

QUALITY MODEL BASED ON COTS QUALITY ATTRIBUTES

Khaled Musa¹ and Jawad Alkhateeb²

¹Department of Software Engineering,
Alzaytoonah University of Jordan, Amman, Jordan
Informatics, University of Huddersfield, UK
Dr.khalid@zuj.edu.jo

²College of Computer Science and Engineering, Taibah University, KSA
jkhateeb@taibahu.edu.sa

ABSTRACT

The quality of software is essential to corporations in making their commercial software. Good or poor quality to software plays an important role to some systems such as embedded systems, real-time systems, and control systems that play an important aspect in human life. Software products or commercial off the shelf software are usually programmed based on a software quality model. In the software engineering field, each quality model contains a set of attributes or characteristics that drives its quality model. These quality attributes reflects the quality of a software product or one of the commercial off the shelf software. This study is a comparative study we will discuss the software quality models: Boehm's quality model, McCall's quality model, FURPS quality model, Dromey's quality model and ISO 9126 quality model. This study will use empirical data to compare quality models based on their quality attributes to determine which of these models are used in the most known commercial off the shelf software based on their quality attributes. The importance of this study is to determine which of the most used quality models by software developers in the computer software industry.

KEYWORD

Software Quality, Software Quality Models, and COTS.

1. INTRODUCTION

The quality of software is important to corporations in making their commercial software, as it plays an important role to some systems such as embedded systems, real-time systems, control systems and others which all play an essential aspect in human life. Software used products can be custom made or commercial of the shelf software (COTS) that is built to fulfill the general needs of an industry. The use of COTS as software system products has grown for both industry and government agencies. COTS products represent the already made software products for commercial use rather than building a custom made software system from scratch. COTS products are usually programmed based on a set of characteristics of one of the known software quality models. The quality of COTS is the derived force to adapt a software product. The quality of COTS is derived from the types of quality attributes implemented while building the software system product. Quality attributes are a set of characteristics that are contained in a specific quality model. There are a number of quality models, where each consist of quality attributes or characteristics that shows the quality of a software product. This paper consists of four sections: Section 2, discusses recent related work that are done in quality models and commercial off the shelf software. Section 3, discusses the contents of the five quality models

Boehm's Quality Model, McCall's Quality Model, FURPS Quality Model, Dromey's Quality Model, and ISO 9126 Quality Model. Section 4, quality models analysis, discusses the quality attributes based on these quality models. Section 5, COTS classification, the classification of the mostly used COTS will be discussed. Section 6, presents the conclusion of this comparative study that discusses which quality model that is most used based on the implemented COTS quality attributes.

2. RELATED WORK

Several studies exist for the COTS products and for quality models. Some studies are done on COTS to guide organizations to evaluate appropriate COTS products [1], the reuse of COTS [4], classify COTS in categories based on their attributes [10], and Defining and Classifying COTS products [11]. Some studies are done as analytical and comparative Study about quality models such as Boehm's Quality Model, McCall's Quality Model, FURPS Quality Model, Dromey's Quality Model, and ISO 9126 Quality Model [2], selecting software package using quality models [3], and a Systematic study of software quality models to measure the quality of software products [5].

3. QUALITY MODELS

This section introduces the five software quality models which are Boehm's Quality Model, McCall's Quality Model, FURPS Quality Model, Dromey's Quality Model, and ISO 9126 Quality Model. All software quality models are used in software system product development [3][5][6][7][8][9].

3.1. Boehm's Quality Model

Boehm introduced his quality model to evaluate the quality of software [2]. This model attempts to qualitatively define the quality of software by predefined set of attributes and metrics. Boehm quality model consists of three models high-level (Primary characteristics), intermediate-level, and lower-level (primitive characteristics), figure 1. Each level contains a set of quality characteristics to produce quality in most used COTS. The quality attributes on Boehm [2] quality standards focuses on portability, reliability, efficiency, human engineering, testability, understandability, and modifiability.

