Literature Study of Component Based Software Development

[1]Divya Saharan, [2]Yogesh Kumar
[1]Student, [2]Assistant Professor
[1]Department of Computer Science and Engineering
[1]University Institute of Engineering & Technology. M.D.U Rohtak, Haryana, India City,

Abstract: The primary aim of the successful software technology is to provide reliable software but with this also comes the factors such as cost and time. An expansion in the field evaluating cost and time factors and trying to minimize these two factors has led to the development of Component Based Software Development. In this various component are made individually and then they are binded together to form software system. The key factor in this is the fact that since the components are constructed individually they are reused to make other software system. To reuse the components, component identification and selection is the foremost task; for this various technique have been deployed. After the component has been selected, component integration takes place which includes adaptation, testing & validation. The major difference between CBSE and other software system is in its development phase. The main objective of CBSE is to increase quality and productivity & to decrease the cost. It has also been learned that CBSE has not been able to succeed in determining standard principles for component models. In this paper, we have reviewed Component Based Software, its lifecycle & component models and also highlighted the attributes of quality assurance in context of reusability concept.

Index Terms: CBSE, CBSD, Component, Reusability, Quality assurance, Component model (keywords)

INTRODUCTION

Component-based development (CBD) has received significant concentration among software creators, traders and IT unions. [1] CBD is classified as combination of a subclass as well as an addition of the present software engineering techniques. The probability and possibility of augmented product consistency and constancy with shorter development period and low-price stays to boost the current appeal in CBD. A primary and frequently used description from Szyperski [2] states that "A software component is a unit of composition with contractually specified interfaces and explicit context dependencies only. A software component can be deployed independently and is subject to composition by third party."

A component-based system recognizes three attributes that are components, their fundamental platform, and the binding mechanisms of those components. The component model can be defined as a model that defines the standard form and interface between various components. It also maintains the integrity as well as quality of the various components by compelling them in certain standards.

Heinemann and Council [3] describe a component model as: "A component model defines a set of standards for component implementation, naming, interoperability, customization, composition, evolution, and deployment". This characterisation uncovers that a component model includes various angles of the development method, associating with guidelines for the composition and assembly of different components into the software system. To understand about CBD, one must understand that   component itself is well-defined comparatively to a particular component model [4]. Definition: A component model describes standards for 1) properties that discrete components requisitely satisfy and 2) techniques for comprising components.

COMPONENT MODEL LIFECYCLE-

 It consists of four phases namely a) Requirement Gathering b) Analysis & Design c) Implementation & Testing d) Maintenance. In the First phase, requirements are gathered from the clients for software development. Then the components are also analysed for the requirements as it is requisite to check if any of the components required can be taken from the established component repository. The Second phase is Analysis & Design, in this analysis of components i.e. component identification and selection is done effectively. After which the system design is established and specified. The Third phase is Implementation & Testing, in this the emphasis is anticipated on the performance of component to confirm that the revealed behaviour is appropriate. The core structure of the components is unidentified, therefore the appropriate technique for this is BBT (black-box testing). The fourth and the last phase is Maintenance, in this the components individually are checked and maintained and if any fault occurs that particular component is then replaced.

Requirement Gathering

Analysis & Design

Implementation & Testing

Maintenance
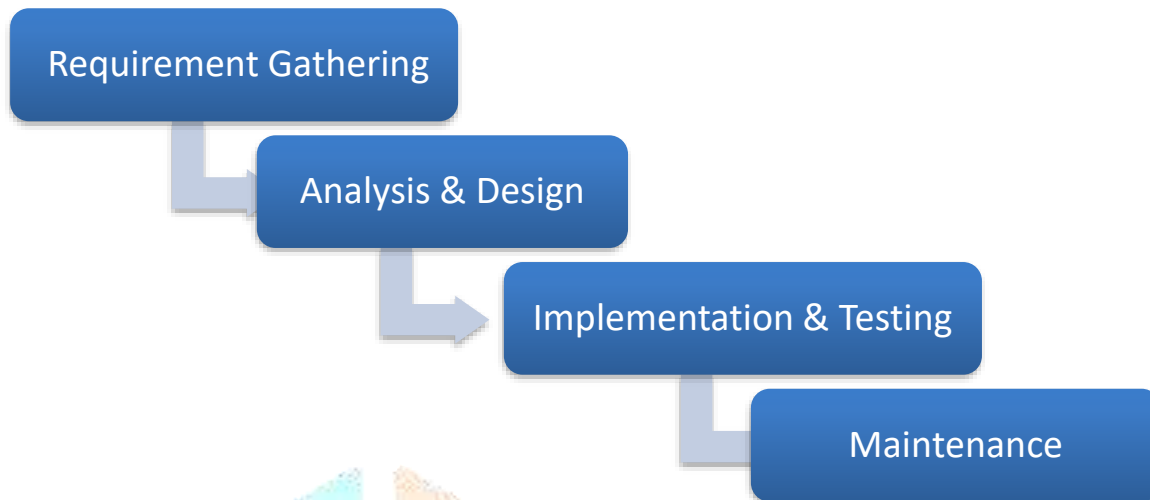
Component Model LIFECYCLE

ADVANTAGES OF CBSD:

