### Licence Agreement



You are about to download material which is subject to strict copyright conditions. Please read these terms and conditions carefully. By accepting them, you are entering into a binding contract. In all countries, there are civil and criminal penalties for copyright infringements.

The document you download is the copyright of ISO, and may not be stored, reproduced, transferred or resold by any means, except as follows.

The document is a single-user, non-revisable Adobe Acrobat PDF file. You are purchasing a single-user licence to store this file on your personal computer.

You may print out and retain ONE printed copy of the PDF file.

This single-user licence and permission to print one copy is valid for each purchased and paid copy.

This printed copy is fully protected by national and international copyright laws, and may not be photocopied or reproduced in any form. Under no circumstances may it be resold. Under no circumstances may the electronic file you are licencing be copied, transferred, or placed on a network of any sort.

If you have any difficulties concerning the above terms or if you have any question regarding the ISO copyright, please contact us:

ISO copyright Office Case postale 56 CH-1211 Geneva 20

Fax

+41 22 749 09 47

E-mail

copyright@iso.org

### ISO/IEC JTC1/SC7/WG6

### **EVALUATION AND METRICS**

TITLE

ISO/IEC 9126-2: Information Technology - Software product quality

-Part 2: External Metrics

DATE:

Nov. 17, 1999

SOURCE:

ISO/IEC JTC1/SC7/WG6

WORK ITEM:

Project 7-13-01-02

STATUS:

Version 5.8 - reflects ISO/IEC JTC1/SC7/WG6 Crituba, Brazil meeting in May,

1999 and comments of 2<sup>nd</sup> PDTR Ballot

REFERENCE:

6N/443(7N2073), 6/N 425,WG6-ROMA12R, 7N1809, 6/N 403R(7N 1767),7N1669,

6/N 363 (7N1563)

DOCUMENT TYPE:

Preliminal Draft Technical Report

ACTION:

For preliminary work for DTR Ballot.

EDITOR:

Atsushi Yamada

System Integration Technology Center, Information and Industrial Systems &

Services Company, Toshiba Corporation 3-22, Katamachi, Fuchu-shi, Tokyo 183-8512

, Japan

TEL: +81-42-340-6399, FAX: +81-42-340-6013

E-mail: atsu.yamada@toshiba.co.jp

**CO-EDITOR:** 

Vipula GODAMUNNE

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

REVIEWERS

Ms. Antonia Jeanrenaud (Italy), Mr. Dave Hufton (Singapore),

Dr. Jili Vanicek (Czech Republic),

Mr. Vipula Godamunne (Australia), Dr. Alain Abran (Canada),

Ms. Carole Nadeau (Canada: Previous Co-Editor), Mr. Alain April (Canada: Previous Co-Editor)

Dr. Chris Vermaak(Sauth Africa: Previous Co-Editor)

### **REVISION HISTORY**

Document#	Version	Date	Description
6/N 456	5.8	Nov. 99	Prelimianal work for Draft techinical report for DTR ballot reflecting 2 <sup>nd</sup> PDTR ballot comments and Critiba, Brazil meeting in May 1999. It is prepared for Knazawa, Japan meeting in Nov. 99.
6/N 443	5.7	Jan. 99	Proposed draft techinical report for 2 <sup>nd</sup> PDTR ballot reflecting Sydney meeting discussion
6/N	5.6	Nov.98	Version 5.6 – revised working draft after PDTR registration (for Sydney, Australia meeting Nov. 23, 1998)
6/N 425	5.5	May. 98	Working version for South Africa meeting discussion, which is under revising to resolve both of ballot comments before Roma meeting and additional comments after Roma.
Roma 12R	5.4	Nov.97	Working version reflecting discussion and work to coordinate 9126 part2 and 3 as a meeting work product.
Roma 12	•	Nov. 97	Working version reflecting discussion and work to coordinate 9126 part2 and 3 as a intermediate meeting work product.
7N 1809	-	Nov. 97	PDTR letter ballot summary - approved with comments
7N 1767 6/N 403R	5.3.2	Jun. 97	Working version after the review in California June 1997 and circulated in SC7 for PDTR Ballot
403R	5.3.2,	Jun. 97	Working version after the review in California June 1997
	5.3.1 5.3		(3 <sup>rd</sup> Walnut Creek meeting version proposing more widely resolutions to the ballot comments)
Walnut Creek	5.2	Jun. 97	Working version for review in California June 1997
14R			(2nd Walnut Creek meeting version proposing more widely resolutions to the ballot comments)
Walnut Creek	5.1	Jun. 97	Working version for review in California June 1997
14			(1st Walnut Creek meeting version proposing more widely resolutions to the ballot comments)
7N 1669	-	Feb. 97	WD & PDTR Registration letter ballot summary
			- approved with comments
	5.0	Jan. 97	Working version for review in California June 1997
Curitiba	4.9	Nov. 96	Working document for review at Curitiba meeting
6/N 363	4.8	July 96	Working document for PDTR after Prague meeting May 1996 and circulated in SC7 for WD & PDTR Registration Ballot
6/N 320		May. 96	Working document for review at Prague meeting May 1996
		May. 95	Working document for review at Brisbane meeting May 1995
		May. 94	Working document for review at Ottawa meeting May 1994

Reference number of working document: ISO/IEC JTC1 /SC7 /WG6 6N456

Date: 1999-11-17

Reference number of document: ISO/IEC DTR 9126-2

Committee identification: ISO/IEC JTC1 /SC 7/WG 6

Secretariat: Canada and Japan

### Information Technology –

Software Product Quality - part2: External Metrics

Titre — Titre — Partie n: Titre

Document type: International technical report

Document subtype: if applicable Document stage: (40) Enquiry Document language: E

ISO Basic template Version 3.0 1997-02-03

### Contents

4	SCOPE	
E	SOUTE	
2	CONFORMANCE	
٤	CONFORMANCE	
3	REFERENCE(S)	
,		
1	TERM(S) AND DEFINITION(S)	2
•		
5	SYMBOLS (AND ABBREVIATED TERMS)	3
3	GENERAL INTENT OF THE QUALITY METRICS	3
c	3.1 CONCEPT DESCRIPTIONS	
•	6.1.1 External metrics	, 4
	6.1.2 Internal metrics	4
	6.1.3 Quality in use metrics	5
6	3.2 INTERPRETATION OF MEASUREMENT	6
	6.2.1 Direct measure for software	٠ و
	6.2.2 Indirect measure for software 6.2.3 Indicators	C
a	5.3 METRICS DESIRABLE PROPERTIES	6
(		
7	BASIC USE OF EXTERNAL METRICS FOR QUALITY CHARACTERISTICS (METRICS TABLES)	7
7	7.1 FUNCTIONALITY METRICS	g
•	7.1.1 Suitability metrics	£
	7.1.2 Accuracy metrics	£
	7.1.3 Interoperability metrics	10
	7.1.4 Security metrics	1U
	7.1.5 Compliance metrics	IL 11
	Functional implementation completeness	11
	Functional implementation coverage	11
	Functional specification stability (volatility)	12
	Table 7.1.2 Accuracy metrics	13
	Accuracy to expectation	13
	Computational Accuracy	13 13
	Precision	14
	Data exchangeability (Data format based)	14
	Data exchangeability (User's success attempt based)	14
	Table 7.1.4 Security metrics.	15
	Access auditability	15
	Access controllability	10 46
	Data corruption Prevention	10
	Functional compliance	17
	Intersystem standard consistency	17
7	7.2 RELIABILITY METRICS	18
•	7.2.1 Maturity metrics	18
	7.2.2 Fault tolerance metrics	18
	7.2.3 Recoverability metrics	18
	7.2.4 Compliance metrics	IԾ 10
	Table 7.2.1 Maturity metrics  Estimated latent failure density	19 19
	Estimated latent fault density	19
	Failure density	20
	(Fault density)	20
	Failure Resolution	0.4

Fault Removal	2:
Mean time between failures (MTBF)	.,2
Test coverage	2
Test overcome	2
Table 7.2.2 Fault tolerance metrics.	21
Breakdown avoidance	21
Failure avoidance	2
Incorrect operation avoidance	2
Table 7.2.3 Recoverability metrics	
Availability	
Mean down time	21
Restartability	29
Restorability	36
Restore effectiveness	31
Table 7.2.4 Compliance metrics (compliance for reliability)	3
Reliability compliance	3 <sup>,</sup>
7.3 USABILITY METRICS	32
7.3.1 Understandability metrics	31
7.3.2 Learnability metrics	32
7.3,3 Operability metrics	33
7.3.4 Attractiveness metrics	33
7.3.5 Compliance metrics	
Table 7.3.1 Understandability metrics	34
Completeness of description	34
Demonstration Availability	34
Evident functions	34
Function understand-ability	32
Understandable Input and Output	34
Table 7.3.2 Learnability metrics	
Ease of function learning	30
Ease of performing task learning	٠عد
Ease of use of help system  Effectiveness of help system	?f
Effectiveness of user documentation and help systems	36
Tutorial Readiness	36
Table 7.3.3 Operability metrics.	37
Conforms with operational user expectations	
Operational Consistency	37
Controllable	38
Error correction	38
User operation cancelability	38
Suitable for the task operation	39
Default value availability	39
User operating time adequacy	39
Self descriptive (Guidable)	41
Guidability	4.1 4.1
Message readiness	4444 2
Self descriptive (Guidable)	
Self-explanatory error messages	مه
Operational error recoverability	
Time Between Human Error Operations	46
Undoability	
Suitable for individualisation	
Customisability	
Suitable for individualisation	49
Operation procedure reduction	49
Table 7.3.4 Attractiveness metrics.	50
Attractive interface	50
Interface appearance customisability	50
User operational frequency	50
Table 7.3.5 Compliance metrics (compliance for usability)	52
Satisfaction coverage of compliance items relating to usability	52
7.4 EFFICIENCY METRICS	
7.4.1 Time behavior metrics	
7.4.2 Resource utilization metrics	54

	Compliance metrics	
Dagner	rse time	• • • • • • • • • • • • • • • • • • • •
Respon	ponse time	
Maa	n response fulfillment ratio	
Wor	st case response time ratio	
Throug	hput	
Theo	oughput time	
Man	n throughput fulfillment ratio	
Throng	hputhput	
Wor	st case throughput ratio	
Turnar	ound time	
Turnare	naround time	
Maa	n turnaround fulfillment ratio	
IVICA	ound time	
War	st case turnaround time ratio	
WUII Table 7	4.2 Resource utilisation metrics	
1 and 7	rices resource utilization	,
1/O dev	levices utilisation satisfaction	
1/O ( Maon 1	I/O fulfillment ratio	
Daar	aiting time of I/O devices utilisation	***************************************
OSCI W	rices resource utilization	
770 aev	ble I/O utilisation	
V ISIL	st case I/O utilisationst	
wor:	y resource utilization	***************************************
Memor	y resource utilization	
Mea	n memory fulfillment ratio	******
VISI	ble memory utilisation	
Wors	st case memory utilisation	
Transm	ission resource utilization	*******
Mear	n transmission fulfillment ratio	•••••
Tran	smission capacity utilisation satisfaction	
Visit	ole transmission utilisation	
Wors	st case transmission utilisation	
Visit	ole synchronisation	
Table 7	4.3 Compliance metrics (compliance for efficiency)	
Satis	faction coverage of compliance items relating to efficiency	
	AINTAINABILITY METRICS	
7.5.1	Analyzability metrics	
7.5.2	Changeability metrics	
7.5.3	Stability metrics	
7.5.4	Testability metrics.	
7.5.5	Compliance metrics	
l'able 7.	5.1 Analysability metrics	
Diag	nostic function support	
Data	recording during operation	
Failu	re analysis time	
Findi	ing results of failure case	
Statu	s monitoring during operation	
Гable 7.	5.2 Changeability metrics	
Chan	ge recordability	
Ease	of parameterisa-tion	
Read	iness for change	
Time	spent to implement the change for user's satisfaction	
Time	spent to implement change by the maintainer	
Γable 7.	5.3 Stability metrics	
Less	encountering failures after change	
Local	lisation of modification ( Emerging failure after change)	
lable 7.	5.4 Testability metrics	
Effor	tless testing	
Readi	iness of built-in test function	
Test r	restartability	*******
Table 7.	5.5 Compliance metrics (compliance for maintainability)	*******
Satisf	action coverage of compliance items relating to maintainability	
Po	DRTABILITY METRICS	-
'.6.1	Adaptability metrics	******
	- market in the second	
	Installability metrics	
7.6.2 7.6.3	Installability metrics	

		77.0
7.6.4	Co-existence metrics	76
7.6.5	Otimes matrice	,,,,,,,,,,,,,,,,,,,
Table 7.	C 1	
Adap	table data	77
Envi	onmental adaptability (Organization adaptability to infrastructure of organization)	78
r ·		
* 7	68. 41	• •
90 J. L. E 71	( ) In at Hability matrice	
ъ,		
<b>^</b>	-t'1 installation floribility	
T 11 7	∠ 2 D - la sachility matrice	
D 4		
13	d. d	,,,,,,,,,,,,,,,,,,,
60.11.6	< 1.0	.,,,,,,,,,,,,,,,,,,,,,,,
Conc	urrent multiple software use with less constraints	84
Table 7	6.5 Compliance metrics (compliance for portability)	84
Satis	action coverage of compliance items relating to portability	O.F
ANNEX A	(INFORMATIVE) DESCRIPTIONS OF THE METRICS TABLES	00
A 1 Mere	NOS MAME	85
A 2 Dunn	OSE OF THE METRICS	
4.2 Meri	TOD OF ADDITION	
A A MEA	SUREMENT, FORMULA AND DATA ELEMENT COMPUTATIONS	85
A.4 IVIEA	RPRETATION OF MEASURED VALUE	85
A.3 INTE	RIC SCALE TYPES	86
A.O IVIETE	SUREMENTS TYPES	87
1 7 1	6'	,,,,,,,,,,,,,,,,,,,,,,,
A 77 1	1 Possetional aire tuna	
4 7 1	O Due - same aims tomo	, ,
A 77 1	2 Hilling decourage size maggire type	,,
A 77 1	A Chasified appreciage procedure step type	,,
. 70	The amount time	
1 77	1 Custom anaestian tima tuna	
A 7 1	2 Evacution time type	
470	2 Haar time type	.,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
A.7.2	.4 Effort type	91
A.7.2	4 Enort type5 Time interval of events types	91
A.7.3 C	ount measure type	92
	A.Dttttttt	,
171	2 No b af datastad inconsistency type	🗸 🖰
A 7 2	A Number of change type	.,.,,,,,,,,,,,,,,,,,,,,,,,,,,
4 7 1	E Number of detected failure type	
472	6 Number of attempt (trial) type	
A 7 3	7 Stroke of human operating procedure type	
A 7 2	Q Coore type	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
A 0 Ixinte	P TO ME A CLIDE MACNIT	
4 0 ICO/I	EC 122A7 SLCP REEDENCE	
A.10 BEN	EFICERIES	
ANNEX B	(INFORMATIVE) REMARKS FOR BETTER USE OF METRICS	94
,	ACCOUNTS OF CONSTRAINTS OF APPLIED METRICS	92
B.1 TAKE	ACCOUNTS OF CONSTRAINTS OF APPLIED METRICS	96
B.2 VALI	DITY DEMONSTRATION AND USE OF METRICS	97
B.3 PRED	ICTION USE OF METRICS	Qf
B.4 Dete	CT QUALITY PROBLEM PRONE COMPONENTS	as
B.5 DISPI	ATYING MEASUREMENT RESULTS	
ANNEX C	(INFORMATIVE) EXAMPLE OF QUALITY MEASUREMENT IN THE INDUSTRY USING I	SO/IEC 9126
CONCEPT	· · · · · · · · · · · · · · · · · · ·	00
C.1 Ove	erview of development and quality process	98 40^
C 1 Ow	dity Anneagh Stage	
Ston	#1 Goal Quality#2: Design Conformity	
Step	#2: Design Conformity#3: Software Detailed Design Conformity	104
Sten	# Y ADDIWATE DESIRED DESIRED CONCERNITY	

Step #4: Software Coding Conformity	105
Step #4: Software Coding Conformity	106
Step #5,6,7: Software Testing	400
C.3 Measuring Quality	100

### **Foreword**

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for world-wide 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.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 3.

