RESEARCH ARTICLE OPEN ACCESS

Service Oriented Architecture for Web Service Based Applications

Sirisha K L S [1], V.Santosh Kumar [2]
Associate Professor
Department of Computer Science and Engineering
Keshav Memorial Institute of Technology [1].
Sreyas Institute of Engineering and Technology [2]
Hyderabad - India

ABSTRACT

Web service is the technology that is related to the concept of distributed computing. With the emergence businesses that collaborate each other, distributed computing has become an important phenomenon where organizations can have applications that work together irrespective of their platforms. This has led to a distributed architecture known as Service Oriented Architecture (SOA). With the emergence of SOA, many applications in the real world are realized for rendering services. Services are given importance rather than individual applications. With SOA applications of different companies are integrated to have a seamless service to be rendered to end users. In this paper, we provide the insights on the technologies behind machine to machine integration and their applications in the real world. It throws light into different application scenarios and their utility in the real world.

Keywords **:–** Distributed computing, Service Oriented Architecture (SOA), Web Services, distributed applications

I. INTRODUCTION

Computing has been around for many decades. Many types of computing evolved. They include standalone, client server, centralized and distributed computing technologies. Standalone computing refers to the computing that occurs in the local machine. It has, generally, no computing connectivity with other computers. Due to the emergence of networking technologies and hardware, we moved to other computing models such as centralized, client/server and distributed technologies. As shown in Figure 1, centralized computing came into existence. It contains only one server and many terminals. Terminal is a machine that has no processing power of its own. All terminals are dumb terminals that are used for participating in computing. As the terminals are dumb in nature, it reduces hardware cost and improves security. However, it causes issues with overloading of server as server needs to do everything.

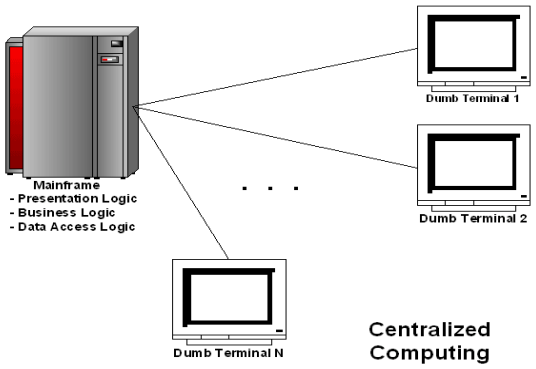

Figure 1**:** Centralized computing

As presented in Figure 1, it is evident that dumb terminals are connected to server and centralized computing and centralized storage, centralized security is maintained in the server. As hardware prices came down, later on, client/server computing came into existence. It is the networking architecture in which client and server machines are involved. Client is the machine here which has client program which makes requests to server program. Server is the machine in which server operating system exists. The server and client both are having computing capabilities. The burden of server is shared by clients in terms of storage and computing locally. That is the reason why client/server computing uses smart terminals. It is illustrated in Figure 2.

