BS EN 61850-10:2013

BSI Standards Publication

Communication networks and systems for power utility automation

Part 10: Conformance testing

bsi.

...making excellence a habit.™

National foreword

This British Standard is the UK implementation of EN 61850-10:2013. It is identical to IEC 61850-10:2012. It supersedes BS EN 61850-10:2005 which is withdrawn.

The UK participation in its preparation was entrusted to Technical Committee PEL/57, Power systems management and associated information exchange.

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

Published by BSI Standards Limited 2013

ISBN 978 0 580 76419 6

ICS 33.200

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 July 2013.

Amendments issued since publication

Date	Text affected

EUROPEAN STANDARD

NORME EUROPÉENNE

EUROPÄISCHE NORM

EN 61850-10

July 2013

ICS 33.200

Supersedes EN 61850-10:2005

English version

Communication networks and systems for power utility automation - Part 10: Conformance testing

(IEC 61850-10:2012)

Réseaux et systèmes de communication pour l'automatisation des systèmes électriques -
Partie 10: Essais de conformité
(CEI 61850-10:2012)

Kommunikationsnetze und -systeme in Stationen -
Teil 10: Konformitätsprüfung
(IEC 61850-10:2012)

CENELEC

European Committee for Electrotechnical Standardization
Comité Européen de Normalisation Electrotechnique
Europäisches Komitee für Elektrotechnische Normung

Management Centre: Avenue Marnix 17, B - 1000 Brussels

Ref. No. EN 61850-10:2013 E

Foreword

The text of document 57/1284/FDIS, future edition 2 of IEC 61850-10, prepared by IEC TC 57 "Power systems management and associated information exchange" was submitted to the IEC-CENELEC parallel vote and approved by CENELEC as EN 61850-10:2013.

The following dates are fixed:

• latest date by which the document has to be implemented at national level by publication of an identical national standard or by endorsement	(dop)	2014-01-05
• latest date by which the national standards conflicting with the document have to be withdrawn	(dow)	2016-01-18

This document supersedes EN 61850-10:2005.

EN 61850-10:2013 includes the following significant technical changes with respect to EN 61850-10:2005:

– server device conformance test procedures have been updated;

– client device conformance test procedures have been added;

– sampled values device conformance test procedures have been added;

– (engineering) tool related conformance test procedures have been added;

– GOOSE performance test procedures have been added.

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

Endorsement notice

The text of the International Standard IEC 61850-10:2013 was approved by CENELEC as a European Standard without any modification.

Annex ZA
(normative)

Normative references to international publications with their corresponding European publications

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

NOTE When an international publication has been modified by common modifications, indicated by (mod), the relevant EN/HD applies.

Publication	Year	Title	EN/HD	Year
IEC/TS 61850-2	-	Communication networks and systems in substations - Part 2: Glossary	-	-
IEC 61850-3	-	Communication networks and systems for power utility automation - Part 3: General requirements	EN 61850-3	-
IEC 61850-4	2011	Communication networks and systems for power utility automation - Part 4: System and project management	EN 61850-4	2011
IEC 61850-5	2003	Communication networks and systems in substations - Part 5: Communication requirements for functions and device models	EN 61850-5[1)]	2003
IEC 61850-6	2009	Communication networks and systems for power utility automation - Part 6: Configuration description language for communication in electrical substations related to IEDs	EN 61850-6	2010
IEC 61850-7-1	2011	Communication networks and systems for power utility automation - Part 7-1: Basic communication structure - Principles and models	EN 61850-7-1	2011
IEC 61850-7-2	2010	Communication networks and systems for power utility automation - Part 7-2: Basic information and communication structure - Abstract communication service interface (ACSI)	EN 61850-7-2	2010
IEC 61850-7-3	2010	Communication networks and systems for power utility automation - Part 7-3: Basic communication structure - Common data classes	EN 61850-7-3	2011
IEC 61850-7-4	2010	Communication networks and systems for power utility automation - Part 7-4: Basic communication structure - Compatible logical node classes and data object classes	EN 61850-7-4	2010

[1)] EN 61850-5 is superseded by EN 61850-5:2013, which is based on IEC 61850-5:2013.

Publication	Year	Title	EN/HD	Year
IEC 61850-8-1	2011	Communication networks and systems for power utility automation - Part 8-1: Specific Communication Service Mapping (SCSM) - Mappings to MMS (ISO 9506-1 and ISO 9506-2) and to ISO/IEC 8802-3	EN 61850-8-1	2011
IEC 61850-9-2	2011	Communication networks and systems for power utility automation - Part 9-2: Specific Communication Service Mapping (SCSM) - Sampled values over ISO/IEC 8802-3	EN 61850-9-2	2011
IEC 62439-3	2012	Industrial communication networks - High availability automation networks - Part 3: Parallel Redundancy Protocol (PRP) and High availability Seamless Redundancy (HSR)	EN 62439-3	2012
ISO 9001	-	Quality management systems - Requirements	EN ISO 9001	-
ISO 9506	Series	Industrial automation systems - Manufacturing Message Specification	-	-
ISO/IEC 9646	Series	Information technology - Open Systems Interconnection - Conformance testing methodology and framework	EN ISO/IEC 9646	-
IEEE 1588	2008	IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems	-	-

CONTENTS

INTRODUCTION

This part of IEC 61850 is part of a set of specifications which details a layered power utility communication architecture.

This part of IEC 61850 defines:

- the methods and abstract test cases for conformance testing of client, server and sampled values devices used in power utility automation systems, and
- the methods and abstract test cases for conformance testing of engineering tools used in power utility automation systems, and
- the metrics to be measured within devices according to the requirements defined in IEC 61850-5.

The intended readers are IEC 61850 developers, test engineers and test system developers.

NOTE 1 Tests regarding EMC requirements and environmental conditions are subject to IEC 61850-3 and not included in this part of IEC 61850.

It is recommended that IEC 61850-5 and IEC 61850-7-1 be read first in conjunction with IEC 61850-7-2, IEC 61850-7-3, and IEC 61850-7-4.

NOTE 2 Abbreviations used in IEC 61850-10 are listed in Clause 4 or may be found in other parts of IEC 61850 that are relevant for conformance testing.

COMMUNICATION NETWORKS AND SYSTEMS FOR POWER UTILITY AUTOMATION –

Part 10: Conformance testing

1 Scope

This part of IEC 61850 specifies standard techniques for testing of conformance of client, server and sampled value devices and engineering tools, as well as specific measurement techniques to be applied when declaring performance parameters. The use of these techniques will enhance the ability of the system integrator to integrate IEDs easily, operate IEDs correctly, and support the applications as intended.

NOTE The role of the test facilities for conformance testing and certifying the results is beyond the scope of this part of IEC 61850.

2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IEC 61850-2, *Communication networks and systems for power utility automation – Part 2: Glossary*

IEC 61850-3, *Communication networks and systems for power utility automation – Part 3: General requirements*

IEC 61850-4:2011, *Communication networks and systems for power utility automation – Part 4: System and project management*

IEC 61850-5:2003, *Communication networks and systems for power utility automation – Part 5: Communication requirements for functions and devices models*

IEC 61850-6:2009, *Communication networks and systems for power utility automation – Part 6: Configuration description language for communication in electrical substations related to IEDs*

IEC 61850-7-1:2011, *Communication networks and systems for power utility automation – Part 7-1: Basic communication structure – Principles and models*

IEC 61850-7-2:2010, *Communication networks and systems for power utility automation – Part 7-2: Basic information and communication structure – Abstract communication service interface (ACSI)*

IEC 61850-7-3:2010, *Communication networks and systems for power utility automation – Part 7-3: Basic communication structure – Common data classes*

IEC 61850-7-4:2011, *Communication networks and systems for power utility automation – Part 7-4: Basic communication structure – Compatible logical node classes and data object classes*

IEC 61850-8-1:2011, *Communication networks and systems for power utility automation – Part 8-1: Specific communication service mapping (SCSM) – Mappings to MMS (ISO/IEC 9506-1 and ISO/IEC 9506-2) and to ISO/IEC 8802-3*

IEC 61850-9-2:2011, *Communication networks and systems for power utility automation – Part 9-2: Specific Communication Service Mapping (SCSM) – Sampled values over ISO/IEC 8802-3*

IEC 62439-3:2012, *Industrial communication networks – High availability automation networks – Part 3: Parallel Redundancy Protocol (PRP) and High Availability Seamless Redundancy (HSR)*

ISO/IEC 9646 (all parts), *Information technology – Open Systems Interconnection – Conformance testing methodology and framework*

ISO 9001 (all parts), *Quality management systems*

ISO 9506 (all parts), *Industrial automation systems – Manufacturing Message Specification*

IEEE 1588:2008, *Standard for a precision clock synchronization protocol for networked measurement and control systems*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in IEC 61850-2 and the following apply.

3.1
factory acceptance test
FAT
customer-agreed functional tests of the specifically manufactured power utility automation system or its parts using the parameter set for the planned application as specified in a specific customer specification

Note 1 to entry: The FAT will be carried out in the factory of the manufacturer or other agreed-upon location by the use of process simulating test equipment.

3.2
hold point
point, defined in the appropriate document beyond which an activity shall not proceed without the approval of the initiator of the conformance test

Note 1 to entry: The test facility shall provide a written notice to the initiator at an agreed time prior to the hold point. The initiator or his representative is obligated to verify the hold point and approve the proceeding of the activity.

3.3
interoperability
ability of two or more IEDs from the same vendor (or different vendors) to exchange information and use that information for correct co-operation.

Set of values having defined correspondence with the quantities or values of another set

3.4
model implementation conformance statement
MICS
statement that details the standard data object model elements supported by the system or device

3.5
negative test
test to verify the correct response of a system or a device when subjected to:

– IEC 61850 series conformant information and services which are not implemented in the system or device under test;
– non IEC 61850 series conformant information and services sent to the system or device under test

3.6
protocol implementation conformance statement
PICS
statement with the summary of the communication capabilities of the system or device to be tested

3.7
protocol implementation extra Information for testing
PIXIT
statement with system or device specific information regarding the communication capabilities of the system or device to be tested and which are outside the scope of the IEC 61850 series. The PIXIT is not subject to standardisation.

3.8
routine test
performed by the manufacturer in order to ensure device operation and safety

3.9
site acceptance test
SAT
verification of each data and control point and the correct functionality within the PUAS and between the PUAS and its operating environment at the whole installed plant by use of the final parameter set as specified in a specific customer specification

Note 1 to entry: The SAT is the precondition for the power utility automation system (PUAS) being put into operation.

3.10
SCL implementation conformance statement
SICS
statement with the summary of the capabilities of the SCL engineering tool

3.11
system related test
verification of correct behaviour of the IEDs and of the overall PUAS under specific application conditions

Note 1 to entry: The system related test is part of the final stage of the development of IEDs as belonging to a PUAS-product family.

3.12
test equipment
all tools and instruments which simulate and verify the input/outputs of the operating environment of the PUAS such as switchgear, transformers, network control centres or connected telecommunication units on the one side, and the serial links between the IEDs of the PUAS on the other

3.13
test facility
organisation able to provide appropriate test equipment and trained staff for conformance testing

Note 1 to entry: The management of conformance tests and the resulting information should follow a quality system.

3.14
technical issues conformance statement
TICS
statement with device specific information regarding the implemented technical issues detected after publication of the standard. The TICS is not subject to standardisation.

3.15
type test
verification of correct behaviour of the IEDs of the PUAS by use of the system tested software under the test conditions corresponding with the technical data

Note 1 to entry: The type test marks the final stage of the hardware development and is the precondition for the start of the production. This test is carried out with IEDs, which have been manufactured through the normal production cycle.

3.16
witness point
point, defined in the appropriate document, at which an inspection will take place on an activity

Note 1 to entry: The activity may proceed without the approval of the initiator of the conformance test. The test facility provides a written notice to the initiator at an agreed time prior to the witness point. The initiator or his representative has the right, but is not obligated, to verify the witness point.

4 Abbreviated terms

ACSI	abstract communication service interface
BRCB	buffered report control block
CDC	common data class
DUT	device under test
FAT	factory acceptance test
GI	general interrogation
GoCB	GOOSE control block
GOOSE	generic object oriented substation events
HMI	human machine interface
HSR	high availability seamless ring
ICD	IED capability description
IED	intelligent electronic device
IID	instantiated IED description
IP	internet protocol
LCB	log control block
LD	logical device
LN	logical node
MICS	model implementation conformance statement
MMS	manufacturing message specification (ISO 9506 series)
MSVCB	multicast sampled value control block
PICS	protocol implementation conformance statement
PIXIT	protocol implementation extra information for testing
PPS	pulse per second

PRP	parallel redundancy protocol
PUAS	power utility automation system
SAT	site acceptance test
SAV	sampled analogue values (IEC 61850-9-2)
SCD	substation configuration description
SCL	substation configuration language
SCSM	specific communication service mapping
SGCB	setting group control block
SICS	SCL implementation conformance statement
SNTP	simple network time protocol
SSD	system specification description
SV	sampled values
SVCB	sampled values control block
TCP	transport control protocol
TICS	technical issues conformance statement
TPAA	two party application association
TUT	tool under test
URCB	unbuffered report control block
USVCB	unicast sampled values control block
UTC	coordinated universal time
XML	extensible markup language

5 Introduction to conformance testing

5.1 General

There are many steps involved from the development and production of a device to the proper running of a complete system designed according the specific needs of a customer. Suitable test steps are incorporated in this process.

The quality system of the producer/supplier forms the basis of reliable testing in development and production activities.

Many internal tests during the development of a device (or a system kit) result in a type test (unit level test) performed at least by the provider and – if required by applicable standards – by an independent test authority. In the context of this standard, the term type test is restricted to the functional behaviour of the device.

Continuing routine tests in the production chain are necessary to ensure a constant quality of delivered devices in accordance with the quality procedures of the producer.

A conformance test is the type test for communication and – since communication establishes a system – the system related test of the incorporated IEDs. As a global communications standard, the IEC 61850 series includes standardised conformance tests to ensure that all suppliers comply with applicable requirements.

Type tests and conformance tests do not completely guarantee that all functional and performance requirements are met. However, when properly performed, such tests

significantly reduce the risk of costly problems occurring during system integration in the factory and on-site.

Conformance testing does not replace project specific system related tests such as the FAT and SAT. The FAT and SAT are based on specific customer requirements for a dedicated power utility automation system and are done by the system integrator and normally witnessed by the customer. These tests increase the confidence level that all potential problems in the system have been identified and solved. These tests establish that the delivered power utility automation system is performing as specified.

5.2 Conformance test procedures

In general, conformance testing of the communication behaviour of an IED should address the functional requirements and performance requirements of typical applications supported by these devices in a PUAS. IEC 61850-4 defines a general classification of quality tests, which are used within this part.

Conformance testing demonstrates the capability of the DUT to operate with other IEDs in a specified way according to the IEC 61850 series.

Conformance testing requires consideration of the following issues:

– The problem of all testing is the completeness of the tests. The number of all possible situations can be very large. It may be possible to cover all normal operating cases, but this may not be true for all failure cases.

– It is impossible to test all system configurations using IEDs from different world-wide suppliers. Therefore, a standardized test architecture with device simulators should be used. The use of such a test architecture implies agreement about its configuration and the test procedures applied in order to achieve compatible and reproducible results.

– A communication standard does not standardise the functions of the communicating equipment. Therefore, the failure modes of the functions are outside the scope of this part of the IEC 61850 series. But both, the existence of distributed functions and the impact of function response in devices on the data flow create some interdependence.

– Depending on the definition range of the standard, some properties of the device may be proven by information and documents provided with the DUT for the conformance testing instead of the conformance test itself.

The conformance test establishes that the communication of the DUT works according the IEC 61850 series. The IEC 61850 series is focussed on interoperability using data, function and device models including all services above or at the application level (ACSI). In addition, performance classes are addressed.

Since the IEC 61850 series defines no new communication stacks, the conformance to all seven ISO/OSI layers may be proven by documentation that communication stack software compliant with the corresponding specifications is implemented and may have been pre-tested and optionally certified. In the standard conformance test, only the application according to ACSI can be tested.

5.3 Quality assurance and testing

5.3.1 General

In order to ensure the quality during conformance testing, a quality assurance system has to be in place. This shall be clearly demonstrated by the test facility. This applies to the quality systems of all sub-suppliers.

In general, quality surveillance is used to monitor and verify the status of components during all phases of the conformance tests. For this purpose, inspections are carried out, based on hold and witness points that are indicated by the initiator or its representative in the test and

the inspection plan that is supplied by the test facility. These inspections are process-related and will provide information and confidence on the quality of the tests. Quality surveillance will reduce the risks of failure during the FAT and SAT.

5.3.2 Quality plan

5.3.2.1 Conformance test quality plan

The test facility will supply, for evaluation, a quality plan for the conformance test.

The conformance test quality plan shall meet the requirements of ISO 9001. The plan shall describe all measures for the scope of work and/or deliveries in the areas of budget, organisation, time, information and quality. There is only one plan for the test facility and its sub-suppliers.

The conformance test quality plan shall also contain the following:

– A complete and detailed description of the work methods. This will help ensure that all verifiable activities will fulfil all applicable requirements and conditions as stated in the scope of work during the time allowed.
– A detailed description of all tasks to be performed, including references to the schedule, an overview of the involved staff, materials and work methods as well as relevant methods and procedures.
– A detailed description of the organisation, including the assignments, tasks and responsibilities of mentioned staff during the different stages of the test programs. The description shall include all tests, inspections, research and audits during the various stages of the tests and the dates on which they will take place. These descriptions will be part of the test and inspection plan.
– A method for handling deviations, changes and modifications during all stages of the test.
– A sign off procedure and a description of the documentation to be supplied.

5.3.2.2 Test and inspection plan

The conformance test quality plan shall contain a test and inspection plan. In this plan, the test facility specifies, for all phases of the tests:

– what will be inspected, tested and registered;
– the purpose of the inspections and tests;
– the procedures and standards to which inspections, tests and registrations will be performed;
– the expected results of the inspections and tests;
– by whom the inspections, tests and registrations will be performed.

The test facility is responsible for the correct and timely performance of all activities mentioned in the test and inspection plan.

The test facility shall provide a proposal for so-called hold, witness and review points in the test and inspection plan.

There are several methods to perform a hold or witness point. The initiator of the conformance test or a representative can be present during the execution of a test or inspection. It is also possible to review the associated quality documents, e.g. checklists, verification and validation documents. This review can take place at the test facility’s site during the execution of a test or inspection can be made at the initiator’s site in which case the test facility shall provide all relevant documentation to the initiator.

All hold and witness points will be announced by the test facility at least a predefined time before they take place. A period of at least one week is recommended, depending on the time needed for making travel arrangements and the availability of the needed resources.

5.3.2.3 Audits requested by initiator

The initiator of a conformance test has the right to conduct audits on the quality system of the test facility and its sub-suppliers. The test facility shall co-operate and provide access to all locations applicable for the conformance test. The initiator's right to check the quality of the conformance test does not dismiss the test facility from its responsibilities.

Inspections and tests by the initiator of a conformance test shall be possible at mutually agreeable times at the locations, offices and factories of the test facility and all applicable third parties and sub-suppliers.

5.4 Testing

5.4.1 General

Conformance testing shall be customised for each device under test based on the capabilities identified in the PICS, PIXIT, TICS and MICS provided by the vendor. When submitting devices for testing, the following shall be provided:

– device ready for testing;
– protocol implementation conformance statement (PICS). A standard PICS, also known as PICS proforma shall be supplied (see IEC 61850-7-2, Annex A);
– protocol implementation extra information for testing (PIXIT) statement;
– model implementation conformance statement (MICS);
– technical issues conformance statement (TICS);
– instruction manuals detailing the installation and operation of the device.

The requirements for conformance testing fall into two categories:

a) static conformance requirements (define the requirements the implementation shall fulfil);
b) dynamic conformance requirements (define the requirements that arise from the protocol used for a certain implementation).

The static and dynamic conformance requirements shall be defined in a protocol implementation conformance statement or PICS. The PICS serves three purposes:

1) selection of the appropriate set of tests;
2) ensure that the tests appropriate to a claim of conformance are performed;
3) provide the basis for the review of the static conformance.

A standard PICS shall be supplied.

Concrete PICS shall be as defined for the SCSMs.

A model implementation conformance statement or MICS shall be provided detailing the data object model elements supported by the system or device. The MICS is implemented in the ICD or IID file according to IEC 61850-6.

A technical issues conformance statement or TICS shall be provided detailing the implemented technical issues detected after publication of the standard.

In addition to the PICS, a protocol implementation extra information for testing or PIXIT document shall be provided.

The process of assessing the conformance is shown in Figure 1.

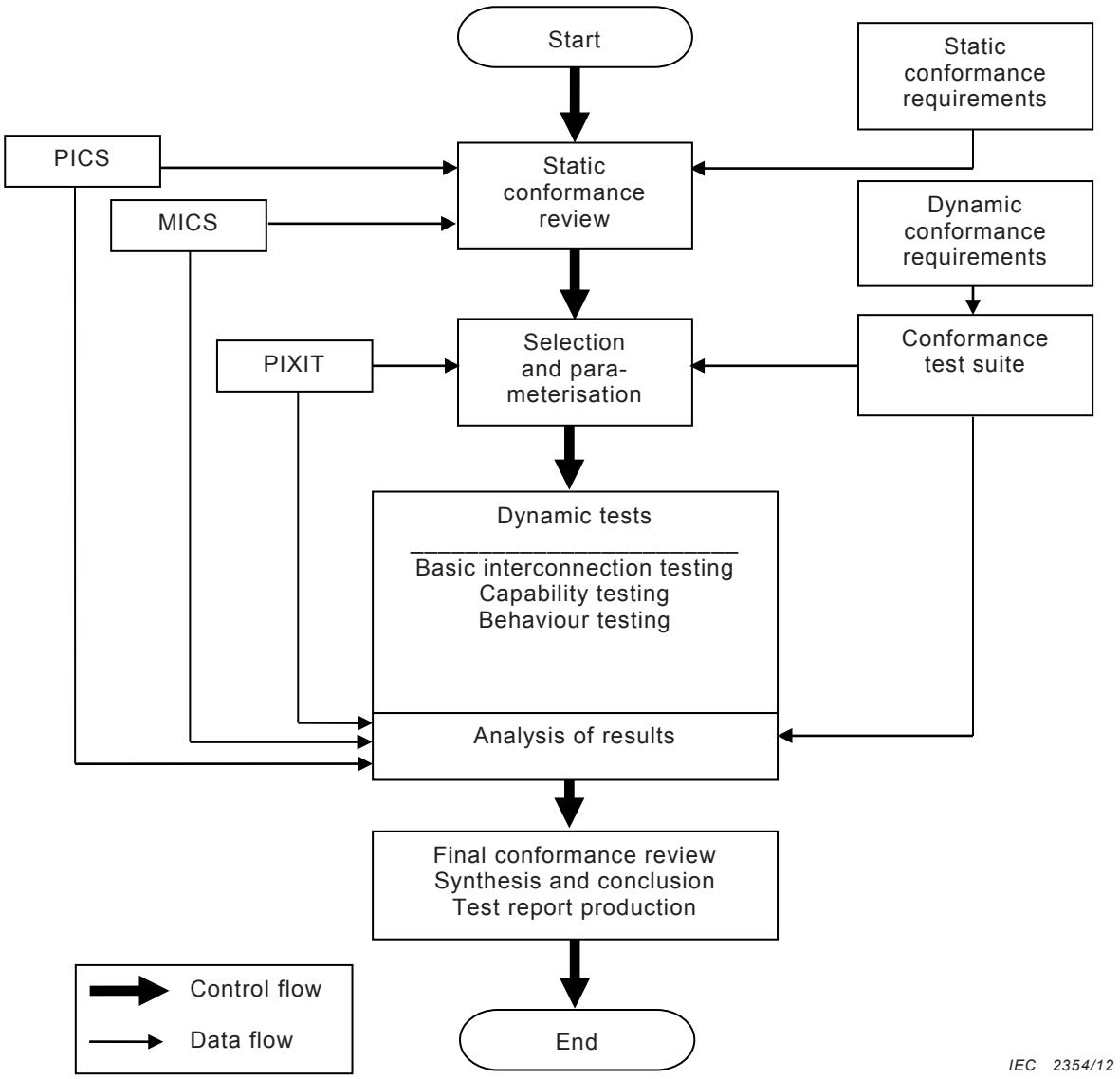

Figure 1 – Conceptual conformance assessment process

5.4.2 Use of SCL files

The DUT shall be delivered with an ICD file.

