INTERNATIONAL STANDARD

ISO/IEC 14598-1

First edition
1999-04-15

Information technology — Software product evaluation —

Part 1:
General overview

Technologies de l'information — Évaluation de produits logiciels —

Partie 1: Aperçu général

Reference number
ISO/IEC 14598-1:1999(E)

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS

Not for Resale

Contents

1 Scope 1

2 Conformance 1

3 Normative references 1

4 Terms and definitions 2

5 Overview of ISO/IEC 14598 and ISO/IEC 9126 6

5.1 Structure of ISO/IEC 14598 and ISO/IEC 9126 6

5.2 Evaluation process 6

5.3 Support for evaluation 6

5.4 Software quality characteristics and metrics 7

6 Evaluation process 8

7 Establish evaluation requirements 9

7.1 Establish the purpose of evaluation 9

7.2 Identify types of product(s) to be evaluated 11

7.3 Specify quality model 13

8 Specify the evaluation 14

8.1 Select metrics 14

8.2 Establish rating levels for metrics 15

8.3 Establish criteria for assessment 16

9 Design the evaluation 16

9.1 Produce evaluation plan 16

10 Execute the evaluation 16

10.1 Take measures 16

10.2 Compare with criteria 17

10.3 Assess results 17

11 Supporting processes 17

Bibliography 18

© ISO/IEC 1999

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and micro-film, without permission in writing from the publisher.

ISO/IEC Copyright Office • Case postale 56 • CH-1211 Genève 20 • Switzerland

Printed in Switzerland

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

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. 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.

International Standard ISO/IEC 14598-1 was prepared by Joint Technical Committee ISO/IEC JTC 1 *Information technology,* Subcommittee SC 7, *Software engineering*.

ISO/IEC 14598-1 is intended for use in conjunction with ISO/IEC 9126-1 (in preparation) which will replace ISO/IEC 9126 (1991).

ISO/IEC 14598 consists of the following parts under the general title *Information technology — Software product evaluation*:

— *Part 1: General overview*

— *Part 2: Planning and management*

— *Part 3: Process for developers*

— *Part 4: Process for acquirers*

— *Part 5: Process for evaluators*

— *Part 6: Documentation of evaluation modules*

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Introduction

As the use of information technology grows, the number of critical computer systems also grows. Such systems include for example, security critical, life critical, economically critical and safety critical systems. The quality of software in these systems is particularly important because software faults may lead to serious consequences.

Throughout the history of software engineering, software quality improvement has been a most important goal. The evaluation of software product quality is vital to both the acquisition and development of software which meets quality requirements. The relative importance of the various characteristics of software quality depends on the mission or objectives of the system of which it is a part; software products need to be evaluated to decide whether relevant quality characteristics meet the requirements of the system.

The essential parts of software quality evaluation are a quality model, the method of evaluation, software measurement, and supporting tools. To develop good software, quality requirements should be specified, the software quality assurance process should be planned, implemented and controlled, and both intermediate products and end products should be evaluated. To achieve objective software quality evaluations, the quality attributes of the software should be measured using validated metrics.

The term "metric" has been used in many senses in software engineering publications. In this international standard it is defined as a quantitative scale and method which can be used for measurement. The word "measure" is used to refer to the result of a measurement.

The ISO/IEC 14598 series of standards give methods for measurement, assessment and evaluation of software product quality. They describe neither methods for evaluating software production processes nor methods for cost prediction (software product quality measurements may, of course, be used for both these purposes).

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Information technology — Software product evaluation — Part 1: General overview

1 Scope

This part of ISO/IEC 14598 introduces the other parts. It provides an overview of the other parts and explains the relationship between ISO/IEC 14598 and the quality model in ISO/IEC 9126. This part of ISO/IEC 14598 defines the technical terms used in the other parts, contains general requirements for specification and evaluation of software quality and clarifies the general concepts. Additionally, it provides a framework for evaluating the quality of all types of software product and states the requirements for methods of software product measurement and evaluation.

ISO/IEC 14598 is intended for use by developers, acquirers and independent evaluators, particularly those responsible for software product evaluation. The evaluation results produced from the application of ISO/IEC 14598 can be used by managers and developers/maintainers to measure compliance to requirements and to make improvements where necessary. The evaluation results can also be used by analysts to establish the relationships between the internal and external metrics. Process improvement personnel can use the evaluation results to determine how processes can be improved through study and examination of the project's product quality information.

NOTE Much of the guidance in ISO/IEC 14598 is not specific to software, but is also applicable to other complex products.

2 Conformance

Specification and evaluation of software conforms to this part of ISO/IEC 14598 if it uses the process in clause 6 and a quality model as required in 8.3. Conformance to ISO/IEC 14598 as a whole shall mean conformance to all applicable published parts of ISO/IEC 14598.

3 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this part of ISO/IEC 14598. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this part of ISO/IEC 14598 are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies. Members of ISO and IEC maintain registers of currently valid International Standards.

ISO/IEC 2382-1:1993, *Information technology — Vocabulary — Part 1: Fundamental terms.*

ISO 8402:1994, *Quality management and quality assurance — Vocabulary.*

ISO/IEC 9126-1:—[1)], *Information technology — Software quality characteristics and metrics — Part 1: Quality characteristics and sub-characteristics.*

ISO/IEC 12207:1995, *Information technology — Software life cycle processes.*

1 To be published. Until this part is published ISO/IEC 9126:1991 should be used.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

4 Terms and definitions

For the purposes of all parts of ISO/IEC 14598, the following definitions apply.

4.1
acquirer
an organisation that acquires or procures a system, software product or software service from a supplier

[ISO/IEC 12207:1995]

4.2
attribute
a measurable physical or abstract property of an entity

NOTE Attributes can be internal or external.

4.3
developer
an organisation that performs development activities (including requirements analysis, design, testing through acceptance) during the software lifecycle process

[ISO/IEC 12207:1995]

4.4
direct measure
a measure of an attribute that does not depend upon a measure of any other attribute

4.5
evaluation module
a package of evaluation technology for a specific software quality characteristic or subcharacteristic

NOTE The package includes evaluation methods and techniques, inputs to be evaluated, data to be measured and collected, and supporting procedures and tools.

4.6
external measure
an indirect measure of a product derived from measures of the behaviour of the system of which it is a part

NOTE 1 The system includes any associated hardware, software (either custom software or off-the-shelf software) and users.