1. Improved Quality has been attained with the help of CBSD approach.
2. Higher Reliability is achieved using various components and binding them together via binding mechanism.
3. The concept of usability and reusability has been widely established.
4. The delivery period and the cost factor has been drastically reduced.
5. The efficient and effective administration of complexity is achieved with the help of this approach.
6. Throughput has been escalated exponentially.

DISADVANTAGES OF CBSD:

1. The system progression takes place via binding of various different components. Therefore, the system developer has less control over how the system is progressing.
2. Since we use the same components which have been developed earlier in order to reduce the cost as well as time, hence we need to compromise on our requirement fulfillment.
3. The maintenance cost increases as the components are reused and were developed earlier.

LITERATURE STUDY:

Sandeep Chopra, M.K. Sharma and Lata Nautiyal, "Comparative Study of Different Models in Component Based Software Engineering" (2017) The authors in this paper have discussed that concept of software Engineering and CBSE. This document stated that CBSD upholds developing software systems by choosing reliable, reusable and robust components of software and then assembling them into the particular software architecture. The authors have examined seven CBSE models in this paper and have determined the efficiency of those seven models. Based on various key factors the authors have made comparative analysis in a tabular form. The selection of key factors was based on the reasoning of the researchers. [5]

Muhammad Tahir et. al. "Framework for Better Reusability in Component Based Software Engineering" (2016) It offers the outline of how to develop component based software. This outline helps to attain effective reusability of components by dispensing ease of selection of component using area knowledge and off-the-shelf-components accessibility. It also helps less proficient developer to develop successful component based software system. In this paper, diverse conventional approaches are studied to attain reusability in (CBSE). It is also stated that outline for the CBSE has been presented but the execution of this outline will be done in the forthcoming years to signify that how this prototypical model benefits in attaining the enhanced reusability. [6]

Prateek Jain, "Towards the Adoption of Modern Software Development Approach: Component Based Software Engineering" (2016) This paper explores the CBSE method alongside its categorisation of numerous methods of CBSE. The life cycle development

model of [CBSD] has been anticipated by the component re-usability notion with the thought of decreasing the budget of software. CBSE emphases on using the current components that are being proposed in .NET or sometimes in Java. This paper states that these components are then reorganised as per the necessities of a latest software and these components can be re-used with the assistance of component warehouse that comprises of all components for re-usability. [7]

Abeer Toheed Quadri, Maria Abubakar and Mehreen Sirshar, "Software Quality Assurance in Component Based Software Development – A Survey Analysis" (2015) In this paper the Authors' scrutinize diverse research papers and examines numerous techniques reviewed to ensure software Quality in CBSD. This paper includes the study on how to increase the quality of a component centred software system without causing the properties of quality feature to be decreased. The stated material is acknowledged from literature study. The advances in component centred systems are increasing as the development period, effort and price is reduced drastically by ways of reusing the same old components. Following examination, it has been investigated that in order to attain the quality in a component centred system we require the components that are qualified through software measures as the probability of software quality attributes of system depends majorly on the quality attributes of the fundamental components, combination process and the outline used. [8]

Tassio Vale et. al. "Twenty-Eight Years of Component-Based Software Engineering" (2015) The authors have provided a systematic analysis of CBSE. To do the analysis they have studied 1,231 articles of the time period 1984 to 2012. With the help of evidence drawn by it, this paper addressed five major factors of CBSE: Objectives, Topic of research, Domains of application, intensity of research and methods of research. The principal objectives established are productivity increase, reduce cost and quality increase. The most important principal domain found in this paper were COTS- commercial off the shelf, embedded & distributed system. [9]

Nikita Soni, S. K. Jha, "Component Based Software Development: A new Paradigm" (2014) The author have given a brief introduction about CBSD methodology, its issues, advantages, disadvantages that it holds. They provided the notion of components and component models and highlighted the crux regarding the performance maintainability and quality aspects of CBSD. It introduced the research results carried out by various researchers regarding the basics of CBSD approach. The authors have surveyed about the components, issues, examples and benefits and limitations of CBSD. It also highlights about component models, performance measure of components and quality assurance models that are to be considered while developing a system using CBSD approach. [10]

