ISO/IEC JTC 1/SC 7

Date: 2003-03-04

ISO/IEC FDIS 18019:2003(E)

ISO/IEC JTC 1/SC 7/WG 2

Secretariat:

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

Élément introductif — Élément central — Élément complémentaire

Document type: International Standard
Document subtype:
Document stage: (50) Approval
Document language: E

Macintosh HD2:Users:cinzia:Documents:AccessAbility SIG work:ISO-IEC_FDIS_18019_(E).doc STD Version 2.1

Contents

Page

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 18019 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information Technology*, Subcommittee SC 7, *Software and systems engineering*, Working Group 2, *Software systems information*, and is based upon British Standards BS 7649:1993 and BS 7830:1996.

Annexes A to G of this standard are for information only.

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.

Software and systems 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 International Standard the following 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. It 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

EXAMPLES 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
co-ordinated 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)

EXAMPLES 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 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 clause 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 clauses 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 clause 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 clause 6.1.2.3);
- evaluation and updating: evaluating the documentation with the rest of the product (see clause 8 and ISO/IEC 12207 clause 6.1.4);
- production: preparing the printed documentation on suitable media (see annex E, and ISO/IEC 12207 clause 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 (clause 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.

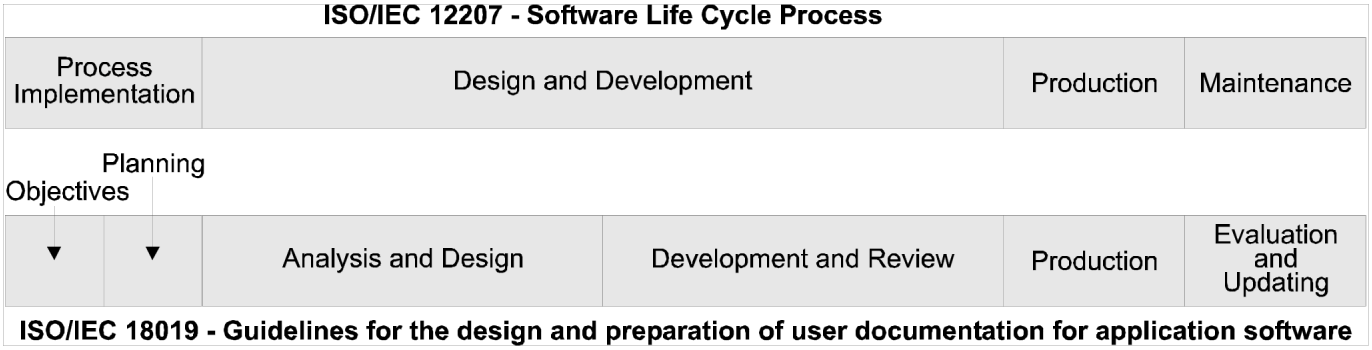


Figure 1 — Comparison of documentation processes between ISO/IEC 12207 and ISO/IEC 18019

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 6.2.7.

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 - 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

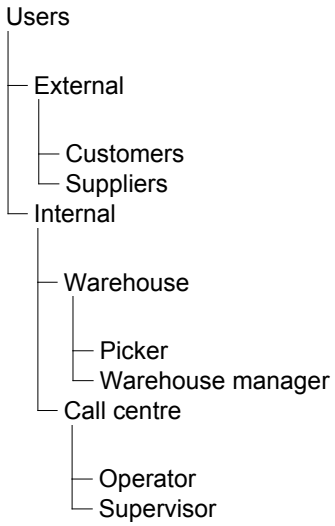


Figure 4 — Example of an audience hierarchy

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:	Ticket agents have knowledge of the travel industry and of customers' needs and concerns. They may not have any previous knowledge of 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, so no users are novices. There may be at least one expert user in each office all the time.
Frequency of use:	Ticket agents may use the application all the time throughout an eight-hour shift (with breaks).
Working environment:	Offices can be noisy and very busy. 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. Offices may have a photocopier, fax machine and scanner. Ticket agents may 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 — 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

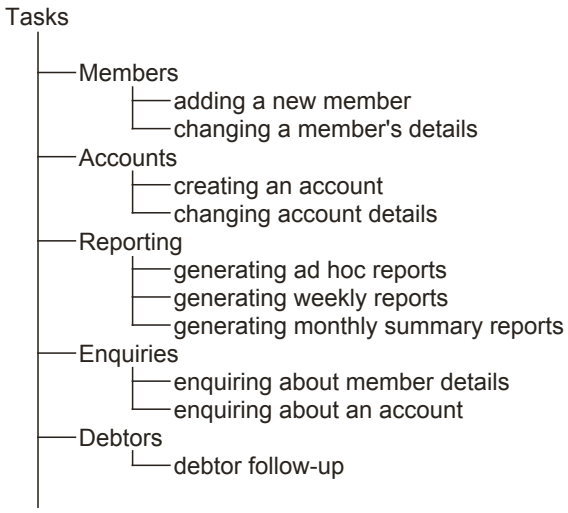


Figure 7 — Task hierarchy

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 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	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.

Medium	Advantages	Disadvantages
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.
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.

Medium	Advantages	Disadvantages
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**		
Information needed	*Volume*	*Same time as the application*	*Permanent or temporary*	*Urgency*	*Medium*
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.

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

Usability cannot be added at the end of documentation development; it must be built into the behaviour and appearance of the documentation and supporting software. Therefore, specify the documentation's usability requirements, and the method of testing them, in the analysis phase where other user needs are being determined.

The usability targets for the computer system and the documentation will affect the usability test methods to be used.

- If the assured level of usability for the whole system and the documentation is high, use quantitative tests.

- If usability of either the documentation or the software and its documentation is less critical or if resources are limited, consider qualitative methods.

Once an analysis of usability requirements has been made and the appropriate testing method selected, translate those requirements into usability goals for the documentation. The goals are the qualitative or quantitative targets which are measured in usability tests. Figure 9 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 summarise 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 a specified time, then the structure and navigation organisation and retrievability 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 at the first try while following the instructions, then the

help was acceptable, task-orientated and complete.

Figure 9 — The process of defining the usability goals of an electronic mail system

6.4.2 Record usability goals

Record the usability goals in a Usability Goals Document.

6.5 Structure of the documentation suite

6.5.1 General

First, work out what information needs to be provided and what will not be provided (see 6.5.2). Second, decide what documents are necessary (see 6.5.3).

6.5.2 Decide what information needs to be provided in the documentation

To decide what information to deliver in the documentation, examine the Information Profile (6.3.5) and details of the context of use (6.3.2) 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. Some elements may be better delivered through training, or in some cases even by defining the expected knowledge and experience of the audiences to include additional training. Some of the information may be delivered by building it into the user interface of the software, or by providing a community of practice for users to share information. Some information may already be available from other sources (user documentation for other products, tertiary education or commercial training courses).

To refine the list of information needs into a list of information to be provided, taking into account the context of use, refer to the descriptions of different sorts of information in clause 3.

6.5.3 Group information needs into documents

For guidance on deciding whether information should be delivered on the screen or on paper, see 3.2.

Take the following into account when deciding how to group information into documents.

- Information for different types of user can be in the same document provided those users carry out the same groups of tasks in the same way.

- If the information is on paper, different quantities may be needed for different documents. For example, each user may need a copy of quick reference information, but a single copy of a getting started tutorial might be shared between several users.

- Information needed by users who work in different environments can be in the same document provided that the document can be designed to suit all the different environments.

- Do not include secure information needed by some users in the same document as information for other users. For example, do not include information about administering the security of a multi-user system in the same document as information for end users. Instead, provide a reference to the location of the information.

- Do not create a proliferation of short printed documents, nor combine a lot of information into one cumbersome document. If a document becomes too large consider splitting it by:

- putting the appreciation and tutorial information in one document, and the reference information in another;

- keeping different types of comprehensive reference information separate, for example, one document for concept-related procedural information and another for task-related information.

Figure 10 shows how the audience and information needs can be grouped into documents, where multiple audiences need the same information. Figure 11 is another example of grouping information into documents, showing the sorts of information to be included in each document.

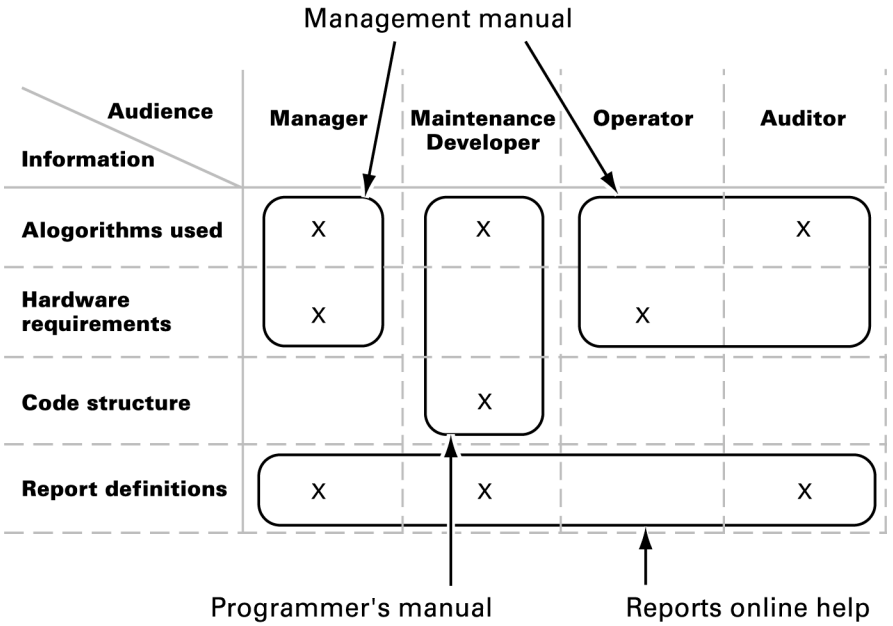


Figure 10 — Example of how the audience and information needs can be grouped into documents

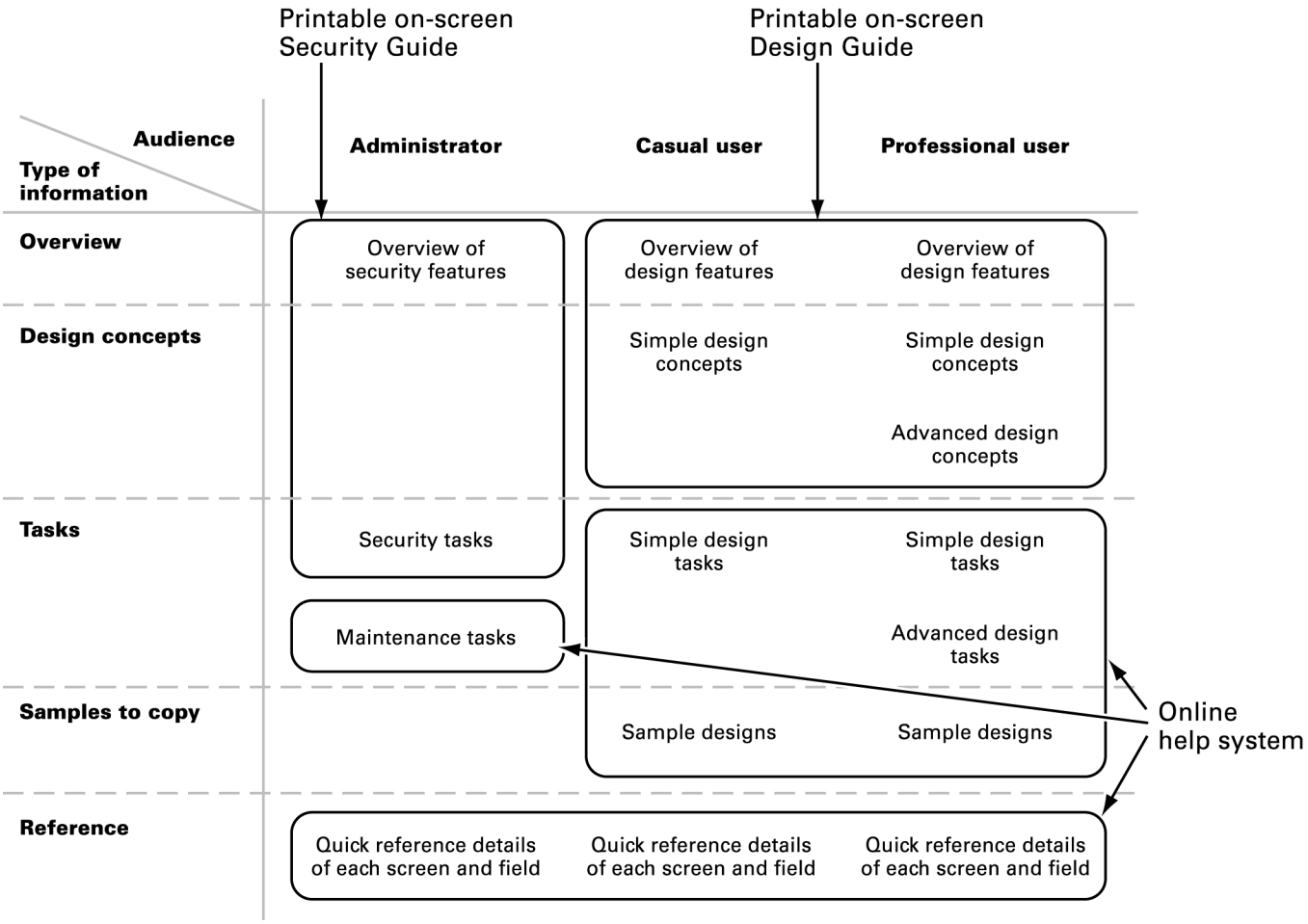


Figure 11 — Example of grouping different sorts of information into documents

6.6 Individual document structures

6.6.1 Prepare a list of contents

Define the 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, give a count of the expected number of pages or topics. Include in the list any navigation pages or topics, such as the contents list itself, indexes, glossary and bibliography.

General advice on structuring different types of information within documents is provided in 6.6.2.1, whilst 6.6.2.2 to 6.6.2.6 gives advice on structuring specific types of information.

6.6.2 Define the document structure

6.6.2.1 Main factors

The main factors for determining the structure of the information within a document are:

- the way in which the user uses the information;

- the way in which the user accesses the information;
- the structure of the software as seen at the user interface.

Select a structure for each document that is appropriate to how the document will be used. For topic-based documents, design the structure as it will be perceived by the user, for example through a contents list or an index. This structure is not necessarily the same as the structure in which the topics are held electronically.

Select the most appropriate of the following schemes for structuring information or displaying the structure to the user:

- tutorial sequence, starting with simple tasks and moving on to more complicated tasks;
- performance sequence, arranged according to the order in which tasks have to be carried out;
- type sequence, grouped by type of task;
- product sequence, grouped by parts of the product;
- alphabetical or numerical sequence, arranged by names of items or tasks.

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

If a document comprises information for different contexts of use, for example, a getting started tutorial and some normal use information:

- for sequential documents, create separate sections or a separate part of the document for each context;
- for topic-based documents, allow the user to find the correct topics for each context, for example, by using different sorts of topic titles or using a table of contents.

6.6.2.2 Awareness or appreciation information

Include one main section or topic for each major subject of interest to the audience, starting with the subjects in which they are most interested. Keep sections short. Use few levels of heading; often, one level is sufficient, but use no more than three.

Consider including references to other documents about this product or a related product that might be of interest to the same audience.

6.6.2.3 Installation instructions

List very clearly at the start of the instructions everything that users need to have available before the software is installed. Then follow the recommendations for task instructions in 6.6.2.5.

6.6.2.4 Tutorials

Divide tutorials into manageable task sections. For getting-started tutorials, if possible design each task section to take a maximum of 10 minutes for an average target user to work through.

Use two section levels: major task and sub-task.

Give general and basic information once, at the beginning of the tutorial. When this information is needed later in the tutorial, remind the user where the information was given. This approach reduces the size of the tutorial and avoids annoying those users who can remember the information.

Arrange getting-started tutorials as the user needs them, to avoid forward references. Since getting-started tutorials are intended to be used sequentially, an index is not required, because it might encourage users to dip into the tutorial, rather than follow it through in the correct sequence.

Do not repeat information that is available in quick reference form; teach users how to use the reference documentation.

6.6.2.5 Normal use task information

You may be providing several types of information for normal use, such as process descriptions, task instructions, reference information and background concept information. For each type of information for which separate topics being provided, decide what tasks, or what subjects a user might want to find when they are working with the product, and provide one topic for each of them. This results in one topic being provided for each process in which the user is interested, one for each task, one for each major concept and so on.

Allow users to find specific sections or topics by including them in the table of contents and the index and making them available through the search mechanism if there is one.

When designing the structure, consider the following for each sort of information being provided.

- Process information. Process topics may explain what tasks are possible, or may specify the sequence in which tasks should be carried out. Provide cross-references or links to the task instructions or the reference information for how to carry out each task.

- Task instructions. If tasks are long or complicated, consider splitting a single task into smaller tasks, perhaps with a process topic to show how the smaller tasks relate to each other. Include any related concept and reference information the user needs to carry out the task, or refer to it using cross-references or links.

- Concept information. Concept information explains concepts that users need to understand before selecting what task to carry out, or before and while carrying out a particular task. If separate concept sections or topics are provided, refer users to related task instructions and related reference information using cross-references or links.

- Reference information. Give cross-references or links to reference information from processes, task instructions and concept topics that need it. Reference information will not be used sequentially. Do not assume that in any section or topic the user will have seen any of the information covered in other sections or topics; use cross-references or links.

- Quick reference information. Either include quick reference information at the place where it is needed in task instructions or refer to it using cross-references or links. If possible, make quick reference information available directly from the product at the points it is needed.

6.6.2.6 Exploitation of advanced features

There are different sorts of exploitation information, such as processes to follow, concept information explaining advanced features or tasks for exploiting advanced features.

Provide separate sections, topics or groups of topics for each category of exploitation information.

Include exploitation information in the table of contents either alongside the normal use information for the same functional area or task, or in its own exploitation section.

For structuring exploitation information, follow the appropriate advice in 6.6.2.5.

6.6.2.7 Quick reference documents

The content of quick reference documents varies so widely that it is not possible to recommend a preferred structure; however, take the following points into account.

- Present the information in quick reference documents so that the user can quickly find the relevant entry when they have a problem, and hence quickly find the answer. This usually means structuring the information by the user's questions or problems, rather than by features of the application. For example, quick reference information explaining which product features or keyboard keys the user needs to use to achieve a particular result should be structured by the result the user wants to achieve, not by the product feature or the keys that they need to use.

- For documents such as cards and charts, do not include a contents list, an index, appendices, a bibliography or a glossary.

6.6.2.8 Comprehensive reference documents

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.

Divide information into topics. The topics can be either divisions of the product, such as modules or entries on a main menu, or tasks the user will want to perform.

The structure chosen depends on how the user needs to see the information. If there is no preference, use the divisions of the product.

In comprehensive reference documentation provided as hypertext documents, allow the user to choose the topic they wish to view by both the product feature to be used and by the task the user may want to carry out. Structure the documentation to support one of these views.

6.7 Document writing style

6.7.1 General

A style guide gives advice to the authors on the specific forms of writing and illustrations to be used in a particular document.

Advice about what to put into writing style guides for particular types of document is addressed in Annex F. Naturally the authors will also be given all of the audience, task, other analysis and design documentation.

Use a writing style that is appropriate to the target users and the learning stage that the document supports, and to the way the document will be used.

Make the writing style appropriate to the users of the document, but do not make any unnecessary assumptions about personal characteristics of individual users, for example national culture, gender and age.

6.7.2 Awareness and appreciation information

Keep paragraphs short.

Use short and clear section headings that motivate the user to read the text.

Avoid complicated tables, except for important reference information such as sizes or prices.

6.7.3 Installation instructions

The task of installing the software is normally carried out only once, so the user is normally installing the software for the first time. Keep the instructions simple, clear, and complete.

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

Avoid using any terms with which users might not be familiar. In particular, although terms might be explained in other user documents, do not assume that the user installing the software has read those other documents.

