sub-requirements
1	General requirements for Description and Publication	<ul style="list-style-type: none"> ➤ Organization description ➤ Organization publication
2	General requirements for Potential business partner and Opportunity identification	<ul style="list-style-type: none"> ➤ Browse Information
3	General requirement for Collaboration	<ul style="list-style-type: none"> ➤ Messaging ➤ Inter-organizational collaboration ➤ Negotiation and agreement
4	General requirements addressing Management aspect	<ul style="list-style-type: none"> ➤ Information management ➤ Policy ➤ Management ➤ Dissemination
5	General requirements for Performance assessment	Performance assessment
6	General requirements for Non-functional aspect	<ul style="list-style-type: none"> ➤ Generality ➤ Comprehensibility ➤ versioning

The main research study on interoperability communications framework is the European interoperability communications frameworks in [14]. The goal is to support the European Union's strategy of making user-centered and e-Government services by enabling, at the European level, the integration of services and systems between public administration and the public (e.g., citizens, businesses). European interoperability communications frameworks provides specifications and defines the requirement aspects of standards with regards to organizational, semantic and technical E-health systems as presented in Figure 1.

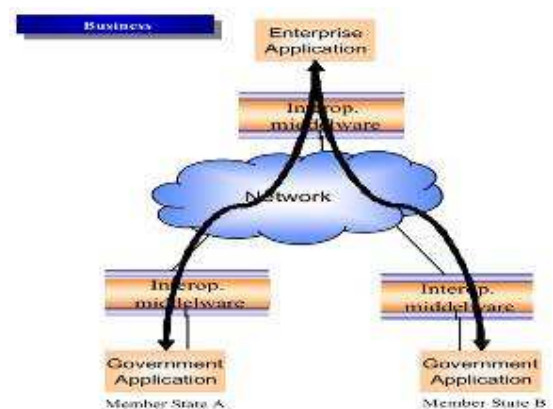


Figure 1: Scenario for European eGovernment Services [14]

Besides that the Cross Compiled Approach (CCA) approach for interoperability communications framework in [32] has shown interoperability communications specifications, such as (1) description specification, (2) communications, (3) naming interoperability (for both components and interfaces/ports), (4) discovery, and (5) creation. A diversity of methods can be used to meet these requirements. Eventually, they can be divided into the following categories of interoperability communications: (1) standard interfaces and protocols, (2) adoption for translation and mapping of constituents, and (3) a proxy to span each framework. This framework can be considered as a very theoretical framework, which does not show a clear diagram for the relationship of each component.

Similarly, the European initiatives to develop interoperability of enterprise applications basic concepts, framework and roadmap in [15]. D. Chena, and G. Doumeings were developed to expand the quality attributes. This framework has been proposed to explore the view that the framework has been developed to address interoperability communications requirements concerning among others the coordination of enterprises, integration of business processes, inter-connection of semantic applications and integration of syntactical applications. From the business point of view, all issues related to the organization and the management of an enterprise is considered as interoperability communications requirements. The conceptual model of the interaction between two enterprises as presented in Figure 2.

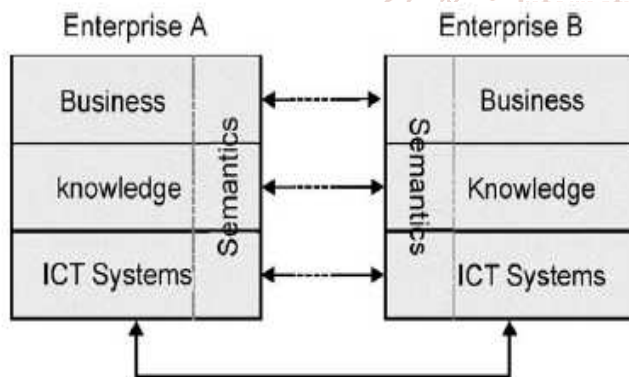


Figure 2: Interoperability on all layers of an Enterprise [15]

Likewise, E-health interoperability communications framework in [16] was developed by National E-Health Transition Authority (NEHTA) in Australia. The idea of this interoperability communications framework is to link organizational aspects, information aspects, and technical aspects, which are related to the delivery of the interoperability communications through health organizations.

