International Journal of Disaster Recovery and Business Continuity
Vol.11, No. 1 (2020), pp. 423-434

REVIEW ON "COMPOSITION OF WEB SERVICES IN SERVICE ORIENTED ARCHITECTURE"

Namrata Kashyap
M.Tech. Computer Science and Engineering
Ajay Kumar Garg Engineering College
Ghaziabad, India

Kirti Tyagi
Department of Computer Science and Engineering
Ajay Kumar Garg Engineering College
Ghaziabad, India

Abstract

With the advancement in technology, the use of distributed systems by the organizations has increased tremendously and the demands of customers have migrated to net-centric operations, so the need for an architecture that is standards based conceptual framework, flexible and adaptable in nature and supports distributed heterogeneous environments has heightened. In response to this needed architecture, Service Oriented Architecture(SOA) is presented. Hence, This paper introduces the review details of the Service Oriented Architecture(SOA), advantages and various challenges to SOA. It also covers a review work on composition of web services. Many researchers have realized that service composition have become a crucial aspect in terms of service oriented systems because sometimes a single service become an incomplete exposure of the functionality needed by the service requestor, so we have presented our review work regarding various techniques for the composition of web services so far. Composition can be done using various techniques like using combinatorial approach, composition methods based on QOS parameters and many more. During this review study, research work from 2000 upto 2013 has been considered.

Keywords—*Atomic service, Candidate service, Composite service, Composition of web-services, Quality of service(QOS), Web-service.*

I. INTRODUCTION

The use of distributed systems by the organizations has gained a pace in recent years as the means for attaining competitive advantage with information technology (IT) has deviated a little from efficiently managing the organization's operations to discovering ways to collaborate and coordinate with industry partners to deliver services as products in the markets that can be otherwise impossible to achieve. In today's expeditiously uncertain environment, new technologies have changed the perspective of looking at business processes to be formed of services and so the technology defined for the design and development of distributed systems is Service Oriented Arhitecture(SOA). SOA enables the business transactions to occur betwixt loosely coupled services.

The technology named as Service Oriented Architecture (SOA) [7,13,17], relates mainly to the design of a system, not to its implementation. A Service Oriented Architecture (SOA) refers to a style of building reliable distributed systems that deliver functionality as a service, with an additional emphasis on loose coupling between the interacting services. It emphasizes the implementation of the components as modular services that can be discovered and used by clients.

ISSN: 2005-4289 IJDRBC

A. Services

Services are specific software components. Services communicate with each other by sending and receiving messages. Services can adopt different roles like when acting as a service provider, a service can advertise its interfaces that can be invoked by other services that acts in the role of the service requestor. A SOA supports location transparency which means that services can be located and invoked by clients irrespective of their location. A SOA should be flexible enough so it could react quickly and efficiently to the changes of the environment by acquiring business strategy for that particular organization.

Services can have various characterstics:

- Services can be used as themselves only or in integrated form(composed) with other services also.
- Services can easily communicate with their clients with the help of messages.
- Services can participate in a workflow. The concept can be said as choreography of services.
- Services can be fully self-contained, or they may depend on the availabiliy of other services.
- Services may also advertise details such as their capabilities, interfaces and supported protocols

B. Loose coupling

The capability of interacting software components of sharing their little in-built knowledge with each other and discovering the required information at the time they need it.

It provides some benefits to the underlying infrastructure which includes:

- Flexibility- A service can be placed on any of the available servers and can also be relocated at any other.
- Scalability- Services can be inserted or deleted according to the demands.
- Replaceability- Keeping the original interfaces preserved, updated and modified implementation of a service can be introduced.
- Fault tolerance- In case of a server crash or other, a client can query the registry for alternate services offering the similar required functionality.