6.7.4 Tutorials and task instructions

If users need to understand certain concepts to carry out some tasks, describe these concepts but clearly separate them from the detailed instructions for carrying out tasks.

Present a sequence of instructions as a numbered list.

- Put into each list item the instructions for just one action, and, if it is helpful, a description of any effects the user should see.
- Include any messages that might be displayed and instructions for responding to them (see 9.9).
- Where possible, start items with a verb, instructing the user to do something such as 'press', 'type' or 'select'.

If it is helpful, consider illustrations of screens or other software displays:

- to show general features of the product's screen displays;
- to help users to check that they have reached the correct stage in a sequence of instructions.

NOTE Be cautious of providing too many illustrations. A simple task can appear long and complicated if too many illustrations are used.

If a screen illustration is used, state clearly whether it shows the display before, during or after an activity in the accompanying text.

Provide worked examples and test them to make sure they are correct. Where examples are used to illustrate a sequence of steps, make the examples consistent throughout the sequence.

If users need to understand the effect of functions of the software (for example, because they have to select them from menus), explain those functions fully.

6.7.5 Quick reference information

Use words and short phrases rather than complete sentences.

Clearly label each set of quick reference information with a relevant but short heading.

Use headings for lists and tables to give clear information about what the entries contain.

Use only good, tested examples, because when using quick reference documents, 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. Ensure that they clearly labelled as wrong or invalid.

6.7.6 Reference information

Use tables, charts and illustrations wherever appropriate.

Only use illustrations of software displays if they are required to clarify the text.

Present information in tables in the sequence in which users need to refer to it, rather than according to what the software does.

Refer to information represented in an illustration or table from the text, do not repeat it in the text. Use text to clarify or explain an illustration or a table.

NOTE Users of reference information often look at the illustrations and tables without reading the related text.

6.7.7 Diagrams

Consider using diagrams for:

- illustrating processes to show the sequence in which activities occur, including the points at which choices can be made;

- illustrating complicated concepts and showing how features relate to each other.

For each type of diagram, examine the user profiles of the target users to ensure all types of user will understand the diagram. 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 can be deduced from the user profiles of all the target users that flowcharts will be understood.

6.7.8 Graphs and charts

Use graphs and charts 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. They are particularly useful in appreciation and awareness documents.

6.7.9 Illustrations of screen displays

Use illustrations of screen displays only in printed documentation, or in on-screen training materials. To avoid confusing the user, do not use screen captures in any documentation the user might display on the screen at the same time as the application.

Scale illustrations of entire screen displays and portions of screen displays – such as menus or dialogue boxes –consistently throughout a single document. At most use two different scales: one for entire screen displays and one for screen fragments, as illustrated in Figure 12.

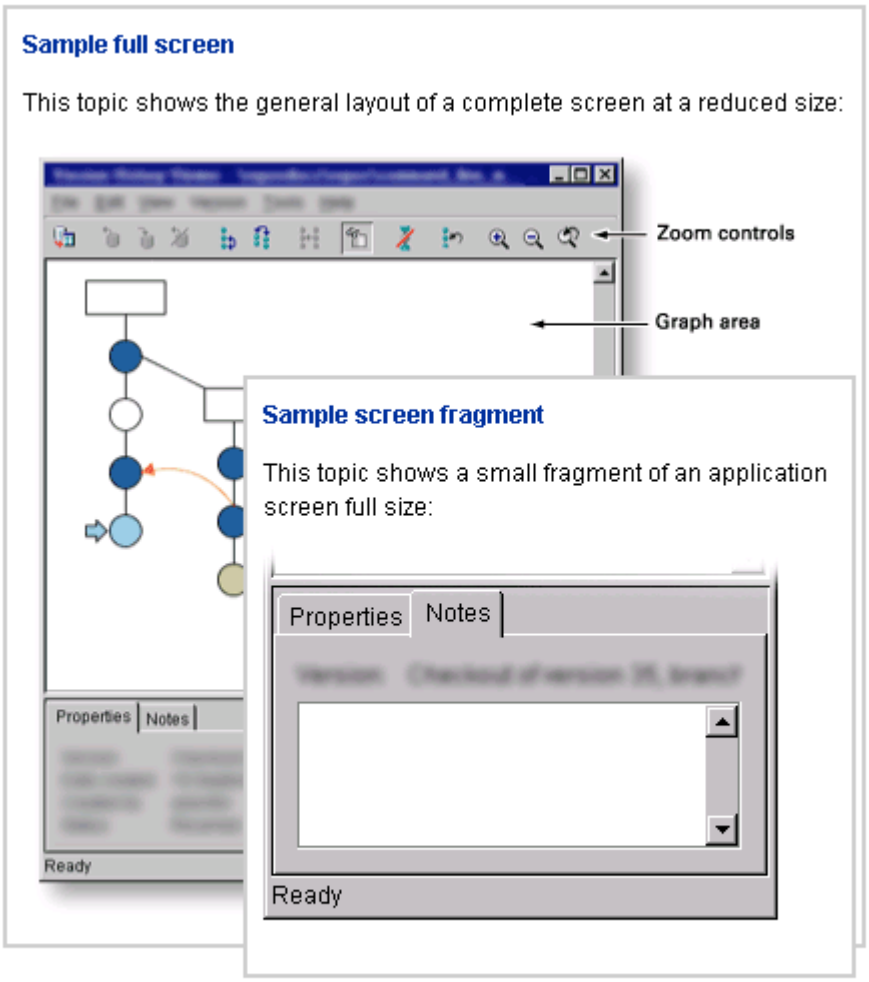


Figure 12 — Use of two scales for screen displays

6.7.10 Illustrations of printed output

When deciding whether to use illustrations of printed output (for example, lines of programming code) and what the illustrations should show, consider why the user is shown the output, in particular whether they need to read all the text on it, or simply to see what sort of text it will contain. These considerations should influence:

- how much of the output is illustrated;
- what type of data is shown;
- the size of the text, and hence the size of the illustration;
- the importance of dates in the output.

7 The development and review phase

7.1 General

The development and review process begins after the design of the documentation is finished. Its purpose is to prepare master versions of technically accurate documents as specified in the documentation plans.

Write as complete a list as is possible of the topics, in addition to all the relevant access and structural information. Implementation consists of writing the relevant texts and drawing the illustrations, and adding the necessary software elements to on-screen documentation to create a working system.

Use the checklists in Annex B to record progress for each document.

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

7.2 Prepare and issue drafts

Before drafting any documentation:

- the corresponding part of the application software should exist, or
- there should be an agreed definition of what features will exist.

For example, before preparing a skeleton set of task instructions, agree about the names of the application features that the tasks use. Before drafting task instructions, define the application features that are required.

Normally prepare three drafts as described in 7.4.

Draft information, using tools and methods specified in the documentation plans. Include only information specified in the detailed document plan. Follow the writing and illustration styles specified in the detailed document plan.

Use the software to check the tasks being documented to ensure that the information is technically accurate and contains everything users need and nothing that is irrelevant.

NOTE 1 A member of the project team may be the first person to study the working software in detail from a user's perspective. Their use of the software can provide a rigorous test of a system and give valuable feedback to the development team, for example by identifying inconsistencies.

Start developing drafts of the documentation when:

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

When developing drafts, emphasise the need for technical accuracy in the documentation. Use the following sources of technical information:

- documentation design information;
- system design information;

- programmer's information;
- personal experiences of using the system or a prototype.

If any of the information needed is not available by the date specified in the schedule:

- advise the project manager;
- review the project schedule.

Write all drafts to conform to the document design for structure and writing style (see Annex F).

NOTE 2 In planning printed documents, decide for each document separately, whether or not first drafts should be presented using the page layout to be used for the finished work. Using the page layout for the finished document helps those commenting on drafts see what the finished pages will look like, but might lead reviewers to concentrate on page layout features instead of the technical content of the documents.

NOTE 3 In planning, decide for each document whether it will be useful to provide a draft or partial index for the first draft, and for reviewers to use the draft index to find topics in the draft and review the format of the index with the rest of the draft.

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

- try out any new documentation-related methods being used in the project, for example for creating or capturing illustrations or for printing master pages;
- have a sample of the text and illustrations produced using the final production method, and check the context-linking of topics to application features, that is, make sure the context-mapping technology is working correctly and that the correct context identifiers are being used.

Clearly identify each draft with a unique issue number. Ensure that both paper and electronic master copies are held securely. Ensure that the requirements of the software configuration management system in force for the project are met.

Distribute all drafts to the respective authorities identified in the project requirements and constraints on the dates specified by the documentation schedule. Accompany all drafts with:

- clear review criteria;
- instructions for how to provide comments and for using review tools;
- an indication of the time to be spent reviewing the material;
- instructions concerning the return of comments to a specified person, by a specified date.

Commission designs and artwork for covers and packaging from illustrators at this stage.

7.3 Check and review drafts

7.3.1 General

Before it is time for drafts to be issued and reviewed, remind all those who will be commenting and authorising changes to drafts that time has to be allocated for this work. Review drafts for at least the following.

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

- **Completeness.** Check each draft to ensure that it contains all the information that users need. If possible, someone who is not familiar with the product should carry out the checks.

- **Ease of understanding.** Check each draft to ensure that users will be able to understand it. If possible, someone who is not familiar with the product should carry out the checks. The person checking drafts should be as imaginative as possible about potential misreadings and misunderstandings and should highlight them.

- **Conformance and consistency.** An author, independent of the project, should check that drafts of all the documents follow the design plans and are consistent with each other in appearance and terminology.

- **Editorial consistency and correctness.** An author, independent of the project but who might be the same one as for conformance and consistency, should check each draft for at least spelling, grammar, punctuation, editorial standards and typography. If a prototype of the software exists, test draft documentation with that prototype, to ensure that the documentation and the software can be used together and are consistent.

- **Legal accuracy.** Check each draft to ensure that the correct legal notices have been included and trademarks have been handled correctly. If possible, an Intellectual Property Law (IPL) attorney should perform this check.

7.3.2 Reviewing the information

7.3.2.1 Purpose

NOTE In this sub-clause, the concept of 'review' applies to both embedded and separate documentation. In some paragraphs, reviews of embedded documentation are referred to as testing because this better describes the act of quality control for such documents.

Review the information to check that each individual item of text or document provides accurate information to the user in a suitable form. Check also that high level descriptions offer the user the chance to find out more, for example:

- a lower level of detail on the same topic (such as explanations of individual fields or values);

- a higher level of detail (such as an overview of this type of function or display);

- procedural information (how to ...);

- other relevant topics.

Review the total set of documentation, printed and on-screen, 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 can be checked for consistency and completeness.

7.3.2.2 Responsibilities

There should be at least three reviewers:

- someone with a detailed technical knowledge of the software, to check the accuracy of the information;
- an author or an editor, preferably one who has not been involved in drafting the information, to check the ease of understanding of the information and the editorial consistency and correctness;
- someone with a legal knowledge to check the legal accuracy.

If possible, a panel of users should also be asked to review the documentation.

7.3.2.3 Methods for reviews

Deliver the documentation to the nominated reviewers at the stages specified in the project plans. Different methods of reviewing information are appropriate for checking different features, consequently different methods of providing the information to reviewers will be required. The following are suitable methods for reviewers to use; generally, a combination of methods should be selected for each type of information.

- Reviewing printed copies of text and illustrations. Use this method for reviewing the accuracy and level of content of information.

NOTE 1 This is a convenient way of checking large amounts of information, such as descriptions of concepts, descriptions of application features or task instructions.

- Reviewing on the screen, together with the software. Use this method to check:
  - the access methods and navigation features;
  - that the right information is displayed at the right times;
  - that the amount of information per section is appropriate;
  - that the level of information is appropriate for the context.

NOTE 2 This method is essential for reviewing the style and presentation.

- Reviewing on the screen independently from the software. Use this method to review:
  - information systems that can operate independently of the software;
  - information consisting of sets of items through which users can navigate to find what they need;
  - indexes, glossaries and searches provided to help users find what they need.

7.3.2.4 Results of reviews

Record the results of reviews and use these results as input to the next cycle.

- Verify any changes made as a result of the first set of checks.
- Verify that any changes to the application as a result of its own reviews and tests are reflected in the documentation system.
- Ensure that there are no unexpected results from other changes (final check).

7.3.3 Usability tests

Carry out usability tests to verify that the usability goals for the documentation have been met. Use methods appropriate to the level of assurance required by the client or customer for the product, including its documentation. The base level of assurance is that the documentation content, navigation, style and presentation conform to the general principles of software ergonomics.

NOTE A suitable test method might be to check that the design and features comply with the guidance in this International Standard, in the relevant parts of ISO 9241 or an equivalent standard. 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.

This level of test should be carried out for all systems.

As an intermediate level of assurance of usability, arrange an assessment, by a usability specialist, of 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.

For a more accurate qualitative assessment, involve typical users working with the system and making subjective assessments of whether the system meets its qualitative usability goals. The assessment of the system may be made by interview, questionnaire or in a group discussion. This level of assurance is adequate unless there is a specific contractual requirement to meet an agreed level of usability. See also 5.4.1.

The highest level of assurance requires a numerical-based 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 is used to assess 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 product. The resulting performance levels are compared with the required levels. This measure of assurance is necessary to guarantee the usability of the documentation.

There is a review of usability evaluation methods and details of the methods used to generate user feedback in Annex C.

7.3.4 System tests

Test the embedded documentation in the same way as the rest of the product. Check the software elements of the embedded documentation. Is the correct text item displayed in each given situation? Do links work correctly?

During the implementation phase, test all the following features of the embedded documentation:

- interface design;
- system design, to test all the routes through the documentation system;
- information design, to test the way the information is divided into independent topics;
- the complete application, including the on-screen documentation;
- the complete application, including the embedded and separate documentation together.

NOTE Tests should be rigorous and systematic, but the degree of completeness should be determined by the level of assurance required. See 5.4.1 for guidance on defining usability goals. For details of qualities that need to be tested, see C.4.

7.3.5 Validation and field trials

Include the documentation in the validation phase of the project. Validate the documentation and the rest of the product together. Use the documentation during the validation. In particular, validate all the examples in the documentation.

During validation, test the documentation to ensure that it meets the usability objectives set for it. If possible, the validators should be unfamiliar with the software.

Include the documentation in the field trials with the rest of the product.

In both validation exercises and field trials, identify problems with the software and its documentation together, and seek solutions to any 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 go through another design phase as part of this process.

7.4 Prepare subsequent drafts

Formally record all requests for changes to drafts and communicate them to all project team members.

Supply comments:

- as mark-up of printed copies;
- in a clearly annotated electronic form. The corrected electronic version should have changes highlighted to enable authors to identify them;
- as a separate electronic review document.

Reviewers should label their comments with their names, in case the author needs to question them, and should keep a copy.

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

Three drafts of each document should be developed.

- Check the first draft thoroughly for all the features listed in 7.3.1.
- Check the second draft for technical accuracy, particularly if the software design has changed since the first draft. Make sure that the comments made on the first draft have been incorporated correctly.
- Check the final draft only for technical or typographical errors before final approval by the appropriate authority.

These drafts may not all be sent to the same set of reviewers. It might be appropriate to send one draft to technical reviewers and other drafts for editorial or peer review.

Partial documents can also be reviewed. For a long document, the author may wish to send a new chapter to a developer for review before the whole document is finished. However, the author may still need to send the entire document for review.

Consider assisting reviewers by highlighting technical changes made in the second or final draft, for example by vertical lines in the margin, to avoid the need to re-read unchanged text. However, use this technique only if all the sections that need to be checked are highlighted, because reviewers will only read the highlighted sections.

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

Handle any changes needed to the documentation as a result of changes in the system design using the formal change control procedure for the entire product, not simply by developing and issuing extra drafts of the documentation.

Prepare plans for possible translation requirements.

When the draft illustrations have been approved, prepare and check final artwork for the illustrations.

When the final draft has been approved, prepare the final indexes and table of contents.

7.5 Prepare document masters

The activities required to build the information systems depend on what types of systems are being built and the degree to which the different types of information are integrated with the software.

Follow the schedules and methods from the implementation plans for the product, as agreed by the rest of the development team, to bring together the software and the documentation elements that have to be integrated. These plans should determine the technical procedures for creating the information, which can be integrated with the rest of the product.

Before preparing document masters, editorial staff, or a different author from the one who wrote the documents, should:

- carry out a final check of spelling, grammar and punctuation using an appropriate dictionary;
- carry out a final check of cross-references;
- review the final versions of illustrations.;
- check that the illustrations are placed correctly (see E.2.3.7) and are clear enough for the chosen method of presentation (on the screen or in printed form);
- proofread the text. Although automatic spelling checkers and other authoring tools may be helpful, always make visual checks;
- if the document is to be printed:
    - decide final pagination, making sure that major sections start on new pages, and that lists or short sections are not split over page breaks;
    - check also that:
        - all the pages are present;
        - all the pages have the correct page numbers;
        - all the headings, tables and figures have the correct numbers.

7.6 Hand over the finished documentation

Hand over the documentation for the next phase, such as a final build of the product or reproduction of free-standing documentation systems, according to the plans agreed at the start of the project.

7.7 Localisation and customisation

If the product is to be translated, instruct the translators, specifying details of what deliverables are required and where and how they are to be delivered.

Follow plans for reviewing, testing and trialing the localised and customised documentation, in the same way as for the base version.

7.8 Archiving

When the documentation is complete, archive sufficient information about it to enable new versions of the documentation to be prepared.

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.

8 The evaluation and updating phase

8.1 General

Use the evaluation and updating process to evaluate the documentation with the rest of the product, so plans can be made for future issues.

8.2 Evaluate the documentation

NOTE Annex C addresses evaluation.

Make plans to obtain feedback from users about the documentation as a part of the overall product. Consider using the documentation to contribute to this process, for example, by including letters to users asking for comment on the documentation and the software and giving addresses where the comments are to be sent. Consider the following to provide feedback about a product and its documentation:

- customer surveys;
- visits to users;
- usability tests;
- reports from support desks and consultants;
- product review magazines and reports;
- training;
- the product's sales team.

NOTE ISO/IEC 14598 addresses software product evaluation

8.3 Update the documentation

Modifications result from:

- the creation of a new or upgraded version of the product. This constitutes a new project;
- the discovery of errors in the existing product or documentation. Consider these errors when preparing a revised version of the product.

If a new version of the product is being created, manage the development of the new version as a completely new project, which should follow all the processes described in this International Standard.

When providing a new version, take into account and incorporate comments from user feedback.

NOTE Much of the work done for previous versions may be found to be relevant, making the processes simpler than they would be for an entirely new product.

9 Guidelines for the design of documentation

9.1 Introduction

Most systems will require only some of the types of information listed in this clause. These will have been identified in the design and analysis phases.

For each type that is provided, ensure that the information:

- is self-contained;
- gives sufficient coverage of the subject, but does not contain unnecessary detail.

If there is a large amount of text that the users can scroll through, structure it so that they can 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 E.

9.2 Product copyright and version details

Take legal advice on what copyright and version details about the application have to be included. This may differ from country to country, or between regions within a country.

If there is any vital information users need to be aware of, ensure that all users see it, for example, by displaying it either the first time or every time the application is started.

Consider the following information for inclusion:

- identification of the software, including the name, operating system, edition, version, language supported and dates;
- date of issue;
- product identification number;
- identification of the documentation, including the title and reference number;

- name of the manufacturer or supplier of the product, with postal address, telephone and fax numbers, email address and URL;
- the name of the publisher of the document, if different from name of the manufacturer or supplier of the product, with postal address, telephone and fax numbers, email address and URL;
- authorship (including, for example, author's name, position and qualifications);
- contact details;
- user licence, number and name of the licence holder;
- copyright notice;
- conditions and terms of the warranty or guarantee;
- the manufacturer's legal responsibilities and the consumer's rights, including training and related assistance, software support, quality assurance and availability of source code;
- reference to any standards followed for the software, indicating the degree or level of conformance;
- acknowledgements.

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

If the product is to be translated, 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.

For printed documents, consider also the following information for inclusion:

- if the document comprises more that one part, a list of all the parts;
- country where printed;
- International Standard Book Number (ISBN).

