BS EN 61850-7-1:2011

Communication networks and systems for power utility automation

Part 7-1: Basic communication structure — Principles and models

National foreword

This British Standard is the UK implementation of EN 61850-7-1:2011. It is identical to IEC 61850-7-1:2011. It supersedes BS EN 61850-7-1:2003 which is withdrawn.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.

© BSI 2011

ISBN 978 0 580 63211 2

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 30 November 2011.

Amendments issued since publication

Amd. No.	Date	Text affected

EUROPEAN STANDARD

NORME EUROPÉENNE

EUROPÄISCHE NORM

EN 61850-7-1

October 2011

ICS 33.200

Supersedes EN 61850-7-1:2003

English version

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

Réseaux et systèmes de communication pour l'automatisation des systèmes électriques -
Partie 7-1: Structure de communication de base -
Principes et modèles
(CEI 61850-7-1:2011)

Kommunikationsnetze und -systeme für die Automatisierung in der elektrischen Energieversorgung -
Teil 7-1: Grundlegende Kommunikationsstruktur -
Grundsätze und Modelle
(IEC 61850-7-1:2011)

This European Standard was approved by CENELEC on 2011-08-19. CENELEC members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration.

Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the CEN-CENELEC Management Centre or to any CENELEC member.

This European Standard exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CENELEC member into its own language and notified to the CEN-CENELEC Management Centre has the same status as the official versions.

CENELEC members are the national electrotechnical committees of Austria, Belgium, Bulgaria, Croatia, Cyprus, the Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, the Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland and the United Kingdom.

CENELEC

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

Management Centre: Avenue Marnix 17, B - 1000 Brussels

© 2011 CENELEC - All rights of exploitation in any form and by any means reserved worldwide for CENELEC members.

Foreword

The text of document 57/1121/FDIS, future edition 2 of IEC 61850-7-1, 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 61850-7-1:2011.

This document supersedes EN 61850-7-1:2003.

Compared to EN 61850-7-1:2003, EN 61850-7-1:2011 introduces:

- the model for statistical and historical statistical data,
- the concepts of proxies, gateways, LD hierarchy and LN inputs,
- the model for time synchronisation,
- the concepts behind different testing facilities,
- the extended logging function.

EN 61850-7-1:2011 also clarifies the following points:

- the use of numbers for data extension,
- the use of name spaces,
- the mode and behaviour of a logical node,
- the use of range and deadbanded values,
- the access to control actions and others.

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)	2012-05-19
– latest date by which the national standards conflicting with the document have to be withdrawn	(dow)	2014-08-19

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 61850-7-1:2011 was approved by CENELEC as a European Standard without any modification.

In the official version, for Bibliography, the following notes have to be added for the standards indicated:

IEC 61346-1	NOTE Harmonized as EN 61346-1.
IEC 61346-2	NOTE Harmonized as EN 61346-2.
IEC 61400 series	NOTE Harmonized in EN 61400 series (not modified).

Annex ZA
(normative)

Normative references to international publications with their corresponding European publications

The following referenced documents are indispensable for the application of this document. 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 in substations - Part 3: General requirements	EN 61850-3	-
IEC 61850-4	-	Communication networks and systems for power utility automation - Part 4: System and project management	EN 61850-4	-
IEC 61850-5	-	Communication networks and systems in substations - Part 5: Communication requirements for functions and device models	EN 61850-5	-
IEC 61850-6	-	Communication networks and systems for power utility automation - Part 6: Configuration description language for communication in electrical substations related to IEDs	EN 61850-6	-
IEC 61850-7-2	-	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	-
IEC 61850-7-3	-	Communication networks and systems for power utility automation - Part 7-3: Basic communication structure - Common data classes	EN 61850-7-3	-
IEC 61850-7-4	-	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	-
IEC 61850-8-1	-	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	-

Publication	Year	Title	EN/HD	Year
IEC 61850-9-2	-	Communication networks and systems in substations - Part 9-2: Specific Communication Service Mapping (SCSM) - Sampled values over ISO/IEC 8802-3	EN 61850-9-2	-
IEC 61850-10	-	Communication networks and systems in substations - Part 10: Conformance testing	EN 61850-10	-
ISO/IEC 8802-3	-	Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications	-	-
ISO/IEC 8825	Series	Information technology - ASN.1 encoding rules	-	-
ISO 9506-1	-	Industrial automation systems - Manufacturing Message Specification - Part 1: Service definition	-	-
ISO 9506-2	-	Industrial automation systems - Manufacturing Message Specification - Part 2: Protocol specification	-	-

CONTENTS

INTRODUCTION 10
1 Scope 11
2 Normative references 12
3 Terms and definitions 13
4 Abbreviated terms 13
5 Overview of the IEC 61850 series concepts 14
5.1 Objective 14
5.2 Topology and communication functions of substation automation systems 16
5.3 The information models of substation automation systems 16
5.4 Applications modelled by logical nodes defined in IEC 61850-7-4 18
5.5 The semantic is attached to data 21
5.6 The services to exchange information 23
5.7 Services mapped to concrete communication protocols 24
5.8 The configuration of the automation system 25
5.9 Summary 26
6 Modelling approach of the IEC 61850 series 27
6.1 Decomposition of application functions and information 27
6.2 Creating information models by stepwise composition 28
6.3 Example of an IED composition 31
6.4 Information exchange models 31
6.4.1 General 31
6.4.2 Output model 33
6.4.3 Input model 36
6.4.4 Model for statistical and historical statistical data 46
6.4.5 Model for system functions 50
7 Application view 52
7.1 General 52
7.2 First modelling step – Logical nodes and data 53
7.3 Mode and behaviour of a logical node 57
7.4 Use of measurement ranges and alarms for supervision functions 57
7.5 Data used for limiting the access to control actions 58
7.6 Data used for blocking functions described by logical nodes 58
7.7 Data used for logical node inputs/outputs blocking (operational blocking) 58
7.7.1 General 58
7.7.2 Blocking incoming commands 59
7.7.3 Blocking process outputs 59
7.7.4 Blocking oscillating inputs 60
7.8 Data used for testing 60
7.8.1 General 60
7.8.2 Multicast signals used for simulation 60
7.8.3 Input signals used for testing 61
7.8.4 Test mode 62
7.9 Logical node used for extended logging functions 62
8 Device view 63
8.1 General 63

8.2 Second modelling step – logical device model ... 64
8.2.1 The logical device concept ... 64
8.2.2 The device nameplate ... 65
8.2.3 Gateways and proxies ... 66
8.2.4 Logical devices for monitoring external device health ... 67
8.2.5 Logical devices management hierarchy ... 68
9 Communication view ... 70
9.1 General ... 70
9.2 The service models of the IEC 61850 series ... 70
9.3 The virtualisation ... 72
9.4 Basic information exchange mechanisms ... 73
9.5 The client-server building blocks ... 75
9.5.1 Server ... 75
9.5.2 Client-server roles ... 76
9.6 Logical nodes communicate with logical nodes ... 77
9.7 Interfaces inside and between devices ... 78
10 Where physical devices, application models and communication meet ... 79
11 Relationships between IEC 61850-7-2, IEC 61850-7-3 and IEC 61850-7-4 ... 80
11.1 Refinements of class definitions ... 80
11.2 Example 1 – Logical node and data class ... 81
11.3 Example 2 – Relationship of IEC 61850-7-2, IEC 61850-7-3, and IEC 61850-7-4 ... 85
12 Formal specification method ... 86
12.1 Notation of ACSI classes ... 86
12.2 Class modelling ... 87
12.2.1 Overview ... 87
12.2.2 Common data class ... 88
12.2.3 Logical node class ... 91
12.3 Service tables ... 92
12.4 Referencing instances ... 93
13 Name spaces ... 96
13.1 General ... 96
13.2 Name spaces defined in the IEC 61850-7-x series ... 97
13.3 Specification of name spaces ... 101
13.3.1 General ... 101
13.3.2 Specification ... 101
13.4 Attributes for references to name spaces ... 102
13.4.1 General ... 102
13.4.2 Attribute for logical device name space (ldNs) ... 103
13.4.3 Attribute for logical node name space (lnNs) ... 103
13.4.4 Attribute for data name space (dataNs) ... 104
13.4.5 Attribute for common data class name space (cdcNs) ... 104
14 Common rules for new version of classes and for extension of classes ... 104
14.1 General ... 104
14.2 Basic rules ... 104
14.3 Rules for LN classes ... 105
14.3.1 Use of standardized LN classes ... 105
14.3.2 Extensions to standardized LN classes made by third parties ... 106
14.3.3 New LN classes ... 106

14.3.4 New versions of standardized LN classes made by name space owners 107
14.4 Rules for common data classes and control block classes 107
14.4.1 New common data classes and control block classes 107
14.4.2 New versions of standardized common data classes 107
14.4.3 New versions of control block classes 107
14.5 Multiple instances of LN classes for dedicated and complex functions 108
14.5.1 Example for time overcurrent 108
14.5.2 Example for PDIS 108
14.5.3 Example for power transformer 108
14.5.4 Example for auxiliary network 108
14.6 Specialisation of data by use of number extensions 109
14.7 Examples for new LNs 109
14.8 Example for new Data 109
Annex A (informative) Overview of logical nodes and data 110
Annex B (informative) Allocation of data to logical nodes 113
Annex C (informative) Use of the substation configuration language (SCL) 116
Annex D (informative) Applying the LN concept to options for future extensions 118
Annex E (informative) Relation between logical nodes and PICOMs 123
Annex F (informative) Mapping the ACSI to real communication systems 124
Bibliography 132

Figure 1 – Relations between modelling and mapping parts of the IEC 61850 series 14
Figure 2 – Sample substation automation topology 16
Figure 3 – Modelling approach (conceptual) 17
Figure 4 – Logical node information categories 20
Figure 5 – Build-up of devices (principle) 20
Figure 6 – Position information depicted as a tree (conceptual) 21
Figure 7 – Service excerpt 23
Figure 8 – Example of communication mapping 25
Figure 9 – Summary 26
Figure 10 – Decomposition and composition process (conceptual) 27
Figure 11 – XCBR1 information depicted as a tree 30
Figure 12 – Example of IED composition 31
Figure 13 – Output and input model (principle) 32
Figure 14 – Output model (step 1) (conceptual) 33
Figure 15 – Output model (step 2) (conceptual) 34
Figure 16 – GSE output model (conceptual) 34
Figure 17 – Setting data (conceptual) 35
Figure 18 – Input model for analogue values (step 1) (conceptual) 37
Figure 19 – Range and deadbanded value (conceptual) 38
Figure 20 – Input model for analogue values (step 2) (conceptual) 39
Figure 21 – Reporting and logging model (conceptual) 40
Figure 22 – Data set members and reporting 41
Figure 23 – Buffered report control block (conceptual) 42

Figure 24 – Buffer time 43
Figure 25 – Data set members and inclusion-bitstring 44
Figure 26 – Log control block (conceptual)..44
Figure 27 – Peer-to-peer data value publishing model (conceptual) 45
Figure 28 – Conceptual model of statistical and historical statistical data (1) 47
Figure 29 – Conceptual model of statistical and historical statistical data (2) 49
Figure 30 – Concept of the service tracking model – Example: control service tracking.........51
Figure 31 – Real world devices 52
Figure 32 – Logical nodes and data (IEC 61850-7-2) 53
Figure 33 – Simple example of modelling...55
Figure 34 – Basic building blocks 55
Figure 35 – Logical nodes and PICOM 56
Figure 36 – Logical nodes connected (outside view in IEC 61850-7-x series)........................56
Figure 37 – Mode and behaviour data (IEC 61850-7-4)..57
Figure 38 – Data used for limiting the access to control actions (IEC 61850-7-4) 58
Figure 39 – Data used for logical node inputs/outputs blocking (IEC 61850-7-4) 59
Figure 40 – Data used for receiving simulation signals...60
Figure 41 – Example of input signals used for testing 61
Figure 42 – Test mode example..62
Figure 43 – Logical node used for extended logging functions (GLOG) 63
Figure 44 – Logical device building block 64
Figure 45 – Logical devices and LLN0/LPHD 65
Figure 46 – The common data class DPL...66
Figure 47 – Logical devices in proxies or gateways..67
Figure 48 – Logical devices for monitoring external device health 68
Figure 49 – Logical devices management hierarchy 69
Figure 50 – ACSI communication methods...71
Figure 51 – Virtualisation 73
Figure 52 – Virtualisation and usage...73
Figure 53 – Information flow and modelling 74
Figure 54 – Application of the GSE model...74
Figure 55 – Server building blocks 75
Figure 56 – Interaction between application process and application layer
(client/server) 76
Figure 57 – Example for a service...76
Figure 58 – Client/server and logical nodes 77
Figure 59 – Client and server roles 77
Figure 60 – Logical nodes communicate with logical nodes..78
Figure 61 – Interfaces inside and between devices 79
Figure 62 – Component hierarchy of different views (excerpt) 80
Figure 63 – Refinement of the DATA class..81
Figure 64 – Instances of a DATA class (conceptual) 84
Figure 65 – Relation between parts of the IEC 61850 series 85

Figure 66 – Abstract data model example for IEC 61850-7-x ... 87
Figure 67 – Relation of TrgOp and Reporting ... 90
Figure 68 – Sequence diagram ... 92
Figure 69 – References ... 93
Figure 70 – Use of FCD and FCDA ... 94
Figure 71 – Object names and object reference ... 95
Figure 72 – Definition of names and semantics ... 96
Figure 73 – One name with two meanings ... 97
Figure 74 – Name space as class repository ... 98
Figure 75 – All instances derived from classes in a single name space ... 99
Figure 76 – Instances derived from multiple name spaces ... 100
Figure 77 – Inherited name spaces ... 100
Figure 78 – Basic extension rules diagram ... 105
Figure B.1 – Example for control and protection LNs combined in one physical device ... 113
Figure B.2 – Merging unit and sampled value exchange (topology) ... 114
Figure B.3 – Merging unit and sampled value exchange (data) ... 114
Figure C.1 – Application of SCL for LNs (conceptual) ... 116
Figure C.2 – Application of SCL for data (conceptual) ... 117
Figure D.1 – Seamless communication (simplified) ... 118
Figure D.2 – Example for new logical nodes ... 119
Figure D.3 – Example for control center view and mapping to substation view ... 121
Figure E.1 – Exchanged data between subfunctions (logical nodes) ... 123
Figure E.2 – Relationship between PICOMS and client/server model ... 123
Figure F.1 – ACSI mapping to an application layer ... 124
Figure F.2 – ACSI mappings (conceptual) ... 125
Figure F.3 – ACSI mapping to communication stacks/profiles ... 126
Figure F.4 – Mapping to MMS (conceptual) ... 126
Figure F.5 – Mapping approach ... 127
Figure F.6 – Mapping detail of mapping to a MMS named variable ... 128
Figure F.7 – Example of MMS named variable (process values) ... 128
Figure F.8 – Use of MMS named variables and named variable list ... 129
Figure F.9 – MMS information report message ... 130
Figure F.10 – Mapping example ... 131

Table 1 – LN groups ... 18
Table 2 – Logical node class XCBR (conceptual) ... 29
Table 3 – Excerpt of integer status setting ... 36
Table 4 – Comparison of the data access methods ... 41
Table 5 – ACSI models and services ... 71
Table 6 – Logical node circuit breaker ... 82
Table 7 – Controllable double point (DPC) ... 83
Table 8 – ACSI class definition ... 86
Table 9 – Single point status common data class (SPS) ... 88

Table 10 – Quality components attribute definition 89
Table 11 – Basic status information template (excerpt) 89
Table 12 – Trigger option 90
Table 13 – GenLogicalNodeClass definition 91
Table 14 – Excerpt of logical node name plate common data class (LPL) 103
Table 15 – Excerpt of common data class 103
Table A.1 – Excerpt of data classes for measurands 111
Table A.2 – List of common data classes (excerpt) 112

INTRODUCTION

This part of the IEC 61850 series provides an overview of the architecture for communication and interactions between systems for power utility automation such as protection devices, breakers, transformers, substation hosts etc.

This document is part of a set of specifications which details a layered communication architecture for power utility automation. This architecture has been chosen to provide abstract definitions of classes (representing hierarchical information models) and services such that the specifications are independent of specific protocol stacks, implementations, and operating systems.

The goal of the IEC 61850 series is to provide interoperability between the IEDs from different suppliers or, more precisely, between functions to be performed by systems for power utility automation but residing in equipment (physical devices) from different suppliers. Interoperable functions may be those functions that represent interfaces to the process (for example, circuit breakers) or substation automation functions such as protection functions. This part of the IEC 61850 series uses simple examples of functions to describe the concepts and methods applied in the IEC 61850 series.

This part of the IEC 61850 series describes the relationships between other parts of the IEC 61850 series. Finally this part defines how interoperability is reached.

NOTE Interchangeability is the ability to replace a device from the same vendor, or from different vendors, utilising the same communication interface and as a minimum, providing the same functionality, with no impact on the rest of the system. If differences in functionality are accepted, the exchange may also require some changes somewhere else in the system. Interchangeability implies a standardisation of functions and, in a strong sense, of devices which are outside the scope of this standard. Interchangeability is outside the scope, but it will be supported following this standard for interoperability.

This part of the IEC 61850 series is intended for all stakeholders of standardised communication and standardised systems in the utility industry. It provides an overview of and an introduction to IEC 61850-7-4, IEC 61850-7-3, IEC 61850-7-2, IEC 61850-6, and IEC 61850-8-1.

