

Australian/New Zealand Standard™

**Software and system engineering—
Guidelines for the design and
preparation of user documentation for
application software**



AS/NZS ISO/IEC 18019:2007

This Joint Australian/New Zealand Standard was prepared by Joint Technical Committee IT-015, Software and Systems Engineering. It was approved on behalf of the Council of Standards Australia on 3 November 2006 and on behalf of the Council of Standards New Zealand on 17 November 2006.
This Standard was published on 17 January 2007.

The following are represented on Committee IT-015:

Australian Computer Society
Australian Electrical and Electronic Manufacturers Association
Australian Society of Technical Communications
Australian Software Metrics Association
Engineers Australia/ACTS Joint Board in Software Engineering
Griffith University
National Association of Testing Authorities Australia
National ICT Australia
New Zealand Organisation for Quality
Software Quality Association, ACT
Software Quality Association, NSW
Systems Engineering Society of Australia
The University of Queensland
University of Auckland, NZ
University of South Australia
University of Technology, Sydney
Vendor Interests, NZ

Keeping Standards up-to-date

Standards are living documents which reflect progress in science, technology and systems. To maintain their currency, all Standards are periodically reviewed, and new editions are published. Between editions, amendments may be issued. Standards may also be withdrawn. It is important that readers assure themselves they are using a current Standard, which should include any amendments which may have been published since the Standard was purchased.

Detailed information about joint Australian/New Zealand Standards can be found by visiting the Standards Web Shop at www.standards.com.au or Standards New Zealand web site at www.standards.co.nz and looking up the relevant Standard in the on-line catalogue.

Alternatively, both organizations publish an annual printed Catalogue with full details of all current Standards. For more frequent listings or notification of revisions, amendments and withdrawals, Standards Australia and Standards New Zealand offer a number of update options. For information about these services, users should contact their respective national Standards organization.

We also welcome suggestions for improvement in our Standards, and especially encourage readers to notify us immediately of any apparent inaccuracies or ambiguities. Please address your comments to the Chief Executive of either Standards Australia or Standards New Zealand at the address shown on the back cover.

This Standard was issued in draft form for comment as DR 06507.

Australian/New Zealand Standard™

Software and system engineering— Guidelines for the design and preparation of user documentation for application software

First published as AS/NZS ISO/IEC 18019:2007.

COPYRIGHT

© Standards Australia/Standards New Zealand

All rights are reserved. No part of this work may be reproduced or copied in any form or by any means, electronic or mechanical, including photocopying, without the written permission of the publisher.

Jointly published by Standards Australia, GPO Box 476, Sydney, NSW 2001 and Standards New Zealand, Private Bag 2439, Wellington 6020

ISBN 0 7337 7976 X

PREFACE

This Standard was prepared by the Joint Standards Australia/Standards New Zealand Committee IT-015, Software and Systems Engineering.

The objective of this Standard is to provide people responsible for specifying, designing and preparing user documentation for application software and people who manage these activities, with guidelines for the design and preparation of user documentation for application software.

This Standard is identical with, and has been reproduced from ISO/IEC 18019:2004, *Software and system engineering—Guidelines for the design and preparation of user documentation for application software*.

As this Standard is reproduced from an international standard, the following applies:

- (a) Its number appears on the cover and title page while the international standard number appears only on the cover.
- (b) In the source text ‘this International Standard’ should read ‘this Australian/New Zealand Standard’.
- (c) A full point substitutes for a comma when referring to a decimal marker.

The term ‘informative’ has been used in this Standard to define the application of the annex to which it applies. An ‘informative’ annex is only for information and guidance.

CONTENTS

	<i>Page</i>
1	Scope..... 1
2	Terms and definitions..... 2
3	Overview 8
3.1	Forms of documentation 8
3.2	Deciding what form of documentation to use..... 9
3.2.1	General 9
3.2.2	Information that needs to be on the screen 9
3.2.3	Information that generally needs to be on paper..... 9
3.3	Overview of the structure of this International Standard 10
3.3.1	General 10
3.3.2	Checklists 12
4	The objectives phase..... 12
4.1	General 12
4.2	Collect and interpret project requirements and constraints 12
4.2.1	General 12
4.2.2	Product objectives 13
4.2.3	Sales objectives 14
4.2.4	Scheduling objectives 14
4.2.5	Usability objectives..... 15
4.2.6	Accessibility objectives..... 15
4.2.7	Modification requirements 15
4.2.8	Internationalisation and national cultural requirements..... 16
4.2.9	Translation requirements 16
4.2.10	Packaging requirements 17
4.2.11	Legal requirements 17
4.2.12	Security 18
4.2.13	Standards and conventions 18
4.2.14	Cost constraints..... 19
4.2.15	Documentation delivery and viewing mechanisms..... 19
4.2.16	Quality management..... 19
4.2.17	Provision of technical information 19
4.2.18	Approval authorities 20
4.2.19	Configuration management 20
4.2.20	Availability of resources..... 20
4.3	Documentation Proposal..... 21
5	The planning phase 23
5.1	General 23
5.2	Documentation plan..... 23
5.2.1	General 23
5.2.2	Standards..... 24
5.2.3	Version control and change control 24
5.2.4	Personnel 24
5.2.5	Equipment..... 25
5.2.6	Responsibilities..... 25
5.2.7	Cost estimates..... 26
5.2.8	Schedules 27
5.2.9	Prototypes and drafts 27

5.2.10	System tests	28
5.2.11	Reviews	28
5.2.12	Usability testing	29
5.2.13	Localisation and customisation	29
5.2.14	Approval	29
5.2.15	Maintenance, updating and future developments	30
5.3	Review of detailed documentation plans	30
6	The analysis and design phase	30
6.1	Audiences	31
6.1.1	Audience analysis	31
6.1.2	Learning stages and frequency of use	32
6.1.3	Working environments	32
6.1.4	Audience profiles	33
6.2	Tasks	34
6.2.1	Task analysis	34
6.2.2	Mapping audiences to tasks	35
6.2.3	Task characteristics	36
6.2.4	Task profiles	36
6.3	Information	36
6.3.1	Information needs	36
6.3.2	Context of use	36
6.3.3	Volume/amount of documentation	37
6.3.4	Media	37
6.3.5	Information profile	39
6.4	Usability	39
6.4.1	Define usability goals	39
6.4.2	Record usability goals	41
6.5	Structure of the documentation suite	41
6.5.1	General	41
6.5.2	Decide what information needs to be provided in the documentation	41
6.5.3	Group information needs into documents	41
6.6	Individual document structures	43
6.6.1	Prepare a list of contents	43
6.6.2	Define the document structure	43
6.7	Document writing style	46
6.7.1	General	46
6.7.2	Awareness and appreciation information	46
6.7.3	Installation instructions	46
6.7.4	Tutorials and task instructions	47
6.7.5	Quick reference information	47
6.7.6	Reference information	47
6.7.7	Diagrams	48
6.7.8	Graphs and charts	48
6.7.9	Illustrations of screen displays	48
6.7.10	Illustrations of printed output	49
7	The development and review phase	50
7.1	General	50
7.2	Prepare and issue drafts	50
7.3	Check and review drafts	51
7.3.1	General	51
7.3.2	Reviewing the information	52
7.3.3	Usability tests	53
7.3.4	System tests	54
7.3.5	Validation and field trials	54
7.4	Prepare subsequent drafts	55
7.5	Prepare document masters	55
7.6	Hand over the finished documentation	56
7.7	Localisation and customisation	56
7.8	Archiving	56

