A Guide to
Writing World Class Standards

ETSI

Contents

Introduction

ETSI standards are used the world over and have enabled technologies which have changed the way people live, work and do business. Today we produce about 2500 standards every year and our work touches almost every area of modern living.

This success owes much to the high quality of the standards we produce. We have earned a reputation for producing world class standards – a reputation we work hard to maintain. Writing good standards is not easy. Writing the sort of world class standards for which ETSI is well known is a complex and challenging task. Over the years, we have developed working methods and detailed drafting rules to ensure the quality and the reliability of our standards.

This document explains how to write ETSI standards. It starts with the basics for readers who are new to standardization or new to ETSI – what is a standard? It then goes on with guidance on how to choose the right type of standard, how to structure a standard and draft it, how to express the technical requirements so that you achieve your goals and finally how to validate, test and maintain a standard.

This guide is intended for anyone involved in creating ETSI standards, especially ETSI Secretariat staff, committee officials, Rapporteurs, Editors and Specialist Task Forces. There is something for everyone – even the most experienced among us can benefit from being reminded of the essentials of good standards writing.

By following this guide, you will produce standards that are accurate and complete, easy to read, clear, unambiguous and consistent – in short, you too will write world class standards.

What is a world class standard?

What is a standard?

A standard is a collection of the minimum requirements necessary for something to:

- Co-exist and interoperate correctly with another
- Meet national and international regulations
- Operate safely without causing harm to people or equipment.

What makes a standard 'world class'?

It is not easy to say exactly what makes one standard better than another but the following points are probably the most important:

- A world class standard should have well-defined objectives that respond to real needs in a timely manner.
- Its technical content should be complete and accurate.
- It should be easy to understand (or as easy as the subject matter allows!) and easy to implement.
- Its requirements should be expressed clearly and unambiguously.
- It should be validated.
- It should be well-maintained.

A response to real needs in a timely manner

A standard will be widely used if it has well-defined objectives that meet real needs in a timely manner. A standard produced too early may become quickly out of date because of a rapid evolution of the technology. On the other hand, a standard that is published too late risks being ignored in favour of earlier, competing standards or proprietary solutions.

Like so much else in standards-making, getting the timing right needs experience and good judgement – in this case, awareness of the state of maturity of the technology being specified. Remember that the type of standard that you choose to produce may affect its timing, because of the different approval processes involved.

Complete and accurate

To be judged world class, a standard should specify all of the requirements necessary to achieve its objectives and only include essential supporting information.

Some formal definitions of a standard

'A document established by consensus and approved by a recognized body that provides for common and repeated use, rules, guidelines or characteristics for activities or their results, aimed at the achievement of the optimum degree of order in a given context.'
- ISO/IEC Guide 2:1996

'A technical specification, adopted by a recognized standardization body, for repeated or continuous application, with which compliance is not compulsory, and which is one of the following:

- *'international standard' means a standard adopted by an international standardization body*
- *'European standard' means a standard adopted by a European standardization organization*
- *'harmonized standard' means a European standard adopted on the basis of a request made by the Commission for the application of Union harmonization legislation*
- *'national standard' means a standard adopted by a national standardization body.'*
- Council Regulation 1025/2012/EC

Easy to understand

A standard should be easy to understand and implement. This applies both to the way you structure the document and how you express individual requirements. For example, a clear structure can help readers follow the document, while good construction and expression of the requirements will make them easier to understand and implement. It also helps if there is a clear distinction between the normative parts (those parts which tell the readers exactly what they must do to comply with the standard) and the informative parts (the descriptive material that is there to improve general understanding of the concept). Unnecessary detail or informative text can make it difficult for readers to identify and understand what is really essential to ensure compliance.

Normative = prescriptive = how to comply with the standard

Informative = descriptive = help with conceptual understanding

Requirement = the criteria to be fulfilled to comply with the standard

Be careful in your use of English. The contributors and users of your standard may not all be native English speakers and may have varying language skills. To produce a world class standard, you should aim to create a document that can be understood by all readers.
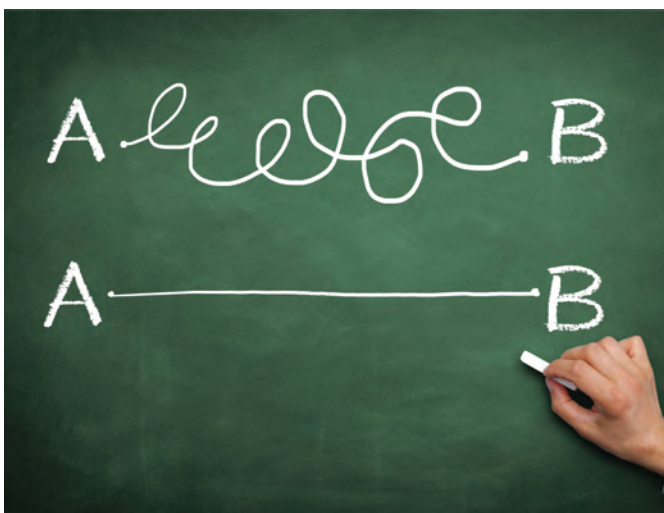
Clear and unambiguous requirements

Requirements specify the individual characteristics that a system or product must implement if it is to fully comply with the standard. Each requirement should be:

- **Necessary:** It should specify only what is required to meet its objectives, and not impose a particular approach to implementation.
- **Unambiguous:** It should be impossible to interpret the normative parts of the standard in more than one way.
- **Complete:** The requirement should contain all the information necessary to understand that requirement, either directly or by reference to other documents. The reader of a standard should not need to make assumptions about the implementation of any particular requirement.
- **Precise:** The requirement should be worded clearly and exactly, without unnecessary detail that might confuse the reader.
- **Well-structured:** The individual elements of the requirement should all be included in an appropriate and easy-to-read manner.
- **Consistent:** There should be no contradiction between different requirements within the standard, nor with other related standards.
- **Testable:** There should be clear and obvious means of demonstrating that an implementation complies with the requirement.

Validated

The purpose of validation is to confirm that the requirements expressed in a standard do, in fact, achieve their objectives and are fit for purpose. Validation assures the quality expected of a world class standard.

Maintained

ETSI standards are 'living' documents that develop over time in order to remain relevant to technological progress and changing needs and circumstances. To accommodate this and to maintain the high quality of our standards, the responsible committee should plan a programme of maintenance and evolution for each of its standards.

Before you get started

Who is your standard intended for?

In most instances, a standard is produced mainly for engineers who will implement it, and procurement specialists who will expect suppliers' products to conform to it. Often a standard is used to formalize the technical aspects of a procurement agreement or contract. Other groups (for example, regulators) may also have an interest in the standard.

The first thing to do when writing a standard is to identify how it is likely to be used, and by whom. Then you will be able to identify how to meet their requirements. In particular, bear in mind that the standard must be understandable by readers who have not been involved in its preparation, so you will need to explain things clearly and fully.

> ### Requirements and provisions
> Standards specify **requirements, recommendations** and **permissible actions** of some kind. Collectively, these are referred to as **provisions**.
>
> For clarity, in this document we use the term *requirement* in a general sense to mean provision.

The content of a standard

Standards consist of a series of requirements to be met when implementing a product or service. Typically, requirements might address:

- **Equipment behaviour** which can be observed at a defined, accessible interface, such as a protocol or service
- **Architectures:** communication and network architectures usually specify the functional (or logical) environment where the standard is expected to be implemented. Although abstract architectures do not generally specify testable or measurable requirements, they do define requirements (such as those related to a reference point) that should be complied with by the writers of protocol standards.
- **Physical characteristics** that can be measured, such as dimensions, voltage, frequency or colour

- **Policy requirements on operation and management practices:** for example related to the process of issuing qualified certificates by Certification Authorities
- **Test specifications:** detailed specifications of the tests that would demonstrate whether an implementation conforms to a standard, or of the tests that could be used to check that two implementations are interoperable
- **Methodologies** applicable to ETSI standardization (which may also be used more widely).

Do not try to standardize implementation details or constraints. Implementers need to know only what to implement and you must give them the freedom to achieve this in the way that suits them best.

Choosing the type of standard

ETSI produces several types of standard: each has its own particular purpose and each requires a different process of approval. The table below summarizes these different types of standard; the ETSI Directives and the Rapporteurs' Guide provide more complete explanations and will help you determine which type is appropriate for specific situations.

You should consider carefully which type of standard to create since your choice determines matters such as the extent of approval and the speed of publication. For example, the approval of Technical Specifications can be quite rapid, typically carried out in a Technical Committee meeting or by correspondence. By contrast, ETSI Standards and (in particular) European Standards usually take longer to approve.

