TECHNICAL REPORT

ISO/IEC TR 25021

First edition
2007-10-15

Software engineering — Software product Quality Requirements and Evaluation (SQuaRE) — Quality measure elements

Ingénierie du logiciel — Exigences de qualité et évaluation du produit logiciel (SQuaRE) — Éléments de mesure de la qualité

Reference number
ISO/IEC TR 25021:2007(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

In exceptional circumstances, the joint technical committee may propose the publication of a Technical Report of one of the following types:

— type 1, when the required support cannot be obtained for the publication of an International Standard, despite repeated efforts;

— type 2, when the subject is still under technical development or where for any other reason there is the future but not immediate possibility of an agreement on an International Standard;

— type 3, when the joint technical committee has collected data of a different kind from that which is normally published as an International Standard ("state of the art", for example).

Technical Reports of types 1 and 2 are subject to review within three years of publication, to decide whether they can be transformed into International Standards. Technical Reports of type 3 do not necessarily have to be reviewed until the data they provide are considered to be no longer valid or useful.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC TR 25021, which is a Technical Report of type 2, was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Software and systems engineering*.

The SQuaRE series of standards consists of the following divisions under the general title *Software product Quality Requirements and Evaluation:*

— Quality Management Division,

— Quality Model Division,

— Quality Measurement Division,

— Quality Requirements Division, and

— Quality Evaluation Division.

Introduction

The purpose of this Technical Report is to define an initial set of quality measure elements to be used throughout the software product life cycle for the purpose of Software Product Quality Requirement and Evaluation (SQuaRE). While the quality measure elements can be used for standalone measurement, their main purpose is to be used as the building blocks for other SQuaRE measures as described in ISO/IEC TR 9126-2, ISO/IEC TR 9126-3 and ISO/IEC TR 9126-4. The content of this Technical Report constitutes the link between ISO/IEC 9126 and the subsequent SQuaRE series of standards (Figure 1).

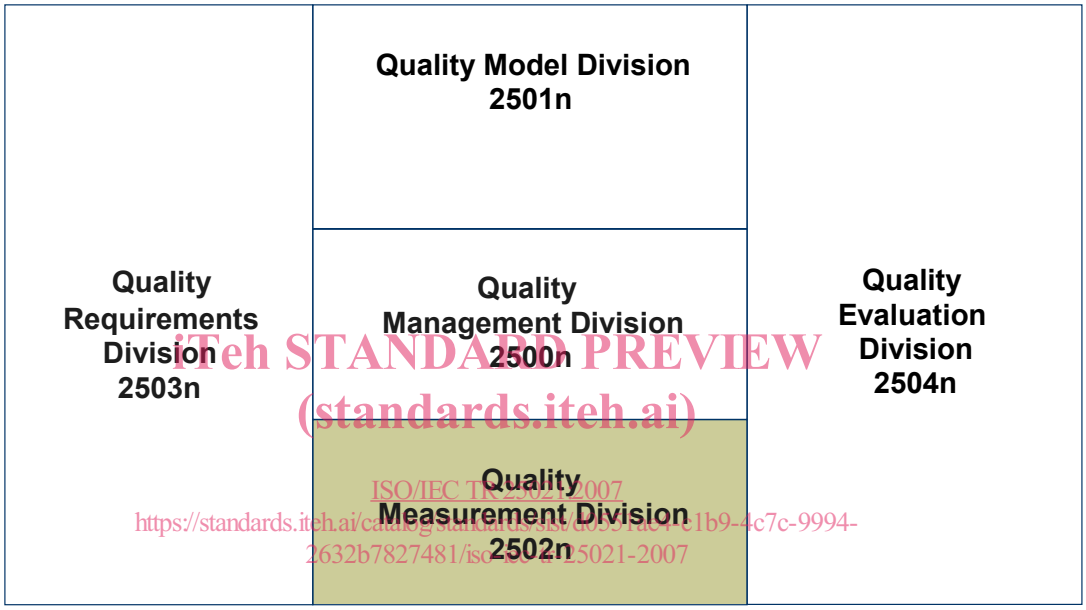

Figure 1 — Organization of SQuaRE series of standards

During the period of transition from a Technical Report to an International Standard, the intention of this Technical Report is to be only a source of support for the collection of quality measure elements. During this trial period, the applicability of this Technical Report and its content will be evaluated in the user environment. The support provided by this Technical Report for the measurement of software product quality is twofold: (a) assist in the selection of the required quality measure elements for a given quality measure, and (b) provide guidance for collecting the selected measurements.

