ISO/IEC JTC1/SC7
Software Engineering
Secretariat: CANADA (SCC)

ISO/IEC JTC1/SC7 N2083

1999/03/30

Document Type	PDTR Ballot
Title	PDTR Ballot of PDTR 9126-3: Software Engineering - Product Quality Part 3 - Internal Metrics.
Source	JTC1/SC7 Secretariat
Project	07.13.01.03
Status	PDTR Ballot
References	N2082
Action ID	FYI or ACT
Due Date	1999/06/30
Mailing Date	1999/03/30
Distribution	SC7_AG
Medium	Encoded Acrobat
No. of Pages	31
Note	Please send early comments directly to Prof. Azuma so they can be discussed at the upcoming Plenary meeting.

Address reply to: ISO/IEC JTC1/SC7 Secretariat
Bell Canada - Quality Engineering & Research
1050 Beaver Hall Hill, 2nd Floor, Montréal (Québec) Canada H2Z 1S4
Tel.: +1 (514) 391-8286 Fax: +1 (514) 870-2246
sc7@qc.bell.ca

ISO/IEC JTC1/SC7 PDTR 9126-3	
Date 1999/03/30	Reference number ISO/JTC 1/SC 7 N2083
Supersedes document	

THIS DOCUMENT IS STILL UNDER STUDY AND SUBJECT TO CHANGE. IT SHOULD NOT BE USED FOR REFERENCE PURPOSES.

ISO/JTC 1/SC 7 Committee Title **Software Engineering** Secretariat: Standards Council of Canada (SCC)	Circulated to P- and O-members, and to technical committees and organizations in liaison for: X voting by (P-members only) **1999/06/30** Please return all votes and comments in electronic form directly to the SC 7 Secretariat by the due date indicated.

ISO/IEC JTC1/SC7

Title: PDTR Ballot of PDTR 9126-3: Software Engineering - Product Quality Part 3 - Internal Metrics.

Project: 07.13.01.03

Introductory note: See page ii of the document

Medium: Encoded Acrobat

No. of pages: 31

Vote on PDTR 9126-3	
Date of circulation **1999/03/30** Closing date 1999/06/30	Reference number ISO/JTC 1/SC 7 N2083

ISO/JTC 1/SC 7 Committee Title **Software Engineering** Secretariat: Standards Council of Canada (SCC)	Circulated to P-members of the committee for voting Please return all votes and comments in electronic form directly to the SC 7 Secretariat by the due date indicated.

ISO/IEC JTC1/SC7

Title: PDTR Ballot of PDTR 9126-3: Software Engineering - Product Quality Part 3 - Internal Metrics.

Project: 07.13.01.03

Vote:

__ APPROVAL OF THE DRAFT AS PRESENTED

__ APPROVAL OF THE DRAFT WITH COMMENTS AS GIVEN ON THE ATTACHED

__ general:

__ technical:

__ editorial:

__ DISAPPROVAL OF THE DRAFT FOR REASONS ON THE ATTACHED

__ Acceptance of these reasons and appropriate changes in the text will change our vote to approval

__ ABSTENTION (FOR REASONS BELOW):

P-member voting:
National Body (Acronym)

Date:
YYYY-MM-DD

Submitted by:
Your Name

ISO/IEC JTC1/SC7/WG6

EVALUATION AND METRICS

TITLE	**ISO/IEC 9126-3 : Software Engineering – Product quality - Part 3 : Internal Metrics**
DATE:	**Jan. 18, 1999**
SOURCE:	**ISO/IEC JTC1/SC7/WG6**
WORK ITEM:	**Project 7-13-01-03**
STATUS:	**Version 1.0 – reflects ISO/IEC JTC1/SC7/WG6 Sydney, Australia meeting(Nov. , 1998) and revised for 1st PDTR Ballot**
REFERENCE:	**6/N 427c, WG6/Rome-13R, SC7/N1713, 6/N 375, 6/N 345**
DOCUMENT TYPE:	**Proposed Draft Technical Report**
ACTION:	**For 1st PDTR Ballot.**

EDITOR: **Yukio TANITSU**
IBM Japan
1623-14, Shimo-Tsuruma, Yamato,
Kanagawa 242, Japan
TEL: +81-462-73-6190 , FAX: +81-462-73-7415
e-mail : tanitsu@jp.ibm.com

CO-EDITOR: **Vipula GODAMUNNE**
IBM Global Services Australia
TEL: +61-2-9353-3727 , FAX: -
E-mail: vipula@au1.ibm.com

REVIEWERS

ISO/IEC JTC1/SC7
Software Engineering
Secretariat: CANADA (SCC)

ISO/IEC JTC 1/SC 7 N

Date: 1999-01-15

WD

ISO/IEC JTC 1/SC 7/WG 6

Secretariat: ISO/IEC JTC 1/SC 7

9126-3: Information Technology - Software Product Quality - Part 3: Internal Metrics.

—

Docum
Docum
Docum
Document language: E

Address reply to: ISO/IEC JTC1/SC7 Secretariat
Bell Canada - IT Procurement & Supplier Quality
2265 Roland Therrien, Room 226, Longueuil (Québec) Canada J4N 1C5
Tel: +1 (514) 448-3100 Fax: +1 (514) 448-2090 or +1 (514) 647-3163

Contents

Foreword

ISO (the International Organization for Standardization) and IEC (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.

Proposed Draft Technical Report ISO/IEC 9126-3 was prepared by the Joint Technical Committee ISO/IEC JTC1, Information Technology, Subcommittee SC7, Software Engineering.

Introduction

This draft technical report provides internal metircs for measuring attributes of six quality characteristics defined in ISO/IEC 9126-1. This report also provides data elements which are commonly used for composing metrics.

The software quality model, set out in the ISO/IEC Standard 9126-1 Information Technology - Software product quality - Part 1: Quality model, presents the framework for the specification and evaluation of software quality. The ISO/IEC technical reports 9126-2, 9126-3(this technical report) and 9126-4 provide the metrics to determine the softwre product quality levels.

This technical report defines metrics from a internal view in developers, aquirers, evaluators, and maintainers. The metics and data elements listed in this technical report are considered to be a basic set.

