International Journal of Innovative Research in Computer Science & Technology (IJIRCST)
ISSN: 2347-5552, Volume- 8, Issue- 5, September- 2020
https://doi.org/10.21276/ijircst.2020.8.5.6
www.ijircst.org

Managing Object Oriented Software Understandability: A Design Perspective

Dr. Abdullah, Teklay Teklu, Haftay Gebrezgabiher, Manoj Kumar

ABSTRACT- Estimating understandability of object oriented software early in the development process; particularly at design phase greatly reduce the overall development cost and effort. To design and deliver quality products inside time and financial plan understandability plays a very important role. This paper shows the need and significance of understandability at design phase and build up a multivariate linear model "Understandability Estimation Model" for Object-Oriented Design. Developed model estimates the understandability of class diagrams in respect of their internal design properties. In this research paper an attempt has been made to propose an understandability estimation framework as a first contribution. A relation between object oriented design properties and understandability has been established as a second contribution. In third contribution, despite the fact that, in order to estimate class diagram's understandability the researcher further developed multivariate models. In conclusion the proposed Understandability models have been authenticated via experimental test.

.

KEYWORDS- Understandability, Modifiability, Object Oriented Software, Design Phase, Design Construct.

I. INTRODUCTION

Now day's software engineering has turn into tremendously vital discipline of study, research and practice. Everyone are working hard to decrease the problems and to meet the purpose of developing high-quality understandable, testable software that is delivered on time, within budget, and furthermore satisfies the requirements [1, 3, 5]. Software has become significant to expansion in almost all areas of human endeavor. The ability of programming merely is no longer sufficient to make big programs. There are severe problems in the cost, timeliness, testing, maintenance and quality of several software products [2, 6]. Software engineering has the purpose of solving these problems by producing feature-quality understandable, testable and maintainable software within time, financial plan. To attain this goal, we encompass to center in a closely controlled way on both the quality of the product and on the process used to develop the product [4, 7].

In today's world, the importance of delivering quality software is no longer an advantage but a necessary factor. However, with the growing Modifiability, pervasiveness and criticality of software, major factor of assuring that it behaves according to the desired level of quality and dependability has become more crucial, increasingly difficult and expensive. Moreover, the Modifiability of applications and environments has substantially increased in the last two couple of decades. Unluckily, the majority of the software industries not only fails to deliver a quality product to their customers, but in addition does not recognize the appropriate quality attributes [8, 9, 21].

II. UNDERSTANDABILITY

Software understandability has always been an indefinable idea and its exact valuation or assessment a problematic exercise [11, 12, 14]. Most of the studies measure Understandability or more precisely the attributes that have impact on Understandability but at the source code level. Though, Understandability estimation at the source code level is a good indicator of effort Measurement, it leads to the late arrival of information in the software development procedure. A choice to change the design in order to increase understandability after coding has started might be very costly and error prone [10, 13]. While estimating Understandability early in the development process may greatly reduce the overall cost. This paper provides a roadmap to industry personnel and study to assess, and preferably, quantify software Understandability in design phase [25, 26, 28]. A prescriptive framework has been proposed in order to integrate Understandability within the development life cycle. It may be used to point of reference for software products according to their Understandability.

Manuscript received September 20, 2020

Dr. Abdullah, Assistant Professor, Department of Information Technology, Adigrat University (A Public University), Adigrat- Tigray, Ethiopia-Africa, (e-mail: abdullahkhan.ap@adu.edu.et)

Teklay Teklu1, Lecturer, Department of Information Technology, Adigrat University (A Public University), Adigrat- Tigray, Ethiopia-Africa,

Haftay Gebrezgabiher, Lecturer, Department of Information Technology, Adigrat University (A Public University), Adigrat- Tigray, Ethiopia-Africa,

Manoj Kumar, Senior Assistant Professor, Department of Information Technology, International Institute for Special Education, Lucknow, Uttar Pradesh, India

III. Generic Guidelines for Managing Object Oriented Software Understandability

As a matter of fact, study and practitioners highly recommend an efficient and accurate measure of software Understandability early in design phase. There is a common consensus among industry professionals and academicians in integrating Understandability within the development life cycle in order to deliver quality software. Unfortunately, there is no standard methodology or guideline available to quantify software Understandability [15, 17, 19, 22, 24]. Therefore, such a roadmap or framework, which can be followed by industry personnel and study to quantify Understandability early in design phase, appears highly desirable and significant. A prescriptive framework as depicted given below has been proposed to estimate Understandability of object oriented software at design level. The framework comprised of seven phases including a common phase of 'Design Review'. A brief description of the framework components is given as follows.

