MULTI CRITERIA BASED RETRIEVAL TECHNIQUES FOR REUSABLE SOFTWARE COMPONENTS FROM COMPONENT REPOSITORY

Shambhu Kumar Jha
Research Scholar
Mewar University
Chittorgarh, India

Dr. R.K. Mishra
National Informatics Centre,
New Delhi, India

ABSTRACT - In recent year's software development community have started developing advance, economical and more reliable software to handle complex, distributed and sophisticated business process through the concept of component based software development (CBSD) [8]. This is due to the fact that Object Oriented Concept was inadequate to handle the ever changing requirement of the customer's business process[1][12]. Concept of CBSD is totally based on foundation of software reuse and develops software by selecting appropriate off the shelf software components from component library and integrating them in order to achieve its desired task[8][7]. Software reuse has a very constructive impact on software productivity, maintainability cost and development time [8]. Retrieval of appropriate software components from its repository and its reuse is very important as it has key contribution to the software quality factors. In this paper efforts are made to provide some suitable guiding principle to the software component integrators about how to retrieve most suitable software components from existing software repository in the development of component based software. This paper also discusses the spirit of suitable component classification and further customization that helps the software developer in selecting the best component which will improve the overall quality of system and of course the software reuse.
Keywords: CBSD, COTS, Component Classification, Software Reuse

I. INTRODUCTION

The main center of attraction of component based software engineering is to purchase but not develop theory. This theory is basically inherited from the principle of hardware engineering, where individual hardware component is not built or manufactured as per specific customer needs but they are selected from already built component of the shelf (COTS). Reusable Software Components provides significant enlightenment to software development community about various challenges faced by them [4]. It is acknowledged with improving the productivity, maintainability and quality of software development and many organizations have claimed significant benefits from it. When an individual software component undergoes repeated reuse under different operating environment as per varying customer needs, it definitely improves the overall software quality as well as performance of individual software component. Due to its black box nature where its internal structure details are unavailable to the component integrator, organizations have developed policies for systematic reuse program as a resulted they have developed in-house libraries of reusable components**.**

II. RELATED WORK

Retrieval of most suitable software component from component repository is considered as most tedious task. Selection of a software component depends on multiple criteria due to varying technical and business needs and objectives [2]. These criteria's are even conflicting in nature at many occasions and making a right decision about selection of most suitable component becomes challenging. Varying component dependencies among the components are also a major bottleneck in component selection process. A most suitable software component may not be well match with

already preexisting components in the component based software [5][3]. Consequently, many organizations have started spending much time in development and selection process of reusable component selection [3]. Choice of the appropriate software components has a major impact on the final products [3]. It is very much needed to set standard procedures for component selection process. Due to its never-ending potential, in terms of development time, cost and quality of the product, the technology of software reuse is becoming popular day by day. In a recent survey conducted on the component based software development from one hundred and eighteen software companies around the world, it is found that 53% of the organizations are using component based approach in its development [1]. Lack of proper selection techniques of software component also may introduce high level of risk like inclusion of software component with unknown quality parameter i.e. reliability, security, dependability and maintenance.

It has been observed during the various literature surveys that software component users adopt a range of approaches for component selection process. It includes incorporation of selection criteria's and calculation of score of each alternative with respect to selection criteria which determines which is the most suitable alternatives as per our business and technical needs. Calculation of score for each alternative is based on appropriate decision making algorithm.

Among various informal criteria's like hands on trial, past experiences with already developed components and customer feedback, success rate for hand on trial method is high as it leads to careful observation of each software component in local and most trusted environment. Researchers, practitioners and component developers have also tried to apply some of the metrics of object oriented system which is also applicable in accessing reusability of component based system [12]. Reusability of software component can further be improved through improved portability, better documentation, higher customizability, understandability and interoperability. Reusability is further not limited to reusability of entire component but it is extended up to reusability of documentation or test suits for testing of software components or interface designed for integration of one component with another. During the study of related work efforts are made to highlights various key issues which can provide suitable guidelines to component integrator while selecting reusable components from components repository.
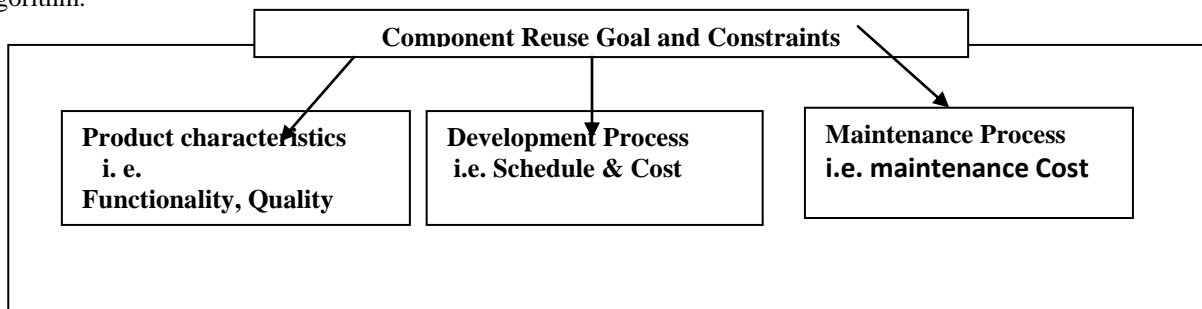