Josip Maras, Luka Lednicki and Ivica Crnkovic, "15 Years of CBSE Symposium: Impact on the Research Community" (2012) The authors here have given an investigation of 15 years of happenings of CBSE procedures. This paper states that in the time period of 15 years beyond 300 co-authors have underwritten 318 contributions to the CBSE community. The authors have also concluded that CBSE has reached a turning point where countless challenges detailed in initial years have been resolved, which were earlier stated as being unknowable. Throughout this phase, CBSE has been applied to many diverse areas, and has become a fundamental part in most of them. It has been proven that new areas and new technologies compel to new and accepted CBSE methods (for instance dynamic conformations, upgraded and effective certifications, dynamic acclimatisation, diverse types of component models, and many more). This gives an assurance to the persistence in this area for the community. At the same phase, since CBSE become an inherent part of today's software engineering, there is a severe threat that CBSE, as a distinct and single subject, will not get the same importance from the researchers and experts. [11]

Ivica Crnkovic et. al. "A Classification Framework for Software Component Models" (2011) This paper describes the concept of component models and component binding in CBSE. It then further discusses about classification framework which consist of its lifecycle, construction and extra functional properties. It gives the detailed insight about the binding mechanism of various components and their interaction styles. Finally, it gives the detailed survey of Component Models. [12]

S. Mahmood et. al. "Survey of Component Based Software Development" (2007) This paper gives the detailed inspect of CBSE, its benefits, risks & challenges. The author also explained the development process of CBSE and component identification & selection. This paper states various approaches for component identification & selection that are hierarchical approach, clustering approach, iterative approach & knowledge base approach. Further the author explained the component integration & deployment method of CBSE and at the end how component evolution can take place. [13]

Ivica Crnkovic et. al. "Component Based Development Process & Component Lifecycle" (2006) This paper highlights to the fact that the major difference between a component based system and other software system is in the development process. It analysis the basic characteristics of CBSE approach and what impact it would hold on the development process and its lifecycle models. The author discussed about the development process of CBSE which integrated as two ways: the first one as Product Line Development and the other one as COTS based development. [14]

Nasib S. Gill, "Reusability Issues in Component-Based Development" (2003) The author has given an insight regarding the reusability concept in component based system. It explains how various components can be reused in other software and what would be the guidelines for reusability in CBSE. Then the author in this explained the cost benefit analysis for reusing the components. To perform cost benefit analysis the author has also identified reuse metrics which can perform the task. It also stated in the conclusion that due to the reusability concept of components, the productivity as well as maintainability of software has increased. It also represents what are the issues and challenges regarding software reusability for CBSE. [15]

Padmal Vitharana, "Risks and Challenges of Component-Based Software Development" (2003) The authors have discussed the role of stakeholders in CBSD. It states that there are three types of stakeholders that are component developers, application assemblers and customers & developers. Further the author discussed about the component based application assembly lifecycle. It is also explained that to understand the impact each and every stakeholder must analyse its risks and make sound strategies to address all of them. [16]

Wiley Blackwell, "Component Based Software Engineering – New Challenges in Software Development" (2001) This paper at first discuses about the software development and the challenges of the same. It explains in detail about what CBSE is and what is meant by component specification. Further it explains the lifecycle of component based system development and compares it with the waterfall model. It also states that software development specifications must be aligned with the new approach, and various new procedures be developed for the same. The author also discussed about the architecture of CBSE and development in component based system. Then it stated the future scope of CBSE in the coming years. [17]

Xia Cai et. al. "Component Based Software Engineering Technologies, Development Frameworks and Quality Assurance Schemes" (2000) This paper describes the approach of component based system software in detail. It gives an insight of what are known as off-the-shelf components and how they can be developed. The authors also explained the system architecture of CBSE and the component technology of CBSE i.e. CORBA, COM, EJB. The paper gave the comparison of the three-described component technology. Further it discussed the quality assurance for component based software engineering. In it the authors stated the lifecycle of CBSE and its key activities. At the end, they discussed that quality assurance techniques for CBSE are still in developing stage i.e. are not mature because specific characteristics of it are different from the other systems. [18]

CONCLUSION:

This paper summarizes the research findings of numerous researchers in the domain of component based software development. It gives an insight of what is the meaning of component, how various components can be binded together. The motive of this paper is to make the user understand the concept of component and reusability of various different components in the software so that the time and efforts to make a new software can be decreased without affecting the quality and productivity. The Future scope lie in the context of reusability and quality attributes that need to be taken care of while using the components in CBSE.