IV. FRAMEWORK DEVELOPMENT

PREMISES:

A framework is a supposed explanation of a composite process. It offers a realistic base for upcoming research direction. The framework for Understandability Estimation of object oriented design has the following assumptions:

- Understandability of an object oriented software design is affected by several factors
- Understandability of OO design is affected by object oriented design constructs.
- Understandability of an OO design is affected by properties of an object oriented design.
- The framework estimate the Understandability by controlling internal object oriented design properties

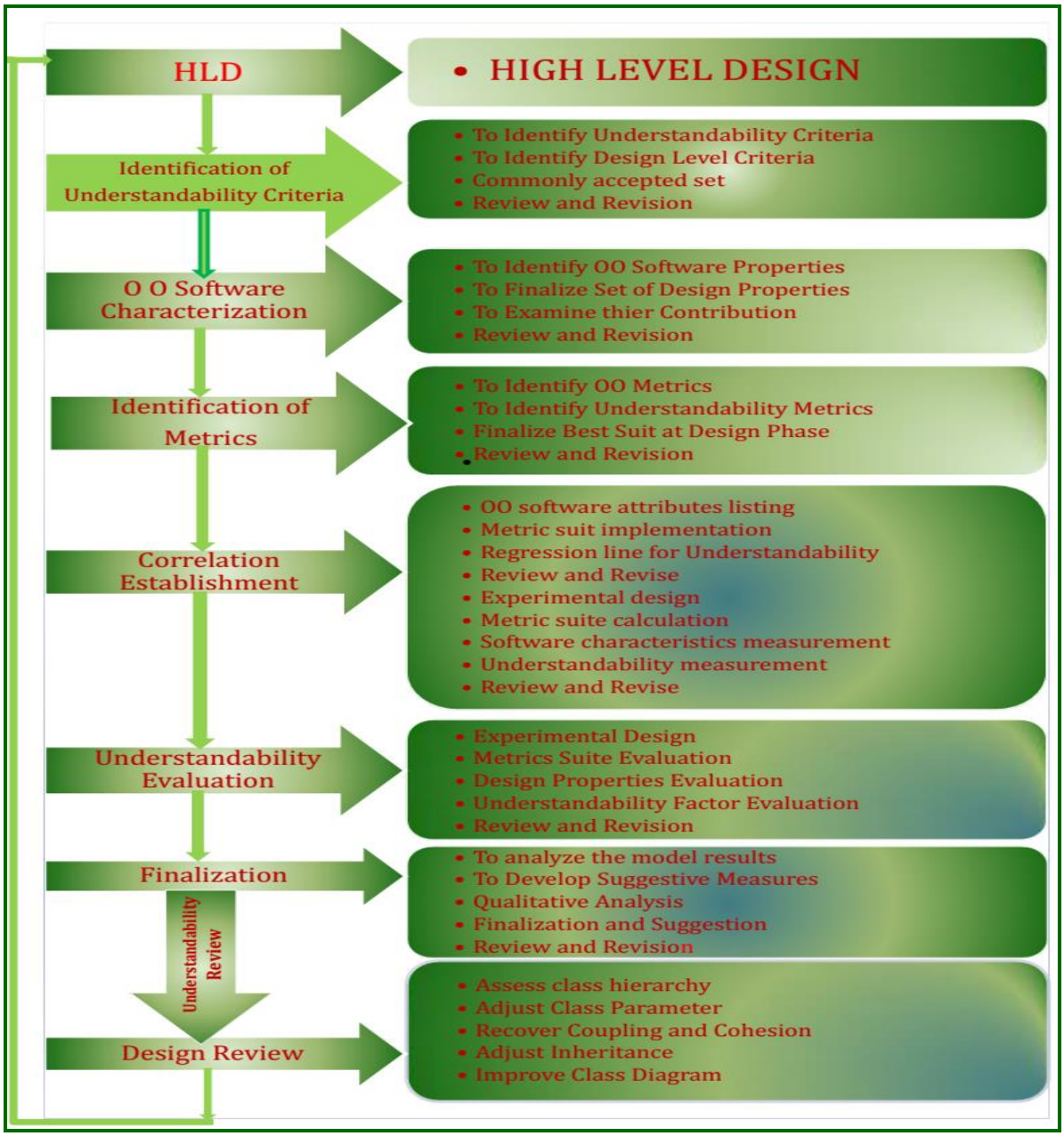

Fig 1: Understandability Estimation Framework

International Journal of Innovative Research in Computer Science & Technology (IJIRCST)
ISSN: 2347-5552, Volume- 8, Issue- 5, September- 2020
https://doi.org/10.21276/ijircst.2020.8.5.6
www.ijircst.org

A. CONCEPTUALIZATION

Software Understandability Estimation is an activity that determines current Understandability by applying statistical inference technique to failure data obtained during system test or system operation [19, 27]. Understandability Estimation framework of an object oriented design believes that estimation is a tool for quantifying the effectiveness of any Estimation activity [18]. Software Understandability estimation includes both Estimation and prediction with the help of software Understandability models [16]. The estimation framework are called Understandability estimation framework.

V. OBJECT ORIENTED DESIGN CONSTRUCTS

Encapsulation is the mechanism to hide the internal specification of an object and shows only the external interface. Inheritance is an approach where an object acquires the characteristics from another object by sharing of attributes and operations among classes through their hierarchical relationship [22]. The new classes of objects that inherit much of their behavior from previously defined classes [17, 26].The two more, most important design constructs may be included, that have been generally used in designing