

An Elite Model for COTS Component Selection Process

Asif Irshad Khan

Department of Computer Science, FCIT
King Abdulaziz University
Jeddah, Saudi Arabia

Abstract—Component-based software development (CBD) promises development of high-quality trustworthy software systems within specified budget and deadline. The selection of the most appropriate component based on specific requirement plays a vital role for high-quality software product. Multi-Agent software (MAS) engineering approach played a crucial role for selection of the most appropriate component based on a specific requirement in a distributed environment. In this paper, multi agent technique is used for component selection. A semi-automated solution to COTS component selection is proposed. It is evident from the result that (MAS) plays an essential role and is suitable for component selection in a distributed environment keeping in view of the system design and testing strategies.

Keywords- Component selection, Component based development, COTS, Multi Agent Software Engineering

I. INTRODUCTION

Developing high-quality software product within budget and time is always a challenging concern to the software industries. Component-based software development plays a crucial role in addressing challenges faced by the software industries. CBD supports development of software using software components, also, known as commercial of the shelf system (COTS). COTS components are sold in open market and manufacturers of COTS define how a component can plug and play into a system based on the specified requirement [1].

Ideally, COTS components are like black box solutions, one has to learn how a component can be plugged into the system by using its required and provide interfaces as defined by its manufacturers. The component is developed by following software development phases such as System requirement specification, System Modeling and Design, System implementation and Testing [10]. COTS Components are highly reliable and trustworthy as they are rigorously tested by its manufacturer and most of the

components have a rating that is evaluated by its customers [12].

However, Selection of component plays a vital role in developing high-quality software systems within specified constraints like budget, efforts and development time. Selection of candidate component as per specified requirement is a challenging task as the selection is usually based on multiple criteria such as functionality support, cost, reliability, security, adoptability etc.

Multi-Agent System (MAS) is intelligent software system consists of software components also known as Agent. These agents can interact with each other or can coordinate with each other to accomplish assigned task. For example, agents can be used to identify most suitable candidates for a particular set of requirements.

Multi-Agent approach is ideal for this research as the agent can explore all available servers in the distributed network and recommend potential candidate components.

This paper is organized as follows: section I describes introduction of the paper, sections II discusses about the related work, section III puts forward the proposed model, section IV explains the role of different agents in the proposed mode, and section V conclusion and future work.

II. RELATED WORK

Software engineering research community has suggested different strategies and methodologies for the selection of candidate COTS components.

Rikard et al. surveyed component selection state of the art in the embedded systems domain and presented a meta-model for selection method based on common activities and practices. The authors also gave suggestions which can be utilized as a schema when making the technique and methodology for component selection [2].

X. Burgués et al. suggested a process model for combined selection of components based on the features of two levels, a global one matching to the combined selection itself, and a local level where all the individual selection processes take place in specific areas. The main contribution of this study is to consider the aggregate expense for a system rather than specifying in advance the individual expenses for various components [3].

Lawrence et al. proposed an approach (CARE/SA) which supports the iterative selection (matching and ranking) of COTS components. The author considered COTS component's representation within architecture as an aggregate of their functional and non-functional requirements with its own set of attributes. The proposed framework could be viewed as an extension to previous methodologies with a systematic approach to matching, ranking, and selection COTS components as shown in figure 1. [4]

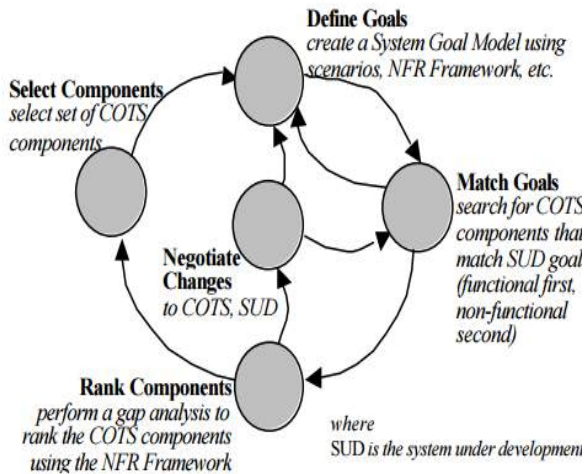


Figure 1. Overview of the CARE Process [4].

