

INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY

WINGS TO YOUR THOUGHTS.....

A REVIEW ON MAINTAINABILITY OF OBJECT ORIENTED DESIGN METRICS

Bhavna Katoch¹, Rupinder Kaur²

¹Student, Lovely Professional University,
Phagwara, Pin no. 144402
bhavnaguleria16@gmail.com,

²Asst. Professor
Lovely Professional University
Phagwara, Pin no. 144402
rupinder.16835@lpu.co.in

Abstract: MOOD metric is the object oriented metrics. In the MOOD metrics many factors are used to enhance the system performance. The maintainability of the MOOD metrics is big issue. Maintainability is that phase which may help the designer for improving the quality of the software system before deliver to a customer. In the development of the software systems the object oriented developers are promises to reduce the maintenance effort. The object oriented development methods models the system components as the objects. In our purposed work, we are going to enhance the maintenance of MOOD metrics by adding the two factors. The first factor is conciseness and the other factor is scalability. Conciseness helps to minimizing the redundant information whereas scalability is used to enhance the performance of software system.

Keywords: Maintenance, performance, software engineering, MOOD metric.

1. INTRODUCTION

A software engineering is related to all the aspects that are used in the software production. Software is basically a generic term, which is used for organizing the data and instructions that are collected to develop it. The software is broken into the two categories: system software and the application software. The system software is used to manage the hardware components, so that other software or user see it as a functional unit. The software contains the operating system and some more utilities like disk formatting, file managers, display managers, etc. The application software used for accomplished the specific tasks. Application software may or may not contain the single program. Software is the program or set of programs. It is different from the program in many ways. As in software many things are includes: as it consists of the programs, the complete documentation of that program, the procedure that is use to set up the software and the various operation of the software system.[1] any program is the subset of the software. As on the other hand, program is the combination of source code and object code.

Need of Software Engineering For Software Development: As software requirement is increases day by day. So it is necessary to maintain the good quality software.

To develop good quality software, software engineering is required. For this, the developer's needs to adopt the software engineering concepts, strategies, and practices to avoid the conflicts that are occur during the development process. Software engineering is an approach to develop, maintain and operate the software. The software development plays a crucial role in software engineering. Many specific techniques are required to develop software. The most common thing in development process is the requirement gathering and customer needs. If a developer fails to complete the needs of the customer than he or she may fails to develop good quality software. Software can be said to of good quality, if it is able to fulfill the needs of the customer. The customer can be satisfied in terms of quality, cost and design of the system software. Many developers adopt the techniques like systematic and organized approach to develop software.[2]

Unified Modeling Language:

UML stands for unified modeling language. UML is the standard language. It is used to writing the software blueprints. It is used for the various software intensive systems. It is used to visualize, construct, specify, and document the various characters of the project. UML is the very expressive language. UML is very easy to understand and use. To learning the UML there are basic three learning

INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY

WINGS TO YOUR THOUGHTS.....

concepts of UML are required, i.e. Building Blocks of UML, Rules, mechanisms. UML is basically used in the software engineering fields. UML is defined by the OMG i.e. Object management group.[3] Unified modeling language is not restricted to the modeling software. It is use to build the models for business process and for the various organizations Unified Modeling Language is important due to the number of reasons. It has been used as a catalyst for the advancement of technologies which are model driven. UML is very helpful for the IT professionals, as with the help of UML the IT professionals now read and disseminate system structure and the designs. UML is a standard language for the software blueprints. UML is the basic language for the visualizing, specifying, constructing, documenting. UML is the language which is useful for both the enterprise system to the web based distributed system. It is a very expressive language.

MOOD Matrix: Mood matrix stands for maintainability of object oriented design. In the development of the software systems the object oriented developers are promises to reduce the maintenance effort. The object oriented development methods models the system components as the objects. These objects are helpful in allow the designer to separate the interface from the implementation. Earlier the maintainability can be defines in three different ways as,