of the software that is coupling and Cohesion. Coupling is the process to interact or communicate between two objects by passing messages. It refers to the degree of association from one object to another [15]. Cohesion is the process to measures the degree of connectivity among the elements of a single class or object [20]. It refers to the degree, to which the methods in a class are related to each other. The internal consistency occurs within the parts of the design, and it is focused on data that is encapsulated within an object and how the methods communicate with data to provide well bounded behavior [15].

VI. OVERVIEW OF THE PROPOSED MODEL

Understandability is directly related to testability and maintainability and continuously plays a significant role to provide high class maintainable and testable software within time and financial plan. It is one of the most important concepts in design and testing of software programs and components. It always supports for improved software design at early stage of software development life cycle that is to say at design phase that have positive impact on the overall Understandability estimation cost and effort [10, 14, 15]. We have established an understandability estimation model that proves the estimation process of software understandability design phase perspective. The proposed model is shown in Figure 2. The model creates an appropriate relationship between understandability and object oriented design properties and the related metrics. The values of these metrics can be effortlessly identified with the help of class diagram. The quantifiable evaluation of Understandability is very supportive to get Understandability index of software design for low cost Understandability estimation.

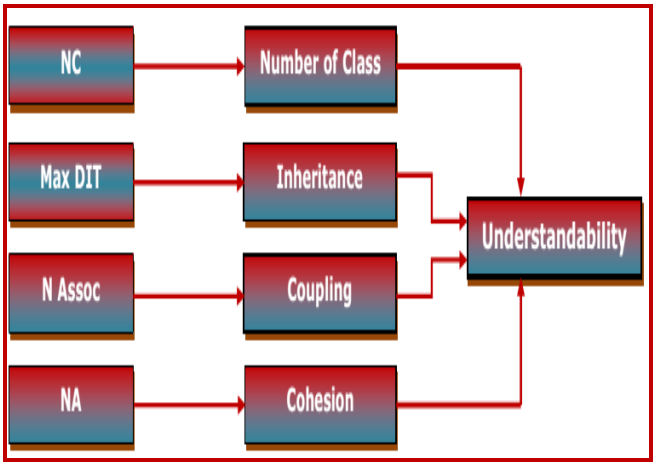

Fig 2: Mapping among Object Oriented Design Properties and Understandability

A. MODEL DEVELOPMENT

In order to set up a model for Understandability, multiple linear regression method has been used. Multivariate linear model is specified below in Equation number (1) which is as follows.