The test entity shall generate from the ICD file the corresponding SCD file based on the configuration of the test system. If the test entity requires that the initiator of a conformance test also provides the SCD file, then the test entity shall provide the SSD file and the SCD/SED file of the test system.

5.4.3 Device testing

A single device shall be conformance tested against a test device.

The device-specific conformance tests contain the positive and negative testing of the following, as appropriate:

– inspection of the documentation and version control of the device (IEC 61850-4);
– test of device configuration file against standardised syntax (schema) (IEC 61850-6);
– test of device configuration file against the device related object model (IEC 61850-7-4, IEC 61850-7-3);
– test of communication stack implementation against applicable SCSM (IEC 61850-8-1, IEC 61850-9-2);
– test of implemented ACSI services against ACSI definition (IEC 61850-7-2);
– test of device specific extensions according to rules given by the IEC 61850 series in general.

5.5 Documentation of conformance test report

A conformance test report shall include the following information:

– A reference list of all documents that describe or specify any qualifying tests that have been performed. These documents may include the vendor's standard operating and testing procedures, and local, national and international standards. International standards shall be cited by document number, date, clause and subclauses. References to other documents shall include a complete source address and document identification. A complete and contextually accurate summary or extract of the document may be included for convenience.
– A list of any specialised test equipment or computer programs used for performing the conformance tests.
– Name and address of the vendor.
– Name and address of the initiator of the conformance test (if different from vendor name).
– Name of the tested device.
– All of the variants (hardware, firmware, etc.) of the tested device.
– Name and address of the test facility.
– Date of issue of the test report.
– Name and signature of the tester.
– Unique reference number.
– A list of test items performed to verify conformance.
– Comments and problems found.
– For each test item, the following subjects shall be documented:
    - description of the test item with the objective of the test, the procedure how to perform the test and the expected result;
    - reference to the IEC 61850 series part, clause and subclause;
    - unique identifier per test item;
    - test result: passed, failed, inconclusive, not applicable or <empty> = not tested;
    - comparison of the test result to the expected result.

Changes or alterations to the device made at any point in the test, particularly those made to correct a test deficiency, shall be completely described. The consequences and requirements of re-testing of a server device – if required – shall be specified in corresponding test plans and test reports.

Conformance test documentation shall be supplied to the initiator.

6 Device related conformance testing

6.1 Test methodology

Communication testing needs at least two devices to communicate with each other. Comprehensive interoperability testing of all possible products is not feasible. Therefore, the test concept shall include test devices, test configurations, and test scenarios. The dynamic behaviour should be tested properly by using well-defined test cases.

Messages are generated to test the communication capabilities. Hardwired stimuli (contacts, voltages, currents, etc.) and stimuli coming over a serial link if applicable should be used if applicable.

Special attention shall be given to communication equipment such as star-couplers, switches, etc. which shall support all requested features of the standard but not introduce additional contingencies and limitations. The impact of the communication method (client-server, GOOSE, SV, etc.) used by the DUT shall be considered properly in the test procedures. Verification of functional applications (use of GOOSE messages) is not part of a conformance test even if advanced tools may offer such analysis.

6.2 Conformance test procedures

6.2.1 General

This subclause describes the test procedure requirements, test structure, the abstract test cases (what is to be tested). The format and a few examples of detailed test procedures (how to perform the test) are given in Annex A.

6.2.2 Test procedure requirements

The test procedure requirements are:

– The abstract test cases describe what shall be tested, the detailed test procedures describe how a test engineer or a test system shall perform the test.

– Test cases include a reference to the applicable paragraph(s) in the referenced document(s).

– The test results shall be reproducible in the same test lab and in other test labs.

– Support automated testing with minimal human intervention, as far as reasonably possible.

– The tests shall focus on situations that cannot easily be tested during, for example, a factory or site acceptance test, and prevent inter-operability risks, for example:

  - check behaviour of the device on delayed, lost, double and out of order packets,
  - configuration, implementation, operation risks,
  - mismatching names, parameters, settings, or data types,
  - exceeding certain limits, ranges or timeouts,
  - force situations to test negative responses,
  - check all (control) state machine paths, and
  - force simultaneous control operations from multiple clients.

– The ACSI tests focus on the application layer (mapping).

– The device under test (DUT) is considered as a black box. The I/O and the communication interface are used for testing.

– The test includes testing the versions, data model and configuration file, and the use of applicable ISO/IEC 9646 series terminology.

The test procedures shall be formatted as outlined in Figure 2. With this format, the test procedures document can also be used as test report. A few test procedure examples are depicted in Annex A.

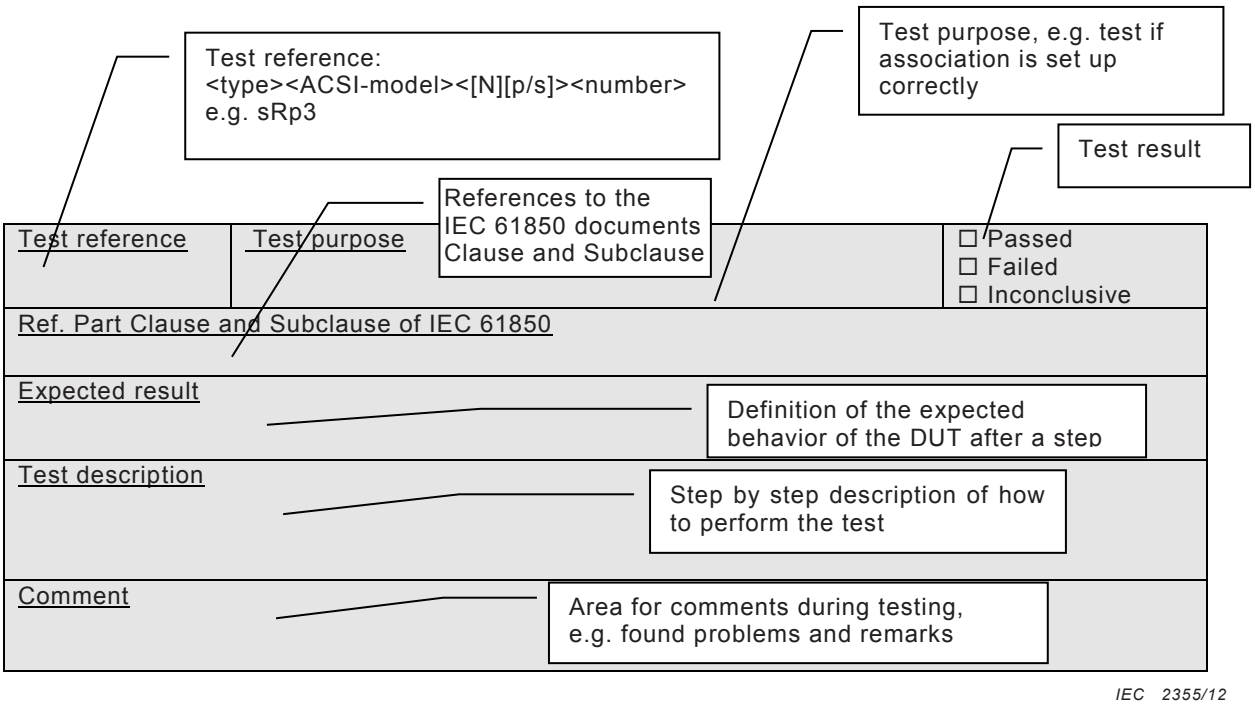

Figure 2 – Test procedure format

6.2.3 Test structure

The test cases are structured as follows:

– documentation and version control (IEC 61850-4);
– configuration file (IEC 61850-6);
– data model (IEC 61850-7-3 and IEC 61850-7-4);
– mapping of ACSI models and services (IEC 61850-7-2 and applicable SCSM).

6.2.4 Test cases to test a server device

6.2.4.1 General

This part of the IEC 61850 series specifies the test system architecture and abstract test cases for server devices. The abstract test cases shall be used for the definition of test procedures to run in tests.

NOTE The SCSM specific test procedures are provided by test facilities agreed upon by the market participants.

6.2.4.2 Test system architecture to test a server device

In order to be able to perform a server device test, a minimum test set-up is necessary. The test architecture contains (see Figure 3):

– DUT;
– client simulator to initiate and generate TPAA messages;
– GOOSE simulator to send correct and incorrect GOOSE messages;
– test master to start/stop test cases, start/stop the analyzer and archive test results;
– time master;

– engineering tool to configure the DUT;
– protocol analyzer to store all the network traffic for each test case;
– signal generator to force binary and analogue events, controlled by the test master or test engineer.

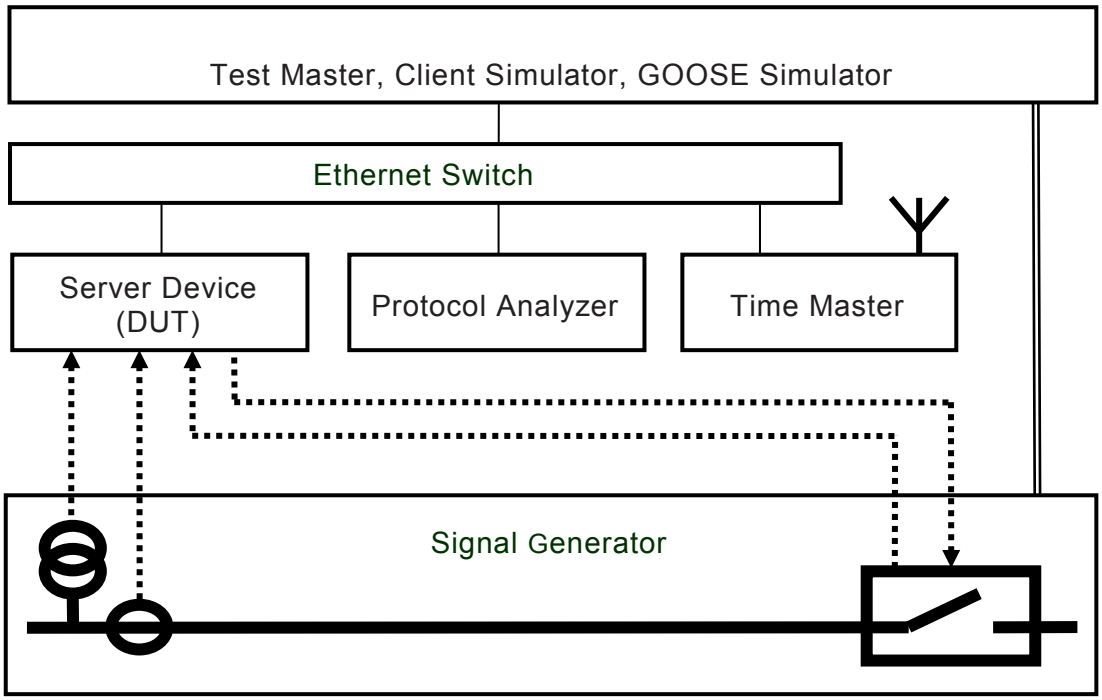

Figure 3 – Test system architecture to test a server device

The test system shall include documentation regarding test system hardware and test system software.

6.2.4.3 Documentation and version control test procedure overview

The test cases listed in Table 1 shall apply.

Table 1 – Server documentation test cases

Test case	Test case description
sDoc1	Check if the major/minor software version in the PICS documentation and the DUT do match (IEC 61850-4). PICS shall contain the ACSI conformance statement according to IEC 61850-7-2 Annex A
sDoc2	Check if the major/minor software version in the PIXIT documentation and software version of the DUT does match (IEC 61850-4). PIXIT shall indicate the required information as requested in the test cases
sDoc3	Check if the major/minor software version in the MICS documentation and software version of the DUT does match (IEC 61850-4). MICS shall indicate the semantics of all non-standard Logical Nodes, Data Objects, Data Attributes and enumeration
sDoc4	Check if the major/minor software version in the TICS documentation and software version of the DUT does match (IEC 61850-4). TICS shall indicate the implemented technical issues.

6.2.4.4 Configuration file test cases

The test cases listed in Table 2 shall apply.

Table 2 – Server configuration test cases

Test case	Test case description
sCnf1	Test if the ICD configuration file conforms to the SCL schema (IEC 61850-6)
sCnf2	Check if the ICD configuration file corresponds with the actual data names, data types, data-sets, pre-defined data values exposed by the DUT on the network. When more data or services are exposed, attach a list and set the test result to Passed. When less data or services are exposed the test result is Failed.
sCnf3	Change at least 5 end-user configurable parameters that are exposed by the DUT on the network in the SCD configuration file, configure the DUT using the SCD configuration file (using the supplied configuration tool) and check the updated configuration using online services corresponds with the updated SCD file. Restore the original SCD file and re-configure the DUT to its original state.
sCnf4	Check if the server capabilities in the ICD “services” section do match with the IED capabilites
sCnf5	In case the control model is fixed (not configurable) check if the ICD correctly initializes the ctlModel values for all controllable objects
sCnf6	Check the edition 2 SCL changes: – version="2007" with revision="A" or higher – nameLength = 64
sCnf7	Check the "ldName" naming structure when supported. All online object references (including data sets and control block references) shall start with the "LDevice ldName" value instead of the "IED name" + "LDevice inst"

6.2.4.5 Data model test cases

The test cases listed in Table 3 shall apply.

Table 3 – Server data model test cases

Test case	Test case description
sMdl1	Verify presence of mandatory objects for each LN. Passed when all objects/attributes are present
sMdl2	Verify presence of conditional presence true objects for each LN. Passed when all objects/attributes are present
sMdl3	Verify non-presence of conditional presence false objects. Passed when these objects/attributes are not present
sMdl4	Verify data model mapping according to applicable SCSM concerning name length and object expansion. Passed when mapping is according to applicable SCSM
sMdl5	Verify data model mapping according to applicable SCSM concerning organisation of functional components. Passed when mapping is according to applicable SCSM
sMdl6	Verify data model mapping according to applicable SCSM concerning naming of control blocks and logs. Passed when mapping is according to applicable SCSM
sMdl7	Verify data type of all objects for each LN. Passed when data type of all objects/attributes do match with the IEC 61850-7-3, IEC 61850-7-4 and the applicable SCSM
sMdl8	Verify that the preconfigured enumerated data attribute values from the device and SCL are in specified range. Passed when all values are in range
sMdl9	Check if manufacturer specific data model extensions are implemented according to the extension rules in IEC 61850-7-1 Clause 14 (only when extensions are implemented). Passed when all extensions are implemented according to the rules
sMdl10	Check if the order of the data attributes within the functional constraints of the Data Object types match with IEC 61850-7-3. Passed when all attributes are in matching order
sMdl11	Check the maximum name length of Logical Device, Logical Node, data sets and control blocks according to IEC 61850-7-2 Subclause 22.2 and SCSM

Test case	Test case description
sMdl12	Check that the rules for multiple data object instantiation are kept (IEC 61850-7-1,14.6, IEC 61850-7-4)
sMdl13	Check the logical device name space or the LLN0 logical node name space refers to edition 2
sMdl14	Check the correct use of name spaces for non-substation power utility applications like for example Hydro and DER

6.2.4.6 Mapping of ACSI models and services test cases

Test items shall be grouped together in tables. The tables shall reflect the applicable service models specified in Figure 3 of IEC 61850-7-2:

– application association (sAss);
– server, Logical device, Logical node, Data, and Data Attribute model (sSrv);
– data set model (sDs);
– service tracking (sTrk);
– substitution model (sSub);
– setting group model (sSg);
– unbuffered report control model (sRp);
– buffered report control model (sBr);
– log control model (sLog);
– generic object oriented substation events (sGop and sGos);
– control model (sCtl);
– time and time synchronisation model (sTm);
– file transfer model (sFt).

Test cases are defined for each ACSI model and services in the following categories:

- positive = verification of normal conditions, typically resulting in response+
- negative = verification of abnormal conditions, typically resulting in response–

A test case is mandatory when the applicable ACSI model and ACSI service is supported by the DUT. This is specified in the PICS according to IEC 61850-7-2, Annex A. The test result interpretation (passed/failed) depends on the declared IED capabilities e.g. in the ICD file as well as on the test result.

6.2.4.7 Application association model

6.2.4.7.1 Positive test cases

The test cases listed in Table 4 shall apply.

Table 4 – Association positive test cases

Test case	Test case description
sAss1	Associate and client-release a TPAA association (IEC 61850-7-2 Subclause 8.3.2)
sAss2	Associate and client-abort TPAA association (IEC 61850-7-2 Subclause 8.3.2)
sAss3	Associate with maximum number of clients simultaneously (PIXIT)

6.2.4.7.2 Negative test cases

The test cases listed in Table 5 shall apply.

Table 5 – Association negative test cases

Test case	Test case description
sAssN1	Check that with incorrect authentication parameters and authentication turned on at server the association fails, and with authentication turned off the server associates (IEC 61850-7-2 Subclause 8.3)
sAssN2	Check that with incorrect association parameters at server or client the association fails (IEC 61850-7-2 Subclause 8.3, PIXIT)
sAssN3	Set up maximum+1 associations, verify the last associate is refused
sAssN4	Disconnect the communication interface, the DUT should detect association lost within a specified period
sAssN5	Interrupt and restore the power supply, the DUT should accept an association request when ready
sAssN6	Verify the re-use of dropped association resources

6.2.4.8 Server, logical device, logical node, and data model

6.2.4.8.1 Positive test cases

The test cases listed in Table 6 shall apply.

Table 6 – Server positive test cases

Test case	Test case description
sSrv1	Request GetServerDirectory(LOGICAL-DEVICE) and check response (IEC 61850-7-2 Subclause 7.2.2)
sSrv2	For each GetServerDirectory(LOGICAL-DEVICE) response issue a GetLogicalDeviceDirectory request and check response (IEC 61850-7-2 Subclause 9.2.1)
sSrv3	For each GetLogicalDeviceDirectory response issue a GetLogicalNodeDirectory(DATA) request and check response (IEC 61850-7-2 Subclause 10.2.2)
sSrv4	For each GetLogicalNodeDirectory(DATA) response issue a – GetDataDirectory request and check response (IEC 61850-7-2 Subclause 11.4.4) – GetDataDefinition request and check response (IEC 61850-7-2 Subclause 11.4.5) – GetDataValues request and check response (IEC 61850-7-2 Subclause 11.4.2)
sSrv5	Issue one GetDataValues request with the maximum number of data values and check response
sSrv6	For each write enabled DATA object issue a SetDataValues request and check response (IEC 61850-7-2 Subclause 11.4.2)
sSrv7	Issue one SetDataValues request with the maximum number of data values and check response
sSrv8	Request GetAllDataValues for each functional constraint and check response (IEC 61850-7-2 Subclause 10.2.3)
sSrv9	Evaluate the semantic of selected (volt/amp) analogue measurements: – Verify analogue value (plausibility check, not accuracy) – Verify quality bits, force situations to set specific quality bits – Verify (UTC) timestamp value and quality (plausibility check, not accuracy) – Verify scaling, range and units, change a setting and verify resulting value – Verify dead band, change dead band and verify result – Verify limit indications
sSrv10	Evaluate the semantic of selected status points: – Verify status value – Verify quality bits, force situations to set specific quality bits – Verify (UTC) timestamp value and quality (plausibility check, not accuracy)

Test case	Test case description
sSrv11	Verify that when blkEna is set to true by an operator the quality bit oldData and operatorBlocked is set by the server and the process data is not updated anymore (IEC 61850-7-3 Subclause 6.2.6)
sSrv12	Verify Mod/Beh values: off, test, blocked – When Mod/Beh is off process data is not updated, Mod and Beh are updated, quality is set to invalid – When Mod/Beh is test or test-blocked the process data quality test is set – When Mod/Beh is on-blocked the process data quality operatorBlocked is set (IEC 61850-7-4 Annex A)
sSrv13	Verify logical device hierarchy; – the LLN0.GrRef should reference a valid logical device – the reference shall not result in a hierarchy loop – Beh value at higher level influences the lower levels correctly (i.e. like LD Beh influences LN behaviour dependent on LN Mod)

6.2.4.8.2 Negative test cases

The test cases listed in Table 7 shall apply.

Table 7 – Server negative test cases

Test case	Test case description
sSrvN1	Request following data services with wrong parameters (unknown object, name case mismatch, wrong logical device or wrong logical node) and verify response– service error – GetServerDirectory(LOGICAL-DEVICE) (IEC 61850-7-2 Subclause 7.2.2) – GetLogicalDeviceDirectory (IEC 61850-7-2 Subclause 9.2.1) – GetLogicalNodeDirectory(DATA) (IEC 61850-7-2 Subclause 10.2.2) – GetAllDataValues (IEC 61850-7-2 Subclause 10.2.3) – GetDataValues (IEC 61850-7-2 Subclause 11.4.2) – SetDataValues (IEC 61850-7-2 Subclause 11.4.3) – GetDataDirectory (IEC 61850-7-2 Subclause 11.4.4) – GetDataDefinition (IEC 61850-7-2 Subclause 11.4.5)
sSrvN2	Request SetDataValues of ENUMERATED data with out-of-range value and verify response– service error (IEC 61850-7-2 Subclause 11.4.3)
sSrvN3	Request SetDataValues with mismatching data type (e.g. int-float) and verify response– service error (IEC 61850-7-2 Subclause 11.4.3)
sSrvN4	Request SetDataValues for read-only data values and verify response– service error (IEC 61850-7-2 Subclause 11.4.3)

6.2.4.9 Data set model

6.2.4.9.1 Positive test cases

The test cases listed in Table 8 shall apply.

