Journal of Information and Operations Management
ISSN: 0976–7754 & E-ISSN: 0976–7762 , Volume 3, Issue 1, 2012, pp-191-193
Available online at http://www.bioinfo.in/contents.php?id=55

IMPACT OF COHESION ON RELIABILITY

YADAV A. AND KHAN R.A.

Department of Information Technology, Babashaeb Bhimrao Ambedkar University, Lucknow, India
*Corresponding Author: amitabha.engg@yahoo.com

Received: December 12, 2011; Accepted: January 15, 2012

Abstract- Modern computerization and electronic systems typically contain significant amount of software. Over the last two decades, modern society has become increasingly dependent on these significant software systems, resulting in demand for increased software reliability. OO software characteristics impacts on software reliability. An effort in this paper has been made to notify whether software reliability can be increased by increasing cohesion of object oriented software. Cohesion measurement has been used since decades for quality assessment, fault proneness etc. In object oriented software, cohesion is often measured at class level. High cohesion and low coupling among classes are aimed to reduce complexity. Higher Complexity makes the system more error prone, difficult to understand and unreliable. The paper examines the role of cohesion in making the software less complex. It analyses the impact of cohesion on complexity and reliability. An inference has been made that more cohesion decreases complexity which results reliable system.
Keywords- Cohesion, Reliability, Complexity

Citation: Yadav A and Khan RA (2012) Impact of Cohesion on Reliability. Journal of Information and Operations Management ISSN: 0976–7754 & E-ISSN: 0976–7762, Volume 3, Issue 1, pp-191-193.

Introduction

To produce high quality object oriented software, a strong emphasis on design aspects is necessary. Object-oriented languages stand in a very high peak due to the increase in the productivity of software and the need of higher reliability for the developed system. Object oriented (OO) design is the only phase where programmer has an authority of taking how much design construct is needed for designing reliable software. Several incidences have been reported because of the unreliable software. It has been well defended by the experts that object oriented design provides the most appropriate features to improve quality of software by improving its factors.

Reliability is one of important factor of quality and there is an increasing demand of reliable software. Dependency on computer system increases the problems of reliability failure. Software reliability breaks has been observed in many cases including failures in Tharc 25 radiation therapy machine, in a London ambulance service, change in the three lines of code in a single program [1]. These software reliability breaks has been reported and investigated by several studies [2]. Software reliability breaks leads to transient failures in software systems, performance degradation of the software or crash/ hang failure or both, memory bloating and leaking, unreleased file locks, data corruption, storage space fragmentation, and accumulation of round-off errors. Thus, it is concluded that there is need of higher reliability.

Reliability in itself is affected by several factors. Sixteen factors have been identified [3], among which complexity has been taken as a major factor to estimate reliability [4]. Complexity as being a major factor has been considered to further elaborate and address in terms of reliability. Complexity decreases reliability. The approach is to control complexity by controlling object oriented constructs. There is no universally accepted direct measure of complexity [5]. Complexity of software can be decreased by adjusting design characteristic. Hence, it is necessary to maintain complexity of OO software.

Once complexity is controlled, reliability will also increases to some extent. Design constructs are used to control complexity of OO design. Object oriented design supports inheritance, encapsulation, coupling, cohesion. Cohesion will be taken as a major construct to maintain complexity of software design in this paper.

The following section summarizes the related work. Section II describes cohesion of object oriented design. Section III presents correlation between Cohesion, complexity and reliability. Impact of cohesion on complexity and reliability is included in section IV. Section V presents findings and future work. At last paper concluded in section VI.

Related Work

In 2010, Michael Bowman, Lionel C. Briand, Yvan Labiche proposed an approach based on a multi-objective genetic algorithm (MOGA) which uses class coupling and cohesion measurement for defining fitness functions. Authors work has some similarity with refactoring. Most of the work is done on source code refactoring, though it is believed that refactoring achieves higher levels of abstraction, such as refactoring of UML models [6].

In 2008, Andrian Marcus, Denys Poshyvanyk discussed conceptual cohesion of classes for fault prediction in object oriented system. Authors proposed a new metric conceptual cohesion of classes (C3) to measure cohesion of classes in OO software based on the analysis of the unstructured information in the source code. Authors used case study which compares the C3 metric with an extensive set of existing metric and uses them to construct models that predict software faults [7].