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 Technical Report ISO/IEC 9126-2 was prepared by the Joint Technical Committee ISO/IEC JTC1, Information Technology, Subcommittee SC7, Software Engineering.

vii

ISO/IEC TR 9126-2 © ISO/IEC

### Introduction

This international technical report (ITR) provides external metrics 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 metrics and data elements listed in this ITR are considered to be a basic set. Developers, evaluators, quality managers and acquirers may select metrics and data elements from this technical report for defining requirements, evaluating software products, measuring quality aspects and other purposes. They may also use metrics and data elements, which are not included here. This report is applicable to any kind of software product, although each of the metrics is not always applicable to every kind of software.

ISO/IEC 9126-1 defines terms for global characteristics and how these characteristics are decomposed into subcharacteristics. ISO/IEC 9126-1, however, does not describe how to state requirements with respect to ubcharacteristics, or how, for a given product, any of these subcharacteristics could be measured. These two objectives cannot be achieved by further decomposing the subcharacteristics recursively in the same way. Something new is required. To attempt to partially fill this gap between the subcharacteristic concept (ISO/IEC 9126-1) and a measurable characteristic, technical reports are presented: 9126-2 on external metrics, 9126-3 on internal metrics and 9126-4 quality-in-use metrics.

### Software Product Quality - part2 : External Metrics

### 1 Scope

This international technical report (ITR) presents external metrics for quantitatively measuring software quality in terms of characteristics and subcharacteristics defined in ISO/IEC 9126-1 and explains methods for determining characteristics values from attributes values.

This ITR 9126-2 External Metrics presents:

- a general intent of quality metrics (Clause 6)
- which measurable attributes contribute to which software quality characteristics (including subcharacteristics) so that they can serve as metrics for these characteristics
- the metrics' desirable properties

a quality approach experimentation example

This report contains the terminology related to the metrics measurements, the usage of metrics in the life cycle process, and the introductory suggested sets of external and quality-in-use metrics for each software quality characteristic and subcharacteristic. This report provides a guide to the user of metrics for planning evaluations, selecting metrics, designing metrics, applying metrics, and interpreting measurement data.

NOTE: Out of scope

This ITR 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. Some attributes may have a desirable range of values, which does not depend on specific user needs but depends on generic factors; for example, human cognitive factors.

This ITR is allowed to be applied to any kind of software for any application, because this ITR does not include any requirements for specific application domain. Therefore, users of this ITR can select or develop and apply the suitable metrics and measures which may be from this ITR or any other references for their individual application domain. For example, the specific measurement of quality characteristics such as safety, security and human factors when these are mission-critical. For critical applications, it is highly recommended to use other specific standards that give precise guidance on how to achieve and measure the required critical quality characteristics such as safety, security, and human factors (for example, these may be found in IS or ITR provided by IEC 65 and JTC1/SC27).

Users of this ITR include, as an example:

- 1. Acquirer (an organization that acquires or procures a system, software product or software service from a supplier);
- 2. Evaluator (an organization that performs an evaluation. An evaluator may, for example, be a testing laboratory , the quality department of a software development organization, a government organization or an user);

3. Developer (an organization that performs development activities, including requirements analysis, design, and testing through acceptance during the software life cycle process);

- 4. Maintainer (an organization that performs maintenance activities);
- 5. Supplier (an organization 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) when validating software quality at qualification test;
- 6. User (an individual or organization that uses the software product to perform a specific function) when evaluating quality of software product at acceptance test;
- 7. Quality managers (an organization that perform an systematic examination of the software product or software services) when evaluating software quality at qualification test.

### 2 Conformance

Since this is an ITR, there are no conformance requirements. Organizations and/or users (acquirers, developers, etc.) should stipulate which clauses of this ITR are needed, as stated in the Introduction.

### 3 Reference(s)

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

ISO/IEC 9126 : 1991, 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: 1999, Information Technology - Software product evaluation - Part 1: General overview

SO/IEC 14598-2: 1999, Information Technology - Software product evaluation - Part 2: Planning and management

ISO/IEC 14598-3: 1999, Information Technology - Software product evaluation - Part 3: Process for developers

ISO/IEC 14598-4: 1999, Information Technology - Software product evaluation - Part 4: Process for acquirers

ISO/IEC 14598-5: 1998, 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: 1998, Functional size measurement

ISO 2382-20:1990, Information technology, Vocabulary

### 4 Term(s) and definition(s)

For the purposes of this ISO/IEC 9126-2, the definitions contained in ISO/IEC 14598-1 and ISO/IEC 9126-1apply.

© ISO/IEC ISO/IEC TR 9126-2

### 5 Symbols (and abbreviated terms)

The following symbols and abbreviations are used in this ITR:

SQA Software Quality Assurance (Group)

### 6 General intent of the quality metrics

These reports (9126-2, 9126-3 and 9126-4) provides a suggested set of quality metrics (external, quality in use and internal metrics) to be used with the 9126-1 quality model. The user of these technical reports may modify the metrics defined, and/or may also use metrics not listed. When using a modified or a new metric not identified in these ITRs, the user should specify how the metrics relate to the 9126-1 quality model or any other substitute quality model that is being used.

The user of these ITRs may select the quality characteristics and subcharacteristics to be evaluated, from ISO/IEC 9126-1; identify the appropriate direct and indirect measures to be applied, identify the relevant metrics and then interpret the measurement result in a objective manner. The user of these ITRs also may select product quality evaluation processes during software life-cycle from ISO/IEC 14598 series of standards. These give methods for measurement, assessment and evaluation of software product quality. They are intended for use by developers, acquirers and independent evaluators, particularly those responsible for software product evaluation.

The metrics desirable properties (clause 6.3) should be applied to the new or modified metric.

### 6.1 Concept descriptions

This sub clause will reinforce the ISO/IEC 9126-1 concepts of internal, external and quality in use metrics before explaining how to use these quality metrics.

The internal metrics may be applied to a non-executable software product during it's development stages (such as request for proposal, requirements definition, design specification or source code). Internal metrics provide the users with the ability to measure the quality of the intermediate deliverables and thereby predict the quality of the final product. This allows the user to detect quality issues and take corrective actions during the early stages of the development life cycle process.

The external metrics may be used to measure the quality of the software product by measuring the behavior of the system of which it is a part. The external metrics should only be used during the testing stages of the life cycle process and during any operational stages. This is achieved by executing the software product in the system environment that it is intended for. (It can be performed in in-house testing site or independent testing laboratory over the system environment site that it is intended for.)

The quality in use metrics measure the extent to which a product meets the needs of specified users to achieve specified goals with effectiveness, productivity, safety and satisfaction in a specified context of use. This can be only achieved in real life system environment.

What bind these three types of metrics (internal, external and quality in use) in an objective manner during an evaluation process are the quality criteria to be used for determining the quality of the software product. The metrics should be applied in a manner that supplements each other during the evaluation.

The process driving mentioned in the above paragraphs could be implemented in the following way. When the software quality requirements are defined, the software quality characteristics or subcharacteristics, which represent the quality requirements, are listed. Then, the appropriate external metrics and acceptable ranges are specified to quantify the quality criteria, which validate that the software meets the user needs. The internal quality attributes of the software are then defined and specified to plan to achieve the required external quality characteristics finally and to build them into the intermediate product during development. Appropriate internal metrics and acceptable range are specified to quantify the internal quality characteristics so that they can be used for verifying that the intermediate software meets the internal quality specifications during the development. Quality in use metrics measure the extent to which a product meets the needs of specified users to achieve specified goals with effectiveness, productivity, safety and satisfaction, when the software is consequently operated as a part of the

ISO/IEC TR 9126-2 © ISO/IEC

system in the individual user's environment. The relationship with other quality characteristics depends on the type of the user.

It is recommended the use of internal metrics having a relation as strong as possible with the target external metrics, so that they can be used to predict the values of external metrics. However, it is often difficult to design a rigorous theoretical model, which provides a strong relationship between internal metrics and external metrics. Therefore, a hypothetical model containing ambiguity may be designed and the extent of the relationship may be modeled statistically during the use of metrics.

In relation to these definitions the next sub clause will present the interpretation of these new concepts.

### 6.1.1 External metrics

External metrics should be designed to:

- 1.represent software product quality which includes software quality characteristics and subcharacteristics defined in ISO/IEC 9126-1, during testing or operation;
- 2. validate that the software satisfies external quality requirements;
- 3.predict the actual quality in use;
- 4.describe the extent to which the software product keeps on satisfying user's stated or implied needs during the actual operation by user.

External metrics should be designed to employ the following measurements:

- 1.Measurements of software behavior in testing and operating, and in cooperation with other software, hardware, or system;
- 2.Measurements of user behavior.

### NOTES:

- 1. Measurements of software behavior include measurements of incidents which are threats to human life and health, environmental natural resources, data destruction, inconsistency or misleading of information, broken security, degrade of service, advantages or profits in a market, gain or loss of economy etc.
- 2.The external metrics may be used to predict and to indicate the density of potential faults remaining in the software.
- 3.Internal measurements may be used to calculate external measures. For example, the number of program steps, which is an internal measure, may be used to normalize an external measure, such as failure occurrences against size of software, that is, failure density during testing.

### 6.1.2 Internal metrics

Internal metrics provide users, evaluators, testers, developers, quality managers, or managers with benefits whereby they are able to evaluate software quality of intermediate and final products during the time period before the software product is executable.

Internal metrics should be designed to:

© ISO/IEC ISO/IEC TR 9126-2

1.represent software quality of intermediate and final products for characteristics which includes software quality characteristics and subcharacteristics defined in ISO/IEC 9126-1, during software product is not executable;

2.guide to planning and implementation of designs, programs or processes improvement, which affect intermediate and final software products;

3.verify that the intermediate and final software products satisfy internal quality requirements which are quality improvement plans for designs, programs or processes;

4.predict the external metrics or quality.

Internal metrics should be designed to employ following measurement:

Measurements of static software attributes which have appeared in the text of source code, in a graph or table representation of control, data flow, or state transition structure, or in documentation of the product itself.

### 6.1.3 Quality in use metrics

The objective of software quality is to achieve quality in use. For systems with end users, this means that specified types of users should be able to carry out specific types of tasks to a required level of productivity and user satisfaction in specified environments. Evaluating quality in use validates software quality in specific user-task scenarios.

Quality in use is the user's view of the quality of a system containing software, and is measured in terms of the result of using the software, rather than properties of the software itself. Quality in use is the combined effect of the software quality characteristics for the end user.

Quality in use may be influenced by any of the quality characteristics, and is thus broader than usability, which is only concerned with the ease of use and attractiveness. Quality in use can be measured by the extent to which the specified users can achieve their goals with effectiveness, productivity (task efficiency), safety and satisfaction. Effectiveness can be measured by the accuracy and completeness with which users achieve specified goals, productivity (task efficiency) by the resources expended in relation to task effectiveness, safety by the level of impact with possibility against to specific risk arising scenarios, and satisfaction by attributes to the use of the product. Quality in use metrics should be designed to:

- 1. describe achieved level of specific types of tasks which are carried out by the specified types of users;
- 2. validate software quality in specific user-task scenarios.

Quality in use metrics should be designed to employ the measurements of the extent to which the specified users can achieve their goals of users' specific tasks.

NOTE: External usability metrics such a particulary operability and attractiveness one may employ measurements of user behavior during operation, as well as, quality in use metrics. However, There are themajor differences between external usability and quality in use metrics which are the followings:

- usability metrics are intended to give the measures represent the extent of which user support function is avilable or influences is appeared to user behaviour;
- -quality in use metrics are intended to give the measures represent the extent of which user can achieve goals of users' specific types of tasks.

### 6.2 Interpretation of measurement

The purpose of this sub clause is to clarify to the reader the interpretation of the measurement concept according to the ISO/IEC 9126-1 and ISO/IEC 14598-1.

### 6.2.1 Direct measure for software

A direct measure is a measure of an attribute that does not depend upon a measure of any other attribute.

Some measurements of software attributes can be done without being influenced by factors such as the choice of the server where the software is executed, behavior of a user, or other external factor. Those measurements are referenced here as direct measures. Some examples are size of source and executable code, total number of menu options of an application or number of configuration items.

### 6.2.2 Indirect measure for software

An indirect measure is derived from (direct or indirect) measures of one or more attribute. For example, measure of response time is not only affected by the evaluated software itself, but also by the operating environment including but not restricted to computer hardware and operating system software. So, the response time of software is perived from the response time of the computing environment as well.

### 6.2.3 Indicators

Some measures can be estimated or predicted from other measures. Those measures are referenced here as indicators, and may be useful to estimate or predict attributes that cannot be measured directly, or can not be measured at all without a model. For example, the response time is not measurable while the software is still a non-executable intermediate product. Therefore, program path length may be measured and used as an indicator to predict the future response time before the software becomes executable.

As another example, in the case that the software is executed in in-house testing, the response time under a testing environment is measurable but may vary from the response time experienced by an actual user in the final operating environment. Therefore, the response time of the software in the final user environment can be predicted from the response time of the testing environment.

