An introduction to

Achieving technical excellence

Contents

Introduction

For nearly 30 years ETSI has maintained a reputation as a global leader in standardisation.

ETSI standards are used the world over and have enabled technologies which have changed the way people live, work and do business. We produce about 2500 standards every year and our work touches almost every area of modern living. Many of today's information and communication technologies (ICT) owe their success and widespread adoption to the dedication and diligence of ETSI and our members.

One of our deepest commitments is to *quality* – the quality of our standards and processes, but also of the products and services that they define.

ETSI is an organisation built on innovation. As we grew, we did things that other standards bodies were not doing – and what we did made a difference. Our creative approach included giving high priority to specification methodology, the validation of our standards and specifications, and the importance of testing.

This document outlines ETSI's approach to these critical tasks and points you towards practical steps that you can take to achieve technical excellence in your standards-making.

However times are changing and yesterday's achievements, though still valid, are having to be extended to address the impact of new technologies, new technical communities with different working methods and the ever-accelerating pace of innovation and customer demand. The ways in which technologies are being developed are changing too. Today they are characterised by greater flexibility and dynamism, and they are challenging some traditional standards-making methodologies. In particular, standards-making is being complemented by 'implementation-driven' standardisation that reflects the development processes of the 'Open Source' world.

So this document also gives a brief introduction to these new trends and how ETSI's approach to standardisation and testing is adapting to its evolving environment.

Achieving world-class products

Standardisation provides a solid foundation upon which to develop new technologies and to enhance existing practices. Technical standards lay down the essential parameters for products and systems, and also help designers and manufacturers understand the intended functions and interactions of the devices they are creating. In fact, standardisation can *drive* technology.

Standards are equally important for ensuring safety, reliability and environmental care.

Standards are a key facilitator of interoperability, giving manufacturers a wider potential market. Manufacturers can benefit from economies of scale; consumers can benefit from reduced prices and greater choice. Wider markets stimulate innovation and increase demand. Interoperability is therefore an essential factor in the success of modern technologies.

The testing of products and services before they are placed on the market is crucial for quality assurance and may also be necessary to demonstrate compliance with legislation. Standards provide the benchmark against which something can be tested. Clearly the quality of the actual standard is critical to the effectiveness of the testing.

Ensuring standards that work

ETSI standards are used to implement ICT products according to agreed requirements. Our *Guide to Writing World Class Standards* shows that well-defined, clear and unambiguous requirements are essential if implementers are to build high-quality, interoperable products.

However, technical excellence is more than just a paper specification – it also means that a standard must be rooted in real practice, that it can be shown to work and that it is relevant to real product development. Of course, the deep expert knowledge present in the ETSI technical committees ensures that this is mostly the case. But, like any human engineering endeavour, “seeing is believing”. To achieve technical excellence, standards themselves need to be validated, whilst additional standards may be needed to describe how the testing of products and services should be carried out. This document shows how ETSI technical committees use practical working methods to confirm that their standards really are fit for purpose.

The rise of Open Source

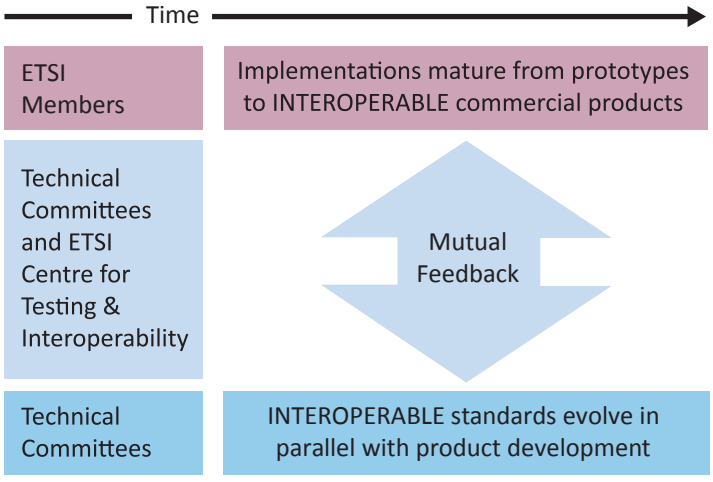
The availability of Open Source Software (OSS) and the creation of OSS projects are growing rapidly. Open Source offers many advantages, such as innovation, fast development and the involvement of a committed and creative global community.

On the other hand, standards are valued widely for providing long-term stability, wide consensus and a cohesive view of large and complex systems. They also ensure interoperability, inspire confidence in products and services and offer economies of scale.

So, will standards and standards bodies such as ETSI be relevant in this ubiquitous and ever expanding world of Open Source (which now goes well beyond just software)? ETSI believes the answer is a firm yes, even more so when traditional standards-making and Open Source work together. Whilst some might see these two approaches as being in competition, we firmly believe that they are complementary. Later in this document we explore the value of co-operation between the two communities.

The necessity for feedback

In practice, standards and products based on those standards should evolve in parallel, each contributing to the development of the other. Through many years of experience the ETSI technical committees know that integrating validation and testing activities into the standards development process can contribute significantly to the production of interoperable standards and, ultimately, to the release of interoperable products based on those standards. Mutual feedback between the processes helps to maximise the quality of both the implementations and the standards, notably with respect to interoperability, which is so vital to many of today's technologies.



Mutual evolution of products and standards

With correct timing, especially in the early stages, validation and testing activities can also reduce the overall development time of a standard, leading to the earlier placement of interoperable products on the market. This is especially true in the case of technologies that rely on multiple standards which may come from a variety of sources.

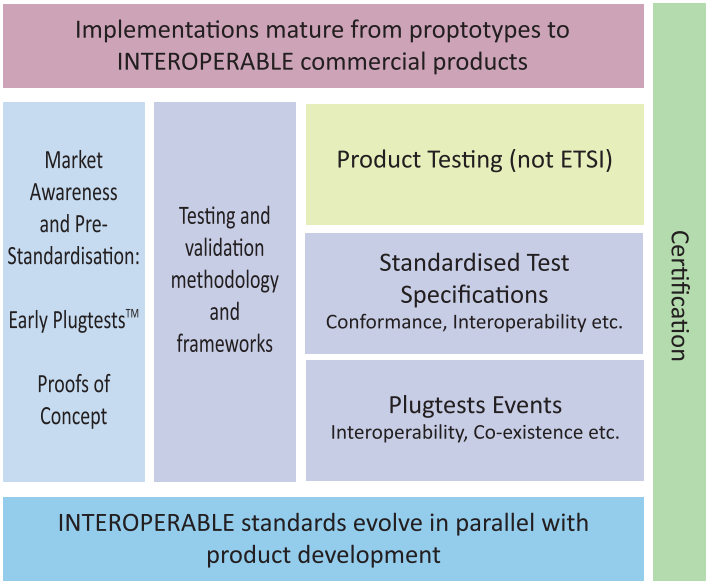
Validation and testing activities offer many benefits, including:

- improved quality of standards and products
- greater guarantee of conformant, interoperable and robust implementations
- shared feedback among product developers, standardisers and testers
- enabling third party certification programmes.