9.3 Overview of the documentation

Give users an overview of the documentation for the product. Explain what information is provided on the screen and what forms the information takes. Answer questions such as 'How can I get help?' and 'What information is there available about this application?'

Include in the structure of the document:

- a table of contents and (excluding exceptional circumstances) an index for all documents, except quick reference documents such as cards. Follow the recommendations in 9.15.3.2 and 9.15.3.3;
- a glossary if it will be helpful;
- a list of all the documents in the set;
- a list of other documents to which reference is made;

- a bibliography if it will be helpful.

Provide details of where to find any information users might need that is not provided on the screen.

If some documentation is being provided as sets of related items that users can navigate around:

- design the documentation so it is as easy as possible to use, so that even the most inexperienced users will be able to use it immediately with minimum guidance;
- show the structure;
- show the scope of each element of the structure;
- explain how to navigate around the structure;
- if flexible or complicated facilities are provided, such as annotation, the facility to insert an electronic bookmark, or complex search mechanisms, then also provide instructions for using them (see 9.15.3.4).

9.4 Process descriptions

Although individual users are likely to be carrying out individual tasks, explain the overall process into which those tasks fit to help users to see their tasks in context. Use process descriptions to answer questions such as 'What have I got to do now?, 'Why am I doing this?' and 'What happens next?'.

If the same user will be carrying out all the tasks for a process but at different times, make it clear when each task should be carried out, for example, what event should cause the user to carry out the next task.

If the tasks are not to be carried out by the same user, make it clear which users carry out each task and, again, when they carry them out.

Diagrams are often helpful for describing processes. Ensure all the information needed is included and that users will be able to see the whole of each diagram in one view. For some products, include a process description as a section in the overview of the application.

If possible, allow users to link directly to the task descriptions for the individual tasks.

9.5 Task descriptions

Task descriptions answer questions such as 'How do I do this?', 'Do I need to do this?', 'Is this the right thing to do?' and 'Why am I doing this?'.

Give the task descriptions for related or similar tasks a common structure, using the same subheadings. Include:

- task name; identify the task, differentiating it from other tasks;
- purpose of the task; explain what can be achieved by carrying out the task (the goal) or why the task has to be carried out;
- details of what needs to have happened before the task is carried out and what should happen afterwards;
- starting point; explain the prerequisites and what inputs are needed;
- numbered instructions for the sequence in which the steps should be taken;

- end point, describing what the output is or when the task is finished.

Where needed, include optional links, or references for:

- any prerequisite tasks;
- other related tasks;
- tasks that may be required afterwards;
- process descriptions for any process including this task.

Figure 13 is an example of a task description.

If an on-screen tutorial is available, consider giving users access to an appropriate section.

If the application screen is visible at the same time as the on-screen documentation, do not use pictures of the application screen in the on-screen documentation.

If the on-screen documentation covers up or replaces the application screen, include illustrations of parts of the application screen in the documentation.

Task descriptions do not include:

- general usage instructions, such as how to use the user interface controls;
- explanations of general terms.

If users need this information, provide a method of seeing it, such as a reference to where it can be found (for example, a glossary) or the option to display it in a temporary display.

Task name	**Creating a new contact**
Purpose	Creating a new contact when you find out about a person in a client company who is not already recorded in the system.
Client Prerequisites	The client company must already be recorded in the system. Before you enter the contact, use Entity **->** Open to search for their surname to find out if we have dealt with them before. You might need to link any notes to the previous entry, or mark the previous entry as 'left' if they have left their previous organisation.
Instructions	To create a new contact: 1. Select the client company record. 2. Select Contact **->** New from the menu. 3. The system will display a box showing a choice of contact types - select Contact and choose the OK button. 4. The system will display a blank contact panel enter the contact details. 5. Close the panel and save the details.
Related topics	**See also:** How to setup a subsidiary record How to delete a contact

Figure 13 — Example of a task description with the elements marked

9.6 Explanations of fields and options

Explanations of fields and options answer questions such as 'What is this for?', 'What can I do using this?' and 'What does this mean?' when asking about fields or options displayed on the screen.

Include an explanation of what the field is for in all field descriptions. Give other information based on why the users need the information, such as for entry fields, to find out:

- how to enter data, for example using sliders, buttons, selecting from displayed lists, or typing;
- what choices are available;
- what values have to be used to select particular choices.

Include the full range of options.

NOTE This can depend upon the type of documentation. For guidance documentation and tutorials, it may be sufficient to document only the typical options, leaving the complete list of options for the reference documentation.

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. Explain all the options and the conditions that apply or include only the available options.

Explain what the information displayed in a field means. Give the meaning of the displayed value; where appropriate explain how this value relates to other possible values.

9.7 Names and uses of user interface options

9.7.1 Names

Reinforce the meanings of icons or codes by providing a method for displaying their names. Provide the name only, such as 'commit order' or 'display empty form', using the same types of names as would be used on a menu.

For icons, display the name:

- permanently next to (above, below or to the side of) the icon; both the name and the icon should be active, or
- temporarily when the user needs it, for example, in a help status line or in a bubble.

See also the guidelines in 9.17.3.

If the names of functions and controls are given different grammatical forms, make it clear to users what each one is for. For example, one set of functions could have names 'New order', 'Commit order', 'Cancel order', 'Add item', 'Unit price'.

If a user can see the names of user interface elements on the screen, provide names for all such elements, not just some of them.

9.7.2 Uses

Users may need to find out what the elements of the user interface do. Give such 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'.

If a user can see details of what user interface objects are used for, describe all such objects.

9.8 Descriptions of application functions

Application functions are the parts of the application providing facilities for users to carry out their tasks. They include the modules, transactions, screens, dialogue boxes, commands and their equivalents.

Function descriptions answer questions such as 'What is it?', 'What does it do?' and 'What can I use it for?'.

Figures 14 and 15 are examples of function descriptions.

For descriptions of application functions, determine whether or not users will be able to see the application screen at the same time as the on-screen documentation. See also 9.16.2 on the use of windows.

Provide a function description for each particular type of function, for example, commands or dialogue boxes, giving one for each function of that type.

Because of the widely differing nature of application functions, there are no general rules about what should or should not be included. However, the following guidance may be helpful.

- Use the function name as a main heading or the title of the information window.
- Include a short statement of the purpose of the function at the very beginning, to help users discover whether this is the function they are interested in.
- Explain the contexts in which the function can and cannot be used.
- Describe what users can achieve when using the function.
- If users need to understand how the application works to decide whether or how to use the function, include this information.
- For functions of similar types, subdivide the information in the same way for each and use the same set of headings.

Provide a way of finding other information related to the functions, either allowing links directly to it, if such facilities are available, or explaining where users should look to find it. For example, include references to tasks using the functions or to relevant parts of an on-screen tutorial. If direct links are available, present them consistently, for example, always at the beginning of the topic or always at the end.

In descriptions of application functions, avoid:

- general usage instructions, such as how to use the user interface;
- explanations of general terms.

NOTE 1 Restrict this information to the general sections of the documentation, to enable users to concentrate on the specific functions.

NOTE 2 If users might need this information, provide a method of accessing it, such as a reference to where it can be found (for example in a glossary) or the option to display it temporarily.

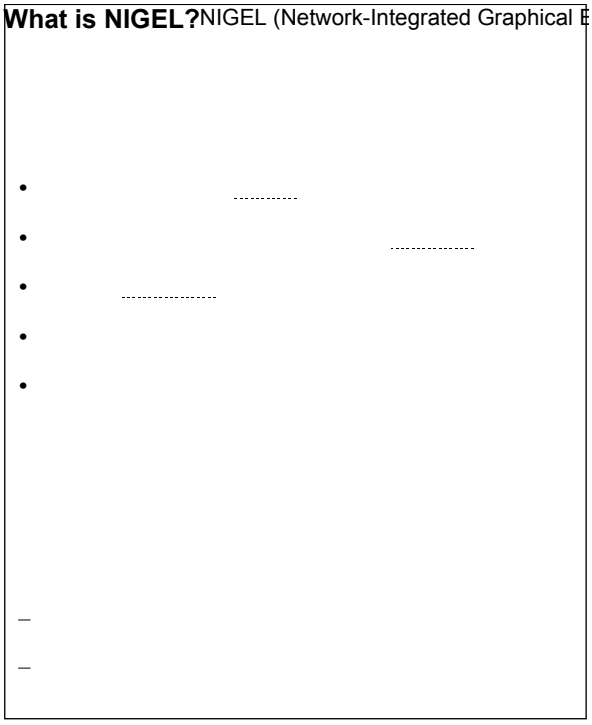


Figure 14 — Example function description for a product module

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 15 — Example function description for a spreadsheet function

9.9 Information messages

9.9.1 Format

When a condition occurs or when it is possible to avoid an impending condition, display a message to:

- advise users in advance, to help them avoid the condition, and before the user or the system has to carry out the action to which it applies.
- give corrective action information to help users avoid, or recover from, the condition.

Write information message text clearly and concisely in terms that the user will understand.

- State what the condition is.
- State the consequences to the user, equipment, software, data or service.

For important information affecting security or safety, use different types of messages. Decide for each message what its purpose is, and select an appropriate writing style. The following different types of message may be needed:

- information message;
- attention message, drawing the user's attention to something;
- action message, requiring the user to take some action.

NOTE Do not issue messages at random: they can annoy the user. For example, if it is really obvious that a record has been created, there is little value in emphasising this with a message.

Avoid the use of technical jargon and system-oriented information. Use no more lines of lines of text than are necessary.

Refer to established platform and operating system information messages in order to avoid giving conflicting advice and the possibility of taking on the overall responsibility for the total system rather than just the software application information messages.

Refer to the appropriate operating system, international and national standards for current information on the symbols to be used to bring the attention of readers to hazards and safety information. If the application can be run on multiple platforms, which use different symbols or conventions, ensure that information messages are modified accordingly.

Avoid defining new information message graphics or icons to replace existing operating system conventions, international and national standards.

9.9.2 On-screen messages

Software applications typically use operating system conventions to display a message relating to a specific situation or condition. As far as possible use the same conventions for symbols in the application information as are used by the particular operating system on which the application is to run.

9.9.2.1 Presentation

Use a message box, displayed in a secondary window, is 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 inform the user of a serious condition that requires intervention before the process can continue. An appropriate symbol usually indicates the type of message being presented.

Do not use established safety symbols that are used to warn users of potential physical dangers or life-threatening situations, where none exists, no matter how severe the consequences to the user, e.g. erasure of data.

Refer to the operating system user interface guidelines for further information on secondary windows and suggestions for textual presentation.

If the information in the message may affect the user's decision about whether or not to carry out an action, display the message at the point where the decision is being made.

Present messages that apply generally to a topic or section at the beginning of the topic or section.

Precede a message by a suitable word, such as 'Note' or 'Information'.

Use the following guidelines when constructing information messages.

- Use complete sentences.

- Do not use abbreviated forms of words or terms. Contractions may slow comprehension, especially in technical messages.
- State the condition, its probable cause, and the action the user can take, no matter how obvious the solution.
- Make messages specific. Avoid combining more than two or three conditions in a single message. For example, if an action cannot be performed for several reasons, present a specific message for each reason.
- Try to offer the user "Yes" or "No" as the choice of response to the message content.
- Avoid referring to assistance that may not be available to the user.
- Describe the effect of the user's choice of action.
- Avoid multiple step solutions. If multiple steps are unavoidable, provide separate instructions, or add a Help button to display an appropriate window. Always present the steps in the correct sequence.
- Present only as much information to the user as is necessary to understand the message. The information should be sufficient to allow an advanced user or support person to help to diagnose the condition. Use a 'context sensitive' help button to display an appropriate Topic.

9.9.2.2 Terminology

Use only terminology that the user will understand.

- Be consistent with words and phrasing for similar conditions. Provide guidance to product developers to ensure consistent and correct terminology in the product interface, messages and documentation.
- Avoid phrases that 'blame' the user or imply user error.
- Do not imply that the application or hardware can think or feel.
- Avoid the use of the word 'please'.

D.6.12 provides further guidance on writing messages.

9.10 Definitions of terms

Definitions of terms answer the user's questions, such as 'What does this mean?' and 'What is that?'. An example is given in Figure 16.

Contact

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

Figure 16 — Example of a definition of a term

Assess which terms the user will already be familiar with, and the possible meanings the user is likely to attach to them. Then prepare a list of the terms for which definitions are to be included. D.4 provides guidelines on choice of terms. If there is any doubt about whether a term should be defined, include it.

Distinguish between those terms used when applying the software, and the specialist terms used within the application itself. For example, if some specialist term is used as the name of a field, make it clear which meaning of the term is intended each time it is used.

It is preferable to collect all the terms with their definitions in a separate chapter, e.g. a glossary. If the definition of a term is also embedded in the text, the definition should be identical in all positions where it appears.

When defining terms, ensure that no jargon or colloquialisms are used in the definitions. Keep definitions simple. Also remember that the readers may not be reading the documentation in their native language. See 4.2.9 for guidance on the translation of terms and their definitions.

If it is necessary for the users to understand some terms before using the documentation, advise them where they can find this information, e.g. a Glossary, or Terms and Definitions, or Terminology.

9.11 Concepts

Because the content of concepts differs according to what users may need or want information about, it is not possible to give specific guidance. However, follow these principles.

- Give each concept item a title or heading indicating clearly what it covers.
- If a concept makes reference to particular functions of the product, or specific tasks, allow users to link to information about those functions or tasks.

An example is given in Figure 17.

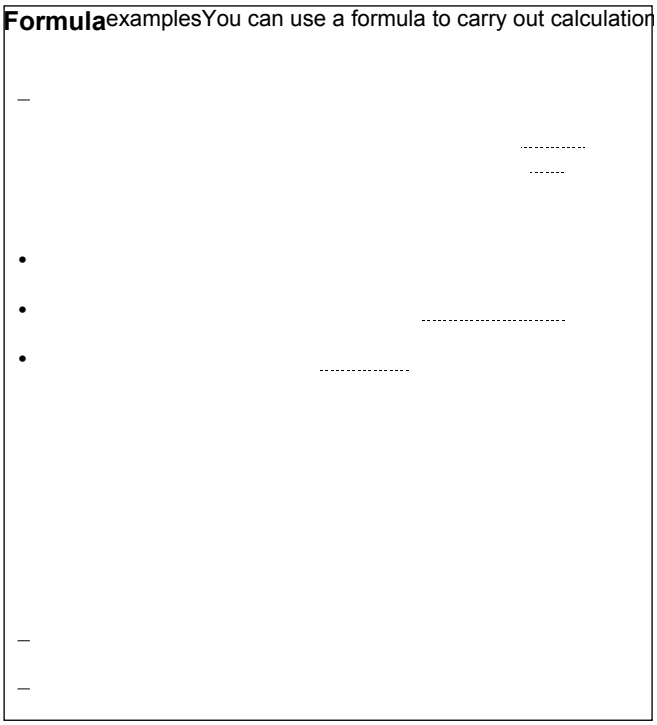


Figure 17 — Example of a concept

9.12 Exploitation information

There are three types of exploitation information:

- information about how to exploit the functions provided by the application, possibly to achieve complicated, unusual or expert effects, or to use a range of different features in combination. Examples of this type of exploitation information include:
  - hints and tips;
  - explanations of how functions can be used together to produce particular types of results;
- customising information, explaining how users can adapt the supplied application to suit their own needs, for example, to alter algorithms used for some calculations;
- information about the application area that will enhance the user's knowledge and hence lead the user to exploit the functions of the application. Figure 18 shows how links can direct the user to such information.

Base decisions about which types of exploitation information to use on the results of the analysis phase.

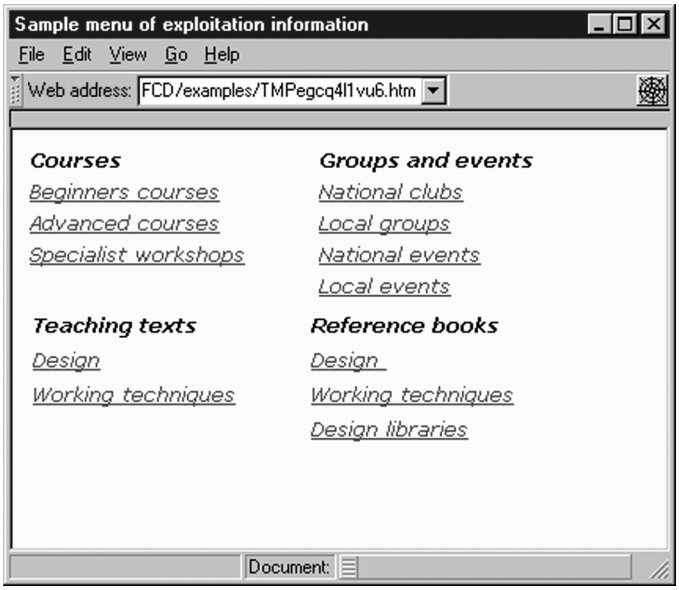


Figure 18 — Example of links to information about the application area

9.13 Frequently asked questions

Produce a library of the types of questions users want to ask about an application; this will reduce the amount of support needed for the product.

Group the information conveniently, so that users can find the topics they are interested in, without having to look at a number of irrelevant questions. 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.

Answer questions specifically and comprehensively or give access to, or references to, all the information users need. For example, rather than instructing users to 'reset the printer page size', tell users which function they need to use and what details they need to change, or give a link to the appropriate task topic.

9.14 User-supplied content

If the technology allows, and if the analysis shows that there is a need, give a user facilities for adding their own information to the existing on-screen documentation. The facilities provided could range from adding simple annotations to existing text and adding new fragments or topics, to linking to the user's own on-screen information systems or documents (see also 9.15.14).

For networked systems used by several users, there may be different categories of user-supplied content, such as:

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

In all cases, ensure that:

- the user cannot delete information that others might need;
- mistakes can be corrected;
- the original information can be restored as supplied.

Pay attention to what will happen to the user-supplied information when the system is updated. Since it is not known whether this information will still apply to the new version of the product, it is not likely to be helpful to provide an automatic method for incorporating this information into the new version of the supplied on-screen documentation. Leave this to the discretion of the user organisation.

If facilities are to be provided for users to add their own content and the technology being used means users will lose that information when a new version of the application is installed, warn users of this before they start to add their information.

9.15 Navigation

NOTE In this International Standard the term "navigation" is used within the context of on-screen documentation only. See 2.37.

9.15.1 Introduction

Users usually see each topic in isolation; so, when a topic is displayed, give users a route to find out more, for example:

- a lower level of detail on the same topic (such as explanations of individual fields or values);
- a higher level of detail (such as an overview of this type of function or display);
- procedural information (how to…); other relevant topics;
- immediate access to a table of contents, an index or a glossary.

Consider the following features where the technology is available:

- move to other items of information, possibly because the current one is not the one the user needs or because the user now needs to see different information (see 9.15.9);
- obtain clarification or amplification of some of the current information (see 9.15.10);
- browse through a large amount of information, possibly to see what information is available or to see what subjects are covered (see 9.15.11);
- go through a complete set of topics from beginning to end (see 9.15.12);
- exit from the documentation (see 9.15.13).

There are guidelines in 9.15.14 about linking to users' own annotations or documentation.

When some information consists of a collection of topics, follow the guidance in 9.15.15 about the size of topics.

Apply to the navigation features the conventions that are used for the documentation, as described in 4.2.13. If the application's conventions are being followed, then, where similar navigation functions are needed, use similar navigation mechanisms.

Consider the following general principles when deciding what navigation features to provide.

- Make navigation easy to use, understand and remember; do not provide too many different methods which may confuse users.
- Use methods of navigation consistently, so users can achieve the same effects in different situations with the same actions.
- Make the structure of the documentation very clear, so users do not become disoriented.

9.15.2 Accessing on-screen information

Make the different methods for users to access on-screen information appropriate for the different types of information. Consider the context of use (see 5.3.2) to determine the best methods of access.

Select a suitable, consistent access method from those available with the technology being used.

Consider context-sensitivity for information about:

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

If it is not possible to determine the context 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 any item specific to the context;
- display the relevant information at the top of the information area so it is the first information users see, if the required information is a piece of some longer topic.

Table 3 provides some examples of access methods that may be suitable for different types of information.