Other factors that influence the speed of approval include the technical complexity of the document and the size of the approving community. Issues such as the strength of diverging views will have an impact as well. Misunderstandings and lack of clarity in the text can also hold things up at the approval stage and in the final pre-publication editing; by adopting practices that build high quality into the draft, you will help to minimize these delays.

Choosing the right deliverable

When deciding the appropriate type for deliverables with normative provisions, you should focus on the stakeholders who will be involved in approving the document. The approval process is likely to be more complex and time-consuming if the stakeholder group is large. You should therefore opt for deliverables that involve a large stakeholder group only when really necessary.

Within ETSI, the most commonly used deliverable type containing normative provisions is the Technical Specification (TS), which requires approval by just the Technical Committee that created it. You should consider whether this is sufficient in your case.

Document type	Approved by	Content
ETSI Technical Specification (TS)	*Responsible ETSI Technical Committee*	*Normative*
ETSI Standard (ES)	*ETSI membership*	*Normative*
European Standard (EN) *(which may be designated as a Harmonized Standard or a Community Specification)*	*Responsible ETSI Technical Committee and European National Standards Organizations and/or ETSI National Delegations*	*Normative, intended to be transposed into national standards of European Union Member States*
ETSI Technical Report (TR)	*Responsible ETSI Technical Committee*	*Informative – the preferred informative type of deliverable unless other considerations demand an EG (or SR)*
ETSI Guide (EG)	*ETSI membership*	*Informative – guidance for the ETSI Technical Organization in general*
ETSI Special Report (SR)	*Responsible ETSI Technical Committee*	*Informative – information made publicly available for reference purposes*
ETSI Group Specification (GS)	*Responsible ETSI Industry Specification Group*	*Normative or informative*

Harmonized Standards

Harmonized Standards are European Standards (ENs) with a special status. Produced by one or more of the European Standards Organizations (which include ETSI) in response to a mandate issued by the European Commission (EC), their purpose is to provide the technical detail necessary to achieve the 'essential requirements' of an EU Directive. These 'essential requirements' are usually written in very non-specific terms (e.g. *"the equipment shall not cause injury to the user"*) so the Harmonized Standards define, in detail, how those requirements are to be met.

ETSI supports various EU Directives through the production of Harmonized Standards including those concerning radio equipment, Electromagnetic Compatibility (EMC) and access to emergency services. Harmonized Standards enable manufacturers, suppliers, network operators and others to claim a product's compliance with the relevant Directives. These products can then be placed on the market and used in all Member States of the European Union.

ETSI Guide EG 201 399 provides guidance on writing Harmonized Standards.

Community Specifications

In response to mandates from the European Commission, we also produce ENs under the Single European Sky Interoperability Regulation (i.e. in civil aviation). These ENs are produced in co-operation with EUROCAE (the European Organization for Civil Aviation Equipment) and acquire the status of Community Specifications (CSs) when they are published in the Official Journal of the European Union.

System Reference Documents

ETSI produces a specific type of Technical Report called a 'System Reference document' (SRdoc). These provide technical, legal and economic background to new radio systems, services or applications. They advise on the need for an allocation of spectrum, in particular either when a change in the current frequency designation or its usage, or a change in the regulatory framework for the proposed band(s) is needed to accommodate a new radio system or service.

The basic steps of standards development

The development process can be quite complex within a standardization body such as ETSI. Not only do the diverse interests of our members have to be respected but there may be other constraints, such as those imposed by EC Mandates. ETSI's Standards Development Process has been created to guide and manage the drafting of standards in a way that keeps the intended purpose of the standard in focus, whilst addressing the different views and changes of direction that are inevitable in any collaborative development task.

The key stages of the standards development process are to:

1. **Create the Work Item**

2. **Develop the draft standard**

3. **Validate the draft**

4. **Submit the draft for editorial checking**

5. **Approve and publish the standard**

6. **Maintain and evolve the standard**

At each stage there is the possibility of feedback into the drafting process. This can provide enormous benefits for the quality of the final document and helps to maintain its 'world class' status.

Create the Work Item

"If you don't know where you are going, you may have difficulty getting there!"

One of the most important steps towards producing a 'good' standard is the preparation of the Work Item proposal. If this is well-written it provides a strong and clear platform from which the standard can be developed. On the other hand, a badly constructed or incomplete Work Item will make it difficult for the authors and reviewers to know what the content should be. Technical Officers can provide their committees with considerable assistance, as well as background knowledge, to simplify the task of developing a 'good' Work Item proposal.

As you start to develop the Work Item proposal, you should consider the following:

- Is the document to be **normative** or **informative**? What **type of deliverable** should it be (EN, ES, TS etc…)?
- What will the **title** of the standard be? It should give a concise indication of the content. If appropriate, it should make clear to which grouping of standards the document belongs. It should be simple; any necessary detail, such as constraints and areas of application, can be provided in the Scope.
- What is the **Scope** of the work? The Work Item Scope should specify the intended contents of the deliverable.
  - The Work Item Scope helps to keep the developers focused on what they are creating, whilst the document's Scope clause provides the reader with a summary of the subject and the area of application of the standard.
  - The Work Item Scope and the Scope clause of the standard may not be the same as they serve different purposes. However, the two Scopes should not contradict one another!
  - For a new standard (i.e. not a revision), it should be possible to use the Work Item Scope as the basis for the document's Scope clause.
  - For the revision of an existing standard, the Work Item Scope need only specify the additional drafting to be undertaken. So it is unlikely that it will be possible to transpose the Work Item Scope directly into the standard. However, if the revision modifies the original Scope of the standard then, of course, the document Scope clause should be changed as necessary.
- Is the Work Item part of a **Hierarchical Work Item**? If so, the Work Item description should make it clear where the proposed document fits in the hierarchy.

Hierarchical Work Items
Committees may decide to organize or present all or part of their work programme in a hierarchical manner. The hierarchy may be organized by any criteria they decide (e.g. 'release', 'technical area', 'project', 'stage' etc.).

- **When is the deliverable likely to be completed**? The schedule of milestones should be realistic. It should take account of the Scope of the document being drafted and the resources that will be available during development.

- **Validation:** The quality of a standard is likely to be improved considerably if you include activities to validate its content as it evolves. Will you need additional activities such as the creation of test specifications or conformance tests, which may need to be available by the time the standard has been implemented as commercial products? Remember to include these activities in your schedule. EG 202 107 provides guidance on how to plan for validation and testing in the standards-making process.
- **Who is responsible** for producing the document? Have the Technical Committee, Working Group and Rapporteur been identified and agreed? Is there a commitment from members? Is a Specialist Task Force (STF) needed?

Develop the draft standard
You can help to enhance the quality of your standard by adopting a structured approach in its drafting. Your aim is to achieve a document that is clear, unambiguous, complete and accurate. It is vital to write requirements in well-constructed English and to avoid unnecessary narrative and tutorial text. Specialized notations such as ASN.1 and graphical forms of specification such as State Charts and Sequence Diagrams are also valuable in improving understanding and precision.

As you develop the standard, you should also keep in mind the need to be able to test, in practice, every requirement that you include. You must ensure the feasibility of showing, by testing, that each requirement has been correctly implemented.

Validate the draft
The development process of any standard should already include document reviews, even if only as part of the approval cycle. Any secondary activity that involves close scrutiny of the standard (such as test development, implementation of the standard in products and the production of implementation and design guides related to the standard) will provide useful feedback on deficiencies, inconsistencies and ambiguities. The only option that you should not consider is not validating the standard at all!

The ETSI Drafting Rules
In order to encourage consistency, ETSI has established a basic structure for all its standards, described in the ETSI Drafting Rules (EDR). Other international standards organizations have similar rules. You can download the Drafting Rules in a useful format at http://portal.etsi.org/edithelp/home.asp.

Submit the draft for editorial checking

The Technical Officer of your committee has a good awareness of the technology to which your standard relates. He or she also has in-depth knowledge of the various ETSI rules and procedures that help to ensure high-quality standards. The Technical Officer is therefore a valuable partner to you in your standards-making task and you should not hesitate to consult him or her regularly as you develop your draft. The Technical Officer is there to support and advise you.

In addition, the ETSI Secretariat has centres of expertise that can provide specialist help when needed. Among these is the *editHelp!* Service which provides advice and tools for the writers of ETSI standards, as well as overseeing the approval and publication of the standards.

