

An Appraisal of Web Applications vs. Web Services with respect to Performance Engineering using Software Performance Engineering Approach

Ch. Ram Mohan Reddy
Research scholar
M S Ramaiah Institute of
Technology, Bangalore – 54
India

D. Evangelin Geetha
M S Ramaiah Institute of
Technology,
Bangalore – 54 India

T. V. Suresh Kumar
M S Ramaiah Institute of
Technology,
Bangalore – 54
India

ABSTRACT

Web Applications (WA) and Web Services (WS) are emerging as predominating technologies ineffectively usefulness in distributed/service oriented systems. These systems present the challenges for achieving the quality of service (QoS). The Quality of Service can be measured by how well the systems are responding to the customers. The performance of these systems from the user point of view is to be estimated, and appropriate measures have to be taken if the objectives are not being met. At the foremost, a feasibility study needs to be done and correlated to technical and environmental factors. The present paper explains how to consider these factors in assessing the performance of different categories of WA vs. WS. It is illustrated by modeling deployment diagram for web application and web services and also by presenting macro and micro flow diagrams.

Keywords

Web Services, Web Applications, Technical Factors, UML, Software Performance Engineering

1. INTRODUCTION

With the increase in the internet usability and availability on any device, everybody can access the internet with the help of devices ranging from mobiles to computers. Anybody who has a basic smart mobile device can access the internet with ease. This is the reason many companies/businesses try to develop web based applications which will increase the usability of the application by a wide spectrum of people across the globe. A company that has a solution for a problem prefers to develop it as a Web-based application or simply Web Application. These application programs are very easy to develop as there are many frameworks that help in the development of a Web Application within no time. Moreover, these Web Applications are user friendly.

Some services are used for general purposes by all the Web Applications. These specifications help in the incorporation of setting certain standards for some web services. These services are general and provide the information about the network being used and different protocols used in the communication process when the user tries to access a Web Application. The components described are operated out using the standards like Web Service Description Language (WSDL), Simple Object Access Protocol API (SOAP) and Universal Description, Discovery and Integration API (UDDI) respectively [10]. The Web Application is for the user so it should have a Graphical User Interface that helps the user in effectively interacting with the software and it also has to have an interface for the web service according to the standards so that the web service provided can help the Web

Application work properly as was intended. Figure 1 shows the intricate working of a Web Application and a Web Service using the specified interfaces.

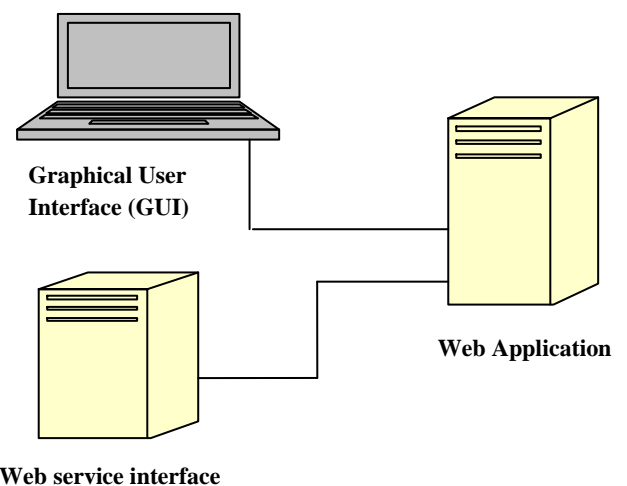


Fig. 1 General view of Web service and Web application

The three roles that interact with each other to form the basis of a Web Service Architecture are shown in Figure 2. These three roles are Service providers, Service registry, and Service requestor.

The Web Applications have software modules with some functionality to support the overall application. These functionalities have to be registered in the service registry of a web service. The service requestor issues the find operation, to the service registry that contains information about the service needed in the service description. The internet is searched for a service provider that provides the same or similar service using the publish operation. These services are the certain general type of services which are identified for all applications trying to incorporate similar services. If a service provider is found, then the developer who needs the web service is bound to the service provider using the bind operation, both exhibiting similar characteristics as they are logical constructs for the same service.

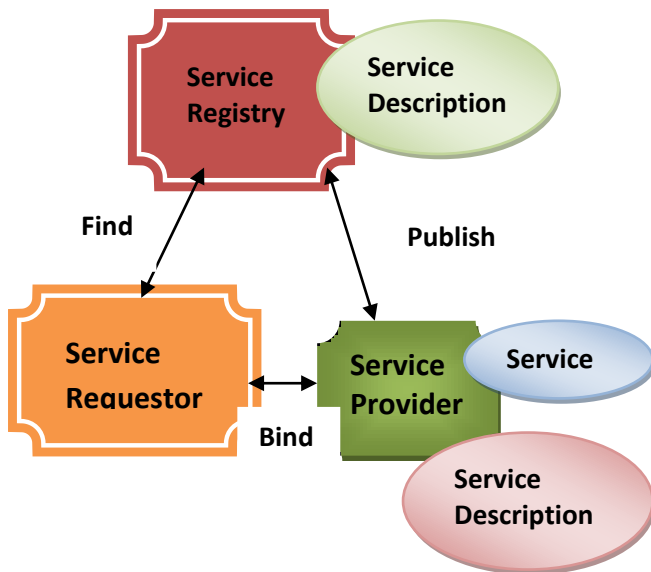


Fig. 2 Web Services Architecture

1.1 Web Applications workflow

Web applications developed nowadays are split into smaller components where each component is independent and provides a well-defined function. Web applications have become popular because of their ready usability over a range of devices.

The Web Application sends a request to the Web Server over a firewall through the Internet or the Intranet to utilize a Web Service for its proper working. This request may or may not go through firewalls as many Intranets do not have them. The request from the web browser is examined by the web server and response is sent back to the web application. Here again, we may have a different set of firewalls. Many web applications also use middleware packages to connect to various components. The web application collects the users request from the browser, understands the requirements of the user, sends the request to the web server by the user's request, gets the response and displays the information back in a proper format for the user. Figure 3 presents the three-layered web application model. The first layer is the user interface or web browser and the dynamic generation is the second, and the database is the third layer, which contains customer data.