In addition, the Enterprise interoperability communications framework in [17] can be used to structure knowledge. This framework shows the concept of interoperability communications barriers. It also shows the use of the framework to identify and categorize the knowledge that removes both syntactic and semantic barriers but is limited to the process level and a few common attributes were not considered such as negotiation, agreement, learning and performance.

In a different perspective, the Evaluative Framework for interoperability communications in [18] has been developed to resolve various interoperability communications issues concerning specific testing mechanisms. For example, interoperability communications testing for Object Request Broker (ORB) products is currently being carried out directly between ORB vendors rather than by an independent testing body, except for CORBA net. CORBA net explores ORB communications by linking hardware and operating software. There are proposed five main requirements attribute for interoperability communications, which included performance, scalability, security and reliability.

In summary, all of these studies have demonstrated the importance of requirement attributes for interoperability communications. This is due to the fact that the standard requirements for interoperability communications are not available. Moreover, the research also found that some of the proposed requirements redundant and can be considered as a sub-requirement. The research presented in this thesis address yet another kind of interoperability communications problem, which is adaptive interoperability communications. Due to the fast growing of software development and required to have collaborated, the proposed interoperability communications framework will also gain the benefit of the required attributes for interoperability communications as discussed in this subtopic. The outcomes of this research, hence, will provide significant contributions in the area of the current study.

ADAPTIVE INTEROPERABILITY COMMUNICATIONS

Adaptive is one of the significant elements in today's application where it has many advantages over the manual system. The manual systems are totally relying on the user to be accurate. The problem is that human is not consistent, in which each individual may respond differently. Without intelligent systems, the level of service is dependent on individuals and this puts a requirement on management to provide training continuously to keep them motivated and to ensure they are following the correct procedures. This has affected a great impact on quality trustworthy and depends on actions [19]. In addition, the manual system can be time consuming and expensive. It is also can increase the chance of duplication in data entry.

The increasing number of collaboration in multiple types of system derives the significant challenges for interoperability communications to be more adaptive. Variety of existing solutions based on self-regulating and automation components have been proposed. Abundant research studies have recently discovered on the significant research in the intelligent systems to accommodate interoperability interaction and communications, which will be discussed in the next section. Nevertheless, these research studies are typically conceived with specific issues or isolate solutions in mind and mostly address the need for reducing management costs rather than the need for enabling complex interoperability communications issues. The existing frameworks are poorly designed to support intelligent interaction between multiple platforms. The traditional framework has been constructed and coordinated based on a manual process for a single goal to resolve a specific issue. On the other hand, next-generation systems are expected to grow more rapidly with no centrally-control, and no specific message type to be used. By the release of central control

over the system has the potential to release an innovation for the future business process [8], [11], [15], [17].

In addition, over the last decade, artificial intelligent (AI) has shown great potential for solving complications in large scale, distributed and interoperability applications. The reason for the growing success of AI in this area is the inherent distribution allows a regular decomposition of the application into multiple interactions to achieve the global goal by using agent technology [42]. Agent-based technology can significantly enhance performance in the following three conditions. Firstly, the domain problem of geographically distributed and interoperability communications. Secondly, the sub-application exists in a dynamic environment. Lastly, the sub-application require more flexible interactive with each other. There are several types of artificially intelligent, one of them is agent-based

Agent-based approach is well suited for the domain of interoperability communications because of its geographically distributed and dynamic changing in nature [20]. The literature study shows that the techniques and methods resulting from the field of agent-based application have been applied to the many aspects of distributed and interoperability applications, including modeling and simulation for interoperability communications. In this section, will discuss some of the existing research works on improvement of agent-technology for interoperability communications. Such as Agent-based Middleware, Agent platform for reliable asynchronous distributed application and Multi-Agent System as a new paradigm for distributed and interoperability communications.