ETSI does not carry out product certification. However, our test specifications may be used by third party certification schemes.

Reaching excellence through validation and testing

What we have described rather vaguely above as 'mutual feedback' is of course quite complex in reality. As can been seen in the diagram below, a whole range of different approaches and physical tools can be applied to help reach the goal of technical excellence. ETSI does not follow a rigid methodology. Rather, we adapt our approach to different needs and levels of ambition. In some cases, a very strict process needs to be applied. In others, a more relaxed, but nonetheless systematic, course of action can be followed. Flexibility and suitability are the watchwords.



Bridging the gap

Whichever approach is adopted, the key is to plan ahead. The activities involved in validating a standard and defining test specifications should be interleaved with the development of the standard itself. Even if validation and testing seem to be far off on the horizon, recognising that either one or both may be needed should be taken into account, even if only to ensure that you are writing requirements which are testable!

Encouraging market awareness

The complexity of modern technologies, plus the high costs often associated with their development, creates a need for early verification of their viability. Pre-standardisation activities such as Proof of Concept are proving valuable tools in the early stages of technology development and play an important role in kick-starting standardisation work.

As a major tool for spawning and fostering emerging technologies, the timing and quality of standardisation is critical. The task of ensuring the high quality of specifications is made all the more difficult in that market opportunities are dynamic – so the perception of market requirements may change during the development of a specification.

The standards-making process must therefore be flexible and adaptable, able to take account of changing requirements. However, stability is also essential and the standards-making process will suffer if it is allowed to become too erratic. The resulting specifications must remain consistent and meet well-understood and documented requirements.

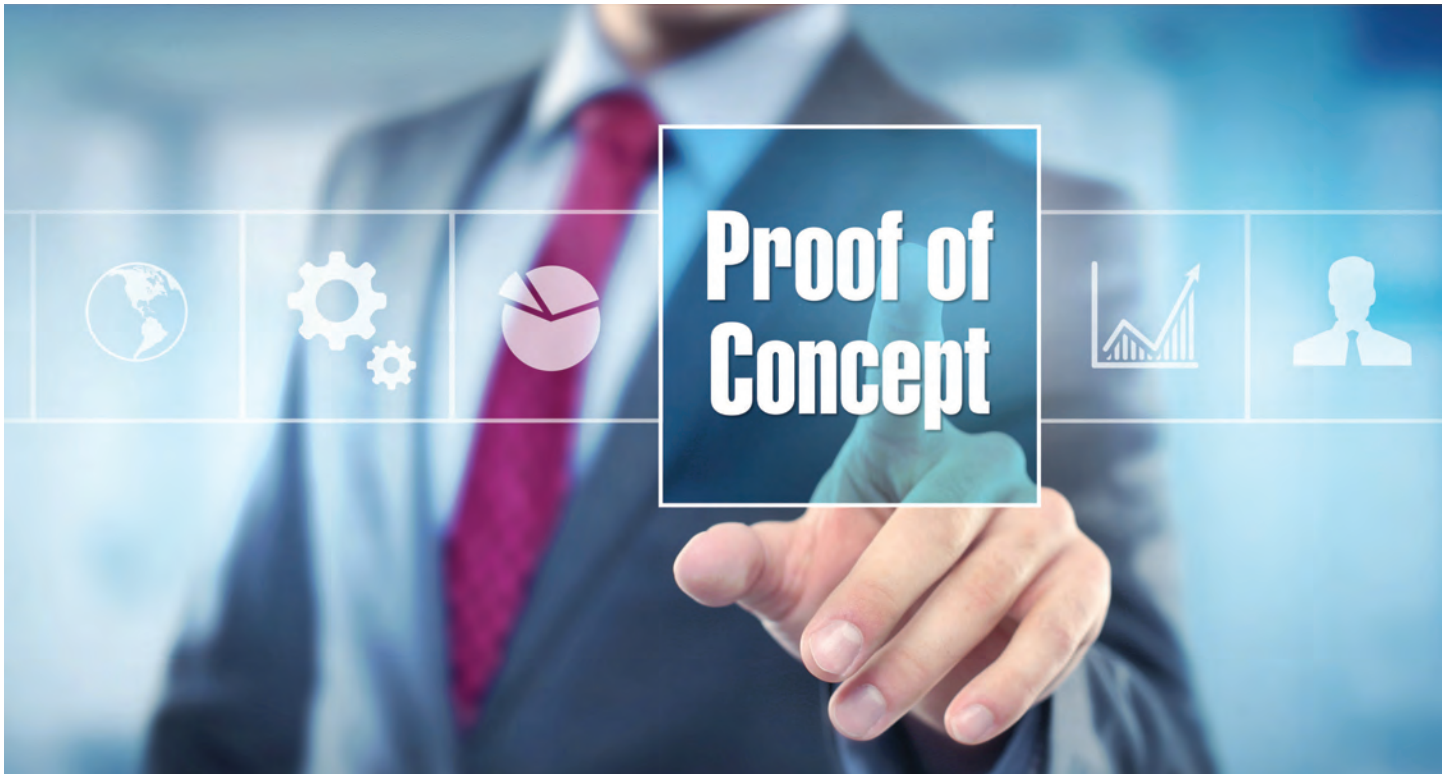
Proof of Concept

A Proof of Concept (PoC) is an important means of demonstrating the viability of new technologies, as well as creating awareness of their commercial scope. PoCs show that ideas and technical approaches are feasible and that the standardisation work is relevant. ETSI has created a generic Framework for PoCs to facilitate their wider adoption.

Within ETSI, our Network Functions Virtualisation (NFV) Industry Specification Group (ISG) has been a leader in the use of Proofs of Concept. To date over 40 PoCs have been created which have been effective in illustrating key aspects of ETSI's NFV work. They have demonstrated NFV as a viable technology and have helped to develop a diverse, open and interoperable NFV ecosystem. Results are fed back to ISG NFV to improve its technical specifications.

Results from PoCs can identify issues relating to interoperability and other technical challenges, which can be addressed in the related standardisation work. PoCs help build communities and consensus and drive the standardisation forward. They can act as an early promotional tool and as an excellent public showcase for the state of the technology.

Early Proofs of Concept or ideas for standards could be developed using hackathons. Hackathons (or hackfests) are a valuable activity where developers come together to collaborate intensively on technical challenges. Hackathons usually last a few days and participants have the freedom to work as they wish.



Validation – Is this the standard we want?

