IEEE Std 1465-1998(R2004)™
[Adoption of
ISO/IEC 12119: 1994(E)]

IEEE Standard

Adoption of International Standard ISO/IEC 12119: 1994(E)

Information Technology—Software packages—Quality requirements and testing

Sponsor

Software Engineering Standards Committee
of the
IEEE Computer Society

Reaffirmed 8 December 2004

IEEE-SA Standards Board

Approved 25 June 1998

IEEE-SA Standards Board

Abstract: Quality requirements for software packages and instructions on how to test a software package against these requirements are established. The requirements apply to software packages as they are offered and delivered, not to the production process (including activities and intermediate products, such as specifications).

Keywords: quality requirements, software, software engineering, software package, testing

The Institute of Electrical and Electronics Engineers, Inc.
345 East 47th Street, New York, NY 10017-2394, USA

Print: ISBN 0-7381-0343-8 SH94665
PDF: ISBN 0-7381-1408-1 SS94665

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. Members of the committees serve voluntarily and without compensation. They are not necessarily members of the Institute. The standards developed within IEEE represent a consensus of the broad expertise on the subject within the Institute as well as those activities outside of IEEE that have expressed an interest in participating in the development of the standard.

Use of an IEEE Standard is wholly voluntary. The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE Standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard. Every IEEE Standard is subjected to review at least every five years for revision or reaffirmation. When a document is more than five years old and has not been reaffirmed, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE Standard.

Comments for revision of IEEE Standards are welcome from any interested party, regardless of membership affiliation with IEEE. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments.

Interpretations: Occasionally questions may arise regarding the meaning of portions of standards as they relate to specific applications. When the need for interpretations is brought to the attention of IEEE, the Institute will initiate action to prepare appropriate responses. Since IEEE Standards represent a consensus of all concerned interests, it is important to ensure that any interpretation has also received the concurrence of a balance of interests. For this reason, IEEE and the members of its societies and Standards Coordinating Committees are not able to provide an instant response to interpretation requests except in those cases where the matter has previously received formal consideration.

Comments on standards and requests for interpretations should be addressed to:

Secretary, IEEE-SA Standards Board
445 Hoes Lane
P.O. Box 1331
Piscataway, NJ 08855-1331
USA

Note: Attention is called to the possibility that implementation of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. The IEEE shall not be responsible for identifying patents for which a license may be required by an IEEE standard or for conducting inquiries into the legal validity or scope of those patents that are brought to its attention.

Introduction

[This introduction is not part of IEEE Std 1465-1998, IEEE Standard—Adoption of International Standard ISO/IEC 12119: 1994(E)—Information Technology—Software packages—Quality requirements and testing.]

The following persons were on the Software Engineering Standards balloting committee at the time this standard was adopted:

Syed Ali
Pamela K. Arya
Robert E. Barry
Leo Beltracchi
H. Ronald Berlack
Alan L. Bridges
Kathleen L. Briggs
Edward R. Byrne
Jaya R. Carl
Keith Chan
John P. Chihorek
Antonio M. Cicu
Theo Clarke
Rosemary Coleman
Darrell Cooksey
W. W. Geoff Cozens
Gregory T. Daich
Raymond Day
Bostjan K. Derganc
Carl Einar Dragstedt
Sherman Eagles
Leo Egan
William Eventoff
Jonathan H. Fairclough
John W. Fendrich
Jay Forster
Kirby Fortenberry
Eva Freund
Simon Gabrihelidis
Barry L. Garner
David Gelperin
Adel N. Ghannam
Hiranmay Ghosh
Marilyn Ginsberg-Finner
John Garth Glynn
Julio Gonzalez-Sanz

L. M. Gunther
John Harauz
Rob Harker
Carol J. Harkness
Robert T. Harley
Herbert Hecht
Mark Henley
Umesh P. Hiriyannaiah
John W. Horch
Charles C. Howell
Jerry Huller
Peter L. Hung
Fabrizio Imelio
George Jackelen
Frank V. Jorgensen
Vladan V. Jovanovic
William S. Junk
George X. Kambic
Myron S. Karasik
Ron S. Kenett
Dwayne L. Knirk
Shaye Koenig
Thomas M. Kurihara
John B. Lane
J. Dennis Lawrence
William M. Lively
Dieter Look
David Maibor
Tomoo Matsubara
Sue McGrath
Bret Michael
Alan Miller
James W. Moore
R. Muralidharan
Pavol Navrat
Myrna L. Olson

Mike Ottewill
Indradeb P. Pal
Lalit M. Patnaik
Donald J. Pfeiffer
Alex Polack
Peter T. Poon
Kenneth R. Ptack
Patricia Rodriguez
Andrew P. Sage
Helmut Sandmayr
Stephen R. Schach
Hans Schaefer
Norman Schneidewind
Robert W. Shillato
Carl A. Singer
Melford E. Smyre
Alfred R. Sorkowitz
Donald W. Sova
Fred J. Strauss
Christine Brown Strysik
Toru Takeshita
Carmen J. Trammell
Patricia Trellue
Leonard L. Tripp
T. H. Tse
Richard D. Tucker
Arthur van der Molen
Thomas E. Vollman
Dolores Wallace
David M. Waxman
Camille S. White-Partain
Scott A. Whitmire
P. A. Wolfgang
Natalie C. Yopconka
Geraldine Zimmerman
Peter F. Zoll

