IEEE STANDARDS ASSOCIATION

IEEE Standard for Adoption of ISO/IEC 26514:2008 Systems and Software Engineering—Requirements for Designers and Developers of User Documentation

IEEE Computer Society

Sponsored by the
Software & Systems Engineering Standards Committee

IEEE
3 Park Avenue
New York, NY 10016-5997
USA

27 January 2011

IEEE Std 26514™-2010

IEEE Standard for Adoption of ISO/IEC 26514:2008 Systems and Software Engineering—Requirements for Designers and Developers of User Documentation

Sponsor

Software & Systems Engineering Standards Committee
of the
IEEE Computer Society

Approved 8 November 2010

IEEE-SA Standards Board

Approved 19 July 2011

American National Standards Institute

Abstract: This standard provides requirements for the design and development of software user documentation as part of the life cycle processes. It defines the documentation process from the viewpoint of the documentation developer. It also covers the documentation product. It specifies the structure, content, and format for user documentation, and also provides informative guidance for user documentation style. It is independent of the software tools that may be used to produce documentation, and applies to both printed documentation and on-screen documentation. Much of this standard is also applicable to user documentation for systems including hardware.
Keywords: information design, information development, procedures, software user documentation, user manual

The Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York, NY 10016-5997, USA

PDF: ISBN 978-0-7381-6505-9 STD97051
Print: ISBN 978-0-7381-6506-6 STDPD97051

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. The IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While the IEEE administers the process and establishes rules to promote fairness in the consensus development process, the IEEE does not independently evaluate, test, or verify the accuracy of any of the information or the soundness of any judgments contained in its standards.

Use of an IEEE Standard is wholly voluntary. The IEEE disclaims liability for any personal injury, property or other damage, of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, or reliance upon this, or any other IEEE Standard document.

The IEEE does not warrant or represent the accuracy or content of the material contained herein, and expressly disclaims any express or implied warranty, including any implied warranty of merchantability or fitness for a specific purpose, or that the use of the material contained herein is free from patent infringement. IEEE Standards documents are supplied "**AS IS**."

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, or every ten years for stabilization. When a document is more than five years old and has not been reaffirmed, or more than ten years old and has not been stabilized, 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.

In publishing and making this document available, the IEEE is not suggesting or rendering professional or other services for, or on behalf of, any person or entity. Nor is the IEEE undertaking to perform any duty owed by any other person or entity to another. Any person utilizing this, and any other IEEE Standards document, should rely upon his or her independent judgment in the exercise of reasonable care in any given circumstances or, as appropriate, seek the advice of a competent professional in determining the appropriateness of a given IEEE standard.

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 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. A statement, written or oral, that is not processed in accordance with the IEEE-SA Standards Board Operations Manual shall not be considered the official position of IEEE or any of its committees and shall not be considered to be, nor be relied upon as, a formal interpretation of the IEEE. At lectures, symposia, seminars, or educational courses, an individual presenting information on IEEE standards shall make it clear that his or her views should be considered the personal views of that individual rather than the formal position, explanation, or interpretation of the IEEE.

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. Recommendations to change the status of a stabilized standard should include a rationale as to why a revision or withdrawal is required. Comments and recommendations on standards, and requests for interpretations should be addressed to:

Secretary, IEEE-SA Standards Board
445 Hoes Lane
Piscataway, NJ 08854
USA

Introduction

This introduction is not part of IEEE Std 26514-2010, IEEE Standard for Adoption of ISO/IEC 26514:2008, Systems and Software Engineering—Requirements for Designers and Developers of User Documentation.

Well-designed documentation not only assists the user and helps to reduce the cost of training and support, but also enhances the reputation of the product, its producer, and its suppliers. This standard specifies the processes for designing and developing software user documentation, and provides the minimum requirements for these activities. It covers establishing project requirements, objectives, and constraints; audience and task analysis; user documentation design, development, and review. It is relevant to project managers, information designers and usability specialists, and information developers such as writers, editors, and illustrators. It applies to both printed and on-screen user documentation, whether produced concurrently with the software or subsequently. This standard is independent of the software tools that might be used to produce documentation. Much of its guidance is applicable to user documentation for systems including hardware as well as to software user documentation.

In addition to defining a standard process for the design and development of user documentation, this standard also covers the documentation product. This standard specifies the structure, content, and format for use documentation, and also provides informative guidance for user documentation style.

Earlier standards tended to view the results of the documentation process as a single book or multivolume set: a one-time deliverable. Increasingly, documentation designers recognize that most user documentation is now produced from managed re-use of previously developed information (single-source documentation), adapted for new software versions or presentation in various on-screen and printed media. While this standard does not describe how to set up a content management system, it is applicable for documentation organizations practicing single-source documentation.

The IEEE contributed IEEE Std 1063™-2001, IEEE Standard for Software User Documentation, as a source for this standard. IEEE Std 26514-2010 completely supersedes and replaces IEEE Std 1063-2001.

This standard is part of a series including IEEE Std 26513™-2010, IEEE Standard for Adoption of ISO/IEC 26513:2009, Systems and Software Engineering—Requirements for Testers and Reviewers of Documentation. Other standards in the 2651N series are forthcoming to address requirements for user documentation management and for acquisition and supply of user documentation. It conforms to the information management and software documentation management processes required in ISO/IEC 15288:2008 (IEEE Std 15288™-2008), *Systems and software engineering — System life cycle processes*, and ISO/IEC 12207:2008 (IEEE Std 12207™-2008), *Systems and software engineering — Software life cycle processes*.

Notice to users

Laws and regulations

Users of these documents should consult all applicable laws and regulations. Compliance with the provisions of this standard does not imply compliance to any applicable regulatory requirements. Implementers of the standard are responsible for observing or referring to the applicable regulatory requirements. IEEE does not, by the publication of its standards, intend to urge action that is not in compliance with applicable laws, and these documents may not be construed as doing so.

Copyrights

Updating of IEEE documents

Users of IEEE standards should be aware that these documents may be superseded at any time by the issuance of new editions or may be amended from time to time through the issuance of amendments, corrigenda, or errata. An official IEEE document at any point in time consists of the current edition of the document together with any amendments, corrigenda, or errata then in effect. In order to determine whether a given document is the current edition and whether it has been amended through the issuance of amendments, corrigenda, or errata, visit the IEEE Standards Association web site at http://ieeexplore.ieee.org/xpl/standards.jsp, or contact the IEEE at the address listed previously.

For more information about the IEEE Standards Association or the IEEE standards development process, visit the IEEE-SA web site at http://standards.ieee.org.

Errata

Errata, if any, for this and all other standards can be accessed at the following URL: http://standards.ieee.org/reading/ieee/updates/errata/index.html. Users are encouraged to check this URL for errata periodically.

Interpretations

Current interpretations can be accessed at the following URL: http://standards.ieee.org/reading/ieee/interp/index.html.

Patents

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 is not responsible for identifying Essential Patent Claims for which a license may be required, for conducting inquiries into the legal validity or scope of Patents Claims or determining whether any licensing terms or conditions provided in connection with submission of a Letter of Assurance, if any, or in any licensing agreements are reasonable or non-discriminatory. Users of this standard are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility. Further information may be obtained from the IEEE Standards Association.

Participants

At the time this standard was submitted to the IEEE-SA Standards Board for approval, the 26514 User Documentation Design Working Group had the following membership:

Annette Reilly, *Chair for Adoption*
James Moore, *Computer Society Liaison Representative to ISO/IEC JTC 1/SC 7*

The following members of the individual balloting committee voted on this standard. Balloters may have voted for approval, disapproval, or abstention.

Chris Bagge
Bakul Banerjee
Hugh Barrass
H Stephen Berger
Juris Borzovs
Pieter Botman
Lyle Bullock
Juan Carreon
Keith Chow
Paul Croll
Geoffrey Darnton
David Deighton
Thomas Dineen
Scott Duncan
Sourav Dutta
Kenneth Echeberry
Harriet Feldman
Andrew Fieldsend
Eva Freund
David Friscia
David Fuschi
Gregg Giesler
Randall Groves
John Harauz
George Hayhoe
Mark Henley
Frank Hill
Werner Hoelzl
Robert Holibaugh
Bernard Homes
Atsushi Ito
Mark Jaeger
Piotr Karocki
Rameshchandra Ketharaju
Thomas Kurihara
Susan Land
David Leciston
Daniel Lindberg
Greg Luri
Edward Mccall
James Moore
Finnbarr Murphy
Michael S. Newman
William Petit
Ulrich Pohl
Iulian Profir
Gerald Radack
Annette Reilly
Robert Robinson
Randall Safier
Helmut Sandmayr
Bartien Sayogo
Robert Schaaf
Maud Schlich
David J Schultz
Stephen Schwarm
Gil Shultz
Carl Singer
Steven Smith
Luca Spotorno
Thomas Starai
Walter Struppler
Marcy Stutzman
K Subrahmanyam
Thomas Tullia
David Walden
Paul Work
Oren Yuen
Janusz Zalewski

When the IEEE-SA Standards Board approved this standard on 8 November 2010, it had the following membership:

Robert M. Grow, *Chair*
Richard H. Hulett, *Vice Chair*
Steve M. Mills, *Past Chair*
Judith Gorman, *Secretary*

Karen Bartleson
Victor Berman
Ted Burse
Clint Chaplin
Andy Drozd
Alexander Gelman
Jim Hughes
Young Kyun Kim
Joseph L. Koepfinger*
John Kulick
David J. Law
Hung Ling
Oleg Logvinov
Ted Olsen
Ronald C. Petersen
Thomas Prevost
Jon Walter Rosdahl
Sam Sciacca
Mike Seavey
Curtis Siller
Don Wright

*Member Emeritus

Also included are the following nonvoting IEEE-SA Standards Board liaisons:

Satish Aggarwal, *NRC Representative*
Richard DeBlasio, *DOE Representative*
Michael Janezic, *NIST Representative*

Lisa Perry
IEEE Standards Program Manager, Document Development

Malia Zaman
IEEE Standards Program Manager, Technical Program Development

Jodi Haasz
IEEE Senior Program Manager, International Standards Programs

Contents of IEEE Adoption of ISO/IEC 26514:2008

IEEE Standard for Adoption of ISO/IEC 26514:2008 Systems and Software Engineering—Requirements for Designers and Developers of User Documentation

INTERNATIONAL STANDARD

ISO/IEC 26514

First edition
2008-06-15

Systems and software engineering — Requirements for designers and developers of user documentation

Ingénierie du logiciel et des systèmes — Exigences pour les concepteurs et les développeurs de la documentation de l'utilisateur

Reference number
ISO/IEC 26514:2008(E)

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

COPYRIGHT PROTECTED DOCUMENT

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

List of Figures

List of Tables

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

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

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

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

ISO/IEC 26514 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Software and systems engineering*.

Introduction

Anyone who uses application software needs accurate information about how the software will help the user accomplish a task. The documentation may be the first tangible item that the user sees and therefore influences the user's first impressions of the software product. If the information is supplied in a convenient form and is easy to find and understand, the user can quickly become proficient at using the product. Hence, well-designed documentation not only assists the user and helps to reduce the cost of training and support, but also enhances the reputation of the product, its producer, and its suppliers.

Although software developers aim to design user interfaces that behave so intuitively that very little separate documentation is needed, this is rarely possible. Today's software offers increasingly robust functionality, not only within applications, but also across applications that intelligently exchange information with one another. Further, most software designs include underlying rules and calculations, or algorithms, that affect the results a user can obtain when using the software. Such underlying programming mechanics are discernable by users, but only through laborious testing. For these reasons and more, user documentation remains an essential component of usable software products.

Documentation is often regarded as something done after the software has been implemented. However, for high-quality software documentation, its development should be regarded as an integral part of the software life cycle process. If done properly, documentation or information management is a big enough job to require process planning in its own right.

This International Standard was developed to assist users of ISO/IEC 15288:2008, *Systems and software engineering — System life cycle processes*, or ISO/IEC 12207:2008, *Systems and software engineering — Software life cycle processes*, to design and develop documentation as part of the software life cycle processes. It defines the documentation process from the documentation developer's standpoint.

NOTE Other International Standards in the ISO/IEC 265NN family are in preparation or planned to address the documentation and information management processes from the viewpoints of managers, assessors and testers, and acquirers and suppliers.

In addition to defining a standard process, this International Standard also covers the documentation product. This International Standard specifies the structure, content, and format for documentation, and also provides informative guidance for user documentation style.

Earlier standards tended to view the results of the documentation process as a single book or multivolume set: a one-time deliverable. Increasingly, documentation designers recognize that most user documentation is now produced from managed re-use of previously developed information (single-source documentation), adapted for new software versions or presentation in various on-screen and printed media. While this International Standard does not describe how to set up a content management system (CMS), it is applicable for documentation organizations practicing single-source documentation.

This International Standard is independent of the software tools that may be used to produce documentation, and applies to both printed documentation and on-screen documentation. Much of its guidance is applicable to user documentation for systems including hardware as well as software user documentation.

This International Standard conforms to ISO/IEC 12207:2008 as an implementation of subclause 7.2.1, Software Documentation Management Process, for software user documentation. This International Standard may be used as a conformance or a guidance document for documentation products, projects, and organizations claiming conformance to ISO/IEC 15288:2008 or to ISO/IEC 12207:2008.

The primary sources for this International Standard are previous standards IEEE Std 1063-2001, *IEEE standard for software user documentation,* and ISO/IEC 18019:2004, *Software and system engineering — Guidelines for the design and preparation of user documentation for application software.*

Systems and software engineering — Requirements for designers and developers of user documentation

1 Scope

This clause presents the scope, purpose, organization, and candidate uses of this International Standard.

This International Standard supports the interest of software users in consistent, complete, accurate, and usable documentation. It includes both approaches to standardization: a) process standards, which specify the way in which documentation products are to be developed; and b) documentation product standards, which specify the characteristics and functional requirements of the documentation.

The first part of this International Standard covers the user documentation process for designers and developers of documentation. It describes how to establish what information users need, how to determine the way in which that information should be presented to the users, and how to prepare the information and make it available. It is not limited to the design and development phase of the life cycle, but includes activities throughout the information management and documentation processes.

The second part of this International Standard provides minimum requirements for the structure, information content, and format of user documentation, including both printed and on-screen documents used in the work environment by users of systems containing software. It applies to printed user manuals, online help, tutorials, and user reference documentation.

This International Standard neither encourages nor discourages the use of either printed or electronic (on-screen) media for documentation, or of particular documentation development or management tools or methodologies.

This International Standard may be helpful for developing the following types of documentation, although it does not cover all aspects of them:

- documentation of products other than software;
- multimedia systems using animation, video, and sound;
- computer-based training (CBT) packages and specialized course materials intended primarily for use in formal training programs;
- documentation produced for installers, computer operators, or system administrators who are not end users;
- maintenance documentation describing the internal operation of systems software;
- documentation incorporated into the user interface itself.

This International Standard is applicable to documentation designers and developers, including a variety of specialists:

- information designers and architects who plan the structure and format of documentation products in a documentation set;
- usability specialists and business analysts who identify the tasks that the intended users will perform with the software;
- those who develop and edit the written content for user documentation;
- graphic designers with expertise in electronic media;
- user interface designers and ergonomics experts working together to design the presentation of the documentation on the screen.

This International Standard may also be consulted by those with other roles and interests in the documentation process:

- managers of the software development process or the documentation process;
- acquirers of documentation prepared by suppliers;
- usability testers, documentation reviewers, subject-matter experts;
- developers of tools for creating on-screen documentation;
- human-factors experts who identify principles for making documentation more accessible and easily used.

This International Standard is intended for use in all types of organizations, whether or not a dedicated documentation department is present, and may be used as a basis for local standards and procedures. Readers are assumed to have experience or knowledge of software development or documentation development processes.

Users of this International Standard should adopt a style manual for use within their own organizations to complement the guidance provided in the annexes to this International Standard, or adopt an industry-recognized style guide. Annex A provides guidance for the content of a style guide, and Annexes B and C provide guidance on style.

The order of clauses in this International Standard does not imply that the documentation should be developed in this order or presented to the user in this order.

In each clause, the requirements are media-independent, as far as possible. Requirements specific to either print or electronic media are identified as such, particularly in Clause 12. Annex D provides guidance for the design of printed documentation.

The checklists in Annex E may be used at each phase of the documentation process to check that the appropriate steps have been carried out and that the finished documentation satisfies quality criteria.

The checklists in Annexes F and G may be used to track conformance with the requirements of this International Standard for documentation processes and products.

The bibliography lists works that provide guidance on the processes of managing, preparing, and testing user documentation.

2 Conformance

This International Standard may be used as a conformance or a guidance document for projects and organizations claiming conformance to ISO/IEC 15288:2008 or ISO/IEC 12207:2008.

2.1 Application of conformance

Whether the organization or project has tailored the selected software life cycle processes or adopted them in full, the organization or project may claim conformance to this International Standard for its documentation process, for the documentation, or for both.

This International Standard is meant to be tailored so that only necessary and cost-effective requirements are applied to documentation. Tailoring may take the form of specifying approaches to conform to its mandatory requirements, or altering its non-mandatory recommendations and approaches to reflect the particular software and documentation product more explicitly. Tailoring decisions made by the acquirer should be specified in the contract.

Throughout this International Standard, “shall” is used to express a provision that is binding, “should” to express a recommendation among other possibilities, and “may” to indicate a course of action permissible within the limits of this International Standard. When using this International Standard as a guide, replace the term “shall” with “should”. Use of the nomenclature of this International Standard for the parts of user documentation (that is, chapters, topics, pages, screens, windows) is not required to claim conformance.

NOTE All “shall” statements are listed in Annex F and Annex G.

2.2 Conformance situations

Conformance of user documentation may be interpreted differently for various situations. The relevant situation shall be identified in the claim of conformance:

1) When conformance is claimed for an organization, the organization shall make public a document declaring its tailoring of the life cycle process.

   NOTE 1 One possible way for an organization to deal with clauses that cite “the documentation plan” is to specify that they shall be interpreted in the project plans for any particular documentation project.

2) When conformance is claimed for a project, the project plans or the contract shall document the tailoring of the documentation requirements.

   NOTE 2 A project’s claim of conformance is typically specified with respect to the organization’s claim of conformance.

3) When conformance is claimed for a multi-supplier program, it may be the case that no individual project may claim conformance because no single contract calls for all the required activities. Nevertheless, the program, as a whole, may claim conformance if each of the required activities is produced by an identified party. The program plans shall document the tailoring of the required tasks, and their assignment to the various parties, as well as the interpretation of clauses of this International Standard that reference “the contract”.

4) When conformance is claimed for documentation products, the organization or project should specify whether conformance applies to a single document, a documentation set, or all user documentation produced through the organization's content management processes.

This International Standard may be included or referenced in contracts or similar agreements when the parties (called the acquirer and the producer or supplier) agree that the supplier will deliver documentation in accordance with this International Standard. It may also be adopted as an in-house standard by a project or organization that decides to produce documentation in accordance with this International Standard.

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 12207:2008, *Systems and software engineering — Software life cycle processes*

ISO/IEC 15288:2008, *Systems and software engineering — System life cycle processes*

IEEE Std 100-2000, *The Authoritative Dictionary of IEEE Standards Terms,* Seventh Edition

4 Terms and definitions

For the purposes of this document, the terms and definitions given in IEEE Std 100-2000 and the following apply.

NOTE Throughout this International Standard, the term “documentation” refers to software user documentation. Use of the terminology in this International Standard is for ease of reference and is not mandatory for conformance with this International Standard.

4.1
accessibility
usability of a software or documentation product, service, environment, or facility by people with the widest range of capabilities

NOTE 1 Adapted from ISO 9241-171.

NOTE 2 Although “accessibility” typically addresses users who have disabilities, the concept is not limited to disability issues.

4.2
action
element of a step that a user performs during a procedure

4.3
active area
⟨on-screen documentation⟩ area that responds to user input

NOTE A hot-spot on a graphic and a link in text are examples of active areas.

4.4
analysis
investigation and collection phase of development that aims to specify types of users and their informational needs

4.5
application software
software designed to help users perform particular tasks or handle particular types of problems, as distinct from software that controls the computer itself

cf. **software**

4.6
audience
category of users sharing the same or similar characteristics and needs (for example, reason for using the documentation, tasks, education level, abilities, training, experience)

NOTE There may be different audiences for documentation (for example, management, data entry, maintenance) that determine the content, structure, and use of the intended documentation.

4.7
caution
advisory information in documentation that states that performing some action may lead to consequences that are unwanted or undefined, such as loss of data or an equipment problem

cf. **warning**

4.8
change control procedure
actions taken to identify, document, review, and authorize changes to a software or documentation product that is being developed

NOTE The procedures ensure that the validity of changes is confirmed, that the effects on other items are examined, and that those people concerned with the development are notified of the changes.

4.9
configuration management
CM
technical and organizational activities comprising configuration identification, control, status accounting, and auditing

NOTE See ISO 10007:2003, *Quality management systems — Guidelines for configuration management*.

4.10
context-sensitive help
type of on-screen documentation in which the information that is displayed depends upon the user's view of the software

cf. **embedded documentation**, **printed documentation**

4.11
critical information
information describing the safe use of the software, the security of the information created with the software, or the protection of the sensitive personal information created by or stored with the software

4.12
customization
adaptation of a software or documentation product to the needs of a particular audience

4.13
design
stage of documentation development that is concerned with determining what documentation will be provided in a product and what the nature of the documentation will be

4.14
development
activity of preparing documentation after it has been designed

4.15
document
separately identified piece of documentation which could be part of a documentation set

4.16
documentation
information that explains how to use a software product

NOTE 1 It can be provided as separate documentation or as embedded documentation or both.

NOTE 2 In this International Standard, the term “documentation” is synonymous with the terms “user documentation” and “software user documentation”. Other forms of documentation (for example, “system documentation”) are clearly identified as such.

NOTE 3 Printed manuals, on-screen information and stand-alone online help are examples of documentation.

4.17
document set
collection of documentation that has been segmented into separately identified volumes or files for ease of distribution or use

4.18
embedded documentation
documentation that is accessed as an integral part of software

cf. **separate documentation**

NOTE Pop-up help and help text on a screen are examples of embedded documentation.

4.19
entry field
area on a screen or in a window in which a user enters data

4.20
escrow
source code and documentation that is kept in the custody of a third party until specified contractual conditions have been fulfilled

4.21
function
part of an application that provides facilities for users to carry out their tasks

4.22
icon
graphic displayed on the screen that represents a function of the computer system

NOTE Adapted from ISO/IEC 11581-1:2000, 4.7.

4.23
illustration
graphic set apart from the main body of text and normally cited within the main text

NOTE In this International Standard, the term “illustration” is used as the generic term for tables, figures, exhibits, screen captures, flow charts, diagrams, drawings, icons, and other types of graphics.

4.24
implementation
phase of development during which user documentation is created according to the design, tested, and revised

4.25
instructional mode
usage mode that is intended to teach the use of software in performing tasks

4.26
internationalization
process of developing information so that it is suitable for an international audience

cf. **localization**

4.27
link
⟨on-screen documentation⟩ active area that displays either a new topic or a different part of the current topic

4.28
localization
creation of a national or specific regional version of a product

cf. **internationalization**

NOTE Localization may be carried out separately from the translation process.

4.29
menu
⟨on-screen documentation⟩ list of topics from which the user may choose

4.30
navigation
act of accessing documentation and viewing different topics

4.31
on-screen documentation
documentation that is intended to be read on the screen by the user while using the software

cf. **printed documentation**, **embedded documentation**

NOTE Pop-up help and help text on a screen are examples of on-screen documentation.

4.32
picture
illustration that shows the actual appearance of physical objects

NOTE Photographs and drawings are examples of pictures.

4.33
printed documentation
documentation that is either provided in printed form, or provided in electronic form for the customer or user to print

4.34
procedure
ordered series of steps that specify how to perform a task

4.35
process
set of interrelated or interacting activities which transforms inputs into outputs

[ISO 9000:2005]

4.36
product
complete set of software and documentation

4.37
product authority
person or persons with overall responsibility for the capabilities and quality of a product

4.38
project
set of activities for developing a new product or enhancing an existing product

4.39
project manager
person with overall responsibility for the management and running of a project

4.40
quality management
coordinated activities to direct and control an organization with regard to quality

[ISO 9000:2005]

4.41
real-world object
entity that exists in a three-dimensional form and, by association, implies similar properties or behavior to software functions

NOTE Printer, filing cabinet, file folder and sheet of paper are examples of real-world objects.

4.42
reference mode
usage mode that is intended to provide quick access to specific information for software users who are generally familiar with the software functions

4.43
secondary window
window that contains information that depends on information in another window (the primary window)

NOTE The information in the secondary window supplements the information in the primary window.

4.44
separate documentation
documentation that may be used independently of the software

cf. **embedded documentation**

NOTE Printed manuals and freestanding hypertext systems are examples of separate documentation.

4.45
signpost
text, symbol, or small graphic that helps the user identify where particular types of information are located or where the information in the current display fits into the whole document

NOTE Information of different types may be indicated by symbols or graphics of different types.

4.46
software
program or set of programs used to run a computer

cf. **application software**

NOTE For the purposes of this International Standard, the term “software” does not include on-screen documentation.

4.47
step
one element (numbered list item) in a procedure that tells a user to perform an action (or actions)

NOTE Responses by the software are not considered to be steps.

4.48
style
set of language-specific editorial conventions covering grammar, terminology, punctuation, capitalization, and word choice of documentation

4.49
technical contact
person responsible for providing a documentation developer with technical information about a software product or for checking the technical accuracy of drafts of documentation; also called subject-matter expert

4.50
topic
small part of a document that deals with a single subject

NOTE 1 In printed documentation, a topic is equivalent to a section (heading; subheading) and its content. In on-screen documentation, a topic consists of a title (heading) and information about a subject (typically, a task or a concept or reference information).

NOTE 2 For on-screen documentation, the system may present a topic without user intervention.

NOTE 3 Instructions on how to print the current document are an example of a topic.

4.51
tutorial
instructional mode documentation in which the user exercises software functions using sample data that is supplied with the software or documentation

4.52
usability
extent to which a software or documentation product may be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use

NOTE Adapted from ISO 9241-11:1998, 3.1.

4.53
usage mode
primary manner in which the documentation developer expects the document to be used

NOTE This International Standard recognizes two usage modes: instructional and reference.

4.54
user
person who performs one or more tasks with software; a member of a specific audience

4.55
user documentation
information to describe, explain, or instruct how to use software

cf. **documentation**

NOTE Printed user manuals, embedded on-screen information, and help are examples of user documentation.

4.56
user interface
ensemble of software and hardware that allows a user to interact with a computer system

4.57
warning
advisory information in documentation that states that performing some action may lead to serious or dangerous consequences

cf. **caution**

4.58
window
area with visible boundaries that presents a view of a software object or through which a user conducts a dialog with a computer system

4.59
wizard
procedural form of help that guides a user through each step of a task through dialog with the user

5 User documentation process within the systems/software life cycle

This International Standard covers the phases involved in designing, specifying, and producing user documentation.

Designers and developers of user documentation work within the life cycle processes of the software product, which is defined in ISO/IEC 12207:2008 and includes the management of software documentation within the software support processes (*Clause 7.2.1*).

Thus, development of the user documentation should be part of the same processes as the software product life cycle, and ideally performed in conjunction with development of the software, so that the software and the user documentation may be tested, distributed, and maintained together. The specification of all the documentation, including on-screen documentation and printed documentation, should be a part of the development of the software product as a whole, not a separate exercise. Although accurate user documentation cannot be completed until the software product has been fully developed, the user documentation and the software product both benefit from concurrent development.

The classic documentation development process applies to the life cycle of a single new software product with a single new user manual, but it is much more likely that software and user documentation are designed and developed under more complex circumstances, such as the following:

1) A previously documented software product is being upgraded, offered in a new version or on different operating system platforms, or customized as part of system integration, and previous documentation must be revised.

2) Previous documentation must be converted to a different format or different media, or in different languages or versions, such as tutorials, online help, or advanced reference guides.

3) Previous documentation must be adapted or used as models for different software products acquired or supplied by an organization.

Designing and developing user documentation is greatly assisted by the presence of other documentation produced during the software life cycle, such as a documentation plan, system design document, system test plan, release records, and problem reports. Other documentation specific to the documentation process may be produced, such as style guides and organizational procedures for content management and documentation reviews.

NOTE: ISO/IEC 15289:2006, *Systems and software engineering — Content of systems and software life cycle process information products (Documentation)*, provides recommended contents for the required documents throughout the systems/software life cycle.

Thus, rather than focusing on the production of a single book or help system, user documentation designers should regard their tasks as part of the information management process, whose activities are defined in ISO/IEC 15288:2008, 6.3.6.3:

Information Management Process

a) **Plan Information Management.** This activity consists of the following tasks:

1) Define the items of information that will be managed during the system life cycle and, according to organizational policy, agreements, or legislation, maintained for a defined period beyond.

2) Designate authorities and responsibilities regarding the origination, generation, capture, archiving and disposal of items of information.

3) Define the rights, obligations and commitments regarding the retention of, transmission of and access to information items.

NOTE Due regard is paid to information and data legislation, security and privacy, e.g. ownership, agreement restrictions, rights of access, intellectual property and patents. Where restrictions or constraints apply, information is identified accordingly.

Staff having knowledge of such items of information are informed of their obligations and responsibilities.

4) Define the content, semantics, formats and medium for the representation, retention, transmission and retrieval of information.

NOTE The information may originate and may terminate in any form (e.g. verbal, textual, graphical, numerical) and may be stored, processed, replicated and transmitted using any medium (e.g. electronic, printed, magnetic, optical). Pay due regard to organization constraints, e.g. infrastructure, inter-organizational communications, distributed project working. Relevant information storage, transformation, transmission and presentation standards and conventions are used according to policy, agreements and legislation constraints.

5) Define information maintenance actions.

NOTE This includes status reviews of stored information for integrity, validity and availability and any needs for replication or transformation to an alternative medium. Consider the need to either retain infrastructure as technology changes so that archived media can be read or the need to re-record archived media using newer technology.

b) **Perform Information Management.** This activity consists of the following tasks:

1) Obtain the identified items of information.

NOTE This may include generating the information or collecting it from appropriate sources.

2) Maintain information items and their storage records according to integrity, security and privacy requirements.

NOTE Record the status of information items, e.g. version description, record of distribution, security classification. Information should be legible and stored and retained in such a way that it is readily retrievable in facilities that provide a suitable environment, and that prevent damage, deterioration and loss.

3) Retrieve and distribute information to designated parties as required by agreed schedules or defined circumstances.

NOTE Information is provided to designated parties in an appropriate form.

4) Provide official documentation as required.

NOTE Examples of official documentation are certification, accreditation, license and assessment ratings.

5) Archive designated information, in accordance with the audit, knowledge retention, and project closure purposes.

NOTE Select the media, location and protection of the information in accordance with the specified storage and retrieval periods, and with organization policy, agreements and legislation. Ensure arrangements are in place to retain necessary documentation after project closure.

6) Dispose of unwanted, invalid or unverifiable information according to organization policy, and security and privacy requirements.

The following sections include requirements and guidance for the process phases:

- process implementation:
    - setting objectives: Identifying goals, policies, and constraints, such as enterprise, project, and software product (internal) standards that will affect the document design;
    - project planning, management, and control.

NOTE: Process implementation may involve use of the acquisition and supply process when the documentation is designed and developed by a different organization than the software.

- development and review:
    - analysis and design: Drawing up the documentation designs for the project: collecting information about the software product and users, their tasks, and their needs for information, and designing documentation based on those needs;
    - development and review: Structuring the content for usability, articulating the documentation design by creating the written and graphic content, implementing the information in the chosen media, and evaluating the user documentation with the rest of the product;
- production: Packaging and making the documentation available to the users through printed or on-screen media; maintaining configuration control of master files and released versions;
- maintenance: Keeping the documentation accurate throughout the software product life cycle, including adaptations for improved usability.

For simplicity this International Standard describes the life cycle as if there were a clear starting point for developing documentation and a clear end point. There is not a single sequence of activities, however, that can be followed in all cases for all products and all types of information. For example, design and implementation activities for on-screen documentation are often closely inter-linked, as are analysis and design, and the way they link together varies between projects.