Table 3 — 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. Free-standing on-screen book.	Choose from a menu (see 9.15.3.1) Find from a contents list (see 9.15.3.2) Find from an index (see 9.15.3.3)

9.15.3 Finding the right information - linking information in on-screen documentation

NOTE The analysis and design phases 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 5.4).

9.15.3.1 Menus

Menus allow users to select the item of on-screen information they wish to see. Menus normally display only one or two levels at a time from a hierarchic structure. An example of a menu allowing users to select exploitation information is shown in Figure 19.

Figure 19 — Example of a text menu

A menu is usually displayed in the information window or frame and is replaced by a topic when the user selects one of the entries. For example, if it is not possible to predict what sort of information a user might want at a particular stage, present the user with a menu allowing them to choose the information they want. When they make their choice, the requested information is displayed in place of the menu.

Present menus as:

- a list or set of lists;
- a diagram illustrating the different levels or types of information;
- a picture of some object, such as a screen or a map.

If a menu uses icons and text, make an icon and the corresponding text active together.

Allow users to select the level or topic they wish to view directly from the displayed menu.

The names used in menus need to indicate clearly what information can be obtained using each entry. Consider giving a brief description of each entry in a menu on the menu display. Do not mix menus and information. (For details of including hypertext links to other topics, see 9.15.9)

In a hierarchy of menus, decide on the number of entries to show at each menu level and the number of levels of menu. To reduce the number of levels offer more choices at each level. Group large numbers of entries at one level, so users do not have to scan long lists of entries. If performance times may be a problem, display several levels at once on each menu.

9.15.3.2 Contents lists

A contents list is usually displayed in its own navigator window, or as a separate frame in the documentation window. The contents list usually remains displayed while the user is selecting and reading topics in the topic window or frame, until the user chooses to close it. A contents list often shows the structure of the information using, for example, a tree structure.

Provide contents lists for on-screen books, such as those that give users additional guidance on using the product. Also consider providing contents lists for on-line help systems where the set of topics can be viewed as a hierarchical structure. Contents lists help users find the information they need while understanding the structure of the information available.

Allow users to expand and contract the different levels in the structure, to see different levels of an outline. Provide a way to access the information contained in the documentation described by the contents list directly from the list.

Use headings that indicate clearly what information is contained in each section.

Consider using icons to indicate the different types of information contained in different sections of the documentation, or to indicate different statuses. Make both the icon and the associated text active, not just the icon.

Figure 20 shows an example of a contents list, partly expanded.

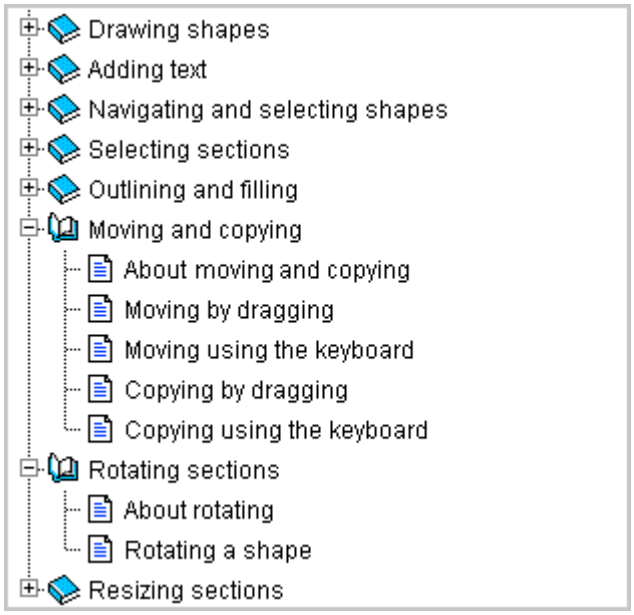


Figure 20 — Example of a contents list

9.15.3.3 Indexes

Provide an index for on-screen books, such as those that give users additional guidance on using the product. Also consider providing an index for on-line help systems where the user may want to find topics about a particular subject. Provide indexes where users may not know exactly what vocabulary to use to identify topics to look at.

Allow users to request topics using alternative vocabulary from that used in the on-screen documentation. To determine the different terms users might choose, consider why they might want to look up the topics.

Make clear for each entry what type of information is provided at each reference, for example, whether or not the information is simply a definition, a description of a function, a task description and so on.

Provide simple methods to select topics from an index.

9.15.3.4 Searches

If the user will be able to search for information, determine the types of searches to be provided, based on the methods it is expected that users would use when looking for information.

Searches may not be appropriate for novice users; give novice users another method of finding information.

Design search methods carefully. Make the complexity of the user facilities for specifying search terms and conditions appropriate for the audience. Avoid complex interfaces for searches and complex search mechanisms of the types often used by librarians and researchers (which allow flexible searches for complicated combinations of words and phrases).

Find out, preferably directly from users, what terms users are likely to use when searching for information, and what types of information they are likely to want to search for, and include corresponding search facilities.

Ensure that the cursor is in the search text box so that the user can begin typing the search criteria without first having to place the cursor in the box.

Avoid the following:

- search mechanisms that find every occurrence of a word; some selection of the set of subjects found is needed;
- searches that provide only the same information as would be contained in a menu, a table of contents or an index.

9.15.3.5 User-defined navigation

Where possible, provide facilities for:

- choosing the level of information displayed, if there are different levels available;
- saving details of the current context, so it can be restored. For example, if several tasks are needed for an activity, let the user quit the activity part way through, and return later;
- labelling sections of the on-screen documentation so that the user can go straight to those sections.

9.15.4 Knowing what the current information is

Make obvious to the user the subject of the displayed information and what type of information it is.

Use the following techniques; select at least one method for each type of item.

- Display the information on or close to an object, for example, next to or beneath an icon or screen field.
- Display a title in the information window identifying an application function, such as the name of a dialogue box or a command.
- Use the first heading in the text to explain the type of information. For example, task descriptions can have headings such as 'How to' or 'Instructions'. Where possible, give the name of the task, including an indication of the process and the task within the process that the instructions apply to.
- Use different symbols to distinguish between different types of information.

When deciding which method to use, take account of the types of question that may have caused users to look at the on-screen documentation. Where possible, use titles and headings to indicate what questions will be answered.

9.15.5 Knowing the current position within a topic

If a topic is longer than can be viewed in a single window, give the user a clear way of knowing where they are within the topic. Apply the following techniques where appropriate; use at least one method for each such topic.

- Structure topics of the same type 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'.

- Number the headings 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.

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

- If none of the above devices is available, consider splitting the topic 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.

Always make it clear if there is more information to follow and how to access it.

9.15.6 Finding the same information again

Allow for the user's need to return to information they found before, either in the current dialogue or later in the present session or in another session.

Context-sensitive information can always be found again, by returning to the same context. But this may rely on the user remembering a sequence of actions.

In other cases, for going back to the same information in the current dialogue:

- provide a navigation control for returning to the previous topic;

- let the user return to any topics they have seen during the current dialogue.

Consider these other techniques for enabling users to find the same topics again.

- Maintain a cumulative list of what topics the user has looked at either in this session or in this and previous sessions and allow them to select topics from this list.

- Allow the user to mark places they may want to return to; make bookmarks or other markers very easy to find.

- Make the structure clear so the same paths can be followed again.

- Annotate a menu to show which topics the user has looked at.

Whatever techniques are used, make them consistent and easy 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.

9.15.7 Switching between the application and the documentation

If the user needs information to be available while they are using the application, but there are no facilities available for displaying the two at the same time, provide a fast, easy-to-use (and easy-to-remember) method for users to switch between the two.

When switching between the application and the information, do not let the user lose their place in either.

9.15.8 Printing information

Enable the user to print out some of the on-screen information, selecting for themselves how much information to print. Enable them 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 fragments of on-screen information (such as those used to give the names of icons).

9.15.9 Moving to a different topic

If the user is allowed to move from one topic to another, make it clear:

- what they have to do to move;
- what the effect will be.

In particular, make it clear whether the new information will be displayed as well as or instead of the current information.

This sub-clause describes the situation where the new information will replace the current information (referred to as linking). For situations where the new information is displayed in addition to the current information, see 9.15.10**.**

Where possible, provide in a consistent way links:

- back to the previous topic the user viewed, if any;
- back to the beginning, or top-level menu or contents list;
- up a level in a hierarchic structure;
- forward to a related topic;
- forward to the next topic in a sequence (see 9.15.11).

If facilities are provided to link to topics in different parts of the hierarchy, give the user some way of understanding where they are in the hierarchy after the link. For example, in the topic heading include words or a diagram explaining where it belongs in the structure, or highlight the new position in a contents list.

If facilities are provided for the user to link by selecting active text, distinguish this text from the surrounding text and from active text used to obtain clarification. Collect such elements together at the beginnings or ends of topics under headings such as 'Related topics' or 'See also'.

Provide a method for returning to the original topic (see 9.15.6).

9.15.10 Obtaining clarification or amplification of current information

If the user needs more explanation about some information, such as a definition of a term, amplification of some text or some part of a picture, then provide a mechanism for displaying the information temporarily, or give users details of where the information can be found, for example, in an independent glossary of terms.

Avoid causing the user to link out of the current topic to see the extra information when it is only amplification of something within the current topic. If they do need to link, provide a simple method to return to the current topic (see 9.15.6).

Make the method provided for displaying the extra information clearly different from the method used for linking to a completely different topic, otherwise the user may confuse the two.

9.15.11 Browsing through information

Provide several navigation aids for general information that the user may want to browse through, perhaps to learn about new subjects, perhaps simply to see what information is available.

- Display the hierarchic structure of the information and allow users to select paths through it (see 9.15.4).
- Provide a default path through the topics (see 9.15.12).
- In each topic include details of related topics (see 9.15.9).
- Provide an index (see 9.15.3.3).
- Allow searches (see 9.15.3.4).

9.15.12 Viewing topics in sequence

If there is a natural sequence for some groups of topics (a browse sequence), 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, display these on the screen.

In such sequences, clearly indicate to the user when they have reached the end of the sequence, for example, by de-emphasising or dimming the 'Next' or 'Previous' control, or including a marker at the end of the sequence. Consider showing the start of a sequence.

9.15.13 Exiting from the on-screen documentation

Provide a simple method of exiting from the on-screen documentation, unless there is a standard method used by the operating system or the product suite.

Make the exit method the same for all information of the same type.

9.15.14 Finding user-supplied information

If facilities are available for allowing the user to annotate the supplied information:

- make the method for adding the annotations clear and simple;
- make the annotations easy to find.

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, distinguish it from other information. For example, if it is text, highlight it in a different way from other text or enclose it in special symbols (see 9.16.5.6).

Make the meanings of icons clear.

Make the meaning of a signpost and the method of accessing the annotation clear.

9.15.15 Sizes of topics and fragments

Determine the sizes of individual topics or fragments delivered to users by the needs of the users, not by the tools being used.

When deciding the sizes of topics, consider:

- how much information the user can see at once, without scrolling through it;
- how much information the user actually needs, as opposed to how much information there is available;
- what information the user might want, rather than need, which might lead to topic design where some information is always displayed, and the availability of additional information is indicated so those who want it can ask for 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 all the information were not supplied in one place.

Where possible, make the whole of each topic visible at once. Where this is not possible, provide a way of scrolling or paging and make it clear how far through the information the user has moved (see 9.15.5).

Make the design flexible so it can adapt easily to a change in the volume of information resulting from amendment or updating, or through translation into another language.

9.16 Presentation

9.16.1 Introduction

NOTE 1 ISO 9241-12 gives detailed guidance on presenting information on a screen.

NOTE 2 In several places there is advice on spacing of text, character size and line length. For the background to this advice, see reference 16 in the Bibliography.

Give on-screen documentation the 'look and feel' of one of the following:

- the application;
- other products in the same suite;
- the operating system;
- other on-screen documentation with which users may be familiar.

Make it clear to users when they are viewing the application and when they are viewing on-screen documentation. Use methods such as:

- a different style of window;
- clear titles;
- different colours (but see 9.16.4).

9.16.2 Windowing

9.16.2.1 Relationship of information displays to the application's displays

Apply the analysis of user needs to design how documentation windows are positioned in relation to the application (see 5.3.2).

If information is needed all the time, do not allow it to disappear when the user presses a key or selects another window.

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

Present information that is needed at the same time as the application interface in at least one of the following ways:

- provide the information on the application screens, instead of as documentation;
- display the information and the application so both are visible;
- provide a simple method of switching between the two;
- provide the information in printed documents.

9.16.2.2 Arrangement of windows

9.16.2.2.1 Introduction

When several windows are to be visible on screen at the same time ensure that:

- the information in each window follows the guidelines on the size and layout of text and illustrations in 9.16.5 and 9.18;
- users can see a meaningful amount of information in each window;
- the title of each window is clearly visible.

9.16.2.2.2 Default window sizes and arrangement

Use a fixed arrangement of windows only if it is known exactly how many windows users will need at one time and exactly what type of information and how much information will be required in each window. Otherwise, design a flexible arrangement whereby:

- the system determines a default placement of each window;
- users can override the default placement and sizes of windows and choose their own arrangement.

Apply a default size and placement of windows so that users do not have to move or resize the windows to see the information they need or to carry on working with the application if necessary. Do not obscure information such as navigation controls.

9.16.2.2.3 Arrangement of windows

Where possible, allow users to define their own arrangement of windows.

Consider the following arrangements of windows on a single screen:

- tiled, in which the windows are arranged so that they appear side-by-side;
- overlapping, when:
  - the number and sizes of windows to be displayed at different times varies;
  - 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.

A simple example of overlapping windows is shown in Figure 21.

NOTE This arrangement is sometimes referred to as “cascading”.

9.16.3 Layout and grids

9.16.3.1 Grids

Draw a grid for each type of on-screen documentation window to show how all the elements required in the window will be positioned. Take into account:

- any 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.

If users can resize windows, allow for this in the design; plan what will happen to the following when the window is resized:

- information area;
- size of text and illustrations;
- non-scrolling areas.

Test samples of screen layouts in user tests.

Figure 21 shows an example grid for a help system navigator and a topic window.

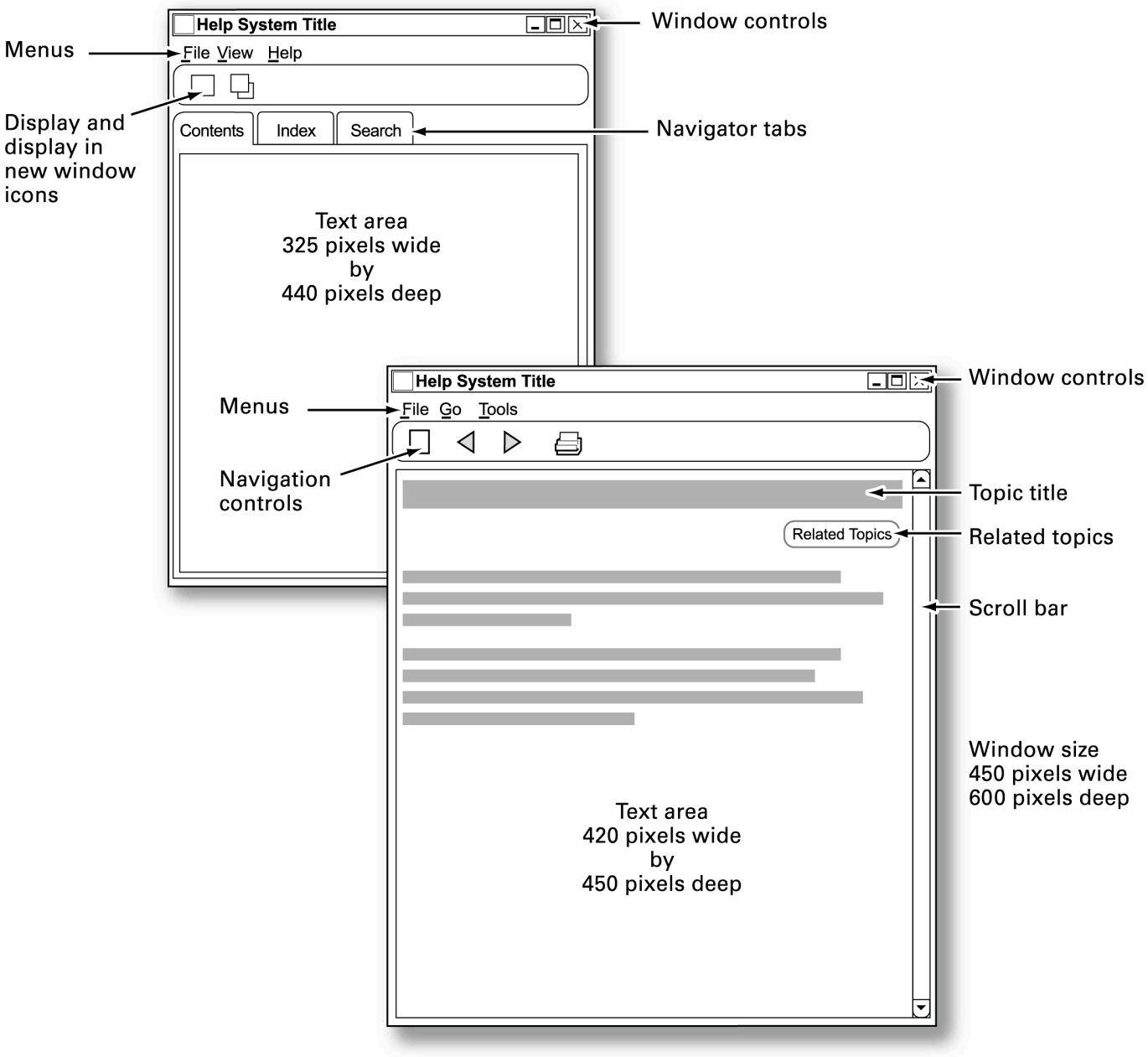


Figure 21 — Example grid for a help system navigator and a topic window

9.16.3.2 Information area

Use generous spacing for large amounts of continuous text.

In general, present text in a single column. Avoid long lines of text in the default window size. See 9.16.5.3 for additional guidelines. If continuous text is presented in tables, ensure that the columns are not excessively narrow, because the sentences will be difficult to read.

NOTE For the English language, 60 characters is the recommended maximum line length [18].

Design the information area to allow users to scroll vertically through information too long to fit all at once. Do not make users scroll horizontally to see the whole width of text or illustrations. Make text wrap round within the text area.

When users scroll vertically through information in the information area, allow scrolling in single units and multiple units clear to the user, for example, a line at a time and a complete window at a time. Ideally, keep visible the title of the information.

When displaying a web browser window of information, predefine the chrome to avoid unnecessary window controls.

9.16.3.3 Placement of illustrations

Specify in the window grid how illustrations should be placed within the information area. Display in a consistent way illustrations of the same type, serving the same type of user need. Follow the guidelines in 9.18.

9.16.3.4 Blank space

Clearly separate the elements presented in a window. Use space rather than drawn lines or other devices that distract and give the window a uniform appearance, making it more difficult to identify the separate elements (see also 9.16.3.2).

NOTE Blank space is also referred to as "white" space.

9.16.3.5 Title and path

Display a consistently positioned title in each window that contains more than a simple name. (Do not title fragments of information such as names of icons.)

Where users move between topics, indicate where the current topic fits into the total structure; allow for this in the grid.

If the information area is a scrolling area, maintain the display of the title when the text scrolls.

9.16.3.6 Navigation controls

Use consistently named and placed navigation controls on all displays of the same type.

9.16.3.7 Non-scrolling areas

If possible, display the following in a non-scrolling area:

- topic title (in a web browser window this can be displayed in the title bar);
- navigation controls for moving to other topics (in a web browser navigation these can be displayed in a non-scrollable frame or as selected browser chrome);
- ‘related topics' or 'examples' sections giving access to other topics.

9.16.4 Colour

9.16.4.1 Use of colour

Consider specialist advice about the use of colour both on paper and on a screen from, for example, a usability expert. Consider specialist advice about the production of colour for use on paper from a publishing house or printing organisation and for use on a screen from a technical expert.