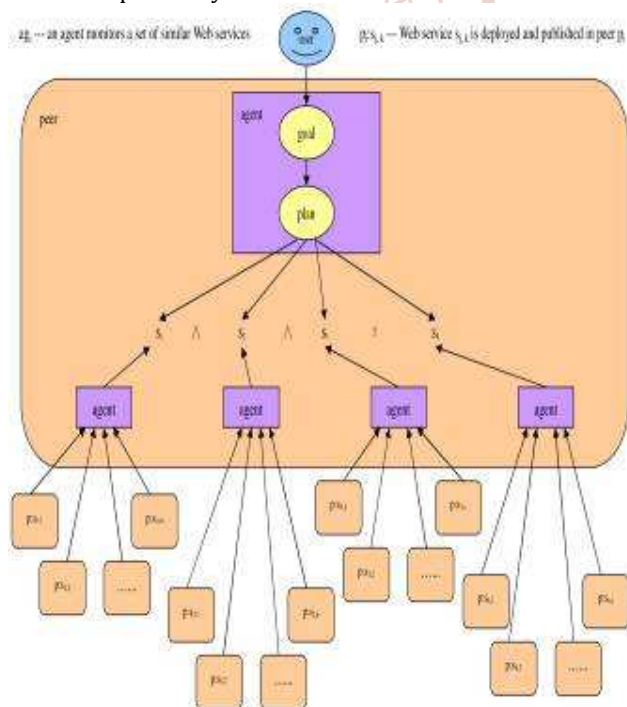


Figure 3: Web service Dynamic Integration Model [20]

One of the important research works in the adoption of agent technology for a distributed system is an agent-based middleware (AbM). A. Lin and P. Maheshwari [20] proposed to construct an Agent-based middleware for web service dynamic integration on peer-to-peer networks. It is to facilitate the integration of optimal quality of web services for application integration. The proposed solution only supports peer-to-peer network, however, it has a significant

focus on web service dynamic integration. This approach dynamically accomplishes the task on behalf of a user by employing the best quality web services that are purely distributed. Therefore, system developers can save time and resources by asking agents to collect as many web services as they need and understand their usages. This capability is very important for SOA environment where the web service is the main component.

INTELLIGENT-BASED WEB SERVICES INTERGRATION

Agent technology can be described as an adaptive and intelligent system to facilitate distributed and cross-platform environment. Therefore, interaction among distributed and cross-platform applications has been widely used agent technology. Consequently, there are valid significant needs of agent technology to access and manipulate web services. By allowing agent technology to access and manipulate web services, it will provide several advantages in ensuring the cross-platform communications between different SOA-based applications. In general, an agent will access a web service to read the content and encapsulate the message into agent communication language (ACL) to distribute the message within multi-agent-system [74, 121]. Multi-agent system (MAS) is the extensive subject of research to support the systems developed of intelligent software entities. The agents in MAS are able to act, react, participate, and collaborate independently.

In recent years, the interest in MAS has grown significantly, and it is being used in a large range of application. It is growing from information management, electronic commerce, business intelligence to distributed and cross-platform application. All these applications have one thing in common where agents must be able to communicate with each other to select the decision and action to be taken and how this action can be coordinated with other actions. The language used by the agent for this exchange is the agent communications language (ACL). An ACL is required to communicate and coordinate the action of an agent with other agents. It can be used to share information and knowledge among agents in a distributed and cross-platform computing environment, and also to request the performance of a task [88-90].

Agent Communications Language (ACL) has been proposed as standard language for agent communications by the Foundation for Intelligent Physical Agents (FIPA). Knowledge Query and Manipulation Language (KQML) meanwhile is another proposed standard as well. The most widespread ACLs are FIPA-ACL proposed by FIPA and KQML. Both rely on the speech act theory developed by Searle in 1960 and enhanced by Winograd and Flores in the 1970s. They defined a set of per-performatives, which are the list of FIPA communicative act specifications and their meanings. The content of the per-performatives is not standardized, however various systems are using this per-performative. To make agents understand each other, the agents not only have to speak the same language, but also have to rely on a common ontology. Ontology is a part of the agent's knowledge-base that describes what kind of situation an agent can deal with and how they are related to each other [38]. The main idea of an ACL is to represent an appropriate framework that allows different agents to interact and communicate with each other's. An important part of the agent approach is the concept that agents can be executed more effectively in dynamic and multiple environment [87].

Agent programs are designed to independently collaborate with each other in order to satisfy the global goal. Each agent makes the balance between collaboration and fulfilling its own goals individually which depending on the situation.