6 Project requirements, objectives, and constraints

The user documentation developer shall gather or receive information about the wider context of the whole project, to understand the requirements that affect the design for the documentation components. The purpose of the process implementation (initiation) phase is to understand the project objectives, requirements, and constraints, particularly:

- the requirements and objectives for the software product;
- the requirements and constraints for the user documentation, such as the documentation policy and standard formats and styles set by the producer of the software product;
- the constraints that may apply to the project cost, schedule, staffing, and equipment;
- the intended acquirer and end-users for the software product and the documentation, whether through contract or commercially offered to the public;
- the usability requirements.

If the stated requirements limit the design options so that users cannot be given a suitable set of documentation, the documentation designer should:

- question the requirements, explaining the reasons for the discrepancy;
- suggest alternative solutions.

The organization should keep a record of the source of each requirement, so that it may be tracked back to its origins and its validity may be reaffirmed.

6.1 Project objectives

Ideally, the project objective should be to develop a software product that satisfies user expectations and meets user needs. Documentation developers need the following information:

- Who are the users and in what contexts they will use the product?
- Are there users with special needs?
- What is the purpose of the software product? What will it do?
- Is there a previous version? If so, which features have to be changed and which have to remain the same?
- Is the product stand-alone, or is it part of a suite of products?
- When will the product be available to document? (The product schedule may help determine the types and amount of documentation.)
- Are there plans for future versions?
- On what platforms will the product run at this release? Are there plans for other platforms later?
- Is the product being developed for a specific organization or organizations? If so, will those organizations be the exclusive users?
- Will localized or customized versions of the product be required?

The project shall maintain information on the standards the software product is required to follow, such as:

- international standards (such as ISO) publications;
- national standards for the countries in which the product will be used;
- industry standards for the system on which the product will run;
- industry standards for the system on which the on-screen documentation will be viewed;
- accessibility standards and requirements;
- company, product, or operating system standards and conventions;
- restrictions on copying or modifying the software and its documentation;
- the protection of sensitive information.

6.2 User documentation requirements and constraints

The following information should be considered in the documentation plan:

- Does the organization have a formal documentation or information management policy and procedures, and if so, how will they be applied to this project?
- What metadata is required for each piece of documentation to support document management, indexing, and searchability?
- Are there organizational standard formats, templates, styles, or systems? Are they consistent with the requirements and recommendations in this International Standard? If they do not exist, they should be established consistent with this International Standard;
- Are there constraints on required content?
- Is there a need to provide background or conceptual information, such as the working procedures of the users' organizations or information about the software product itself? If so, is that information readily available?
- Are there resources for supplying the documentation developer with technical information about the product, such as technical contacts, written specifications, and the product itself (possibly development or prototype versions)?

Documentation may have to satisfy legal and regulatory requirements. The documentation designer and developer should consider the following and take legal advice if necessary:

- requirements set by national legislation;
- copyright status of the document itself;
- copyright issues for text included in the documentation from elsewhere;
- protection of sensitive data;
- acknowledgments;
- trademarks;
- escrow conditions;
- licensing;
- any presentation requirements that apply (such as use of special logotypes on the packaging and in the documentation to identify proprietary platform-compatible documentation products);
- intellectual property rights;
- warranties and guarantees.

In terms of security, the documentation developer should determine whether it is necessary to control the integrity of on-screen documentation or allow for deliberate or accidental changes to its content made by the user.

The supplier and client should also determine how the documentation will be maintained and updated.

6.3 Project goals and constraints

The business goals of a project depend on whether the software product is produced under contract or for general commercial sale. The important features of the software product and documentation's sales objectives are:

- the target number of copies of the software product to be distributed over its projected lifetime;
- the expected profile of sales over time (will demand be steady or will there be a period of peak demand?).

The documentation designer and developer should review plans for making modified versions of the software product available to users because modifications may affect the documentation design. The plans for future modifications of the application may be identified using the following questions:

- At what intervals will the software product and documentation be modified?
- What different levels of modification will there be, for example, for issuing:
    - temporary corrections;
    - interim versions;
    - major upgrades or substantially new versions.

- What are the time scales for making modifications?
- What are the cost constraints for maintenance?
- How will the documentation be updated?
- Will previous versions be maintained or removed from support?
- How will amendments be made available to existing users?

6.3.1 Project infrastructure and tools

Project constraints may include infrastructure systems and tools, schedules, cost, and staffing.

If a new software product is part of a suite of software products, tools already specified for developing, maintaining, delivering, and viewing documentation for that suite should be considered. If the customer or organization for which the software product is being developed has existing systems with which they want the new software product to be consistent, the documentation designer and developer should consider whether these existing systems dictate the use of particular documentation delivery and viewing mechanisms.

Constraints for the documentation should not be allowed to restrict the software product from handling the real problems that it should solve, and the range of hardware and operating systems for which it is being developed. The documentation designer should obtain information on the capabilities and limitations of the planned or required tools for:

- using the software, so documentation developers may find out how it operates and how users will have to use it;
- writing, laying out, and editing the text;
- writing on-screen information and compiling help files;
- drawing the illustrations;
- reviewing the documentation;
- testing the documentation;
- conducting user tests, including tests of accessibility.

6.3.2 Schedule constraints

Designers and developers need to know the following about the software product development schedule:

- When will the alpha, beta, and acceptance testing start and be completed?
- If there will be multiple releases, is there a phased or packaged approach to the releases? What features/functions will be delivered in what phase?
- Are there work processes that need to be defined to change the way business is currently being done?
- Will the software product be submitted for government certification and accreditation?
- What is the delivery date for the finished software product, and is there a worldwide release on the same day or a different release date for each region?
- How long before the delivery date does the documentation need to be ready?
- What other milestones apply to the project, such as dates for early releases?
- What are the major dependencies between different activities in the overall project?
- If the product is to be localized or customized, what are the required delivery dates for the various localized and customized versions and how long before those delivery dates does the documentation need to be ready?
- Will the documentation be translated and what is the translation schedule, including lead time, drop-off dates to translation, and delivery dates from translation?
- Will the documentation need to pass security or legal reviews prior to delivery?
- Will the documentation be printed locally but shipped internationally, and if so, what are the shipping time estimates to build into the documentation schedule to allow for physical delivery?
- If the documentation product is to be sent to a printer or published on removable media, what is the lead time required?

6.4 Users and usability objectives

Designers should be aware of usability requirements from the beginning of the project. From the start, usability should be built into the structure, content, and format of the documentation and supporting software; it cannot be added at the end of documentation development. Therefore, the usability requirements and the method of testing them shall be specified in the analysis phase when other user needs are being determined.

The usability of the documentation is an integral part of the usability of the software product. When the usability targets are set and measured for the software product, the documentation shall be treated as an integral part of the software product. The documentation developers and software developers should work together to ensure usability. Measures of the usability of the documentation, independent of the usability of the software, shall include the following:

- the time taken to learn how to do a specified task using the instructions in the documentation;
- the ability of the user to accomplish the task at hand using the documentation.

There are two aspects to ensuring that the documentation is usable:

- Applying human-centered and software ergonomics principles to the design. ISO 13407:1999, *Human-centred design processes for interactive systems*, and the ISO 9241 series [ISO 9241-1 to ISO 9241-17, *Ergonomic requirements for office work with visual display terminals (VDTs)*; ISO 9241-110:2006, *Ergonomics of human-system interaction — Part 110: Dialogue principles*; and ISO 9241-171, *Ergonomics of human-system interaction — Part 171: Guidance on software accessibility*] address such design advice.

- Evaluating the design and the software product to assess how usable the software product will be in practice. ISO 13407 addresses such design advice.

Quantitative tests measure qualities such as task completion time and accuracy, while qualitative tests measure user attitudes and preferences.

The combination of quantitative and qualitative measurements selected for a specific usability test will depend on the purpose of the particular test and the point in the life cycle when the test occurs.

Once an analysis of usability requirements has been made and the appropriate testing methods have been selected, the documentation designer and developer should translate those requirements into usability goals for the documentation. The goals are the qualitative or quantitative targets, which are measured in usability tests. Figure 1 illustrates the process in the context of an electronic mail system.

Step 1. Define the user's objectives.

The user's objectives are:

- to find in the help menu the instructions for sending a message;
- to summarize the task in the user's own words;
- to send the message.

Step 2. Define the usability measures for those objectives.

The measures are:

- effectiveness (Is the right information found?);
- efficiency (How long does it take to find the information? Is the shortest search route or method used? Does the help text have to be re-read to be understood or remembered?);
- satisfaction (What are the user's attitudes towards the help?).

Step 3. Define acceptance criteria.

The criteria are:

- If the user finds the information within 20 seconds, then the structure and navigation are acceptable.
- If the user's summary of the task is correct, then the information is accurate and suitably clear.
- If the task could be performed correctly on the first try while following the instructions, then the help was acceptable, task-oriented, and complete.

Figure 1 — Sample process of defining the usability goals for an electronic mail system

Accessibility requirements extend usability requirements to ensure that the documentation is provided in suitable format for all those who may use the software product. The documentation should be provided in media and formats that allow its use by those with vision, hearing, or other physical limitation if they are able to use the software and to perform the tasks supported by the software.

EXAMPLE: The US government has published specific requirements for software accessibility, known as Section 508 of the Rehabilitation Act. Details of Section 508 may be found at the URL listed in *Annex H*.

Internationalization and national cultural requirements have a significant impact on the design of documentation, including schedules, costs, presentation formats, writing style, and usability testing. The designer should determine whether the application will be used by an international audience and in which countries. The answers to these questions will determine the translation requirements.

For software products to be made available in other countries using the same language as the source country, the documentation designer and developer should consider whether special localized versions of the documentation should be prepared for those countries, using appropriate local language, dialects, or conventions. Cultural issues should be considered, both in the software product and in the documentation, especially in examples. A national of the target country should check the user documentation to ensure that it is suitable for use in that country.

6.5 Interviewing technical contacts and other experts

Document designers and developers should conduct formal interviews as part of the process of gathering information and reaching consensus on design decisions. Interviews may be conducted through in-person meetings, telephone, instant messaging, e-mail, audio- or video-conferencing, Internet-based real-time chat tools, and other communication media used by the project team. Documentation designers and developers should first read the available software life cycle design documents, and then plan and conduct formal interviews with technical contacts and other experts to resolve questions. Interviews, whether conducted individually or in groups, should be scheduled as far in advance as practical. Frequent, unstructured, unscheduled meetings or questions may be perceived as interruptions by other team members. Documentation designers and developers should begin planning for interviews by identifying the information that they need to collect, along with the name or role of the persons or groups most likely able to provide the information, and then should carefully develop questions that will elicit the answers needed.

Group interviews often produce more information and resolve discrepancies in assumptions and approaches among members of the project team better than a series of individual interviews with the same persons. Documentation designers and developers should balance the schedules and time requirements of participants in determining which interview technique is best for a particular project. During the interview, no matter what medium is used to conduct it, those participating should have access to documents, prototypes, or software to reference or otherwise elicit information. Decisions and information gained from interviews and meetings should be published in meeting minutes to those participating and project managers, so that miscommunications may be promptly corrected and any continuing information gaps may be identified.

6.6 Project planning

The objective of documentation project planning is to identify the deliverable documents, specify the quality and usability standards they will meet, and define the project tasks, activities, schedules, resources, and cost. This clause describes factors in planning and controlling documentation projects from the viewpoint of the designers, writers, editors, artists, software developers, and others involved in user documentation development. Documentation designers and developers provide input to project managers for realistic configuration management, schedules, and cost estimates.

Documentation designers and developers should participate in meetings at which the requirements, objectives, and constraints of the project are identified and discussed.

NOTE: Documentation process planning and management will be addressed in a future International Standard (ISO/IEC 26511, *Software and systems engineering — User documentation requirements for managers*), the standard for the management of user documentation, which should be consulted by those responsible for developing documentation plans, controlling documentation development, or managing the content of user documentation so that it may be more easily reused.

The results of the documentation planning process shall be recorded in a documentation plan covering the documentation to be produced, and including individual plans (outlines or specifications) for each separate document item. The detailed documentation plans should be thoroughly reviewed along with the project plan. The documentation plan shall assign responsibility and authority for each major activity:

- technical accuracy of the documentation;
- usability of the documentation with the software product and vice versa;
- suitability of the documentation for the market;
- editorial quality of the documentation, including both text and illustrations;
- conformance to legal requirements;
- production of the documentation, including tools support;
- translation, localization, or customization;
- packaging and delivery of the software and the documentation;
- final approval of the documentation.

The audiences for the document plans are:

- the writing team;
- the reviewers;
- the people doing the analysis and planning;
- the product authority;
- the project managers who will be supervising the document development;
- customers who may need to know what the documentation will look like when it is complete;
- the people doing the translation, production, and usability testing;
- others in the organization responsible for marketing, training, and customer support.

The documentation plan should be approved with the rest of the product and shall be under change control procedures with the rest of the project plans.

NOTE: It is more difficult and more expensive to make major alterations to document structures or styles once development has started than it is to alter the design at the planning stage.

6.6.1 Quality management

The documentation development activities recommended in this International Standard should be carried out under the control of the quality management system (QMS) being used for the software product development. Users of this International Standard are recommended to operate a quality management system, which may be independently assessed for ISO 9000 compliance.

6.6.2 Version control and change control

A configuration management (CM) process shall be used to control the user documentation. The document designer should consider the CM policies that may influence the documentation design and affect its maintainability. These policies may include:

- the level at which version control is to apply; for example, whether it applies to a single file of information or to the information for a particular module, topic, object, or task;
- the landmarks within the project at which new versions are to be created; for example, after each set of tests;
- the method for controlling changes to each version;
- the method for applying version control to localized and customized versions;
- the back-up and archive procedures;
- the method for keeping records about the history of each version, to enable recreation of old versions if necessary;
- at what stage, if any, the documentation becomes part of the software for version control purposes.

Change control procedures for the project shall take account of the requirements of the documentation activities. Documentation developers should be members of any body responsible for approving changes. Documentation may be greatly affected by changes to the design of the software or the training or support plans. Therefore, the implications of such changes should be assessed before approval is given. When software product changes are made, documentation developers should be informed promptly, so that they work only with up-to-date information.

NOTE: Small changes to a system may cause major changes to documentation, although major changes to a system do not always require major changes to the documentation. For example, altering a system menu might be a small change to the software itself but might have major implications for the entire structure of a user manual. In contrast, completely revising the methods used within the software for some complex operation might be a major task in developing the software, but might not affect the user's view of the software at all, and therefore might require no changes to the documentation.

The same CM system used for the software development may support the documentation. This CM system will usually have some features appropriate for the documentation, particularly the on-screen elements.

NOTE: ISO/IEC TR 15846:1998, *Information technology — Software life cycle processes – Configuration management*, provides guidance on configuration management systems.

6.6.3 Availability of resources

During project planning, the designer and project manager should determine the availability of the resources needed for developing the documentation. The plan should include the time and effort to include acquisition of resources or services that are not already available in the organization. Resources include facilities, services, tools, and human resources, such as:

- the hardware and software for using a prototype or other versions of the software;
- a prototype or other versions of the software to be used by documentation developers to obtain information about the software being documented;
- the hardware and software for producing the documentation itself, for content management and storing master copies, and for translations;
- a laboratory for usability testing, and suitable space for other testing such as unit testing and functional testing;
- technical contacts who will supply information, discuss technical details with documentation developers, answer technical questions, and check drafts of the documentation;
- documentation developers or other staff who will editorially review drafts of the documentation;
- graphic designers and illustrators;
- printers and packaging vendors;
- usability staff;
- legal reviewers or contacts;
- translators.

6.6.4 Schedules

During the project implementation phase, the organization shall prepare a preliminary schedule for documentation activities. More detailed schedules for developing each separate document, as part of a master schedule for other parts of the project, shall be produced during the design and development phase. Documentation designers and developers, software developers, and the project manager should agree on the overall schedules for the project. Once the schedules are agreed upon, the team is committed to design documentation that may be prepared within the required time and cost.

When the project schedules are being defined, the documentation designer and developer should:

- remember that documentation development cannot be finished until after the software design is frozen;
- remember that the amount of time needed for the following activities may be significant and make adequate allowance for them in the project's schedules:
    - observing and using the software or prototypes of the software during the development stage;
    - learning how to use new tools;
    - obtaining information from technical contacts and having them check the accuracy of drafts;
    - creating the graphics and screen captures;
    - making draft documentation available for software product validation, field trials, and usability trials (documentation development schedules for preparing such drafts will affect the timing of these exercises and trials);
    - for each activity, particularly system tests, reviews, edits, and user tests, incorporating the results of the activities in corrected documentation; with subsequent retests and further revisions of the software and the documentation;
    - identifying technical work necessary to include the embedded documentation in the software product;

- obtaining legal reviews;
- having the documentation translated, if required;
- having the documentation printed and packaged, if required.

Once a date for delivering the application has been agreed upon, it should be applied in all phases of the planning. Changes in the software product and in planned delivery dates shall be promptly communicated to those concerned, and the impact of these changes on the documentation shall be evaluated and communicated to project management. If schedules need to be adjusted during the project, activities such as system tests, reviews, and usability tests should not be removed from the schedule to save time at the expense of quality.

6.6.5 Cost estimating

Documentation developers should provide input for estimates of the cost of developing the proposed documentation by including where relevant:

- documentation developer costs for development and for reviewing drafts;
- technical staff costs for briefing documentation developers and reviewing drafts;
- illustration costs;
- project management costs;
- editorial costs;
- tools support costs;
- training costs;
- expenses for support staff;
- equipment costs;
- materials costs;
- bulk copying and production costs (for example, copying and packaging media);
- costs of system validation tests, reviews, and user tests;
- evaluation costs;
- maintenance costs;
- costs of production, printing, delivery, or distribution to customers or users;
- translation costs for each language, which may include fees, travel, and communications.

NOTE 1: If documentation design is not taken into account in the design of the software product, the documentation might not be adequately concise, and hence documentation development costs might be higher than necessary.

NOTE 2: Similarly, if the product is to be translated or localized and those costs are not taken into account in the design of the software and the documentation, they may be significantly higher than necessary. The higher costs will be incurred for every translated or localized version.

NOTE 3: If future versions of the product are planned but not considered in the design of the documentation, the costs may be significantly higher. For example, if the document designer knows from the outset that a basic and enterprise

version of a product may be released, they may use publishing features such as variables for software product names and use those throughout the document during the original creation. Later, changing a single variable value will be much faster than searching for and replacing long and short software product names in one or more documents.

The documentation developer may assist in obtaining estimates of the costs of carrying out documentation development and production activities for a variety of documentation types to assist in the negotiation of documentation cost constraints.

6.6.6 Planning for localization and customization

The same factors considered in planning for the documentation project as a whole should be applied in planning localized or customized versions:

- Who is responsible for the process?
- Who will prepare the instructions for localization or customization, and when?
- Where will the localization or customization be carried out, and by whom?
- When and how will the product be delivered to the staff, and who will carry out the work?
- How will the localized and customized versions be tested?
- When and how will the localized or customized versions be delivered?
- Who is responsible for the quality of the localized or customized versions?
- Who is responsible for the usability of the localized or customized versions?

6.7 Documentation proposal

When documentation is to be produced by a separate organization (internal or external), the outcome of the objectives and planning activities may be a documentation proposal from the supplier to the acquirer. Figure 2 shows a sample contents list for a documentation proposal.

Documentation proposal contents

1. Executive summary

2. Overview of the objectives and requirements *(This section sets out the requirements specified for the project and briefly explains how they will be met.)*

- The product to be documented; future modifications; customization; security and safety concerns
- Target audiences; Internationalization and localization; accessibility concerns
- Critical tasks to be documented
- Legal requirements (including intellectual property issues)
- Organizational policies, procedures, and development tools affecting the project

3. Overview of the documentation to be provided

- Preliminary high-level design of the deliverable documentation
- International Standards and conventions to be followed
- Documentation delivery and viewing constraints

4. **Documentation development and review processes** *(This section may include separate sections for localization, translation, or customization projects.)*

- Process overview
- Acquirer's resources and responsibilities
- Supplier's resources and responsibilities
- Preliminary schedule
- Tools and systems to be used
- Project risks

5. Business proposal

- Basis of estimate (comparison of the project to previous schedules, costs, and resources)
- Ownership of intellectual property and delivered documentation
- Cost and payment terms

Figure 2 — Sample contents list for a documentation proposal

7 Analysis and design

Analysis and design are the first tasks in the documentation design and development process. Analysis is the process of developing detailed requirements and acceptance criteria for the documentation. Documentation shall be based on audience and task analyses. Taking these analyses into consideration, design is the process of specifying the structure, content, format, and style to be used in the documentation.

7.1 Audience and task analysis

7.1.1 Audience analysis

The designer shall list the intended types of users of the software product and classify users into audiences. Each audience consists of a group of users whose work tasks and use of the application are similar. The classification should consider:

- the users' background, experience, and training;
- the language familiar to the users;
- the way that the application will be used;
- the users' learning stages (for example, how much experience they have with this application);
- the frequency of use (some users use an application or feature frequently, some infrequently);
- the users' working environments (do the users do most of their work at a desk, or up a telephone pole?).

NOTE: An analysis of the tasks performed by members of each audience will help identify their information requirements (see *Clause 7.1.3*). However, the above factors will influence the level of detail required, the ratio of conceptual to procedural information presented, as well as the best way to present and organize the information.

Figure 3 shows a sample list of audiences for part of an order fulfillment system:

Call centre operator
Call centre supervisor
Webmaster
Warehouse picking staff
Warehouse manager
Finance manager
Accounts manager
Accounts payable clerk
Accounts receivable clerk

Figure 3 — Sample list of audiences for part of an order fulfillment system

The documentation designer should use both a “bottom-up” and a “top-down” approach to check that all user types have been considered:

- Bottom-up—The documentation designer should consider who will use the software product and determine all the types of user. For some applications, the audiences may be identified by their job titles, such as accountant. For other applications, audiences may have to be identified by the role they are performing, such as letter writer.

- Top-down—The documentation designer should consider the whole organization or the total functionality of the application, and break this down until a set of audiences or roles is reached. This set of audiences may be used to check the results of the bottom-up analysis.

User roles may not correspond to people in a 1:1 relationship; the same person might have more than one role: for example, salesperson and inventory taker.

Audiences may be grouped into a hierarchy, so that individual documents may be targeted at several audiences. The documentation designer should use the hierarchy to group together audiences who will have the same type of interaction with the software. The documentation designer should not (necessarily) reproduce the user organization's organization chart. Figure 4 is an example of such a hierarchy.

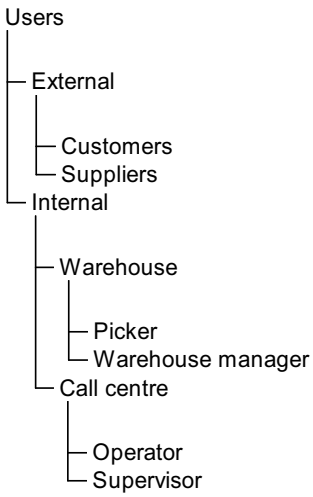


Figure 4 — Sample of an audience hierarchy

Audiences who are similar in other respects require different types of documentation according to how long they have been using the application or how often they use it. For example:

- learning to use (explanation of concepts and tutorials on procedures);
- using occasionally or infrequently (procedural instructions);
- using regularly (quick-reference cards);
- performing advanced or complex tasks (reference material and additional procedures).

If the software product being developed is a consumer software package, it may not be possible to gather useful information about typical users' jobs and background, for example, because their jobs and skills are varied. In this case, the designer should concentrate on the tasks that the users will carry out and the learning stages through which the user will go. Where there might be a wide variety of types of potential user, with a wide variety of experience, skills, and knowledge, the documentation should be adequate for use by the least experienced of the expected users.

The documentation designer should collect details of users' working environments. These details are one factor in deciding the most convenient media for presenting information. For example, if the software is to be used in a storeroom or warehouse:

- Will printed documents have to be carried around?
- Will there be convenient places to store them?
- Will users have the equipment to view the information on-screen?

If you expect the documentation to be used by a wide range of users with varying levels of experience and usage frequency, it is important to "layer" the information carefully. Layering ensures that the material is presented in such a way that all users may find the information that they need, at the level of detail that they need. In printed documentation, layering usually involves presenting information in progressively more detail, using clear headings to enable users to select the right information. In online documentation, layering may be achieved by various methods, including the use of hypertext links from overview-level items to items containing detailed information.

7.1.2 Audience profiles

For each type of user, the documentation designer should draw up an audience profile that records the pertinent information. This profile will be useful both in planning and as a guide for writers and illustrators. The audience profile may be recorded either in absolute terms (for example, special qualifications of one audience) or in a comparison of users' experience to the norm. Figure 5 illustrates the types of information that might be included in an audience profile.

Audience: Ticket Agent	
Background	Ticket agents have knowledge of the travel industry, customers' needs and concerns, and other computer booking systems.
Languages	Ticket agents may have a command of English, but not necessarily as their first language.
Use of the application	Ticket agents use the application while they are on the telephone to customers or when the customer is present, and make ticket bookings immediately.

Learning stage	All ticket agents will have attended a one-day training course; consequently, no users are novices. There is at least one expert user in each office all the time.
Frequency of use	Ticket agents frequently use the application all the time throughout an eight-hour shift (with breaks).
Working environment	Offices may be noisy and very busy. Ticket agents sit at desks to work. Agents have access to a computer throughout the shift but may share printers. Offices usually have a photocopier, fax machine, and scanner. Ticket agents use headsets for the telephone. Not all offices will have: • shelf space to hold a small library of books; • drawer space in the desks; • wall space available for displaying wall charts.

Figure 5 — Sample audience profile

7.1.3 Task analysis

The documentation designer should collect information about what users do, if possible by asking users questions, or by observing users and recording what they do. If this cannot be done (for example, if the software is still under development), then the documentation designer and developer should consider the tasks, or use other sources such as use-case design documents.

Figure 6 shows a sample task list for an electronic mail system.

Check for incoming mail.

Send or forward a message.

Print a message.

Attach a file.

Read an attachment.

Create a mail folder.

Move mail to a folder.

Figure 6 — Sample task list for an electronic mail system

The documentation designer should consider the following information about each task:

- why the task is carried out;
- how frequently the task is carried out (to help determine whether users will remember how to do it);
- how the user is likely to be operating, for example, doing tasks that take several hours, or working in a real-time situation where transactions have to be carried out while a customer is present;
- what discretion the user has in how or when the task is carried out;
- in what sequence the user carries out the set of tasks;
- what the prerequisites for the task are;
- how fault-tolerant the task is; that is, how important it is that the user carries out the task correctly;
- what the consequences of the task being completed or not completed are.

The designer should group related or similar tasks, or tasks that involve similar steps, in a hierarchy as illustrated in Figure 7. The designer should record the details of each task in a task profile.

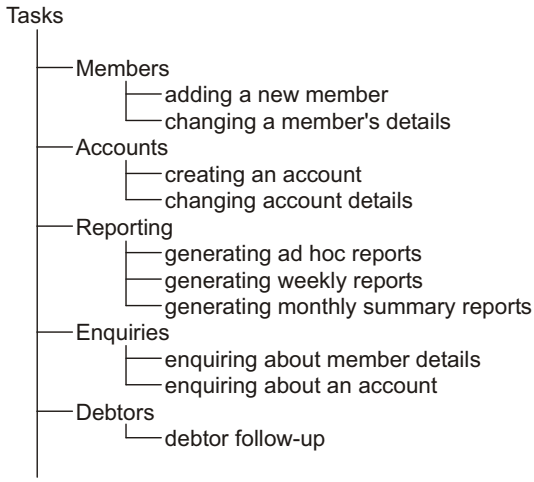


Figure 7 — Sample task hierarchy

The documentation designer should document which audiences carry out which tasks. The designer should consider using a matrix, as illustrated in Table 1. This example shows which types of user will be carrying out a set of security-related tasks and what their learning stages will be. As a result of this audience and task analysis, the documentation could be structured into a guide for the functional analyst, a guide for the security officer, and a quick reference (help topic) for users on changing their passwords. Common procedures could be written for the task of viewing the security log to be included in both guides.

Table 1 — Audience mapping matrix

	Audience								
Task	**DIC**	**SUP**	**MAN**	**AUD**	**ACC**	**FC**	**SO**	**FA**	**SA**
Adding a new user							N		
Deleting a user							N		
Giving a user permission to access a function								N	
Withdrawing a user's permission to access a function								N	
Viewing a user's permission to access functions								N	
Changing your password	O	O	O	O	O	O	O	O	O
Changing another user's password							NE		
Viewing the security log							N	N	
Investigating security breaches							NE	NE	
Printing security reports							NE	NE	

Legend to Table 1

Key to audiences		**Key to learning stages**	
DIC	Data input clerk	**O**	Occasional use
SUP	Supervisor	**N**	Normal use
MAN	Manager	**E**	Exploiting advanced features
AUD	Auditor		
ACC	Accountant		
FC	Financial controller		
SO	Security officer		
FA	Functional analyst		
SA	System administrator		

7.2 User documentation design

The user documentation designer shall perform two principal activities:

- use the audience and task analysis to determine what information needs to be provided (the documentation content) and what will not be provided;
- structure the content into necessary items in the documentation set, and determine the structure for each documentation product.

Depending on the software in question, a number of the items in this clause may apply more to a documentation set than to a particular document. For example, one piece of software may have a tutorial for end users, a manual for administrators, a reference for the application program interface, online help, a quick reference guide, and so on.

Subsequent clauses of this International Standard address the structure and content required for user documentation. For guidance on deciding whether information should be delivered on the screen or on paper, see *Clause 12.2*.

7.2.1 Designing for use of content

The documentation designer should prepare a list of the topics, in addition to the relevant access and structural information. The major factors that should influence decisions about where and how information is provided to the user are:

- audience and task profiles;
- information characteristics;
- the environment in which the information will be used;
- convenience to users;
- the range of technical facilities, including resources and the software product, available for developing, delivering, and viewing documentation;
- cost of delivery and maintainability.

The documentation designer should select a measure of how much information there will be, and use it to make sensible choices about delivery methods and delivery media. Volume estimates will vary according to the type of information. It may not be possible to find the volume of the information precisely, but the designer should have a concept of the scale of the documentation. The documentation designer should define the preliminary structure of each document using a draft table of contents, a topic list, or a list of web pages. To specify the size of each document, the documentation designer should give a count of the expected number of pages or topics. The documentation designer should include in the list navigation pages or topics, such as the contents list itself, indexes, glossary, and bibliography. The following techniques will be helpful.

- The designer may count identifiable items, for example, icons, tasks, dialog boxes, transactions.
- The designer may assess how much information is needed about each item, for example, the average and maximum number of steps for a task, or the approximate number of words needed about a concept.
- The designer may prepare a sample for one item and use its size as a basis for calculations.
- The designer may refer to existing documentation for comparable users, tasks, and applications, and assess from that how much information will be needed for the current application.

7.2.2 Designing formats

Consistent documentation is more easily used, because the user can focus on the content rather than the format. The organization shall set conventions for the whole product, project, or organization for entire documents and document sets, topics, chapters, front matter, back matter, prefaces, technical terminology, icons and symbols, navigational controls, and system messages.

For details of documentation formatting, refer to *Clause 12*.

8 Development and review

Although prototypes or samples of documentation may be developed as part of the documentation design process, the full development and review process should begin after the design of the documentation is finished. With a complete documentation plan, the documentation developer should follow the writing and illustration styles specified in the detailed document plan, using the specified tools and methods and including only the information specified in the detailed document plan.

Documentation development consists of writing the relevant texts, preparing the illustrations, and applying the necessary software elements to on-screen documentation to create a working system. In the documentation development process, the documentation developer prepares master versions of technically accurate documentation topics (also called objects or chunks) for assembly into the documentation products as specified in the documentation plans.

NOTE: The process of developing individual documents is an iterative one. If changes are made to the software, then changes might be needed in the documentation plan and the documentation.

8.1 Prototypes and drafts

Prototypes may be used to validate documentation designs for each type of document. Prototypes may be developed in detail and used to test that the documentation approach will provide users with what they need within the usability objectives. Documentation prototypes may be developed in conjunction with software prototypes. Prototypes may also be prepared for combinations of equipment, and for languages or localized versions, which will be used for the finished application. For each prototype, the documentation developer should consider the following questions:

- What is its purpose? What information is intended to be collected based on testing or trials of the prototype?
- What is its scope? What part of the application will be covered, and what types of documentation will be included?
- How will the tests and trials be carried out, recorded, and used in subsequent phases?

Prototypes are also useful for on-screen documentation to check that technology issues are resolved before the documentation design is finalized.

