UGC APPROVED JOURNAL - 47746

© INNOVATIVE RESEARCH THOUGHTS | Refereed | Peer Reviewed | Indexed

ISSN : 2454 – 308X | Volume : 03 , Issue : 10 | October – December 2017

# Component Based Software System Metrics : A Review

**Neha Taneja**
**nehataneja89@gmail.com**

**Abstract:** Measurements have been used to quantify the amount to which a system, component, or process affects a certain property. Software quality can't be judged without metrics. Predictability of software quality is improved by using it. There are a variety of ways to create a software metric. Object-oriented, distributed, and component-based techniques are all examples of these approaches. Developing software using component-based engineering is a relatively new practise. It is the primary goal of component-based metrics to facilitate reuse. It also saves the development costs and time. Such measures have been used to evaluate quality and to control related risks. Component-based metrics are the focus of this research. Several component-based metrics were compared in this study. This comparison is based on the software's functional and non-functional aspects.
The advantages and disadvantages of the suggested system for software development are examined.

***Keywords: CBSE, Software development, Software Quality, Software Metrics, Reusability, Complexity.***

## I. INTRODUCTION

“The complexity of software development is on the rise right now. Traditional techniques and technologies are no longer enough for software development. As a consequence, software's overall quality suffers. It is the most recent method for developing software, Component Based Software Engineering. Design and construction criteria are adhered to throughout the process of component-based software engineering. It takes advantage of a computer-based system's reusable software component. It is made up of both commercially available and in-house components. Components created by a third party are referred to as commercial-off components. Figure 1 depicts an example of one of these components in action. Components of this kind have already been thoroughly tested, and their code has not been altered [2, 3]. On the basis of their functionality, quality, cost, and degree of change required [2], these COTS components may be selected. Software metrics play an important role in enhancing software quality and risk management since they influence several areas of software complexity. Metrics have been shown to be beneficial in providing information about the quality of software. Reusability, portability, and the ability to comprehend [4] are among these properties. As a result, the developer is able to identify potential risks and take remedial measures. Metrics play an important function in bringing attention to the system. Increasing the end product's quality is how it's accomplished. Metrics may be used to build a large-scale software system. [5] To achieve abstraction at a high level, metrics have the capacity to promote reusability [3, 4]. Figure 2 depicts one approach to software development that makes use of already-existing components. Software quality is being improved as a result of this. It is enhancing the software's maintainability and productivity. It speeds up software development by lowering the amount of time and effort required [1]. In figure 1 the different reusable software components have been represented. Active X controls, Active X DLL, Active X exe are part of visual studio 6.0. Assemblies are available in modern .net platform that is used for usability. Component object models are providing reusability by embedding one application in another application.

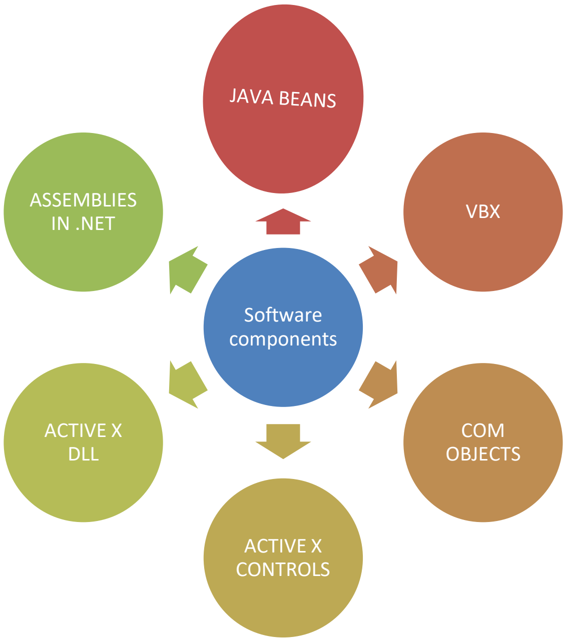

Figure 1: Software Components

## III. EXISTING RESEARCHES

In this field, several researches have been executed related to component based software metrics. Some of them are given below:

In 2014, Sachin Kumar et al.'s [1] proposed a metric which is developed using black box components. They determine the coupling complexity of software.

In 2003, Model Joaquina et al.'s [2] has designed the CQM model (component Quality Model). In this they evaluate component quality on four dimensions and have main focus on usability metrics.

In 2014 Chander Diwaker, et al.[3] explained the metrics used in component based software engineering. In their research they discussed the metrics which are applied in software engeenring.