The *editHelp!* Service has an essential role in working with you to finalize your draft. Once you and your committee consider your draft standard to be reasonably complete, you should send it to the *editHelp!* Service (normally via your Technical Officer, unless other arrangements have been made in your committee – your Technical Officer can tell you). *editHelp!* will check many different aspects of the editing, including the use of English, the validity and consistency of the references, the abbreviations and the definitions of terms.

This activity follows approval by the technical committee, but you are also welcome to submit your draft to *editHelp!* earlier during its development, for example to make sure that it is following the correct shape and style.

However, do not treat the *editHelp!* Service as a substitute for the proper preparation of a standard! The service will suggest corrections to structure, presentation and consistency but does not evaluate the technical content.

Remember that your Technical Officer is your key interface to *editHelp!* and other Secretariat services.

Approve and publish the standard

The approval process prior to publication is different for each type of standard. The Technical Officer will manage the passage of the draft through each stage of approval. ETSI's web-based voting tools make it easy for our members and others to record comments and objections: this feedback is very valuable for improving the technical quality of the standard and should be considered as part of the validation process.

Maintain and evolve the standard

Standards require on-going maintenance that needs to be planned for and managed. The Scope of the revision may include such matters as the correction of defects found after publication and alignment with changes in related standards.

Dealing with sets of standards

Develop a framework

If you are involved in the production of a set of standards, you may find it helpful to agree with your fellow writers to produce a framework document. This will provide guidelines for the writers of all the standards in the set to achieve a broad degree of commonality and consistency in their presentation. There are no specific requirements for what should be included in a framework document but you might want to include the following:

- Document structure
- Requirement numbering format
- Naming conventions
- Recommendations on the use of specialized languages
- Recommendations on the use of tools
- Identification of supporting material

Adopt a systematic approach

Complex subjects require particularly careful organization. The general principles of the well-proven three-stage approach described in ITU Recommendation I.130 are particularly useful in encouraging and facilitating a systematic method, especially for the development of protocol and service standards. This approach can also be adapted for other specifications. However, some techniques in I.130 may have been superseded; these will need to be aligned with current practices.

The Recommendation proposes a design process divided into three stages of activity, typically resulting in a corresponding series of specifications:

Stage 1 Specify objectives from the user's perspective

Stage 2 Develop a functional model to meet those objectives

Stage 3 Develop a specification of the detailed implementation requirements.

You can find more detailed information about the three-stage approach, with examples, on ETSI's *Making Better Standards* website at http://portal.etsi.org/mbs/protocolStandards/stagedApproach.htm.

How to draft a good standard

Clarity and consistency are essential characteristics of a world class standard. You can achieve them by taking a logical approach to the document's structure and content, and by respecting well-established standards-writing practices.

Developing standards is similar to the task of engineering a product. Both are only effective and efficient if the overall process is systematic, and both require a top-down approach; any other approach is unlikely to deliver what was intended.

Use the ETSI pre-defined deliverable structure

Using a consistent structure for a standard will not only help you, the writer, to collect essential information and requirements together in a logical manner, it will also help the reader to navigate through the document. Consistency improves the quality of the standard as well as the perception of quality. For a set of associated standards, consistency between the standards makes them easier to read and implement.

ETSI has also produced skeleton documents for each deliverable type and these can help you considerably in your task. You can download them from the *editHelp!* pages on the ETSI Portal (http://portal.etsi.org/edithelp/home.asp). These skeletons contain all the obligatory elements, as well as some optional ones, in the required order:

- Front cover
- Table of contents
- Intellectual Property Rights
- Foreword
- Introduction (optional)
- Scope
- References
- Abbreviations, Symbols and Definitions

These pre-defined elements set the context for the standard and define the terms the reader will need to understand the rest of the document. They are followed by the technical content – the heart of the standard.

You must not write any requirements in these pre-defined elements.

Draft the content of the pre-defined elements

A summary of what goes into each element is described below; you can find full details in the ETSI Drafting Rules and the relevant skeleton document.

Foreword
The Foreword starts by mentioning the document type and the committee that prepared the document. If appropriate you may also include other information that would be helpful to the reader, such as:

- The relationship between this standard and other documents
- Documents that this standard supersedes or replaces
- Significant technical changes from a previous version of the standard
- The existence of an electronic attachment.

For European Standards (ENs), the Secretariat will insert a table containing the transposition dates. The Secretariat will also add the special text required if the document is to be a Harmonized Standard.

There are additional rules if the document is part of a multi-part standard.

The example Foreword below contains most of the above elements.

> # Foreword
>
> This European Standard (EN) has been produced by ETSI Technical Committee Digital Enhanced Cordless Telecommunications (DECT).
>
> The present document contains text pertaining to approval testing of the Digital Enhanced Cordless Telecommunications (DECT) Common Interface. Such text should be considered as guidance to approval (or licensing) authorities.
>
> Details of the DECT Common Interface may be found in EN300 175-1 [1] to EN 300 175-8 [8]. Further details of the DECT system may be found in the ETSI Technical Reports, TR 101 178 [i.1] and ETR 043 [i.7].
>
> The present document is part 2 of a multi-part deliverable covering the approval test specification for Digital Enhanced Cordless Telecommunications (DECT), as identified below:
>
> Part 1:   "Radio";
>
> **Part 2:   "Audio and speech".**
>
> | National transposition dates | |
> | --- | --- |
> | Date of adoption of this EN: | 12 July 2012 |
> | Date of latest announcement of this EN (doa): | 31 October 2012 |
> | Date of latest publication of new National Standard or endorsement of this EN (dop/e): | 30 April 2013 |

Introduction
The Introduction is optional, but can help readers to understand the document. The Introduction may give specific information or commentary about the technical content of the deliverable, and the reasons prompting its preparation. For example:

> # Introduction
>
> The European Union launched the Legislation "Single European Sky" (SES) in 2002 which was adopted in 2004 and amended in 2009 [i.6].
>
> The SES legislation is based on a framework of 4 regulations, which includes the Interoperability Regulation [i.6]. The objective of the Interoperability Regulation is to ensure interoperability of the European Air Traffic Management Network (EATMN) consistent with air navigation services. Under this regulation, the use of a European Standard referenced in the Official Journal of the European Union as Community Specification(CS) is a means of compliance to the essential requirements of the Regulation and/or the relevant implementing rules for interoperability.
>
> The present document takes into account the Council Decision 2009/320/EC [i.8] endorsing the European Air Traffic Management Master Plan for the Single European Sky ATM Research (SESAR) project.

Scope

The aim of the Scope clause is to provide readers with a succinct, factual statement of the purpose of the document. This may include the subject of the standard, the area of applicability, the type of product or service and other relevant information such as the relationship of the standard to other standards – as long as such details clarify the Scope of your document.

You should review the Scope text regularly as drafting proceeds to ensure that it remains correct, complete and in alignment with the Work Item scope. If you find that the draft document is deviating from the Scope you should inform the responsible committee.

The Scope

The ETSI Drafting Rules provide guidance on how to structure the Scope:

The present document
- *specifies   the functional requirements for …*
                  a method of …
                  the characteristics of …
- *establishes   a system for …*
                  general principles for …
- *gives guidelines for …*
- *gives terms and definitions …*

The present document is applicable to …

---

1   Scope

The present document specifies the minimum requirements for shipborne radio transmitters and receivers for fixed installations operating in the VHF frequency bands between 156 MHz and 174 MHz allocated to the maritime mobile service, using both 25 kHz and 12,5 kHz channels and capable of Radiotelephony and Digital Selective Calling communications within the Global Maritime Distress and Safety System. The present document incorporates the requirements of the relevant resolutions of the International Maritime Organization (IMO) and is primarily intended to specify equipment suitable for fitting to ships subject to the SOLAS Convention[1] and complying with the Council Directive 96/98/EC [i.11] of 20 December 1996 on marine equipment as amended (the European Marine Equipment Directive).

The present document does not address the testing of ancillary equipment on a stand-alone basis, i.e. separately from the radio equipment with which it is to be used.

References

References form a vital part of a standard and should be handled carefully and consistently. If your standard has both normative and informative references, you should include them in two distinct clauses within the *References* clause:

- *Normative references* identify other documents that are directly referenced within normative provisions in the main body of the standard. They are effectively part of the standard, even though they are actually another document, so it is **imperative that you ensure that all referenced documents are publicly available!**
- *Informative references* identify other documents that are directly referenced from within the main body of the standard purely to provide additional information for the purposes of justification or clarification.

References can relate to either a specific edition of a document or, if the date and version are omitted, they relate generically to the latest edition. Be careful! If the user of the standard uses a different version of the referenced document to what you intended, **their implementation may be wrong!**

If your standard makes references to **specific clauses** in another document you should always specify which edition you are referencing. Normative references in a Harmonized Standard must always refer to a specific edition.