To avoid problems arising later in the project, at the first draft stage, the documentation developer should:

- try out new documentation-related methods and tools being used in the project, for example for creating or capturing illustrations or for output of content in various media;
- produce a sample of the text and illustrations using the final production method;
- for online documentation, check the context-linking of topics to application features; that is, make sure that the context-mapping technology is working correctly and that the correct context identifiers are being used.

NOTE 1: In planning printed documents, the documentation developer should decide for each document separately whether or not prototypes and drafts should be presented using the screen or page layout to be used for the finished work. Using the layout for the finished document helps those commenting on drafts see what the finished documentation will look like, but might lead reviewers to concentrate on page layout features instead of the technical content of the documents.

The documentation should be drafted when:

- outlines in the detailed documentation plans have been authorized;
- functional requirements documents have been accepted;
- sufficient stable information about the software is available.

Before drafting documentation, the corresponding part of the application software should exist, or there should be an agreed set of requirements and nomenclature for the software. If the information needed is not available by the date specified in the schedule, the documentation developer should advise the project manager to review the project schedule.

Before the documentation developer drafts task instructions, the project team should agree on the names of the application features that the tasks use, such as the names of menu selections, icons, and navigation features. The documentation developer should prepare a project glossary and apply it consistently. Clarity and simplicity of style are particularly important in documents that will be translated. The documentation developer should avoid slang, jargon, and humor, and use similes and analogies with care. In addition, the documentation developer should be aware that idioms and metaphors might not translate correctly, might have no equivalent in the target language, or could actually be offensive in another culture.

NOTE 2: For documentation that will be translated, the project team should agree which terms in the project glossary will remain in the source language and consistent terminology should be used for terms translated into the target language.

The documentation developer should ensure that the information contains everything users need and nothing that is irrelevant. The documentation developer should use the software to check the tasks being documented. The drafts should be technically accurate, based on the following sources of technical information:

- documentation design information;
- system design information;
- SME or software developer’s information;
- personal experiences of using the system or a prototype of it.

NOTE 3: A member of the documentation team may be the first person to study the working software in detail from a user's perspective. The documentation developer’s use of the software may provide a rigorous test of a system and give valuable feedback to the development team, for example by identifying inconsistencies.

NOTE 4: In planning, the documentation designer should decide whether it will be useful to provide reviewers with a draft or partial index for the first draft of each document so that they can find topics in the draft and review the format of the index with the rest of the draft.

8.1.1 CM during development

The organization responsible for documentation development shall ensure that the requirements of the CM system for the project are enforced. Approved documentation versions shall be maintained securely and separately from copies checked out for development. Each draft issued for review shall be uniquely identified. Back-up copies of master documentation should be stored securely and separately from the systems being used for documentation development.

8.1.2 Development of translated and localized documentation

While it is simpler to translate or adapt complete, approved documentation, documentation must often be released in multiple languages simultaneously to support software product releases. For some software products that are to be exported, there might be a legal requirement for the documentation to be translated.

Translations should be made by native speakers of the target language, not by native speakers of the source language. Localized versions may be needed for variations in the target language: for instance, European Spanish versus Mexican Spanish, and UK English versus US English.

As a first step in the translation process, the project glossary (list of terms and their definitions) shall be translated. The documentation shall be translated only after the translated list of terms has been approved. This is especially important when more than one translator is involved for a language (the structure of the glossary is described in *Clause 11.12*).

If inaccuracies or ambiguities in the source text are found during the translation process, the documentation developers should revise the source at the earliest opportunity.

Translated and localized documentation should be reviewed and tested in the same way as the base version.

8.2 Evaluation of documentation

The purpose of documentation evaluation is to ensure that documentation is fit for purpose. Documentation evaluations are performed throughout the document's development, production, and maintenance, from first draft to revision, for reuse or consideration of enhancements. The evaluation method selected will depend on a variety of factors including:

- the reasons for carrying out the evaluation;
- the phase in the life cycle at which the evaluation is being carried out;
- the resources available;
- the time available;
- the amount of information available about the documentation;
- the availability of users with the appropriate ranges of experience and skill;
- the availability of experts in documentation design;
- the availability of experts in usability.

Different methods may be used at different phases in the development, after the documentation is complete, and when the documentation has been used for a specified period. A combination of methods is likely to be appropriate in each case.

This clause describes the processes of documentation evaluation in two forms: review and testing.

The documentation review process includes reviews of:

- documentation structure, format, and style compared to plans, requirements, and established standards;
- scope and level of detail of the information presented;
- range of topics covered;
- technical accuracy;
- consistency with the product;
- safety (provision of critical information to protect against hazards or errors).

The documentation test process includes:

- operational testing to ensure that the documentation performs correctly and responsively (for example, on-screen documentation is appropriately linked to the software topic, and navigation is consistent and as expected; the index in printed documentation has accurate references);
- usability testing of documentation with the software product, to determine whether the intended audience can perform their tasks with the aid of the documentation.

Documentation test requirements shall be specific and measurable.

After documentation is released, evaluation continues in the form of feedback from users (through comment forms and messages) and from independent software product review magazines and reports. Surveys and interviews may also be useful in gathering users' and customers' opinions and attitudes on released documentation. Problem reports gathered by trainers, sales staff, and customer service also indicate where documentation or software products need improvement.

NOTE: Documentation is also included in evaluation of the software product, as in ISO/IEC CD 25012, *Software engineering — Software product Quality Requirements and Evaluation (SQuaRE) — Data quality model*. The ISO/IEC 9126 series (*Software engineering — Product quality)* may also be used for this purpose. The ISO/IEC 14598 series (*Software engineering — Product evaluation)* addresses software product evaluation.

Documentation evaluation shall include the following four activities:

- Plan: The documentation designer and developer should identify the requirements for acceptable performance or quality, and prepare for the evaluation exercise by specifying when the test will be held, what resources are needed, how the evaluation will be carried out (test scenarios and scripts), how the results will be measured and recorded, how the results will be analyzed, and what the pass/fail criteria for the tests will be.

- Do: The documentation designer and developer should evaluate the documentation against the requirements, taking measured results and recording them. Members of the project team normally carry out tests during development following test scripts that are drawn up in advance to ensure that the tests are systematic and complete.

- Check: The documentation designer and developer should analyze and report the results of the evaluation, recommending next steps.

- Act: The documentation designer and developer should revise the documentation, advise changes to the software product based on the test results and recommendations, and determine whether further evaluation cycles are needed.

To produce acceptable results, revisions may entail changes in the project schedule and documentation plan, or even changes in the requirements. For example, documentation of advanced functions might be deferred until a later project phase, or product release dates might be deferred until acceptable documentation is produced.

8.2.1 Other roles in evaluation of documentation quality

Documentation evaluation shall be based on the required features and qualities. Evaluation of documentation quality depends on the recognition of various perspectives for acceptability. Ultimate acceptance comes from the end-users, but unless others find the documentation to be of acceptable quality, it will never reach the users. Those other perspectives include:

- **Manager's view.** Managers may be more concerned with overall quality than with specific quality characteristics. They may assign different weights to certain characteristics to reflect the business needs of the organization, comparing the documentation to what is commercially available in the market and what is less costly to produce. Managers should be aware that the quality of documentation may appreciably affect costs for customer support and future sales.

- **Technical editor's view.** The technical editor assesses overall document quality and consistency and works to ensure that documentation fulfills the vision and plans that were laid out during the analysis phase.

- **Developer's view.** Software developers may be concerned with how the software product operates in its novel or advanced functions, more than with how it supports the users' tasks.

  NOTE: However, developers are very close to the software product and are more sophisticated users.

- **Maintainer's view.** Those who will have to maintain the documentation systems will have special requirements for quality in addition to those of other documentation developers. They will be concerned with, for example, the simplicity and clarity of the documentation structures, the ease with which new versions of the documentation may be created, and portability to new technology for content management and production.

- **User's view.** Users are likely to measure quality in terms of the inclusion of accurate and relevant information, as well as ease of finding the required information and applying it.

8.2.2 Documentation review procedures

Documentation reviews should precede usability testing to improve documentation quality, reducing the number of flaws and defects remaining to be identified in testing and thus the amount of rework and retest required at a late stage. Documentation drafts should be reviewed for the following:

- **Technical accuracy.** The approval authority identified in the documentation plan has responsibility for the technical accuracy of the documentation for the software product and for resolving conflicting comments from technical contacts.

  NOTE 1: Technical accuracy of instructional procedures is better verified by testing the documentation with the software than by reviews. Technical reviews of embedded documentation (such as pop-up help) are best performed in conjunction with software product testing.

- **Safety.** An expert should review the documentation to ensure that it contains warnings, cautions, and appropriate avoidance and recovery instructions for risks associated with use of the software product. The reviewers shall verify that warnings and cautions are appropriately placed and worded to be noticeable and understandable by the intended users.

- **Ease of understanding.** At least one person who is not familiar with the software product should carry out the checks for ease of understanding. The person checking drafts should be alert to potential misreadings and misunderstandings, and should highlight them.

- **Conformance and consistency.** Documentation developers or editors should check that the documents conform to the requirements of selected standards and organizational policies and style guides, follow the documentation and design plans, and are consistent with each other in appearance and terminology. The reviewers should check for consistency and accuracy in references or links to other parts of the document or document set. This level of assessment will not guarantee that the documentation will satisfy its users, but should ensure that no user will find the documentation totally unsatisfactory.

- **Completeness.** The documentation should include or refer to the information that users need. If possible, at least one person who is not familiar with the software product should carry out the checks. (See *Clause 11.1*.) The total set of documentation, printed and on-screen, should be reviewerd, including the items that will be integrated with the software. The documentation should be reviewed together, or be reviewed by the same people, so that it may be checked for consistency and completeness.

- **Editorial consistency and correctness.** An editor should check near-final drafts for spelling, grammar, punctuation, editorial standards, and all documentation structure, format, and style requirements.

- **Security and legal accuracy.** The organization's security group and legal counsel should check near-final drafts to ensure inclusion of the correct notices for protection of sensitive and proprietary information, compliance with legal requirements, and proper handling of trademarks. If possible, an intellectual property law (IPL) attorney should perform this check.

The documentation developer and project manager should determine the review participants and procedures. Reviewers may be peers, editors, technical experts, trainers, managers, or customers. Reviewers should be selected based on their expertise, familiarity with the requirements and standards, and ability to provide thorough and usable comments and corrections. It is rare for reviewers to focus on all aspects of documentation at one time. Separate reviews or reviewers should evaluate the documentation's organization; technical accuracy; suitability for the intended audience; and consistency of grammar, style, and format.

Moreover, it is inefficient to hold detailed editorial reviews before the technical content is accurate and consistent.

Topics, sections, and partial documents may be reviewed. For a long document, the documentation developer may send a new chapter to a software developer for review before the whole document is finished. Before production, the documentation shall be reviewed in its entirety.

Reviewers may offer divergent opinions and comments. The organization should determine who on the project team may see and apply other reviewers' comments. Review and acceptance procedures shall specify who is the final authority for accepting and implementing changes.

NOTE 2: To ensure timely review responses, the documentation developer, reviewers, and management should agree on the number and the length of review cycles, such as one week for each review. The stakeholders should further agree that if review comments are not submitted within the agreed time period, such omission grants approval to the document developer to proceed to the next phase of documentation development. This avoids schedule delays, for example, when one or more reviewers are away from the office without naming a substitute review participant.

Different methods of reviewing information are appropriate for checking different features. The following are suitable methods for reviewers to use:

- Reviewers may examine printed copies of text and illustrations. This method should be used for reviewing:
    - the accuracy of information;
    - the editorial style;
    - the appropriateness of the amount of information per section (level of detail of content).
- Reviewers may examine documentation on the screen independently from the software. This method should be used to review:
    - information systems that can operate independently of the software;
    - information consisting of sets of items through which users may navigate to find what they need;
    - indexes and searches provided to help users find what they need.

NOTE 3: These methods are essential for reviewing the style and presentation of print and on-screen documentation.

Testing on-screen documentation on the screen, together with the software, is described in *Clause 8.3*.

Drafts should be distributed to reviewers with:

- clear review criteria, including a clear statement of what reviewers need to concentrate on;
- instructions for how to provide comments and for using review tools;
- an indication of the time to be spent reviewing the material;
- details of whom to contact during the review in case of questions or problems;
- instructions concerning the return of comments to a specified person, by a specified date.

The documentation developer should consider assisting reviewers by highlighting changes made from previous drafts, for example by vertical lines in the margin, to avoid the need to re-read unchanged text. This technique is appropriate only if all the sections that need to be checked are highlighted, because reviewers will only read the highlighted sections.

NOTE 4: If there are too many changes, highlighting may become counter-productive. One technique is to apply the 25% rule: if more than 25% of the document has changed, all highlighting should be removed, and the reviewer should be instructed to read the whole document.

The review instructions should indicate how reviewers should supply comments:

- as mark-up of printed copies;
- in a clearly annotated electronic form (the corrected electronic version should have changes highlighted to enable documentation developers to identify them);
- as a separate on-screen review document.

Review instructions should also specify whether:

- comments should be identified with the reviewer's name (this may be handled automatically by the review tool);
- reviewers are permitted to keep a copy of the draft, due to security or configuration control restrictions.

Reviewers' comments shall be retained under configuration control at least until the next cycle of reviews has been completed. Many reviewers expect to see copies of their previous comments while reviewing subsequent drafts.

Revised drafts should be reviewed for editorial accuracy (making sure that the comments made on the first draft have been incorporated correctly and do not introduce inconsistencies) and for technical accuracy, particularly if the software design has changed since the first draft. The documentation developer should check with those responsible for software product CM to identify changes to the software application as a result of its own reviews and tests, which should be reflected in the documentation.

The final draft should be prepared when no further changes in content are allowed, and should be reviewed only for format and typographical errors before final approval by the appropriate authority.

Where comments conflict, the product authority should make the final decision.

8.3 Documentation testing

Documentation tests validate and verify documentation in conjunction with the software. The documentation development team should test the documentation at each stage of application development when the software is available, including unit testing and system integration testing, as well as usability testing, before the product is generally released.

8.3.1 Types of documentation tests

System testing should include both embedded and separate documentation. System testing should verify that:

- the access methods and navigation features for context-sensitive documentation and wizards perform properly;
- the links and cross-references for related information work correctly;
- the correct information (such as an error message) is displayed in each given situation;
- the instructions in the documentation have the desired effect when carried out;

  NOTE: In particular, the examples in the documentation and tutorials should be thoroughly verified on the system.

- the documentation headings, search, and index entries quickly lead users to the information needed to perform their tasks.

8.3.2 Usability tests

User tests are the most acceptable method of checking that the information provided in the documentation is what users need, and that users are able to find it, understand it, and apply it. This method is useful for evaluating systems that are partially developed, while there is still time to change the documentation, as well as for near-final software and documentation. As an intermediate level of assurance of usability, a usability specialist may review documentation to assess whether the documentation will meet its specified, qualitative usability goals. This assessment may be subjective or may use analytical methods appropriate to the particular goals.

Empirical evaluation is carried out by experts observing users (actual users or others who represent the intended audience) using the documentation to perform selected tasks. A range of different recording techniques may be used, such as an observer taking notes during the test, videoing the users, or asking the users to explain what they are doing and why. The assessment of the system may be made by interview, questionnaire, or group discussion.

Other opportunities for structured and unstructured observation of documentation users may be arranged if there are validation tests or field trials before general release of the product. In both validation exercises and field trials, the documentation developer should identify problems with the software and its documentation together, and seek solutions to problems by considering the software and its documentation together.

If validation and field trials highlight major problems with the product, the whole product is likely to require another design stage to resolve the problems. The documentation, therefore, will also go through another design stage as part of this process.

Planning for usability tests includes the following:

- definition of the test's purpose;
- consent from the sponsor to undertake usability testing;
- identification of the tasks to be tested and acceptance criteria;
- provision of the resources, including facilities;
- schedules for the tests;
- methods for conducting the tests;
- methods for recording the results of the tests;
- methods for determining whether the objectives have been met;
- methods and plans for incorporating the results of the tests into the future design and development activities for the project;
- identification and recruitment of test participants.

The highest level of assurance requires a numeric scale of performance of the system from managed user tests. In the trials, users are observed by a usability specialist as they work with the documentation, performing tasks designed to test the documentation against the defined usability goals stated in terms of user effectiveness and efficiency. A questionnaire or user comments during the test are valuable in assessing the level of user satisfaction. The tests are performed in a specified context of measurement, which is related to the context of use of the software product. The resulting performance levels are compared with the required levels. This measure of assurance is necessary to ensure the usability of the documentation.

NOTE: Documentation usability comments may reveal problems with the software product as well.

9 Production

Production is the process of integrating, preparing, reproducing, packaging, and delivering a documentation product once the content has been developed and tested. Production thus includes final assembly and review. The activities required to produce the documentation depend on the degree to which the different types of information are integrated with the software.

9.1 Final assembly and review

When the final draft has been approved, publishing/production staff, editorial staff, or a different person than the one who wrote the documents, shall:

- prepare the final table of contents and indexes;
- proofread the text;

  NOTE: Although automatic spelling and grammar checkers and other authoring tools may be helpful, always make visual checks.

- carry out a final check of cross-references;
- prepare and check final artwork for the illustrations;
- check that the illustrations are placed correctly (see *Clause 12.18*) and are clear enough for the chosen method of presentation (on the screen or in printed form);
- if the document is to be printed, review final pagination, making sure that major sections start on new pages (if that is the house style of the organization), and that lists or short sections are not split over page breaks;
- check also that:
  - all the pages or topics are present;
  - all the pages are correctly numbered;
  - all the headings, tables, and figures are correctly numbered.

9.2 Approval

Documentation shall be approved as specified in the documentation plan or contract before it is released. The supplier should verify that all conditions for documentation development have been followed, such as:

- the documentation life cycle process has been followed;
- the testing has been carried out and is complete;
- a draft has been reviewed and corrected;
- usability objectives have been met;
- legal approval has been obtained.

9.3 CM

The CM process shall be exercised to maintain a controlled copy of the released documentation version. When the documentation is complete, the documentation developer should supply sufficient information about it to enable new versions of the documentation to be prepared. The organization should maintain a record of the versions of all the items (including text and illustrations) that were included in the published version. This will enable the creation of a version of the documentation exactly as it was prepared.

9.4 Updating and maintenance

Properly designed and developed documentation is more easily updated and maintained. The designer should expect that modifications are likely to result from:

- the creation of a new, customized, rehosted, or upgraded version of the software product;
- the discovery of errors in the existing software product or documentation;
- the development of new software products that can be documented by reusing existing content.

Any changes that need to be made to the documentation as a result of changes to the system design should use the formal change control procedure for the entire product or the documentation management system. Plans for modifications to the on-screen and printed documentation should be integrated with the plans for modifications to the software.

Wherever possible, documents should be designed so that, when a new version of the software is issued, new versions of the documentation are issued as well. The project requirements and constraints for on-screen and printed documentation should specify whether the documentation is to be modified or updated and if so, how this should be done.

Release notes may be used to provide concise summaries of changes in software products for existing users. However, release notes are confusing to new users and should not be used as a quick substitute for a thorough documentation update. Documentation content affected by the new release (including translations), especially instructional procedures, shall be updated and provided to customers who have contracted for the software and documentation maintenance.

NOTE: If a completely new version of the product, including the printed documentation, may be bought or is distributed each time the product is updated, there should be no need to issue changed pages to the users. Where alterations to the product are issued, it is more convenient for users of printed documentation if complete, printed or on-screen documents are reissued: replacement pages can be lost and do not always reach all holders of a document, so there is a risk that users will be working with out-of-date documents. Only large documents with few users should be updated by issuing replacement printed pages.

10 Structure of documentation

The structure of documentation, both printed and on-screen, includes the way that it is organized into segments and the order in which the segments are presented to the user. Documentation may be structured into a single document or a document set of printed or on-screen documents. The structure of documentation should assist the user in locating and understanding the information content. When a document set will address audiences with widely differing needs, at least one of the following structures shall be used:

- separate sections devoted to the needs of specific audiences (the audiences and their needs shall be identified specifically in the introduction, enabling each user to identify the sections of interest);
- separate documents or document sets for each specific audience.

10.1 Overall structure of documentation

A document set may consist of one or more documents, and each document of a document set may be one or more volumes. For example, a printed command manual might have one volume covering one half of the commands and a second volume covering the other half of the commands. Documents shall be structured into units with unique content. Well-structured documentation makes information available where it is needed without redundancy.

For purposes of this International Standard, the following non-mandatory nomenclature is used for the structural parts of documentation. A printed document is structured into logical units called *chapters*, subdivided into *topics*, which may be subdivided into *subtopics*, and printed on physical units called *pages.* Lengthy reference material is presented in *annexes or appendixes.* An on-screen document is structured into logical units called *topics* and presented in physical units called *pages* or *screens.* Because of variations among physical display devices, the term *screen* as used here does not necessarily refer to the information displayed to the user at a single instant, but rather to the entire collection of information available by simple scrolling operations.

Documentation headings should clearly and uniquely indicate what information is contained in each section.

Each page or screen shall be uniquely labeled (for example, with a page or topic number, or screen name or number). When viewed or printed, each topic in an on-screen document should be identifiable as belonging to the parent document. For example, a status bar or header shows the document or file name.

The documentation structure, length of a chapter or topic, and amount of information presented on a page or screen (physical unit) depend on several considerations:

- access to the documentation while using the software;
- amount of information;
- complexity of information;
- number and size of illustrations needed;
- audience familiarity with the information;
- limitations of the media;
- the structure of the software as seen at the user interface;
- usage modes.

The documentation designer should not create a proliferation of short printed documents, nor combine a lot of information into one cumbersome document. If a document becomes too large, the documentation designer should consider splitting it by:

- putting the conceptual and instructional information in one document, and the reference information in another;
- keeping different types of reference information separate; for example, one document for concept-related information and another for quick reference look-up.

The hierarchy of subsections should be limited, so users know where they are in the document. Printed documents shall be structured with no more than three levels of subdivision within a chapter. For example, a subtopic numbered 1.2.3.4 would be at the lowest level of subdivision. On-screen documents shall be structured so that information may be accessed with no more than three jumps (that is, requiring no more than three links) from the initial page of a topic (not counting action required to open the document).

The documentation designer should make the documentation structure flexible so it may adapt easily to a change in the volume of information resulting from amendment or updating, or through translation into another language.

The organization of documentation shall support its usage mode (instructional or reference). When a document contains both instructional and reference material, the two shall be clearly separated into different chapters or topics, or distinguished by formatting within the chapter or topic.

10.1.1 Structure of instructional mode documentation

Task-oriented instructional mode documentation shall include procedures structured according to the user's tasks. Related tasks should be grouped in the same chapter or topic. Chapters and topics should be organized to facilitate learning by presenting simpler, more common, or initial tasks before more complex, less frequently used, or subsequent tasks. Alternatively, topics and chapters may be organized by the logical order in which they need to be performed.

10.1.2 Structure of reference mode documentation

Reference mode documentation should be arranged to facilitate random access to individual units of information. Comprehensive reference documents will not be used sequentially. Do not assume that, at any point in the document, the user will have seen any of the information covered in earlier sections; use cross-references or links. Present the information in reference documents so that the users may quickly find the relevant entry when they have a problem, and hence quickly find the answer. The information should be searchable or indexed by key words in the users' questions or problems, as well as by features of the application. For example, quick-reference information explaining which software product features or keyboard keys the user needs to use to achieve a particular result could be structured by the result the users want to achieve, not only by the software product feature or the keys that they need to use.

A list of software commands or error messages should be arranged alphabetically or by message number.

NOTE: If alphabetical sequence is used and the documentation is translated, the sequence might be changed.

Problem resolution or troubleshooting information should be structured for quick reference. For example, group questions according to which part of the application they apply to, or to the types of task the users will be carrying out.

10.2 Structure of documentation according to audience needs

To decide what information to deliver in the documentation, the documentation designer should examine the information profile (see *Clause 12.3*) and details of the context of use, and decide which needs will be supported by the documentation and which will not.

Not all of the information in the information profile needs to be delivered in the documentation. The information needs of the target audiences should be identified (as far as possible) in the documentation plan, and where possible, the delivery mechanism identified. For each audience, the documentation designer needs to specify the procedural and conceptual information required and how that information will be delivered. When the plan is signed off, any assumptions about suitability of specific delivery channels may be confirmed or revised. Alternative channels for delivering information to the audiences include:

- training;
- building it into the software user interface;
- providing a community of practice for users to share information;
- using other sources such as user documentation for other products, tertiary education, or commercial training courses.

Security information needed by some users should be structured into a separate document from information for other users. For example, information about administering the access control security of a multi-user system should not be included in the same document as information for end users. Instead, the documentation designer should provide a reference to the location of the information.

Figures 8 and 9 show how the audience's information needs may be supported by the structure of the documentation set.

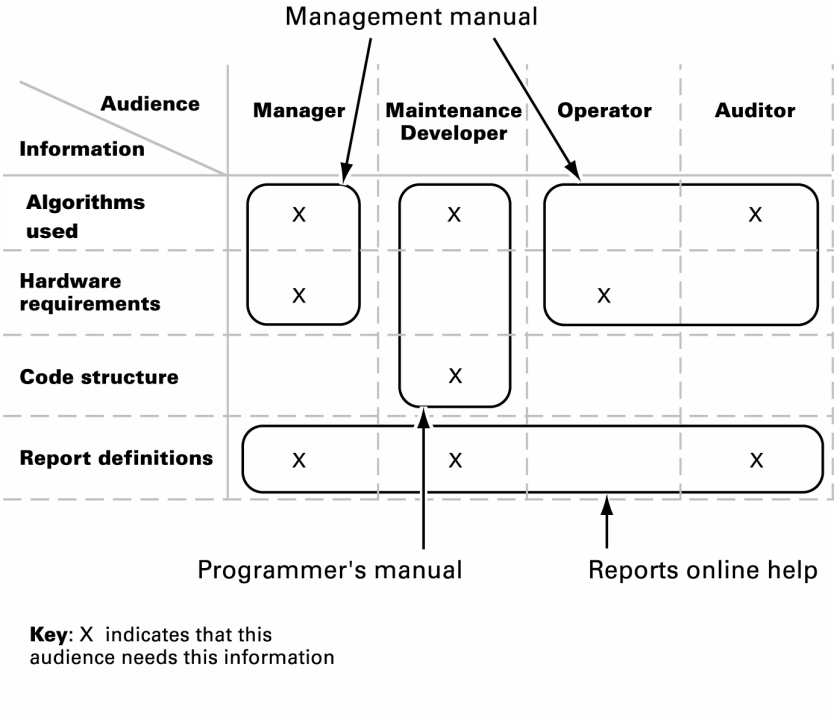


Figure 8 — Using audience information needs to determine document content

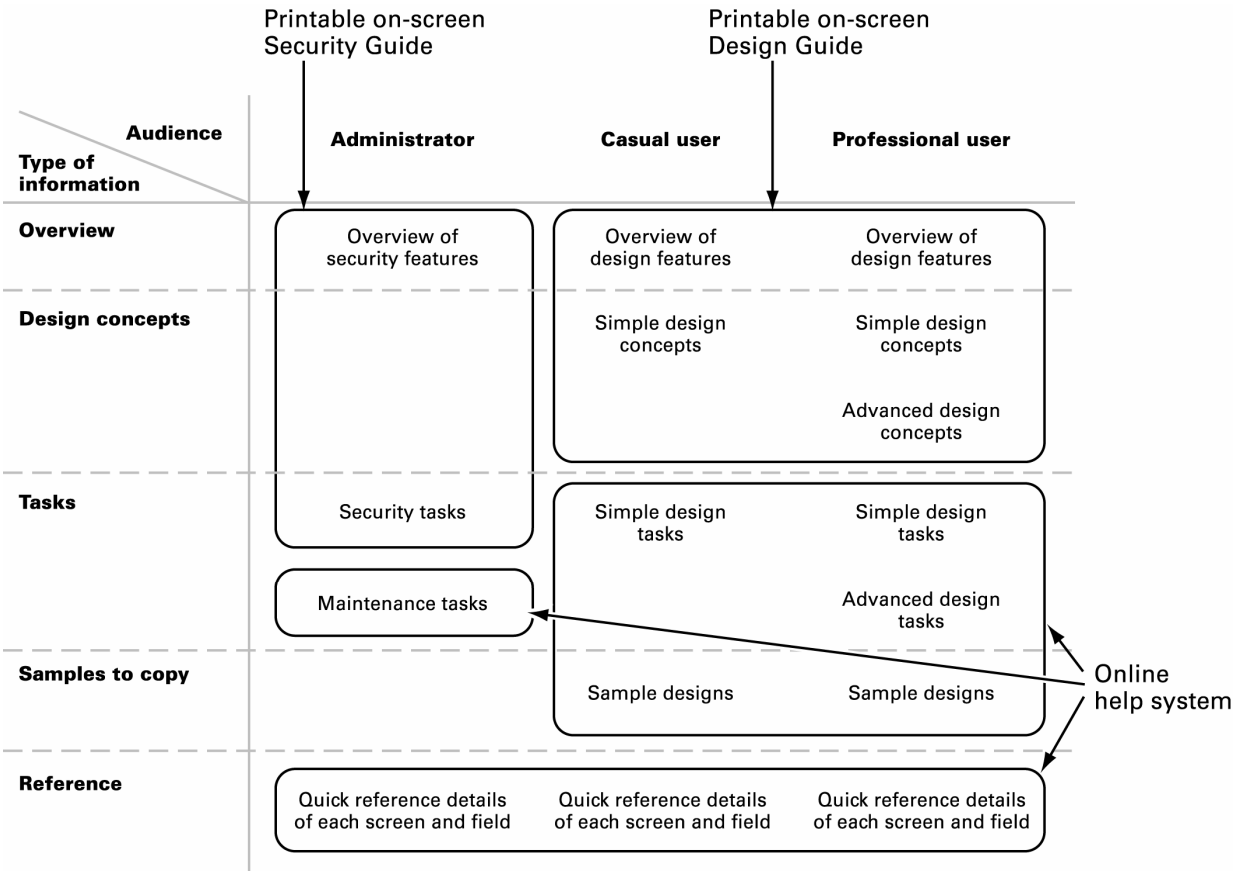


Figure 9 — Using information type and usage to determine delivery method

10.3 Size of topics in onscreen documentation

The documentation designer should determine the sizes of individual topics delivered to users by the needs of the users, not by the tools being used. When deciding the sizes of topics, the documentation designer should consider:

- how much information the user may see at once with the most commonly used output device, without scrolling through it;
- how much information the user actually needs, as opposed to how much information there is available;
- what information the user must have, so that some information is always displayed, and the availability of additional information is indicated so those who want it may navigate to it;
- how much information the user is likely to be able to absorb, taking into account the urgency of the need for help;
- how many extra navigation steps the user would have to take if the information were not supplied in one place.

Where possible, the documentation designer should make the entire topic visible at once. Where this is not possible, the documentation designer should provide a way of scrolling or paging, and make it clear how far through the information the user has moved (see *Clause 12.13*). If there is a large amount of text that users may scroll through, the documentation designer should structure it so that users may easily find the particular subjects that they are interested in.

NOTE: Aspects of design for documents specifically intended for printing on paper are included in Annex D.

10.4 User documentation components

Table 2 lists required and optional structural, content, and format components of the document. The components may be arranged in this order in printed documentation.

The required components shall be included in documentation unless the information does not exist or is not applicable for a specific document. For example, a description of conventions may not be applicable in an overview document. The components may be renamed; for example, information suggested for the introduction could go in a section labeled “Preface.” The requirements specified in Table 2 may be tailored to meet an organization’s or project’s needs (see *Clause 2.1, Definition of conformance*).

Table 2 — Components of documentation

Component	Clause	Required?
Identification data (package label/title page)	11.3	Yes
Table of contents	12.14.1	Yes, in documents of more than about 8 pages after the identification data
List of illustrations (or separate lists of figures and tables)	12.14.3	Optional
Introduction	10.5.1	Yes
Information for use of the documentation	11.4	Yes
Concept of operations	11.5	Yes
Procedures	11.7	Yes (instructional mode)
Information on software commands	11.8	Yes (reference mode)
Error messages and problem resolution	11.10	Yes
Glossary	11.12	Yes, if documentation contains unfamiliar terms
Related information sources (bibliography, list of references)	11.13	Optional
Navigational features	12.13	Yes
Index	12.14.4	Yes, in documents of more than about 40 pages
Search capability	12.14.5	Yes, in on-screen documents of more than about 8 pages

Initial components for a document within a set of documents may include a chapter defining the document in the context of the set, rather than the subject of the document. Often called "About this document," this chapter is a form of "Read me first" information, such as identification, navigational directions, disclaimers, and warnings, often common to the set rather than the individual document. Segregation of this information allows for the Introduction to be restricted to the context of the subject matter of the specific document rather than the document set. Because this chapter is outside the context of the subject, the chapter is often provided as front matter, preceding the Contents, and can be an alternative to a separate "road map" document.