COMMUNICATION NETWORKS AND SYSTEMS FOR POWER UTILITY AUTOMATION –

Part 7-1: Basic communication structure – Principles and models

1 Scope

This part of the IEC 61850 series introduces the modelling methods, communication principles, and information models that are used in the various parts of the IEC 61850-7-x series. The purpose of this part of the IEC 61850 series is to provide – from a conceptual point of view – assistance to understand the basic modelling concepts and description methods for:

– substation-specific information models for power utility automation systems,
– device functions used for power utility automation purposes, and
– communication systems to provide interoperability within power utility facilities.

Furthermore, this part of the IEC 61850 series provides explanations and provides detailed requirements relating to the relation between IEC 61850-7-4, IEC 61850-7-3, IEC 61850-7-2 and IEC 61850-5. This part explains how the abstract services and models of the IEC 61850-7-x series are mapped to concrete communication protocols as defined in IEC 61850-8-1.

The concepts and models provided in this part of the IEC 61850 series may also be applied to describe information models and functions for:

– hydroelectric power plants,
– substation to substation information exchange,
– information exchange for distributed automation,
– substation to control centre information exchange,
– information exchange for metering,
– condition monitoring and diagnosis, and
– information exchange with engineering systems for device configuration.

NOTE 1 This part of IEC 61850 uses examples and excerpts from other parts of the IEC 61850 series. These excerpts are used to explain concepts and methods. These examples and excerpts are informative in this part of IEC 61850.

NOTE 2 Examples in this part use names of classes (e.g. XCBR for a class of a logical node) defined in IEC 61850-7-4, IEC 61850-7-3, and service names defined in IEC 61850-7-2. The normative names are defined in IEC 61850-7-4, IEC 61850-7-3, and IEC 61850-7-2 only.

NOTE 3 This part of IEC 61850 does not provide a comprehensive tutorial. It is recommended that this part be read first – in conjunction with IEC 61850-7-4, IEC 61850-7-3, and IEC 61850-7-2. In addition, it is recommended that IEC 61850-1 and IEC 61850-5 also be read.

NOTE 4 This part of IEC 61850 does not discuss implementation issues.

2 Normative references

The following referenced documents are indispensable for the application of this document. 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 in substations – Part 2: Glossary*

IEC 61850-3, *Communication networks and systems in substations – Part 3: General requirements*

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

IEC 61850-5, *Communication networks and systems in substations – Part 5: Communication requirements for functions and device models*

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

IEC 61850-7-2, *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, *Communication networks and systems for power utility automation – Part 7-3: Basic communication structure – Common data classes*

IEC 61850-7-4, *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, *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*

IEC 61850-9-2, *Communication networks and systems in substations – Part 9-2: Specific Communication Service Mapping (SCSM) – Sampled values over ISO/IEC 8802-3*

IEC 61850-10, *Communication networks and systems in substations – Part 10: Conformance testing*

ISO/IEC 8802-3, *Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements – Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications*

ISO/IEC 8825 (all parts), *Information technology – ASN.1 encoding rules*

ISO 9506-1, *Industrial automation systems – Manufacturing Message Specification – Part 1: Service definition*

ISO 9506-2, *Industrial automation systems – Manufacturing Message Specification – Part 2: Protocol specification*

3 Terms and definitions

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

3.1
information
knowledge concerning objects, such as facts, events, things, processes, or ideas, including concepts, that within a certain context has a particular meaning

[IEC 60050-101:1998, 101-12-01]

3.2
information model
knowledge concerning power utility functions and devices in which the functions are implemented

This knowledge is made visible and accessible through the means of the IEC 61850 series. The model describes in an abstract way a communication-oriented representation of a real function or device.

3.3
model
a representation of some aspect of reality

The purpose of creating a model is to help understand, describe, or predict how things work in the real world by exploring a simplified representation of a particular entity or phenomenon. The focus of the model defined in IEC 61850-7-x is on the communication features of the data and functions modelled.

4 Abbreviated terms

ACSI	Abstract communication service interface
ASN.1	Abstract syntax notation one
API	Application program interface
CDC	Common data class
CT	Current transformer
DST	Daylight saving time
GOOSE	Generic oriented object system event
IED	Intelligent electronic device
LD	Logical device
LN	Logical node
LLN0	Logical node zero
LPHD	Logical node physical device
MMS	Manufacturing message specification
PHD	Physical device

PICOM	Piece of communication
SAS	Substation automation system
SCSM	Specific communication service mapping
SoE	Sequence of events
SMV	Sample values
UCAIug	UCA international users group
UTC	Universal time coordinated
VMD	Virtual manufacturing device
VT	Voltage transformer
XML	extended markup language

5 Overview of the IEC 61850 series concepts

5.1 Objective

IEC 61850-7-4, IEC 61850-7-3, IEC 61850-7-2, IEC 61850-6, and IEC 61850-8-1 are closely related. This subclause provides an overview of these parts and it describes how they are interwoven. The modelling and implementation methods applied in the different parts of the standard and their relation are shown in Figure 1.

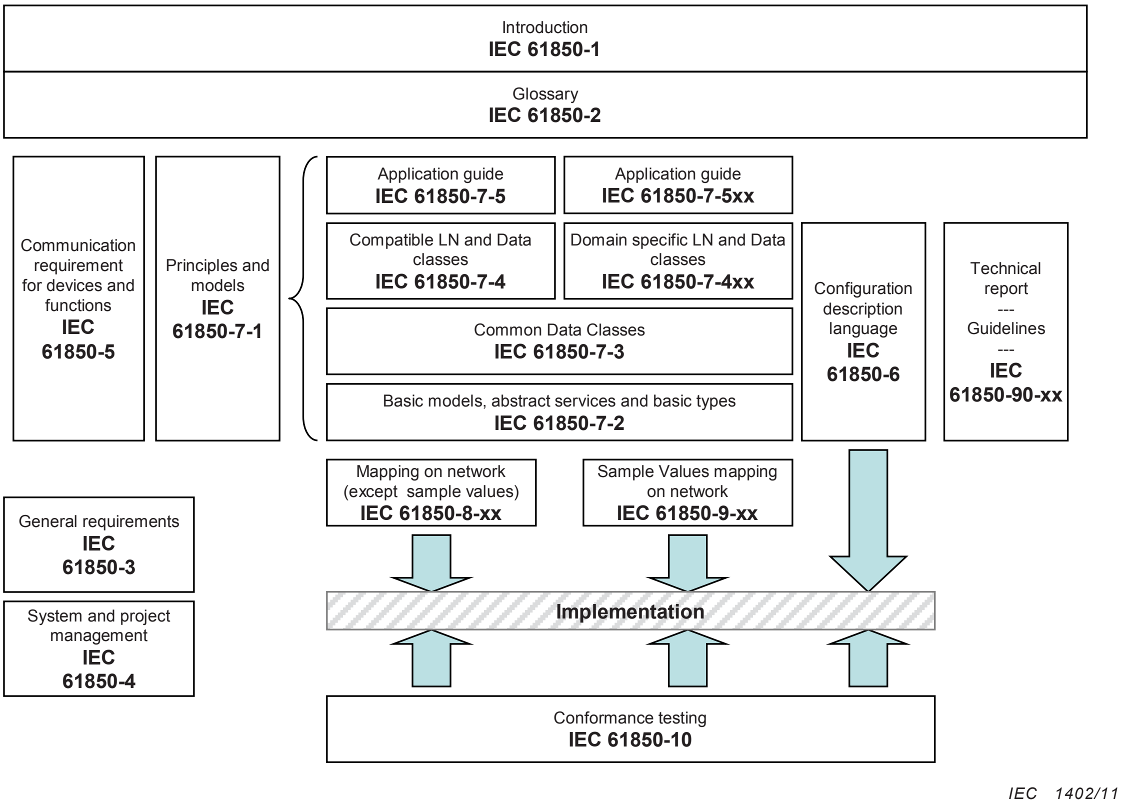


Figure 1 – Relations between modelling and mapping parts of the IEC 61850 series

Each part defines a specific aspect of a substation IED:

– IEC 61850-1 gives an introduction and overview of the IEC 61850 series,
– IEC 61850-2 contains the glossary of specific terminology and definitions used in the context of power utility automation systems within the various parts of the standard,
– IEC 61850-3 specifies the general requirements of the communication network with regard to the quality requirements, environmental conditions and auxiliary services,
– IEC 61850-4 pertains to the system and project management with respect to the engineering process, the life cycle of the SAS and the quality assurance from the development stage to the discontinuation and decommissioning of the SAS.
– IEC 61850-5 specifies the communication requirements of the functions being performed in systems for power utility automation and to device models. All known functions and their communication requirements are identified,
– this part of IEC 61850 defines the basic principles and modelling methods,
– IEC 61850-6 specifies a file format for describing communication related IED (intelligent electronic device) configurations and IED parameters, communication system configurations, switchyard (function) structures, and the relations between them. The main purpose of the format is to exchange IED capability descriptions, and system level descriptions between engineering tools of different manufacturers in a compatible way. The defined language is called substation configuration description language (SCL). Mapping specific extensions or usage rules may be required in the appropriate parts.
– IEC 61850-7-5 defines the usage of information models for substation automation applications. It gives clear examples on how to apply LNs and data defined in IEC 61850-7-4 for different substation applications. The examples cover applications from monitoring function to protection blocking schemes. Other domain specific application guides which are within the scope of IEC technical committee 57 are defined in the IEC 61850-7-5xx series[1]. Examples are hydropower and distributed energy resources domains,
– IEC 61850-7-4 defines specific information models for substation automation functions (for example, breaker with status of breaker position, settings for a protection function, etc.) – what is modelled and could be exchanged. Other domain specific information models within the scope of IEC technical committee 57 are defined in the 61850-7-4xx series,
– IEC 61850-7-3 has a list of commonly used information (for example, for double point control, 3-phase measurand value, etc.) – what the common basic information is,
– IEC 61850-7-2 provides the services to exchange information for the different kinds of functions (for example, control, report, get and set, etc.) – how to exchange information,
– IEC 61850-8-1 defines the concrete means to communicate the information between IEDs (for example, the application layer, the encoding, etc.) – how to serialise the information during the exchange,
– IEC 61850-9-2, and particularly the subset 9-2LE described in the “Implementation Guideline for Digital Interface to Instrument Transformers using IEC 618509-2” by the UCAIug, defines the concrete means to communicate sampled values between sensors and IEDs,
– IEC 61850-10 defines the methods and abstract cases for conformance testing of devices and engineering tools as well as the metrics to be measured within devices according to the requirements defined in IEC 61850-5,
– there may be object classes defined for various other application domains outside the scope of IEC technical committee 57. They are relevant to Figure 1 only if they are built according to the approach of the IEC 61850 series.

---

1 IEC 61850-7-4xx, -7-5xx, -8-xx, -9-xx and -90-xx are series of documents whose scope is similar. For example, IEC 61850-7-4 deals with data object classes used for substations while IEC 61850-7-410 deals with data object classes used for hydroelectric power plants. IEC 61850-90-xx series is reserved for technical reports or guidelines.

5.2 Topology and communication functions of substation automation systems

As shown by the topology in Figure 2, one focus of the IEC 61850 series is the support of substation automation functions by the communication of (numbers in brackets refer to the figure):

– sampled value exchange for CTs and VTs (1),
– fast exchange of I/O data for protection and control (2),
– control signals (3),
– trip signals (4),
– engineering and configuration (5),
– monitoring and supervision (6),
– control-center communication (7),
– time-synchronisation,
– etc.

Support for other functions such as metering, condition monitoring, and asset management is provided as well.

Many functions are implemented in intelligent electronic devices (IED). Several functions may be implemented in a single IED or one function may be implemented in one IED and another function may be hosted by another IED. IEDs (i.e., the functions residing in IEDs) communicate with functions in other IEDs by the information exchange mechanisms of this standard. Therefore, functions distributed over more than one IED may be also implemented.

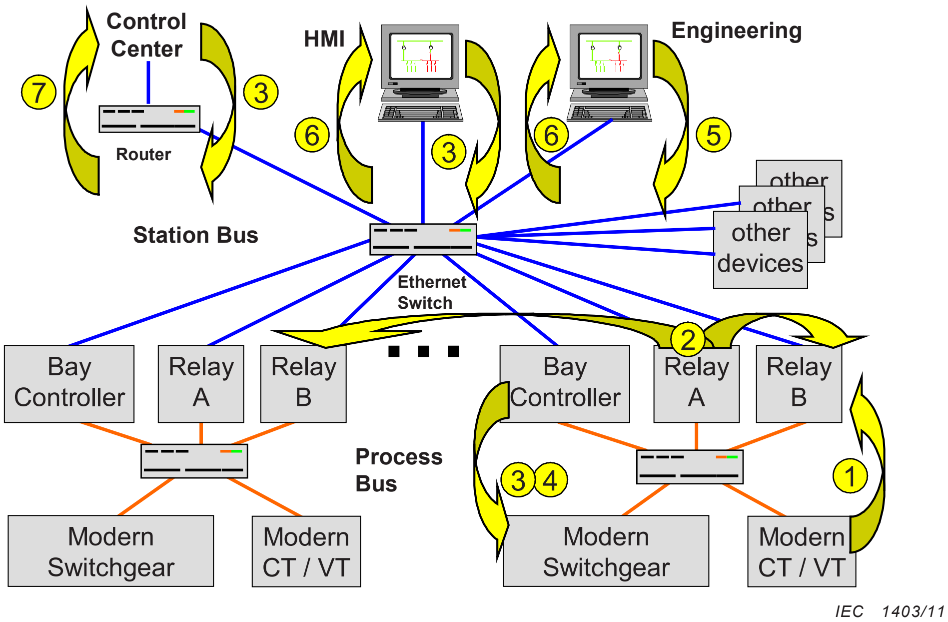


Figure 2 – Sample substation automation topology

5.3 The information models of substation automation systems

The information exchange mechanisms rely primarily on well-defined information models. These information models and the modelling methods are at the core of the IEC 61850 series. The IEC 61850 series uses the approach to model the common information found in real devices as depicted in Figure 3. All information made available to be exchanged with other devices is defined in the standard. The model provides for systems for power utility automation an image of the analogue world (power system process, switchgear).

NOTE 1 "The common information" in the context of the IEC 61850 series means that the stakeholders of substation automation systems (users and vendors) have agreed that the information defined in the IEC 61850 series is widely accepted and required for the open exchange of information between any kind of substation IEDs.

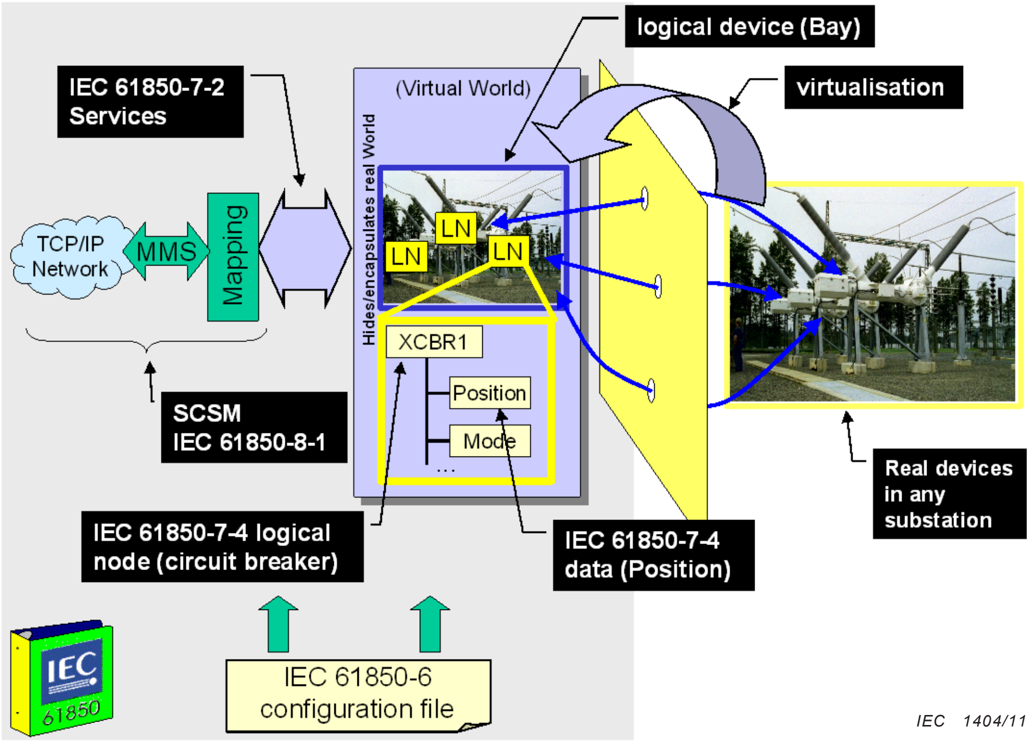


Figure 3 – Modelling approach (conceptual)

Implementations to reach interoperability have to be based on common understanding of definitions. Therefore, the parts describing the data model contain mandatory semantic tables which have to be considered very carefully.

The IEC 61850 series defines the information and information exchange in a way that it is independent of a concrete implementation (i.e., it uses abstract models). The standard also uses the concept of virtualisation. Virtualisation provides a view of those aspects of a real device that are of interest for the information exchange with other devices. Only those details that are required to provide interoperability of devices are defined in the IEC 61850 series.