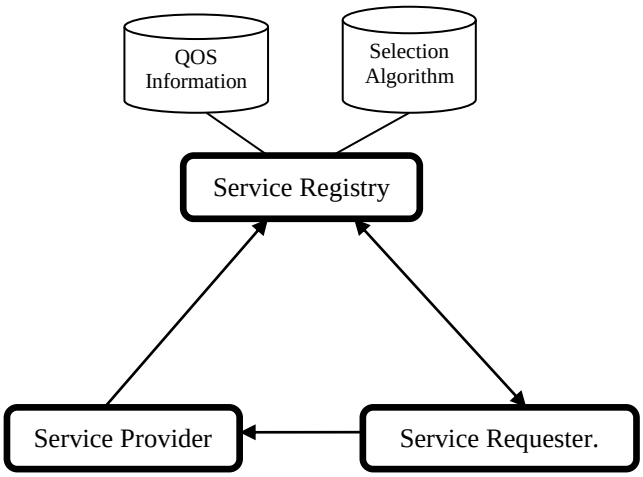

Fig. 1: Basic SOA Paradigm

ISSN: 2005-4289 IJDRBC

C. Statelessness

Statelessness [7], can be understood as a multistep process as:

- A service should provide sufficient state information at the end of each and every intermediate step to the client makes any qualified service to identify and continue the transaction.
- The client should always handover the state information to that particular service if it wants to process the next step of the transaction using that service.

The selected service should be able to accept and process the state information supplied by the client, regardless of the fact that whether it processed the earlier steps itself.

D. Web services

Web services [7,9], are the autonomous software systems identified by URIs which can be easily advertised, published, placed, and accessed through messages encoded using XML based standards (e.g., SOAP, WSDL, and UDDI) and transmitted through Internet protocols. Web services encapsulate all the application functionality and useful information resources and make them available through programmatic interfaces.A simple web service can be designed to represent a particular device or a legacy application in grid computing.

A simple difference between SOA and web services is that, service orientation is an architectural style while web services are an implementation technology.

E. Grid computing

A Grid is a group of autonomous devices and allows the shared use of data and computational resources of all the constituent devices.

Grid computing [7], is a form of distributed computing in which the use of contrasting resources such as compute nodes, storage spaces, applications and data, covering different physical locations and administrative domains, is optimized through virtualization and collective management.

Service Oriented grids are those simple grids in which services are used as resources themselves and other grid functionalities also.

F. Workflow technology

A Workflow schema [17], describes a workflow under different paronamas. It mainly represents the prototype showing process sequence. A workflow schema can be said as the complete description of the business process.

G. Component-based Software development(CBSD) Technique

The idea of this approach is to select appropriate components and then assembling(or integrate) them together with a well-defined architecture [23].

A component is a software object, noted to interact with other components, encapsulating some functionality or a set of functionalities [9]. A component always has a clearly defined interface and confirms to a already stated and specified behaviour common to all components within an architecture.A Component Based Software Development(CBSD) can eloquently reduce development cost, specify time-to-market, and improve maintainability,reliability and also overall quality of software systems [23].

ISSN: 2005-4289 IJDRBC

H. Service Oriented Architecture(SOA): Benefits and challenges

*Benefits:*Services reuse,Promptness to collaborate,Agility to adapt,Cost reduction,Improvement in efficiency [13].

*Challenges:*A procedure to direct how and when services are built, making sure that the services meet atleast basic technical criteria to enable interoperability, and identifying and avoiding unnecessary duplication of services [13].

II. LITERATURE SURVEY

As demands for the better services is increasing day by day rapidly in the market so it becomes a very hactic job for the clients to search for the best services from different vendors in search space and in minimum possible time, so this becomes the main reason why composition of services becomes the key solution for the most of the clients to find all services at one single place.So, Various challenges focusing on the techniques of performing composition of various services keeping in consideration that their natural and indistinguishable behaviour remains untouched and to make them available at a single service provider have been continuously raised which indirectly increases the cost of the project, unnecessary delays in the project and many more at the basic as well as at the bigger levels. That is why, the idea showing the need to combine various web-services has gained acceleration these days which can prove itself advantageous for the service consumers thereby enhances the efficiency of the respective composition model(project).