Finally, in the case that measuring efficiency of the software is dominantly dependent on time measurement, the response time may be used as an indicator to represent efficiency quantitatively in the software quality evaluation.

### .3 Metrics desirable properties

The accuracy and correctness of a quality evaluation relies strongly on the metrics used on that process. In order to improve metrics confidence, a list of validation requirements, which should be present on every metric applied in an evaluation, is given below.

Whenever a metric does not satisfy these validation requirements, the metric description should explain the associated constraint and, as far as possible, how that situation can be handled.

NOTE: Those properties are suited to the requirements on measurements of ISO/IEC 14598-1, and to the requirements on the evaluation of ISO/IEC 14598-5.

Reliability (of metric): Reliability is associated with random error. A metric is free of random error if random variations do not affect the results of the metric:

- a) Repeatability: repeated use of the metric in the same product to the same evaluation specification (including the same environment) by the same evaluators, test users and environment should produce results that can be accepted as being identical,
- b) Reproducibility: use of the metric in the same product to the same evaluation specification (including the same environment) by different evaluators, test users and environment should produce results that can be accepted as being identical.

© ISO/IEC ISO/IEC TR 9126-2

NOTE: It is recommended to use statistical analysis to mitigate random variations within acceptable range.

Indicativeness (of metric): Capability of the metric to identify parts or items of the software which should be improved, given the measured results compared to the expected ones.

Availability (of metric): The metric should make it clear the conditions (e.g.: presence of specific attributes) which constrain its usage.

NOTE: The selected or proposed metric should provide documented evidence of the availability of the metric for use.

Cost Effectiveness (of metric): The metric should have a good cost-benefit relation, that is, the most expensive its application, the most important the results obtained. NOTE: As an example, metrics which demand users and hardware for testing could be more expensive than those requiring project inspection only.

Correctness (of metric): The metric should have good property which encompasses objectivity, impartiality and precision.

a) Objectivity: the metric results and its data input should be factual, i.e. not colored by the feelings or the opinions of the evaluator, test users, etc.

NOTE: Since it is impossible to avoid subjective factors, especially on usability evaluation, it is recommended that procedure for assigning the number or category is enough well reviewed to be agreeable. This does not mean that it is not applicable such a metric with user sensitive testing based on questionnaire or interview.

- b) Impartiality: the measurement should not be biased towards any particular result.
- c) Precision: Precision is determined by the design of the metric, and particularly by the choice of the material definition used as the basis for the metric. The metric user will describe the precision and the sensitivity of the metric.

Meaningfulness (of metric): the measurement should produce meaningful results about the software behavior or quality characteristics. User of metrics will describe the goal or question the metric result aims at fulfilling.

### 7 Basic use of external metrics for quality characteristics (Metrics tables)

The following set of metrics is recommended to be used as basic metrics which gives measures or may be applied as checklists to represent software quality characteristics. Although there are some practical experiences in progress, these metrics are draft and need more validation and feedback. They are listed in order of software quality characteristics and subcharacteristics.

Additional specific metrics for particular purposes are not included, but are provided in other related documents, such as functional size measurement or precise time efficiency measurement.

Metrics, which may be applicable, are not limited to these listed here. If there is a trial to apply a new metric, the appropriate model and the practical experiences should be evaluated and specified.

NOTE: It is recommended to refer a specific metric or measurement form specific standards, technical reports or guidelines Functional size measurement is defined in ISO/IEC 14143s. An example of precise time efficiency measurement can be refered from ISO/IEC 14756. For critical applications, it is recommended to use other specific standards that give precise guidance on how to achieve and measure the required critical quality characteristics such as safety, security, and human factors.

The title terms used in these metrics tables are followings.

a) Metrics name: Metrics name characterizes a measureable attribute of software and represents an unique or a group of measurements. Metrics name has the same or similar name in internal, external and quality in use metrics, when they are intended to be mutually corresponded respectively.

ISO/IEC TR 9126-2 © ISO/IEC

b) Purpose of the metrics: This helps to identify what user of metric can know by using the metric. This is described as a questionary style to look up easily in accordance with Goal / Question / Metric flamework.

- c) Method of application: This helps to understand what way are useful and recommended to apply metrics.
- d) Measurement, formula and data element computations: This helps to understand what kind of measurement, formula and data element are used to compute measure.

NOTE: In many situations one or more measurement formulas for one metric is proposed. That is why a lot of metrics can employ some measurements such are counting up the result with checked each function item or counting up occurrences of problems on the executed software or on user behavior during operational testing. It is recommended to regard metric has not a single number or category, but a data structure.

- e) Interpretation of measured value: This helps to understand the range of measured value and the interpreted better range.
- f) Metric scale types: The measurement metric scale types should be identified for each measure, when a user of letrics has the result of a measurement and use measure to calculate together or compared with other. The average, ratio or difference values may have no meaning for some measures. Such scale types are: Nominal scale, Ordinal scale, Intervals scale, Ratio scale, Absolute scale. M'=F(M), where F is the admissible function, explain what the admissible function is (if M is a metric then M'=F(M) is also a metric).

### NOTE:

- 1. Measurement (metric) scale type categories measurement mapping, from an observed (empirical) relation system to some numerical relation system. Measurement (metric) scale type does not depend on its calculation formula.
- 2. As an appropriate statistics, Mode and Frequency can be used for Nominal type, Median and Percentile for Ordinal type, Mean and Standard deviation for Interval type, and Geometric mean and Coefficient of variation for Ratio type. All arithmetic analysis of resulting count is meaningful for Absolute type.
- 3. Be aware line of code is Ratio type, because there are many different way to measure it( such as LOC, number of characters, and number of bytes). A count of the number of failure and the number of people working on a software project are Absolute type, because they can be measured only in one way.
- g) Measurements types: For designing procedure for collecting data, interpreting fair meanings, and normalizing measures for comparison, a user of metrics should identify and take account of measure type of measurement mployed by a metric.

**Size measure type**: A measure of this type represents a particular size of software according what it claims to measure within its definition. Normalizing other measures with a size measure can give comparable values in terms of units of size. The size measures in this report can be used for software quality metrics. Functional size, as well as source lines of code, is an example of one type of size that software may have.

Time measure type: The user of metrics of time measure type should record time periods, how many examined sites and how many users took part of measurements. The user of metrics should be aware that there are many ways in which time can be measured as a unit.

Count measure type: This measure type identifies number of counted number, turn, event or incident with investigation activity. Investigation activity includes reviewing, testing and operating, and using from view of human-engineering and ergonomics.

- h) Input to measurement: This helps to understand what kinds of information or documents are generally required to do measurement.
- i) ISO/IEC 12207 SLCP Reference: This suggest processes of software life-cycle processes (SLCP) defined in ISO/IEC 12207, in which the metric is beneficially applicable.

© ISO/IEC ISO/IEC TR 9126-2

j) Beneficeries: This helps to understand whose view and benefit are strogly related with the metric, though any other personel and party related to the software are also receive benefit.

### NOTE:

- 1. All the title terms used in these tables are described farther more in annex A.
- 2. All the definitions related to this clause are presented in ISO/IEC 9126-1.
- 3. All metrics described in this clause are related to the following:

The behavior of the system may be observed with following aspects:

- a) differences between actual executed results and quality requirements specifications (a view of quality validation testing);
- b) occurrences of unexpected time behavior or resource utilization during operation, which may be due to not stated implied needs.

### 7.1 Functionality metrics

An external functionality metric should be able to measure an attribute such as behavior of system containing the software. The behavior of the system may be observed with following aspects:

a) differences between actual executed results and quality requirements specification (a view of quality in use validation testing);

NOTE: Quality requirements specification for functionality is usually described as functional requirements specification.

b) insufficient performed function during actual user operation that may be implied not stated (a view of quality in use). NOTE: When a lot of implied operation or function are detected, they should be again reviewed, approved and stated. Their extent to be fullfilled should be agreed.

### 7.1.1 Suitability metrics

An external suitability metric should be able to measure an attribute such as the occurrence of the unsatisfied functions or occurrence of unsatisfied operations from behavior of system during testing and user operation.

Unsatisfied function implies:

a) a function performing task which does not conform to specified ones in requirements specifications or user manuals, or functions specified in the user manual that are simply not performed;b) a function, which does not meet appropriately the tasks for user's reasonable, intended specific use.

### 7.1.2 Accuracy metrics

An external accuracy metric should be able to measure an attribute such as the frequency of users encountering the occurrence of inaccurate matters which includes:

- a) incorrect calculation results or less precision results beyond the range of erroneous of calculation, for example, it is caused why it is too shortage of significance to calculate;
- b) inconsistency between actual operation procedures and described ones in the operation manual;

ISO/IEC TR 9126-2 © ISO/IEC

c) differences between the actual and reasonable expected results of tasks performed during operation.

### 7.1.3 Interoperability metrics

An external interoperability metric should be able to measure an attribute such as the number of functions or occurrences of less communicativeness involving data and commands, which is transferred easily between the software product and other systems, other software products, or equipment which are connected.

### 7.1.4 Security metrics

An external security metric should be able to measure an attribute such as the number of functions with, or occurrences of security problems, which are:

- failing to prevent leak of secure output information or data;
- failing to prevent lost of important data;
- failing to defend against illegal access or illegal operation.

NOTE: It is recommended that penetration tests be performed to simulate attack, because such a security attack does not normally occur in the usual testing. Real security metrics may only be taken in "real life system environment", that as "quality in use".

### 7.1.5 Compliance metrics

An external functionality compliance metric should be able to measure an attribute such as the number of functions with, or occurrences of compliance problems, which is of the software product to failing to adhere to standards, conventions or regulations in laws and similar prescriptions which are required to be adhered.

## Table 7.1.1 Suitability metrics

© ISO/IEC

Metric Name	Purpose metrics	of	Metric Name Purpose of the Method of application metrics	Measurement, form data element computations	formula outations	and Interpretation Metric of measured scale value type	on Metric red scale type	Measure	and Interpretation Metric Measure Sources of ISO/IEC Beneficerie of measured scale type input to 12207 s value type measurem SLCP	SC Benefice S
Functional	How muc	th com	How much complete Do functional tests (black $X = 1 - (A/B)$	ck X = 1 - (A / B)		0<=X<=1	Absolut	A=Count	ent Reference  0<=X<=1 Absolut A=Count Require- 6.5	ence Developer.

Functional	How much comp	How much complete Do functional tests (black $X = 1 - (A/B)$	(black X = 1 - (A/B)	0<=X<=1 Absol	0<=X<=1 Absolut A=Count Require- 6.5 Developer.	6.5 Developer.
implementation	n is the implementat	implementation is the implementation box test) of the system	system	The closer to e.	B= Count ment	Validation, SQA
completeness	according	to according to	the $A = Number of missing functions detected 1.0 is the$	ed 1.0 is the	specifica- 6.	6.3
•	requirement	requirement	in evaluation	better.	X=Count/ tion	Quality
	specifications?	specifications.			Count	Assurance,
			B = Number of functions described in	.E		5.3
		Count the number	of			Qualificati
		missing functions.	,		Evaluation	Svaluation on testing
					report	

NOTE: 1. Input to the measurement process is the updated requirement specification. Any NOTE: 2. This metric is suggested as experimental use. changes identified during life cycle must be applied to the requirement specifications before using in measurement process.

ı								
Developer.	ı, SQA			۶,		• • •		
6.5	Validation, SQA	6.3	Quality	Assurance,	5.3	Qualificati	Evaluation on testing	
Require-	B= Count ment	specifica-	tion				Evaluation	report
A=Count	B= Count		X=Count/ tion	Count				
Absolut	to e.	ગ્ર						
0<=X<=1 Absolut A=Count Require- 6.5 Developer.	ed The closer	1.0 is th	better.	i.				
(black X=1-(A/B)	functional box test) of the system A = Number of missing functions detected The closer to e.	the in evaluation.		B = Number of functions described in	requirement specifications	of	are	ones:
te Do functional tests	box test) of the	according to	requirement	specifications.		Count the number	functions that	complete versus the ones that are not.
How correct is the	1 functional	implementation?						
Functional	implementation functional	coverage	j					

NOTE: 1. Input to the measurement process is the updated requirement specification. Any NOTE: 2. This measure represents a binary gate checking of changes identified during life cycle must be applied to the requirement specifications before determining whether presence or absence of a feature. using in measurement process.

Maintainer m	
6.8 Problem Resolution 5.4 Operation	
A= Count Require- B= Count ment X= Count' specifica- Size tion Evaluation report	
0<=X<= 1 Absolut c. The closer to 1.0 is the better.	NOTE: This metric is suggested as experimental use.