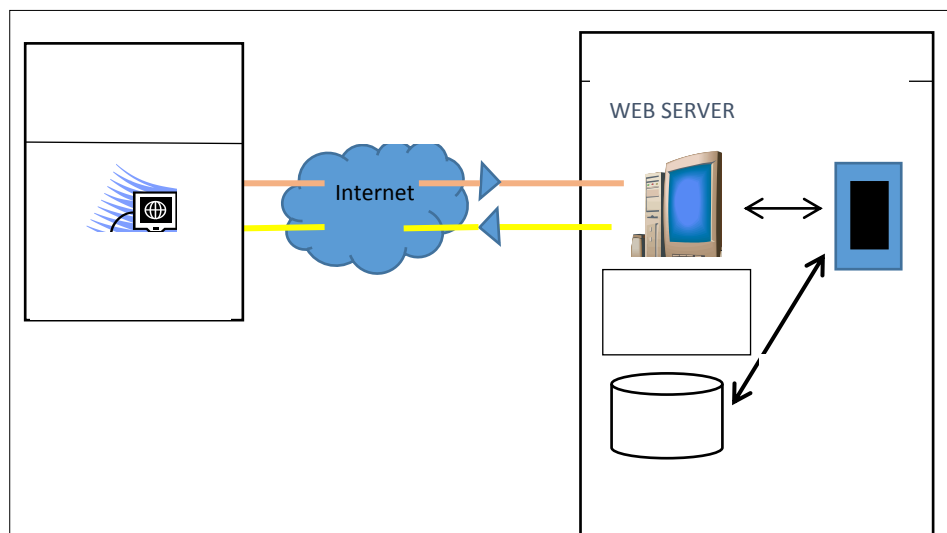


Fig. 3 Web Application Work Flow

1.2 Web Services workflow

Web Service is often described as “a software application identified by a Uniform Resource Identifier (URI), whose interfaces and bindings are capable of being defined, described, and discovered as XML artifacts. A web service supports direct interactions with other software agents using XML-based messages exchanged via Internet-based protocols” [16]. Web Services elements are SOAP, WSDL,

and UDDI. By using SOAP protocol, Web services service provider registers itself with the UDDI directory. The service requestor application queried the Directorate to locate service providers. The service requestor now looks at the WSDL document published by the provider to find out how its web services works. The service requestor now calls the web service which in turn sends the results back. SOAP protocol is used for communication. The workflow of services is shown in Figure 4.

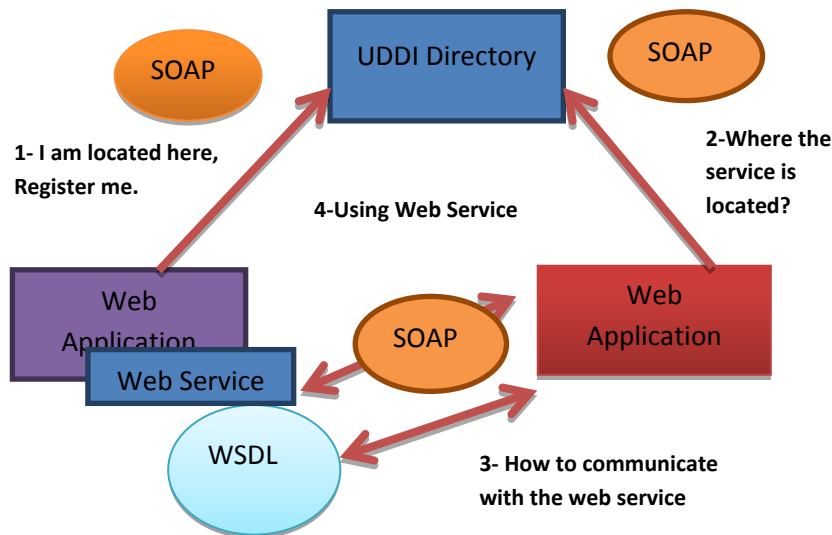


Fig. 4 Web Services Work Flow

The rest of this paper is organized as follows. Related work is discussed in Section 2. Section 3 represents the concepts of high level architecture of web applications & web services. In Section 4, we discussed the differences between web applications and web services. We discuss the actors and technical factors applicable for web applications and web services in Section 5. Section 6 is illustrated with the case study explaining the technical factors applicable for web applications and web services. Finally, the paper is concluded along with future work in Section 7.

2. RELATED WORK

The general design and structure of any Web Application would be like the client – server technology. It means that all the data is stored on the server and is accessed using a browser with an interface or web pages (presentation layer) which work on the client system. The server is accessed by the client using a set of different protocols some being HTTP protocols. A generalized mechanism is defining the security and encryption mechanisms also called as GET or POST methods are also adopted for the same interaction between the client and the server. A web application is used by a human and has to have a GUI user interface to communicate. The human need not be aware of the complexities happening behind the system or, in other words, the technical aspect of how the web services are being utilized in the Web Application he has been accessing for his personal purposes. A web service hence doesn't need to have a graphical interface as it is accessed by a well-knowledge person who can utilize it in his web application. Though the web services need not always have a GUI, it has to be more appropriately developed as all the error handling is not done mostly by a human being. Hence, a whole new set of acronyms like SOAP, UDDI, etc. were developed to automate the troubleshooting aspects of a web service.

As part of a literature survey, we discuss few significant literature related to web applications and web services.

WA2WS (Web Applications to Web Services) framework which constructs the web services from the web application in two phases is proposed in [4]. Using domain ontology, UML conceptual schema is extracted for web application in the first abstraction phase for which a Java code is generated in the second implementation phase. This approach uses reverse engineering technique, first to convert data-driven web application to a high level of abstraction for which mapping

rules are applied to create web service from functionalities of initial web application and forward engineering approach later.

With the development of growing interest in number of web services, it is required to map the services at early stages of development, i.e., analysis and design phase. It is interesting to use UML in various stages of the web services development process. In addition, the non-functional requirements also can be modeled using UML. [5] proposed the study and classify various UML based approaches that are used to model web services.

Modernization of web-based systems to more elaborate services, business services and their interactions is the important step when migrating from the traditional legacy WA to WS. It is useful in the modern world of complex inter business services and interactions. A semi-automated approach which migrates legacy web applications to web services based Service Oriented Architecture (SOA) application by presenting the simpler version of the Moodle login service is described in [1]. It is based on three goals which are user interface plasticity, code refactoring, and load balancing. It is a technique to develop efficient, module-based web application.