Some key points which should be elaborated to explain our approach and are used by various authors in the same field are as follows:

A. Composition of Web services

Web services composition [5], can be defined as combining web services, developed by various organizations and offering contrasting functional (e.g.,token purchase, payment), behavioral (remuneratable or not) and non-functional properties (i.e. QOS values- e.g. execution cost, response time, success rate) to offer more complex services.

B. Orchestration and Choreography

Orchestration [2], refers to an executable business process that can interact with both internal and external Web services and with the interactions occur at the message level. Orchestration always represents control from one party's point of view.

Choreography [2], tracks the message sequences among multiple parties and sources(typically the public message exchanges that occur between web services rather than a specific business process that a single party executes). So, It is more collaborative and allows each involved party to describe its part in the interaction.

C. Atomic service

A single service is called atomic service. An atomic service can be provided by a single or heterogeneous service provider(stake holder) and its result may depend upon the architecture on which it is executing.

D. Composite service

A simple composite service is a combination of many related atomic services. A composite Web service is an umbrella structure aggregating multiple other elementary and composite web services, which interect each other according to a process model [10].

ISSN: 2005-4289 IJDRBC

E. Quality Of Service(QOS)

All non-functional properties of a service which determines its quality and not consider the detailed process of its execution such as response time, etc.

F. Execution plan

This is a process of assigning a task to every service to create finest results. A task can be executed by a number of alternative services, but it is not possible for a service to execute a multiple tasks in a "single shot". To enable a combination of tasks to get executed by a single service, these tasks need to be assembled into a single one, which in turns can be called as a composite tasks and if we talk in terms of web services,it is called as composite service [9].

G. Quality criteria / QOS parameters for elementry services

Various quality criteria's [9,10], presenting distinct features, quality and abilities have been defined for elementary services, some of them can be defined in the following table:

TABLE1. Table elaborating various QOS parameters

No.	QOS Parameter	Description
1.	Execution price,$q_{pr}(p)$	Sum of the execution prices of all the operations invoked over the services that participate in an execution plan p.
2.	Execution duration,$q_{du}(p)$	The expected delay in seconds from the moment when a request is sent till the moment when the results are delivered.
3.	Reputation,$q_{rep}(s)$	A measure of a service's trustworthiness which mainly depends on user's experience on the service.
4.	Successful execution rate,$q_{rat}(s)$	The probability that a request is correctly responded within the maximum expected time limit indicated in the web- service description.
5.	Availability,$q_{av}(s)$	The probability that the service is accessible.

Diverse approaches adopted by various authors in this field can be summarized as-

In [3], a new decisional aspect as a support architecture has been added to the information system of the companies. Decisional aspect should maintain tha flexibility of the information system and should avoid the redevelopment of whole system when request for decisions comes around defining how decisions affects the organization.So as a solution, this paper presents an extended SOA with IDC (Intelligence Choice Decision) model- SOAda for supporting decisional aspect and a mete-

ISSN: 2005-4289 IJDRBC

model MDS (Model Decisional of Service) which clarifies why we need decisions at all the different levels and on various factors inside the company. This SOAda(SOA+IDC) adds a new Decision level to the already present Business and Information levels and hence proposes three types of services as Intelligent , Design and choice services.

In [24], author has discussed about the trust factor between the service providers and consumers when they interact with each other which says that there must be maximum trust between the interacting partners on each other in general Service Oriented Open and Dynamic Environments. Like in earlier Trust models, there exists no intermediator who can evaluate the trust level between the two interacting partners so, PB Trust model presents an evaluation criteria for the calculation of trustworthiness of a potential service provider by considering provider's experience on the service it provides, its suitability for the requested service and many more. This model uses two types of agents- provider's agent and consumer's agent responsible for their interaction. For calculating the trust factor, it focuses on some attributes of the service like service description, service request, etc. In this model, Agent's experience is also recorded and updated dynamically as a additional factor.