S	
metrics	
me	
Accuracy	
ccura	
S	
1.2	
<u> </u>	
ıble	
<u>_</u>	

Table /.1.	Table /.1.2 Accuracy metrics	etrics				
Меtric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation Metric of measured scale value type	Measure Input to ISO/IEC type measure 12207 ment SLCP Reference	Beneficerie s
Accuracy to expectation	Are differences allowable between the actual and reasonable expected results?	Do input .vs. output test cases and compare the output to reasonable expected results.  Count the number of test cases beyond allowable	X=A / T A= Number of users encountered cases beyond allowable difference against reasonable expected results T= Operation time	0<=X The closer to Ratio. 0 is the better.	A= Count Req. spec. 6.5 Deve T= Time Validation User X= Count/ User 6.3 Time operation Quality manual Assurance	Developer User
		difference against reasonable expected results.	NOTE: Reasonable expected results may be identified in a requirement specification. a user operation manual, or hearing to users		users USETS Test report	
Computational Accuracy	Computational How often does the Accuracy end users encounter inaccurate results?	Record the number of inaccurate computations based on specifications.	X=A / T  A= Number of inaccurate computations encountered by users.  T= Operation time	0<=X The closer to Ratio 0 is the better.	A= Count Req. spec. 6.5 Deve T= Time Test report Validation User X= Count/ 6.3 Time Quality Assurance	Developer 1 User
Precision	How often does the end users encounter results with inadequate precision?	Record the number of results with inadequate precision.	X=A/T  A= Number of results encountered by the users with level of precision different from required  T= Operation time	0<=X The closer to Ratio 0 is the better.	A= Count Req. spec. 6.5 Deve T= Time Validation User X= Count/ Test report 6.3 Time Quality Assurance	Developer 1 User

Table 7.1.	(7)	ility metrics						
Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation Metric Measure of measured scale type value type	Metric N scale t	Measure type	Input to ISO/IEC measure 12207 ment SLCP Reference	Beneficeri es
Data exchangeability (Data format based)	Data How complete are exchangeability the downstream (Data format interface functions based) for specified data transfer?	Test each downstream interface function output record format of the system according to the data fields specifications.	Data exchangeable format  X= A / B  A= Number of data formats which are approved to be exchanged with other software or system during testing on data exchanges, B= Total number of data formats to be exchanged  NOTE: It is recommended to test specified data transaction.	0<=X<= 1 The closer to 1.0 is the better.	Absolut A=Count c. B=Count X=Count Count	A=Count B=Count X=Count Count	6.5 Validation	Developer on
Data exchangeability (User's success attempt based)	Data How often successful exchangeability are the data transfers (User's success between target attempt based) software and the other software?  Can user usually succeed to exchange data?	bata How often successful Count the number of exchangeability are the data transfers times that an interface (User's success between target functions was used and attempt based) software and the failed.  Can user usually succeed to exchange data?	User successful data exchange ratio  Y= 1 - (A / B)  A= Number of cases which user fail to exchange data with other software or systems  B= Number of cases which user attempt to exchange data	0<=Y<= 1 The closer to 1.0 is the better.	Absolut A=Count c. B= Count Y=Count Count	A=Count B= Count Y=Count Count	5.4 Operation	Maintainer

Table 7.1.	Table 7.1.4 Security metrics	etrics					
Mctric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation Metric of measured scale value type	Measure type	Input to ISO/IEC measure 12207 ment SLCP Reference	Beneficeri es
Access auditability	How complete is t audit trail concernithe user access to t system and data?	the Evaluate the amount ing access record that the system records in access history database.	of Access history record holding ratio 0<=X<=1 the X= A / B The closer to the A= Number of "user accesses to the system 1 is the better. and data" recorded in the access history database B= Number of "user accesses to the system and data" done during evaluation  NOTE :1. Accesses to data may be measured only with testing activities. 2. This metric is suggested as experimental use. 3. It is recommended that penetration tests be performed to simulate attack, because such a security attack does not normally occur in the usual testing		Absolut A=Count e. B=Count X=Count Count	Test spec. 6.5 Test report Validation	Developer
			Real security metrics may only be taken in "real life system environment", that as "quality in use"	u s			
Access controllability	How well is access the syste controlled?	How well is access to Try to get access to the Illegal of the system by unauthorized X= A / B controlled?  Ways.  B= Numb in specific NOTE:  unexpected, detected, operation	the Illegal operation access detectable ratio 0<=X<=1 zed X= A / B The close A= Number of detected illegal operations 1 is the be B= Number of illegal operations anticipated in specification NOTE: If it is necessary to complement unexpected illegal operations to be detected, additional intensive abnormal operation testing should be conducted.	r to	Absolut A=Count Test e. B=Count Test X=Count 5.4 Count Ope	Test spec. 6.5 Test report Validation 5.4 Operation 6.3 report Quality Assurance	Developer

Table 7.1.	4 Security me	Table 7.1.4 Security metrics (continued)						
Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation Metric of measured scale value type	Metric scale type	Measure type	Input to ISO/IEC measure 12207 ment SLCP Reference	Beneficeri es
Data corruption Prevention	How often fatal data corruption occur?	Count the occurrences of data corruption, and of fatal data corruptions events.	Frequency of fatal data corruption events X= 1 - A / N A= Number of times that a major data corruption event occurred N= Number of test cases tried to occur data corruption event corruption event Frequency of data corruption events Y= 1- B / N B= Number of times that a minor data corruption event occurred NOTE: I. Intensive abnormal operation testing is needed to obtain minor and major data corruption events.  2. It is recommended to grade impact of data corruption event such are following examples.  3. Major data corruption event: - reproduce and recover impossible: - second affection distribution to wide; - importance of data itself.  B) Minor data corruption event: - reproduce or recover possible and - no second affection distribution; - importance of data itself.	0<=X<= 1 The closer to 1.0 is the better.		Absolut A=Count e. B=Count X=Count Y= Count Count	Test spec. Test report Operation report	Maintainer
			use.					

lity)
tiona
func
e for
pliance
(com
e metrics
omplianc
5 C
Τ.
e 7
ble
Tab

		TALL AND INCHES	the combination of the complete to the company)				
Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation Metric of measured scale value type	Metric Measure scale type type	input to ISO/IEC measure 12207 ment SLCP	Beneficeri es
Functional	How completely is compliance to applicable regulations, standards and conventions being satisfied?	Previously specify Ratio of satisfied required compliance items X = 1 — (A / B) based on standards, conventions or regulations A= Number of fin laws which to be during testing adhered by software.  B= Number of the Design test cases in accordance with NOTE; compliance items.  In It may be used measured values compliance items.  Conduct functional testing trend of increasi for these test cases. items and to deta for these test cases. fully satisfied or compliance items that because problem have been satisfied.  Counting and recogniting a	Previously specify  Ratio of satisfied compliance items  required compliance items X = 1 - (A / B)  based on standards,  conventions or regulations A= Number of failed compliance items  and hered by software.  B= Number of total compliance items  Design test cases in  accordance with  NOTE:  compliance items.  1. It may be useful to collect several  measured values along time, to analyse the  measured values along time, to analyse the  Conduct functional testing trend of increasing satisfied compliance items and to determine whether they are fully satisfied or not.  Count the number of  2. It is suggested to count number of fail,  because problem detection is an objective  of effective testing and also suitable for  counting and recording.	0<= X <= 1 The closer to 1 is the better.	Absolut A= Co e. B= Co X= Count Count	Absolut A= Count Specifica- 5.3 Supple.  E. B= Count ion of Qualifica- X= Count ce and related 6.5 standards, Validation conventions or regulation s  Test specification and report	Supplier g User
Intersystem standard consistency	How consistent are the standards in the systems concerned?	Test each interface of the system for consistency with the standard identified in the specifications.	Interface standard consistent ratio X= A / B A= Number of check items of intersystem interface which are approved at testing that they are consistent with standard/rule of intersystem. B= Total number of check items of intersystem interface	0<=X<= 1 The closer to 1.0 is the better.	Absolut A=Count c. B=Count X=Count Count	nt 6.5 nt Validation nt/	Developer

ISO/IEC TR 9126-2 © ISO/II

### 7.2 Reliability metrics

An external reliability metric should be able to measure attributes related to the behaviors 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. Systems and software are not distinguished from each other in most case.

### 7.2.1 Maturity metrics

An external maturity metric should be able to measure such attributes as the software freedom of failures caused by faults existing in the software itself.

### 7.2.2 Fault tolerance metrics

An external fault tolerance metrics should be related to the software capability of maintaining a performance level in cases of occurrence of operation faults or infringement of its specified interface.

### 7.2.3 Recoverability metrics

An external recoverability metric should be able to measure such attribute as the software with system being able to re-establish its adequate level of performance with minimal lost of data.

### 7.2.4 Compliance metrics

An external reliability compliance metric should be able to measure an attribute such as the number of functions with, or occurrences of 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.

metrics	Method
Maturity	Purpose of the
Table 7.2.1	Metric Name

			T	ı		J C
Metric Name	Purpose of the Method of application metrics	Measurement, Iormula and data element computations	Interpretation Metric of measured scale	Measure	input to ISO/IEC measurem 12207	S
					ent SLCP Reference	
Estimated	How many problems	Estimated residuary latent failure density	0<=X Ratio	NPFI=	Test report 5.3	Developer
latent failure	still potentially exist	$X = \{ABS(NPFI - NAFI)\} / SIZE$	it depends on	Count NAFI=	Oneration 5.3	n i ester
density	and may emerge; predict potential number		Stage Of	I TVNI	TOT	
		٠,	testing.	Count	report Qualifica-	. 9
	come across failures growin estimation model.  or faults in the	n. INFET total fluither of predicted fateful failures	rinally, une smaller is the	Size	Problem 5.4	oŭ
	future?	NAFI= total number of actually detected	better.	=X		_
		failures		Count		
		SIZE= product size		Size	Validation	_
					6.3	
					Quality Assurance	4
	NOTE: I. When total number of actually dete	NOTE: I.When total number of actually detected failures becomes larger than total number	NOTE:			
	of predicted latent failures, it is recommended to predict again and estimate more larger number.	to predict again and estimate more larger	<ol><li>It may be helpful t failures.</li></ol>	o predict upp	<ol><li>It may be helpful to predict upper and lower number of latent failures.</li></ol>	latent
	2. It is recommended to use several reliability	2. It is recommended to use several reliability growth estimation models and chose the most	4. It is necessary to c	onvert this v	4. It is necessary to convert this value $(X)$ to the $<0.1>$ interval if	terval if
	likelihood fit one and repeat prediction with mo	nonitoring detected failures.	making summarization of characteristics	on of charact	eristics	
Estimated	How many problems Count the number of	Estimated residuary latent fault density	0<=X Ratio	NPFT=	Test report 5.3	Developer
latent fault	still potentially exist detected faults and prec	detected faults and predict X= {ABS(NPFU - NAFU )} / SIZE	It depends on	Count		u
density	and may arise potential number by		stage of	NAFT=	Operation 5.3	Tester
£		ABS()= Absolute Value	testing.	Count	report Qualifica-	ı.
	growth estimation model	els. NPFU= total number of predicted latent	Finally, the	SIZE=	tion testing SQA	ng SQA
	,	faults in a software product	smaller is the	Size	Problem 5.4	
		NAFU= total number of actually detected	better.	X= Cou	X= Count/ report Operatio	6
		faults		Count	6.5	
		SIZE= product size			Validation	Ę,
		•			6.3	
					Quality	
					Assurance	ę,
	NOTE: 1. When total number of actually deterpredicted latent faults, it is recommended to predicted latent faults.	NOTE: 1. When total number of actually detected faults becomes larger than total number of predicted latent faults, it is recommended to predict again and estimate more larger number.	1	to predict up	NOTE: 3. It may be helpful to predict upper and lower number of latent faults.	f latent faults
	2. It is recommended to use several reliability growth estimation model likelihood fit one and renest prediction with monitoring detected faults	2. It is recommended to use several reliability growth estimation models and chose the most likelihood fit one and reneat prediction with monitoring detected faults.	4. It is necessary to convert this value (X) making summarization of characteristics	convert this v	4. It is necessary to convert this value $(X)$ to the $<0.1>$ interval if making summarization of characteristics	ıterval if
	ilkeiinood iii one and repeat prediction with i	noninornig detected radits.	maning summanical	יסוויס ויסווסו	ici istics	

<b>Table 7.2.</b>	.1 Maturity me	Table 7.2.1 Maturity metrics (continued)						
Metric Name	Purpose of the	Method of application	Measurement, formula and	Interpretation N	etric Measu	Interpretation Metric Measure Input to ISO/IEC	SO/IEC	Beneficerie
	metrics		data element computations	of measured scale	ale type	measurem 12207	2207	s
				value ty	type	ent S	SLCP	
						æ	Reference	
Failure density	Failure density How many problems Count the number of	Count the number of	a) Failure density X= NFAI / SIZE	0<=X.Y R	Ratio NFAI=	Test report	5.3	Developer
•	were detected?	detected failures (or	b) Fault density Y= NFAU / SIZE	It depends on	Count		Integration	
(Fault density)		faults) and calculate	•	stage of	NFAU=	Operation	5.3	Tester
		density.	NFAI= number of detected failures	testing.	Count	report	Qualifica-	
			NFAU= number of detected faults	Finally, the	SIZE=	ŧ	tion testing SQA	SQA
			SIZE= product size	smaller is the	Size	Problem	5.4	
			•	better.	X,Y=	report	Operation	
					Count	•	5.3	
					Size		Quality	
						7	Assurance	

NOTE: I. The larger is the better, in early stage of testing. On the contrary, the smaller is the NOTE: better, in later stage of testing or operation. It is recommended to monitor trend of this 3. It is necessary to convert this value (X,Y) to the <0,1> interval if measure along with time.

2. The number of detected failures (or faults) divided by the number of test cases indicates effectiveness of test cases.

# Table 7.2.1 Maturity metrics (continued) Metric Name Purpose of the Method of sandianing a

© ISO/IEC

		,					
Metric Name	Purpose of the	Method of application	Measurement, formula and	Interpretation Met	ric Measure	Interpretation Metric Measure Input to ISO/IEC Beneficerie	Beneficerie
	metrics		data element computations	of measured scale	ic type	measurem 12207	S
				value type		ent SLCP	
1						Reference	63
Failure	Evaluate the amount	Evaluate the amount Failure resolution is	a) Ratio to actually detected number	0 <= X <= 1	a)Absol NRFI=	Test renort 5.3	Maintainer
Resolution	of failure conditions	of failure conditions ensured by observation	X= NRFI / NAFI	The closer to ute.	Count	Operation Integration	
	that are resolved.	that the same type of		1.0 is the		(test) 5.3	ı
		failure never occurs again		better. More	Count	report Oualifica-	
		with the execution		failures	NPFI	•	ō
		condition being		resolved b)	Count	5.4	Į)
		acceptably identical to the	b) Ratio to estimated number		Absolut $X = Count/$		
		condition with which	Y=NRFI/NPFI	\_=\\ \-\>0	Count		•
		failure once occurred.	NPFI= total number of predicted latent	The closes to	V= Count.		
			ביי כשו וושווסכו כד הוכתוכם ומוכזוו	וויך בוספבו נס	IIIOO – I	_	
			tailures	1.0 is the	Count		
		Maintain a problem		hetter			
		resolution report					
		describing status of all the					
		failures.					
	NOTE:			MOTE.			
	1 It is recommended	to monitor trans and at	It is recommonded to monitor trans of this manner of the		;		
	1. 1. 1. 1. 1. COMMITTELLICATION		us measure along with time.	of the necessary to	Convert this va	it is necessary to convert this yallow (V) to the <0.15 interior of	+ 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1

It is recommended to monitor trend of this measure along with time.
 Total number of predicted latent failures may be estimated with one of reliability growth models adjusted by parameter data based on the past history.

5. It is necessary to convert this value (Y) to the <0.1> interval if making summarization of characteristics

3. It is recommended to monitor this estimated failure resolution ratio Y andwhen Y > 1, to investigate whether it is going well to detect and to resolve early more enough defects before delivery than estimation, or it is going worse because of more defects contained in the software than usual.

Otherwise, when Y < 1,

then it is recommended to investigate whether it is going well because of more defects contained in the software than usual, or it is going worse because of defects detected less than usual.

4. Multiple problem reports may be received for the same failure, when the software is under user beta testing or multiple sites testing.