Fig 1: Reuse Goals for Software Component

III. SOFTWARE COMPONENT REUSE AND REUSABILITY

Component Based Software Development helps in planning, designing and developing complex, reliable and economical software using preexisting software components as the basic elements. Component Based Software Development can be broadly classified into two types:
a) Developing software with reuse of previously developed component without any change.

b) Developing software with reuse of previously developed component with customization to suit the current requirement.
During the process of identification and selection of suitable software component if an existing component from component repository is completely fulfilling the requirement of new software then those components are directly plugged into new software being developed and used without any further alteration [9]. On the other side reusing an component without any further

modification is not a easy choice. This is because use of different programming language, operating environment, business logic and hardware software environment for individual software component.
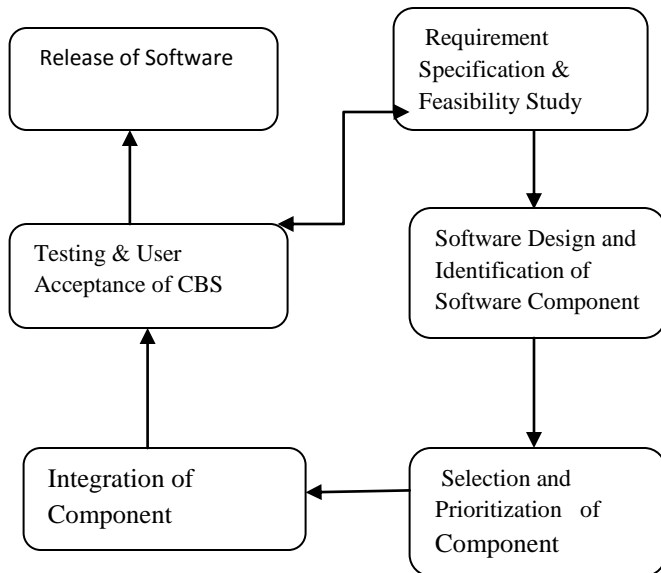

Fig2:   Component Based Reuse Model using preexisting component without change.

Due to black box nature of software component, reuse with customization is more difficult choice. Another challenging task in case of reuse with customization is to identify those parts of the software components which really require modifications [4]. Even after changes the modified components need to be thoroughly tested before plugging it into new system

Selection of the best component for reusability is decided on the basis of which component is more easily adapted and has less dependability with other component

Guiding principle for improving reusability of customized software components are [6]:

a) Simple and smaller code structure,
b) Use of platform independent language.
c) Superior documentation.