10.5 Placement of user documentation components

10.5.1 Initial components

Each individual document shall be structured to begin with identification data (*Clause 11.3*), followed by a table of contents (see *Clause 12.13*) and an introduction; that is, the introduction is the first chapter or topic of the document.

The introduction shall describe the intended audience, scope, and purpose for the document and include a brief overview of the software's purpose, functions, and operating environment. If it is necessary for the users to understand some terms before using the documentation, the documentation developer should advise them where they may find this information, for example, a glossary or concept of operations.

Introductions shall be provided within a document for each chapter and topic. Introductory sections should be provided for each major feature or function of the software being documented. The introductory sections shall provide an overview of the topic, the purpose of the function, and environmental requirements, warnings, cautions, or user requirements unique to the topic.

10.5.2 Placement of critical information

Critical information should be placed in a prominent location in the documentation. General warnings or cautions that apply throughout the use of the software or documentation shall appear in the initial components. If there is vital information that users need to be aware of, the documentation developer should ensure that all users see it, for example, by displaying it either the first time or every time the application is started.

Specific warnings and cautions shall appear on the same page or screen and immediately before the procedure or step that requires care.

NOTE: Documentation developers should consult the style guide being used for the documentation to distinguish correctly among cautions, notes, and warnings.

11 Information content of user documentation

This clause specifies characteristics of information contained in user documentation, including completeness and accuracy (see *Clause 11.1* and *11.2*). This clause also defines the required information for inclusion in user documentation (see *Clause 11.3 and following*). The information required in this clause shall be included in the documentation unless the information does not exist or is not applicable for a specific document.

The content of documentation is related to its usage mode. Users of software need documents either to learn about how to use the software (instructional mode) or to refresh their memory about it (reference mode). Instructional mode documents may be either information- or task-oriented. Information-oriented documents instruct the user on the concepts and technical information needed to use the software properly (see *Clause 11.5*). Task-oriented instructional documents show the user how to complete procedures to reach a goal (see *Clause 11.7*). Reference mode documentation may be context-sensitive and integrated in the software; for example, pop-up or drop-down lists of acceptable data values or commands.

In either instructional or reference documentation, the content of documentation may be improved by the inclusion of examples and illustrations.

NOTE: ISO/IEC 15289, *Systems and software engineering — Content of systems and software life cycle process information products (Documentation)* provides guidance on the content of information items (documentation) required during the life cycle, including user documentation.

11.1 Completeness of information

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). Instructional mode documentation shall include complete information to enable performance of selected tasks using the software functions by the least experienced members of the audience. Reference mode documentation shall include all instances of the selected elements being documented. For example, if reference mode documentation covers a subset of software commands, it shall include the relevant user-entered and system-displayed commands and error messages in that subset.

Considerable information may be available about the software and about how it carries out its functions, but instructional mode documentation should include only what is necessary for the user to perform the task.

11.2 Accuracy of information

Documentation shall accurately reflect the functions and results of the applicable software version. If the previous documentation version is no longer accurate, current documentation shall be available for customers acquiring software updates or upgrades. Documentation corrections and updates may be provided via a new manual, a read-me file or errata sheet, or a downloadable file on a web site.

11.3 Content of identification data

Documentation shall contain unique identification data. That identification data shall include the following:

- documentation title;
- documentation version and date published;
- software product and version;
- issuing organization.

Identification data shall appear on a package label, legible without opening the package, and on a title page. A package label is not required if the title page is legible without opening the package. Each document in a document set shall have a unique title page. For single-page documents, such as quick reference cards, the identification data may appear on the same page as the rest of the document.

The title of the document should indicate its nature and scope and should not include abbreviations or acronyms unless they are familiar to the intended audience. If different versions of the software are available for different operating environments, the title page should specify the applicable operating environments, including versions of hardware, communications, and operating systems.

Other information may be included on the package label and on the title page or following pages:

- document and software product part numbers;
- serial numbers;
- restrictions on use.

NOTE: Identification of software products may include the name, operating system, edition, version, languages supported, and dates.

The package label and pages immediately following the title page should include:

- International Standard Book Number (ISBN);
- copyright and trademark notices;
- restrictions on copying or distributing the documentation;
- information for contacting the issuing organization (for user's comments);
- warranties, contractual obligations, or disclaimers, such as the manufacturer's legal responsibilities and the consumer's rights, including training and related assistance, software support, quality assurance, and availability of source code;
- if the document comprises more than one part, a list of all the parts;
- country where the document was printed;
- general warnings and cautions;
- reference to standards followed for the software, indicating the degree or level of conformance;
- acknowledgments.

The documentation designer should obtain legal advice on what copyright and version details about the application must be included. This data will differ from country to country.

Contact information may be provided for the manufacturer or supplier of the product, and for the publisher or developer of the document. Contact information may include postal address, telephone and fax numbers, e-mail address, and URL.

The identification of the document and the software shall be consistent with the CM practices of the issuing organization or the acquiring organization. Information (change history) shall be provided in the document set to document the date of issue and version number of the current version, and may include information for each previous version of the documentation.

If users need to quote details about the application when they are asking for support, these should be easy to find.

If the software product is to be translated, the documentation designer should carefully decide whether or not graphics are to be used as part of the product identification information, for example, as a company or product logo. Different translated versions of a graphic might be required for the different language versions of the product.

11.4 Information for use of the documentation

The documentation shall include information on how it is to be used (for example, help on help), and an explanation of the notation (a description of formats and conventions—see *Clauses 11.12 and 12.11*). At least one document in a document set shall identify all of the documents in the set by title and intended use, including recommendations on which members of the audience should consult which sections of the documentation. In document sets comprising many volumes or documentation products, this information may be provided in a separate “road map” or guide to the document set. Documentation may include identification and discussion of notable changes from the previous version of the document set and the software.

11.5 Concept of operations

Documentation may explain the conceptual background for use of the software, using such methods as a visual or verbal overview of the process or workflow; or the theory, rationale, algorithms, or general concept of operation. Explanations of the concept of operation should be adapted to the expected familiarity of the users with specialized terminology for user tasks and software functions. Documentation shall relate each documented function to the overall process or tasks. Overview descriptions of workflow processes should identify the roles and responsibilities of those involved. Conceptual information may be presented in one section or immediately preceding each applicable procedure. Diagrams are often helpful for describing processes. If a concept description refers to particular functions of the software product, or specific tasks, the documentation may allow users to link to information about those functions or tasks.

Command or data entry input examples should always be clearly explained, especially for dates:

EXAMPLE: Enter the customer's date of birth. For example: 12/10/1980.

With no explanation (or information about the required format: mm/dd/yyyy), it may not be clear to the user whether to use the US or European format and hence whether the date is 10th December 1980 or 12th October 1980.

An example is given in Figure 10.

Formula

examples

You may use a formula to carry out calculations on values, much as a calculator does, and put the result into a cell.

The values used in formulae may be:

- numbers that you type in;
- numbers held in other cells, given by cell references;
- numbers worked out using functions provided by the system.

You specify the calculations using operators, such as + - * /

EXAMPLE: =(D6/D10)*100

This formula calculates and displays as a percentage the ratio between the numbers stored in cells D6 and D10.

Whenever you change any of the values in cells used in a formula, the program automatically updates the result of the formula.

See also

List of functions

Manual recalculation mode

Figure 10 — Sample presentation of an example

User documentation should be organized to support the users' tasks rather than the organization of software functions. However, explanations of the application functions may be helpful for new users. Figure 11 provides an example.

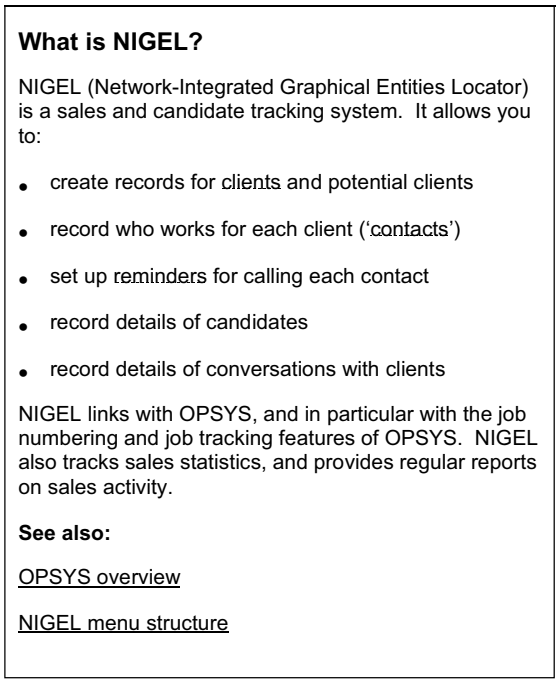


Figure 11 — Sample overview for a software product module

For on-screen descriptions of application functions, the documentation developer should determine whether users will be able to see the application screen at the same time as the on-screen documentation. The following guidance should be applied for explanations of functions:

- The function name should be used as a main heading or the title of the information window.

  NOTE: "Title" refers to the first-level heading of an information window or screen. "Heading" refers to a second- or third-level heading.

- A short statement of the purpose of the function should be included at the very beginning, to help users confirm that this is the function they are interested in.

- The contexts in which the function can and cannot be used should be explained.

- What users may achieve when using the function should be described.

- If users need to understand how the application works to decide whether or how to use the function, this information should be included.

11.6 Information for general use of the software

Task-oriented instructional mode documentation shall include instructions for routine activities that are applicable to the general use of the software:

- software installation and uninstallation, if performed by the user;

- orientation to use of the features of the graphical user interface (see *Clause 12.11*);

- access, or log-on to and sign-off from the software;

- navigation through the software to access and to exit from functions;

- data operations (enter, save, read, print, update, and delete); the instructions should cover any unusual features of how to enter data, for example, using sliders, buttons, special keys; or selecting from displayed lists;

- methods of canceling, interrupting, and restarting operations.

NOTE: Documentation should indicate if a typical method of navigation or data entry would result in error. For example, “Do not use your browser's BACK button or press any other keys until the 'Payment accepted' message is received."

These common procedures should be presented once to avoid redundancy when they are used in more complex functions.

11.7 Information for procedures and tutorials

Instructional mode documentation provides directions for performing procedures. Procedures shall include:

- preliminary information;

- instructional steps;

- completion information.

11.7.1 Preliminary information for procedures

Preliminary information common to several sets of procedures may be grouped and presented once to avoid redundancy. Preliminary information for procedures shall include the following:

- a brief overview of the purpose of the procedure and definitions or explanations of necessary concepts not included elsewhere;

- identification of any technical or administrative activities that must be done before starting the task;

- a list of resources the user will need to complete the task, which may include data, documents, passwords, additional software, and identification of drivers, interfaces, or protocols;

  NOTE 1: Software dependencies may be critical during the installation process. These dependencies may include legacy software, previously installed versions, proprietary installation software.

- relevant warnings, cautions, and notes that apply to the entire procedure.

  NOTE 2: Relevant warnings, cautions, and notes shall immediately precede each applicable instructional step or group of steps. If the information in the message may affect the user's decision about whether or not to carry out an action, the message should be displayed at the point where the decision is being made.

  NOTE 3: Documentation developers should consult the style guide being used for the documentation to distinguish correctly among cautions, notes, and warnings.

11.7.2 Procedural steps

Procedural steps shall be numbered using Arabic numbers and presented in the order of performance. Alternative or repeated steps should be clearly indicated, to enable the user to easily determine which alternate or repeated steps to perform or skip and where to rejoin the main procedure. Where possible, documentation developers should use the active voice. Additional guidance on active vs. passive voice constructs is provided in *Clause B.9.6*. Existing company or client style should be observed, however.

In English, French, and other languages where the imperative exists and is culturally acceptable, the documentation developer should use the imperative to describe the user’s action.

Procedural steps shall indicate the expected result or system response.

Procedural steps shall include or provide references to documentation of the acceptable range, maximum length and applicable format, and unit of measurement of data fields for user-supplied data.

Procedural steps shall include or provide references to explanations of error messages and recovery procedures.

Quick-reference information should be included where needed in task instructions using cross-references or links.

NOTE 1: Acceptable data formats and values are commonly documented through pop-up lists.

Procedures should break down complex tasks or ideas into simpler components so that they are explained fully. However, the documentation developer should avoid making complicated tasks appear easier than they really are. If users need to understand how the software works to carry out their tasks effectively, the documentation developer should provide a brief explanation before the action or as part of the preliminary information.

NOTE 2: Procedures with many steps should be structured into short sets of steps. If tasks are long or complicated, the documentation developer should consider splitting a single task into smaller tasks, with a process topic to show how the smaller tasks relate to each other.

11.7.3 Completion information for procedures

All procedures shall ensure that it is clear to the user that the procedure has been successfully completed. This clarification may be achieved simply by providing the user with feedback about the result of the final procedure step. If the procedure contains conditional steps or jumps, it is possible for the procedure to have several logical end points. In this case, the phrase "Procedure complete" should be inserted after the feedback information for those steps to ensure that the user does not try to continue with any remaining steps. Where appropriate, it should also describe how the user should exit from the procedure and what is a typical next activity (in modular documentation, this usually is not known).

Table 3 is an example of procedures.

Table 3 — Example of procedures with the elements marked

Type of Content	**Content Example**
Title	**Creating a new contact**
Purpose	Create a new contact when you find out about a person in a client company who is not already recorded in the system.
Preliminary information	The client company must already be recorded in the system. You will need at least the client's first or last name, and phone number or email address.
Prerequisites	Before you enter the contact, use Entity=>Open to search for the surname and find out if we have dealt with them before. You might need to link notes to the previous entry, or mark the previous entry as “left” if the client is now with a different organization.
Instructional steps	To create a new contact: 1. From the Company drop-down list, select the client company record. 2. Select Contact => New from the menu. 3. From the box showing a choice of contact types, select Contact and click the OK button. 4. On the Contact panel, enter the detailed information for the contact. 5. Click the Save button.

Type of Content	Content Example
Completion information	The system will return the message, Saved, add another? See also: Setting up a subsidiary record (client notes) Deleting a contact

NOTE: In on-screen procedures, a shorter form of instruction is often used. In the example in Table 3, in step 3, “click OK” and in step 5, “click Save” would be preferred.

11.7.4 Tutorials

Tutorials should be structured into manageable task sections. For getting-started tutorials, each task section should take a maximum of 10 minutes for an average target user to work through. Complex tasks should be structured into sub-tasks for instructional purposes. Tutorials should point users to reference material so they are accustomed to finding information quickly. Tutorials should allow the user to save their work without altering the original sample files provided for training. The examples in tutorials should be tested to make sure they are correct. Where examples are used to illustrate a sequence of steps, make the examples consistent throughout the sequence.

11.8 Information on software commands

Observing the syntax of the system being documented, the documentation developer shall explain the formats and procedures for user-entered software commands, including required parameters, optional parameters, default options, order of commands, and syntax (see *Clause 12.11.1*). Documentation may be provided on the development and maintenance of macros and scripts.

Reference mode documentation shall contain a reference listing of reserved words or commands.

Examples should illustrate the use of commands. To find a solution to a problem, users usually search in manuals for concrete examples, similar to their own problem situation.

Documentation shall explain how to interrupt and undo an operation during execution of a command and how to restart it, if possible. Documentation shall describe how to recognize that the command has successfully executed or abnormally terminated. Figure 12 is an example of documentation of a software command function.

acos

Gives the arccosine of x in radians. The arccosine is the angle whose cosine is x.

Syntax

acos(x)

where x is a decimal number in the range -1 < = x < = 1

Result

An angle A in radians, in the range 0 < = A < = π

If x is out of range, ERR is displayed

Examples

acos(0.5) returns 1.047 (π/3 radians)

acos(-1) returns 3.142 (π radians)

Related functions

To convert radians to degrees: rad2deg

To find the cosine of an angle: cos

Figure 12 — Sample function description for a spreadsheet function

11.9 Explanations of data entry fields

The design of the GUI should make it clear when the user may enter data or change the displayed data. The design of the GUI should include the field description or name in terminology understandable to the user. A brief explanation of the field may be included in a pop-up or embedded help. This practice may replace lengthy separate documentation of every field or option on a screen. Documentation of data entry fields should indicate the full range of acceptable options, for example, "Passwords must be between 8 and 10 characters and contain at least one number and one capital letter."

For conditional fields, that is, fields that depend on the context or on the settings of other fields, explain what options are available in the current context. The documentation may explain the options and the conditions that apply, or include only the available options.

NOTE: Guidance documentation and tutorials may document only the commonly used options, leaving the complete list of options for the reference documentation.

The documentation may explain what the information displayed in a field means and give the meaning of the displayed value, explaining how this value relates to other possible values.

11.10 Content of error messages and problem resolution

Software systems should provide messages to users to confirm normal operation, especially when transaction processing does not return immediate results, and to notify the users of errors. Reference mode documentation shall include each error message with an identification of the problem, probable cause, and corrective actions that the user should take.

NOTE: As early as possible, the documentation developer should provide guidance to the software developers in relation to wording error messages to ensure that they are clear, concise, accurate, and useful.

Documentation should address all known problems in using the software in sufficient detail such that the users may either recover from the problems themselves or clearly report the problem to technical support personnel. A quick reference card may address error messages by referring the user to more detailed reference documentation. The documentation on resolving problems shall also include contact information for reporting problems with software or its documentation, and suggesting improvements.

Messages should describe the effect of the user's choice of action. For example, "If you press cancel, your session will end without a purchase." The system response should offer the user specific task choices such as “Delete/Do not delete” or “Delete/Cancel” rather than “Yes” or “No,” so that users know exactly what they are doing when they choose a response. System messages should give the user the chance to cancel or correct the request, before the application takes actions that could have significant consequences, such as deleting unsaved data, leaving the application, or opening the application to security risks.

Error messages should avoid multiple step solutions. If multiple steps are unavoidable, the software may refer the user to separate instructions, or add a context-sensitive help button to display an appropriate window.

The documentation should answer questions about corrective actions specifically and comprehensively with access to, or references to, the information that users need. For example, rather than instructing users to “reset the printer page size,” the documentation should tell users which function they need to use and what details they need to change, or give a link to the appropriate procedure topic.

Messages relating to security access errors should not provide extra information that would help unauthorized persons break in.

The documentation developer should be familiar with established platform and operating system information messages to avoid giving conflicting advice and the possibility of taking on the overall responsibility for the total system, rather than just the software product.

Error messages to users should avoid the use of technical jargon and system-oriented information. The documentation developer should use the following guidelines when constructing messages:

- Abbreviated forms of words or terms should not be used. Contractions may slow comprehension, especially in technical messages, and they may be difficult to translate successfully.
- Phrases that blame the user or imply user error should be avoided.
- Expressions or phrasing that implies that the application or hardware may think or feel should not be used.

11.11 Content of warnings and cautions

A warning or caution shall include the following parts:

- flag word; for example, “Warning” or “Caution”;
- graphic symbol;
- brief description of the hazard;
- instructional text on avoiding the hazard;
- consequences of incurring the hazard to the user, equipment, software, data, or service;
- user’s remedies for the adverse consequences, if any.

Established safety graphics and symbols that are used for warnings of potential physical dangers or life-threatening situations should not be used for other purposes. The documentation should use different graphic symbols for cautions, no matter how severe the consequences to the user; for example, erasure of data.

For information about formatting cautions and warnings, see *Clause 12.15*.

NOTE: Documentation developers should consult the style guide being used for the documentation to distinguish correctly among cautions, notes, and warnings.

11.12 Information on terminology

Documentation should use simple vocabulary familiar to the users. If specialist vocabulary is used in a particular application area, the documentation should use that specialist vocabulary rather than inappropriate general usage vocabulary.

Documentation shall include a glossary if terms or their specific uses in the software user interface or documentation are likely to be unfamiliar to the novice users in the audience. The glossary shall include an alphabetic list of terms and definitions. The glossary may also include information items such as field definitions and names and uses of interface elements. The glossary should define vocabulary in user's terms, not in software developer's terms.

NOTE: The terminology and the order of terms in a glossary will be changed by translation.

In printed documentation, terms should always include definitions, either in a glossary or similar reference document, or in the descriptive documentation.

The documentation developer should provide guidance to software developers to ensure consistent and correct terminology in the software interface, messages, and documentation. Documentation should be consistent with words and phrasing for similar conditions. Embedded definitions should be identical wherever they appear.

An example is given in Figure 13.

Contact

A person recorded in NIGEL who is associated with a *client* (parent company). Compare with *candidate*.

Figure 13 — Sample definition of a term

11.13 Information on related information sources

Documentation may contain information on accessing related information sources, such as a bibliography, list of references, or links to related web pages (Figure 14). The documentation should indicate whether the references contain mandatory requirements or informative background material.

Related information sources and references may include the following:

- other, relevant, documents in the current documentation set, such as the command reference guide or administrator manual;
- requirement specifications, design specifications, and applicable standards for the software and the documentation;
- test plans and procedures for the software and the documentation;
- CM policies and procedures for the software and the documentation;

- documentation for the hardware and software environment;
- explanations of the concept of operations or scientific, technical, or business processes embodied in the software.

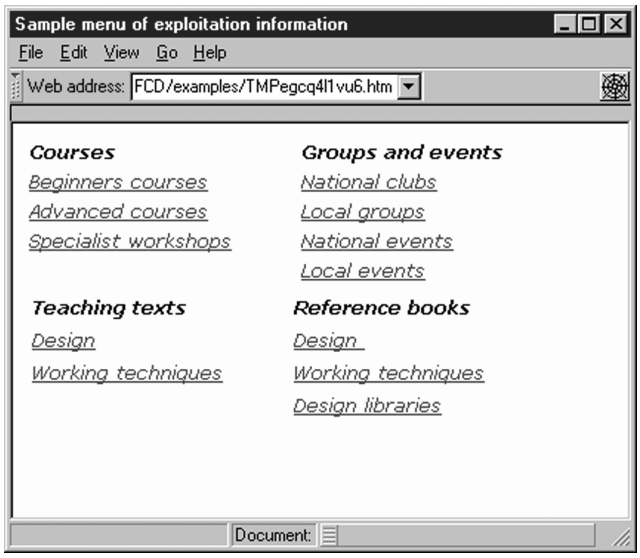


Figure 14 — Sample links to related information

11.14 User-supplied content

On-screen documentation may include facilities to enable users to add their own content or customize the existing content. The facilities provided could include adding company logos, lists of permitted values, simple annotations to existing topics, or links to the user's own on-screen information systems or documents (see also *Clause 12.17*).

User-supplied or customized content may be:

- global, which may be seen by all users;
- group, which may be seen by specific groups of users;
- local, which may be seen only by the user supplying it.

The product supplier should take steps to preserve its intellectual property rights and to clarify what liability the supplier and the customer have for customized documentation. If documentation is supplied in modifiable form, the supplier shall ensure that:

- the user may not delete information that others must see, such as critical warnings;
- mistakes may be corrected;
- the original information may be restored as supplied.

The acquirer should consider what will happen to the user-modified information when the application or standard documentation is updated, since it is not known whether this customized information will still apply to the new version of the software product. The supplier may provide an automatic method for incorporating this information into the new version of the supplied on-screen documentation.

12 Presentation format of documentation

12.1 General

This clause covers the format for presenting user documentation:

- how to present information so that it is easily understood;
- what facilities to provide for navigating the on-screen documentation;
- what styles and techniques to use in writing documents and producing illustrations.

NOTE 1: Additional information is given in Annex D for printed documentation.

The documentation format includes the selected on-screen or print media and presentation conventions for stylistic, typographic, and graphic elements. This clause specifies formats for various documentation components.

A fundamental decision in formatting user documentation is whether it will be delivered in printed or on-screen media. Printed documentation may come in a variety of formats, from quick-reference cards to comprehensive sets of manuals.

On-screen documentation may also have several formats:

- intended to be viewed on-screen, but independently of the software (the user exits the software application to view the on-screen documentation, such as help files, supporting web sites, and tutorials; sometimes viewable documentation may also be printed);
- embedded in the software, such as context-sensitive help, error messages, pop-ups, wizards.

NOTE 2: Animation or video segments may be incorporated into on-screen documents and are very useful for showing processes and concepts not easily presented in static illustrations such as human intervention tasks, for example, "how to load paper into the printer."

A range of factors determine what formats and media will be most effective for presenting information. These factors include:

- audience characteristics and information needs;
- nature of tasks documented;
- context of use;
- volume of information;
- cost of development and maintenance;
- usability;
- marketability.

The designer of the documentation should work within the constraints to give the user the most convenient documentation possible. For example, the documentation designer should not assume that all printed documentation has to be a book or a card unless this is a stated requirement. Other forms such as keyboard templates, wall charts, and labels should be considered if these will better match the users' needs.

12.2 Use of printed or on-screen formats

If a system becomes unusable, printed documentation is the only help available to the user. Whether or not on-screen documentation is provided, the following documentation shall be presented in printed form, outside of the product software:

- hardware, operating system, browser, and other software requirements for the software and the documentation;
- installation and configuration information, including recovery and troubleshooting information for installation instructions;
- instructions for starting the software;
- instructions for accessing on-screen documentation;
- information for contacting technical support or performing recovery actions available to users.

For software available electronically, the supplier may provide this information in printable format on a web site, so that the user may print this information before downloading or installing the software. For software packaged and delivered in a box, this information shall be printed and included in the box.

As a convenience to the user, other parts of the documentation may be preferred in printed form, enabling the user to learn while away from the system:

- initial familiarization information;
- long examples;
- structured reference material, particularly for advanced features of the software;
- checklists for processes.

The system should include software features designed to enable printing of on-screen documentation for those software systems designed for use when connected to a printer. A description of how to print the on-screen documentation should be included in both the on-screen and the printed documentation. The system should enable users to print:

- the current topic;
- a single illustration;
- a set of topics;
- a glossary, if there is one;
- quick-reference information, however it is provided on the screen, unless an equivalent printed document is provided.

NOTE: Users do not normally need to print out short items of on-screen information (such as those used to give the names of icons).

When on-screen documentation is provided, it shall be available for display whenever user input to the software is possible. The user should be able to perform a function and read the relevant function-specific on-screen documentation simultaneously. The user should be able to view the on-screen documentation and navigate to related software functions during system operations.

12.3 Selection of appropriate media and format

The design of documentation formats should consider each combination of audience and task. The context of use should influence the information format, for example:

- Is the information needed only temporarily, or should it remain visible (persistent)?
- Does the information have to be available at the same time as the user is working with the application?
- How quickly does the user need an answer to be provided?
- Does the information need to be available when the application is not available, such as information about finding help when the software product will not run?

The documentation designer should record the source data and design decisions for each audience/task combination in an information profile (Table 4).

Table 4 — Sample Information Profile for one task and one audience

Task: **Get a quotation for a journey** *Audience:* **Ticket Agent**					
Information needed	*Volume*	*Same time as the application*	*Persistent or temporary*	*Urgency*	*Medium*
Task instructions for getting a quotation including finding and entering departure and destination locations; entering dates, party sizes, and ticket types; checking availability; and finding a price	Up to 10 task steps	✓	Persistent	Quick	On-screen help system task topics
Reference list of location codes	Hundreds	✓	Temporary	Quick	Context-sensitive help
Reference list of ticket types	Up to 50	✓	Temporary	Quick	Context-sensitive help
Reference list of passenger categories	Maximum of 10	✓	Temporary	Quick	Context-sensitive help
Field use information for date formats	One line	✓	Temporary	Quick	Display on the user interface

12.3.1 Comparison of media

Often the solution involves the use of multiple media. The documentation designer should consider media from Table 5, plus others that might be available or invented to suit the particular project.

NOTE: Most of these media are for separate documentation. Other media may be either for separate or embedded documentation.

Table 5 — Advantages and disadvantages of various media

Medium	Advantages	Disadvantages
Removable media	Can contain many documents. Easily searched. Low production cost.	Requires suitable hardware.
Book	Easy to annotate pages. Portable, may be used away from the system.	If large and heavy, may be difficult to carry around. Can be expensive to produce. Difficult to update.
Card	Suitable for a small amount of quick-reference information. Information that is needed frequently may be on one side and information that is needed infrequently may be on the other side.	Only one side is visible. Easy to misplace.
Wall chart	Suitable for a large amount of quick-reference information.	Requires a relatively large clear wall space for display.
Notice to attach to the computer	Suitable for a small amount of information that is always needed at the computer.	Danger of cluttering the computer with too many small notices.
Leaflet or brochure	Suitable for awareness information that does not need to be retained. Can be convenient for tutorials.	Not durable for continued use (unless it is laminated).
On-screen help	Available at the press of a key or a mouse click. Suitable for quick-reference information.	Not suitable for long passages of text. Embedded documentation is only available when the application is running.
Video	Easy to absorb complex information. Widely used medium, which is acceptable to most users.	Can be expensive to develop. Needs appropriate support software for viewing. If audio is included, the user needs earphones or computer speakers.
Web site	Updatable without contacting users. Can be launched automatically from the application.	Users need web access.

12.3.2 Relationship of information displays to the application's displays

The presentation of all controls in the on-screen documentation should be consistent with the controls for the application or the system. However, on-screen documentation should be presented so that users may distinguish when they are viewing the application and when they are viewing the documentation. Use methods such as:

- a different style of window;
- clear titles for the window;
- different colors.

NOTE: The ISO 9241 series defines the ergonomic requirements for office work with visual display terminals (VDTs). ISO 9241-12:1998, *Ergonomic requirements for office work with visual display terminals (VDTs) — Part 12: Presentation of information*, gives detailed guidance on presenting information on a screen.

Information that is needed at the same time as the application interface should be displayed in at least one of the following ways:

- provide the information as embedded documentation;
- display the information and the application so both are visible at the same time;
- provide a simple method of switching between the two;
- provide the information in printed documents.

If on-screen reference mode documentation is provided, it shall be accessible from the software it documents, and shall provide a clear means of exiting the documentation and returning to the software. Software may be linked to online help, tutorials, or reference mode documentation in various ways, such as the following:

- through a help menu linked to a listing of topics or a point of entry to the help system;
- through help buttons on the software screens providing information on a particular topic (dialog box and field level help);
- through context-sensitive help and pop-up text (tool tips).

If information is needed all the time, it should not be closed when the user presses a key or selects another window.

If information is needed only temporarily, the display window should close when the user takes the next action with the application, so that the user does not have to take special steps to remove it. For example, a pop-up window or a line of text in a standard position may be used.

12.4 Context-sensitive information

Documentation may provide context-sensitive information about:

- the current field;
- the current task;
- the current application function (such as a dialog box, a transaction, or a command);
- the current message;
- a user interface object.

If it is not possible to determine the context that the user requires help with, select one of the above for display, and provide facilities for the user to choose other information.

If the information is context-sensitive:

- display an item specific to the context;
- display the relevant information at the top of the information area so that it is the first information users see, if the required information is a piece of some longer topic.

Table 6 provides some examples of access methods that may be suitable for different types of information.

Table 6 — Examples of access methods

Types of information	Access method
Task description for the current task Function description for the current function Explanation of the current message	Press a special key. Click on a help button or icon.
Names of icons Uses of icons Uses of fields Definitions of terms	Position the pointer over an object on the screen. Select an object on the screen, for example highlight, click on, or touch the object, using a different technique from that for activating the object.
Process descriptions Concepts Exploitation information Frequently asked questions Overview of the application Overview of the documentation Instructions for using on-screen information	Choose from a menu (see *Clause 12.14.2*) Find from a contents list (see *Clause 12.14.1*) Find from an index (see *Clause 12.14.4*)

12.5 Accessible documentation

Documentation shall be accessible to and usable by the expected groups of users in their work environments.

12.5.1 Provide understandable documentation

Product documentation, whether print or on-screen, should be written using a clear and simple language to the extent that this can be done using the vocabulary of the task.

NOTE: The use of technical terms are permitted where they are required to clearly explain the functionality or product.

EXAMPLE: The documentation of a CAD (Computer Aided Design) system can use terminology from the field of technical drawing.

12.5.2 Provide user documentation in accessible electronic form

All user documentation, both print and on-screen, shall be delivered in electronic form that meets applicable documentation accessibility standards. This documentation shall be provided with the product, or upon request on a timely basis and without extra cost.

NOTE: The category of “users” includes administrators. For software development software, users would include software developers.