Do not use colour alone to convey meaning. Where appropriate, use colour for illustrations but, where possible, use also devices such as different types of hatching or shading patterns.

Use colour only to guide the user, for example, to emphasise certain text or graphics, or link related elements. Do not use colour arbitrarily.

When using colour, follow these guidelines.

- Consider the effect of the use of colour on a colour-blind reader.
- Do not refer to the names of the colours on the screen; some users may be colour blind, have a monochrome screen, or have selected a personalised colour combination.
- For embedded information, check, and provide advice regarding, the colour palettes available on the user's display devices. Might some or all users have only monochrome displays? What range of colours will be available? Choose a colour scheme and palette that is suitable for all the users.
- Where consistency of colour is important between the embedded and printed documentation, check the proposed colours thoroughly.
- Test which colours and combinations of colours reproduce or display best and are least likely to cause confusion, especially with colour-blind users.
- Consider cultural interpretation and usage of colours.
- Colour printing can be expensive. Verify cost estimates for reproduction with printing organisations before making a decision.
- Choose a printing method capable of handling colour. Consistent reproduction of colour can be difficult; some methods distort colour more than others.
- Be aware of how the colours will appear when the content is printed on colour and monochrome printers.

9.16.4.2 Customisable colours

Allow the user to alter the default background and foreground colours. Maintain the user's preferences until the user requests another change. Allow users the option to restore the system defaults.

9.16.4.3 Colour palette

Use only colours from the standard palette of the documentation delivery tool (if there is one), to display colours correctly on users' screens. (Avoid different palettes; the user may not be able to display them at the same time.)

9.16.4.4 Background and foreground colours

Use solid colour for background or to fill areas of the display. Avoid colour causing patterned effects on the screen. Select a neutral background colour because the perceptibility of objects depends on background colour. Take care when placing coloured objects on a non-white background.

For international audiences, consider the cultural connotations of different colour combinations.

9.16.4.5 Design in monochrome

Check how the documentation will appear on a monochrome display to ensure that they will achieve the required effects when displayed in monochrome or are viewed by people who are colour-blind.

9.16.4.6 Number of colours

Colours can easily distract the user's attention from the working context and may therefore decrease the user's performance. Two or three colours are usually sufficient. Start with a monochrome design and add colour where it provides additional emphasis to the other coding techniques being used.

9.16.5 Presentation of text

NOTE This clause provides guidelines for many aspects of the presentation of text; some of these aspects may already be dictated by the presentation technology or overriding corporate style guide.

9.16.5.1 Typefaces and sizes

Use typefaces that are readily available for the planned presentation technology or reproduction method. Before starting work on a project, check that the selected typefaces provide all the special characters that are needed. Do not use typefaces in which the lower-case **L**, the upper-case **I** and the numeral **1** are indistinguishable.

When selecting typefaces and sizes to use for on-screen documentation, take into account:

- the range of different display screens that will be used;
- the range of typefaces and sizes that users' systems will have available; try to use, or take account of, the default type fonts installed on computers;

  NOTE The use of fonts which may not be installed on a user’s computer will result in the computer replacing the font with one of the 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;
- the character sizes:
  - for on-screen documentation, select a size that will be displayed clearly on the types of screens that might be used. At the point of display to users, upper-case letters need to be at least 3 mm high. Do not rely on internal settings of point sizes as the measure of text displayed on the screen; measure the sizes of displayed text on all types of screens that will be used;

- for printed documentation select a font with upper-case letters not less than 2.75 mm high and the height of a lower-case x (the x-height) not less than 2 mm. This corresponds to a size between 9 point and 11 point.

Use of no more than three different typefaces and use them consistently.

- Headings. Use one typeface for headings (see 9.16.5.5).
- Normal text. The typeface can be the same as that used for headings.
- Possibly use a third (perhaps mono-spaced) typeface to represent code and information that users have to type (see 9.16.5.12 and 9.16.5.13).
- For header and footer information in printed documentation:
  - use one of the typefaces used for the text or for headings, to avoid having too many different typefaces;
  - select sizes and weights that take into account how the information will be used.

9.16.5.2 Use of bold and italic type

The bold and italic forms of the normal text typeface can be used to convey meaning. Also use these forms to ensure users pay attention to important information when notes, cautions and warnings are not appropriate.

Limit the use of bold to:

- headings (see 9.16.5.5);
- column headings in tables;
- titles of illustrations and tables.

Limit the use of italic to:

- text that users need to type;
- variables;
- identification of screen elements, in particular, menu items;
- identification of particular words or terms, such as names of fields on a screen form;
- introduction of new terms;
- quotations;
- article and book titles.

NOTE 1 Italic fonts can be difficult to read on a screen. Always check the legibility of planned type fonts on computer displays.

NOTE 2 Use bold and italic forms consistently throughout the same document or set of documents.

9.16.5.3 Lines of text

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

NOTE 1 This format is not used in this International Standard, as it is not a software document, and follows ISO drafting rules.

Make the spacing between lines of text, measured from baseline to baseline, at least one thirtieth of the line length and at least twice the x-height. Check this feature; do not assume that automatic settings will be satisfactory, because the x-heights of different typefaces vary quite significantly.

Use non-breaking spaces when appropriate, for example:

The sampling rate is 1

kHz

is not ideal and should use a non-breaking space to give:

The sampling rate is

1 kHz

For normal text in printed documentation use a maximum line length that has been found suitable for the language being used.

NOTE 2 For English a suitable maximum is 60 [18].

For printed documentation use mixed upper and lower case for text; text in all upper case is difficult to read, and consequently the user might ignore it.

9.16.5.4 Vertical spacing

Use vertical spacing 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.
- Separate each heading 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 will not stand out.

Select the text line spacing (the "leading") so that lines of text do not appear to be too close together, but also to ensure that users can 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 (i.e. 10 on 12 point).

Separate paragraphs with vertical space. This helps users to see where one paragraph finishes and the next one starts, and reduces the density of information. Do not fill the space between paragraphs 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.

Make the vertical spacing between paragraphs greater than or equal to half the line spacing of the text but less than or equal to twice the line spacing, provided the vertical hierarchy of headings and text is maintained. For example, for a line size of 12 points, make the space between paragraphs between 6 points and 24 points.

9.16.5.5 Headings in text

Use different sizes and weights of type for different levels of heading in a sensible hierarchy, limiting their use. Two or three different heading styles are enough for most documentation. For example:

- one style for a topic or chapter title;
- one or two styles for subheadings within a topic or chapter.

Set typographic conventions for different types of heading. Consider:

- different sizes: limited to two or three different sizes;
- bold and italic: use in combinations to indicate different types of heading (bold adds emphasis; italics appear lighter than the Roman equivalents);
- colours: use only in conjunction with at least one of the other devices;
- underlining: avoid unless there is no other method available for distinguishing headings, for example on text-only displays.

If information is being displayed in topics, each of which fits in one window, use vertical spacing to separate the headings from the text.

If the user can scroll through text on the screen, consider numbering the headings to provide signposts to users to indicate the position they have reached within the text.

If the headings indicate a hierarchy, carry out trials to ensure that the different heading styles accurately represent the different levels of the hierarchy.

When users are reading documentation on the screen, they may have only one level of heading visible at a time, so they are not able to determine the position of the heading in a hierarchy. For this reason, two levels of heading are normally enough for on-screen documentation: one for the major heading at the top and another for headings within a topic.

9.16.5.6 Highlighting

If it is necessary to highlight some text, draw up conventions and apply them consistently.

The following highlighting methods can be used.

- Bold: adds emphasis to text and attracts the reader's attention;
- Italics: can be difficult to read in large amounts and in some typefaces; use only for small amounts of text;
- Underlining: avoid if possible, but if different types of underlining are used for highlighting text, for example for active text and for new terms, make the different types of underlining immediately distinguishable;
- Enclosing text in special characters: use square brackets and braces to enclose special types of text, particularly active text;

- Colour: do not use colour as the only method of highlighting.

Avoid using the following to highlight text unless there is no other method available:

- underlining that crosses the descenders of characters;
- blinking: do not highlight words by blinking the text itself; if necessary, display a marker next to the text and make the marker blink;
- upper-case letters: use according to the guidance in D.6.11; do not use upper-case letters for emphasis, in particular, avoid long lengths of upper-case only text.

9.16.5.7 Lists

Set presentation conventions for lists. Decide:

- the number of levels: try to stay at one level of list; avoid more than two nested levels;
- the numbering scheme: set conventions for:
  - the numbers or letters to be used for each level;
  - whether the numbers or letters are to be followed by, or enclosed by, some characters;
  - the characters to be used to introduce bullet list items.

Use numbers only when the sequence of the items in the list matters, for example, to number a set of steps the user must follow in sequence.

Avoid Roman numerals because they vary so much in length.

If the bullets are being used as navigation controls, apply the conventions in use for navigation controls. Do not use bullets larger than the normal text characters.

Decide whether list items can have headings and, if so, how the headings are to be presented.

If a list is too long to be visible all at once on the screen, break it up into separate pieces of text, accessible using the chosen navigation techniques.

9.16.5.8 Tables

In a table, put items that are to be read together, close together in rows not in columns.

Separate items using space. Do not use drawn lines to separate cells that should be read together. See 9.16.3.4 for guidelines on the use of space as a separator.

Remember that when the text is translated, the length of the text will change; the translated headings and text may require more space.

When preparing tables for display on the screen:

- if the user has to scroll through a table, ensure that the headings are always displayed;
- if scrolling to the right is used, keep the left-most column on screen;

- remember that if users resize a window, some text may not be visible or the alignment of the text in the columns may be altered.

If there are problems, consider:

- splitting the table into several smaller tables;
- presenting the table as a series of lists.

9.16.5.9 Boxes and borders

Consider separating items with space rather than drawn lines. Box text and illustrations if it helps users to distinguish information of particular types or to keep associated information together.

Do not use boxes and borders only for decoration.

9.16.5.10 Annotations

If users can annotate text in the on-screen documentation, provide a signpost or icon to identify where the annotation belongs.

Distinguish the information itself, 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, or one of the highlighting methods in 9.16.5.6.

9.16.5.11 Representing the information displayed by the software application

If all the characters displayed on the screen have the same width (mono-spacing), rather than proportional spacing, show those characters in the documentation using mono-spacing. For example:

```
Do you want to continue? For Yes, press Y. For No, press N.
```

Copy text from software dialogues exactly, character for character.

9.16.5.12 Representing information that users need to type in

Clearly distinguish information users need to type in from other text and information that the software presents. For example, use mono-spaced bold type.

Present the information users have to type, character by character. Represent general terms for which the user needs to substitute a value (known as variables) in italic type. If possible, avoid using brackets, braces, the characters < and >, and single or double quotation marks. Define any formal notation in every document that uses it, with the exception of […] for references in a bibliography.

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`*

9.16.5.13 Representing individual keyboard keys

In text, distinguish terms or symbols referring to keyboard keys from the same characters used in other contexts with a convention such as one of the following.

- Use a special typeface or font, for example, small upper-case letters or bold characters.
- Use special characters such as < and > to enclose the term or symbol, but 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.

If possible, copy the characters printed on the top of the key exactly.

If different keyboards, with different characters on the key tops, might be used, set conventions the for names of keys such as Enter, Shift, Tab. Define these conventions in every document needing them. Use them consistently.

A convention is frequently needed for representing a space character; if so, use a character not used for any other purpose in either the documentation or the software. Explain the convention in every document that uses it.

Light characters on a dark background should be avoided, because they give undue emphasis to the characters.

Use pictures of key tops if possible, but if the pictures are larger than the normal text of a document, consider putting the pictures to the side of the text, in a list or in a table.

9.16.5.14 Variables

Variables represent information for which values have to be substituted. For example, the variable *filename* might be used as a general term for which a real filename has to be substituted. Present variables in italic text (see example in 9.16.5.12).

9.16.5.15 Controls

The presentation of all controls should be designed to the same conventions as the controls for the application or to the conventions used for the system with which the on-screen documentation is to be consistent.

The different types of control should be presented in clearly different ways. In particular, active text of different types should be very clearly distinguished, so that users understand what will happen if they select it. The following need to be distinguished:

- active text or active areas causing a link to another topic, replacing the current topic;
- active text or active areas causing additional information to be displayed as well as the current topic.

Consider the following methods and select at least one for each type of active text or active area.

- Change the shape of the pointer as it moves over the active area.
- Present the active areas in different ways, using different methods of highlighting or different methods of displaying the area.
- Use different icons for the different actions.

9.17 Icons and signposts

9.17.1 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. This saves space in the window, allowing more information to be displayed. Also, appropriate pictures can often show the meaning better than many lines of text.

NOTE 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 simply a piece of text, a symbol or a small graphic signalling a specific place. Use signposts 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;
- signal a particular type of information, for example some instructions, a topic or a concept.

Make graphics or symbols self-explanatory within the context in which they are being used.

9.17.2 Design of icons and signposts

NOTE ISO/IEC 11581 addresses the design and presentation of common icons for software interfaces.

9.17.2.1 General guidelines

Use the following guidelines when designing graphics for use in icons or as signposts.

- An icon should not distract the user's attention, but should be large enough to be selected without excessive precision with the pointer.
- Use representations of real world objects or actions if possible.
- Emphasise the graphical elements that distinguish this object from other objects.
- Simplify the representation by eliminating or de-emphasising the elements that do not contribute to the object's identification.
- Use a consistent approach when designing and constructing icons and signposts, such as the re-use of visual metaphors, scale, orientation, colour and location of graphical components.
- Use as few graphical components as possible; excess detail obstructs recognition and comprehension.
- If the software uses specific symbols or icons:
    - use the same symbols or icons in the documentation, so that users have a simple and consistent method of relating the documentation to the software;
    - do not invent alternative sets of symbols or icons purely for the purposes of the documentation;
    - never use different symbols or icons for the same object or function;
    - never represent different objects or functions by the same symbol or icon.
- Set icon and signpost standards for all of the product’s documentation; use them consistently.
- Do not use text in icons or signposts, unless a translated version can be substituted when another language is used.

9.17.2.2 Choice of icon graphics

Select the icon graphics carefully to avoid misinterpretation. For example, if a picture of a printer is used, users might be unsure whether it represents on-screen information about printers and printing, or represents a way of obtaining a printed copy of the current topic.

For international audiences avoid the use of culturally dependent graphics. If culturally dependent graphics are used, they should be replaced as part of the translation or localisation processes.

If an icon represents changes in the status of an item, use the metaphor 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.

Assess the suitability of the graphics during user tests.

9.17.3 Displaying the names of icons

If the names of the objects or functions represented by icons are displayed, consider:

- are the words 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 users be allowed to choose alternative graphics to use on the icons?
- should the users be allowed to select whether to display the words permanently in addition to the graphics?
- will those names be translated?

9.18 Presentation of illustrations

For diagrams (and line drawings if they are appropriate), set conventions for the following, to ensure all the illustrations within a document are consistent.

- Line thickness: make drawn lines thick enough to be clearly visible and to reproduce well using the intended reproduction or presentation technologies. If the diagrams include text, avoid lines so thick that they detract from the main message.
- Typefaces and sizes of text: use the same typeface for all diagrams and illustrations.
- Sizes of arrowheads and other connectors: select a size appropriate to their importance in the diagram (generally visible but small).

  NOTE 1 ISO 4196 defines recommended arrow styles and their uses.

- Shading: use shading only to convey information; avoid decorative shading, for example large areas of grey.

  NOTE 2 Shading causes difficulties with some reproduction and presentation technologies.

- Presentation of titles: place the illustration title either above or below the illustration and use either bold or italic, but be consistent. Use a style similar to table titles.

These recommendations apply to the finished size of the illustrations; if illustrations are to be reduced in size in the finished documentation, allow for this reduction in the conventions.

D.9 contains further guidance on illustrations.

Annex A(Informative) Process Checklists

NOTE Copyright is waived by ISO in respect of the checklists in Annex A.

These checklists provide a means of recording decisions and actions at each stage of documentation development. Each checklist is presented on one page, for ease of use.

A.1 Objectives

	Notes
Collect all relevant information that exists such as (see 4.2):	
• Customer requirements	
• Marketing requirements	
• Product plans	
• Task analysis	
• Evaluation reports	
• Customer feedback	
• Previous versions of the product and product documentation	
Find out the objectives for the following:	
• The product (see 4.2.2)	
• Scheduling (see 4.2.4)	
• Usability (see 4.2.5)	
• Accessibility (see 4.2.6)	
• Future modifications (see 4.2.7)	
• Internationalisation, localisation and customisation (see 4.2.8)	
• Translation (see 4.2.9)	
• Packaging (see 4.2.10)	
• Legal requirements (see 4.2.11)	
• Security (see 4.2.12)	
• International Standards and conventions (see 4.2.13)	
• Costs (see 4.2.14)	
• Documentation delivery and viewing mechanism (see 4.2.15)	
Write the Documentation Proposal (see 4.3)	

A.2 Planning

This checklist should be used at each planning stage. Not all the subjects will be included in plans at every stage. Some may be added in the later stages.

	Notes
Collect relevant documents, such as:	
• Previous version of the Documentation Plan	
• Specifications	
• Requirements documents	
• Documentation Proposal (see 4.3)	
• Quality Plan	
Decide the standards and conventions to be followed (see 5.2.2)	
Decide the version and change control procedures (see 5.2.3)	
Plan provision of personnel (see 5.2.4)	
Plan provision of equipment (see 5.2.5)	
Assign responsibilities for aspects of the documentation (see 5.2.6)	
Estimate the financial costs (see 5.2.7)	
Prepare the schedules (see 5.2.8)	
Plan the prototypes (see 5.2.9)	
Plan the system tests (see 5.2.10)	
Plan the documentation reviews (see 5.2.11)	
Plan the user tests (see 5.2.12)	
Plan the localisation (see 5.2.13)	
Plan the customisation (see 5.2.13)	
Plan the approval mechanism for the documentation (see 5.2.14)	
Decide how to handle updates and future developments (see 5.2.15)	
Plan the reproduction method for the printed documentation (see E.2.1)	

Agree on the form of the deliverable to be handed over for release (see E.2)	
Write the Documentation Plan (see 5.3)	

A.3 Analysis

	Notes
Collect all relevant documents that exist, such as:	
• Documentation Proposal (see 4.3)	
• Documentation Plan (see 5.2)	
• Output from systems analysis	
List the audiences (see 6.1.1)	
Write a profile of each type of audience including:	
• Learning stages (see 6.1.2)	
• Working environment (see 6.1.3)	
Draw up or find a task analysis (see 6.2.1)	
Map the audiences to the tasks (see 6.2.2)	
List the characteristics of each task (see 6.2.3)	
Work out the information needed by users including (see 6.3):	
• Context of use (see 6.3.2)	
• Volume of documentation (see 6.3.3)	
• Kind of information (see 6.5.2 and 6.5.3)	
• Document structures (see 6.6)	
Specify usability goals (see 6.4.1)	

A.4 Design

	Notes
Collect all relevant documents that exist, such as:	
• Documentation Plan (see 5.2)	
Decide what medium to use for information (see 6.3.4)	
Decide what tools to use (see 5.2.5)	
Design the details of the documentation including:	
• Content (see 9.3 to 9.16)	
• Structure of printed documents (see Annex E)	
• Navigation in on-screen documentation (see 9.15)	
• Style (see Annex D)	
• Presentation (see 9.3 and 9.18)	
Specify the usability goals and measures (see 6.4.1)	
Develop prototypes (see 5.2.9)	
Plan user tests (see 5.2.12)	

A.5 Development and review

	Notes
Collect all relevant documents that exist, such as the Documentation Plan (see 5.2)	
Examine the prototypes and the feedback from them (see 6.2.9)	
Draft the information (see 7.2)	
Review the information (see 7.3.2)	
Carry out usability tests (see 7.3.3)	
Carry out system tests (see 7.3.4)	
Carry out validation and field trials (see 7.3.5)	
Prepare subsequent draft and retest (see 7.4)	
Hand over the finished documentation (see 7.6)	
Hand over the documentation for localisation or customisation (see 7.7)	
Prepare the archive (see 7.8)	

Annex B (Informative) Design Checklist

NOTE Copyright is waived by ISO in respect of the checklists in Annex B.