The set of quality measures listed in this Technical Report arise from surveys involving several large commercial and academic institutions conducted during the following research projects:

— Quality Measure Validation Survey (Prague University of Economics research project GACR 201/06/0175 and Czech University of Life Sciences in Prague research projects MSM6046070904 and 2C06004);

— Conformity of Industrial Software to International Standards ES (Excellent Software) Mark (Korean Agency for Technology and Standards, Korea);

— Project for Development fund of MII (China Ministry of Information Industry): software engineering standardization.

NOTE This set does not contain quality measure elements for all ISO/IEC 9126-1 Quality Model subcharacteristics. Some subcharacteristics were omitted because survey results did not find evidence of their use. It does not imply the ISO/IEC 9126-1 Quality Model should be changed, but rather new quality measures may need to be defined in the future.

Subsequently the quality measure elements necessary for measurement of the above quality measures set have been identified and documented. These quality measure elements represent an initial set of measures which can be used during the construction of quality measures referenced in ISO/IEC TR 9126-2, ISO/IEC TR 9126-3 and ISO/IEC TR 9126-4, as well as other measures for other purposes. Quality measures described in the SQuaRE series (Figure 2) are derived from one or more quality measure elements described in this Technical Report. When evaluating a selected quality measure, the user should first review and evaluate the relevant quality measure element(s) listed in this Technical Report.

Some important benefits from using the measures in this Technical Report are the following.

— To improve measurement productivity and consistency which will minimize measurement effort: When using quality measures such as internal, external and quality in use, there is a possibility of duplicating the attribute measurement tasks, because they are usually performed as separate activities. However by identifying the set of quality measure elements that are uniquely required to derive all the quality measures for a given product, these measurements need to be performed only once at the attribute level. This approach will improve the productivity of the measurement processes.

— Usability (guidance, cross reference): This Technical Report allows users to identify the possible indicators of quality (quality measures) that can be derived by measuring one or more measures from a selected set of quality measure elements, and thereby maximizing the benefits from the measurement process.

The quality measure elements are the common components of quality measures. The intention here is that users of this Technical Report will select measures from quality measure elements for the purpose of defining internal, external or quality-in-use measures. These are then used for the definition of quality requirements (ISO/IEC 25030), software product evaluation (ISO/IEC 2504n), quality assessment and other purposes. It is therefore strongly recommended to use this Technical Report and the other documents in the Quality Measurement Division together with the other relevant documents in the ISO/IEC 25000 series.