As described in IEC 61850-5, the approach of the standard is to decompose the application functions into the smallest entities, which are used to exchange information. The granularity is given by a reasonable distributed allocation of these entities to dedicated devices (IED). These entities are called logical nodes (for example, a virtual representation of a circuit breaker class, with the standardised class name XCBR). The logical nodes are modelled and defined from the conceptual application point of view in IEC 61850-5. Several logical nodes build a logical device (for example, a representation of a Bay unit). A logical device is always implemented in one IED; therefore logical devices do not contain logical nodes from different IEDs.

Real devices on the right-hand side of Figure 3 are modelled as a virtual model in the middle of the figure. The logical nodes defined in the logical device (for example, bay) correspond to well-known functions in the real devices. In this example, the logical node XCBR represents a specific circuit breaker of the bay to the right.

NOTE 2 The logical nodes of this example may be implemented in one or several IEDs as appropriate. If the logical nodes are implemented in different IEDs, they need exchange information over a network. Information exchange inside a logical node is outside the scope of the IEC 61850 series.

Based on their functionality, a logical node contains a list of data (for example, position) with dedicated data attributes. The data have a structure and a well-defined semantic (meaning in the context of systems for power utility automation or, e.g. more specifically, of substation automation systems). The information represented by the data and their attributes are exchanged by the services according to the well-defined rules and the requested performance as described in IEC 61850-5. The services are implemented by a specific and concrete communication means (SCSM, for example, using MMS, TCP/IP, and Ethernet among others).

The logical nodes and the data contained in the logical device are crucial for the description and information exchange for substation automation systems to reach interoperability.

The logical devices, the logical nodes and the data they contain need to be configured. The main reason for the configuration is to select the appropriate logical nodes and data from the standard and to assign the instance-specific values, for example, concrete references between instances of the logical nodes (their data) and the exchange mechanisms, and initial values for process data.

5.4 Applications modelled by logical nodes defined in IEC 61850-7-4

Table 1 lists all groups of logical nodes defined in IEC 61850-7-4. Over one hundred logical nodes covering the most common applications of substation and feeder equipment are defined. While the definition of information models for protection and protection related applications is important because of the high impact of protection for safe and reliable operation of the power system, the covered applications include many other functions like monitoring, measurement, control and power quality.

Table 1 – LN groups

Group indicator	Logical node groups
A	Automatic control
C	Supervisory control
D	DER (Distributed Energy Resources)
F	Functional blocks
G	Generic function references
H	Hydro power
I	Interfacing and archiving
K	Mechanical and non-electrical primary equipment
L	System logical nodes
M	Metering and measurement
P	Protection functions
Q	Power quality events detection related
R	Protection related functions
S	Supervision and monitoring
T	Instrument transformer and sensors
W	Wind power
X	Switchgear
Y	Power transformer and related functions
Z	Further (power system) equipment

IEC 61850 has well-defined rules to define additional logical nodes and data, for example for additional functions within substations or for other application domains such as wind power plants. For details on the extension rules, see Clauses 13 and 14 of this standard.

The following excerpt of the logical nodes has been included to provide an example of what kind of real applications the logical nodes represent:

- distance protection;
- differential protection;
- overcurrent;
- undervoltage;
- directional over power;
- volts per Hz relay;
- transient earth fault;
- directional element;
- harmonic restraint;
- protection scheme;
- zero speed or underspeed;
- measurement;
- metering;
- sequence and imbalance;
- harmonics and interharmonics;
- differential measurements;
- switch control;
- circuit breaker;
- circuit switch;
- and others.

Most logical nodes provide information that can be categorised as depicted in Figure 4. The semantic of a logical node is represented by data and data attributes. Logical nodes may provide a few or up to 30 data. Data may contain a few or even more than 20 data attributes. Logical nodes may contain more than 100 individual information (points) organised in a hierarchical structure.

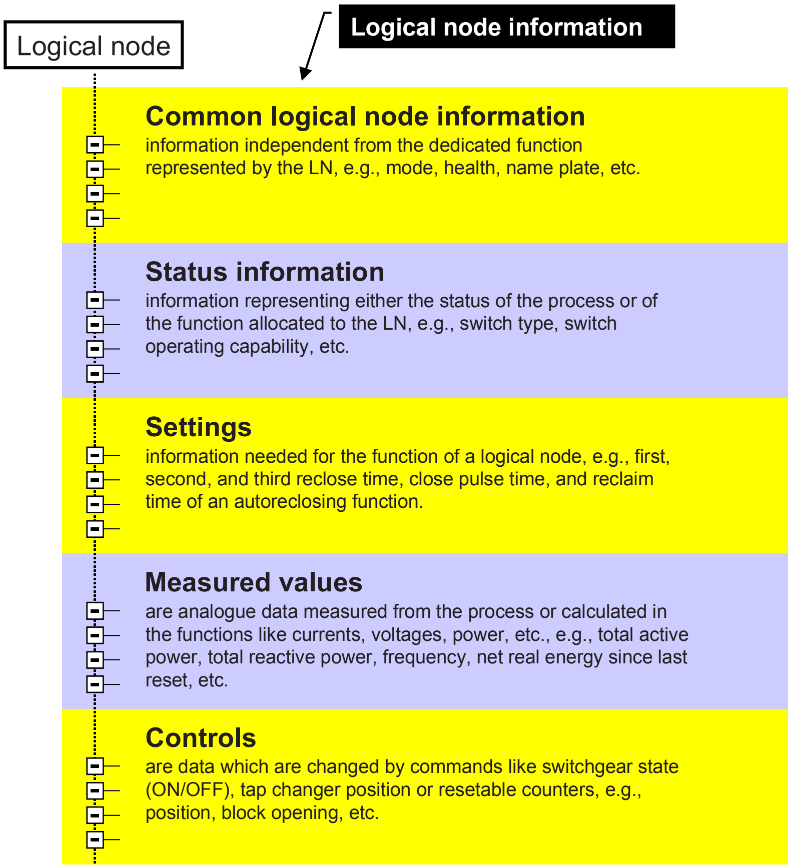


Figure 4 – Logical node information categories

IEDs are built up by composing logical nodes as depicted in Figure 5. The logical nodes are the building blocks of substation IEDs, for example, circuit breaker (XCBR) and others. In the example for each phase, one instance of XCBR is used.

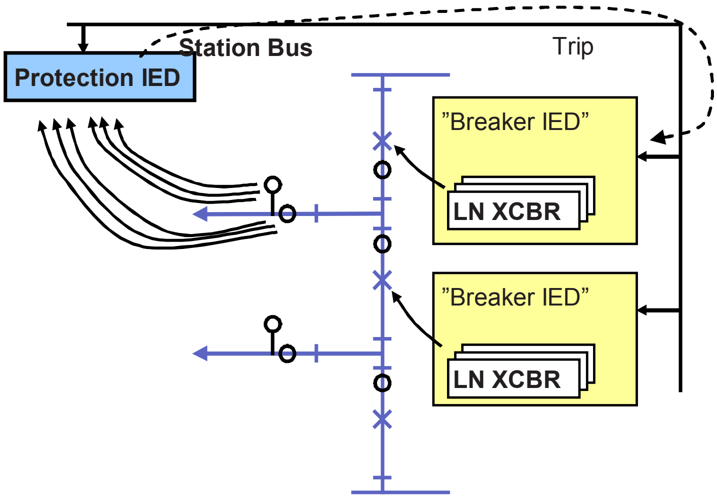


Figure 5 – Build-up of devices (principle)

In Figure 5, the protection IED receives the values for the voltage and current from conventional VT and CT. The protection functions in the protection device may detect a fault and issue or send a trip signal via the station bus. The standard supports also IEDs for digitizing VTs and CTs sending voltage and current as samples to the protection over a serial link. The output of conventional VTs and CTs may also be converted at the source to samples and transmitted over this serial link.

The logical nodes are used to build up substation IEDs.

5.5 The semantic is attached to data

The mean number of specific data provided by logical nodes defined in IEC 61850-7-4 is approximately 20. Each of the data (for example, position of a circuit breaker) comprises several details (the data attributes). The position (named “Pos”) of a circuit breaker is defined in the logical node XCBR (see Figure 6). The position is defined as data. The category of the position in the logical node is “controls” – the position can be controlled via a control service.

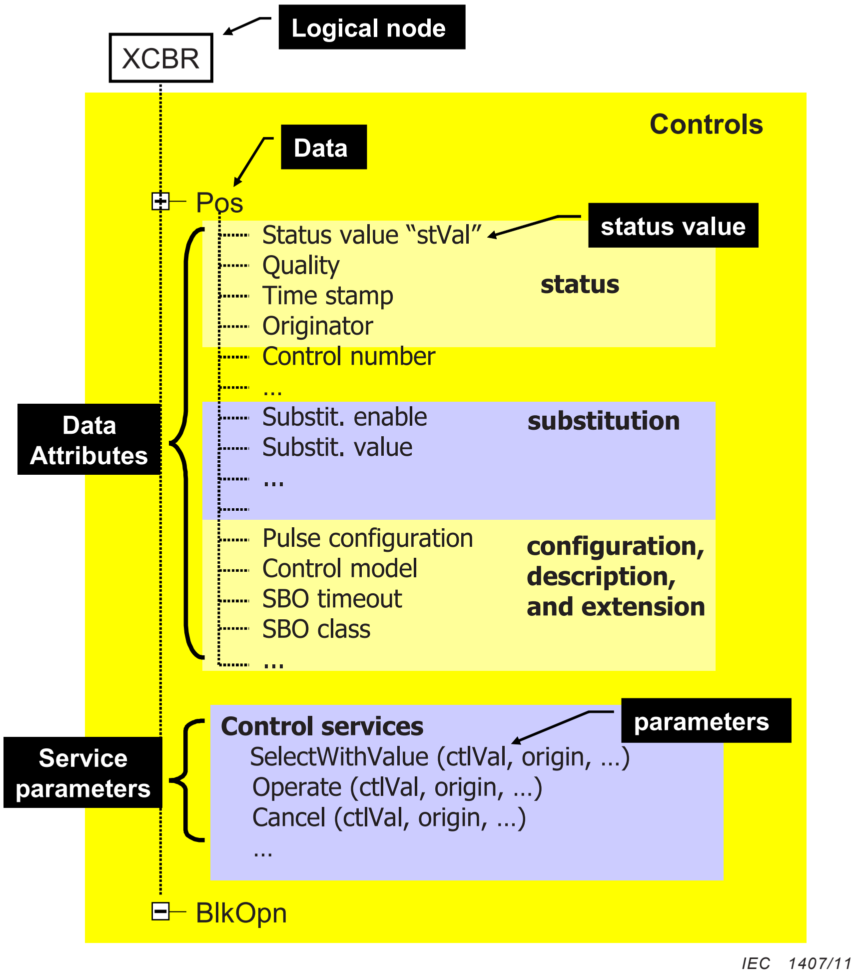


Figure 6 – Position information depicted as a tree (conceptual)

The position Pos is more than just a simple “point” in the sense of simple RTU protocols. It is made up of several data attributes. The data attributes are categorised as follows:

– status (or measured/metered values, or settings),
– substitution,
– configuration, description and extension.

The data example Pos has approximately 20 data attributes accessible through different services. The data attribute Pos.stVal represents the position of the real breaker (could be in intermediate-state, off, on, or bad-state).

The position Pos can be controlled by use of control services and the associated service parameters. It is important to understand that these service parameters are not part of the data model: they do not represent data attributes. They only "live" the time of the command execution.

The position also has information about the originator that issued the command and the control number (given by the originator in the request). Furthermore, the position contains the cause diagnosis of a negative control response. The quality and time stamp information indicate the current validity of the status value and the time of the last change of the status value.

The current values for stVal, the quality and the time stamp (associated with the stVal) can be read, reported or logged in a buffer of the IED.

The values for stVal and quality can be remotely substituted. The substituted values take effect immediately after enabling substitution.

Several data attributes are defined for the configuration of the control behaviour, for example, pulse configuration (single pulse or persistent pulses, on/off-duration, and number of pulses) or control model (direct, select-before-operate, etc.).

Data attributes are defined primarily by an attribute name and an attribute type:

Attribute name	**Attribute type**	**FC**	**TrgOp**	**Value/value range**	**M/O/C**
stVal	CODED ENUM	ST	dchg	intermediate-state | off | on | bad-state	M
ctlModel	CtlModels	CF	dchg	status-only | direct-with-normal-security | sbo-with-normal-security | directwith-enhanced-security | sbo-with-enhanced-security	M

Additional information provides further details (one could say provides meta-data) on:

– the services allowed: functional constraint -> FC=SV means that specific services shall be applied only (for example SV refers to the substitution service),

– the trigger conditions that cause a report to be sent: TrgOp=dchg means that a change in the value of that attribute causes a report,

– the value or value range,

– the indication if the attribute is optional (O), mandatory (M), conditional mandatory (X_X_M), or conditional optional (X_X_O). The conditions result from the fact that not all attributes are independent from each other.

The data attribute names are standardised (i.e., they are reserved) names that have a specific semantic in the context of the IEC 61850 series. The semantic of all data attribute names is defined at the end of IEC 61850-7-3; for example:

Data attribute name	**Semantics**
stVal	Status value of the data.
ctlModel	Configuration information about the control model.

The names of the data and data attributes carry the crucial semantic of a substation IED.

The position information Pos as shown in Figure 6 has many data attributes that can found in many other switching-specific applications. The prime characteristic of the position is the data attribute stVal (status value) which represents four states: intermediate-state | off | on | bad-state. These four states (represented usually with two bits) are commonly known as "double point" information. The whole set of all the data attributes defined for the data Pos (position) is called a "common data class" (CDC). The name of the common data class of the double point information is DPC (controllable double point).

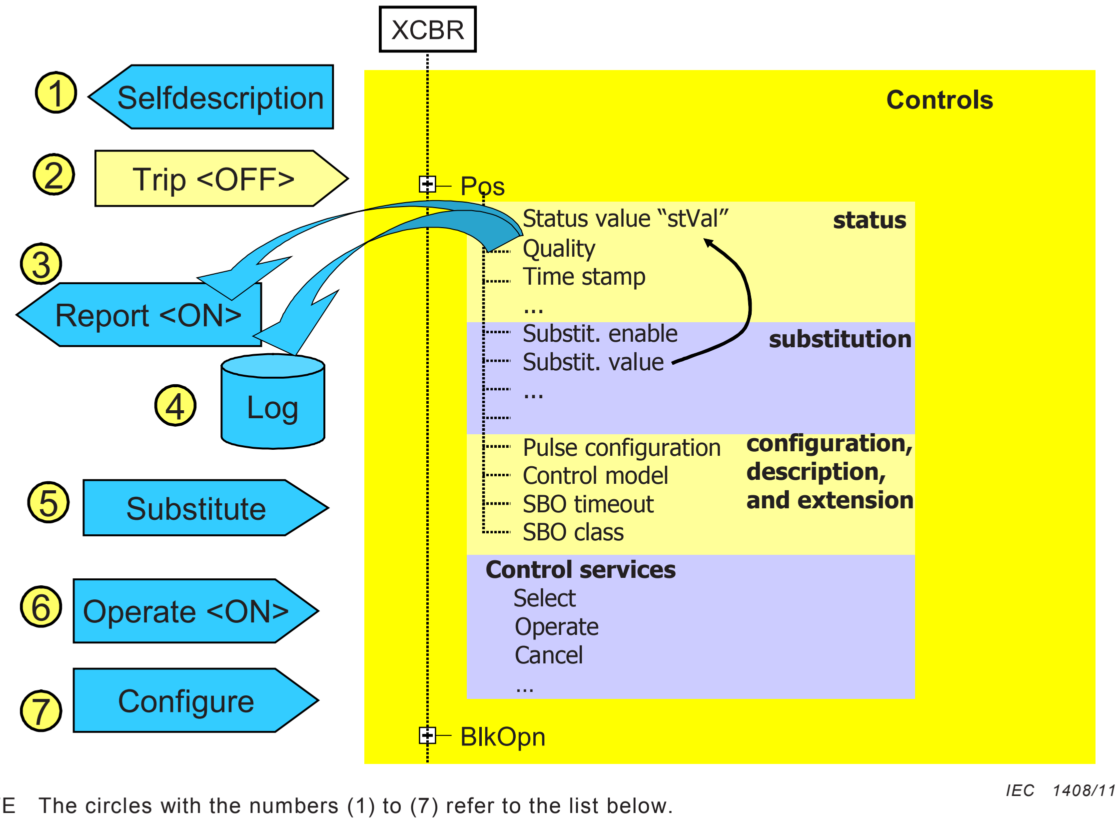
Common data classes provide a useful means to reduce the size of data definitions (in the standard). The data definition does not need to list all the attributes but needs to just reference the common data class. Common data classes are also very useful to keep the definitions of data attributes consistent. A change in the double point control CDC specific data attributes only needs to be made in a single place – in the DPC definition of IEC 61850-7-3.

IEC 61850-7-3 defines common data classes for a wide range of well-known applications. The core common data classes are classified into the following groups:

– status information,
– measurand information,
– controllable status information,
– controllable analogue information,
– status settings,
– analogue settings, and
– description information.

5.6 The services to exchange information

The logical nodes, data, data attributes and service parameters are defined mainly to specify the information required to perform an application, and for the exchange of information between IEDs. The information exchange is defined by means of services. An excerpt of the services is displayed in Figure 7.



NOTE The circles with the numbers (1) to (7) refer to the list below.