In this document we use the term *validation* to describe any means of checking whether a standard really does specify the requirements as intended by the technical committee producing it – and whether it does it well. In other words: will the standard do the job it is supposed to do? Validation should be an ongoing activity throughout the drafting process and not something carried out just prior to publication when it may be too late to put things right.

Validation is a process which gives evidence that a standard can be fully implemented and that it is able to provide the intended level of functionality and performance at minimum cost. Validation has several objectives:

- **Produce the standard more quickly and cheaper by detecting errors early.** Errors in standards which are detected late in the development lifecycle are very expensive to fix and may lead to unacceptable delays in publication – and the standards may then miss the market!

- **Secure the investment made by product and service suppliers.** Suppliers need to be assured that they can base their development on sound and stable standards, that their investments will not be wasted because the standard contains errors only discovered after its publication, or because it was incomplete.

- **Make the testing process cheaper.** Preparation of test specifications will frequently bring to light flaws in a standard since the task involves a kind of technical review of the standard. Naturally, the production of test specifications will be easier and cheaper if the base standard has already been validated. The test specifications themselves also need to be validated to detect any errors: this will ensure a faster, cheaper and more reliable test.

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. ETSI Guide EG 201 015 details various validation techniques. These techniques are not all suited to the full range of subjects standardised by ETSI but there are few, if any, standards that cannot be validated at all.

Validation can be explicit – activities where the main purpose is validation itself, such as peer review or interoperability events – or implicit, where the main purpose is something other than validation, but with validation as a by-product. Examples of implicit validation include 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.

The nature of standards development implies a continuous process of iteration and reviewing of the draft document. Properly conducted, this process can make an effective contribution to validating the draft. However, greater rigour can be achieved by dedicated (explicit) validation activities.

Peer review

A peer review is a method of evaluating specifications that can identify many of the inaccuracies and inconsistencies that might be 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.

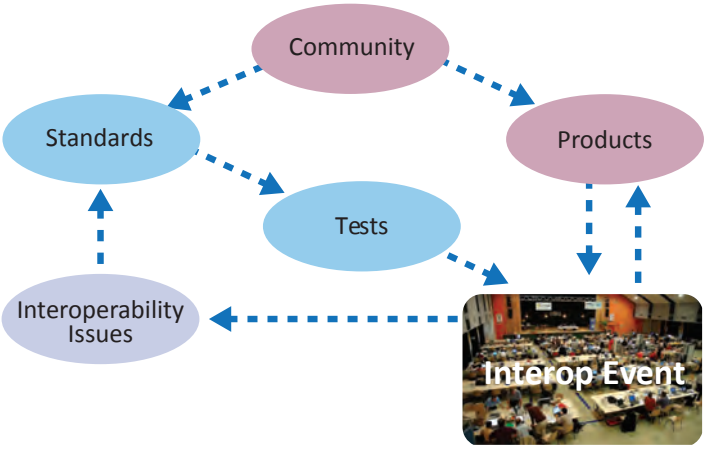
The implicit impartiality of the ETSI Technical Officers makes them ideal candidates for leading reviews.

ETSI interoperability events

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

The organisation of regular events, in step with the development of the standard and as new or enhanced implementations become available, is therefore an excellent means of obtaining prompt, relevant feedback into the standardisation process.

ETSI has run a comprehensive Plugtests programme since 1999, performing on average 12 events per year and covering well over 60 different technologies.



Feedback loop for interoperability events

Please visit www.etsi.org/about/what-we-do/plugtests for more information.

Testing – Is the product doing it right?

The term *testing* is used in this document to describe any means of checking whether a product adheres to a certain set of criteria. Many ETSI technical committees develop standardised test specifications as a complement to the base standards. These tests have many uses, for example by implementers as an integral part of their proprietary development processes or by third-party regulatory or certification programmes.

Writing world class test standards

In ETSI we usually produce test specifications for each of our base standards – this is particularly helpful for those standards that specify protocols or services. A well-written base standard will contain testable requirements that will be easy to identify. Test specifications are derived from these requirements so, the better these requirements are written, the better the test specifications will be. Our *Guide to Writing World Class Standards* provides a great deal of useful advice on this subject. For example, if you have used the verbs SHALL, SHOULD and MAY correctly, the task of categorising the corresponding tests as being mandatory or optional will be reasonably straightforward.

The criteria for a world class testing standard are essentially the same as for a good base standard, namely:

- it will **respond to real needs in a timely manner**
- it will be **complete and accurate**
- it will be **easy to understand**
- the requirements it specifies will be **clear and unambiguous**
- it will be **validated**
- it will be **maintained** regularly
- and, specifically for a test specification, it will clearly define **what** is to be tested and **how** to perform the tests.

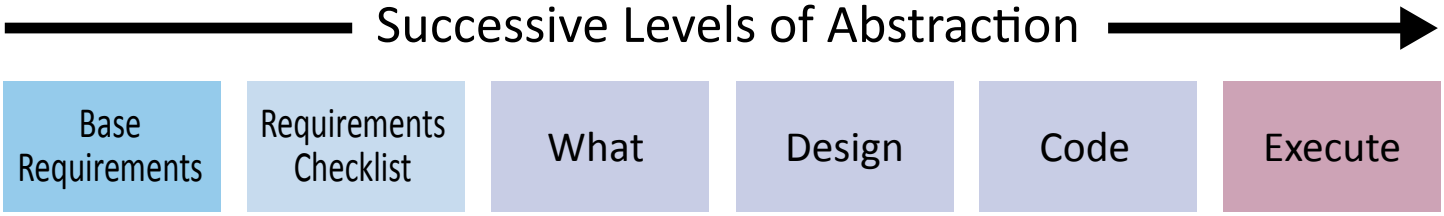
ETSI's Centre for Testing and Interoperability (CTI) works with our technical committees to set up test programmes and to assist the development of test specifications which, like any other standards, can be freely used by industry. However, the actual testing of a product is, for the most part, outside ETSI's responsibilities.

ETSI test specifications – an example of good practice

ETSI test specifications define either conformance or interoperability test suites and typically include the following components:

- a **Requirements** or **Features checklist** – a standardised pro forma (such as a Protocol Implementation Conformance Statement (PICS) or an Interoperable Functions Statement (IFS)) in which the individual requirements or features specified in a base standard are collected together
- a set of **Test Purposes** – a prose description of what is to be tested
- a set of **Test Descriptions** – a more detailed specification of how to carry out the test to achieve the purpose
- a **Test Suite** – a collection of test scripts, usually in TTCN-3 (Testing and Test Control Notation version 3) code.

In all cases we follow a top-down approach of successive levels of abstraction as illustrated below, with the ultimate goal of enabling the creation of executable code that performs the required tests. While there may be more emphasis on certain steps than others, with some steps possibly omitted, we never go straight from requirement to coding. This principle applies to all test development whether it be for conformance, interoperability, co-existence, performance or whatever.