8	The evaluation and updating phase	57
8.1	General	57
8.2	Evaluate the documentation	57
8.3	Update the documentation	57
9	Guidelines for the design of documentation.....	57
9.1	Introduction	57
9.2	Product copyright and version details	58
9.3	Overview of the documentation	59
9.4	Process descriptions	59
9.5	Task descriptions.....	60
9.6	Explanations of fields and options	61
9.7	Names and uses of user interface options.....	62
9.7.1	Names.....	62
9.7.2	Uses	62
9.8	Descriptions of application functions.....	62
9.9	Information messages	64
9.9.1	Format	64
9.9.2	On-screen messages	65
9.10	Definitions of terms	66
9.11	Concepts	67
9.12	Exploitation information.....	67
9.13	Frequently asked questions.....	68
9.14	User-supplied content	68
9.15	Navigation.....	69
9.15.1	Introduction	69
9.15.2	Accessing on-screen information	70
9.15.3	Finding the right information - linking information in on-screen documentation.....	71
9.15.4	Knowing what the current information is	74
9.15.5	Knowing the current position within a topic	75
9.15.6	Finding the same information again	75
9.15.7	Switching between the application and the documentation	75
9.15.8	Printing information	76
9.15.9	Moving to a different topic	76
9.15.10	Obtaining clarification or amplification of current information	77
9.15.11	Browsing through information	77
9.15.12	Viewing topics in sequence	77
9.15.13	Exiting from the on-screen documentation.....	77
9.15.14	Finding user-supplied information.....	77
9.15.15	Sizes of topics and fragments	78
9.16	Presentation	78
9.16.1	Introduction	78
9.16.2	Windowing	79
9.16.3	Layout and grids	80
9.16.4	Colour.....	82
9.16.5	Presentation of text.....	84
9.17	Icons and signposts	90
9.17.1	When to use icons and signposts.....	90
9.17.2	Design of icons and signposts.....	90
9.17.3	Displaying the names of icons	91
9.18	Presentation of illustrations.....	92
Annex A (informative)	Process checklists	93
Annex B (informative)	Design checklist	98
Annex C (informative)	Evaluation of documentation.....	114
Annex D (informative)	Writing style and techniques	119
Annex E (informative)	Design and preparation of printed information.....	131
Annex F (informative)	Writing style guides — Contents	144

	<i>Page</i>
Annex G (informative) ISO/IEC 18019 and related standards.....	145
Bibliography.....	146

INTRODUCTION

Anyone who uses application software needs accurate information about the correct way to use it. If the information is supplied in a convenient form and is easy to find and understand, the users can quickly become proficient at using the product. Consequently their view of the product is positive, with the result that their view of the supplier is positive too. 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 many software products aim to have user interfaces that behave so intuitively that very little separate documentation is needed, this is rarely possible.

Documentation is an essential component of any product. Documentation design is crucial; the success or failure of an entire product can depend on it. The documentation can be the first tangible item that the user sees, and so influences the user's first impressions of the product.

Users of application software products generally have one important feature in common: they might be experts in the tasks for which they wish to use the software, but they are not, initially, experts in using the application software itself.

Although the guidance given in this International Standard covers all the activities and all the design decisions that need to be made, some of the activities can be extremely simple to carry out in some environments, as demonstrated by the following examples.

- If there are already established typographic and illustration standards and established development and production routes, very little design and planning will be needed in these areas.
- If the product being developed is for a single type of user with well-known user characteristics and well-defined tasks, very little user analysis will be needed.

ISO/IEC 18019 is based upon British Standards BS 7649:1993 and BS 7830:1996.

AUSTRALIAN/NEW ZEALAND STANDARD

Software and system engineering — Guidelines for the design and preparation of user documentation for application software

1 Scope

This International Standard gives guidelines for the design and preparation of user documentation for application software. 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 then to prepare the information and make it available.

For the purposes of this International Standard, application software includes the types listed below.

- Consumer software packages, that is, software products designed and sold to carry out identified tasks, where the software and its associated documentation are packaged for acquisition as a unit.
- Software for office applications such as word processors, spreadsheets, databases and electronic mail.
- Business software, for example, software for recording and monitoring business activities, such as stock control and order processing.
- Specialist software for use by professionals, such as accounting systems, graphic design systems and engineering design systems.

These guidelines may also be helpful for developing documentation for the following, although it does not cover all the issues relating to them.

- Software engineering products for use by computer professionals.
- Software for programmable electronic or mechanical systems.

This International Standard is for use by people responsible for specifying, designing and preparing user documentation for application software and people who manage these activities, including.

- Developers of tools for creating hardcopy documentation.
- Product designers.
- Application developers.
- Project managers.
- Authors.
- Programmers.
- Translators.
- Localisation staff.

It is intended for use in all types of organisations, whether or not a dedicated documentation department is present. In all cases, it can 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.

This International Standard may also be useful to.

- Developers of tools for creating on-screen documentation.
- People who are evaluating existing or proposed application software.

2 Terms and definitions

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

2.1

accessibility

successful access to information and use of information technology by people who have disabilities

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

2.2

active area

area of a screen interface that responds to user input

EXAMPLE A window, icon or text field.

2.3

active text

text displayed on the screen that responds to user input

2.4

alpha testing

first stage of testing before a product is considered ready for commercial or operational use; often performed only by users within the organisation developing the software (see also 2.11)

2.5

analysis

investigation and collection phase of development, that aims to specify types of users and their information needs

2.6

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

2.7

application window

window (on-screen location) that presents an environment or application

2.8

appreciation information

awareness information

information that introduces the product to potential users, tells them what the software can do, how it can be used and helps them decide whether the product is appropriate to their needs

2.9**audience**

category of users sharing the same or similar characteristics and needs (e.g. purpose in using the documentation, tasks, education level, abilities, training, experience) that determine the content, structure and use of the intended documentation

NOTE There may be a number of different audiences for a software product's documentation (e.g. management, data entry, maintenance).

[ISO/IEC 15910, definition 4.3]

2.10**author**

person designing or developing user documentation

2.11**beta testing**

final stage of testing for a computer product prior to commercial or operational release; normally involves sending the product to beta test sites outside of the company for real use exposure (see also 2.4)

2.12**bubble help**

hover help

flyover help

embedded documentation, in the form of on-screen help information consisting of small boxes containing concise text that describe items on the screen

NOTE A help bubble appears when the user moves the pointer to an item and disappears when the user moves the pointer away from the item.

2.13**change control procedures**

actions taken to identify, document, review and authorise any changes to a product being developed

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

2.14**chrome**

that part of the application or web browser window that lies outside the content area of the window

NOTE Title bar, status bar, scroll bars, menu bar, tool bar and location bar, are all examples of elements of the browser window that are part of the chrome. Web browser windows can be opened with or without elements of chrome visible by the inclusion of appropriate programming in the page to be displayed.