An integration approach which includes of Converting Hyper Text Markup Language (HTML) response documents to XML documents in exploiting the interface of a web application using a technology known as wrapper technology which extracts the appropriate information from the HTML document and stores it in XML format which can be automatically treated as and when required is proposed in [24].

A web service based framework called Component Based Software Engineering (CBSE) is discussed in [18]. It also uses Software as a Service (SaaS) model, TIA technologies and web service architecture which are a composition of various web components.

A black box migration approach which uses black box reverse engineering technique provided for the toolkit that generates wrappers in a semi-automatic and efficient manner. To migrate from existing functionality web application to web services for modeling the web application interface using black box reverse engineering techniques is discussed in [14]. This approach thinks over web application as a from-based

system and transforms the user-oriented interface to be programmatic by presenting the predictable framework of wrapper, relocation platform, migration process and toolkit.

Performance is one of the most important QoS attributes. It is important to consider the factors involved in the efficient performance of different services that contribute to client satisfaction. [9] proposed the literature on performance and scalability attributes related to QoS. This paper also discussed the quality attributes related to web services and web applications. A web service composition system based on Relaxable QoS based service selection algorithm which presents flexibility and performance of the algorithm is proposed in [23]. This system is compatible with WS-BEPL standards which are used to develop composite web application development using intuitive graphical interface. This algorithm helps in fulfilling functional and nonfunctional requirements and also satisfies the non-relaxable QoS constraints having a lower failure rate in finding the feasible solution.

The performance of any Web application can be increased by using GZIP for HTTP Conversion, avoiding redirects, minimizing the use of Iframes, minimize Cookies/CSS size, using link tags, etc. is discussed in [3]. The evaluation of web server based performance on parallel architecture for more reliability by parallel technology is discussed in [21]. The new architecture is deployed for web application server, portal server and databases are also proposed. This architecture reduces CPU load and fastens processor time by reducing delay.

In heterogeneous platforms, the security of a Web Service has to be enhanced. The Web Service has to be properly ensured safety from unwanted theft and data scrambling issues for which a security processing model names JASYPT (Java simplified encryption technique) based on SOAP and authentication is presented in [2]. It provides a framework that manages service security, composition and semantics by making client authentication, signing and encryption of SOAP messages. An approach to Enterprise Application Integration (EAI) using extensible Web services by building a real-world application for EAI with the main purpose of integrating the various web services and assess load balancing routing technique and clustering mechanism is proposed in [19]. It presents the manifestation of web services and functional extensibility in functional services domain. Optimization techniques can be used to increase the performance of a Web Service as proposed in [17]. When multiple messages reach the server from the client, there has to be a mechanism to tackle such situations to improve the server's response time. This is done by effective execution of packing and unpacking of SOAP messages. This approach implements application and protocol processing as separate threads which increase the performance up to 10x.

A page-transition based disintegration and page's access abstraction model with framework extension for service

extraction using a framework named H2W, which constructs the Web service wrapper from multi-pages web applications using simple workflow is discussed in [20]. In this framework, the service provided by Web Service Provider is designed in a generalized format and kept as "libraries" or components. These modules are stacked up and do not require a User Interface. These modules and package structure of a Web Service helps in its effective utilization by a wide variety of Web Applications that needs to incorporate those services into their web applications.

The reason for the web data traffic fluctuation to study the performance of web applications and define the measures of performance indices such as response time, transfer time, TCP connection time, etc. is analyzed in [22]. The study states that the web data fluctuation depends on the characteristics of a web application, the complexity of network path, self-similarity of web traffic and congestion control mechanism in TCP/IP protocol.

Analyzing performance web service that contains intranets and web servers is unique and requires attention Hundreds of millions of online customers are raising. The uncertainty associated with the behavior of web applications and web services makes the issue of workload estimation and planning difficult. Significant concepts that help users, administrators, and managers understand performance problems in web environments. Work load characterization and performance models that take into consideration the unique aspects of Web environments are focused using Technical factors and environmental factors.

In summary, research gaps are identified.

3. BASIC CONCEPTS

Web application is the one that can run on the client's machine accessed through the web browser whereas a system of software, which allows various machines to interact with each other over a network, is a web service. Majority of the times, web services do not necessarily have a user interface as it is used as a component in an application whereas a web application is a complete GUI application. In addition, to communicate or transfer data, web services are used between web applications running on different platforms.

For any analysis of either WA/WS with initial data, we have to know the key characteristics at a high level. There are several works done to represent characteristics of software systems during feasibility or initial inception phase. This leads to interest in research platforms that need to be studied [12] related to their performance. While discussing data of feasibility study, we have to map the elements of low level like SOAP protocol, various network layers to a high level abstraction. High level Web application and web service architecture is given in Figure 5 and Figure 6.