You may be aware of documents that are not referred to directly within the main body of the standard but which might be of interest or assistance to readers. You can help your readers by listing these documents in a **Bibliography** annex (rather than in the Informative references clause).

Definitions, symbols and abbreviations

If your standard is to be implemented correctly, you need your readers to understand it in exactly the same way that you do. In order to avoid misunderstanding, you should define terms that are included in requirements – this is particularly important if those terms are specialized or not widely known.

You should also define terms that are used regularly in the document and, where appropriate, identify a defined term by an abbreviation. This helps the reader, as well as simplifying your task because it allows you to refer repeatedly to objects or concepts without having to describe them in detail each time you mention them.

Where you have defined a term or abbreviation, you should use it – always and consistently. For example, it could be confusing (and certainly irritating!) to read a standard that sometimes refers to "ADSL", sometimes to "Asymmetric Digital Subscriber Line", sometimes to "Asymmetric Digital Subscriber Loop" and sometimes to "a digital line with different bit rates in the uplink and downlink". Having

Definitions

If you use any specialist terms in the main body of the standard that require specific interpretation you should list and define them in the *Definitions* clause. Each entry in the clause should comprise the term followed by a definition of that term, for example:

> **interface:** *common boundary between two associated systems*

The definition of a term should never include a requirement.

Symbols

All technical symbols (including mathematical and scientific symbols and units) used in the main body of the standard should be listed together in the *Symbols* clause. Each entry should comprise the symbol and a very brief explanation of what it represents, for example:

Ro	*reference distance*
μ	*microns*
λ	*wavelength*

Abbreviations

All acronyms used in the main body of the standard should be listed together in the *Abbreviations* clause. Each entry should comprise the acronym itself and the corresponding full term, for example:

IPv6	*Internet Protocol version 6*
ITS	*Intelligent Transport System*
ITS-S	*ITS Station*
RX	*Receiver*
TX	*Transmitter*
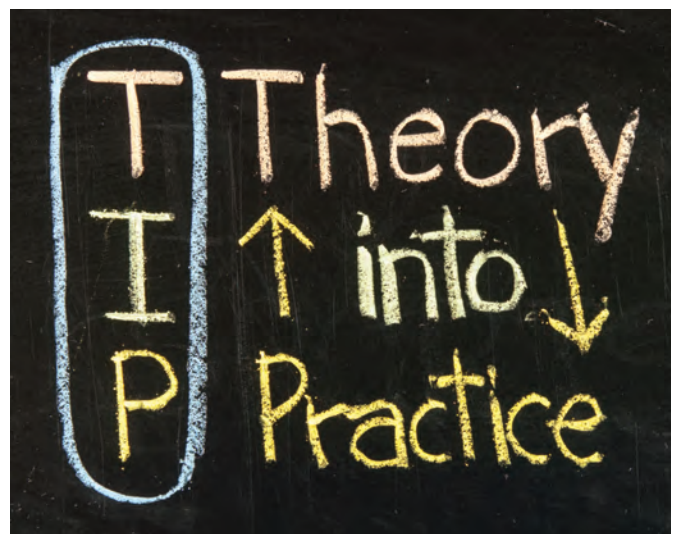
defined the term, "Asymmetric Digital Subscriber Line", and its abbreviation, "ADSL", you should use either the full term or the abbreviation whenever the concept is mentioned.

In your standard you should provide lists of the definitions, symbols and abbreviations that you have used in the document, together with the meanings that you have given them. As far as possible of course, those meanings should also be the same as those used in related standards and, where appropriate, throughout the industry. This need for consistency applies not just to individual documents but across related standards, which is why it can be useful to create a separate document that contains all the abbreviations, symbols and definitions of terms used within a set of related standards. It also avoids wasting your effort, and that of other standards-writers, in defining terms that have already been defined.

The Terms and Definitions Database, Interactive (TEDDI) on the ETSI Portal (http://webapp.etsi.org/Teddi/) is a helpful tool for achieving consistency in your documents, since it records definitions and abbreviations that have already been defined in other documents. However, you should make sure that TEDDI definitions are applicable and consistent with your needs.

The technical content of the document
The structure of the technical content will depend on the subject of the standard but the following principles will help you achieve a good result:

- Group related topics in clauses or sub-clauses with appropriate titles.
- Order the topics in a logical way so that preceding text helps the reader to understand what follows.
- Clearly distinguish between normative and informative text
  - Put informative descriptions in separate clauses and consider marking them as informative.
  - If it is not possible to insert informative material as a separate clause and it has to sit within a normative part of the standard, present the material as 'Notes' to distinguish it. (Do not use footnotes!)
- Improve the readability of the main body of the standard by moving self-contained parts to annexes. Such parts might be:
  - the normative specification of a complete functionality or service
  - informative descriptions for understanding and background
  - specifications in specialized notations such as ASN.1, XML or TTCN-3.
- If you are producing a set of associated standards, adopting a common structure for all technical content will make the documents easier to read.

Numbering of clauses, figures and tables
The numbering of clauses, figures and tables is clearly specified in the ETSI Drafting Rules and all standards should follow these guidelines. The reason for numbering is simple: the reader should be able to identify every requirement in a standard by a reference.

Bear in mind that the numbering within a document can change if the document is updated, bringing a risk of confusion and possible mis-application of requirements.

Follow the style guide

In addition to defining a structure, the ETSI Drafting Rules specify a comprehensive range of Microsoft® Word® styles for use in ETSI deliverables. This helps ensure a consistent appearance to all published documents. These styles address most editorial requirements and you should not need to add new styles to a document – neither should you modify the styles provided.

The technical content of the standard – How to write good requirements

The technical content of a standard should specify the minimum requirements necessary to achieve the objectives of the standard, while leaving room for product differentiation. In other words, standards should specify what is to be achieved by an implementation rather than how to achieve it. They are not product specifications. The technical committee has the responsibility for deciding what functionality should be specified in a standard and what should not.

ETSI standards are limited to defining normative provisions such as essential physical or electrical characteristics, or open interfaces and common data formats for communication protocols. Requirements range from the *high-level*, such as stating that equipment shall support a specific functionality or capability, to the *low-level*, such as setting specific values or characteristics or behaviour.

Some basic principles

Any suitably qualified reader should be able to understand your standard – without misinterpretation! Remember that many readers of your standard will not have participated in the committee that produced it. Misinterpretation due to deficiencies in a standard is a major cause of incorrect implementation and can lead to non-interoperable products. It is therefore essential that all requirements in a standard are *unambiguous*.

Use of English

ETSI deliverables are written in the English language but idiom, spelling and grammar differ from country to country. Given ETSI's European context, British English is preferable to any other variant.

As standards are contribution-driven, they may contain a mixture of varieties of English and styles of writing. During the later phases of drafting you will need to rationalize the language in order to:

- align the writing styles of different contributors to make the standard read like a consistent document rather than a collection of contributions
- make it consistently British English throughout
- eliminate grammatical errors and mis-spellings (poor spelling and poor grammar not only lead to ambiguity but also give the impression of a low-quality deliverable that cannot be trusted)
- eliminate non-standard idiom that might not be understood by many readers.

Unambiguous requirements have a single, clear interpretation. The clear and concise use of English is important. For example, writing
> the bitstreams shall be aggregated

does not make it obvious what or who will perform the task of aggregating the bitstreams. The ambiguity is removed when the requirement is written as
> the multiplexer shall aggregate the bitstreams.

Lack of precision also leads to ambiguity:
> If the timer T1 expires before receipt of a T-DISCONNECT_ indication the SPM requests a transport_disconnection with a T-DISCONNECT_request. The timer is cancelled on receipt of a T-DISCONNECT_indication.

In this example there are two sources of ambiguity. Firstly, the use of the present tense ("requests" and "is cancelled") makes it unclear whether this is actually a requirement or not. Secondly, it is ambiguous whether or not the second sentence is part of the condition in the first sentence. Most probably these are two completely separate requirements as the cancellation of an expired timer makes little sense. A better approach would be:
> The timer T1 shall be cancelled on receipt of a T-DISCONNECT_indication.

> If the timer T1 expires before receipt of a T-DISCONNECT indication the SPM shall request a transport disconnection by sending a T-DISCONNECT_ request.

If the cancellation of the timer was indeed intended (even after expiry), then the second requirement should be as follows (the first requirement remains unchanged):
> If the timer T1 expires before receipt of a T-DISCONNECT_indication the SPM shall request a transport disconnection by sending a T-DISCONNECT_ request and the SPM shall cancel Timer T1.

