INTERNATIONAL STANDARD

ISO/IEC 25051

First edition
2006-04-01

Corrected version
2006-06-15

Software engineering — Software product Quality Requirements and Evaluation (SQuaRE) — Requirements for quality of Commercial Off-The-Shelf (COTS) software product and instructions for testing

Ingénierie du logiciel — Exigences de qualité pour le logiciel et évaluation (SQuaRE) — Exigences de qualité pour les progiciels et instructions d'essai

Reference number
ISO/IEC 25051:2006(E)

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

Published in Switzerland

Contents

Page

Foreword iv

Introduction v

1 Scope 1

2 Conformance 2

3 Normative references 3

4 Terms and definitions 3

5 Requirements for COTS software product 4
5.1 Requirements for product description 4
5.2 Requirements for user documentation 8
5.3 Quality requirements for software 9

6 Requirements for test documentation 12
6.1 General Requirements 12
6.2 Requirements for the test plan 12
6.3 Requirements for the testing description 14
6.4 Requirements for the test results 15

7 Instructions for conformity evaluation 16
7.1 General Principles 16
7.2 Conformity evaluation pre-requisites 16
7.3 Conformity evaluation activities 16
7.4 Third-party conformity evaluation process 17
7.5 Conformity evaluation report 17
7.6 Follow up conformity evaluation 18

Annex A (informative) **Definitions from others standards** 19

Annex B (informative) **Guidance for application of cots software product in business or safety critical applications** 22

Annex C (informative) **How to use ISO/IEC 25051** 25

Bibliography 26

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

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

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

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

ISO/IEC 25051 was prepared by Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Software and system engineering*. This first edition cancels and replaces the first edition of ISO/IEC 12119:1994.

The IEEE Computer Society participated as a liaison organization in the revision of this International Standard. Some suggestions from the IEEE adoption of the 1994 edition of this International Standard have been incorporated in this revision.

This corrected version of ISO/IEC 25051:2006 incorporates the following corrections:

— English and French titles corrected.

Introduction

Commercial Off-The-Shelf (COTS) software products are used in an increasingly wide variety of application areas and their correct operation is often vital for business, safety or personal applications.

COTS software products are ready-made packages sold off-the-shelf to the acquirer who had no influence on its features and other qualities. Typically the software is sold pre-wrapped with its user documentation. The information provided on the cover of the package is often the only means whereby the manufacturer or marketing organization can communicate with the acquirer and user. It is therefore important that essential information is given to enable acquirers to evaluate the quality of the COTS software products for their needs.

Selecting high quality COTS software products is of prime importance, because COTS software products may have to be operational in various environments and selected without the opportunity to compare performance among similar products. Suppliers need a way to ensure confidence in services given by the COTS software product to the users. Some suppliers may choose third-party evaluation or certification to assist them in providing this confidence.

In addition, when users require assurances that business or safety critical risks are involved, those assurances may need to be addressed by the user using techniques chosen by the user after the purchase. It is not the intent of this International Standard to specify minimum safety or business critical quality requirements for COTS, however, informative guidance is given. (See Annex B.)

ISO/IEC 12119:1994, was developed to support these needs. This International Standard took into account ISO/IEC 9126:1991, which defined quality characteristics.

The environment has changed. ISO/IEC 9126 has been revised, issued as ISO/IEC 9126-1 (including, for example, concept about quality in use), and in the new SQuaRE series it will become ISO/IEC 25010. ISO/IEC 12119:1994 has been used by certification bodies, which have identified some difficulties and ambiguities to correctly use the first edition.

These items are the major points for revising this International Standard, which provides a set of requirements for COTS software product and requirements for testing a COTS software product against its requirements.

This document is then a revision of ISO/IEC 12119:1994 to:

– be consistent with ISO/IEC 25010;

– take into account the experience resulting from usage of the standard, particularly by certification bodies;

– take into account the new normative context;

– add a clause about testing;

– delete Annex B for consistency with ISO 9127.

Software engineering — Software product Quality Requirements and Evaluation (SQuaRE) — Requirements for quality of Commercial Off-The-Shelf (COTS) software product and instructions for testing

1 Scope

This International Standard is applicable to COTS software products.

In this International Standard, the term “COTS” is used as an adjective and stands for “Commercial Off-The-Shelf”.

EXAMPLE Examples of COTS software products include but are not limited to text processors, spreadsheets, data base control software, graphics packages, software for technical, scientific or real-time embedded functions, such as real-time operating systems or local area networks for aviation/communication, automated teller machines, money conversion, human resources management software, sales management, and web software such as generators of web sites/pages.

This International Standard establishes:

a) Quality requirements for COTS software products;

b) Requirements for test documentation for the testing of COTS software products, including test requirements, test cases, and test reporting;

c) Instructions for conformity evaluation of COTS software products.