In 2006, Jean-François Gélinas, MouradBadri, Linda Badri measured cohesion for aspect oriented software development based on dependencies analysis. Author proposes a new metric aspect oriented metric (ACoh) and compare it with existing aspect oriented metrics using several case studies [8].

In 2005,Andrian Marcus, Denys Poshyvanyk proposed a new set of measures for the cohesion of single classes with in OO software. Authors have taken case study which compares the new measure with an existing metrics. Authors also discussed existing approaches to cohesion measurement in object oriented software which are largely based on the structural information of the source code. OO analysis & design methods try to control complexity. High cohesion & low coupling for classes are design constructs aimed to reduce the system complexity [9].

In 1998, Ken S.Lew, TharamS.Dillon and Kevin E.Forward, proposed a software complexity metric which includes internal and external complexity of software. Authors shown that complexity measure are useful in quantifying the design and provide a guide for designing reliable software. Author discussed software complexity and its impact on software reliability. Complexity of software affects reliability. To produce reliable software, its complexity must be controlled [4].

In 1997, Lionel C. Briand, John W. Daly, and Jürgen K. Wüst provided a framework for coupling measures in object oriented system. The proposed framework aimed to be exhaustive and integrates new ideas with existing measurement frameworks. Authors also used the framework to review the state of the art [10].

According to literature survey, it is brought out that cohesion can be taken as a measure construct to reduce complexity of object oriented software. Cohesion and complexity are closely related with other. Complexity of software affects reliability. Complexity is factor of reliability. Higher complexity decreases the reliability of software. High cohesion makes the software more reliable and less complex.

Cohesion

Cohesion is one of the important design construct among encapsulation, inheritance, coupling and polymorphism. Cohesion is defined as a measure of the strength of functional association between elements within a module. It is universally accepted that strong cohesion is desirable for good software design because it makes the code easy to understand and modify. Cohesion also helps in supporting low coupling between modules.

Several metrics have been proposed to measure cohesion. LCOM (lack of cohesion metrics) is the first metric developed by S.R Chidamber and C.F Kemerer in 1991[11]. After that several more definitions of LCOM have been proposed and still research is going on [11] Chidamber -Kemerer Defines LCOM metric is a value of dissimilarity of the methods in a class [10]. LCOM defined as follows:

Let a class C_1 having methods M_1, M2……M_n.

Let (M_i) be the method which uses set of instance variables (I_i). There are n such sets {I_1}…… {I_n}.

Let P be the set containing disjoints sets and Q be the set containing common sets.

$P = \{(I_i, J_j)\ |I_i \cap J_j = \Phi\}$

$Q = \{(I_i, J_j)\ |I_i \cap J_j \neq \Phi\}$

LCOM= |PI-IQ ,Iif P>Q otherwise 0.

Correlation between Cohesion, Reliability & Complexity

While developing software, software engineers and developers have a responsibility to take care of high quality of software. Reliability is one such attribute of quality for which developers has to take more care. High reliability is possible only when there is high cohesion and low coupling [12]. According to literature survey very less work is done in the field of software reliability in respect of OO design. Higher cohesion increases reliability of software by making software more qualitative.

Reliability is the property of referring 'how well software meets its requirements' and also 'the probability of failure free operation for the specified period of time in a specified environment' [13]. Reliability is affected by complexity. High complexity decreases the reliability of software. Complexity of a system can be controlled by increasing cohesion at the time of designing the software [14]. Strong cohesion maintains the complexity to some extent. Cohesion and complexity are closely related to each other. It is concluded that less complex system increases reliability of software by maximizing cohesion and minimizing coupling in the system. Fig 1 gives a pictorial representation of correlation among cohesion, complexity and reliability.