When the IEEE-SA Standards Board approved this standard on 25 June 1998, it had the following membership:

Richard J. Holleman, *Chair*
Donald N. Heirman, *Vice Chair*
Judith Gorman, *Secretary*

Satish K. Aggarwal
Clyde R. Camp
James T. Carlo
Gary R. Engmann
Harold E. Epstein
Jay Forster*
Thomas F. Garrity
Ruben D. Garzon

James H. Gurney
Jim D. Isaak
Lowell G. Johnson
Robert Kennelly
E. G. "Al" Kiener
Joseph L. Koepfinger*
Stephen R. Lambert
Jim Logothetis
Donald C. Loughry

L. Bruce McClung
Louis-François Pau
Ronald C. Petersen
Gerald H. Peterson
John B. Posey
Gary S. Robinson
Hans E. Weinrich
Donald W. Zipse

*Member Emeritus

Yvette Ho Sang
IEEE Standards Project Editor

Contents

IEEE Standard

Adoption of International Standard ISO/IEC 12119: 1994(E)

Information Technology—Software packages—Quality requirements and testing

Overview

IEEE Std 1465-1998 is an adoption of ISO/IEC 12119: 1994(E). IEEE Std 1465-1998 may be employed to

a) Specify quality requirements for software, and provide instructions on how to test against these requirements.

b) Manage and improve the organization's quality processes and personnel.

c) Establish management and engineering environments based on the quality requirements and methods in ISO/IEC 12119: 1994(E).

d) Foster improved understanding between customers and vendors, and among other parties involved in the software product life cycle.

e) Facilitate world trade in software.

References

This standard shall be used in conjunction with the following publications. When the following standards are superseded by an approved revision, the revision shall apply.

IEEE Std 610.12-1990, IEEE Standard Glossary of Software Engineering Terminology.[1]

IEEE Std 829-1998, IEEE Standard for Software Test Documentation.

IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications.

[1] IEEE publications are available from the Institute of Electrical and Electronics Engineers, 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331, USA (http://www.standards.ieee.org/).

IEEE Std 1008-1987 (Reaff 1993), IEEE Standard for Software Unit Testing.

IEEE Std 1012-1998, IEEE Standard for Software Verification and Validation.

IEEE Std 1012a-1998, IEEE Standard for Software Verification and Validation: Content Map to IEEE/EIA 12207.1-1997.

IEEE Std 1016-1998, IEEE Recommended Practice for Software Design Descriptions.

IEEE Std 1063-1987 (Reaff 1993), IEEE Standard for Software User Documentation.

IEEE Std 1233, 1998 Edition, IEEE Guide for Developing System Requirements Specifications.

IEEE adoption implementation considerations

The following implementation notes relate to IEEE interpretation of ISO/IEC 12119: 1994(E).

Terminology correlation

Some differences exist between the terminology used in IEEE Std 1465-1998 and that in IEEE Std 610.12-1990. The reader should use the local definitions in IEEE Std 1465-1998 where there is a conflict with those of IEEE Std 610.12-1990. In some cases, the definition in IEEE Std 610.12-1990 may provide further clarification. However, it is important to keep in mind that full compatibility of definitions has yet to be achieved in the evolving field of software engineering.

Page 5: In subclause 3.1.6, change “efficiency” in the heading to “performance.”

Page 7: In subclause 3.3.4, change “efficiency” in the heading and text to “performance.”

Page 8: In clause 4, change “ergonomic” in the second complete paragraph to “human factor.”

Errata

Page 2: In subclause 2.6, change “test objective” in the first dash to “test objective(s).”

Page 2: Replace the text of the second dashed item with the following:

– The functions to be tested, and, if possible, a mapping to their related requirements.

Page 3: In subclause 3.1.1, change “easy to overview” in the first paragraph to “organized.”

Page 6: In subclause 3.2.5, change “easy to overview” in the first paragraph to “organized.”

Page 9: In subclause 4.3, add the following text above the first dashed item:

– a detailed list of all test objectives to be covered in the test cases;

Page 9: In subclause 4.4, change “objects” in the first paragraph to “objectives.”

Page 9: In subclause 4.4, in the last paragraph, delete “and the total number of its pages.” The intent of listing the total number of pages is to provide a way to verify that no part of the test report is missing.

INTERNATIONAL STANDARD

ISO/IEC 12119

First edition
1994-11-15

Information technology — Software packages — Quality requirements and testing

Technologies de l'information — Progiciel — Exigences qualité et essais

Reference number
ISO/IEC 12119:1994(E)

Contents

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

Printed in Switzerland

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) together form a system for worldwide standardization as a whole. 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 liason with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for approval before their acceptance as International Standards. They are approved in accordance with procedures requiring at least 75 % approval by the national bodies voting.

International Standard ISO/IEC 12119 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*.

Information technology – Software packages – Quality requirements and testing

1 Scope

This International Standard is applicable to software packages. Examples are text processors, spreadsheets, data base programs, graphics packages, programs for technical or scientific functions, and utility programs.