Increasing levels of abstraction from base requirements to test execution

All of our test specification activity follows the principle of Black Box testing, which must be done over normative (standardised) interfaces, i.e. interfaces that have been specified in the corresponding standard. It is also important to ensure that:

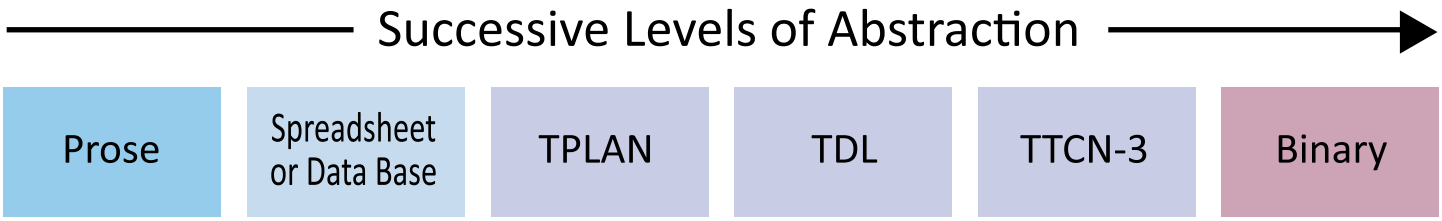
- the tests are **useful and timely** (for example with respect to maturity of the standard, the availability of products and the availability of suitable test equipment)
- they are **relevant** to individual (proprietary testing) needs as well as to the industry at large (e.g. for formal approval testing)
- there is a **good balance** between coverage, quality and cost (of development)
- the experience of developing the tests is **fed back** into the relevant standards committee (implicit validation).

Let's now take a look at the various steps in the process.

Different tools for different steps

Each of the test specification components listed above is meant for a different audience and group of users. ETSI's Technical Committee on Methods for Testing and Specification (TC MTS) has standardised three specialised languages for these components – TPLAN (Test Purpose Language), TDL (Test Description Language) and TTCN-3.

Tool support for these languages is vital. At present the greatest availability of tools is for TTCN-3 (development environments and compilers). TDL tools are starting to appear and the next version of TDL (currently being specified by TC MTS) will enable direct generation of TTCN-3 from TDL scripts. It will also incorporate the capabilities of TPLAN, thus providing a seamless development environment from Test Purposes to executable code.



Different languages or formats for different stages of test development

Drawing up a requirements checklist

The starting point for the development of any test specification is its relationship to the base standards. This is done by collecting and categorising the individual requirements (for example whether they are mandatory or optional). Often this takes the form of a tabulated set of questions that can usually be answered by an implementer with a simple 'yes' or 'no', or by entering a specific value, such as a frequency, timer value or message length. This information is best presented in a standardised pro forma, such as an Implementation Conformance Statement (ICS), an Interoperable Functions Statement (IFS) or possibly in a database.

The ICS or IFS have several roles:

- as input to the test developer on which tests to write and to link them to the requirements in the standard
- to indicate whether a test is mandatory, optional or conditional (a conditional test is one that becomes mandatory under certain conditions)
- as input to the test process for a particular implementation where tests can be selected or deselected as appropriate and to set product-specific values
- for use in a procurements process to evaluate which requirements or features have or have not been implemented.

Specifying what needs to be tested

Test Purposes are described in prose or a prose-like, but structured, use of English in TPLAN. The Test Purposes are intended for the standards-developer community as much as for the testers. They are often used to facilitate a sensible discussion of the tests with the creators of the base standards without going into any implementation details.

A conformance Test Purpose describes in broad terms *what* is to be tested. In the simplest case it will be related to a single requirement in the corresponding standard. For a typical conformance test this will usually be a low-level protocol aspect (for example testing one or more parameter values over a sequence of message interactions) or, in the case of radio frequency (RF) or electromagnetic compatibility (EMC) testing, measuring a particular physical characteristic, the potential value or values of which are well-defined in the base standard.

For an interoperability test the requirement will most likely be a high-level functionality which implicitly will cover many detailed protocol requirements, for example setting up a secure connection between two parties.

In both cases, the Test Purpose will define the criteria upon which verdicts (such as a PASS or FAIL) can be assigned.

Designing a test

Test descriptions are more detailed and form the bridge between the Test Purposes and the TTCN-3 code. The form of TDL can be both textual and graphical. Again, this form is useful for the protocol specifiers as well as the test designers. In a case of interoperability testing (which often cannot be automated), this is the actual form of the test script to be manually executed at the time of testing.

For more information visit: www.etsi.org/technologies-clusters/technologies/test-description-language.

Writing test code

Finally, the TTCN-3 code is written and used by test programmers and the test implementers. It is a fully specified piece of code defining all data and behaviour paths necessary to execute the test, and includes the handling of error conditions. TTCN-3 code can be compiled and executed on a real test system.

The official ETSI website for TTCN-3 can be found at: www.ttcn-3.org.

Validation of ETSI test specifications

Test suites cannot easily be validated by means of a design review as they are written in TTCN-3 or some other programming language. A manual method can partially validate the structure and coverage of a test suite but the correctness and effectiveness of a TTCN-3 test suite can only be validated by execution of the code in either a simulation or a controlled testbed.

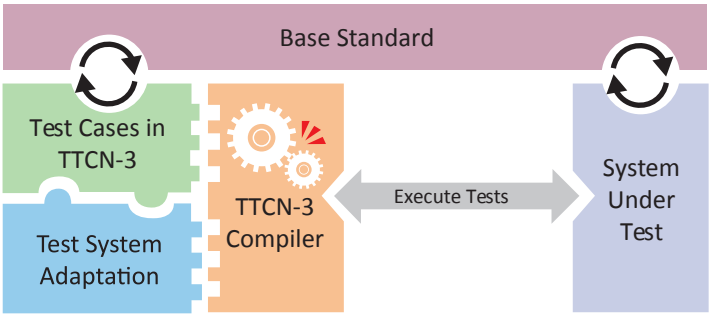
Nevertheless, in order to ensure high quality executable test suites some form of practical validation is essential. ETSI's CTI has defined validation levels for standardised TTCN-3 test specifications which can be selected, as appropriate, by our technical committees when developing a particular test suite:

- **Level 1:** The test suite is successfully analysed and compiled on one or more TTCN-3 test tools. Optionally, a design review of the test suite is performed.
- **Level 2:** The tests are executed on one test platform against at least one System Under Test (SUT). Where possible, tests will be run to completion. Optionally, a back-to-back validation (mirror test cases) and/or a data-driven codec validation are performed.
- **Level 3:** The tests are executed on several test platforms against several SUTs. Tests will be run to completion and trace analysis will be done.

Annex B of ETSI Guide EG 201 015 contains further details of the validation processes for these three levels.