REFERNCES:

[1] Councill Bill and Heinemann George T. 2000. Component- Based Software Engineering and the Issue of Trust, in Proceedings of International Conference on Software Engineering (ICSE 2000), pp. 661-664.

[2] C. Szyperski. 1997. Component Software: Beyond Object-Oriented Programming. Addison-Wesley Professional.

[3] G.T. Heinemann and W.T. Councill. 2001. Component-Based Software Engineering: Putting the Pieces Together. Addison Wesley Longman Publishing Co.

[4] M.R.V. Chaudron and I. Crnkovic. 2008. Software Engineering: Principles and Practice, third ed., ch.18. Wiley.

[5] Sandeep Chopra, M.K. Sharma and Lata Nautiyal. 2017. Comparative Study of Different Models in Component Based Software Engineering. International Journal on Emerging Technologies (Special Issue NCETST-2017) 8(1): 441-445(2017).

[6] Muhammad Tahir et. al. 2016. Framework for Better Reusability in Component Based Software Engineering. J. Appl. Environ. Biol. Sci., 6(4S)77-81, 2016 ISSN: 2090-4274Journal of Applied Environmental and Biological Sciences.

[7] Prateek Jain. 2016. Towards the Adoption of Modern Software Development Approach: Component Based Software Engineering. Indian Journal of Science & Technology, Volume 9, Issue 32, August 2016.

[8] Abeer Toheed Quadri, Maria Abubakar and Mehreen Sirshar. 2015. Software Quality Assurance in Component Based Software Development – A Survey Analysis. World Academy of Science, Engineering and Technology International Journal Computer and Information Engineering Vol:9, No:2, 2015

[9] Tassio Vale et. al. 2015. Twenty-Eight Years of Component-Based Software Engineering.The Journal of Systems & Software.

[10] Nikita Soni, S. K. Jha. 2014. Component Based Software Development: A new Paradigm. International Journal of Scientific Research and Education, Volume 2, Issue 6, Pages 969-974, 2014 ISSN (e): 2321-7545.

[11] Josip Maras, Luka Lednicki and Ivica Crnkovic. 2012. 15 Years of CBSE Symposium: Impact on the Research   Community. CompArch 2012 June25-28,2012, Bertinoro, Italy.

[12] Ivica Crnkovic et. al. 2011. A Classification Framework for Software Component Models. IEEE Transactions On Software Engineering, VOL. 37, NO. 5, September/October 2011

[13] S. Mahmood et. al. 2007. Survey of Component Based Software Development. IET Software, Vol. 1, No. 2, April 2007

[14] Ivica Crnkovic et. al. 2006. Component Based Development Process & Component Lifecycle. Conference: Conference: Information Technology Interfaces, 2005. 27th International Conference.

[15] Nasib S. Gill. 2003. Reusability Issues in Component-Based Development. ACM SIGSOFT Software Engineering Volume 28 Issue 4, July2003.

[16] Padmal Vitharana. 2003. Risks and Challenges of Component-Based Software Development. Communications of the ACM, Vol. 46 No. 8, Pages 67-72

[17] Wiley Blackwell. 2001. Component Based Software Engineering – New Challenges in Software Development. Issue TOC, Volume 2, Issue 4 Winter 2001 Pages 127–133.

[18] Xia Cai et. al. 2000. Component Based Software Engineering Technologies, Development Frameworks and Quality Assurance Schemes. Conference: Software Engineering Conference, 2000. APSEC 2000. Proceedings. Seventh Asia-Pacific.

[19] Meenu, Yogesh Kumar. 2015. Comparative Study of Automated Testing Tools: Selenium, SoapUI, HP Unified Functional Testing and Test Complete in International Journal of Emerging Technologies and Innovative Research, Volume 2, Issue 9, Sept-2015, Pg. 42-48.

[20] Yogesh Kumar, Neeraj Varshney. 2017. Comparative analysis of software size estimation techniques in project management, in International journal for research in applied science & engineering technology, Vol. 5, Issue VIII, Aug-2017. Pg. 1470-1477.

[21] Mani, Yogesh Kumar. 2018. Software development life cycle models, in International Journal of Emerging Technologies and Innovative Research, Vol 5, Issue 2, Feb-2018. Pg. 107-113.

[22] Tannu, Yogesh Kumar. 2014. Comparative Analysis of Different Software Cost Estimation Methods, International Journal of Computer Science and Mobile Computing, Volume 3, Issue 6, 04 July 2014, pg.547-557.

[23] Yogesh Kumar. 2014. A Review on Effort Estimation Techniques used in Software Projects, in International Journal of Computer Science & management Studies, Volume 14, Issue 3, March 2014. Pg. 25-31.