Keep it impersonal

Do not use the 1st person (*I, we*), 2nd person (*You*), or the 3rd person personal (*He, she, they*).

Thus, these examples are acceptable:
> …the equipment shall respond with a CALL_ACCEPTED message
> …the potential shall be set to +5 volts.

The following are not acceptable:
> …you should set the audio level to 0dB
> …we recommend that it should not exceed 100µA.

Shall, should and may

In an ETSI deliverable, requirements are each expressed using specific verbs: **SHALL**, **SHOULD** and **MAY**.

- **SHALL** is used to express *mandatory requirements* (provisions that have to be followed):
  - the negative form is **SHALL NOT**.
  - **Do not use** the verb **MUST** as an alternative to SHALL. It is forbidden by the ETSI Drafting Rules!

  Example:
  During call set-up, TVP <u>shall not</u> be incremented during call setup synchronization bursts but <u>shall</u> be repeated across each slot of the frames containing the DM-SETUP or DM-SETUP PRES messages

- **SHOULD** is used to express *recommendations* (provisions that an implementation is expected to follow unless there is a strong reason for not doing so):
  - the negative form is **SHOULD NOT.**

  Examples:
  For best subjective speech quality the CNF parameters <u>should</u> be estimated using periods of no input speech activity.

  An MS that does not support authentication <u>should not</u> select a cell that broadcasts "authentication required".

- **MAY** is used to express *permissible actions* (provisions that an implementation is able to follow or not follow):
  - the negative form is **NEED NOT** (in English MAY NOT is ambiguous, so NEED NOT is used instead).
  - **Do not use** the verb **CAN** as an alternative to MAY.

  Examples:
  Members of an SCK set <u>may</u> be shared amongst different DMO nets. They <u>may</u> be allocated in either the home V+D network of the MS or by an external body.

  Speech frames marked "NO_IMPORTANCE" <u>need not</u> be transmitted over the air interface.

Alternative forms

The Drafting Rules offer alternatives for some situations (such as *is required to*, *ought to*, and, *will* and *is*) but they can make the text very difficult to read and may cause confusion. For this reason…

…avoid any of these alternative forms!

Warning!

Do not use SHALL in informative text, as the word implies a requirement. Similar terms that imply a requirement, such as 'have to' and 'required to', are also not permitted in informative text.
If you find that your informative text seems to need the use of such words, you should check that it really is purely informative!

Remember that this restriction applies not just to informative material in a standard but equally to informative types of documents (EG, TR, SR and possibly GS).
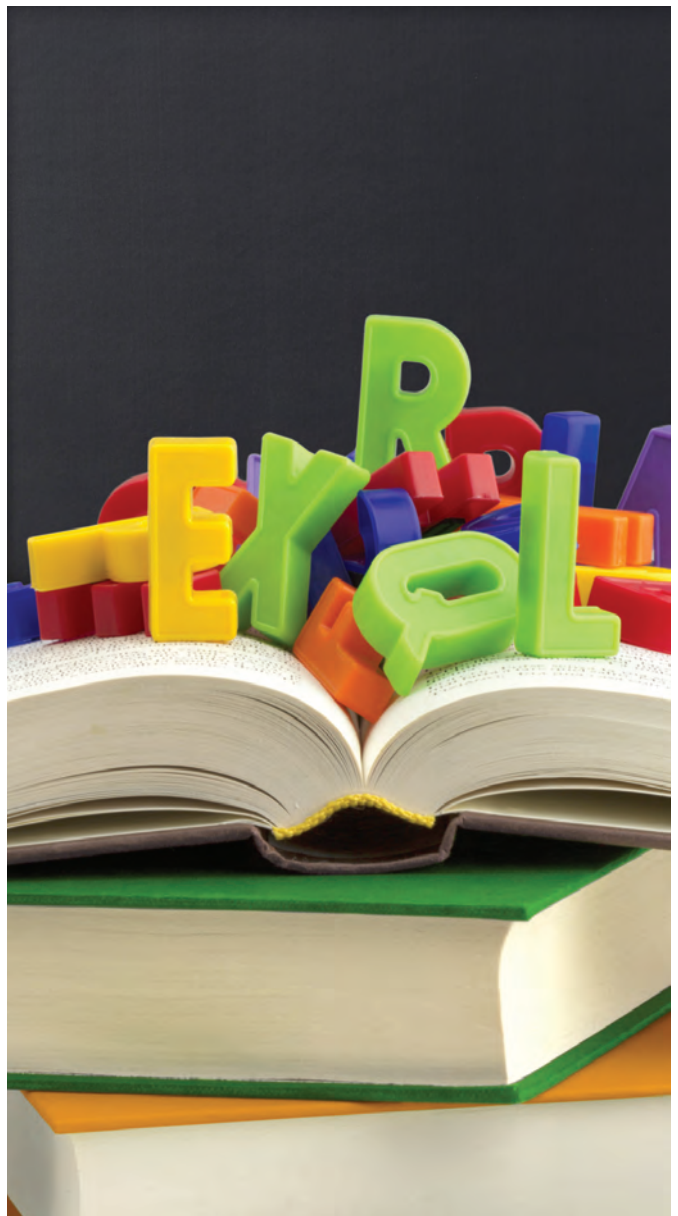
Structuring the requirements

Writing *well-structured* requirements that are *consistent, complete* and *precise* will go a long way to removing ambiguities and potential misunderstandings.

To ensure that your standard is understandable and usable, structure the requirements in a logical and consistent manner:

- Keep each requirement as simple as possible
  - Avoid unnecessary complexity (e.g. long sentences covering several requirements or long requirements covering several sentences).
  - Use numbered or bulleted lists rather than complex sentences.
- Make the logical structure of each requirement, or combination of requirements, clear
  - Ensure that the requirements specified in a clause are directly related to the title of that clause.
  - Clearly identify *the context* in which a requirement appears, any *pre-conditions* that may apply to the requirement and any subsequent or *alternative* requirements that may need to be specified.
- Specify the requirement in one place
  - Avoid splitting a requirement (for example, partially defining the requirement on one page and finishing it off several pages later). If splitting is unavoidable, make sure the link to the two parts is clearly explained and referenced.
  - Avoid multiple definitions of the same requirement. If *similar* requirements are found, decide whether they are truly different or just poor expressions of the same thing. If there are strong similarities between two *different* requirements make it clear that they are separate to prevent the reader thinking one is just an inconsistent version of the other!
  - Group related requirements (architectures, protocols, service primitives etc.) in separate clauses.
  - Avoid implicit requirements (for example, having the definition of one requirement implied by another).
- Specify the requirements in a logical order
  - Avoid listing requirements that assume requirements which only appear later in the document.
  - Use functional decomposition

Functional decomposition

Start with high-level requirements and work down to the detailed or lower-level requirements. For example, a high-level requirement such as

The equipment shall support functionality F1

is a simple statement even though functionality F1 may need a large number of subsequent requirements to specify it completely. For example:

The Base Band Interface shall support the Radio Application Interface for baseband signal processing.

- Use well-established design practices for protocol layering, reference points, naming etc.
- Consider the use of specialized notations to increase precision
  - Some requirements can be very difficult, if not impossible, to express precisely in text, such as the structure of communication messages. Specialized notations exist to overcome these difficulties.
  - Specialized notations can also be used to add clarity to requirements which can be expressed in text.
- "A good picture is worth a thousand words."
  - Charts and diagrams can also help explain concepts that are difficult to express in text.

Identifying requirements

Remember that it must be possible to reference every requirement. This can be done through clause numbering, but it can also be done by giving each requirement a unique identifier. There is no fixed format but it should be:
- An alphanumeric string
- Unique within the standard (or set of standards)
- Easily identifiable as a requirement identifier (for example, it has the format *REQ nnnnn*).

Static and dynamic requirements

Static requirements specify characteristics that do not change over time and are not dependent on previous events. They can be expressed as single statements and may be written in any order.

No-load conditions shall not exceed 0.3 Watt.

Dynamic requirements express the behavioural aspects of equipment and usually state what is required to happen when something else happens – in other words, cause and effect. In the example below the cause is in bold text:

When a new SIM is inserted in the preferred reader, *the user shall be offered the ability to select the new SIM after the call is terminated.*

Usually, dynamic requirements are more complex than static requirements and there may be dependencies between them that mean they cannot be written in an arbitrary order. In these situations, careful attention to the overall composition of the standard will ensure clarity.

Putting requirements into context

Often, a requirement or set of requirements only applies within a given context. In such a situation, you should take care to ensure that the scope of that context is well-defined and can be easily related to the relevant requirements. There are essentially two kinds of context: *environmental* and *descriptive*.