The diagram below shows an example of how TTCN-3 test cases are validated. It comprises the Test Suite which is the collection of all, or a subset of, the Test Purposes implemented as TTCN-3 Test Cases. Another building block of this framework is the Test System Adaptation, which provides an interface for various plugins to allow the Test Suite to communicate with the System Under Test (SUT). The TTCN-3 source code and the Test System Adaptation source code are compiled with a commercial or Open Source TTCN-3 compiler. Execution of the code both validates the test specifications and provides an assessment of SUT compliance to the base standard.

All the source code is managed and made available via the ETSI Source Forge for test specifications. This allows the community to monitor, contribute, feedback and use the test system for its in-house testing.



Example of validation of TTCN-3 Test Cases

Combining conformance testing with interoperability testing

Conformance and interoperability testing are both important and useful approaches and it is unlikely that one will ever fully replace the other. Both have their strengths, and it can be useful to consider using the techniques together to give a consolidated result. Performing limited conformance monitoring and analysis during interoperability events can provide an added level of technical confidence. In some cases, the conformance analysis can be done in real-time; in others an offline approach may be preferred. ETSI's CTI can provide guidance in this complex area.

Planning for validation and testing

The complexity of the tasks of validation and testing is such that careful planning is essential. Before even getting to the planning stage, a number of questions may need to be considered. These include:

- Do you want to run a practical validation programme?
  - If so, bear in mind that interoperability events have three key advantages:
    - they validate the technical content and quality of the standard
    - they demonstrate and enable interoperability
    - they test products

- What sort of testing, if any, is anticipated?
  - Interoperability, conformance or both?
  - Will it support some third-party approval scheme?

- Will you use a dedicated Working Group or will the task be integrated as a Work Item in an existing Working Group?

- Will you require an ETSI Specialist Task Force (STF)?

Other issues to consider when planning for validation and testing include:

- the degree of maturity of the base standard
- the intended or probable availability of products.

In most instances validation can start before formal conformance testing. Similarly, test specifications can be developed once the testing requirements have been established and need not wait for the base standard to be finalised.

Adding structure to the task

The task of developing test specifications should be treated as a project. Defining a validation and test framework is one way to handle this. Typically the framework will contain all the background and planning information related to the project. It is a living document or set of documents in which some of the information and material is necessary in the early stages of the project, whilst other parts will be added as the project evolves.

Typically a validation and test framework will include:

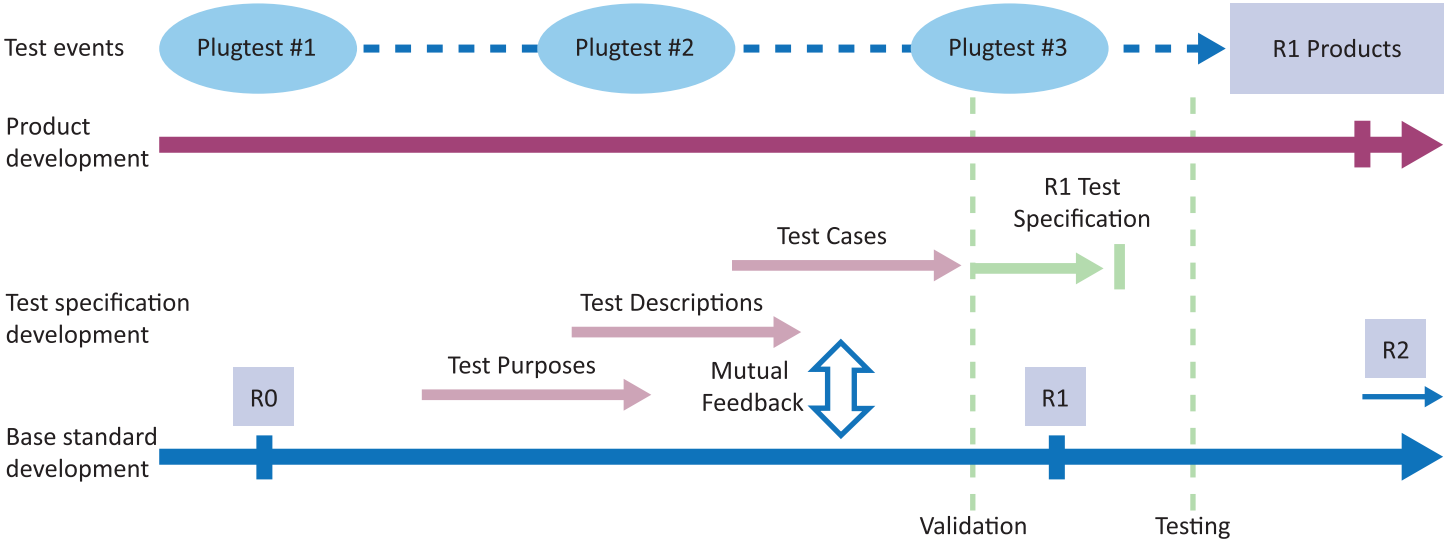
- a list of base standards
- a list of testing standards to be produced (conformance and/or interoperability)
- a roadmap or timeline for delivery of test specifications and interoperability events, preferably well-aligned with the standards release schedule
- the methodologies to be used
- applicable test methods
- a description of the basic test configurations that will be used
- languages and tools needed
- a description of test platforms that will be required
- a list of possible implementations to test.

Aligning testing and validation with standards development

As has already been noted, one key aspect of the testing framework is getting the timing right. If work on the test specification begins too early, while the standard is immature, there is the risk of having to perform constant updates, making the process very inefficient and error-prone. On the other hand, starting too late risks not having the tests available when needed.

The figure below illustrates the ideal case for conventional standardisation. In this example, definition of the testing requirements starts at a time when Release 1 (R1) of the base standard is becoming stable. This needs to occur early enough to provide useful feedback to the standards-makers but it must not be so delayed that Release 1 of the test specification is published too late to be of use to the product implementers.

In parallel, interoperability events can begin as soon as there are enough (early) implementations to make an event viable.



Alignment of testing and validation with conventional standards development

Standardisation and Open Source

As mentioned earlier, the ICT industry is increasingly implementing interfaces and components using Open Source Software (OSS), with the support of major industrial and commercial players. OSS development typically employs a faster, less formal approach compared with older technologies, and this has consequences for establishing standards, including their validation and testing.