Table 7.2.1 Maturity metrics (continued)

Metric Name Purpose of the Method of application N

,								
Metric Name	Purpose of the	Method of application	Measurement, formula and	Interpretation Metric		Measure	Input to ISO/IEC	Beneficerie
	metrics		data element computations	of measured	scale ty	type	measurem 12207	s
				value	type		ent SLCP	
!		,					Reference	٥ ن
Fault Removal		Count the number of	a) Ratio to actually detected number	0<=X<= 1	(a)Absol NCFU=		Test report 5.3	Developer
	have been corrected	detected faults and	X= NCFU / NAFU	The closer to	ute.		Integration	
	¢.	permanently corrected	NCFU= number of corrected faults	1.0 is the	Z	II	Organizati 5.3	
		faults.	NAFU= total number of actually detected	better, less	)		on Oualifica-	<u>.</u>
		Fault correction is ensured		faults remain.	Z.		abase	50
		by observation of resolved	יסי			Count		۵
		failure or by testing or	b) Ratio to estimated number	Y=>0	×	X= Count/	Validation	g
		reviewing modification.	Y= NCFU / NPFU	The closer to	(b)	Count	6.3	
		Maintain a problem	NCFU= number of corrected faults	1.0 is the	Absolut	Y= Count/	Ouality	
		resolution report	NPFU= total number of predicted latent	better less	٠	Count	Assurance	4
		describing status of all the		faults shall	;	1	Window.	Ą
		failures.		remain				
	NOTE:	ANALALA (PRINCIPAL PROPERTY OF THE REPRESENTATION OF THE PROPERTY OF THE PROPE	THE THE PERSON NAMED OF TH	NOTE				,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
	1. It is recommended	1. It is recommended to monitor trend of the	this measure along with time.	4. It is necessa	ary to conve	rt this valu	4. It is necessary to convert this value $(Y)$ to the $\leq 0.1 > \text{interval if}$	iterval if
			)	making summarization of characteristics	arization of	characteri	stire	Y
	2. Total number of p	redicted latent faults may be	2. Total number of predicted latent faults may be estimated with one of reliability growth	Q.	10 100 100		conc.	
	models adjusted by p	models adjusted by parameter data based on the past history.	past history.					
	3. It is recommended	3. It is recommended to monitor this estimated faults resolution ratio V and	aults resolution ratio V and					
	when Y > 1.							
	to investigate wheth	er it is going well to detect	to investigate whether it is going well to detect and to resolve early more enough defects					
	before delivery than	estimation, or it is going we	before delivery than estimation, or it is going worse because of more defects contained in the	6.				
	software than usual.	) )						
	Otherwise, when Y < 1							
	then it is recommend	led to investigate whether it	then it is recommended to investigate whether it is going well because of more defects					
	contained in the sort usual.	comained in the software than usual, of it is goin usual.	ng worse because of defects detected less than	c				

Metric Name	Purpose of the	Method of application	Measurement, formula and	Interpretation Metric Measure	Metric	Measure	Input to ISO/IEC	l .	Beneficerie
	metrics		data element computations	of measured scale	scale	type	measurem 12207	s 7(	
				value	type		ent SLCP	൧	
							Refe	Reference	
Mean time	How frequent are t	How frequent are the Trace failures occurred	Mean time between failures	0 <x.y< td=""><td>(a)</td><td>NAFI=</td><td>Test report 5.3</td><td>Z</td><td>Maintainer</td></x.y<>	(a)	NAFI=	Test report 5.3	Z	Maintainer
between	failures of the syste	failures of the system during observed testing or	a) $X = TOPT / NAFI$	The longer is Ratio	Ratio	Count	Integ	Integration	
failures	in operation?	operation time.	b) $Y = TSIB / NAFI$	the better.		TOPT =	Operation 5.3	D	User
(MTBF)					(p)	Time		Qualifica-	
			TOPT = operation time	The longer	Ratio	TSIB=		testing	
			TSIB = sum of time interval between	time can be		Time		5.4	
			failure occurrence and the next one	expected			Ope	Operation	
			NAFI = total number of actually detected	between		X=Time	/ testing	я. 2	
			failure (Failures occurred during observed	failures.		Count	5.4	ı	
			operation time)			Y=Time/	Ope	Operation	
						Count			
	NOTE:			NOTE:					
	<ol> <li>The following it</li> </ol>	1. The following investigation may be helpful: -	distribution of time interval between failure		or hazar	d rate calcul	2. Failure rate or hazard rate calculation may be alternatively used.	matively	used.
	occurrence to the next;	next;						•	
	- changes of mean	- changes of mean time along with interval operation time period;	ition time period;	3. It is necess:	ary to cor	ivert this va	3. It is necessary to convert this value to the <0,1> interval if making	nterval if	making
	- distribution indic	- distribution indicating which function has frequ	- distribution indicating which function has frequent failure occurrences and operation	summarization of characteristics	n of chara	acteristics			

Metric Name	Α	`					
	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation Metric of measured scale value type	Measure Input type measu	Input to ISO/IEC measurem 12207 ent SLCP Reference	Beneficerie s
Test coverage	Have enough user operation scenario test cases been executed?	Count the number of test cases from user operation view which are actually tested.	Specified operation scenario testing coverage  X= A / B  A= Number of actually performed test cases representing operation scenario during testing  B= Number of test cases to be performed  NOTE:  1. Test cases may be normalized by software size, that is: test density coverage Y= A / C, where C= Size of product to be tested.  The larger Y is the better. Size may be functional size that user can measure.	0<=X<=1 Absolut The closer is e. to 1, the better test coverage.		A= Count Test spec. 5.3 Deve B= Count Test report Qualifica- Tests X= Count report 6.5 Count report 6.5 Validation 6.3 Quality Assurance	Developer Tester g SQA
Test overcome	Is product going to pass successfully a lot of test cases?	Count the number of passed test cases which have been actually executed.	Passed test case ratio X= A / B A= Number of passed test cases during testing or operation B= Number of test cases to be performed.  NOTE: 1. It is recommended to perform heavily stress testing, for example, to use actual log data at highly peak period of operation for testing.  Therefore, it is recommended to ensure the following test cases are executed enough and passed successfully:  - User operation scenario: - Peak stress: - Overloaded data input.	0<=X<=1 The closer is Absolut to I the better e. overcome of tests.	A= Count B= Size X= Count/ Size	Test spec. 5.3 Deve Test report Qualifica- Teste Operation tion testing SQA report 6.3 Quality Assurance	Developer Fester g SQA

								The second name of the last	
Metric Name	Purpose of the	Method of application	Measurement, formula and	Interpretation	Metric	Measure	interpretation Metric Measure Input to ISO/IEC Beneficerie	EC Be	neficerie
	metrics		data element computations	of measured scale		type	measurem 12207	s	
				value	type		ent SLCP		
			The state of the s				Reference	ence	
Breakdown	How often can user	How often can user Count the number of	Breakdown avoidance ratio	0<=X<= 1		A=Count	A=Count Test report 5.3	Us	User
avoidance	avoid breakdown of	avoid breakdown of breakdowns occurrence	X = 1 - (A/B)	The closer to	Absolut	B=Count	The closer to Absolut B=Count Operation Integration Maintainer	ation Ma	aintainer
	system, even if	with respect to number of	A= Number of breakdowns	1.0 is the	ಬ	X=Count/ report	report 5.3		
	critical failures	failures.	B= Number of failures	better.		Count	Oualifica-	fica-	
	occurred?		NOTE: 1. The breakdown means executing				tion to	tion testing	
		If it is under operation,	of any user task is suspended until system is				5.4	)	
		analyse log of user	restarted up, or its control is lost until				Operation	tion	
		operation history.	system is enforced to be shut down.				•		
			2. When none or a few failures observed,						
			time between breakdown may be more						
			suitable.						

$\overline{}$	
$\boldsymbol{\sigma}$	
دتة	
n	
_	
•	
_	
0	
ິວ	
$\overline{}$	
Ś	
۳.	
۳.	
$\overline{}$	
Ų	
₽	
43	
S	
$\simeq$	
_	
ಡ	
Ľ	
40	
$\stackrel{\sim}{-}$	
0	
=	
ᇽ	
نس	
~7	

Fault tole	Fault tolerance metrics (continued)	(continued)					
Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation Metric of measured scale value type	Measure Input to ISO/IE type measurem 12207 ent SLCP Refere	3C snce	Beneficerie s
Failure avoidance	How often can user avoid critical or serious failure?	Count the number of critical and serious failure occurrence with respect to number of causes.	X=A / B  A = Number of avoided critical and serious The closer to A of failure emergence against test cases of fault 1 is the better, e. pattern  B = Number of executed test cases of fault more often pattern (almost causing failure) during avoid critical testing or serious failure.	0<=X<= 1 The closer to Absolut I is the better, e. the user can more often avoid critical or serious failure.	A= Count Test report B= Count X= Count/ Operation Count report	5.3 User Integration Maintainer 5.3 Qualifica- tion testing 5.4 Operation 6.5 Validation	ntainer
	NOTE:  1. It is recommended to categorize failt impact of faults, for example:  -Critical: entire system stops / or seriou-Serious: important functions become (workaround);  -Average: most functions are still avail alternate operation (workaround);  -Small: a few functions experience lim-None: impact does not reach end user	NOTE:  1. It is recommended to categorize failure avoidance levels wh impact of faults, for example:  -Critical: entire system stops / or serious database destruction: -Serious: important functions become out of service and no alt (workaround); -Average: most functions are still available, but limited perfor alternate operation (workaround); -Small: a few functions experience limited performance with langest does not reach end user	NOTE:  1. It is recommended to categorize failure avoidance levels which is the extent of mitigating impact of faults, for example:  -Critical: entire system stops / or serious database destruction:  -Serious: important functions become out of service and no alternative way of operating (workaround);  -Average: most functions are still available, but limited performance occur with limited or alternate operation (workaround):  -Small: a few functions experience limited performance with limited operation:  -None: impact does not reach end user	NOTE: 2. failure avoidance levels severity of consequenc ISO/IEC 15026 System 3. Fault pattern examples - out of range data deadlock Fault tree analysis techni 4. test case can include th	NOTE: 2. failure avoidance levels may be based on a risk matrix composed by severity of consequence and frequency of occurrence provided by ISO/IEC 15026 System and software integrity. 3. Fault pattern examples - out of range data - deadlock - Fault tree analysis technique may be used to detect fault patterns 4. test case can include the human incorrect operation	sk matrix compe currence provide y. etect fault patter cration	sed by d by by ls.
Incorrect operation avoidance	How often can user avoid failure, even if user operates incorrectly?	How often can user Do operational test or avoid failure, even if analyse log of user user operates operation history. incorrectly?  Count the number of critical and serious failure occurrence with respect to number of user's incorrect operation.	X=A / B A= number of avoided user's erroneous The clooperations or inputs during operation 1.0 is B= number of user's erroneous operation or more input during operation incorracte of the control of the	<= 1 loser to better. ect user tion ed	A=Count Test report 5.3 Absolut B=Count Operation Inte E. X=Count report 5.3 Count Qu tion 5.4	Test report 5.3 User Operation Integration Maintainer report 5.3 Qualifica- tion testing 5.4 Operation	intainer intainer

bility metrics	Method of applica
- 44	Purpose of the
Table 7.2	Metric Name

Table 1.7	able 1:2:3 theory elability interites	אורץ אוויינו וכי				ł
Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation Metric of measured scale value type	Measure Input to ISO/Ik type measurem 12207 ent SLCP	to ISO/IEC Beneficerie m 12207 s SLCP
Availability	Can user use software system enough every time?	Investigate recovery testing.	(a) Availability X= { To / (To + Tr) } Pc=X<=1 The larger (b) User operation available ratio Y= C / D the better, (This is from software system operation user can uview.) To = operation time Tr = time to repair C= total available turns of user's successful software use when user attempt to use D= total number of turns of user's attempt to use the software during observation time. The largen to use the software during observation time. the better. This is from user callable function	0<=X<=1 (a),(b) The larger is Absolut the better, the e. user can use the software for more time.  0<=Y<=1 The larger is the better.	To = Time Test report Tr = Time Operation X= report Time/ Time/ Time C= Count D= Count Y= Count Count	5.3 User Integration Maintaine 5.3 Qualifica- tion testing 5.4 Operation
			NOTE: It is recommended that this metric includes only the automatic recovery provided by the software and excludes the maintenance work of human.			
Mean down time	How long is usually system down?	/ Investigate recovery testing.	Mean time: X= T / B	0 <x be="" better.="" down="" for="" is="" ratio="" shorter="" smaller="" system="" td="" the="" time.<="" will=""><td>T=Time Test report 5.3 B=Count Operation Into X=Time/C report 5.3 ount Qu tion tion 5.4 Op Op</td><td>Test report 5.3 User Operation Integration Maintainer report 5.3 Qualification testing 5.4 Operation 6.5</td></x>	T=Time Test report 5.3 B=Count Operation Into X=Time/C report 5.3 ount Qu tion tion 5.4 Op Op	Test report 5.3 User Operation Integration Maintainer report 5.3 Qualification testing 5.4 Operation 6.5
	NOTE: 1. It is recommende by the software.	TE: It is recommended that this recovery metric includes only the a by the software and excludes the maintenance work of human	NOTE:  I. It is recommended that this recovery metric includes only the automatic recovery provided by the software and excludes the maintenance work of human.		NOTE:  2. It is necessary to convert this value (X) to the <0.1> interval if making summarization of characteristics	<0.1> interval if

Metric Name	Purpose of the	Method of application	Measurement, formula and	Interpretation		Measure	Input to ISO/IEC		Beneficerie
	merrics		data element computations	of measured value	scale type	type	measurem 12207 ent SLCP Refere	2	sa
Recovery	How long it will take	How long it will take Let system to be down	Recovery time	XX 0	Ratio	T=Time	Test report 5.3		User
	to recover if system		Mean time $=X= Sum(T) / B$	The smaller is		B=Count	Operation Integration Maintainer	egration	Maintaine
	went down?	time with actual time to		the better.		X=Time/	report 5.3		
		re-establish its adequate				Count	om')	Qualifica-	
		level of performance with enough less lost of data	system at each opportunity R= Number of times which observed				tion	tion testing	
			software system downs entered into				ť ő	eration	
			recovery				6.5	6.5	
	NOTE:			3. It is recom	mended to	distinguis	Validation  3. It is recommended to distinguish the grades of recovery difficulty.	Validation of recovery (	lifficulty.
	I. It is recommended time for many cases.	ed to measure maximal time o	<ol> <li>It is recommended to measure maximal time of the worst case or distribution of recovery time for many cases.</li> </ol>	for example, recovery of destroyed recovery of destroyed transaction.	ecovery o	f destroyed ansaction.	for example, recovery of destroyed data base is more difficult than recovery of destroyed transaction.	re difficu	lt than
	2. It is recommende by the software and	2. It is recommended that this recovery metric includes only the by the software and excludes the maintenance work of human.	2. It is recommended that this recovery metric includes only the automatic recovery provided by the software and excludes the maintenance work of human.		ary to conv arization	vert this val of character	4. It is necessary to convert this value $(X)$ to the $<0,1>$ interval if making summarization of characteristics	,1> inter	val îf
Restartability	Can user restart easily?	Let system to be down, and compare estimated	X=A/B	0<=X<=1 The larger	Absolute	A=Count	report	coration	5.3 User Integration Maintainer
		time with actual time to	A= Number of restarts which met to	and closer to		X=Count/	report	To and the second	1.1441111441111
		restart system.	required time during testing or user	1.0 is better,		Count		Qualifica-	
			operation support	the user can			tion	tion testing	
			B= Total number of restarts during testing	restart easily.			5.4		
			or user operation support				Ope 5.5	Operation 6.5	
	NOTE.						Val	Validation	
	1. It is recommended to confidence of down, such as data base temporal data destruction.	ed to estimate different time t ta base destruction, lost mult uction.	1. It is recommended to estimate different time to restart to correspond to the severity level of down, such as data base destruction, lost multi transaction, lost single transaction, or temporal data destruction.						
	2. It is recommended	2. It is recommended that this recovery metric includes only the	2. It is recommended that this recovery metric includes only the automatic recovery provided						