It establishes

- requirements for software packages (quality requirements);
- instructions on how to test a software package against these requirements (instructions for testing, in particular for third party testing).

It deals only with software packages as offered and delivered. It does not deal with their production process (including activities and intermediate products, e.g. specifications). The quality system of a supplier is outside the scope of this International Standard.

NOTE – Some software needs additional requirements, e.g. safety-critical software.

The intended users of this International Standard include

a) suppliers when

1) specifying requirements for a software package;

2) designing a form to describe products;

3) assessing their own products;

4) issuing declarations of conformity [ISO/IEC Guide 22];

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

b) certification bodies which may wish to establish a third-party certification scheme (international, regional or national) [ISO/IEC Guides 16, 28 and 44];

c) testing laboratories which will have to follow the instructions for testing when testing for a certificate or a mark of conformity [ISO/IEC Guide 25];

d) accreditation bodies for accrediting certification bodies and testing laboratories [ISO/IEC Guides 40 and 58];

e) auditors of testing laboratories when assessing their competence [ISO/IEC Guide 58];

f) buyers who may

1) compare their requirements with those described here;

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

3) look for certified products;

4) check otherwise if the requirements are met;

g) users who may profit from better products.

2 Definitions

For the purposes of this International Standard, the following definitions apply. Definitions from other standards used in this International Standard are reproduced in annex A for convenient reference.

2.1 function: The implementation of an algorithm in the program with which the user or the program can perform part or all of a work task.

NOTES

1 A function does not need to be callable by the user (e.g. automatic backup or saving of data).

2 The notion of a function here is narrower than that used in ISO 2382-14:1978 (in the definitions of failure, fault, maintenance and reliability), but wider than those defined in ISO 2382-2 and ISO 2382-15.

2.2 requirements document: A document containing any combination of recommendations, requirements or regulations to be met by a software package.

NOTE – Examples are a technical or ergonomic standard, a requirements list (or model requirements specification) from a group (e.g. a market sector, technical or user association), a law or a decree.

2.3 product description: A document stating properties of a software package, with the main purpose of helping potential buyers in the evaluation of the suitability for themselves of the product before purchasing it.

NOTE – This term is more specific than the term "system description" in ISO/IEC 2382-20:1990. The purpose of the product description includes that of the "cover information" in ISO 9127. The product description is not a specification; it serves a different purpose.

2.4 user documentation: The complete set of documents, available in printed or non-printed form, that is provided for the application of the product and also is an integral part of the product.

2.5 package documentation: The product description and the user documentation.

2.6 test case: A documented instruction for the tester that specifies how a function or a combination of functions shall or should be tested. A test case includes detailed information on the following issues:

- the test objective;
- the functions to be tested;
- the testing environment and other conditions (configuration details and preparatory work);
- the test data;
- the procedure;
- the expected behaviour of the system.

2.7 maintenance: That part of system maintenance (see A.5.2) which is concerned with modifying a software package.

3 Quality requirements

Subclauses 3.1 to 3.3 contain

- the requirement that each software package has a product description and user documentation;
- requirements for the product description. In particular, there is a requirement that it shall contain specified information and that all its statements shall be testable and correct;
- requirements for the user documentation;
- requirements for the programs and for data, if any, included in the software package.

NOTES

1 The requirements on user documentation, programs and data contain many general requirements (independent from what may be promised in a product description), but they do not include all the properties that users may desire.

2 Certain properties, for example "understandability" and "ease of overview" of the user documentation and of the program messages, are admittedly required in the view of the user. However, due to the difficulty of testing for them with clear-cut and reproducible results these are formulated currently as recommendations only.

3 The requirements in 3.1 to 3.3 are arranged in the same order as the characteristics which appear in ISO/IEC 9126.

A software package conforms to this International Standard if it complies with all the requirements in 3.1 to 3.3. The recommendations (indicated by the use of the verbal form "should") are optional.

NOTE 4 – Conformity of a product to the requirements in 3.1 to 3.3 may be difficult or impossible to prove. However, a test (including review of documents) according to clause 4 is deemed sufficient to provide the confidence required for a certificate of conformity according to ISO/IEC Guide 2. No formal proof is needed.

3.1 Product description

Each software package shall have a product description.

The product description defines the product. It is a part of the package documentation of the product. It provides information on the user documentation, the programs and, if any, the data.

The main purposes of the product description are

- to help the user or potential buyer in their evaluation of the suitability of the product for themselves. To this extent it is also sales information;

- to serve as a basis for testing (see clause 4).

The product description shall be available for people interested in the product.

3.1.1 General requirements on contents

The product description should be sufficiently understandable, complete and easy to overview for helping potential buyers in the evaluation of the suitability for themselves of the product before purchasing it.

It shall be free from internal inconsistencies. Each term should have the same meaning everywhere.

The statements of the product description shall be testable and correct.

NOTE – This requirement extends to the statements in external requirements documents invoked, if any (see 3.1.2 e).

The following subclauses 3.1.2 to 3.1.8 specify what the product description shall or should include. It may include additional statements on the product.