The **environmental context** describes matters that directly relate to the standard. These may be interactions with peer entities or other systems, typically specified in other standards.

For example:

A standard specifying the behaviour of a terminal needs to describe responses to messages received from a network that is the subject of a separate standard.

*The **network standard** specifies that it* <u>shall</u> *send a CALL_TERMINATED message in response to a received DISCONNECT message and that it* <u>may</u> *send a CALL_PROCEEDING message in response to a received SETUP message.*

*The **terminal standard** then specifies that the network* <u>will</u> *respond with a CALL_TERMINATED message to the terminal's DISCONNECT and that it* <u>can</u> *respond with a CALL_PROCEEDING message in response to its SETUP.*

The **descriptive context** provides additional information to readers to help them understand the requirement. In the following example, the bold text is the context and the non-bold text is the actual requirement:

The address field range is extended by reserving the first transmitted bit of the address field octets to indicate the final octet of the address field. The presence of a 1 in the first bit of an address field octet signals that it is the final octet of the address field. *The double octet address field for LAPD operation shall have bit 1 of the first octet set to a 0 and bit 1 of the second octet set to 1.*

Note that the requirement would not be changed in any way if the descriptive context were removed.

In practice, the context may not be adjacent to the requirement text. It may be a clause introducing a whole set of subsequent requirements.

Conditions under which a requirement applies

A requirement may depend on one or more **conditions** (sometimes referred to as pre-conditions) that must be met before it can apply.

In this example, the bold text is the condition and the non-bold text is the actual requirement:

> **With the ME switched off and with the power source connected to the ME** the voltages on contacts C1, C2, C3, C6 and C7 of the ME shall be between 0 and ±0.4 V referenced to ground (C5).

It usually helps the reader if you put the condition first in the sentence, followed by the requirement, as in the example above. However, in some instances it might be clearer if the condition comes after the requirement or is even embedded in it, as illustrated below. It is up to you as the author to decide which is preferable.

> When a new SIM is inserted in the preferred reader **during a call**, the user shall be offered the ability to select the new SIM after the call is terminated.

In some cases, a requirement may be the condition for another requirement, typically of the form:

> If REQ1 is implemented then REQ2 applies.

For example:

> **If the ME is able to detect a 5V only SIM according to the procedure in clause 5.1.3, or if the procedure cannot be completed,** then the ME shall deactivate and reject the SIM immediately (maximum of 5s) without issuing any further command.

Several requirements may be valid within a single condition, for example:

> If additional carriers are supported by the equipment, **the applicant shall declare the band edge limits FL and FU** <u>and</u> shall declare **the carriers supported**.

In such cases, it is important to be very clear about the scope of the condition and to which requirement or requirements it applies. You need to be careful to avoid ambiguity. A sentence such as the following may confuse the reader:

> If COND1 is fulfilled then REQ1 and REQ2 apply.

Does the condition apply only to REQ1 or to REQ2 as well? You can avoid any confusion by careful use of language: for example:

> If COND1 is fulfilled then both REQ1 and REQ2 apply.

Remember that it can happen that the condition is defined in a different part of the document, so you should take care to ensure that it can be easily identified and understood by the reader.

Specifying alternatives to conditional requirements

Your reader needs to be clear what should happen if a condition is not met. Often, this will be obvious but you need to take care to ensure that the reader is not left in doubt. This may mean that you will need to define the consequences if the condition is not met.

> If COND1 is fulfilled then REQ1 applies **otherwise** REQ2 applies.

A simple conditional requirement with an alternative might be:

> If installed in an emergency vehicle the equipment shall be painted red otherwise it shall be painted blue.

Expressing this as two separate requirements (with mutually exclusive conditions) may make it clearer:

> If installed in an emergency vehicle the equipment shall be painted red.

> If installed in any vehicle except an emergency vehicle the equipment shall be painted blue.

Combining requirements

The specification of an item may consist of several requirements. You could simply list them but this may read awkwardly. For example:

The equipment shall have length 15cm.
The equipment shall have width 20cm.
The equipment shall have height 10cm.

It might be better to write this as a single requirement:

The equipment shall have the dimensions of 15cm x 20cm x 10cm (length x width x height respectively).

Or as a **combination** of three requirements in a single statement:

*The equipment shall have length 15cm **and** shall have width 20cm **and** shall have height 10cm.*

The requirements in the above examples are all **static requirements** and you could list them in any of these forms because their order is not important.

However, the order of a combination of **dynamic requirements** may sometimes be significant. In the following example, the standard does not require that the two actions (sending the SETUP message and starting the timer T30) are done in any particular order.

When requested by the user the terminal equipment shall send a SETUP message and shall start timer T30.

Conversely, if the sequence is important you should make it clear by writing the requirement in the following manner:

When requested by the user the terminal equipment shall send a SETUP message and then shall start timer T30.

As you can see, it is important to describe dynamic requirements correctly so that the reader is not confused or made to misunderstand what is intended. Words such as and, or, either, neither and not are useful for combining requirements, but care is still needed.

For example, the text defining a set of requirements could take the form:

REQ1 and REQ2 or REQ3 but not REQ4.

But this is ambiguous! Does REQ3 apply as an alternative to REQ2?

REQ1 and (REQ2 or REQ3) but not REQ4

Or is REQ3 an alternative to both REQ1 and REQ2?

(REQ1 and REQ2) or REQ3 but not REQ4

In these abstract examples, parentheses help to clarify the intentions but, in a real standard, words or phrases such as *either, neither, both, at least* can be used to achieve the same effect. For example:

REQ1 and (REQ2 or REQ3) could be expressed in prose as REQ1 and either REQ2 or REQ3.

Ensuring that your requirements are testable

All requirements should be ***testable***. At least in principle, it should be possible to devise a test to verify whether a requirement has been correctly implemented in a product. In practice (usually for economic reasons), it may not be feasible to implement a particular test but, even in such cases, the requirement must be theoretically testable.

Testability and ***observability*** go hand-in-hand. If it is truly impossible to test a requirement, this implies that there is no possible way of observing the specified characteristic or behaviour, and therefore you should question whether that requirement is valid.

A requirement that appears to be testable because it has an observable outcome may, in fact, not be testable because it lacks precision or completeness. Consider:

When the equipment receives a SERVICE_REQUEST message, it shall respond immediately with a SERVICE_ACK message.

This is not a testable requirement because it does not specify how soon "immediately" is. Rewriting it as:

When the equipment receives a SERVICE_REQUEST message, it shall respond with a SERVICE_ACK message within 30ms

makes the requirement both implementable and testable.

Expressing optional requirements

You may find it necessary to make some of the requirements optional within a standard. However, this can result in imprecise standards, as each option may need to be considered in combination with every other option. The resulting lack of precision may well lead to interoperability or interaction problems in implementations of the standard. You should therefore aim to minimize the number of optional requirements although you may be unable to eliminate them altogether. In addition, you must ensure that your standard clarifies the consequences of whether or not a particular optional requirement is implemented.

Options can range from the optional implementation of entire features or capabilities (high-level requirements) to choices about the support of specific characteristics or behaviour (low-level requirements). Implementations that employ an optional feature must interwork with implementations that do not.

For example, when specifying the dimensions for equipment racking, it would clearly be unacceptable to specify that

The equipment may accept 19inch (482.6mm) wide front panels or 23inch (584.2mm) wide front panels.

Expressed in this way, the implementer has to decide which width to support, making it impossible to accommodate front panels of the size that he did not select.

If the intention is that the racking should be able to accept panels of either width, the following alternative texts make the requirement clear and remove any problems of implementation:

The equipment shall accept 19inch (482.6mm) wide front panels and may, optionally, also accept 23inch (584.2mm) wide front panels.

The equipment shall accept both 19inch (482.6mm) wide front panels and 23inch (584.2mm) wide front panels.

Optional features

If you need to include optional features in a standard, you should make this clear by using appropriate introductory text such as:

The following feature (or features) may be supported.

You should follow this statement by specifying the requirements that define the feature. Remember that the correct use of the verbs **SHALL**, **SHOULD**, **MAY** and their negative forms is crucial.

Optional requirements

A standard that includes optional requirements should clearly specify:

• under what conditions optional requirements apply and which other optional requirements may become 'mandatory' as a result of an optional requirement being implemented

• the consequences, if any, of omitting optional requirements.

Profiles

It is important to avoid an excess of options, but some standards may contain many options for good reasons. The development of profiles of the standard can help ensure interoperability between implementations. Profiles state which options must be implemented for a particular situation (effectively making those options mandatory) and which should not be implemented. A profile should itself be a standard. There may be a different profile of the base standard for each context in which it applies.