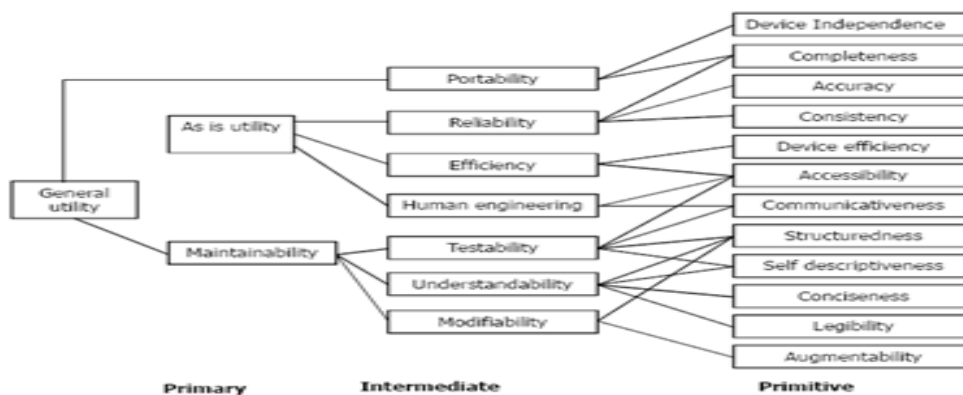


Figure 1. Boehm Software Quality

3.2. McCall Quality Model

McCall software quality model is aimed towards system developers and system development process [5]. McCall's model is to bridge the gap between users and developers by focusing on a number of software quality factors that indicate the views of both users and developers.

The structure of McCall's quality model is three major categories (product revision, product transition, product operation), figure 2, each type of quality attributes consists of a set of quality attributes.

The category product revision consists of maintainability, flexibility, and testability quality attributes. The product transition category consists of portability, reusability, interoperability quality attributes. The product operation category consists of a set of quality attributes that includes correctness, reliability, usability, integrity, and efficiency.

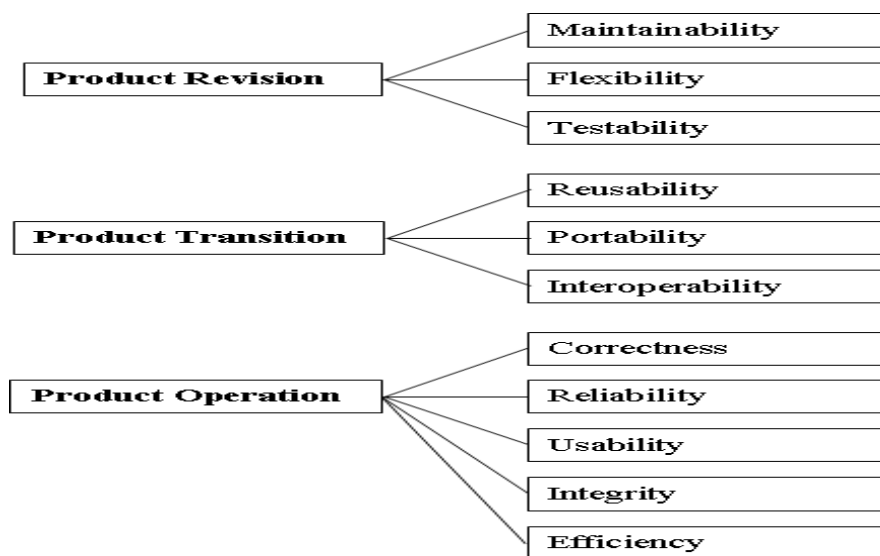


Figure 2. McCall Software Quality Model

3.3. FURPS Quality Model

FURPS software quality model is created by Robert Grady [6]. The structure of FURPS software quality model consists of five quality attributes or characteristics that includes supportability, performance, reliability, usability, and functionality.

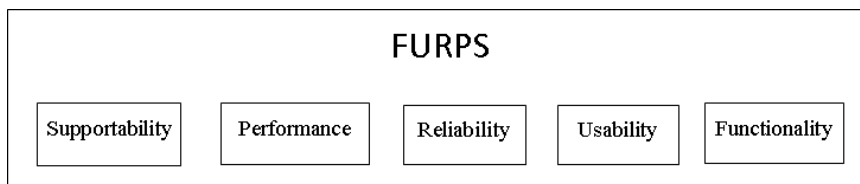


Figure 3. FURPS Software Quality Model

3.4. Dromey's Quality Model

Dormey's software quality model is created by Dromey [7][8] and presents four quality categories where each category consists of quality attributes. The four categories are correctness, internal, contextual, and descriptive, figure 4. The four categories share similar quality attributes along with each owns designated attributes [8]. The correctness category includes functionality and reliability. The internal category includes maintainability, efficiency. The contextual category includes reusability and portability. The descriptive category includes usability.