Checklists should be drawn up for reviews of the documentation. The checklists given here may be useful as a starting point.

B.1 Content checklist

This checklist should be applied to the complete documentation suite, both embedded and separate, taken as a whole.

	Notes:
General information:	
• Is it clear what version of the software the documentation applies to?	
• Is the name of the manufacturer clear?	
• Is the information that users need when asking for support included?	
• Is there a copyright statement?	
Overview of the application:	
• Is there an overview of the application?	
• Does it explain what the application is for?	
• Does it explain what application functions are available?	
• Does it explain the structure of the application?	
Overview of the documentation:	
• Is there an overview of the documentation?	
• Does it explain what documentation there is?	
• Does it explain how to use the documentation?	
Task descriptions:	
• Is there a task description for each task that users can perform?	
• Are there process descriptions that put the tasks in context?	
Fields:	
• Are all fields explained?	
• Are all options explained?	
• Is the information about different types of field appropriate?	
User interface elements:	
• Are all the elements of the application's user interface explained?	
Application functions:	

• Are all the functions of the application described?	

Messages:	
• Are messages explained, if necessary?	
Terms:	
• Are all the terms used either defined in the documentation or already familiar to users (see 9.10)?	
• Is the terminology used consistently?	
Concepts:	
• Are all the important concepts explained?	
Exploitation:	
• Is there information on how to exploit the advanced features of the application?	
Questions and problems:	
• Does the documentation answer questions that the users may have?	
• Is there any problem-solving information provided, if necessary?	
• Does it cover all the problems users may be expected to encounter?	
• Does it provide solutions?	
Examples:	
• Are there sufficient examples?	
• Are the examples suitable?	
• Are examples presented consistently?	
Captions and callouts:	
• Are captions and callouts for illustrations, tables, photographs and other graphics effective and consistent.	
User-supplied content:	
• Can the users annotate the text?	
• If so, can the original text be restored?	

B.2 Navigation checklist

B.2.1 On-screen documentation

	Notes:
General:	
• Is it easy to find the documentation?	
• Are the access methods consistent?	
• Is it easy to exit from the documentation?	
Finding the right information:	
• Is context-sensitive access available?	
• If so, is the right information for the context displayed?	
• Are topics easy to find?	
• Is there an index facility, is it easy to use?	
• Is the index comprehensive, accurate and well-edited?	
• Does it make effective use of synonyms and cross-references?	
• Is it designed from the reader's viewpoint of how to access information?	
• Is the speed of access adequate?	
Structure:	
• Is the information structured in a logical way?	
• Is the structure of the information clear or else explained to the users?	
• Is the information structured in the way users need to understand it, rather than the way the program was written?	
• Is the material split into suitable chunks?	
• Is the order of information within each chunk sensible?	
Orientation:	
• Is it clear what type of information is displayed?	
• Does each topic have a clear heading?	

• Can users get back to a previous topic?	
• Can each user find the best starting point for their needs?	
Links:	
• Are different types of link clearly distinguished?	
• Are the links appropriate?	
• Are all the links necessary?	
• Are the links sufficient?	
• Do the links go to the correct places?	
Browsing:	
• Is it possible to browse through the on-screen information in a logical sequence?	
• Can users browse backwards as well as forwards?	
Scrolling:	
• Do the text and graphics scroll at an adequate speed?	
• Do titles and headings remain displayed?	
• Can users see how near the end they are in a scrollable topic?	

B.2.2 Paper-based and printable documentation

	Notes:
General:	
• Is it easy to obtain the documentation?	
• Does each user have the documents that they need?	
Finding the right information:	
• Is it clear in which documents to find each type of information?	
• Is it easy to find the right information within each document?	
• Is there a table of contents for each document?	
• Is each table of contents comprehensive, useful, accurate and well-edited?	
• Is there an index for each document in which users need to find particular information?	
• Is the index comprehensive, accurate and well-edited?	
• Does it make effective use of synonyms and cross-references?	
• Is it designed from the reader's viewpoint of how to access information?	
• Are references from the contents list and index easy to follow?	
Structure:	
• Is the information structured in a logical way?	
• Is the structure of each document obvious or else explained to the users?	
• Is each document or each part structured into chapters or major sections?	
• Is it easy to find the start of each part and each chapter or section?	
• Are appendices or annexes used appropriately for reference or lengthy information?	
• Is the information structured in the way users need to understand it, rather than in the way the program is written?	
• Is the material split into sections of a suitable length?	
• Is the sequence of information within each section suitable for the expected method of use?	

Orientation:	
• Is it clear what type of information is contained in each section?	
• Does each section have a clear heading?	
• Can each user find the best starting point for their needs?	
Sequence of information:	
• Are subjects presented in a logical sequence in each document?	
• Does the sequence of information in each document or each part correspond to the way users need to use it?	
Numbering:	
• For documents divided into volumes, are the volumes clearly labelled?	
• For documents divided into parts, are the parts numbered?	
• Are chapters or sections clearly numbered or labelled in some prominent way?	
• Do the page numbers in each volume or each part make it clear to which volume or part they belong?	
• Is the page numbering scheme suitable for the type of document?	
• Is the page numbering clear?	
• Is the page numbering scheme consistent with the section numbering scheme?	
• Is the section numbering clear?	
• Are illustrations that are referred to from the text numbered?	
• Are tables that are referred to from the text numbered?	
• Are preliminary pages numbered in a separate sequence from the rest of the document? (see E.2.3.2.1)	

Cross-references:	
• Are there cross-references between separate documents?	
• Are there sufficient cross-references between sections containing related information?	
• Are the cross-references appropriate?	
• Are all the cross-references necessary?	
• Are the cross-references easy to follow?	

B.3 Style checklist

This checklist should be applied to each separate document individually, that is, all the separate documents should be examined.

	Notes:
Vocabulary:	
• Are terms and abbreviations defined where necessary?	
• Are terms used consistently?	
• Are definitions provided for the terms?	
• Is the vocabulary suitable for the intended readership?	
• Are most words in the vocabulary short and simple?	
• Are definitions provided for terms?	
Grammar: NOTE When writing in the English language be aware of the differences between UK and US English (the default setting of the available spelling and grammar checker may be US English).	
• Is the spelling correct?	
• Is the grammar correct?	
• Are tenses used correctly?	
• Are instructions in the imperative?	
• Is the punctuation correct?	
Style:	
• Does the sentence length vary?	
• Are most sentences short and simple to understand?	
• Are most verbs active?	
• Is the reader addressed directly (in the second person)?	
• Is the writing concise?	
• Do sentences lead the user from the known to the unknown?	
• Do paragraphs lead the user from the known to the unknown?	

• Is the style appropriate for the intended readership?	
• Is the text easy to understand?	
• Are goals given for the instructions?	
• Are pre-requisite conditions given at the beginnings of instructions?	
• Is hyphenation used where necessary to avoid misunderstandings?	
• Are facts written the right way round?	
• Does each message explain what types of message it is?	
• Has giving the computer or the application an human personality been avoided?	
• Have clichés been avoided?	
• Are upper-case letters used appropriately?	
Illustrations:	
• Are illustrations used in appropriate places?	
• Are the types of illustration suitable for the information?	
• Are illustrations simple and clear?	
• Are illustrations of similar types consistent?	
• Are illustrations complete?	
• Is meaning conveyed using methods other than colour alone?	

B.4 Presentation checklist

B.4.1 On-screen documentation

	Notes:
Windows:	
• Does the windowing scheme accommodate the information characteristics the user needs?	
• Are the default sizes and placements of windows suitable?	
• Can users resize and move information windows?	
• Can the default sizes and placements be restored?	
Window layouts:	
• Are similar topics displayed in windows with similar layouts?	
• Are window layouts appropriate for the types of information they display?	
• Is there sufficient blank space so that the windows do not seem crowded?	
• Is the title of each window clear?	
• Are the navigation controls in a consistent place?	
• Are topic titles and controls contained in non-scrolling areas?	
Colours:	
• Are colours used appropriately?	
• Are colours used consistently?	
• Are only a few colours used?	
• Does the presentation work in monochrome?	
• Can the user change the foreground and background colours?	
• Is the polarity consistent (for example, dark text on light background)?	
• Is the contrast between foreground and background colours suitable?	
Typography:	
• Are only a few typefaces used?	

• Are typefaces used consistently?	
• Are the typefaces legible?	
• Are the typefaces used suitable for on-screen use (for example Verdana or Arial)?	
• Are the different levels of heading easy to distinguish?	
• Do the different heading levels represent the hierarchy correctly?	
• Is most of the text in mixed upper and lower case?	
• Are the typefaces available on the users' screens?	
• Are the upper-case letters at least 3 mm high on the screen?	
• Have underlines that cut through the descenders been avoided?	
• Is the method used for highlighting words and phrases consistent?	
• Is the method used for highlighting words and phrases effective?	
• Is the line spacing at least twice the x-height?	
• Is the line spacing at least one thirtieth of the line length?	
Layout of information:	
• Do important messages stand out sufficiently?	
• Are similar elements positioned consistently?	
• Is blank space used to separate elements?	
• Are paragraphs separated clearly?	
• Are lists used where appropriate?	
• Are lists consistent?	
• Are lists presented as numbered lists where the sequence is significant?	
• Are lists presented as bullet lists where the sequence is not significant	
• Are tables used appropriately?	
• Are rules and borders used consistently?	
• Is the presentation simple?	
• Can annotations be distinguished easily?	

Illustrations:	
• Is the speed of display of the illustrations adequate?	
• Can illustrations be seen at the same time as text that relates to them?	
• Are the drawn lines thick enough to be seen clearly?	
• Is text in illustrations sufficiently large (upper-case letters at least 3 mm high)?	
• Can users print the illustrations?	
• Are active areas clear and easy to use?	
Icons and signposts:	
• Are the icons and signposts self-evident or easy to learn?	
• Are the graphics clear?	
• Do the graphics represent actual objects or actions?	
• Do the graphics use culture-independent images?	
• Are metaphors used consistently?	

B.4.2 Paper-based and printable documentation

	Notes:
Documents:	
• Does the presentation of each document make it clear what the document contains?	
• Is the overall size of each document suitable for the place in which the document will be used?	
• Is the orientation of each document suitable for its intended use?	
• Do the majority of the documents have portrait layout?	
Binding:	
• Is the binding of each document suitable for the place in which the document will be used?	

• Is the binding of each document suitable for how the document will be used?	
• For documents with a printed spine, is the printing either across the spine or running from top to bottom?	
• For documents in ring binders, is there only one document in each binder?	
• Are cards and charts folded in a suitable way?	
Paper:	
• Is the paper suitable for the expected use of the document?	
• Are divider pages made of heavy paper or thin card?	
• Are divider pages durable enough for their intended use?	
• Is opaque paper used in documents printed double-sided?	
• Does the paper used for documents that are not leaflets or brochures have a matt finish?	
• Is the material used for pages suitable for the environments in which they are to be used (especially for damp or dirty environments)?	
• Unless there is a special need for another colour, is the paper white?	
Page layouts:	
• Are page layouts simple?	
• Are page numbers clear?	
• Is the document reference number included on the pages?	
• Are common elements on pages presented in the same position on each page?	
Section layouts:	
• Do major sections start on a new page?	
• Is the heading of each section clear?	
• Are sections clearly distinguished from each other?	
• Are similar subjects displayed using similar section layouts?	
• Are section layouts appropriate for the types of information they contain?	
• Is there sufficient blank space so that pages do not seem crowded?	

Layout of information:	
• Are line lengths short enough for the text to be easy to read?	
• Do important messages stand out sufficiently?	
• Are similar elements positioned consistently?	
• Are paragraphs separated clearly?	
• Are lists used where appropriate?	
• Are lists consistent?	
• Are lists, where the sequence is significant, presented as numbered lists?	
• Are lists, where the sequence is not significant, presented as bullet lists?	
• Are tables used appropriately?	
• Are rules and borders used consistently?	
Colours:	
• Are colours used appropriately?	
• Are colours used consistently?	
• Are only a few colours used?	
• If colours are used in printable documents, is the presentation clear when printed in monochrome?	
• Is the majority of the text presented as dark characters on a light background?	
• Is the contrast between text and paper colours suitable?	
Typography:	
• Are only a few typefaces used?	
• Are typefaces used consistently?	
• Are the typefaces legible?	
• Are the different levels of heading easy to distinguish?	
• Do the different heading levels represent the hierarchy correctly?	
• Is most of the text in mixed upper and lower case?	

• For printable documents, do the typefaces print correctly on all printers available to the users?	
• Is the text size suitable for the environment in which the document will be used?	
• Is the method used for highlighting words and phrases consistent?	
• Is the method used for highlighting words and phrases effective?	
• Is the line spacing sufficient?	
Illustrations:	
• Do illustrations occur after the text that refers to them?	
• Can text and illustrations that need to be together be seen at the same time?	
• Are the drawn lines thick enough to be seen clearly?	
• Is the text in illustrations legible?	
• For printable documentation, do the illustrations print correctly, with all drawn lines and text visible?	

B.5 Overall

	Notes:
Does the documentation fulfil its objectives for the intended audience?	
Does the documentation integrate all elements into a readable and usable publication?	
Does the documentation project a professional image of the publication's sponsor?	

Annex C (Informative) Evaluation of documentation

C.1 General

This annex is concerned with evaluating new documentation for application software; the ISO/IEC 9126 series may also be used for this purpose. The information can also be used to evaluate existing documentation to determine:

- whether it is suitable for a given purpose, for example, whether the documentation is suitable for use in a particular organisation for a particular group of users;
- what enhancements or alterations should be made to the documentation in the future, to enhance its qualities (see C.4);
- whether it meets the objectives set for it.

Most evaluations of documentation are carried out as part of an evaluation exercise for a complete application, and there will be the following two activities in the evaluation exercise:

- evaluating the entire application, including all its documentation both on-screen and on paper;
- evaluating the on-screen documentation elements independently.

This annex contains suggestions for:

- a procedure for planning and carrying out an evaluation (see C.2);
- a set of viewpoints from which the documentation may be evaluated (see C.3);
- a set of qualities for which an existing documentation system can be tested (see C.4);
- evaluation methods (see C.5).

These methods are used in the reviews described in clause 7 and can also be used in the post-production evaluation described in 8.2.

C.2 Procedure

To evaluate existing documentation, the following three stages should be carried out.

- Draw up statements about the required levels of performance or quality. Consider all the qualities in C.4 from all the viewpoints in C.3, selecting all those that are relevant. The statements should be specific and measurable.
- Prepare for the evaluation exercise by specifying what measures will be used, how the evaluation will be carried out, how the results will be recorded and how they will be analysed.
- Evaluate the documentation against the statements, using evaluation techniques from C.5, taking measured results and analysing them.

If the documentation was designed and developed following the guidance in this International Standard, the objectives set for the system as described in clause 4 will be useful as follows.

- For the existing documentation, to evaluate whether or not it meets its objectives.
- For future planning, as a basis for setting the objectives for future versions.

C.3 Viewpoints

The quality of the documentation should be looked at not only from the user's point of view. For example, the manager of a group of users may have different requirements for the documentation from that of the users themselves. Similarly, the developers will have yet another different view. All these views are important and they should all be considered. A list of viewpoints relevant to the documentation being evaluated should be drawn up. The following are some suggestions.

- **User's view.** Users are likely to measure quality in terms such as:
  - Does it give me all the information I need?
  - How reliable is it?
  - Is it easy to use?
- **Users' manager's view**. Managers may be more concerned with overall quality than with specific quality characteristics, and so will probably assign different weights to certain characteristics to reflect the business needs of the organisation.
- **Developer's view.** Developers can have the same views of quality as the users, because they need their systems to meet the users' requirements. However, developers are also concerned with the quality of intermediate products as well as of the finished product.

  NOTE However, developers are very close to the product and are more sophisticated users.
- **Maintainer's view.** Those who will have to maintain the on-screen documentation systems will have special requirements for quality in addition to those of other developers. They will be concerned with, for example, the simplicity and clarity of the documentation structures and the ease with which new versions of the documentation systems can be created.

It is also useful to test the quality characteristics of the existing documentation in relation to the current state of technology.

C.4 Quality characteristics

To evaluate documentation, it is important to decide in advance what features or qualities are required. To help in drawing up such lists, the following list of software qualities, adopted from ISO/IEC 9126, can be used. For each quality characteristic, a set of possible sub-characteristics under which the characteristic may be considered is given. Although the words used for these sub-characteristics are complicated, the meanings they convey may be helpful.

Use whichever of the following apply to the type of document being evaluated; they may all be appropriate for on-screen documentation but may not all be appropriate for paper documentation.

- **Functionality:** what does it do? This concerns the degree to which the documentation provides users with what they need. The following sub-characteristics may be useful:
  - suitability;
  - accuracy;
  - interoperability (working with other applications, the operating system, and so on);
  - security;
  - compliance.
- **Reliability:** how reliable and resilient is it? This concerns how well the documentation maintains its level of performance under different conditions over periods of time. The following sub-characteristics may be useful:
  - maturity;
  - fault tolerance;
  - recoverability;
  - compliance.
- **Usability:** how easy or hard is it to use? This concerns the effort needed to use the documentation and users' views about the documentation's ease of use. The following sub-characteristics may be useful:
  - understandability;
  - learnability;
  - operability;
  - attractiveness;
  - compliance.
- **Efficiency:** how efficient in time and resources is it? This concerns the level of performance of the documentation. The following sub-characteristics may be useful:
  - time behaviour;
  - resource behaviour;
  - compliance.
- **Maintainability:** how easy is it to maintain? This concerns the effort needed to make alterations to the documentation. The following sub-characteristics may be useful:
  - analysability (how easily it can be analysed for updating);
  - changeability;
  - stability;

- testability;
- compliance.

- **Portability:** how easy is it to transfer. This concerns transferring the documentation from one environment, organisation or platform to another. The following sub-characteristics may be useful:
  - adaptability;
  - installability;
  - replaceability;
  - co-existence;
  - compliance.

C.5 Evaluation methods

This clause lists some methods that can be used in evaluating documentation. The method selected will depend on a variety of factors including:

- the reasons for carrying out the evaluation;
- the phase in the development 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.

In all cases, the evaluation should be carried out to measure the features specified in the statements drawn up as advised in C.2.

Some suggested methods are the following.

- **System test.** Embedded documentation can be tested as described in 7.3.4. The aim of the tests is to ensure that the documentation performs as required. Tests are normally carried out by members of the project team during development. They are conducted following test scripts, which are drawn up in advance to ensure that the tests are systematic and complete.

- **Review.** The information contained in the documentation can be reviewed as described in 7.3. The checklists in annex B may be useful. Reviews of the information itself can be carried out in part using a printed copy of the information. The navigation and presentation are reviewed using the documentation on the screen, along with the application. The reviews will help determine some but not all of the required qualities.

- **Documented evidence**. Evaluation is carried out by examining the analysis and design definitions of the documentation and comparing them with the requirements. This is useful for checking whether and how particular requirements apply to the documentation.

- **Observation.** Observation means simply to inspect the documentation to see whether or not particular features are present. Observation can be carried out by anyone who has the necessary skill to check the documentation systematically against a set of requirements.

- **Analytical evaluation.** Evaluation is carried out by experts. They make informed judgements about the properties of the documentation. The experts need to be experts in the properties they are evaluating.

- **Empirical evaluation**. Evaluation is carried out by experts observing users (who are not members of the project team) using the documentation and analysing the users' behaviour. This method is useful for evaluating those features of documentation that can only be evaluated in the context of use. A range of different techniques can be used, such as an observer sitting with the users, videoing the users, asking the users to explain what they are doing and why. This method is also useful for evaluating systems that are partially developed, while there is still time to change the documentation.

- **Surveys.** Opinions and attitudes are gathered using surveys, which may include interviews with users and others and may include the use of questionnaires. The data gathered is then analysed. This method is particularly useful for evaluating released products.

Annex D (Informative) Writing style and techniques

NOTE The guidance in this annex refers extensively to 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.

D.1 General

This annex covers the writing and illustration styles and techniques appropriate for user documentation.