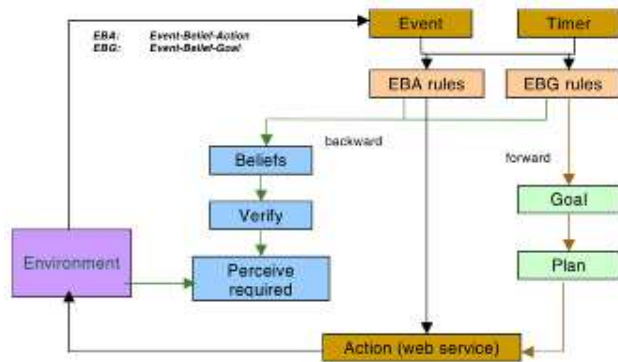


Figure 4: The Goal-driven Agent [20]

A. Lin and P. Maheshwari also proposed web service dynamic integration (WSDI), which is the concept of replacing web services with other better quality web services in integration prototypes at run time to maximize the quality of the performance. They provided agent panel to assign the tasks and monitor the options set of web services $\{si,1, si,2, \dots, si,mi\}$ as shown in Figure 3. It is responsible to choose one web service with the best quality. Then, web services are deployed and published the service where the agent perceives the joining peers of web services to remove the existing peer or web services. The authors also introduced the goal-driven agent as shown in Figure 4. It is based on intelligent control loop as explained early. The goal-driven are agent software components that employ reactive and proactive (goal-driven) to achieve goals. However, this research work is only focused on the peer-to-peer network, which is not suitable for distributed and interoperability application. Thus, it is significantly important to consolidate on publish-subscribe that is capable to support interoperability communications and dynamic environment.

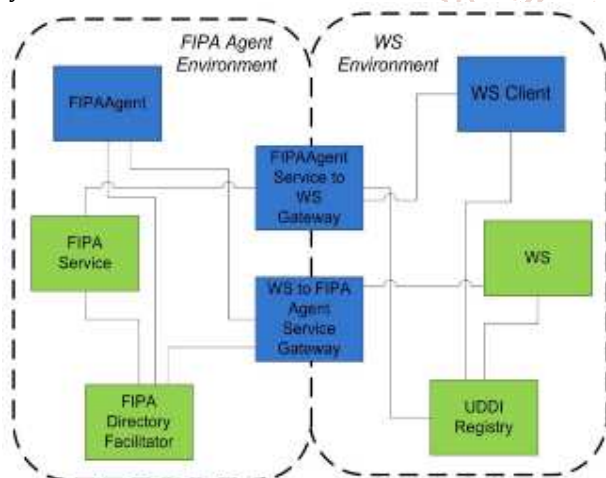


Figure 2: Agents and web services Communicate Gateways [20]

Xuan, T., N., and Ryszard, K., in [20] also proposed integrating web service and JADE agents (WS2JADE) as shown in Figure 2.11. The authors present the framework between WS and FIPA Agent environments, it allows FIPA agents to access web services by using agent service gateway to translate ACL messages into web services. Nevertheless, WS2JADE only aims to describe SOAP messages into java messages (ACL) and vice versa. It would face a difficulty to interact with others different type of web service standard and specification. As a final point, there has been accumulative amount of agent extension to support distributed and cross-platform application in the SOA environment. This is due to its great capability, facility, and flexibility of agent technology. These works have been carrying on by varieties of approaches by using multi agent system (MAS).

THE FINDING

Table 2 analyzes the improvements of MOM concerning eight requirement attributes for adaptive interoperability communications in SOA. The requirements attributes are specified in the first row of the table and the existing improvements are specified in the following of the table. Existing improvement of MOM is divided into two categories, synchronize and asynchronous communications. The improvements were selected from different kinds of extensions, in which some of them are the basic extension of MOM, which consist of significant highlighted point of view for interoperability communications. The attributes used in the evaluation were mainly selected from an interoperability communications requirements.