Develoers, evaluators, quality managers and acquirers may select some metrics and data elements from this technical report for defining requirements, evaluating software products, measuring quality aspect and other purposes. This document provides a set of intenal metrics to measure subcharacteristics. At the same time this document describes a set of internal subcharacteristics of software products that influence the subcharacteristics defined in ISO/IEC 9126-1. Regarding general intent of all the quality metrics including external metrics and quality in use metrics in addition to intenral metircs, developers, evaluators, quality managers and acqiures can refer to Clause 6 in 9126-2, which is designed to commonly used among internal metrics, external metrics and quality in use metrics.

The technical report does not assign ranges of values of these metrics to rated levels or to grades of compliance, because these values are defined for each software product or a part of the software product , by its nature, depending on such factors as category of the software, integrity level and users' needs.

PDTR

Information Technology — Software quality characteristics and metrics — Part 3: Internal Metrics

1 Scope

This international technical report defines sets of internal metrics and provide suggested sets of internal metrics used for predicting quality characteristics of developed software products or software products under development from a developer/maintainer perspective. This report also defines a set of internal metrics that measure the internal quality characteristics. The internal metrics are intended to measure software products from an internal perspective.

The software quality characteristics and subcharacteristics are defined in ISO/IEC 9126-1, related external metrics are defined in ISO/IEC 9126-2, and related quality in use metrics are defined in 9126-4.

This report should be used to guide the user of metrics for planning evaluation, selecting metrics, designing metrics, applying metrics, and interpreting metric results.

Audiences of this international technical report include:

1. Acquirer (an organisation that acquires or procures a system, software product or software services from a supplier)

2. Developer (an organisation that performs development activities, including requirements analysis, design, testing through acceptance during the software life cycle process)

3. Evaluator (an organisation that performs an evaluation. An evaluator may, for example, be a testing laboratory, the quality department of a software development organisation, a government organisation or a user)

4. Supplier (an organisation that enter into a contract with the acquirer for the supply of a system, software product or software services under the terms of the contract) when validating software quality at qualification test

5. Maintainer (an organisation that performs maintenance activities)

6. User (an individual that uses the software product to perform a specific function) when evaluating quality of software product at acceptance test

7. Quality manager (an organisation that perform an systematic examination of the software product or software services) when evaluating software quality at qualification test

1.1 Out of Scope

This Technical Report does not attempt to address the following items,

- counting rules

- expected range of values to achieve rated levels of compliance

- identification of metrics which are usable as indicators for other metircs

- recommended prediction rules when using the metrics

- provision of a complete or a recommended set of metrics applicable to any subcharacteristics.

2 Normative reference(s)

ISO 8402 : 1994, Quality management and quality assurance – Quality Vocaburary

ISO/IEC 9126 : 1988, Information technology - Software product evaluation – Quality characteristics and guidelines for their use

ISO/IEC 9126-1(new) : Information Technology - Software product quality - Part 1: Quality model

ISO/IEC 9126-2(new) : Information Technology - Software product quality - Part 2: External metrics

ISO/IEC 9126-4(new) : Information Technology - Software product quality - Part 4: Quality in use metrics

ISO/IEC 14598-1(new) : Information Technology - Software product evaluation - Part 1: General overview

ISO/IEC 14598-2(new) : Information Technology - Software product evaluation - Part 2: Planning and management

ISO/IEC 14598-3(new) : Information Technology - Software product evaluation - Part 3: Process for developers

ISO/IEC 14598-4(new) : Information Technology - Software product evaluation - Part 4: Process for acquirers

ISO/IEC 14598-5(new) : Information Technology - Software product evaluation - Part 5: Process for evaluators

ISO/IEC 14598-6(new) : Information Technology - Software product evaluation - Part 6: Documentation of evaluation modules

ISO/IEC 12207 : 1995, Information technology – Software life-cycle processes.

ISO/IEC 14143 : Functional size measurement

ISO 2382-20:1990, Information technology, Vocabulary

3 Term(s) and definition(s)

For the purpose of this International Technical Report, the definitions in Clause 6 of 9126-2 and contained in Annex A of this report apply.

4 General Intent of the Quality Metrics

When using internal metrics defined in this report, the reader should refer to Clause 6 and Annex A and B in 9126-2 for their concept, description, relationship, and interpretation.

5 Basic use of internal metrics for quality characteristics

Internal metrics provides the user with the ability to measure the quality of the intermediate deliverables, and thereby predict the quality of the final products. This allows the users to detect quality issues and to take corrective actions during early stages of software development life-cycle processes.

Following set of metrics is a suggested set which gives measures or may be applied as checklists to represent software quality characteristics. They are sorted by software quality characteristics and subcharacteristics. In

trying to use new metrics, appropriate model and practical experiences should be evaluated and specified.

5.1 Functionality metrics

Internal functionality metrics indicate if the software product in question satisfies prescribed requirements and implied user needs from a developer's perspective.

5.1.1 Suitability metrics

Internal suitability metrics indicate a set of attributes for describing explicitly functions to prescribed tasks, and for determining their adequacy for performing the tasks. The determination can be done by the following;

- coverage of implemented functions versus all functions defined in functional specifications during development.

5.1.2 Accuracy metrics

Internal accuracy metrics indicate a set of attributes for demonstrating that a correct or agreeable result or effect is achieved. For example, incorrectness of computational results or inconsistency between description in user manuals and designed functions.

5.1.3 Interoperability metrics

Internal Interoperability metrics indicate a set of attributes for demonstrating the capability of interaction with designated systems.

5.1.4 Security metrics

Internal security metrics indicate a set of attributes for demonstrating capability of avoiding illegal access to the system and/or data.

5.1.5 Compliance metrics

Internal compliance metrics indicate a set of attributes causing compliance problems, which is of the software product to failing to adhere to stadards, conventions or regulations in laws and similar prescriptions which are required to be adhered.