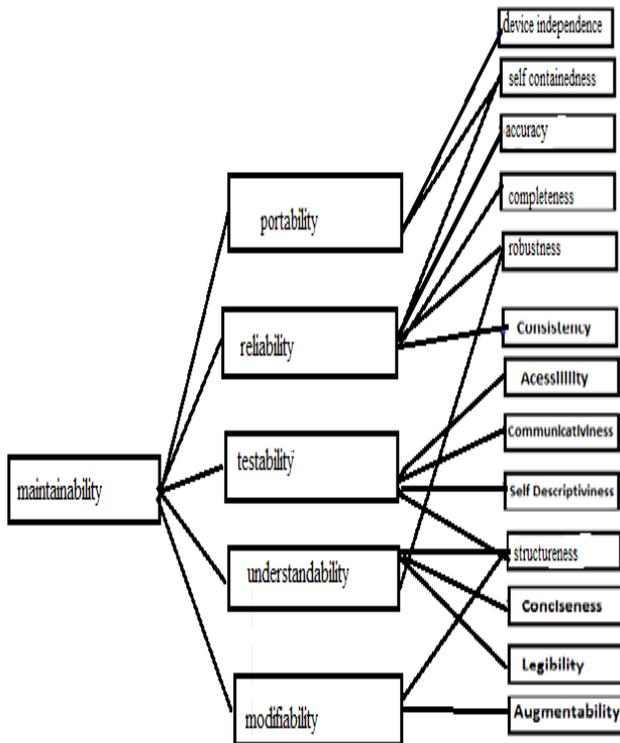


Figure 1: Existing Maintainability Model

In the figure 1, the existing maintainability model is shown, which contains the five various factors.

2. LITERATURE REVIEW

Heuristics Based on Object Oriented (OO) Metrics Rakesh Kumar, Deepali Gupta, [2012]: In this paper, author discuss that the object oriented metrics are generally used to predict the quality of the object oriented software products. There are various attributes that are used to determine the quality of the software which includes the maintainability, understandability, reusability etc. This paper is generally focus on the object oriented metrics and it also propose some heuristics based approach. [4] The software metrics are applicable to all phase of software development life cycle. Metrics are basically of three types:

Product metrics: these metrics are use to measure the software product properties.

Process metrics: these metrics are emphasizes on the software development process.

Resource metrics: these metrics are emphasizes on the human, hardware and software resources.

New Proposed Inheritance Metrics to Measure the Software Complexity, Preeti Gulia, Rajender S, [2012]:

In this paper author discuss that the inheritance is an important aspect in developing the software system. Inheritance provides the class hierarchy design and it also provides the relation between classes and inheritance. Complexity of software helps to increases the testing and maintenance efforts. The developers of the project always try to reduce the software complexity. The software complexity is helps to reduce the testing and maintainability. In this paper, two new inheritance metrics are proposed. These metrics are based on the class complexity due to depth of inheritance tree and class complexity due to number of children.[5] From the last decades, the object-oriented technologies come into existence, which is used for fast development of software and also used to reduce the time and cost. For developing the better quality software, it is necessary to identify the complexity at the various levels of the software development. Complexity may effects the many factors, such as coupling, cohesion and inheritance. Inheritance is act as the key feature of object oriented technology, it is used to increases the reusability. With the help of inheritance, it helps to reduce the software maintenance and testing efforts.

Analysis of Reusability of Object Oriented System using CK Metrics, Brig Mohan Goel, Pradeep Kumar Bhatia [2012]:

In this paper author discuss that the object oriented environment is one of the most important aspects which have

INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY

WINGS TO YOUR THOUGHTS.....

the strong influence on the quality of software system is the design complexity. The object oriented model offers the technology to create components that are used for general programming. Design complexity to play an important role in the quality of the resulting software system. In this paper, author tells about the design of CK suit of metrics and also describes the evaluation to these metrics. Hence these metrics should reflect accurate and precise results for object oriented based systems. [6] The object oriented systems share a major portion of software development systems. In this paper, the object oriented metrics are act as predictor for software complexity. This is use to calculate and analyze the object oriented metrics.

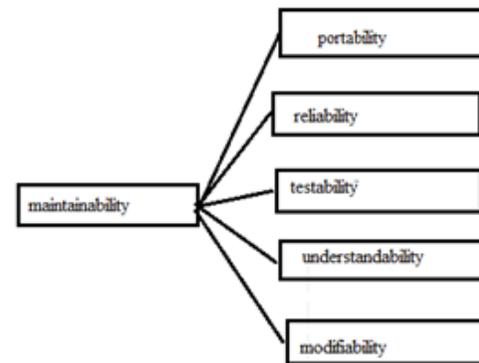


Figure 2: Existing Maintainability Model

In the existing maintainability model three basic factors are there. These factors are:

- **Testability:**

Testability is the non functional requirement. It defines the property of measuring the ease of testing. It basically measure the piece of code and functionality of the system. When software is tested, firstly a piece of code is tested. The errors are finds out in that piece of code. After that the whole system is tested. Hence testability increases the maintainability of the system. In the testable systems, whenever user receives the correct output, but the internal processes are not the same as specify in the requirements, the system found the defects.

- **Understandability:**

Understandability of the software is an important aspect of software engineering. It helps to increase the quality of the software system. The software understandability is use to influence the cost and reliability of the system.

- **Modifiability:**

Modifiability defines that the make some correction in the system, to make it more reliable. It includes the corrections and improvements of the software. It is also use to locate the error and fix that error. In the modifiability, the changes are take place in the environment, requirements or functional specification.