NOTE 2 The number of failures found during testing is an external measure of the number of faults in the program because the number of failures are counted during the operation of a computer system running the program.

NOTE 3 External measures can be used to evaluate quality attributes closer to the ultimate objectives of the design.

4.7
external quality
the extent to which a product satisfies stated and implied needs when used under specified conditions

4.8
failure
the termination of the ability of a product to perform a required function or its inability to perform within previously specified limits

4.9
fault
an incorrect step, process or data definition in a computer program

NOTE This definition is taken from IEEE 610.12-1990.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

4.10
implied needs
needs that may not have been stated but are actual needs when the entity is used in particular conditions

NOTE Implied needs are real needs which may not have been documented.

4.11
indicator
a measure that can be used to estimate or predict another measure

NOTE 1 The predicted measure may be of the same or a different software quality characteristic.

NOTE 2 Indicators may be used both to estimate software quality attributes and to estimate attributes of the development process. They are imprecise indirect measures of the attributes.

4.12
indirect measure
a measure of an attribute that is derived from measures of one or more other attributes

NOTE An external measure of an attribute of a computing system (such as the response time to user input) is an indirect measure of attributes of the software as the measure will be influenced by attributes of the computing environment as well as attributes of the software.

4.13
intermediate software product
a product of the software development process that is used as input to another stage of the software development process

NOTE In some cases an intermediate product may also be an end product.

4.14
internal measure
a measure of the product itself, either direct or indirect

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

4.15
internal quality
the totality of attributes of a product that determine its ability to satisfy stated and implied needs when used under specified conditions

NOTE 1 The term "internal quality", used in ISO/IEC 14598 to contrast with "external quality", has essentially the same meaning as "quality" in ISO 8402.

NOTE 2 The term "attribute" is used with the same meaning as the term "characteristic" used in 4.21, as the term "characteristic" is used in a more specific sense in ISO/IEC 9126.

4.16
maintainer
an organisation that performs maintenance activities

[ISO/IEC 12207:1995]

4.17
measure (verb)
make a measurement

4.18
measure (noun)
the number or category assigned to an attribute of an entity by making a measurement

4.19
measurement
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

NOTE Measurement can be qualitative when using categories. For example, some important attributes of software products, e.g. the language of a source program (ADA, C, COBOL, etc.) are qualitative categories.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

4.20
metric
the defined measurement method and the measurement scale

NOTE 1 Metrics can be internal or external, and direct or indirect.

NOTE 2 Metrics include methods for categorising qualitative data.

4.21
quality
the totality of characteristics of an entity that bear on its ability to satisfy stated and implied needs

NOTE 1 In a contractual environment, or in a regulated environment, such as the nuclear safety field, needs are specified, whereas in other environments, implied needs should be identified and defined (ISO 8402 :1994, note 1).

NOTE 2 In ISO/IEC 14598 the relevant entity is a software product.

[ISO 8402:1994]

4.22
quality evaluation
systematic examination of the extent to which an entity is capable of fulfilling specified requirements

NOTE The requirements may be formally specified, as when a product is developed for a specific user under a contract, or specified by the development organisation, as when a product is developed for unspecified users, such as consumer software, or the requirements may be more general, as when a user evaluates products for comparison and selection purpose.

[ISO 8402:1994]

4.23
quality in use
the extent to which a product used by specified users meets their needs to achieve specified goals with effectiveness, productivity and satisfaction in specified contexts of use

NOTE This definition of quality in use is similar to the definition of usability in ISO 9241-11. In ISO/IEC 14598 the term usability is used to refer to the software quality characteristic described in ISO/IEC 9126-1.

4.24
quality model
the set of characteristics and the relationships between them which provide the basis for specifying quality requirements and evaluating quality

4.25
rating
the action of mapping the measured value to the appropriate rating level. Used to determine the rating level associated with the software for a specific quality characteristic

4.26
rating level
a scale point on an ordinal scale which is used to categorise a measurement scale

NOTE 1 The rating level enables software to be classified (rated) in accordance with the stated or implied needs (see 10.2).

NOTE 2 Appropriate rating levels may be associated with the different views of quality i.e. ‘Users’, ‘Managers’ or ‘Developers’.

4.27
scale
a set of values with defined properties

NOTE Examples of types of scales are: a nominal scale which corresponds to a set of categories; an ordinal scale which corresponds to an ordered set of scale points; an interval scale which corresponds to an ordered scale with equidistant scale points; and a ratio scale which not only has equidistant scale point but also possess an absolute zero. Metrics using nominal or ordinal scales produce qualitative data, and metrics using interval and ratio scales produce quantitative data.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

4.28
software
all or part of the programs, procedures, rules, and associated documentation of an information processing system

NOTE Software is an intellectual creation that is independent of the medium on which it is recorded.

[ISO/IEC 2382.1:1993]

4.29
software product
the set of computer programs, procedures, and possibly associated documentation and data

NOTE Products include intermediate products, and products intended for users such as developers and maintainers.

[ISO/IEC 12207:1995]

4.30
supplier
an organisation that enters into a contract with the acquirer for the supply of a system, software product or software service under the terms of the contract

[ISO/IEC 12207:1995]

4.31
system
an integrated composite that consists of one or more of the processes, hardware, software, facilitires and people, that provides a capability to satisfy a stated need or objective

[ISO/IEC 12207:1995]

4.32
user
an individual that uses the software product to perform a specific function

NOTE Users may include operators, recipients of the results of the software, or developers or maintainers of software.

4.33
validation
confirmation by examination and provision of objective evidence that the particular requirements for a specific intended use are fulfilled

NOTE 1 In design and development, validation concerns the process of examining a product to determine conformity with user needs.

NOTE 2 Validation is normally performed on the final product under defined operating conditions. It may be necessary in earlier stages.

NOTE 3 "Validated" is used to designate the corresponding status.

NOTE 4 Multiple validations may be carried out if there are different intended uses.

[ISO 8402:1994]

4.34
verification
confirmation by examination and provision of objective evidence that specified requirements have been fulfilled

NOTE 1 In design and development, verification concerns the process of examining the result of a given activity to determine conformity with the stated requirement for that activity.

NOTE 2 "Verified" is used to designate the corresponding status.