3.1.2 Identifications and indications

a) Identification of the product description

The product description shall possess a unique document identification. It may be named differently from "product description", for example: "Functional Description", "Product Information", "Product Sheet".

b) Identification of the product

The product description shall identify the product. The product identification shall have at least the product name and a version or date. If there are two or more variants mentioned in the product description, then each variant shall have at least the product name, a variant name and a version or date.

c) Supplier

The product description shall contain the name and address of at least one supplier.

NOTE – The name and address need not be printed; the rubber stamp of a dealer is sufficient.

d) Work task

The product description shall identify the intended work tasks that can be performed with the product.

e) Conformity to requirements documents

The product description may refer to requirements documents to which the product conforms, in which case the relevant editions shall be identified.

f) Required system

The required system (hardware, software and their configuration) for putting the product into use shall be specified, including manufacturer names and type identifiers of all parts, for example

- processing unit including co-processors;

- main memory size;

- types and sizes of peripheral storage;

- extension cards;

- input and output equipment;

- network environment;

- system software and other software.

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

The statement "(or any other ..., if compatible)" may appear in a product description if previously a particular hardware or software product has been identified. The statement "or an updated version if compatible" may appear if previously a version of a product has been identified. The statement "from version X to at least version Y" may appear; "from version X" shall not appear.

NOTE – A statement "from version X" could later become false by the appearance of a version X+3 with which the software package would fail to operate.

g) Interfaces to other products

If the product description makes reference to interfaces to other products, the interfaces or products shall be identified.

h) Items to be delivered

Every physical component of the supplied product shall be identified, in particular all printed documents and all data media.

The form of the supplied programs shall be stated, e.g. source programs, object modules or load modules.

NOTE – Media formats (e.g. diskette formats) need not be indicated because the set of possible formats is determined by the required system (see 3.1.2 f).

i) Installation

It shall be stated whether or not the product installation can be carried out by the user.

j) Support

It shall be stated whether or not support for operating the product is offered.

k) Maintenance

It shall be stated whether or not maintenance is offered. If maintenance is offered, it shall be stated what is specifically included.

3.1.3 Statements on functionality

a) Overview of functions

The product description shall provide an overview of the user-callable functions of the product, the necessary data and the facilities offered.

It shall be clearly stated for each function mentioned (especially for an option or variant) whether it is a part

- of the product;
- of a product extension fully described in the product description;
- of a product extension referenced in the product description;
- of a supplement without guarantee.

NOTE – Not every user-callable function need be mentioned, and not every detail on how a function is called need be given.

b) Boundary values

If the usage of the product is limited by product specific boundary values, these shall be provided. Examples are

- minimum or maximum values;
- lengths of keys;
- maximum number of records in a file;
- maximum number of search criteria;
- minimum sample size.

In the case when it is not possible to provide fixed boundary values (when, for example, they depend upon the application type or upon the input data), then the limitations shall be stated. Permissible combinations of values may be provided and reference shall be made to more specific information in the user documentation.

c) Security

The product description should include information on means, if provided, for preventing unauthorized access, whether accidental or deliberate, to programs and data.

3.1.4 Statements on reliability

The product description shall include information on data saving procedures.

NOTE – It is sufficient to state, for example, that backup is possible by functions of the operating system.

Additional product properties should be described that ensure the functional capability of the product. Examples are

- checks whether input is plausible;
- protection against serious consequences of a user mistake;
- error recovery.

3.1.5 Statements on usability

a) User interface

The type of user interface shall be named, for example, command line, menu, windows, function key, help function.

b) Required knowledge

Specific knowledge required for the application of the product shall be specified. Examples are

- 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.

All the natural languages of the user documentation and of the user interface (including error messages and visible data) shall be stated, both of the software package itself and of all other products mentioned in the product description.

NOTE – This requirement exceeds the one in ISO 9127:1988, 6.1.7, where the mention of the languages used is optional.

c) Adaptation to the user's needs

If the product can be adapted by the user, then the tools for this adaptation and the conditions of their use shall be identified. Examples are

- changing of parameters;
- changing of algorithms for computation;
- assignments to function keys.

d) Protection against copyright infringement

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

- technical protection against copying;
- programmed expiry dates for usage;
- interactive reminders to pay for copies.

e) Efficiency of usage and user satisfaction

The product description may include data on efficiency of usage and user satisfaction.

NOTE – Such data may follow the guidance of ISO 9241-11.

3.1.6 Statements on efficiency

The product description may include data on the time behavior of the product such as response times and throughput rates for given functions under stated conditions (for example the system configurations and load profiles).

3.1.7 Statements on maintainability

The product description may contain statements on maintainability.

3.1.8 Statements on portability

The product description may contain statements on portability.

3.2 User documentation

3.2.1 Completeness

The user documentation shall contain the information necessary for the use of the product. All the functions stated in the product description and all user-callable functions in the program shall be completely described in the user documentation.

All boundary values given in the product description shall be repeated in the user documentation.

If installation can be carried out by the user, the user documentation shall include an installation manual containing all necessary information (see 3.3.1 a). The installation manual should state the minimum and maximum sizes of the files once installed.