Table 5.1.1 Suitability metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Functional adequacy	How adequate are the checked functions?	To measure the ratio of implemented functions that are suitable for performing the specified tasks to those implemented. The following may be measured; -all or parts of design specifications -test completed modules/parts of software products	X=1-A/B A= Number of functions in which problems are detected in evaluation B= Number of functions checked	0 <= X <= 1 The closer to 1, the more adequate.	X=absolute	X=count/count A=count B=count	Req spec Design Source code Review report	6.5Validation 6.6Joint review	Requirer Developer
Functional implementation completeness	How much complete is the implementation?	Review NOTE) 1. Comparing with requirement specification 2. Review by functional items	X=1-A/B A=Number of missing functions detected in evaluation. B=Number of functions described in requirement specifications NOTE: Input to the measurement process is the updated requirement specifications. Any changes identified during life cycle must be applied to the requirement specifications before using in measurement process.	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Source code Review report	6.5Validation 6.6Joint review	Requirer Developer
Functional implementation coverage	What is the ratio of correctly implemented functions to what are specified as required?	Review Note) 1. Comparing with requirement spec 2. Review by functional item.	X=A/B A= Number of correctly implemented functions confirmed in evaluation B= Number of functions described in requirement specifications Note) Input to the measurement process is the updated requirement specifications. Any changes identified during life cycle must be applied to the requirement specifications before using in measurement process.	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Source code Review report	6.5Validation 6.6Joint review	Requirer Developer
Functional specification stability	How much stable are the functional	Count the number of existing functional specifications that had	X=1-A/B A=Number of functions changed during development life cycle phases	0 <= X <= 1 The closer to 1 is the	X=absolute	A=Count B=Size X=Count/Size	Requirement specifications	6.5 Validation 6.3 Quality Assurance	Developer Maintainer

(volatility)	specifications of the system during development life cycle?	to be changed during subsequent development life cycle phases.	B=Number of functions described in requirement specifications (or any function size measure)	better.	Review report	5.3 Qualification testing 6.8 Problem Resolution 5.4 Operation	

Table 5.1.2 Accuracy metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Computational accuracy	How completely are the accuracy requirements being considered?	Count the number of functions that have implemented the accuracy requirements	X=A/B A= Number of functions in which specific accuracy requirements had been considered, as confirmed in evaluation. B= Number of functions for which specific accuracy requirements need to be considered.	0 <= X <= 1 The closer to 1, the more complete.	X=absolute	X=count/count A=count B=count	Requirement specification Design Source code Review report	Verification Joint review	Requirers Developers
Precision	How complete was the implementation of specific levels of precision for the data items?	Count the number of data items that meet the requirements of specific levels of precision and compare to the total number of data items with these requirements. The following may be measured: -all or parts of design specification; -test of completed modules or parts of software products.	X=A/B A= Number of data items implemented with specific levels of precision, confirmed in evaluation B= Number of data items that require specific levels of precision	0 <= X <= 1 The closer to 1, the more complete.	X=absolute	X=count/count A=count B=count	Requirement specification Design Source code Review report	Verification Joint review	Requirers Developers

Table 5.1.3 Interoperability metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input t measureme nt	12207 referenc e	Beneficiari es
Data exchangeability (data format based)	What is the ratio of correctly implemented data format interfaces to what are specified as required?	Count the number of data formats that an interface was implemented correctly as in the specifications.	X=A/B A=Number of data formats implementing consistent format confirmed in review B=Number of data formats to be exchanged	0 <= X <= 1 The closer to 1, the more consistent.	X=absolute	X=count/count A=count B=count	Req spec Design Source code Review report	Verification Joint review	Developers Requirers
Interface consistency (interface format based)	What is the ratio of correctly implemented interface format to what are specified as required?	Count the number of interface formats that an interface was implemented correctly as in the specifications.	X=A/B A=Number of interface formats implementing consistent format confirmed in review B=Number of interface formats to be exchanged	0 <= X <= 1 The closer to 1, the more consistent.	X=absolute	X=count/count A=count B=count	Req spec Design Source code Review report	Verification Joint review	Developers Requirers
Intersystem standard consistency	How much consistent are the standards in the systems concerned?	Count the number of standards applied that implementation was consistent among related systems.	X=A/B A=Number of implemented items consistent with standard/rule confirmed in review B=Number of items to be consistent with standard/rule	0 <= X <= 1 The closer to 1, the more consistent.	X=absolute	X=count/count A=count B=count	Req spec Design Source code Review report	Verification Joint review	Developers Requirers

Table 5.1.4 Security metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Access auditability	How complete is the implementation of access login instances considering the auditability requirements?	Review with specifying most of all illegal/incorrect operation. The following may be measured: -all or parts of design specifications; -test completed modules/parts of software products	X=A/B A= Number of information recording access log confirmed in review B=Number of information requiring access log	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Requirement specification Design Source code Review report	6.5 Validation 6.6 Joint review	Requirer Developer
Access controllability	How complete is the detection of user access to the system?	Review with specifying most of all illegal/incorrect operation	X=A/B A= Number of detected incorrect/illegal operations confirmed in review B=Number of illegal/incorrect operations to be detected	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Requirement specification Design Source code Review report	6.5 Validation 6.6 Joint review	Requirer Developer
Data corruption prevention	What is the ratio of implemented data corruption prevention in operations to the total number of operations capable of corrupting data?	Review with specifying most of all operation/access capable of corrupting/destroying data operation. The following may be measured: -all or parts of design specifications; -test completed modules/parts of software products	X=A/B A= Number of implemented operation/access with prevention confirmed in review B=Number of operation/access identified in requirements as capable of corrupting/destroying data Note: To consider security class(e.g., CS1/CS2/CS3…), data importance, recoverability…	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Requirement specification Design Source code Review report	6.5 Validation 6.6 Joint review	Developer
Data encryption	What is the data encryption ratio?	To measure the ratio of implemented instances of data encryption to the instances requiring encryption. The following may be measured:	X=A/B A=Number of data items implementing data encryption/decryption facility confirmed in review B= Number of data items requiring data encryption/decryption facility NOTE:	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Requirement specification Design Source code Review report	6.5 Validation	Developer

		-all or parts of design specifications -test completed modules/parts of software products.	Data encryption: e.g., data in open database, data in public communication facility,…						

Table 5.1.5 Compliance metrics (Compliance for Functionality)

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Satisfied compliance items coverage	How completely are the compliance items being satisfied with the standards, conventions or regulations relating to functionality?	Count the number of compliance items that have been satisfied.	X=A/B A=Number of correctly implemented items related to functionality compliance confirmed in evaluation B= Total number of compliance items	0 <= X <= 1 The closer to 1, the more complete.	X=absolute	X=count/count A=count B=count	Specification of compliance and related standards, conventions or regulations. Design Source code Review report	Verification Joint review	Requirers Developers

5.2 Reliability metrics

Internal reliability metrics indicate a set of attribute affecting to the behaviours of the system of which the software is a part during execution testing to indicate the extent of reliability of the software in that system during operation.