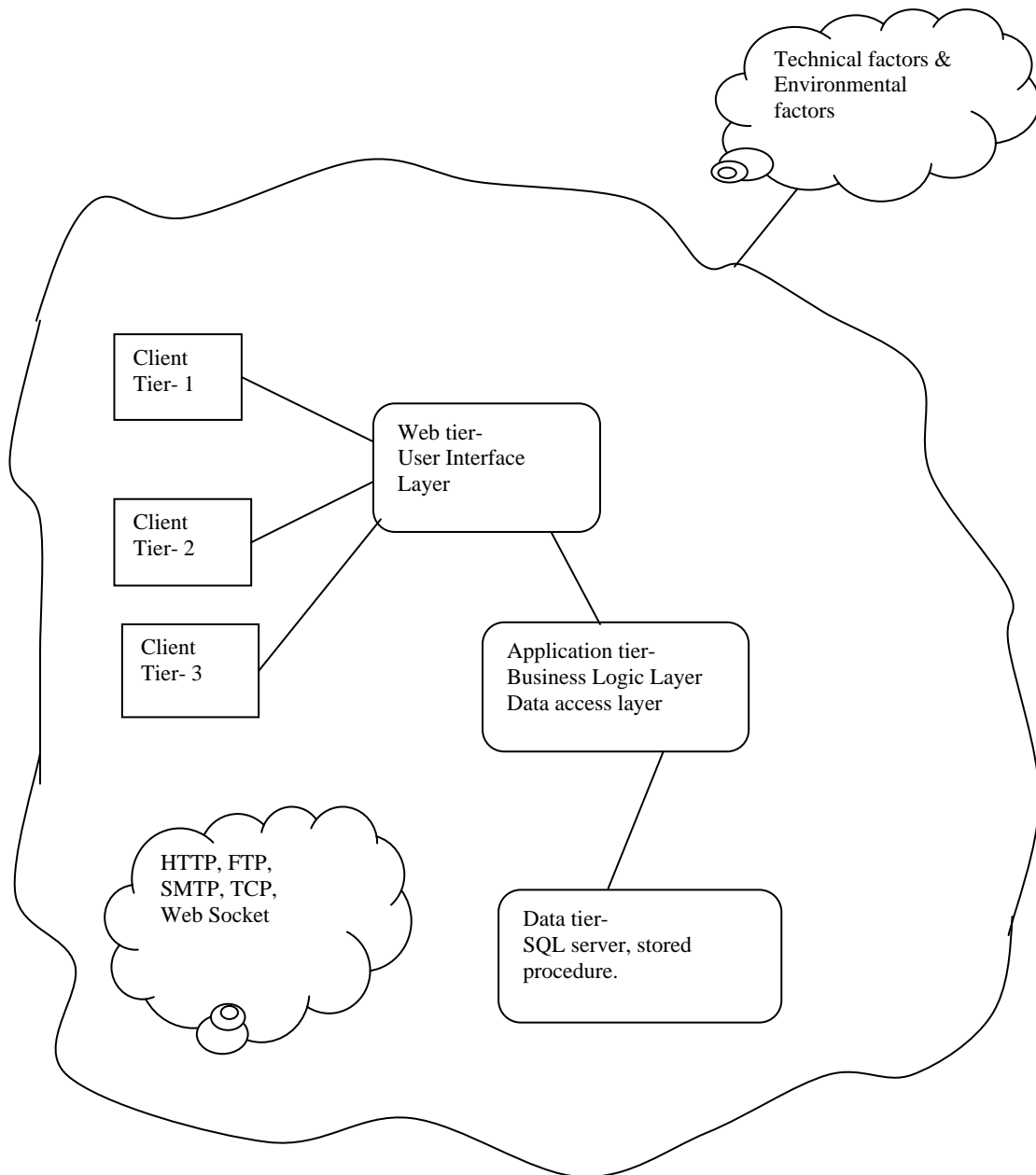


Fig. 5 High Level Representation of Web application Architecture

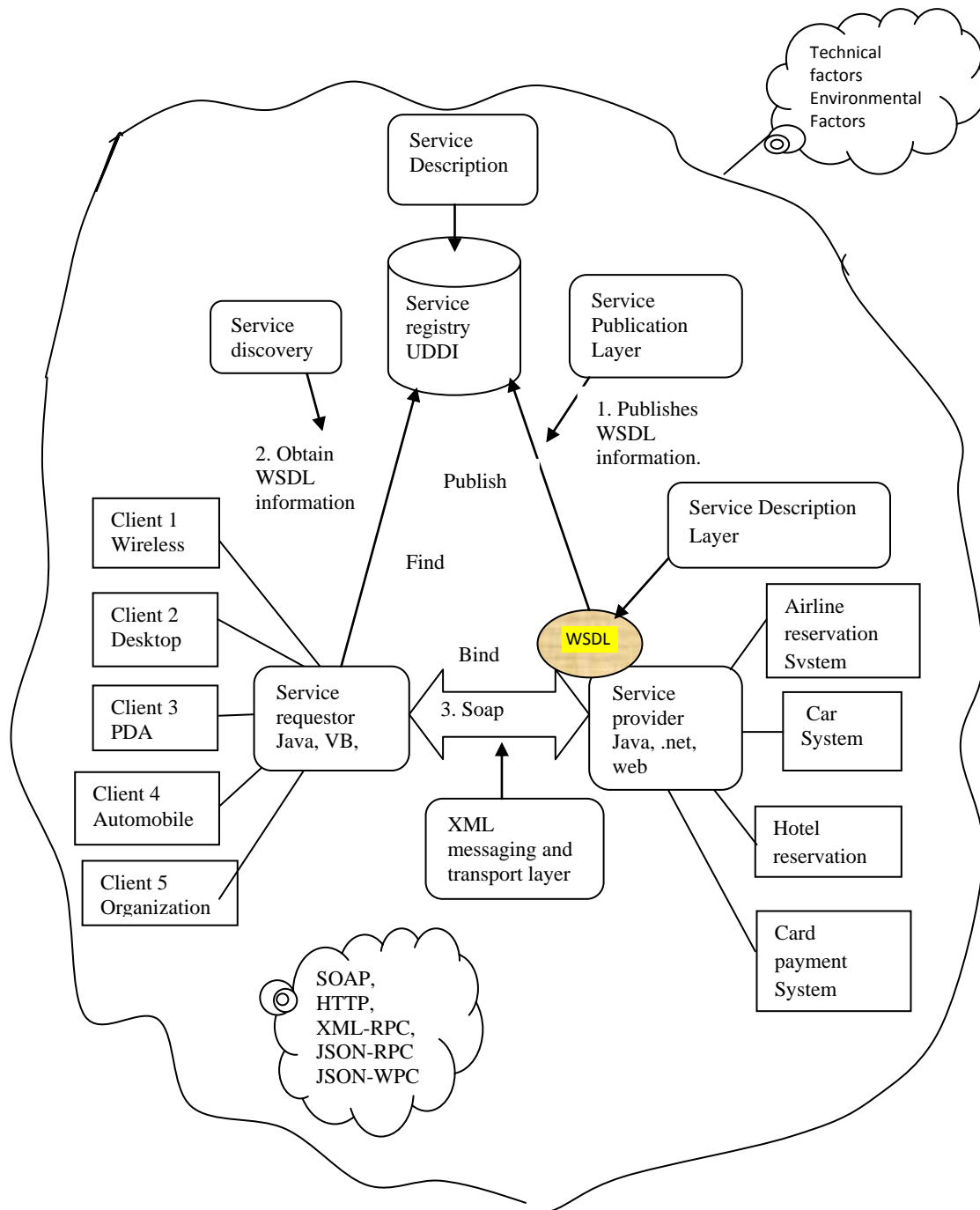


Fig. 6 High Level Representation of Web service Architecture

We have shown in Table 1 mapping of elements of low level architecture elements to high level elements. These factors are termed as technical and environmental factors. Technical factors and Environmental factors are considered as we

explore user based assessment of performance prediction. These factors that are available during feasibility study help in the performance estimation of web applications/web services.