Zahid et al. in their work evaluated the recent practice of evaluation and selection of COTS components from software houses in Pakistan. Closed questionnaire methodology was adopted in the study. The result shows that no criterion is utilized for evaluating and selecting COTS components by the majority of respondents. Authors proposed a framework for COTS selection, integration and evaluation [5].

Tarawneh et al. [6] suggested a framework support and improve the COTS software evaluation and selection processes in industry. To achieve this objective the authors have shown that specific objectives have to be addressed:

- a) Identify the processes which support COTS software evaluation and selection.
- b) Determine the criteria or requirements which are important for successful evaluation and selection process.
- c) Propose methods and techniques to address the mismatch between COTS features and customer requirements.
- d) Develop a repository to manage information from previous selection cases that support the decision-making process.

M. Shakeel et al. [7] proposed component selection methodology for component-based software engineering. The author claimed that most of the IT developers in development countries (UDCs) are unaware about the COTS evaluation and selection methodology. This methodology will reduce development times, cost effective and less strenuous efforts on the basis of software quality model ISO/IEC 25010 as shown in figure 2.

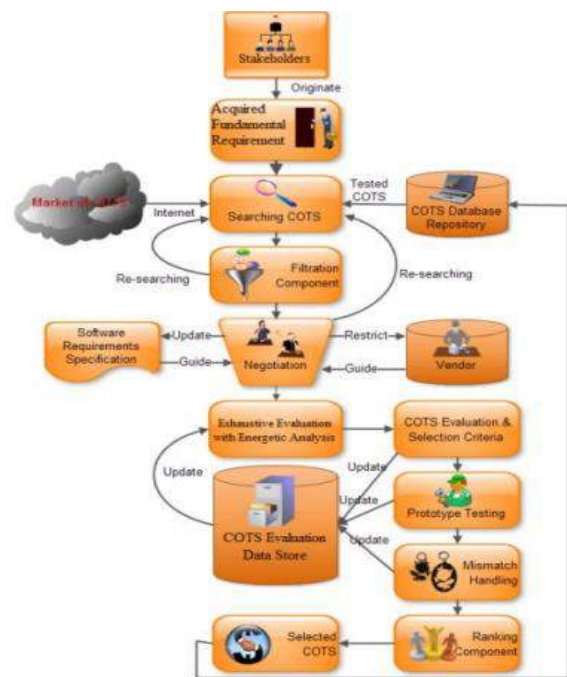


Figure 2. IROTS Process [7].

Shah et al. proposed a method for component selection using fuzzy logic. Several software quality factors like reusability, portability, security, maintainability etc were considered in the proposed method. This methodology was evaluated using hypothetical case study [8].

Agora [9] is a web-based component search prototype proposed by Carnegie Mellon software engineering Inst. Agora provides agents that crawl the

Web for components. In this model, the whole process is divided into two main tasks.

One of the task which is an automated background task is performed by an agent, i.e. a background agent is responsible to automatically search the location and indexes of COTS on the web for component specific/specification model. While the component selection and retrieval task is a manual and done by the expert as shown in figure 3.

The main advantage of agora approach is its ability to automatically build an index of available worldwide components on the web [9].

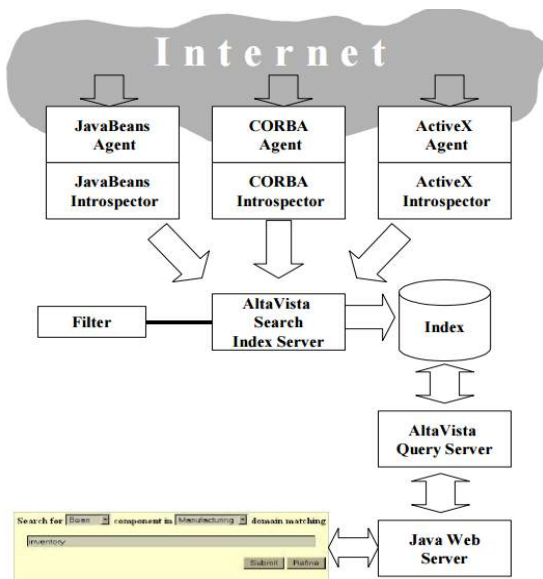


Figure 3: Agora Architecture [9].