NOTE The collection of documents for test is called “test documentation”.

It includes also recommendations for safety or business critical COTS software products.

This International Standard deals only with providing the user confidence that the COTS software product will perform as offered and delivered. It does not deal with the production process (including activities and intermediate products, e.g. specifications). The quality system of a supplier is outside the scope of this International Standard.

The intended users of this International Standard include:

a) suppliers when:

   1) specifying requirements for a COTS software product;

   2) advertising performance claims for their product (ISO 9127);

   3) assessing their own software products against the claimed performance;

   4) issuing declarations of conformity (ISO/IEC 17050);

   5) applying for certificates or marks of conformity (ISO/IEC Guide 23);

b) certification bodies that may wish to establish a third-party certification scheme (international, regional or national) (ISO/IEC Guide 28);

c) testing laboratories which will have to follow the instructions for testing when testing for a certificate or a mark of conformity (ISO/IEC 17025);

d) accreditation bodies for accrediting registration or certification bodies and testing laboratories;

e) potential acquirers who may:

   1) compare the requirements for the intended work task with the information in product descriptions of existing software products;

   2) look for certified COTS software product;

   3) check if the requirements are otherwise met;

f) end users who may profit from better software products;

g) organizations:

   1) establishing management and engineering environments based on the quality requirements and methods of this international standard; and

   2) managing and improving their quality processes and personnel;

h) regulatory authorities who may require or recommend the requirements of this International Standard for COTS software products used in safety or business-critical applications.

Annex C provides guidance on the use of this International Standard.

2 Conformance

A COTS software product conforms to this International Standard if:

a) it has the properties specified in Clause 5;

b) it has been tested by producing test documentation that meets the requirements of Clause 6;

c) anomalies found during testing are documented and resolved prior to product release. Anomalies against advertised performance claims must be fixed or the performance claim must be removed. Known anomalies may be considered acceptable if:

   1) the anomaly is not a violation of a performance claim; and

   2) the supplier has duly considered the nature and the impact of the anomaly on the potential acquirer and deemed it negligible, and has preserved the documentation of the anomalies for future improvement.

Subclause recommendations are optional.

NOTE To facilitate the conformity evaluation, requirements of the present standard are drafted in a way that they are level 3 subclauses (numbered X.X.X.X). Informative notes complete these clauses and can serve as a guide.

3 Normative references

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

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

ISO/IEC 9126-1:2001, *Software engineering — Product quality — Part 1: Quality model*

Note The reference to ISO/IEC 9126-1 will be replaced by a reference to ISO/IEC 25010, *Software product Quality Requirements and Evaluation (SQuaRE) — Quality Model*

Refer to Bibliography for additional informative documents.

4 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

4.1
application administration function
functions performed by users which include installation, configuration, application backup, maintenance (patching and upgrading) and de-installation

4.2
conformity evaluation report
document that describes the conduct and results of the evaluation carried out for a COTS software product

NOTE This was adapted from IEEE Std 610.12-1990.

4.3
COTS software product
Commercial-Off-The-Shelf software defined by a market driven need, commercially available, and whose fitness for use has been demonstrated by a broad spectrum of commercial users

NOTE 1 COTS software product includes:

– the product description (including all cover information, data sheet, web site information, etc.),

– the user documentation (necessary to install and use the software),

– the software contained on a computer sensible media (disk, CD-ROM, internet downloadable, etc.).

NOTE 2 This was adapted from ISO/IEC 14598-4:1999.

NOTE 3 Software is mainly composed of programs and data.

NOTE 4 This definition apply also to product description, user documentation and software which are produced and supported as separate manufactured goods, but for which typical commercial fees and licensing considerations may not apply.

4.4
function
implementation of an algorithm in the software with which the end user or the software can perform part or all of a work task

NOTE A function does not need to be callable by the end user (e.g. automatic backupor saving of data).

4.5
product description
document stating properties of software, with the main purpose of helping potential acquirers in the evaluation of the suitability for themselves of the software before purchasing it

4.6
requirements document
document containing any combination of requirements or regulations to be met by a COTS software product

EXAMPLE These documents may be technical reports, standards, requirements list (or model requirements specification) for a kind of users, or a statute or regulation imposed by a governing or regulatory body.

4.7
test documentation
collection of the documentation inherent to the testing activities

4.8
test environment
hardware and software configuration necessary to conduct the test case

4.9
test objective
identified set of software features to be measured under specified conditions by comparing actual behavior with the required behavior

NOTE This was adapted from IEEE Std 610.12-1990.

4.10
test plan
document describing the scope, approach, resources, and schedule of intended testing activities

NOTE This was adapted from IEEE Std 610.12-1990.