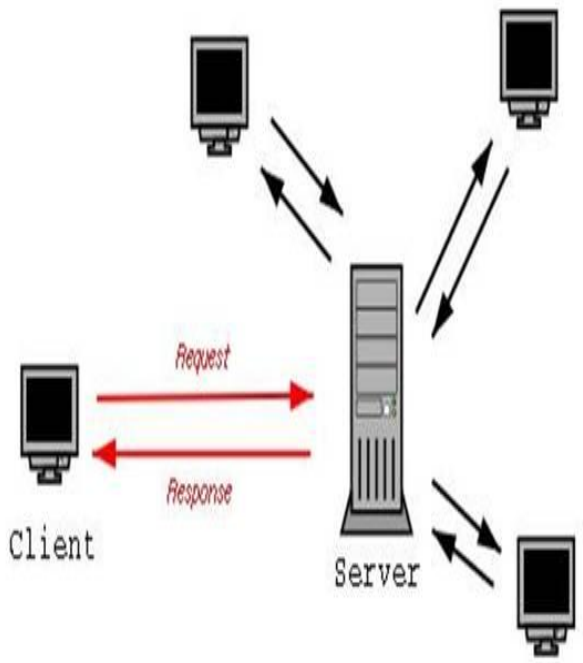

Figure 2: Illustrates client/server

As presented in Figure 2, it is evident that multiple terminals are connected to server. The client machines are able to interact with server. Clients are smart terminals. The client machine makes request to server and server gives response. This kind of computing has many advantages. Server is not overloaded as the clients can have their own storage. Clients can relieve server from storage burden. It provides sufficient security as well as the clients cannot connect to server with due authentication. Having understood the centralized computing and client-server computing, it is the time now to understand distributed computing.

This paper throws light into distributed computing, service oriented architecture, and applications that are based on SOA. The remainder of the paper is structured as follows. Section 2 provides related work. Section 3 presents distributed computing. Section 4 provides web services technology. Section 5 presented applications in distributed technologies and SOA. Section 5 concludes the paper and gives directions for future work.

II.RELATED WORKS

This section provides review of literature related to web services and distributed computing besides SOA. Valls et al. [1] explored web services architecture for creation, testing and reconfiguration of test cases for distributed applications. Al-Khanjari et al. [2] studied SOA for mobile applications in distributed environments. In [3] SOA is studied which is cloud based and used for e-governance. In [4] SOA is used to study the feasibility for games that are operated by individuals across the globe. On the other hand in [5] Geographical Information System (GIS) is studied for making them based on SOA. Nazim et al. [6] studied web of thing in terms of embedded resources and their dynamic management. In [7] Mobile Cloud Computing (MCC) and SOA are combined to have a seamless integration of web services with cloud computing. In [8] Model Driven Development (MDD) is used for SOA and market oriented architecture applications. In [9] SOA is explored for e-Commerce domain where distributed programming and distributed application scenarios are important. In [10] smart cities are studied using SOA.

In [11], different kinds of ways used to build web services are explored. Especially two important means include SOAB based and RESTful web services. The former is used based on SOAP while latter uses different technology known as RESTful which is based on the URLs in order to consume web services in distributed applications. Web services discovery and reputation based approach are studied in [12]. Testing of web services that are composed into different applications is the main focus of [13]. They used composed web services to form a flow of services and then tested them. There are anti-patterns in distributed environments. Such patterns are studied in [14] for understanding the dynamics of service composition. Web service aggregation pattern is studied in [15] for integrating enterprise applications built in different platforms. Another distributed computing technology such as Common Object Request Broker Architecture (CORBA) is compared with web services in [16].

For industrial applications, knowledge based integration of web services is explored in [17]. Component based and reconfigurable middle-ware components are understood and explored in [18]. This is a signifying research idea to have components and reuse them instead of reinventing the wheel. Many applications that are based on SOA are analyzed in [19] for giving useful insights. SOA based applications in distributed environments are widely used in the real world. Service oriented middleware platform that is device level and self-manageable is explored in [20] for distributed energy resources. In this paper, web services that can be used to realize SOA is given importance with technology description and use cases.

III.DISTRIBUTED COMPUTING

It is the technology which is currently being used by applications in the real world. It is the computing phenomenon which needs multiple servers to work together. Many servers

geographically located in different places can participate in computing. This kind of technology can help in leveraging computational power that is required to process huge amount of data in short span of time. Moreover, it brings about many advantages such as fault tolerance, availability, scalability and modular approaches. It helps in building load balanced distributed applications as well. Load balancing is one of the features of distributed computing. The architecture appears as shown in Figure 3.

