Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Technology 4 (2012) 536 – 542

Procedia Technology

C3IT-2012

Analyzability Quantification Model of Object Oriented Design

Suhel Ahmad Khan[a], Raees Ahmad Khan[a]

[a]*Department of Information Technology, Babasaheb Bhimrao Ambedkar University (Central University), Lucknow, 226025, India*

Abstract

In this paper a significant effort is made to quantify analyzability using object oriented software. Analyzability is the key attribute of maintainability for high quality products. A metric based model for analyzability quantification has been proposed by establishing the relation between analyzability and object oriented design constructs. The proposed model is empirically validated and statistical significance of the study discusses the high correlation for model acceptance. Design complexity of software is also an influencing factor of analyzability with negative impact.

Keywords: Analyzability; Object oriented Design; Complexity; Design Metrics;

1. Introduction

Maintainability is one of the most significant characteristics of software quality. Most of software companies splurge 60 to 70 percent of resources for correcting, adopting and maintaining the existing software [1]. Most of companies spend over 70 percent amount on testing maintenance of the software to control the quality [2]. The ISO/IEC 9126 standard describes a model for software product quality that dissects the overall notion of quality into 6 main characteristics: functionality, reliability, usability, efficiency, maintainability, and portability [3].Maintainability measurement helps to analyze the maintenance effort and easiness of software at design level. The maintainability definition according to IEEE glossary of Software Engineering is *"the ease with which a software system or component can be modified to correct faults, improve performance or other attributes, or adapt to a change environment"*. Software maintenance accounts for more effort than any other -----

doi:10.1016/j.protcy.2012.05.085

software engineering activity. The maintainability of software is not possible directly but with the help of their internal characteristics measurement. Early estimation of maintainability helps to utilize its attributes more efficiently to control/improve the quality of software. The design phase quantification of software maintainability is more feasible for software development and maintenance economically. The proposed model for analyzability quantification point outs the probable influences of design constructs and also discusses the impact of analyzability with design complexity.

2. Maintainability Characteristics

Maintainability is one of the most accurate quality indicators. Accurate measurement of maintainability is an indicator of improved designing, high quality product and low maintenance cost. Characteristics of software product quality central to the quality model of ISO 9126 is its breakdown of the notions of internal and external software product quality into 6 main characteristics which are further subdivide into a total of 27 quality sub characteristics [4]. According to the ISO/IEC 9126-4:2004 standard, maintainability can be subdivided into four measurable sub characteristics: Analyzability, Changeability, Stability and Testability & related definitions are depicted in Table 1[5].

Table 1: Maintainability Attributes Definition Table

Maintainability Characteristics	**Definitions**
Analyzability	According to ISO 9126: 2001, 6.5.1, the analyzability is defined as the capability of the software product to be diagnosed for deficiencies or causes of failures in the software, or for the parts to be modified to be identified [6, 7]. To find out the deficiencies at early stage or early identification of location where failure occurred is a valuable effort to mitigate the problem.
Changeability	M. Ajmal Chaumun discuss *Changeability is key in application areas such as telecommunications, in which software systems are evolving at a rapid pace. Moreover, there are organizations which do not develop the software they operate, but purchase it. They are less interested in analyzability, testability, and stability, but primarily in the software's ability to sustain an on-going flow of changes, that is, in its changeability* [8]. If the design patterns increases the changeability will get affected.
Stability	Stability is defined as the attributes of a software product that have an influence on the risk of unintended consequences as a result of modifications. At design time modules are interconnected to each other and the changes made may affect the whole or part of the design. The stability of design is concerned with how resistance of the design against the interclass propagation [9].
Testability	As per IEEE standard glossary, testability is about the degree to which a system or component facilitates the establishment of test criteria and the performance of tests to determine whether those

	criteria have been met: and the degree to which a requirement is stayed in terms that permit the establishment of test criteria and performance of tests to determine whether those criteria have been met. Testability is software quality characteristics whose major part is concerned with defect identification and removal for improved maintainability and test cost control [10].

3. Analyzability and Complexity

According to the definition of Analyzability, it's a process to diagnose deficiencies or causes to failure. Analyzability discusses the effort estimation and reason of break down. The definition can be summarized as follows:

1. Effort to diagnose & rectifying the defects
2. Causes of failure

For any complex software application or design, the effort estimation or rectification of causes of failure is complicated. Complexity is a key factor for secure design up to certain optimal point. Due to enough complexity the things are harder to understand and possibility of defects increases. To diagnose the deficiencies easily is only possible when design complexity kept optimal. Beyond that optimal point as complexity increases, the analyzability of software decreases from that certain acceptance level. There is need to control/adjust the design complexity for better analyzability. The more complex application is facilitating humanity like space shuttle program, but due to enough complexity the analyzability of software are difficult [11]. The effects of complexity and analyzability with quality and its attributes are discussed in Figure 1. From the figure 1 the following inference can be made.

