Analysis of the Meanings of the Measurement, Assessment, and Evaluation in ISO/IEEE Software Engineering Standards

Rafa E. AL-QUTAISH, PhD
Department of Management Information Systems
Al Ain University of Science & Technology - Abu Dhabi Campus
P.O. Box: 112612, Abu Dhabi
UAE
rafa@ieee.org

Laila CHEIKHI, PhD
École Nationale Supérieure d'Informatique et d'Analyse des Systèmes
Université Mohammed V- Souissi
BP: 713, Rabat
Morocco
cheikhi@ensias.ma

Abstract: - Software engineering international standards have been developed to eliminate the gaps between the practitioners in understanding and implementing the software engineering terms and techniques. However, in presence of the coordination lack between the standardization bodies and even between the different work-groups within the same standardization body, some of these standards came up with a set of conflicted definitions and meanings of the same terms. Thus, this will make it difficult or impossible for the practitioners to implement such standards in a correct and rigorous way. In this paper, the meanings of the '*measurement'*, '*assessment'*, and '*evaluation'* terms and the relationship between them in different ISO and IEEE standards will be discussed.

Key-Words: - Software Measurement, Software Assessment, Software Evaluation, Software Product Quality, ISO Standards, IEEE Standards, Software Engineering Standards.

1 Introduction

The International Organization for Standardization (ISO) and the Institute of Electrical and Electronic Engineers (IEEE) developed a set of software engineering international standards. Such standards are developed to assist the practitioners in dealing with the required procedures and terms in order to apply the software engineering techniques and frameworks in a standard way.

However, the international standards which have been produced by ISO are completely agreed on by the country members of ISO organization. Also, the IEEE software engineering standards are well known and widely used by practitioners around the globe. These agreements and usability should make such standards common and usable in software engineering enterprises.

One of the main problems with the ISO and IEEE standards is that there is no coordination between these standardization bodies when producing standards and the lake of coordination between the different work-groups which are responsible for developing different ISO standards. For example, some software engineering ISO and IEEE standards use the term '*measures*' while others use the term '*metrics*' to refer to the same concept.

In this paper, an analysis of three terms (i.e. *measurement*, *assessment*, and *evaluation*) will be conducted based on a number of ISO and IEEE software engineering standards, and the relationship between these terms will be discussed.

The rest of this paper is organized as the following: section 2 contains a brief overview of the related standard, that is, ISO 9126, ISO 14598, ISO 15939, and IEEE Std. 1061. In section 3, the detailed analysis of the meanings of the three terms along with the relationship between them will be discussed. Section 4 discusses the outcomes of the analysis. Finally, section 5 concludes the paper and introduces a potential future work.

2 Related ISO/IEEE Standards

2.1 ISO 9126

The ISO 9126 on software product quality is divided into four standards, that is:

1. ISO 9126-1 on quality models [1],
2. ISO 9126-2 on external quality metrics [2],
3. ISO 9126-3 on internal quality metrics [3], and
4. ISO 9126-4 on quality in-use metrics [4].

The first part contains two quality models for the internal/external quality and quality in-use. The internal/external software quality model contains six characteristics in which they contain twenty-seven sub-characteristics, while the quality in-use model contains four characteristics. In addition, this part introduces the definitions of the related terms and concepts.

The second, third, and fourth parts of the ISO 9126 international standard contains huge sets of metrics to be used to measure a specific characteristic for the internal/external quality or quality in-use for the intended software product.

2.2 ISO 14598

In addition to the four documents of the ISO 9126 series, the ISO also published a set of documents for guidelines on how to apply ISO 9126, which is called ISO 14598 and named as software product evaluation. The ISO 14598 series of standards consists of six parts, that is:

1. ISO 14598-1 on general overview [5],
2. ISO 14598-2 on planning and management [6],
3. ISO 14598-3 on process for developers [7],
4. ISO 14598-4 on process for acquirers [8],
5. ISO 14598-5 on process for evaluators [9], and
6. ISO 14598-6 on documentation of evaluation modules [10].

The first part of this series of international standards - *general overview* - contains an overview of the contents and the objectives of the other parts, defines a number of terms used in the other parts, and illustrates the relationship between the other five parts [5]. In addition, it clarifies the relationship between the quality model in the ISO 9126 part-1 and the ISO 14598 series of standards, includes the general requirements for the specification and the evaluation of the software quality, and presents a framework to evaluate the quality of all types of software product [5].

The second part - *planning and management* - presents details about the planning and management requirements that are associated with the software product evaluation, and it defines the requirements which should be provided by the organization in order to ensure the success of the evaluation process [6].