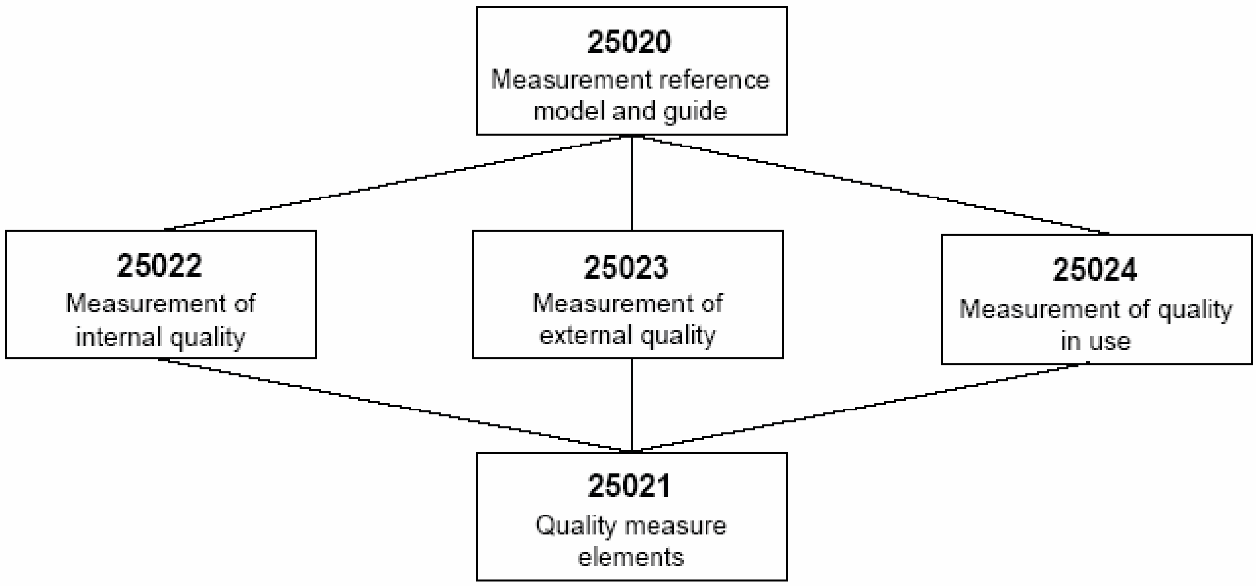

Figure 2 — Structure of the Quality Measurement Division

Quality measure elements, together with the other support documentation offered in this Technical Report, provide added value to the ISO/IEC 9126 Technical Reports and International Standards by making them more understandable to users by clearly defining the relevant quality measure elements (Figure 3). In this sense, this Technical Report acts effectively as the common link between ISO/IEC 9126 and its follower, the SQuaRE series.