In 2012, Prakriti Trivedi and Rajeev Kumar [4] provided a software matrix set to check the interconnection among the software components and its function. By using software component reusability, they provide a software metrics to evaluate the software quality. To make approximate calculation of software quality using software component reusability they provided a software metrics

In 2012, MajdiAbdellatief et al.'s [5] shows the dependency between components is major factor affecting the structural design of Component Based Software System (CBSS). For this they provided dependency metrics based on components information.

In 2003,TullioVernazza, et al. [6] wrote on defining metrics for software components.

They explained the metrics which are used with software elements.

In 2012, MajdiAbdellatiefab, et al. [7] did research on component-based software system dependency. metrics based on component information flow measurements.

In 2012, DanailHristov, et al. [8] provided his views on structure of software reusability metrics. They explained that software reusability metrics are capable to develop the component related software.

In 2015, N. Gehlot et al. [9] discussed the dynamic inheritance coupling metric-design. In this research, it has been analyzed to enlarge the reusability.

In 2012,V. Subedha, et al. [10] proposed on dynamic component design. They also considered the reuse and reusability metrics library. These metrics are to reuse the software components.

In 2014, PoojaRana et al. [11] discovered different types of complexity metrics based on component. Component information flow metrics and component coupling metrics are the two sets of metrics which are proposed by them.

In 2007, V.LakshmiNarasimhan, et al. [12] analyzed the various metrics. They also evaluated the existing researches. Theses exiting researches are proposed by many researchers. After analysing the previous paper, they conclude and made some useful results.

In 2011,P. Edith Linda, et al. [13]provided a review on metrics for component based measurement tools.

In 2012, Vinay Tiwari, et al. [14] explained the open source software and reliability metrics.

In 2015, A.Aloysiusi and K.Maheshwaran [15] suggested further improvement of various component based metrics. They took a review of existing researches and concluded that there is requirement to expend the component related metrics.

In 2009, V. Lakshmi Narasimhan, et al. [16] did research on evaluation of a suite of metrics for component based software engineering.

In 2005,Hironori Washizaki, et al.[17] wrote on a metrics suite for measuring reusability of software components.

In 2014,K.P. Srinivasan, et al.[18]discussed the software metrics validation methodologies in software engineering.

## IV. COMPONENT BASED SOFTWARE METRICS

Component based software metrics calculate the quality of component based software. They also manage the risk related to component based software. In order to formulate a sufficient metrics, first of all it is necessary to classify the chief feature of task. The second step is to divide them or break down into sub-feature [15]. Then these refined sub features are appearing in traits. These traits are related to metric definitions. Those are capable to obtain the wanted metrics.

### *A. Metric Suite:*

This metrics structure is denoted as a tree. There are two types' metrics in component based software engineering:

(a) Non-functional Metrics (b) Functional Metrics.

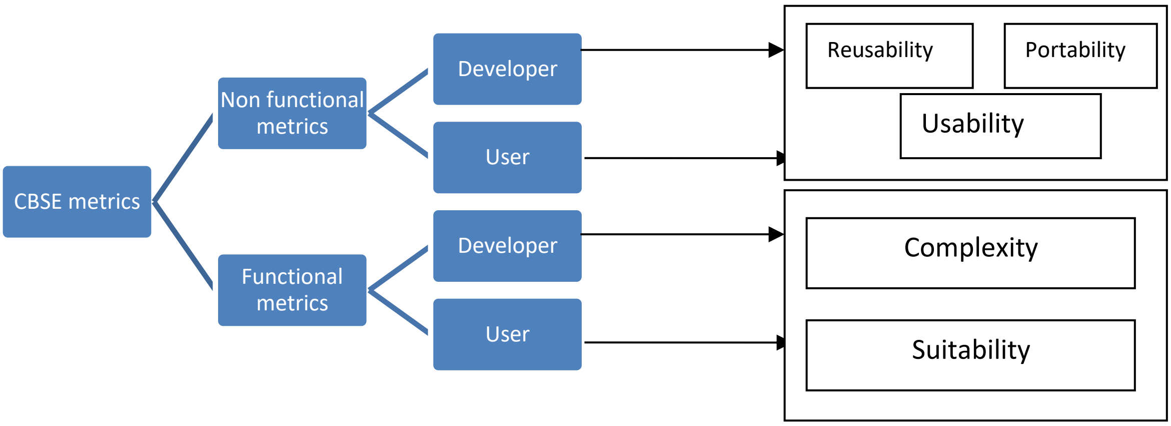

**Figure 3: CBSE Metrics Tree**

In this we are having various metrics based on various dimensions.

i) *Suitability Metrics:* The degree to which components fulfil the confine requirement. The component suitability is the nature that can be determined after the component gets installed [16]. For suitability there are two types of metrics based on different perspective:

Required Functionality (RF): It comes under producer perspective. In this only required functionality need to be checked that must be satisfied.

$$RF = \frac{\text{No. of necessary functionality given by the components}}{\text{Total no. of functionalities necessary by the component based system}}$$

Increase in the value of RF will increase suitability of component.