2.15**configuration management**

technical and organisational activities comprising configuration identification, control, status accounting and auditing (see also 2.50)

[ISO 10007:1995, definition 3.9]

2.16**context sensitive help**

information relevant to the user's current context in the application displayed when requested by the user

2.17**context sensitivity**

ability of on-screen documentation systems to react differently according to the state of the user's interaction with the application

2.18**customisation**

process of adapting a product to the needs of a particular user or group of users

2.19**design**

phase of development concerned with determining what documentation will be provided in a product and what the nature of the documentation will be (see also 2.33)

2.20**development**

process of preparing documentation including phases for objectives, analysis, design and implementation, which are planned and controlled as a unit

2.21**display**, noun

information presented on a screen or in a window of a screen

2.22**document**, noun

equivalent to an item of documentation

[ISO/IEC 15910, definition 4.10]

2.23**documentation**

printed user manuals, on-screen information and help text that describe how to use a software product

[ISO/IEC 15910, definition 4.11]

2.24**documentation plan**

written statement of the essential elements of the documentation project

[ISO/IEC 15910, definition 4.13]

2.25**documentation suite**

complete collection of documents comprising CDs, online help, printed manuals, etc. provided to support the user of a software product

2.26**embedded documentation**

information that is delivered as an integral part of a piece of software (see also 2.53 and 3.1)

EXAMPLE On-screen help.

2.27**entry field**

area on a screen or in a window in which data is entered

2.28**escrow**

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

2.29**fragment**

small piece of information about a single object such as an icon, a word or a field, that can be retrieved or displayed separately (see also 2.58)

EXAMPLE Text that appears in a help bubble (see 2.12).

2.30**function**

part of an application that provides facilities for users to carry out their tasks, such as a module, a command, a dialogue box, a transaction screen and their equivalents

2.31**hypertext**

means of presenting information online with connections (called hypertext links) between one piece of information and another.

2.32**icon**

graphic displayed on the screen that represents a function of the computer system or application or on-screen documentation system

[ISO/IEC 11581-1, definition 4.7]

2.33**implementation**

phase of development in which documentation is prepared (see also 2.19)

2.34**internationalisation**

process of developing information so that it is suitable for an international audience and can be localised

2.35**link**

navigation method that takes the user from one item of on-screen documentation to another item

2.36**localisation**

process of creating a national or specific regional linguistic version of a product

NOTE Localisation can be carried out separately from the translation process.

2.37**menu**

list of options from which the user can choose

2.38**navigation**

process of accessing on-screen documentation and moving between different items of information

2.39**on-screen documentation**

information about the software that is intended to be read on the computer screen by the user while using the software

2.40**printed documentation**

information about the software which is either provided in printed form, or in electronic form and primarily intended to be printed by the customer or user

2.41**picture**

illustration that shows the actual appearance of physical objects

EXAMPLE Photographs, drawings or a reproduction of a screen display.

2.42**platform**

computing environment with a particular user or programming interface, including hardware and operating system, supporting execution of application programs

2.43**pop-up**

menu that, when requested, is displayed next to the object it is associated with; it contains choices appropriate for a given object or set of objects in their current context

2.44**primary window**

window in which the main dialogue between the user and the application takes place

2.45**process**

set of interrelated activities, which transform inputs into outputs

[ISO/IEC 12207:1995, definition 3.17]

2.46**product**

software product

complete set of computer programs, procedures and associated documentation and data designed for delivery to a user

2.47**product authority**

person with overall responsibility for the capabilities and quality of a product

2.48**project**

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

2.49**project manager**

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

2.50**quality management**

coordinated activities to direct and control an organisation with regard to quality (see also 2.15)

[ISO 9000:2000, definition 3.2.8]

2.51**real world object**

entity that exists in a three dimensional form, and by association infers similar properties or behaviour to software functions

EXAMPLES Printer, filing cabinet, file folder, sheet of paper.

2.52**secondary window**

window containing information that is dependent on information in a primary window and is used to supplement the information in the primary window

2.53**separate documentation**

information that is provided independently of the software (see also 2.26 and 3.1)

EXAMPLE Printed manuals and free-standing hypertext systems.

2.54**signpost**

text, symbol or a small graphic used to help the user identify where particular types of information are given or where the information in the current display fits into the whole

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

2.55**software**

the part of a product that is the computer program or the set of computer programs

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

2.56**technical contact**

person responsible for providing an author with technical information about a product or for checking the technical accuracy of drafts of user documentation

2.57**technical specification**

data, listing the document grid, use of colour, typographic requirements, page sizes, etc.

2.58**topic**

individually named chunk of information on a single subject that is presented within the printed documentation or that can be retrieved and displayed separately as part of the on-screen documentation (see also 2.29)

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

EXAMPLE Instructions on how to print the current document.

2.59**usability**

extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use

[ISO 13407:1999, definition 2.3 and ISO 9241-11:1998]

2.60**user**

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

2.61**user documentation**

information that is supplied with the software to help the user in their use of that software

2.62**user interface**

ensemble of software and hardware that enables a user to interact with a computer system

2.63**window**

information window

area with visible boundaries that presents a view of an object or through which a user conducts a dialogue with a computer system

2.64**wizard**

form of user guidance that automates a task through dialogue with the user

3 Overview**3.1 Forms of documentation**

There are a number of different ways of classifying software documentation, for example:

- embedded documentation, which is delivered as an integral part of the software versus separate documentation, which is provided independently of the software;
- printed documentation versus on-screen documentation.

These classifications overlap. This International Standard covers software documentation in all of these classifications.

Examples of embedded documentation are:

- context-sensitive help;
- wizards;
- material that appears automatically without a request from the user;
- bubble help.

NOTE Embedded documentation is intended primarily to be read on-screen.

Examples of separate documentation are:

- printed documentation, such as manuals;
- supporting web sites;
- help files which are not context sensitive;
- documentation delivered as read-only, viewable or printable electronic files on a CD.

NOTE Separate documentation can be read either as on-screen information or be printed and read on paper.

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 video and sound;
- on-screen tutorials and computer-based training (CBT) packages;

- maintenance documentation describing the internal operation of software;
- dynamically generated documentation.

3.2 Deciding what form of documentation to use

3.2.1 General

The major factors that should influence decisions about where and how information is provided to the user are:

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

3.2.2 Information that needs to be on the screen

There is usually some information that the user needs to have as on-screen information, including:

- names of all elements of the user interface;
- functions of all elements of the user interface, including commands, icons, buttons and dialogue boxes;
- error and other messages;
- help information;
- information explaining what the system will do, is doing or has done as a result of the user taking some action.

3.2.3 Information that generally needs to be on paper

There is usually some information that the user needs as printed documentation. Often, when a system becomes unusable, it can be the only help available to the user. This information includes:

- product and version information needed to select the package

NOTE ISO/IEC 9127 provides guidance on the minimum information recommended for inclusion on software packaging.

- instructions for handling the package;
- pre-requisite information, saying what equipment, system environment and so on, are needed to run the application;
- instructions for configuring the hardware to run the application;
- installation instructions;

- diagnostic and migration information;
- instructions for starting the application;
- what to do when all else fails, such as where to call for support or help.

In addition to the items listed above, the following types of information should be considered for presentation on paper or in a printable electronic form:

- initial familiarisation information;
- long examples;
- structured reference material, particularly for advanced features of the software;
- checklists;
- guides to use of the keyboard and other input devices, which may be better provided on the device itself or on a template attached to the device.

NOTE Video clips can be incorporated into electronic documents and are very useful for showing sequences that can not be adequately covered by a series of VDU screens, for example "How to load paper into the printer", that is "human intervention tasks".