[ISO 8402:1994]

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

5 Overview of ISO/IEC 14598 and ISO/IEC 9126

5.1 Structure of ISO/IEC 14598 and ISO/IEC 9126

The ISO/IEC 9126 series defines a general purpose quality model, quality characteristics and gives examples of metrics. The ISO/IEC 14598 series gives an overview of software product evaluation processes and provides guidance and requirements for evaluation. ISO/IEC 14598-2 and ISO/IEC 14598-6 relate to corporate or department level evaluation management and support, while ISO/IEC 14598-3, ISO/IEC 14598-4 and ISO/IEC 14598-5 give requirements and guidance for evaluation at the project level. Figures 1 and 2 show the relationships between these standards and technical reports.

5.2 Evaluation process

The ISO/IEC 14598 series of International Standards provides guidance and requirements for the evaluation process in three different situations:

- development (enhancement) (ISO/IEC 14598-3);
- acquisition (ISO/IEC 14598-4);
- independent evaluation (including third-party evaluation) (ISO/IEC 14598-5).

5.2.1 Process for developers

ISO/IEC 14598-3 should be used by organisations that are planning to develop a new product or enhance an existing product and intending to perform product evaluation using members of its own technical staff. It focuses on the use of those indicators that can predict end product quality by measuring intermediate products developed during the life-cycle.

5.2.2 Process for acquirers

ISO/IEC 14598-4 should be used by organisations that are planning to acquire or reuse an existing or pre-developed software product. It can be applied for the purposes of deciding on the acceptance of the product or for selecting a product from among alternative products. (A product may be self contained, a part of system, or it may be part of larger product.)

5.2.3 Process for evaluators

ISO/IEC 14598-5 should be used by evaluators carrying out an independent assessment of a software product. This evaluation could be performed at the request of either a developer, acquirer or some other party. This part is intended for those who perform independent evaluation. Often they work for third party organisations.

5.3 Support for evaluation

Each of the evaluation process standards can be used in conjunction with ISO/IEC 14598-2 (Planning and Management) and ISO/IEC 14598-6 (Documentation of evaluation modules) (Figure 1).

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

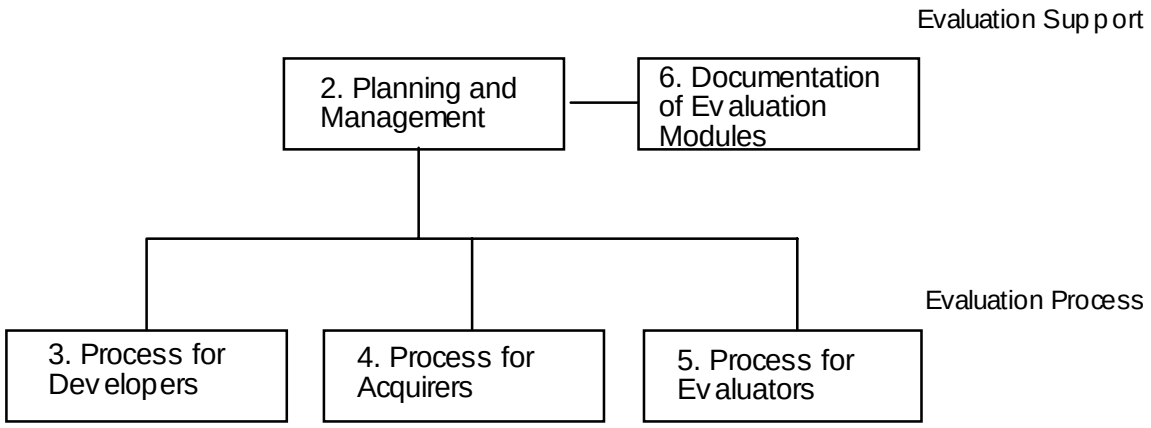

Figure 1 - Relationship of evaluation process to evaluation support standards

5.3.1 Planning and management

ISO/IEC 14598-2 Planning and Management contains requirements and guidance for supporting functions for software product evaluation. The support is related to planning and management of a software evaluation process and associated activities, including development, acquisition, standardisation, control, transfer and feedback of evaluation expertise within the organisation. This part of ISO/IEC 14598 can be used by managers to produce a quantitative evaluation plan.

5.3.2 Evaluation modules

ISO/IEC 14598-6 provides guidance for documenting evaluation modules. These modules contain the specification of the quality model (i.e. characteristics, subcharacteristics and corresponding internal or external metrics), the associated data and information about the planned application of the model and the information about its actual application. For each evaluation appropriate evaluation modules are selected. In some cases it may be necessary to develop new evaluation modules. This part of ISO/IEC 14598 can be used by organisations producing new evaluation modules.

5.4 Software quality characteristics and metrics

Each of the parts of ISO/IEC 14598 should be used in conjunction with the planned parts of ISO/IEC 9126 describing software quality characteristics and metrics:

- Quality Characteristics and Subcharacteristics (ISO/IEC 9126-1);
- External Metrics (ISO/IEC 9126-2);
- Internal Metrics (ISO/IEC 9126-3).

ISO/IEC 9126-1 defines quality characteristics, associated subcharacteristics and the relations between the top three levels of the ISO/IEC 9126 quality model. ISO/IEC 9126-2 and ISO/IEC 9126-3 identify the relationships of each metric (internal and external) to their corresponding characteristics and subcharacteristics (see Figure 2). Note that some internal metrics have corresponding external metrics.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

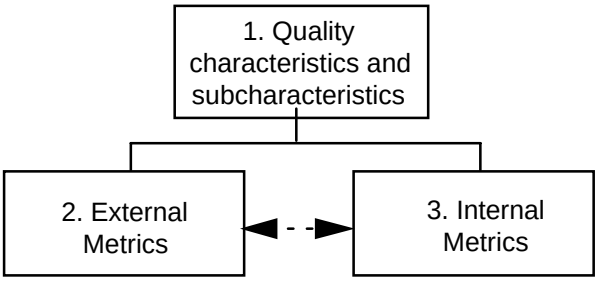

Figure 2 - ISO/IEC 9126 Software Quality Characteristics and Metrics

5.4.1 Quality Characteristics and Subcharacteristics

ISO/IEC 9126-1 defines a set of quality characteristics and corresponding subcharacteristics. These subcharacteristics are manifested externally when the software is used as a part of a computer system, and are a result of internal software attributes. ISO/IEC 9126-1 is used as the foundation for constructing the top three levels of the quality model. The overall objective of quality from the users' perspective is quality in use.