5.2.1 Maturity metrics

Internal maturity metrics indicate a set of attributes for such as the software freedom of failures caused by faults existing in the software itself.

5.2.2 Fault tolerance metrics

Internal fault tolerance metrics indicate a set of attributes related to the software capability of maintaining a performance level in cases of occurrence of operation faults or infringement of its specified interface.

5.2.3 Recoverability metrics

Internal recoverability metrics indicate a set of attributes such as the software with system being able to re-establish its adequate level of performacne with minimal lost of data.

5.2.4 Compliance metrics

Internal compliance metrics relating to reliability indicate a set of attributes causing compliance problems, which is of the software product to failing to adhere to standards, conventions or regulations relating to reliability which are required to be adhered.

Table 5.2.1 Maturity metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Fault detection	How many faults were detected in reviewed product?	Count the number of detected faults in review and compare it to their estimated number.	X=A/B A=Absolute number of faults detected in review B=Number of estimated faults to be detected in review (using past history)	0 <= X A high value for X implies good product quality, while A=0 does not necessarily imply fault free status of the reviewed item. NOTE: 1. It is necessary to convert this value(X) to the <0,1> interval if making summarization of characteristics.	X=ratio	X=count/count A=count B=count	Value A comes from review report Value B comes from the organization database.	Verification Joint review	Requirers Developers
Fault removal	How many faults corrected?	Measure the ratio of faults removed during design/coding to the number of faults detected in review.	X=A A=Number of corrected faults in design/coding B=Number of faults detected in review	0 <= X A high value of X implies, that less faults remain.	X=ratio	X=count A=count	Value A comes from fault removal report. Value B comes from review report.	Verification Joint review	Requirers Developers
	What is the proportion of faults removed?		Y=A/C A=Number of corrected faults design/coding C= Number of faults detected in review	0 <= Y <= 1 The closer to 1, the better. (more faults removed) NOTE: 1. It is necessary to convert this value (X) to the <0,1>	Y=absolute	Y=count/count C=count			

				interval if making summarization of characteristics.					
Test adequacy	How many required test cases are covered by the test plan?	Measure the ratio of test cases planned to the number of test cases required to obtain full test coverage.	X=B/A A= Number of tests required B=Number of test cases designed in test plan and confirmed in review	0 <= X The closer to 1, the better is the test plan compliance to testing requirements.	X=absolute	X=count/count A=count B=count	Value A comes from requirements Value B comes from test plan	QA Problem resolution Verification	Developers Maintainers

Table 5.2.2 Fault Tolerance metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Failure avoidance	How many fault patterns were brought under control to avoid critical and serious failures?	Measure the ratio of avoided fault patterns to the number of fault patterns considered *need to consider implied user needs - something other than specified. IEEE to be referred to as fault categories.	X=A/B A=Number of fault patterns having avoidance in design/code B=Number of fault patterns considered Note: Fault pattern examples - - -	0 <= X The closer to 1, the better the failure avoidance. NOTE: 1.X>1 means that additional fault patterns were avoided. 2.It is necessary to convert this value (X) to the <0,1> interval if making summarization of characteristics.	X=absolute	X=count/count A=count B=count	Value A comes from review report Value B comes from requirement specification document.	Verification Validation Joint review Problem resolution	Developers Requirers Maintainers
Incorrect operation avoidance	How many functions are implemented to avoid incorrect operations?	Measure the ratio of implemented functions to avoid critical and serious failures caused by	X=A/B A=Number of functions implemented to avoid incorrect operation patterns. B=Number of incorrect operation patterns considered	0 <= X <= 1 The closer to 1, the better. More incorrect operation	X=absolute	X=count/count A=count B=count	Value A comes from review report Value B comes from	Verification Validation Joint review Problem resolution	Developers Requirers Maintainers

		incorrect operations to the number of incorrect operation patterns that exist. NOTE: Also data damage in addition to system failure.	NOTE: Incorrect operation pattern, - - - NOTE: Pattern may be used by Fault Tree Analysis.	patterns avoided			requirement specification document.		

Table 5.2.3 Recoverability metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Restorability	How is the product capable to restore in defined cases?	Measure the ratio of implemented restoration cases to the number of items that require restoration. Cases: e.g., database checkpoint, transaction checkpoint, redo function, undo function,...	X=A/B A=Number of cases capable for restore confirmed in review B=Number of cases capable for restore stated in requirements.	0 <= X <= 1 The closer to 1, the product is more capable to restore in defined cases.	X=absolute	X=count/count A=count B=count	A comes from review document B comes from requirements or design document	Verification Joint review	Developers Maintainers
Restore effectiveness	How effective will be the restoration process in the prosuct?	Measure the ratio of restorations meeting the desired time constraint to the number of restorations tried.	X=A/B A=Number of restore cases meeting target restore time by calculation or simulation in design/code B=Number of restore cases which shall meet the target restore time considered	0 <= X <= 1 The closer to 1, more effective will be the restoration process in the product.	X=absolute	X=count/count A=count B=count	A comes from review document B comes from requirements or design document.	Verification Joint review	Developers Maintainers

Table 5.2.4 Compliance metrics (Compliance for Reliability)

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Satisfied coverage of compliance items relating to reliability	How completely are the compliance items being satisfied with the standards, conventions or regulations relating to reliability?	Count the number of compliance items that have been satisfied.	X=A/B A=Number of correctly implemented items related to reliability compliance confirmed in evaluation B= Total number of compliance items	0 <= X <= 1 The closer to 1, the more complete.	X=absolute	X=count/count A=count B=count	Specification of compliance and related standards, conventions or regulations. Design Source code Review report	Verification Joint review	Requirers Developers

5.3 Usability metrics

Internal usability metrics indicate a set of attributes affecting to its operation, with regard to easiness of use and adaptation of new operators.