Table 1 Mapping of Low Level Elements into WA/WS high Level Elements

Resource (High Level)	Actor	Technical Factors (Low level)
Client (User interface)	Human User (Complex)	Response, End user efficiency, Reusable code, Easy to install, Easy to use, portable, Easy to change, Special User Training, Facilities are required
Application Tier	Application Programming	Distributed System, Throughput, Complex internal processing, Portable, Concurrent, Includes Security Features

	Interface (Simple)	
Data Tier	Application Programming Interface (Simple)	Distributed System, Throughput, Complex internal processing, Portable, Concurrent, Includes Security Features
Service Requestor	Average	Response, End user efficiency, Reusable code, Easy to install, Easy to use, portable, Easy to change, Special User Training, Facilities are required
Service Provider	Average (System interacting with Protocol)	Technical Factors: Distributed System, Throughput, Complex internal processing, Reusable code, Concurrent, Includes security features, Provides access for third parties
Service Registry	Average	Technical Factors: Distributed System, Throughput, Concurrent, Includes Security Features, Provides access for third parties

The technical factors/environmental factors considered are generally discussed in various software developments at various stages [15]. Throughout our paper, our discussion represents Table1 mapping and high level in nature.

4. WEB APPLICATIONS VS. WEB SERVICES (WA VS. WS)

- The internet is a combination of different types of networks and machines. Our Web Application needs to work on these different networks in the same way despite the fact that it utilizes web services for data transfers and its working. The data present in HTML format in a Web Application while it is in XML format at the Web Services structure.
- Because a Web Service is not always accessed or controlled by human intervention all the time, it has to be designed using specific standards like (WSDL/UDDI). It is based on XML for interoperable applications.
- Web Application is designed using HTML language, and it is a collection of web pages interacting with a database at the backend. A web service, however, is a package collection of modules each designed with a specific purpose to provide a specific service for the better working over heterogeneous networks that too remotely.
- The GUI helps the user to interact with the Web application and the APIs and HTTP protocols help in the internal working of the web services.
- Web Application takes the support of the services of a Web service either for its development or implementation.
- A Web Application has a user interface for the user to interact the software to get a solution to the problem for

which the Web Application is developed. The logistics behind the user interface, i.e., the communication aspects the server and web page, the storing and retrieving of data using protocols, the mechanisms for downloading and uploading scripts to run on the browser like HTTP POST and GET requests are all handled by the Web Services

- Web Services extensively use many open standards like XML, SOAP, WSDL and UDDI to make the Web Application developed on a particular architecture, run on top of many varied types of architectures and machines in the same way as designed.
- Web services do not need a user interface while the most important component of a Web Application is a user interface.
 - Web services are used to help communicate different web applications running on different platforms; web applications use user interfaces to receive information from more than one client running in different applications.
 - Web Service provides the service interface instead of a user interface. The service interface is an XML document that is used to call the web service.
 - Web services work with SOAP protocol, and SOAP protocol is based on XML and HTTP protocol, Since HTTP protocol is stateless protocol any service which is developed on HTTP SOAP protocol is stateless service SOAP protocol is dependent on HTTP.

5. ACTORS AND TECHNICAL FACTORS APPLICABLE FOR WEB APPLICATION AND WEB SERVICES

SPE (Software Performance Engineering) [6], [7] approach helps to assess the performance metrics during the early stages of Software Development Life Cycle (SDLC). SPE requires five parameters, namely: performance goal, workload, software execution structure, execution environment and resource usage. One of the data required for this assessment is the size of the software components. During feasibility study, the information regarding the technical factors and environmental factors [15] are available to the system analysts. These factors are useful to estimate the size of the software components. This size is one of the key parameters and is essentially performance of web service using SPE.

Table 2 Weighting factor for Actor

Category of actor	REPRESENTATION	Weighting Factor
Simple	Application Programming Interface (API)	1
Average	System interacting with protocol (TCP/IP)	2
Complex	Person interacting with GUI or web page	3

Table 3 Unadjusted Weights for Activities

No. of Tables/Classes User Interface (complexity level)	1	2	3 OR MORE
Simple	1	2	3
Average	2	3	4
Complex	3	4	5

Table 4 Technical Factors

Factor	Description	Weight	Rate	Total
T1	Distributed system	2	4	8
T2	Response or throughput performance objectives	2	5	10
T3	End-user efficiency	1	2	2
T4	Complex internal processing	1	3	3
T5	Reusable code	1	2	2
T6	Easy to install	0.5	2	1
T7	Easy to use	0.5	2	1
T8	Portable	2	3	6
T9	Easy to change	1	2	2
T10	Concurrent	1	3	3
T11	Includes security features	1	3	3

T12	Provides access for third parties	1	5	5
T13	Special user training facilities are required	1	1	1
	Total			47

Table 5 Environmental Factors

Factor	DESCRIPTION	WEIGHT	RATE	TOTAL
E ₁	Familiar with Rational Unified Process	1.5	3	4.5
E ₂	Application experience	0.5	3	1.5
E ₃	Object-oriented experience	1	3	3
E ₄	Lead analyst capability	0.5	2	1
E ₅	Motivation	1	3	3
E ₆	Stable requirements	2	3	6
E ₇	Part – time workers	-1	0	0
E ₈	Difficult programming language	2	3	6
	Total			25

A methodology called Activity Based Performance Prediction (ABPP) is used to assess the performance of software systems activity wise using these technical and environmental factors [11] are using SPE. It considers the complexity of users (actors) and the activities of the WA vs. WS in two aspects i) User based ii) high level architecture. The estimation of response time is based on the amount of resources required to execute each activity that can be quantified. The typical weighting factor for the actor [15] and activities is presented in Table 2, Table 3, Table 4 and Table 5 respectively.