If maintenance can be carried out by the user, the user documentation shall include a program maintenance manual containing all the information that is necessary for the kind of maintenance concerned.

3.2.2 Correctness

All information in the user documentation shall be correct. In addition, its presentation should be free from ambiguities and errors.

3.2.3 Consistency

The documents of the user documentation shall be free from contradiction within themselves, with each other and with the product description. Each term should have the same meaning everywhere.

NOTE – Consistency with programs and data is dealt with in 3.3.1.d.

3.2.4 Understandability

The user documentation should be understandable by the user population that normally performs the stated work task, for example by using an appropriate selection of terms, graphical exhibits, detailed explanations and by referencing useful information sources.

3.2.5 Ease of overview

The user documentation should be easy to overview so that relationships are recognizable.

Every document should have a table of contents and an index.

If a document is not provided in printed form the procedure for printing it should be indicated.

3.3 Programs and data

3.3.1 Functionality

a) Installation

If installation can be carried out by the user, it shall be possible to install the programs successfully by following the information in the installation manual. Each of the required systems given in the product description shall be sufficient for installing the programs.

Following installation it shall be recognizable whether or not the programs can function, for example using supplied test cases or by self-testing with corresponding messages.

b) Presence of functions

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

NOTE – Since all functions mentioned in the product description shall also appear in the user documentation, it follows that they too shall be executable.

c) Correctness

The programs and data shall correspond to all the statements in the product description and the user documentation. The functions shall be executed in a correct manner for the work task. In particular, programs and data shall comply with all the requirements in any requirements document referenced by the product description.

d) Consistency

The programs and data shall be free from contradictions within themselves and with the product description and the user documentation. Each term should have the same meaning everywhere.

The control of the program operation by the user and the program behaviour (for example, messages, input screen formats and printed reports) should be uniformly structured.

3.3.2 Reliability

The system, comprising hardware, required software and those programs that belong to the product, shall not fall into such a state that the user cannot control it, and shall neither corrupt nor lose data.

This requirement shall even 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 user or from another of the programs listed in the product description;
- explicit instructions in the user documentation are violated.

Only those possibilities for hardware and operating system interruption that cannot be caught by any program (for example, the key or combination of keys for system operation reset) are excluded.

The programs shall recognize violations of syntactic conditions for input. In the case where a program recognizes an input as erroneous or as undefined, it shall not process this as permissible input.

3.3.3 Usability

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.

Note – In particular parts 10 and 13 of the ISO 9241 series should be considered.

a) Understandability

The questions, messages and results of the programs should be understandable, for example,

- by an adequate selection of terms;
- by graphical representations;
- by provision of background information;
- by the explanations of a help function.

Error messages shall offer detailed information explaining the cause or the correction of the corresponding usage errors (for example by a reference to an item in the user documentation).

b) Ease of overview

Each data medium shall bear the product identification and, if there is more than one medium, a distinguishing number or text.

For the user, when working with the programs it shall always be possible to find out which function is being executed.

The programs should provide information to the user in such a form that it is easily visible and easy to read. The user should be guided by appropriate coding and grouping of the information. Where necessary, the programs should alert the user.

Messages from the programs should be so designed that the user can easily differentiate them by type, for example,

- acknowledgement;
- queries from programs;
- warnings;
- error messages.

Input screen formats, reports and other inputs and outputs should be designed to be clear and easy to overview. Possibilities for this include

- alphanumeric fields are left-aligned;
- numeric fields are right-aligned;
- in tables, decimal points or commas are arranged in the same vertical line;
- field limits are recognizable;
- fields, the use of which is obligatory, are recognizable as such;
- identified input failures are immediately highlighted in the input screen format;
- the user's attention is directed to a change in screen content by a visual or audible sign.

c) Operability

The execution of functions that have serious consequences shall be reversible, or the programs shall give a clear warning of the consequences and request confirmation before executing the command. In particular, erasure and overwriting of data, as well as interruptions of a lengthy processing operation, have serious consequences.

If documenting text is offered in the dialogue, the user should be able to access subclauses of the text in a direct manner, for example, by selection from a displayed table of contents and by a search function based on keywords.

3.3.4 Efficiency

Nothing is required. However, statements on efficiency in the product description shall be conformed to.

3.3.5 Maintainability

Nothing is required. However, statements on maintainability in the product description shall be conformed to.

3.3.6 Portability

Nothing is required. However, statements on portability in the product description shall be conformed to.

4 Instructions for testing

The instructions in 4.1 to 4.5 specify how a product shall be tested against the quality requirements. They include both testing for properties required from all conforming products and testing for properties promised by the product description. They include both testing by inspection of documents and blackbox testing of programs and data.

These instructions describe functional testing (blackbox testing). Structural testing is not included because it would require the availability of the source code.

Only the product in its required systems is tested. The ergonomic evaluation at the computer workplace is not considered in this International Standard.

NOTES

1 These instructions are primarily aimed at third party testing according to some certification scheme (see clause 1, item c). During production it may be cheaper and more effective to use structural testing.

2 Clause 4 does not contain requirements on software packages (all these are contained in clause 3). A software package may conform without having been tested according to clause 4, and such a test may fail to discover an existing non-conformity.