5.3.1 Understandability metrics

Internal Understandability metrics indicate a set of attributes affecting to effort of users who, before regularly use of software, try to understand :

a) what kind of tasks are performed or outputs produced by the software product ;

b) what kind of users' manual tasks, operations, or inputs are needed to use the software.

These assist users to select the software product which is suitable to their intended use.

5.3.2 Learnability metrics

Internal Learnability metrics indicate a set of attributes affecting to the behaviour of user who is learning how to use the software.

5.3.3 Operability metrics

Internal operability metrics indicate a set of attributes affecting to user's human behaviour during operational testing, usability testing or user operation.

5.3.4 Attractiveness metrics

Internal attractiveness metrics indicate a set of attributes affecting to user's human behaviour expressing the extent of which user likes the software during operational testing, usability testing or user operation.

5.3.5 Compliance metrics

Internal compliance metrics relating to usability indicate a set of attributes causing compliance problems, which is of the softwarre product to failing to adhere to standards, conventions, style guides or regulations relating to usability which are required to be adhered.

Table 5.3.1 Understandability metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Completeness of description	What proportion of function (or types of functions) are described in the product description ?	Inspection (by expert)	X= A/B A= Number of functions (or types of functions) described B= Total of number of functions (or types of functions)	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint review	Requirers Developers
Demonstration availability	What proportion of functions have demonstrations?	Inspection (by expert)	X=A/B A= Number of demos implemented, confirmed in review B= Total number of functions	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint review	Requirers Developers
Evident functions	What proportion of functions(or types of functions) are evident to the user based upon start up condistions?	Inspection (by expert)	X= A/B A= Number of functions (or types of functions) evident to the user B= Total of number of functions (or types of functions)	0 <= X <=1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint review	Requirers Developers
Function understandability	What proportion of interface functions are understandable?	Inspection (by expert)	X= A/B A= Number of interface functions whose purpose is correctly described by the user B= Number of functions available from the interface	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint review	Requirers Developers

Table 5.3.2 Learnability metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Completeness of user documentation and help systems (V)	What proportion of functions are described in the user documentation and help systems	Inspection (by expert)	X= A/B A= Number of functions described B= Total of number of functions provided	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint review	Requirers Developers

Table 5.3.3 Operability metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Default value availability									
Default Value Availability	What proportion of input items provide default values?	Inspection (by expert or users)	X=A/B A=Number of input items which provide appropriate default values B=Number of input items which could provide default values	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint review	Developers Requirers
User operation cancellability									
User operation cancellability	What proportion of functions which may take more than 1 second to complete can be cancelled prior to completion?	Inspection (by expert)	X=A/B A=Number of functions which can be cancelled by the user B=Number of functions which may take more than 1 second to complete	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint review	Developers Requirers
Guidability									
Customisability	What proportion of functions can be customised?	Inspection (by expert)	X=A/B A=Number of functions which can be allocated to any available interface elements(such as keyboard, menu and icons) B=total number of functions	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint review	Developers Requirers
User understandable status or progress	What proportion of status or progress are self-explanatory?	Inspection (by expert or users)	X=A/B A=Number of status or progress giving a clear explanation and instructions for action. B=Number of status or progress to be provided to the users	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint review	Developers Requirers
Consistency									
Operational consistency	What proportion of operations behave the same way as similar operations in other parts of the system?	Inspection (by expert)	X=A/B A=Number of operations with consistent behaviour B=Total number of functions	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint review	Developers Requirers
Message readiness									

Comprehensive messages	What proportion of messages are self-explanatory?	Inspection (by expert or users)	X=A/B A=Number of messages giving a clear explanation and instructions for action. B=Number of messages to be provided to the users.	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint review	Developers Requirers
Interactive memorability	What proportion of panels having adequate number of messages which can be easily understood by the user?	Inspection (by expert or users)	X=A/B A=Number of panels giving a adequate number of messages. B=Number of panels to be provided to the users.	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint review	Developers Requirers
Operation time adequacy									
Operation time	What proportion of operation which may reduce required time to complete can be used for time comfortable?	Inspection (by expert or users)	X=A/B A=Number of operations having reduced operation time. B=Number of operations to be provided to the users.	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint review	Developers Requirers
Task time	What proportion of task which may reduce required task time to complete can be used for time comfortable?	Inspection (by expert or users)	X=A/B A=Number of tasks having reduced task time. B=Number of tasks to be provided to the users.	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint review	Developers Requirers
Operational procedure reduction									
Operational procedure	What proportion of operation which may reduce required procedures to complete can be used for operational steps comfortable?	Inspection (by expert or users)	X=A/B A=Number of operations having reduced operation procedures. B=Number of operations to be provided to the users.	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint review	Developers Requirers
Human error free									
Human error free	What proportion of functions can be operated in preventing human errors?	Inspection (by expert or users)	X=A/B A=Number of functions having human error prevention. B=Number of functions to be provided to the users.	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint review	Developers Requirers

Table 5.3.4 Attractiveness metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Attractive interaction	How attractive is the interface to the user?	Questionnaire to users	Questionnaire to access the attractiveness of the interface to users, taking account of attributes such as colour and graphical design.						
Interface appearance customisability	What proportion of interface elements can be customised in appearance.	Inspection (by expert)	X=A/B A=Number of types of interface elements that can be customised. B=Total number of types of interface elements.	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint reivew	Requirers Developers

Table 5.3.5 Compliance metrics (Compliance for Usability)

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Satisfied coverage of compliance items relating to usability	How completely are the compliance items being satisfied with the standards, conventions or regulations relating to usability?	Count the number of compliance items that have been satisfied.	X=A/B A=Number of correctly implemented items related to usability compliance confirmed in evaluation B= Total number of compliance items	0 <= X <= 1 The closer to 1, the more complete.	X=absolute	X=count/count A=count B=count	Specification of compliance and related standards, conventions or regulations. Design Source code Review report	Verification Joint review	Requirers Developers

5.4 Efficiency metrics

Internal efficiency metrics indicate a set of attributes affecting to the behaviour of computer system including software during testing or operating. To measure efficiency metrics, the stated conditions should be defined, i.e., that the hardware configuration and the software configuration of a so-called reference environment(which has to be defined in the software specifications) should be defined before measuring. If citing measured time behaviour values the reference environment should be referred.