The third part of the ISO 14598 series of standards - *process for developers* - may be used to apply the concepts explained in the ISO 9126 series of standards and the ISO 14598 parts 1, 2, and 6 [7]. It gives recommendations and requirements for the practical implementation of the software product evaluation, in parallel with the development, by the developer [7]. This part of the ISO 14598 series of standards may be used by the project manager, software designer, quality assurance audit, maintainer, and/or software acquirer [7].

The fourth part - *process for acquirers* - includes requirements, recommendations and guidelines for the systematic *measurement*, *assessment*, and *evaluation* of the software product quality [8]. The evaluation process explained in this part of the ISO 14598 series of standards helps to meet the objectives and the goals of deciding on the acceptance of a single product or for selecting a product [8]. This part may be used by the project manager, system engineers, software engineering staff, and/or end users [8].

The fifth part - *process for evaluators* - is used to apply the concepts explained in ISO 9126 series of standards by providing requirements and recommendations for the practical implementation of the software product evaluation when several parties need to understand, accept, and trust the evaluation results [9]. The evaluation process explained in this part defines the activities needed to analyze the evaluation requirements, to specify, design, and perform the evaluation actions and to conclude the evaluation of any kind of software product [9]. This part of the ISO 14598 series may be used by testing laboratory evaluators, software suppliers, software acquirer, software users, and/or certification bodies [9].

Finally, the sixth part - *documentation of evaluation modules* - clarifies and defines the contents, the formation, and the structure of the documentation to be used to illustrate an evaluation module [10]. This part of the ISO 14598 series may be used by testing laboratories, research institutions

ISBN: 978-960-474-281-3

and organizations, and any others who need to produce new evaluation modules [10].

2.3 ISO 15939

Within this international standard, ISO produced an information model to help in determining what has to be specified during measurement planning, performance and evaluation [11].

In this international standard a specific measurement method is used to collect a base measure for a specific attribute (characteristics). Then, the values of two or more base measures can be used within a computational formula (by means of a measurement function) to produce and construct a specific derived measure.

These derived measures are then used in the context of an analysis model to arrive at an indicator which is a value, and to interpret the indicator's value to explain the relationship between it and the information needed, in the language of the measurement user, to produce an Information Product for his Information Needs [11].

This ISO standard contains a very mature measurement terminology, and it is well documented in the ISO International Vocabulary of Basic and General terms in Metrology (VIM) [12]. This terminology is widely accepted and used in most fields of science, and has been adopted in ISO 15939 as the agreed upon measurement terminology for software and system engineering related ISO standards.

2.4 IEEE Std. 1061

This IEEE standard provides a set of definitions, an overview of framework for software quality metrics, and a methodology for software quality metrics [13].

The framework is structured in three levels. In the first level, the quality requirements are expressed in terms of quality factors, which are decomposed in quality sub factors in the second level. Each quality sub factor is decomposed into metrics that allows measuring some aspects of software product or process during the whole software life cycle from top to bottom and bottom to up.

This standard use the term metric and it defines the software quality metric as: "*A function whose inputs are software data and whose output is a single numerical value that can be interpreted as the degree to which software possesses a given attribute that affects its quality*" [13]. This definition is adapted from the second definition of quality metric provided by the IEEE Std. 610.12 (IEEE Standard Glossary of Software Engineering Terminology). This (IEEE Std.610.12) standard provides a set of standardized vocabularies to be used by software engineering community allowing therefore conflicted definitions and meanings of the same term. Furthermore, it also uses the term '*metric*' and is defined as: "*a quantitative measure of the degree to which a system, component, or process possesses a given attribute*" [14].

3 The Different Definitions

The following is a list of the definitions for the *measurement* term taken from different software engineering international standards:

Definition-(1): *act or process of assigning a number or category to an entity to describe an attribute of that entity* [13].

***Definition-(2*)**: the assignment of numbers to objects in a systematic way to represent properties of the object [15].

Definition-(3): *the use of a metric to assign a value (which may be a number or category) from a scale to an attribute of an entity* [5].

Definition-(4): *set of operations having the object of determining a value of a measure* [12].

Definition-(5): *an algorithm or calculation performed to combine two or more base measures* [11, 16].

Definition-(6): *the assignment of values and labels to aspects of software engineering (products, processes, and resources) and the models that are derived from them, whether these models are developed using statistical, expert knowledge or other techniques* [15].

Definition-(7): *assigning relative value* [17].

Definition-(8): *a figure, extent, or amount obtained by measuring* [13].

In addition, the following is the only definition for the *assessment* term:

Definition-(9): *an action of applying specific documented criteria to a specific software module, package or product for the purpose of*

determining acceptance or release of the software module, package or product [18].