3 As the required system is determined by the product description, any nonconformity of the product on the required system is treated as a nonconformity of the product.

4 A certification scheme may make testing against recommendations optional.

5 Guidance on ergonomic evaluation is contained in ISO 9241-11.

4.1 Test pre-requisites

4.1.1 Presence of product items

For testing a software package all items to be delivered (see 3.1.2 h) as well as the requirements documents identified in the product description (see 3.1.2 e) shall be present.

4.1.2 Presence of system constituents

For testing a software package it is required that the constituent parts of all the computer systems as named in the product description are available.

4.1.3 Training

If training is mentioned in the product description, the tester shall have access to the training material and the training program.

4.2 Testing activities

The product description, user documentation, programs and any data to be delivered as parts of the software package

- shall be tested for compliance with the requirements in clause 3, and
- should be tested for compliance with the recommendations in clause 3.

The test objectives shall be derived from, and include all, the requirements in clause 3 (completeness, consistency etc.).

If other products are mentioned in the product description, these other products need only be tested for the claims made in the product description of the product under test.

Details in the product description, in the user documentation, in functions or in data of the product do not need to be tested if according to the judgement of the tester

- they have negligible influence on the suitability for the named work task, and
- they can be tested in principle but not with justifiable expenditure.

Such details which are not tested shall be mentioned in the test records and in the test report. The reasons for not testing them shall be documented in the test records.

4.2.1 Product description

The fulfilment of the requirements in clause 3 shall be tested, and the fulfilment of the recommendations in clause 3 should be tested.

4.2.2 User documentation

The fulfilment of the requirements in clause 3 shall be tested, and the fulfilment of the recommendations in clause 3 should be tested.

4.2.3 Programs and data

The fulfilment of the requirements in clause 3 shall be tested, and the fulfilment of the recommendations in clause 3 should be tested.

The programs shall be tested in all the computer systems that are named in the product description.

If there are several program variants, each shall be tested. Functions that, according to the product description and user documentation, are identical in a number of variants may be tested each in one variant.

The programs and the data supplied with them shall be tested using test cases constructed on the basis of the product description and the user documentation. Further material (for example, source programs) need not be considered unless the testing of statements in the product description or user documentation makes this necessary.

The test cases shall be constructed methodically and systematically.

NOTE – Methodical random testing is permitted.

If examples are given in the user documentation they shall be used as test cases, but testing shall not be restricted to these examples.

Test cases provided by the supplier of the software package may be used but testing shall not be restricted to these test cases.

a) Installation

If, according to the product description, installation can be carried out by the user, it shall be tested that the programs can be installed and tested for successful installation as described in the installation manual.

Otherwise it shall be ensured that the hardware and software environment of the installed programs corresponds to the statements in the product description for the computer system under consideration.

b) Program execution

The test cases shall cover all functions described in the product description and user documentation and they shall take into account combinations of functions that are representative for the work task.

The programs shall be tested for all boundary values (according to product description and user documentation) in the required systems for which these values apply.

Input or command sequences that are explicitly deprecated or declared as forbidden in the user documentation (see 3.3.2) shall be used in the testing.

4.3 Test records

The records for each test shall contain sufficient information to permit the repetition of the test [ISO/IEC Guide 25]. They shall include

- a test plan or test specification containing test cases (each test case stating its objectives, see 2.6);
- all the results associated with the test cases, including all the failures occurring during the test;
- the identity of personnel involved in testing.

4.4 Test report

The objects and the results of the test (as recorded in the test records) shall be summarized in a test report. The test report shall have the following structure:

1 Product identification

2 Computer systems used for testing (hardware, software and their configuration)

3 Documents used (with their identifications)

4 Results of the test of the product description, user documentation, programs and data

5 List of the non-conformities to requirements

6 Either a list of the non-conformities to recommendations, or a list of the recommendations that were not followed, or a statement that the product was not tested for conformity to recommendations

7 Date of the termination of the test

Chapter 4 of the test report (Results of the test) shall contain a statement corresponding to each headline of 3.1 to 4.2.

In addition to the statement that the product was not tested for conformity with recommendations, chapter 6 of the test report may provide a list of observed non-conformities to recommendations.

The identification of the test report (testing laboratory, product identification, date of the test report) and the total number of its pages shall appear on each page of the test report.

The test report shall include

- a statement to the effect that the test results relate only to the items tested;

- a statement that the test report shall not be reproduced except in full without the written approval of the testing laboratory [ISO/IEC Guide 25].

The test report should comply with the provisions for test reports in ISO/IEC Guide 25.

4.5 Follow up test

When a product, which has already been tested, is tested again (taking into consideration the previous test), then

- all changed parts in the documents, functions and data shall be tested as if it were a new product;

- all unchanged parts that are expected to be influenced by the changed parts or by changes in a required system (according to the specialized knowledge of the tester) shall be tested as if it were a new product;

- all other parts shall at least be tested by samples.

Annex A
(informative)

Definitions from other standards

For easy reference, definitions are quoted here for some terms used in this International Standard but defined in other standards. At the time of publication, the editions indicated were valid, and the definitions quoted from them were used or considered.