In [5], author has shown composition as well as the criteria, which every service has to follow if it wants to be a component service and it should provide maximum user satisfaction in terms of QOS criterian and also must satisfy the transactional requirement set by the the user. For simplicity, they have considered only two activities in the workflow and each web service(ws) is assigned to each activity to obtain a Transactional Composite Web Service(TCWS). From many patters of composition, they have used two- sequential and parallel patterns. A properly analysed algorithm which focuses on the workflow patterns and transactional properties of the component web service(ws) is used. Local QOS aware selection technique have been used. This whole paper focuses on automatic service composition and keeps user untouched with the process of composition and execution of web services. Providing the capability to the user to set the transactional criteria for a unsuccessfully executed service can even turn in to a risk for the system.

In [9], a middleware platform supporting the selection of web services for their composition is presented which can be used to satisfy user in any of the ways by defining some functions on QOS attributes of web services fulfilling all the constraints applied by the user and also the whole structure of the composite web service. Web services can be defined in terms of functional and non-functional(QOS) properties such as execution rate, reputation, etc. They have used a model named as AgFlow which is a middleware platform for web service composition based on quality of services. It introduces a new term Service Ontology(concepts used in description of web services), Service Class(describing capabilities of web service and how to use them) and Service Level Agreements(SLA)-do's and donot's of the quality of service being offered to the service requestors. For the service selection in this web service composition approach, two selection techniques i.e. local optimization and global planning(using integer planning) are used.

In [11], author says that in a dynamic sharing of workflow processes of a cross-enterprise environment over the web, the current workflow technology(defined as a statechart) can be used. It generally uses two types of technologies- workflow and agent technology. A process agent mainly combines various tasks in a certain way to form a cross-enterprise instance. All the service providers are assigned tasks and executes them in a distributed environment and the services that executes the workflow tasks are combined using agents. Our system searches for agents which matches the capabilities of the tasks and hence it uses four types of agents- user agent, process agent, discovery agent and service agent. All the working of the system is done using these agents. In general, a user agent consists of two main components- workflow specification editor and workflow execution monitor, and a discovery agent consists of service registry and a query engine.

ISSN: 2005-4289 IJDRBC

In [22], author presents an algorithm known as improved Particle Swarm Optimization Algorithm which removes the drawbacks of previous algorithms on composition by selecting the services with the same functionality but a greater QOS value to provide the user best composite service based on the non-funtional requirements such as availability, service price, etc. of the services. Selection is based on a multiple criteria decision making with a weighted sum model to confirm the QOS performance of service. After this, some parameters of PSOA are redefined to construct improved PSOA which are velocity, position and updating operations. To improve the performance of the algorithm some techniques like Non-Uniform Mutation(NUM) applied to global best particle to overcome the prematurity of PSOA, Adaptive Weight Adjustment(AWA) and Local Best First(LBF) strategies are used to enhance the convergence speed at both local and global levels respectively. This algorithm is advantageous here because it is easy to operate, efficient and easy to distributed execution. With this,it also has a limitation that it converges rapidly which can result in lost of diversity so we have used Non-Uniform Mutation technique in this algorithm.

In [15], author has said that in general, the relationship between the Web service QOS and environments are not considered usually in a workflow-based service composition approach which results in inaccurate information about the QOS for the composite service selection and hence makes the composite service inefficient. To remove this problem, an advanced service composition approach based on production QOS rules is proposed. As normally, we cannot directly judge the influence of various environmental factors on Web service QOS so, this paper proposes a"black-box" analysis method to optimize the composite services and discovering the knowledge that "the QOS of a Web service will be higher in a particular environment". In the proposed approach, firstly, an execution repository storing the execution information of the composite service is constructed, secondly, extraction of the QOS data of the Web service is done, then thirdly, Estimation of the Timely TQOS (True QOS) is done from the obtained QOS from the execution information of the Web service, after that mining of the production QOS rules to express the performance of the Web services in different environments in done and finally, the optimization of the Web service selection using the mined knowledge is performed.