Table 8 – Data set positive test cases

Test case	Test case description
sDs1	Request GetLogicalNodeDirectory(DATA-SET) and check response (IEC 61850-7-2 Subclause 10.2.2) For each response issue a – GetDataSetValues request and check response (IEC 61850-7-2 Subclause 13.3.2) – GetDataSetDirectory request and check response (IEC 61850-7-2 Subclause 13.3.6)
sDs2	Request a persistent CreateDataSet with one member and with maximum possible members and check response (IEC 61850-7-2 Subclause 13.3.4) and verify that the persistent data set is visible for another client
sDs3	Request a non-persistent CreateDataSet with one, maximum members and check response (IEC 61850-7-2 Subclause 13.3.4) and verify that the persistent data set is not visible for another client
sDs4	Create and delete a persistent dataset, create the dataset again with the same name with one extra data value / re-ordered member and check the members
sDs5	Create and delete a non-persistent dataset, create the dataset again with the same name with one extra data value / re-ordered member and check the members
sDs6	Create a non-persistent dataset, release/abort the association, associate again and check the dataset has been deleted (IEC 61850-7-2 Subclause 13.1)
sDs7	Create a persistent dataset, release/abort the association, associate again and check the dataset is still present (IEC 61850-7-2 Subclause 13.1)
sDs8	Create and delete a persistent data set several times and verify every data set can be created normally
sDs9	Create and delete a non-persistent data set several times and verify every data set can be created normally
sDs10	Verify SetDataSetValues / GetDataSetValues with GetDataValues and SetDataValues
sDs11	Verify that the maximum number of persistent data sets with the maximum number of members can be created as specified in SCL
sDs12	Verify that the maximum number of non-persistent data sets with the maximum number of members can be created as specified in SCL
sDs13	Verify that a persistent data set can be created with the maximum name length for data set and a data set member (IEC 61850-7-2 Subclause 22.2)
sDs14	Verify that a non-persistent data set can be created with the maximum name length for data set and a data set member (IEC 61850-7-2 Subclause 22.2)

6.2.4.9.2 Negative test cases

The test cases listed in Table 9 shall apply.

Table 9 – Date set negative test cases

Test case	Test case description
sDsN1	Request following data set services with wrong parameters (unknown object, name case mismatch, wrong logical device or wrong logical node) and verify response– service error: – GetDataSetValues (IEC 61850-7-2 Subclause 13.3.2) – SetDataSetValues (IEC 61850-7-2 Subclause 13.3.3) – CreateDataSet (IEC 61850-7-2 Subclause 13.3.4) – DeleteDataSet (IEC 61850-7-2 Subclause 13.3.5) – GetDataSetDirectory (IEC 61850-7-2 Subclause 13.3.6)
sDsN2	Create a persistent dataset with the same name twice, and verify response– service error
sDsN3	Create a non-persistent dataset with the same name twice, and verify response– service error
sDsN4	Continue to create persistent data sets until a correct response– service error is returned
sDsN5	Continue to create non-persistent data sets until a correct response– service error is returned
sDsN6	Create a persistent dataset with unknown member verify response– service error
sDsN7	Create a non-persistent dataset with unknown member verify response– service error
sDsN8	Delete a (pre-defined) non-deletable dataset, and verify response– service error
sDsN9	Delete a persistent dataset twice, and verify response– service error
sDsN10	Delete a non-persistent dataset twice, and verify response– service error
sDsN11	Delete a persistent dataset referenced by a (report) control class, and verify response– service error (IEC 61850-7-2 Subclause 13.1)
sDsN12	Delete a non-persistent dataset referenced by a (report) control class, and verify response– service error (IEC 61850-7-2 Subclause 13.1)
sDsN13	Request SetDataSetValues with a dataset with one or more read-only members, and verify response– service error

6.2.4.10 Service tracking model

6.2.4.10.1 General

The tracking services can be verified by verifying the tracking information while executing the corresponding test cases defined in the other clauses. For example during the execution of the control model test cases the tracked AddCause value shall be verified as well.

6.2.4.10.2 Positive test cases

The test cases listed in Table 10 shall apply. The control block tracking test cases shall be executed with the maximum control block and data set name length.

Table 10 – Service tracking test cases

Test case	Test case description
sTrk1	Verify the tracking of control block services: Buffered reporting, LTRK.BrcbTrk
sTrk2	Verify the tracking of control block services: Unbuffered reporting, LTRK.UrcbTrk
sTrk3	Verify the tracking of control block services: Log control block, LTRK.LocbTrk
sTrk4	Verify the tracking of control block services: GOOSE control block, LTRK.GocbTrk
sTrk5	Verify the tracking of control block services: Multicast sampled values control block, LTRK.MsvcbTrk
sTrk6	Verify the tracking of control block services: Unicast sampled values control block, LTRK.UsvcbTrk
sTrk7	Verify the tracking of control block services: Setting group control block, LTRK.SgcbTrk
sTrk8	Verify the tracking of control services: Single point control, LTRK.SpcTrk
sTrk9	Verify the tracking of control services: Double point control, LTRK.DpcTrk
sTrk10	Verify the tracking of control services: Integer control, LTRK.IncTrk
sTrk11	Verify the tracking of control services: Enumerated control, LTRK.EncTrk
sTrk12	Verify the tracking of control services: Analogue process value control with float command, LTRK.ApcFTrk
sTrk13	Verify the tracking of control services: Analogue process value control with integer command, LTRK.ApcIntTrk
sTrk14	Verify the tracking of control services: Binary step control, LTRK.BscTrk
sTrk15	Verify the tracking of control services: Integer step control, LTRK.IscTrk
sTrk16	Verify the tracking of control services: Binary analogue process value control, LTRK.BacTrk
sTrk17	Verify the tracking of other supported common services, LTRK.GenTrk

6.2.4.11 Substitution model

6.2.4.11.1 Positive test cases

The test cases listed in Table 11 shall apply.

Table 11 – Substitution positive test cases

Test case	Test case description
sSub1	Disable subEna and set subVal, subMag, subCMag, subQ, subID and verify the substituted values are not transmitted when subEna is disabled and are transmitted when subEna enabled (IEC 61850-7-3 Table 64).
sSub2	Verify that in case the association fails, the substituted values shall remain unchanged
sSub3	Verify setting subVal, subMag, subCMag, subQ and subID is allowed and the substituted values are transmitted and Quality.Source is set to Substituted when subEna is enabled

6.2.4.12 Setting group control model

6.2.4.12.1 Positive test cases

The test cases listed in Table 12 shall apply.

Table 12 – Setting group positive test cases

Test case	Test case description
sSg1	Request GetLogicalNodeDirectory(SGCB) and check response+. For each SGCB request GetSGCBValues and check response+
sSg2	Verify the following setting group state machine path (IEC 61850-7-2 Clause 16, Figure 22); – SelectEditSG – Use SetEditSGValue [FC=SE] to change values – Use GetEditSGValue [FC=SE] to verify the new values – ConfirmEditSGValues
sSg3	Verify SelectActiveSG (IEC 61850-7-2 Clause 16, Figure 22); – SelectActiveSG of the first setting group – GetSGCBValues to verify active setting group and last activation time – Use GetDataValues/GetEditSGValue [FC=SG] to verify the values are of fist setting group – Repeat for all setting groups
sSg4	Verify that after loss of association the server cancels the editing (EditSG=0) and the client can use SelectEditSG again to copy the values to the edit buffer (IEC 61850 7-2 Subclause 16.3.3)
sSg5	Verify that when SGCB ResvTms is present – The first client can edit the setting group when ResvTms = 0 – A second client cannot edit the setting group when ResvTms > 0 – A server resets the ResvTms when it does not receive a ConfirmEditSG within the reservation time
sSg6	Verify that when SGCB ResvTms is not present – The first client can edit the setting group – A second client cannot edit the setting group within a certain time (PIXIT)
sSg7	Verify that editing and activating the active setting group is allowed
sSg8	Verify that a client can cancel the editing of a setting group and that the original setting group values remain unchanged
sSg9	Request SelectEditSG of the first setting group, change one value and SelectEditSG of the second setting group without (ConfirmEditSGValues). Verify the response+
sSg10	Verify that when a setting group is being edited the SG values of that group can be read

6.2.4.12.2 Negative test cases

The test cases listed in Table 13 shall apply.

Table 13 – Setting group negative test cases

Test case	Test case description
sSgN1	Request following setting group selection services with wrong parameters (out of range values, or non- existent/null setting group) and verify response– service error – SelectActiveSG (IEC 61850-7-2 Subclause 16.3.2) – GetDataValues/GetEditSGValue [FC=SG] (IEC 61850-7-2 Subclause 16.3.6) – GetSGCBValues (IEC 61850-7-2 Subclause 16.3.7)
sSgN2	Request following setting group definition services with wrong parameters (out of range values, or non- existent/null setting group) and verify response– service error – SelectEditSG (IEC 61850-7-2 Subclause 16.3.3) – SetEditSGValue (IEC 61850-7-2 Subclause 16.3.4) – ConfirmEditSGValues (IEC 61850-7-2 Subclause 16.3.5) – GetEditSGValue [FC=SE] (IEC 61850-7-2 Subclause 16.3.6)
sSgN3	Request SetEditSGValue on an setting group value with FC=SG, verify response– service error
sSgN4	Request SetEditSGValue (FC=SE) without SelectEditSG (EditSG = 0), verify response- service error
sSgN5	Verify that when a client is editing settings, another client cannot edit settings

6.2.4.13 Unbuffered reporting model

6.2.4.13.1 Positive test cases

The test cases listed in Table 14 shall apply.

Table 14 – Unbuffered reporting positive test cases

Test case	Test case description
sRp1	Request GetLogicalNodeDirectory(URCB) and check response Request GetURCBValues of all responded URCB's
sRp2	Verify the reporting of optional fields of a URCB Configure/enable a URCB with all optional fields combinations: sequence-number, report-time-stamp, reason-for-inclusion, data-set-name, and/or data-reference (IEC 61850-7-2 Subclause 17.2.3.2.2.1), force/trigger a report and check the reports contain the enabled optional fields
sRp3	Verify the trigger conditions of a URCB – Configure and enable a URCB with optional fields: sequence-number, report-time-stamp, reason-for-inclusion, data-set-name and data-reference and check the reports are transmitted according to the following (supported) trigger conditions: • on integrity • on update (dupd) • on update with integrity • on data change (dchg) • on data and quality change • on data and quality change with integrity period – Verify the validity of the ReasonCode (IEC 61850-7-2 Subclause 17.2.3.2.2.9) – Verify that when more trigger conditions are met preferably only one report is generated (IEC 61850-7-2 Subclause 17.2.3.2.3.2) – Verify that reports are only sent when RptEna is set to True (IEC 61850-7-2 Subclause 17.2.2.5), when reporting is disabled no reports should be transmitted
sRp4	General interrogation (IEC 61850-7-2 Subclause 17.2.2.13) Setting the GI attribute of an URCB shall start the general-interrogation process. One report with the current data values will be sent. After initiation of the general-interrogation, the GI attribute is reset to False.
sRp5	Segmentation of reports Verify that if a long report does not fit in one message, the report is split into sub-reports. Enable sequence-number and report-time-stamp optional field and check validity of: (IEC 61850-7-2 Subclause 17.2.3.2.2.5) – SqNum (not changed) – SubSqNum (0 for first report, incrementing, roll-over) – MoreSeqmentsFollow – TimeOfEntry (not changed as SqNum is not altered) (IEC 61850-7-2 Subclause 17.2.3.2.2.9) Verify that an update of a data value during sending of a segmented report caused by an integrity or general-interrogation trigger can be interrupted by a report with change of one of the data values with a new sequence number. (IEC 61850-7-2 Subclause 17.2.3.2.3.5) A new request for general-interrogation shall stop the sending of remaining segments of the GI-report that is still going on. A new GI-report shall start with a new sequence number and the sub-sequence number shall be 0 (IEC 61850-7-2 Subclause 14.2.3.2.3.4)
sRp6	Configuration revision (IEC 61850-7-2 Subclause 17.2.2.7) – Verify that ConfRev represents a count of the number of times the configuration of the data set referenced by DatSet has been changed. Changes that are counted are: • deletion of a member of the data-set • re-ordering of members in the data-set – Verify that after a restart of the server, the value of ConfRev is restored to its original value of the base local configuration OR the value is retained from the configuration prior to restart (PIXIT) – Verify that the server increments the ConfRev in case the data sets changes due to processing of ACSI services – ConfRev should never be 0 (zero) in case DatSet is not null.

Test case	Test case description
sRp7	Verify that after a restart of the server, the value of ConfRev is restored to its original value of the base local configuration OR the value is retained from the configuration prior to restart (PIXIT)
sRp8	Buffer Time (IEC 61850-7-2 Subclause 17.2.2.9) – Verify that in the case where a second internal notification of the same member of a DATA-SET has occurred prior to the expiration of BufTm, the server: (IEC 61850-7-2 Subclause 17.2.2.9) • shall for status information behave as if BufTm has expired and immediately send the report, restart the timer with value BufTm and process the second notification or • may for analogue information behave as if BufTm has expired and immediately transmit the report for transmission, restart the timer with value BufTm and process the second notification or • may for analogue information substitute the current value in the pending report with the new one. – Configure Buffer Time to 1 000 ms and force a data value change of multiple dataset members within buffer time. Server should send not more than one report per buffer time with all the data values changes since last report. – Verify that the value 0 for buffer time indicates that the buffer time attribute is not used. (IEC 61850-7-2 Subclause 17.2.2.9) – Verify that the BufTm value can contain at least the value 360 0000 (= 1 h in milliseconds)
sRp9	Verify the DUT can send reports with data objects
sRp10	Verify the DUT can send reports with data attributes
sRp11	Verify the DUT send any buffered events before the integrity report
sRp12	Verify the DUT send any buffered events before the GI report
sRp13	Verify that the server sets URCB Owner to a non-NULL value when the URCB is configured by a client and reset to NULL when a client releases the URCB. For a pre-assigned URCB the server resets the Owner to the pre-assigned client address
sRp14	Verify that the DUT can process an URCB with maximum name length for RptID and DatSet (IEC 61850-7-2 Subclause 22.2)

6.2.4.13.2 Negative test cases

The test cases listed in Table 15 shall apply.

Table 15 – Unbuffered reporting negative test cases

Test case	Test case description
sRpN1	Request GetURCBValues with wrong parameters and verify response– service error (IEC 61850-7-2 Subclause 17.2.5.3)
sRpN2	Configure reporting with trigger option GI (not dchg, qchg, dupd, integrity). When enabled only GI reports are transmitted. No reports should be sent when generating events (IEC 61850-7-2 Subclause 17.2.5.4)
sRpN3	Setting the integrity period to 0 with TrgOps = integrity will result in no integrity reports will be sent (IEC 61850-7-2 Subclause 17.2.2.12)
sRpN4	Incorrect configuration of a URCB: configure when enabled, configure ConfRev and SqNum and configure with unknown data set
sRpN5	Exclusive use of URCB and lost association Configure a URCB and set the Resv attribute and enable it. Verify another client cannot set any attribute of that URCB (IEC 61850-7-2 Subclause 17.2.4.5)
sRpN6	Configure unsupported URCB options (PIXIT); Configure unsupported trigger conditions, optional fields and related parameters
sRpN7	Verify another client cannot configure a pre-assigned URCB
sRpN8	Verify that when TrgOps – GI is not set the request to set GI to true shall fail

6.2.4.14 Buffered reporting model

6.2.4.14.1 Positive test cases

The test cases listed in Table 16 shall apply.

Table 16 – Buffered reporting positive test cases

Test case	Test case description
sBr1	Request GetLogicalNodeDirectory(BRCB) and check response Request GetBRCBValues of all responded BRCB's
sBr2	Verify the reporting of optional fields of a BRCB Configure/enable a BRCB with all optional fields combinations: sequence-number, report-time-stamp, reason-for-inclusion, data-set-name, data-reference, buffer-overflow, and/or entryID (IEC 61850-7-2 Subclause 17.2.3.2.2.1), force/trigger a report and check the reports contain the enabled optional fields
sBr3	Verify the trigger conditions of a BRCB – Configure and enable a BRCB with optional fields: sequence-number, report-time-stamp, reason-for-inclusion, data-set-name, data-reference, buffer-overflow, and entryID and check the reports are transmitted according to the following (supported) trigger conditions: • on integrity • on update (dupd) • on update with integrity • on data change (dchg) • on data and quality change • on data and quality change with integrity period – Verify the validity of the ReasonCode (IEC 61850-7-2 Subclause 17.2.3.2.2.9) – Verify that when more trigger conditions are met preferably only one report is generated (IEC 61850-7-2 Subclause 17.2.3.2.3.2) – Verify that reports are only sent when RptEna is set to True. (IEC 61850-7-2 Subclause 17.2.2.5), when reporting is disabled no reports should be transmitted
sBr4	General interrogation (IEC 61850-7-2 Subclause 17.2.2.13) Setting the GI attribute of a BRCB shall start the general-interrogation process. One report with the current data values will be sent. After initiation of the general-interrogation, the GI attribute is reset to False.
sBr5	Segmentation of reports Verify that if a long report does not fit in one message, the report is split into sub-reports. Enable sequence-number and report-time-stamp optional field and check validity of: (IEC 61850-7-2 Subclause 17.2.3.2.2.5) – SqNum (not changed) – SubSqNum (0 for first report, incrementing, roll-over) – MoreSeqmentsFollow – TimeOfEntry (not changed as SqNum is not altered) (IEC 61850-7-2 Subclause 17.2.3.2.2.9) Verify that an update of a data value during sending of a segmented report caused by an integrity or general-interrogation trigger can be interrupted by a report with change of one of the data values with a new sequence number. (IEC 61850-7-2 Subclause 17.2.3.2.3.5) A new request for general-interrogation shall stop the sending of remaining segments of the GI-report that is still going on. A new GI-report shall start with a new sequence number and the sub-sequence number shall be 0 (IEC 61850-7-2 Subclause 17.2.3.2.3.4) Verify that when OptFlds=sequence-number is NOT set, neither SubSqNum nor SqNum are present in the sub-reports (IEC 61850-7-2 Subclause 17.2.3.2.2.4 and 17.2.3.2.2.5)

Test case	Test case description
sBr6	Configuration revision (IEC 61850-7-2 Subclause 17.2.2.7) – Verify that ConfRev represents a count of the number of times the configuration of the data set referenced by DatSet has been changed. Changes that are counted are: • deletion of a member of the data-set • re-ordering of members in the data-set – Verify that after a restart of the server, the value of ConfRev is restored to its original value of the base local configuration OR the value is retained from the configuration prior to restart (PICS) – Verify that the server increments the ConfRev in case the data sets changes due to processing of ACSI services – ConfRev should never be 0 (zero) in case DatSet is not null
sBr7	Verify that after a restart of the server, the value of ConfRev is restored to its original value of the base local configuration OR the value is retained from the configuration prior to restart (PIXIT)
sBr8	Buffer Time (IEC 61850-7-2 Subclause 17.2.2.9) – Verify that in the case where a second internal notification of the same member of a DATA-SET has occurred prior to the expiration of BufTm, the server: (IEC 61850-7-2 Subclause 17.2.2.9) • shall for status information behave as if BufTm has expired and immediately send the report, restart the timer with value BufTm and process the second notification or • may for analogue information behave as if BufTm has expired and immediately transmit the report for transmission, restart the timer with value BufTm and process the second notification or • may for analogue information substitute the current value in the pending report with the new one. – Configure Buffer Time to 1 000 ms and force a data value change of multiple dataset members within buffer time. Server should send not more than one report per buffer time with all the data values changes since last report. – Verify that the value 0 for buffer time indicates that the buffer time attribute is not used (IEC 61850-7-2 Subclause 17.2.2.9) – Verify that the BufTm value can contain at least the value 360 0000 (= 1 h in ms)
sBr9	Verify the DUT can send reports with data objects
sBr10	Verify the DUT can send reports with data attributes
sBr11	Verify that all buffered events shall be sent before integrity reports can be sent (IEC 61850-7-2 Subclause 17.2.3.2.3.3)
sBr12	Verify that all buffered events shall be sent before the GI report can be sent (IEC 61850-7-2 Subclause 17.2.3.2.3.3)
sBr13	Verify that the server sets BRCB Owner to a non-NULL value when the BRCB is configured by a client and reset to NULL when a client releases the URCB. For a pre-assigned BRCB the server resets the Owner to the pre-assigned client address
sBr14	Verify that the DUT can process a BRCB with maximum name length for RptID and DatSet (IEC 61850-7-2 Subclause 22.2)
	Specific to BRCB (leave a gap for future RP test cases)
sBr20	Buffered reporting (BRCB) state machine (IEC 61850-7-2 Subclause 17.2.2.5 figure 20) – Verify events are buffered after the association is released – Verify reporting is disabled after the association is lost – Verify that not received reports while not associated are received now in the correct order (SOE) (IEC 61850-7-2 Subclause 17.2.1, IEC 61850-7-2 Subclause 17.2.2.5) – Do the same but now set PurgeBuf to True before enabling the reporting. No stored buffered reports should be send (IEC 61850-7-2 Subclause 14.2.2.14) – Force buffer overflow, the OptFlds buffer-overflow should be set in the first report that is sent with events that occurred after the overflow. (IEC 61850-7-2 Subclause 17 2.3.2.2.8)
sBr21	Buffered reporting (BRCB); buffering events (IEC 61850-7-2 Subclause 17.2.3.2.3.6) – Verify that after the association is available again and after the client has set the EntryID, and enabled the BRCB, the BRCB shall start sending the reports of events that have been buffered. The BRCB shall use the sequence and subsequence numbers so that no gaps occur.
sBr22	Verify that integrity reports are buffered

Test case	Test case description
sBr23	Verify successful ResvTms behaviour – On ResvTms = –1 the BRCB can be used by the pre-assigned client – On ResvTms = 0 a client can reserve the BRCB by writing a value and configure the BRCB – On lost association the reserved BRCB is released after the ResvTms number of seconds (ResvTms set to zero) On lost association, within ResvTms time none of other clients can reserve the BRCB except the one who did it originally (the client restores association)
sBr24	Verify that a SetBRCBValues request, for setting ResvTms, shall: • Generate a negative response if the BRCB's ResvTms value = –1. • Generate a negative response if the BRCB's ResvTms value is non-zero and if the SetBRCBValues request is being issued by another client for whom the BRCB is not reserved. Generate a negative response if the ResvTms value to be set is negative.
sBr25	Verify that a change of one of the following BRCB parameters purges the buffer: RptID, BufTm, TrgOps, IntgPd, DatSet. A change of OptFlds should not purge the buffer. (IEC 61850-7-2 Subclause 17.2.2.5)
sBr26	Verify that after setting an invalid, null or non-existing EntryID the DUT sends all reports in the buffer
sBr27	Verify that when the BRCB state is RptEna=FALSE a GetBRCBValues shall return the EntryID value that represents the last (newest) entry that has been entered into the buffer. And when the BRCB RptEna=TRUE: The value of EntryID, returned in a GetBRCBValues response, shall be the EntryID of the last EntryID formatted and queued for transmission.
sBr28	Verify that only the last buffered GI report is transmitted after a resync

6.2.4.14.2 Negative test cases

The test cases listed in Table 17 shall apply.