It is essential that user documentation is clear, unambiguous and easy for users to understand, which means that:

- the writing and illustrations should be simple and clear;
- the vocabulary should be straightforward;
- all terminology should be explained simply.

D.2 Conventions

Set conventions for the whole product in the following areas.

- **Technical terminology.** Where appropriate, use the same terminology as that used for documenting the hardware and the operating system. For software-specific items, use the same terminology in the documentation and in displays generated by the software. However, beware of being too technical: know your audience.
- **Icons.** If the application uses icons, the on-screen documentation should explain what they represent. The documentation may use the same icons to represent the same objects. The documentation should not use those icons to represent any other objects.
- **Symbols.** If the application uses symbols, the same considerations apply as for icons.
- **Controls.** Where controls are needed within the on-screen documentation, for example for navigation, they should obey the conventions agreed for the documentation (see 9.16.5.15).
- **Messages.** Follow established software platform conventions and International Standards for safety notices and messages (see 9.9).

D.3 Vocabulary

Write the documentation using simple vocabulary familiar to the users.

Define a product glossary. Use it for the software and the documentation. The glossary may be used as one source of information for a glossary provided for users, and for information fragments such as field definitions and names and uses of interface elements. However, the users' information should define vocabulary in user's terms, not in developer's terms.

If specialist vocabulary is used in a particular application area, the documentation should use that specialist vocabulary rather than different general usage vocabulary. If there is a known user vocabulary, the on-screen documentation can include definitions of it.

Check the meanings of words using a standard dictionary.

Take particular care to avoid vocabulary that is sometimes used imprecisely. The precise meanings of terms that are frequently misused should not be relied upon, because the user might assume that the misused meaning is intended, rather than the precise meaning.

Do not invent words, for example, by turning a noun into a verb, such as 'moused' to mean 'clicked the mouse button' or 'characterise' to mean 'make into a character'.

If there is a danger that a term might be misunderstood, do not use it. If words that have more than one meaning are used, make clear which meaning is intended.

Where the same meaning is to be conveyed, use the same words; avoid elegant variation, where different vocabulary is used simply to make sentences sound more interesting. Using different words can 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.

Avoid the following:

- '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*.

D.4 Terminology

The terms used in the documentation should be easy for users to understand and should be used consistently. The following guidelines should help to achieve this.

- Use industry-specific terminology correctly. For example, in process descriptions use the job titles that are used in the target industry.

- Introduce as few new terms as possible. If new terms are needed, make them short, easy to remember and pronounceable.

- Avoid jargon. Where an accurate English alternative to jargon exists, use it instead of the jargon term. If no alternative exists, define the correct terms.

- If industry-specific or other common acronyms are used, define them in the documentation. If the documentation is on-screen, allow users to display the definition each time an acronym is used. Do not create new acronyms simply for the purposes of the documentation.

- Use terminology that is common in the user's environment or the application area and use it correctly. This means that the author should investigate the user's environment or the application area fully.

Avoid terminology that is common in the author's environment but not in the user's. If there is doubt, conduct a survey of users, or consult an authority in the user's environment.

D.5 Writing style

D.5.1 General

For each fragment or complete topic of information, consciously select the best technique to use. Consider using:

- text;
- a list;
- a table;
- an illustration.

Each piece of text should use the appropriate writing style and layout. This sub-clause gives guidance on writing style; D.6 gives guidelines for writing technique; D.7 and D.8 cover some aspects of layout. Guidelines for illustrations are in D.9.

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.

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 following are useful classifications of types of writing style needed in on-screen documentation:

- names (see 9.7.1);
- instructions (see D.5.2);
- descriptions and explanations (see D.5.3);
- facts (see D.5.4).

Each piece of text will be constructed from combinations of these classes. A task description, for example, may require text of all these types:

- the name of the task;
- an explanation of what the task is intended to achieve;
- the instructions for the actions to be taken to carry out the task;
- facts to be used in the actions;
- descriptions of what the user should observe.

Decide for each piece of information what type of information it is, and hence what writing style should be used for it.

D.5.2 Instructions

It should be clear what each set of instructions will achieve, what the user has to do and what the effects will be. Do the following.

- Explain the goal first. The goal activates the user's mental model for the task and so helps the user to interpret the instructions and to anticipate possible instruction steps. The goal holds over several sentences.

- Give the instructions using the imperative, for example:

  'Press Enter'

  'Check that the totals are displayed and then type SUM'

- If conditions are associated with an action, give them at the beginning of the sentence, separated from the rest of the sentence by a comma. For example:

  'If the light is still on, do not remove the diskette'

- Use the present tense (see D.6.6).

- Number instructions that have to be carried out in a strict sequence. Do not number instructions that do not have to be performed in a strict sequence.

- Include descriptions of what effect carrying out each instruction should have. Where it is helpful, explain why particular actions are being taken.

D.5.3 Descriptions and explanations

Use a direct writing style and simple vocabulary for descriptions and explanations.

Concentrate on the information and understanding that the user needs, rather than on what information there is available about a subject.

Take the user from what is known to what is unknown. This principle applies at the sentence as well as the topic level.

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 the user has to do and which the system does.

D.5.4 Facts

Facts are simply special cases of descriptions or instructions. They are separated out to emphasise the difference.

Write facts taking into account why users need them, so that users are taken from what they know to what they want to find out. The difference is important, if users are not to waste time looking for what they need.

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*').

D.6 Writing techniques

D.6.1 General

Take the following into account when writing on-screen information:

- the size of the window's information area. Because the space is constrained, it is important that the writing is succinct. The following techniques are recommended:
  - 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.
- the lack of a visible framework into which the text fits, such as a previous page, next chapter and so on. Each topic needs to be self-contained; authors cannot assume that users have read or can see any other topics.

On-screen documentation should be clear, unambiguous and easy to understand. Follow the guidelines in D.6.2 to D.6.14.

D.6.2 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 D.7 and D.8). In general, use shorter paragraphs in on-screen documents than would be used in printed documents.

It takes users longer to read sentences that refer to information introduced several sentences earlier than those that refer to recently introduced information. So keep related ideas together.

Keep references to other parts of the same topic 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 can follow the explanation. For example:

> 'If you want to save your work, make sure that the diskette is in the disk drive. This is because the software ...'

It is also helpful to use similar techniques to link related paragraphs.

D.6.3 Sentences

D.6.3.1 Sentence structures

Sentence structures should be simple, but a variety of sentence structures should be used. Avoid complicated grammatical constructions, because users may find them difficult to understand. Also, avoid lengthy sentences that would make translation difficult.

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 any new information. Where the user has discretion over whether or how actions are carried out, construct sentences to 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 can 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'

D.6.3.2 Sequence of ideas

It is essential that sentences present actions and effects in exactly the sequence in which they should take place.

In most cases, if information is presented in the wrong sequence it may simply annoy the user.

In some cases, however, if information is not presented in the correct sequence, it can 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:

‘Save the file. Exit from the program by selecting Exit'

D.6.3.3 Hanging participles

Take care to avoid hanging participles because they can be misleading. For example, in the following, the participle 'saving' is hanging, meaning that it is not related to anything.

'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 are saving a file, you should keep details of the date and time'

'When the system is saving a file, it also saves details of the date and time'

In some cases, a hanging participle can cause serious misunderstanding, because the participle appears to relate to a nearby noun. For example, in the following, it appears as if the administrator is the person who is noting down the details, although the intention is that the user should note them down:

'After recording details of the failure, the administrator ...'

Instead, the following should be used:

'After you have recorded details of the failure, the administrator ...'

D.6.3.4 Tautologies and redundant phrases

Avoid tautologies and redundant phrases because, although they do not often lead to misunderstandings, they can make sentences more complicated than necessary. For example, in the following tautology the prefix 'Re' and the word 'again' say the same thing:

'Re-enter the information again'

The following could be used instead:

'Enter the information again'

D.6.3.5 Articles and pronouns

Articles and pronouns should not be omitted because this makes sentences difficult to parse. 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'

D.6.3.6 Positive and negative constructions

Where possible, use positive rather than negative constructions, unless a negative is being 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'

D.6.4 Conditions

Explain all conditions clearly with the condition first, separated from the rest of the sentence by a comma. If the comma is not included, 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 emphasise the multiple conditions using terms such as 'both', 'at least one of', 'either one of' and 'only one of'.

For complicated conditions, authors should use a list or a table (see D.7 and D.8), rather than giving all the conditions in words. For example, the information in Table D.1 is much easier for the user to understand when presented in a table than it would be in continuous text.

Table D.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 can be used to illustrate complicated conditions, but the author should assess whether the target users will be able to understand them (see D.9).

D.6.5 Active and passive voice

Where possible, 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 colour to red, you will not be able to see the warnings'

would be clearer if written as:

'If the background has been changed to red, the warnings will not be visible'

D.6.6 Tenses

Pay careful attention to the use of tenses, taking into account that:

- users might be following instructions one by one, so all 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:

a) Select the Change User Group function. A dialogue box is displayed showing your current user group

b) From the list of user groups displayed in the New Group field, select ...

c) ...'

Future: 'The system will use the name of the new user group in the titles of all future reports'

D.6.7 Singular and plural verbs

A verb has to 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, ...'.

Avoid using plurals unnecessarily, because they can 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.

D.6.8 Punctuation

Simple, accurate punctuation makes written English easy to understand.

- Use punctuation consistently, with the sole aim of making the meaning of the text clear.
- Examine long sentences carefully and, where possible, break them down into shorter ones. If the resulting separate sentences need to remain linked, separate them by semi-colons, and not by commas.
- Do not use parentheses to enclose essential information because this might reduce the apparent importance of that information.
- Do not use square brackets or braces in text, although they may be necessary in mathematical or syntactic information. In the latter case they should be used strictly according to the rules of the relevant discipline. (Square brackets and braces may be used as ways of highlighting active text, see 9.16.5.6).
- Use all single or all double quotation marks to enclose quotations. Otherwise, neither single nor double quotation marks should be used unless the software uses them and the documentation needs to include copies of what the software displays or uses. In particular, quotation marks should not be used to enclose any 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 9.16.5.12).
- Be aware of the differences between UK and US English, especially where quoted speech occurs at the end of a sentence.
- Do not use short-hand methods for covering several options in one construction. They are difficult to understand and can cause confusion.
- Avoid the use of 'and/or', and the use of a solidus (/ character) in any other phrase to cover several options in one construction. For example, instead of:

  'You can now save the file and/or exit from the system.',

  give the information in full:

'You can now:

1 Save the file if you wish

2 Exit from the system

To save the file ...

To exit from the system….'

- Do not attempt to cover both the singular and plural options in a single sentence using brackets to enclose the alternatives. For example, instead of:

'click on the names of the images that you want to display'

use:

'click on the name of each image that you want to display'

D.6.9 Hyphenation

Hyphenation is used to join two words together so that they act as a single word.

Use hyphens wherever they are helpful, even if in some cases they are not 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'

D.6.10 Infinitives

Where an infinitive consists of two parts, that is, the word 'to' and the verb, the two parts should normally be kept together. Although infinitives can usually be split by a single word without causing the user any problems, split infinitives can annoy some users, and for that reason they should be avoided. Do not split infinitives by a whole sequence of words because the text might be difficult to understand.

D.6.11 Upper-case letters

The following recommendations for use of upper-case letters should be used for both the user interface and the documentation.

- Keep upper-case letters to a minimum.

- In names of products and companies, use upper-case letters exactly as in the registered product and company names.
- For titles and headings, including headings of columns in tables and lists, use either of the following conventions, but use it consistently.
  - Use an initial upper-case letter for only the first word.
  - Use initial upper-case letters for all significant words.

D.6.12 Anthropomorphisms

Avoid attributing human personality to the computer or application, particularly in on-screen documentation, because this can strengthen 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.

D.6.13 Analogies and metaphors

The use of an analogy or a metaphor can 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 itself; 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 localise correctly, so, if the product will be translated or localised, metaphors should be checked to ensure that they are suitable, or they should be avoided.

D.7 Lists

Lists are useful for:

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

All the entries in a list should have the same construction. For example, all the instructions in a list should use the imperative.

Use a lead-in sentence. If each list item starts with the same word, make that word part of the lead-in sentence.

For guidelines on the presentation of lists, see 9.16.5.7.

D.8 Tables

Tables should be used where the information is easier for the user to absorb in tabular form than in continuous text, and where the information consists of sets of related items from which the user may need to choose one set.

For guidelines on the presentation of tables, see 9.16.5.8

D.9 Illustrations

D.9.1 When to use an illustration

Consider using illustrations:

- 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 describe something. Sometimes an illustration is the best way of describing 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.

D.9.2 Types of illustration

For each illustration, decide which of the following types is appropriate:

- a picture, which shows 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:
  - organisation chart or a tree structure;
  - graph;
  - bar chart or pie chart;
  - flowchart.

D.9.3 Styles for illustrations

Use the following principles for the style of illustrations.

- Keep them simple. Focus on the purpose of the illustration, avoid unnecessary information and unnecessary levels of detail. Illustrations in on-screen documentation should be as simple as they can be while still conveying the information.

- Consider the quality of the display screens. Illustrations should be of an acceptable quality when displayed on all the types of display screen that may be used. Drawn lines may need to be thicker than they would be for an equivalent illustration on paper. Ensure that any characters are clearly readable. Conduct trials to test illustrations on all types of screen that may be used.

- Make them clear. Pictures should be easy to recognise. Add text, if necessary, to clarify the meaning.

- Make them consistent. For a series of illustrations, establish a house style to ensure consistency of line thickness, colours, hatching, and fonts. Illustrations in a series should all have a consistent level of detail.

- Make them complete. The whole of each illustration should be visible at once. 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.

- Do not use colours to convey meaning. If illustrations use colours, the colours should not be the only way of conveying certain meanings; some other method should be used as well, such as different types of shading or hatching. Carry out tests to ensure that the illustrations are clear when displayed on a monochrome screen or viewed by people who are colour blind. Consider that a screen may be printed out on a monochrome printer.

- If the product will be translated, give attention to the text on illustrations. The translated text may not fit in the same area as the original text; the process for translating the text on diagrams may be longer and may cost more than translating ordinary text. It is recommended that any illustrations used only for cosmetic purposes do not contain text if the product will be translated.

- If the product is to be used by an international audience, culture-bound images should not be used.

9.18 contains further guidance on the presentation of illustrations.

Annex E (Informative) Design and preparation of printed information

E.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.

E.2 Design

E.2.1 Decide how the printed documents will be produced

Consult several printing organisations for advice, prices and time scales for different methods of production. Get samples for every solution that is being considered.

Table E.1 shows some methods of producing multiple copies, giving the main features of each.

If reproduction is to be carried out by a third party, allow extra time for transferring documents.

Table E.1 - Methods of producing multiple copies

Method	Quality	Speed and convenience	Cost
Printing on local printer	Depends upon the quality of the printing device used	Very convenient Useful where only a few copies are needed Not always fast Very easy to make last minute changes, by simply replacing pages	Depends upon the type of printing device and labour 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 last minute changes, by simply replacing pages	Can be cheap, especially for black and white only and for small quantities
Offset lithography	Results are excellent Any effects can be achieved Often preferred for colour	Needs a specialist printing organisation Can take a long time	Can be expensive for small quantities

	printing		

E.2.2 Hierarchic structure of documents in the form of books

Structure user documents in the form of books in one or more of the following ways:

- split over several separate numbered volumes;
- divided into numbered parts;
- each document or part divided into:
  - chapters or main sections for the body of the document;
  - appendices or annexes at the back for reference information or lengthy information that is needed for several chapters or sections in the body of the document;
  - chapters, main sections, appendices and annexes divided into subsections, either numbered or unnumbered. Restrict the number of levels of subsection to two wherever possible, so users know where they are in the document.

E.2.3 Style and presentation

E.2.3.1 Structure of comprehensive reference documents

Use divider pages to distinguish different sections or chapters, or patches of black or a colour on the edges of pages (bleeding tabs) at different vertical levels for different sections.

Number sections. For commands or procedures with short names, use Alphabetical or numerical sequence.

NOTE If Alphabetical sequence is used and the documentation is translated, the sequence of the items might change.

Consider the following for each section:

- start the section on a new page;
- put large headings on the edges of pages to help users to find the procedures that they need to use.

Do not repeat detailed information; give it once and refer to it when it is needed.

Include an index in all comprehensive reference documents. Include a bibliography and a glossary where useful.

NOTE The word sequence in a glossary will be changed by translation.

E.2.3.2 Numbering schemes for pages, sections, illustrations and tables

E.2.3.2.1 Page numbers

For each document, select an appropriate method of numbering pages. For example:

a) Number all the pages of a document sequentially with a simple Arabic page number, starting from 1. For any 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.

b) Number pages within each major section and include with the page number, the number, letter or name identifying the section. This style is known as "folio-by-chapter": it has the advantage of clearly showing which chapter the reader is in. Separate the two parts of the page number 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

   This two-part numbering is recommended for loose-leaf publications.

   Number appendices or annexes in the format:

   A-1, A-2, …,B-1, …

c) If the document is divided into physically separate volumes, include the volume number with the page number, for example:

   Volume 1, 1-1, Volume 1, 1-2, ...

d) Number the preliminary pages of a document, that is the pages containing the preface and the contents list, in a separate sequence from the rest of the document; for example, using Roman numerals:

   i, ii, iii, iv ...

e) Number all pages, 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 counts as pages i and ii.

f) Number all pages 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. This form of page numbering can be used with methods a), c) and e). The benefits in using this method are:

   - The reader soon knows if pages have been missed off the end, and
   - It can make the document template design easier, as the same page numbering method is used for both the preliminary pages and the body.

E.2.3.2.2 Section numbers

Ensure a clear distinction between the page numbers and the section numbers used.

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:

- use Arabic numbers;
- number main sections sequentially starting from 1;

- divide each main section into any appropriate subsections (second level) and number these sequentially within the highest level, for example, 1.1, 1.2. Continue this method of subdivision to further levels, for example, 1.1.1, 1.1.2. Whenever possible, restrict the hierarchy to no more than three levels of subdivision;

- separate the numbers designating subdivisions of different levels with a full stop (period); do not use a full stop (period) after the final number.

E.2.3.2.3 Illustration and table numbers

Number illustrations and tables, unless they are all in-line in the text and are not referenced from elsewhere.

When numbering is used:

- use separate numbering sequences for illustrations and for tables;

- if the pages of the document are numbered within sections, number the illustrations and tables within the same level of section. If the pages of the document are numbered straight through, number illustrations and tables straight through the document;

- include the type of item in the titles of illustrations and tables, for example, Figure 6 or Table 10. Use consistent titles throughout the documents.

The following are examples of illustration and table numbering schemes:

Table 1, Table 2, ...

Figure 1, Figure 2, …

Illustration 1-1, Illustration 1-2, …

Table 1.1, Table 1.2, … .

E.2.3.3 Illustrations of printed output

Consider the two commonly used methods for preparing illustrations of printed output.

- Use actual output as master artwork; it can be photocopied and reduced and then used directly or it can be scanned into an electronic system and reproduced electronically.

- Create text that resembles the printed output and include it in the master artwork.

If illustrations of printed output will be included, take into account the reason for including the output.

Wherever possible, present illustrations of printed output 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, rotate the text 90 degrees anti-clockwise (landscape), so that the top of the report is on the left-hand side of the page.

E.2.3.4 Page layouts

E.2.3.4.1 Page size and orientation

For electronically supplied documents that users may subsequently decide to print, take into account the paper sizes likely to be available to users in the different countries where the product will be used.

For pre-printed documents, work with the printing organisation to choose a page size and orientation that will be convenient in the users' environment; 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.

- Use a suitable A size that is similar in size to the initial approximation. ISO 216 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 can be created using a simple fraction of an A size, 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, define it, but check details of the production method and cost with the printing organisation that will produce the documents.

  NOTE 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 travelling, decide whether or not the same document can be used for all the different needs. If so, design the document so that the text will be legible in all the different sizes. If not, design different versions of the documentation to satisfy the different needs.

Orient pages with either the long edges on the left and right (portrait) or at the top and bottom (landscape).

Portrait documents bound along the long edge are the most usual; use them 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; use them only if there is a special user need.