4.11
testing description
description of the test execution conditions (i.e. test procedure)

Refer to Annex A for additional definitions from other standards.

5 Requirements for COTS software product

5.1 Requirements for product description

NOTE The paragraph concerning the Cover information of ISO/IEC 9127 *Software engineering — User documentation and cover information for consumer software package* can be used as input for creating a product description.

5.1.1 Availability

5.1.1.1 The product description shall be available for potential acquirers and users of the product.

5.1.2 Contents

5.1.2.1 The product description shall contain information needed by potential acquirers to evaluate the suitability of the software for their needs.

5.1.2.2 The product description shall be free from internal inconsistences.

5.1.2.3 The statements included in the product description shall be testable or verifiable.

5.1.3 Identification and indications

5.1.3.1 The product description shall display a unique identification.

5.1.3.2 The COTS software product shall be designated by its name, a version, and a date.

5.1.3.3 The product description shall contain the name and address (postal or web) of the supplier and at least one seller, e-commerce seller or distributor (if applicable).

5.1.3.4 The product description shall identify the intended work tasks and services that can be performed with the software.

5.1.3.5 When requirements defined by law or by a regulatory body apply to the COTS Software product and the supplier want to claim conformity to the corresponding requirements documents, the product description shall identify those requirements documents.

5.1.3.6 The product description shall indicate whether the COTS software product is intended for multiple concurrent end users or for a single end user on a single system, and shall state the maximum number of concurrent end users feasible at a stated level of performance on the required system.

5.1.3.7 If the product description makes reference to known user callable interfaces to other software, these interfaces or software shall be identified.

5.1.3.8 The product description shall indicate where the COTS software product relies on specific software and/or hardware with appropriate references.

EXAMPLES The reference may include:
– name of software and/or hardware;
– version;
– specific operating system.

5.1.3.9 The product description shall state whether support for operating the COTS software product is offered or not.

5.1.3.10 The product description shall state whether maintenance is offered or not. If offered, the product description shall describe the maintenance services offered.

5.1.4 Statements on functionality

5.1.4.1 The product description shall contain, as applicable, statements on functionality, taking into account suitability, accuracy, interoperability, security, and functionality compliance, written such that verifiable evidence of compliance can be demonstrated, based on ISO/IEC 9126-1:2001.

5.1.4.2 The product description shall provide an overview of end user callable functions of the product.

5.1.4.3 The product description shall describe all critical functions.

NOTE Refer to Annex B and to ISO/IEC 15026 for more information.

5.1.4.4 If there are options and versions for software components, they shall be indicated.

5.1.4.5 All known limitations to user functionality shall be described.

EXAMPLES These limitations may be:
– minimum or maximum values;
– lengths of keys;
– maximum number of records in a file;
– maximum number of search criteria;
– minimum sample size.

5.1.4.6 If prevention of unauthorized access, whether accidental or deliberate, to the software is provided, the product description shall include this information.

5.1.5 Statements on reliability

5.1.5.1 The product description shall contain, as applicable, statements on reliability, taking into account maturity, fault tolerance, recoverability, and reliability compliance, written such that verifiable evidence of compliance can be demonstrated, based on ISO/IEC 9126-1:2001.

NOTE No statement claiming reliability should be made unless the developer can substantiate the claim with in-service data or other verifiable data.

5.1.5.2 The product description shall address the ability of the software to continue operating (i.e. to be available) in the case of user interface errors, errors in the application's own logic, or errors due to availability of system or network resources.

5.1.5.3 The product description shall include information on data saving and restoring procedures.

NOTE An indication affirming that data backup may be executed by functions of the operating system is acceptable.

5.1.6 Statements on usability

5.1.6.1 The product description shall contain, as applicable, statements on usability, taking into account understandability, learnability, operability, attractiveness, and usability compliance, written such that verifiable evidence of compliance can be demonstrated, based on ISO/IEC 9126-1:2001.

5.1.6.2 The product description shall specify the type of user interface.

EXAMPLES These interfaces may be:
– command line;
– menu;
– windows;
– web browser;
– function key;
– help function.

5.1.6.3 The product description shall specify the specific knowledge required for the use and operation of the software.

EXAMPLES They can be:
– knowledge of the database calls and protocol used;
– knowledge of a technical area;
– knowledge of an operating system;
– knowledge obtainable by special training;
– knowledge of a language other than that in which the product description is written.

5.1.6.4 If the user can adapt the software, then the tools or procedures for this adaptation and the conditions of their use shall be identified.

EXAMPLES Conditions can be:
– changing of parameters;
– changing of algorithms for computation;
– interface customization;
– assignments to function keys.

5.1.6.5 If technical protection against copyright infringement can hamper usability, then this protection shall be stated.