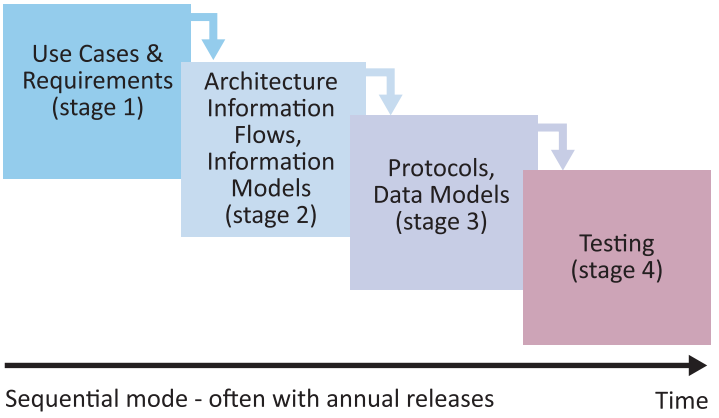
The traditional ETSI approach

Standards usually remain stable over time and, unless they have been developed in support of a particular technology platform, they tend to be technically neutral, allowing innovation to take place in implementations. The ETSI standards development process is mostly *linear*, and is heavily meeting-based. In the ideal case, a three-stage approach such as the following is used:

- Stage 1: High-level user requirements
- Stage 2: Architecture and functional model(s)
- Stage 3: Detailed (concrete) requirements (protocols, services or APIs)

Testing may be considered as a "Stage 4".

The order of this development is mostly sequential. Of course, the work is iterative to some extent, but a certain stability of



Idealised 3-stage approach

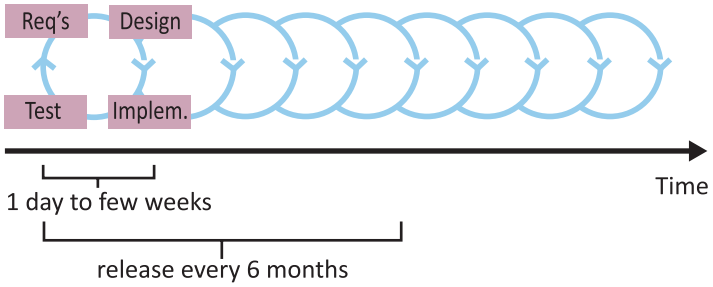
stage 1 is needed before stage 2 is started and stability of stage 2 is needed before stage 3 is started. Complex end-to-end systems need to be broken down into manageable components (features), each with its own stage 1, 2 and 3 release cycle. A particular compilation of many features, comprising all the stages 1, 2 and 3, constitutes a major release.

The picture is completed by the so-called fourth stage – testing. In most cases, the testing and validation activities described previously will not begin much before the start of stage 3. Note, however, that pre-standardisation and market awareness initiatives such as PoCs should be started as early as possible, ideally during stage 1 development.

The OSS approach

Open Source solutions rely on a community that is responsible for the development, provisioning and maintenance of the software. This community is usually agile, innovative and fast-moving, often looking to solve immediate technical issues without consideration of a related standard. Indeed, by intention, a definition of the end product may not even be available at the start and good documentation may not be the priority. In such cases the OSS often emerges to become the *de facto* standard.

The OSS approach is *highly iterative* and work is mostly done on-line. Code is usually developed very rapidly, and contributions are incrementally integrated into the main repository branch and tested at a rate that is usually much faster than typical standards development. In short, OSS development does not follow the 'waterfall' model described above – by its very nature it is agile.



Idealised OSS approach

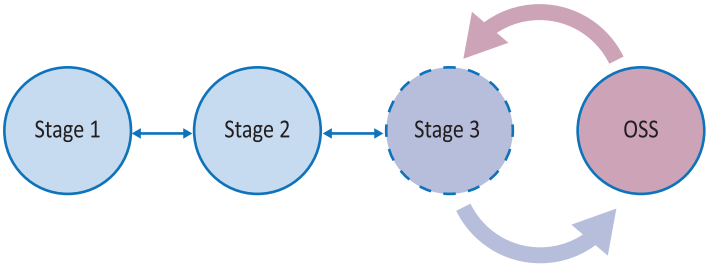
Implementation-driven standardisation

As we have discussed, traditional standardisation works at a different pace to OSS because it is consensus-driven and thorough. The focus is on specifying normative requirements which are well separated from any particular implementation. Iteration periods (the time between versions of standards) may be many months or even longer, whilst for OSS they may occur daily or even every few hours.

However, Open Source offers a way to develop solutions collaboratively and in 'real-time' with the development of an associated standard. If the respective standardisation and Open Source communities are to collaborate, for instance to share their solutions or expertise, they will have to find ways of aligning their working processes as far as this is possible. The word *aligned* is used intentionally here – both approaches have their strengths and it should not be the intention of one to change the other.

To some extent the OSS process is not so different from the ETSI process except the focus of attention at any given moment may be narrower and the turn-around time much quicker. The

standards community's response may imply adopting a more implementation-driven approach. The important thing is that, if an OSS – Standards collaboration is to be successful, there needs to be an almost continuous feedback loop between the two communities.



Implementation-driven development of a Stage 3 specification

In the illustration above an early stage 3 specification may or may not exist. The development of code can help by speeding up or even driving the standardisation, potentially leading to a Reference Implementation.

An implementation may be considered to be a Reference Implementation if:

it fully and correctly implements the standardised requirements – demonstrated, for example, by an ETSI acceptance (compliance) testing programme;

and

it has a strong reputation among the developer community for quality (a smart implementation, good performance and robustness, well-structured code, high-quality documentation etc.) and is regarded as a de facto *basis for commercial implementations.*

Open Source and testing

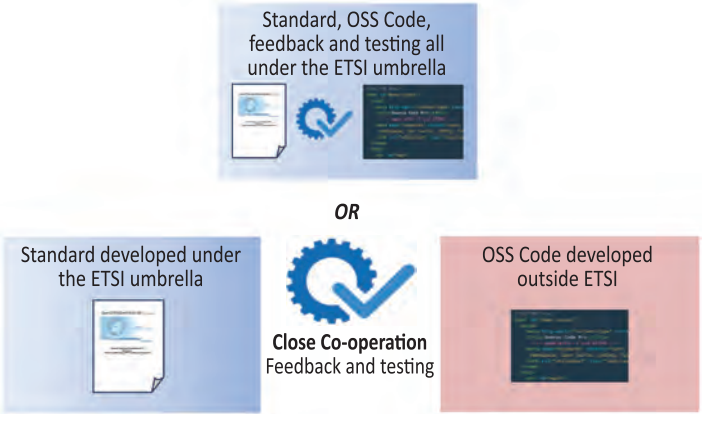
It is clear that the respective approaches of Open Source and traditional standardisation present very different constraints for requirements setting, consensus, validation and testing. Among the areas in which traditional standards-making is already adapting with greater flexibility and dynamism is that of testing. Increasingly, planning for validation and testing takes account of the following:

- Open Source activities relating to requirements definition, design and implementation are highly iterative – validation and testing need to respond at the same pace
- frequently, complete specifications may not exist or may not be up to date – testing strategies need to be devised to cope with this
- testing of the OSS as a stand-alone package may only be partially useful – further testing may need to be applied to products which integrate the OSS
- flexibility is part of the fundamental nature of Open Source – validators and testers need to be prepared for the possibility of requirements changing frequently and at short notice.