As an example, the DECT™ General Access Profile (GAP) specifies an interoperability profile for simple telephone capabilities that most manufacturers implement. There are several other interoperability profiles that specify the use of DECT in applications such as data services and radio local-loop services.

## Specifying requirements with specialized notations

Natural languages (English etc.) all have their limitations that can result in misunderstanding and confusion. Some requirements are difficult to express precisely in text and can sometimes be described more precisely and more effectively by using specialized languages such as ASN.1 (Abstract Syntax Notation 1) and UML (Unified Modelling Language®). They can be used either as the primary means of specifying requirements, or to complement the text of a standard in order to illustrate and clarify requirements.

In any case, you should only use languages that are internationally standardized and known in the industry, and you should respect their syntax and semantic rules. Dedicated tools exist for most of these languages, which can be helpful in detecting errors in specifications. ETSI's Centre for Testing and Interoperability can advise you about their use.

Readers need to be clear about which parts of a standard are normative and which are informative. Since specialized languages can be used for either purpose, it is important that the reader is not left in doubt. A statement such as the following will make the situation clear:

In the present document, ASN.1 is used for the specification of the content of all protocol primitives (normative) and service primitives (informative).

If both the English language text and alternative descriptions (in ASN.1 etc.) are considered to be normative, the text needs to make clear how any apparent misalignments between two descriptions will be resolved. Typically, this is done by stating which takes precedence.
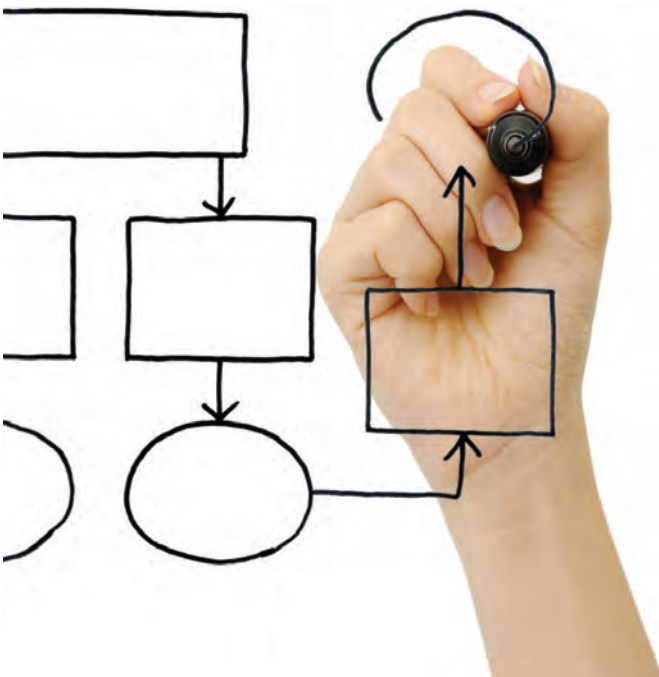
Advantages of XML and ASN.1
- They have well-defined syntax and semantics.
- Specialized tools are available to check the correctness and completeness of specifications written in XML and ASN.1.
- Specifications in XML or ASN.1 can be used directly in products implementing the standard, independent of the programming language used, thus eliminating the risk of conflicting interpretations.
- Accompanying standards exist that define how data values are encoded and transmitted. The tools can generate encoders and decoders.

Specifying interactions and use cases

Interaction is a fundamental part of communication standards. The text of a standard can provide a complete or partial description of an interaction, but the inclusion of a graphical description may help the reader considerably. Languages that are particularly suitable for this purpose are Message Sequence Charts (MSCs) and UML Sequence Diagrams. They are very similar and either may be chosen. Their simplicity and intuitive nature has made them popular. Despite their simplicity, they provide a powerful notation with built-in mechanisms to describe interactions that include such things as timers, loops, optional, alternative and exceptional behaviour. Some guidelines for the use of MSCs in standardization are given in EG 201 383; these guidelines can be extended to UML sequence diagrams.

UML Use Case Diagrams are particularly well-suited to the initial requirements capture phase of standardization, for documenting interactions between systems and with users.

Specifying messages

Communication standards enable devices to interoperate by specifying precisely the messages that can be exchanged, including their detailed sub-structuring and encoding. For this particular task, plain textual description is never adequate but, fortunately, two suitable languages exist, XML (Extensible Markup Language) and ASN.1.

Specifying the behaviour of communicating systems

If a standard describes complex interactions and timing patterns, it may be beneficial to use some form of state-oriented description to complement the text. There are two standardized notations particularly well-suited for this purpose, SDL process diagrams and UML state charts. EG 202 106 provides guidance for their use.

Validation of standards

If you have followed this guide, you will have included validation activities in your schedule. Even the simplest of validation activities can improve the quality of a draft standard but a well-planned and systematic validation process may identify technical and editorial inaccuracies, inconsistencies and ambiguities that might otherwise have remained in the published document. The process is therefore important for maximizing the quality of the standard as well as providing valuable feedback into on-going standards-making activities.

EG 201 015 details various validation techniques. These techniques are not all suited to the full range of subjects standardized by ETSI but almost every standard can be validated in one way or another. Two common validation methods described in EG 201 015 are peer reviews and interoperability events.

Peer review

A peer review is a method of evaluating specifications that can identify many of the inaccuracies and inconsistencies present in the specifications. In fact, reviews are effective in validating most types of standard, regardless of their content or presentation. The results of the review can provide feedback automatically into the standard itself as the rapporteur and other contributors should be present during the review.

The implicit impartiality of the ETSI Technical Officers makes them ideal candidates for leading reviews.

Interoperability events

During these events, companies are able to interconnect prototype or production implementations of standards in order to test for interoperability and, where needed, conformance to requirements. Interoperability events provide a highly cost effective and practical way of identifying inconsistencies in either an implementation or the standard itself.

Holding regular events, in time with the development of the standard and as new or enhanced implementations become available, is an excellent way to obtain prompt, relevant feedback into the standardization process.

ETSI's Centre for Testing and Interoperability (CTI) organizes interoperability events, known as Plugtests, for ETSI's technical committees and its partners. For more details visit: www.etsi.org/services/plugtests

Implicit methods of validation

In addition to these 'explicit' validation methods there are also a number of 'implicit' methods such as the preparation of a requirements catalogue, the development of test specifications and the implementation of the standard in a product; in other words, any activity which requires close scrutiny of the requirements specified in the standard. Although the explicit methods described above are the most effective means of validation, implicit methods may often produce results faster.

Maintenance and evolution of standards

Maintenance of standards

When a standard requires revision to correct defects ETSI publishes a new version. In the early stages of standardization, it may be possible to plan revisions at regular intervals because the number of change requests is likely to be high. These should diminish as the standard matures and a more ad hoc (or 'as required') approach to maintenance can be used. In any case, the efficient maintenance of standards requires an effective change management system to collect, evaluate and implement change requests.

Evolution of standards

Sometimes the capabilities or characteristics of a particular technology are so extensive that it would be unreasonable to delay publication of the standard until every aspect has been specified. It may then be advisable to publish the standard progressively as a carefully planned and evolving set of 'releases'.

Before the first release is prepared, a clear release strategy should be devised and documented. The content of each release should be identified and a schedule drawn up to indicate broadly when each release is expected to be published. It may not be possible or desirable to include all of the capabilities planned in each release, so the release strategy should have some flexibility and evolve with the standards. It should not be treated as a rigid set of requirements.

Maintaining compatibility

There is always a danger, even with planned revisions and releases, that an updated standard will become incompatible with earlier versions of itself, and also with other standards. Standards-writers should make compatibility one of their highest objectives.

Incompatibility should be less of a problem in a planned set of releases. However, although it is not desirable, incompatibility between releases can sometimes be an intentional part of a release strategy; in such cases the nature of the expected incompatibility should be made very clear from an early stage.

In addition to maintaining compatibility between versions of a single standard, it is also necessary to consider its compatibility with other standards. This can be more difficult to achieve because it is not always possible to know which other standards make normative references to the standard being revised. It is equally difficult to avoid incompatibilities caused by references to other documents, particularly if they are not ETSI standards.

Therefore:
- Before you decide to implement any change request you should review it in the context of the whole standard and those that are directly related to it.
- You should plan any change to the technical content of a standard together with any consequent changes to related standards in order to minimize periods of inconsistency or incompatibility.
- The use of hierarchical Work Items will help to identify the range of standards that might be affected by any given change request.
- Normative references to specific versions of other standards should only be used when necessary (i.e. when requirements depend on the content of a specific clause in the referenced document).
- Technical committees should systematically review the status and validity of the normative references in each of their published standards on a regular basis.