EXAMPLES These protections may be:
– programmed expiry dates for usage;
– interactive reminders to pay for copies.

5.1.6.6 The software shall include provision for accessibility, particularly for users with disabilities and language differences.

5.1.7 Statements on efficiency

5.1.7.1 The product description shall include statements of efficiency, taking into account time behavior, resource utilization, and efficiency compliance, written such that verifiable evidence of compliance can be demonstrated, based on ISO/IEC 9126-1:2001.

EXAMPLES Stated conditions may be:
– system configurations;
– resources needed for efficient working with the COTS software product, e.g., hard disk space, RAM, video card, wireless internet card, etc.

5.1.8 Statements on maintainability

5.1.8.1 The product description shall contain, as applicable, statements on maintainability, taking into account analyzability, changeability, stability, testability, and maintainability compliance, written such that verifiable evidence of compliance can be demonstrated, based on ISO/IEC 9126-1:2001.

5.1.8.2 The product description shall include information on maintenance for the user.

EXAMPLES Information may be:
– monitoring ongoing dynamic performance of the app;
– monitoring unexpected failures and significant conditions;
– monitoring operational indicators such as logs, alert screens;
– monitoring local data which is operated upon by the application.

5.1.9 Statements on portability

5.1.9.1 The product description shall contain, as applicable, statements on portability, taking into account adaptability, installability, replaceability, co-existence, and portability compliance, written such that verifiable evidence of compliance can be demonstrated, based on ISO/IEC 9126-1:2001.

5.1.9.2 The product description shall specify the different configurations or supported configurations (hardware, software) for putting the software into use.

NOTE Different configurations may be specified, e.g. for different work tasks, different boundary values or different efficiency requirements.

EXAMPLES These systems may be:
– operating Systems;
– processing unit including co-processors;

– main memory size;
– types and sizes of peripheral storage;
– extension cards;
– input and output equipment;
– network environment;
– system softwareand other software.

5.1.9.3 The product description shall provide information on the installation procedure.

5.1.10 Statements on quality in use

5.1.10.1 The product description shall contain, as applicable, statements on quality in use, taking into account effectiveness, productivity, safety, and satisfaction in a specified context of use, written such that verifiable evidence of compliance can be demonstrated, based on ISO/IEC 9126-1:2001.

NOTE Statements on quality in use highly depends on in the context of use. It is impossible to take into account all possible users of COTS. Typical expected users and their intended use should be described.

EXAMPLES Statements on quality in use may be:
– the percentage of product that was audited,
– the number of open problem reports or observations in the COTS development process,
– the user survey results that have been corrected or uncorrected.
Refer to ISO/IEC 25062 for information on test efficiency, effectiveness, and satisfaction.

5.1.10.2 A reference to the test report shall be given.

NOTE The format of the test report is given in ISO/IEC 9126-4.

5.2 Requirements for user documentation

NOTE ISO/IEC 9127 *Software engineering — User documentation and cover information for consumer software package* can be used for creating user documentation.

5.2.1 Completeness

5.2.1.1 The user documentation shall contain the information necessary for the use of the software.

5.2.1.2 The user documentation shall describe all the functions stated in the product description and all functions that the end user can call.

5.2.1.3 The user documentation shall describe the reliability features and their operations.

5.2.1.4 The user documentation shall list the errors and failures that are handled and cause application failure or termination, particularly, those conditions ending in application termination which end in loss of data.

5.2.1.5 The user documentation shall give guidance to backup and restore the necessary data.

5.2.1.6 The user documentation shall provide complete instructional and reference information for all critical software functions (software whose failure could have an impact on safety, or could cause large financial or social loss).

NOTE See Annex B for more information.

5.2.1.7 The user documentation shall state all limitations given in the product description.

5.2.1.8 The user documentation shall state the minimum and maximum required disk space for installation.

5.2.1.9 The user documentation shall include all information necessary for user performed application administration functions.

5.2.1.10 Information allowing the user to verify the successful completion of application administration functions shall be included in the information for user performed application administrative functions.

5.2.1.11 If the user documentation is provided in several parts, at least one item in the set shall identify all the parts.

5.2.2 Correctness

5.2.2.1 All information in the user documentation shall be correct.

NOTE All information in the user documentation should be traceable to an authoritative source for correctness.

5.2.2.2 The user documentation shall present the information free from ambiguities.

5.2.3 Consistency

5.2.3.1 The documents of the user documentation shall be free from contradiction within themselves, with each other, and with the product description.

NOTE Consistency with the software is dealt with in subClause 5.3.1.5.

5.2.4 Understandability

5.2.4.1 The user documentation shall be understandable by the end user population for which the COTS software product is primarily targeted by using terminology and style understandable by its specialized audience.