Table 17 – Buffered reporting negative test cases

Test case	Test case description
sBrN1	Request GetBRCBValues with wrong parameters and verify response- service error (IEC 61850-7-2 Subclause 17.2.3.3.2)
sBrN2	Configure reporting with trigger option GI (not dchg, qchg, dupd, integrity). When enabled only GI reports are transmitted. No reports should be sent when generating events (IEC 61850-7-2 Subclause 17.2.3.2.3.4)
sBrN3	Setting the integrity period to 0 with TrgOps = integrity will result in no integrity reports will be sent (IEC 61850-7-2 Subclause 17.2.2.12)
sBrN4	Incorrect configuration of a BRCB: configure when enabled, configure ConfRev and SqNum and configure with unknown data set
sBrN5	Exclusive use of BRCB and lost association Configure a BRCB and enable it. Verify another client cannot set attributes value in this BRCB. (IEC 61850-7-2 Subclause 17.2.1)
sBrN6	Configure unsupported BRCB options (PIXIT); Configure unsupported trigger conditions, optional fields and related parameters
sBrN7	Verify another client cannot configure a pre-assigned BRCB
sBrN8	Verify that when TrgOps – GI is not set the device does not send reports with reason code GI

6.2.4.15 Log model

6.2.4.15.1 Positive test cases

The test cases listed in Table 18 shall apply.

Table 18 – Log positive test cases

Test case	Test case description
sLog1	Request GetLogicalNodeDirectory(LOG) and check response+
sLog2	Request GetLogicalNodeDirectory(LCB) and check response+
sLog3	Request GetLCBValues with functional constraint LG of all responded LCB's
sLog4	Request SetLCBValues with functional constraint LG when LCB is disabled
sLog5	Verify that logging is independent of a limited set of external application associations or other communication transactions
sLog6	Configure and enable logging and check that the following logging trigger conditions place a correct entry in the log with the correct members of the data set – on integrity – on update (dupd) – on update with integrity – on data change (dchg) – on quality change (qchg) – on data and quality change – on data and quality change with integrity period
sLog7	Request QueryLogByTime and check response+
sLog8	Request QueryLogAfter and check response+
sLog9	Request GetLogStatusValues and check response+, verify that the responded entries indicate the oldest/newest entry ID/time available in the log
sLog10	Check that data is logged as defined in the settings of logical node GLOG. The corresponding reason code shall be "application-trigger"
sLog11	Verify that server can process a LCB and LOG with maximum name length for LCBRef, LogRef and DatSet (IEC 61850-7-2 Subclause 22.2)
sLog12	Verify that log entries are non-volatile and not lost after reboot and power loss

6.2.4.15.2 Negative test cases

The test cases listed in Table 19 shall apply.

Table 19 – Log negative test cases

Test case	Test case description
sLogN1	Request the following log services with wrong parameters (out of range entries, or non-existent Dataset, LCB or Log) and verify response – service error – GetLCBValues (IEC 61850-7-2 Subclause 17.3.2.5) – SetLCBValues (IEC 61850-7-2 Subclause 17.3.2.6) – QueryLogByTime (IEC 61850-7-2 Subclause 17.3.5.2) – QueryLogAfter (IEC 61850-7-2 Subclause 17.3.5.3) – GetLogStatusValues (IEC 61850-7-2 Subclause 17.3.5.4)
sLogN2	Request SetLCBValues when LCB is enabled and disabled and verify response– service error

6.2.4.16 Generic substation events model

6.2.4.16.1 Positive test cases

The test cases listed in Table 20, Table 21 and Table 22 shall apply. To verify the device processes the subscribed GOOSE message it is recommend to configure an observation mechanism, for example copy a subscribed state value to a local data object which is published.

NOTE The performance of sending and receiving GOOSE messages is verified by the GOOSE performance test procedures (see 8.2.3).

Table 20 – GOOSE publish positive test cases

Test case	Test case description
sGop1	Request GetLogicalNodeDirectory(GoCB) and request GetGoCBValues (IEC 61850-7-2 Subclause 18.2.2.5 and 10.2.2)
sGop2	GOOSE messages are published with a long (SCL maxtime) cycle time, check the GOOSE data with configured data; (IEC 61850-7-2 Subclause 18.2.3) – gocbRef is a valid GoCB reference – timeAllowedtoLive > 0 and the next GOOSE message is transmitted within the specified value of the current GOOSE message – datSet is same as the GoCB and contains a valid dataset reference – goID is same as the GoCB and SCL, the default value is the GoCB reference – t contains the time of the status increment or start-up – sqNum is incremented, stNum>0 and is not changed – Simulation is not present or if present with value FALSE – confRev >0 and is same as the GoCB and SCL (IEC 61850-7-2 Subclause 18.2.1.6) – needsCommisioning is not present or if present same as GoCB – numDatSetEntries matches with the number of data entries in allData – allData values match with the datSet element type
sGop3	Verify that a newly activated device sends the initial GOOSE message with stNum initial value one (1) (IEC 61850-7-2 Subclause 18.1, 18.2.3)
sGop4	Force a data change of a data value in the GOOSE dataset, DUT should publish GOOSE messages as specified/configured (SCL mintime), stNum is incremented, sqNum = 0
sGop5	When supported, verify that the DUT publishes GOOSE messages with the simulation flag set (IEC 61850-7-2 Subclause 18.2.3.8)
sGop6	Disable GoCB, verify that changing parameters with SetGoCBValues are active (IEC 61850-7-2 Subclause 18.2.1.3, 18.2.2.5 and 18.2.2.6) and no GOOSE messages are transmitted anymore
sGop7	Verify that a restart of the device shall not reset the Configuration revision value (IEC 61850-7-2 Subclause 18.2.1.6)
sGop8	Verify that ConfRev increments every time when the configuration of the data set referenced by DatSet has been changed (IEC 61850-7-2 Subclause 15.2.1.6). Changes that are counted are: – deletion of a member of the data-set – re-ordering of members in the data-set – changing the value of the attribute DatSet
sGop9	Verify that GoCB attribute NdsCom is set when DatSet is not yet configured (is NULL) (IEC 61850-7-2 Subclause 18.2.1.7)
sGop10	Verify the DUT can send GOOSE messages with data attributes and/or data objects
sGop11	Verify that the server can process a GoCB with maximum name length for DatSet, GoCBRef and GoID (IEC 61850-7-2 Subclause 22.2)

Table 21 – GOOSE subscribe positive test cases

Test case	Test case description
sGos1	Send GOOSE messages with/without the VLAN tag with new data and check if the message is received and the data has the new value by e.g. check binary output, event list, logging or MMI
sGos2	Send GOOSE messages with the ndsCom parameter set. Verify that on a status change the values are not used for operational purposes (IEC 61850-7-2 Subclause 18.2.3.8)
sGos3	Proper detection and action roll-over of sqNum with no status change (sqNum=max -> sqNum = 1) and with status change (sqNum=max -> sqNum = 0)
sGos4	Verify the logical node LGOS data object attribute values on receiving valid GOOSE messages, no GOOSE messages and GOOSE messages with mismatching ConfRev
sGos5	Verify that the server can subscribe to GOOSE messages with structured data (FCD)
sGos6	Send subscribed GOOSE messages with the Simulation parameter set (IEC 61850-7-2 Subclause 18.2.3.8). Verify that – when the subscriber is not in simulation mode (LPHD.Sim.stVal=false) the simulated values are ignored. The subscriber shall keep on using the "real" GOOSE messages – when the subscriber is in simulation mode (LPHD.Sim.stVal=true) the simulated values are used for operational purposes. The subscriber shall ignore the "real" GOOSE messages after a first simulated one has been received. The corresponding LGOS.SimSt shall be set when the first simulated message is received and cleared when LPHD.Sim.stVal is set to false.
sGos7	Verify that the server can subscribe GOOSE messages with maximum name length for DatSet, GoCBRef and GoID (IEC 61850-7-2 Subclause 22.2)

Table 22 – GOOSE management positive test cases

sGom1	Verify GOOSE services: request service with legal parameters and check response (IEC 61850-7-2 Subclause 15.2.2) – GetGoReference (IEC 61850-7-2 Subclause 18.2.2.3) – GetGOOSEElementNumber (IEC 61850-7-2 Subclause 18.2.2.4)
sGom2	Verify GOOSE management request: Check DUT request service with valid parameters and simulate valid response (IEC 61850-7-2 Subclause 15.2.2) – GetGoReference (IEC 61850-7-2 Subclause 18.2.2.3) – GetGOOSEElementNumber (IEC 61850-7-2 Subclause 18.2.2.4)

6.2.4.16.2 Negative test cases

The test cases listed in Table 23, Table 24 and Table 25 shall apply.

Table 23 – GOOSE publish negative test cases

Test case	Test case description
sGopN1	When GoEna=TRUE, no attributes of the GoCB control block can be set except for GoEna. (IEC 61850-7-2 Subclause 18.2.1.3)
sGopN2	Verify that if the number or size of values being conveyed by the elements in the dataset exceeds the SCSM determined maximum number, NdsCom is set to True. (IEC 61850-7-2 Subclause 18.2.1.7)

Table 24 – GOOSE subscribe negative test cases

Test case	Test case description
sGosN1	Check behaviour of DUT as specified in PIXIT on Missing GOOSE message
sGosN2	Check behaviour of DUT as specified in PIXIT on Double GOOSE message
sGosN3	Check behaviour of DUT as specified in PIXIT on Delayed GOOSE message, with and without exceeding timeAllowedToLive
sGosN4	Check behaviour of DUT as specified in PIXIT on Out of order GOOSE message
sGosN5	Check behaviour of DUT as specified in PIXIT on No GOOSE messages
sGosN6	Check behaviour of DUT as specified in PIXIT on invalid GOOSE messages – gocbRef different from GoCB and NULL – timeAllowedtoLive = 0 – datSet different from GoCB and NULL – goID different from GoCB and NULL – t contains the time of a status change minus/plus one hour – confRev different from GoCB and NULL – numDatSetEntries 0, more, less with the number of data entries in the allData – allData values do not match with the datSet element type

Table 25 – GOOSE management negative test cases

Test case	Test case description
sGomN1	Client request GOOSE management services with illegal parameters and verify DUT response-service error (IEC 61850-7-2 Subclause 18.2.2), Verify that NULL for MemberReference in GetGOOSEElementNumber indicates that no member of the referenced data set is defined. (IEC 61850-7-2 Subclause 18.2.2.4.2.2)

6.2.4.17 Control model

6.2.4.17.1 Test cases

The test cases listed in Table 26 shall apply.

Table 26 – Control test cases

Test case	Test case description
sCtl1	Force and check each path in control state machine for several control objects with control models – direct with normal security (IEC 61850-7-2 Subclause 20.2.1) – SBO-control with normal security (IEC 61850-7-2 Subclause 20.2.2) – direct with enhanced security (IEC 61850-7-2 Subclause 20.3.2) – SBO-control with enhanced security (IEC 61850-7-2 Subclause 20.3.3) Compare detailed test cases for each control model
sCtl2	Change control model using online services and verify that the control object responds according to the new control model
sCtl3	Time Operate a second enhanced security control object before the activation time of the first control object (PIXIT)
sCtl4	Verify that the stSeld attribute value is set/reset as specified in the state machines
sCtl5	Verify test flag in selectwithvalue/operate and Beh = test (IEC 61850-7-4 Annex A Table A.1) – When LN Beh is "on" the control Requests are rejected with AddCause "Blocked-by-mode" – When LN Beh is “test/blocked” the control requests are accepted – When LN Beh is "test" the control requests are accepted
sCtl6	Select all SBO control objects and cancel them in opposite order. In case a control action is blocked because another control is already running the AddCause shall be "1-of-n-control"

Test case	Test case description
sCtl7	Verify that with interlock or synchro check conditions the specified checks are performed and the command is executed accordingly (IEC 61850-7-2 Subclause 20.5.2.5) – When the interlock check fails with AddCause "Blocked-by-interlocking" – When the interlock check passes – When the synchro check fails with AddCause "Blocked-by-synchrocheck" – When the synchro check passes
sCtl8	Operate (without select) a SBO control object and verify that the request is rejected with AddCause "Object-not-selected" (IEC 61850-7-2 Table 47)
sCtl9	Select the same control object twice, verify that the second select request is rejected with AddCause "Object-already-selected" (IEC 61850-7-2 Table 47) and the object remains in selected state (Operate.req is accepted)
sCtl10	Operate control value is the same as the actual status value (On-On or Off-Off) and verify that the control request is rejected with AddCause "Position-reached" (IEC 61850-7-2 Table 47, PIXIT)
sCtl11	Select the same control object from 2 different clients. Verify that the control requests from the second client are rejected with AddCause "Locked-by-other-client" (IEC 61850-7-2 Table 47)
sCtl12	Select / Operate an unknown control object and verify that the control requests are rejected with AddCause "Unknown" (IEC 61850-7-2 Table 47)
sCtl13	Verify that the Select request on a direct operate control object is rejected with Addcause "Unknown" (IEC 61850-7-2 Table 47)
sCtl14	Operate the same direct control object twice from 2 clients (IEC 61850-7-2 Table 47, PIXIT) and verify that the last control request is rejected with AddCause "Command-already-in-execution"
sCtl15	Verify that the SBOes Operate or Cancel request with different control parameters than the SelectWithValue is rejected with AddCause: Inconsistent-parameters"
sCtl16	Verify that when Loc is set remote control requests are rejected with AddCause "Blocked-by-switching-hierarchy"
sCtl17	Verify that with station level control authority (LocSta=T) remote control requests are rejected with AddCause "Blocked-by-switching-hierarchy".
sCtl18	Verify that on CmdBlk.stVal is set the control requests are rejected with AddCause "Blocked-by-command" (IEC 61850-7-2 Table 54)
sCtl19	Verify that when the blkEna is set the control requests are **terminated** with AddCause "Time-limit-over"
sCtl20	Verify that when parameters are changed after the select respond, the operate request is rejected with AddCause "Parameter-change-in-execution" (IEC 61850-7-2 Table 54)
sCtl21	Verify that when tap changer has reached the limit (EndPosR or EndPosL in YLTC) control requests are rejected with AddCause "Step-limit" (IEC 61850-7-2 Table 54)
sCtl22	Verify that with insufficient access authority control requests are rejected with AddCause "No-access-authority". (IEC 61850-7-2 Table 54)
sCtl23	Verify that when an APC control action end position has overshoot the command terminates with AddCause "Ended-with-overshoot". (IEC 61850-7-2 Table 54)
sCtl24	Verify that when an APC control action is aborted due to deviation between the command value and the measured value the control terminates with AddCause "Abortion-due-to-deviation". (IEC 61850-7-2 Table 54)
sCtl25	Verify that a cancel request is successful when the control object is in the unselected state (IEC 61850-7-2 Table 47)
sCtl26	Verify that when the control object is in the WaitForExecution state the cancel or SelectWithValue request is rejected with AddCause "Command-already-in-execution" (IEC 61850-7-2 Table 54)
sCtl27	Verify that the SelectWithValue request on a SBOns control object is rejected with AddCause "Unknown" (IEC 61850-7-2 Table 54)
sCtl28	Verify that the DUT can control an object with maximum name length for IED and Logical Device (IEC 61850-7-2 Subclause 22.2)

6.2.4.17.2 Control model specific test cases

The test cases listed in Table 27, Table 28, Table 29 and Table 30 shall apply.

Table 27 – SBOes test cases

Test case	Test case description
sSBOes1	Send a correct SelectWithValue and Operate request Verify each of these paths will return the device to the Unselected state and verify the CommandTermination: – force the equipment simulator to move to the requested new state – force the equipment simulator to keep the old state (AddCause: Time-limit-over or Invalid-position) – force the equipment simulator to move to the 'between' state (AddCause: Invalid-position)
sSBOes2	Send a correct SelectWithValue request Verify each of these paths will return the device to the Unselected state: – Send a correct Cancel request – Wait for select timeout – Send a Release request – Send an Operate request resulting in 'Test not ok'
sSBOes3	Send a correct SelectWithValue and TimeActivatedOperate request, resulting in respond-
sSBOes4	Send a correct SelectWithValue request Send a correct TimeActivatedOperate Once request Verify the TimeActivatedOperateTermination+ Verify each of these paths will return the device to the Unselected state and verify the CommandTermination: – force the equipment simulator to move to the requested new state – force the equipment simulator to keep the old state (AddCause: Time-limit-over or Invalid-position) – force the equipment simulator to move to the 'between' state (AddCause: Invalid-position)
sSBOes5	Send a correct SelectWithValue request Send a correct TimeActivatedOperate request Verify each of these paths will return the device to the Ready state and the TimeActivatedOperateTermination-: – Force a 'Test not ok' – Send a correct Cancel request
sSBOes6	Select device using SelectWithValue with improper access rights. Access should be denied (IEC 61850-7-2 Subclause 20.2.2) or send incorrect SelectWithValue request
sSBOes7	Send a correct SelectWithValue request Verify that sending multiple Operate Many requests will return the device to the Ready state Verify that sending a Cancel request will return the device to the Unselected state
sSBOes8	Verify that the Operate or Cancel request with different control parameters than the SelectWithValue is rejected with AddCause: Inconsistent-parameters

Table 28 – DOns test cases

Test case	Test case description
sDOns1	Send a correct Operate request
sDOns2	Send an Operate request, resulting in 'Test not ok'
sDOns3	Send an TimeActivatedOperate, request resulting in respond-
sDOns4	Send a correct TimeActivatedOperate request Verify the TimeActivatedOperateTermination+
sDOns5	Send a correct TimeActivatedOperate request Verify each of these paths will return the device to the Ready state and the TimeActivatedOperateTermination-: – Force a 'Test not ok' – Send a correct Cancel request

Table 29 – SBOns test cases

Test case	Test case description
sSBOns1	Send a correct Select request Send correct Operate request
sSBOns2	Send a correct Select request Verify each of these paths will return the device to the Unselected state: – Send a correct Cancel request – Wait for select timeout – Send a Release request – Send an Operate request, resulting in 'Test not ok'
sSBOns3	Send a correct Select request – Send an incorrect TimeActivatedOperate request resulting in respond-
sSBOns4	Send a correct Select request Send a TimeActivatedOperate request, thereby making sure the device will generate a 'Test Ok'. Verify the TimeActivatedOperateTermination+
sSBOns5	Send a correct Select request Send a correct TimeActivatedOperate request Verify each of these paths will return the device to the Ready state and the TimeActivatedOperateTermination-: – Force a 'Test not ok' – Send correct Cancel request
sSBOns6	Send a Select request resulting in respond-. Verify the device returns to the Unselected state.
sSBOns7	Send a correct Select request Verify that sending multiple Operate Many requests will return the device to the Ready state Verify that sending a Cancel request will return the device to the Unselected state

Table 30 – DOes test cases

Test case	Test case description
sDOes1	Send a correct Operate request Verify each of these paths will return the device to the Ready state and verify the CommandTermination: – force the equipment simulator to move to the requested new state – force the equipment simulator to keep the old state (AddCause: Time-limit-over or Invalid-position) force the equipment simulator to move to the 'between' state (AddCause: Invalid-position)
sDOes2	Send an Operate request, resulting in 'Test not ok'.
sDOes3	Send a TimeActivatedOperate request, resulting in respond-
sDOes4	Send a correct TimeActivatedOperate request Verify the TimeActivatedOperateTermination+ Verify each of these paths will return the device to the Ready state and verify the CommandTermination: – force the equipment simulator to move to the requested new state – force the equipment simulator to keep the old state (AddCause: Time-limit-over or Invalid-position) – force the equipment simulator to move to the 'between' state (AddCause: Invalid-position)
sDOes5	Send a correct TimeActivatedOperate request Verify each of these paths will return the device to the Ready state and the TimeActivatedOperateTermination-: – Force a 'Test not ok' – Send a correct Cancel request

6.2.4.18 Time and time synchronisation model

6.2.4.18.1 Positive test cases

The test cases listed in Table 31 shall apply.

Table 31 – Time positive test cases

Test case	Test case description
sTm1	Verify the DUT supports and executes the SCSM time synchronisation as configured in SCL
sTm2	Check report/logging timestamp accuracy matches the documented timestamp quality of the server
sTm3	Verify that when the device supports time zones and daylight saving the time stamp of events and disturbance files are UTC time
sTm4	Verify the time management settings in logical node LTIM
sTm5	Verify the time master supervision in logical node LTMS

6.2.4.18.2 Negative test cases

The test cases listed in Table 32 shall apply.

Table 32 – Time negative test cases

Test case	Test case description
sTmN1	Verify that when time synchronisation communication lost is detected after a specified period
sTmN2	On synchronisation error, deviation beyond time stamp tolerance should be detected

6.2.4.19 File transfer model

6.2.4.19.1 Positive test cases

The test cases listed in Table 33 shall apply.

Table 33 – File transfer positive test cases

Test case	Test case description
sFt1	Request a GetServerDirectory(FILE) with correct parameters and verify the response (IEC 61850-7-2 Subclause 8.2.2, PIXIT)
sFt2	For each responded file: – request a GetFile with correct parameters and verify the response (IEC 61850-7-2 Subclause 23.2.1) – request a GetFileAttributeValues with correct parameters and verify the response (IEC 61850-7-2 Subclause 23.2.4) – request a DeleteFile with correct parameters and verify the response (IEC 61850-7-2 Subclause 23.2.3)
sFt3	Verify the SetFile service with a small and large file and the maximum number of maximum sized file
sFt4	Request a GetFile from two clients simultaneously if more than one client association is supported (PIXIT)
sFt5	Request a GetServerDirectory(FILE) with the wildchar parameter and verify the response (IEC 61850-7-2 Subclause 7.2.2)

6.2.4.19.2 Negative test cases

The test cases listed in Table 34 shall apply.

Table 34 – File transfer negative test cases

Test case	Test case description
sFtN1	Request following file transfer services with an unknown file name and verify the appropriate response– service error – GetFile (IEC 61850-7-2 Subclause 23.2.1) – GetFileAttributeValues (IEC 61850-7-2 Subclause 23.2.4) – DeleteFile (IEC 61850-7-2 Subclause 23.2.3)

6.2.4.20 Network redundancy

6.2.4.20.1 Test cases

The test cases listed in Table 35 shall apply.

Table 35 – Network redundancy test cases

Test case	Test case description
sPrp1	Verify that the device support PRP redundancy according to IEC 62349-3
sPrp2	Verify that if one channel fails no packets are lost in the device and LCCH data values are updated
sHsr1	Verify that the device support HSR redundancy according to IEC 62349-3
sHsr2	Verify that if one channel fails no packets are lost in the device and LCCH data values are updated

6.2.5 Test cases to test a client device

6.2.5.1 General

This part of the IEC 61850 series specifies the test system architecture and abstract test cases (see 6.2.5.7 to 6.2.5.19) for client devices. The abstract test cases shall be used for the definition of test procedures to run in tests.

NOTE The SCSM specific test procedures are provided by test facilities agreed upon by the market participants.