The characteristics of web application and web services can be captured by adjusting appropriate rates for these factors while assessing the performance of a given web applications and web services. For any software, the hardware and software resources available in software architecture may differ. The rate of technical factors varies depending on these resources and the category of the software. The deployment diagram of unified Modeling Language (UML) helps to model the location of the hardware as well as the software resources. By using the ‘note’ of UML, the possible technical factors and actors can be represented and attached to the corresponding resources.

The deployment diagram of the web application architecture and web services architecture are presented along with actors and technical factors using the Unified Modeling Language (UML) ‘note’ in Figure 7 and Figure 8 respectively. This ‘note’ will give us more elaborate mapping of low level elements to high level elements

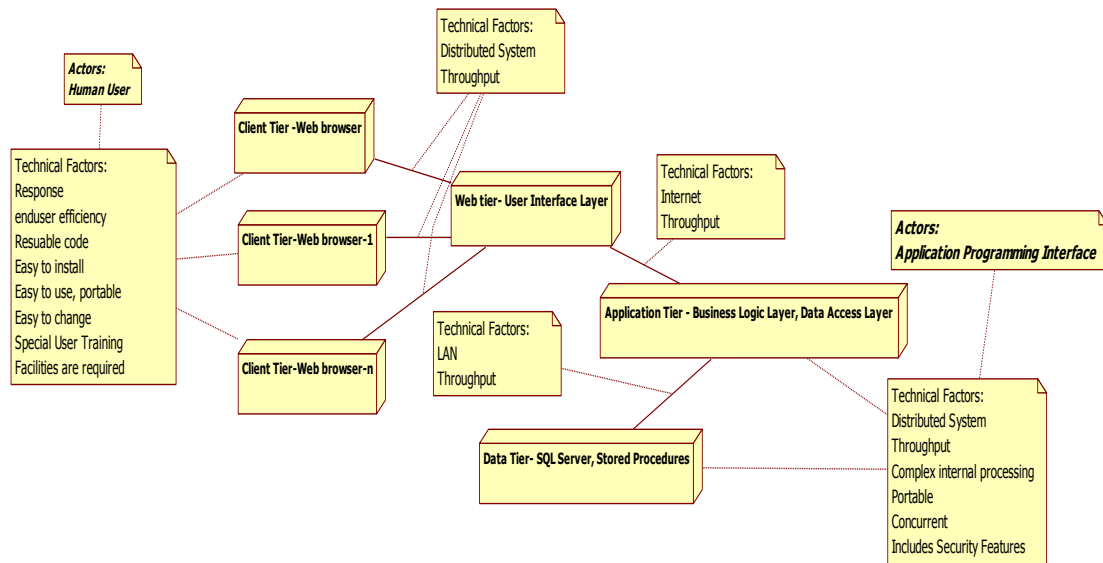


Fig. 7 Web Application Architecture with Technical Factors

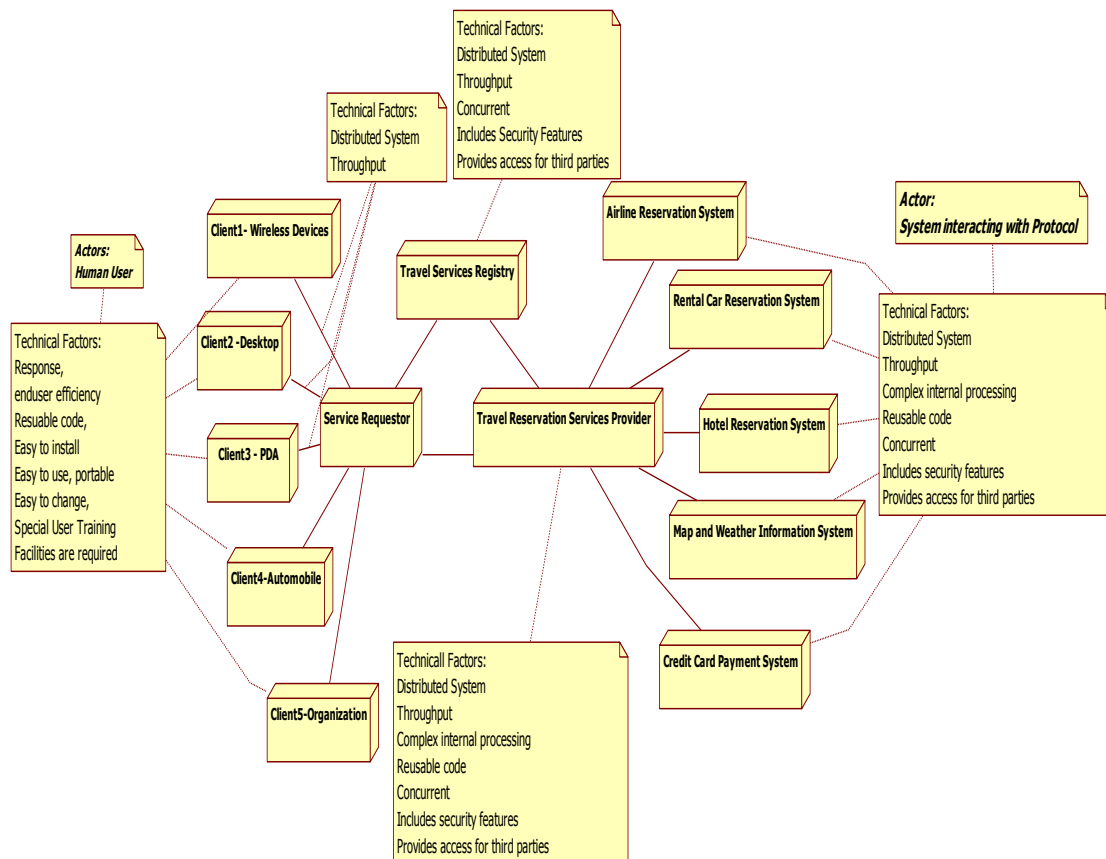


Fig. 8 Web Service Architecture with Technical Factors

In a web application, the application server can be accessed by its own users only. Hence, Application Programming Interface (API) is sufficient for the access. Hence, API is represented as an actor of the application server. We developed a novel algorithm to transform GUI to Activity model [8]. But, in web services architecture, the services can be utilized from any other websites also. Hence, protocols are

required for accessing the web services. The actor of the application server is protocols in case of web services. The detail explanation for the assignment of these factors based on the characteristics and functionalities of web application and web services is explained in the following section by illustrating a case study.