III. PROPOSED MODEL

The currently available models for COTS component selection are complex and difficult to implement and also its time consuming. The analysis indicated that there is a need of improvement to COTS component selection process.

This paper proposed an efficient semi-automated COTS component selection technique which is easy to understand and easy to implement as well.

A semi-automated model [ABCS] for COTS component selection is proposed. This model is based on the multi-agent technique in which a group of agents works together with the single objective of completing a specific task i.e. identifying the list of suitable candidate components. Figure 4 shows a conceptual view of this approach.

The whole process of identifying the candidate component requires multiple agents to work together. A multi-agent system is a loosely coupled network of several agents which interact among themselves or with external environments to solve problems that are

beyond one's individual abilities or knowledge of each problem solver.

As multi-agent is composed of several autonomous agents, there is a need of coordination and cooperation among the agents as shown in the Figure4.

Firstly the task of searching COTS component as per defined criteria is broken into several sub-tasks and each of the sub-tasks is assigned to specific agents. For example, sub-tasks like matching requirement against integration complexity, security features verification, cost-benefit analysis, adaptation, and quality assurance verification is assigned to different agents as shown in figure 4.

These agents, after solving their respective sub-tasks, co-ordinate with the main agent to produce the result's set of candidate components as per the selection criteria in the form of a weighted matrix [11]. Finally, an expert team will decide the candidacy of the most suitable component available from the list of choices.

ABCS model take the outline of the user requirement [User requirements may be in plain English or in script form or any other defined form] and application domain [like health, agriculture, business, education etc] as the input parameter and select the candidate component from the list of available components from different repositories in a distributed network as shown in figure 5.

As it is often seen that based on the requirements, there might be several COTS products that match the requirements for different degrees, but in few cases most likely none of the several candidates would completely match the user requirements.

The proposed model also has an agent integrating tool [ABIT] that takes system architecture and list of candidate components and integrates them to generate source code for integrated components as shown in figure 6.

In case of any compatibility issues it is handled by compatibility test agent. It tests the compatibility issues while integrating the selected candidate components with the system architecture.

This agent automatically verify and validate the compatibility of selected COTS component for any risk of failure since selected component has to integrate into the system and there is always a risk of component failure at later stages of development.

In case the test shows compatibility problem while integrating the selected candidate components, the system will select the next possible candidate component from the ABCS list as shown in figure 6. Interactions between these agents are shown in figure 7.

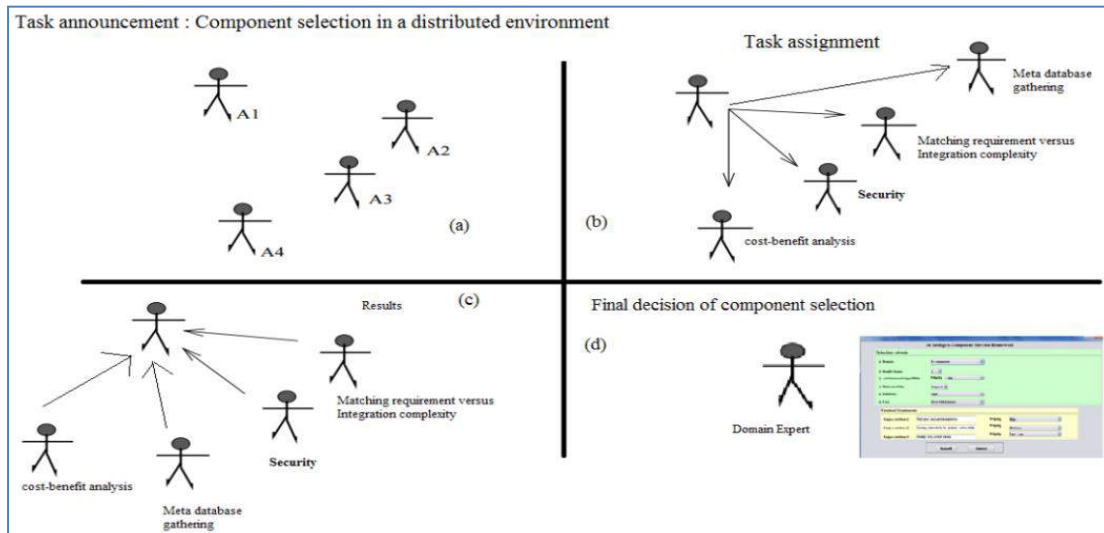


Figure 4. Conceptual view of multi-agent approach for component section.