Figure 7 – Service excerpt

The operate service manipulates the control specific service parameters of a circuit breaker position (open or close the breaker). The report services inform another device that the position of the circuit breaker has been changed. The substitute service forces a specific data attribute to be set to a value independent of the process.

The categories of services (defined in IEC 61850-7-2) are as follows:

– retrieving the self-description of a device, see (1) in Figure 7,
– fast and reliable peer-to-peer exchange of status information (tripping or blocking of functions or devices), see (2) in Figure 7,
– reporting of any set of data (data attributes), SoE – cyclic and event triggered, see (3) in Figure 7,
– logging and retrieving of any set of data (data attributes) – cyclic and event, see (4) in Figure 7,
– substitution, see (5) in Figure 7,
– handling and setting of parameter setting groups,
– transmission of sampled values from sensors,
– time synchronisation,
– file transfer,
– control devices (operate service), see (6) in Figure 7, and
– online configuration, see (7) in Figure 7.

Many services operate directly on the attributes of the information model (i.e., on the data attributes of data contained in logical nodes). The pulse configuration of the data attribute Pos of a specific circuit breaker can be set directly by a client to a new value. Directly means that the service operates on the request of the client without specific constraints of the IED.

Other services provide a more complex behaviour which is dependent on the state of some specific state machine. A control request may be required to follow a state machine associated with the data attribute, for example, select-before-operate.

There are also several application-specific communication services that provide a comprehensive behaviour model which partially act autonomously. The reporting service model describes an operating-sequence in which the IED acts automatically on certain trigger conditions defined in the information model (for example, report on data-change of a status value) or conditions defined in the reporting service model (for example, report on a periodical event).

5.7 Services mapped to concrete communication protocols

The services defined in IEC 61850-7-2 are called abstract services. Abstract means that only those aspects that are required to describe the required actions on the receiving and sending side of a service request are defined in IEC 61850-7-2. They are based on the functional requirements in IEC 61850-5. The semantic of the service models with their attributes and the semantic of the services that operate on these attributes (including the parameters that are carried with the service requests and responses) are defined in IEC 61850-7-2.

The specific syntax (format) and especially the encoding of the messages that carry the service parameters of a service and how these are passed through a network are defined in a specific communication service mapping (SCSM). One SCSM – IEC 61850-8-1 – is the mapping of the services to MMS (ISO 9506-1 and ISO 9506-2) and other provisions like TCP/IP and Ethernet (see Figure 8), another is IEC 61850-9-2 – the direct mapping on Ethernet.

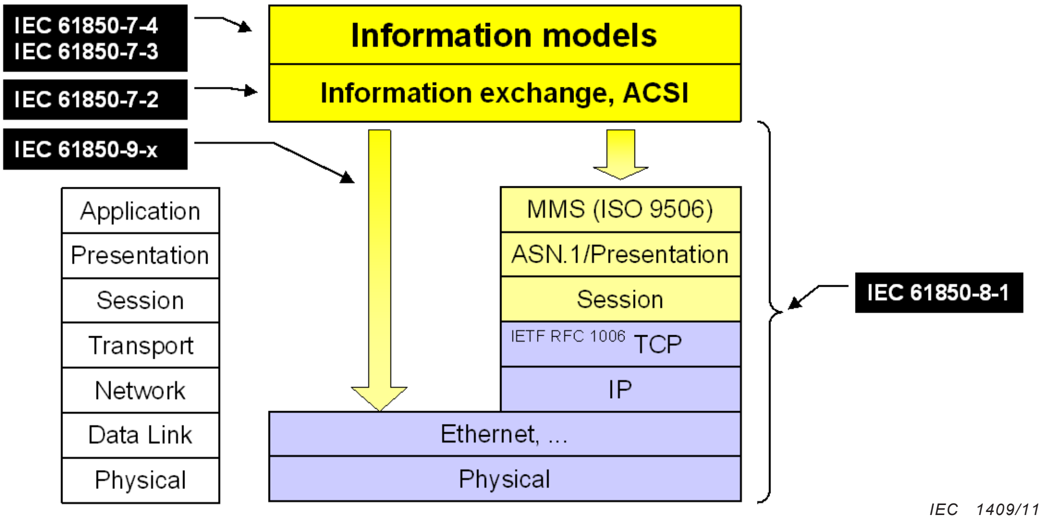


Figure 8 – Example of communication mapping

Additional mappings to other communication stacks are possible. However, to not jeopardize interoperability, the number of accepted mappings in the standard shall be a minimum. The main purpose of this flexibility is to be able to follow over time the evolution in communication technology. The ACSI is independent of the mappings.

5.8 The configuration of the automation system

IEC 61850-6 specifies a file format for describing communication related IED configurations and IED parameters, communication system configurations, switchyard (function) structures, and the relations between them. The main purpose of this format is to exchange IED capability descriptions, and SA system descriptions between IED engineering tools and the system engineering tool(s) of different manufacturers in a compatible way.

The defined language is called substation configuration description language (SCL). The configuration language is based on the extensible markup language (XML) version 1.0.

To support the intended engineering process, the SCL is capable to describe:

a) a system specification in terms of the single line diagram, and allocation of logical nodes to parts and equipment of the single line to indicate the needed functionality,
b) pre-configured IEDs with:
   - the logical node, datasets and report control block definitions,
   - the supported services: GOOSE, sampled values, logging, file handling,
c) pre-configured IEDs with no semantic or a pre-configured semantic for a process part of a certain structure, for example a double busbar GIS line feeder,
d) complete process configuration with all IEDs bound to individual process functions and primary equipment, enhanced by the access point connections and possible access paths in subnetworks for all possible clients,
e) as item d) above, but additionally with all predefined associations and client server connections between logical nodes on data level. This is needed if an IED is not capable of dynamically building associations or reporting connections (either on the client or on the server side).

The scope of SCL is focussed on these purposes:

1) SAS functional specification (point a) above),
2) IED capability description (points b)and c) above), and

3) SA system description (points d) and e) above).

These purposes shall support in a standardised way system design, communication engineering and the description of the readily engineered system communication for the device engineering tools.

5.9 Summary

Figure 9 exhibits a summary of Clause 5. The four main building blocks are

– the substation automation system specific information models,
– the information exchange methods,
– the mapping to concrete communication protocols, and
– the configuration of a substation IED.

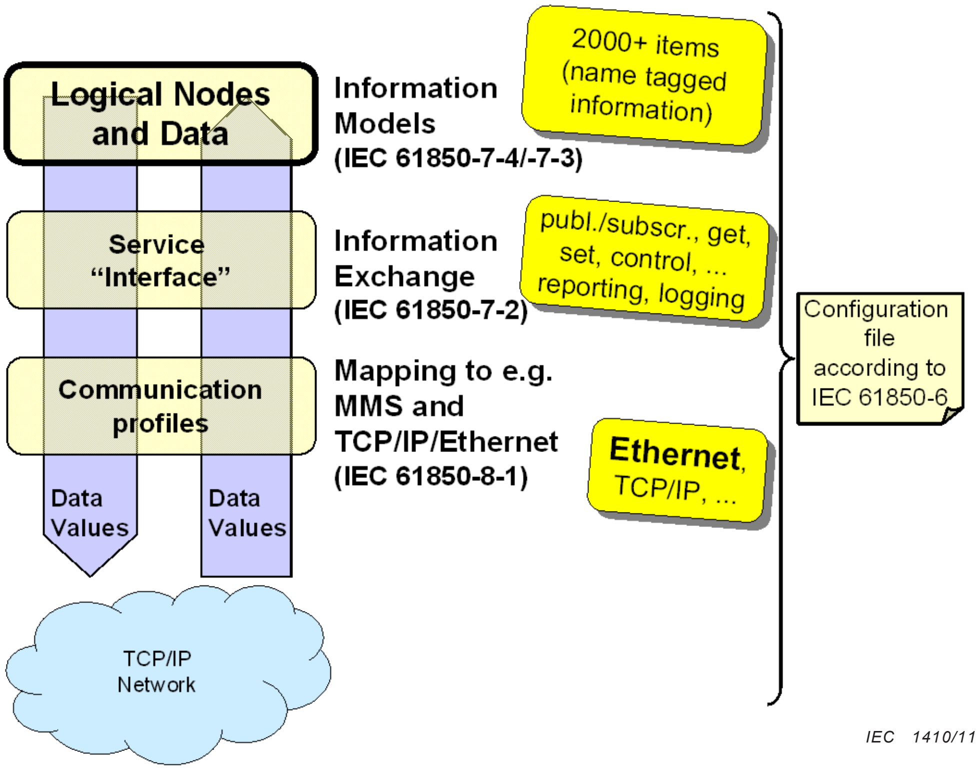


Figure 9 – Summary

These four building blocks are to a high degree independent of each other. The information is separated from the presentation and from the information exchange services. The information exchange services are separated from the concrete communication profiles. It means that the information models can easily be extended, by definition of new logical nodes and new data according to specific and flexible rules, as required by another application domain. In the same way, different communication stacks may be used following the state-of-the-art in communication technology. But to keep interoperability simple, one stack only should be selected at one time. For the selection, see IEC 61850-8-x and IEC 61850-9-x.

Clause 6 provides a more detailed view of the four building blocks.

6 Modelling approach of the IEC 61850 series

6.1 Decomposition of application functions and information

As described in IEC 61850-5, the general approach of the IEC 61850 series is to decompose application functions into the smallest entities, which are then used to communicate. The granularity is given by a reasonable distributed allocation of these entities to dedicated devices (IED). The entities are called logical nodes. The requirements for logical nodes are defined – from an application point of view – in IEC 61850-5.

Based on their functionality, these logical nodes comprise data with dedicated data attributes. The information represented by the data and the data attributes are exchanged by dedicated services according to well-defined rules and the performance requested as required in IEC 61850-5.

The decomposition process (to get the most common logical nodes) and the composition process (to build up devices using logical nodes) are depicted in Figure 10. The data classes contained in logical nodes have been defined to support the most common applications in an understandable and commonly accepted way.

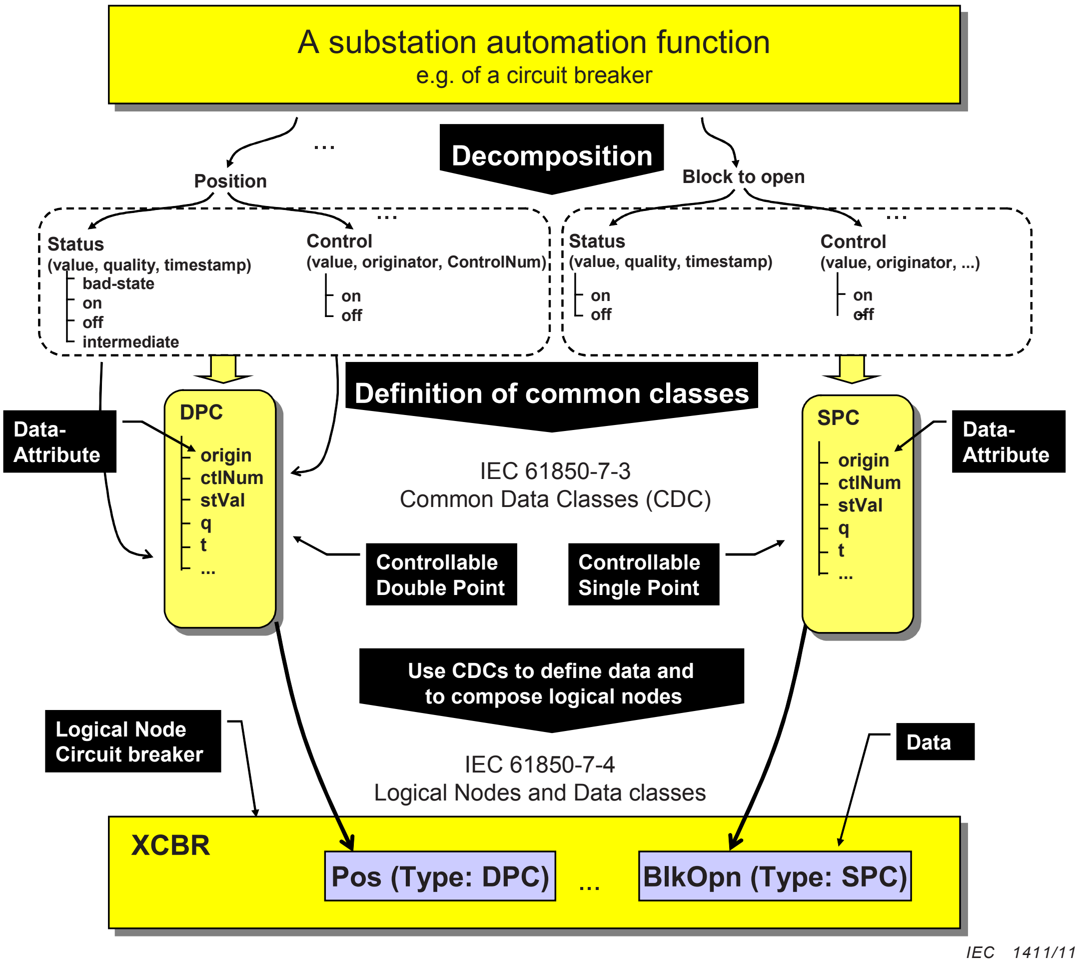


Figure 10 – Decomposition and composition process (conceptual)

A small part of a function (an excerpt of a circuit breaker model) has been selected as an example to explain the decomposition process. The circuit breaker has, among many other attributes, a position which can be controlled and monitored and the capability to prevent the switch being opened (for example, by an operator in service situations, block to open). The position comprises some information that represents the status of the position providing the value of the status (on, off, intermediate, bad state), the quality of the value (good, etc.), and the timestamp of the time of the last change of the position. In addition, the function provides the capability to control the switch: Control value (on, off). To keep track of who controlled the switch, the originator stores the information about the entity that issued the last control command. A control number stores the sequence number of the last control command.

The information grouped under the position (status, etc.) represents a very common group of a four-state value that can be reused many times. Similarly, the "block to open" groups information of a two-state value. These groups are called common data classes (CDC):

– four-state reusable class is defined as controllable double point (DPC), and
– two-state reusable class is defined as controllable single point (SPC).

IEC 61850-7-3 defines many common data classes for status, measurands, controllable status, controllable analogue, status settings, and analogue settings.

6.2 Creating information models by stepwise composition

IEC 61850-7-5xx series, IEC 61850-7-4xx series, IEC 61850-7-3, and IEC 61850-7-2 define how to model the information and communication in power utility applications according to the requirements defined in IEC 61850-5. The modelling uses the logical nodes (and their data that represent a huge amount of semantical definitions) primarily as building blocks to compose the visible information of a power utility automation system. The models are used for description of the information produced and consumed by applications and for the exchange of information with other IEDs.

The logical nodes and data classes introduced in IEC 61850-5 are refined and precisely defined in the IEC 61850-7-4xx series. They have been defined in a joint effort of domain experts of the various power utility application domains and modelling experts. The logical nodes and their data are defined with regard to content (semantic) and form (syntax). The approach uses object oriented methods.

NOTE 1 The logical node classes and data classes modelled and defined in IEC 61850-7-4 meet the requirements listed in IEC 61850-5.

In the next step, the common data classes are used to define the (power utility domain-specific) data classes, see lower half of Figure 10. These data classes (defined in IEC 61850-7-4) are specialised common data classes, for example, the data class Pos (a specialisation of DPC) inherits all data attributes of the corresponding common data class DPC, i.e., the stVal, q, t, etc. The semantic of the class Pos is defined at the end of IEC 61850-7-4.

A logical node groups several data classes to build up a specific functionality. The logical node XCBR represents the common information of a real circuit breaker. The XCBR can be reused to describe the common information of circuit breakers of various makes and types.

IEC 61850-7-4 defines several tens of logical nodes making use of hundreds of data names. The logical node XCBR comprises over 15 data classes. A brief description of the logical node XCBR is given in Table 2.

Table 2 – Logical node class XCBR (conceptual)

Common logical node information
Mode
Behaviour
Health
Name plate
Logical node information
External equipment health
External equipment name plate
Operation counter
Controls
Switch position (see below for details)
Block opening
Block closing
Charger motor enabled
Control authority at station level. Switches between station and higher level.
Metered values
Sum of switched amperes, resetable
Status information
Local operation (Indicates the switchover between local and remote operation; local = TRUE, remote = FALSE)
Circuit breaker operating capability
Point on wave switching capability
Circuit breaker operating capability when fully charged

NOTE 2 IEC 61850-7-4 defines a standardised name for each item such as Pos for the switch position. Additionally, the tables for logical nodes contain the common data class to be used for the corresponding data class. Finally, the tables define if the data class in the table is mandatory or optional. These details are explained later in this part.

The content of the marked “switch position” (name = Pos) is introduced in Figure 11.

IEC 61850-7-x series use tables for the definition of the logical node classes and data classes (see IEC 61850-7-4), the common data classes (see IEC 61850-7-3) and service models (see IEC 61850-7-2). Data classes and data attributes form a hierarchical structure as depicted in Figure 11. The data attributes of the data class Pos are functionally grouped (status, substitution, configuration, etc.).

The data attributes have a standardized name and a standardized type. On the right-hand side the corresponding references (object reference) are shown. These references are used to provide the path information to identify the information in the tree.