EXAMPLE A COTS software product targeted for architects.

5.2.4.2 Understanding of the user documentation shall be facilitated by an organized document list.

5.2.5 Learnability

5.2.5.1 The user documentation shall provide the information necessary to learn how to use the software.

NOTE the user documentation may reference additional information contained within the COTS software package itself, or within auxiliary materials such as training.

5.2.6 Operability

5.2.6.1 If user documentation is not provided in printed form, the documentation shall indicate whether it can be printed, and if so, how to obtain a printed copy.

5.2.6.2 User documentation other than cards and quick reference guides shall have a table of contents, or list of topics, and an index.

5.2.6.3 User documentation shall define uncommonly used terms and acronyms.

5.3 Quality requirements for software

NOTE Software whose failure could have an impact on safety or business critical objectives may take into account guidance and recommendations in Annex B.

5.3.1 Functionality

5.3.1.1 Following installation, it shall be recognizable whether or not the software can function.

EXAMPLE The verification of good functioning can be done by using supplied test cases or by self-testing with corresponding messages, or by other tests conducted by the user.

5.3.1.2 All functions mentioned in the user documentation shall be executable with the corresponding facilities, properties, and data, and within the limitations given there.

5.3.1.3 The functions of the software shall be executable according to all the statements in the user documentation.

5.3.1.4 The software shall comply with all the requirements in any requirements document referenced by the product description.

5.3.1.5 The software shall be free from contradictions within itself and with the product description and user documentation.

EXAMPLE Two identical actions shall return the same result.

5.3.1.6 The control of the software operation by the end user following user documentation and the software behavior shall be consistent.

5.3.2 Reliability

5.3.2.1 The software must perform in accordance with the reliability features defined in the user documentation.

5.3.2.2 The function related to error handling shall be consistent with corresponding statements in the product description and in the user documentation.

NOTE The software can not be held responsible for many kinds of failures originating in the operating system or network.

5.3.2.3 The software shall not lose data when used within the limitations stated in the user documentation.

NOTE This requirement may be met in the case that:
– capacity is exploited up to the specified limits;
– attempts are made to exploit capacity beyond the specified limits;
– an incorrect input is made by the end user or from other software listed in the product description;
– explicit instructions in the user documentation are violated.

5.3.2.4 The software shall recognize violations of syntactic conditions for input and it shall not process this as permissible input.

5.3.3 Usability

5.3.3.1 The questions, messages, and results of the software execution shall be understandable.

EXAMPLE The understandability can be achieved:
– by an adequate selection of terms;
– by graphical representations;
– by provision of background information;
– by the explanations of a help function.

NOTE With respect to usability, parties to agreements based on this International Standard are encouraged to investigate the possibility of applying the most recent editions of standards in the ISO 9241 series. In particular parts 1, 2, 10 to 17 of the ISO/IEC 9241 series and ISO/IEC 25062 should be considered.

5.3.3.2 The software error message shall indicate how to correct the error or who to contact to report errors.

EXAMPLE This information can be a reference to an item in the user documentation.

5.3.3.3 The software shall provide information in such a form that it is easily understandable by the end users, i.e. text or graphic output visible and easy to read, audio output easy to hear.

5.3.3.4 Messages from the software shall be so designed that the end user can easily understand the type of message.

EXAMPLE These messages can be:
– acknowledgement;
– queries from software;
– warnings;
– error messages.

5.3.3.5 Input screen formats, reports, and other outputs shall be clear and understandable by the users.

5.3.3.6 The execution of functions that have serious consequences shall be reversible, or the software shall give a clear warning of the consequences and request confirmation before executing the command.

EXAMPLE Erasure and overwriting of data, as well as interruptions of a lengthy processing operation, having serious consequences.

5.3.3.7 The end user shall be able to learn how to use a function by means provided by the user interface, help function or user documentation.

5.3.3.8 The end user shall be advised when executing a function with response time beyond common expected limits is encountered.

5.3.3.9 Each element (data medium, file, ...) shall bear the product identification and, if there is more than one , an identification number or text.

5.3.4 Efficiency

5.3.4.1 Statements on efficiency in the product description shall be conformed to.

EXAMPLE Message to end user when time to wait for a response is unreasonable.

5.3.5 Maintainability

5.3.5.1 Statements on maintainability in the product description shall be conformed to.

EXAMPLE Capability to diagnose for deficiencies, capability to enable a modification.

5.3.6 Portability

5.3.6.1 If the user can carry out the installation, the software shall be installed successfully by following the information in the installation documentation.

5.3.6.2 Successful installation and correct operation of the software application shall be verified for all supported platforms and systems listed in the product description

5.3.6.3 If the user can carry out the installation and the software has any co-existence constraint to any component installed, it shall be stated before installation occurs.