3.3 Overview of the structure of this International Standard

3.3.1 General

This International Standard contains two types of guideline (process guidelines and design guidelines) and is structured to reflect this.

NOTE Annex G provides an illustration of the relationships of ISO/IEC 18019 to other related standards.

3.3.1.1 Process guidelines

The process guidelines cover the phases involved in designing, specifying and producing user documentation.

If the separate components of an application are to appear to the user as a coherent package, all the components need to be specified together. Therefore, the specification of all the documentation, including on-screen documentation and printed documentation, should be a part of the development of the product as a whole, not a separate exercise. Authors should be included in the development team as equal members; they can also be very useful early testers of the product.

The six process phases are as follows:

- objectives: finding out the documentation policy of the enterprise, what the project is about and what has to be achieved (see Clause 4 and ISO/IEC 12207, 6.1.1.1);
- planning and control: drawing up the documentation plans for the project, including monitoring and controlling it (see Clause 5, and ISO/IEC 12207, 6.1.2.1 and 6.1.3.2);
- analysis and design: collecting information about the product and users, their tasks and their needs for information, and designing documentation based upon those needs (see Clause 6 for process guidance and Clause 9 for design guidance and ISO/IEC 12207, 6.1.2.1);
- development and review: using the enterprise, project and product (internal) standards (for example, templates) to develop the product, and preparing master versions of the documents (see Clause 7 and ISO/IEC 12207, 6.1.2.3);

- evaluation and updating: evaluating the documentation with the rest of the product (see Clause 8 and ISO/IEC 12207, 6.1.4);
- production: preparing the printed documentation on suitable media (see Annex E, and ISO/IEC 12207, 6.1.3).

Although there is probably a clear starting point for developing documentation, and a clear end point, there is not a single sequence of activities 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 very closely inter-linked, as are analysis and design, and the way they link together varies between projects.

In developing each product, therefore, draw up a detailed plan showing what documentation activities will be carried out and in what sequence.

Although accurate user documentation cannot be completed until the software product has been fully developed, the user documentation and the product both benefit from concurrent development. In particular, the activities described in this standard should be initiated along with the following activities of the ISO/IEC 12207 development process (5.3).

NOTE ISO/IEC 15910 can be used as a contractual document to define and enforce the process between the organisation developing the documentation and the organisation buying it.

ISO/IEC 12207 addresses documentation as part of the software life cycle. In many cases ISO/IEC 12207 may state that the result of a process shall be documented or may imply the need for a document resulting from a task. Figure 1 shows a comparison between the documentation processes in ISO/IEC 12207 and ISO/IEC 18019.

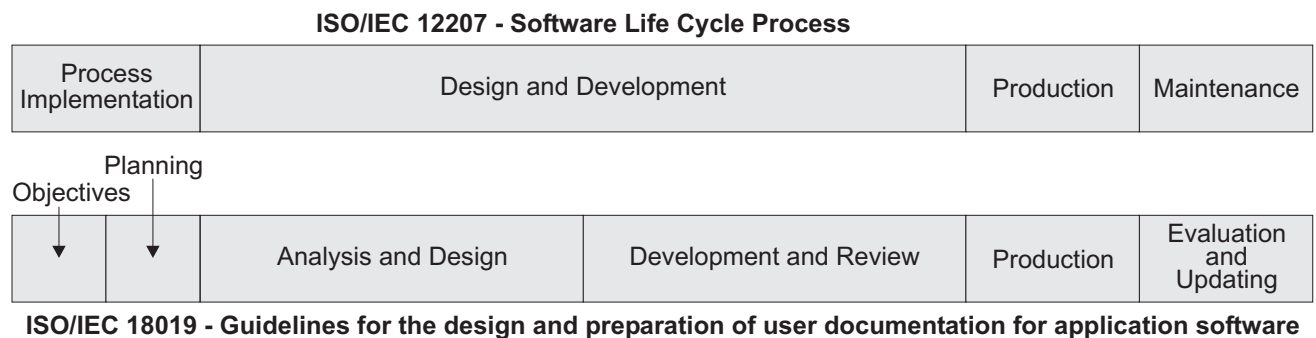


Figure 1 — Comparison of documentation processes between ISO/IEC 12207 and ISO/IEC 18019

3.3.1.2 Design guidelines

The design guidelines cover the wording, format and presentation of the user documentation. They include:

- what to tell the user in each type of document. For example, what information to include in each sort of topic being provided in the documentation;
- how to structure the information and what facilities to provide for navigating the on-screen documentation;
- styles and techniques for writing documents and producing illustrations. Additional information is given in Annex D;
- how to present information so that it is easily understood.

Design guidelines are addressed in Clause 9, with additional recommendations for printed documentation given in Annex E.

3.3.2 Checklists

The checklists in Annexes A and B can be used at each stage of the documentation development to check that the appropriate steps have been carried out.

4 The objectives phase

4.1 General

The objectives phase has the following purposes:

- to understand the project, particularly the work that will require user documentation, including:
 - the project objectives;
 - the requirements and objectives for the product.
- to gauge the scale of the project particularly the work that will require user documentation;
- to understand the documentation policy of the producer of the software product;
- to formulate, where possible, an initial strategy on how the software is to be documented.

NOTE It may be necessary to view the wider context of the whole project, to understand the requirements for the documentation components.

The objectives are used to find out:

- what has to be achieved;
- what opportunities exist;
- what constraints may apply to project finances, time, staffing and equipment.

4.2 Collect and interpret project requirements and constraints

4.2.1 General

The types of objectives, requirements and constraints that should be considered for every project are covered in 4.2.2 to 4.2.20.

Interpret the initial requirements and check them thoroughly to ensure they are accurate and realistic. Take account of the following:

- cost or scheduling constraints;
- market requirements. For example a document with a particular title, such as a 'User guide' or a 'Reference card';
- suggestions about the style and content of the documentation. Following review, these may be rejected, or may become formal requirements;
- page size and book format;
- specific constraints on the maximum number of pages (for printed documentation) or file size (for on-screen documentation);

- number of languages the documentation is to be translated into;
- how to make information easily translatable;
- selling price of books, identifying which ones to distribute free with the software.

NOTE 1 For example, subjective requests for 'lots of pictures' or for 'not too much detail' may be resolved into statements about the expected audience or usability targets.

NOTE 2 If the same information is needed in both embedded and separate documentation for users carrying out the same tasks, then that information may be used in both types of documentation, in each place structured appropriately.

NOTE 3 Constraints based solely on a personal preference should not be treated as requirements for the documentation.

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

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

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

4.2.2 Product objectives

The overriding project objective should be to develop a product that meets user needs. Constraints for the documentation should not be allowed to change the product in terms of the real problems which it should solve, and the range of hardware and operating systems for which it is being developed.

Authors need the following information.

- What is the purpose of the 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? (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 organisation or organisations? If so, will that always be the case?
- Will localised versions of the product be required?
- Is there a need to provide extra information, such as the working procedures of the users' organisations or information about the product itself? If so, is that information readily available?
- How will the documentation be maintained and updated?

4.2.3 Sales objectives

For printed documentation, the important features of the product's sales objectives are:

- the target quantity of the product over its projected lifetime;
- the expected profile of sales over time: will demand be steady or will there be a period of peak demand?