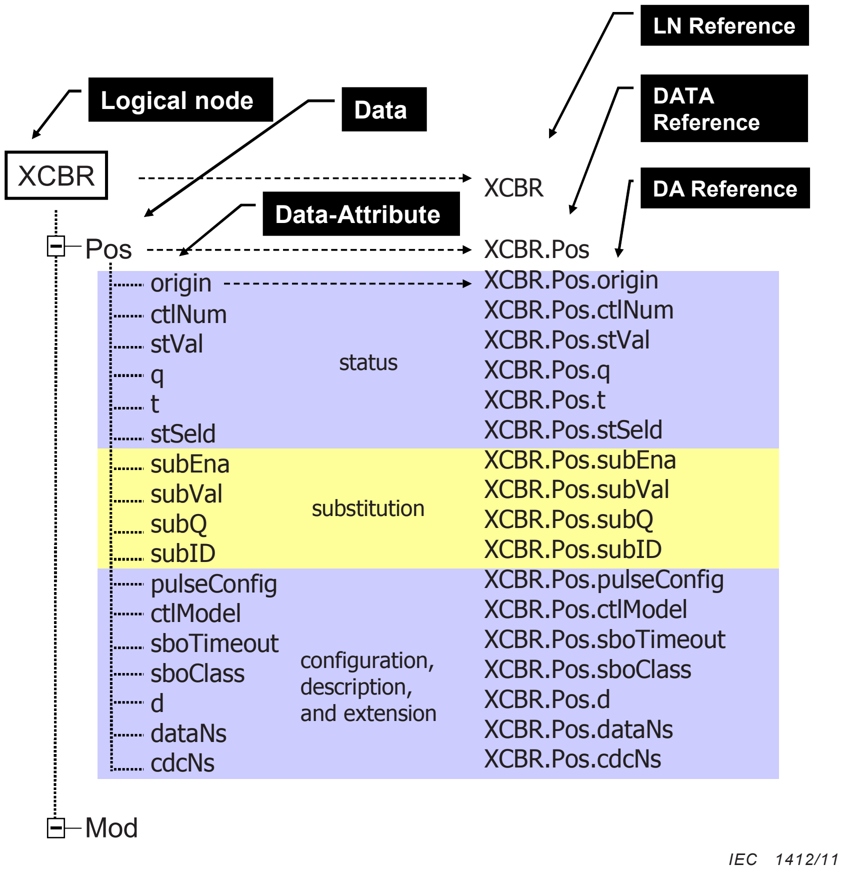


Figure 11 – XCBR1 information depicted as a tree

XCBR is the root at the level of logical nodes. The object reference XCBR references the complete tree below. XCBR contains data, for example, Pos and Mode. The data Pos (position) is precisely defined in IEC 61850-7-4 (see excerpt of the description):

Description of data

Data Name	Semantics
…	…
Pos	This data is accessed when performing a switch command or to verify the switch status or position. Possible states for position are: intermediate state I off I on I bad state.
…	…

The content of the position Pos is a list of some 20 data attributes and 7 controllable parameters. The attributes are derived from the common data class DPC (double point control). The data attributes defined in the DPC are partly mandatory and others are optional. Only those data attributes that are required for a specific application are inherited by a data object. For example, if the position does not require the support of substitution, then the data attributes subEna, subVal, subQ, and subID are not required in the data object Pos.

The information exchange services that access the data attributes make use of the hierarchical tree. The data attribute XCBR.Pos.ctlModel defines the type of control service which is supported. The status information could be referenced as a member (XCBR.Pos.stVal) of a data set named “AlarmXCBR”. The data set could be referenced by a reporting control block named “Alarm”. The report control block could be configured to send a report to a specific computer each time a circuit breaker changes its state (from open to close or from close to open).

6.3 Example of an IED composition

Figure 12 shows examples of different logical nodes being parts of IEDs. The logical nodes involved are PTOC (time overcurrent protection), PDIS (distance protection), PTRC (trip conditioning) and XCBR (circuit breaker). Case 1 shows a protection device with two functions, which are hardwired with the circuit breaker. Case 2 shows a protection device with two functions where the trip is communicated via a trip message over a network to the circuit breaker LN. Case 3 shows the two protection functions in dedicated devices, which may operate both in a fault and where the trips are transmitted as trip messages via the network independently to the circuit breaker LN (XCBR).

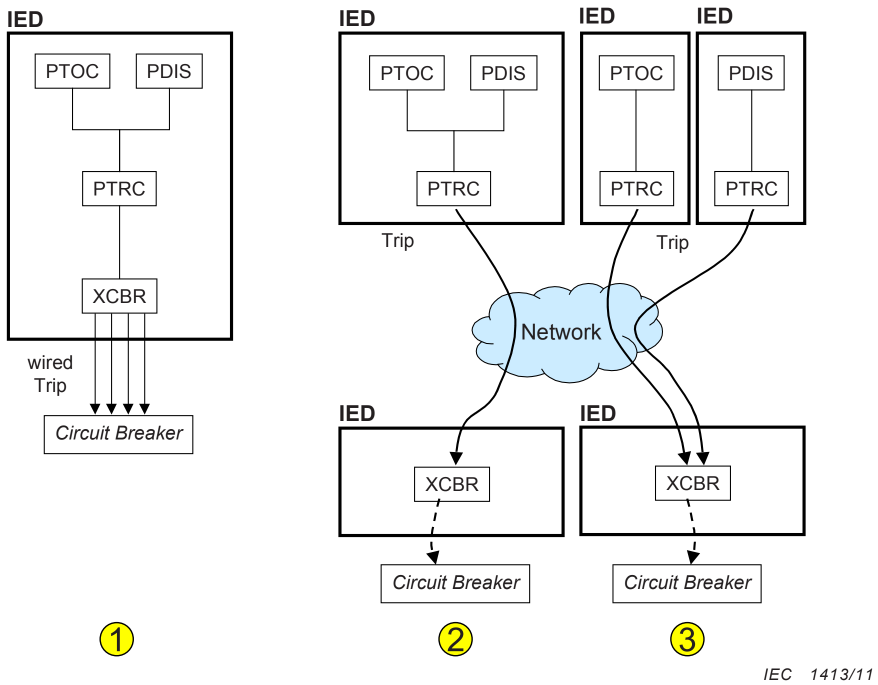


Figure 12 – Example of IED composition

In cases 2 and 3, the IED that hosts the XCBR LNs may be integrated in the real circuit breaker device or hardwired with it as in case 1, but this is outside the scope of the IEC 61850 series. The real breaker is represented for the substation automation system according to the IEC 61850 series by the XCBR LNs.

The IED composition is very flexible to meet current and future needs.

6.4 Information exchange models

6.4.1 General

The information contained in the hierarchical models of IEC 61850-7-4 can be communicated using services defined in IEC 61850-7-2. The information exchange methods (depicted in Figure 12) fall mainly into three categories:

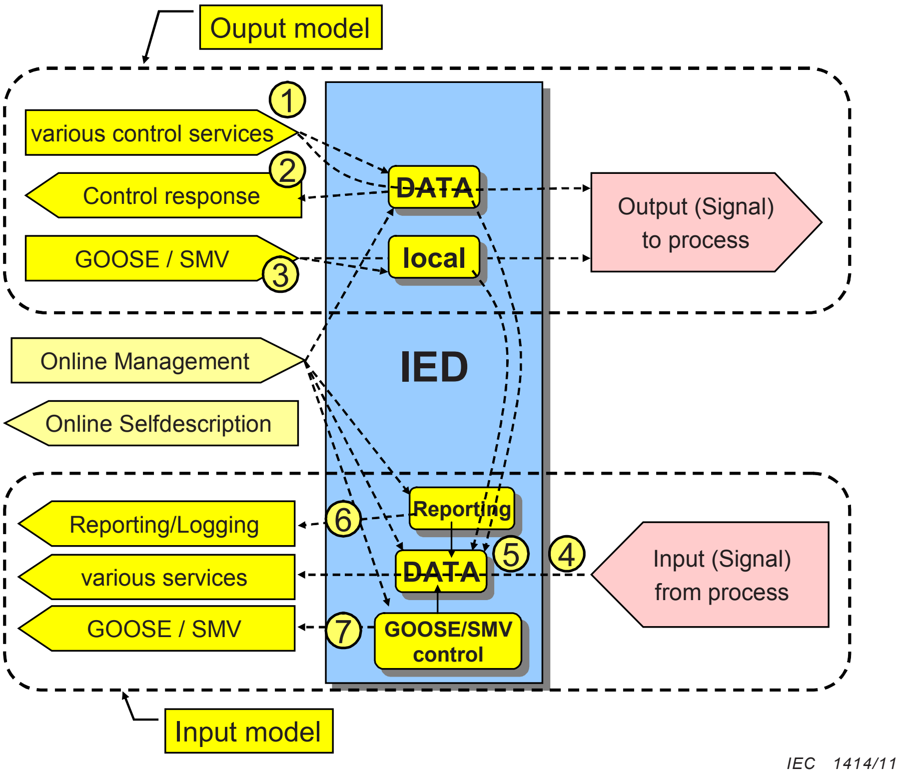
– the output model,
– the input model, and
– the model for online management and self-description.

Several services are defined for each model. The services operate on data, data attributes, and other attributes usually contained in logical nodes.

NOTE 1 Services operate actually on instances of data. To increase the readability, the term “instance of” has been omitted in most places throughout this part of IEC 61850.

Services for the output model may have an impact on an internal process only, may produce an output signal to the process via a process interface, or may change a state value of a data attribute triggering a report. If the process interface is an IED in conformance with the IEC 61850 series, this service will produce an output signal to the process directly.

NOTE 2 The terms “input” and “output” are relative to the direction from the IED to the process (output) and from the process to the IED (input).



The numbers in the circles in this figure are used in 6.4.2, 6.4.3 and related figures as references for the description.

Figure 13 – Output and input model (principle)

Several services are defined for the input model. The services communicating input information may carry information directly from the process interface or may have been computed inside an IED.

There are also several services that may be used to remotely manage the IED to some (restricted) degree, for example, to define a data set, to set a reference to a specific value, or to enable sending specific reports by a report control block. The information models (logical nodes and data classes) and the service models (for example, for reporting and logging) provide means to retrieve comprehensive information about the information model and the services that operate on the information models (self-description).

The following description of the output and input models are conceptual only. Details on the information and services involved in the models are defined in IEC 61850-7-4, IEC 61850-7-3, and IEC 61850-7-2.

6.4.2 Output model

6.4.2.1 Control model concept

The concept of the control model is depicted in Figure 14. The example is a circuit breaker logical node (XCBR) with the controllable data object XCBR.Pos (shown in Figure 15). A control service request is issued to the controllable data object. The service request contains service parameters like the control value, the originator of the request, the time when the originator sent the request and others. The data attribute XCBR.Pos.ctlModel (shown in Figure 15) indicates the type of control service to use. Before the right control service request performs the change of the position of a real device, some conditions have to be met, for example, the output can be generated only if the local/remote switch is in the "remote" position and the interlocking node (CILO) has released this operation. The chain of conditions to be met may possibly include:

– the local/remote behavior of the circuit breaker (data object XCBR.Loc) and/or local/remote behaviour of the logical device (LD) (data object LLN0.Loc),
– the control authority condition at the station level (data object LLN0.LocSta),
– the control output signal is not being blocked, either by the process or:
  - by the mode of the circuit breaker (data object XCBR.Mod) or,
  - following an external control request (data object XCBR.BlkOpn and/or XCBR.BlkCls),

  NOTE For controllable data in LNs not having specific data like BlkOpn and BlkCls, the blocked control output signal indication could be CmdBlk.
– check conditions of the device, and
– other attributes of the controllable data, for example, interlocking, pulse configuration, control model, sbo class, and sbo timeout as defined in the common data class DPC (controllable double point in IEC 61850-7-3).

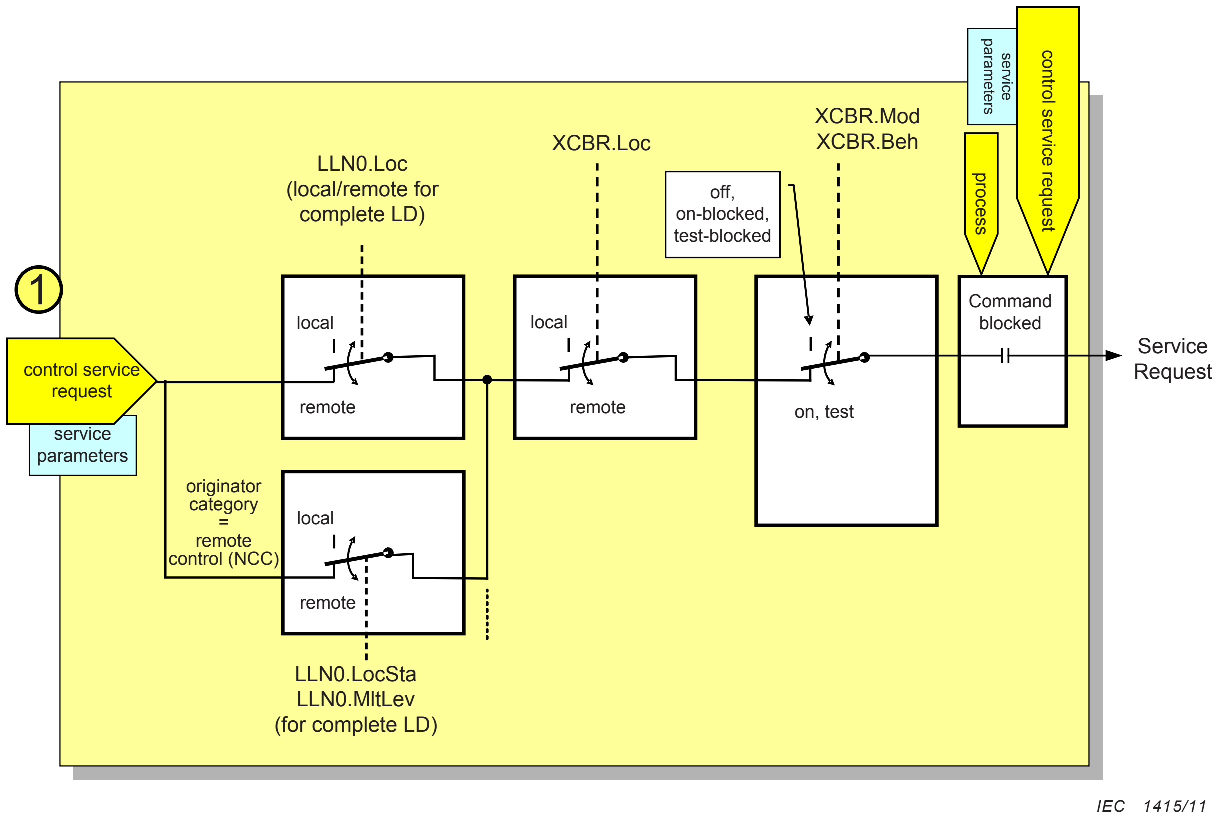


Figure 14 – Output model (step 1) (conceptual)

After all conditions have been met and all checks are positive, the output signal can be conditioned and control the real equipment (the circuit breaker – not shown).

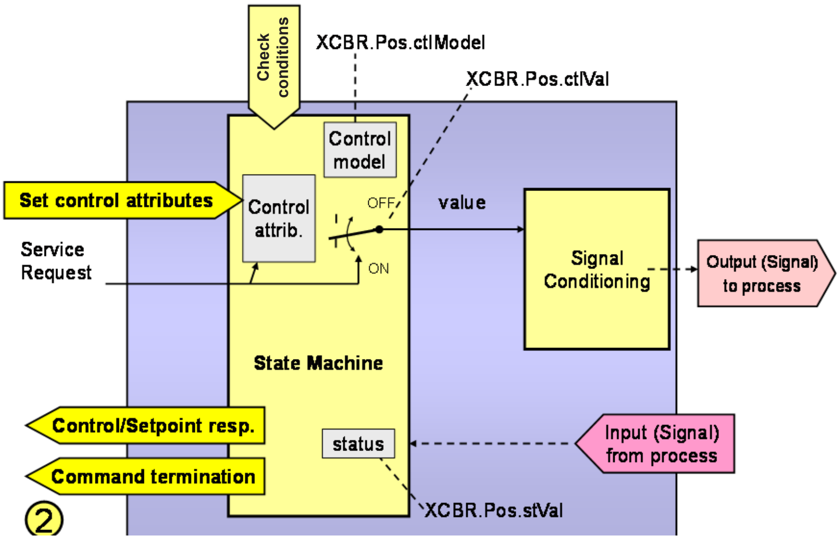


Figure 15 – Output model (step 2) (conceptual)

The state change of the real circuit breaker causes a change in the status information modelled with the data attribute XCBR.Pos.stVal. The status change issues a control service response. A command termination completes the control transaction.

6.4.2.2 GSE and SMV model concept

The generic substation event (GSE – GOOSE) and the sampled measured values (SMV), as shown in Figure 16, provides the peer-to-peer information exchange between the input data values of one IED to the output data of many other IEDs (multicast). The GOOSE and SMV messages received by an IED may be used to compute data for internal purposes also. An example for internal purposes are received switch positions to calculate the interlocking conditions locally or line current sample values to calculate the fundamental or RMS values.

NOTE 1 The GOOSE/SMV data values are defined in the input model described in 6.4.3.