Both portrait and landscape documents bound along the short edges are extremely inconvenient and are not recommended for application software documents.

E.2.3.4.2 Basic page layout

Use simple page layouts that can easily be achieved using the chosen method for preparing master pages.

Prepare a page grid for each type of page. The different types of pages are:

- normal text pages;

- imprint page;

- title page;

- contents list;

- start-of-section pages;

- index.

When preparing the page grids, 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.

On each page grid:

- show where the margins of the pages will be;

  NOTE Be aware of the different UK and US spacing for drilling;

- make the margin on the inside edge of a page wide enough to accommodate the method of binding. In the simplest case, the left and right margins of all pages can be the same. However, consider using a wider margin on the inside edges of pages, that is, the left of a right-hand page and the right of a left-hand page. Figure E.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 E.2 shows an example of an A5 page set using the page grid in Figure E.1;

- show the number of columns to be used for the text. For normal text pages, a single column for text is usually appropriate, although awareness documents in the form of leaflets or brochures might benefit from a two-column or a three-column layout. Determine the width of the text column in association with the typography of the text;

- 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, indent the body of the text from the left margin;

- for the index, use two or three columns of text to reduce the number of pages used;

- show the placement of illustrations and tables. Make the placement strategy as simple as possible, so 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;

- decide how cards and charts will be folded. Take 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;

  - the sizes of other documents with which it will be packaged.

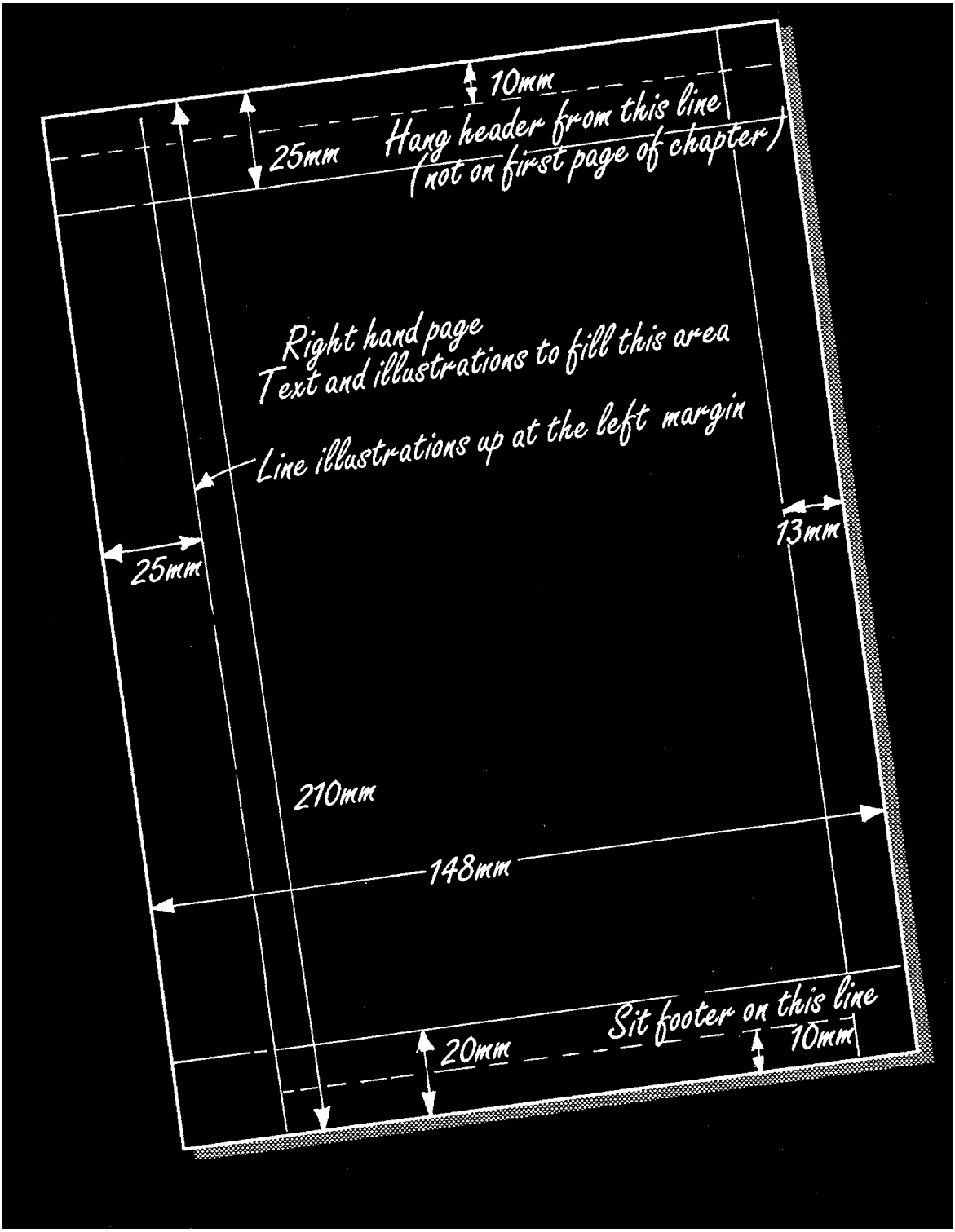
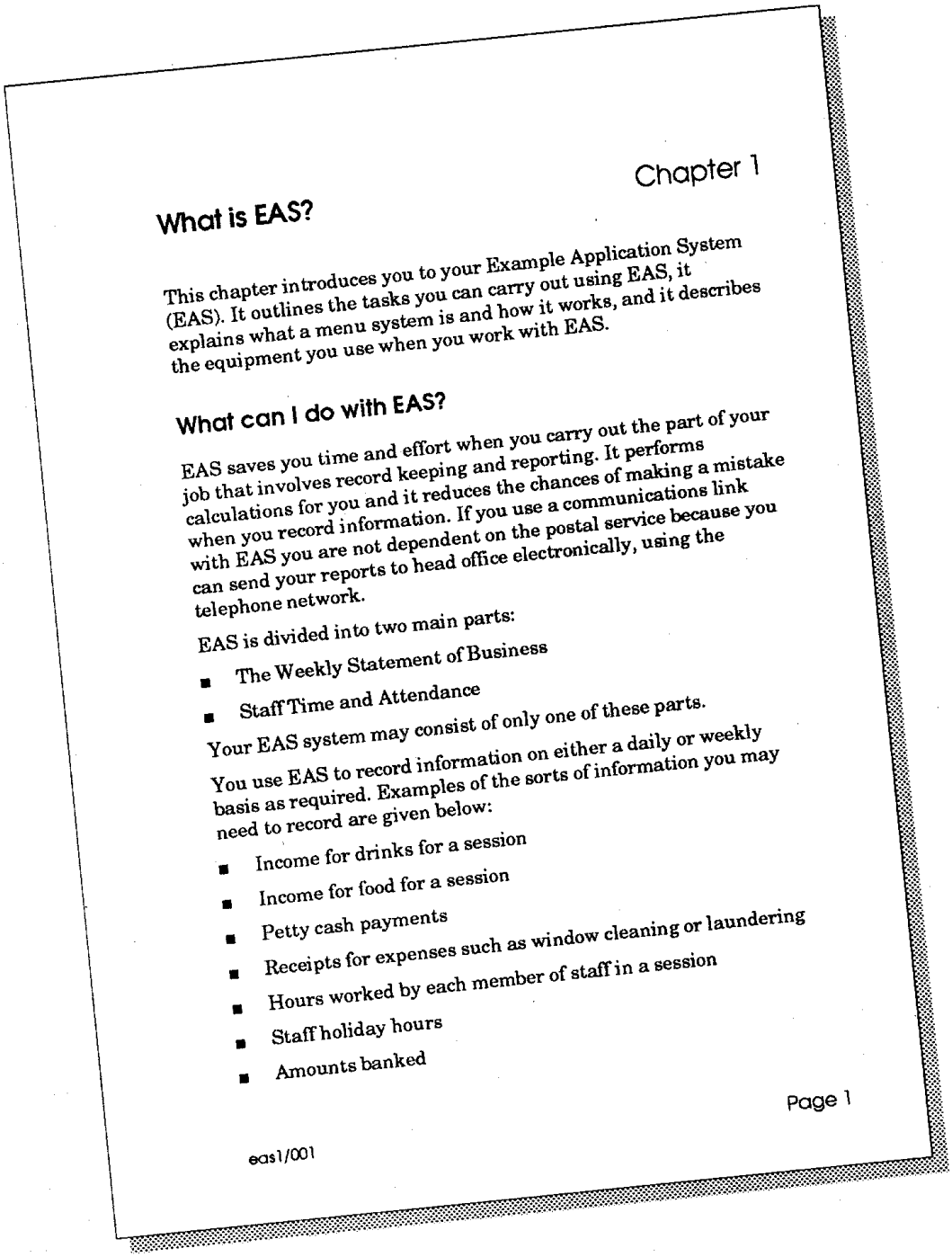


Figure E.1 - Example A5 Page Grid



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 E.2 - Example of an A5 page

E.2.3.4.3 Basic page elements

On the page grid, define the precise positions of the following basic elements that will appear on all 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 can be used to separate header and footer text from the body of the page.

NOTE Avoid the use of vertical lines unless there is a special need for them.

For headings, distinguish the different levels from each other and from the text.

- Consider using different styles, sizes and weights of character.
- Consider representing the hierarchic structure with vertical spacing.
- Use horizontal spacing with care, to avoid making insignificant headings appear more important than significant ones. This can happen if section numbers are aligned on the right and not on the left, with longer numbers looking more prominent than shorter numbers.

E.2.3.5 Use of colour

Consider specialist advice about the use of colour from, for example, a publishing house or a printing organisation.

Use colour only to guide the user, to emphasise certain text or to decorate a document. Do not use colour to convey meaning, unless it is inherent in the software application.

When using colour, follow these guidelines.

- Choose a method of production capable of handling colour. Consistent reproduction of colour can be difficult; some methods distort colour more than others.
- Test which colours and combinations of colours reproduce best and are least likely to cause confusion.
- Colour reproduction can be expensive. Verify cost estimates for reproduction with printing organisations before making a decision.

E.2.3.6 Presentation of text

E.2.3.6.1 Footnotes

Avoid footnotes if possible. Place the information in the text where it is accessible. A smaller type size or different font may be used to indicate notes in text.

E.2.3.7 Presentation of illustrations

If illustrations are too wide to fit on a page (even after reducing):

- if users need to refer to the illustration from different parts of the text, for example, a keyboard layout chart, present a fold-out page;
- rotate the illustration 90 degrees anti-clockwise.

Place each illustration after the text that refers to it, preferably visible at the same time; 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 page, leave white space and move the illustration and text onto a new double-page spread.

E.2.4 Production phase

E.2.4.1 Quality and durability of the finished documents

E.2.4.1.1 Weight of paper

Choose a weight of paper that:

- suits the document type and size;
- can carry both a strong image and fine detail.

For large reference documents, use thin paper to keep down the weight and thickness. For awareness documents, use thick paper.

Paper weights of between 80 grammes per square metre (gsm) and 130 gsm suit most user documentation needs.

For divider pages, use heavy paper or thin card, for example, 300 gsm board.

For double-sided documents, use opaque paper so that print does not show through from one side to the other.

Take advice from a printing organisation or a paper supplier. Test samples using the chosen production method.

E.2.4.1.2 Paper surface

Use paper with a matt finish for user documents such as tutorial or reference documents. However, for brochures and leaflets, consider a gloss finish to 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, consider laminating the pages or using plastics. Give similar consideration to divider pages, particularly tabbed dividers. Use matt laminates wherever possible, to avoid glare.

E.2.4.1.3 Paper colour

Unless there is some special user need, use plain, white paper.

E.2.4.2 Binding

To select the type of binding a document should have, 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?

Consult a printer to determine what sort of binding is suitable for each document, based on the answers to the questions above.

Bind documents individually. Avoid, for example, two documents in the same ring binder. (Users perceive the contents of one ring binder to be one 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 and local conventions may determine the orientation. The spine information should include the title, volume number and name of the document; consider also the name or logo of the company. Recommendations for the presentation of spine titles are given in ISO 6357.

E.2.4.3 Submit master pages for production

Before master pages are sent for production, agree the following with the organisation that will produce the documents:

- Number of copies of each document (see 4.2.3)
- Exact page size for each document (see E.2.3.4.1)
- Type of paper to be used (see E.2.4.1.1)
- Details of any lamination or other special finish needed for any of the pages (see E.2.4.1.2)
- For cards and charts, folding instructions (see E.2.3.4.2)
- Type of binding for each document and details of any cover and spine information (see E.2.4.2)
- Instructions for use of colour (see E.2.3.5)
- Instructions for collating documents, for example where divider cards should be included (see E.2.3.1)

When master pages are submitted for production, agree the following with the organisation that is to produce the copies:

- 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 documents
- Cost

Archive securely electronic and paper copies of master pages of all text and illustrations, so a new set of master pages can be created if the original set is lost or damaged.

Check complete copies of the documents.

E.2.4.4 Package the documents and software

Consider packaging several documents in open-fronted boxes (slipcases), and consider packaging single documents in this way too.

Slipcases are often used to hold ring-bound documents, but consider them for all sorts of bindings if they are used to hold the full set of documents for one type of user. Cases can be made to order. They improve the appearance of the 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 disks, CD-ROMs or other media will be packaged with the printed documentation 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 moulded plastic media holders in boxes of similar size to binders or books, held in slip cases along with the documents.

Before documents and software are packaged, agree the following with the packaging organisation:

- details of how separate documents and the software will be packaged together;
- information to be included on the outside of the package, including any artwork that is needed;

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

- details of the form of packaging that will be used;
- the number of packages required.

Consider producing packaging diagrams to describe how documents and software are to be packaged.

If packaging is to be carried out by a different organisation from the one that produced the copies of individual documents, arrange for the copies of the documents to be checked before they are packaged and for the copies to be transferred to the packaging organisation.

If the packaging is being carried out by the same organisation as the one that produced copies of documents, check sample documents together with sample packages.

If the software is to be packaged with the documentation, arrange for copies of the software to be delivered to the packaging organisation.

Give packaging instructions to the packaging organisation, 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.

E.2.4.5 Check the finished documents and package

Check sample copies of individual documents and samples of the entire packaged product, not all the copies.

It is normal practice for the production and the packaging organisations to produce a few sample copies for checking. Correct any mistakes in these samples before larger quantities are produced. Check random samples from these larger quantities.

Check the results of the production process and the packaging process, including 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 through the samples;
- the quality of the illustrations is suitable, for example, all drawn lines and text are clear;
- if colours are used, they are correct;
- 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.

Annex F (Informative) Writing style guides – Contents

Make the writing style simple and straightforward, so users are not distracted by complicated sentence structures or complicated vocabulary. Make the meaning clear and easy to assimilate, but do not patronise users by over-explaining simple ideas.

Remember that the language used may not be the first language of the reader of the document.

Large amounts of continuous text are difficult to follow. Divide text 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.

Use humour with caution. It may be helpful in some awareness documents to use cartoons to illustrate points. Humour 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.

Break down complicated ideas into simpler components so that they are explained fully. 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, provide an explanation.

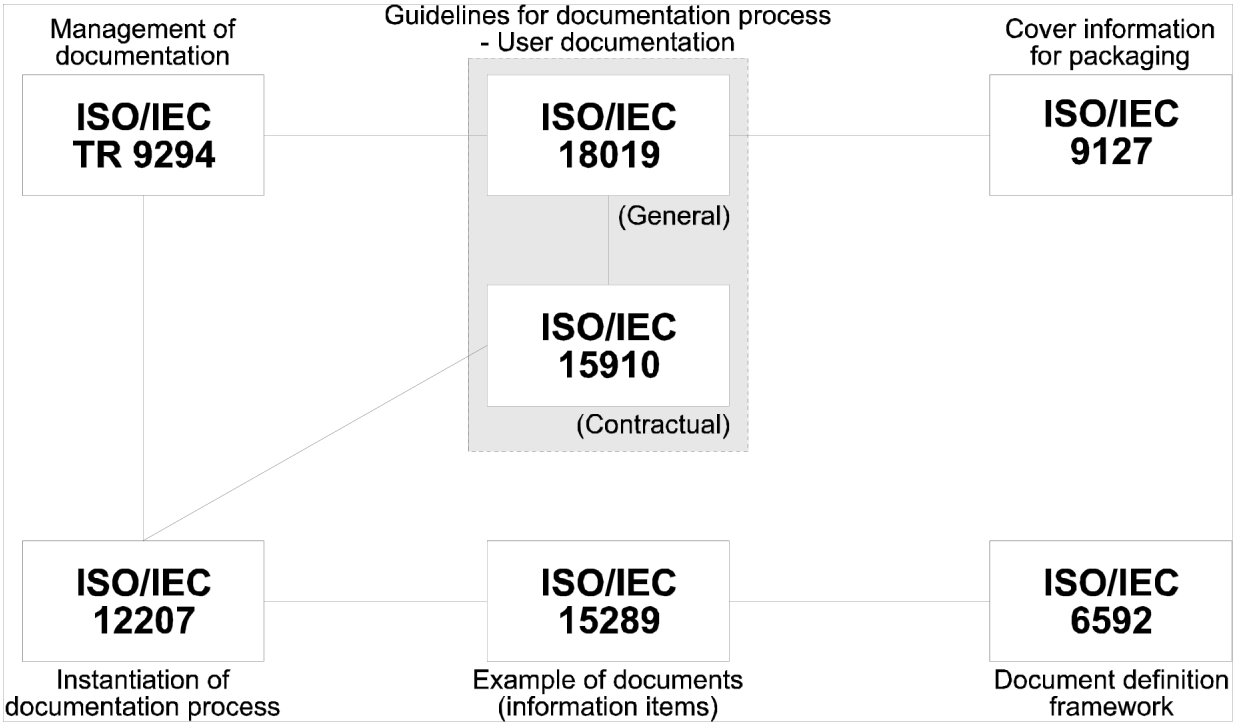
Give cautions and warnings in the imperative form. State clearly the action required, with a brief description of the hazard, if necessary.

Consider using illustrations instead of text if:

- an idea or relationship can 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.

Searching is important in all information processing. Illustrations may be very helpful when searching in documents. Illustrations are suitable for concretisation of structures. Users trying to find a solution to a problem, usually search in manuals for concrete examples, similar to their own problem situation.

Annex G (Informative) ISO/IEC 18019 and related standards



Bibliography

Icons and symbols:

[1] ISO 4196:1984, Graphical symbols – Use of arrows

[2] ISO/IEC 11581 (series), Information technology - Icon symbols and functions

Software management and quality:

[3] ISO/IEC 9126:2001, Information technology - Software quality characteristics and metrics

[4] ISO/IEC TR 9294:1990, Information technology - Guidelines for the management of software documentation

[5] ISO/IEC 12207:1995, Information technology - Software life cycle processes

[6] ISO/IEC TR 15846:1998, Information technology - Software life cycle process - Configuration management for software

[7] ISO/IEC 14598:1999, Information technology - Software product evaluation

[8] ISO 9000 (series), Quality management and quality assurance standards

Software design and development:

[9] ISO 9241(series), Ergonomics requirements for office work with visual display terminals

[10] ISO 13407:1999, Human-centred design processes for interactive systems

[11] ISO/IEC 15289, Software engineering - Guide for the application of ISO/IEC 12207 to the documentation process

Software documentation:

[12] ISO/IEC 9127:1988, User documentation and cover information for consumer software packages.

[13] ISO/IEC 15910:1999, Information technology – Software user documentation process.

Documentation printing and publishing:

[14] ISO 216:1975, Sizes of paper and board. Specification for A and B series of trimmed sizes of writing paper and certain classes of printed matters

[15] ISO 6357:1985, Recommendations for presentation of spine titles

[16] ISO/IEC 6592:2000, Information technology - Guidelines for the documentation of computer-based application systems

Accessibility:

[17] Section 508 of the US Rehabilitation Act: http://www.section508.gov/

Typography:

[18] Communicating or just making pretty shapes, A study of the validity - or otherwise of some elements of typographic design. Colin Wheildon, Newspaper Advertising Bureau of Australia Ltd.