6. CASE STUDY

For airline services, the service requirements include User Management, Reservation Management, and Payment Management. The service functionalities designed in airline system include customer service, Flight service, Reservation service and Payment Service. A Macro Flow diagram opts for web application and Micro Flow diagram for web services which include Check Flight, Book Flight, and Pay Ticket [13].

6.1 Airline System as a Web Application.

Airline system is a web application that provides the users to check for availability of flights as well as booking and cancellation of tickets. If the customer wants to book a ticket or cancel a ticket, the user logs into the application directly. Figure 9 presents the Macro Flow Diagram for Check Flight. This diagram can be extended to other business processes such as Book Flight and pay Ticket.

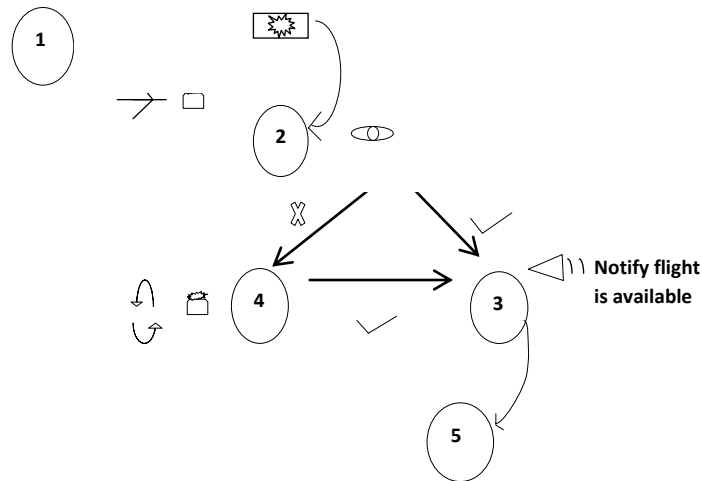


Fig. 9 Macro Flow Diagram: Check Flight

6.2 Travel agent as a web service

Travel Agent is a web service consists of the services airline, hotel, car rental, banking, etc. If a customer wants to book an airline ticket, he or she simply logs into the travel agent web service. It displays the available services. When the customer selects a particular service (airline service), the travel agent connects to the corresponding service provider. The flow of the process is presented as Micro Flow Diagram for Check Flight are in Figure 10. The Micro Flow diagram can be extended to other business processes such as Book Flight and Pay Ticket.

One of the data required for this assessment is the size of the software components. During feasibility study, the information regarding the technical factors and environmental factors are available with the system analysts. The rates assigning for technical factors namely distributed system, response or throughput, performance objectives,

complex internal processing and concurrent processing help to represent the protocols used for communication between the components of the proposed software or between different software systems to be assessed inherently. The weighting factor for Actor indicates the user of the software. The weights for activities and the technical factors such as distributed system, includes security features, provides access to third parties help to represent the category and complexity of the proposed software system. If we consider a web application, the actors may be the users (person interacting with GUI or web page). In the case of web services, the actors will be services provided by the service providers for the travel agent in the given case study. Application programming interfaces help the service providers to communicate with the web services in case of a web service is used.

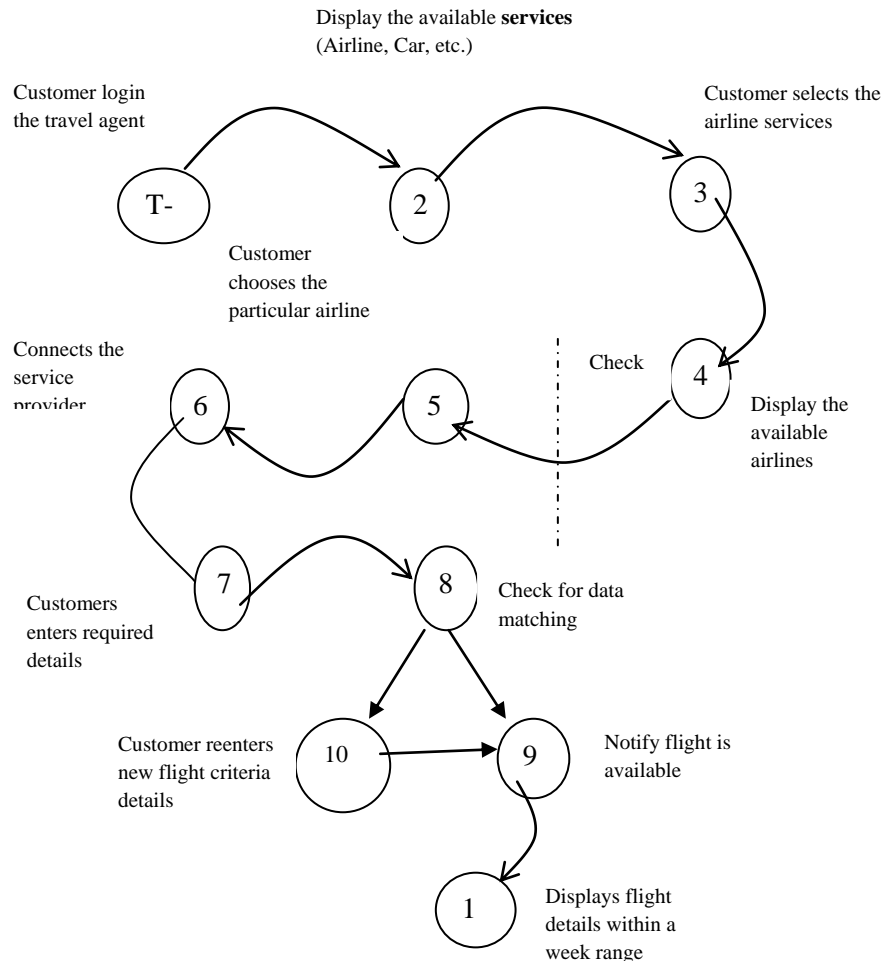


Fig. 10 Micro Flow Diagram: Check Flight

7. CONCLUSION

The software that can access through the web browser is a web application. Web services can be used to communicate or transfer data between web application running on various platforms. In this paper, we have studied various characteristics of web services and web applications. The narrow gaps between web applications and web services are identified. In the case of web applications user interaction place an important role. However, in web services other web applications, will be interacting with the system. The differences are captured by annotating the deployment diagram with 'note' and presented using Macro and Micro flow diagram. We studied WA vs. WS and differences between them with respect to user point of view and web service users/web service developers/web application developers/ web application users view to assess the performance and address the various problems. SPE requires five parameters, namely: performance goal, workload, software execution structure, execution environment and resource usage. One of the data required for this assessment is the size of the software components. We have extended SPE to assess performance. This will help in the fine tuning of deployment diagrams of WA/WS and serve as the guidelines to web service providers as well as web service developers.