5.4.1 Time behaviour metrics

Internal behaviour metrics indicate a set of attributes affecting to the time behaviour of computer system including software during testing or operating.

5.4.2 Resource utilisation metrics

Internal resource utilisation metrics indicate a set of attributes affecting to the utilised resources behaviour of computer system including software during testing or operating.

5.4.3 Compliance metrics

Internal compliance metrics relating to efficiency indicate a set of attributes causing compliance problems, which is of the software product to failing to adhere to standards, conventions or regulations relating to efficiency which are required to the adhered.

Table 5.4.1 Time behaviour metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Response time	What is the estimated time to complete a specified task?	Evaluate the efficiency of the operating system and the application system calls. Estimate the response time based on this. The following may be measured, -all or parts of design specifications -test complete transaction path -test complete modules/parts of software product -complete software product during test phase	X=time (calculated or simulated)	The shorter the better.	X=ratio	X=time	Known operating system. Estimated time in system calls.	Verification Joint review	Developers Requirers
Throughput time	What is the estimated number of tasks that can be performed over a unit of time?	Evaluate the efficiency of handling resources in the system. Make a factor based upon the application calls to the system in handling the resources.	X=time (calculated or simulated)	The shorter the better.	X=ratio	X=time	Known operating system. Estimated time in system calls.	Verification Joint review	Developers Requirers
Turnaround time	What is the estimated time to complete a group of related tasks as a job lot?	Evaluate the efficiency of the operating system and the application system calls. Estimate the response time to complete a group of related takss based on this. The following may be	X=time (calculated or simulated)	The shorter the better.	X=ratio	X=time	Known operating system. Estimated time in system calls.	Verification Joint review	Developers Requirers

		measured, -all or parts of design specifications -test complete transaction path -test complete modules/parts of software product -complete software product during test phase.							

Table 5.4.2 Resource utilisation metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
I/O Utilisation	What is the estimated I/O to complete a specified task?	Estimate the I/O requirement for the application.	X=number of buffers(calculated or simulated)	The shorter the better.	X=ratio	X=size	Source code	Verification	Developer
I/O Utilisation Message Density	What is the density of messages relating to I/O in the lines of code responsible in making system calls.	Estimate the number of lines of code responsible in system calls. Count the number of errors pertaining to I/O failure and warnings.	X=A/B A=number of I/O related error messages. B=number of lines of code directly related to system calls.	The greater the better.	X=ratio	X=size	Source code	Verification	Developer

Memory utilisation	What is the estimated memory size that a program will occupy to complete a specified task?	Estimate the memory requirement for the application.	X=size in bytes (calculated or simulated)	The lesser the better.	X=ratio	X=size	Estimated size of memory utilisation.	Verification	Developer
Memory utilisation message density	What is the density of messages relating to memory utilisation in the lines of code responsible in making system calls?	Estimate the number of lines of code responsible in system calls. Count the number of error messages pertaining to memory failure and warnings.	X=A/B A=Number of memory related error messages. B=Number of lines of code directly related to system calls.	The greater the better.	X=ratio	X=size	Source code	Verification	Developer
Transaction Utilisation	What is the estimated ability of the application to transmit?	Estimate the transmission requirement for the application.	X=bits/time (calculated or simulated)	The shorter the better.	X=ratio	X=time	Known operating system. Estimated time in system calls.	Verification	Developer

Table 5.4.3 Compliance metrics (Compliance for Efficiency)

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Satisfied coverage of compliance items relating to efficiency	How completely are the compliance items being satisfied with the standards, conventions or regulations relating to efficiency?	Count the number of compliance items that have been satisfied.	X=A/B A=Number of correctly implemented items related to efficiency compliance confirmed in evaluation B= Total number of compliance items	0 <= X <= 1 The closer to 1, the more complete.	X=absolute	X=count/count A=count B=count	Specification of compliance and related standards, conventions or regulations. Design Source code Review report	Verification Joint review	Requirers Developers

5.5 Maintainability metrics

Internal maintainability metrics indicate a set of attributes affecting to the effort of maintainer, user, or system behaviour including the software, when the software is maintained or modified during testing or maintenance.

5.5.1 Analysability metrics

Internal analysability metrics indicate a set of attributes affecting the maintainer's or user's effort or spent resources when trying to diagnose for deficiencies or cause of failures, or for identification of parts to be modified.

5.5.2 Changeability metrics

Internal Changeability metrics indicate a set of attributes affecting the maintainer's or user's effort by measuring the behaviour of system including the software when trying to implement a specified modification.

5.5.3 Stability metrics

Internal stability metrics indicate a set of attributes causing unexpected behaviour of system including the software when the software is tested or operated after modification.

5.5.4 Testability metrics

Internal testability metrics indicate a set of attributes that refer to the amount of designed and implemented autonomous functions aiding in testing.

5.5.5 Compliance metrics

Internal compliance metrics relating to maintainability indicate a set of attributes causing compliance problems, which is of the software product to failing to adhere to standards, conventions or regulations relating to maintainability which are required to be adhered.