Maintaining numbering in new versions of standards

Be alert to the danger that re-numbering clauses, figures and tables in an updated version of a standard may cause problems when it is referenced by other standards. This is especially a risk where the document uses a contiguous unique numbering scheme. In each new version of the standard, the same clauses, figures or tables could possibly have different numbers compared with the previous version – a potential source of great confusion!

An alternative approach is to maintain the same number for each item (clause, table or figure) across all versions of the standard, to mark deleted items as void and to use alphanumeric suffixes for additional items (for example, if a new clause was added between clause *5.1.3* and *5.1.4*, it would be given the number *5.1.3a*).

Both methods have their merits and their drawbacks; the committee should decide which to use.

Change management

ETSI uses several approaches to change management. The simplest of these are oriented towards the collection, allocation and resolution of requests for improvement and defect reports. Other approaches are more concerned with the processing of fully marked-up changes to existing standards. It is the responsibility of each technical committee to select its preferred method and the tools to support it.

Irrespective of the method chosen, you should ensure that certain principles are respected:
- Change Management should be implemented for all deliverables once they have been approved by the relevant technical committee.

- Every request for a change should identify:
  - the name of the individual raising the request
  - the number, title and version of the deliverable for which the change is requested (this should be the latest published version of the standard and not an earlier version or an unpublished version)
  - the proposed change, either in summary or in detail
  - the reason for the change.

- Each change request should be evaluated to determine whether the proposed change is:
  - Necessary – does it correct a defect that prevents accurate or consistent implementation of the standard?
  - Desirable – does it extend the specification with a capability that is attractive to the market? Does it modify the approach to an existing capability, for example to improve its efficiency or effectiveness in an implementation?
  - Feasible – can the defect be addressed within the constraints of the technology, cost and timescale?

- No guarantees should be made to the proposers that every submitted request will result in the proposed change to the deliverable. Only those ultimately approved by the technical committee will be implemented in a subsequent revision of the standard.

Test specifications

It is normal practice within ETSI to publish test specifications for each of our base standards. This is particularly helpful for those specifying protocols or services. You should decide as early as possible in the development of the base standard whether you will also create test specifications.

If you follow the guidelines in this guide, your standards will contain testable requirements that will be easy to identify. Test specifications are derived from the requirements in the base standard so, the better those requirements are written, the better the test specifications will be. If you have used the verbs **SHALL**, **SHOULD** and **MAY** correctly, the task of categorizing the corresponding tests as being mandatory or optional will be reasonably straightforward.

ETSI test specifications define either conformance or interoperability test suites and typically consist of three components:

- An **Implementation Conformance Statement** (ICS) – a standardized proforma in which the requirements specified in a base standard are collected together into a table of questions (to be answered by an implementer or equipment supplier) with a limited set of possible answers. The most common example of a specific ICS within ETSI is the Protocol (specification) ICS, or PICS.
- **Test Purposes** – a prose description of what is to be tested
- A **Test Suite**, usually in TTCN-3 code.

Standardized testing has a narrower scope than proprietary testing because it can only use the normative interfaces. By contrast, proprietary testing will typically access internal functions and internal interfaces of the product that may not have been defined in the standard. Performance testing is a good example of this. Most standards have little to say about product performance: in the case of protocols, this is usually expressed in the standard as response times to certain messages whereas proprietary tests will also handle characteristics such as load, throughput and capacity.

Conformance and interoperability testing

Conformance testing determines whether a single product or service conforms to the requirements specified in the standard(s) that it implements. The development of all ETSI conformance test standards should follow the well-established methodology defined in ISO/IEC 9646 *Conformance Testing Methodology and Framework*.

Interoperability testing considers operations at the functional or user level. Its purpose is to determine the extent to which multiple implementations of a standard (or a set of standards) will interoperate. It is concerned less with strict adherence to the requirements specified in the base standard and more with the overall functionality implied by that standard. EG 202 237 contains the methodology that should be followed when creating interoperability test standards.

Closing remarks

You are not alone!

Do not hesitate to call upon the support and assistance of the Secretariat. There is a wealth of experience and expertise for you to draw on. Over the years ETSI has produced more than 30.000 standards and enabled key technologies that have revolutionized telecommunications and IT. It is only by sharing our skills and experience that we can write the world class standards that our industries need.

Within the ETSI Secretariat we provide a range of resources that are there to help you. These include:

* The Technical Officers and support staff assigned to each committee, who have extensive technical and practical knowledge in the areas in which those committees work. They can guide you to other sources of expertise as necessary.

* The *editHelp!* Service, for matters relating to the drafting, approval and publication of your standard

* The Centre for Testing and Interoperability (CTI) which can help you ensure that the requirements in your standard are testable in practice, as well as providing expertise in the use of specification languages and the production of test specifications

* Secretariat staff dedicated to matters such as relations with the European Commission, relations with other partner organizations, use of standards in regulation, and the use and interpretation of the ETSI Directives

* The Specialist Task Force (STF) service, which can arrange dedicated teams of technical experts to work on specific standards-related tasks

* The ETSI website and Portal, which contain a wide range of information about the work of ETSI, its standards, meetings, key people and much more

* Numerous guides and other resources, some of which are listed in the References and Resources section of this guide.

We could wish you "good luck" but luck has very little to do with producing world class standards! It takes dedication, care and commitment – and a respect for the rules and good practices. We wish you every success!

References and useful resources

ETSI resources

ETSI Drafting Rules (EDR)	http://portal.etsi.org/edithelp/home.asp
ETSI Directives	http://portal.etsi.org/Directives/home.asp
Use of English (interactive)	http://portal.etsi.org/edithelp/Files/other/UOE.chm
Use of English (PDF)	http://portal.etsi.org/edithelp/Files/pdf/Use%20of%20English.pdf
Making Better Standards	http://portal.etsi.org/mbs/
Standards-making process	http://portal.etsi.org/chaircor/process.asp
Terms and definitions database	http://webapp.etsi.org/Teddi/
Editing checklist	http://portal.etsi.org/edithelp/Files/zip/ETSI_editing_checklist.zip
Testing and interoperability	http://www.etsi.org/services/testing-interoperability-support
ETSI Seminar	http://www.etsi.org/news-events

The *editHelp!* web pages (http://portal.etsi.org/edithelp/home.asp) contain a wide range of guides, tools and skeleton documents to help you in the standards-making task.

Various documents relating to methodologies and best practices are available at http://portal.etsi.org/edithelp/StandardsDevelopment/home.htm?page=Documentation.

3GPP™ resources

TR 21.801	Specification drafting rules
TR 21.905	Vocabulary for 3GPP Specifications
TS 21.101	Technical Specifications and Technical Reports for a UTRAN-based 3GPP system
TS 41.101	Technical Specifications and Technical Reports for a GERAN-based 3GPP system

ETSI Guides on standards-writing, testing and validation

EG 201 058	Methods for Testing and Specification (MTS); Implementation Conformance
EG 201 383	Use of SDL in ETSI deliverables; Guidelines for facilitating validation and the development of conformance tests
EG 201 399	A guide to the production of Harmonized Standards for application under the R&TTE Directive
EG 201 015	A Handbook of validation methods
EG 202 106	Guidelines for the use of formal SDL as a descriptive tool
EG 202 107	Planning for validation and testing in the standards-making process
EG 202 237	Internet Protocol Testing (IPT); Generic approach to interoperability testing

ITU-T three-stage process

ITU-T Recommendation I.130: Method for the characterization of telecommunication services supported by an ISDN and network capabilities of an ISDN

Specification languages

ITU-T Recommendation Z.100-Z.106 (2011)   Specification and Description Language

ITU-T Recommendation X.680-X.683 (2008) | ISO/IEC 8824-1 to 4:2008
                                         Information technology – Abstract Syntax Notation One (ASN.1)

ITU-T Recommendation X.690 (2008) | ISO/IEC 8825-1:2008
                                         Information technology – Abstract Syntax Notation One (ASN.1):
                                         Information technology – ASN.1 encoding rules: Specification of Basic
                                         Encoding Rules (BER), Canonical Encoding Rules (CER) and
                                         Distinguished Encoding Rules (DER)

ITU-T Recommendation Z.120 (2011)   Message Sequence Chart (MSC)
OMG (2011): Unified Modeling Language UML, v 2.4.1
W3C (2008): Extensible Markup Language (XML) 1.0
W3C (2006): Extensible Markup Language (XML) 1.1

Other resources

ISO/IEC 9646   Conformance Testing Methodology and Framework

Oxford Guide to English Usage (ISBN: 9780192800244)

02/0314