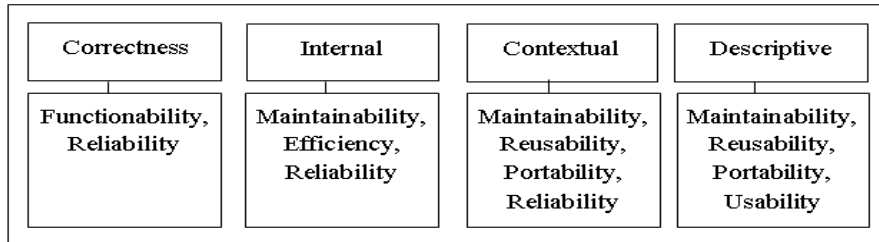


Figure 4. Dromey's Software Quality Model

3.5. ISO 9126 Quality Model

The ISO 9126 software quality model was created in 1991 by the country members of the ISO organization as the ultimate software quality standard and revised many times to contain four parts [9]. The software quality model is described into two categories: a) internal quality and external quality, and b) quality in use figure 5. The first part is further divided into six categories where each consists of a set of quality attributes. The second part consists of four quality attributes. The quality in use category consists of four software quality attributes such as effectiveness, productivity, safety, and satisfaction, figure 6.

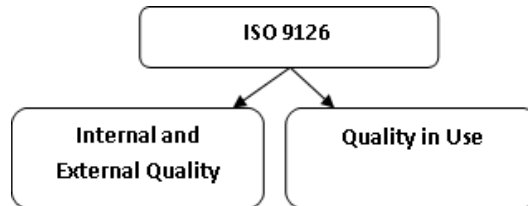


Figure 5. ISO Software Quality

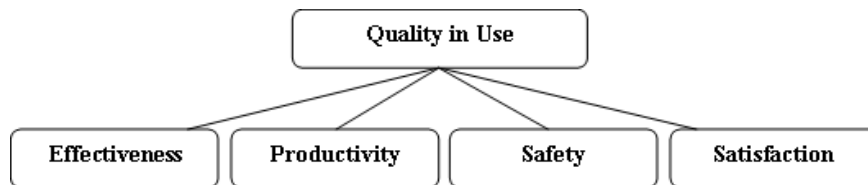


Figure 6. ISO Quality in Use

The internal and external quality category consists of six quality categories where each category consists of several quality attributes, figure 7. The categories are 1) Functionality that includes suitability, accuracy, interoperability, security, and functionality. 2) Reliability that include maturity, fault tolerance, recoverability, and reliability. 3) Usability that includes understandability, learnability, operability, attractiveness, and usability. 4) Efficiency that includes time behavior, resource utilization, and efficiency. 5) Maintainability that includes analyzability, changeability, stability, testability, and maintainability. 6) Portability that includes adaptability, installability, co-existence, replaceability, and portability.