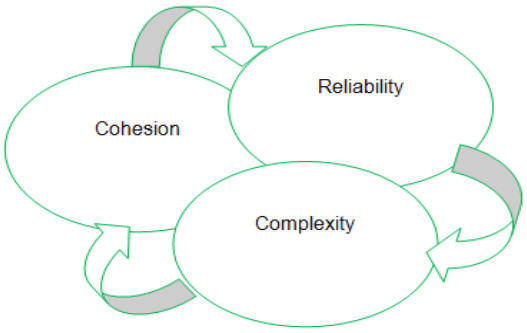

Fig. 1- Correlation between Cohesion, Complexity and Reliability

Impact of Cohesion on Complexity & Reliability

Cohesion is strongest design construct of OO design. Tight cohesion increases reliability by decreasing complexity [7]. Cohesion affects complexity negatively and affects reliability positively shown in table 1. Reliability is directly proportional to cohesion and complexity is inversely proportional to cohesion.

$$Coh \propto R \quad \text{---------- (1)}$$

$$Coh \propto 1/C \quad \text{---------- (2)}$$

From equation 1 and 2,

$$Coh \propto R/C \quad \text{---------- (3)}$$

Where,

Coh= Cohesion
R=Reliability
C=Complexity

Table 1- Impact of Cohesion on Complexity and Reliability

	Complexity	Reliability
Cohesion	-ve	+ve

Findings and Future Work

Complexity is major factor of reliability which can be controlled by maximizing cohesion. Complexity has a negative impact on reliability of software. The paper presents some advances towards the correlation of cohesion and complexity in design phase of an object oriented software system with reliability. The design constructs cohesion is strongly connected with reliability attribute complexity. The cohesion is closely related to complexity which expresses in terms of reliability point of view. Cohesion decreases complexity of the system which makes the system easy to understand and maintain. It is necessary to maintain cohesion of software in design phase to control complexity to make the system highly reliable.

The contribution of the work is summarized as follows:

A relation between design principal cohesion and reliability factor complexity has been taken into consideration.

Impact of coupling on reliability is shown.

Impact of coupling on complexity is shown.

A conclusion has been made that high cohesion makes the software less error/fault prone. High cohesion decreases complexity of the software which results in ceasing severe damages caused by software reliability break.

In future, complexity will be estimated in terms of cohesion of object oriented software with the help of a case study of class hierarchy. Reliability of objected oriented software will be judged on the basis of cohesion because complexity decreases with increase of cohesion.

Conclusion

Cohesion is a measure of the strength of functional association between elements within a module. It is saying that coupling should always be minimum and cohesion should always be maximum because high cohesion increases reliability and quality of software. Complexity is a major factor of reliability and it negatively affects reliability. By controlling complexity, reliability of object oriented software can be increased. Complexity can be controlled by making the system highly cohesive. Higher the cohesion lesser will be the complexity and grater will be the reliability of the system.

References

[1] Yadav A. and Khan R.A. (2009) *International Journal of Computing Science & Communication Technologies,* 1(2),172-179.

[2] Singh H., Cortellessa V., Cukic B., Gunel E. and Bharadwaj V. (2001) *IEEE,* 12-21.

[3] Yadav A. and Khan R.A. (2010) *CSI communication,* 29-32.

[4] Lew K.S., Dillon T.S. and Forward K.E. (1988) *IEEE Transaction on Software Engineering,* 14(11), 1645-1654.

[5] Singh S., Kahlon K.S. and Sandhu P.S. (2010) *IEEE*.

[6] Bowman M., Briand L.C. and Labiche Y. (2010) *IEEE Transactions on Software Engineering*,1-50.

[7] Marcus A. And Poshyvanyk D. (2008) *IEEE Transaction on Software Engineering,* 34(2), 287-299.

[8] Gelinas J.F., Badri M. and Badri L. (2006) *Journal of Object Technology,* pp. 97 – 114.

[9] Marcus A. and Poshyvanyk D. (2005) 21st *IEEE International Conference on Software Maintenance*.

[10] Briand L.C., Daly J.W. and Wüst J.K. (1997) *4th International Software Metrics Symposium,* 43-53.

[11] Chidamber S.R and Kemerer C.F. (1991) OOPSLA'9, 197-211.

[12] Succl G., Pedrycz W., Djokic S., Zuliani P. and Russo B. (2005) *Empirical Software Engineering, Springer Science,* 81–103.

[13] Khan R.A, Mustafa K. and Ahson S.I (2004) *Operation Profile-A key Factor for Reliability Estimation,* pp.347-354.

[14] Dallal J.A. (2010) *International journal of computers,* 2(4),45-52.