5.3.6.4 The software shall provide a means for the user to remove or uninstall all its installed components.

5.3.7 Quality in use

5.3.7.1 Statements on quality in use in the product description shall be conformed to.

6 Requirements for test documentation

6.1 General Requirements

6.1.1 Purpose

6.1.1.1 The test documentation purpose is to demonstrate the conformity of the software to the requirements defined in the subclause 5.3. It contains all the elements allowing this demonstration.

6.1.2 Consistency

6.1.2.1 Information contained in each document of the test documentation shall be verifiable and correct.

6.1.2.2 Each document of the test documentation shall be free from contradiction within themselves and with product description and user documentation.

6.1.3 Requirements for content

6.1.3.1 The test documentation shall contain:

a) the test plan;

b) the test description;

c) the tests results.

6.1.3.2 The test documentation shall contain a list of all the documents that compose it, with their titles and their identifiers.

6.1.3.3 Each document of the test documentation shall include:
– a title;
– a single identifier (reference, number of version, date of issue).
– a history of the modifications or any other element describing the evolution of the document;
– contents or a description of the content;
– the identifier of the documents referred to in the body of the document;
– information relating to the authors and the inspectors;
– a glossary.

6.1.3.4 The test documentation may be composed of one or more documents.

6.2 Requirements for the test plan

6.2.1 Approach

NOTE No specific test techniques or methods are recommended.

6.2.1.1 All quality characteristics mentioned in the product description and in subclause 5.3, Quality requirements for software, shall be subject to test cases.

6.2.1.2 Each quality characteristics mentioned in the product description and in subclause 5.3, Quality requirements for software, shall be the objective of at least one test case

NOTE The test plan can refer any other document, providing that there is a relation between this document and the user documentation.

6.2.1.3 All the functions described in the user documentation, as well as the combinations of functions representative of the task to be achieved, shall be subject to test cases.

6.2.1.4 Each function described in the user documentation shall be the objective of at least one test case.

6.2.1.5 The test cases shall demonstrate the conformity of the software to the statements in the user documentation.

6.2.1.6 When requirements documents are mentioned in the product description, they shall be subject to test cases.

6.2.1.7 The level of functional decomposition selected as basis for the test case design shall be indicated.

EXAMPLE A function can be:
– a paragraph of the user documentation;
– a command of a shell;
– a button on the user interface;
– a language command.

6.2.1.8 The design method of test cases shall be indicated.

EXAMPLE Possible design methods are:
– boundary value analysis;
– checklist;
– data flow analysis;
– fault insertion;
– volume testing.

6.2.1.9 All the installation procedures shall be subject to test cases.

6.2.1.10 All the operational limits indicated in the product description and user documentation shall be subject to test cases.

6.2.1.11 Identified violations of syntactic conditions for input shall be subject to test cases.

6.2.1.12 If examples are indicated in the user documentation, they shall be used as test cases but the whole test shall not be limited to these examples.

6.2.1.13 If any requirement in Clauses 5.3, Quality requirements for software, is not applicable, the reason shall be stated.

6.2.2 Pass/fail criteria

6.2.2.1 The criteria used to decide if the test results demonstrate the conformity of the software to the product description and user documentation shall be indicated.

6.2.3 Test environment

6.2.3.1 The test plan shall specify the hardware and software configuration in which the tests are to be executed.

6.2.3.2 The software shall be tested in all the configurations of application mentioned in the product description.

NOTE Demonstration of equivalence of configurations can be used.

6.2.3.3 The test plan shall identify the tools necessary for the execution of the test cases.

6.2.4 Schedule

6.2.4.1 The test plan shall specify the schedule for each testing activity and test milestone.

6.3 Requirements for the testing description

6.3.1 Test case description

6.3.1.1 The description of each test case shall include:

a) its test objective;

b) a unique identifier;

c) input data and test boundaries for test;

d) the detailed steps to perform;

e) the expected behavior of the system;

f) the expected output from the test case;

g) the criteria for the result interpretation;

h) the criteria used to decide on positive or negative result of the test case.

6.3.1.2 Environment and other test conditions (detailed configuration and preliminary works) should be stated if it is necessary to bring additional information compared to those provided in the test plan.

6.3.2 Test procedures

6.3.2.1 The test procedure shall include:

a) the test preparation;

b) the actions necessary to begin and to execute the test;

c) the actions necessary to record the test results;

d) the conditions and actions to stop and eventually restart the tests.

6.3.2.2 Test procedures shall be sufficiently detailed to provide for repeatability and reproducibility of the tests.

6.3.2.3 Following correction, there shall be a procedure for re-testing of the functions concerned and any related functions.