Table 5.5.1 Analysability metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Activity recording	How many recording data provided to record system status.	To measure the ratio of items logged in the activity log to the number of items required to be logged. Data contains activity log related and system status related.	X=A/B A=Number of recording data confirmed in review B=Number of data to be recorded defined in the specifications	0 < X <= 1 The closer to 1, more data provided to record system status. NOTE: 1.It is necessary to convert this value to the <0,1> interval if making summarization of characteristics	X=absolute	X=count/count A=count B=count	Value A comes from review resport. Value B comes from requirement specifications.	Verification Joint review	Maintainer User
Readiness of diagnostic function	How many diagnostic function provided to analyse?	Measure the ratio of implemented diagnostic functions to their required number . This metric also contains failure analysis time oriented and cause finding oriented.	X=A/B A=Number of implemented diagnostic functions confirmed in review B=Number of diagnostic functions required	0 < X The closer to 1, the better implementation of diagnostic functions. NOTE: 1.It is necessary to convert this value to the <0,1> interval if making summarization of characteristics.	X=absolute	X=count/count A=count B=count	Value A comes from review resport. Value B comes from requirement specifications.	Verification Joint review	Maintainer User

Table 5.5.2 Changeability metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Change recordability	Are changes to specifications and program modules recorded adequately?	Record ratio of module change information	X=A/B A=Number of functions/modules having change log confirmed in review B=Total number of functions/modules	0 <= X <= 1 The closer to 1, the better. The change control 0 indicates poor change control or little changes, high stability.	X=absolute	X=count/count A=count B=count	Configuration control system Version logs Specifications	Verification Joint review	Developers Maintainers Requirers

Table 5.5.3 Stability metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Less encountering failures after change	What is the frequency of side-effect failures after modification?	Measure the frequency of side effects as a result of modification.	X=1-(A/B) A=Number of detected side-effect failures confirmed in review B=Number of failures corrected	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	A comes from review report B comes from review report	Joint review Verification	Developers Maintainers Requirers
Localisation of modification	How big is the impact of the modification?	Measure the ratio of affected variables from a modification to the total number of variables. NOTE: 1.Impacted variable is a) all variables in the instruction which was changed b) Variable which is in	X=A/B A=Number of affected variable data by modification, confirmed in review B=Total number of variables	0 <= X <= 1 The closer to 1, the bigger impact of the modification.	X=absolute	X=count/count A=count B=count	A comes from review report B comes from review reoprt	Joint review Verification	Developers Maintainers Requirers

		the same instruction with the variable defined by a).							

Table 5.5.4 Testability metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Completeness of built-in test function **Autonomy of testability**	How complete is the set of built in tests of this function?	Calculate the ratio of required built in test to implemented tests	X=A/B A=Number of implemented built-in test function confirmed in review B=Number of built-in test function required	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	A comes from review document B comes from requirements or design document	Verification Joint review	Developers Maintainers Requirers
Test progress observability	How complete is the set of built in checkpoints?	To measure the ratio of implemented checkpoints to the number of checkpoints required by design.	X=A/B A=Number of implemented checkpoints confirmed in review B=Number of designed checkpoints	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	A comes from review document B comes from design document	Verification Joint review	Developers Maintainers Requirers

Table 5.5.5 Compliance metrics (Compliance for Maintainability)

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Satisfied coverage of compliance items relating to maintainability	How completely are the compliance items being satisfied with the standards, conventions or	Count the number of compliance items that have been satisfied.	X=A/B A=Number of correctly implemented items related to maintainability compliance confirmed in evaluation B= Total number of compliance items	0 <= X <= 1 The closer to 1, the more complete.	X=absolute	X=count/count A=count B=count	Specification of compliance and related standards, conventions or	Verification Joint review	Requirers Developers

	regulations relating to maintainability?		regulations. Design Source code Review report

5.6 Portability metrics

Internal Portability metrics indicate a set of attributes affecting to the behaviour of the operator or system during the porting activity.

5.6.1 Adaptability metrics

Internal adaptability metrics indicate a set of attributes affecting to the behaviour of user who is trying to adapt software to different specified environments.

5.6.2 Installability metrics

Internal Installability metrics indicate a set of attributes affecting to the behaviour of user who is trying to install the software in a user specified environment.

5.6.3 Replaceability metrics

Internal Replaceability metrics indicate a set of attributes affecting to the behaviour of user who is trying to use the software in place of other speccified software in the environment of that software.

5.6.4 Co-existence metrics

Internal Co-existence metrics indicate a set of attributes affecting to the behaviour of user who is trying to use the software with other independent software in a common environment sharing common resources.

5.6.5 Compliance metrics

Internal compliance metrics relating to portability indicate a set of attributes causing compliance problems, which is of the software product to failing to adhere to standards, convemtions or regulations relating to portability which are required to be adhered.

Table 5.6.1 Adaptability metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Adaptable data	To measure the limitation-free operation after adapting data		X=A/B A=Number of data which are operable and has no limitation after adaptation, confirmed in review B=Total number of data reviewed as being operable after software adaptation	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint review	Developers Maintainers Requirers
Environmental adaptability (Organisation)	To measure the organizational/business adaptability readiness		X=A/B A=Number of functions ready for getting adequate level of results in different organisation or business environment, confirmed in review B=Total number of functions reviewed	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint review	Developers Maintainers Requirers
Environmental hardware adaptability (HW, network)	To measure the environmental hardware adaptability readiness		X=A/B A=Number of functions ready for getting adequate level of results in different hardware, confirmed in review B=Total number of functions reviewed	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint review	Developers Maintainers Requirers
Environmental software adaptability (OS, concurrent application)	To measure the environmental software adaptability readiness		X=A/B A=Number of functions ready for getting adequate level of results in different systems or applications, confirmed in review B=Total number of functions reviewed	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint review	Developers Maintainers Requirers
User effortless adaptation	To measure the functional availability for supporting ease-of-adaptation		X=A/B A=Number of functions ready for supporting ease-of-adaptation by user, confirmed in review B=Total number of functions for supporting ease-to-adapt functions reviewed	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Req spec Design Review report	Verification Joint review	Developers Maintainers Requirers

Table 5.6.2 Installability metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Easiness of setup retry	What is the success ratio of setup operation retry?	Keep note of retry failure during setup operation	X=1-(A/B) A=Number of retry failure during setup operation, confirmed in review B=Total number of retry operation reviewed	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Review reoprt	6.5 Validation	Developer
Installation effort reduction	What is the implementation ratio of effort reduction?	Keep note of installation steps/time failure during installation operation	X=A/B A=Number of installation steps/time confirmed in review B=Number of installation steps/time prescribed NOTE: Prescribed: e.g., number of panels/commands/manual operation to reach target operation,…	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Review report	6.5 Validation	Developer
Installation flexibility	What is the implementation ratio of flexibility?	Keep note of the number of customisable installation operation that you are able to do during installation operation	X=A/B A=Number of implemented customisable installation operation confirmed in review B=Number of customisable installation operation specified NOTE: Customisable: e.g., nesting depth, numner of panels,…	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Requirement specification Review report	6.5 Validation	Developer