8. REFERENCES

- [1] Asil A. Almonaies Manar H. Alalfi James R. Cordy Thomas R.Dean, "Towards a Framework for Migrating Web Applications to Web Services", Proceedings of the 2011 Conference of the Center for Advanced Studies on Collaborative Research, CASCON,2011.
- [2] A.Sudha, K.Vanitha, A.NooralShaba; Effective Method For Secured Soap Based Webservice Interaction Using Jasypt; IRACST – International Journal Of Computer Networks And Wireless Communications (IICNWC), ISSN: 2250-3501 Vol.3, No6, December 2013.
- [3] Balamurugan Subrayen, Gurumoorthi Elangovan, Vasuki Muthusamy, Angayarkanni Anantharajan, "A Case Study for Improving the Performance of Web Application", International Journal of Web Technology Vol 02, Issue 01, June 2013.
- [4] Bouchiha Djelloul, Malki Mimoun, and Mostefai Abd El Kader, "Towards Reengineering Web Applications to Web Services", The International Arab Journal of Information Technology, Vol. 6, No. 4, October 2009.
- [5] Ch Ram Mohan Reddy, RV Raghavendra Rao, D Evangelin Geetha, T V Suresh Kumar, K Rajani Kanth, "Survey on UML based modeling for Web Services" NCSE'14 Conference Proceedings, International Journal of Engineering Research & Technology (IJERT), pp. 57-65, 2014.

- [6] C.U., Smith., “Performance Engineering of Software Systems,” Addison Wesley,1990.
- [7] C.U., Smith and L.G., Williams., Performance Solutions: A Practical Guide to Creating Responsive, Scalable Software. S.I. : Addison-Wesley, 2002.
- [8] Ch Ram Mohan Reddy, D Evangelin Geetha, KG Srinivasa, T V Suresh Kumar, K Rajani Kanth, “Prototype based Performance Prediction of Web Services”, DPPR 2011, CCIS 205, © Springer-Verlag Berlin Heidelberg, CCSEIT 2011, pp. 346–354, 2011.
- [9] Ch Ram Mohan Reddy, Rv Raghavendra Rao, D Evangelin Geetha, T V Suresh Kumar, K Rajani Kanth, “QOS of Web Service: Survey on Performance and Scalability” Natarajan Meghanathan et al. (Eds) : ITCSE, ICDIP, ICAIT, CS & IT-CSCP, pp. 65–73, DOI : 10.5121/csit.2013.390, 2013.
- [10] C. Ferris and J. A. Farrell. “What are web services? Communication”. ACM, 2003, 46(6):31.
- [11] Ch Ram Mohan Reddy, D Evangelin Geetha, T V Suresh Kumar, K Rajani Kanth, “Activity Based Performance Prediction for Software Systems”, Technical Report, Department of Computer Applications, M S Ramaiah Institute of Technology, Bangalore, June 2015.
- [12] Evangelin Geetha D, Suresh Kumar T V and Rajani Kanth K, “Predicting the Software Performance during Feasibility Study”.IET Softw. Volume 5, Issue 2, pp.201–215.doi:10.1049/iet-sen.2010.0075, April 2011.
- [13] Fang-Fang Chua, Syahrul Bin Ngazizan, and Musa Bin Hassan, “Design and Implementation of Airline Reservation Web Services Using Service-Oriented Architecture”, Proceedings of the World Congress on Engineering, Vol 1, WCE, 2010.
- [14] Giusy Di Lorenzo, Anna Rita Fasolino, Lorenzo Melcarne, Porfirio Tramontana, Valeria Vittorini, “Turning Web Applications into Web Services by Wrapping Techniques”,14th working Conference on Reverse Engineering, WCRE, IEEE, 2007.
- [15] Gautam Banerjee, "Use Case Points -An Estimation Approach”, Technical Report, August 2001.
- [16] Heather Kreger, “Web Services Conceptual Architecture (WSCA 1.0)”, May 2001.
- [17] Hao Wang, Yizhu Tong, Hong Liu, Taoying Liu, “Application-aware Interface for SOAP Communication in Web Services”, International Conference on Cluster Computing, 2006.
- [18] J. L. Herrero, F. Lucio, P. Carmona, “Web services and web components”, International Conference on Next Generation Web Services Practices, (NWeSP), IEEE, 2011.
- [19] K. Tharun Kumar Reddy, B. Ramana Reddy, “Integration of Web Services and Routing Techniques for Distributed Applications”, International Journal of Advanced Research in Computer Science and Software Engineering Volume 3, Issue 9, September 2013.
- [20] Michiaki Tatsubori, Kenichi Takahashi, “Decomposition and Abstraction of Web Applications for Web Service Extraction and Composition”, IEEE International Conference on Web Services (ICWS'OG), 2006.
- [21] Nurul Huda Nik Zulkipli, Norazlan Idris, “An Empirical Study on Performance Evaluation of Parallel Architecture for Web Application Services”, Symposium on Computers & Informatics, IEEE, 2013.
- [22] Paolo Cremonesi, Giuseppe Serazzi, “End-to-End Performance of Web service”, Performance Evaluation of Complex Systems: Techniques and Tools, Volume 2459 of the series Lecture Notes in Computer Science pp 158-178,2002.
- [23] Ruey-Kai Sheu, Win-Tsung Lo, Chia-Feng Lin, Shyan-Ming Yuan, “Design and Implementation of A Relaxable Web Service Composition System”, International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, IEEE, 2010.
- [24] Robert B., Georg G., Marcus H., and Wolfgang S., “Interactively Adding Web Service Interfaces to Existing Web Applications,” in Proceedings of Symposium on Applications and the Internet (SAINT'04), California, pp. 74-74, 2004.