Furthermore, the definitions of the *evaluation* term are following:

Definition-(10): *a systematic determination of the extent to which an entity meets its specified criteria* [19].

Definition-(11): *an action that assesses the value of something* [20].

4 Discussion

The *measurement definitions (1), (2), (3), (5), (6), (7),* and *(8)* above did not contain any kind of measurement function. However, it is well-known that the measurement needs a measurement function to explain the procedure and to get a result. Furthermore, *definitions (2), (4), (7),* and *(8)* state that the *measurement* is an assignment of numbers, figures, values; whereas, *definitions (1), (3),* and *(6)* state that the *measurement* is an assignment of numbers (values) or labels (categories). The result of the measurement process should be numbers (objective measurement) or labels (subjective measurement). The meaningful definition should state that the *measurement* is the process of using a measurement function to derive and assign a number or category to describe an attribute (characteristic) of an entity.

The *assessment* definition contains some thing not clear, that is, "*for the purpose of determining acceptance or release*". However, the acceptance or rejection of a software product should be based on an evaluation process and not only an assessment, see Figure 1.

For the *evaluation* term, *definition (10)* is completely misunderstood since the "*action that assesses the value of something*" related to the *measurement* and not to the *evaluation* process.

The evaluation process should be specified, for example, Sahari *et al.* [21] has specified that the evaluation is used - in their paper - to determine the suitability and accuracy of content. Also, Arh and Blažič [22] have used the evaluation to determine the usability of end users.

A misunderstanding of the differences between *measurement*, *assessment*, and *evaluation* processes is appear in some researches, such as, in [21], [22], and [23].

The standard bodies should recognize the differences between the objective and subjective measurements. However, some of the measurement definitions in Section 3 can work with the objective only and other can work with the subjective. As well-known, the objective measurement is a real measurement since it based on numbers; while, the subjective measurement is completely based on judgment.

Figure 1 below illustrates the intended relationships between the *measurement*, *assessment*, and *evaluation*. Thus, the evaluation process cannot be performed without conducting an assessment process, and the assessment process needs the results of a measurement process.

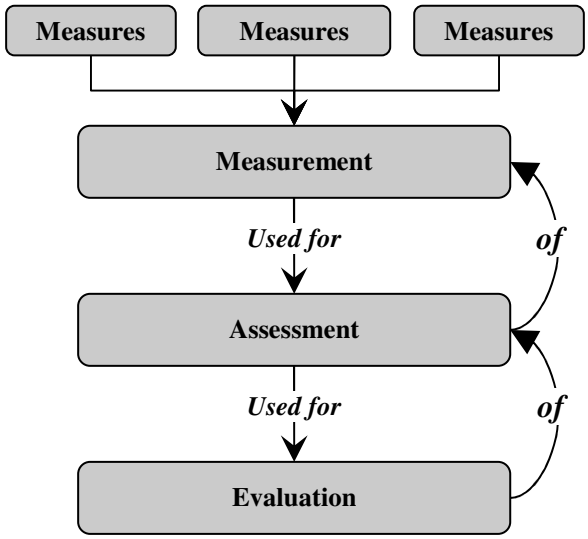

Figure 1: The Relationship between *Measurement*, *Assessment*, and *Evaluation*.

The *measurement* process needs to use a set of measures (metrics) in order to apply a *measurement* function to determine the results of such *measurement* process. As a standard, such measures (metrics) can be taken only from ISO 9126 parts 2, 3, and 4.

5 Conclusion and Future Work

Software engineering international standards have been developed to eliminate the gaps between the practitioners in understanding and implementing the software engineering terms and procedures.

ISBN: 978-960-474-281-3

However, in presence of the coordination lack between the standardization bodies and even between the different work-groups within the same standardization body, some of these software engineering standards came up with conflicted definitions and meanings of the same terms. Thus, this will make it difficult or impossible for the practitioners to implement such standards in a correct and rigorous way. In this paper, the meanings of the *measurement*, *assessment*, and *evaluation* terms and the relationship between them in different ISO and IEEE standards are discussed.

This paper concludes that some of the definitions are incomplete or not understandable while others are incorrect. In addition, the relationship between such terms is valuable since the *measurement* of a software product will lead to assessment of that software product, and its *assessment* will lead to its *evaluation*, at the last.

As a future work, a software measurement program framework will be constructed based on the results of this paper and based on the relationship between the *measurement*, *assessment*, and *evaluation* processes. In addition, a comprehensive definition of each of these terms will be introduced to abstain the misunderstanding of the previous-discussed week definitions.