Table 5.6.3 Co-existence metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
			X=A/B A= Number of entities which product can co-exist B= Number of entities in production environment that needs co-existency						

Table 5.6.4 Replaceability metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Data continuity	What's the amount of data that remain unchanged?	Count the number of data that continue to be used.	X=A/B A=Number of data of the software to be replaced that continue to be used, confirmed in evaluation. B=Number of data to be continued from old software	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Design Source code Review report Test report	Verification Joint review	Requirer Developer Maintainer
Functional inclusiveness	What's the amount of functions that remain unchanged?	Count the number of functions that present the results.	X=A/B A=Number of functions covered by new software that produces similar results, confirmed in review B=Number of functions checked in old software	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count	Design Source code Review report Test report	Verification Joint review	Requirer Developer Maintainer

Table 5.6.5 Compliance metrics (Compliance for Portability)

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Satisfied coverage of compliance items relating to portability	How completely are the compliance items being satisfied with the standards, conventions or regulations relating to portability?	Count the number of compliance items that have been satisfied.	X=A/B A=Number of correctly implemented items related to portability compliance confirmed in evaluation B= Total number of compliance items	0 <= X <= 1 The closer to 1, the more complete.	X=absolute	X=count/count A=count B=count	Specification of compliance and related standards, conventions or regulations. Design Source code Review report	Verification Joint review	Requirers Developers

5.7 Pure internal metrics

Table 5.7.1 Pure internal metrics

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Coherence									
Traceability	To measure effectiveness of documentation and design structure and code of software product in mapping functions from requirements to implementation.		X=A/B A=Number of traceable items confirmed in review B=Number of items checked	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count			
Cyclomatic number	To measure the level of complexity of the software design and coding structure		e-n+2p e: # of sides n: # of edges p: # of adjacent components						
	To measure complexity of design control structure. (refer to IEEE 982.1)		IFC(Information Flow Complexity) =(fanin x fanout)2						
Self-descriptiveness									
Modularity									
Self-containedness									
Program size			(N1+N2)log2(n1+n2) N1: operator occurrences N2: operand occurrences n1: total # of operators n2: total # of operands						
Conditional statement	To measure the complexity level of coded modules		X=A A= Number of conditional statements	0 <= X	X=	X=size A=size			

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Scale type	Measure type	Input to measurement	12207 reference	Beneficiaries
Unified data reference	To measure the data unification		X=A/B A=Number of data references with unified name confirmed in review B=Total number of data references	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count			
Adequacy of variable names	To measure the variable names adequacy		X=A/B A=Number of variables with adequate names confirmed in review B=Total number of variables	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count			
Data-coupled module ratio	To measure the data-coupled module ratio		X=A/B A=Number of data-coupled modules confirmed in review B=Total number of all modules	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count			
Program statements	To measure the program source statement		X=A A=Total number of program statements	0 <= X	X=	X=size A=size			
Average module size	To measure the average module size		X=A/B A=Total lines of source statements in all modules B=Total number of all modules	0 <= X	X=absolute	X=size A=size			
Function-coupled module ratio	To measure the function-coupled module ratio		X=A/B A=Number of function-coupled modules confirmed in review B=Total number of all modules	0 <= X <= 1 The closer to 1, the better.	X=absolute	X=count/count A=count B=count			

Annex A (Informative)

A.1 Definitions

Definitions are from ISO/IEC 14598-1 and ISO/IEC 9126-1 unless otherwise indicated.

A.1.1 Quality

External quality: The extent to which a product satisfies stated and implied needs when used under specified conditions.

Internal quality: The totality of attributes of a product that determine its ability to satisfy stated and implied needs when used under specified conditions.

NOTES:

1. The term "internal quality", used in this technical report to contrast with "external quality", has essentially the same meaning as "quality" in ISO 8402.

2. The term "attribute" is used (rather than the term "characteristic" used in 3.1.3) as the term "characteristic" is used in a more specific sense in ISO/IEC 9126 series.

Quality: The totality of characteristics of an entity that bear on its ability to satisfy stated and implied needs. [ISO 8402]

NOTE:

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).

Quality in use: The extent to which a product can be used by specified users to meet their needs to achieve specified goals with effectiveness, task efficiency and satisfaction in a specified context of use.

Quality model: The set of characteristics and the relationships between them which provide the basis for specifying quality requirements and evaluating quality.

A.1.2 Software and user

Software: All or part of the programs, procedures, rules, and associated documentation of an information processing system. (ISO/IEC 2382-1: 1993)

NOTE:

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

Software product: The set of computer programs, procedures, and possibly associated documentation and data designated for delivery to a user. [ISO/IEC 12207]

NOTE:

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

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.

A.1.3 Measurement

Attribute: A measurable physical or abstract property of an entity.

Direct measure: A measure of an attribute that does not depend upon a measure of any other attribute.

External measure: An indirect measure of a product derived from measures of the behaviour of the system of which it is a part.

NOTES:

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

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

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

Indicator: A measure that can be used to estimate or predict another measure.

NOTES:

1. The measure may be of the same or a different characteristic.

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

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.

Internal measure: A measure derived from the product itself, either direct or indirect; it is not derived from measures of the behaviour of the system of which it is a part.

NOTE:

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

Measure (noun): The number or category assigned to an attribute of an entity by making a measurement.

Measure (verb): Make a measurement.

Measurement: The process of assigning a number or category to an entity to describe an attribute of that entity.

NOTE:

"Category" is used to denote qualitative measures of attributes. For example, some important attributes of software products, e.g. the language of a source program (ADA, C, COBOL, etc.) are qualitative.

Metric: A measurement scale and the method used for measurement.

NOTES:

1. Metrics can be internal or external.

2. Metrics include methods for categorising qualitative data.