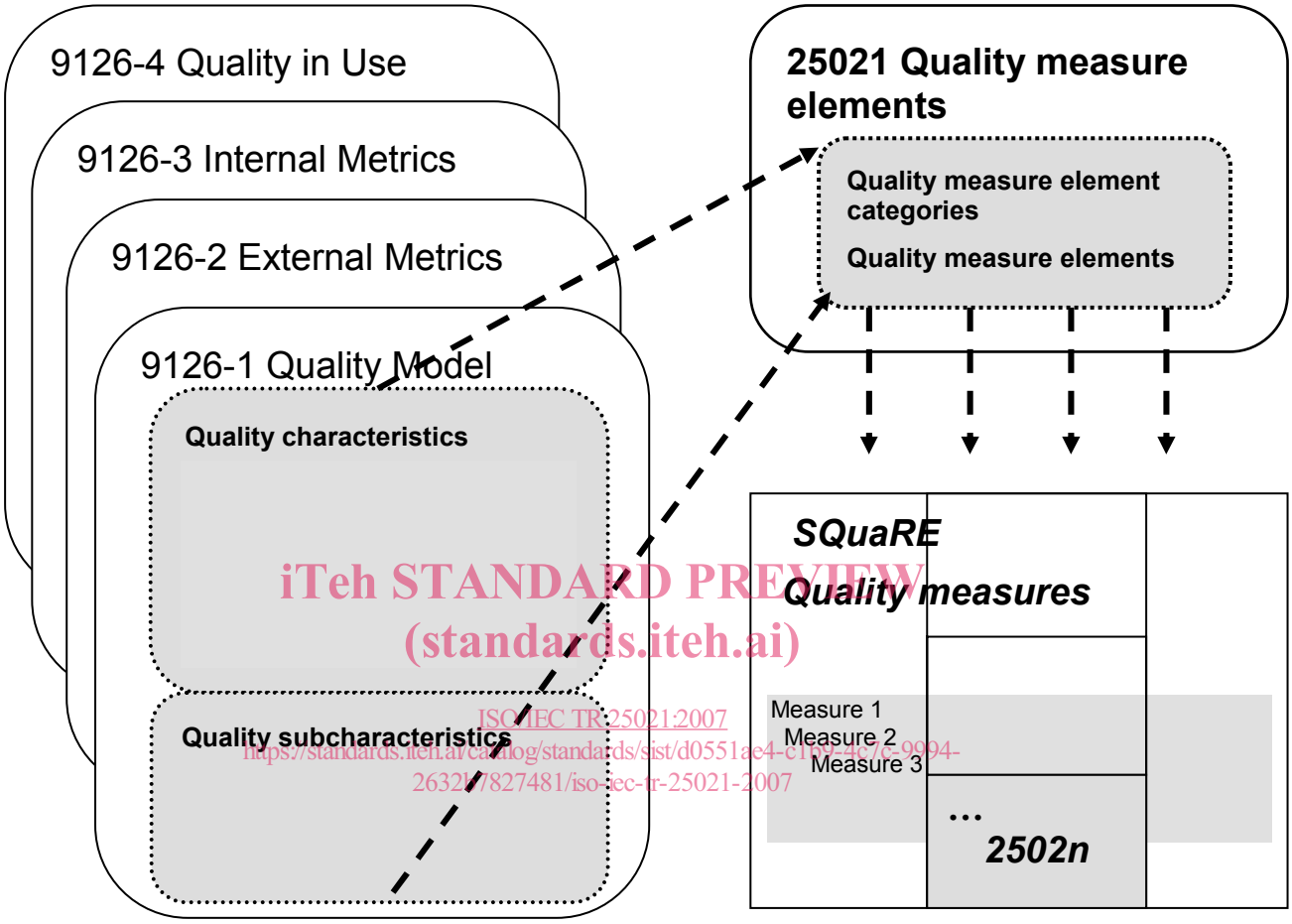

Figure 3 — The relationship of ISO/IEC TR 25021 as a link between the ISO/IEC 9126 and the SQuaRE series of standards

Software engineering — Software product Quality Requirements and Evaluation (SQuaRE) — Quality measure elements

1 Scope

This Technical Report specifies an initial set of quality measure elements in order to assist users of ISO/IEC TR 9126-2, ISO/IEC TR 9126-3 and ISO/IEC TR 9126-4 and users of the SQuaRE series of quality measurement standards ISO/IEC 2502n in the selection and use of the quality measures for software product quality evaluation and in the selection of the entities to be measured in the software product lifecycle.

This Technical Report contains

a) a description of the concept of quality measure elements,

b) considerations for using quality measure elements,

c) a set of quality measure elements.

This Technical Report is intended for, but not limited to, developers, acquirers and independent evaluators of software product, particularly those responsible for defining software product quality requirements and for software product evaluation.

2 Conformance

There are no conformance requirements in this Technical Report.

3 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the cited edition applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 25000, *Software Engineering – Software product Quality Requirements and Evaluation (SQuaRE) – Guide to SQuaRE*

ISO/IEC 25020, *Software engineering – Software product Quality Requirements and Evaluation (SQuaRE) – Measurement reference model and guide*

ISO/IEC 15939, *Systems and software engineering – Measurement process*

ISO/IEC TR 9126-2, *Software engineering — Product quality — Part 2: External metrics*

ISO/IEC TR 9126-3, *Software engineering — Product quality — Part 3: Internal metrics*

ISO/IEC TR 9126-4, *Software engineering — Product quality — Part 4: Quality in use metrics*

4 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 25000, ISO/IEC 25020 and ISO/IEC 15939 apply.

NOTE The following definitions are replicated here for the convenience of the user.

4.1
attribute
inherent property or characteristic of an entity that can be distinguished quantitatively or qualitatively by human or automated means

NOTE 1 Based on ISO/IEC 15939:2007.

NOTE 2 ISO 9000 distinguishes two types of attributes: a permanent characteristic existing inherently in something; and an assigned characteristic of a product, process or system (e.g. the price of a product, the owner of a product). The assigned characteristic is not an inherent quality characteristic of that product, process or system.

4.2
base measure
measure defined in terms of an attribute and the method for quantifying it

NOTE A base measure is functionally independent of other measures.

[ISO/IEC 15939:2007, based on the definition in the International Vocabulary of Basic and General Terms in Metrology, 1993]

4.3
derived measure
measure that is defined as a function of two or more values of base measures

[ISO/IEC 15939:2007, based on the definition in the International Vocabulary of Basic and General Terms in Metrology, 1993]

NOTE A transformation of a base measure using a mathematical function can also be considered as a derived measure.

4.4
external software quality
capability of a software product to enable the behaviour of a system to satisfy stated and implied needs when the system is used under specified conditions

NOTE Attributes of the behaviour can be verified and/or validated by executing the software product during testing and operation.