Table 7.2.3 Recoverability metrics (continued)

Metric Name Purpose of the Method of annihization Measurement

COLO IN COLOR	December of Alex							
INCUIC MAINE	rupose of the	Method of application	Measurement, formula and	Interpretation	Interpretation Metric Measure Input to ISO/IEC	Input to		Beneficerie
	metrics		data element computations	of measured	scale type	measurem 12207		s
				value	type	cnt	SLCP	
							Reference	
Restorability	How is the product	Let the system to perform	X=A/B	(X=X<=1	Absolute A= Count Test report 5.3	nt Test report	1 5.3	User
	capable to restore in	defined cases	A= Number of cases successfully restored	The larger is	B= Cour	B= Count Operation Integration Maintainer	Integration	Maintainer
	defined cases?		B= Number of cases performed	the better, the	X= Count/ report	it/ report	5.3	
				product is	Count	•	Qualifica-	
			NOTE: It is recommended that this metric	more capable			tion testing	
			includes only the automatic recovery	to restore in			5.4	
			provided by the software and excludes the	defined cases			Operation	
			maintenance work of human.				6.5	
							Validation	
Restore	How effective is the	How effective is the Let the system to perform	X=A/B	1=>X=>0	Absolute A= Count Test report 5.3	nt Test report	t 5.3	User
effectiveness	restoration process in defined cases	n defined cases	A= Number of cases successfully restored	The larger is	B= Com	B= Count Operation Integration Maintainer	Integration	Maintainer
	the product?		meeting the target restore time	the better, the	X= Count/ report	it/ report	5.3	
			B= Number of cases performed	restoration	Count	•	Qualifica	
				process in			tion testing	
			NOTE: It is recommended that this metric	product is			5.4	
			includes only the automatic recovery	more			Operation	
			provided by the software and excludes the	effective.			6.5	
			maintenance work of human.				Validation	

Table 7.2.4 Compliance metrics (compliance for reliability)

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation Metric Measure Input to ISO/IEC of measured scale type measure 12207 value type ment SLCP Reference	Measure type	Input to ISO/IEC measure 12207 ment SLCP Reference	Beneficeri es
Reliability compliance	How completely Previously specify does the software required complianc adhere the standards, based on standards.	Previously specify Ratio of satis required compliance items to reliability based on standards.	Ratio of satisfied compliance items relating 0<= X <=1 to reliability  The closer of the better the	0<= X <=1 A= Count Specifica- 5.3  The closer to Absolut B= Count tion of Qualifica- 1 is the better. e. X= Count/ complian- tion testing	A= Count B= Count X= Count	A= Count Specifica- 5.3 Supp B= Count tion of Qualifica- X= Count/ complian- tion testing User	Supplier g User
	conventions or regulations relating	conventions or regulations $X = 1 - (A/B)$ relating to reliability	(X = 1 - (A/B))		Count	ce and related 6.5	
	to reliability?	which to be adhered by	A= Number of failed compliance items			standards, Validation	
		software.	during testing			conven-	
						tions or	
		Design test cases in	B= Number of total compliance items			regulation	
		accordance with				s	
		compliance items.	NOTE:				
			It may be useful to collect several measured			Test	
		Conduct functional testing	Conduct functional testing values along time, to analyse the trend of			specifica-	
		for these test cases.	increasing satisfied compliance items and			tion and	
			to determine whether they are fully satisfied			report	
			or not.				

# 7.3 Usability Metrics

External usability metrics should be able to measure software attributes related to its operation, with regard to easiness of use and adaptation of new operators. Although the evaluation focus is the software, it's expected to be difficult to distinguish between actual software attributes, and the host system influence over usability characteristics. This does not invalidate the measurements, since the evaluated software is ran under explicitly specified conditions, which encompass the required system."

An external usability metric should be able to measure such attributes as user behavior, operator or system including software, to represent the extent of ease of use.

## NOTE: User tests

Most external usability metrics are based on testing users. A sample of users who are representative of an identified user group should carry out the test without any hints or external assistance. These measures may vary widely among different individuals. For reliable results a sample of at least eight users is necessary, although useful information can be obtained from smaller groups.

It should be possible for the measures taken to be used to establish acceptance criteria or to make comparisons between products. This means that the measures should be counting items of known value. Results should report the mean value and the standard error of the mean.

# 7.3.1 Understandability metrics

Understandability metrics should be capable of evaluating the behavior of users without previous knowledge on software operation and measure their difficulty on understanding software functions, operation and concepts. On doing this, it may be considered entities such as documentation (in all available formats, as on-line or printed), software interface and vocabulary. The condition of observing users without any background on the evaluated software is not always neccesarry, but is expected to be helpful on measuring the capability of the software to communicate with new users; at that point, usability problems are more related to software problems then to user psychological aspects, as stress on trying to operate the system.

An external understandability metric should be able to measure such attribute as users' effort by measuring the behavior of user 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.

# 7.3.2 Learnability metrics

Learnability metrics should be capable of evaluating or drawing the user curve of performance on software operation, from a start point of no knowledge about the evaluated software.

© ISO/IEC TR 9126

The main measurable factors are time and success ratio on completing tasks by using software operations. When using prepared or chosen test tasks, the evaluator must be aware of the effects of inducing user behavior, like ordering those tasks in an increasing difficulty list.

Learnability measuring is strongly related to understandability; in the evaluation results analysis, and during the evaluation process, understandability measurements probably can be indicators of the learnability potential of the evaluated software."

An external learnability metric should be able to measure such attribute as the behavior of user who is learning how to use the software. Measurement may include measures derived from interviewing to user.

# 7.3.3 Operability metrics

An external operability metric should be able to measure such attribute as user's human behavior during operational testing, usability testing or user operation.

The followings should be remarked:

- a) Operation testing should be performed and operation of users should be observed and analyzed;
- b) The software or functions which are specified to be used infrequently or intermittently could be excluded (depending of the nature of the needs). Examples are emergency call function or restart function:
- c) Frequency distribution for each of functions may be helpful to understand user's operational profile.

NOTE:1. When user's operation profile is comprehended, it is useful to apply usability testing to frequently used function with priority derived from profile, and to get close operability measurement results to user's actual evaluation.

2.Users may be observed who are participants of operation testing.

# 7.3.4 Attractiveness metrics

An external attractiveness metric should be able to measure such attribute as user's human behavior expressing the extent of which user likes the software during operational testing, usability testing or user operation.

## 7.3.5 Compliance metrics

An external usability compliance metric should be able to measure an attribute such as the number of functions with, or occurrences of compliance problems, which is of the software product to failing to adhere to standards, conventions, style guides or regulations relating to usability which are required to be adhered.

Table 7.3.1 Understandability metrics
Metric Name Purpose of the Method of anticonia.

Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Metric scale type	Measure type	Input to ISO/IE measurem 12207 ent SLCP	to ISO/IEC n 12207 SLCP	Beneficerie s
Completeness of description	What proportion of functions (or types of function) are understood after reading the product description?	User test	X = A/B A = Number of functions (or types of functions) understood B = Total of number of functions (or types of functions)	0<=X<= 1 The closer to 1.0 is the better.	Absolut e.	A=Count B=Count X=Count/ Count	Operation (test) report	5.3 User Qualifica- tion testing Maintainer 5.4	User Maintainer
Demonstration Availability	i i	Observe user behavior who is trying to see demonstration. Observation may employ human cognitive action monitoring approach with video camera.	User's demonstration success ratio $X = A / B$ A= Number of turns which user successfully see demonstration when user attempts to see demonstration. B= Number of turns which user attempts to see demonstration during observation period.	0<=X<= 1 The closer to 1.0 is the better.	Absolut e.	Absolut A=Count e. B=Count X=Count/ Count	Operation (test) report User monitoring record (video tape	Operation 5.3 User Qualifica- tion testing Maintainer 5.4 Operation	User Maintainer
Evident functions	What proportion of functions (or types of function) can be identified by the user based upon start up conditions?	User test f	X = A / B A = Number of functions (or types of functions) identified by the user B = Total of number of actual functions (or types of functions)	0<=X<= 1 The closer to 1.0 is the better.	Absolut e.	Absolut A=Count e. B=Count X=Count Count	Operation (test)	5.3 User Qualifica- tion testing Maintainer 5.4 Operation	User Maintainer
Function understand- ability	What proportion of interface functions are understandable?	User test	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 1.0, the better.	Absolut	A=count B=count X=Count/ Count	Operation (test) report	5.3 User Qualifica- tion testing Maintainer 5.4	User Maintainer
Understandabl Input and Output	Understandable Is easy for user to Input and understand what is Output required as input data and what is provided as output by software system?	Is easy for user to Observe user behavior understand what is who is trying to required as input data understand input and and what is provided output of the software. as output by software system?	Understandable input and output data items ratio X=A/B A= Number of input and output data items which user success to understand during enough short time within a few trial use B= Number of input and output data items with which user attempts to understand during observation period	0<=X<= 1 The closer to 1.0 is the better.	Absolut e.	A=Count B=Count X=Count Count	Operation (test) report User monitorin g record	6.5 Validation 5.3 Qualification testing 5.4 Operation	User Maintainer

	ý	2
•	ř	
	Ĺ	
	d	
	E	
	_	•
,	Ľ	,
į	_	
٠	7	
•	700	
	È	
	ĭ	
	ç	į
	٠	i
,		
•	`	ł
•	4	
t	_	۰
٠	_	
,	4	
,	7	2
r	Ç	1
	<u>م</u>	1

Table /	LADIC 1.3.4 L'CALHABITILY MICHINS	אוורנו ורט					
Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation Metric of measured scale value type	Measure type	Input to ISO/IEC measurem 12207 ent SLCP Reference	Beneficerie s
Ease of function learning	How long does the user take to learn to use a function?	User test	Time required to learn function  T= Mean time taken to learn to use a function correctly  NOTE: This metric is generally used as one of experienced and justified.	0 <f better.<="" is="" ratio="" shorter="" td="" the=""><td>T=Time</td><td>Operation 6.5 (test) Validation report 5.3 Qualifica- User tion testing monitorin 5.4 g record Operation</td><td>User Maintainer g</td></f>	T=Time	Operation 6.5 (test) Validation report 5.3 Qualifica- User tion testing monitorin 5.4 g record Operation	User Maintainer g
Ease of performing task learning	How long does the Observe user take to learn from the how to perform the till becaspecified task smoothly?  NOTE:1. It is recommended to Such user's operating time mafirst use as the fair proportion.	Observe user behavior from they started to learn till became to operate smoothly.  mended to determine an ext g time may be the threshold roportion.	How long does the Observe user behavior Time required to learn operation to perform 0<1 between the learn task to perform the till became to operate to learn task the specified task smoothly.  The Sum of user operation time until user achieved to perform the specified task within a short time.  NOTE:1. It is recommended to determine an expected user's operating time as a short time. NO Such user's operating time may be the threshold, for example, which is 70% of time at the farst use as the fair proportion.	0 <t 2.="" better.="" effort="" is="" may<="" note:="" ratio="" shorter="" td="" the=""><td>T=Time</td><td>OcT Ratio T=Time Operation 6.5 User The shorter is report Asimatines the better. 5.3 User Qualifica- monitorin tion testing g record 5.4 Operation NOTE: 2. Effort may alternatively represent time by person-hour unit.</td><td>User Maintainer g</td></t>	T=Time	OcT Ratio T=Time Operation 6.5 User The shorter is report Asimatines the better. 5.3 User Qualifica- monitorin tion testing g record 5.4 Operation NOTE: 2. Effort may alternatively represent time by person-hour unit.	User Maintainer g
Ease of use of help system	Can the user find online help?	User test	X = A/B A = Number of tasks for which correct online help is located B = Total of number of tasks tested	0<=X Absolor The closer to e 1.0 is the better.	Absolut A= Count Operation e., B= Count (test) report X= Count Count monitorin g record	l i	6.5 User Validation 5.3 Human Qualifica- interface tion testing designer 5.4 Operation