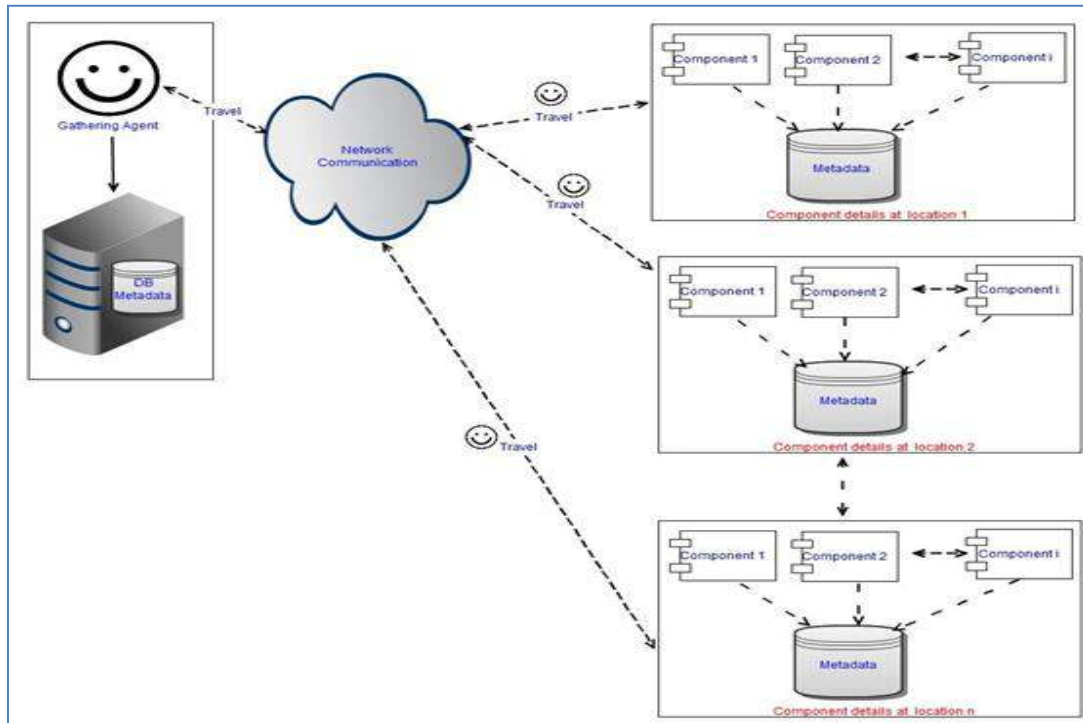


Figure 5. Conceptual Model of Component Gathering Agent.

IV. AGENT ROLE IN THE MODEL

A. Gathering Agent

The main responsibility of this agent is to maintain in-house component meta-data database. Once requirement is specified to this agent, it travels in distributed network, collect meta-data details related to available matching components and classify them

based on the component model to update in-house database.

Vendor of the component maintain meta-data details in online repositories, as shown in figure 5. Following are some of the listed categories of component classified based on the component model.

- Microsoft’s COM/DCOM/COM+ components.