4.2.4 Scheduling objectives

When the project schedules are being defined:

- remember that documentation development cannot be finished until after the software design is frozen;
- remember, the amount of time needed for the following activities may be significant; make adequate allowance for them in the project's schedules;
 - allow time for authors to observe and use the software or prototypes of the software during the development stage;
 - identify technical work necessary to include the embedded documentation in the product;
 - allow time for the graphics to be created and screen captures to be taken;
 - allow time for technical contacts to supply information and to check the accuracy of drafts;
 - make draft documentation available for product validation, field trials and usability trials (documentation development schedules for preparing such drafts will affect the timing of these exercises and trials);
 - when the documentation has been developed, allow time for translation, if required, and for printing and packaging.

Authors need to know the following about the product schedule.

- When will the alpha, beta and acceptance testing start, and be completed?
- What is the delivery date for the finished product?
- 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 localised or customised, what are the required delivery dates for the various localised and customised versions and how long before those delivery dates does the documentation need to be ready?

Once a date for delivering the application has been agreed, it is important that it is used in all phases of the planning (see Clause 6).

Authors and the project manager should agree on the overall schedules for the project. Once the schedules are agreed, authors can design documentation that can be prepared within the required time and cost.

4.2.5 Usability objectives

When the usability targets are set and measured for the product, treat the documentation as an integral part of the product. Measures of the usability of the documentation, independent of the usability of the software, might include the following:

- the time taken to learn about the contents of the documentation, particularly if more than one document is supplied;
- the time taken to understand the document structure and to learn how to use it;
- the time taken to find information, once the user is familiar with the documentation;
- the time taken to learn how to do a specified task using the instructions in the documentation;
- the time taken to incorporate amended pages.

Measures of satisfaction for the documentation could use responses to questions such as the following.

- How satisfied are you with the layout of the document?
- How satisfied are you with the time that it took you to find the information you needed?
- Treating each topic separately, how satisfied are you with the completeness of the information?
- Can you accomplish the task in hand using the documentation?

Set durability targets for individual printed documents according to the type and amount of use they will have.

4.2.6 Accessibility objectives

To meet accessibility needs or objectives, the documentation may be required to:

- provide a high level topic titled "Accessibility" and describe the accessibility features of the software product;
- be accessible itself. For on-screen documentation, file formats would need to be able to be read by screen reading programs, which in turn can output to speech synthesis or refreshable-Braille output devices. Graphics and illustrations should have meaningful titles and descriptions;
- be given special consideration concerning the logical structure, graphics, "page turning", alert messages, entry fields, headings, hypertext links, etc, of on-screen documentation.

NOTE The US Government has published specific requirements for software accessibility, known as Section 508 of the Rehabilitation Act. Details of Section 508 can be found at the URL listed in the bibliography.

4.2.7 Modification requirements

Review plans for making modified versions of the product available to users because modifications can affect all aspects of the documentation design.

Find out the plans for future modifications of the application, using the following questions.

- Will completely new versions of the product be issued?
- Will amendments be made available to existing users?
- At what intervals will the product be modified?

- What different levels of modification will there be, for example, for:
 - issuing temporary corrections?
 - issuing interim versions?
 - issuing major upgrades?
- What are the time scales for making modifications?
- What are the cost constraints?
- How will the documentation be updated?

4.2.8 Internationalisation and national cultural requirements

Determine the following:

- Will the application be used by an international audience?
 - If so, in what countries?
 - Are localised versions to be developed?
- Are customised versions to be developed?
 - If so, for which organisations?

The answers to these questions affect the design of the documentation. They also affect some or all of the following:

- schedules;
- costs;
- user tests;
- writing style;
- presentation.

For products to be made available in other countries using the same language as the source country, consider whether special versions of the documentation should be prepared for those countries, using appropriate local language, dialects, or conventions. Consider cultural issues, which need to be taken into account, both in the main product and in the documentation, especially in examples. A national of the target country should check all the user documentation to ensure it is suitable for use in that country.

4.2.9 Translation requirements

Make a list of the human languages that are required. Consider the following.

- For some products that are to be exported, there might be a legal requirement for the documentation to be translated.
- Avoid translating documents until they are complete and have been approved.

NOTE 1 To give the translators more time to complete their work, the translation process could begin when the documents are almost complete. When complete, any subsequent changes to the documentation are identified electronically and then translated.

- Arrange for all translations to be made by native speakers of the target language, not by native speakers of the source language.

NOTE 2 Be aware of variations in the target language: for instance Spanish versus Mexican Spanish, and UK English versus US English.

- After translation, check that the main product and its documentation are consistent.

As a first step in the translation process, translate the list of terms and their definitions that has been prepared as described in 9.10. Do not translate the documentation until the translated list of terms has been approved. This is especially important when more than one translator is involved for a language.

If inaccuracies or ambiguities in the source text are found during the translation process, remove them at the earliest opportunity. Clarity and simplicity are particularly important in documents that will be translated. Avoid the vernacular, jargon, humour, idioms and metaphors that might not translate correctly. Use similes and analogies with care.

NOTE 3 Annex E of ISO/IEC 15910 provides further guidance to authors writing in English for translation.

4.2.10 Packaging requirements

The packaging requirements for the entire product affect the documentation design. Consider at least the following:

- whether the main 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 the product;
- whether the packaging should be the same as used for previous versions or for similar products.

NOTE Further guidance on the minimum information recommended for inclusion on software packaging is in ISO/IEC 9127.

4.2.11 Legal requirements

A product may have to satisfy legal requirements. 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;
- data protection;
- acknowledgements;
- trademarks;
- escrow conditions;

- licensing;
- what presentation requirements apply (for example, proprietary platform-compatible products are typically identified by special logotypes on the packaging and in the documentation);
- intellectual property rights;
- professional bodies;
- warranties and guarantees.

4.2.12 Security

Specify security requirements, including:

- the prevention of copying of the software and its documentation;
- the protection of sensitive information.

Determine whether it is necessary to check the integrity of on-screen documentation to allow for deliberate or accidental changes to its content made by the user.

4.2.13 Standards and conventions

Maintain a list of the standards the product has to follow. Consider:

- International Standards (e.g. 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.

List local rules and house styles if they exist. Check whether any local, company or product standards are suitable by comparing them with the recommendations in this International Standard. If they do not exist, arrange that they be established consistent with this International Standard.

On-screen documentation should retain the 'look and feel' of one of the following:

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

ISO 9241 defines the ergonomic requirements for office work with visual display terminals (VDTs). Apply all relevant parts.

4.2.14 Cost constraints

Allow for the documentation costs, including the following:

- design and development costs;
- production costs;
- maintenance and updating costs.

NOTE 1 If documentation design is not taken into account in the design of the main 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 localised 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 localised 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.

Obtain 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. Recommendations for estimating costs are given in 5.2.7.

4.2.15 Documentation delivery and viewing mechanisms

If the new product is part of a suite of products, consider the use of any tools already specified for developing, maintaining, delivering and viewing documentation for that suite.

If the customer or organisation for which the product is being developed has existing systems with which they want the new product to be consistent, consider how this may dictate the use of particular documentation delivery and viewing mechanisms.

4.2.16 Quality management

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

4.2.17 Provision of technical information

At the start of the project, the project manager decides who is to be responsible for supplying the author with technical information about the product.

Together with the author, those responsible should specify the methods to be used to supply information, including:

- consultations with technical contacts;
- written specifications;
- the product itself (possibly development or prototype versions).