EXAMPLE The number of failures found during testing is an external software quality measure related to the number of faults present in the program. The two measures are not necessarily identical since testing may not find all faults, and a fault may give rise to apparently different failures in different circumstances.

4.5
indicator
measure that provides an estimate or evaluation of specified attributes derived from a model with respect to defined information needs

[ISO/IEC 15939:2007]

NOTE In ISO/IEC 14598, this definition of “indicator” was: “a measure that can be used to estimate or predict another measure”.

4.6
information need
insight necessary to manage objectives, goals, risks and problems

[ISO/IEC 15939:2007]

4.7
internal software quality
capability of a set of static attributes of a software product to satisfy stated and implied needs when the software product is used under specified conditions

NOTE 1 Static attributes include those that relate to the software architecture, structure and its components.

NOTE 2 Static attributes can be verified by review, inspection and/or automated tools.

EXAMPLE The number of lines of code, complexity measures and the number of faults found in a walk through are all internal software quality measures made on the product itself.

4.8
measure, noun
variable to which a value is assigned as the result of measurement

NOTE The term “measures” is used to refer collectively to base measures, derived measures, and indicators.

[ISO/IEC 15939:2007]

4.9
measure, verb
make a measurement

[ISO/IEC 14598-1:1999]

4.10
measurement
set of operations having the object of determining a value of a measure

[ISO/IEC 15939:2007, based on the definition in the International Vocabulary of Basic and General Terms in Metrology, 1993]

NOTE Measurement can include assigning a qualitative category such as the language of a source program (ADA, C, COBOL, etc.).

4.11
measurement function
algorithm or calculation performed to combine two or more base measures

[ISO/IEC 15939:2007]

4.12
measurement method
logical sequence of operations, described generically, used in quantifying an attribute with respect to a specified scale

[ISO/IEC 15939:2007, based on the definition in the International Vocabulary of Basic and General Terms in Metrology, 1993]

4.13
quality in use (measure)
extent to which a product used by specific users meets their needs to achieve specific goals with effectiveness, productivity, safety and satisfaction in specific contexts of use

4.14
quality measure element
base measure or derived measure that is used for constructing software quality measures

NOTE The software quality characteristics or subcharacteristics of the entity are derived afterwards by calculating a software quality measure.

5 Symbols and abbreviated terms

For the purposes of this Technical Report, the symbols and abbreviations given in ISO/IEC 25000, ISO/IEC 25020 and the following apply.

QME Quality Measure Element

SPQM-RM Software Product Quality Measurement Reference Model

6 Quality measure elements concept

Quality measure elements are used throughout the software product lifecycle as an input for the internal, external and quality in use measures listed in ISO/IEC TR 9126-2, ISO/IEC TR 9126-3 and ISO/IEC TR 9126-4. They measure:

— Attributes of resources consumed, or activities performed during the software product development, testing, and maintenance that relate to software product quality,

— Attributes of the software product itself,

— Attributes of the specific context of use,

— Attributes of the software product when used in a specific context of use, and the effects connected with such use (effects on the user and the result of their task).

6.1 Quality measure elements in the Software Product Quality Measurement Reference Model (SPQM-RM)

International Standard ISO/IEC 25020 defines the Software Product Quality Measurement Reference Model (SPQM-RM) to be used for the software product quality requirements specification process (top-down approach) and for the software product quality evaluation process (bottom-up approach).

SPQM-RM (Figure 4) defines the position of quality measure elements in the software product quality measures definition process. Any single quality measure element by itself generally will not indicate the quality of the measured entity. The quality measure is derived afterwards by computing the results of quality measure element values as stated in the formula of the quality measure that is being measured.