A.1 General terms

A.1.1 software: All or part of the programs, procedures, rules, and any associated documentation of an information processing system. [ISO/IEC 2382-1:1993, without the Note]

A.1.2 software package: A complete and documented set of programs supplied to several users for a generic application or function. [ISO/IEC 2382-20:1990, without the Note]

A.1.3 system software: Application-independent software that supports the running of application software. [ISO/IEC 2382-20:1990]

A.1.4 utility routine, utility program: A routine [A computer program] that provides general, frequently needed services for computer users and service personnel. [ISO 2382-7:1989, without the examples]

A.1.5 functional unit: An entity of hardware or software, or both, capable of accomplishing a specified purpose. [ISO/IEC 2382-1:1993]

A.1.6 (computer) program: A syntactic unit that conforms to the rules of a particular programming language and that is composed of declarations and statements or instructions needed to solve a certain function, task, or problem. [ISO/IEC 2382-1:1993]

A.1.7 interface: A shared boundary between two functional units, defined by functional characteristics, common physical interconnection characteristics, signal characteristics, and other characteristics, as appropriate. [ISO 2382-9:1984, without the Note]

A.1.8 user interface: An interface that enables information to be passed between a human user and hardware or software components of a computer system. [ANSI/IEEE Std 610.12-1990]

A.1.9 configuration: The manner in which the hardware and software of an information processing system are organized and interconnected. [ISO/IEC 2382-1:1993]

A.2 Characteristics of a product

A.2.1 functionality: A set of attributes that bear on the existence of a set of functions and their specified properties. The functions are those that satisfy stated or implied needs. [ISO/IEC 9126:1991, without the Notes]

A.2.2 reliability: A set of attributes that bear on the capability of software to maintain its level of performance under stated conditions for a stated period of time. [ISO/IEC 9126:1991, without the Notes]

A.2.3 usability: A set of attributes that bear on the effort needed for use, and on the individual assessment of such use, by a stated or implied set of users. [ISO/IEC 9126:1991, without the Notes]

A.2.4 efficiency: A set of attributes that bear on the relationship between the level of performance of the software and the amount of resources used, under stated conditions. [ISO/IEC 9126:1991, without the Note]

A.2.5 maintainability: A set of attributes that bear on the effort needed to make specified modifications. [ISO/IEC 9126:1991, without the Note]

A.2.6 portability: A set of attributes that bear on the ability of software to be transferred from one environment to another. [ISO/IEC 9126:1991, without the Note]

A.3 Data

A.3.1 data: A reinterpretable representation of information in a formalized manner suitable for communication, interpretation, or processing. [ISO/IEC 2382-1:1993]

A.3.2 data medium: A material in or on which data can be recorded and from which data can be retrieved. [ISO/IEC 2382-1:1993]

A.4 Testing

A.4.1 test: Technical operation that consists of the determination of one or more characteristics of a given product, process or service according to a specified procedure. [ISO/IEC Guide 2:1991]

A.4.2 test data: The data used for a check problem. [ISO 2382-8:1986]

A.4.3 check problem: A problem with a known solution used to determine whether a functional unit is operating correctly. [ISO 2382-8:1986]

A.4.4 test method: Specified technical procedure for performing a test. [ISO/IEC Guide 2:1991]

A.4.5 test plan, system test and evaluation plan: A plan that establishes detailed requirements, criteria, general methodology, responsibilities, and general planning for test and evaluation of a system. [ISO/IEC 2382-20:1990]

A.4.6 test report: Document that presents test results and other information relevant to a test. [ISO/IEC Guide 2:1991]

A.5 Other terms

A.5.1 program maintenance manual: A document that provides all the information necessary to maintain a program. [ISO/IEC 2382-20:1990]

A.5.2 system maintenance: The modification of a system to correct faults, to improve performance, or to adapt the system to a changed environment or changed requirements. [ISO/IEC 2382-20:1990]

A.5.3 work task: An intended outcome of the work system. [ISO 6385:1981]

A.5.4 work system: The work system comprises a combination of people and work equipment, acting together in the work process, to perform the work task, at the work space, in the work environment, under the conditions imposed by the work task. [ISO 6385:1981]

Annex B
(informative)

Example of a product description

The following example describes, in accordance with this International Standard, a simple imaginary software package, in order to show information that shall be present in every product description.

Product description sheet
FIREatWORK Version 2.6

FIREatWORK – screen saver and password protection

The program FIREatWORK will save your screen by displaying a stunning – and on Red-Green-Blue screens colourful – firework while you are not working with your computer. If you enter a password you will be warned if anyone has tampered with your computer in your absence.

FIREatWORK is installed in the memory. It will activate itself when you don't press any key and don't move the mouse (if any) for some (adjustable) time. It will stop as soon as you press any key or move the mouse. However, if you define a password, FIREatWORK will wait for that password to be typed.

You can define your favourite settings for

– the time FIREatWORK will wait before activating itself (1 to 999 minutes, or never);

– the number of fireworks that will explode together (1 to 19).

For this, FIREatWORK will use line dialog or a window (as your operating system does for changing system date and time).

In the same way you may define a password (6 to 45 characters). Then, if FIREatWORK stops upon typing an arbitrary character or does not stop upon typing your password, someone has interrupted FIREatWORK (e.g. by switching off the power) and restarted it without a password or with a different one.