The existing MOM improvements are capable to communicate with complex tasks and provide the competence to expire the message after a certain amount of time. Nevertheless, most of them are still lack of consideration on requirement attributes for interoperability communications. From the comparison study, there are two main significant perspectives can be concluded. Firstly, none of the existing research works are supported by adaptive interoperability communications. Secondly, even though SOAP is widely used for communication, there are also numerous different message type has been used. Therefore, it would be very challenging for these extensions to communicate with other different types of SOA application, which were build based on different web services standard and specifications. To overcome these limitations, the comprehensive interoperability communications framework to manage the communications process is significant to consolidate.

TABLE 2: THE COMPARISION TABLE

	Authors	Objectives	Message type	Adaptive Interoperability Communications at runtime	Gap
1	Xu, Y.-z., D.-x. Liu, and F. Huang [35]	To resolve issues of asynchronous communications systems.	Compensating messages	Do not support	Do not support multiple types of interoperability communications in runtime and only focus on Compensating messages web services message.
2	Goel, S., H. Shada, and D. Taniar [37]	The work for loosely integrating interoperability and distributed systems.	SOAP	Do not support	The proposed communications model requires on time respond to complete the transaction.
3	Pietzuch, P.R. and S. Bhola [38]	Provide reliable and scalable message-oriented middleware by following the publish / subscribe communications style.	DCQ and UCA	Do not support	The proposed research work implemented by using different type of web services message.
4	Parkin, S., D. Ingham, and G. Morgan [39]	Provides reliable messaging across organizational limitations while implementing appropriate mechanisms for non-repudiation.	SOAP	Do not support	This research work emphasis on clients and servers that use SOAP RPC for interaction and WSDL to de-script services.
5	Yuan, P. and H. Jin, A [36]	Enables the integration of services and resources within and across enterprises or other organizations.	AMQP	Do not support	cMOM using AMQP message type for communications.
6	Ahn, S. and K. Chong [40]	To make use of an intelligent connection for information exchange in heterogeneous network environment.	SOAP	Do not support	Do not support one-to-many interoperability communications and only focus on SOAP web services message.
7	Tai, S., T.A. Mikalsen, and I. Rouvellou [41]	Comprehensive MOM for reliable web service messaging.	SOAP	Do not support	Do not support multiple types of interoperability communications and only focus for reliable SOAP web services message.
8	Laumay, P. [42]	Provide guarantee scalability through matrix clocks in Message Oriented Middleware (MOM).	ACL	Do not support	Using different message type for communications, which may have a difficulty to communicate between them.

CONCLUSION & FUTUREWORK

Interoperability mechanism is essential for different types of SOA applications that ensure the communications will be well maintained. The need for interoperability communications within SOA has attracted different research efforts as reported in various kinds of literature. This research interest is specifically focused on adaptive interoperability communications for multiple types of SOA applications. The literature survey includes several areas, such as SOA, Web services, MOM and Agent-based technology. The aim of this paper is to introduce and review these technologies and to present the current efforts as reported in various kinds of literature regarding interoperability communications within SOA environments.

Interoperability framework plays a significant seamless role in communication between different SOA-based applications. The lack of interoperable awareness, motivate researchers to focus on seamless attributes of web services and SOA of different deployment style. We found that some researchers focused on some perspective while others have attempted to integrate several specifications such as enterprise coordination, business process integration, semantic applications, syntactical applications, physical interaction, etc. We believe that more research in this area is still needed.