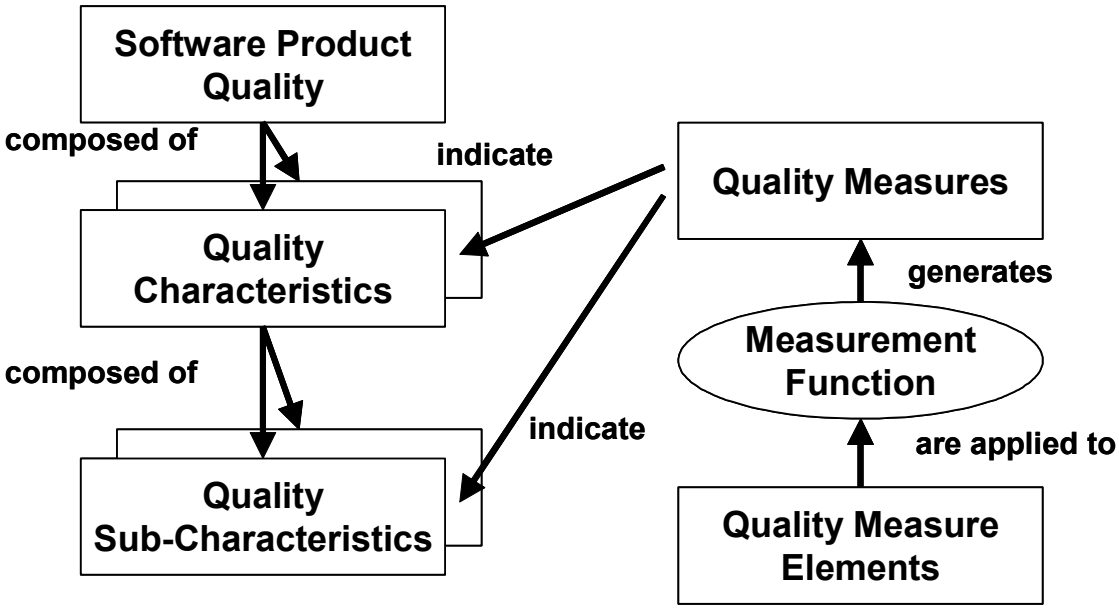

Figure 4 — Quality measure elements concept in SPQM-RM

A quality measure is derived from one or more quality measure elements. A quality measure element can be an input to more than one quality measure. This relationship can be represented in the form of a two-dimensional cross-reference table, which allows a user of this Technical Report to select the proper quality measure elements that are necessary for measuring an individual set of quality measures. It also allows the user of this Technical Report to identify quality measures, which could be derived from a given set of quality measure elements.

The layout of the cross-reference table is shown in Table 1. In this table the quality measures and quality measure elements can be any of those stated in this document. This table works in both ways: by row, a cross associates a quality measure element to all the quality measures where its usage applies; by column, all the crosses indicate the quality measure elements that are required for the quality measure in that column.

Table 1 — Layout of the cross-reference table representing the relationship between QMEs and quality measures

	Quality measure 1	Quality measure 2	Quality measure 3
Quality measure element 1	X	X	
Quality measure element 2	X		X
Quality measure element 3			X

When evaluating the quality measure elements, a user should be familiar with the information given in the quality measure element table and consider all the limitations and rules given for a listed quality measure element, prior to applying the formula to obtain the value of the quality measure.

6.2 Quality measure elements and their categories

Quality measures in ISO/IEC TR 9126-2, ISO/IEC TR 9126-3 and ISO/IEC TR 9126-4 are described in different level of detail. Some of them are specified precisely, such that the QME and their measurement methods could be easily derived from them; others are described in a way that permits different QME and measurement methods to be used. The concept of the QME Category is used to indicate this difference.

6.3 Aspects of quality measure elements

Users of this Technical Report should consider each of the aspects listed in this section (while evaluating the quality measure elements) in order to avoid accidental misuse of the measurement process, and possible omission of essential measurement steps.

The purpose of quality measure elements is to measure different attributes of a software product or process at any stage during the software product lifecycle. This requires consideration of different aspects of the quality measure elements construction and application of specific practices, such as data selection, measurement process, documentation, arithmetic operations, etc. that are described in each clause.

For the purpose of this Technical Report these aspects are:

— Measurement scale type – the scale type used for measurement,

— Measurement focus in the software product quality lifecycle – the scope and objective of the measurement (e.g. the software product itself, software product in a system, software product in a system used by a specified user in a specified scenario),

— Measurement method type – the measurement method type relating to the quality measure element used for measurement.