5.4.2 External metrics

ISO/IEC 9126-2 (in preparation) describes those metrics that represent the external perspective of software quality when the software is in use. The external measures are taken over some predefined period while the software is in use. Values for quantities like time and effort are used as the basis for these measures. These measures apply in both the testing and operation phases. When used during test they are meant to be early predictors of the levels of quality that can be expected once the software is used and operated. These measures generally represent the quality in terms that are relevant to users.

5.4.3 Internal metrics

ISO/IEC 9126-3 (in preparation) describes those metrics that measure internal attributes of the software related to design and code. These “early” measures are used as indicators to predict what can be expected once the software system is in test and operation. Therefore the internal measures are most important to development managers since they are a valuable device for forestalling down stream problems. Internal measures are used to predict the values of corresponding external measures. ISO/IEC 9126-3 shows which internal metrics have corresponding external metrics.

6 Evaluation process

To evaluate software quality, first establish the evaluation requirements, then specify, design and execute the evaluation (see Figure 3). Each step is described in more detail in the clauses indicated. This part of ISO/IEC 14598 gives an overview of the process. The other parts of ISO/IEC 14598 explain how the process is applied in different circumstances.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

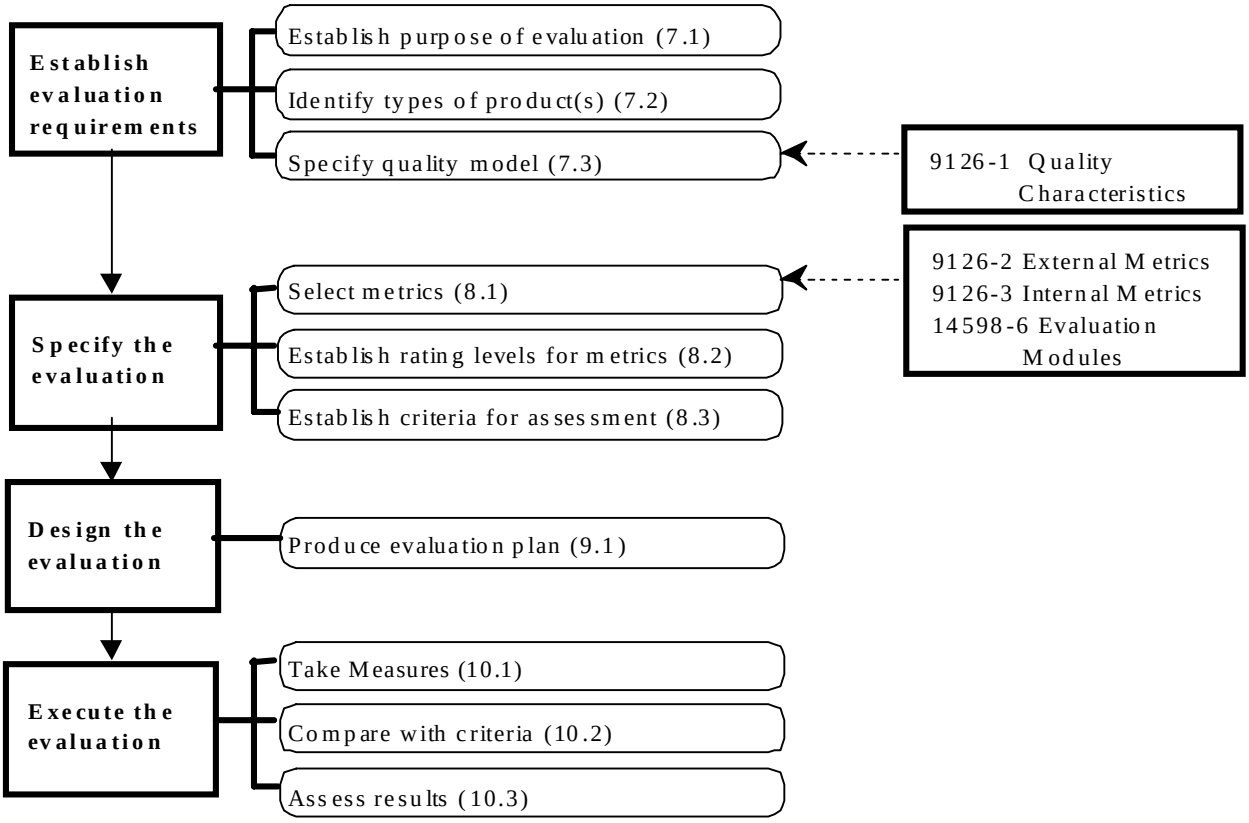

Figure 3 - Evaluation process

7 Establish evaluation requirements

7.1 Establish the purpose of evaluation

7.1.1 General

The purpose of software quality evaluation is to directly support both the development and acquisition of software which meets user and customer needs. The ultimate objective is to ensure that the product provides the required quality - that it meets the stated and implied needs of the users (including operators, recipients of the results of the software, or maintainers of software).

The purpose of evaluation of intermediate product quality may be to:

- decide on the acceptance of an intermediate product from a subcontractor;
- decide on the completion of a process and when to send products to the next process;
- predict or estimate end product quality;
- collect information on intermediate products in order to control and manage the process.

The purpose of evaluation of end product quality may be to:

- decide on the acceptance of the product;
- decide when to release the product;

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

- compare the product with competitive products;
- select a product from among alternative products;
- assess both positive and negative effect of a product when it is used;
- decide when to enhance or replace the product.

Software quality can be evaluated within a defined quality structure throughout the product development and acquisition life-cycle processes defined in ISO/IEC 12207.

7.1.2 Acquisition

When acquiring a custom made software product, the acquirer should establish external quality requirements, specify the requirements to the supplier, and evaluate potential purchases against these requirements before acquisition.

When a product is being developed, the objective of specifying quality requirements is to ensure that the product meets the stated and implied needs of the user (see ISO/IEC 14598-3).

When purchasing a software product, evaluation can be used to compare alternative products and to ensure that the selected product meets the quality requirements (see ISO/IEC 14598-4 for the process for acquirers and ISO/IEC 14598-5 for the process for evaluators, including third party evaluation requirements).