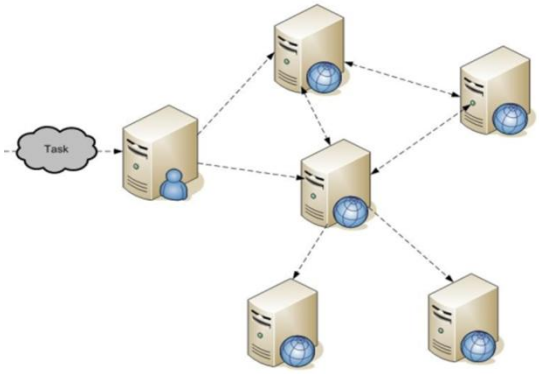

Figure 3: Distributed computing

As illustrate in Figure 3, it is evident that many servers across the globe are involved in distributed computing. Distributed computing is therefore a server side phenomenon which enables computational power to be spread across multiple machines in the world. Cloud computing and grid computing are based on distributed computing. Many platforms in the real world came into existence for realizing distributed computing. With respect to Java it is Remote Method Invocation (RMI) as shown in Figure 4 where communication takes place between objects running in different JVMs. The JVMs typically run in remote machines. This technology enables diversified applications to have seamless integration. At the same time, this technology enables component development that can help in promoting reusability instead of reinventing the wheel gain.

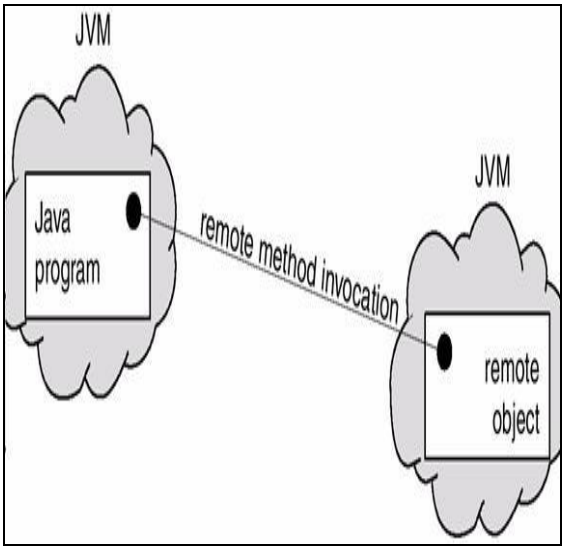

Figure **4:** Distributed computing technology with Java

As shown in Figure 4, it is evident that two programs running in different machines are able to interact with each other with technology known as RMI. RMI is the distributed computing technology that enables Java applications to have interaction even when they are running in different machines. RMI enables applications that are developed in Java platform to have seamless integration. Many real time scenarios like net banking, insurance, e-Commerce etc. need this kind of technology. However, RMI supports homogenous applications to get connected. But the need of the hour is to go with heterogeneous applications and integrate them. Towards this end, a distributed technology that is language independent is Web Services. The subsequent section throws light into web services and its utility in realizing distributed applications.

IV. WEB SERVICES

Web services as explored in [2], [4], [7], [10] are widely used in research and academics. Web services are the technology that helps in developing distributed and heterogeneous applications. Such applications can integrate businesses to form chain of businesses. Many use cases such as e-Commerce are found on top of web services. Distributed computing with heterogeneous environment is possible with web services. The technology is used to build applications that make use of XML based standards like Simple Object Access Protocol (SOAP), Web Service Description Language (WSDL), and Universal Description, Discovery, and Integration (UDDI). These standards help in building and consuming web services. They are used to realize SOA.