- Microsoft's .Net components.
- Object Management Group OMG's CORBA.
- Oracle J2EE or JEE 5 components
- Oracle JAVA RMI components

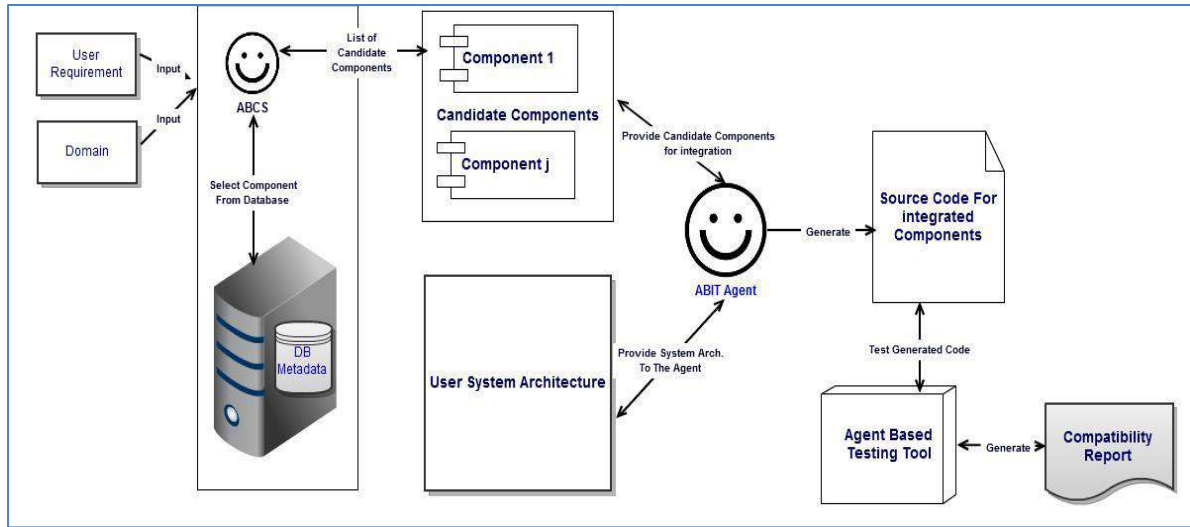


Figure 6. Model of Multi-agent based Component selection and Integration with Testing Tool.

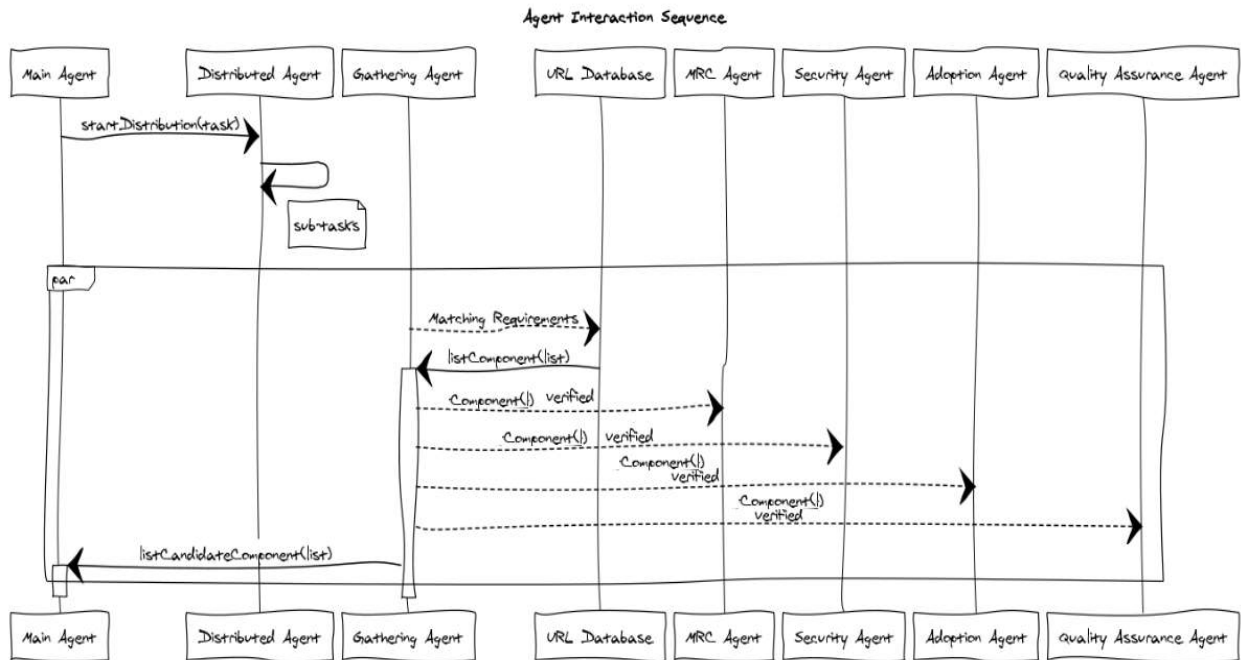


Figure 7. Agent Interaction Sequence diagram for Multi-agent based Component selection.