In [25], author has proposed an approach which answers the question that how to deal with the requirements freely expressed by the user and an unplanned interaction of agents for service composition. This whole architecture is based upon Multi-Agent Systems. Agents play the lead role during the composition of services and utilizes the semantic information like non-functional characterstics of web services and whole process uses local and reactive processing in MAS which thereby avoids centralized system's pitfalls and improves the scalability by making the system more distributive. Selection and composition, both are done using agents. The agents which provides the semantic web services are interactive agents and an interaction protocol is issued which allows agents to respond to a complex query which requires the coordination of various candidate services. In MAS, services are used as a constituent of an agent. The service selection is done using a mediator agent which exploits a time out mechanism and fetches the best combination of answers from a set of various combinations of answers.

In [18], author has proposed an advanced algorithm which is an extension of optimal winner determination problem in combinatorial auctions and it works on a different type of graphs of shared and unshared items. Auctions are where an e-buyur can buy not only a single item but a bunch of items. These above graphs are the essence of work process and are used for an exploration process of the bids which inturns accelerates the search to the optimal solution. The efficiency of the graphs fully depends upon the number of bids. Firstly, an example is used to explain the steps of the search algorithm and the use of graphs. After that algorithms are applied to partioning of bids which were

ISSN: 2005-4289 IJDRBC

used for building of graphs and then used for exploring the graphs in between the process of building of optimal solution tree. It mainly brings about various tools like graphs and their properties, are generic and can be easily stuffed to any of the present algorithms to enhance the performance of the overall search process. This can easily refine the buyer's needs and preferences so a better allocation of items can be done in accordance with the buyer's needs.

The literature by many researchers containing the pros and cons, classification on which they are differentiated and their area of use is shown in the following table:

TABLE2. Table showing the pros and cons of the Research work

Author	Year of Development	Classification	Area of use	Pros (+) and cons (-)
Joyce El Haddad, Maude Manouvrier, and Marta Rukoz	2010	Transactional requirement and QOS Criteria	Web service	(+) alleviates user from composition and execution process by focusing on automatic service selection (-) unable to use full context policies in case of failure in choosing and detailing transactional requirement
Xing Su, Minjie Zhang, Yi Mua,and Quan Bai	2013	Soft computing	Web service	(+) good for open and dynamic environments (-) requests only for single service were included
Liangzhao Zeng, Boualem Benatallah, Anne H.H. Ngu, Marlon Dumas, Jayant Kalagnanam, and Henry Chang	2004	Integer programming	Web service	(+) global planning and local optimization- service selection techniques for composite service are used (-) not a computation cost-efficient model, more composite service's execution time
Senol Arikan	2012	Reliability management model	SOA and reliability	(+) competent solution for optimizing the reliability of SOA (-) fluctuating degrees of reliability of atomic web services
Fatima Boumahdi, Rachid Chalal	2013	Architecture based	Web services and SOA	(+) introduction of decisional aspect to existing SOA based organizations (-) cannot be applied to real systems
Rainer Berbner, Tobias Grollius, Nicolas Repp	2005	Architecture based	Management of SOA	(+) SoA with an additional management functionality, e.g. monitoring mechanisms and SLA