Testing is an integral part of an agile quality assurance (QA) approach and most OSS projects will include unit, integration, performance, robustness and internal compliance testing programmes, supported by test specifications, tools, test platforms, training, methodology and even testing services.

However, one of the strengths of OSS, its potential diversity, is also a potential weakness. Fragmentation may lead to non-interoperable solutions. Therefore, from a standards perspective, it is necessary to add testing for conformance to the base standard, plus interoperability testing between different products that may use the Open Source (OS) code.

Two models of co-operation are envisaged, as illustrated in the figure below where the Open Source activity may be hosted within ETSI or in close co-operation with an external Open Source project or foundation.



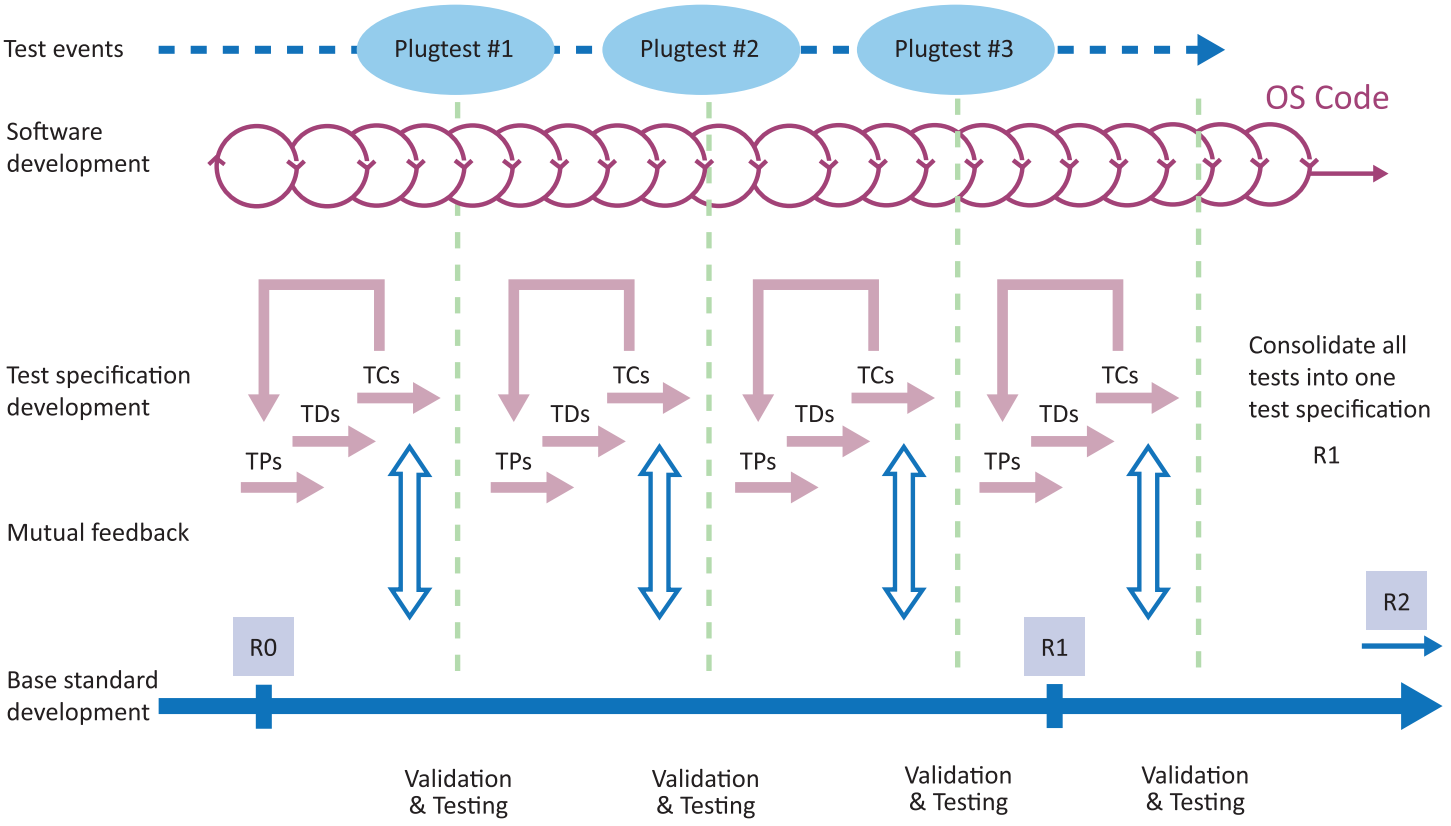
Two examples of how Open Source code may be developed in parallel with a standard

Aligning testing and validation with an agile development process

Interoperability events and regular conformance testing will debug the implementation and provide essential feedback to the standards-writers. This feedback is crucial not only for correcting any possible defects in the standard but, significantly, for proposing new or improved features or technical solutions.

Once again, the timing is critical. So is flexibility and speed. In principle, a tightly iterated 'waterfall' approach of *Test Purposes – Test Descriptions – Test Cases* will work if the cycle time is in months and if the project is following a well-defined release schedule. At some point, at a major Release for example, the total set of test cases can be collected and published as a single test suite, as illustrated below.

However, if a really fast turn-around time is needed this approach is unlikely to be adequate. A possible solution is to use a language such as TDL and concentrate everything into a Test Description that is abstract enough for non-testers to understand (i.e. the Test Purpose) but from which executable code such as TTCN-3 can automatically be generated. While this is not possible at the time of writing, ETSI's MTS Technical Committee foresees such a capability in the near future.



Alignment of testing and validation with an agile OSS-like development

ETSI resources

Methods for Testing and Specification Technical Committee

ETSI's aim has always been to produce documents that are easy to understand and easy to use. Our Technical Committee on Methods for Testing and Specification (TC MTS) develops standards related to testing and specification languages, and provides frameworks and methodologies to enable other ETSI committees to achieve their goal of producing effective standards.

TC MTS works very closely with our Centre for Testing and Interoperability (CTI) by developing the background material which the CTI then uses in its support of other ETSI committees, as well as other relevant standardisation bodies such as ITU Study Group 17. Much of the work undertaken by the committee has been adapted and used beyond ETSI by other organisations, fora, and industry globally.

The committee comprises a range of experts from major network operators, test service providers, telecommunication equipment vendors, test tool vendors and acclaimed research institutes. As members of the committee, they develop, profile and adapt state-of-the-art test and specification technologies to enable a competitive tool market for these technologies, and to draw up guidelines on how to use them in the standardisation context.

ETSI's Centre for Testing and Interoperability

ETSI's Centre for Testing and Interoperability (CTI) supports our technical committees in the use of best practices for the specification and validation of base standards, the development of test specifications, organising Plugtests events, overseeing PoC Frameworks, assisting with hackathons and participating in ETSI Open Source projects.