B. MRC Agent

Matching requirement versus Integration complexity is addressed by MRC agent. It plays a major role in the evaluation of every component based on its matching

requirement and its relative complexity to integrate.

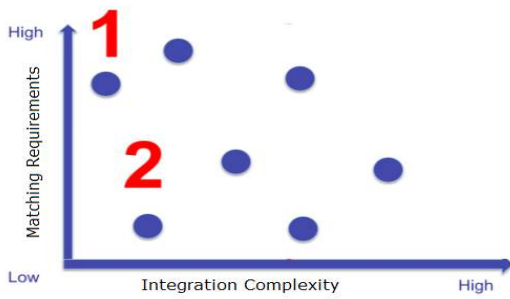


Figure 8. MRC Agent.

The component that has highest matching requirement and lowest effort to integrate will be considered from the component candidate list as shown in figure 8 and figure 9.

Recommended Candidate Components		
Component name	Matching index	Recommendation
Component A	0.76	
Component B	1	High
Component C	0.82	
Component D	0.76	
Component E	0.95	High

Figure 9. MRC Agent.

C. Security Agent

The role of this agent is to verify the security features in the selected candidate components. The components should be verified for any breach of security. This agent has authentication and authorization manager which checks how certificate authority, usable security policies etc. are defined and security mechanisms are implemented and supported.

D. Cost Agent

The main responsibility of this agent is to take part in the cost-benefit analysis so as to compare the component features & functionalities with the cost. Costs should be seen in broad aspects; because low cost components may result in higher total cost along the product’s life cycle. Buying component with loads of features also results in a higher cost and most of the features remain unused by the users.

E. Adoption Agent

The main responsibility of this agent is to verify component adoption process by studying all pre-requisite in a way that components can be easily integrated into the system architecture effectively.

F. Quality Assurance Agent

It is well-known fact that no software can be 100% error-free. To check component high-quality features, knowing that it is developed by the vendor company and may have some issues, it is necessary to check the rating of the component as most of the vendors maintain a rating of the components.

Also, it is necessary to find out what problems other users are encountering from vendor site where issue tracker related to the component are posted. The role of this agent is to validate and verify above mentioned high-quality attributes and generate a software quality metric for measurement that is used to evaluate software quality in a system as shown in figure 10.

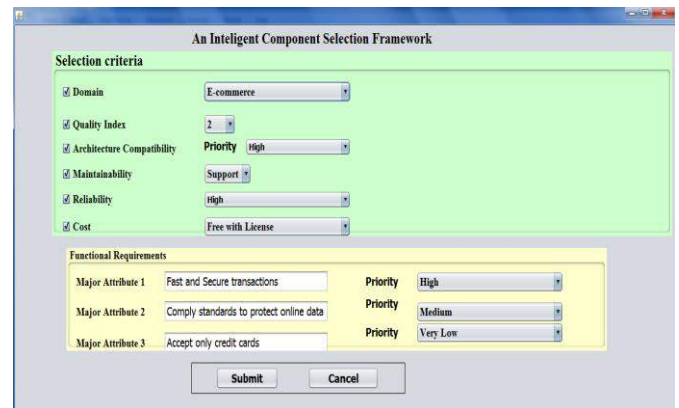


Figure10. MRC Agent.

V. CONCLUSION

In this paper, an approach to COTS component selection was presented. The software component selection is done based on different attributes like availability of features and functionalities in the component as per requirements, component adoption easiness in the architecture, security, cost, quality of service etc.

The proposed approach attempts to find the best candidate component list based on the requirements in a distributed environment. The multi-agent technique is considered in this approach by breaking a task into several sub-tasks and these sub-tasks are assigned to specific agents to be solved.

All the rigid conditions are taken into consideration and based on this; the most optimal component sets are generated. Finally, domain experts will decide the candidacy of the most suitable component available from the list of choices.