6.2.5.2 Test system architecture to test a client device

In order to be able to perform a client device test, a minimum test set-up is necessary. The test architecture contains:

- DUT with optional HMI;
- multi server simulator to respond to TPAA messages from the DUT;
- test master to start/stop test cases, start/stop the analyze and archive test results;
- time master;
- engineering tool to configure the DUT;
- protocol analyzer to store the all network traffic for each test case.

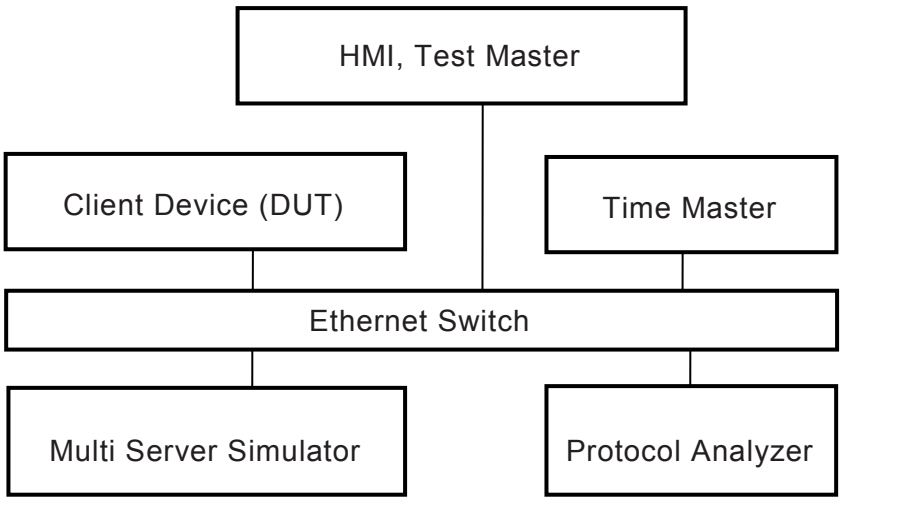

Figure 4 – Test system architecture to test a client device

The test system shall include documentation regarding test system hardware and test system software.

6.2.5.3 Documentation and version control test procedure overview

The test cases listed in Table 36 shall apply.

Table 36 – Client documentation test cases

Test case	Test case description
cDoc1	Check if the major/minor software version in the PICS documentation and the DUT do match (IEC 61850-4)
cDoc2	Check if the major/minor software version in the PIXIT documentation and software version of the DUT does match (IEC 61850-4). PIXIT shall indicate the required information as requested in the test cases
cDoc3	Check if the major/minor software version in the MICS documentation and software version of the DUT does match (IEC 61850-4). MICS shall indicate which CDC's and/or CDC parts are supported by the DUT, for example arrays
sDoc4	Check if the major/minor software version in the TICS documentation and software version of the DUT does match (IEC 61850-4). TICS shall indicate the supported technical issues.

6.2.5.4 Configuration file test cases

The test cases listed in Table 37 shall apply.

Table 37 – Client configuration test cases

Test case	Test case description
cCnf1	Check if the DUT processes the data names, data types as configured in the SCL configuration file.
cCnf2	Change at least 5 end-user configurable parameters that are displayed by the DUT in the SCL configuration file, configure the DUT using the SCL configuration file (using the supplied configuration tool) and check the updated configuration. Restore the original SCL file and re-configure the DUT to its original state.
cCnf3	Verify that client can handle the ConfigRev management in SCL and exposed by the server in LLN0.NamPlt.configRev as described in the PIXIT. On a mismatch the DUT shall behave as described In the PIXIT (note that, if the PIXIT describes that the DUT does not check such a mismatch, no action is required by the DUT)

6.2.5.5 Data model test cases

The test cases listed in Table 38 shall apply.

Table 38 – Client data model test cases

Test case	Test case description
cMdl1	Verify that the client can handle the maximum name length according to IEC 61850-7-2 Subclause 22.2 and SCSM and expands objects like SDOs correctly (PIXIT)
cMdl2	Verify that DUT supports the following naming conventions for the supported control blocks a) unbuffered report control block – not indexed b) unbuffered report control block – indexed c) buffered report control blocks d) setting group control block e) GOOSE control block f) Log control block
cMdl3	Verify that DUT can read and process the mandatory and optional attributes from the CDCs in IEC 61850-7-3 unless stated otherwise in the MICS

6.2.5.6 Mapping of ACSI models and services test cases

Test items shall be grouped together in tables. The tables shall reflect the applicable service models specified in IEC 61850-7-2 Figure 3:

– application association (cAss);
– server, Logical device, Logical node, Data, and Data Attribute model (cSrv);
– data set model (cDs);
– service tracking (cTrk);
– substitution model (cSub);
– setting group model (cSg);
– unbuffered report control model (cRp);
– buffered report control model (cBr);
– log control model (cLog);
– GOOSE control block model (cGcb);
– control model (cCtl);
– time and time synchronisation model (cTm);
– file transfer model (cFt).

Test cases are defined for each ACSI model and services in the following categories:

- positive = verification of normal conditions, typically resulting in response+;
- negative = verification of abnormal conditions, typically resulting in response–.

A test case is mandatory when the applicable ACSI model and ACSI service is supported by the DUT. This is specified in the PICS according to IEC 61850-7-2, Annex A. The test result interpretation (passed/failed) depends on the declared capabilities.

6.2.5.7 Application association model

6.2.5.7.1 Positive test cases

The test cases listed in Table 39 shall apply.

Table 39 – Association positive test cases

Test case	Test case description
cAss1	Associate and force the DUT to release or abort a TPAA association (IEC 61850-7-2 Subclause 8.3)
cAss2	Force the DUT to associate with maximum number of servers simultaneously (PIXIT).
cAss3	Verify that the DUT restores the association after the association of one server is lost and that this has no effect on the other active associations of the other servers
cAss4	Verify the DUT can handle servers with small and large MMS PDU size, the DUT should keep on proposing it's original MMS PDU size (PIXIT)

6.2.5.7.2 Negative test cases

The test cases listed in Table 40 shall apply.

Table 40 – Association negative test cases

Test case	Test case description
cAssN1	Associate and server responds with negative answer due to AccessPointReference.
cAssN2	Associate and server responds with negative answer due to AuthenticationParameter.
cAssN3	Associate and server releases TPAA association (IEC 61850-7-2 Subclause 8.3). DUT should try to re-establish the association after the configured period (PIXIT).
cAssN4	Associate and server-abort TPAA association (IEC 61850-7-2 Subclause 8.3). DUT should try to re-establish the association after the configured period (PIXIT).
cAssN5	Associate and server denies TPAA association (IEC 61850-7-2 Subclause 8.3). DUT should try to re-establish the association after the configured period (PIXIT).
cAssN6	Disconnect the communication interface between server and the Ethernet switch such that the link between DUT and the Ethernet switch stays active. The DUT shall detect link lost within a specified period. Once the link is re-established the DUT should try to establish the association again.
cAssN7	Interrupt and restore the power supply, the DUT shall establish the configured associations when ready (PIXIT).

6.2.5.8 Server, logical device, logical node, and data model

6.2.5.8.1 Positive test cases

The test cases listed in Table 41 shall apply.

Table 41 – Server positive test cases

Test case	Test case description
cSrv1	If the DUT implements Autodescription, (See Note 1) force the DUT to start the autodescription and check the DUT requests a GetServerDirectory(LOGICAL-DEVICE) to all the logical devices of the configured servers (see Note 2) (IEC 61850-7-2 Subclause 7.2.2)
cSrv2	If the DUT implements Autodescription, for each GetServerDirectory(LOGICAL-DEVICE) response check the DUT issues a GetLogicalDeviceDirectory request (IEC 61850-7-2 Subclause 9.2.1)
cSrv3	If the DUT “implements Autodescription”, for each GetLogicalDeviceDirectory response check the DUT issues a GetLogicalNodeDirectory(DATA) request (IEC 61850-7-2 Subclause 10.2.2)
cSrv4	If the DUT “implements Autodescription”, for a subset of the GetLogicalNodeDirectory(DATA) response check the DUT issues at least one of the following services: a) GetDataDirectory request and check response (IEC 61850-7-2 Subclause 11.4.4) b) GetDataDefinition request and check response (IEC 61850-7-2 Subclause 11.4.5)
cSrv5	Verify that after start up the DUT is able to update the process values of the configured servers.
cSrv6	Request a SetDataValues of the different basic types (with for example FC=CF) and check the services (IEC 61850-7-2 Subclause 11.4.3)
cSrv7	Request GetDataValues and check if the DUT updates its model (IEC 61850-7-2 Subclause 11.4.2)
cSrv8	Request GetAllDataValues for the required functional constraints and check if the DUT updates its model (IEC 61850-7-2 Subclause 10.2.3)
cSrv9	Verify that the client is able to set/reset blkEna (IEC 61850-7-3 Subclause 6.2.6)

Implement Autodescription means that there is a way to configure the DUT to update the image of the model of one of the servers it has to communicate with using the ACSI services.

Configured servers means the servers the DUT is configured to communicated with. The DUT at least needs to know the parameters to establish an association with them.

6.2.5.8.2 Negative test cases

The test cases listed in Table 42 shall apply.

Table 42 – Server negative test cases

Test case	Test case description
cSrvN1	If the DUT implements autodescription, force the DUT to start the autodescription and check the DUT still communicates with other servers when it requests the following services with negative response: a) GetServerDirectory(LOGICAL-DEVICE), b) GetLogicalDeviceDirectory, c) GetLogicalNodeDirectory(DATA), d) GetDataDirectory, e) GetDataDefinition.
cSrvN2	Check that the DUT is able to communicate with other connected servers after a request for GetAllDataValues fails in the following circumstances: a) The response is negative. b) The response comes with mismatching data objects.
cSrvN3	Check that the DUT is able to communicate with other connected servers after a request for GetDataValues fails in the following circumstances: a) The response is negative. b) The response comes with mismatching data objects. c) The value is out of the valid range for this data.
cSrvN4	Check that the DUT is able to communicate with other connected servers after a request for SetDataValues fails in the following circumstances: a) The response is negative. b) One of the data values is read-only
cSrvN5	If the DUT detects/notifies changes in the “Quality” attribute, use the SERVER SIMULATOR to force different values in the Quality of the measured/status values monitored by the DUT and check the behaviour described in the PIXIT.
cSrvN6	If the DUT detects/notifies changes in the timeStamp’s “TimeQuality” attribute, use the SERVER SIMULATOR to force different values in the TimeQuality of the measured/status values monitored by the DUT and check the behaviour described in the PIXIT.

6.2.5.9 Data set model

6.2.5.9.1 Positive test cases

The test cases listed in Table 43 shall apply.

Table 43 – Data set positive test cases

Test case	Test case description
cDs1	If the DUT implements autodescription, force it to start autodescription and check if it requests a GetLogicalNodeDirectory(DATASET) of the Logical Nodes of the configured servers (IEC 61850-7-2 Subclause 10.2.2)
cDs2	If the DUT implements autodescription, force it to start autodescription and check it requests a GetDataSetDirectory of all the DataSets of the server (IEC 61850-7-2 Subclause 13.3.6)
cDs3	Check the DUT can request a GetDataSetValues and handle the respond (IEC 61850-7-2 Subclause 13.3.2)
cDs4	Check the DUT can request a SetDataSetValues and handle the respond (IEC 61850-7-2 Subclause 13.3.3)
cDs5	Verify that the DUT checks the pre-configured datasets in the SCD file. If any deviation is detected the DUT behaves as specified in the PIXIT
cDs6	If the DUT creates persistent / non-persistent datasets dynamically after starting up check that the DUT sends the CreateDataSet services according to configuration. PIXIT (IEC 61850-7-2 Subclause 13.3.4)
cDs7	Request a DeleteDataSet service and check the DUT sends the request properly and is able to process the response of the server (IEC 61850-7-2 Subclause 13.3.5)

Test case	Test case description
cDs8	Verify that a persistent data set can be handled with the maximum name length for data set and a data set member (IEC 61850-7-2 Subclause 22.2)
cDs9	Verify that a non-persistent data set can be handled with the maximum name length for data set and a data set member (IEC 61850-7-2 Subclause 22.2)

6.2.5.9.2 Negative test cases

The test cases listed in Table 44 shall apply.

Table 44 – Data set negative test cases

Test case	Test case description
cDsN1	If the DUT implements autodescription, force the DUT to start the autodescription and check the DUT still communicates with other servers when it request the following services with negative response: a) GetLogicalNodeDirectory(DATA-SET) b) GetDataSetDirectory
cDsN2	Check that the DUT still communicates with other servers properly when it requests a GetDataSetValues to one of them and the following situations happen: a) The response is negative. b) The response comes with more/less elements than expected c) The response comes with reordered elements of different types d) The response comes with reordered elements of the same type
cDsN3	Check that the DUT still communicates with other servers properly when it requests a SetDataSetValues to one of them and the response is negative.
cDsN4	If the DUT creates persistent / non-persistent datasets dynamically after starting up check the DUT still communicates with other servers when it requests a CreateDataSet with negative response
cDsN5	If the DUT configures the datasets dynamically after starting up check the DUT still communicates with other servers when it requests a DeleteDataSet with negative response

6.2.5.10 Service tracking model

6.2.5.10.1 Test cases

The test cases listed in Table 45 shall apply.

Table 45 – Service tracking test cases

Test case	Test case description
cTrk1	Verify that the DUT can process tracking of control block services: Buffered reporting, LTRK.BrcbTrk
cTrk2	Verify that the DUT can process tracking of control block services: Unbuffered reporting, LTRK.UrcbTrk
cTrk3	Verify that the DUT can process tracking of control block services: Log control block, LTRK.LocbTrk
cTrk4	Verify that the DUT can process tracking of control block services: GOOSE control block, LTRK.GocbTrk
cTrk5	Verify that the DUT can process tracking of control block services: Multicast sampled values control block, LTRK.MsvcbTrk
cTrk6	Verify that the DUT can process tracking of control block services: Unicast sampled values control block, LTRK.UsvcbTrk
cTrk7	Verify that the DUT can process tracking of control block services: Setting group control block, LTRK.SgcbTrk
cTrk8	Verify that the DUT can process tracking of control services: Single point control, LTRK.SpcTrk
cTrk9	Verify that the DUT can process tracking of control services: Double point control, LTRK.DpcTrk
cTrk10	Verify that the DUT can process tracking of control services: Integer control, LTRK.IncTrk
cTrk11	Verify that the DUT can process tracking of control services: Enumerated control, LTRK.EncTrk
cTrk12	Verify that the DUT can process tracking of control services: Analogue process value control with float command, LTRK.ApcFTrk
cTrk13	Verify that the DUT can process tracking of control services: Analogue process value control with integer command, LTRK.ApcIntTrk
cTrk14	Verify that the DUT can process tracking of control services: Binary step control, LTRK.BscTrk
cTrk15	Verify that the DUT can process tracking of control services: Integer step control, LTRK.IscTrk
cTrk16	Verify that the DUT can process tracking of control services: Binary analogue process value control, LTRK.BacTrk
cTrk17	Verify that the DUT can process tracking of other supported common services, LTRK.GenTrk

6.2.5.11 Substitution model

6.2.5.11.1 Test cases

The test cases listed in Table 46 shall apply.

Table 46 – Substitution test cases

Test case	Test case description
cSub1	Verify the DUT can enable substitution, enter a substituted value and disable substitution
cSub2	Verify the DUT can display the source “substituted” for substituted values
cSub3	Verify the DUT can display the source “substituted” for values substituted by another client
cSub4	Verify that the DUT can handle the maximum name length for substitution values (IEC 61850-7-2 Subclause 22.2)

6.2.5.12 Setting group control model

6.2.5.12.1 Positive test cases

The test cases listed in Table 47 shall apply.

Table 47 – Setting group positive test cases

Test case	Test case description
cSg1	If the DUT implements autodescription, force it to start autodescription and check if it requests GetLogicalNodeDirectory(SGCB) and check response+
cSg2	Verify the DUT can select a setting group (IEC 61850-7-2 Clause 16, Figure 22); a) SelectActiveSG of the first setting group (IEC 61850-7-2 Clause 16.3.2) b) GetSGCBValues to verify active setting group (IEC 61850-7-2 Clause 16.3.7) c) Repeat for another setting group
cSg3	Verify the DUT can get setting group values [FC=SG] (IEC 61850-7-2 Clause 16, Figure 22); a) SelectActiveSG of the first setting group b) Use GetDataValues [FC=SG] to verify the values of the fist setting group c) Repeat for another setting group
cSg4	Verify the DUT can edit setting group values a) SelectEditSG of the first setting group b) Request GetEditSGValue to read the edit value (IEC 61850-7-2 Clause 16.3.6) c) Use SetEditSGValue to change the edit value (IEC 61850-7-2 Clause 16.3.4) d) Use ConfirmEditSGValues to confirm the changes (IEC 61850-7-2 Clause 16.3.5)
cSg5	Verify the device can cancel the edit procedure a) SelectEditSG of the first setting group b) Cancel processing with SelectEditSG where SettingGroupNumber is 0 (zero)
cSg6	If the device is able to read the optional ResvTms, verify the DUT does not request SelectEditSG if ResvTms >0 (IEC 61850-7-2 Clause 16.2.2.8)
cSg7	If the device is able to read the optional EditSG, verify the DUT does not request SelectEditSG if EditSG >0 (IEC 61850-7-2 Clause 16.2.2.5)

6.2.5.12.2 Negative test cases

The test cases listed in Table 48 shall apply.

Table 48 – Setting group negative test cases

Test case	Test case description
cSgN1	Force SERVER SIMULATOR to return response- for the following services and verify the DUT continues as before a) SelectActiveSG (IEC 61850-7-2 Subclause 16.3.2) b) GetSGCBValues (IEC 61850-7-2 Subclause 16.3.7)

6.2.5.13 Unbuffered reporting model

6.2.5.13.1 Positive test cases

The test cases listed in Table 49 shall apply.

Table 49 – Unbuffered reporting positive test cases

Test case	Test case description
cRp1	If the DUT implements autodescription, force it to start autodescription and check if it requests a GetLogicalNodeDirectory(URCB) of the logical nodes declared in the PIXIT of all configured servers.
cRp2	If the DUT configures the server's Unbuffered ReportControlBlock parameters after startup using SetURCBValues, check that the SetURCBValues are sent with the configured values. (IEC 61850-7-2 Subclause 17.2.5.4)
cRp3	Verify the DUT is able to process the reports with different optional fields: Force the DUT to configure/enable a URCB with useful optional fields combinations: sequence-number, report-time-stamp, reason-for-inclusion, data-set-name and/or data-reference, force/trigger a report and check the DUT is able to process the reports and updates its database. (IEC 61850-7-2 Subclause 17.2.2.8)
cRp4	Verify the DUT is able to process the reports with different trigger conditions (IEC 61850-7-2 Subclause 17.2.2.11) Configure and enable a URCB with all supported optional fields and check the reports are transmitted according to the following (supported) trigger conditions: a) on integrity b) on update (dupd) c) on update with integrity d) on data change (dchg) e) on data change and quality change (dchg+qchg) f) on data change and quality change with integrity period (dchg+qchg)
cRp5	Verify the DUT is able to process segmented reports
cRp6	Verify the DUT can change the (pre-)configured Buffer Time (IEC 61850-7-2 Subclause 17.2.2.9)
cRp7	Verify the DUT can force a General interrogation (IEC 61850-7-2 Subclause 17.2.2.13)
cRp8	Verify that after start up the DUT configures and enables the URCBs as specified in the SCD file. The DUT only may write to the "dyn" URCB fields in the SCL.
cRp9	Verify that the DUT can handle reporting of complex structured data (for example WYE and DEL data objects)
cRp10	Verify that the DUT can handle reporting of basic data (for example stVal and quality)
cRp11	Verify that the DUT can handle a URCB, RptID and DatSet with maximum name length (IEC 61850-7-2 Subclause 22.2)
cRp12	Verify that the DUT can change the dataset elements of a dynamic dataset previously used in a URCB resulting in a ConfRev increment by the server
cRp13	Verify that the DUT configures another indexed URCB when the another client has reserved the indexed URCB before

6.2.5.13.2 Negative test cases

The test cases listed in Table 50 shall apply.

Table 50 – Unbuffered reporting negative test cases

Test case	Test case description
cRpN1	If the DUT implements autodescription, force the DUT to start the autodescription and check that the DUT still communicates with other servers when it request GetLogicalNodeDirectory (URCB) with negative response.
cRpN2	Check that the DUT still works properly when it requests a GetURCBValues when the response is negative.
cRpN3	Check that the DUT still works properly when it requests a SetURCBValues when the response is negative.
cRpN4	Check that the DUT still works properly when it request a SetURCBValues and the URCB is reserved (Resv=TRUE, PIXIT)
cRpN5	Report with not supported OptFlds. Check that the DUT does not collapse if it receives a Report with a non-configured or non-supported OptFlds.
cRpN6	Report with not supported TrgOps. Check that the DUT does not collapse if it receives a report with a non-configured or non-supported Trigger Option.
cRpN7	Mismatching reports: a) Report with unknown DataSet. b) Report with unknown RptId c) Report with incorrect references of the Data. d) Report with incorrect types in the Data. Check the behaviour described in the PIXIT.
cRpN8	Verify that the DUT detects a change in the ConfRev attribute (Configuration revision, IEC 61850-7-2 Subclause 14.2.2.7) of the Report Control Block. When the DUT does not perform the ConfRev check it should check the dataset elements. The means of detection needs to be specified in the PIXIT.

6.2.5.14 Buffered reporting model

6.2.5.14.1 Positive test cases

The test cases listed in Table 51 shall apply.