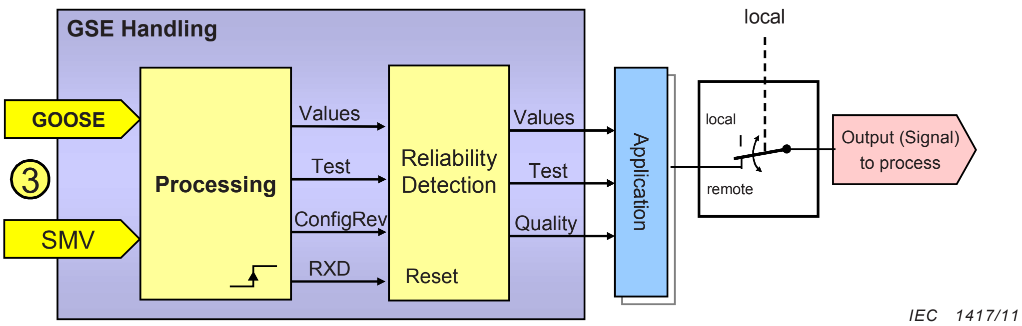


Figure 16 – GSE output model (conceptual)

The conditions to be met and the checks to run before the values are used as output signals such as interlocking are partly described within the IEC 61850 series and partly defined by the local application outside the scope of the IEC 61850 series.

NOTE 2 Many GOOSE messages may be transmitted in certain cases, for example, fault detected by a protection relay. A SCSM usually filters these messages at the data link layer to prevent flooding the IEDs.

6.4.2.3 Attributes of data and control blocks

Many data attributes of the hierarchical information model can be set with a Set-service, for example, SetDataValues and SetDataSetValues. Setting the values of data attributes is usually constrained only by the application.

The various control blocks, for example, the setting group control block (SGCB), the buffered report control block (BRCB) and log control block (LCB), have control block attributes that can usually be set to a specific value. The services to set these attributes are defined with the control blocks in IEC 61850-7-2. Setting the values of the control block attributes is constrained by the state machine of the corresponding control block.

The control blocks behave according to the values of their attribute set. The values may also be configured using the SCL file or by other local means.

All control block attributes can be read by another IED.

6.4.2.4 Setting data and setting group control block

A special treatment of output data values is required for setting data contained in several logical nodes as defined in IEC 61850-7-4, for example, the settings for the voltage controlled overcurrent protection logical node PVOC (see Figure 17). The setting data (for example AVCrv, TmACrv, TmMult, etc.) have as many values as setting groups are defined. Each setting group has a consistent set of values.

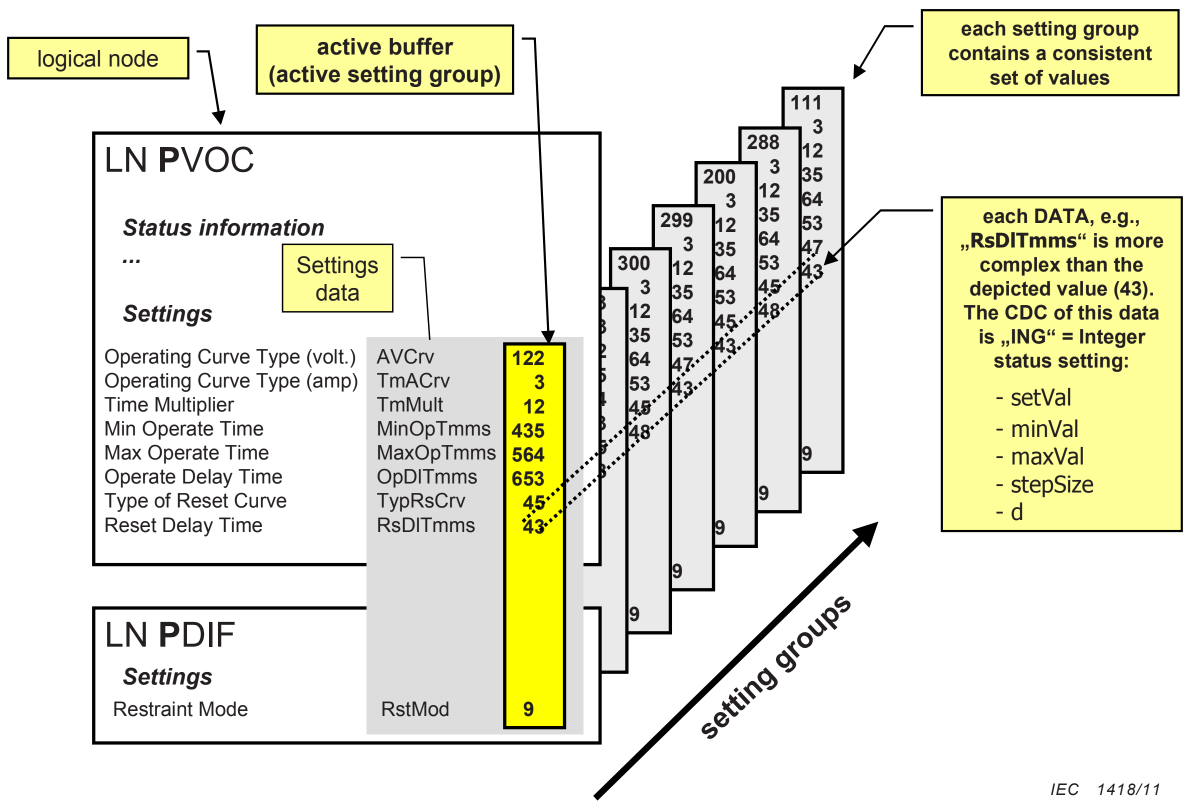


Figure 17 – Setting data (conceptual)

The values depicted are complex in the sense that each data has a type derived from a common data class. The RsDlTmms is derived from the common data class ING. The ING has several data attributes as listed in Table 3.

Table 3 – Excerpt of integer status setting

ING class					
Attribute name	**Attribute type**	**FC**	**TrgOp**	**Value/value range**	**M/O/C**
...	...				
DataAttribute					
				setting	
setVal	INT32	SP	dchg		AC_NSG_M
setVal	INT32	SG, SE			AC_SG_M
				configuration, description and extension	
minVal	INT32	CF	dchg		O
maxVal	INT32	CF	dchg		O
stepSize	INT32U	CF	dchg	1 … (maxVal – minVal)	O
units	Unit	CF	dchg		O
d	VISIBLE STRING255	DC		Text	O
...	...	...			...

The values of a specific setting group contained in the setting data can be set only if that group is in the “EDIT” state (indicated by the FC=SE; edit setting data). After all values of that group are set, the values of that group can be confirmed as containing a consistent set of values. This newly confirmed set of values can then be selected for use by the application (setting group in active state: FC=SG; active setting data).

The setVal of FC=SP means “simple” setting data (set point); applied when the setting group control model is not supported. This value can be set as a regular data attribute.

6.4.3 Input model

6.4.3.1 Input analogue signal acquisition

The concept of the input analogue signal acquisition is depicted in Figure 18. Normally, the signal is conditioned by a signal conditioner. In the IEC 61850 model, an analogue input does not exist as data before it is converted from analogue to digital. The sample rate (data attribute smpRate of a configurable data) determines how often the value shall be sampled. Alternatively, the raw digital values may be directly obtained from sample values transmitted over communication links (see also Figure 27). The method used for processing the signal (True RMS, Peak amplitude, …) may be set through the data ClcMth. In absence of the data ClcMth, the calculation method shall be considered as UNSPECIFIED, meaning UNKNOWN.

The conditions to be met before the value can be communicated (modelled as the data attribute instMag of the data, for example, a voltage of a specific phase – see Figure 18) may comprise the values of the following attributes:

– substitute/unsubstitute “switch” of the data (modelled as the data attribute subMag of the data, for example, a voltage of a specific phase),
– operator blocked or unblocked “switch”.

The result of these first steps is the “intermediate value” (still an analogue value) accompanied by the corresponding quality information.

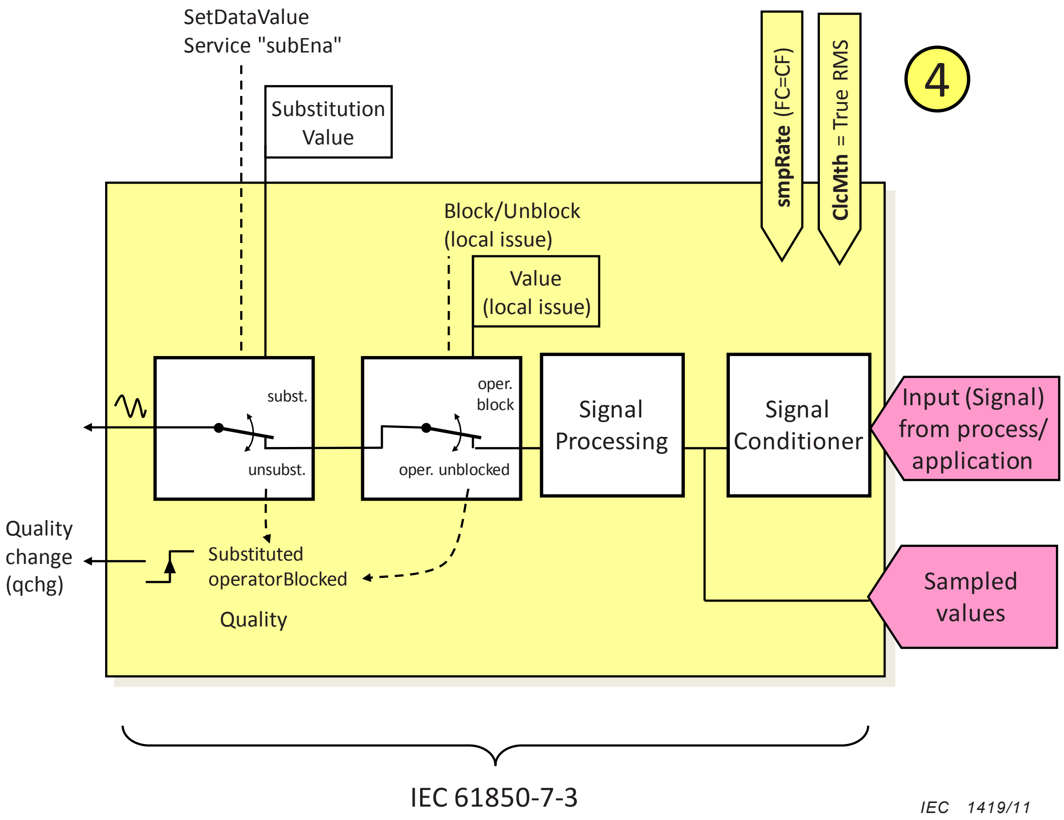


Figure 18 – Input model for analogue values (step 1) (conceptual)

6.4.3.2 Data attribute value processing, monitoring and event detection

The "intermediate value" is used for various purposes. As shown in Figure 20, the first use is to provide this value as the instantaneous data attribute value (magnitude) of the data. The data attribute has the name instMag; with the functional constraint FC=MX (indicating a measurand value). There is no trigger option associated with the instantaneous value.

The second application is the calculation of the deadbanded value, the mag value. The deadbanded value shall be based on a deadband calculation from instMag as illustrated in Figure 19. The value of mag shall be updated to the current value of instMag when the value has changed according the value of the configuration parameter db of this data.

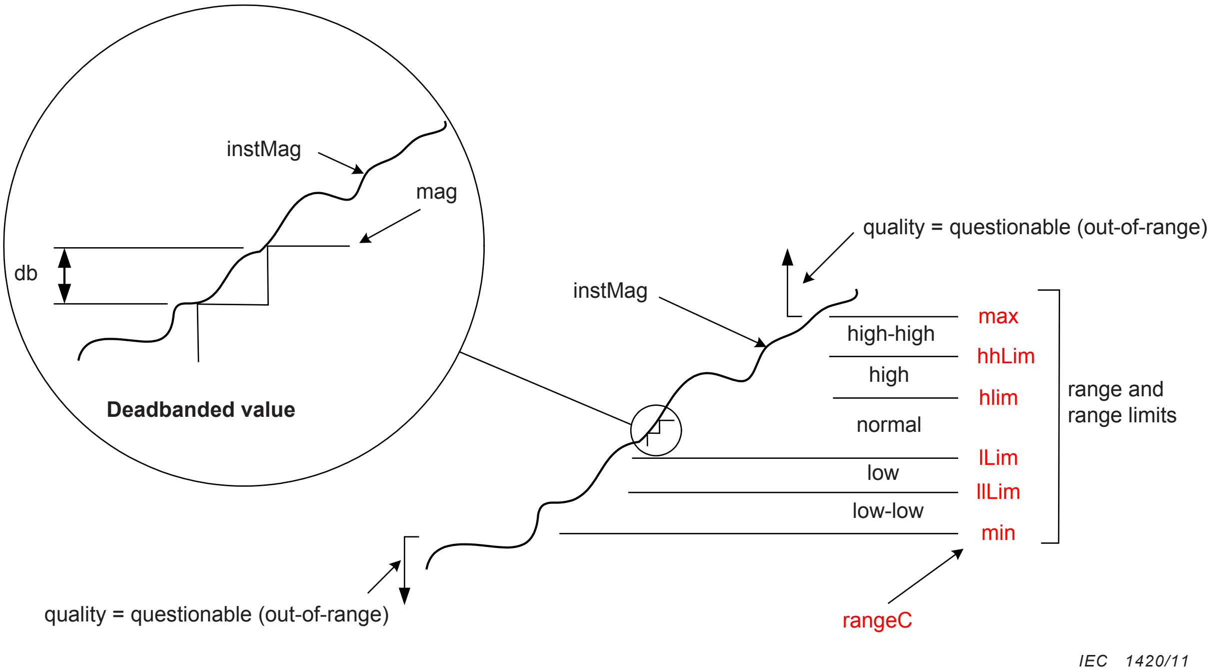


Figure 19 – Range and deadbanded value (conceptual)

The value of the deadband configuration db shall represent the percentage of difference between the max and min parameter value in units of 0,001 %.

NOTE The db value has nothing to do with the accuracy of the data defined both by the accuracy of the analogue transducer and by the accuracy of the A/D conversion.

An internal event is created any time the mag value changes. The deadbanded value mag and the event (data change – according to the trigger option TrgOp=dchg) are made available for further actions, for example, reporting or logging.

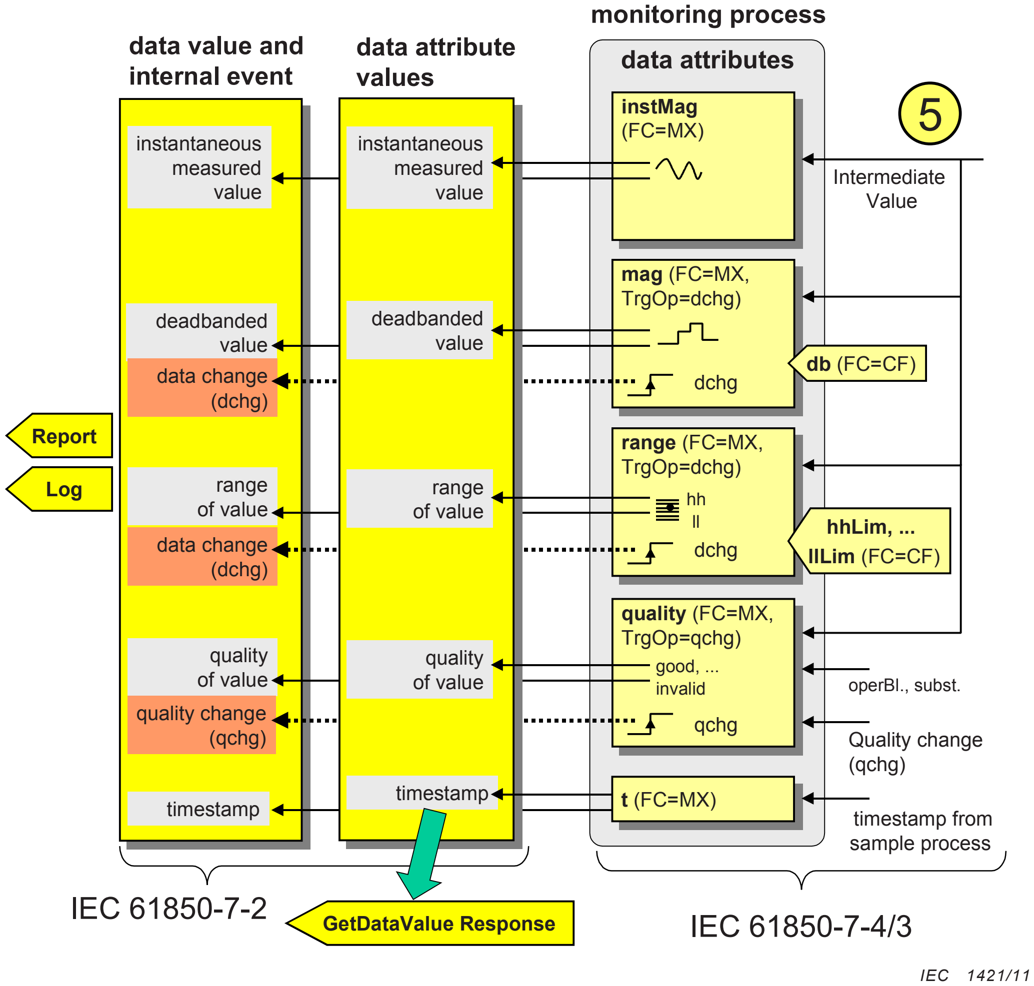


Figure 20 – Input model for analogue values (step 2) (conceptual)

A third application is to monitor the "intermediate value" to determine the current range of the value.