This approach gives acceptable results. This semi-automated solution to COTS component selection has the main improvement in reduction time of COST selection process which leads to acceleration of development and time to market. Further, knowledge of available COTS component is another advantage of the proposed solution.

REFERENCES

- [1] M. Elammari and R. Ali, "Towards a Multi-Agent Mechanism for Software Component Selection", The International Arab Conference On Information Technology (ACIT2014), Zarqa University in Jordan.
- [2] Rikard Land, Laurens Blankers, Michel Chaudron, Ivica Crnković, "COTS Selection Best Practices in Literature and in Industry", Mälardalen University, School of Innovation, Design and Engineering, Västerås, Sweden.
- [3] X. Burgués, C. Estay, X. Franch, J.A. Pastor, and C. Quer, "Combined selection of COTS components", Proceedings of ICCBSS, February, Orlando, Florida USA, 2002, pp. 54-64.
- [4] Lawrence Chung, Kendra Cooper, "COTS-Aware Requirements Engineering and Software Architecting", Department of Computer Science, University of Texas at Dallas.
- [5] Zahid Javed, Ahsan Raza Sattar, Salman Afsar, Muhammad Shakeel Faridi, "An Empirical Study of COTS components Persuasion, Evaluation & Selection and Integration in software houses Faisalabad, Pakistan", IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 6, No 2, November 2012.
- [6] Feras Tarawneh, Fauziah Baharom, Jamaiah Hj. Yahaya and Faudziah Ahmad, "Evaluation and Selection COTS Software Process: The State of the Art", International Journal on New Computer Architectures and Their Applications, The Society of Digital Information and Wireless Communications, (IJNCAA) 1(2): 344-357, 2011 (ISSN: 2220-9085)
- [7] M. Shakeel Faridi & Zahid Javed, M. Haris Abid, Mudassar Ahmed, Dr. Md Asri Bin Ngadi, "ILOTS: A Proposed COTS Evaluation & Selection Methodology for Component Based Software Engineering in Under-Development Countries", Department of Komputing Universiti Teknologi 2nd International Conference on Advances in Computer Science and Engineering (CSE 2013), Malaysia, Johor Bahru, Malaysia
- [8] S Nazir, MA Khan, S Anwar, H Khan, M Nazir "A novel fuzzy logic based software component selection modeling", 2012, International Conference on Information Science and Applications, 1-6.
- [9] R.C. Seacord, S.A. Hissam, K.C. Wallnau, "Agora: A Search Engine for Software Components", IEEE Internet Computing, vol. 6, no. 2, pp. 62-70, 1998.
- [10] Khan, A.I., Khan, U.A., 2012. "An Improved Model for Component Based Software Development." Software Engineering 2, no. 4 (2012): 138-146.
- [11] Khan, A.I., Alam, M.M., Shariq, M., May 2015. " A Perspective Study of Intelligent System for Component based Development". International Journal of Computer Applications 117(4):11-17.
- [12] A. Barnawi, M. Rizwan Jameel Qureshi and A. IrshadKhan, "A framework for next generation mobile and wireless networks application development using hybrid component based development model", Int. J. Res. Rev. Next Gener. Netw. (IJRRNGN), vol. 1, no. 2, (2011), pp. 51-58.

AUTHOR PROFILE

Asif Irshad Khan, Ph.D., is working as a faculty member in the department of Computer Science, FCIT, King Abdulaziz University, Jeddah, Saudi Arabia. Twelve years of experience as a professional academician and researcher. Dr. Khan received Ph.D. in Computer Science and Engineering from Singhania University, Rajasthan, India and Master & Bachelor degrees in Computer Science from the Aligarh Muslim University (A.M.U), Aligarh, India. He has published several research articles in leading journals and conferences. He is a member of the editorial boards of international journals and his current research interest includes Software Engineering with a focus on Component Based and Software Product Line Engineering.



© 2017 by the author(s); licensee Empirical Research Press Ltd. United Kingdom. This is an open access article distributed under the terms and conditions of the Creative Commons by Attribution (CC-BY) license. (<http://creativecommons.org/licenses/by/4.0/>).