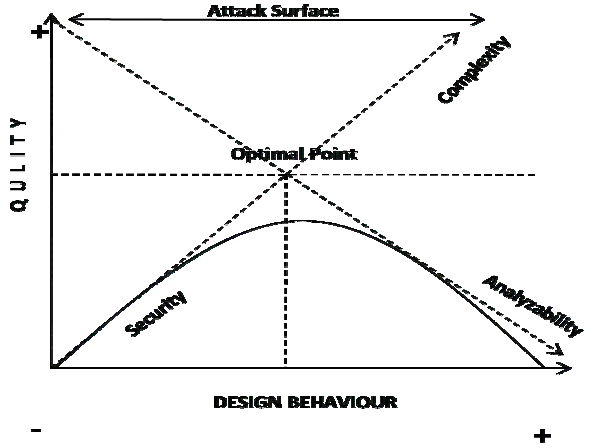


Figure 1: Impact diagram of complexity and analyzability

$$A \propto \frac{K}{C} \quad ¥ \quad A > 0, C > 0,$$

Where C= Complexity
A= Analyzability

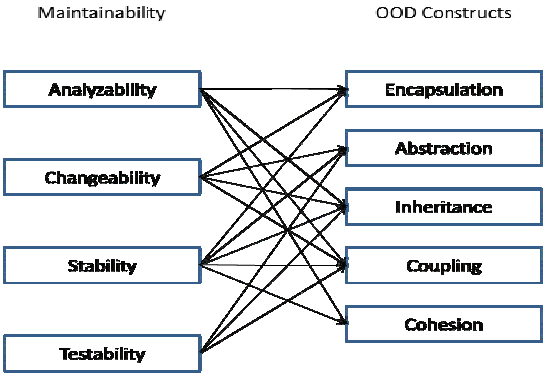
3. Model Development

In order to provide the significant and improved measurement of object oriented software development products, a concerted effort to find quantifiable way to relate measurable object

oriented characteristics to the high level desirable software quality attributes is required.

4.1 Quantifying Statement between Analyzability Attributes & OOD Constructs

It is evident from literature survey that there is no known comprehensive and complete model or framework for evaluating the analyzability of designs developed using an object oriented approach based on its internal design property [12, 13, 14]. This model has the low level design metrics well defined in terms of design characteristics. The set of empirically identified and weighted object oriented design properties are used to assess the analyzability to improve maintainability of software. It is observed that each characteristics having positive or negative impact on factors affects the analyzability of the object oriented software. An exhaustive review correlating object oriented design constructs with analyzability to control maintainability is depicted in Figure 2[8, 9, 10, 15].



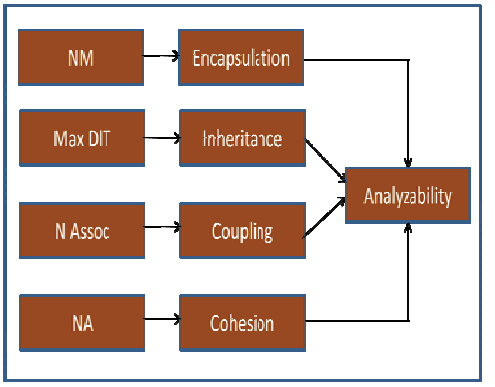


Figure 2: Correlation diagram, Figure 3: Analyzability Quantification Model

4.2 Identification of Object Oriented Design Metrics & Data Collection

Researchers have proposed several object oriented metrics in past decades for qualitative assessment of object oriented design [17, 18, 19]. These metrics are valid and widespread accepted by a large community of researchers and developers due to its accuracy. After a regress review of these accepted metrics some of existing metrics which are well suited for object oriented designing & required dataset has been taken from Genero [17].These metrics are helpful for qualitative and quantitative assessment of analyzability of software design for improved results for maintainability.