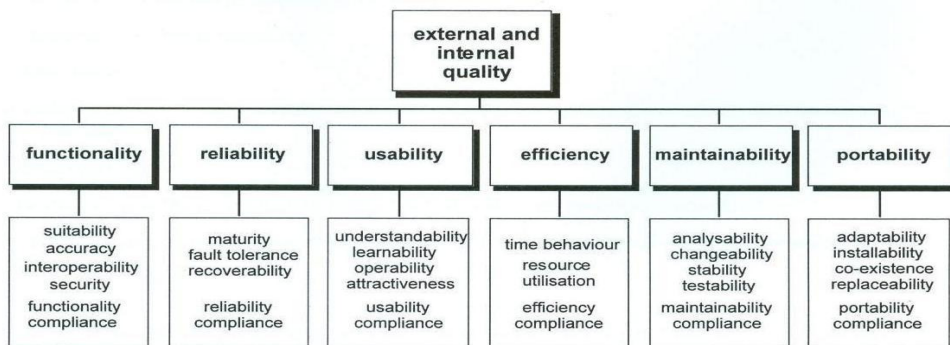


Figure 7. ISO External and Internal

4. QUALITY MODELS ANALYSIS

Analyzing all quality models by comparing their quality attributes or characteristics, Table 1 shows that total number of attributes or characteristics for each quality model. From the 17 quality attributes only one attribute 'reliability' that is common to all five quality model. Also, there are attributes such as 'portability', 'efficiency', and 'usability' that is common to four quality models. The attributes 'maintainability' and 'functionality' are common to three quality models. The attributes 'testability' and 'reusability' is common to two quality models. The remaining attributes 'human engineering', 'understandability', 'modifiability', 'interoperability', 'correctness', 'integrity', 'supportability', and 'performance', 'adabtability', 'installability', 'changeability', 'accuracy', 'maturity', 'suitability', 'resource utilization', 'analyzability', 'suitability', 'attractiveness', 'operability', each are common to only one quality model.

Out of the total number of attributes, it can be noted there are many attributes are shared with all quality models, but only the ISO 9126 covers more software quality attribute when compared to other quality models. Based on this quality models analysis, it is determined that ISO 9126 in its characteristics and sub-characteristics is most dependable as software quality metric to be used by software developers than other software quality models, figure 8.

Quality Models Attributes	Boehm	McCall	FURPS	Dromey	ISO 9126
Portability	√	√		√	√
Reliability	√	√	√	√	√
Efficiency	√	√		√	√
Human engineering	√				
Testability	√	√			√
Understandability	√				
Modifiability	√				
Maintainability		√		√	√
Flexibility		√			
Reusability		√		√	
Interoperability		√			
Correctness		√			
Usability		√	√	√	√
Integrity		√			
Supportability			√		
Performance			√		
Adaptability					√
Installability					√
Changeability					√
Accuracy					√
Maturity					√
Suitability					√
Resource Utilization					√
Analysability					√
Stability					√
Attractiveness					√
Operability					√
Functionalability			√	√	√
Total Number of Attributes	7	11	5	7	18

Figure 8. ISO 9126 Quality

5. COTS CLASSIFICATION

Commercial of the Shelf products can be classified into three categories architectural level, product kind, and life-cycle phase. The categories have several classification dimensions. With the different COTS, COTS are divided into classes such as 1) server side languages, 2) server side engines, 3) DBMS, 4) client execution engines, 5) client development tools, 6) programming languages, 7) development standards, 8) executable components, and 9) development tools [10]. Based on the software classes, it is found that the most number of COTS are to belong to the Server-Side Languages with (23 items) including Java Servlet, Java Beans, Pearl, CGI, CORBA, PHP, and SOAP. The second most used software belongs to classes such as client development tools, and client execution engines. The client development tools consists a large variety of software products including Paint Shop Pro, Macromedia, WAP, MS Frontpage, Netscape, and Oracle think9i. The client execution engines consists a large variety of software products including Acrobat reader, Lynx, Opera, and Winamp. The third category is Client-Side language That includes HTML, Java Applet, Java phone, Java script, and Macromedia. The forth category is DBMS that includes MySQL, Oracle, and Sybase. The fifth category is a server-side engine that includes Oracle Application Server and Apache HTTP server [11].

Most COTS products are to ensure many quality model attribute that can be achieved by following McCall or ISO 9126 quality models. To make sure that COTS products are to comply with software standards, COTS must pass their tests for quality attributes such as ‘adabtability’, ‘changeability’, ‘accuracy’, ‘suitability’, ‘resource utilization’, ‘analyzability’, ‘attractiveness’, ‘operability’, and fault tolerance’.