4.2.18 Approval authorities

At the start of the project, appoint the product authority and specify who is responsible for the following:

- technical accuracy of the documentation;
- usability of the documentation with the main product and vice versa;
- suitability of the documentation for the market;
- editorial quality of the documentation, including both text and illustrations;
- legal issues;
- production of the documentation, including tools support;
- translation;
- packaging and shipping of the software and the documentation;
- final approval of all the documentation.

4.2.19 Configuration management

4.2.19.1 Application of configuration management

All the documentation development activities recommended in this International Standard should be carried out under the control of the configuration management system being used for the product development. Users of this International Standard are recommended to operate a configuration management system, as addressed by ISO/IEC TR 15846.

4.2.19.2 Configuration control

Set change control procedures for the project in accordance with the configuration management system in use. Take account of the requirements of the documentation activities as follows.

- Documentation can be greatly affected by changes to the design of the software or to the training or support plans. Therefore the implications of such changes should be assessed before approval is given.
- When product changes are made, immediately inform authors, so that they work only with up-to-date information.

NOTE Small changes to a system can cause major changes to documentation, whilst 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 need no changes to the documentation.

Authors should be members of any body responsible for approving changes.

4.2.20 Availability of resources

Establish the availability of resources needed for developing documentation at the start of the project. Consider:

- hardware and software for using prototype or other versions of the software;

- laboratory space;
- prototype or other versions of the software to be used by authors to obtain information about the software being documented;
- technical contacts who will supply information, discuss technical details with authors, answer technical questions, and check drafts of the documentation;
- graphic designers and illustrators;
- printers and packaging vendors;
- authors or other staff who will editorially review drafts of the documentation;
- hardware and software for producing the documentation itself;
- usability staff;
- testing to ensure the documentation matches the installation and operational use of the software;
- legal reviewers or contacts;
- translation.

4.3 Documentation Proposal

The Documentation Proposal should record the decisions made during the objectives phase. Keep a record of the source of each requirement, so that it can be tracked back to its origins and its validity can be reaffirmed.

The subjects described in 4.2 can be used as a starting point for the contents list for the Documentation Proposal. Figure 2 shows an example contents list for a Documentation Proposal.

NOTE Not all of these sections will be used in all proposals. The sections used will depend upon the needs of the particular project.

Documentation Proposal

Contents

1 Introduction

2 Overview of the product

3 Objectives *(This section sets out the requirements specified for the project, and gives the source of each requirements statement)*

Target audiences

Usability

Schedule

Cost

Internationalisation and localisation

Accessibility

Customisation

Future modifications

Legal requirements

Security requirements

International Standards and conventions to be followed

Documentation delivery and viewing constraints

Safety

Document maintenance

4 Investigation *(This section outlines the investigation work that has been done to establish the proposals in section 5)*

History of documentation development schedules

History of documentation development costs

History of the use of documentation development tools

Evaluation of existing documentation

Evaluations of localised and/or internationalised documentation

Customer or user feedback from previous versions

New product features

Possible solutions

5 Proposal *(This section outlines the decisions that can be made about the documentation based upon the constraints arising from section 3)*

Preliminary schedule

Development of base documentation

Localisation and customisation of separate documentation

Localisation and customisation of embedded documentation

Costs

International Standards

Dependencies

Assumptions

Prerequisites

Resources

Risks
6 Future modifications
Software
Documentation

Figure 2 — Example contents list for a Documentation Proposal

5 The planning phase

5.1 General

This clause describes how to plan and control documentation projects. Most of the principles outlined here are standard IT project management practice.

Although Planning is shown here before Analysis to be consistent with ISO/IEC 12207, analysis and design may have to be completed before planning is done, particularly in small projects.

The audiences for the document plans and documentation suite plan are:

- the people doing the analysis and planning;
- the product authority;
- the project managers who will be looking after the document development;
- customers who may need to know what the documentation will look like when it is complete;
- writers, editors, artists, software developers, and others involved in product development;
- the people doing the translation, production and usability testing;
- printers and other external suppliers.

Record the results of the planning process in:

- a set of documentation plans, one for each document;
- if necessary, a documentation suite plan, which brings together all of the individual document plans.

NOTE ISO/IEC 15910 refers to a 'Documentation plan', which incorporates both of the above. ISO/IEC 15910 is intended as a contractual document, and specifies the content of a documentation plan in such a way that compliance can be determined in a court of law.

5.2 Documentation plan

5.2.1 General

The documentation plan should include (or reference) all of the material developed in accordance with Clause 6.

New material that should be developed for each documentation plan (or referenced, if it is to be common across multiple documentation plans) is described in 5.2.2 to 5.2.15.

5.2.2 Standards

Identify standards that need to apply to the document, including house standards, documentation suite standards, national and international laws and standards.

5.2.3 Version control and change control

Wherever possible, use the same configuration management system as used for the software development. This will usually have some features appropriate for the documentation, particularly the on-screen elements. The essential version control and change control aspects are:

- the level at which version control is to apply, for example, whether it applies to a single file of information or to all the information for a particular module;
- the landmarks within the project at which new versions are to be created, for example, after each set of tests;
- how changes to each version are to be controlled;
- how version control is to be applied to localised and customised versions;
- the back-up and archive procedures;
- how records are kept about the history of each version, so old versions can be recreated if necessary;
- at what stage, if any, the documentation becomes part of the software for version control purposes.

5.2.4 Personnel

The personnel resources for the project should include:

- experienced authors for developing user documentation (rather than using software designers or software engineers to do the work);
- 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;
- usability specialists for conducting usability tests and advising on matters concerning human abilities, limitations and accessibility.

Make plans to provide people to carry out those of the following tasks that are relevant; the number of people required for each activity will depend on the scope of the work needed in each area:

- to write the text;
- to draw the illustrations;
- to edit the text;
- to build the on-screen documentation systems;
- to provide specialised accessibility features (see 4.2.6);
- to test the on-screen documentation at each planned testing stage;
- to carry out reviews at each technical review stage;

- to run user tests at each planned trial;
- to be the users in user tests at each planned trial stage;
- to provide legal review;
- to translate the documentation;
- to prepare the localisation instructions and manage the localisation;
- to prepare the customisation instructions and manage the customisation;
- to prepare and distribute material to the printers (production staff);
- to coordinate the packaging and shipping of the documentation and the product.

5.2.5 Equipment

Make plans to provide equipment and software tools for:

- using the software, so authors can 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.

5.2.6 Responsibilities

Identify people who are to take management responsibility for the following, as necessary:

- technical specification of the documentation;
- interface design for the documentation;
- provision of technical information to authors;
- technical accuracy of the documentation;
- correct operation of the system features of on-screen documentation;
- style and quality of illustrations;
- editorial quality of the documentation;
- tools support;
- training for authoring tools;
- usability of the documentation;

- suitability of the documentation for the market;
- legal issues concerning the documentation;
- approval of completed documentation as ready for localisation and customisation;
- translation of the documentation;
- approval of the translated versions of the documentation;
- approval of localised and customised versions of the documentation;
- final approval of all the documentation.