7.1.3 Supply

The supplier can use the results of software product evaluation to ensure that products meet required quality criteria which may be set by the acquirer, or by comparison with other products.

7.1.4 Development

Software product requirements express the users' needs for the software product under consideration, and are defined prior to the development. As a software product is decomposed into major components, the requirements derived from the overall product may differ for the different components, and may require different evaluation criteria. Prior to quality evaluation, quality requirements should be specified in terms of quality characteristics and subcharacteristics (see ISO/IEC 9126-1).

At the initial stage of evaluation, these quality requirements should be studied and identified, for planning and implementing evaluation. The developer should establish external quality requirements for each relevant quality characteristic. The completeness and correctness of the quality requirements specification should be evaluated to ensure that all the necessary requirements have been specified and unnecessary requirements excluded. The developer needs to evaluate the product against these requirements before delivery.

To achieve quality both stated and implied needs have to be met, so it is important to check that implied needs are specified in sufficient detail for all relevant quality characteristics. If possible, requirements should be assessed by procurers or purchasers, and by end users to assess the implied needs. User experience with prototypes frequently leads to a more accurate statement of requirements for quality in use.

The developer should identify internal quality requirements. When internal quality requirements are used, the developer should identify these using a quality model which relates them to the external quality requirements, and use the internal requirements to verify quality of intermediate products during development.

Software evaluation should be used to predict and verify quality during development, by specifying internal quality requirements for intermediate products in the development process. The external quality of the complete product for specific intended uses can subsequently be evaluated against initial requirements. (See ISO/IEC 14598-3 for the evaluation process for developers.)

--`,,```,,,,````-`-`,,`,,`,`,,`---
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

The results of software quality evaluation can be used to obtain feedback on the extent to which different development processes, design methods or CASE tools can be used to meet quality requirements.

7.1.5 Operation

The organisation which operates a software system can use software quality evaluation to validate that quality requirements are met under different operating conditions, and to provide feedback on the need for any changes to those responsible for maintenance.

7.1.6 Maintenance

The organisation which maintains the software system can use software evaluation to validate that quality requirements are still met, and requirements for maintainability and portability are achieved.

7.2 Identify types of product(s) to be evaluated

The type of intermediate or final software product to be evaluated will depend on the stage in the life cycle and the purpose of the evaluation (see Figure 4).

The objective is that when the software product is actually used by the user it meets the stated and implied needs. External quality can only be assessed for a complete hardware/software system of which the software product is a part. External metrics are applied when executing the software. The values of external measures necessarily depend on more than the software, so the software has to be evaluated as part of a working system.

Quality in use is the combined effect of the relevant quality characteristics for a particular user (who may be an end user, operator or maintainer). For software to have quality in use it has to meet the user needs to carry out particular tasks in particular hardware and software environments. Software which performs satisfactorily in one environment may show faults in another environment. External evaluation of quality characteristics should therefore take place under conditions which emulate as closely as possible the expected conditions of use. External measurements of characteristics are made when the code is complete, though as it may not be possible to emulate the exact conditions of use (e.g. network environment and user characteristics), external measures are often only indicators of the actual quality in use.

If the external quality requirements are not achieved the results of the evaluation can be used as feedback to modify the software characteristics in order to improve the external quality, thus supporting a iterative improvement process.

--`,,```,,,,````-`-`,,`,,`,`,,`---
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