Table 51 – Buffered reporting positive test cases

Test case	Test case description
cBr1	If the DUT implements autodescription, force it to start autodescription and check if it requests a GetLogicalNodeDirectory(BRCB) of the logical nodes declared in the PIXIT of all configured servers.
cBr2	If the DUT configures the server's Buffered ReportControlBlock parameters after startup using SetBRCBValues, check that the GetBRCBValues/SetBRCBValues are sent with the configured values (IEC 61850-7-2 Subclause 17.2.3.4)
cBr3	Verify the DUT is able to process the reports with different optional fields: Force the DUT to configure/enable a BRCB with the useful optional fields combinations: sequence-number, report-time-stamp, reason-for-inclusion, data-set-name, data-reference, buffer-overflow, entryID and conf-revision , force/trigger a report and check the DUT is able to process the reports and updates its database (IEC 61850-7-2 Subclause 17.2.2.8)

Test case	Test case description
cBr4	Verify the DUT is able to process the reports with different trigger conditions (IEC 61850-7-2 Subclause 17.2.2.11) Configure and enable a BRCB with all useful optional fields: sequence-number, report-time-stamp, reason-for-inclusion, data-set-name, data-reference, buffer-overflow, entryID and conf-revision and check the reports are transmitted according to the following (supported) trigger conditions: a) on integrity b) on update (dupd) c) on update with integrity d) on data change (dchg) e) on data and quality change (dchg+qch) f) on data and quality change with integrity period (dchg+qchg)
cBr5	Verify the DUT is able to process segmented reports
cBr6	Verify the DUT can change the (pre-)configured Buffer Time (IEC 61850-7-2 Subclause 17.2.2.9)
cBr7	Verify the DUT can force a General interrogation (IEC 61850-7-2 Subclause 17.2.2.13)
cBr8	Verify that after startup the DUT configures and enables the BRCBs as configured in the SCD file (and actually used). The DUT only may write to the “dyn” BRCB fields in the SCL.
cBr9	Verify that the DUT can handle reporting of complex structured data (for example WYE and DEL data objects)
cBr10	Verify that the DUT can handle reporting of basic data (for example stVal and quality)
cBr11	Verify that the DUT can handle a BRCB, RptID and DatSet with maximum name length (IEC 61850-7-2 Subclause 22.2)
cBr12	Verify that the DUT can change the dataset elements of a dynamic dataset previously used in a BRCB resulting in a ConfRev increment by the server
cBr13	Verify that the DUT configures another indexed BRCB when the another client has configured the indexed BRCB before
cBr20	Verify the DUT is able to process reports buffered during an lost association a) without buffer overflow (PIXIT) b) with buffer overflow
cBr21	Verify the DUT is able to request specific buffered reports after restoring a lost association by setting the EntryID
cBr22	Verify the DUT is able to purge buffered reports
cBr23	Verify the client first sets the ResvTms attribute if this attribute is available and has value 0

6.2.5.14.2 Negative test cases

The test cases listed in Table 52 shall apply.

Table 52 – Buffered reporting negative test cases

Test case	Test case description
cBrN1	If the DUT implements autodescription, force the DUT to start the autodescription and check that the DUT still communicates with other servers when it request GetLogicalNodeDirectory (BRCB) with negative response.
cBrN2	Check that the DUT still works properly when it requests a GetBRCBValues when the response is negative.
cBrN3	Check that the DUT still works properly when it requests a SetBRCBValues when the response is negative.
cBrN4	Check that the DUT still works properly when it requests a SetBRCBValues and the BRCB is used by or pre-assigned to another DUT. (PIXIT)
cBrN5	Report with not supported OptFlds. Check that the DUT does not collapse if it receives a Report with a non-configured or non-supported OptFlds.
cBrN6	Report with not supported TrgOps. Check that the DUT does not collapse if it receives a Report with a non-configured or non-supported Trigger Option.
cBrN7	Mismatching reports: a) Report with unknown DataSet. b) Report with unknown RptID c) Report with incorrect references of the Data (when data references are enabled). d) Report with incorrect types in the Data. Check the behaviour described in the PIXIT.
cBrN8	Verify that the DUT detects a change in the ConfRev attribute (Configuration revision, IEC 61850-7-2 Subclause 14.2.2.7) of the Report Control Block. When the DUT does not perform the ConfRev check it should check the dataset elements. The means of detection needs to be specified in the PIXIT.
cBrN9	Verify the DUT can handle a severe buffer overflow with SetBRBValues(EntryID) response-

6.2.5.15 Log model

6.2.5.15.1 Positive test cases

The test cases listed in Table 53 shall apply.

Table 53 – Log positive test cases

Test case	Test case description
cLog1	If the DUT implements autodescription, force it to start autodescription and check if it requests a GetLogicalNodeDirectory (LOG) of the logical nodes declared in the PIXIT of all configured servers.
cLog2	If the DUT implements autodescription, force it to start autodescription and check if it requests a GetLogicalNodeDirectory(LCB) of the logical nodes declared in the PIXIT of all configured servers.
cLog3	If the DUT implements autodescription, force it to start autodescription and check if it requests a GetLogStatusValues of the LOGs found with the GetLogicalNodeDirectory(LCB) services
cLog4	If the DUT implements autodescription, force it to start autodescription and check if it requests a GetLCBValues of the LCBs found with the GetLogicalNodeDirectory(LCB) services (IEC 61850-7-2 Subclause 17.3.2.5)
cLog5	If the DUT configures the server's LogControlBlock parameters after startup using SetLCBValues, check that the SetLCBValues are sent with the configured values (IEC 61850-7-2 Subclause 17.3.2.6)
cLog6	Force the DUT to enable the Logging of at least one LOG of the server and check the DUT send the request correctly.
cLog7	Force the DUT to QueryLogByTime or QueryLogAfter and check the DUT updates its database with the Log entries received (IEC 61850-7-2 Subclause 17.3.5)
cLog8	Verify that the DUT can handle a LCB and DatSet with maximum name length (IEC 61850-7-2 Subclause 22.2)

6.2.5.15.2 Negative test cases

The test cases listed in Table 54 shall apply.

Table 54 – Log negative test cases

Test case	Test case description
cLogN1	If the DUT implements autodescription, force the DUT to start the autodescription and check that the DUT still communicates with other servers when it request GetLogicalNodeDirectory (LCB) and GetLogicalNodeDirectory (LOG) with negative response.
cLogN2	Check that the DUT still works properly when it requests a GetLCBValues/GetLogStatusValues when the response is negative.
cLogN3	Check that the DUT still works properly when it requests a SetLCBValues when the response is negative.

6.2.5.16 GOOSE control block

6.2.5.16.1 Test cases

The test cases listed in Table 55 shall apply.

Table 55 – GOOSE control block test cases

Test case	Test case description
cGcb1	Verify the DUT can send a GetGoCBValues request and handle the response (IEC 61850-7-2 Subclause 18.2.2.5)
cGcb2	Verify the DUT can send a SetGoCBValues request and handle the response (IEC 61850-7-2 Subclause 18.2.6)

6.2.5.17 Control model

6.2.5.17.1 General test cases

The test cases listed in Table 56 shall apply.

Table 56 – Control general test cases

Test case	Test case description
cCtl1	Check if the DUT is able to set the TEST field in the SelectWithValue and Operate requests (PIXIT).
cCtl2	Check if the DUT is able to set the CHECK (Synchro-Check or Interlock-Check bits) in the commands (PIXIT) for the supported control models.
cCtl3	Check if the DUT is able to change control model using online services (PIXIT).
cCtl4	Verify the values of originator category & identification and the control number values (PIXIT)
cCtl5	Check if the DUT reacts in a proper way when it detects a control model mismatch (PIXIT): a) Server status-only, DUT expects controllable b) Server SBO, DUT expects direct operate c) Server direct operate, DUT expects SBO
cCtl6	Check if the DUT reacts in a proper way when it detects a control model is not initialized in the SCL file (PIXIT)

6.2.5.17.2 Control model specific test cases

The test cases listed in Table 57, Table 58, Table 59 and Table 60 shall apply.

Table 57 – SBOes test cases

Test case	Test case description
cSBOes1	SelectWithValue [test not ok] resp-: Select device using SelectWithValue resulting in test not ok. Check the DUT indicates an error.
cSBOes2	SelectWithValue [test ok] resp+ and Operate[test ok] resp+ Select device using correct SelectWithValue. Perform a correct Operate request. Check the DUT indicates no error after receiving the command termination+
cSBOes3	SelectWithValue [test ok] resp+ and Operate[test not ok] resp- Perform a SelectWithValue and Operate request. The Operate results in test not ok. Check that the DUT realizes the operation failed.
cSBOes4	SelectWithValue [test ok] resp+ and Cancel Perform a correct Cancel request. Check the DUT indicates no error.
cSBOes5	SelectWithValue [test ok] resp+ and TimeActivatedOperate [test ok] resp+ Perform a correct TimeActivatedOperate request. Check that the DUT realizes the operation succeeded after the WaitForActivationTime and detects the CommandTermination with the result of the order.
cSBOes6	SelectWithValue [test ok] resp+ and TimeActivatedOperate [test ok] resp- Perform a SelectWithValue and TimeActivatedOperate request. The TimeActivatedOperate results in test not ok. Check that the DUT realizes the operation failed.

Table 58 – DOns test cases

Test case	Test case description
cDOns1	Operate[test ok] resp+ Perform a correct Operate request. Check that the DUT does not generate an error.
cDOns2	Operate[test not ok] resp- DUT requests Oper resulting in Test not ok. Check that the DUT realizes the operation failed.
cDOns3	TimeActivatedOperate [test not ok] resp- DUT requests TimeActivatedOperate resulting in Test not ok. Check that the DUT realizes the time operation failed.
cDOns4	TimeActivatedOperate [test ok] + TimerExpired[test ok] resp+ Send a TimeActivatedOperate request, thereby making sure the device will generate a 'test Ok'. Verify the WaitForActivationTime results in a timer expired ‘Test ok’ and that the DUT realizes the operation succeeded.
cDOns5	TimeActivatedOperate [test ok] + TimerExpired[test not ok] resp- Send a TimeActivatedOperate request, thereby making sure the device will generate a 'test Ok'. Force situation that the WaitForActivationTime results in a timer expired ‘Test not ok’. Check that the DUT realizes the operation failed.

Table 59 – SBOns test cases

Test case	Test case description
cSBOns1	Select[test not ok] resp-: DUT requests Select resulting in Test not ok. Check that the DUT realizes the select failed (PIXIT).
cSBOns2	Select[test ok] resp+ and Operate[test ok] resp+ Select a controllable object using Select. Perform a correct Operate request. Check that the DUT does not generate an error.
cSBOns3	Select[test ok] resp+ and Operate[test not ok] resp- of selected object. Perform a correct Operate request resulting in Test not ok. Check that the DUT realizes the operation failed.
cSBOns4	Select[test ok] resp+ and Cancel Perform a correct cancel request.
cSBOns5	Select[test ok] resp+ and TimeActivatedOperate [test ok] resp+ Perform a correct TimeActivatedOperate request. Check that the DUT realizes the operation succeeded after the WaitForActivationTime.
cSBOns6	Select[test ok] resp+ and TimeActivatedOperate [test not ok] resp- Perform a correct TimeActivatedOperate request resulting in test not ok. Check that the DUT realizes the operation failed.

Table 60 – DOes test cases

Test case	Test case description
cDOes1	Operate[test ok] resp+: Send a correct Operate request. a) Check that the DUT notices the operation ended positively when it receives the CommandTermination+. b) Check that the DUT notices the operation ended negatively when it receives the CommandTermination- (PIXIT)
cDOes2	Operate[test not ok] resp-: Send an Operate request, thereby making sure the device will generate a 'test not ok'. Check that the DUT realizes the operation failed (PIXIT)
cDOes3	TimeActivatedOperate [test not ok] resp-: Send a TimeActivatedOperate request, thereby making sure the device will generate a 'test not ok'. Check that the DUT realizes the operation failed.
cDOes4	TimeActivatedOperate [test ok] resp+: Send a correct TimeActivatedOperate Operate request. a) Check that the DUT realizes the operation request succeeded. b) Check that the DUT notices the operation ended positively when it receives the CommandTermination+. c) Check that the DUT notices the operation ended negatively when it receives the CommandTermination-.

6.2.5.18 Time and time synchronisation model

6.2.5.18.1 Positive test cases

The test cases listed in Table 61 shall apply.

Table 61 – Time positive test cases

Test case	Test case description
cTm1	Verify the DUT supports the SCSM time synchronisation, Change the time in the time server and verify the DUT uses the new time
cTm2	Check the DUT timestamp accuracy matches the documented timestamp quality

6.2.5.18.2 Negative test cases

The test cases listed in Table 62 shall apply.

Table 62 – Time negative test cases

Test case	Test case description
cTmN1	Verify that a lost time synchronisation is detected after a specified period and the timestamp quality invalid is set
cTmN2	Verify the DUT handles the time stamp quality coming from the time server

6.2.5.19 File transfer model

6.2.5.19.1 Positive test cases

The test cases listed in Table 63 shall apply.

Table 63 – File transfer positive test cases

Test case	Test case description
cFt1	Verify that the DUT requests a GetServerDirectory(FILE) with correct parameters and handles the response (IEC 61850-7-2 Subclause 7.2.2)
cFt2	Verify that the DUT requests a GetFileAttributeValues with correct parameters and verify the DUT handles the response (IEC 61850-7-2 Subclause 23.2.4)
cFt3	Verify that the DUT requests a GetFile with correct parameters and verify the DUT handles the response (IEC 61850-7-2 Subclause 23.2.1)
cFt4	The DUT requests a SetFile service with a small and large file and verify the DUT sends the resulting file(s)
cFt5	Verify the DUT requests a DeleteFile with correct parameters and verify the DUT handles the response

6.2.5.19.2 Negative test cases

The test cases listed in Table 64 shall apply.

Table 64 – File transfer negative test cases

Test case	Test case description
cFtN1	Force SERVER SIMULATOR to respond– on GetFile request, and verify the DUT reports an error
cFtN2	Force SERVER SIMULATOR to respond– on GetFileAttributeValues request, and verify the DUT reports an error
cFtN3	Force SERVER SIMULATOR to respond– on SetFile request, and verify the DUT reports an error

6.2.6 Test cases to test sampled values device

6.2.6.1 General

This part of the IEC 61850 series specifies test system architecture and abstract test cases for sampled values devices. The abstract test cases shall be used for the definition of test procedures to run in tests.

NOTE The SCSM specific test procedures are provided by test facilities agreed upon by the market participants.

6.2.6.2 Test system architecture to test a sampled values publishing device

In order to be able to perform a sampled values publishing device test, a minimum test set-up is necessary. The test architecture contains:

– DUT – SV publisher;
– time master;
– engineering tool to configure the DUT;
– high performance protocol analyzer to store the all network traffic for each test case;
– signal generator to generate current and/or voltage signals.

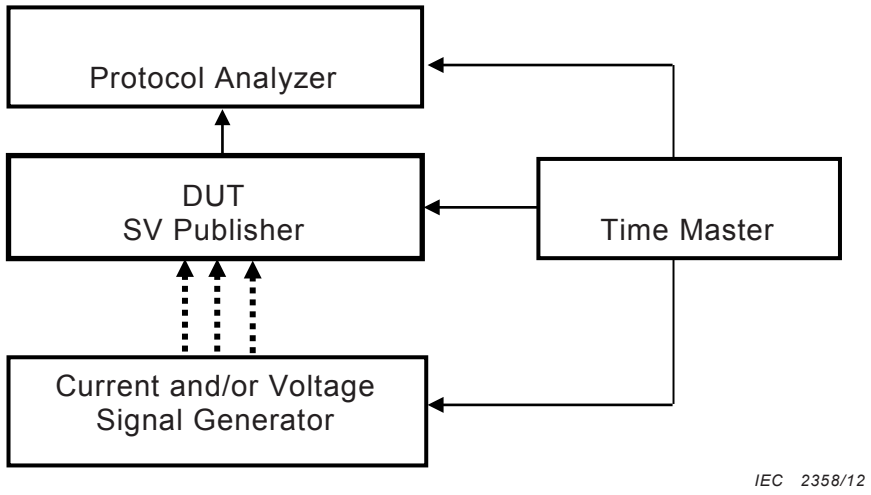

Figure 5 – Test system architecture to test a sampled values publishing device

The test system shall include documentation regarding test system hardware and test system software.

6.2.6.3 Test system architecture to test a sampled values subscribing device

In order to be able to perform a sampled values subscribing device test, a minimum test set-up is necessary. The test architecture contains:

– DUT – SV subscriber;
– engineering tool to configure the DUT;
– high performance protocol analyzer to store the all network traffic for each test case;

– SV simulator to publish correct and incorrect SV messages

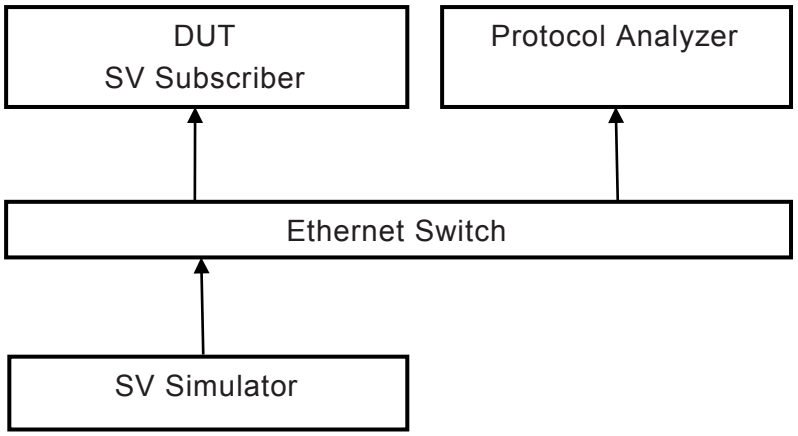

Figure 6 – Test system architecture to test a sampled values subscribing device

The test system shall include documentation regarding test system hardware and test system software.

6.2.6.4 Documentation and version control test procedure overview

The test cases listed in Table 65 shall apply.

Table 65 – Sampled values documentation test cases

Test case	Test case description
svDoc1	Check if the major/minor software version in the PICS documentation and the DUT do match (IEC 61850-4)
svDoc2	Check if the major/minor software version in the PIXIT documentation and software version of the DUT does match (IEC 61850-4). PIXIT shall indicate the required information as requested in the test cases
svDoc3	Check if the major/minor software version in the MICS documentation and software version of the DUT does match (IEC 61850-4). MICS shall specify the contents of the sampled values message, including validity and source of each data object
svDoc4	Check if the major/minor software version in the TICS documentation and software version of the DUT does match (IEC 61850-4). TICS shall indicate the supported technical issues.

6.2.6.5 Configuration file test cases

The test cases listed in Table 66 shall apply.

Table 66 – Sampled values configuration test cases

Test case	Test case description
svCnf1	Test if the ICD configuration file conforms to the SCL schema (IEC 61850-6)
svCnf2	Check if the SCL configuration file corresponds with the actual names, data-sets, and values exposed by the DUT on the network.
svCnf3	Check if the server "SMVSettings" capabilities in the ICD “services” section do match with the IED capabilities
svCnf4	Verify the name and logical nodes in the SCL
svCnf5	Verify the logical node LLN0 in the SCL: – dataset – sampled value control block
svCnf6	Verify the sampled value dataset in the SCL
svCnf7	Verify the common data class SAV and scale factor values in the SCL
svCnf8	Verify the Multicast sampled value control block in the SCL
svCnf8	Verify the Unicast sampled value control block in the SCL
svCnf9	Verify that if the device does not supply all samples, ‘dummy’ SAV data objects might be referenced in the data set. To detect the difference between dummy and real samples in the SCL, the ICD shall have all LNs included but the ones that are not supported have the LN Mode preconfigured to "Off".

6.2.6.6 Data model test cases

The test cases listed in Table 67 shall apply.

Table 67 – Sampled values datamodel test cases

Test case	Test case description
svMdl1	Verify the presence of sampled values objects
svMdl2	Verify that the MSVCB is located in LLN0
svMdl3	Verify that the USVCB is located in LLN0

6.2.6.7 Mapping of ACSI models and services test cases

Test cases are defined in the following categories:

– Sampled value control block (svSvcb);
– Send SV message publish (svSvp);
– Send SV message subscriber (svSvs).

A test case is mandatory when the applicable ACSI model and ACSI service is supported by the DUT. This is specified in the PICS according to IEC 61850-7-2, Annex A. The test result interpretation (passed/failed) depends on the declared IED capabilities e.g. in the ICD file as well as on the test result.

6.2.6.8 Transmission of sampled values model

6.2.6.8.1 Sampled value control block test cases

The test cases listed in Table 68 shall apply.

Table 68 – Sampled value control block test cases

Test case	Test case description
svSvcb1	Request GetLogicalNodeDirectory(MSVCB) and check response+
svSvcb2	Request GetLogicalNodeDirectory(USVCB) and check response+
svSvcb3	Verify that MSVCB attributes can be read using GetMSVCBValues (IEC 61850-7-2 Subclause 19.2.2.3)
svSvcb4	Verify that USVCB attributes can be read using GetUSVCBValues (IEC 61850-7-2 Subclause 19.3.2.3)
svSvcb5	Verify that MSVCB attributes can be changed using SetMSVCBValues and no SV messages are transmitted anymore while SvEna=False (IEC 61850-7-2 Subclause 19.2.2.4)
svSvcb6	Verify that USVCB attributes can be changed using SetUSVCBValues and no SV messages are transmitted anymore while SvEna=False(IEC 61850-7-2 Subclause 19.3.2.4)
svSvcb7	Verify that ConfRev represents a count of the number of times the configuration with regard to xSVCB has been changed (IEC 61850-7-2 Subclause 19.2.1.6). Changes that shall be counted are: – deletion of a member of the data-set – re-ordering of members in the data-set – any change of a value of the attribute of the data set whose functional constraint equals CF – changing a value of an attribute of xSVCB – ConfRev shall never be 0 (zero) – Verify that after a restart of the publisher, the value of ConfRev remains unchanged
svSvcb8	Verify that when a SVCB is enabled, no changes of the attributes of the SVCB other than disabling shall be allowed
svSvcb9	When SVCB is disabled, set non-configurable attributes in the SVCB and verify the response– service error
svSvcb10	Verify that the transmission of Send SV messages matches the settings in the xSVCB

6.2.6.8.2 Send SV message publish test cases

The test cases listed in Table 69 shall apply.

Table 69 – Send SV message publish test cases

Test case	Test case description
svSvp1	Verify that the maximum delay time from taking the sample to sending the corresponding message is within the limit specified in PIXIT
svSvp2	Verify that the physical layer and connector match the SCSM and PIXIT
svSvp3	Verify that the format of the link layer matches the SCSM
svSvp4	Verify that the format of the application layer matches the SCSM
svSvp5	Verify the supported quality bits of the sample values
svSvp6	Verify that the samples are transmitted at specified number messages per cycle (PIXIT, SVCB)
svSvp7	Verify that SmpCnt will be incremented each time a new sampled value is taken.
svSvp8	Verify that the sampled values match with the analog signals
svSvp9	Verify that the voltage scaling parameters are configured as specified in the PIXIT and correctly applied
svSvp10	Verify that the current scaling parameters are configured as specified in the PIXIT and correctly applied
svSvp11	Verify that SmpSynch is set as follows: SmpSync = 2; global area time synchronization signal is present SmpSync = 1; local area time synchronization signal is present SmpSync = 0; no time synchronization signal is present
svSvp12	Verify that after restoring the power the DUT shall publish valid SV messages within specified time (PIXIT)
svSvp13	Verify that in SIMULATION mode the DUT publishes SV message with Simulation = TRUE (PIXIT)
svSvp14	Signals that are not measured or calculated shall have the corresponding Quality = Invalid

6.2.6.8.3 Send SV message subscribe test cases

The test cases listed in Table 70 and Table 71 shall apply.