An internal event is created any time the instMag value crosses a range limit (see Figure 20). The range value and the event (data change – according to the trigger option TrgOp=dchg) are made available for further actions, for example, reporting or logging. The attribute rangeC is used to configure the parameters associated with the different limits.

In addition to the various values, the two attributes quality and t (time stamp) are available at any time. The time stamp is determined at the time that the value change of the data attributes mag and range has been detected. A change in the quality can be used to issue an internal event as well.

The definitions conceptually depicted on the right-hand side of Figure 19 are defined in IEC 61850-7-4 and IEC 61850-7-3. The left hand side and Figure 21 show the definitions (conceptual) found in IEC 61850-7-2.

6.4.3.3 Data reporting and logging

6.4.3.3.1 General

The internal events (process values, corresponding trigger values that caused the event, time stamps and quality information) are used as a trigger foundation for reporting and logging (see Figure 21). This information is grouped using a data set. The data set is the content basis for reporting and logging. It is also the content basis for GOOSE and SMV messages (see 6.4.3.4 and 6.4.3.5). The data set contain references to the data and data attribute values.

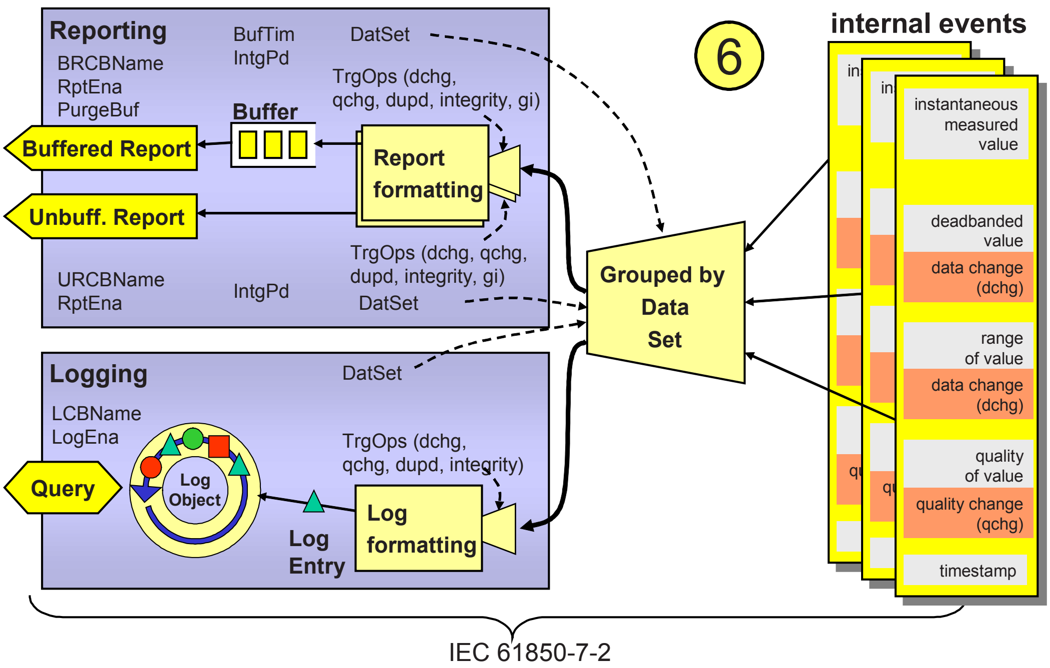
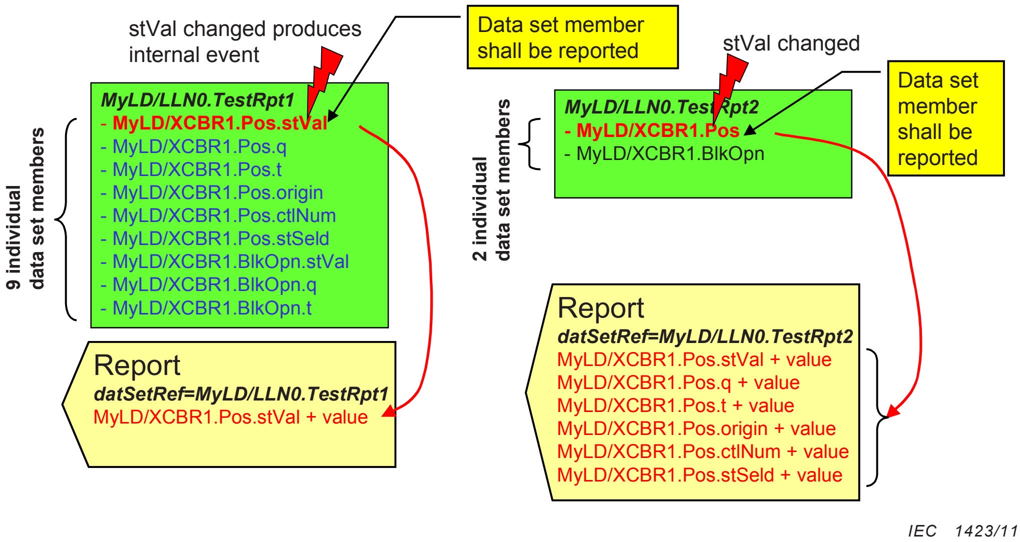


Figure 21 – Reporting and logging model (conceptual)

Which data and data attribute values are to be reported and logged is specified in the data sets. The following example explains the concept.

6.4.3.3.2 Data reporting

The data attribute stVal of the data MyLD/XCBR1.Pos (Position) in Figure 22 is referenced in two different data sets. The figure displays two different instances of data sets that reference the data attributes of the position. In the case on the left, the data set references 9 individual data set members (all of functional constraint ST): Pos.stVal is one of the nine members. In case of the change triggered by the member stVal, the value for exactly that member shall be included in the report. The data set in the example on the right-hand side has just two members. The data Pos (which has six data attributes: stVal, q, t, etc.) is one of the two members. A change triggered in the member Pos (for example, by the change in the DataAttribute stVal) shall cause the inclusion of the values of all data attributes of the data set member Pos (i.e., the complete member comprising all six data attributes stVal, q, t, etc.).



NOTE All data attributes are functionally constrained by FC=ST.

Figure 22 – Data set members and reporting

The data set specifies which data is to be monitored and reported. The next task is to define when and how to report or log the information. The reporting model provides two kinds of report control blocks:

a) the unbuffered, and

b) the buffered control blocks.

The log model has the log and log control block.

The principle characteristics of the data access methods provided by IEC 61850-7-2 are shown in Table 4.

Table 4 – Comparison of the data access methods

Retrieval method	Time-critical information exchange	Can lose changes (of sequence)	Multiple clients to receive information	Last change of data stored by	Typical client (but not exclusive)
Polling (GetDataValues)	NO	YES	YES	–	Browser
Unbuffered Reporting	YES	YES	NO	–	Real-time GUI
Buffered Reporting	YES	NO	NO	Server	Data concentrator
Log (used for SOE logging)	NO	NO	YES	Client	Engineering stations

Each of the four retrieval methods has a specific characteristic. There is no single method that meets all application requirements. During system design, the designer has to analyse the requirements and to check them against the (implemented) methods provided by a device compliant with the IEC 61850 series.

The basic buffered reporting mechanism is shown in Figure 23. The buffered and unbuffered reporting starts with the configuration of the report control blocks. The reporting starts with setting the enable buffer attribute to TRUE; setting to false stops the reporting.

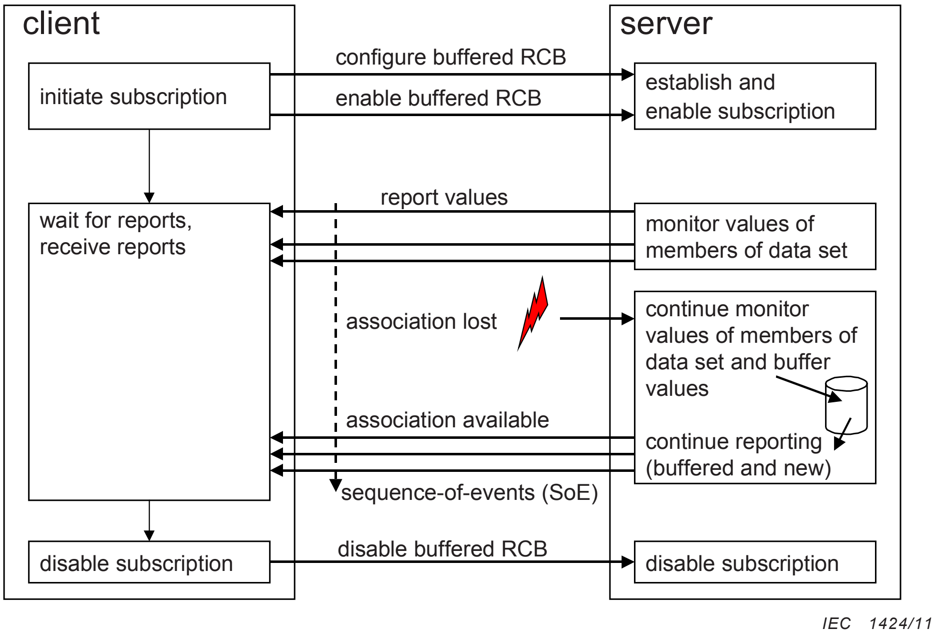


Figure 23 – Buffered report control block (conceptual)

The specific characteristic of the buffered report control block is that it continues buffering the event data as they occur according to the enabled trigger options in case of, for example, a communication loss. The reporting process continues as soon as the communication is available again. The buffered report control block guarantees the sequence-of-events (SoE) up to some practical limits (for example, buffer size and maximum interruption time).

The unbuffered report control block does not support SoE in case of loss of communication.

The buffered report control block has several attributes that control the reporting process, for example:

– RptID handle provided by the client to identify the buffered report control block,
– RptEna to remotely enable/disable the reporting process,
– DatSet references the data set whose values are to be reported,
– ConfRev contains the configuration revision to indicate addition/deletion of a member of the data set or the reordering of the members,
– OptFlds indicates the optional fields which are to be included in the report; among the optional fields are:
  - sequence-number to get the correct order of events,
  - report-time-stamp to inform the client when the report has been issued,
  - reason-for-inclusion to indicate the trigger that has caused the value to be reported,
  - data-set-name to indicate from which data sets the values have been generated,
  - data-reference to include the object references for the values,
– BufTm contains the time to wait after the first event within a data set has occurred before issuing the report (see Figure 24),
– SeqNum is the current sequence number of the reports,
– TrgOps (trigger options) contains the reasons which cause the control block to report a value into the report. The reasons for reporting may be: data change dchg, data update dupd, or quality change qchg of the data attribute in a logical node,

– IntgPd (integrity period): reporting all values initiated by the server based on this period,
– GI (general interrogation): reporting all values initiated by the client, and
– PurgeBuf set to TRUE indicates to delete all events not yet sent.
– ResvTms indicates if the the buffered report control block is reserved or not by a set of specific clients based upon configuration. The value represents the number of seconds that the reservation will be maintained after association loss.

If it is likely that – after a first event – several other events occur in the direct neighbourhood of the first event (see Figure 24), then the server can reduce the number of reports applying the buffer time attribute. Changes occurring during that time result in a report at the end of the buffer time reporting all changes (according to the reasons and according to the definition of the corresponding data set for a specific report control block).

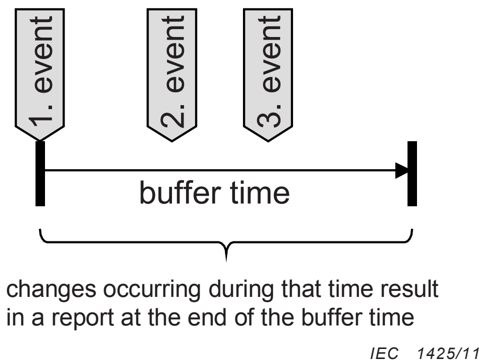


Figure 24 – Buffer time

A report allows sending just the values (according to the reason and according to the definition of the corresponding data set for a specific report control block) without any object reference of the data and data attributes. Then the object references may be retrieved out of the data set definition (see below). The report may also transmit the object references of the data and data attributes together with the data.

If firstly, no object references are sent with the values and secondly, values for a subset of members of a data set only are to be reported, then a provision is provided to determine to which members the reported values belong. The SCSM defined in IEC 61850-8-1 defines an inclusion-bitstring to indicate the member of the data set. The order of the members of the data set is as they are defined in the data set. Figure 25 shows an example.

The data set has two members in the order shown. The report with the inclusion-bitstring has two bits whose values indicate from which members the values have been derived. The first bit is TRUE thus indicating that the values in curly brackets are the values from the member MyLd/XCBR1.Pos. For the second member, no values are reported (second bit is FALSE). The report with the inclusion-bitstring optimises the length of the report message.

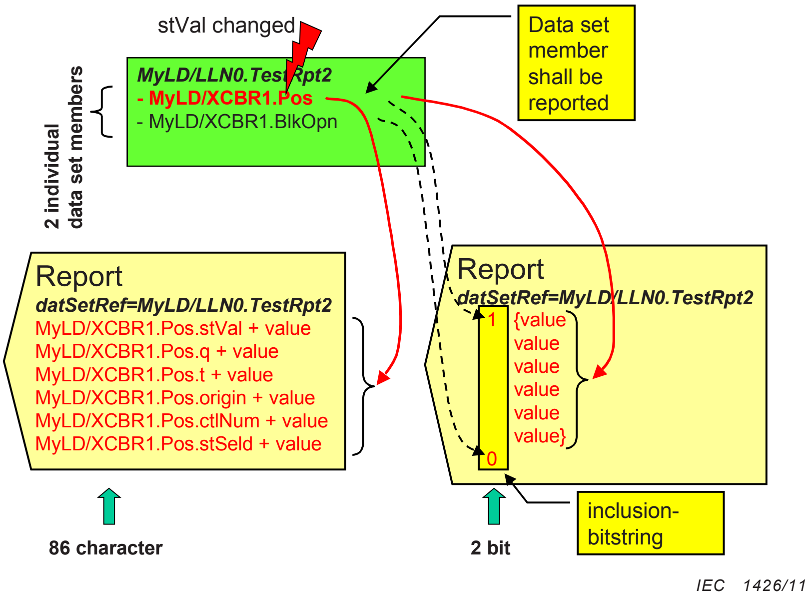


Figure 25 – Data set members and inclusion-bitstring

6.4.3.3.3 Data logging

The logging model provides a log to store values (the log entries). The log control block controls which data values and when these data values are to be stored into the log. The log is organised as a circular buffer as shown in Figure 21. The number of entries that can be stored depends on the size of log entries and on the buffer size.

NOTE Several factors have an impact on the design of a log. The system needs to be designed carefully to implement or configure the log and log control blocks in a way that meets the application requirements. Recommendations with regard to the system design are outside the scope of this standard.

Figure 26 shows an example of a log and three log control blocks. The first step is to configure and enable log control blocks. After enabling the association with that server may be closed. The log entries are stored into the log as they arrive for inclusion into the log. The logs are stored in time sequence order. This allows retrieval of a sequence-of-events (SoE) list.

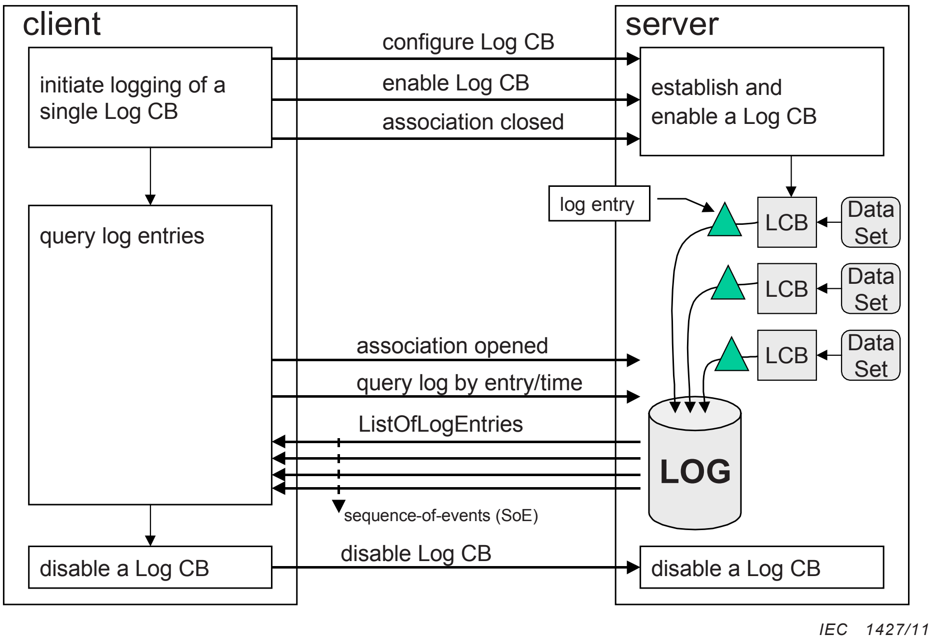


Figure 26 – Log control block (conceptual)

The log (not the log control block) is enabled at any time. The different log control blocks allow storage of information from different data sets into the log. Each log control block is independent of the other control blocks.