These three guidelines may be used in developing component based system to rank high candidate component for reuse with customization.
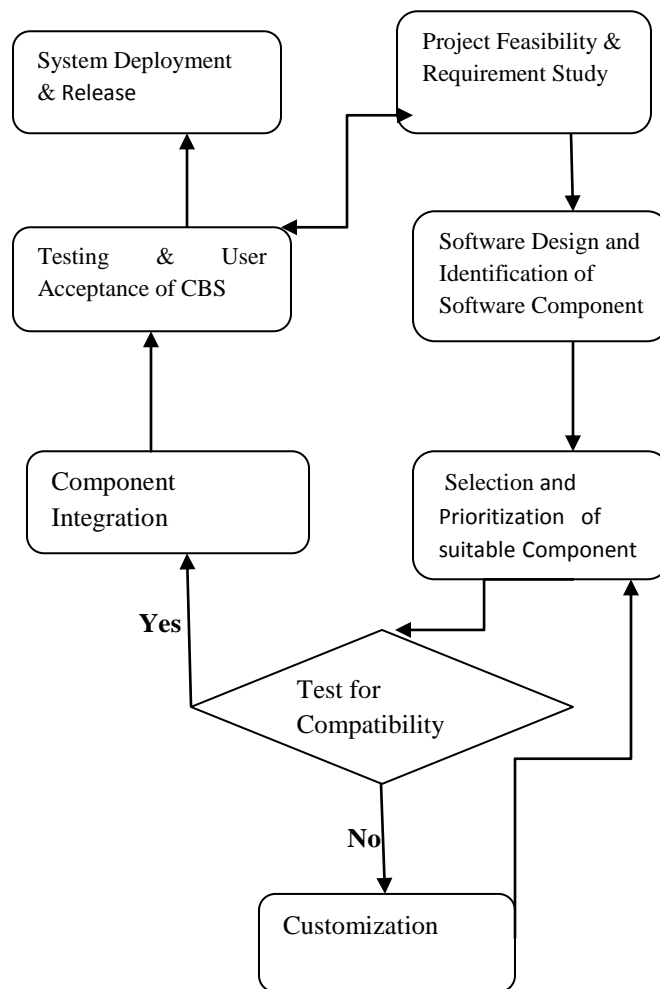

Fig3: Component Based Reuse Model using preexisting component with change.

Good documentation of software component makes easier many of the complex and challenging problem faced by the component integrator in case of reuse of component with change. Simple program structure and small size code play significant role in case of white box nature of component. In case of white box nature of component code structure is visible to the component integrator so reuse of component is much easier and retrieval process of software component is much easier [4].

IV. CONCLUSION AND FUTURE WORK

It has been observed that quality of component based software product solely depends on selection of most suitable software component. The various selection criteria mention in this paper may be of great use as a guiding principle for researchers, component integrators and practitioners. Further correct use of customizable software component can reduce the maintenance effort and improve the security and reliability of the end product up to great extent [11]. Considering the various selection criteria for reusable software component, researcher and practitioners can work together on developing different reusability metrics, which can be further validated and used for more correct estimation for selecting reusable software component.

V. REFERENCES

[1] Arun Sharma, Rajesh Kumar, P S Grover, June 2006., "Investigation of reusability, complexity and customizability for component-based systems", ICFAI Journal of IT, Vol.2 Issue. 1,

[2] Loveleen Kaur, Dr. Hardeep Singh, 2014 "Software Component Selection techniques-A review" International Journal of Computer Science and Information Technologies, Vol. 5, Issue 3,

[3] Nikita Soni , S. K Jha , 2014 "Component Based Software Development: A New Paradigm" , International Journal of Scientific Research and Education ,Volume 2, Issue 6.

[4.]Nasib Singh Gill, "Importance of Software Component Characterization for Better Software Reusability", ACM SIGSOFT SEN Vol. 31, No. 1.

[5].Nasib Singh Gill,"Reusability Issues in Component-based Development", ACM SIGSOFT SEN Vol. 28 No. 6, pp. 30.

[6]. Shrddha Sagar, Prtistha Mathur and Arun Sharma, 2015, "Multi Criteria Selection of Software Components using Fuzzy-AHP Approach",International Journal of Innovative Computing ,Information and Control Vol 11 ,No 3.

[7].S. K.. Jha, 2012 "Quality Management for Component Based Software System", LAMBERT ACADMIC PUBLICATION GERMANY,

[8] Shambhu Jha; R K Mishra, April 2015 "Accessing Software Quality for Component – Based Software through Trustworthiness and Dependability Analysis" , International Journal of Development Research(IJDR) of India, ISSN: 2230-9926 [online] , Volume 5 & Issue 4 , Page: 4259-4261-18,

[9] T. Birgerstaff and C. Richter, 1987 "Reusability Framework, Assessment, and Directions, IEEE Software, vol. 4, March. pp. 41-49,.

[10] V. R. Basili, G. Caldiera, and H. D. Rombach., , 1994 "Goal Question Metric Paradigm," Encyclopedia of Software Engineering, ed. J. J. Marciniak.New York John Wiley & Sons.pp. 528-532.

[11] Arun Sharna , Kirrti Tyagi, November 2011 "Reliability of component based systems: a critical survey" ACM SIGSOFT Software Engineering Notes Volume 36, Issue 6.

[12] Judith Barnard , March 1998 " A new reusability metric for object-oriented software" , Software Quality Journal published by Springer , Volume 7, Issue 1, pp 35-50