NOTE A pseudo-language or a command language may be used to describe the test procedures.

6.4 Requirements for the test results

6.4.1 Execution report

6.4.1.1 The execution report shall include an overall summary of the results of the test cases.

6.4.1.2 The execution report shall demonstrate that all test cases have been executed according to the test plan.

6.4.1.3 For each test case, the execution reports shall include:

a) the identifier of the test case;

b) the date of the test execution;

c) the name and the function of the person having carried out the test;

d) the list of the found anomalies;

e) for each anomaly, the reference to the corresponding anomaly report.

6.4.2 Anomaly report

6.4.2.1 The anomaly report shall include an overall summary of the anomalies found and, if any, the corrections and the verifications by re-testing.

6.4.2.2 The descriptive part of the anomaly report shall include for each anomaly:

a) the identifier of the anomaly;

b) the identifier of the software;

c) the anomaly description;

d) the point in the test case the anomaly occurred;

e) the nature of the anomaly.

EXAMPLE The nature may be “blocking”, “major”, “minor”.

6.4.2.3 The correction part of the anomaly report shall demonstrate that all anomalies found have been corrected.

6.4.2.4 The correction part of the anomaly report shall include for each correction:

a) the identifier of the correction;

b) the correction date;

c) the name of the corrector;

d) the identifier of the modification corresponding to the correction;

e) the possible impact of the correction;

f) the possible comments of the corrector.

6.4.2.5 The verification part by re-testing of the anomaly report shall demonstrate that all corrected functions have the behaviour defined in the user documentation.

6.4.2.6 The verification part by re-testing of the anomaly report shall include for each verification:

a) the identifier of the verification;

b) the verification date;

c) the name of the verifier;

d) the test cases used for the verification;

e) the results of verification.

6.4.3 Assessment of the test results

6.4.3.1 The assessment of the execution report and anomaly report shall demonstrate that all expected behaviors were obtained, within the limits of the criteria used to decide if the test results show the conformity of the software.

7 Instructions for conformity evaluation

7.1 General Principles

The product description, the user documentation, and the software to be delivered, as parts of the COTS software product, shall be evaluated for conformity with the requirements in Clause 5.

NOTE The term “conformity evaluation” does not imply any technique or tool: testing, validation, verification, review, analysis, …

These instructions are primarily aimed at third-party evaluation. The third-party can be a testing laboratory working in accordance with some certification scheme or an in house testing laboratory that is independent from the supplier of the COTS software product.

7.2 Conformity evaluation pre-requisites

7.2.1 Presence of COTS software product items

For evaluation of a COTS software product, all items to be delivered (see 5.2.1.11) as well as the requirements documents identified in the product description (see 5.1.3.5) shall exist.

7.2.2 Presence of system elements

All components of all the computer systems as described in the product description shall exist and be available for conformity evaluation.

7.3 Conformity evaluation activities

7.3.1 Product description conformity evaluation

A conformity evaluation is carried out to determine the conformity of the product description to the requirements in subclause 5.1.

NOTE No specific techniques or tools are recommended.

7.3.2 User documentation conformity evaluation

A conformity evaluation is carried out to determine the conformity of the user documentation to the requirements in subclause 5.2.

NOTE No specific techniques or tools are recommended.

7.3.3 Software conformity evaluation

A conformity evaluation is carried out to determine the conformity of the software to the requirements in subclause 5.3 by producing test documentation conforming to the requirements in Clause 6, but without the part related to anomalies corrections and to verification by re-testing (subclauses 6.4.2.3 to 6.4.2.6).

NOTE The test documentation includes the descriptive part for the anomalies found, however correction of discovered anomalies is beyond the scope of a third party conformity evaluation.

7.4 Third-party conformity evaluation process

The supplier provides the COTS software product to the third-party. The supplier can also provide test documentation.

If the supplier provides only the COTS software product, without the test documentation, the third-party shall:

a) carry out a conformity evaluation of the product description, the user documentation, and the software according to subclause 7.3;

b) record the results in a conformity evaluation report, according to subclause 7.5.

If the supplier provides the COTS software product and the test documentation, the third-party shall:

a) carry out a conformity evaluation of the product description and the user documentation according to subclauses 7.3.1 and 7.3.2;

b) carry out a conformity evaluation to determine the conformity of the test documentation to the requirements in Clause 6;

c) record the results in a conformity evaluation report, according to subclause 7.5.

NOTE The conformity of the test documentation to the requirements in Clause 6 establishes the conformity of the software to the requirements in subclause 5.3.

7.5 Conformity evaluation report

The third-party shall prepare the conformity evaluation report.

The conformity evaluation report shall establish the conformity of a COTS software product to the requirements of Clause 5.