$$Y=a_0+a1X1+a2X2+a3X3+-------+anXn \qquad \textbf{Eq (1)}$$

Where,

- Y is dependent variable.
- X1, X2, X3--------Xn (be independent variables) associated to Y.
- a1, a2, a3--------an., are the coefficient of the exacting independent variables.
- a_0 is the intercept.

The data used for establishing Understandability model is taken from Genero [27] that have been collected through large commercial object oriented systems. The relationship between Understandability and object oriented properties has been established as depicted in Figure 2. As per the mapping, Metrics 'NC, MaxDIT, NAssoc, NA' are selected from [27] as independent variable to build up the Understandability estimation model via SPSS, values of coefficient are calculated in addition to Understandability assessment model is formulated as specified below.

B. UNDERSTANDABILITY ESTIMATION MODEL

In order to create a multivariate model for Understandability of class diagram, metrics listed in Genero, will play the role of independent variables whereas Understandability will be in use as dependent variable. Data used during the study has taken from Genero et al. [27]. It contains Understandability data, collected through a controlled experiment. It comprises a group of 28 class diagrams (indicated as D0 to D27) and the metrics value of every one diagram. Additionally, the mean value of the expert's score of understandability of these class diagrams is also given and called as 'Known Value' in this paper.

Understandability = 1.133+ .235 x **NC -**.083 x **MaxDIT**

+ .032 x **NAssoc +**.002 x **NA** Eq. (2)

Table 1: Coefficients values for Understandability estimation Model

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	1.133	.350		3.239	.018	.277	1.989
	Number_of_Class	.235	.203	.911	1.157	.291	-.263	.733
	Inheritance	-.083	.388	-.085	-.213	.838	-1.031	.866
	Coupling	.032	.209	.076	.153	.884	-.479	.543
	Cohesion	.002	.068	.015	.027	.980	-.165	.169

a. Dependent Variable: Understandability

The model summary table 2 outcomes are best cooperative when accomplishment multiple regressions. R is the coefficients determinant that communicates us how strongly the multiple independent variables are related to the dependent variable. R Square is highly helpful as it contributes us the coefficient of determination.

Table 2: Understandability estimation Model Summary

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics				
					R Square Change	F Change	df1	df2	Sig. F Change
1	.949[a]	.900	.834	.48185	.900	13.574	4	6	.004

a. Predictors: (Constant), Cohesion, Inheritance, Coupling, Number_of_Class

Table 3: ANOVA[b] for Understandability estimation Model

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	12.607	4	3.152	13.574	.004[a]
	Residual	1.393	6	.232		
	Total	14.000	10			

a. Predictors: (Constant), Cohesion, Inheritance, Coupling, Number_of_Class

b. Dependent Variable: Understandability

VII. EXPERIMENTAL TRYOUT

Empirical justification is a dynamic phase of proposed research work. Empirical justification is the normal approach to justify the proposed model approval as well as verification. In view of this actuality, practical authentication of the Understandability model has been done with sample tryouts. In order to validate developed Understandability model the data has been taken from [10].