Table 70 – Send SV message subscribe positive test cases

Test case	Test case description
svSvs1	Verify that the physical layer and connector matches the SCSM and PIXIT
svSvs2	Send SV messages from one or more sources with new data and check if the DUT processes the messages (PIXIT)
svSvs3	Send SV messages with SmpSynch = 0, 1 and 2 and check if DUT processes the messages according to the PIXIT
svSvs4	Verify that after restoring the power the DUT is subscribing valid SV messages within specified time (PIXIT)
svSvs5	Verify the behaviour of the DUT when the Simulation is set in the SV messages (PIXIT)
svSvs6	Verify the behaviour of the DUT when the Quality-Test is set in the sampled data of SV messages (PIXIT)
svSvs7	Verify the behaviour of the DUT when the Quality-Invalid is set in the sampled data of SV messages (PIXIT)

Table 71 – Send SV message subscribe negative test cases

Test case	Test case description
svSvsN1	Check behaviour of DUT as specified in PIXIT on – Missing some SV messages – Missing all SV messages – Double SV message – Delayed SV message – Out of order SV message
svSvsN2	Check behaviour of DUT when the SV message SvID, ConfRev, SmpRate, DatSet mismatches (PIXIT)
svSvsN3	Check behaviour of DUT when the SV data set configuration mismatches: too many elements, not enough elements, element out of order, or element with wrong type (PIXIT)
svSvsN4	Verify that the DUT behaves a specified in the PIXIT when the SV message SmpSynch is set to 1 or 2 and restored to 0 again

6.2.7 Acceptance criteria

Evaluation criteria for testing the device under test (DUT) include:

– specific design characteristics to be validated;

– checkpoints identified for anomalous conditions.

There are three possibilities for a test result according to the ISO/IEC 9646 series:

- Pass (verdict) – A test verdict given when the observed test outcome gives evidence of conformance to the conformance requirement(s) on which the test purpose of the test case is focused, and when no invalid test event has been detected.
- Fail (verdict) – A test verdict given when the observed test outcome either demonstrates non-conformance with respect to (at least one of) the conformance requirement(s) on which the test purpose of the test case is focused, or contains at least one invalid test event, with respect to the relevant specification(s).
- Inconclusive (verdict) – A test verdict given when the observed test outcome is such that neither pass nor fail verdict can be given. Such a result shall be always resolved to find out if this behaviour results from the standard, from the implementation or from the test procedure.

In general, a test case is passed when the DUT behaves as specified in the IEC 61850 series and the PIXIT, the test cases are failed when the DUT behaves different as specified in the IEC 61850 series and PIXIT. When not specified in the IEC 61850 series and in the PIXIT, the DUT shall keep on responding to syntactically correct messages and shall ignore syntactically incorrect messages.

7 Tool related conformance testing

7.1 General guidelines

7.1.1 Test methodology

IEC 61850 tool testing needs at least two tools exchanging SCL files with each other. Comprehensive interoperability testing of all possible tools with all possible devices and system configurations is not feasible. Therefore, the test concept shall include test devices, test configurations, and test scenarios. The behaviour should be tested properly by using well-defined test cases. In addition, the mandatory tests the test selection and judgement depend on the SICS provided together with the tool to be tested.

NOTE SCL files are generated to test the data exchange and engineering capabilities.

7.1.2 Test system architecture

In order to be able to perform a tool test, a minimum system configuration triggering all engineering capabilities as system test set-up is necessary (see Figure 7). For configurator tool tests, no online system is needed at all. TUT1 (tool under test) is the IED configurator tool to be tested by using a system configurator simulator, TUT2 is the System configurator to be tested by using an IED tool simulator. The configurator tools import and export SCL formatted files with file extension sed, scd, icd and iid specified in IEC 61850-6.

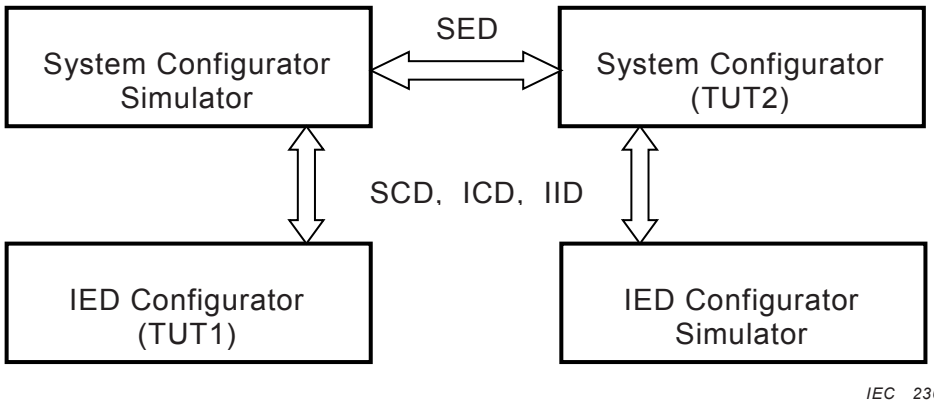

Figure 7 – Test system architecture to test a configurator tool

7.2 Conformance test procedures

7.2.1 General

This subclause describes the test procedure requirements, test structure, the test cases (what is to be tested) and the format. A few examples of test procedures (how to be tested) are given in Annex A.

7.2.2 Test procedure requirements

The test procedure requirements are the same as for device test procedures.

7.2.3 Test structure

The test cases are structured as follows:

– IED configurator (tool) tests (related to IEC 61850-6, Table G.1 SICS)

– System configurator tests (related to IEC 61850-6, Table G.2 SICS)

7.2.4 Test cases to test an IED configurator tool

7.2.4.1 General

This part of the IEC 61850 series specifies abstract test cases for IED tools. The abstract test cases shall be used for the definition of test procedures to run in tests.

NOTE The concrete syntax of test cases depends on the test system environment, i.e., mainly on the test script language. The concrete test cases are provided by test facilities agreed upon by the market participants.

7.2.4.2 ICD export test procedure overview

The test cases listed in Table 72 and Table 73 shall apply. Observe that most are only relevant if SICS I12 is claimed.

Table 72 – ICD test cases

Test case	Test case description
tIce1	Check if the major/minor software version in the tool documentation and the SICS do match with the tool (IEC 61850-4)
tIce2	Test if the ICD configuration file conforms to the SCL schema (IEC 61850-6)
tIce3	Check that the data model name space is stated in the ICD file (LLN0.NamPlt.ldNs; I13)
tIce4	Check that any predefined / fixed configuration values are within the ICD file (I14)
tIce5	Check that the supported SCL versions are stated in the SICS (I15, I16)
tIce6	Check if the ICD file contains a communication section with the default address (only for SICS I110).
tIce7	Check that the ICD file is UTF-8 coded

In case that SICS I12 is supported, the tests of Table 73 need to be performed additionally, and Table 72 tests need to be repeated with the generated ICD file from tIce8.

Table 73 – ICD export test cases

Test case	Test case description
tIce8	Modify the IED pre-configuration with the IED tool, and generate an ICD file. Perform tests tIce1 to tIce7 on the generated file.
tIce9	Check the communication and engineering capability (Services) section of the generated ICD file against the supplied ICD file (should be identical, if not changed at IED engineering)
tIce10	Check that the generated ICD file contains the correct valKind values (I111)
tIce11	Check that IED internal addresses for pre-engineered input signals appear in the ICD Input section (if I112 is claimed)
tIce12	Check that exported IED internal addresses in the Input section have the expected Service type (if I113 is claimed)

7.2.4.3 SCD import test cases

The test cases listed in Table 74 shall apply. Prerequisite is that an SCD file is produced with a validated system tool, which contains a readily engineered IED from the ICD file supplied with or produced by the IED tool. The SCD file shall use all communication and engineering capabilities of the IED / IED tool specified in the ICD file concerning the tested IED / IED tool. The SCD file shall have UTF-8 format.

Table 74 – SCD Import test cases

Test case	Test case description
tSci1	Import the SCD file into the IED tool (I214). Select the IED to be handled from the IEDs named in the SCD file by IED name (I21).
tSci2	Complete the signal engineering for incoming signals from other IEDs (I42). Verify that this is based either on I213/I43 or I29 (or both) as specified in the SICS.
tSci3	Test that the configuration values are correctly loaded (I210) and the *valKind* restrictions on reading / writing configuration data does work as specified (I211)
tSci4	An LD name change in the IED tool is possible (I45)
tSci5	An LD name specified in the SCD file is used by the IED (I212)

7.2.4.4 Tool functionality test cases

The test cases listed in Table 75 shall apply.

Table 75 – IED configurator data model test cases

Test case	Test case description
tTf1	For an edition 2 tool (SCL 2007) create an SCL edition 1 (2003) SCD, and import into the tool. The import shall work, ignoring all features the tool does not understand (I41)
tTf2	Generate a CID file (if I44 is supported). Check the CID file on SCL schema conformance.
tTf3	Modify some LN prefix / instance number in the SCD file, reconfigure the IED and load onto the IED. Browse the data model and check that changes are in, check that the IED functionality behind still works correctly (if I46 is supported).

7.2.4.5 IID export test cases

The test cases listed in Table 76 and Table 77 shall apply.

Table 76 – IID export test cases

Test case	Test case description
tIie1	Modify IED data model (add LN or add data objects, remove unused data objects / LNs). Export an IID file. Check the file on SCL conformance and the performed model changes (if I35 is claimed)
tIie2	Modify IED data object values (either configuration values I32, or setting parameters I33). Export an IID file. Check the file on SCL conformance, and that the changed values are in.
tIie3	Verify that the IID file header information (versioning) is as required (I34).
tIie4	Export of model changes from an existing system. Add a LN instance and a data object instance to an existing LN instance, and remove an unused non-mandatory data object from an LN instance (whatever the IED tool supports). Check that the IID file contains the modifications (I35) and the data model version (LLN0.NamPlt.configRev) is modified.

Table 77 – Negative IID export test case

Test case	Test case description
tIieN1	Try to remove data objects / LNs which are contained in a data set allocated to a control block allocated to a client. This shall not be allowed / possible (I35).
tIieN2	Try to remove a data set allocated to a control block allocated to a client. This shall not be allowed / possible (I35).

7.2.5 Test cases to test a system configurator tool

7.2.5.1 General

This part of the IEC 61850 series specifies abstract test cases for system configurator tools. The abstract test cases shall be used for the definition of test procedures to run in tests.

NOTE The concrete syntax of test cases depends on the test system environment, i.e., mainly on the test script language. The concrete test cases are provided by test facilities agreed upon by the market participants.

7.2.5.2 Documentation and version control test case

The test cases listed in Table 78 shall apply.

Table 78 – System configurator documentation test case

Test case	Test case description
tDoc1	Check if the major/minor software version in the tool documentation and the SICS do match with that of the tool (IEC 61850-4)

7.2.5.3 IED file import test cases

The test cases listed in Table 79 and Table 80 shall apply. The input ICD respective IID shall be in UTF-8 format. In case that support of other formats is claimed in the SICS, an appropriate ICD file in this other format shall also be imported.

Table 79 – ICD / IID import test cases

Test case	Test case description
tSie1	Import ICD file in supported file format (UTF-8 at least) (S111)
tSie2	Verify that predefined data sets and control blocks are imported (S12, S13), i.e. visible in the tool or at least in the later exported SCD file.
tSie3	For an edition 2 tool (2007) import another ICD file from Edition 1 (2003). Check that all understandable parts according to the version are imported and accessible (S14, S15)
tSie4	Import an ICD file with LNode links and coordinates according to IEC 61850-6 Annex C.1. Instantiate the bay template as bay and the IED template as IED. Check that all bay elements and LNode connections are imported (if S16 is claimed) and (if S19 is claimed) the coordinates are also imported. If coordinates are not visible in the tool, export an SCD file and check that coordinates are kept.
tSie5	Import the same ICD a second time, instantiate for another IED. Ensure that the already imported Data type templates are reused and not doubled (S17)
tSie6	Provide an ICD file with private XML elements and attributes and import it. Check the exported SCD file that these elements are still there (if S18 is claimed)
tSie7	Export a SCD file to the IED tool. Provide an IID file for one IED with changes in configuration values, setting values, added LN instances, removed LN instances or data objects (not referenced in data sets). Import this IID file. Check that the imported changes are reflected in the tool (S110)
tSie8	Export a SCD file to the IED tool. Provide an IID file for one IED with removed control blocks and some changed values (Configuration, settings). Import this IID file. Check that the removed control blocks are still in the system tool project, and the modified values are updated (S110)
tSie9	For an edition 2 tool (2007) import another extended ICD file. Check that all understandable parts according to the version are imported and accessible (S14, S15)

Table 80 – ICD / IID negative test case

Test case	Test case description
tSieN1	Export a SCD file to the IED tool. Provide an IID file for one IED with removed LN instances or data objects referenced in data sets. Import this IID file. Check that the import is refused, or at least the removed objects are still in the system tool (S110)

7.2.5.4 Communication engineering test cases

The test cases listed in Table 81 and Table 82 shall apply. Verify all results inside an exported SCD file.

Table 81 – Communication engineering test cases

Test case	Test case description
tSce1	Import an ICD file, and give the instance an IED name (S21)
tSce2	Create a SubNetwork with type 8-MMS (IEC 61850), and connect the IED to this SubNetwork with some IP address (S22)
tSce3	Import an ICD file of a client IED, connect it with IP address to the SubNetwork (S23)
tSce4	Import an ICD file for a master clock, and connect with IP address to the SubNetwork (S23)
tSce5	Configure physical connections between the first IED, the client IED and the master clock (S24)
tSce6	For an IED capable of having a LD name different to the concatenation of IED name and LD inst, configure the LD name differently (S25)
tSce7	For an IED allowing to configure this, modify the LN prefix and/or LN instance number (S26)

Table 82 – Communication engineering negative test case

Test case	Test case description
tSceN1	Try to modify LN prefix and LN instance number for an IED, which forbids this. Try to change the LD inst or set the LD name for an IED which does not allow this. All should be prohibited by the tool (S56).

7.2.5.5 Data flow engineering test cases

The test cases listed in Table 83 and Table 84 shall apply. They shall be performed on the project started with the Communication engineering tests, i.e. after these test steps have been performed. Verify all results inside an exported SCD file.

Table 83 – Data flow test cases

Test case	Test case description
tDfe1	Create a data set on an IED allowing this (S33)
tDfe2	Configure an existing control block with this data set, and appropriate reporting options (S31), using: report control, GOOSE control, Logging control, Sampled Value Control
tDfe3	Configure the data flow from this control block to the client / subscriber IED (S36) using: report control, GOOSE control, Logging control, Sampled Value Control
tDfe4	Create a new control block (if IED allows) and a new data set (if IED allows). Configure the control block with this data set and data flow to the same client as the previous one (S32, S34). Try for report control, GOOSE control, Logging control, Sampled Value Control.
tDfe5	Modify a data set allocated to a control block. Observe that the control block confRev is incremented (S34, S35) by the TUT for report control, GOOSE control, Logging control, Sampled Value Control.
tDfe6	Create an Input section at the client with two incoming data items from the source IEDs (S37)
tDfe7	Automatically create a client input section based on the data flow to this client (S38)
tDfe8	Provide the source control block reference for incoming signals at the Input section (S39). Might be automatically created, or might need manual creation.

Table 84 – Data flow negative test cases

Test case	Test case description
tDfeN1	Try to modify a preallocated data set for an IED, which forbids this. The TUT should prohibit this (S56).
tDfeN2	Try to create a control block for an IED which does not allow this. All should be prohibited by the tool (S56) for: Report control, GOOSE, Logging, Sampled Value. The TUT should prohibit this.

7.2.5.6 Substation section handling test cases

The test cases listed in Table 85 shall apply.

Table 85 – Substation section handling test cases

Test case	Test case description
tSsh1	Import the substation section from an SSD file to the configuration from the previous tests, or from an SCD file together with the IEDs. Verify that it is correctly represented (S41)
tSsh2	Add another bay to the Substation (S41, S42)
tSsh3	Allocate some LN instances to elements of the Substation section (e.g. a CSWI to a disconnector, a PTOC or MMXU to a bay) (S43)
tSsh4	Import a bay template or an IED with bay template, and instantiate this bay template as new bay in the substation (S44)
tSsh5	Connect the new bay electrically to the HV bus(ses) of the existing substation (S45)
tSsh6	Modify names and description of one imported bay (S46)
tSsh7	If no terminal exists, edit terminals to one primary equipment. Change the terminal name at the selected equipment (S47)
tSsh8	Create a Function / SubFunction hierarchy (e.g. protection / overcurrent below a bay element) and allocate some LN instances to this (S48)
tSsh9	Export an SCD file and check, that the final state is correctly contained in the SCD file. Further on, the SCD Header should contain a new / modified revision index (S58)

7.2.5.7 SCD modification test case

The test cases listed in Table 86 shall apply.

Table 86 – SCD modification test cases

Test case	Test case description
tSmo1	Assign basic information to the project header. Perform some modification in Substation section or data flow. Check that either a revision index is automatically set in the SCD Header section, or do this manually (S51, S58)
tSmo2	Set or change the values of some CF attributes, which allow this change (valKind=Set) (S52)
tSmo3	Set some setting values for SP parameters, and different values in different setting groups for SG parameters (S53)
tSmo4	Move a Substation object. Observe if coordinates in exported SCL change appropriately (S54)
tSmo5	Try to make the IED capabilities visible. Check if this corresponds to the ICD input (S55)
tSmo6	Take an attribute with valKind=Set, modify its value, and set valKind=RO (S57)

7.2.5.8 SCD export test cases

The test cases listed in Table 87 shall apply.

Table 87 – SCD export test cases

Test case	Test case description
tSse1	Export an SCD file either in 2003 or 2007 format and UTF-8 coding, as claimed. Check syntactical correctness (S61, S62). Repeat for any other version claimed (S63)
tSse2	Observe that all Private sections imported from ICD/IID files are again exported at the same places.
tSse3	Observe that even if the DataTypeTemplate section is restructured, the resulting LN / DO / DA instances for instantiated IEDs are identical (except possibly allowed renaming of prefix and LN instance number) to the ICD files (S65)
tSse4	Import another ICD file using the same type identifiers as already exist, but with different structure / contents. Observe that type renaming takes place, and the resulting IED related LN / DO / DA instances are identical to those of the ICD file (S66)
tSse5	Export SCD file in claimed codings different to UTF-8. Check that the logical content is identical to that of UTF-8 format (S67)

7.2.5.9 SCD import test cases

The test cases listed in Table 88 shall apply.

Table 88 – SCD import test cases

Test case	Test case description
tSsi1	Import an SCD file in 2003 syntax. Observe that all parts are correctly visible (S71)
tSsi2	Import an SCD file in 2007 syntax. Observe that all imported parts are correctly visible (S72), or at least the 2003 compatible parts are imported (S71)
tSsi3	Import an SCD file in claimed syntax. Observe that all parts are correctly visible (S73)
tSsi4	Create an SCD file with additional LN instance allocations to the Substation section. Import this to the previous project. Observe that the old LN instance associations are kept, and the new ones added (S75)
tSsi5	Create an SCD file, and modify attribute values (configuration values, settings). Import the SCD file. Observe that the values are updated in the project model (S77, S78).
tSsi6	Add new IEDs to the previous SCD file. Import this new SCD file. Observe that the new IEDs and their relation to the Substation section are added to the project model.
tSsi7	Export an SCD file. Check that all modifications imported via SCD or IID files are contained in it. Note that if this is performed inside each of above tests, no separate test is needed.
tSsi8	Create an SCD file with an additional bay linked to the existing bus bar. Import this file. Observe that the new bay inclusive bus bar links is added in the TUT to the existing project. (S74)

7.2.5.10 SED file handling test cases

The test cases listed in Table 89 shall apply.

Table 89 – SED file handling test cases

Test case	Test case description
tSeh1	Select one IED for export with data flow engineering right. Export an SED file. Check syntactical correctness, and that it contains the IED with dataflow right, and all IEDs sending to it with fixed engineering right (S81). Observe that the IED exported with data flow right is now set to 'fix' in the system tool.(S83)
tSeh2	Try to modify the data set of the IED exported with data flow right. This should be blocked by the tool (S83)
tSeh3	Add an IED to the exported SED file, and engineer some data flow from the exported IED to this new IED. Import the modified SED file. Observe that the new IED and the data flow definitions to it are imported, and that the exported IED now again has full engineering right (S82)
tSeh4	Import an SED file from another project. Add data flow to an 'own' IED with a data set modification at an imported IED with data flow right. Export an SED file with these modifications. Check the correct header ID setting, and that the 'own' IED is contained with 'fix' engineering right (S84)
tSeh5	Import an SED file with Substation section. Add any new substation elements, and add any new LN instance associations to the substation elements (S85)
tSeh6	Import the communication addresses existing in an SED file for the IEDs in the SED file, and overwrite or add to own existing address(es). Do not remove any address (S86).

7.2.6 Acceptance criteria

Evaluation criteria for testing the tool under test (TUT) include:

– specific design characteristics to be validated;

– checkpoints identified for anomalous conditions.

There are three possibilities for a test result according to the ISO/IEC 9646 series:

- Pass (verdict) – A test verdict given when the observed test outcome gives evidence of conformance to the conformance requirement(s) on which the test purpose of the test case is focused, and when no invalid test event has been detected.
- Fail (verdict) – A test verdict given when the observed test outcome either demonstrates non-conformance with respect to (at least one of) the conformance requirement(s) on which the test purpose of the test case is focused, or contains at least one invalid test event, with respect to the relevant specification(s).
- Inconclusive (verdict) – A test verdict given when the observed test outcome is such that neither pass nor fail verdict can be given. Such a result shall be always resolved to find out if this behaviour results from the standard, from the implementation or from the test procedure.

In general, a test case is passed when the TUT behaves as specified in the IEC 61850 series and the SICS and PIXIT, the test cases are failed when the TUT behaves different as specified in the IEC 61850 series, SICS and PIXIT.

8 Performance tests

8.1 General

IEC 61850-5 identifies several specific performance requirements for applications operating in the IEC 61850 series environment. This clause defines the metrics to be measured within devices such that documented product claims supporting those requirements can be compared across vendors.

The performance tests may require a base load generator. The definition of base load is outside this part of the standard. The use of priorities according to IEC 61850-8-1 and IEC 61850-9-2 mitigates the use of base load simulation for time critical information exchange such as for GOOSE and sampled value exchange.

IEDs requiring a very high time accuracy may use a directly connected external time source (radio or satellite clock).

8.2 Communications latency

8.2.1 Application domain

IEC 61850-5 defines application communications requirements in terms of “Transfer time” (IEC 61850-5 Subclause 13.4), the time required to deliver a process value from a sending physical device to the process logic of a receiving device. The transfer time is defined (Subclause 13.4 and Figure 16 of IEC 61850-5) in terms of three intervals:

– t_a: the time required for the sending device to transmit the process value;

– t_b: the time required for the network to deliver the message; and

– t_c: the time required for the receiving device to deliver the value to its process logic.

The interval t_b is determined by the network infrastructure and is not an attribute of the IED. From an IED testing point of view, only output and input latencies can be measured, t_a and t_c are estimated from the measured latencies.

measured output latency = estimated input processing time + estimated t_a

measured input latency = estimated output processing time + estimated t_b

The vendors of network components such as switches shall define and document the amount of the latency time that is due to estimated processing time for all priorities supported by the network components.

The estimated input processing time of an IED is the time required for input signal conditioning (e.g., debouncing, sampling, etc.).

The estimated output processing time of an IED is the time required for output signal activation (e.g., contact delays, I/O scan rate, etc.).