12.5.3 Provide text alternatives in on-screen documentation

Information presented in pictures and graphics by software shall also be provided as descriptive text suitable for screen reading, printing, or Braille conversion so that it can be read by an alternative method.

NOTE: Using both text and graphics simultaneously (in the default presentation) to communicate information is often helpful to readers who use one to reinforce the other and for people who differ in terms of their preferred style of information processing (for example, visual vs. verbal).

EXAMPLE: A user can print the text portion of the on-line help and read text descriptions of any embedded graphics.

12.5.4 Write instructions without unnecessary device references

Instructions for software, whether print or on-screen, should be written so that they refer to the users' actions and resulting output without reference to a specific device. References to devices, such as the mouse or the keyboard, should only be made when they are integral to and necessary for understanding of the advice being given.

NOTE: For contexts where operation of a specific device such as a mouse is required, a generic description may not be possible. However, such specific descriptions need only occur in help about using that device, not in all contexts.

EXAMPLE 1: The task description in help does not require a user to recognise the color of a user interface element to use it, so the text does not state “click on the green icon”. Instead, the name is reported.

EXAMPLE 2: An application provides a description of how to perform tasks using as many different input/output modalities as are available (for example, mouse, keyboard, voice, etc.).

12.5.5 Provide documentation on accessibility features

Print and on-screen documentation shall provide general information on the availability of accessibility features and information about the purpose of and how to use each feature.

NOTE: It is important for users to be able to easily discover the accessibility features of the software.

EXAMPLE 1: On-line help provides a section describing features of interest for people who have disabilities.

EXAMPLE 2: On-line help explains keyboard-only use of the software.

EXAMPLE 3: On-line help describes how to adjust the font size.

EXAMPLE 4: A product has multiple color schemes, and documentation and on-line help describe which color schemes are available for people with color vision deficiencies.

12.6 Consistency of formats

Documentation shall use consistent terminology throughout a document set for elements of the user interface, data elements, field names, functions, pages, topics, and processes.

Formatting conventions shall be applied consistently throughout a documentation set. Formatting conventions may include colors, borders, indenting, spacing, and font variations (see *Clause 12.13*).

If the application uses icons or symbols, the on-screen documentation shall explain what they represent. The documentation should use the same icons to represent the same objects. The documentation shall not use those icons to represent other objects.

Where controls are needed within the on-screen documentation, for example for navigation, they should obey the conventions agreed to for the documentation.

Formatting conventions for highlighting information of special importance, such as warnings, cautions, and notes, shall be applied consistently throughout the document set.

NOTE: Documentation developers should consult the style guide being used for the documentation to distinguish correctly among cautions, notes, and warnings.

The documentation may use special formatting to identify new or changed content.

Similar material, such as sets of instructions, shall be presented in a consistent format.

If documentation is adapted for use in another operating environment, language, or culture, a common glossary and style guides for text and illustrations should be used to assist documentation developers and translators in maintaining consistency. Consideration should be given to selecting graphics and colors that are not culture-specific, to enable documentation to be more easily adapted, localized, or translated while preserving the intended meaning of the original.

On-screen documentation should be consistent with the “look-and-feel” format of one of the following:

- the software being documented;
- other software products in the same suite, or the parent software product of the suite;
- the operating system;
- other on-screen documentation with which the user may be familiar.

12.7 Consistency of terminology

Documentation projects and organizations should develop and maintain standard dictionaries for consistent terminology. Where the same meaning is to be conveyed, the writer should use the same words, avoiding elegant variation, where different vocabulary is used with the aim of making sentences sound more interesting. Using different words may lead the user to think that different meanings are intended. The problem is perpetuated if documentation is to be translated, because translators may assume that a different translation is needed for the different words and costs may increase.

Documentation should avoid vocabulary that is sometimes used imprecisely. If words that have more than one meaning are used, the documentation should make clear which meaning is intended. The precise meanings of terms that are frequently misused should not be relied upon, because the user might then assume that the alternate meaning is intended, rather than the intended meaning.

The following guidelines should help to achieve consistent terminology:

- Industry-specific terminology should be used correctly. For example, process descriptions should use the job titles that are used in the target industry.
- As few new terms as possible should be introduced. If new terms are needed, they should be short, easy to remember, and pronounceable.
- Jargon should be avoided. Where an accurate alternative to jargon exists, it should be used instead of the jargon term. If no alternative exists, the jargon term should be defined.

- If industry-specific or other uncommon acronyms are used, they should be defined in the documentation. If the documentation is on-screen, users should be able to display the definition in each topic in which an acronym is used. New acronyms should not be introduced as a form of shorthand to reduce the length of the documentation.

- Terminology that is common in the user's environment or the application area should be used, and used correctly. Terminology that is common in the documentation developers' environment, but not in the user's, should be avoided. The documentation developer should investigate the user's environment or the application area fully. If there is doubt, conduct a survey of users, or consult an authority in the user's environment.

12.8 Layout of screens and pages

The documentation designer shall prepare a consistent layout for formatting similar information.

12.8.1 Grids

The designer should determine the minimum and maximum pixel area for viewing the documentation. The designer should prepare a grid for each type of on-screen documentation window to show how the elements required in the window will be positioned, bearing in mind that users may be able to resize the window. For example, for online help, the grid should include:

- the title of the topic;

- window controls (such as close window);

- navigation controls (such as scroll bars);

- information area;

- blank space;

- non-scrolling and scrolling areas;

- signposts indicating where the current information is within the topic;

- placement of annotations;

- positioning of illustrations.

The design should use consistently named and positioned navigation controls on displays of the same type. The designer should plan what will happen to the information area, the size of text and illustrations, and non-scrolling areas when the window is resized. Figure 15 shows a sample grid for a help system navigator and a topic window.

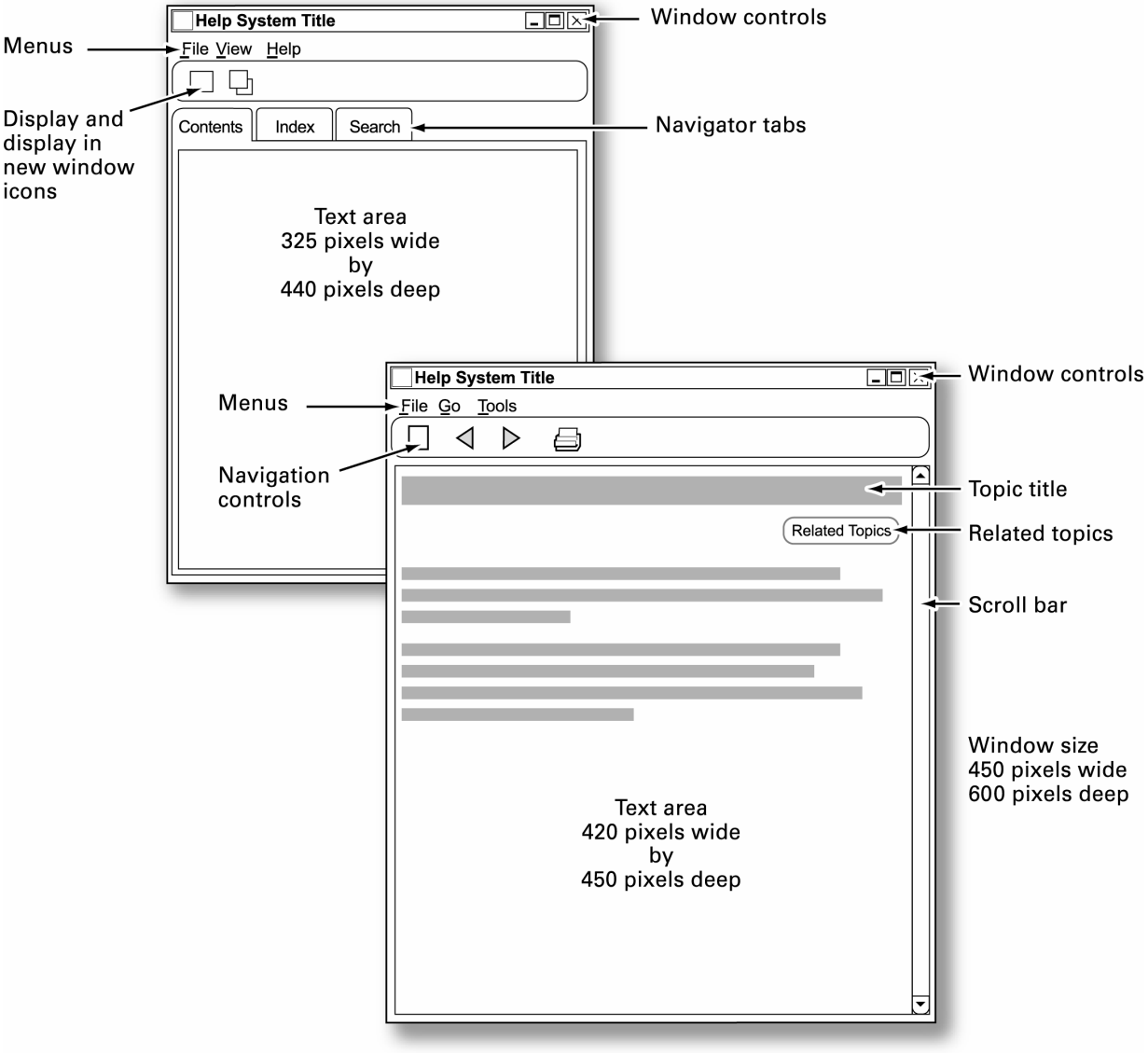


Figure 15 — Sample grid for a help system navigator and a topic window

12.8.2 Non-scrolling areas

The design should continue to display the following reference information in a non-scrolling area:

- topic title (in a web browser window this may be displayed in the title bar);
- navigation links for returning to the main topic or home page;
- navigation controls for moving to related topics.

12.8.3 Arrangement of windows

When several windows are to be visible on screen at the same time, the design should enable users to see a meaningful amount of information in each window. The on-screen application and documentation may use a fixed default arrangement of windows in which users may override the default placement and sizes of

windows and choose their own arrangement. The default size and placement of windows should enable users to see the information they need and to carry on working with the application, without moving or resizing the windows. Overlapping (cascaded) windows should not obscure navigation controls. Cascaded windows are appropriate when:

- the number and sizes of windows to be displayed at different times vary;
- the resolution of the display screen is so low that users would not be able to view meaningful amounts of information in tiled windows;
- the overlapped areas are not needed.

12.8.4 Formats for information area (text)

The format of text should assist comprehension. Large amounts of continuous text are difficult to follow. Text should be divided into short sections, with clear section headings, which help users to establish where they are in a document and to find the information they need about particular topics.

The design should use generous spacing for large amounts of continuous text. The documentation designer should consider starting each section on a new page, and putting large headers on each page to help users find the information. For printed documentation, divider pages may be used to distinguished different sections or chapters. Patches of black or a color on the edges of pages (bleeding tabs) may also be used at different vertical levels for access to different sections.

A page layout may include multiple columns, such as one column for headings and notes and another for text. Continuous text should be presented in a single column to avoid up-and-down or back-and-forth scrolling. The information area should enable users to scroll vertically through information too long to display at the same time. Users should not need to scroll horizontally to see the whole width of text or illustrations. Text should wrap around within the on-screen text area. The title of the information should remain visible when the user scrolls vertically. If continuous text is presented in tables, the design should ensure that the columns are not excessively narrow, because the sentences will be difficult to read.

NOTE: For the English language, a maximum line length of approximately 60 characters may improve comprehension.

12.8.5 Formats for headings

The levels of headings in a hierarchy of topics should be indicated by typographic conventions, such as the font size and use of bold. Indenting is often used to represent subheadings. However, when users are reading documentation, they may have only one level of heading visible at a time, so they are not able to quickly determine the position of the heading in a hierarchy by its indentation.

The design should use horizontal spacing with care, to avoid making insignificant headings appear more important than significant ones. This may happen if section numbers are aligned on the right and not on the left, with longer numbers looking more prominent than shorter numbers.

The headings should be numbered if it is important for users to be aware of their unique location in the document or of progress through a long procedure. In practice, a three-level hierarchy with distinctive formatting allows users to understand whether they are reading a chapter opening, a topic, or a subtopic.

When preparing headings, the documentation developer should consider the types of question that may have caused users to look at the documentation. Where possible, titles and headings should indicate what questions will be answered.

12.8.6 Blank space and borders

The design should clearly separate the elements presented in a window. Using space rather than drawn lines or other borders reduces visual distraction and gives the window a uniform appearance. Box text and illustrations (borders) may be used to help users to distinguish information of particular types (such as

warnings and cautions) or to keep associated information together. Boxes and borders should not be used only for decoration.

NOTE: Blank space is also referred to as "white" space. See *Clause 12.12*.

12.8.7 Vertical spacing

The design should use vertical spacing consistently above and below headings to indicate the hierarchical structure of the document:

- Except for the first heading on a page or on a screen topic, there should be more space before first-level headings than before second-level headings. Likewise, there should be more space before second-level headings than third.
- Each heading should be separated from its text with vertical space at least the size of the space between paragraphs of text.
- Run-in headings and headings with no space below at the lowest level of heading have less emphasis.

Paragraphs should be separated with vertical space. This space helps users to see where one paragraph finishes and the next one starts, and reduces the density of information. The space between paragraphs should not be illustrated with drawn lines or other presentation features.

NOTE: Where there are spaces between paragraphs, it is not necessary to indent the first line of a paragraph.

The vertical spacing between paragraphs should be greater than or equal to half the line spacing of the text, but less than or equal to twice the line spacing, provided that the vertical hierarchy of headings and text is maintained. For example, for a line size of 12 points in printed documentation, the space between paragraphs should be between 6 points and 24 points.

Select the text line spacing (the "leading") such that lines of text do not appear to be too close together, but also to ensure that users may easily move from one line to the next. A line spacing of two or three points greater than the size of text is recommended. For example, for 10-point text, select 12 point for line spacing.

The spacing between lines of text, measured from baseline to baseline, should be at least one thirtieth of the line length and at least twice the x-height (the height of a lower-case x in the chosen typeface). The designer should check this feature, rather than assuming that automatic settings will be satisfactory, because the x-heights of different typefaces vary quite significantly.

12.9 Legibility

Printed and on-screen documentation shall be legible to the user, taking into consideration the distance between the user and the documentation in the expected work environment. Documentation shall use a font style and color that is legible against the expected background (paper color or screen background color).

Upper-case (all capital) letters should not be used for continuous text of more than one sentence, except where legally required. Text in all upper case is difficult to read, and consequently the user might ignore it.

Printed text, including text in illustrations, shall be no smaller than 2.75 mm for capital letters measured at the page. The designer should not rely on internal settings of point sizes as the measure of text displayed on the screen but should measure the sizes of displayed text on all types of screens that will be used.

On-screen documentation should be legible in the users' expected work environment, which includes the anticipated combination of computer monitor or display and software graphics drivers. Legibility may be affected by output devices (monitors and printers) that are monochrome, have limited resolution, render colors differently, or support a limited range of colors. Some output devices may apply substitute fonts or special characters if the specified font is not available.

Distinctions that depend on more than two gradations of colors or shades of gray may not be visible. Because some users cannot distinguish between colors, documentation should provide text cues rather than using colors such as red or green as the only way to convey meaning. Iconic cues may also be used to support colored text.

12.9.1 Typefaces and text size

The designer should select typefaces that are readily available for the planned presentation technology or reproduction method. When selecting typefaces and sizes to use for on-screen documentation, the designer should take into account:

- the range of different display screens that will be used and the users' ability to resize the text;
- the range of default typefaces and sizes that users' systems will have available;

  NOTE 1: If the documentation uses fonts that are not installed on a user's computer, the computer will replace those fonts with default fonts.

- the different physical environments in which the documentation will be used;
- translation of the documentation: the required character sets for the required languages need to be available;
- availability of special characters (such as the copyright symbol) that are needed for the documentation;
- the character sizes in the selected font.

  NOTE 2: The same nominal font size may appear quite different in different typefaces. Always check the legibility of planned type fonts on computer displays.

The design should not use typefaces in which the lower-case L, the capital I, and the number 1 are indistinguishable, or those in which the number 0 and the capital letter O are indistinguishable.

The designer should select no more than three different typefaces and use them consistently. The selected sizes and weights of the typeface should take into account how the information will be used. The selected typefaces should assist in distinguishing between the information elements:

- headings;
- normal text (the typeface may be the same as that used for headings).

To avoid having too many different typefaces, header and footer information in printed documentation should use one of the typefaces used for the text or for headings.

12.9.2 Highlighting text

Highlighting text may be accomplished by consistently using the variant forms of the normal text typeface to convey meaning.

Bold text may be used consistently to highlight important information, such as:

- headings;
- column headings in tables;
- titles of figures and tables.

Italic text may be difficult to read, particularly on-screen, for more than a few words. Italic text may be used for the following:

- command variables;
- introduction of new terms;
- references to article and book titles.

Underscored text should be used only to represent active hyperlinks. In on-screen documentation, the link should also change color when selected. Underscore should not be used for headings that are not active areas of text.

Enclosing text in special characters, such as square brackets [ ] and braces { } may be used to identify special types of text, particularly user-entered text.

The following methods of highlighting should be avoided as visually distracting for the user:

- underlining that crosses the descenders of characters;
- blinking and animation (words should not be highlighted by blinking the text itself; if necessary, a marker, which may blink, should be displayed next to the text);
- complete words or phrases in upper-case (all capital) letters.

12.9.3 Lines of text

For scripts read from left to right, the text should be aligned on the left, but not on the right (ragged right).

NOTE: This format is not used in this International Standard, as it is not software documentation, and follows ISO standard formatting rules.

Non-breaking spaces should be used to improve comprehension. For example, there should not be a line break between a quantity and the unit of measure:

Poor formatting	Preferable
The sampling rate is 1 kHz	The sampling rate is 1 kHz

12.10 Formats for lists

Lists are useful for:

- sets of options, so that users may see at a glance which option they need;
- activities that need to be carried out in sequence;
- information that may be viewed as a series of separate points.

Strictly speaking, lists are text rather than illustrations. However, lists are sometimes enclosed in borders for emphasis and may be treated as figures with titles.

Presentation conventions for lists should support users' comprehension of the information. The format of a list provides space between items to focus attention on the individual items. Lists should have no more than two nested levels of subdivision for easier comprehension; an excessively subdivided list should be separated into several lists. If a list is too long to be visible at once on the screen or on a single page, it should be subdivided into separate pieces of text, accessible using the chosen navigation techniques.

The designer should establish consistent conventions for identifying items in lists and sub-lists and for punctuating (or not) the end of each list item. Generally, lists should be introduced by bullets or other special characters. Distinct characters and indentation should be used for second-level lists. List items should be introduced by numbers only if the sequence of events or the exact number of items in the list is important. Documentation should not use Roman numerals because they vary so much in length, are harder to read quickly, and not all cultures are familiar with them.

NOTE: The documentation developer should use Arabic numbers for a set of steps that the user must follow in sequence. If a procedural step includes substeps, these should be indicated by sequential letters beginning with the first letter of the alphabet in which the document is written.

If list bullets are being used as navigation controls (radio or option buttons), their appearance should be distinct from bullets in non-active areas of text.

12.11 Formats for representing user interface elements

Graphical user interface (GUI) elements of the software, such as buttons, icons, and variable cursors and pointers; special uses of keyboard keys or combinations of keys; and system responses shall be represented in documentation by consistent graphic or typographical formats so that the various elements are each distinguished from the text. The documentation should include a representation of the element, its purpose, and an explanation of its action (functional consequence), with examples of actual operational instances. On-screen documentation may include pop-up text labels for GUI elements.

Documentation incorporating text from software dialogs should copy the text exactly, character for character.

12.11.1 Representing control and command input

Documentation formats for user-entered commands or codes shall clearly distinguish between literals (to be input exactly as shown) and variables (to be provided by the user). For example, the variable *filename* might be used as a general term for which a real filename has to be substituted. Documentation should represent variables in a distinctive type (for example, italics).

Formal notation, such as the use of brackets, braces, greater than (>) and less than (<) characters, and other marks, shall be defined in every document that uses it, with the exception of […] for references in a bibliography. Quotation marks shall not be used in command representations unless the user should input them literally.

EXAMPLE:

The following message is displayed on the screen:

```
Enter first command
```

To create the first text record, type the following, substituting the name you wish the record to have in place of *record-name*, and then press Enter:

```
create record-name
```

12.11.2 Representing special keyboard keys

The documentation shall establish consistent conventions for representing special keyboard keys and explain the conventions in documentation for software that uses special keys for input. In text, the documentation should distinguish terms or symbols referring to individual keyboard keys from the same characters used in other contexts with a convention, such as one of the following:

- a special type style or font; for example, use of upper-case, small capital letters, or bold characters;
- pictures of key tops; if the pictures are larger than the normal text of a document, the pictures may be placed in the margin, in a list, or in a table;

- special characters such as < and > to enclose the term or symbol. Check carefully that those characters themselves do not need to be represented in the same way. Avoid text such as < > > to represent the > key or < " > to represent double quotation marks.

Documentation should address variations in keyboards or data entry devices in the expected users' work environment. Documentation should indicate whether different keyboards, with different characters on the key tops, might be used. The documentation should set consistent conventions for the names of keys such as Enter, Shift, and Tab. Instructions and help for software should be written so that they refer to the users' actions and resulting output without reference to a specific device. References to devices, such as the mouse or the keyboard, should only be made when they are integral to and necessary for understanding of the advice being given.

A convention is frequently needed for representing a space. The documentation should use a character not used for other purposes in either the documentation or the software.

Illustrations of keys should use dark characters on light background. Reverse text (light characters on a dark background) is not easily legible in illustrations.

12.12 Use of color

Although color is a valuable method of conveying meaning, the designer should consider whether color is essential to the documentation. In some documentation, as for graphics or visualization software, exact representations of color are indispensable, and monochrome documentation would be inadequate. In most other documentation, color is used for emphasis, but the software may also be used by color-blind or visually impaired users who may not be able to perceive or distinguish certain colors. In this situation, the design should not use color alone to convey meaning; but should employ another form of highlighting as well, such as different types of hatching or shading patterns in illustrations.

Colors may easily distract the user's attention from the working context and may therefore decrease the user's performance. Two or three colors are usually sufficient. The designer should start with a monochrome design and add color where it provides additional emphasis to the other highlighting techniques being used. Color choices should be validated when converted to grayscale or black-and-white, to ensure that there is enough distinction in contrast and tonal value.

The designer should select a neutral background color because the perceptibility of objects depends on background color. The designer should take care when placing colored objects on a non-white background Use solid color (such as white) for background or to fill areas of the display. The designer should also avoid color causing patterned effects on the screen.

When using color, the designer should:

- determine the color palettes available on the users' display devices. Might some or all users have only monochrome displays? What range of colors will be available? Choose a color scheme and palette that is suitable for all the users. Use only colors from the standard palette of the documentation delivery tool (if there is one), to display colors correctly on users' screens. Avoid multiple or alternate palettes; the user may not be able to display them at the same time;

- determine whether the users are likely to print the documentation on monochrome printers, or if they are expected to have color printers. Be aware of how the colors will appear when the content is printed on color and monochrome printers;

- test which colors and combinations of colors reproduce and display best and are least likely to cause confusion;

- where consistency of color is important between the embedded and printed documentation, check the proposed colors thoroughly;

- consider the effect of the use of color on a color-blind user. Avoid selecting color combinations that are often indistinguishable, such as blue and green, or red and green;

- consider cultural interpretation and usage of colors. For international audiences, consider the cultural connotations of different color combinations;

- choose a printing method for print documentation capable of handling color. Consistent reproduction of color may be difficult; some methods distort color more than others. In particular note that the colors selected using the RGB gamut available on a computer monitor or an inkjet printer may not look the same when printed on a device that uses the CMYK gamut (offset or color laser technologies);

- remember that full-color printing is usually more expensive than black or one color plus black. Verify cost estimates for reproduction with printing organizations before making a decision.

12.13 Navigational features

Features for navigation include chapter and topic titles and headings; page or screen titles; chapter, title, page, and screen numbers; tabs; page headers and footers; bookmarks; links; cross references; navigational icons; and buttons. Documentation shall include explanations of unusual, flexible, or complicated system- and document-specific navigational features.

NOTE: The analysis and design process will have set usability goals for speed and ease of access, for example, to be able to find the right information within a certain amount of time or using a certain number of steps (see *Clause 6.4*).

Navigation features should indicate the locations to which users may move from their current location. If facilities are provided to link to topics in different parts of the hierarchy, the documentation may display where the topic is in the hierarchy. Where users move between topics in on-screen documentation, the documentation may indicate where the current topic fits into the total structure (a "breadcrumb trail" or path from Home or the main menu). For example, in the topic title, the documentation should include words or a diagram explaining where it belongs in the structure, or highlight the new position in a contents list.

Navigation features shall enable documentation users to go to the following locations:

- back, to return to the section/page most recently linked from;

- next, next logical topic/page in the sequence of topics (if any);

- previous, logical topic/page just prior to the one being viewed (if any);

- table of contents (if any) or top-level menu;

- index (if any).

Documentation should provide a simple method of exiting from the on-screen information. The exit method should be the same throughout the on-screen information.

12.13.1 Using formats to indicate position within a topic

If a topic is longer than may be viewed in a single window, the documentation should give the users a clear way of knowing where they are within the topic. The following techniques may be applied:

- Topics of the same type should be structured in the same way, so the user knows where to look for certain types of information. For example, give task descriptions headings such as “Purpose,” “Instructions,” “Examples,” and “Related tasks.”

- Headings should be numbered so that the user is given a view of their place within the structure, rather than simply a measure of how far they have moved through the total amount of information.

- Scroll bars or other indicators, such as section numbers and end markers, should show the user how close they are to the end of the information (particularly helpful for lengthy sets of instructions).

- Long topics should be split into smaller topics to be viewed one after the other. Use menus to show the full set of topics and use markers to show which topics have been viewed already.

The documentation should always make it clear whether there is more information to follow and how to access it.

12.13.2 Finding the same information again

Features for navigation shall be provided such that users may determine their location within the printed or on-screen documentation. In printed documentation, each page shall have a unique page number. In on-screen documentation, each page or screen shall have a unique identifier (alphanumeric or caption) that can be seen by the user. Navigation features should remain accessible in a static location if on-screen documentation allows scrolling through the text. If the information area is lengthy, the title should continue to be displayed when the text scrolls.

Documentation should enable the user to return to information later in the present session or in another session. This return may be accomplished by these techniques:

- a cumulative list of the topics the user has looked at either in this session or in this and previous sessions, from which they may select topics;

- bookmarks that enable users to mark places they may want to return to;

- a clear structure that allows users to follow the same paths;

- annotated or highlighted menus that show which topics the user has looked at.

Techniques used should be consistent and simple to use. For example, if a list is being maintained, provide the same method for the user to access this list wherever it is available.

12.13.3 Viewing topics in sequence

If there is a natural sequence for some groups of topics (a browse sequence), the documentation should provide a method for the user to view the topics in sequence both forwards and backwards.

If the method provided requires the user to use particular key combinations to move forwards and backwards, these should be displayed on the screen.

In such sequences, the on-screen documentation should clearly indicate to the users when they have reached the start and end of the sequence, for example, by de-emphasizing or dimming the “Next” or “Previous” control, or including a marker.

12.13.4 Formats for active areas

Interactive text of different types should be clearly distinguished, so that users understand what will happen if they select the text. The following need to be distinguished:

- active areas of text causing a link to another topic, replacing the current topic;

- active areas of text causing additional information to be displayed as well as the current topic (usually in a separate or pop-up window).

The designer should select a distinct method for representing each type of active area of text, such as one of the following:

- The shape of the pointer may change as it moves over the active area.
- The active areas may be indicated by using different methods of highlighting or different methods of displaying the area.
- Different icons may be used for the different actions.

NOTE: Text marked with a solid underscore should be used only to represent active hyperlinks (jumps). In on-screen documentation, the hyperlink should also change color when selected.

12.13.5 Linking information

Links should provide information that the user expects in one jump, rather than requiring that the user follows one or more additional links to reach the required information.

If the destination is outside the documentation, the users should be notified that they are leaving the documentation. The documentation should provide users with an alternate way of locating the information, in case the external link has been broken or the destination removed.

Links between related topics shall be bidirectional, so that whichever topic the users access first, they may jump to the related information on the other topic.

Links shall provide a clear indication of the destination of the link. For example, use More troubleshooting tips rather than Click here.

If facilities are provided for the user to link by selecting active areas of text, the documentation should distinguish this text from the surrounding text and from active areas of text used to obtain clarification. Clarification elements should be collected at the start or end of topics under headings such as “Related topics” or “See also.”

12.14 Documentation formats for finding information

As presented in *Clause 10.4*, documentation shall contain features to provide access to information, such as a table of contents, an index, and search capabilities.

12.14.1 Table of contents

The table of contents shall list the headings of the chapters or topics of a document with an access point for each (its initial page number or an electronic link). Documents with fewer than about eight pages after the identification data may omit the table of contents. In a printed document, the table of contents shall immediately follow the identification data (see *Clause 11.3*).

The table of contents may be comprehensive or simple. A comprehensive table of contents lists chapter or topic titles and headings down to the third level. A simple table of contents includes only the first-level headings. At least one volume of a document set shall include a simple table of contents for all volumes in the set (see *Clause 10.4*).

In large printed documents, both types of tables of contents may be used to provide the user with more convenient access to the required information. Printed documents may also include secondary comprehensive tables of contents at the beginning of each chapter or topic to assist navigation.

On-screen documentation may display tables of contents in expandable and collapsible formats (tree structure) to provide top-level and detailed access to headings without excessive scrolling. Secondary, comprehensive tables of contents may be accessible through menus, expandable lists, or secondary windows.

The table of contents shall include all portions of the documentation, including front matter that follows the table of contents and back matter (for example, appendixes, glossary, and index). The headings in the table of contents shall be identical in wording to those in the document, including chapter or topic numbers. The format of the table of contents shall distinguish the hierarchy of headings by consistent typography or indentation. In printed documentation, the table of contents shall list the headings in the same order as in the text. For on-screen documentation, the table of contents should order the headings according to browse sequence, task, topic type, or some other logical criteria.

In on-screen documentation, the contents list may be displayed in its own navigation window, or as a separate frame in the documentation window. The contents list should remain displayed while the user is selecting and reading topics in the topic window or frame, until the user chooses to close it. Each item in the contents list should be linked directly to the information it describes.

Icons may be used in contents lists to indicate the different types of information contained in different sections of the documentation, or to indicate different statuses. Both the icon and the associated text should be active links, not just the icon.

Figure 16 shows an example of a contents list, partly expanded.

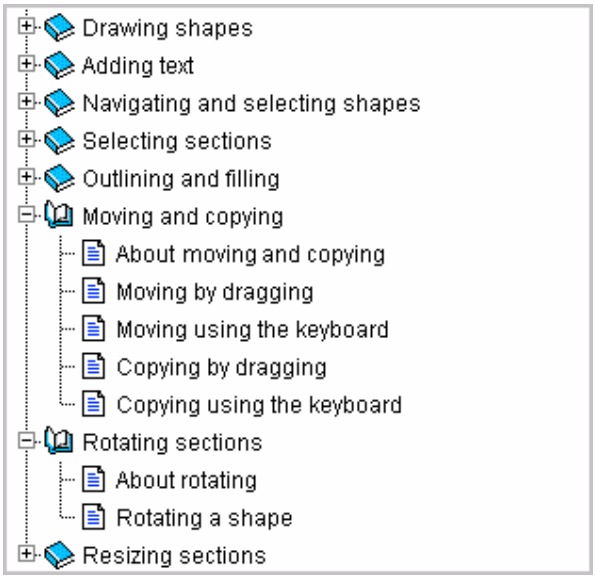


Figure 16 — Example of a contents list

12.14.2 Menus

A menu is a secondary table of contents in software and on-screen documentation (Figure 17). Menus should enable users to select the level or topic and automatically navigate directly to their selection. Menus may be displayed in the information window or frame and may be replaced by a topic when the user selects one of the entries. Menus normally display only one or two levels at a time from a hierarchical structure.

Figure 17 — Sample text menu

If a menu uses icons and text, both an icon and the corresponding text should be active links.

The names used in menus should indicate clearly what information may be obtained using each entry and should be consistent with the topic titles and headings. The menu design may include a brief (pop-up) description of each entry on the menu display. Menus should not include lengthy explanations.