COTS and Quality Attributes	Server-Side Languages	Client Development Tools	Client Execution	Client-Side Language	DBMS	Server-Side Engine
Portability	√	√		√	√	√
Reliability	√	√	√	√	√	√
Efficiency	√	√	√	√	√	√
Human engineering			√			
Testability	√	√	√	√	√	√
Understandability	√	√				
Modifiability	√	√		√		√
Maintainability	√	√		√	√	√
Flexibility	√	√				√
Reusability	√	√	√	√	√	
Interoperability	√	√				
Correctness	√	√				
Usability	√	√	√	√	√	
Integrity	√	√	√	√		
Supportability	√		√			√
Performance	√	√	√			√
Adaptability	√	√	√		√	√
Installability	√	√	√			
Changeability	√				√	
Accuracy	√	√			√	√
Maturity	√	√	√			√
Suitability	√	√			√	
Resource Utilization	√	√	√		√	
Analyzability	√	√	√	√		√
Stability	√	√	√	√	√	√
Attractiveness			√	√	√	√
Operability	√	√	√	√	√	√
Functionalability	√		√	√	√	

Figure 9. COTS and ISO 9126 Quality

6. CONCLUSION

In this paper, we have addressed and presented the different types of the most common used quality models and their characteristics, and presented the most used COTS and their classified attributes to find their conjunction with the standard quality attributes based on their implemented quality attributes.

COTS products represent the already made software products for commercial and government agencies use rather than building a custom made software system from scratch. COTS products are made based on a set of attributes that follows on of the known software quality models. The quality of COTS is the reason to adapt a software product. The quality of COTS comes from the types of quality attributes implemented in it while building the software system product. The software implemented quality attributes are a set of characteristics that are contained in a specific quality model. There are a number of quality models, where each consist of quality attributes or characteristics that is presented in a software product.

COTS cover many quality attributes to make certain that software products are suitable for public use; and based on the COTS classifications quality attributes, different quality models can be use. Some software attributes can not be ignored and must be implemented in the COTS products such as ‘adabtability’, ‘changeability’, ‘accuracy’, ‘suitability’, ‘resource utilization’, ‘analyzability’, ‘attractiveness’, ‘operability’, and fault tolerance’.

Quality model analysis, articulates the different quality models and their quality attributes in their characteristics and sub-characteristics that are listed in the software quality metric. Based on quality models and software quality attributes, there are certain quality attributes used by software developers than other software quality models and they are found in ISO 9126.

Among the many used attributes that are clearly articulated in the ISO 9126 quality model in comparison with other quality models. With the outcome of Quality Models analysis and the results of COTS classifications, it is determined that most developed COTS products follow the ISO 9126 quality model than any other quality models since it covers more quality attributes that must be present in a computer software.

REFERENCES

- [1] Rawashdah, A., Matakah, B. A New Software Quality Model for Evaluation COTS Components. Journal of Computer Science, Science Publication. 2006.
- [2] Al-Qutaish, R. Quality Models in Software Engineering Literature: An Analytical and Comparative Study. Journal of American Science, 2010.
- [3] Franch, X., Carvallo, J. Using Quality Models in Software Package Selection. IEEE Software, IEEE Computer Society, 2003
- [4] Galorath, D. Software Reuse and Commercial Off-the-Shelf Software. Galorath Incorporation, El Segundo, CA.
- [5] Tomar, A. Thakare, V. A Systematic Study of Software Quality Models. In proceedings of the International Journal of Software Engineering & Applications (IJSEA), Vol.2, No.4, October 2011.
- [6] McCall, J. A., Rihcards, P. K., Walters, G. F. Factors in Software Quality, Volumes I, II, and III. US Rome Air Development Center Reports, US Department of Commerce, USA, 1977.
- [7] Dromey, R. G. A model for software product quality. IEEE Transactions on Software Engineering, 1995; 21:146-162.
- [8] Dromey, R. G. Concerning the Chimera [software quality]. IEEE Software, 1996; 13:33-43.
- [9] ISO. ISO/IEC 9126: Software Engineering Product Quality – Part 4: Quality Model. International Organization for Standardization, Geneva, Switzerland, 2004.
- [10] Jaccheri L., Torchiano, M. Classifying COTS Products. Department of Computer and Information Science. University of Science and Technology, Trondheim, Norway.
- [11] Morisio, M., Torchiano, M.: Definition and Classification of COTS: a proposal. Proc. of international conference on COTS based Software Systems (ICCBBS), pp 165-175, Orlando (FL), February 4-6, 2002