You may produce backup copies of the program and the setting by means of your operating system. The password is not saved.

Some technical details:

– FIREatWORK runs on a Quince Hardcore 119xi personal computer (and on compatible computers) with at least 1 MB main memory and a 90 mm (3,5 in) or 130 mm (5,25 in) diskette drive for at least 720 KB. It does not need a hard disk. It supports a serial or parallel Mini-RAT mouse (or any other mouse, if compatible), but no mouse is required;

– FIREatWORK needs a graphics card: Hercules DeLuxe or PowerEGA 16+ (or any other card, if compatible);

– FIREatWORK runs under B.I.T.S 1.01 or Gnome 3.0 (or any operating system compatible with one of these two).

When ordering FIREatWORK please tell us

– whether you want the variant for B.I.T.S or the one for Gnome;

– whether you want FIREatWORK on a 90 mm (3,5 in) diskette or on a 130 mm (5,25 in) diskette.

The packet consists of the program (load module) on one diskette and a documentation booklet which includes the installation guide.

Important for you:

– You don't need any special knowledge to install or use FIREatWORK.

– Program messages and documentation are written in English.

– FIREatWORK fully complies with ISO/IEC 12119:1994 *Information technology – Software packages – Quality requirements and testing.*

Neither support for operating the product nor maintenance is offered.

Where do you get FIREatWORK:

PyroManiac Klaus P Schmidt Ltd
33 Bell Street
Bergheim, SU 53844
Telephone (022) 845 3902

Annex C
(informative)

Bibliography

ISO/IEC Guide 2:1991, *General terms and their definitions concerning standardization and related activities.*

ISO/IEC Guide 16:1978, *Code of principles on third-party certification systems and related standards.*

ISO/IEC Guide 22:1982, *Information on manufacturer's declaration of conformity with standards or other technical specifications.*

NOTE – Such a declaration is now called a "supplier's declaration" [ISO/IEC Guide 2:1991].

ISO/IEC Guide 23:1982, *Methods of indicating conformity with standards for third-party certification systems.*

ISO/IEC Guide 25:1990, *General requirements for the competence of calibration and testing laboratories.*

ISO/IEC Guide 28:1982, *General rules for a model third-party certification system for products.*

ISO/IEC Guide 40:1983, *General requirements for the acceptance of certification bodies.*

ISO/IEC Guide 44:1985, *General rules for ISO or IEC international third-party certification schemes for products.*

ISO/IEC Guide 58:1993, *Calibration and testing laboratory accreditation systems – General requirements for operation and recognition.*

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

ISO 2382-2:1976, *Data processing – Vocabulary – Part 2: Arithmetic and logic operations.*

ISO/IEC 2382-7:1989, *Information technology – Vocabulary – Part 7: Computer programming.*

ISO 2382-8:1986, *Information processing systems – Vocabulary – Part 8: Control, integrity and security.*

ISO 2382-10:1979, *Data processing – Vocabulary – Part 10: Operating techniques and facilities.*

ISO 2382-14:1978, *Data processing – Vocabulary – Part 14: Reliability, maintenance and availability.*

ISO 2382-15:1985, *Data processing – Vocabulary – Part 15: Programming languages.*

ISO/IEC 2382-20:1990, *Information technology – Vocabulary – Part 20: System development.*

ISO 6385:1981, *Ergonomic principles in the design of work systems.*

ISO 6592:1985, *Information processing, Guidelines for the documentation of computer-based application systems.*

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

ISO/IEC 9126:1991, *Information technology – Software product evaluation – Quality characteristics and guidelines for their use.*

ISO 9127:1988, *Information processing systems – User documentation and cover information for consumer software packages.*

ISO 9241-1:1992, *Ergonomic requirements for office work with visual display terminals (VDTs) – Part 1: General introduction.*

ISO 9241-2:1992, *Ergonomic requirements for office work with visual display terminals (VDTs) – Part 2: Guidance on task requirements.*

ISO 9241-10:–[1], *Ergonomic requirements for office work with visual display terminals (VDTs) – Part 10: Dialogue principles.*

ISO 9241-11:–[1], *Ergonomic requirements for office work with visual display terminals (VDTs) – Part 11: Guidance on specifying ad measuring usability.*

ISO 9241-12:–[1], *Ergonomic requirements for office work with visual display terminals (VDTs) – Part 12: Presentation of information.*

[1] To be published.

ISO 9241-13:–[2], *Ergonomic requirements for office work with visual display terminals (VDTs) – Part 13: User guidance.*

ISO 9241-14:–[2], *Ergonomic requirements for office work with visual display terminals (VDTs) – Part 14: Menu dialogues.*

ISO 9241-15:–[2], *Ergonomic requirements for office work with visual display terminals (VDTs) – Part 15: Command dialogues.*

ISO 9241-16:–[2], *Ergonomic requirements for office work with visual display terminals (VDTs) – Part 16: Direct manipulation dialogues.*

ANSI/IEEE Std 610.12-1990, *Glossary of Software Engineering Terminology.*

[2] To be published.

Alphabetical index