Metric Name Purpose of the Method of application data element computations of measurem in the parties and the page of measurem 1207 s value and the page of measurement of transitions of the closer to a page of measurement of mea	Table 7.3.2 Learnability metrics (contin	z real nadinty							
triveness of What proportion of User test X=A/B  system completed correctly completed after accessing online help systems of What proportion of User test X=A/B  mentation correctly after accessing online help systems?  functions can be used after accessing online help better.  Systems of What proportion of User test X=A/B  mentation correctly after accessing online help better.  Systems?  A=Number of functions that can be used functions that can be used for all the correctly after accessing online help better.  Systems?  Incidions can be used A=Number of functions that can be used for all the correctly after accessing online help better.  Systems?  Incidions can be used A=Number of functions that can be used for all the correctly after a single help systems?  Incidions can be used A=Number of functions that can be used for all the correctly after a single help systems?  Incidion overed by unorial ratio better after a single dependent and justified merics rather access that the closer to a B=Count (test) validation account they up to learn by X=A/B  NOTE: I (Complementary measurement formula)  NOTE: I (Complementary measurement f	Metric Name	Purpose of the	Method of application	Measurement, formula and	Interpretation	l	Input	SO/IEC	Beneficerie
ethoress of What proportion of User test X=A/B		metrics		data element computations			measurem	12207	s
triveness of What proportion of User test  asks can be completed after accessfully completed after accessfully after using the help system.  completed after accessfully after accessfully after using the help systems?  A=Number of tasks tested better.  correctly after a per accessfully and a proportion of User test and of mumber of tasks tested better.  A=Number of functions and better.  A=Number of tasks tested better.  A=Number of functions that can be used a particular statement and justified metrics are tasks tested better.  A=Number of tasks tested better.  B=Number of tasks tested better.  A=Number of tasks tested better.  A=Number of tasks tasks tested better.  B=Number of tasks task tasks to use demonstration.  A=Number of tasks task tasks to use demonstration.  B=Number of tasks task tasks to use demonstration.  A=Number of tasks task tasks to use demonstration.  A=Number of tasks task tasks to use tasks ta						ypc	ent	SLCP	
system completed correctly completed for tasks tested better.  System completed correctly completed for tasks tested on the closer to completed services and the completed correctly completed for tasks tested after using the help systems?  A = Number of tasks tested better.  Systems  A = Number of tasks tested better.  Systems  A = Number of tasks tested better.  Systems  Systems  Systems  A = Number of tasks tested better.  Systems  Systems  A = Number of tasks tested better.  Systems  Systems  A = Number of tasks tested better.  Systems  A = Number of functions that can be used monitoring that can be used line title correctly after a correctly after a correctly after a correctly after a systems  A = Number of functions that can be used line title closer to e B = Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e B = Count Operation 6.5 and the closer to e Count								Reference	
system casks can be completed ornerly completed after accessing online help system state the help systems?  The closer to be used mining the help systems?  The closer to count the help systems?  The closer to count the help systems of the help sy	Effectiveness of	<ul> <li>What proportion of</li> </ul>	User test	X= A / B	X=>0	A=	nt Operation	6.5	User
after using the help systems?  after using the systems?  after using the help systems?  after using the help systems?  after using the help systems?  after using the systems?  after using the systems?  after using the systems?  because the systems?  after using success ratio  because the systems?  after using success ratio  because the systems after the systems and systems and systems attempts to use demonstration.  because the systems after the systems and systems and systems and systems and systems and systems and systems attempts to use unortal functions are used well when user attempts to use. implementation.  because the systems of which user attempts to use unortal functions are used well when user attempts to use. implementation.  because the systems are used well when user attempts to use. implementation.  after after after after a system and systems are used well when user attempts to use unortal functions are used well and governation period.	help system	tasks can be		A= Number of tasks successfully	The closer to		nt (test)	Validation	
after using the help systems?    A		completed correctly		completed after accessing online help	1.0 is the		report	5.3	Human
count User tion testing monitorin 5.4  mentation  corrections can be used functions that can be used The closer to e B= Count (test) Validation  mentation  corrections can be used functions that can be used The closer to e B= Count (test) Validation  mentation  corrections can be used the closer to e B= Count (test) Validation  mentation  reading the commentation or of experienced and justified metrics rather  documentation or of experienced and justified metrics rather  than the others.  NOTE: This metric is generally used as one count User tion testing than the others.  Infiness  utorial? when they up to learn by X= A/B  when they up to learn by the test of turnial functions during observation  period.  D= Number of functions of which user attempts to use uponing observation  period.  D= Number of functions of which user attempts to use unplementation.  D= Number of functions of which user attempts to use unplementation.  D= Number of functions of which user attempts to use unplementation.  D= Number of functions of which user attempts to use unplementation.		after using the help		B = Total of number of tasks tested	better.	X= Cour		Qualifica-	interface
tinetions can be used  functions and be used  functions can be used  functions can be used  functions browled  functions provided  functions divided metrics rather  functions divided metrics rather greater  functions divided metrics functions divided metrics functions divided metrics  functions divided metrics functions divided metrics functions divided metrics  functions divided metrics functions divided metrics  function functions divided metr		systems?				Count	User	tion testing	designer
the functions of What proportion of User test    X = A / B							monitorin	5.4	1
training help systems?  The closer to councetly after reading the concentration correctly after reading the concentration or cerebility and the others.  The closer to counced by after reading the commentation or correctly after a control of functions of the closer to count of functions of which user attempts to use.    A = Number of functions of functions of functions of functions of functions of functions of which user attempts to use.   A = Number of functions of functions of functions of which user attempts to use.   A = Number of functions of functions of which user attempts to use.   A = Number of functions of functions of which user attempts to use.   A = Number of functions of which user attempts to use.   A = Number of functions of which user attempts to use tutorial functions of w				THE PROPERTY OF THE PROPERTY O			g record	Operation	
functions can be used the correctly after	Effectiveness of	What proportion of	User test	X = A/B		=Absolut A= Cou	int Operation	ı	User
B = Total of number of functions provided 1.0 is the report 5.3  B = Total of number of functions provided 1.0 is the report 5.3  NOTE: This metric is generally used as one of experienced and justified metrics rather than the others.  y use Observe user behavior User's tutorial using success ratio of c=X=C = 1 = Absolut A=Count Operation 5.4  when they try to learn by X= A / B  themselves. A= Number of turns which user 1.0 is the second operation of turns which user attempts to use demonstration.  B=Number of turns which user attempts to use demonstration.  mmplementary measurement formula)  NOTE: 2 It is recommended to determine previously functions of which user attempts to use uttorial functions during observation of functions of which user attempts to use uttorial functions during observation  NOTE: 2 It is recommended to determine previously function of functions of which user attempts to use uttorial functions during observation  Total A= Count (test) (Qualification of turns which user attempts to use uttorial functions during observation of thinctions of which user attempts to use uttorial functions during observation of the count of the extent of the extent of the count of the count of turns of the count of the c	nser	functions can be used		A = Number of functions that can be used		B≡	int (test)		
reading the documentation or using help systems?  NOTE: This metric is generally used as one using help systems?  Can user easily use Observe user behavior user attempts to use tutorial functions of which tutorial functions are used well when user attempts to use tutorial functions during observation  NOTE: I (Complementary measurement formula)  Period.  NOTE: This metric is generally used as one using success ratio and oc=x<= 1	documentation	correctly after		B = Total of number of functions provided	1.0 is the		report	5.3	Human
documentation or using help systems?  Can user easily use Observe user behavior  Library themselves.  Can user easily use observe user behavior  Can user easily use observe user behavior  A=Number of turns which user attempts to use attempts to use attempts to use user demonstration.  NOTE: 1 (Complementary measurement formula)  Function covered by tutorial functions are used well when user attempts to use. Implementation.  D=Number of functions of which tutorial functions during observation  D=Number of functions of which user attempts to use attempts to use used well when user attempts to use. Implementation.  NOTE: 2 It is recommended to determine previously function beriod.  NOTE: 2 It is recommended to determine previously function beriod.  NOTE: 3 It is recommended to determine previously function beriod.  NOTE: 1 (Complementation of which tutorial functions during observation beriod.	and help	reading the			better.	X = Cour		Oualifica-	interface
using help systems?  using help systems?  using help systems?  than the others.  Can user easily use Observe user behavior User's tutorial using success ratio  tutorial?  when they try to learn by X= A / B  tutorial?  themselves.  The closer to c B=Count (test) Validation 6.5  NOTE: I (Complementary measurement formula) Function overed by tutorial ratio Y=(C / D)  C=Number of functions of which user attempts to use tutorial functions during observation  D=Number of functions of which user attempts to use tutorial functions during observation  D=Number of functions of which user attempts to use tutorial functions during observation  D=Number of functions of which user attempts to use tutorial functions during observation  D=Number of functions of which user attempts to use tutorial functions during observation  D=Number of functions of which user attempts to use tutorial functions during observation  D=Number of functions of which user attempts to use tutorial functions during observation  D=Number of functions of which user attempts to use tutorial functions during observation  D=Number of functions of which user attempts to use tutorial functions during observation  D=Number of functions of which user attempts to use tutorial functions during observation	systems	documentation or		NOTE: This metric is generally used as one		Count		tion testing	designer
Can user easily use Observe user behavior User's tutorial using success ratio  Can user easily use Observe user behavior User's tutorial using success ratio  (Can user easily use Observe user behavior User's tutorial using success ratio  (Can user easily use Observe user behavior User's tutorial using successfully user uttorial functions when they try to learn by X=A/B  A=Number of turns which user attempts to use demonstration.  B=Number of functions of which tutorial functions are used well when user attempts to use.  Can user attempts of turns which user attempts to use demonstration.  C=Number of functions of which tutorial functions are used well when user attempts to use. implementation.  D=Number of functions of which user attempts to use tutorial functions during observation period.		using help systems?		of experienced and justified metrics rather			monitorin		)
Can user easily use Observe user behavior User's tutorial using success ratio  tutorial? when they try to learn by X= A / B  tutorial? when they try to learn by X= A / B  tutorial? when they try to learn by X= A / B  themselves. A=Number of turns which user attempts to use demonstration.  B=Number of functions during observation  NOTE: 1 (Complementary measurement formula)  Function covered by tutorial functions are used well when user attempts to use. Tutorial functions during observation  NOTE: 2 It is recommended to determine previously functions of which tutorial functions during observation  NOTE: 2 It is recommended to determine previously functions are used well when user attempts to use, implementation.  D=Number of functions of which user attempts to use utorial functions during observation  D=Number of functions of which user attempts to use used well when user attempts to use, implementation.				than the others.			g record		
tutorial? when they try to learn by X= A / B  tutorial? when they try to learn by X= A / B  themselves. A= Number of turns which user the curve themselves.  A= Number of turns which user attempts to use tutorial functions during observation  NOTE: 1 (Complementary measurement formula)  Function covered by tutorial ratio Y=(C / D)  C= Number of functions of which user attempts to use tutorial functions during observation  NOTE: 2 It is recommended to determine previously functions of which user attempts to use tutorial functions during observation  NOTE: 2 It is recommended to determine previously functions of which user attempts to use tutorial functions during observation  NOTE: 2 It is recommended to determine previously function need tutorials for beginners and to examine later the extent of the period.	Tutorial	Can user easily use	Observe user behavior	User's tutorial using success ratio	0<=X<= 1	=Absolut A=Coun			User
better. Count report 5.3  Count User Qualification in testing g record 5.4  NOTE: 2 It is recommended to determine previously function need tutorials for beginners and to examine later the extent on implementation.	Readiness	tutorial?	when they try to learn by	X=A/B		B=Count			
better. Count 5.3 User Qualificanonin ton testing grecord grecord 5.4  NOTE: 2 It is recommended to determine previously function need tutorials for beginners and to examine later the extent or implementation.			themselves.	A= Number of turns which user	1.0 is the	X=Coun			Maintainer
0 45				successfully use tutorial functions when	better.	Count	ı	5.3	
O 45				user attempts to use demonstration.			User	Qualifica-	
AS				B= Number of turns which user attempts to			monitorin		
AŠ.				use tutorial functions during observation					
a ś				period.			ı	5.4	
AŠ								Operation	
ΔŠ		NOTE: 1 (Compleme Function covered by 1	entary measurement formulationial ratio $V=(C/D)$	la)	NOTE: 2 It is n	ecommended to de	etermine prev	riously functi	ons which
Destruction of functions of which used attentifies to use futorial functions during observation period.		C= Number of functi	ons of which tutorial funct	ions are used well when user attempts to use.	implementation	or oegimiers and to 1.	o examine iai	er une extent	5
		period.	ons of winch user autempts	to use tutorial functions during observation					

Table 7.3.3 Operability metrics

Conforms with	Conforms with operational user expectations	oectations							
Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation Metric of measured scale value type		Measure type	Input to ISO/IE measurem 12207 ent SLCP Refere	ISO/IEC 12207 SLCP Reference	Beneficerie s
Operational Consistency	How easily user sen his/her intention to receive his/her expecting results from software through what user see?	How easily user send Observe user behavior his/her intention to or who is operating software receive his/her expecting results from software through what user see?	a) Smooth Communicativeness  Conformity to WYSIWYG (What You See 0<=X<=1 Is What You Get)  X = 1 - (A / B)  A= Number of reports or functions which better. resulted within unacceptable conformity against user's expectation derived from what are visible at the screen  B= Number of reports or functions which are visible at the screen	0<=X<=1 The closer to 1.0 is the better.	a) Absolute	Operation (test) A= Count report B= Count User X= Count monitorin Count g record	Operation (test) report User monitorin g record	6.5 User Validation 5.3 Human Qualifica- interface tion testing designer 5.4 Operation	User Human interface designer
			b) Easy to derive operation Frequency of expected results Y = N / UOT N= Number of operations which resulted within unacceptable conformity against user's expectation derived from experience of operation UOT= user operating time (observation period)  NOTE: User's experience of operation is usually helpful to recognize several operation pattern which derives user's expectation.	0<=Y<= 1 The smaller and closer to 0.0 is the better.	Batio	UOT= time Y= Count/ Time			

Operability metrics (continued)

Controllable									
Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation Metric of measured scale value type	1	Measure type	Input to measurem ent	ISO/IEC 12207 SLCP	Beneficerie s
Error correction	Can user easily correct error on tasks?	User test	Time for Error Correction on Task T=FCC - TSC TCC = Time completing correct specified type errors of performing task TSC = Time starting correct specified type errors of performing task	0 <t The shorter is the better.</t 	Ratio	T=Time			
User operation cancelability	Can user easily recover his/her error or retry tasks?	Observe user behavior who is operating software	a) Frequency of cancel success  X= A / UOT  A= number of turns which user success to cancel their error operation  UOT= user operating time during observation period.	0<=X<= 1 The higher and closer to 1.0 is the better.	Ratio	A=Count UOT = Time X = Count / Time	Operation (test) report User monitorin g record	6.5 User Validation 5.3 Human Qualifica- interface tion testing designer 5.4 Operation	User Human interface designer
			NOTE:  1. When function is tested each by each, the ratio can be also calculated, that is the ratio of number of functions which user success to cancel their operation to all functions.	0					
	Can user easily recover his/her input?	Observe user behavior who is operating software	<ul> <li>b) User's successful input change ratio</li> <li>X = A / B</li> <li>A= Number of screens or forms where the input data were successfully modified or changed before being elaborated</li> <li>B = Number of screens or forms where user tried to modify or to change the input data during observed user operating time</li> </ul>	0<=X,Y<=1 The closer to 1.0 is the better.	e. e.	A= Count, Opera B= Count (test) X= Count/ report Count User moniti	Absolut A= Count, Operation e. B= Count (test) X= Count/ report Count User monitorin g record	6.5 User Validation 5.3 Human Qualifica- interface tion testing designer 5.4 Operation	User Human interface designer

Suitable for the task operation	task operation								
Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation Metric of measured scale value type	Metric scale type	Measure type	Input to measurem ent	ISO/IEC 12207 SLCP Reference	Beneficerie s
Default value availability	Can user easily select parameter values for his/her convenient operation?	Can user easily select Observe user behavior parameter values for who is operating software. his/her convenient Count how many times user attempts to establish or to select parameter values and fails to do them, because user can not use default values provided by the software.	Frequency of default value available  X = 1 - (A / B)  A = The number of turns which user fail to establish or to select parameter values in a short period (because user can not use default values provided by the software)  B = Total number of turns which user attempt to establish or to select parameter values  NOTE:  1) It is recommended to take accounts of operator's behavior and decide how long period is allowable to select parameter values as "short period".  2) When parameter setting function is tested by each function, the ratio of allowable function can be also calculated.  3) It is recommended to conduct functional test which covers parameter setting functions	0<=x<= 1 The closer to 1.0 is the better.	Absolute	A=Count B=Count Count	Operation (test) report User monitorin g record	6.5 User Validation 5.3 Human Qualifica- interface tion testing designer 5.4 Operation	User Human interface designer
User operating time adequacy	Is user become to deal task with very short operating time after sufficient training?	Observe user behavior who is operating software before/after sufficiently trained.	a) Ultimate Operating Time  Tul = Operating time needed to perform a specified task for ultimately short time skillful or trained user)  b) Beginner's Operating Time  The = Operating time needed for beginner to perform a specified task  c) Reminder's Operating Time  C) Reminder's Operating Time  Tre = Operating time needed for reminder to perform a specified task	0 <tul 0<tbe 0<tre 0<t The shorter is the better.</t </tre </tbe </tul 	Ratio	Tul. Tbe, Tre. If,Ts and T = Time	Operation (test) report User monitor record	6.5 User Validation 5.3 Human Qualifica- interface tion testing designer 5.4 Operation	User Human interface g designer

d) Operating time needed to complete task Time to complete a task of user's attempt T = Tf - Ts

Tf = Time completing a specified task Ts= Time starting operation for task

NOTE:

1. Beginner means that a specified task has been never done by that user or hc/she is beginner of the software.

NOTE:
2. Reminder means that task is used only few times with very long intervals and user may forget details of operation.