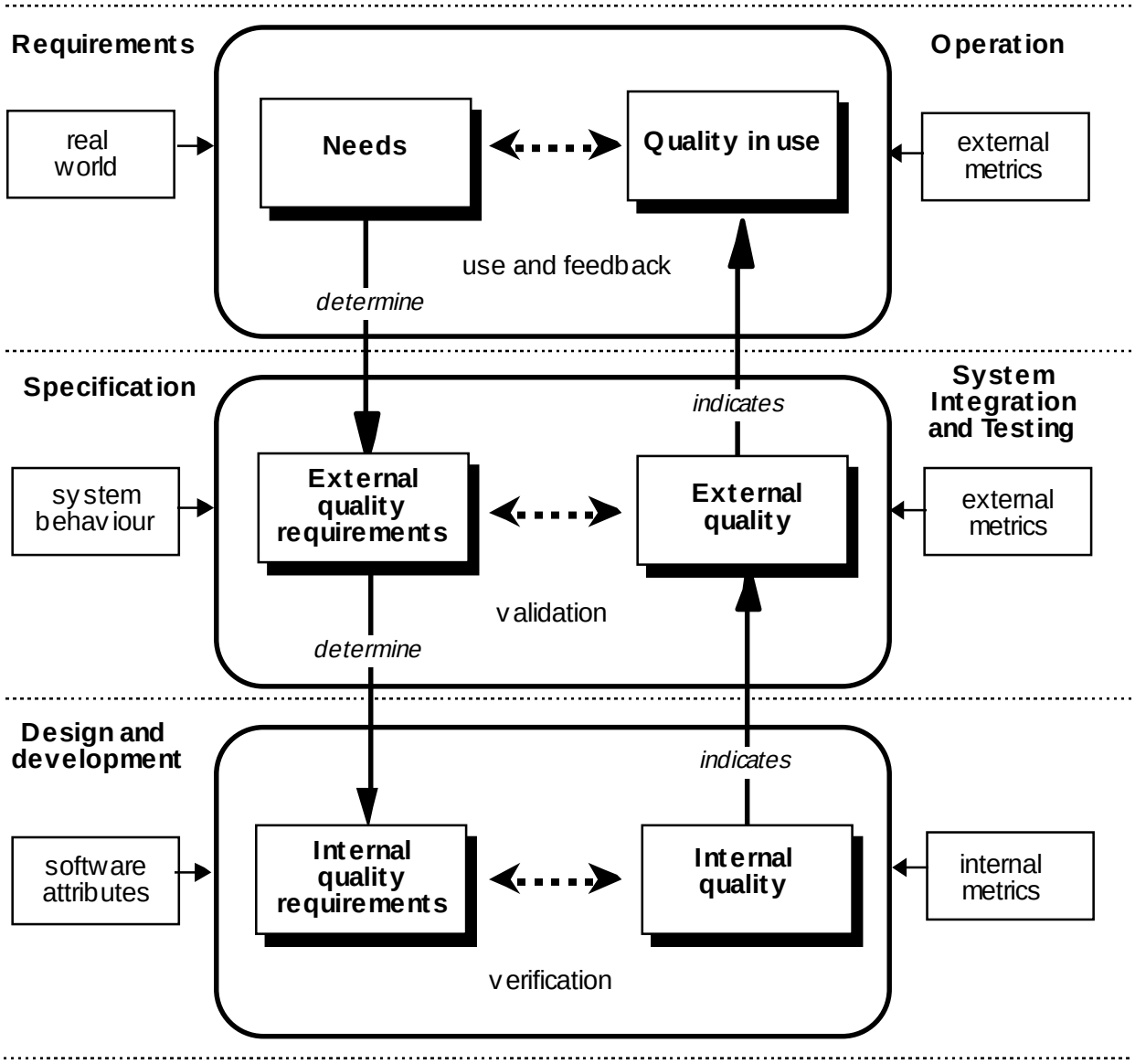

Figure 4 - Quality in the software lifecycle

For the purposes of development, internal quality requirements are defined which enable the quality of intermediate products to be verified. The internal properties (e.g. the specification or source code) of the software can be measured by internal metrics. Internal metrics are of most interest during the development process. Internal measures can be used as indicators of external attributes. Modularity and traceability are examples of internal attributes which can be measured. Achievement of the required internal quality will contribute to meeting the external requirements of the software in use. Internal software quality measures can thus be used as indicators to estimate quality in use (see Figure 5).

For example, response time is an important measure required to evaluate the usability and the efficiency of the software, but response time cannot be measured during development. In order to evaluate the efficiency of the product during development, path length could be measured based on the intermediate product or specifications. This could be used as an indicator which provided rough estimates of response time under certain conditions.

It is very important that internal software quality attributes are related to external quality requirements, so that the quality characteristics of software products under development (both intermediate and end item software products) can be assessed with respect to final system in-use quality needs. Internal measures are of little value unless there is evidence that they are related to external quality.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

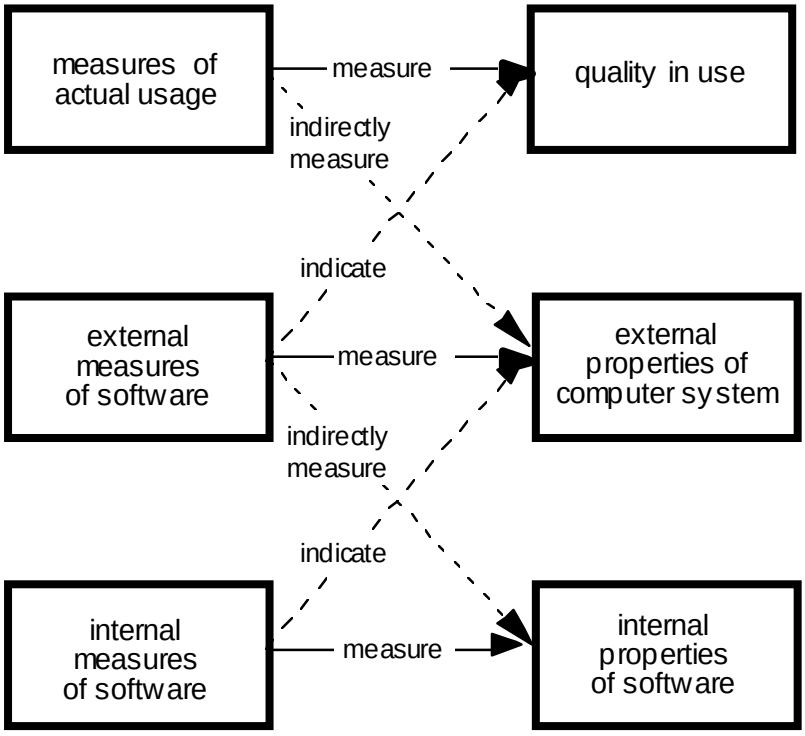

Figure 5 - Relationships between measures

The specific attributes which are relevant to final quality will depend on the intended conditions of use - for an interactive product this will depend on the needs of the eventual end user and task. Other issues which will influence the needs for software product quality include whether the product is being purchased or developed, the stage of development, and the hardware, software and network environment in which the product will be used.

External measures of a computer system can also be used as indirect measures of internal software quality. Thus the response time of a computer system can be used to measure the efficiency of the software in a particular computing environment.

7.3 Specify quality model

The first step in the evaluation of software is to select relevant quality characteristics, using a quality model which breaks software quality down into different characteristics. Quality models for software evaluation generally represent the totality of software quality attributes classified in a hierarchical tree structure of characteristics and subcharacteristics. The highest level of this structure consists of quality characteristics and the lowest level consists of software quality attributes. ISO/IEC 9126-1 provides a general-purpose model which defines six broad categories of software quality characteristics: functionality, reliability, usability, efficiency, maintainability and portability. These can be further broken down into subcharacteristics which have measurable attributes. The combined effect of the quality characteristics in a particular usage situation is defined as quality in use.

Internal software product quality attributes are the measurable properties of a software product that influence its ability to satisfy stated and implied needs. One or more attributes can be used to assess a particular software quality characteristic or subcharacteristic (Figure 6).