The performance metrics to be measured in the IEDs depend on which of the IEC 61850 series services are used to deliver the process values. The standard defines four basic mechanisms: GOOSE, SV, Reporting, and Controls. When tested from a black box perspective, each of these mechanisms yields two possible metrics that can be tested.

The measured output (input) latency shall be less than or equal to 40 % of the total transmission time defined for the corresponding message type in IEC 61850-5 Subclause 13.7.

The value of 40 % on each end of the connection leaves over 20 % for network latencies. This maximum time applies mainly to the message types 1 (Fast messages) and 4 (Raw data messages); these messages make use of the priority mechanisms of the networks components defined in IEC 61850-8-1 and IEC 61850-9-2. Messages of type 2 may be assigned to a high priority.

NOTE The values for the total transmission times are not repeated for consistency reasons.

The tests may require a base load generator. The definition of base load is beyond the scope of this part of IEC 61850. The use of priorities according to IEC 61850-8-1 and IEC 61850-9-2 mitigates the use of base load simulation for time critical information exchange like GOOSE, SV, Reporting, and Controls.

8.2.2 Methodology

The following time interval measurements shall be made between a physical input (or message) change and the appearance of a message on the output media (or physical output):

– GOOSE output latency;

– sampled value output latency;
– report output latency;
– control output latency.

A test system (see Figure 8) shall measure an output latency time by generating a sequence of physical input triggers to the IED and measuring the time delay to the corresponding message generated by the IED. The worst case, mean latency time and the standard deviation shall be computed across the responses to 1 000 input triggers. The vendor shall define and document the amount of the latency time which is due to estimated output processing time.

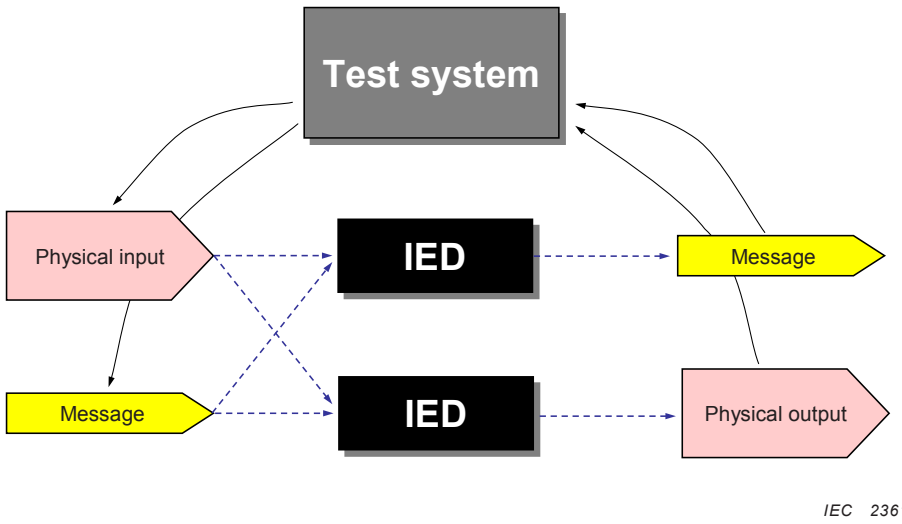

Figure 8 – Performance testing (black box principle)

The test results to be documented for each latency shall be the measured values and the two corresponding estimated values. The measured values shall be the worst case and mean values and the standard deviation of the latency time computed across 1 000 tests.

8.2.3 GOOSE performance test

8.2.3.1 General

The scope of the GOOSE performance test is to benchmark the GOOSE performance against the performance classes as defined in IEC 61850-5. Clause 13 of IEC 61850-5 states that the Type 1A messages are the most demanding messages with the shortest transmission times:

– For Performance Class P1, the total transmission time shall be in the order of half a cycle. Therefore, 10 ms is defined.
– For Performance Class P2/P3, the total transmission time shall be below the order of a quarter of a cycle. Therefore, 3 ms is defined.

To measure the transmission time as defined in IEC 61850-5 is not possible without special access to the internal data of the device. To enable "black-box" testing we need a different test methodology further referred to as the "GOOSE ping-pong" method. This method is already in use for GOOSE server device conformance testing.

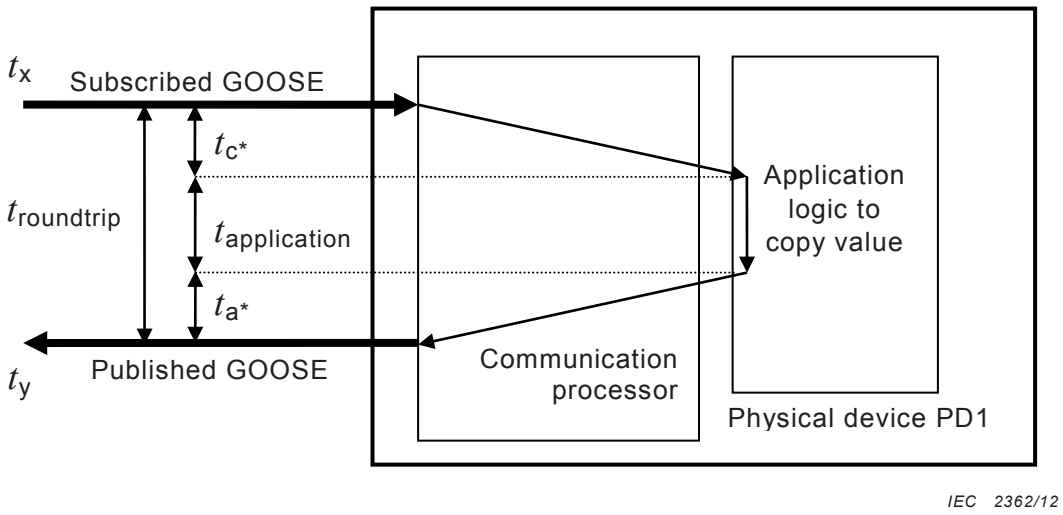

Figure 9 – Measure round trip time using GOOSE ping-pong method

The GOOSE ping-pong method focuses on the round trip time as defined in Figure 5. The round trip time is the time interval between the arrival of a subscribed GOOSE message and the departure of the published GOOSE message. A protocol analyzer shall be used to timestamp the GOOSE messages and archive the performance test results.

The relation between the transfer time and roundtrip time is as follows:

- $t_{transfer} = t_a + t_b + t_c$
- $t_{roundtrip} = (t_y - t_x) = t_{c*} + t_{application} + t_{a*}$

When the IEDs are the same we assume that the GOOSE publish and subscribe communication processing times are the same. In that case we can combine these equations into:

- $t_{transfer} = t_{roundtrip} - t_{application} + t_b$

For a single Ethernet switch as used during the test, the network delay will be minimal (< 0,1 ms). Then we get

- $t_{transfer} = t_{roundtrip} - t_{application}$

t_a = GOOSE publish communication processing

t_b = network delay of one GOOSE message

t_c = GOOSE subscribe communication processing

$t_{application}$ = application logic time

The application time typically is the sum of the scan cycle delay and the actual application logic processing time. On a scan cycle of for example 4 ms the average scan cycle delay is about 2 ms (50 % of scan cycle). The difference between the maximum and the minimum of the measured roundtrip times will be close to the scan cycle. These metrics can be used to perform a plausibility check on the documented figures in the device PIXIT document.

NOTE The scan cycle is defined as the inverse of the number of input scans per second. For example if an input is scanned 100 times per second the scan cycle is 10 ms.

The following items may have an impact on the GOOSE performance:

– size of the published/subscribed GOOSE message (number of data set elements);
– type of data set elements;
– use of Functionally Constrained Data (FCD) or Functional Constrained Data Attributes (FCDA) in the dataset;
– number of subscribed GOOSE messages;
– time correlation of subscribed GOOSE messages state changes;
– number of background GOOSE messages on the network;
– other communication tasks like MMS reporting, file transfer and/or Sampled Values when supported.

This test method is intended as a benchmark for comparing relative performance of different IEDs. It defines standardized tests aimed at mimicking typical workload conditions. It does not test device performance under worst case load, worst case network conditions, or in a specific system application. Please refer to detailed vendor specifications for full description of the device capabilities, behaviour and limitations.

8.2.3.2 Message definitions

To compare the test results the messages during the test shall be as similar as possible. The general message requirements are:

– each GOOSE has unique address, same priority, Test=false, ConfRev=1, NdsCom=false;
– the GOOSE datasets contain functionally constrained data attributes (FCDA);
– the BRCB or URCB datasets contain functionally constrained data (FCD).

The normal "Published GOOSE used for ping-pong" has 4 boolean and 4 quality data values, the large "Published GOOSE used for ping-pong" has 20 double point, 20 boolean and 40 quality data values. In case a device has less then 20 double points available it can publish large GOOSE messages with 5 double point, 35 boolean and 40 quality data values.

The normal "Subscribed GOOSE used for ping-pong" has 4 boolean and 4 quality data values, the large "Subscribed GOOSE used for ping-pong" has 20 double point, 20 boolean and 40 quality data values.

The "Time correlated Subscribed GOOSE not used for ping-pong" has 20 double point, 20 boolean and 40 quality data values.

The background load GOOSE messages have 20 double point, 20 boolean and 40 quality data values. The background load shall be at least 300 GOOSE messages per second with a state change about every 10 ms.

The GOOSE simulator(s) shall be able to send all the subscribed, not subscribed and background load GOOSE messages and send the time-correlated GOOSE messages within 0,2 ms accuracy.

In case the DUT supports reporting, one client shall be connected to the DUT during all test cases. The client enables two BRCBs or when buffered reporting is not supported two URCBs with same data values (as FCD) as the normal and large datasets in the published GOOSE. The report control blocks shall be configured to send reports on data change and integrity 1 second with all supported optional fields.

8.2.3.3 Test cases for GOOSE performance

The test cases listed in Table 90 shall apply.

Table 90 – GOOSE performance test cases

Test case	Subscribe (ping)	Publish (pong)	Time correlated subscribed GOOSE state changes	Background load
Gpf1	Normal	Normal	No	No
Gpf2	LARGE	LARGE	No	No
Gpf3	Normal	Normal	YES	No
Gpf4	LARGE	LARGE	YES	No
Gpf5	Normal	Normal	No	YES
Gpf6	LARGE	LARGE	No	YES
Gpf7	Normal	Normal	YES	YES
Gpf8	LARGE	LARGE	YES	YES

For performance class P1 the transmission limit is defined as 10 ms and 3 ms for P2/P3. The performance results are the average and standard deviation over 1 000 input triggers and the sum of the measured output and input latency shall be less than or equal to 80 % of the total transmission (because 20 % is reserved for network latency).

We already determined: $t_{transfer} = t_{roundtrip} - t_{application.}$ The application time typically is the sum of the internal scan cycle wait time and the actual logic processing time. To represent the worst case transfer time, we set the actual logic processing time to zero (this means that the logic processing time is considered as part of the transfer time). As a result we get:

- Average application time = 50 % of scan cycle
- Maximum application time = 100 % of scan cycle
- Minimum application time = 0 % of scan cycle

Now the transfer times can be calculated as follows:

- Average: $t_{transfer.avg} = t_{roundtrip.avg} - t_{application.avg} = t_{roundtrip.avg}$ – scan cycle/2
- Maximum: $t_{transfer.max} = t_{roundtrip.max} - t_{application.max} = t_{roundtrip.max}$ – scan cycle
- Minimum: $t_{transfer.min} = t_{roundtrip.min} - t_{application.min} = t_{roundtrip.min}$

NOTE It is possible that the calculated maximum transfer time is less than the calculated minimum transfer time.

Plausibility checks:

- Documented scan cycle ≥ Measured scan cycle = $t_{roundtrip.max} - t_{roundtrip.min}$
- Documented scan cycle ≥ Measured standard deviation * $\sqrt{12}$ (for uniform distribution[1])

In case the measured scan cycle is more than the documented scan cycle, the documented scan cycle shall be adjusted. In case the DUT has an event driven method (no scan cycle), the scan cycle for the calculations is set to 0,0 ms.

To pass the performance test the criteria are:

- Gpf1 to Gpf6 test are passed when the calculated average, maximum and minimum transfer times are less than 80 % of the applicable performance class limit (see 8.2.1 Note 1):
  - Performance class P1; $t_{transfer} < 8{,}0$ ms

1 http://en.wikipedia.org/wiki/Uniform_distribution_(continuous)

  - Performance class P2/P3; $t_{transfer}$ < 2,4 ms

- Gpf7 and Gpf8 test are passed when the calculated average, maximum and minimum transfer times are less than 100 % of the performance class limit:
  - Performance class P1; $t_{transfer}$ <10,0 ms
  - Performance class P2/P3; $t_{transfer}$ < 3,0 ms

The PIXIT document shall specify the GOOSE performance class and scan cycle(s).

8.3 Time synchronisation and accuracy

8.3.1 Application domain

The scope of this test is to verify the ability of the IED to communicate time stamp information about an instrumented event. An accurate time stamp relies on several separate functions including clock accurately decoding the received signal, accurate synchronisation of IED clock to the received signal, timely IED detection of change of state and accurate use of IED clock value to time stamp data.

Time synchronisation is used for the synchronisation of the IED clock values when no direct external time source (for example PPS or GPS) is available to the IED. During synchronisation across the power utility LAN, one IED with a precision time source acts as the time master. The time source of the time master IED is typically provided by an external source.

The time accuracy metrics defined in this subclause represent measures of time stamp accuracy for the IED when an external source is provided or when the IED relies on the time synchronisation mechanism with a time master respectively.

This test is essential due to the nature of networked IEDs being used to design systems of interoperable devices working in a coordinated fashion. These, and other device performance measures, are essential information for predicting performance, functionality and reliability of designs executed by networked IEDs. No specific performance benchmarks are expected to be met, however, verification and publication of actual performance measures is necessary to be conformant. Using these published performance measures, system integrators can predict the performance of the interconnected IEDs and thus the performance of system. Furthermore, system integrators will be able to identify suitable devices for specific applications. Performance measures will be made on the device under test connected to a network with pre-defined configuration and traffic. It is understood that if the network traffic changes, the system performance may change. It is also understood that if the processing load on the device changes, the device performance may change.

8.3.2 Methodology

8.3.2.1 General

The time synchronisation test requires a test system (see Figure 10) consisting of a data change generator function and a time master function, each connected to a common external clock source (e.g. radio or satellite clock). The change generator function triggers physical events within the IED, with accurate times recorded for each event. A test system analyser function retrieves the time stamp of each event from the IED and compares it with the recorded time of the event generation.

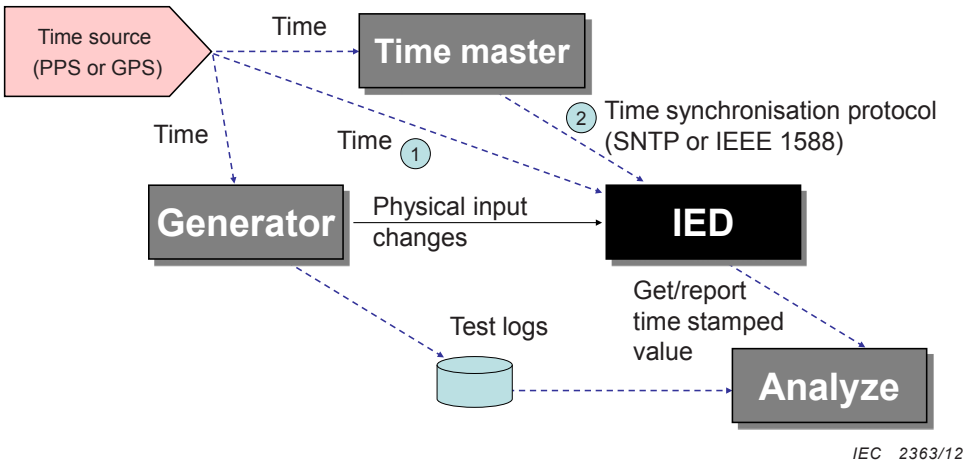

Figure 10 – Time synchronisation and accuracy test setup

8.3.2.2 Time from external source

The first accuracy measurement is made with the IED directly receiving time from the same external source (for example PPS or GPS) used by the test system (1). After the time synchronisation is completed, a sequence of 1 000 change events shall be generated, and the mean and standard deviation from the mean is computed over the differences between the event times and the retrieved time stamps.

8.3.2.3 Time from time synchronisation protocol

The second accuracy measurement (2) is made with the IED using the time synchronisation protocol (for example SNTP or IEEE 1588) with the time master function in the test system. After the time synchronisation is completed, a sequence of 1 000 change events shall be generated, and the mean and standard deviation from the mean is computed over the differences between the event times and the retrieved time stamps. This difference is the overall time stamping accuracy consisting of clock setting accuracy, clock drift between resynchronisations, and I/O scan cycle.

The event sequence generation shall be coordinated with the time synchronisation protocol. The event sequence shall begin just after the IED requests synchronisation with the Time Master function. If synchronisation is requested during the sequence, the sequence is interrupted while the synchronisation protocol exchange is completed.

8.3.3 Testing criteria

Time synchronisation accuracy shall be tested relative to UTC (as provided by the time reference used by the test generator). IEDs shall be tested for the class of accuracy (according to IEC 61850-5) for which they are rated.

NOTE 1 The jitter caused by network components like switches is assumed to be negligible.

The vendors of network components like switches shall define and document the amount of the latency time that is due to estimated processing time for all priorities supported by the network components. The time synchronisation solution in the IED shall estimate and compensate the message delay in the network within the specified accuracy (PICS).

The vendors of IEDs shall define and document the time drift of the IED's internal clock.

NOTE 2 The drift is independent of the time synchronization.

8.3.4 Performance

Values of accuracy and allowable error are documented in IEC 61850-5, Clause 13. These figures can be matched only if both the time synchronisation and the tagging mechanism within the IEDs support these requirements. The IED clock shall be accurate to a higher resolution than the performance class in order to receive and synchronise to a source.

9 Additional tests

The quality assurance requirements contained in IEC 61850-4, Clause 7 comprise several tests that are beyond the scope of this part of IEC 61850. Especially details on the system related test, type test, routine test, factory acceptance test, and site acceptance test shall be defined in specifications other than this part of the IEC 61850 series.

Annex A
(informative)

Examples of test procedure template

A.1 Example 1

sBr1	GetLogicalNodeDirectory(BRCB) and GetBRCBValues	☐ Passed ☐ Failed ☐ Inconclusive
Ref. Part, Clause and Subclause of IEC 61850 IEC 61850-7-2, Subclause 10.2.2 and 17.2.3.3 IEC 61850-8-1, Subclause 12.3.1 and 17.2.2		
Expected result 1) DUT sends GetLogicalNodeDirectory(BRCB) Response+ 2) DUT sends GetBRCBValues Response+		
Test description 1) For each logical node Client requests GetLogicalNodeDirectory(BRCB) 2) For each BRCB Client requests GetBRCBValues()		
Comment		

A.2 Example 2

sRp1	GetLogicalNodeDirectory(URCB) and GetURCBValues	☐ Passed ☐ Failed ☐ Inconclusive
Ref. Part, Clause and Subclause of IEC 61850 IEC 61850-7-2 Subclause 10.2.2 and 17.2.5.3 IEC 61850-8-1 Subclause 12.3.1 and 17.2.4		
Expected result 1) DUT sends GetLogicalNodeDirectory(URCB) Response+ 2) DUT sends GetURCBValues Response+		
Test description 1) For each logical node Client requests GetLogicalNodeDirectory(URCB) 2) For each BRCB Client requests GetURCBValues()		
Comment		

Bibliography

BRAND K.P. et al., Conformance Testing Guidelines for Communication in Substations, Cigré Report 34-01 – Ref. No. 180, August 2002

SCHIMMEL R., Conformance Test Procedures for Server Devices with IEC 61850-8-1 interface, Revision 2.3, UCA international users group, October 2007

SCHIMMEL R. and FLOHIL M., Conformance Test Procedures for Client Systems with IEC 61850-8-1 interface, Revision 1.1, UCA international users group, October 2009

SCHIMMEL R., Test procedures for Sampled Values Publishers according to the "Implementation Guideline for Digital Interface to Instrument Transformers using IEC 61850-9-2", Revision 1.1, UCA international users group, January 2010

SCHIMMEL R. and GERSPACH S., Test procedures for GOOSE performance according to IEC 61850-5 and IEC 61850-10, Revision 1.1, UCA international users group, January 2011

This page deliberately left blank

British Standards Institution (BSI)

BSI is the national body responsible for preparing British Standards and other standards-related publications, information and services.

BSI is incorporated by Royal Charter. British Standards and other standardization products are published by BSI Standards Limited.

About us

We bring together business, industry, government, consumers, innovators and others to shape their combined experience and expertise into standards -based solutions.

The knowledge embodied in our standards has been carefully assembled in a dependable format and refined through our open consultation process. Organizations of all sizes and across all sectors choose standards to help them achieve their goals.

Information on standards

We can provide you with the knowledge that your organization needs to succeed. Find out more about British Standards by visiting our website at bsigroup.com/standards or contacting our Customer Services team or Knowledge Centre.

Buying standards

You can buy and download PDF versions of BSI publications, including British and adopted European and international standards, through our website at bsigroup.com/shop, where hard copies can also be purchased.

If you need international and foreign standards from other Standards Development Organizations, hard copies can be ordered from our Customer Services team.

Subscriptions

Our range of subscription services are designed to make using standards easier for you. For further information on our subscription products go to bsigroup.com/subscriptions.

With **British Standards Online (BSOL)** you'll have instant access to over 55,000 British and adopted European and international standards from your desktop. It's available 24/7 and is refreshed daily so you'll always be up to date.

You can keep in touch with standards developments and receive substantial discounts on the purchase price of standards, both in single copy and subscription format, by becoming a **BSI Subscribing Member**.

PLUS is an updating service exclusive to BSI Subscribing Members. You will automatically receive the latest hard copy of your standards when they're revised or replaced.

To find out more about becoming a BSI Subscribing Member and the benefits of membership, please visit bsigroup.com/shop.

With a **Multi-User Network Licence (MUNL)** you are able to host standards publications on your intranet. Licences can cover as few or as many users as you wish. With updates supplied as soon as they're available, you can be sure your documentation is current. For further information, email bsmusales@bsigroup.com.

BSI Group Headquarters

389 Chiswick High Road London W4 4AL UK

Revisions

Our British Standards and other publications are updated by amendment or revision.

We continually improve the quality of our products and services to benefit your business. If you find an inaccuracy or ambiguity within a British Standard or other BSI publication please inform the Knowledge Centre.

Copyright

Useful Contacts:

Customer Services
Tel: +44 845 086 9001
Email (orders): orders@bsigroup.com
Email (enquiries): cservices@bsigroup.com

Subscriptions
Tel: +44 845 086 9001
Email: subscriptions@bsigroup.com

Knowledge Centre
Tel: +44 20 8996 7004
Email: knowledgecentre@bsigroup.com

Copyright & Licensing
Tel: +44 20 8996 7070
Email: copyright@bsigroup.com

bsi.

...making excellence a habit.™