5.2.7 Cost estimates

Estimate the cost of developing the proposed documentation by including where relevant:

- author costs for development and for reviewing drafts;
- technical staff costs for briefing authors and reviewing drafts;
- illustration costs;
- typography costs;
- project management costs;
- editorial costs;
- tools support costs;
- training costs;
- testing and validation costs;
- expenses for other staff;
- equipment costs;
- materials costs;
- bulk copying and production costs, for example, copying and packaging disks;
- costs of system tests, reviews and user tests;
- evaluation costs;
- maintenance costs;
- costs of delivery to customers or users;
- translation costs for each language, which may include fees, travel and communications.

NOTE Annex F of ISO/IEC 15910 describes how to estimate the time needed for development tasks.

When estimating the cost of producing copies of the printed documentation, consider the following:

- the number of pages in the document;
- the size of the pages;
- the use of colour;
- the preparation of special illustrations;
- the number of copies required of each document;
- printing costs;
- binding and packaging costs, including materials;
- distribution costs.

5.2.8 Schedules

Prepare a separate schedule for developing each document, in step with the schedule for other parts of the product.

NOTE Consider scheduling awareness and appreciation documents to be prepared last, so all the knowledge of the software and the users amassed during the preparation of the other documents can be applied.

For each activity, particularly system tests, reviews and user tests, allow time for the results of the activities to be incorporated. Plan for:

- creating a new version of the documentation to correct errors found in the documentation itself;
- keeping the documentation matched to the software:
 - updating the documentation to take account of changes to the software following software testing;
 - re-testing the changed documentation and also handling the results of those tests.

Allow time for:

- learning how to use any new tools;
- becoming familiar with the software (preferably by using it);
- software developers to provide information to the authors and to answer questions;
- legal reviews.

If schedules need to be adjusted during the project, review the plan; do not remove activities such as system tests, reviews and usability tests from the schedule to save time.

5.2.9 Prototypes and drafts

Validate documentation designs and the development of prototypes or drafts of each type of document, to test that the documentation does provide users with what they need within the usability objectives. Test prototypes for all combinations of equipment that will be used for the finished application.

Plan:

- the stages at which prototypes will be prepared;
- for each prototype:
 - its purpose: what information is intended to be collected based on testing or trials of the prototype;
 - its scope: what part of the application will be covered and what types of documentation will be included;
- how the tests and trials will be carried out;
- how the results of the tests and trials will be recorded;
- how the results will be used in subsequent phases.

5.2.10 System tests

Include tests of both embedded and separate documentation in system testing. Verify that:

- the correct information is displayed in each given situation;
- the links and cross-references work correctly;
- the instructions in the documentation have the desired effect when carried out.

Plan:

- provision of the resources required for testing;
- timings and schedules for the tests;
- preparation of test scripts;
- methods for classifying the results of the tests;
- methods to be used for the tests;
- methods for recording the results of the tests;
- methods for determining whether the tests succeed or fail;
- methods and plans for incorporating the results of the tests into the next version.

5.2.11 Reviews

The review activity includes technical reviews, reviews against standards and reviews of the accuracy and understandability of the text, and focuses on the content of each element of the documentation.

Plan:

- provision of the resources for reviews;
- timings and schedules for the reviews;
- provision of versions of the documentation to be reviewed;

- level of detail (as a percentage of the complete documentation) to be reviewed at each draft;
- methods for conducting the reviews;
- methods for recording the results of the reviews;
- methods for classifying the results of the reviews;
- methods and plans for incorporating the results of the reviews into the next phase of the project.

5.2.12 Usability testing

Plan to carry out user tests for the different audiences at predefined stages in the project.

NOTE User tests are the most acceptable method of checking that the information provided in the documentation is what users need, and that users can find it, understand it and apply it.

Plan:

- provision of the resources for user tests;
- schedules for the trials;
- methods for conducting the trials;
- methods for recording the results of the trials;
- methods for determining whether the objectives have been met;
- methods and plans for incorporating the results of the trials into the future design and development activities for the project.

5.2.13 Localisation and customisation

Specify:

- who is responsible for the process;
- who will prepare the instructions for localisation or customisation and when;
- where the localisation or customisation will be carried out, and by whom;
- when and how the product will be delivered to the staff who will carry out the work;
- how the localised and customised versions will be tested;
- when and how the localised or customised versions will be delivered;
- who is responsible for the quality of the localised or customised versions;
- who is responsible for the usability of the localised or customised versions.

5.2.14 Approval

Plan the mechanism that will be used to approve finished documentation. Specify:

- who has authority to approve and sign off the documentation;

- what conditions must be satisfied before sign off:
 - the 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?
 - if the product is to be translated, that the localisation instructions have been prepared and approved?
 - legal approval has been obtained?
- how the documentation will be signed off.

5.2.15 Maintenance, updating and future developments

Integrate plans for modifications to the on-screen and printed documentation with the plans for modifications to the software. If the on-screen documentation is being provided separately from the software, produce plans for issuing new versions in the same way as for printed documentation.

The project requirements and constraints for on-screen and printed documentation should specify whether a product is to be modified or updated and if so, how this should be done.

For printed documentation, consider estimates of time and costs for both complete replacement and amended pages, taking into account the authoring time and printing costs.

If a completely new version of the product, including the printed documentation, can be bought or is distributed each time the product is updated, there should be no need to issue changed pages to the users.

NOTE Where alterations to the product are issued, it is more convenient for users if complete, printed documents are reissued: replacement pages can be lost and do not always reach all holders of a document, so there is a risk users will be working with out-of-date documents. Only large documents with few users should be updated by issuing replacement pages. Wherever possible, documents should be designed so that, when a new version of the product is issued, new versions of the documentation are issued as well.

5.3 Review of detailed documentation plans

Check the detailed documentation plans thoroughly.

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.

Approve the documentation plans with the rest of the product and place the plans under change control procedures with the rest of the project plans. Reviewer responsibilities are listed in 5.2.6.

6 The analysis and design phase

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

6.1 Audiences

6.1.1 Audience analysis

Prepare a list of all the intended types of user of the product. Classify users into audiences according to:

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

Each audience consists of a group of users whose background, experience and use of the application is similar.

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

Use both a 'bottom-up, and a top-down' approach to check that all user types have been considered.

- Bottom-up — Consider who will use the product and determine all the types of user. For some applications, the audiences can 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 — Consider the whole organisation or the total functionality of the application and break this down until a set of audiences or roles is reached. This set of audiences can 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.

Consider grouping audiences into a hierarchy, so individual documents can be targeted at several audiences. Use the hierarchy to group together audiences who will have the same type of interaction with the software; do not (necessarily) reproduce the user organisation's organisation chart. Figure 4 is an example of such a hierarchy.

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 fulfilment system

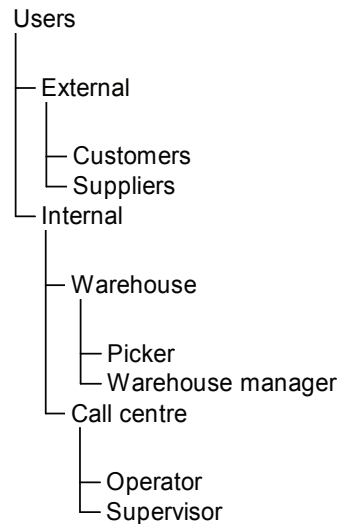


Figure 4 — Example of an audience hierarchy

6.1.2 Learning stages and frequency of use

Sometimes it is necessary to split audiences of otherwise similar users according to how long they have been using the application, or how often they use it. For example:

- learning to use (tutorial);
- using occasionally or infrequently (and so may need reminders);
- using (or normal use);
- exploiting advanced features.