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

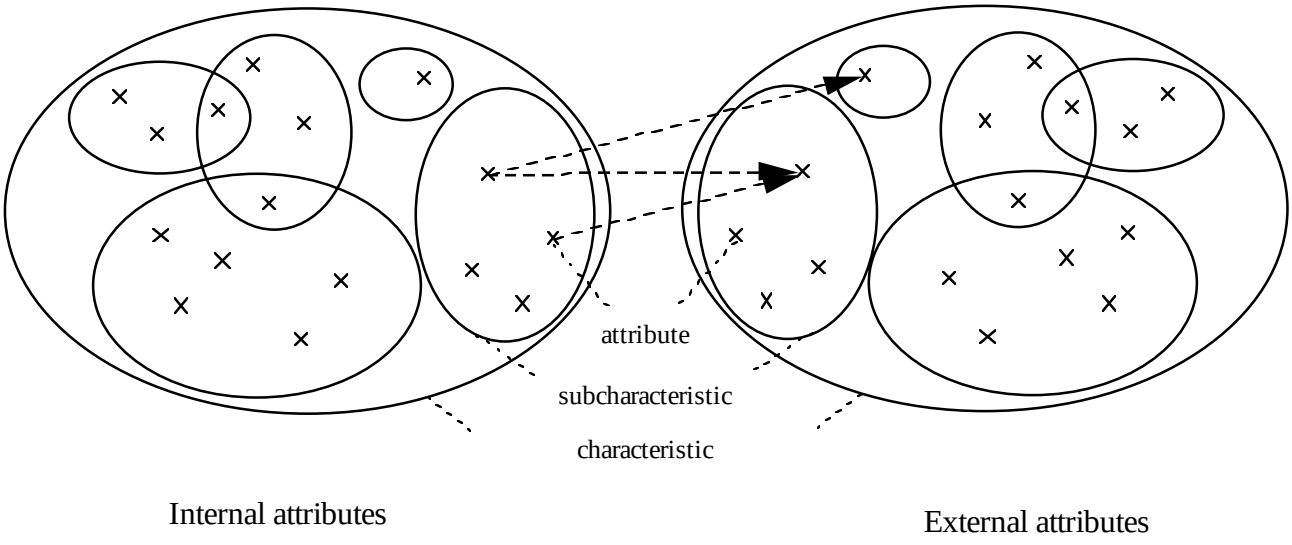

Figure 6 - Quality characteristics, subcharacteristics and attributes

Sufficient internal and external attributes should be identified for each required subcharacteristic.

The actual characteristics and subcharacteristics which are relevant in any particular situation will depend on the purpose of the evaluation, and should be identified by a quality requirements study. The ISO/IEC 9126-1 characteristics and subcharacteristics provide a useful checklist of issues related to quality, but other ways of categorising quality may be more appropriate in particular circumstances.

NOTE An example is IEC 50(191) which defines Dependability as the extent to which a user can justifiably depend on the service delivered by a system. This is divided into characteristics of reliability, availability and maintainability. It could also include usability, recoverability, safety, extendibility and security.

8 Specify the evaluation

8.1 Select metrics

It is important that software product measurements can be made easily and economically and that the resulting measures are easy to use. Many software measurements may conveniently be made with a tool of some kind and may be packaged as an evaluation module (ISO/IEC 14598-6).

The manner in which quality characteristics have been defined does not allow their direct measurement. The need exists to establish metrics that correlate to the characteristics of the software product. Every quantifiable internal attribute of software and every quantifiable external attribute of the software interacting with its environment that correlates with a characteristic can be established as a metric.

Metrics can differ depending on the environment and the phases of the development process in which they are used. Metrics used in the development process should be correlated to the user perspective metrics, because the metrics from the user's view are crucial.

NOTE Examples of metrics and an explanation of how to select and summarise metrics for each subcharacteristic will be given in ISO/IEC 9126 parts 2 and 3 (under development).

8.1.1 Types of measurements

There are two broad objectives of evaluation:

- to identify problems so that they can be rectified, and

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

- to compare the quality of a product with alternative products or against requirements (which may include certification).

The type of measurement required will depend on the purpose of evaluation. If the primary purpose is to understand and correct deficiencies, several measurements may be made on the software to monitor and control improvements. A wide range of measures can be useful for these purposes, including checklists and expert opinion. The primary requirement is that the measurements correctly identify the impact that any changes in the software have on quality.

More rigorous metrics are required to make reliable comparisons, either between products or with criterion values. Measurement procedures should measure the software quality characteristic (or subcharacteristic) they claim to be measuring with sufficient accuracy to allow criteria to be set and comparisons to be made. It is important that the evaluation specification specifies a precise quality model, and measurement methods, scales and rating levels for each metric. Data from checklists and expert opinion may not be reliable when comparing products with different attributes. Allowance should be made for possible measurement errors caused by measurement tools or human error.

8.1.2 Requirements for measurements

Internal metrics should have predictive validity, that is they should correlate with some desired external criterion. For instance an internal measure of a particular software attribute should correlate with some measurable aspect of quality when the software is used. It is important that measurements assign values which coincide with normal expectations; for instance if the measurement suggests that the product is of high quality then this should be consistent with the product satisfying particular user needs.

Further information on the requirements for measurement can be found in ISO/IEC 9126-1.

8.2 Establish rating levels for metrics

Quantifiable features can be measured quantitatively using quality metrics. The result, i.e. the measured value, is mapped on the scale. This value does not itself show the level of satisfaction. For this purpose, the scale has to be divided into ranges corresponding to the different degrees of satisfaction of the requirements. Examples are:

- dividing the scale into two categories: unsatisfactory and satisfactory;

- dividing the scale into four categories bounded by the current level for an existing or an alternative product, the worst case, and the planned level. The current level is stated in order to control that the new system does not deteriorate from the present situation. The planned level is what is considered achievable with the resources available. The worst case level is a boundary for user acceptance, in case the product does not fulfil the planned level (Figure 7).