To expedite navigation to the desired topic in a hierarchy of menus, the designer should carefully allocate the number of entries to show at each menu level and the number of levels of menu. To reduce the number of levels, the design should offer more choices at each level or display several levels at once on each menu.

12.14.3 List of illustrations

Documentation should contain a list of tables, a list of figures, or a list of illustrations (including both tables and figures) if the document contains more than five numbered illustrations and the illustrations are not visible at the same time as text references to them. The list of illustrations shall list the illustration numbers and titles with an access point for each (such as its initial page number or an electronic link). The titles in the list of tables, figures, or illustrations shall be identical in wording to those in the document, including table, figure, or illustration numbers. For information about formats for illustrations, see *Clause 12.18*.

12.14.4 Index

An index is an alphabetical listing of keywords, graphics, or concepts with an access point for each. An index is an important information retrieval device and should be constructed in a way that makes it easy and consistent for users to locate the information they seek. Printed documents of more than about 40 pages should include an index, whose access points may be page numbers, topic numbers, illustration numbers, or references to another index entry. On-screen documents of more than about 40 topics shall include either a search tool (see *Clause 12.14.5*) or an index whose access points are electronic links. An index entry may cross-reference another index entry; however, the referenced entry shall give an access point to the documentation and shall not point to a third index entry.

Following is guidance on good practices for indexing documentation:

- Indexes should include alternative terminology from that used in the documentation. To determine the different terms users might choose, consider why they might want to look up the topics. Use words that users are likely to look up and also list the topics in the document. Use synonyms to provide alternate terms for access. Thoroughly indexed documentation typically has two to five entries per topic.

- The preface, the appendixes, warnings, cautions, and notes should be included. Non-textual items such as graphics (and multimedia in on-screen documentation) should be indexed.

- Overly general keywords and user-descriptive terms for entries should be avoided, and the main word should be placed first. For example, for the text heading, “using file manager,” use the index heading, “file manager, using.”

- Entries should be double-posted. For example, a topic that is indexed as both “site licenses” and “licenses, site” is double-posted.

- The importance of information should be indicated by placing minor key words under major ones. Entries with multiple references in the document should be broken into sub-entries that indicate which aspect of the topic is covered. For example, each entry should clearly indicate whether the information is simply a definition, a description of a function, or a procedure. Detailed primary entries should be used rather than secondary or tertiary entries.

- Optionally, major page references may be identified by marking the page number in bold.

- Optionally, for document sets, a global index may be included in addition to the indexes in the individual volumes. The global index provides users with the means of looking for information in a single index, instead of looking it up in the indexes of the individual documents. Location references for each entry should include an abbreviated document title in addition to the page numbers.

12.14.5 Search capability

On-screen documentation shall provide a method of locating words in the text. Electronic search capabilities may include full text search of the document or document set; ranked returns from the search indicating their closeness to the requested topic; search for words in illustrations; keyword search; finding a text string in the current topic; a Boolean search; and the restriction of a search to specific chapters, topics, or pages. Search for common, non-meaningful words (such as "the" in English) should be turned off.

The designer should determine, preferably directly from users, what terms users are likely to use when searching for information. These should be the menu headings and table of contents entries and index entries as well.

The design should ensure that the cursor is in the search text box so that the user may begin typing the search criteria without first having to place the cursor in the box.

12.15 Formats for warnings, cautions, and notes

Warnings, cautions, and notes shall be displayed in a consistent format that is readily distinguishable from ordinary text or instructional steps. The flag word (for example, “warning,” “caution,” or “note”) shall precede the accompanying text. The term “note” shall not be used to identify hazards. Warnings and cautions shall be identified by consistent and distinct graphical symbols, for example, an exclamation point or lightning bolt inside a triangle.

NOTE: Documentation developers should consult the style guide being used for the documentation to distinguish correctly among cautions, notes, and warnings.

In on-screen documentation, warnings, cautions, and notes may be displayed in a format similar to that used in printed documentation. Alternatively, a message displayed in a secondary window may be used to notify the user about the results of a command, to alert the user to a condition or situation requiring a decision before proceeding, or to inform the user of a serious condition that requires intervention before the process can continue.

12.16 Format for instructions

Instructional steps shall be consecutively numbered. A consistent numbering or lettering convention should be established for major tasks, sub-steps or actions, alternative steps, and repeated procedures. If required, any response of the system to an instruction should be described briefly, should not be numbered, and should be separated from the preceding instruction by a blank line.

NOTE: Arabic numbers should be used; traditional numbers may be used in languages such as Japanese read from top to bottom.

12.17 Formats for user-supplied annotations

If facilities are available for allowing the user to annotate the supplied information, the documentation should display at least one of the following at the place where the user added the annotation: the annotation itself, an icon, or a signpost. If the annotation is displayed, it should be distinguished from other information. For example, if it is text, it should be highlighted in a different way or enclosed in special symbols. The information itself should be distinguished, whether displayed all the time or only when the user asks to see it, from the supplied text by using a different form of presentation, such as a different typeface.

12.18 Formats for illustrations

12.18.1 When to use an illustration

Illustrations include figures and tables. Documentation may use several types of figures:

- a picture, which represents the actual appearance of physical objects; examples are:
    - photograph;
    - line drawing;
    - picture of a screen display.
- a diagram, which gives a spatial representation of abstract concepts; examples are:
    - organization chart or a tree structure;
    - line graph;
    - bar chart or pie chart;
    - flowchart.

Illustrations should be used:

- to draw attention to important information (an illustration is often the first element that the user looks at on a page or a screen);
- to represent processes, relationships, hierarchies, networks, structures, shapes, positions, statistics, trends, directions, proportions, correlations, mappings, and other concepts;
- to show the appearance of a physical object, which helps the user to identify the parts of an object or the object itself;
- to make information easy to remember.

The designer should consider whether:

- an idea or relationship may be better described using a picture than by using words;
- attention needs to be drawn to some detail;
- items need to be identified by their appearance.

Graphs and charts should be used if they will convey the information more clearly and memorably than words or numbers. Graphs and charts are usually more effective in communicating broad differences or general trends than in conveying specific quantities. Tables are more appropriate for presenting specific quantities.

For each type of illustration, the designer should examine the profiles of the target users to ensure that all types of user will understand it. It is sometimes more difficult for users to learn how to interpret diagrams than to understand the same information in the form of a list or a table. For example, use flowcharts only if it may be deduced from the profiles of the target users that flowcharts will be understood.

NOTE: Designers should provide an equivalent for all illustrations to make their content accessible to those who are visually impaired.

12.18.2 Level of detail in illustrations

The level of detail shown in an illustration should be only what is necessary to achieve its purpose. The text should refer to information represented in an illustration but should not repeat the same information. Text should be used to clarify or explain an illustration.

NOTE 1: Users of reference information often look at the illustrations and tables without reading the related text.

The illustration should include the needed information. Users should be able to see the whole of each diagram in one view. In on-screen documentation, illustrations should be sized so they are legible and viewable in their entirety, without scrolling, on the expected viewing device. The designer should consider simplifying or showing only salient features of a large graphic so it is visible at one time without scrolling.

NOTE 2: On a screen, if sufficient detail cannot be seen when the whole illustration is visible, provide a method of zooming or the capability to expand parts of the illustration.

12.18.3 Identification of illustrations

In documentation with more than five illustrations, each illustration should have a unique number and title (see *list of illustrations, Clause 12.14.3*). These illustrations should be included in the list of illustrations. Informal illustrations, referred to only once in the text or having no text reference, may be untitled and unnumbered.

12.18.4 Consistent presentation of illustrations

Representations of GUI elements in documentation should be consistent with the version of the software being documented.

If the documentation product is to be used by an international audience, the image should be consistently understandable and acceptable in various cultures.

If the document will be translated, the illustration should be arranged so that text areas may accommodate additional text length that may arise in translation. Allowing for twice as much translated text as English text should be sufficient for a wide variety of languages. Informal illustrations used only for incidental purposes should not contain text if the product will be translated.

NOTE 1: If text is rasterized for publication of illustrations in print or on-screen documentation, the original application files for illustrations should be retained so that text may be edited or translated and new rasterized illustrations recreated.

NOTE 2: Some target languages may not have a term that corresponds to a term in the source language. In these cases, the translated text may be many times longer than the source text.

Illustrations should be legible when printed or shown on the expected types of screen. On-screen lines may appear thicker than the corresponding lines printed on paper; ensure that lines intended to be of different thickness appear different on screen. The documentation developer should ensure that characters are clearly readable (see *Clause 12.9*). Illustrations should be tested on all of the expected types of screen.

The format of illustrations of similar content shall be consistent for scale, screen size, fonts, line thickness, and use of color. For diagrams (and line drawings if they are appropriate), the designer should set conventions for the following, to ensure that all the illustrations within a document are consistent:

- Line thickness: Drawn lines should be thick enough to be clearly visible and to reproduce well using the intended reproduction or presentation technologies. If the diagrams include text, lines so thick that they detract from the main message should be avoided.

- Typefaces and sizes of text: The same typeface should be used for all diagrams and illustrations.

- Sizes of arrowheads and other connectors: The size should be appropriate to their importance in the diagram (generally visible but small).

  NOTE 1: ISO 4196 defines recommended arrow styles and their uses.

- Shading: Shading should be used only to convey information; decorative shading, such as large areas of grey, should be avoided.

  NOTE 2: Shading causes difficulties with low-resolution reproduction and presentation technologies.

- Presentation of titles: The illustration title should be placed either above or below the illustration, and either bold or plain text should be used consistently for both table and figure titles.

12.18.5 Placement of illustrations

Illustrations that accompany text should appear adjacent to their first reference in the text, so the associated text and illustration may be viewed simultaneously. The grid (see *Clause 12.8.1*) should specify how illustrations should be placed within the information area. Illustrations of the same type, serving the same type of user need, should be displayed consistently.

If the width of an illustration exceeds the margins of the page, it should be placed on a separate page and rotated 90 degrees to the left.

12.18.6 Illustrations of screen displays

Illustrations of screens or other software displays may be used:

- to show general and specific features of the software product's screen displays;

- to help users to check that they have reached the correct stage in a sequence of instructions.

Illustrations of screen displays in documentation should be easily distinguished from the live software product. Fewer illustrations may be needed in on-screen documentation that allows links to the actual software.

NOTE: Documentation designers and developers should be cautious of providing too many illustrations. A simple task may appear long and complicated if too many illustrations are used. User documentation is not ordinarily intended to be the system repository of images of each screen.

If a screen illustration is used, the text should state clearly whether the illustration shows the display before, during, or after an activity in the accompanying text.

Illustrations of entire screen displays and portions of screen displays, such as menus or dialog boxes, should be scaled consistently throughout a single document. At most, two different scales should be used: one for entire screen displays and one for portions of screens (Figure 18).

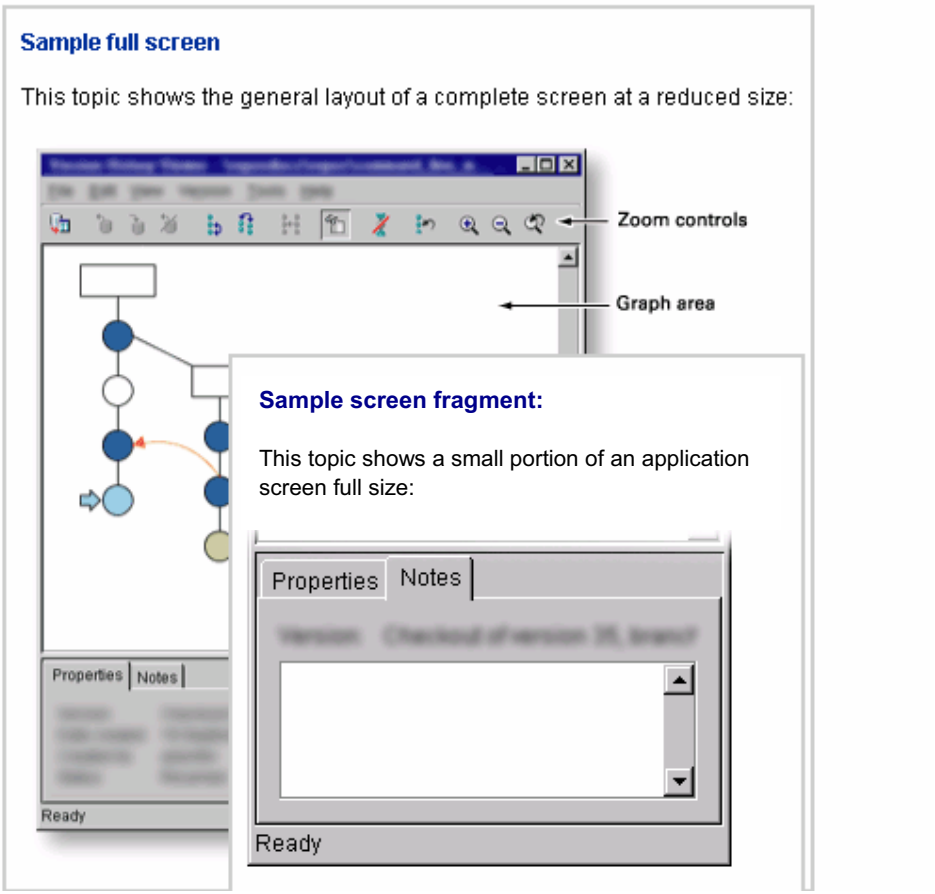


Figure 18 — Use of two scales for screen displays

12.18.7 Illustrations of printed output

Illustrations of printed output (for example, lines of programming code) should be selected based on why the users are shown the output, in particular whether they need to read all the text on it, or simply to see what sort of data it will contain. These considerations should influence:

- the amount of output that is illustrated;
- the type of data that is shown;
- the size of the text and hence the size of the illustration.

NOTE: Documentation developers should include dates in sample printed output only if necessary to the reader's understanding of the material. Dates may give the reader a false impression of the age and hence the relevance or accuracy of the material, especially if the text sample remains unchanged through several releases of the documentation.

12.18.8 Tables

The design of tables should simplify users' access to exact data. In a table, items that are to be compared should be shown in adjacent rows rather than in adjacent columns. Otherwise, the order of rows and columns should simplify reference, such as by alphabetic sequence, by amount, by time, or by frequency of use.

Borders between rows or columns may guide readers to important data. Borders on every cell may distract, but house style should determine their use.

When the text is translated, the length of the text in tables will change and the translated headings and text may require more space.

Long tables that cannot fit on a single page shall repeat the table title and column or row headings on each page or two-page spread. Long tables that span multiple pages should also be identified with a sheet number, for example, “Table 15. Metric units (sheet 2 of 4).”

Large tables may need special attention to display in a meaningful way. If the user has to scroll through a table displayed on-screen, or if a printed table will not fit on one page, the design should ensure that the table title, column headings, and row headings are always displayed. The designer should test the effect of resizing a window, some text may not be visible or the alignment of the text in the columns may be altered.

If there are problems, the designer should consider:

- splitting the table into several smaller tables;
- presenting the table as a series of lists.

If the width of a table exceeds the margins of the page, it should be placed on a separate page and rotated 90 degrees to the left.

12.19 Icons and signposts

12.19.1 When to use icons and signposts

An icon represents an object, action, or concept in a smaller area than an equivalent text label or description. Icons save space in the window, allowing more information to be displayed. Also, appropriate pictures often show the meaning more quickly than many lines of text, and they do not require translation.

NOTE 1: If graphics and symbols are not self-explanatory, users have to interpret and learn the meanings of icons, whereas words are immediately understood. Users need to learn the whole vocabulary of icons used in the software and in the documentation. Consequently, users may not find icons as easy to use as words.

A signpost is a piece of text, a symbol, or a small graphic signaling a specific place. Signposts should be used:

- to help users understand where they are in the structure of some information; for example, to show them what level they are at in a hierarchy;
- to signal a particular type of information; for example, some instructions, a topic, or a concept.

NOTE 2: The ISO/IEC 11581 series addresses the design and presentation of common icons for software interfaces.

12.19.2 Design of icons and signposts

The designer of graphics for use in icons or as signposts should consider the following guidance:

- An icon should not distract the user's attention, but should be large enough to be selected without excessive precision with the pointer.

- The documentation should use the same symbols or icons as the software product so that users have a simple and consistent method of relating the documentation to the software; alternative sets of symbols or icons should not be invented purely for the purposes of the documentation.

- An icon should represent a real-world object (printer, filing cabinet, file folder, sheet of paper) or actions if possible.

- The icon should be recognizable in all cultures where the product will be used. For international audiences, the use of culturally dependent graphics should be avoided.

  NOTE: If culturally dependent graphics are used, they should be replaced as part of the translation or localization processes. This practice imposes extra burdens on maintaining consistency in the software product and the documentation.

- Icons should uniquely represent objects or functions. Different symbols or icons should never be used for the same object or function; different objects or functions should not be represented by the same symbol or icon.

- The design should emphasize the graphical elements that distinguish this object from other objects.

- The design should simplify the representation by eliminating or de-emphasizing the elements that do not contribute to the object's identification. As few graphical components as possible should be used; excess detail obstructs recognition and comprehension.

- The design and construction of icons and signposts should be consistent for the software product or organization, through the reuse of visual metaphors, scale, orientation, color, and location of graphical components. Icon and signpost standards should be established for the software product's documentation and used consistently.

- Text should not be used in icons or signposts unless a translated version may be substituted when the documentation is translated.

- If an icon represents changes in the status of an item, the metaphor should be used consistently. For example, if a picture of a book represents a batch of information, pictures of a closed and an open book could be used to represent two different states. Similarly, a picture of a page could be used to continue the metaphor to a different level.

- The suitability of the graphics should be assessed during user tests to ensure that users interpret them as intended (see *Clause 8.3*). For example, if a picture of a telephone receiver is used, users might be unsure whether it represents on-screen information about dial-up connections or the numbers to dial for customer support.

12.19.3 Displaying the names of icons

The meanings of icons or codes should be reinforced by providing a method for displaying their names, such as "tooltips." Only the name should be provided, such as "commit order" or "display empty form," using the same types of names as would be used on a menu.

If the names of the objects or functions represented by icons are displayed, the designer should consider the following questions:

- Are the words sufficiently concise to save space and avoid concealing other elements on the interface?

- Will users find the icons memorable and easy to use when they have become familiar with them?

- Should the users be allowed to select whether to display the words permanently in addition to the graphics?

- Will those names be translated?

For icons, the name should be displayed:

- permanently next to (above, below or to the side of) the icon, with both the name and the icon active;
- temporarily when the user needs it, for example, in help status line, bubble, or tooltip.

If a user can see the names of user interface elements on the screen, names for all such elements should be provided, not just some of them.

12.20 Documentation packaging

The packaging requirements for the entire product affect the documentation design. The designer should consider at least the following:

- whether the software product and the documentation are to be packaged together;
- whether there is a house style for packaging;
- what medium will be used to distribute the product;
- what methods will be used to physically ship or electronically deliver the product;
- whether the packaging should be the same as used for previous versions or for similar products.

Annex A
(informative)

User documentation style guide content

A style guide provides standards and guidelines for the documentation developers on the specific forms of writing, illustrations, and layout to be used in a particular document. This clause sets out the content of a style specification. Much of the information in a style guide is language-specific.

The documentation developers should also be given access to the audience, task, other analysis and design documentation.

A.1 Writing style

Annex B contains recommendations about the topics to be addressed in a guide to writing style. Specialized standards and style guidelines may be added to the style specification or referenced from the documentation plan.

A.2 Language

The language should be specified, with the country-specific variant if appropriate; for example, French (Canadian).

A.3 Spelling

For languages where it is necessary, a spelling dictionary should be specified, optionally with a list of exceptions.

NOTE: A national standard dictionary appropriate for the nationality of the majority of the audience should be specified. If possible, the dictionary should be available as a spelling checker file.

A.4 Grammar and usage

A grammar and usage style manual should be specified.

NOTE: A grammar and usage standard appropriate for the nationality of the majority of the audience should be specified.

Annex B
(informative)

Writing style and techniques for user documentation

This annex describes English grammar and usage. A national body may prepare its own version of this annex to match the rules and conventions of its own language.

B.1 General

This annex covers the writing and illustration styles and techniques appropriate for user documentation.

User documentation should be clear, unambiguous, and easy for users to understand:

- The vocabulary should be straightforward.
- All terminology should be explained simply.

If users are familiar with a particular style for a particular type of information in printed form, for example, specifications of commands, that same style should be considered for the on-screen documentation. Conversely, if users are familiar with a particular style for a particular type of information on the screen, that same style should be used in the printed documentation unless there is a very strong user case for a different style.

The writing style should be as specific as possible to the user's level of understanding and to the current context.

The writing style should be simple and straightforward, so that users are not distracted by complicated sentence structures or complicated vocabulary. The meaning should be clear and easy to assimilate, but the documentation should not patronize users by over-explaining simple ideas.

It is important to remember that the language used may not be the first language of the reader of the document.

Documentation developers should avoid “e.g.,” and “i.e.,” because some users might not understand the difference and the difference might be vital. Instead, spell out “for example” and “that is” in full, so that it is clear whether what follows are examples only or a complete set;

Other Latin terms such as *viz.*, *sic.*, *cf*, *qv*, *de facto*, and *ad hoc* should be avoided.

The word “please” should not be used except for optional, courteous requests. For example, "Please return the user satisfaction survey card," but not, "Please press Enter to save the data."

Humor should be used with caution. What is amusing when read once will be annoying in frequently referenced documentation. While it may be helpful in some introductory documents to use cartoons to illustrate points, the user should be allowed to skip them. Humor does not necessarily translate into different cultures, so if documentation is to be used in other cultures, check its suitability in all the target cultures.

Paragraphs should be short.

Section headings should be short and clear to motivate the user to read the text.

Tabular data should be presented in as simplified a form as possible.

B.2 Style for sentences

Sentence structures should be simple. The documentation developer should avoid complicated grammatical constructions, because users may find them difficult to understand. Lengthy sentences should also be avoided, as they make translation more difficult and therefore more expensive.

The emphasis in a sentence should be on the important point. Generally, the sentence should start with the information that the user already has and should lead the user on to new information.

B.2.1 Hanging participles

The documentation developer should take care to avoid hanging participles because they may be misleading. For example, in the following, the participle “saving'” is hanging, meaning that it does not logically modify anything in the sentence.

When saving the file, details of the date and time are recorded.

In this case, it is not clear who or what is saving the file. Instead, one of the following could be used:

When you save a file, you should keep details of the date and time.

When the system saves a file, it also saves details of the date and time.

In some cases, a hanging participle may cause serious misunderstanding, because the participle appears to modify a nearby noun. For example, in the following, it appears as if the administrator is the person who is noting the details, although the intention is that the user should note them:

After recording details of the transaction, the administrator ….

Instead, the following should be used:

After you record details of the transaction, the administrator ….

B.2.2 Tautologies and redundant phrases

Tautologies and redundant phrases should be avoided because they may make sentences more complicated than necessary even if they do not lead to misunderstandings. For example, in the following tautology the prefix “re” and the word “again” say the same thing:

Re-enter the information again.

It is unclear whether the information should be entered two or three times.

The following could be used instead:

Enter the information again.

Avoid omitting "that." For example, instead of "Ensure the disk is loaded …," write, "Ensure that the disk is loaded …."

B.2.3 Articles and pronouns

Articles and pronouns should not be omitted because this makes sentences difficult to understand. For example, instead of:

Use arrow keys to move pointer to start of address.

the following should be used:

Use the arrow keys to move the pointer to the start of the address.

B.2.4 Positive and negative constructions

Where possible, the documentation developer should use positive rather than negative constructions, unless a negative is used for emphasis. There should be no more than one negative in a sentence.

For example, instead of:

You will not see the results of the search unless you click on the Display button

the following should be used:

To see the results of the search, click on the Display button.

B.2.5 Style for conditions

The documentation developer should explain conditions clearly with the condition first, separated from the rest of the sentence by a comma. If the comma is missing, the meaning of the sentence might be unclear or, at worst, wrong. For example, the following simple conditional sentences have the conditions at the beginning and a comma separating the condition from the rest of the sentence, and both are clear:

If the screen goes blank when you press the Return key, the system saves your data.

If the screen goes blank, when you press the Return key the system saves your data.

If the comma were missing, the meaning would be unclear.

Where two conditions are needed, the separate conditions should be stated clearly and should be linked correctly using “and” or “or.” The text should emphasize the multiple conditions using terms such as “both,” “at least one of,” “either one of,” and “only one of.”

For complicated conditions, documentation developers should use a list or a table (see *Clause 12.18.8*), rather than giving the conditions in words. For example, the information in Table B.1 is much easier for the user to understand when presented in a table than it would be in continuous text.

Table B.1 — Example of conditions presented as a table

X	Y	Result angle A
Positive	Positive	0 <= A < π/2
Negative	Negative	π/2 <= A < π
Negative	Negative	- π <= A < - π/2
Positive	Negative	- π/2 <= A < 0

Diagrams such as flowcharts may be used to illustrate complicated conditions, but the documentation developer should assess whether the target users will be able to understand them.

B.2.6 Active and passive voice

Where possible, documentation developers should use the active voice. For example, the following sentence:

When new values are entered, the file has to be saved.

would be clearer if written as:

When you enter new values, you need to save the file.

However, there are situations where the passive voice is appropriate, and should be used. For example:

If someone has changed the background color to red, you will not be able to see the warnings.

would be clearer if written as:

If the background is changed to red, the warnings will not be visible.

B.2.7 Tenses

Pay careful attention to the use of tenses, taking into account that:

- users might be following instructions one by one, so actions are in the present;
- references to actions performed earlier are in the past;
- if something that the user is doing now might affect something in the system or something that the user might want to do later, then the future tense is essential to differentiate between the current activity and that later activity.

The following examples illustrate these points.

Past: When the software was installed, the system administrator assigned you to a user group.

Present: If you need to change to another user group, select the Change User Group function.

Future: The system will use the name of the new user group in the titles of future reports.

B.2.8 Singular and plural verbs

A verb must agree in number (singular or plural) with its subject. Particular care is needed where the subject and verb are separated by a word that uses a different number. For example, in the following sentence, the subject, which is “the first (window)”, and the verb “is” are both singular:

When the first of the windows is closed ….

The documentation developer should avoid using plurals unnecessarily because they may introduce ambiguity. For example, in the sentence “Power lights on PCs are usually green,” the user does not know whether there is more than one power light on a PC.

B.2.9 Punctuation

Simple, accurate punctuation makes written English easier to understand:

- Punctuation should be used consistently, with the sole aim of making the meaning of the text clear.
- Long sentences should be broken down into shorter ones where possible. If the resulting separate sentences need to remain linked, they should be separated by semicolons and not by commas.
- Parentheses should not be used to enclose essential information because this might reduce the apparent importance of that information.
- Square brackets or braces should be avoided in text although they may be necessary in mathematical and syntactic information. In the latter case they should be used strictly according to the rules of the relevant discipline. Square brackets and braces may also be used as ways of representing command variables and highlighting interactive text, see *Clause 12.11*.
- All single or all double quotation marks should be used to enclose quotations. Otherwise, neither single nor double quotation marks should be used unless the software uses them. In particular, quotation marks should not be used to enclose text that a user has to type because it would be unclear to the user whether or not the quotation marks themselves should be typed. (For guidelines on how to represent text the user has to type, see *Clause 12.11.1*).
- The documentation developer should be aware of the differences between UK and US English punctuation, especially where quoted speech occurs at the end of a sentence, and the use of the serial comma.
- Shorthand methods for covering several options in one construction are difficult to understand and may cause confusion.
- The documentation developer should avoid “and/or” and the use of a solidus (/ character) in other phrases to cover several options in one construction. For example, instead of:

  You may now save the file and/or exit from the system.

  give the information in full:

  You may now:

  - Save the file if you wish
  - Exit from the system

  To save the file ….

  To exit from the system ….

- The documentation developer should also not attempt to cover both the singular and plural options in a single sentence using brackets or parentheses to enclose the alternatives. For example, instead of:

  Click on the name(s) of the image(s) that you want to display.

  use:

  Click on the name of each image that you want to display.

  It is also possible to couch everything in the plural (but see *Clause 2.8*):

  Click on the names of the images that you want to display.

B.3 Style for paragraphs

Each paragraph should cover only one idea. Paragraphs should be kept short wherever possible. If long paragraphs are needed because ideas are complicated, consider breaking them up using lists or tables (see *Clauses 12.10 and 12.18.7*).

References to other parts of the same topic should be kept to a minimum.

Within each paragraph, the flow from one sentence to another should be clear, showing how each sentence is related to the previous one, so that users may follow the explanation.

EXAMPLE: If you want to save your work, use the USB memory in the computer's drive. This precaution will ensure that you obtain a back-up copy to be stored in another location.

It is also helpful to use similar techniques to link related paragraphs.

B.3.1 Hyphenation

Hyphenation is used to join two words together so that they act as a single word.

Hyphens should be used wherever they are helpful, even if in some cases they are not strictly necessary.

In some cases, hyphens are essential, because if they are missing, there is a danger of misunderstanding. For example, the phrase “Forty eight character fields are used” could mean either of the following, which are clearly different:

Forty eight-character fields are used.

Forty-eight character fields are used.

As another example, the following are clearly different and the difference is important:

a new orders file

a new-orders file

B.3.2 Infinitives

The two parts of an infinitive, that is, the word “to” and the verb, should normally be kept together. Although infinitives may usually be split by a single word without causing the user problems, split infinitives may annoy some users, and for that reason they should be avoided. Infinitives should not be split by a whole sequence of words because the text might be difficult to understand and translate.

B.3.3 Capital letters and lower case

Both the user interface and the documentation should observe the following recommendations for use of capital letters:

- Use of capital letters should be kept to a minimum.
- In names of software products and companies, capital letters should be used exactly as in the registered software product and company names.
- For titles and headings, including headings of columns in tables and lists, either of the following conventions should be used consistently:
    - an initial capital letter for only the first word;

- initial capital letters for significant words.

- For all index entries (headings) apart from proper nouns and acronyms, use lower-case. Index-generating software usually creates separate index entries for upper- and lower-case versions of the same word, so using lower-case reduces editing effort.

B.3.4 Anthropomorphisms

Human traits should not be attributed to the computer or application, particularly in on-screen documentation, because this could introduce the feeling that the system and not the user is in control. It may also give the user a false expectation of the capabilities of the application. Where the user has discretion over whether or how actions are carried out, sentences should enhance the user's perception that they, not the system, are in control. Where the system initiates and controls interaction with the user through a predetermined sequence of steps, sentences should make this clear.

For example, in the following, the user may decide what action to take:

If you want a printed copy of the report, use the Print Report command.

In the following, the system requires the user to take some action:

You must confirm the order using the Confirm Order command before opening another order.

B.3.5 Analogies and metaphors

The use of an analogy or a metaphor may sometimes help a user to understand complex ideas. However, if analogies or metaphors are used in the text, take great care to ensure that they do not lead users to make incorrect inferences.

Text should not explain analogies or metaphors instead of explaining the information; either both should be explained or only the information.

If metaphors are used, they should be used consistently. If the user interface uses metaphors, the documentation should not use different metaphors for the same concepts; it should use either the same ones or none.

Metaphors might not translate or localize correctly, so if the product will be translated or localized, metaphors should be checked to ensure that they are suitable, or they should be avoided.

B.4 Style for quick-reference information

Quick-reference information may use words and short phrases rather than complete sentences, as long as the meaning remains clear.

Each set of quick-reference information should be clearly labeled with a relevant but short heading.

Only good, tested examples should be used because users often just glance at the document to find the information that they need. If wrong or bad examples are used, even simply as illustrations of what to avoid, there is a danger that users may copy them. Therefore, caution is needed if these are used. The documentation developer should ensure that they are clearly labeled as wrong or invalid.

B.5 Style for installation instructions

The task of installing the software is usually done only once, so the user is normally installing the software for the first time. Keep the instructions simple, clear, and complete.

Terms with which users might not be familiar should be avoided. In particular, although terms might be explained in other user documents, the documentation developer should not assume that the user installing the software has read those other documents.

B.6 Style for tutorials and task instructions

Cautions and warnings should be stated in the imperative form, clearly indicating the action required, with a brief description of the hazard, if necessary. See *Clause 11.11* and *Clause 12.15*.

Where possible, items should start with a verb, instructing the user to “press,” “type,” or “select.” If conditions are associated with an action, they should be stated at the beginning of the sentence, separated from the rest of the sentence by a comma.

EXAMPLE: If the green indicator light illuminates, remove the diskette.