ISSN: 2005-4289 IJDRBC

				management (-) not applicable to a simulated environment
Liangzhao Zeng, Boualem Benatallah, Marlon Dumas, Jayant Kalagnanam, Quan Z. Sheng	2003	Linear programming	Web services	(+) optimal(efficient) selection of component services using global planning approach (-) no reaction to runtime exceptions
Ming-Wei Zhang, Bin Zhang, Ying Liu, Jun Na, Zhi-Liang Zhu	2010	Architecture based	Web services	(+) increases the system stability, improves the system efficiency and reduces the possibility of unusuality in dynamic service execution environments (-) systems cannot use this approach at the initial stages of the execution
Latha Srinivasan , Jem Treadwell	2005	Architecture based	SOA, Web service and Grid computing	(+) a criteria for the development of highly scalable distributed applications (-) key standards are a necessity for the development of next-generation grids
Zaigham Mahmood	2009	Review	SOA and Grid computing	(+) grids are termed as an "Enterprise grid" providing a dynamic framework with a management node as a part of it (-) SoA and Grid, both works on completely distinct approaches
Yasmine Charif and Nicolas Sabouret	2006	Architecture based	Agents and Web services	(+) interaction protocols allowing agents to interact and discuss about their services (-) constraint to get the results from a single service
Xia Cai, Michael R. Lyu, Kam-Fai Wong and Roy Ko	2010	Architecture based	CBS	(+) binds both component and system QA(Quality Assurance) and their interactions also (-) actually not applicable to real world projects
Samir Aknine	2004	Algorithm based	Combinatorial auctions	(+) updation of the optimal solution with minimum computations (-) no synchronization of both types of graphs

ISSN: 2005-4289 IJDRBC

Wenbin Wang, Qibo Sun, Xinchao Zhao, Fangchun Yang	2010	Algorithm based	Web services	(+) suitable to solve QoS-aware Web Service Selection problem (+) improves the diversity of PSOA's population and overcome the prematurity situation effectively (+) improve the convergence speed in global and local level (-) does not take adaptive termination of algorithms into account

III. CONCLUSION

The paper talks mainly about the composition of web services in service oriented computing environment. Several important composition techniques which are useful to show how we can combine various candidate services to make a composite service in the rapidly changing and diverse environment of SOA are also discussed. Although different methods and approaches have already been discussed by various researchers have been discussed in the literature section of the paper and the comparision of contrasting techniques based on distinct classification parameters has also been mentioned but that's just not the enough to express the research work and still more research is needed in this area so we have decided to extend the work by enhancing the composition model(doing composition) of web services based on QOS parameters using fuzzy logic as our future work. This composition will be dynamic in nature with a database containing all the candidate services from which suitable and appropriate services can be selected.

References

[1] Amelia Maurizio, James Sager, Peter Jones, Gail Corbitt and Lou Girolami, Sager and Corbitt, Service Oriented Architecture: Challenges for Business and Academia, Proceedings of the 41st Hawaii International Conference on System Sciences – 2008.

[2] Chris Peltz, Web Services Orchestration and Choreography, 2003 IEEE.

[3] Fatima Boumahdi, Rachid Chalal, SOAda: Service Oriented Architecture with a Decision Aspect, Published by Elsevier B.V. Selection and peer review under responsibility of KES International, 2013.

[4] Grace A. Lewis and Dr. Dennis B. Smith, Four Pillars of service-Oriented Architecture, The Journal of Defense Software Engineering, September 2007.

[5] Joyce El Haddad, Maude Manouvrier, and Marta Rukoz, TQoS: Transactional and QoS-Aware Selection Algorithm for Automatic Web Service Composition, IEEE TRANSACTIONS ON SERVICES COMPUTING, VOL.3, NO.1, JANUARY-MARCH 2010.

[6] Kyriakos Kritikos, Sylvain Kubicki and Eric Dubois, Goal-based business service composition, 18 January 2013 © Springer-Verlag London 2013.

[7] Latha Srinivasan and Jem Treadwell, An Overview of Service-oriented Architecture, Web Service And Grid Computing, HP Software Global Business Unit, November 3, 2005.

[8] Li H H, Du X Y, Tian X, A review-based reputation evaluation approach for Web services. Journal of Computer Science and Technology, 2009, 24(5): 893-900.