- **Portability:**

The effort required to transfer a program from one environment to another. To executable version of software in a new environment is an act of porting. It's like software component but in portability the component change its environment. A software system is a collection of software units. We use the term software unit to indicate an

Comparison of Software Quality Metrics for Object-Oriented System, Amit Sharma, Sanjay Kumar Dubey, [2012]: In this paper author discuss that the object oriented design is becoming the more important in software development environment. The software Metrics are used to measure the software complexity, estimating size, quality and project efforts. The various types of tools are used to measure the estimations in lines of codes, function points, and object points. In this paper, author classifies the various metrics, such as, software quality metrics and object oriented metrics. The object oriented design is very beneficial for the software development environment. The object oriented design contains all the properties and quality of software. Object oriented system is a classifying approach; it is capable to classify the problem in terms of object. [7] The software metrics are the crucial source of information, which helps the software developer to take decisions for design good software.

3. PURPOSED WORK

The present work is about increasing the maintainability factors of the MOOD metrics. These factors help to increase the functionality of the software system. With the help of these factors the scalability, conciseness, and maintainability of the software system become easy. UML is related to the object oriented paradigm and thus fits in with the goals of this work. Earlier Maintainability models estimate the maintainability of UML class diagrams in terms of their understandability and modifiability on the basis of non standard metrics which are neither used in any other research.

INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY

WINGS TO YOUR THOUGHTS.....

application program, a system program, or a component of a program.

- **Reliability:**

The software verification process aims to detect and remove defects and make it more reliable. Changes that introduce defects into the software make it less reliable. Reliability can be reduced by reusing components that have not been developed to the same standards as the rest of the software. The effect of a modification on software reliability can be estimated indirectly by measuring its effect on the complexity of the software.

4. METHODOLOGY

In the following, definitions will be given of a few fundamental concepts discussed in this report. While the terms below will be used throughout the report, it must be understood that we can decrease the maintenance cost by adding scalability and conciseness in to maintenance model. Maintainability is an effort required to locate and fix a fault in the program within its operating environment. In the purposed methodology we add two new factors in to maintainability model. These two factors are scalability and conciseness. These factors may reduce the maintenance cost of the software project.

In our proposed methodology, we are going to add two factors.

Scalability: It is the ease with which the system or component can be modified to fit the problem area. For enhancing the performance of software projects, we need the scalability. As it is used to modify the components of the software projects, and it can easily finds the bugs in the software system.

Conciseness: it is used for minimizing the excessive or redundant information. This is important where memory capacity is limited, and it is generally considered good practice to keep lines of code to a minimum. It can be easily modified, because it takes the less space than the other software factors and it helps to make system more efficient.

REFERENCES

- [1] <http://sa.inceptum.eu/tool/modifiability>
- [2] <http://umlguide2.uw.hu/ch02lev1sec1.html>
- [3] http://www.augustana.ab.ca/~mohrj/courses/2003.fall/csc220/lecture_notes/SE_definition.html
- [4] Heuristics Based on Object Oriented (OO) Metrics Rakesh Kumar, Deepali Gupta,[2012]
- [5] New Proposed Inheritance Metrics to Measure the Software Complexity, Preeti Gulia, Rajender S,[2012]
- [6] Analysis of Reusability of Object Oriented System using CK Metrics, Brig Mohan Goel, Pradeep Kumar Bhatia [2012]
- [7] Comparison of Software Quality Metrics for Object-Oriented System, Amit Sharma¹, Sanjay Kumar Dubey²,[2012]

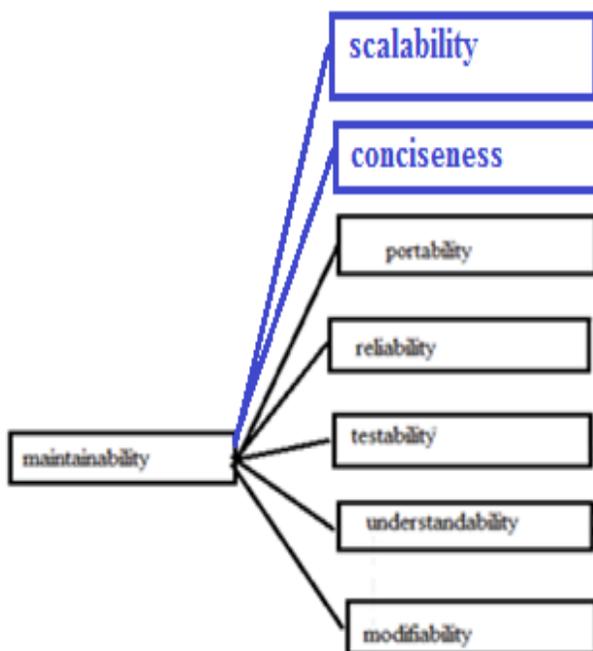


Figure 3: New Purposed Maintainability Model