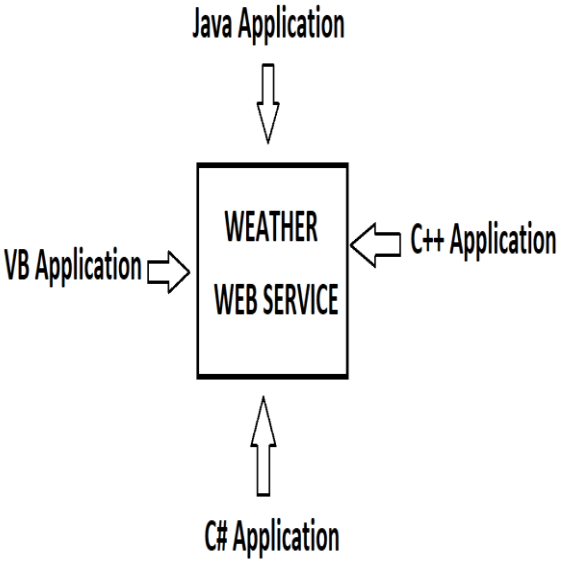

Figure 5**:** Illustrates an interoperable web service

As presented in Figure 5, it is evident that a web application can make use of web services. However, web services may be developed by third parties. Applications written in different languages can invoke web service irrespective of the platform in which web service is developed. A web service developed in one language can be called from a program written in any language making it interoperable. Thus web services can be reused and web services can be used to integrate heterogeneous applications. Figure 6 illustrates how interoperability works out when web services are used in distributed applications.

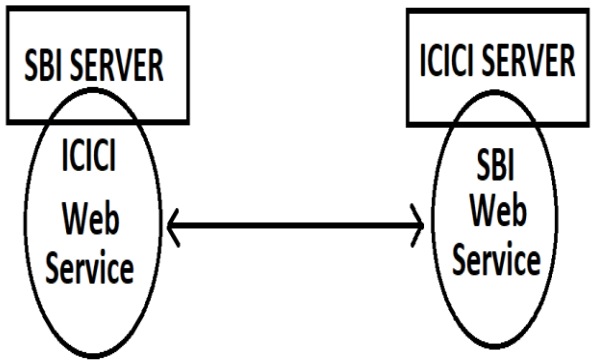

Figure 6: Web services and their utility in integrating heterogeneous applications

As shown in Figure 6, it is evident that a web service provided by ICICI bank can be called by a program of SBI. This is possible with web services. In the same fashion, the web service provided by SBI can be invoked by ICICI applications. The two companies might be using different application development platform. Irrespective of the platform the true interoperability of applications in distributed environment is made possible with web services.

V. APPLICATIONS THAT NEED SERVICE ORIENTED ARCHITECTURE

Here are two case studies or use cases that needs SOA and realized using web services technology. The first use case is related to banking sector. With the introduction of distributed computing in banking sector, banking transactions that involve multiple branches with heterogeneity is made possible. A person having account in one bank can use ATM card in ATM of different bank. This is illustrated in Figure 7. Since banking sector needs dynamic means for transferring amount from one bank to another without physically moving to bank, the web applications provided by banks are able to support distributed transactions where two-phase commit is used for successful transactions. A transaction contains two or more operations that need to be executed as a single unit. When such transactions are made in business or financial outfits, it is important to have SOA approach. Web services can be used to realize the requirements of SOA.