References

- [1] N. M.Ibrahim, and M.F. Hassan, 4. N. M.Ibrahim, and M.F. Hassan, Agent-based Message Oriented Middleware(MOM) for Cross-platform Communication in SOA Systems, International Conference on Computer and Information Sciences (ICCIS2012). University Teknologi PETRONAS, 2012, IEEE: KL.
- [2] H. Zaki and R. Ibrahim, Review on Web Service Architectures and Service Oriented Architecture Modelling Techniques for Monitoring Scholarly Publication Performance, Journal of Information Systems Research and Innovation, 10(1), 8-14, February 2016.
- [3] J. Tihomirovs and J. Grabis, Web Services: Comparison of SOAP and REST Based Web Services Using Software Evaluation Metrics; Information Technology and Management Science, Volume 19, Issue 1, Pages 92–97, ISSN (Online) 2255-9094, DOI: <https://doi.org/10.1515/itms-2016-0017>.
- [4] Mohsin A, Fatima S, Khan AU and Nawaz F, Automated Approach for Web Services Architectural Style Selection. Information Technology & Software Engineering, J Inform Tech Softw Eng 2016, 6:2.
- [5] R. Ahmad, A. Hussain and F. Baharom, Embedding the Concept of Service Oriented Architecture into Software Sustainability Evaluation Model. AIP Conference Proceedings 1891, 020023 (2017); doi: 10.1063/1.5005356.
- [6] M. Garriga, A. Cechich, A. Pablo Flores and A. Zunino, Web Services Composition Mechanisms: A Review. 8 March 2015, DOI: 10.1080.
- [7] M. Garriga, A. Flores, A. Cechich and A. Zunino, Web Services Composition Mechanisms: A Review, in International Conference on Web Services. Pages 376-383 | Published online: 16 Mar 2015.
- [8] M Suresh Kumar, P Varalakshmi, Dynamic web service composition based on network modeling with statistical analysis and backtracking. Computer Science & Education (ICCSE), 2012 7th International Conference, IEEE.
- [9] [64] C. M. Garcia and R. Abilio, Systems Integration Using Web Services, REST and SOAP: A Practical Report. Revista de Sistemas de Informação da FSCMA n. 19 (2017) pp. 34-41.
- [10] S. Shenoy. Roşu, An Adaptive Framework for Web Services Testing Automation Using JMeter, 2014 IEEE 7th International Conference on Service-Oriented Computing and Applications, IEEE.
- [11] L. Duan and H. Tian, CollaborativeWeb Service Discovery and Recommendation Based on Social Link, in Future Internet 2017, 9, 63; doi:10.3390/fi9040063.
- [12] B. Guerineau, C. Zheng, M. Bricogne, A. Durupt, L. Rivest, H. Rowson and B. Eynard, Management of Heterogeneous Information for Integrated Design of Multidisciplinary Systems, in Procedia CIRP 60 (2017) 320 – 325 ScienceDirect 27th CIRP Design 2017.
- [13] C.-M. Chituc, A. r. Azevedo, and S. Toscano, A framework proposal for seamless interoperability in a collaborative networked environment. Computers in Industry, 2009: p. 22.
- [14] IDA, European Interoperability Framework for the pan-European eGovernment services. 2004.
- [15] D. Chena, and G. Doumeingts, European initiatives to develop interoperability of enterprise applications—basic concepts, framework and roadmap. Annual Reviews in Control, 2003. 27.
- [16] Nehta, Towards an Interoperability Framework. 2005.
- [17] D. Chen, Interoperability Research for Networked Enterprises Applications and Software. 2004.
- [18] S. Mahajan, and J. Chen, CORBA On WWW: Evaluative Framework for Interoperability Issues. 1998, IEEE.
- [19] N. M.Ibrahim,. and M.F. Hassan, A Survey on Different Interoperability frameworks of SOA Systems Towards Seamless Interoperability, in ITsim '10. 2010, IEEE: Kuala Lumpur, KLCC.
- [20] A. Lin, and P. Maheshwari, Agent-Based Middleware for Web Service Dynamic Integration on Peer-to-Peer Networks, in AI 2005, LNAI 3809. 2005, Springer: Verlag Berlin Heidelberg. p. 405 – 414, 2005.
- [21] L. Bellissard, N. De Palma, A.Freyssinet, M. Herrmann and S. Lacourte, *An Agent Platform for Reliable Asynchronous Distributed Programming*, IEEE: France, 1999.
- [22] J. Xie and C. Liu, Multi-agent systems and their applications, in Journal of International Council on Electrical Engineering, Volume 7, Issue 1, 2017.
- [23] M. Atif Nazir Raja and H. Farooq Ahmad, SOA Compliant FIPA Agent Communication Language. 2008, IEEE.
- [24] M. Purvis, S. Cranefield, G. Bush and D. Carter, an Agent-Based Distributed Information Systems Architecture, in Proceedings of the 33rd Hawaii International Conference on System Sciences. 2000, IEEE.
- [25] M. W. Khan and E. Abbasi, Differentiating Parameters for Selecting Simple Object Access Protocol (SOAP) vs. Representational State Transfer (REST) Based Architecture, in Journal of Advances in Computer Networks, Vol. 3, No. 1, March 2015.
- [26] K. A. Kadouh, and K. A. Albashiri, Improvement of Data Transfer over Simple Object Access Protocol (SOAP), in World Academy of Science, Engineering and Technology International Journal of Industrial and Manufacturing Engineering Vol:8, No:2, 2014.
- [27] A. Alshammari, A. Abu Bakar and M. Z. Ahmad, Fast dynamic clustering SOAP messages based compression and aggregation model for enhanced performance of Web services. Journal of Network and Computer Applications, January 2014.
- [28] N Nurhayati, HB Suseno, AMR Maulan, Pengembangan Web Service Online Pembayaran Online Menggunakan Simple Object Access Protocol (SOAP) (Study Kasus : Sekolah Dasar Islam Al-Azhar BSD), in Jurnal Pseudocode, 2015.
- [29] Ning Wei, Research on an Information Application Integration Platform Based on SOA and Web Service, International Journal of Multimedia and Ubiquitous Engineering, Vol.10, No.12 (2015), pp.165-174.