4.3 Establishing Correlation

The figure 3 describes the quantification process of software analyzability. The contextual impact relationship between analyzability and object oriented design constructs has been established. The values of these metrics can be easily identified with the help of class diagram. The quantifiable assessment of analyzability is very helpful to achieve maintainability index of software design for low cost maintenance. In order to establish a model for analyzability, multiple linear regression techniques have been used. The proposed multivariate linear model is as follows:

$$Y=\alpha+\beta1*X1+\beta2*X2+\beta3*X3+\text{---------}\ \beta n*Xn\text{-----}\quad Eq(1)$$

Where Y is dependent variable and X1, X2…….Xn are independent variables. β1, β2…… βn are the coefficient of the independent variables. The data is taken for this model is from [17] that is a controlled experiment of 28 class diagrams denoted as (D0 to D27). As per Equation(1) analyzability is taken as independent variable. Using these data, the coefficient calculated for inheritance, coupling and cohesion to show the relationship with design properties. Equation 2 summarizes the computational formula for analyzability with the component weightage.

Analyzability= 1.078 -.029 NM+.659 Max DIT + .228 N Assoc +.064 NA

Analyzability=1.078-.029Encapsulation+.659Inheritance+.228Coupling+.064 Cohesion----------(2)

4.4 Statistical Significance of Model

The coefficient table (2) presents the statistical significance of independent variables. A linear regression relationship has been established between dependent variable and independent variables to check whether it is statistically significant or not. The coefficient table 2 and summary table 3 describes that the metrics are statistically significant at a significant level of 0.05(equivalent to a confidence level of 95%).

Table 2: Coefficient Table

Model	Unstandardized Coefficients		Standardized Coefficients	T	Sig.	95.0% Confidence Interval for B	
	B	Std. Error				Lower Bound	Upper Bound
Constant	1.078	.466		2.315	.068	-.119	2.275
Encapsulation	-.029	.038	-.521	-.763	.480	-.126	.068
Inheritance	.659	.366	.632	1.800	.132	-.282	1.599
Coupling	.288	.217	.578	1.328	.242	-.270	.847
Cohesion	.064	.063	.636	1.012	.358	-.098	.226

a. Dependent Variable: Analyzability

Table 3: Summary Table

Model	R	R Square	Adjust R Square	Std. Error of the Estimate	Change Statistics				
					R Square Change	F Change	df1	df2	Sig. F Change
1	.954	.910	.838	.66864	.910	12.674	4	5	.008

Model Summary : Predictors: (Constant), Cohesion, Inheritance, Coupling, Encapsulation

5. Model Validation

This part of the paper reviews, how well the model effectively quantifies the analyzability of object oriented design using class diagram. For validation of model, same set of data is collected with original set of data for analyzability that has been calculated using evaluated model's equation. Pearson's correlation coefficient is being calculated between the actual values (Analyzability Tabulated) and calculates values (Analyzability Calculated). The values are discussed in Table 4.

Table 4: Summary of Analyzability value

CD	A(Tabulated) (Known Value)	A(Calculated) (Using Model)	CD	A(Tabulated) (Known Value)	A(Calculated) (Using Model)
CD1	1.00	1.390	CD15	3.00	2.953
CD2	2.00	1.402	CD16	4.00	3.700
CD3	2.00	1.507	CD17	6.00	5.760
CD4	2.00	2.042	CD18	6.00	6.007
CD5	2.00	1.653	CD19	5.00	5.596
CD6	2.00	1.690	CD20	6.00	7.912
CD7	3.00	2.106	CD21	3.00	2.454
CD8	3.00	2.547	CD22	5.00	4.613
CD9	2.00	1.594	CD23	6.00	6.748
CD10	3.00	2.629	CD24	5.00	4.708
CD11	3.00	2.711	CD25	6.00	5.412
CD12	3.00	2.680	CD26	5.00	7.339
CD13	3.00	2.994	CD27	5.00	4.809
CD14	2.00	2.218	CD28	4.00	4.937

6. Statistical Analysis

It is mandatory to check the validity of proposed model for acceptance. A 2-sample t test has been introduced to test the significance of Analyzability_Tabulated values to Analyzability_Calculated Values. A hypothesis test based on 2-sample t test is being performed and confidence interval is being observed by the difference of two standard mean. The t test history of analyzability is mentioned in Table 5.