Table 4: Understandability Ranking and their Relation

Projects ↓	Understandability		Understandability Ranking		d^2	$\sum d^2$	r_s	$r_s > \pm.781$
	Known Value	Value achieved using Model	Known Rank	Rank achieved using Model				
P1	7.1	1.7	3	3	0	**14**	**.915**	✓
P2	6.9	1.6	1	2	1			
P3	8.1	1.9	9	7	4			
P4	7.1	1.5	2	1	1			
P5	8.5	2.0	10	9	1			
P6	7.2	1.8	4	4	0			
P7	7.4	1.9	5	6	1			
P8	8.1	2.1	8	10	4			
P9	7.8	2.0	7	8	1			
P10	7.4	1.8	6	5	1			

Speraman's Coefficient of Correlation rs was used to check the significance of correlation among calculated values of Understandability using model and it's 'Known Values'. The 'rs' was estimated using the method given as under:
Speraman's Coefficient of Correlation

$$r_s = 1 - \frac{6\sum d^2}{n(n^2-1)} \qquad -1.0 \le r_s \le +1.0$$

'd' = difference between 'Calculated ranking' and 'Known ranking' of Understandability.

n = Total number of 'projects' used in the experimentation.

The correlation values between Understandability through model and known ranking are shown in table (4) above. Pairs of these values with correlation values rs above [±.781] are checked in table. The associations are up to standard with high degree of confidence, i.e. up to 99%. Therefore we can conclude without any loss of generality that Understandability Estimation model measures are really reliable and significant and applicable.

'd' = difference between 'Calculated ranking' and 'Known ranking' of Understandability.

n = Total number of 'projects' used in the experimentation.

The correlation values between Understandability through model and known ranking are shown in table (4) above. Pairs of these values with correlation values rs above [±.781] are checked in table. The associations are up to standard with high degree of confidence, i.e. up to 99%. Therefore we can conclude without any loss of generality that Understandability Estimation model measures are really reliable and significant and applicable.

VIII. CONCLUSION

The framework proposed in this paper will address understandability during software development life cycle. It may help putting understandability benchmarking of software projects. The framework is generic in nature, and may be used by industry practitioners to estimate understandability in order to make design decisions early in the development life cycle. Strong theoretical basis presented in the paper supports the claim of the framework's usability to estimate understandability of object oriented software at design phase. This research work shows the significance of understandability in general and as a key attribute to software testability and maintainability for creating high quality software within time and budget. Understandability is clearly highly appropriate and significant in the perspective of software maintainability. Understandability model is developed with the help of multiple linear regression method on object oriented design properties. Statistical examination displays that understandability model is statistically too much significance and acceptable. Understandability estimation model has been validated contextually tryout.

REFERENCES

[1] K. K. Aggarwal, Y. Singh, and J.K. Chhabra, "A Fuzzy Model for Measurement of Software Understandability", International Symposium on Performance Evaluation of Computer & Telecommunication Systems, Montreal, Canada (2003).

[2] W.N.Lo Bruce, S. Haifeng, "A Preliminary Testability Model for Object-Oriented Software", in Proceeding of International Conference on Software Engineering, Education, Practice, pp. 330- 337, IEEE (1998).

[3] J. Voas, Miller, "Improving the Software Development Process using Testability Research", pp. 114-121, IEEE Software (1992).

[4] R.V. Binder, "Design for Testability in object-oriented Systems", Communications of the ACM. Vol. 37(9), pp. 87-101 (1994).

[5] L. Zhao, "A new approach for software testability analysis", International Conference on Software Engineering, Proceedings of the 28th International Conference on Software Engineering, Shanghai, pp. 985–988, 2006.

[6] J. Gao, S. Ming-Chih, "A Component Testability Model for Verification and Measurement", In Proceedings of the 29th Annual International Computer Software and Applications Conference, pages 211–218. IEEE Computer Society (2005).

[7] Dr, Reena Srivastava, and M. H. Khan. "Testability Estimation of Object Oriented Design: A Revisit". International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2, Issue 8, pages 3086-3090, August 2013.

[8] G. Jimenez, S. Taj, and J. Weaver, "Design for Testability" in the Proceedings of the 9th Annual NCIIA Conference, 2005.

[9] S. Jungmayr, "Testability during Design", SoftwaretechnikTrends, Proceedings of the GI Working Group Test, Analysis and Verification of Software, Potsdam, pp. 10-11, 2002.

[10] Abdullah, Dr, Reena Srivastava, and M. H. Khan. "Modifiability: A Key Factor To Testability", International Journal of Advanced Information Science and Technology, Vol. 26, No.26, Pages 62-71 June 2014. (IJCSIT), Vol. 7, No 1, February 2015, DOI: 10.5121/ijcsit.2015.7115.