- [30] H. Nassiri, M. Machkour and M. Hachimi, Integrating XML and Relational Data, in *Procedia Computer Science*, Volume 110, 2017, Pages 422-427.
- [31] S. Kanimozhi, A. Kannana K.Selvamani and A. VijayKumar, RETRACTED: A Novel Approach to Discover Web Services Using WSDL and UDDI, in *Procedia Computer Science*, Volume 48, 2015, Pages 480-488.
- [32] M.J. Lewis, M. Govindaraju, and K. Chiu, Exploring the Design Space for CCA Framework Interoperability Approaches. 2005.
- [33] X.T. Nguyen, and R. Kowalczyk, WS2JADE: Integrating Web Service with Jade Agents, in *Verlag Berlin Heidelberg 2007*. 2007, Springer: Berlin.
- [34] S. Goel, H. Shada, and D. Taniar, Asynchronous Messaging Using Message-Oriented-Middleware, in *IDEAL 2003*. 2003, Springer: Verlag Berlin Heidelberg. p. 1118-1122.
- [35] Y.-z. Xu, D.-x. Liu, and F. Huang, Design and Implementation of a Workflow-Based Message-Oriented Middleware, in *APWeb Workshops*. 2006, Springer: Verlag Berlin Heidelberg. p. 842 – 845.
- [36] P. Yuan, and H. Jin, A Composite-Event-Based Message-Oriented Middleware, in *GCC 2003, LNCS 3032*. 2004, Springer: Verlag Berlin Heidelberg. p. 700-707.
- [37] S. Goel, H. Sharda, and D. Taniar, Java Message-Oriented-Middleware in a Distributed Environment, in *IICS 2003*. 2003, Springer: Verlag Berlin Heidelberg. p. 93-103.
- [38] P.R. Pietzuch, and S. Bhola, Congestion Control in a Reliable Scalable Message-Oriented Middleware, in *International Federation for Information Processing*. 2003, IFIP USA. p. 202-221.
- [39] S. Parkin, D. Ingham, and G. Morgan, A Message Oriented Middleware Solution Enabling Non-repudiation Evidence Generation for Reliable Web Services, in *ISAS 2007*. 2007, Springer: Verlag Berlin Heidelberg p. 9 – 19.
- [40] S. Ahn, and K. Chong, A Case Study on Message-Oriented Middleware for Heterogeneous Sensor Networks, in *IFIP International Federation for Information Processing*. 2006, IFIP. p. 945 – 955,
- [41] Y.J. Yoon, K.H. Choi, and D.R. Shin, Design and Implementation of Communication System among Heterogeneous Multi-Agent System, in *Fourth International Conference on Networked Computing and Advanced Information Management*. IEEE, 2008.
- [42] K. Avila, P. Sanmartin, D. Jabba and M. Jimeno, Applications Based on Service-Oriented Architecture (SOA), in *Proceedings of the International Conference on Computers in Education*. Sensors 2017, 17, 1703; doi: 10.3390/s17081703.