References:

[1] ISO, *ISO/IEC IS 9126-1: Software Engineering - Product Quality - Part 1- Quality Model*, International Organization for Standardization Geneva, Switzerland, 2001.

[2] ISO, *ISO/IEC TR 9126-2: Software Engineering - Product Quality - Part 2- External Metrics*, International Organization for Standardization, Geneva, Switzerland, 2003.

[3] ISO, *ISO/IEC TR 9126-3: Software Engineering - Product Quality - Part 3- Internal Metrics*, International Organization for Standardization, Geneva, Switzerland, 2003.

[4] ISO, *ISO/IEC TR 9126-4: Software Engineering - Product Quality - Part 4- Quality in Use Metrics*, International Organization for Standardization, Geneva, Switzerland, 2004.

[5] ISO, *ISO/IEC 14598-1: Information Technology - Software Product Evaluation - Part 1- General Overview*, International Organization for Standardization, Geneva, Switzerland, 1999.

[6] ISO, *ISO/IEC 14598-2: Software Engineering - Product Evaluation - Part 2- Planning and Management*, International Organization for Standardization, Geneva, Switzerland, 2000.

[7] ISO, *ISO/IEC 14598-3: Software Engineering - Product Evaluation - Part 3- Process for Developers*, International Organization for Standardization, Geneva, Switzerland, 2000.

[8] ISO, *ISO/IEC 14598-4: Software Engineering - Product Evaluation - Part 4- Process for Acquirers*, International Organization for Standardization, Geneva, Switzerland, 1999.

[9] ISO, *ISO/IEC 14598-5: Information Technology - Software Product Evaluation - Part 5- Process for Evaluators*, International Organization for Standardization, Geneva, Switzerland, 1998.

[10] ISO, *ISO/IEC 14598-6: Software Engineering - Product Evaluation - Part 6- Documentation of Evaluation Modules*, International Organization for Standardization, Geneva, Switzerland, 2001.

[11] ISO, *ISO/IEC IS 15939: Information Technology - Software Engineering - Software Measurement Process*, International Organization for Standardization, Geneva, Switzerland, 2001.

[12] ISO, *International Vocabulary of Basic and General Terms in Metrology (VIM)*, International Organization for Standardization, Geneva, Switzerland, 1993.

[13] IEEE, *IEEE Std. 1061: IEEE Standard for Software Quality Metrics Methodology*, Institute of Electrical and Electronics Engineers, New York, NY, USA, 1998.

[14] IEEE, *IEEE Std. 610.12: IEEE Standard Glossary of Software Engineering Terminology*, Institute of Electrical and Electronics Engineers, New York, NY, USA, 1990.

[15] ISO, *ISO/IEC TR 19759: Software Engineering - Guide to the Software Engineering Body of Knowledge (SWEBOK)*, International Organization for Standardization, Geneva, Switzerland, 2005.

[16] ISO, *ISO/IEC 25000: Software Engineering - Software product Quality Requirements and Evaluation (SQuaRE) - Guide to SQuaRE*, International Organization for Standardization, Geneva, Switzerland, 2005.

[17] ISO, *ISO/IEC 20926: Software Engineering - IFPUG 4.1 Unadjusted Functional Size Measurement Method - Counting Practices Manual*, International Organization for Standardization, Geneva, Switzerland, 2003.

[18] ISO, *ISO/IEC 14102: Information Technology - Guideline for the Evaluation and Selection of CASE Tools*, International Organization for Standardization, Geneva, Switzerland, 2008.

[19] ISO, *ISO/IEC 12207: Systems and Software Engineering - Software Life Cycle Processes*, International Organization for Standardization, Geneva, Switzerland, 2008.

[20] ISO, *ISO/IEC 15414: Information Technology - Open Distributed Processing - Reference Model - Enterprise Language*, International Organization for Standardization, Geneva, Switzerland, 2006.

[21] Noraidah Sahari, Hairuliza Mohd. Judi, Abdul Azim Abdul Ghani, Hasan Selamat, and Aida Suraya Md. Yunus, Construction and Selection of Usefulness Evaluation Items, *International Journal of Education and Information Technologies*, Vol. 1, No. 3, 2009, pp. 28-35.

[22] Tanja Arh and Borka Jerman Blažič, A Case Study of Usability Testing - the SUMI Evaluation Approach of the EducaNext Portal, *WSEAS Transactions on Information Science and Applications*, Vol. 5, No. 2, 2008, pp. 175-181.

[23] Mihaela Elena Idomir, Valentin Idomir, and Angela Repanovici, Web-Based Application for Self Testing and Assessment, *International Journal of Education and Information Technologies*, Vol. 5, No. 2, 2011, pp. 141-148.