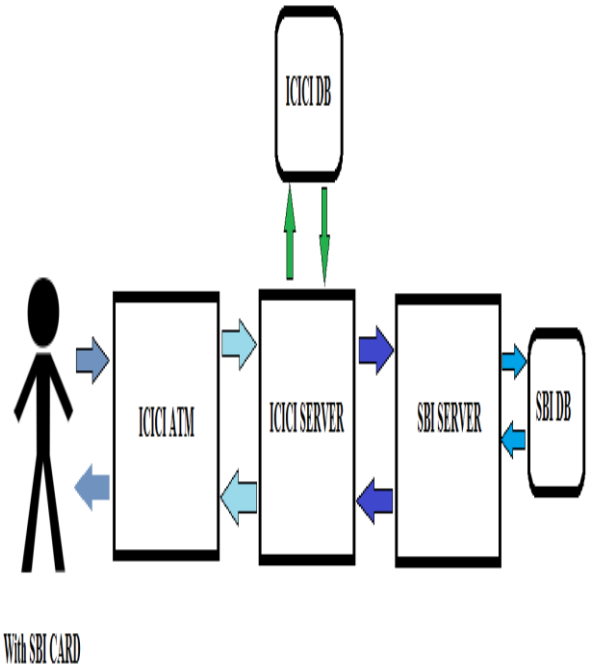

Figure 7: Use case for distributed application with SOA in banking sector

As shown in Figure 7, it is evident that a person with SBI card approaches ATM of ICICI. The customer has ATM of SBI and trying to withdraw money from ICICI ATM. This is very interesting scenario in which it is important to understand technology behind this. When ICICI bank ATM is used with ATM debit card, the ATM client program interacts with ICICI server and then ICICI server needs to interact with SBI server with machine-to-machine interaction phenomenon. Otherwise, it is not possible to have distributed environments. The SBI server then does needful activity of checking its database to know whether customer has sufficient balance before making well informed decisions. The SBI server provides money to ICICI bank server so as to enable the customer to draw money from ICICI bank ATM. This is the typical scenario or use case for SOA which demonstrates the need for web services as the parties involved in the system are using different platforms.

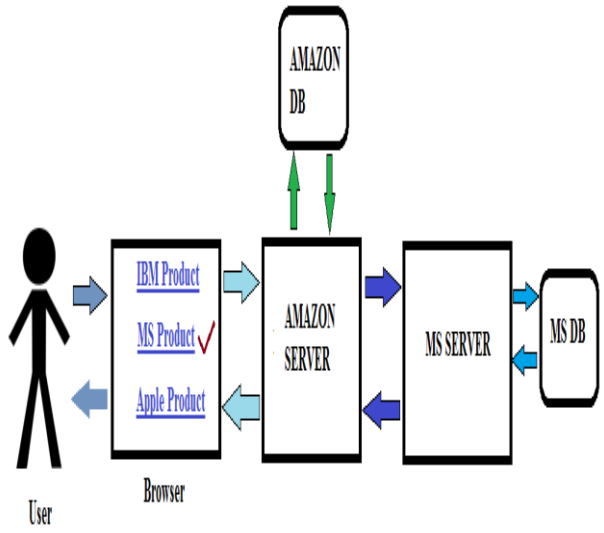

Figure 8: Use case for distributed application with SOA in e-Commerce sector

As shown in Figure 8, it is evident that e-Commerce scenario needs multiple players to work in distributed environment. Different manufacturers, stockist etc. need to have a unified platform for making their online transaction. When user selects products of specific type such as MS product, the browser makes request to Amazon server as the request is made from Amazon web site. Once the product option is chosen, actual product details come from third party Company. When MS product is selected, the request goes to Amazon web server. Then it is forwarded to Microsoft server from where it needs to take data. The Amazon server cannot make a direct connection to MS server. The Amazon server when connected to MS server, the connectivity needs to be done with web services technology. This is the reason why it is an example where SOA is needed. The processing of request is made in distributed environment.

VI. CONCLUSIONS AND FUTURE WORK

In this paper we studied different types of computing and focused on the need for distributed computing and Service Oriented Architecture (SOA). The realization of distributed computing is made with many technologies. We explored web services as the technology as part of distributed computing for realizing applications that are based on SOA. Such applications need specific architecture for providing desired services. Two real time scenarios are provided for SOA applications. Web services are the technology used to build applications that can work together. It is also used to build components that can be reused by many applications. Web services technology has made cloud computing and grid computing possible. Amazon web service is the cloud platform that is based on web services. There is growing need for web services API in the real world. In future we focus on using web services and testing the applications for improving quality of service oriented applications.

REFERENCES

[1] Marisol García Valls,Iago Rodríguez López, and Laura Fernández Villar. (2013). iLAND: An Enhanced Middleware for Real-Time Reconfiguration of Service Oriented Distributed Real-Time Systems. *IEEE*. 9 , p228-235.

[2] Z. Al-Khanjari, Z. Al-Kindi, E. Al-Kindi and N. Kraiem. (2015). Developing Educational Mobile Application Architecture using SOA. *IJMUE*. 9 , p247-254.

[3] Abdulelah Awadh Al-Rashedi. (2014). E-Government Based on Cloud Computing and Service- Oriented Architecture. *IJCEE*. 6 , p201-206.