The conformity evaluation report shall contain the following items:

a) the COTS software product identification;

b) the date of evaluation completion and, if any, testing completion;

c) if any, the computer systems used for testing (hardware, software, and their configuration);

d) the documents used, with their identification;

e) the summary of conformity evaluation activities and, if any, testing activities;

f) the summary of conformity evaluation results and, if any, testing results;

g) the detailed results of conformity evaluation and, if any, testing;

h) if any, the list of non-conformities to requirements.

The results part of the conformity evaluation report (items f to h in previous paragraph) shall contain the product description and the user documentation conformity evaluation results. According to the supplied elements, it shall also contain one of the two following elements:

a) the results of the tests of the software to the requirements in subclause 5.3, i.e. the descriptive part of the anomaly report (subclause 6.4.2.2), in case of the supplier provides only the COTS software product, without the test documentation;

b) the results of the conformity evaluation of the test documentation to the requirements in Clause 6, in case of the supplier provides the COTS software product and the test documentation.

NOTE The conformity evaluation report contains only the descriptive part of the anomaly report because it is not the responsibility of the third-party to correct the anomalies.

For conformity evaluation reports in printed form, the identification of the conformity evaluation report (testing laboratory, COTS software product identification, date of the conformity evaluation report) and the total number of its pages shall appear on each page of the conformity evaluation report.

The conformity evaluation report shall include:

a) a statement to the effect that the evaluation and, if any, test results relate only to the items evaluated and tested;

b) a statement that the conformity evaluation report shall not be reproduced, except in full, without the written approval of the testing laboratory.

7.6 Follow up conformity evaluation

When a COTS software product, which has already been evaluated for conformity, is evaluated again, taking into consideration the previous conformity evaluation, then:

a) all changed parts in the documents and software shall be evaluated as if it were a new COTS software product;

b) all unchanged parts that are expected to be influenced by the changed parts or by changes in a required system shall be evaluated as if it were a new software;

c) all other parts shall at least be evaluated by samples.

Annex A
(informative)

Definitions from others standards

A.1
acquirer
an organization that acquires or procures a system, software product or software service from a supplier

NOTE Acquirer could be one of the following: buyer, customer, owner, user, purchaser

[ISO/IEC 12207:1995]

A.2
anomaly
any condition that deviates from expectations based on requirements specifications, design documents, standards, etc. or from someone's perceptions or experiences

[IEEE Std 1044-1993]

A.3
conformity evaluation
systematic examination of the extent to which a product, process or service fulfils specified requirements

[ISO/IEC Guide 2:1996]

A.4
end user
individual person who ultimately benefits from the outcomes of the system

NOTE The end user may be a regular operator of the software product or a casual user such as a member of the public.

[ISO/IEC 25000:2005]

A.5
fault
an incorrect step, process, or data definition in a computer program

[IEEE STD 610.12-1990]

A.6
maintenance
the process of modifying a software system or component after delivery to correct faults, improve performance or others attributes, or adapt to a changed environment

[IEEE Std 610.12-1990]

A.7
pass/fail criteria
decision rules used to determine whether a software item or a software feature passes or fails a test

[IEEE Std 829.12-1998]

A.8
software
all or part of the programs, procedures, rules, and associated documentation of an information processing system

NOTE 1 Software is an intellectual creation that is independent of the medium on which it is recorded.

[ISO/IEC 2382.1:1993]

NOTE 2 In the present standard, the documentation is not considered as part of the software, but as separate item.

A.9
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

NOTE

1 The term “supplier” is synonymous with contractor, producer, seller, or vendor.

2 The acquirer may designate a part of its organization as supplier.

[ISO/IEC 12207:1995]

A.10
test
an activity in which a system or component is executed under specified conditions, the results are observed or recorded, and an evaluation is made of some aspect of the system or component

[IEEE Std 610.12-1990]

A.11
test case
a set of inputs, execution conditions, and expected results developed for a particular objective, such as exercise a particular program path or to verify compliance with a specific requirement

[IEEE Std 610.12-1990]

A.12
test procedure
detailed instructions for the set-up, execution, and evaluation of results for a given test case

[IEEE Std 610.12-1990]

A.13
testing
the process of operating a system or component under specified conditions, observing or recording the results, and making an evaluation of some aspect of the system or component

[IEEE Std 610.12-1990]

A.14
third-party
person or body that is recognized as being independent of the parties involved, as concerns the issue in question

[ISO/IEC Guide 2:1996]

A.15
user
individual or organisation that uses the system to perform a specific function

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

[ISO/IEC 15939:2002]

user
person or business organisation that uses the software product to perform a specific function

[ISO/IEC 18019:2004]

A.16
user documentation
information that is supplied with the software to help the user in their use of that software

[ISO/IEC 18019:2004]