Sentences should present actions and effects in the sequence in which they should take place. In some cases, if information is not presented in the correct sequence, following the instructions may cause serious problems. For example, in the sentence:

> Select Exit to exit from the program, remembering to save the file first.

there is a risk that a user may select Exit before reading the remainder of the sentence, and hence lose important data. The same information would be better expressed as follows:

1) Save the file.

2) Select Exit to exit from the program.

B.7 Style for describing user interface elements

Users may need to find out what the elements of the user interface do. The documentation should provide the information as a single clause using an active construction and starting with a verb; for example, “commits the order to the master orders file ready for picking” or “displays a new empty deposit form.”

B.8 Style for descriptions and explanations

The documentation developer should focus on the information that the user needs, rather than on the information that is available about the subject.

The documentation developer should take the user from what they know to what they want to find out. This principle applies at the sentence as well as the topic level. To understand this, imagine the type of question that will have given rise to the user’s search for information. For example:

- If a user might ask “what does this mean?” or “what is this?,” a description is needed, and the sequence should be: “*x* means *y*” (not “*y* is what *x* means”).

- If a user might ask “what do I do to get a particular effect?” or “how do I do this?,” the fact should be written as an instruction, such as “to achieve *y*, use *x*” (not “*x* is the means by which you achieve *y*”).

It takes users longer to read sentences that refer to information introduced several sentences earlier than those that refer to recently introduced information; therefore, related ideas should be kept together.

If a description or explanation includes information about actions taken by different users or by the system, it should be clear who or what performs each action, particularly which action the user has to perform and which the system does.

B.9 Style for on-screen information

When writing on-screen information, the documentation developer should consider the size of the window's information area. Because the space is constrained, it is important that the writing is succinct. The documentation developer should:

- allow users to choose whether or not to see definitions of terms, rather than including them every time;
- show users an outline and allow them to select the sections of the outline that they want to look at.

B.10 Style for lists

All the entries in a list should have the same construction. For example, the instructions in a list should use the imperative.

A lead-in sentence may introduce the list. If each list item starts with the same word, that word should be part of the lead-in sentence.

For guidelines on the presentation of lists, see *Clause 12.10*.

Annex C
(informative)

User documentation style for translation and localization

C.1 General

The guidelines in this annex should be followed when writing material in English that is likely to be translated. Most points refer to both paper and on-screen documentation.

C.2 Terminology

Terminology should be as follows:

- General or non-technical terms, as defined in general dictionaries, should be used.
- Glossaries should be created that include:
    - definitions of software product-specific and unfamiliar terms;
    - expanded forms and definitions of acronyms and abbreviations;
    - explanations of unusual word usage, such as nouns used as adverbs.
- A controlled-language glossary of words and terminology specific to a project may be helpful for a particular project.
- Special terminology should be based on national or international terminology standards, recognized dictionaries, or approved glossaries.
- Each acronym should be defined on the first occasion it is used in each topic, or a link should be provided to the glossary.
- A bibliography of specialized dictionaries and international standards may be helpful for a particular project.
- Each term should be used consistently throughout the document, the on-screen information, and the system library.
- Compound phrases such as “file input” should have only one meaning, which should be used consistently.
- Compound phrases should be limited to three words.
- The same word should not be used as different parts of speech (for example, “aid” as both noun and verb).
- All software product-specific and specialized terms should be introduced within an explanatory or self-sufficient context.
- Terms introduced without sufficient context, such as keyboard key names and commands, should be defined in the glossary.

- The term “billion” should be avoided.

  NOTE: The US billion (= 1,000,000,000) is not the same as the old UK billion (= 1,000,000,000,000).

- The use of the term “translation” to refer, for example, to file format translation should be avoided; for a meaning other than translation from one language to another, “conversion” should be used instead. If this is not possible, define the term.

C.3 Style for translation

C.3.1 Abbreviations and symbols

Only widely recognized abbreviations should be used and they should be explained in an abbreviation list. The following should be avoided:

- the US symbol for pound (#);
- the raised period for multiplication.

C.3.2 Confusing words

Documentation developers should beware of the following confusing words:

- who, that, which;
- only, merely, just, mainly, simply;
- while;
- so;
- as;
- can, may;
- since;
- when, if;
- alternate, alternative;
- due to;
- once (instead of "when" or "after").

C.3.3 Syntax

Documentation developers should consider the following syntax items:

- Sentences should be short.
- Construction of sentences that contain a series of concepts separated by commas should be avoided.
- Restrictive and nonrestrictive clauses should be carefully distinguished.

C.3.4 Punctuation

A dash should not be used where a bracket or a semi-colon may be used.

C.3.5 Physical factors

Documentation developers should note the following physical factors:

- Abbreviations should not be used to save space.
- Articles and prepositions should not be omitted from sentences in an attempt to save space.
- Sufficient space should be left, for example, for monetary values.
- Text should not be integrated into illustrations.
- Only graphical symbols that are universally recognized should be used.
- Graphics should be used to replace text wherever possible.

C.3.6 On-screen information

Documentation developers should note the following suggestions for on-screen information:

- If control over the software development is available, on-screen information (text and messages) should be isolated from program logic.
- Each text block or message should have a unique identification code.
- The software should accommodate the maximum anticipated length of input and output fields in the expected countries and languages.
- A separate message should be used for each idea.
- Message variables should contain only untranslatable information such as keywords and return codes.
- The software should accommodate the maximum anticipated length of the text used in menu names and field names in the expected languages.

C.4 Cultural factors

Documentation developers should note the following cultural factors:

- Artwork (such as faces, animals, and telephones) should be culturally neutral.
- Examples that are specific to local culture or the local way of doing business (such as holidays, banking, payroll, sports) should be avoided.
- Idiomatic expressions specific to the documentation developer's national language should be avoided in the text and artwork.
- Humor, especially puns and plays on words, should be avoided.
- Irony should be avoided.
- Slang, jargon, and colloquialisms should be avoided.

- The first person, singular or plural, should not be used.

- Dates should not be expressed in all-numeric form. The month should always be spelled out (6 July 2010); using four digits for the year will clarify the date.

- International conventions should be used for measurement systems. Dual measurement systems may be used if traditional systems are prevalent in the country of use. When metric measurements occur with other measurements, the context should make the meaning clear.

Annex D
(informative)

Design, development, and production of printed information

D.1 Introduction

This annex gives guidance on the additional activities and decisions involved when a version of the documentation is to be provided in printed form.

D.2 Design

D.2.1 Decide how the printed documents will be produced

The documentation designer should consult several printing organizations for advice, prices, and time scales for different methods of production. Get samples for every solution that is being considered.

NOTE: The quality of reproduction depends on the printer resolution. For example, 300 dpi produces legible copies for text and line drawings; at least 600 dpi is better for reproduction of photographs and shading.

If a third party will produce the documents, the documentation designer should allow extra time for transferring documents.

Table D.1 shows some methods of producing multiple copies, giving the main features of each.

Table D.1 — Methods of producing multiple copies

Method	Quality	Speed and convenience	Cost
Printing on local printer	Depends on the quality of the printing device used Often may result in unbound, one-sided copies, which are less usable	Very convenient Useful where only a few pages or copies are needed Not always fast Very easy to make last minute changes, by simply replacing pages	Depends on the type of printing device and labor costs
Bulk printing on fast laser printer from PDF file	Excellent	Very fast	Low
Photocopying	Can give excellent results. Can handle a wide variety of types of paper	Can be very fast Very easy to make late changes by replacing pages	Can be inexpensive, especially for black and white only and for small quantities
Offset lithography	Results are excellent A wide range of effects can be achieved Often preferred for color printing	Needs a specialist printing organization Can take longer	Can be expensive for small quantities

D.2.2 Numbering schemes for pages, sections, illustrations, and tables

D.2.2.1 Page numbers

For each document, the documentation designer should select an appropriate method of numbering pages. For example:

- All the pages of a document may be numbered sequentially with an Arabic page number, starting from 1. For a reference to a page number in the index or contents pages, it will be apparent approximately how far through the book the page will be. This scheme is recommended for use wherever it is practicable.

- Pages may be numbered within each major section, with the number, letter or name identifying the section included with the page number. This style is known as "folio-by-chapter": it has the advantage of clearly showing which chapter the reader is in. Another advantage of this style of page numbering is the reduced impact of significant text edits on pagination. This two-part numbering is recommended for loose-leaf publications. The two parts of the page number should be separated by a hyphen or dash, for example:

  1-1, 1-2, 1-3, … 2-1, 2-2, 2-3

  Introduction-1, Introduction-2, Tutorial-1, Reference-1, Reference-2

  Number appendices or annexes in the format:

  A-1, A-2, …, B-1

- If the document is divided into physically separate volumes, the volume number should be included with the page number, for example:

  Volume 1, 1-1, Volume 1, 1-2.

- The preliminary pages of a document (front matter), that is, the pages containing the preface and the contents list, should be numbered in a separate sequence from the rest of the document (unless house style dictates otherwise). For example, using Roman numerals:

  i, ii, iii, iv.

- All pages should be counted in the numbering sequence, even if a number is not printed on the page. For example, if a document has a title page at the beginning (separate from the front cover), that sheet might not have a page number printed on either side of it, but nevertheless counts as pages i and ii.

- Pages may be numbered in the form "x of y", where "x" is the current page number and "y" is the total number of pages in the document. This is known as "maxpage" page numbering. The benefits in using this method are:

  - The reader soon knows whether pages have been missed off the end.

  - It may make the document template design easier because the same page numbering method is used for both the preliminary pages and the body.

D.2.2.2 Section numbers

The documentation designer should ensure that there is a clear distinction between the page numbers and the section numbers used.

The documentation designer should create a strict hierarchy of section numbers, where all the identifying letters or numbers of all the levels of the hierarchy are given every time a section number is quoted.

The recommended scheme for section numbers is:

- Arabic numbers;

- main sections numbered sequentially starting from 1;

- each main section divided into appropriate subsections (second level) and numbered sequentially within the highest level, for example, 1.1, 1.2. This method of subdivision is continued to further levels, for example, 1.1.1, 1.1.2. Whenever possible, restrict the hierarchy to no more than three levels of subdivision;

- numbers designating subdivisions of different levels separated with a full stop (period), with no full stop (period) after the final number.

D.2.2.3 Figure and table numbers

The documentation designer should number figures and tables unless they are all in-line in the text and are not referenced from elsewhere.

When numbering is used:

- Separate numbering sequences should be used for figures and for tables.

- If the pages of the document are numbered within sections, the figures and tables within the same level of section should be numbered separately. If the pages of the document are numbered straight through, figures and tables should be numbered straight through the document. ~

- The type of item should be included in the titles of illustrations and tables, for example, Figure 6 or Table 10. Titles should be used consistently throughout the documents.

The following are examples of figure and table numbering schemes:

- Table 1, Table 2, …
- Figure 1, Figure 2, …
- Illustration 1-1, Illustration 1-2, …
- Table 1.1, Table 1.2, ….

D.2.3 Page layouts

D.2.3.1 Page size and orientation

For on-screen documents that users may subsequently decide to print, the documentation designer should take into account the paper sizes likely to be available to users in the different countries where the software product will be used.

For pre-printed documents, the documentation designer should work with the printing organization to choose a page size and orientation that will be convenient in the users' environment; the documentation designer should take into account the amount of flat surface that is likely to be available for opening out documents. First, approximate the page size based on the users' needs; then determine the actual page size by the following steps:

- A suitable A size that is similar in size to the initial approximation should be selected. ISO 216:2007, *Writing paper and certain classes of printed matter — Trimmed sizes — A and B series, and indication of machine direction*, gives details of the international A sizes of paper. For example, A4, A5, or A6 might be suitable for different types of user manuals or booklets; larger A sizes might be suitable for wall charts.

- If no A size is suitable but a suitable size may be created using a simple fraction of an A size, the documentation designer should use that size. For example, one-third A4 might be suitable for a reference card, two-thirds A4 in either orientation might be suitable for a user manual, four-thirds A4 might be suitable for a folding reference card.

- If a completely different size is needed, the documentation designer should define it but check details of the production method and cost with the printing organization that will produce the documents.

  NOTE: The documentation designer should be aware of paper sizes in non-metric markets. For example, Mexico uses Letter (8.5" x 11") and Legal (8.5" x 14"). Documents written in the UK for A4, may at some stage have to be re-formatted to US Letter size before they are delivered. Consider this also if softcopy versions are sent with the intention that the customer has the responsibility of printing them.

- If the analysis of user needs determines that some users may need certain documentation in more than one size, for example one to use in the office and one to use when traveling, the documentation designer should decide whether or not the same document may be used to meet the different needs. If so, the document should be designed so that the text will be legible in the different sizes. If not, different versions of the documentation should be designed to satisfy the different needs.

The documentation designer should orient pages with the long edges either on the left and right (portrait) or at the top and bottom (landscape).

Portrait documents bound along the long edge are the most usual; they should be used for the majority of user documents. Landscape documents bound along the long edge have the same shape and size when opened, but the method of turning pages is less conventional; they should be used only if there is a special user need for a wider page. Documents bound along the short edge are less common. Landscape documents bound along the short edge could be used when a large amount of material must be visible at one time and there is

space in the users' expected work environment. Portrait documents bound along the top are sometimes used for stand-up documents used as job aids.

D.2.3.2 Basic printed page layout

The designer should adopt simple page layouts that may readily be achieved using the chosen method for preparing master pages.

The designer should prepare a page grid for each type of page (Figures D.1 and D.2). The typical different types of pages are:

- normal text pages;
- imprint page;
- title page;
- contents list;
- start-of-section pages;
- index.

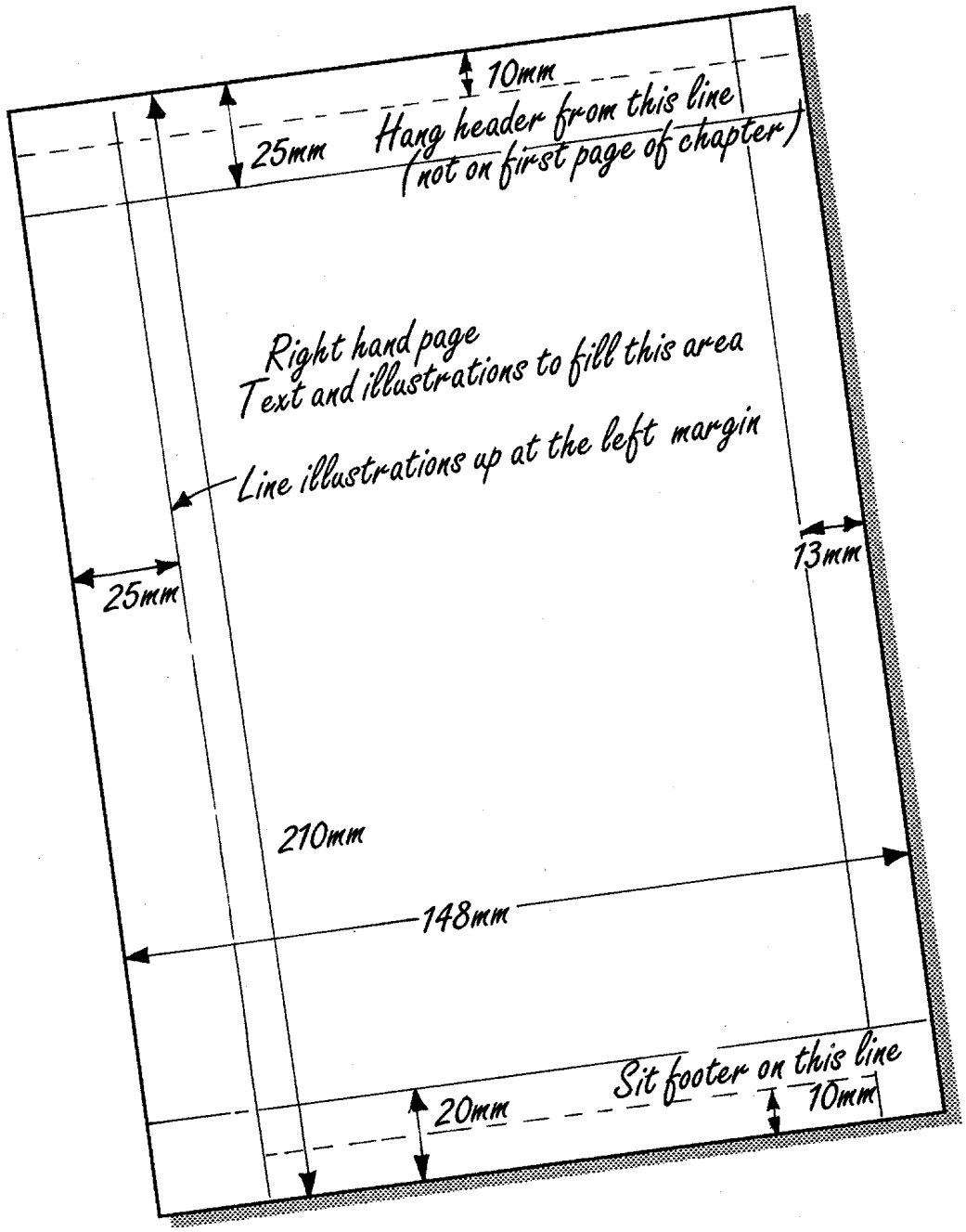
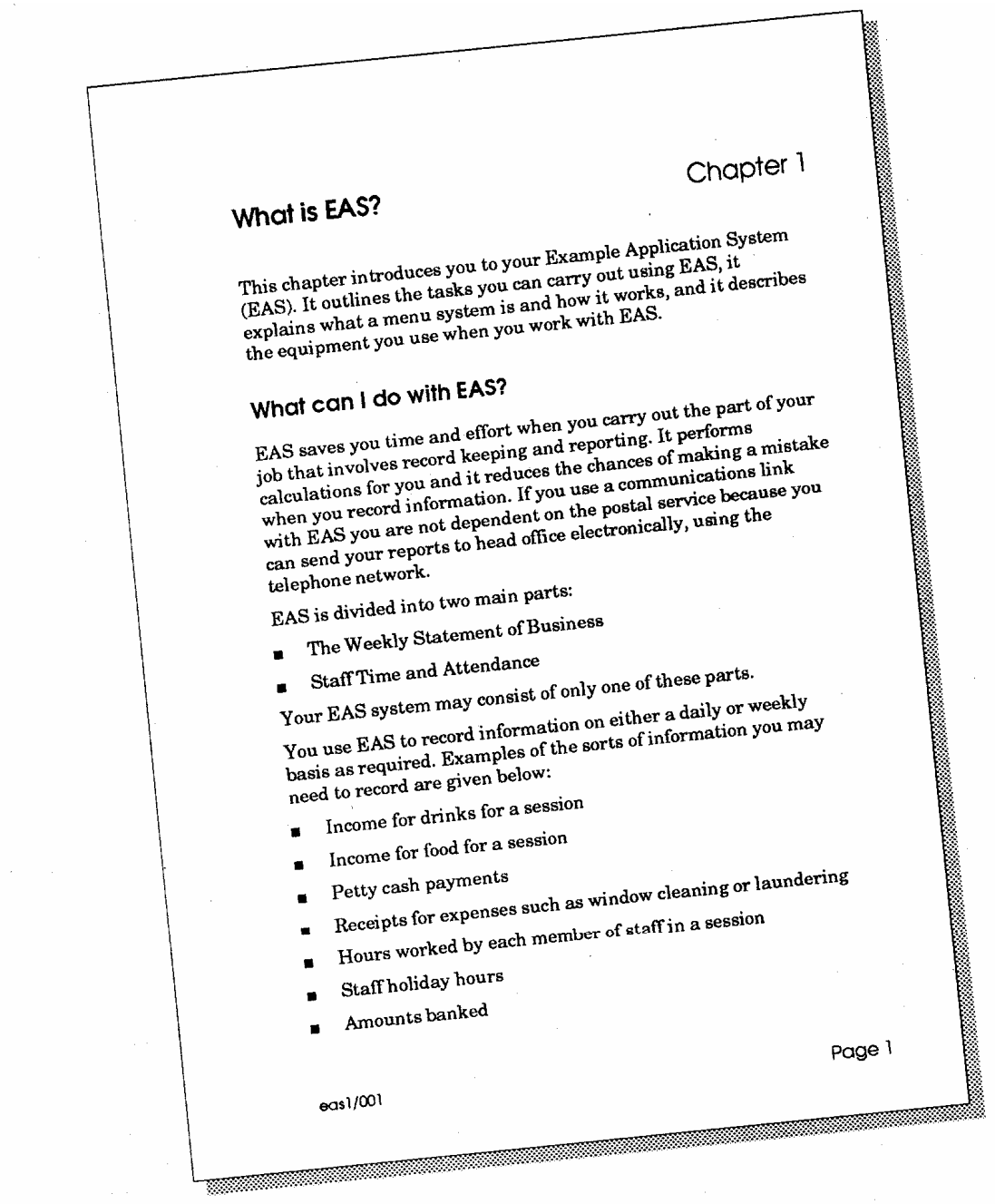


Figure D.1 — Example of an A5 page grid



Chapter 1

What is EAS?

This chapter introduces you to your Example Application System (EAS). It outlines the tasks you can carry out using EAS, it explains what a menu system is and how it works, and it describes the equipment you use when you work with EAS.

What can I do with EAS?

EAS saves you time and effort when you carry out the part of your job that involves record keeping and reporting. It performs calculations for you and it reduces the chances of making a mistake when you record information. If you use a communications link with EAS you are not dependent on the postal service because you can send your reports to head office electronically, using the telephone network.

EAS is divided into two main parts:

- The Weekly Statement of Business
- Staff Time and Attendance

Your EAS system may consist of only one of these parts.

You use EAS to record information on either a daily or weekly basis as required. Examples of the sorts of information you may need to record are given below:

- Income for drinks for a session
- Income for food for a session
- Petty cash payments
- Receipts for expenses such as window cleaning or laundering
- Hours worked by each member of staff in a session
- Staff holiday hours
- Amounts banked

Page 1

eas1/001

Figure D.2 — Example of an A5 page

When preparing the page grids, the documentation designer should consider both left-hand and right-hand pages side-by-side, to show how the document will look when it is open with two pages on view.

- The grid should show where the margins of the pages will be.

  NOTE: The documentation designer should be aware of the different UK and US spacing for drilling.

- The margin on the inside edge of a page should be wide enough to accommodate the method of binding. In the simplest case, the left and right margins of pages may be the same. However, the documentation

designer should consider using a wider margin on the inside edges of pages, that is, on the left margin of a right-hand page and the right margin of a left-hand page. Figure D.1 shows a page grid for an A5 page in a booklet to be bound by staples (saddle-stitched). Again, the inside margin is wider than the outside margin. Figure D.2 shows an example of an A5 page set using the page grid in Figure D.1.

- The grid should show the number of columns to be used for the text. For normal text pages, a single column for text is usually appropriate, although certain documents such as leaflets or brochures might benefit from a two-column or a three-column layout. The documentation designer should determine the width of the text column in association with the typography of the text.

- The documentation designer should consider the amount of text on the page as a proportion of the amount of white space, and the implications for the number of pages that will be needed. If the lines of text seem too long, the documentation designer should indent the body of the text from the margin.

- For the index, the grid should use two or three columns of text to reduce the number of pages used.

- The grid should show the placement of figures and tables. The documentation designer should make the placement strategy as simple as possible such that pages will always look neat and consistent, no matter how many illustrations and tables are used. For some documents, it might be helpful to users if the two pages of a double-page spread were visible together, perhaps with an illustration of a screen display or a report on one page and the corresponding text on the facing page.

- The documentation designer should decide how cards and charts will be folded, taking into account:

    - how users will want to handle the document itself and use the information it contains;

    - whether the document will have holes drilled, and if so, where;

    - what the sizes of other documents with which it will be packaged are.

D.2.3.3 Basic page elements

On the page grid, the documentation designer should define the precise positions of the following basic elements that will appear on pages:

- page numbers;

- document reference numbers;

- information needed at the top and bottom of every page to help the user (header and footer, respectively).

Horizontal lines may be used to separate header and footer text from the body of the page.

NOTE: The documentation designer should avoid the use of vertical lines unless there is a special need for them.

The documentation designer should avoid footnotes if possible, as readers are apt to lose their place. Information should be placed in the text where it is accessible. A smaller type size or different font may be used to indicate notes in text.

D.2.4 Presentation of illustrations

If illustrations are too wide to fit on a page (even after the illustration is reduced):

- The documentation designer may use a fold-out page if users need to refer to the illustration from different parts of the text, for example, a keyboard layout chart.

- The documentation designer may rotate the illustration 90 degrees to the left.

The documentation designer should place each illustration after the text that refers to it, preferably visible at the same time; the documentation designer should plan double-page spreads.

If the user needs to see an illustration at the same time as the associated text and there is not room on the same page, the documentation designer should leave white space and move the illustration and text to a new double-page spread.

D.2.4.1 Illustrations of printed output

If illustrations of printed output will be included, the documentation designer should take into account the reason for including the output. For illustrations in user documentation, it is preferable to capture an electronic text file that results in printed output. However, the output may be available only in printed form or in plain text without formatting. The documentation designer should consider two commonly used methods for preparing illustrations of printed output.

- The documentation designer should use actual output as master artwork; it may be scanned into an electronic system and reproduced as an image.

- The documentation designer should recreate text that resembles the printed output in format.

Wherever possible, illustrations of printed output should be presented in portrait on the printed page. If the entire width of a wide text has to be illustrated and will not fit on a portrait page, the text may be rotated 90 degrees to the left (landscape), so that the top of the report is on the left-hand side of the page.

D.3 Production phase

D.3.1 Quality and durability of the finished documents

D.3.1.1 Weight of paper

The documentation designer should choose a weight of paper that:

- suits the document type and size;

- prints both a strong image and fine detail.

For large reference documents, the documentation designer should use thinner paper to keep down the weight and thickness of the completed document. For awareness documents, use thicker paper.

Paper weights of between 80 grams per square meter (gsm) and 130 gsm suit most user documentation needs.

For divider pages, heavy paper or thin card should be used, such as 300 gsm board.

For double-sided documents, opaque paper should be used so that print does not show through from one side to the other (bleed-through).

The documentation designer should seek advice from a printing organization or a paper supplier, and should test some samples using the chosen production method.

D.3.1.2 Paper surface

Paper with a matt finish should be used for user documents such as tutorial or reference documents. However, for brochures and leaflets, a gloss or semi-gloss finish will present a more suitable image to the user.

For documents that might be used in a damp or dirty environment, for example in a storeroom, the documentation designer should consider laminating the pages or using plastics. Similar consideration should be given to divider pages, particularly tabbed dividers. Matt laminates should be used wherever possible, to avoid glare.

D.3.1.3 Paper color

Unless there is some special user need, plain white paper is the best choice. White papers are available in a range of brightnesses. Consider the likely user environment and consult a printer or paper supplier for help in choosing a paper of suitable brightness.

D.3.1.4 Binding

To select the type of binding a document should have, the documentation designer should consider how the document will be used. For each document, ask questions such as:

- Does the document need to open flat?
- Does the document need to stay open?
- Is the weight of the document important; for example, does the document need to be carried around?
- How frequently is the document expected to be used?
- Is a large format needed?
- Does the document need a visible spine with the document title on it?
- Do replacement pages or sections need to be issued to users?

The documentation designer should consult the printing organization to determine what sort of binding is suitable for each document, based on the answers to the questions above.

Documents should be bound individually. For example, two documents should not be bound in the same ring binder. (Users perceive the contents of one ring binder to be a single document and will expect there to be a single contents list and a single index.) If several sets of information are presented to the same user in the same ring binder, collect the information into a single document – perhaps with separate parts – but with one contents list and one index.

Spine information should read across the spine. If the spine is too narrow, the information should read along the spine; local conventions may determine the orientation. The spine information should include the title, volume number, and version of the document (if greater than 1); the documentation designer should also consider including the name or logo of the company. Recommendations for the presentation of spine titles are given in ISO 6357:1985, *Documentation — Spine titles on books and other publications*.

D.3.1.5 Submit master pages for production

Before master pages are sent for production, the documentation designer should agree with the organization that will produce the documents on the following:

- number of copies of each document;
- exact page size for each document;
- type of paper to be used;
- details of lamination or other special finish needed for any of the pages;
- folding instructions for cards and charts;
- type of binding for each document and details of cover and spine information;
- instructions for use of color;
- instructions for collating documents; for example, where divider cards should be included.

When master pages are submitted for production, agree with the organization that is to produce the copies on the following:

- date, time, and place where copies will be delivered;
- date and time for delivery of samples to check;
- date and time for returning comments on the samples and approval of them;
- date and time for delivery of finished documents;
- cost.

The documentation designer should securely archive on-screen and paper copies of master pages of text and illustrations so that a new set of master pages may be created if the original set is lost or damaged.

The documentation designer should check complete copies of the documents.

D.3.1.6 Package the documents and software

The documentation designer should consider using open-fronted boxes (slipcases) to package several documents or single documents.

Slipcases are often used to hold ring-bound documents, but the documentation designer should consider them for various bindings if they are used to hold the full set of documents for one type of user. Cases may be made to order. They improve the appearance of the software product, increase the durability of the documentation, and give users a convenient storage method. However, they add to the packaging cost of the product.

If removable media will be packaged with the printed documentation, the documentation designer should consider:

- putting these items in plastic holders bound in a ring binder of their own;
- putting these items in plastic holders bound in the same ring binder as the installation instructions or some other user document;
- using specially molded plastic media holders in boxes of similar size to binders or books, held in slipcases along with the documents.

Before documents and software are packaged, the documentation designer should agree with the packaging organization on the following:

- details of how separate documents and the software will be packaged together;
- information to be included on the outside of the package, including artwork that is needed;
- details of the form of packaging that will be used;
- the number of packages required.

The documentation designer should consider producing packaging diagrams (sometimes known as floor plans) to describe how documents and software are to be packaged.

If packaging is to be done by a different organization from the one that produced the copies of individual documents, the documentation designer should arrange for the copies of the documents to be checked before they are packaged and for the copies to be transferred to the packaging organization.

If the packaging is being done by the same organization as the one that produced copies of documents, the documentation designer should check sample documents together with sample packages.

If the software is to be packaged with the documentation, the documentation designer should arrange for copies of the software to be delivered to the packaging organization.

The documentation designer should give packaging instructions to the packaging organization, including the following:

- delivery instructions;
- time scale for delivery of samples to check;
- time scale for returning comments on the samples and approval of them;
- time scale for delivery of finished packages;
- cost.

D.3.1.7 Check the finished documents and package

The documentation designer should check sample copies of individual documents and samples of the entire documentation product, not all the copies.

It is normal practice for the production and the packaging organizations to produce a few sample copies for checking. The documentation designer should correct mistakes in these samples before larger quantities are produced. The documentation designer should check random samples from these larger quantities.

The results of the production process and the packaging process should also be checked to verify that:

- All elements of the documentation are clean and free of marks.
- The covers are correct and the cover information is presented correctly.
- The quality of the print is suitable and is consistent throughout the samples.
- The quality of the illustrations is suitable; for example, drawn lines and text are clear.
- If colors are used, they are correct and consistent.
- The documents have been assembled correctly: the dividers are in the right places, the pages are in the right sequence, and so on.
- The documents have been bound and packaged correctly.

D.3.2 Accessibility

When providing printed documentation in response to a request for alternative formats (see *Clause 12.5*), the documentation developer should consider:

- printing in large type (12 point or above);
- printing on low reflectance or low contrast paper;
- binding the documentation so that it stays open unaided.

Annex E
(informative)

Checklists for user documentation

NOTE: Copyright is waived by ISO with respect to the checklists in Annex E.

The following checklists are excerpts from the checklists that are used to judge manuals and online help in international competitions of the Society for Technical Communication.