# Operability metrics (continued)

Self descriptive (Guidable)	e (Guidable)		THE REAL PROPERTY OF THE PROPE						
Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation Metric of measured scale value type	Metric scale type	Measure type	Input to measurem ent	ISO/IEC 12207 SLCP Reference	Beneficerie s
Guidability	Is everyone adequately guided to success his/her intended task?	Is everyone Observe user behavior adequately guided to who is operating software success his/her Count how many tasks does user attempt and fail/success.	a) Interactive guidability User's attempt task success ratio X=(A/B) A = Number of tasks of which objectives are completed successfully by user with online interactive guide B = Number of tasks attempted by user	0<=X<= 1 The closer to a) 1.0 is the Al better.	a) Absolute	A=Count Absolute X=Count Count	Operation (test) report User monitorin g record	6.5 User Validation 5.3 Human Qualifica- interface tion testing designer 5.4 Operation	User Human interface designer
			NOTE: It is recommended to investigate the followings with this metric: - Can user complete his/her intended task, even if it is the first attempt to do that? - Can beginner user complete his/her intended task?						
	Is everyone adequately guided to success his/her intended task?	Is everyone Count how many users adequately guided to fail/success, who are success his/her participants of operation intended task? testing.	b) Common user guidability Ratio of task success users  Y = (C/N) C = Number of users successfully completing task N = Number of users who attempted to do the task NOTE: Common user may be ordinary people or trained operator. Users may be observed who are participants of operation	0<=Y<= 1 b) The closer to Absolute 1.0 is the better.	Absolute	C=Count N=Count Y=Count Count			
			testing,						

D = Count UOT= Time	Z = Count / Time	
c) Ratio		
0<=Z<= 1 The smaller and closer to 0.0 is the	better.	
c) User understandable status or progress Frequency of user's fail to understand status or progress	Z = D / UOT D= number of turns which user pause for a long period or repeat fail successively at the same operation, because of user not comprehending status or progress of executing task UOT= user operating time (observation period)	NOTE:  1. It is recommended to take accounts of operator's behavior and decide how long period is not allowable to pause as "long period".  2. When function is tested each by each, the ratio of allowable function can be also calculated.
Observe user's fail or hesitation during operation.		

Self descriptive (Guidable)	Self descriptive (Guidable)	ııııaca)							
Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation Metric of measured scale value type	Metric scale type	Measure type	Input to ISO/IE measurem 12207 ent SLCP Refere	ISO/IEC 12207 SLCP Reference	Beneficerie s
Message readiness	Can user easily understand messages from software system? Is there any message which brought delay for user to understand and to start next action? Can user easily memorize important message?	Can user easily Observe user behavior understand messages who is operating software from software system?  Is there any message which brought delay for user to understand and to start next action?  Can user easily memorize important message?	Comprehensive messages Frequency of user's comprehending message X = A / UOT A = number of turns which user pause for a long period or repeat fail successively at the same operation, because of user not comprehending messages UOT = user operating time (observation period)	0<=X<= 1 The smaller and closer to 0.0 is the better.	Ratio	A=Count UOT = Time X = Count / Time	A=Count Operation 6.5  UOT = (test) Val Time report 5.3  X = Count  / Time User tion monitorin 5.4  g record Ope	6.5 User Validation 5.3 Human Qualifica- interface tion testing designer 5.4 Operation	User Human interface designer
	NOTE:  1. The extent of easiness of m brought delay for us Therefore, it is recommended period is not allowable to paus 3. It is recommended to invest comprehending messages.  1) Attentiveness: Attentivenes messages presenting such the warning for careful operation.	NOTE:  1. The extent of easiness of message comprehending is brought delay for user to understand Therefore, it is recommended to take accounts of opperiod is not allowable to pause as "long period".  3. It is recommended to investigate the followings as comprehending messages.  1) Attentiveness: Attentiveness implies that user suc messages presenting such the next user action guidanc warning for careful operation.	ing is represented by how long that n and and to start next of operator's behavior and decide ho s as one of the problems of user's r successfully recognize attentive im idance, name of data items to be look	<ul> <li>2) Memorability: Memorability implies that user memorize important ressage messages presenting such the next user action guidance, name of data action. items to be looked, and warning for careful operation.</li> <li>w long - Can user easily memorize important message?</li> <li>- Is it helpful for user to keep user's remembrance?</li> <li>- Is it required for user to remember only few and not so much?</li> <li>3. When message is tested each by each, the ratio of comprehensive messages to the total can be also calculated.</li> <li>portant ed, when several users are observed who are participants of operation testing, the ratio can be calculated, which is ratio of users who comprehended messages to all users.</li> </ul>	ty: Mementus sucked, and lily memon or user to for user to age is test e total car all users at io can be e messages.	orability im the next to warning for rize importa keep user's o remember ed each by on the also call re observed calculated.	plies that us user action ger action ger action gent message is remembrant only few a sach, the rail culated.  who are pain which is ratiful.	ser memoriz guidance, na cration. ? nce? and not so m tio of compt rticipants of io of users'	e important me of data uch? ehensive operation who

- Can user never fail to watch when user is encountering attentive important messages? - Can user avoid to mistake operation, because of user's recognizing attentive important

messages?

# Operability metrics (continued)

Self descriptive (Guidable)	(Guidable)								
Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation Metric Measure Input to ISO/IEC Beneficerie of measured scale type measurem 12207 s value type ent SLCP Reference	Metric l scale t type	Measure type	Input to measurem ent	ISO/IEC 12207 SLCP Reference	Beneficerie s
Self- explanatory error messages	Self-  Explanatory  Crror conditions does  Error messages the user propose the  Correct recovery  action?	f User test s	X=A/B A=Number of error conditions for which the user proposes the correct recovery action B=Number of error conditions tested NOTE: This metric is generally used as one of experienced and justified.	0 <= X <= 1 Absolute X=count/c Operation 6.5  The closer to ount (test) Validat 1, the better. A=count report 5.3  B=count Oualifit User tion test monitorin 5.4 e g record Operati	Absolute	X=count/c Opera ount (test) A=count report B=count User monit	Operation (test) report User monitorin g record	Operation 6.5 User test) Validation eport 5.3 Human Qualifica- interface User tion testing designer nonitorin 5.4 grecord Operation	5.5 User Validation 5.3 Human Qualifica- interface ion testing designer 5.4 Decration

Operabili Operational er	Operability metrics (continued)	ntinued) error free)							
Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation of measured value	Metric scale type	Measure type	Input to measurem ent	ISO/IEC 12207 SLCP	Beneficerie s
Operational error recoverability	Can user easily recover his/her wors situation?	Can user easily Observe user behavior recover his/her worse who is operating software situation?	recoverable situation frequency  X = 1 - (A / B)  A= Number of unsuccessfully recovered situation (after a user error or change) and user was not informed about this risk by the system B= Number of user errors or change	0 <x The less is the better.</x 	Absolute	A= Count, UOT= Time X= Count/ Time	A= Count, Operation UOT= (test) Time report X= Count/ Time User monitorin	1	User Human interface designer
Time Between Human Error Operations	Can user operate the software long enough without human error?	Can user operate the Observe user behavior software long enough who is operating software without human error?	1	0 <x The higher is the better.</x 	Ratio	T = Time Opera N = Count (test) X = report Time / Count User monit g recc	T = Time Operation  N = Count (test)  X = report  Time /  Count User  monitorin  g record	Operation 6.5 User Validation 5.3 Human Qualifica- interface tion testing designer 5.4 Operation	User Human interface designer
	NOTE: 1. Human error opera a) Simple human error b) Intentional error: c) Operation hesitation NOTE: 2. It is seemed that of It depends on function requested to take accolumn. to 3 min.	ation may be detected by couror. The number of turns which un pause: The number of turn peration pause implies user jon, operation procedure, applicant into them and determination procedure.	NOTE:  1. Human error operation may be detected by counting below user's behavior:  2. Simple human error: The number of turns which user just simply mistakes to input operation;  3. Intentional error: The number of turns which user pause for a long period with hesitation during observation period.  3. Operation hesitation pause: The number of turns which user pause for a long period with hesitation during observation period.  3. It is seemed that operation pause implies user is puzzled and hesitate operation.  4. It is seemed that operation procedure, application domain and user, whether it is long period or not for user to pause operation. Therefore, evaluator is requested to take account into them and determine threshold time. For an interactive operation, it is generally supposed that "long period" threshold range is from Imin. to 3 min.	on; rration with mis ssitation during reriod or not for rit is generally	is-understa observation user to pa	nding durin on period. use operatic	g observati on. Thereft oeriod" thre	on period: ore, evaluato	or is

N. C	Operational error tolerant (Human error free)	error tree)							
Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation Metric of measured scale	Metric scale	Measure	Input to ISO/IE	ISO/IEC	Beneficerie
				value	type	<u>.</u>	ent	SLCP Reference	,
Undoability	How frequently does. User test	User test	a) Input undoability	0<=X<=1	Absolute	A= Count,	Absolute A= Count, Operation 6.5	6.5	User
	correct input errors?		X= A / B	The closer to		B= Count (test)		Validation	;
			A= Number of input errors which the user	better.		A= Count Count		5.3 Oualifica- interface	Human interface
			successfully corrects					tion testing designer	designer
			b—Indinger of attempts to correct input				monitorin	5.4	
							g record	Operation	
			NOTE: This metric is generally used as one of experienced and instifted						
	How frequently does User test	s User test	b) Error undoability	0 <= X <= 1 Absolute X=count/c Operation 65	Absolute	X=count/c	Operation	6.5	I spr
	the user correctly		: :	The closer to	•	ount	(test)	Validation	
	undo errors?		X=A/B	<ol> <li>the better.</li> </ol>		unt		5.3	Human
			A= Number of error conditions which the					Oualifica.	interface
			user successfully corrects				User	tion testing designer	designer
			B= 1 otal number of error conditions tested				monitorin	5.4	ı
							g record	Operation	

Operabili	Operability metrics (continued)	ntinued)						
Suitable for individualisation	dividualisation							
Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation Metric of measured scale value type	Metric Measure scale type type	input to ISO/IE measurem 12207 ent SLCP	ISO/IEC 12207 SLCP	Beneficerie s
Customisabilit	Customisability Can user easily customize operation procedures for his/her convenience? Can user, who instructs end users, easily set customized operation procedure templates for preventing his/her error? What proportion of functions can be customised?	User test	X= A / B A= Number of functions successfully customised B= Number of attempts to customise	0 <= X <= 1 A The closer to 1, the better.	Absolute X=count/c ount A=count B=count	t, t t	NOTE OF THE OFFICE OFFIC	
	NOTE:  1. Ratio of user's fail to customize may be $Y = 1 - (C/D)$ $C = Number of turns which a user fail to to D = Total number of turns which a user at 0 \le Y \le 1. The closer to 1.0 is the better.$	NOTE:  1. Ratio of user's fail to customize may be measured.  Y = 1 - (C / D)  C = Number of turns which a user fail to customize operation  D = Total number of turns which a user attempted to customize 0<= Y<= 1, The closer to 1.0 is the better.	NOTE:  1. Ratio of user's fail to customize may be measured.  Y = 1 - (C / D)  C = Number of turns which a user fail to customize operation  D = Total number of turns which a user attempted to customize operation for his/her convenience.  0<=Y<= 1, The closer to 1.0 is the better.	ience.				

- 2. It is recommended to regard the followings as variations of customise operation:
  chose alternative operation, such as to use menu selection or command input:
  combine user's operation procedure, such as to record and edit operation procedure:
  set constrained template operation, such as to program procedures or to make template for input guidance.
- 3. This metric is generally used as one of experienced and justified.

table for in	Suitable for individualisation							
Metric Name	Purpose of the metrics	Method of application	Measurement, formula and data element computations	Interpretation Metric Measure Input to ISO/IEC of measured scale type measurem 12207 value type ent SLCP	Metric Meas scale type type	ure Input to ISO/IE measurem 12207 ent SLCP	ISO/IEC n 12207 SLCP	Beneficerie s
Operation procedure reduction	Can user easily Count user's strangue operation specified operation procedures for compare them by his/her convenience? before and after customizing operations.	Count user's strokes for specified operation and compare them between before and after customizing operation.	Operation Procedure Reduction Ratio  X = 1 - (A / B)  A = Number of reduced operation procedures after customizing operation B = Number of operation procedures before customizing operation.  NOTE: 1. It is recommended to take samples for each different user task and to distinguish operator who is skillful user or beginner.  2. Number of operation procedures may be represented by counting operation strokes such click, drug, key touch, screen touch etc.	0<=X<1 Ab The closer to e. 1.0 is the better.	bsolut A=Cor B=Cor X=Cor Count	Reference Absolut A=Count Operation 6.5 e. B=Count (test) Valis X=Count report 5.3 Count User tion monitorin 5.4 g record Oper	Reference 1 6.5 User Validation 5.3 Human Qualifica- interface tion testing designer 1 5.4 Operation	User Human interface designer