If the product being developed is a consumer software package, it may not be possible to gather any useful information about typical users' jobs and background, for example, because their jobs and skills are very varied. In this case, concentrate on the tasks that the user 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, define the least experienced of the potential users and prepare the documentation for users at that level.

6.1.3 Working environments

Collect details of users' working environments. These details are one factor in deciding the most convenient media for presenting information to them; there are other factors.

Collect as much information as possible about factors that will influence decisions about the types of printed document to provide. 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?

6.1.4 Audience profiles

For each type of user, draw up an Audience Profile that records all of the information that has been gathered. This profile will be useful both in planning and as a guide for writers and illustrators.

Record this information either in absolute terms, for example special qualifications of one audience, or as how users' experience varies from the norm.

Figure 5 illustrates the types of information that might be included in an audience profile.

Audience:	Ticket Agent
Background:	<p>Ticket agents have knowledge of the travel industry and of customers' needs and concerns.</p> <p>They may not have any previous knowledge of computer booking systems.</p>
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:	<p>All ticket agents will have attended a one-day training course, so no users are novices.</p> <p>There may be at least one expert user in each office all the time.</p>
Frequency of use:	Ticket agents may use the application all the time throughout an eight-hour shift (with breaks).
Working environment:	<p>Offices can be noisy and very busy.</p> <p>Ticket agents sit at desks to work. Agents may have access to a computer to use throughout the shift. In some offices, agents share printers.</p> <p>Offices may have a photocopier, fax machine and scanner.</p> <p>Ticket agents may use headsets for the telephone.</p> <p>Not all offices will have:</p> <ul style="list-style-type: none"> • Shelf space to hold a small library of books. • Drawer space in the desks. • Wall space available for displaying wall charts.

Figure 5 — Example of the types of information to include in an audience profile for a ticket agent

6.2 Tasks

6.2.1 Task analysis

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 consider the tasks, or use other sources such as use-case design documents.

Figure 6 shows an example task list for an electronic mail system.

It might be useful to group related tasks, tasks that are similar, or tasks that involve similar steps, in a hierarchy as illustrated in Figure 7.

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 — Example task list for an electronic mail system

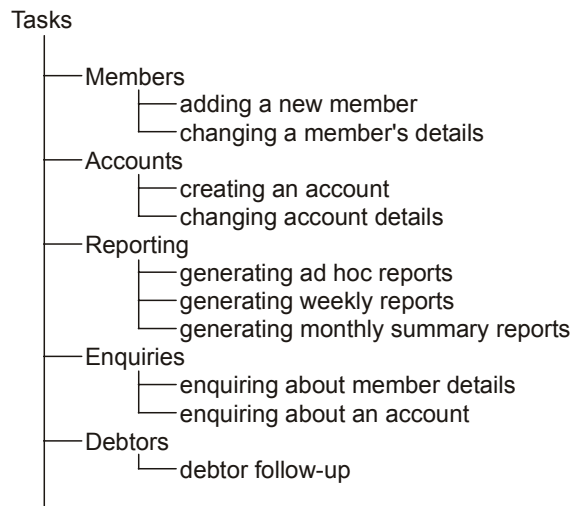


Figure 7 — Task hierarchy

6.2.2 Mapping audiences to tasks

Record details of which audiences carry out which tasks. 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.

Table 1 — Audience mapping matrix

Audience	DIC	SUP	MAN	AUD	ACC	FC	SO	FA	SA
Task									
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	Function allocator		
SA	System administrator		

6.2.3 Task characteristics

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);
- the ways in which 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 whilst a customer is present;
- what discretion the user has in how or when the task is carried out;
- the sequence in which the user carries out the set of tasks;
- the prerequisites for the task;
- how fault-tolerant the task is, that is, how important it is that the user carries out the task correctly;
- the consequences of the task.

6.2.4 Task profiles

Record all of the details of each task in a Task Profile.

6.3 Information

6.3.1 Information needs

Decide what information is needed for each audience, and for each task or group of tasks separately. Include only the information that users actually need to carry out the tasks, and remember that some audiences (because of their prior training or experience) already have some of the information they need.

There may be a lot of information available about the software and how it carries out its functions, but include only what is necessary for the user to perform the task.

Take the user's view of the application. If possible, consult potential users or observe users performing typical tasks.

6.3.2 Context of use

Each combination of audience and task has a context of use, which affects the way in which the information needs to be presented. 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 works 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 product will not run?

6.3.3 Volume/amount of documentation

Select a measure of how much information there will be. Use this 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 set some measure of scale.

Consider the following techniques.

- Count identifiable items, for example, icons, tasks, steps, dialogue boxes, transactions.
- 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.
- Prepare a sample for one item and use its size as a basis for calculations.
- Refer to documentation for comparable users, tasks and applications, and assess from that how much information will be needed for the current application.

6.3.4 Media

The combination of audience, task, information needs, context of use and volume, as well as considerations such as cost, maintainability, usability and marketability will determine what media are used to present information.

Do not assume that one medium will suffice for one product, or for one audience, or even for one task or context. Often a solution involves the use of multiple media. Consider media from Table 2, plus any others that might be available or invented to suit the particular project.

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

Table 2 — Advantages and disadvantages of various media types

Medium	Advantages	Disadvantages
CD ROM/DVD	Can contain many documents. Easily searched.	Requires suitable hardware.
Book	Easy to annotate pages.	If large and heavy, can be difficult to carry around. Can be expensive to produce.
Card	Suitable for a small amount of quick reference information. Information that is needed frequently can be on one side and information that is needed infrequently can be on the other side.	If displayed on a wall, only one side is visible.
Wall chart	Suitable for a large amount of quick reference information.	Requires a relatively large clear wall space for display.

Table 2 (continued)

Medium	Advantages	Disadvantages
Keyboard template or sticky labels for keys	Always in sight for the user.	Care is needed to ensure that users have only one such item to use at a time, taking into account the operating system and other software.
Notice to attach to the computer	Suitable for a small amount of information that is always needed at the computer.	Danger of cluttering up 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.	
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. Distribution on tape can be expensive. If distributed as a tape, needs appropriate hardware for viewing If distributed as a file, needs audio equipment on the computer.
Web site	Updateable without contacting users. Can be launched automatically from the application.	Users need web access.

6.3.5 Information profile

Record all the source data and design decisions for each audience/task combination in an Information Profile.

Figure 8 shows an example of an Information Profile.

Task: Get a quotation for a journey Audience: Ticket Agent					
<i>Information needed</i>	<i>Volume</i>	<i>Same time as the application</i>	<i>Permanent or temporary</i>	<i>Urgency</i>	<i>Medium</i>
Task instructions for getting a quotation including finding and entering departure and destination locations, entering dates, party sizes, ticket types, checking availability and finding a price	Up to 10 task steps	✓	Permanent	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

Figure 8 — Example Information Profile for one task and one audience

6.4 Usability

6.4.1 Define usability goals

The usability of the complete product is very important and is not separate from the usability of its documentation. However, the purpose of this clause is to provide guidance on usability aspects of the software documentation.

Once details of the users, their tasks and the information they need have been collected, define usability goals in detail. Keep usability goals independent of the method to be used to provide the information to users.

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

- Apply human-centred and software ergonomics principles to the design. ISO 13407 and ISO 9241 address such design advice.