## SCOPE OF RESEARCH

This paper has presented overview over component based software metrics. Here several metrics for one characteristics of component based software have been discussed. In order to consider the complexity of component at interfaces three metrics have been considered. CCCM (BB) is considered as best metrics. CCCM has been considered as black box testing metrics". As it is interface complexity metrics it does not consider testing, considers capability and effort. This paper has presented the significance of reusability metrics". It has considered both functional and non-functional aspects of software system. Such metrics are found helpful to confirm the quality characteristics of software. Research has made review on metrics dependent on component. There are very few reusability metrics that have been validated and defined clearly.

## REFERENCES

[1] Sachin Kumar, Pradeep Tomar, Reetika Nagar, SuchitaYadav,"Coupling Metric to Measure the Complexity of Component Based Software through Interfaces", April 2014.

[2] Joaquina Martín-Albo, Manuel F. Bertoa, Coral Calero, Antonio Vallecillo, Alejandra Cechich and Mario Piattini "CQM: A Software Component Metric Classification Model" IEEE TRANSACTIONS JOURNAL, 2003.

[3] ChanderDiwaker, Sonam Rani, Pradeep Tomar "Metrics Used In Component Based Software Engineering", ICFTEM-2014.

[4] Prakriti Trivedi, Rajeev Kumar, "Software Metrics to Estimate Software Quality using Software Component Reusability", IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 2, No 2, March 2012.

[5] MajdiAbdellatief∗, Abu BakarMd Sultan, Abdul Azim Abdul Ghani1, Marzanah A. Jabar, "A mapping study to investigate component-based software system metrics", jo u rn al homepage: www.elsevier.com, 5 October 2012.

[6] TullioVernazza, GiampieroGranatella, Giancarlo Succi, Luigi Benedicenti, Martin Mintchev, "Defining metrics for software components", July 2003.

[7] MajdiAbdellatiefab, Abu BakarMd Sultana, Abdul AzimAbdGhania, MarzanahA.Jabara, "Component-based Software System Dependency Metrics based on Component Information Flow Measurements" ICSEA: The Sixth International Conference on Software Engineering Advances, 2012.

[8] DanailHristov, Oliver Hummel, MahmudulHuq, Werner Janjic, "Structuring Software Reusability Metrics for Component-Based Software Development", ICSEA: The Seventh International Conference on Software Engineering Advances, 2012.

[9] N. Gehlot and J.Kaur, "Dynamic Inheritance Coupling Metric-Design and Analysis for Assessing Reusability", Int. J. Software Engineering Technology and Applications, Vol.1, No.1, PP. 118-133, 2015

[10] V. Subedha, S. Sridhar, "Design of Dynamic Component Reuse and Reusability Metrics Library for Reusable Software Components in Context Level", February 2012.

[11] PoojaRanaRajender Singh, "A Study of Component Based Complexity Metrics", November 2014.

[12] V. L. Narasimhan and B. Hendradjaya, "A New Suite of Metrics for the Integration of Software Components", 2007

[13] P. Edith Linda, V. ManjuBashini, S. Gomathi, "Metrics for Component Based Measurement Tools", International Journal of Scientific & Engineering Research Volume 2, Issue 5, May-2011.

[14] Vinay Tiwari, Dr. R.K. Pandey, "Open Source Software and Reliability Metrics", International Journal of Advanced Research in Computer and Communication Engineering Vol. 1, Issue 10, December 2012.

[15] A.Aloysius and K.Maheswaran, "A Review on Component Based Software Metrics", 22 January 2015

[16] V. Lakshmi Narasimhan, P. T. Parthasarathy, and M. Das, "Evaluation of a Suite of Metrics for Component Based Software Engineering (CBSE)", 2009

[17] Hironori Washizaki, HirokazuYamamotoandYoshiakiFukazawa,"A Metrics Suite for Measuring Reusability of Software Components", 2005.

[18] K.P. Srinivasan1 and T. Devi, "Software Metrics Validation Methodologies InSoftwareEngineering", International Journal of Software Engineering & Applications (IJSEA), Vol.5, No.6, November 2014.