These aspects are followed with an essential, but not exclusive set of considerations. While measuring any quality measure element a user of this Technical Report should examine relevant considerations from all the above listed aspects.

NOTE 1 The following aspects and considerations relate to common issues that are relevant to the collection or measurement of (non-exclusive) sets of QMEs.

NOTE 2 Some of the described considerations (e.g. nominal scale etc.) are not relevant for any of the QME listed in this Technical Report. They have bee included to bring the complete description of each aspect and allow measurement of QMEs not listed in this Technical Report.

6.3.1 Measurement scale type

The type of scale depends on the nature of the relationship between values on the scale. Five types of scales are commonly used and connected considerations are:

NOTE Scale is defined in ISO/IEC 25000. The following are examples of types of scale.

Nominal scale - The purpose of nominal scale type measures is to classify measured attributes. No ordering is implied even if numbers are used. The numbers assigned to measures in nominal scale type measurement identify only the category (type) of measured attribute. Therefore the order, minimum, maximum, median, arithmetic mean, percentage, etc. derived from these numbers do not have any empirical meaning.

EXAMPLE Software product fault types or categories. Type of function based on the software product quality model (e.g., type of quality characteristics or subcharacteristics)

Ordinal scale - The purpose of ordinal scale type measures is to assign an order to measured attributes. The ordinal scale is often useful to augment the nominal scale with information about an ordering of the classes or categories. The minimum, maximum and median derived from the measures in the ordinal scale type

measurement have empirical meaning. The arithmetic mean, percentage etc. do not have an empirical meaning.

EXAMPLE Software product failure by severity (e.g. negligible, marginal, critical, catastrophic)

Interval scale - The purpose of interval scale type measures is to measure a difference between measures. If a ratio is calculated from the quality measure elements of the same type, it does not have an empirical meaning.

EXAMPLE Cyclomatic complexity has a minimum value of one, but each increment represents an additional path. The value of zero is not possible.

Ratio scale - The ratio scale is similar to the interval scale but includes the value zero, representing total lack of an attribute. Ratio scale quality measure elements include ordered rating scales, where the difference between two measures, and the ratio of two measures, has the same empirical meaning. Ratio and average are giving meaning to the values.

EXAMPLE Effort (time) spent to change, Buffer size, Number of detected faults

Absolute scale - The purpose of absolute scale type measures is to measure a proportion between measures with the same units. Absolute scale quality measure elements include measures resulting from dividing one ratio scale measure to another ratio scale measure where the measurement unit is the same in both cases.

EXAMPLE Number of comment lines divided by total lines of code

6.3.2 Measurement focus in software product quality lifecycle

NOTE ISO/IEC 25020 defines the relationship between the software product quality lifecycle and the quality measures. A short summary of these relationships is also described in clause 6.1 Quality measure elements in the Software Product Quality Measurement Reference Model (SPQM-RM).

Internal software quality measure - The purpose of internal software quality measures is to measure internal attributes of a software product. Internal attributes are those that can be measured in terms of the software product itself, i.e., separate from its behaviour. Internal software quality measures provide the users with the ability to measure the internal quality characteristics of the intermediate deliverables and thereby manage the development of the product to achieve the specified quality requirements. The quality measure element can be measured only if the user has access to the software product structure (source code, design specifications etc.).

NOTE Internal software quality is defined in ISO/IEC 25000

EXAMPLE The number of lines of code, complexity measures and the number of faults found in a walk through are all internal software quality measures made on the product itself.

External software quality measure - The purpose of external software quality measures is to measure external attributes of a software product. External attributes are those that can be measured only with regard to how the product relates to its environment, i.e., by observing its behaviour in the system environment in which it is intended to operate, or in a similar environment. External software quality measures provide users, evaluators, testers, and developers with the insight into the quality of the software product by collecting measures during the execution of the software product. The value of a quality measure element of that type is influenced either by the attributes of the product itself or by the attributes of the process of using it.

NOTE External software quality is defined in ISO/IEC 25000

EXAMPLE The number of failures found during testing – it could be influenced by quality of the software product but also by the testing process.

Quality in use measure - The purpose of quality in use measures is to measure quality in use attributes. Quality in use attributes are those that can be measured when executing the software product in a specified