The CTI promotes best practices which encompass techniques which are pragmatic (including the validation of standards through interoperability events) as well as technically advanced (for example, using languages such as TTCN-3 to define test scenarios). These techniques have the added benefit of being applicable well beyond the world of standardisation. Good practice incorporated into standardisation can be carried through to proprietary development processes.

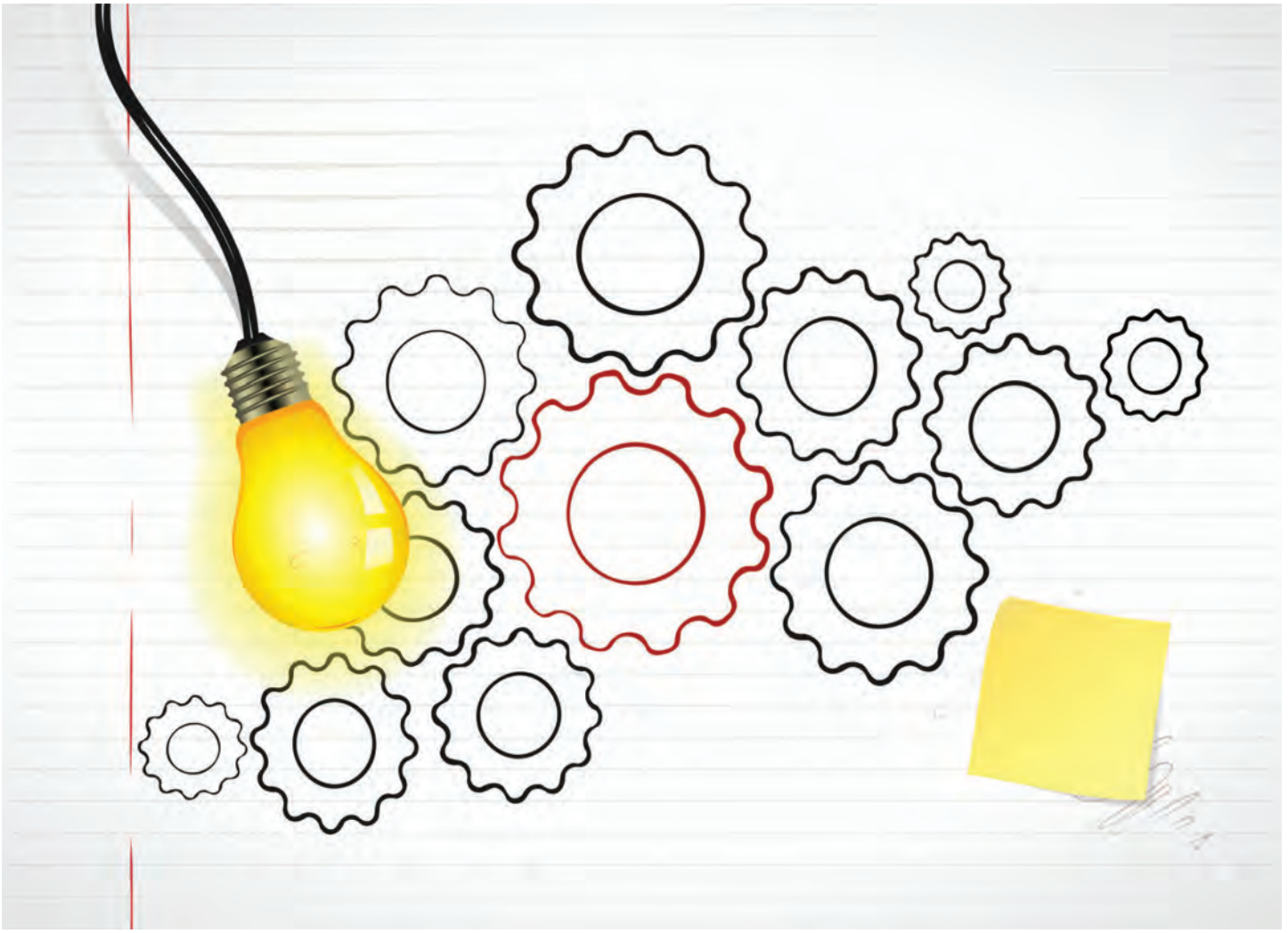
https://portal.etsi.org/Services/CentreforTestingInteroperability.aspx
www.etsi.org/about/what-we-do/plugtests

Responding to a changing world

The pursuit of technical excellence is something that lies at the heart of ETSI and is central to its members' aspirations. This introduction has described various processes and techniques that have been proven to complement the creation of high-quality standards, enabling industry to produce innovative, reliable, interoperable and cost-effective products and services.

Many of the practices discussed in this document were developed or adapted by ETSI. They now form a fundamental part of our standards-making philosophy and are highly regarded within the global standardisation community. In addition to these practices, we have gathered together a wealth of technical expertise both to support our members and continue our search for new solutions to meet the needs of an evolving industry. We are here to help, so please do not hesitate to call upon our support and expertise!

The 'Open Source phenomenon' is a major challenge to traditional standardisation – but, like all technical challenges, one that we in ETSI relish for the opportunities it affords. It impacts well-established methods of working in a number of different ways, not least in how to maintain the same assurance of high-quality, interoperable solutions in a multi-market industry. We are working with Open Source communities to encourage dialogue and the sharing of practices – with the goal of ensuring that the mutual benefits of well-respected procedures and dynamic innovation can be combined in tomorrow's technologies.



References and useful resources

ETSI resources

ETSI Drafting Rules (EDR)	https://portal.etsi.org/Services/editHelp!.aspx
ETSI Directives	https://portal.etsi.org/Resources/ETSIDirectives.aspx
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
Standards-making process	http://portal.etsi.org/chaircor/process.asp
Terms and definitions database	http://webapp.etsi.org/Teddi/
Editing checklist	https://portal.etsi.org/Portals/0/TBpages/edithelp/Docs/etsi_experts_guide.pdf (Annex K)
Testing and interoperability	www.etsi.org/about/how-we-work/testing-and-interoperability
ETSI Seminar	www.etsi.org/news-events/etsi-seminar

The *editHelp!* web pages (https://portal.etsi.org/Services/editHelp!.aspx) contain a wide range of guides, tools and skeleton documents to help you in the standards-making task. These include documents relating to methodologies and best practices.

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

Test specification languages

ES 202 553	Methods for Testing and Specification (MTS); TPLan: A notation for expressing Test Purposes
ES 203 119	Methods for Testing and Specification (MTS); The Test Description Language (TDL)
TTCN-3	www.ttcn-3.org/index.php/downloads/standards

650 Route des Lucioles
06921 Sophia Antipolis CEDEX – France
Tel.: +33 4 92 94 42 00
www.etsi.org