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

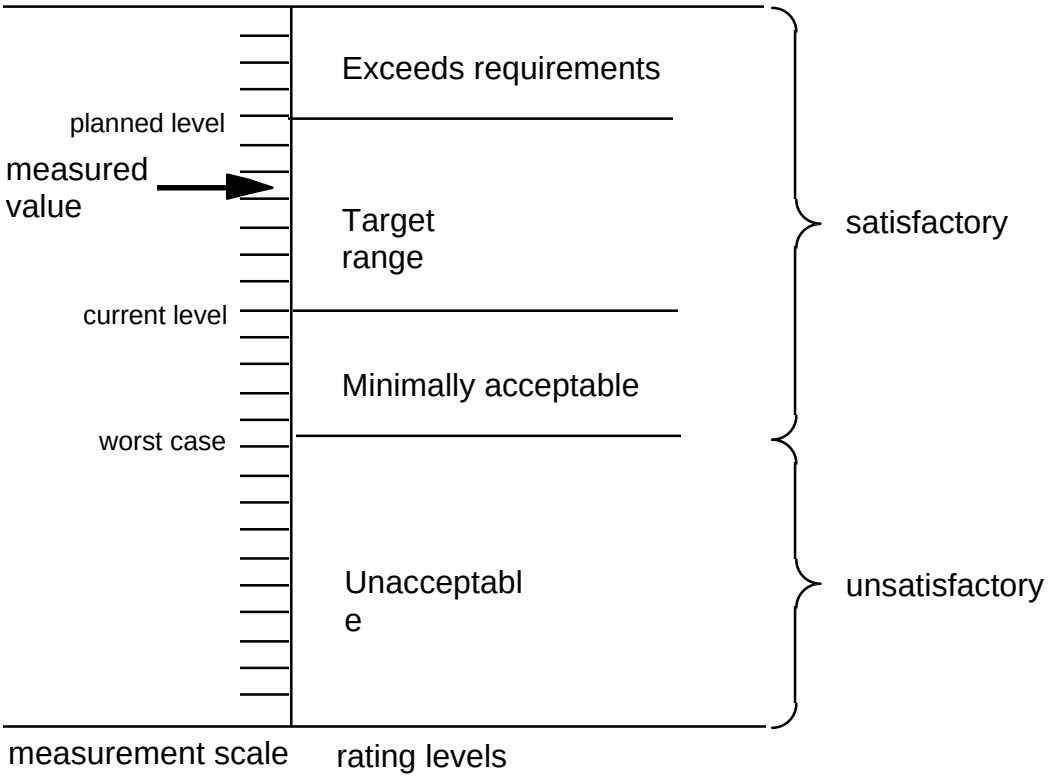

Figure 7 - Rating levels for metrics

8.3 Establish criteria for assessment

Software quality requirements specifications shall be defined using an appropriate well defined quality model. For this purpose the quality model and definitions in ISO/IEC 9126-1 should be used, unless there is a particular reason to use another model.

To assess the quality of the product, the results of the evaluation of the different characteristics need to be summarised. The evaluator should prepare a procedure for this, with separate criteria for different quality characteristics, each of which may be in terms of individual subcharacteristics, or a weighted combination of subcharacteristics. The procedure usually will include other aspects such as time and cost that contribute to the assessment of quality of a software product in a particular environment.

9 Design the evaluation

9.1 Produce evaluation plan

The Evaluation Plan describes the evaluation methods and the schedule of the evaluator actions (see ISO/IEC 14598-3, ISO/IEC 14598-4 or ISO/IEC 14598-5). It should be consistent with the Measurements Plan (ISO/IEC 14598-2).

10 Execute the evaluation

10.1 Take measures

For measurement, the selected metrics are applied to the software product. The result are values on the scales of the metrics.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

10.2 Compare with criteria

In the rating step, the measured value is compared with predetermined criteria (e.g. as shown in Figure 7).

10.3 Assess results

Assessment is the final step of the software evaluation process, where a set of rated levels are summarised. The result is a statement of the extent to which the software product meets quality requirements. Then the summarised quality is compared with the other aspects such as time and cost. Finally a managerial decision will be made based on the managerial criteria. The result is a managerial decision on the acceptance or rejection, or on the release or no-release of the software product.

The results of the evaluation are important for decisions about next steps in the software development life cycle. For instance, do the requirements have to be changed or are more resources needed for the development process?

11 Supporting processes

These are activities for assisting evaluation by collecting information on evaluation tools and methods, developing and validating metrics, and standardising evaluation process, metrics and measures. ISO/IEC 14598-2 contains requirements and guidance for supporting processes for software product evaluation.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Bibliography

International Standards

[1] IEC 60050-191:1990, *International Electrotechnical Vocabulary — Dependability and quality of service.*

[2] IEEE 610.12-1990, *Standard Glossary of Software Engineering Terminology.*

[3] ISO/IEC 9126-2:—[1)], *Information technology — Software quality characteristics and metrics — Part 2: External metrics.*

[4] ISO/IEC 9126-3:—[1)], *Information technology — Software quality characteristics and metrics — Part 3: Internal metrics.*

[5] ISO 9241-11:1998, *Ergonomic requirements for office work with visual display terminals (VDTS) — Part 11: Guidance on usability.*

Other references

[6] Arthur, J.D. and Nance, R.E., Developing an automated procedure for evaluating software development methodologies and associated products- A final report, Technical report SRC-87-007, Systems Research Center and Virginia Tech, 1987.

[7] Azuma, M (Editor), Software Quality Evaluation Guide Book (Japanese), JISA, 1994.

[8] Azuma, M. Software Products Evaluation System: Quality Models, Metrics and Processes - International Standards and Japanese Practice, Information and Software Technology, Vol. No. , 1996, Elsevier.

[9] Basili, V.R. and Rombach, H.D., The TAME Project: Towards improvement oriented software environments, IEEE Trans. Software engineering, June 1988.

[10] Bevan N Measuring usability as quality of use. Journal of Software Quality, 4, p.115-130, 1995.

[11] Bevan N and Azuma M (1997) Quality in use: Incorporating human factors into the software engineering lifecycle. In: Proceedings of the Third IEEE International Software Engineering Standards Symposium and Forum (ISESS'97), p.169-179.

[12] Boehm, B.W., et al., Quantitative evaluation of software quality, Proc. ICSE 1976.

[13] Bush, M and Russel, M., Software Engineering Measurement: A Modular Course, Proc. COMPSAC'91 P571 -576.

[14] Dumke, R. Softwareentwicklung nach Masz (German), Vieweg, 1992.

[15] Fenton NE and Pfleeger SL, Software Metrics: A Rigorous and Practical Approach, International Thomson Computer Press, 1996.

[16] McCabe, T.J. A Software Complexity Measure, IEEE Transaction on Software Engineering, Vol. 2, No. 6, 1976.

[17] McCall et al., Rome Air Develop Center Report, TR-77-369, 1977.

[18] Moller, K.H. and Paulish, D.J. Software Metrics, Chapman & Hall (IEEE Press), 1993.

1 Currently at the stage of draft.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

[19] Miyoshi, Togashi and Azuma, Evaluating software development environment quality, Proc. COMPSAC'89, 1989.

[20] Natale, D. Qualita E Quantita Nei Sistemi Software (Italian), FrancoAngeli, 1995.

[21] Sunazuka, T., and Azuma, M., Software quality assessment technology, Proc. ICSE 1985.

[22] Wallmuller, E. Software Quality Assurance A practical approach, Prentice Hall, 1994.

[23] Zuse, H., Software Complexity - Measures and Methods, Walter de Gruyter, 1991.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

© ISO/IEC

ICS 35.080

Price based on 19 pages

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS

Not for Resale