[4] Maira B. Carvalhoa, Francesco Bellottia, Riccardo Berta, Alessandro De Gloria, Giorgia Gazzarata, Jun Hub and Michael Kickmeier-Rus. (2015). A case study on Service-Oriented Architecture for Serious Games. *Entertainment Computing*, p1-22.

[5] Mortaza Saleh, Tahere Yaghoobi and Ahmad Faraahi. (2012). SUITABILITY OF SERVICE ORIENTED ARCHITECTURE FOR SOLVING GIS PROBLEMS. *IJAIT*. 2 , p1-11

[6] Muhammad Nazim, Munam Ali Shah, Muhammad Kamran Abbasi. (2015). Analysis

of Embedded Web Resources in Web of Things. *Proceedings of the IOARP International Conference on Communication and Networks*, p22-30.

[7] Saeid Abolfazli, Zohreh Sanaei, Muhammad Shiraz and Abdullah Gani. (2012). MOMCC: Market-Oriented Architecture for Mobile Cloud Computing Based on Service Oriented Architecture. *IEEE*, p8-13

[8] David Ameller, Xavier Burgu´es and Oriol Collell, Dolors Costala, Xavier Franch, Mike P. Papazoglou. (2015). Development of Service-Oriented Architectures using Model-Driven Development: A Mapping Study. *Information and Software Technology*, p1-49

[9] Fabian Aulkemeier, Milan Schramm, Maria-Eugenia Iacob, and Jos van Hillegersberg. (2016). A Service-Oriented E-Commerce Reference Architecture. *IJATER*. P26-45.

[10] S. J. Clement, D. W. McKee AND Jie Xu. (2017). Service-Oriented Reference Architecture for Smart Cities. *IEEE*, p81-85.

[11] Kishor Wagh And Ravindra Thool. (2012). A Comparative Study of SOAP Vs REST Web Services Provisioning Techniques for Mobile Host. *Journal of Information Engineering and Applications*. 2 , p12-16.

[12] Emad Elabd. (2015). A Dynamic Reputation-Based Approach for Web Services Discovery. *I.J. Information Technology and Computer Science*, p31-36.

[13] Chetan Phalke and Archana Jadhav. (2017). A Technique for Testing Composed Web Services including Footprint. *International Research Journal of Engineering and Technology*. 4 , p820-823.

[14] Ali Ouni, Marouane Kessentini, Katsuro Inoue and Mel ´O Cinn´eide. (2015). Search-based Web Service Antipatterns Detection. *IEEE*. p1-14.

[15] Nilesh Vishwasrao Patil and M. C. Kshirsagar. (2015). Enterprise Application Integration using Service Oriented Architecture with Web Service Aggregation Pattern. *International Journal of Current Engineering and Technology*. 5, p1-8.

[16] Hayyan R. Sheikh. (2012). Comparing CORBA and Web-Services in view of a Service Oriented Architecture. *International Journal of Computer Applications*. 39 , p47-55.

[17] Borja Ramis, Luis Gonzalez, Sergii Iarovyi, Andrei Lobov, José L. Martinez Lastra and Valeriy Vyatkin William Dai. (2014). Knowledge-based web service integration for industrial automation. *IEEE*, p733-739.

[18] Lionel Seinturier, Philippe Merle, Romain Rouvoy, Daniel Romero, Valerio Schiavoni and Jean-Bernard Stefani. (2012). A component-based middleware platform for reconfigurable service-oriented architectures. *Softw. Pract*. 42 (.), p559–583.

[19] Karen Avila , Paul Sanmartin ,Daladier Jabba and Miguel Jimeno. (2017). Applications Based on Service-Oriented Architecture (SOA) in the Field of Home Healthcare. *Sensors*, p1-16.

[20] Stjepan Sucic, Juraj George Havelka and Tomislav Dragicevic´ . (2014). A device-level service-oriented middleware platform for self-manageable DC microgrid applications utilizing semantic-enabled distributed energy resources. *Elsevier*. 54 , p576–588.