ISSN: 2005-4289 IJDRBC

[9] Liangzhao Zeng, Boualem Benatallah, Anne H.H. Ngu, Marlon Dumas, Jayant Kalagnanam and Henry Chang, QOS-Aware Middleware for Web Services Composition, IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL.30, NO.5, MAY 2004.

[10] Liangzhao Zeng, Boualem Benatallah, Marlon Dumas, Jayant Kalagnanam and Quan Z. Sheng, Quality Driven Web services Composition, www2003, May 20-24, 2003, Budapest, Hungary, ACM 1-58113-680-3/03/0005.

[11] Liangzhao Zeng, Boualem Benatallah, Phuong Nguyen, Anne H.H. Ngu, AgFlow: Agent-based Cross Enterprise Workflow Management System, Procedings of the 27th VLDB Conference, Rome, Italy, 2001.

[12] Liangzhao Zeng, Hui Lei, and Henry Chang, Monitoring the QoS for Web Services, pp. 132–144, © Springer-Verlag Berlin Heidelberg 2007.

[13] Manoj Mansukhani, Sevice Oriented Architecture, June 28, 2005.

[14] Milanovic N,Malek M. Current solutions forWeb service composition.IEEE Internet Computing, 2004, 8(6): 51-59.

[15] Ming-Wei Zhang, Bin Zhang, Ying Liu, Jun Na, and Zhi-Liang Zhu, Web Service Composition Based on QoS Rules, JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 25(6): 1143-1156 Nov. 2010 in Springer Science.

[16] Agrawal R, Imielinski T, Swami A, Mining association rules between sets of items in large databases. In Proc. the ACM Conf. Management of Data, Washington DC, USA, May 25-28, 1993, pp.207-216.

[17] Rainer Berbner, Tobias Grollius, Nicolas Repp, An approach for the Management Of ServiceorientedArchitecture (SoA) based Application Systems, Enterprise Modelling and Information Systems Architectures (EMISA 2005) October 24-25, 2005,Klagenfurt, Austria (in conjunction with ER 2005).

[18] Samir Aknine, Improved Optimal Winner Determination Algorithms Using Graph Structures, Paris, 2004.

[19] Sara Mahdavi-Hezavehi, Matthias Galster , Paris Avgeriou, Variability in quality attributes of service-based software systems: A systematic literature review, Information and Software Technology 55 (2013) 320–343, 2012 Elsevier.

[20] Senol Arikan, Integrated Communication Systems Group University of Kaiserslautern "Automatic Reliability Management in SOA-based critical systems", European Conference on Service-Oriented and Cloud Computing, September 13, 2012.

[21] Tuomas Sandholm, Approaches to winner determination in combinatorial auctions, Washington University, Department of Computer Science.

[22] Wenbin Wang, Qibo Sun, Xinchao Zhao and Fangchun Yang, An improved Particle Swarm Optimization Algorithm for QoS-aware Web Service Selection in Service Oriented Communication, International Journal of Computational Intelligence Systems, Suppl. 1(December,2010).

[23] Xia Cai, Michael R. Lyu, Kam-Fai Wong and Roy Ko, "Component-Based Software Engineering: Technologies, Development Frameworks, and Quality Assurance Schemes", Asia-Pacific Software Engineering Conference (APSEC 2000), July 30, 2010.

[24] Xing Su, Minjie Zhang, Yi Mu, Quan Bai, A robust trust model for service-oriented systems, School of Computer Science and Software Engineering, University of Wollongong, Wollongong, NSW 2522, Australia and School of Computing and Mathematical Sciences, Auckland University of Technology, Auckland, New Zealand, December 2012.

ISSN: 2005-4289 IJDRBC

[25] Yasmine Cherif and Nicolas Sabouret, Dynamic Web Service Selection and Composition: An Approach on agent based dialogues, Paris,2003.

[26] Zaigham Mahmood, Synergies between SOA and Grid computing, Vol 8, 2009 ISSN: 1943-7765; z.mahmoodQderby.ac.uk.

ISSN: 2005-4289 IJDRBC