[11] S. Jungmayr, "Reviewing Software Artifacts for Testability", EuroSTAR'99, Barcelona, Spain, 1999.

[12] I. Sommerville, *Software Engineering*, 5th Edition, Addison Wesley, 1996.

[13] R. A. Khan, K. Mustafa, I Ahson, "An Empirical Validation of Object Oriented Design Quality Metrics, Journal King Saud University, Computer & Information Science, Vol. 19, pp. 1-16, Riyadh (1427H/2007).

[14] Abdullah, Dr, M. H. Khan, and Reena Srivastava. "Testability Measurement Model for Object Oriented Design (TMMOOD)". International Journal of Computer Science & Information Technology (IJCSIT), Vol. 7, No 1, February 2015, DOI: 10.5121/ijcsit.2015.7115.

[15] M. Genero, 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.

[16] T. DeMarco, Controlling Software Projects, Englewood Cliffs, NJ, Yourdon Press, 1982.

[17] C. Valdaliso, O. Eljabiri, F.P. Deek, "Factors Influencing Design Quality and Assurance in Software Development: An Empirical Study", Electronic Proceedings of the First International Workshop on Model-based Requirements Engineering (MBRE 01), San Diego, California, 2001.

[18] Dr. Abdullah, Dr. Mahfuzul Huda, Hagos Yirgaw, "A Methodology to Evaluate Extensibility of Object Oriented Design: A Product Transition Perspective". International Journal of Advanced Research in Computer and Communication Engineering Vol. 8, Issue 10, October 2019. DOI 10.17148/IJARCCE.2019.81001

[19] B. Jacob, L. Niklas, P. Waldermarsson, "Relative Indicators for Success in software development", Deprtment of Communication Systems, Lund University, 2001.

[20] Abdullah, Dr, Reena Srivastava, and M. H. Khan. "Testability Measurement Framework: Design Phase Perspective". International Journal of Advanced Research in Computer and Communication Engineering, Vol. 3, Issue 11, Pages 8573- 8576 November 2014.

[21] Ramesh Kumar , Dr. Abdullah, Abhishek Yadav (2020) Measuring Maintainability of Object Oriented Design: A Revisit IJIRCST Vol-8 Issue-5 Page No-354-360, DOI: 10.21276/ijircst.2020.8.5.4 DOI URL: https://doi.org/10.21276/ijircst.2020.8.5.4

[22] S. Mouchawrab, L. C. Briand, and Y. Labiche, "A Measurement Framework for object-oriented Software Testability, Information and Software Technology, Canada , Vol. 47, Issue 15, pp. 9 7 9- 9 97 , 2005.

[23] E. Mulo, "Design for Testability in Software Systems", Department of Software Technology, Faculty EEMCS, Delft University of Technology, Netherlands, 2007.

[24] Abdullah, Dr, M. H. Khan, and Reena Srivastava. "Flexibility: A Key Factor To Testability", International Journal of Software Engineering & Applications (IJSEA), Vol.6, No.1, January 2015. DOI: 10.5121/ijsea.2015.6108.

[25] R. Marinescu, "Measurement and Quality in Object-Oriented Design", Faculty of Automatics and Computer Science, University of Timisoara, 2002.

[26] Dr. Abdullah1 and Dr. Mahfuzul Huda, "Empirically Validated Simplicity Evaluation Model For Object Oriented Software". International Journal of Software Engineering & Applications (IJSEA), Vol.9, No.6, November 2018. DOI:10.5121/ijsea.2018.9606

[27] M. Genero, E. Manso, and G. Cantone, "Building UML Class Diagram Maintainability Prediction Models Based on Early Metrics," Proc. 9th International Symposium on Software Metrics (METRICS'03), 3 - 5 Sept., 2003, pp. 263 - 275, 2003.

[28] S. Jungmayr, "Identifying Test-Critical Dependencies," in the P ro c e e d i n g s , IEEE International Conference on Software Maintenance, pp. 404-413, 2002