E.1 Checklist for printed manuals

1. Guidelines The following guidelines apply to all manuals, including:

- Software Guides
- Hardware/Software Combination Guides
- Computer Hardware Guides
- Non-computer Equipment Guides

Choose one of the following ratings for each criterion:

SD = Strongly Disagree D = Disagree N/A = Not Applicable
SA = Strongly Agree A = Agree

Content and Organization

Criteria	SD	D	A	SA	NA
Writing tone and style suit the purpose and audience					
Vocabulary and reading level are appropriate for the audience					
Organization and conventions are either inherently understandable or are explained					
Overall strategy for organizing the information suits the subject matter					
Similar kinds of information are presented consistently					
Writing is crisp and clean, with logical development of the subject matter at the right level of detail for user and task					
Writing is free of gender or ethnic bias					
Technical complexity is handled effectively					
Graphic elements are positioned near the text they support					
Notes, cautions, and warnings are clearly identified, positioned appropriately, and follow conventions for their meaning					
Terminology is defined in an effective place					
Quick-start procedures, tutorials, glossaries, appendixes, or reference sections are included when they would be helpful and appropriate					
Easy-to-find customer support information is included, as appropriate					

Copy Editing

Criteria	SD	D	A	SA	NA
Spelling, punctuation, grammar, and capitalization are correct and consistent					
Writing tone and style are consistent					
Treatment of wording in headings is consistent					
Copy is free of obvious technical errors					
Terminology is used consistently					
Treatment of all elements (lists, examples, tables, and so on) is consistent					
References to information within and outside of the manual are correct and consistent					
Labeling, captions, and callouts for tables, illustrations, photos, and other support material are consistent					
Acronyms and abbreviations are spelled out and defined at first occurrence					
Table of contents is comprehensive, useful, accurate, and well-edited					
Index is comprehensive, cross-referenced within, accurate, and well-edited, with effective use of synonyms; it also considers the reader's viewpoint of how to access information					

Visual Design

Criteria	SD	D	A	SA	NA
Overall design, including the cover, is unified and appropriate for the purpose					
Layout of page elements contributes to readability and usability					
Typography is used as an effective design element					
Typography is easy to read					
Headers and footers are visually effective in helping readers find information					
Other navigation devices are used, as appropriate					
Graphics maintain the internal consistency of the publication					
Icons and symbols (if used) are explained and used effectively					
Graphics are suitable for the audience in tone, style, and content					
Graphics support the content effectively					
Graphics are consistently well-designed and legible, and are executed neatly					
Tables, charts, and diagrams are treated as graphic elements					
Captions and callouts are effective for illustrations, tables, photos, and other graphics					
Color (if used) adds to the appeal and usability of the publication and unifies its design effectively					
Size and binding are appropriate for purpose and audience					
Production materials are of appropriate durability and quality					
Print quality supports the readability and usability of the publication					

Overall

Criteria	SD	D	A	SA	NA
Fulfills the purpose for the intended audience					
Integrates all elements into a readable and usable publication					
Shows evidence of creativity or originality					
Projects a professional image of the publication's sponsor					

E.2 Checklist for online help

Scale

5 Exemplifies the highest quality with few or no weaknesses

4 Strengths outweigh the weaknesses

3 Strengths and weaknesses balance

2 Weaknesses outweigh the strengths

1 Weak areas greatly affect the effectiveness

Does the entry fulfill its purpose? (mark the overall section rating with an **X** below the number)	**1**	**2**	**3**	**4**	**5**
Evaluation Factors			**Yes**	**No**	**N/A**
Audience Definition					
Is there a clearly defined audience to which the help is addressed?					
Does the entry effectively meet the needs of the intended audience?					
Goals or Purpose					
Are the goals or purpose of the entry stated clearly?					
Does the entry answer questions or provide appropriate assistance where needed?					
Are there multiple and effective methods for accessing help?					
Is the level of context sensitivity appropriate and does it lead to useful information?					
Given the goals as articulated in the help entry, did the documentation developer(s) make appropriate choices for:					
• Help design					
• Context sensitivity					
• Language					
• Linking					
• Interactivity					
Do the contents of the help adequately address the stated goals?					
Does the help system provide clear and sufficient instructions for how to complete tasks?					

Is the content of high quality and is it communicated effectively? (mark the overall section rating with an **X** below the number)	**1**	**2**	**3**	**4**	**5**
Evaluation Factors			**Yes**	**No**	**N/A**
Writing					
Is the help system well written?					
Is the writing style appropriate for the audience and the topics being covered?					
Is the language consistent throughout the help system?					
Is the language appropriate to the subject?					
Are procedures (if used) presented in clear sequential steps?					
Content Design					
Do the titles and headings clearly identify the information that follows?					
Are lists, tables, and graphics used effectively?					
Does the entry provide signposts to orient the user?					
Overall Quality					
Do all of the navigational elements, such as hyperlinks, resolve correctly and behave in the expected manner without error?					
Is the help navigation (hyperlinks) error free?					
Is the content consistent and appropriate for the audience?					
Is the interface consistent, easy to use, and reliable?					

Is the content well integrated and organized? (mark the overall section rating with an **X** below the number)	**1**	**2**	**3**	**4**	**5**
Evaluation Factors			**Yes**	**No**	**N/A**
Organization/Integration					
Is the help system well organized and is the organization appropriate for the audience?					
Is the organization obvious?					
Is the information organized into appropriate topics?					
Is the information organized into appropriate subtopics?					
Can you easily navigate between topics?					
Are there direct links or text references to relevant external documents, topics, or subtopics?					
Table of Contents (or navigational equivalent)					
Are the contents clearly identified in a table or navigational equivalent?					
Is the table of contents or navigational equivalent complete and comprehensive?					
Does the table of contents or other navigational equivalent provide an easy way to access the contents or move through branches of information?					
Index					
Are the index entries well chosen?					
Does the Index use cross-references and alternate words (synonyms) for topics?					
Can you browse the Index easily using an incremental search field, alphabet navigation buttons, or other facility?					
Search					
Does the help system have an effective search mechanism?					
If there is a full text search capability, is it easy to use? Does it support wildcards, case sensitivity, and word variation?					
Can you specify a search area?					
Navigation					
Is it easy to find specific information, to navigate through information, and to return to where you started?					
Are navigational aids present and are they used consistently throughout the entry?					
Context Sensitivity					
Does the entry employ context sensitive help effectively?					
Usability					
Is the interface intuitive, easily interpreted, and consistent?					
Is information provided to assist the user such as help, help-on-help, action cues, etc.?					

Are the communication media used effectively and appropriately? (mark the overall section rating with an **X** below the number)	**1**	**2**	**3**	**4**	**5**
Evaluation Factors			**Yes**	**No**	**N/A**
Platform Conventions/Features					
Does the entry appropriately use the standard conventions and features of the platform? For example, if the entry is a Windows Help file, does it effectively use the standard WinHelp features appropriately such as the Contents file, indexing features, design options, etc.?					
Speed					
Is the entry reasonably responsive?					
Is the entry designed to optimize performance?					
Interactivity					
Does the entry offer choices and can the user control the pace?					
Can the user omit or repeat parts?					
Can the user exit easily?					
Screen Design and Accessibility					
Is the design appealing and inviting?					
Are the screens easy to read?					
Is the type size easy to read? Can the type size be changed if required?					
Does the design minimize the need to scroll where possible?					
Does the help system provide appropriate accessibility features, such as tool tips, for users who require them?					
Graphics					
Are graphics used effectively to communicate concepts?					
Are the graphics attractive and of high quality?					
Media					
Are media elements (such as sound, video, animation, and interactive elements) easy to use?					
Are media elements (such as sound, video, animation, etc.) of high quality?					
Are media elements used appropriately? Do they enhance the communication objective of the entry?					
Do the media elements balance with each other and the content?					

Annex F
(informative)

Requirements clauses and checklist for the documentation process

For the convenience of users of this International Standard, this annex identifies those clauses which contain requirements (*shall* statements) for the documentation process. The following list identifies the location of the specific requirements for use when verifying conformance to this International Standard.

Clause no.	Guideline	Applicability		Conformance			
		Yes or No	Reason not applicable	Yes	Partial	No	Comments
2.2	**Conformance situations** … The relevant situation shall be identified in the claim of conformance … (list follows)						
6	**Project requirements, objectives and constraints** … shall gather or receive information about the wider context of the whole project …						
6.1	**Project objectives** … shall maintain information on the standards the software product is required to follow.						
6.4	**Users and usability objectives** … the usability requirements and the method of testing them shall be specified in the analysis phase …						
6.4	**Users and usability objectives** … When the usability targets are set and measured for the software product, the documentation shall be treated as an integral part of the software product.						

Clause no.	Guideline	Applicability		Conformance			
		Yes or No	Reason not applicable	Yes	Partial	No	Comments
6.4	**Users and usability objectives** … Measures of the usability of the documentation, independent of the usability of the software, shall include the following … (list follows)						
6.6	**Project planning …** The results of the documentation planning process shall be recorded in a documentation plan …						
6.6	**Project planning …** The documentation plan shall assign responsibility and authority for each major activity … (list follows)						
6.6	**Project planning …** The documentation plans … shall be under change control procedures with the rest of the project plans.						
6.6.2	**Version control and change control …** A configuration management (CM) process shall be used to control the user documentation.						
6.6.2	**Version control and change control …** Change control procedures for the project shall take account of the requirements of the documentation activities.						
6.6.4	**Schedules …** During the project implementation phase, the organization shall prepare a preliminary schedule for documentation activities.						

Clause no.	Guideline	Applicability		Conformance			
		Yes or No	Reason not applicable	Yes	Partial	No	Comments
6.6.4	**Schedules …** More detailed schedules for developing each separate document, as part of a master schedule for other parts of the project, shall be produced during the design and development phase.						
6.6.4	**Schedules …** Changes in the software product and in planned delivery dates shall be promptly communicated to those concerned, and the impact of these changes on the documentation shall be evaluated and communicated to project management.						
7	**Analysis and design …** Documentation shall be based on an audience and task analyses.						
7.1.1	**Audience analysis …** The designer shall list the intended types of users of the software product and classify users into audiences.						
7.2	**User documentation design …** The user documentation designer shall perform two principal activities … (list follows)						
7.2.2	**Designing formats …** The organization shall set conventions for the whole product, project, or organization for entire documents and document sets, topics, chapters, front matter, back matter, prefaces, technical terminology, icons and symbols, navigational controls, and system messages.						

Clause no.	Guideline	Applicability		Conformance			
		Yes or No	Reason not applicable	Yes	Partial	No	Comments
8.1.1	**CM during development** … The organization responsible for documentation development shall ensure that the requirements of the CM system for the project are enforced.						
8.1.1	**CM during development** … Approved documentation versions shall be maintained securely and separately from copies checked out for development.						
8.1.1	**CM during development** … Each draft issued for review shall be uniquely identified.						
8.1.2	**Development of translated and localized documentation …** the project glossary (list of terms and their definitions) shall be translated.						
8.1.2	**Development of translated and localized documentation …** The documentation shall be translated only after the translated list of terms has been approved.						
8.2	**Evaluation of documentation …** Documentation test requirements shall be specific and measurable.						
8.2	**Evaluation of documentation …** Documentation evaluation shall include the following four activities … (list follows)						

Clause no.	Guideline	Applicability		Conformance			
		Yes or No	Reason not applicable	Yes	Partial	No	Comments
8.2.1	**Evaluation of documentation quality …** Documentation evaluation shall be based on the required features and qualities.						
8.2.2	**Documentation review procedures … Safety …** The reviewers shall verify that warnings and cautions are appropriately placed and worded to be noticeable and understandable by the intended users.						
8.2.2	**Documentation review procedures …** Before production, the documentation shall be reviewed in its entirety.						
8.2.2	**Documentation review procedures …** Review and acceptance procedures shall specify who is the final authority for accepting and implementing changes.						
8.2.2	**Documentation review procedures …** Reviewers' comments shall be retained under configuration control …						
9.1	**Final assembly and review …** When the final draft has been approved, publishing/production staff, editorial staff, or a different person than the one who wrote the documents, shall … (list follows)						
9.2	**Approval …** Documentation shall be approved as specified in the documentation plan or contract before it is released.						

Clause no.	Guideline	Applicability		Conformance			
		Yes or No	Reason not applicable	Yes	Partial	No	Comments
9.3	**CM …** The CM process shall be exercised to maintain a controlled copy of the released documentation version.						
9.4	**Updating and maintenance …** Documentation content affected by the new release (including translations), especially instructional procedures, shall be updated and provided to customers who have contracted for the software product and documentation maintenance.						

Annex G
(informative)

Requirements clauses and checklist for documentation products

For the convenience of users of this International Standard, this annex identifies those clauses which contain requirements (*shall* statements) for documentation products. The following list identifies the location of the specific requirements for use when verifying conformance to this International Standard.

Clause no.	Guideline	Applicability		Conformance			
		Yes or No	Reason not applicable	Yes	Partial	No	Comments
10	**Structure of documentation …** When a document set will address audiences with widely differing needs, at least one of the following structures shall be used … (list follows)						
10	**Structure of documentation** … The audiences and their needs shall be identified specifically in the introduction, allowing each user to identify the sections of interest. (list item) …						
10.1	**Overall structure of documentation …** Documents shall be structured into units with unique content.						
10.1	**Overall structure of documentation …** Each page or screen shall be uniquely labeled (for example, with a page or topic number, or screen name or number).						
10.1	**Overall structure of documentation …** Printed documents shall be structured with no more than three levels of subdivision within a chapter.						

Clause no.	Guideline	Applicability		Conformance			
		Yes or No	Reason not applicable	Yes	Partial	No	Comments
10.1	**Overall structure of documentation …** On-screen documents shall be structured so that information may be accessed with no more than three jumps (that is, requiring no more than three links) from the initial page of a topic (not counting any action required to open the document).						
10.1	**Overall structure of documentation …** The organization of documentation shall support its usage mode (*instructional* or *reference*).						
10.1	**Overall structure of documentation …** When a document contains both instructional and reference material, the two shall be clearly separated into different chapters or topics, or distinguished by formatting within the chapter or topic.						
10.1.1	**Structure of instructional mode documentation …** Task-oriented instructional mode documentation shall include procedures structured according to the user's tasks.						
10.4	**User documentation components …** The required components shall be included in documentation unless the information does not exist or is not applicable for a specific document.						
10.5.1	**Initial components …** Each individual document shall be structured to begin with identification data (*Clause 11.3*), followed by a table of contents (see *Clause 12.13*) and an introduction; that is, the introduction is the first chapter or topic of the document.						

Clause no.	Guideline	Applicability		Conformance			
		Yes or No	Reason not applicable	Yes	Partial	No	Comments
10.5.1	**Initial components …** The introduction shall describe the intended audience, scope, and purpose for the document and include a brief overview of the software‘s purpose, functions, and operating environment.						
10.5.1	**Initial components …** Introductions shall be provided within a document for each chapter and topic.						
10.5.1	**Initial components …** The introductory sections shall provide an overview of the topic, the purpose of the function, and any environmental requirements, warnings, cautions, or user requirements unique to the topic.						
10.5.2	**Placement of critical information …** .General warnings or cautions that apply throughout the use of the software or documentation shall appear in the initial components.						
10.5.2	**Placement of critical information …** Specific warnings and cautions shall appear on the same page or screen and immediately before the procedure or step that requires care.						
11	**Information content of user documentation …** The information required in this clause shall be included in the documentation unless the information does not exist or is not applicable for a specific document.						
11.1	**Completeness of information …** Documentation shall provide complete instructional and reference information for all critical software functions …						

Clause no.	Guideline	Applicability		Conformance			
		Yes or No	Reason not applicable	Yes	Partial	No	Comments
11.1	**Completeness of information …** Instructional mode documentation shall include complete information to enable performance of selected tasks using the software functions by the least experienced members of the audience.						
11.1	**Completeness of information …** Reference mode documentation shall include all instances of the selected elements being documented.						
11.1	**Completeness of information …** if reference mode documentation covers a subset of software commands, it shall include the relevant user-entered and system-displayed commands and error messages in that subset.						
11.2	**Accuracy of information …** Documentation shall accurately reflect the functions and results of the applicable software version.						
11.2	**Accuracy of information …** If the previous documentation version is no longer accurate, current documentation shall be available for customers acquiring software updates or upgrades.						
11.3	**Content of identification data …** Documentation shall contain unique identification data. The identification data shall include the following … (list follows)						
11.3	**Content of identification data …** Identification data shall appear on a package label, legible without opening the package, and on a title page.						
11.3	**Content of identification data …** Each document in a document set shall have a unique title page.						

Clause no.	Guideline	Applicability		Conformance			
		Yes or No	Reason not applicable	Yes	Partial	No	Comments
11.3	**Content of identification data …** The identification of the document and the software shall be consistent with the CM practices of the issuing organization or the acquiring organization.						
11.3	**Content of identification data …** Information (change history) shall be provided in the document set to document the date of issue and version number of the current version …						
11.4	**Information for use of the documentation …** The documentation shall include information on how it is to be used (for example, help on help), and an explanation of the notation (a description of formats and conventions—see *Clauses 11.12 and 12.11*).						
11.4	**Information for use of the documentation …** At least one document in a document set shall identify all of the documents in the set by title and intended use, including recommendations on which members of the audience should consult which sections of the documentation.						
11.5	**Concept of operations …** Documentation shall relate each documented function to the overall process or tasks.						
11.6	**Information for general users of the software …** Task-oriented instructional mode documentation shall include instructions for routine activities that are applied to the general use of the software … (list follows)						

Clause no.	Guideline	Applicability		Conformance			
		Yes or No	Reason not applicable	Yes	Partial	No	Comments
11.7	**Information for procedures and tutorials …** Instructional mode documentation provides directions for performing procedures. Procedures shall include … (list follows)						
11.7.1	**Preliminary information for procedures …** Preliminary information common to several procedures may be grouped and presented once to avoid redundancy. Preliminary information for procedures shall include the following … (list follows)						
11.7.1	**Preliminary information for procedures …** Relevant warnings, cautions, and notes shall immediately precede each applicable instructional step or group of steps.						
11.7.2	**Procedural steps …** Instructional steps shall be numbered using Arabic numbers and presented in the order of performance.						
11.7.2	**Procedural steps …** Procedural steps shall indicate the expected result or system response.						
11.7.2	**Procedural steps …** Procedural steps shall include or provide references to documentation of the acceptable range, maximum length and applicable format, and unit of measurement of data fields for user-supplied data.						
11.7.2	**Procedural steps …** Procedural steps shall include or provide references to explanations of error messages and recovery procedures.						
11.7.3	**Completion information for procedures …** All procedures shall ensure that it is clear to the user that the procedure has been successfully completed.						

Clause no.	Guideline	Applicability		Conformance			
		Yes or No	Reason not applicable	Yes	Partial	No	Comments
11.8	**Information on software commands …** Observing the syntax of the system being documented, the documentation developer shall explain the formats and procedures for user-entered software commands, including required parameters, optional parameters, default options, order of commands, and syntax (see *Clause 12.11.1*).						
11.8	**Information on software commands …** Reference mode documentation shall contain a reference listing of all reserved words or commands.						
11.8	**Information on software commands …** Documentation shall explain how to interrupt and undo operation during execution of a command and how to restart it, if possible.						
11.8	**Information on software commands …** Documentation shall describe how to recognize that the command has successfully executed or abnormally terminated.						
11.10	**Content of error messages and problem resolution …** Reference mode documentation shall include each error message with an identification of the problem, probable cause, and corrective actions that the user should take.						
11.10	**Content of error messages and problem resolution …** The documentation on resolving problems shall also include contact information for reporting problems with software or its documentation, and suggesting improvements.						
11.11	**Content of warnings and cautions …** A warning or caution shall include the following parts: … (list follows)						

Clause no.	Guideline	Applicability		Conformance			
		Yes or No	Reason not applicable	Yes	Partial	No	Comments
11.12	**Information on terminology …** Documentation shall include a glossary of terms or their specific uses in the software user interface or documentation are likely to be unfamiliar to the novice users in the audience.						
11.12	**Information on terminology …** The glossary shall include an alphabetic list of terms and definitions.						
11.14	**User-supplied content …** If documentation is supplied in modifiable form, the supplier shall ensure that … (list follows)						
12.2	**Use of printed or on-screen formats …** Whether or not on-screen documentation is provided, the following documentation shall be presented in printed form … (list follows)						
12.2	**Use of printed or on-screen formats …** For software packaged and delivered in a box, this information shall be printed and included in the box.						
12.2	**Use of printed or on-screen formats …** When onscreen documentation is provided, it shall be available for display at any time when user input to the software is possible.						
12.3.2	**Relationship of information displays to the application's display …** If on-screen reference mode documentation is provided, it shall be accessible from the software it documents, and shall provide a clear means of exiting the documentation and returning to the software.						
12.5	**Accessible documentation …** Documentation shall be accessible and usable by the expected groups of users in their work environments.						

Clause no.	Guideline	Applicability		Conformance			
		Yes or No	Reason not applicable	Yes	Partial	No	Comments
12.5.2	**Provide user documentation in accessible electronic form** … All user documentation, both print and on-screen, shall be delivered in electronic form that meets applicable documentation accessibility standards.						
12.5.2	**Provide user documentation in accessible electronic form** … This documentation shall be provided with the product, or upon request on a timely basis and without extra cost.						
12.5.3	**Provide text alternatives in on-screen documentation** … Information presented in pictures and graphics by software shall also be provided as descriptive text suitable for screen reading, printing, or Braille conversion so that it can be read by an alternative method.						
12.5.5	**Provide documentation on accessibility features** … Print and on-screen documentation shall provide general information on the availability of accessibility features and information about the purpose of and how to use each feature.						
12.6	**Consistency of formats …** Documentation shall use consistent terminology throughout a document set for elements of the user interface, data elements, field names, functions, pages, topics, and processes.						
12.6	**Consistency of formats …** Formatting conventions shall be applied consistently throughout a documentation set.						
12.6	**Consistency of formats …** If the application uses icons or symbols, the on-screen documentation shall explain what they represent.						
12.6	**Consistency of formats …** The documentation shall not use those icons to represent any other objects.						

Clause no.	Guideline	Applicability		Conformance			
		Yes or No	Reason not applicable	Yes	Partial	No	Comments
12.6	**Consistency of formats …** Formatting conventions for highlighting information of special importance, such as warnings, cautions, and notes, shall be applied consistently throughout the document set.						
12.6	**Consistency of formats …** Similar material, such as sets of instructions, shall be presented in a consistent format.						
12.8	**Layout of screens and pages …** The documentation designer shall prepare a consistent layout for formatting similar information.						
12.9	**Legibility …** Printed and electronic documentation shall be legible to the user, taking into consideration the distance between the user and the documentation in the expected work environment.						
12.9	**Legibility …** Documentation shall use a font style and color that is legible against the expected background (paper color or screen background color).						
12.9	**Legibility …** Printed text, including text in illustrations, shall be no smaller than 2.75mm for capital letters measured at the screen or page.						
12.11	**Formats for representing user interface elements …** Graphical user interface (GUI) elements of the software, such as buttons, icons, and variable cursors and pointers; special uses of keyboard keys or combinations of keys; and system responses shall be represented in documentation by consistent graphic or typographical formats so that the various elements are each distinguished from the text.						

Clause no.	Guideline	Applicability		Conformance			
		Yes or No	Reason not applicable	Yes	Partial	No	Comments
12.11.1	**Representing control and command input …** Documentation formats for user-entered commands or codes shall clearly distinguish between literals (to be input exactly as shown) and variables (to be selected by the user).						
12.11.1	**Representing control and command input …** Formal notation, such as the use of brackets, braces, the greater than (>) and less than (<) characters, and other marks, shall be defined in every document that uses it, with the exception of […] for references in a bibliography.						
12.11.1	**Representing control and command input …** Quotation marks shall not be used in command representations unless the user should input them literally.						
12.11.2	**Representing special keyboard keys …** The documentation shall establish consistent conventions for representing special keyboard keys and explain the conventions in documentation for software that uses special keys for input.						
12.13	**Navigational features …** Documentation shall include explanations of unusual, flexible, or complicated system- and document-specific navigational features.						
12.13	**Navigational features …** Navigation features shall enable documentation users to go to the following locations … (list follows)						
12.13.2	**Finding the same navigation again …** Features for navigation shall be provided such that users may determine their location within the printed or on-screen documentation.						
12.13.2	**Finding the same navigation again …** In printed documentation, each page shall have a unique page number.						

Clause no.	Guideline	Applicability		Conformance			
		Yes or No	Reason not applicable	Yes	Partial	No	Comments
12.13.2	**Finding the same navigation again …** In on-screen documentation, each page or screen shall have a unique identifier (alphanumeric or caption) that can be seen by the user.						
12.13.5	**Linking information …** Links between related topics shall be bidirectional, so that whichever topic the users access first, they may jump to the related information on the other topic.						
12.13.5	**Linking information …** Links shall provide a clear indication of the destination of the link.						
12.14	**Documentation formats for finding information …** As presented in *Clause 10.4*, documentation shall contain features to provide access to information, such as a table of contents, an index, and search capabilities.						
12.14.1	**Table of contents …** The table of contents shall list the headings of the chapters or topics of a document with an access point for each (its initial page number or an electronic link).						
12.14.1	**Table of contents …** At least one volume of a document set shall include a simple table of contents for all volumes in the set (see *Clause 10.4*).						
12.14.1	**Table of contents …** The table of contents shall include all portions of the documentation, including front matter that follows the table of contents and back matter (for example, appendixes, glossary, and index).						
12.14.1	**Table of contents …** The headings in the table of contents shall be identical in wording to those in the document, including chapter or topic numbers.						

Clause no.	Guideline	Applicability		Conformance			
		Yes or No	Reason not applicable	Yes	Partial	No	Comments
12.14.1	**Table of contents …** In printed documentation, the table of contents shall immediately follow the identification data (see *Clause 11.3*).						
12.14.1	**Table of contents …** The format of the table of contents shall distinguish the hierarchy of headings by consistent typography or indentation.						
12.14.1	**Table of contents …** In printed documentation, the table of contents shall list the headings in the same order as in the text.						
12.14.3	**List of illustrations …** The list of illustrations shall list the illustration numbers and titles with an access point for each (such as its initial page number or an electronic link).						
12.14.3	**List of illustrations …** The titles in the list of tables, figures, or illustrations shall be identical in wording to those in the document, including table, figure, or illustration numbers.						
12.14.4	**Index …** On-screen documents more than about 40 topics shall include either a search tool (see *Clause 12.14.5*) or an index whose access points are electronic links.						
12.14.4	**Index …** An index entry may cross-reference another index entry; however, the referenced entry shall give an access point to the documentation and shall not point to a third index entry.						
12.14.5	**Search capability …** Onscreen documentation shall provide a method of locating words in the text.						
12.15	**Formats for warnings, cautions, and notes …** Warnings, cautions, and notes shall be displayed in a consistent format that is readily distinguishable from ordinary text or instructional steps.						

Clause no.	Guideline	Applicability		Conformance			
		Yes or No	Reason not applicable	Yes	Partial	No	Comments
12.15	**Formats for warnings, cautions, and notes …** The flag word (for example, “warning,” “caution,” or “note”) shall precede the accompanying text.						
12.15	**Formats for warnings, cautions, and notes …** The term “note” shall not be used to identify hazards.						
12.15	**Formats for warnings, cautions, and notes …** Warnings and cautions shall be identified by consistent and distinct graphical symbols, for example, an exclamation point or lightning bolt inside a triangle.						
12.16	**Format for instructions …** Instructional steps shall be consecutively numbered.						
12.18.4	**Consistent presentation of illustrations …** The format of illustrations of similar content shall be consistent for scale, screen size, fonts, line thickness, and use of color.						
12.18.8	**Tables …** Long tables that cannot fit on a single page shall repeat the table title and column or row headings on each page or two-page spread.						

Bibliography

[1] ISO 216:2007, *Writing paper and certain classes of printed matter — Trimmed sizes — A and B series, and indication of machine direction*

[2] ISO 999:1996, *Information and documentation — Guidelines for the content, organization and presentation of indexes*

[3] ISO 6357:1985, *Documentation — Spine titles on books and other publications*

[4] ISO 9000:2005, *Quality management systems — Fundamentals and vocabulary*

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

[6] ISO/IEC TR 9126-2:2003, *Software engineering — Product quality — Part 2: External metrics*

[7] ISO/IEC TR 9126-3:2003, *Software engineering — Product quality — Part 3: Internal metrics*

[8] ISO/IEC TR 9126-4:2004, *Software engineering — Product quality — Part 4: Quality in use metrics*

[9] ISO/IEC TR 9294:2005, *Information technology — Guidelines for the management of software documentation*

[10] ISO 9241-1 to ISO 9241-17, *Ergonomic requirements for office work with visual display terminals (VDTs)*

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

[12] ISO 9241-12:1998, *Ergonomic requirements for office work with visual display terminals (VDTs) — Part 12: Presentation of information*

[13] ISO 9241-110:2006, *Ergonomics of human-system interaction — Part 110: Dialogue principles*

[14] ISO 9241-171:—[1)], *Ergonomics of human-system interaction — Part 171: Guidance on software accessibility*

[15] ISO 10007:2003, *Quality management systems — Guidelines for configuration management*

[16] ISO/IEC 11581-1:2000, *Information technology — User system interfaces and symbols — Icon symbols and functions — Part 1: Icons — General*

[17] ISO/IEC 11581-2:2000, *Information technology — User system interfaces and symbols — Icon symbols and functions — Part 2: Object icons*

[18] ISO/IEC 11581-3:2000, *Information technology — User system interfaces and symbols — Icon symbols and functions — Part 3: Pointer icons*

[19] ISO/IEC 11581-5:2004, *Information technology — User system interfaces and symbols — Icon symbols and functions — Part 5: Tool icons*

[20] ISO/IEC 11581-6:1999, *Information technology — User system interfaces and symbols — Icon symbols and functions — Part 6: Action icons*

1) To be published.

[21] ISO/IEC 12207:2008, *Systems and software engineering — Software life cycle processes*

[22] ISO 13407:1999, *Human-centred design processes for interactive systems*

[23] ISO/IEC 14598-1:1999, *Information technology — Software product evaluation — Part 1: General overview*

[24] ISO/IEC 14598-2:2000, *Software engineering — Product evaluation — Part 2: Planning and management*

[25] ISO/IEC 14598-3:2000, *Software engineering — Product evaluation — Part 3: Process for developers*

[26] ISO/IEC 14598-4:1999, *Software engineering — Product evaluation — Part 4: Process for acquirers*

[27] ISO/IEC 14598-5:1998, *Information technology — Software product evaluation — Part 5: Process for evaluators*

[28] ISO/IEC 14598-6:2001, *Software engineering — Product evaluation — Part 6: Documentation of evaluation modules*

[29] ISO/IEC 15288:2008, *Systems and software engineering — System life cycle processes*

[30] ISO/IEC 15289:2006, *Systems and software engineering — Content of systems and software life cycle process information products (Documentation)*

[31] ISO/IEC 18019:2004, *Software and system engineering — Guidelines for the design and preparation of user documentation for application software*

[32] ISO 80416-2:2001, *Basic principles for graphical symbols for use on equipment — Part 2: Form and use of arrows*

[33] IEEE 1063-2001, *IEEE standard for software user documentation*

[34] Barker, T. T. *Writing Software Documentation: A Task-Oriented Approach*, 2nd ed. New York: Longman, 2002

[35] Barnum, C. M. *Usability Testing and Research*. New York: Longman, 2002

[36] Bias, R. G., and Mayhew, D. J. *Cost-Justifying Usability, Second Edition: An Update for the Internet Age*, 2nd ed. San Francisco: Morgan Kaufmann, 2005

[37] Dicks, R. S. *Management Principles and Practices for Technical Communicators*. New York: Longman, 2004

[38] Farkas, D. K., and Farkas, J. B. *Principles of Web Design*. New York: Longman, 2002

[39] Hackos, J. T., and Redish, J. C. *User and Task Analysis for Interface Design*. New York: John Wiley & Sons, 1998

[40] Hackos, J. T., and Stevens, D. M. *Standards for Online Communication: Publishing Information for the Internet/World Wide Web/Help Systems/Corporate Internets*. New York: John Wiley & Sons, 1997

[41] Hackos, J. T. *Managing Your Documentation Projects*. New York: John Wiley & Sons, 1994

[42] Hargis, G., et al. *Developing Quality Technical Information*. Upper Saddle River: Pearson Education, 2004

[43] Kostelnick, C., and Roberts, D. D. *Designing Visual Language: Strategies for Professional Communicators*. New York: Longman, 1997

[44] Nielsen, J., and Loranger, H. *Prioritizing Web Usability*. Indianapolis: New Riders Press, 2006

[45] Rockley, A. *Managing Enterprise Content: A Unified Content Strategy*. Indianapolis: New Riders Press, 2003

[46] Schriver, K. A. *Dynamics in Document Design: Creating Texts for Users*. New York: John Wiley & Sons, 1994

[47] Section 508 of the US Rehabilitation Act, http://www.section508.gov/

[48] Tufte, E. *The Visual Display of Quantitative Information*. Cheshire, CT: Graphics Press, 1983

[49] Wheildon, C. *Type & Layout: Are You Communicating or Just Making Pretty Shapes?* 2nd ed. Mentone, Australia: The Worsley Press, 2005

ICS 35.080

Price based on 143 pages