Table (5): T Test of Analyzability

T Test For Analyzability Data				
	N	Mean	Std Div	Std Err
Analyzability_Tabulated	28	3.64286	1.56854	0.29643
Analyzability_Calculated	28	3.64682	1.95862	0.37014
Standard Error Difference=0.474 T Value=0.0093 P Value=0.993(Two Tailed)				

Ho: (Null hypothesis): There is significant difference between Analyzability_Tabulated and Analyzability_Calculated.
H1: (Alternate hypothesis): There is no significant difference between Analyzability_Tabulated and Analyzability_Calulated.
Ho: $\mu1-\mu2=\delta0$ verses H1: $\mu1-\mu2 \neq \delta0$
Where $\mu1$ and $\mu2$ are the sample means and $\delta0$ is the hypothesized difference (zero) between the two sample mean. Mean, Standard Div, Standard Error, Standard Error difference have been calculated for given two samples at Table (X). Given samples are trusted by 95% confidence with concluding remarks that samples means are same. There is no difference between tabulated data and calculated data. Therefore the null hypothesis is rejected and alternate hypothesis is accepted. The obtained equation through using design parameters for analyzability calculation is highly accepted.

7. Conclusion

Software analyzability is one of the most significant factors of maintainability of software development. The assessment of analyzability using the model is more appropriate and its validation signifies the valid impact of structural and functional information of object oriented design software. The model is developed using multiple linear regression technique on object oriented design constructs. The applied statistical analysis on this study concludes its statistical significance remarked that calculated data is highly acceptable. This paper expresses the relation between complexity and analyzability and design complexity of software having a negative impact on analyzability.

Acknowledgement

This work is sponsored by University Grants Commission (UGC), New Delhi, India under F. No. 34-107\2008 (SR).

References

1. R. Pressman, "Software Engg: A Practitioner's Approach" , Sixth Ed. Mcgraw-Hill,2005
2. H. Zuse, " A Framework of Software Measurement" Walter de Gruyter,1998
3. ISO, "ISO/IEC 9126-1: Software Engg.-Product Quality-Part-1: Quality Model",Geneva, Switzerland,2001
4. N. Bevan, "Qualtiy and Usability: A New Framework", Van Veenendaal, E, and McMullan, J (eds) Achieving Software Product Quality, Tutein Noithenius, Netherlands,1997
5. ISO/IEC 9126-4:2004, "Software Engg.: Product Quality-Quality in Use Metrics", ISO/IEC 2004
6. J. Rumbaugh, M. Blaha, "Object Oriented Modeling and Design", ISBN:81-7808-738-3, Pearson Education, 2003
7. http://www.issco.unige.ch/en/research/projects/isle/femti/html/212.html
8. M. Ajmal Chauman, H. Kabaili, R. K. Keller, L. and G. S. Denis, "Design Properties and Object Oriented Software Changeability", SPOOL Project BY CSER, Bell Canada, National Science and Research Council of Canada (NSERC) and National Research Council of Canada (NRC), IEEE, Feb 2000, pp:45-54
9. M. O. Elish, D. Rine, "Investigation of Metrics for Object Oriented Logical Stability",CSMR-03, 0-7695-1902-4, IEEE 2003, pp:193-200.
10. R. A. Khan, K.Mustafa, "Metric Based Testability Model for Object Oriented Design (MTMOOD)", SIGSOFT Software Engineering Note, Volume 34, Number 2, March 2009.
11. Software Technologies for Embedded and Ubiquitous Systems Springer; 1 edition | November 14, 2007 | ISBN-10: 3540756639
12. W. Li, S. Henry, "Object Oriented Metrics Which Predict Maintainability", Technical Report 93-05,
13. A.Epping, C. M.Lott, "Does Software Design Complexity Affect Maintainability Effort?", citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.6.5828, 19th Annual Software Engineering Workshop, 30 Nov–1 Dec 1994
14. B. Lilburne, "Measuring Quality Metrics for Web Application", Technical Report No: CIT/08/2004.
15. M. Perepletechikov, C. Ryan, Z. Tari, "The Impact of SERVICE Cohesion on the Analyzability of Service Oriented Software", IEEE Transactions on Services Computing Volume 3 , Number 2, April-June 2010
16. L. Wang, E. Arisholm, "The Effect of Task Order on the Maintainability of Object Oriented Software ", Information and Software Technology, Elsevier 2008, doi:10.1016/j.infsof2008.03.005
17. Genero M., J. Olivas, M. Piattini and F. Romero, "A Controlled Experiment for Corroborating the Usefulness of Class Diagram Metrics at the early phases of Object Oriented Developments", Proceedings of ADIS 2001, Workshop on decision support in Software Engineering, 2001.
18. Shyam R. Chidamber, Chris F. Kemerer, "Towards A Metrics Suit for Object Oriented Design", OOPSLA, ACM, 1991, pp.197-211.
19. Jagdish Bansia, "A Hierarchical Model for Object Oriented Design Quality Assessment", IEEE Transaction of Software Engineering, Volume 28, No. 1, January 2002,pp: 4-17