The log control block has several attributes that control the logging process, for example:

– LogEna to remotely enable/disable the logging process,
– DatSet references the data set whose values are to be logged,
– TrgOps contains the reasons which cause the control block to store an entry into the log. The reasons for storing a log entry into the log may be: data change dchg, data update dupd, or quality change qchg of the data attribute in a logical node,
– integrity period IntgPd: logging all values initiated by the server based on a given period,
– LogRef indicating in which log the entries are to be stored.

6.4.3.4 Peer-to-peer data value publishing

The peer-to-peer communication provides services for the exchange of generic substation events (GOOSE based on multicast) and for the exchange of sampled values (based on multicast or unicast). The GOOSE message receipt is already explained in the output model in 6.4.2.2.

Figure 27 shows the GOOSE and sampled value models.

The GOOSE model uses data values to be published grouped into data sets. All data and data attributes can be used to create a data set (for example, analogue, binary or integer values).

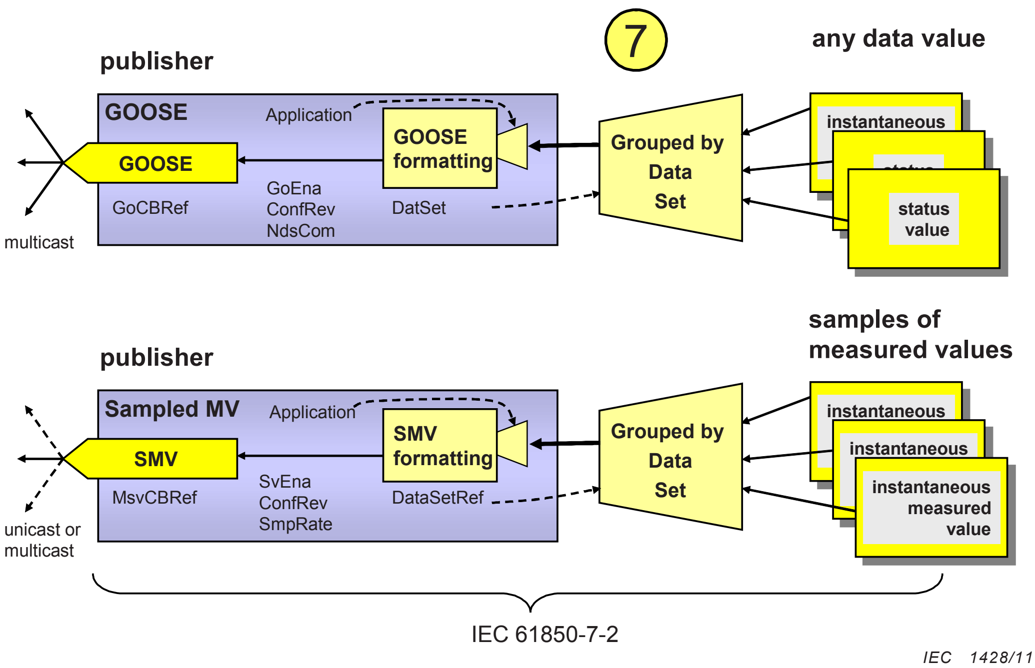


Figure 27 – Peer-to-peer data value publishing model (conceptual)

The GOOSE model has several attributes that control the publishing process, for example:

– GoEna to remotely enable/disable the publishing,
– GoID send in the message to be used as a handle for the receiving application,
– DatSet references the data set whose values are to be published,
– ConfRev contains the configuration revision to indicate deletion of a member of the data set or the reordering of the members, or changing the DatSet reference,
– NdsCom indicates in the message that some commissioning is required.

Which event triggers the publishing of values as well as how often and how fast the values are to be published is outside the scope of this standard.

6.4.3.5 Sampled value publishing

The sampled value publishing model has several attributes that control the publishing process, for example:

– SvEna to remotely enable/disable the publishing,
– MsvID sent in the message to be used as a handle for the receiving application,
– DatSet references the data set whose values are to be published,
– ConfRev contains the configuration revision to indicate deletion of a member of the data set or the reordering of the members, or changing the DatSet reference,
– SmpRate specifies the samples per unit.

6.4.4 Model for statistical and historical statistical data

In many application domains, it is required to provide additional information of a basic analogue value:

– statistical information (for example, minimum value calculated for a specified time period, for example, minimum value of last 1 h),
– historical statistical information (for example, log of minimum values of the sequence of values calculated above, for example, last 24 hourly values).

This additional information may be derived from the basic analogue values. It may be the only information provided – depending on the application requirements.

The models for the statistical and historical statistical data are explained conceptually in Figure 28. On the left hand side are the basic data representing the current values (UNSPECIFIED), i.e. some instantaneous analogue (or integer) values that are contained in the logical node instance XXYZ.

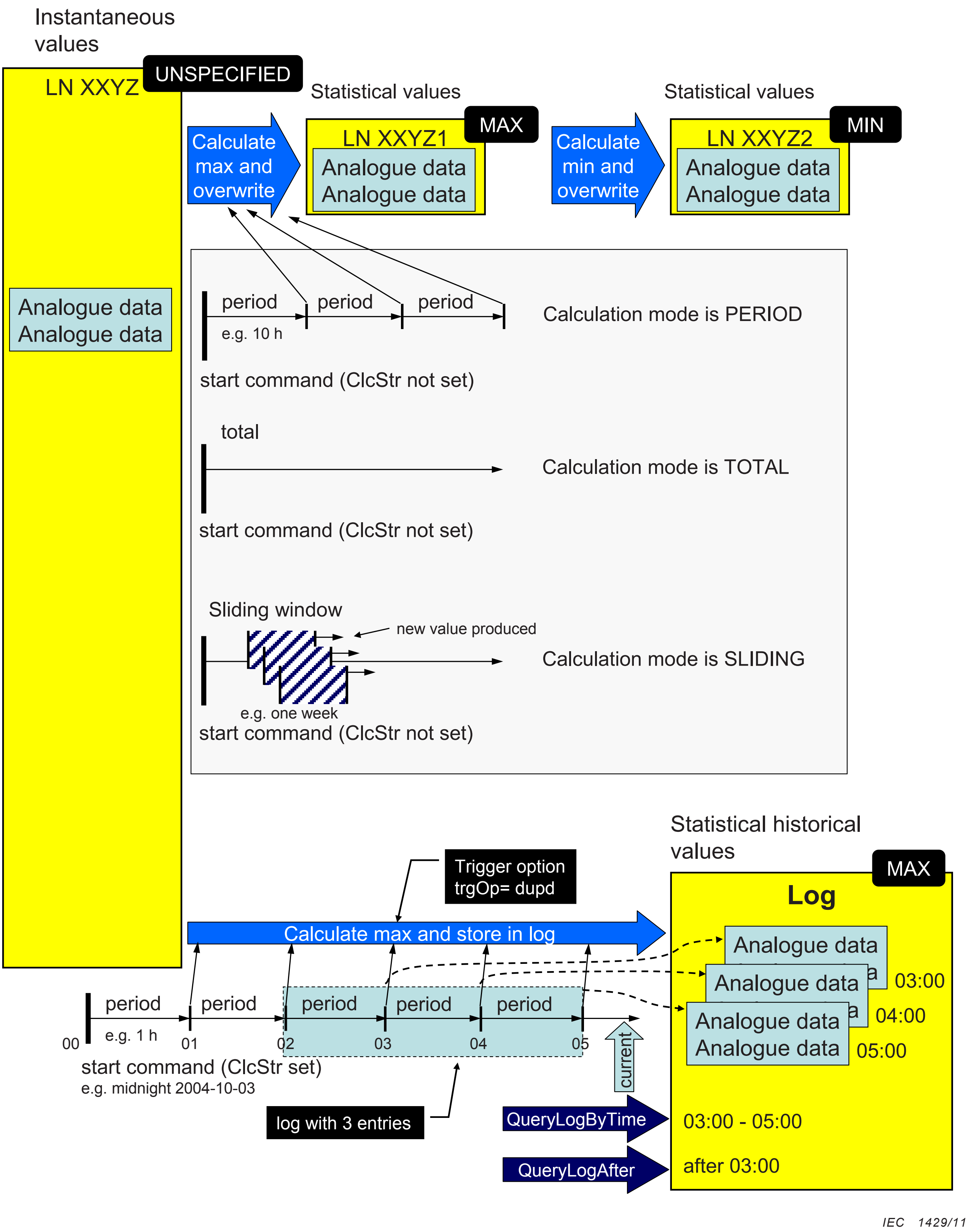


Figure 28 – Conceptual model of statistical and historical statistical data (1)

The upper half depicts the method defined for statistical values. The first example is the instance XXYZ1 of the logical node class XXYZ. The analogue values represent the calculated maximum values derived from the instance XXYZ. The logical node XXYZ1 has special setting data that indicate that the values are maximum values and that the calculation method is "periodic". The period starts after a start command. At the end of the period, the calculated maximum values of the instance XXYZ1 are overwritten by the new values.

The maximum values can be used to calculate the minimum maximum values in – of course – a much longer period than for the maximum calculation in XXYZ1. The instance XXYZ2 may represent the minimum value of the max value of the last 10 days.

Setting parameters other than PERIOD may be used to specify calculation modes. A calculation mode set to TOTAL means that the calculation of maximum values will start as soon as it has been enabled. A calculation mode set to SLIDING means that the calculated maximum values are calculated over a sliding window whose width can be set by means of a special interval type setting (e.g. hour, day, week).

The lower part of the figure shows the conceptual model of the historical statistical data. In this model, the calculated values (in this case the maximum values with calculation mode set to PERIOD) are stored in sequence in a log. The calculation in the example starts at midnight of 2004-10-03. The interval is 1 h. After the first hour, the first log entry is written. After the second hour, the second entry contains the value of the second hour. After 5 h, the log contains the values of the last three hours (intervals 02-03, 03-04, 04-05).

The statistical data model is based on the calculation of analogue values contained in other logical nodes. The Figure 29 comprises three technological logical nodes of the same type (for example MMXU). The top logical node (LN XXYZ) represents the RMS measured values. The second and third logical nodes are the statistical logical nodes, i.e., the logical nodes that represent the calculated values (LN XXYZ1 represents the MIN values, the LN XXYZ2 the MAX values).

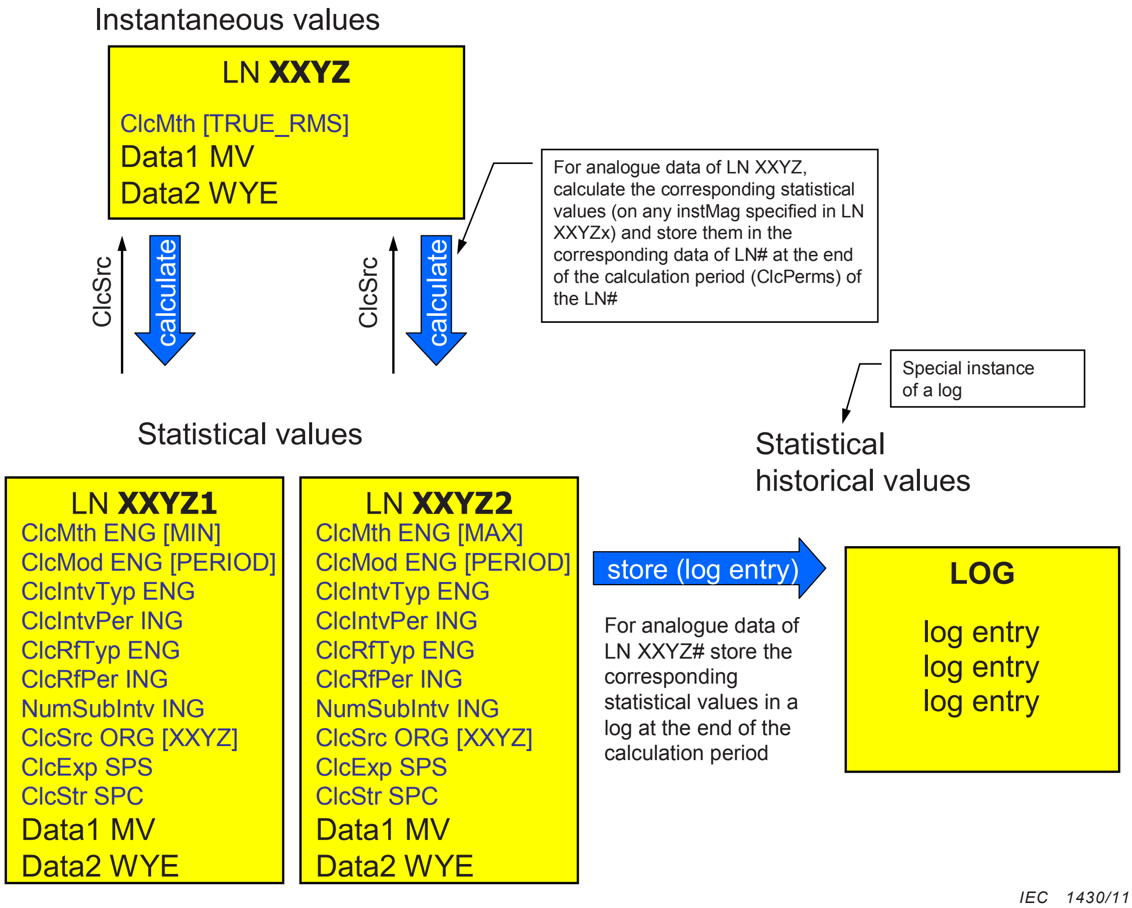


Figure 29 – Conceptual model of statistical and historical statistical data (2)

The two logical nodes on the bottom left in Figure 29 (XXYZ1 and XXYZ2) represent minimum (MIN) and maximum (MAX) values of the analogue data represented in the top logical node (XXYZ). Minimum and maximum values are defined by the setting ClcMth. The two logical nodes make use of the setting data ClcSrc (calculation source). The common data class of ClcSrc is ORG, "object reference setting group" and is used to reference the source logical node for the calculation. For both logical nodes, ClcSrc has the value XXYZ. Each logical node with analogue data can be used as a source.

Four time-related parameters are needed to define the considered statistical calculation:

– the mode of calculation (ClcMod), i.e. PERIOD, TOTAL or SLIDING;
– the calculation interval duration: data ClcIntvTyp and ClcIntvPer which are respectively used to specify the time unit and the number of units to consider for the duration window (e.g. one day);
– the calculation refreshment interval duration: data ClcRfTyp and ClcRfPer which are respectively used to specify the time unit and the number of units to consider for the duration between two updates of the calculated result (e.g. one hour);
– the calculation sub-interval (NumSubIntv), the duration step between two contiguous sliding windows.

The detailed explanation of statistical calculation basis, time interval definitions and calculation start can be found in IEC 61850-7-4.

The settings described before can be used to control the behaviour of the logical node . For periodic calculation, the “event” ClcExp set to TRUE can be used as an event to report the new value (the statistical value) by the report control block or it may be logged as historical statistical data for later retrieval.

NOTE The data names of the “Data” in all logical nodes shown in Figure 29 are the same, i.e., in all three logical nodes. The data are contained in different logical node instances (XXYZ, XXYZ1, and XXYZ2). These result in the following references: XXYZ.Data1, XXYZ1.Data1, and XXYZ2.Data1.

6.4.5 Model for system functions

6.4.5.1 General

System specific information is modelled using logical nodes from the group L. This subclause gives a short description of the functions intended to be modelled by those logical nodes. All “L” logical nodes, except LPHD and LLN0, belonging to an IED shall be in the same logical device (see 8.2). There shall be only one such LD per IED.

6.4.5.2 Time synchronization

The time synchronization model shall provide accurate time to all IEDs in a power utility system for data time stamping with various ranges of accuracy, e.g. millisecond range for reporting, logging and control and microsecond range for sample values.

The different components of the time synchronization model are:

– an external time source with a known level of accuracy (for example, GPS),
– a time server providing the internal time source for all IEDs in the substation,
– a time synchronization protocol providing synchronization between IEDs (for example SNTP for IEC 61850-8-1),
– the time stamp semantics used for the information exchange defined in the ACSI,
– the presentation of the time stamps according to the SCSM,
– the server and clients that need substation-wide synchronized time.

The chosen reference for time stamping is UTC. Using UTC guarantees that time evolution is continuous and is independent of abrupt changes due to daylight savings time (DST). This is very convenient for applications which have to sort events by date and time.

6.4.5.3 Time management

At the user level, it is generally necessary to display local time according to the time zone and DST. The standard considers this translation a local application issue. It is also needed to manage the two moments when DST changes. Time management functions are modelled by means of the logical node LTIM. LTIM provides data objects (settings) in order to read or change the local time parameters depending on the IED capabilities. If an IED is required to display information using local time but does not internally manage any time zone table, this can be handled by a central IEC 61850 client in the system which knows the time zone table parameters.

6.4.5.4 Time supervision

Time supervision is the function responsible for communicating the status of the time synchronization function of an IED. The logical node LTMS has been introduced for that purpose. It allows to know important conditions with regard to time synchronization as the status of the communication channel with the time source or the time accuracy class according to IEC 61850-7-2.

6.4.5.5 GOOSE and SMV supervision

The LGOS and LSVS logical nodes may be used to monitor subscription states to GOOSE or SMV signals. They contain mandatory information like status of the subscription (active, not active) and other optional information e.g. the source GOOSE or SMV control block identification.

6.4.5.6 Service tracking

Service tracking is represented by the logical node LTRK and is defined as the function in charge of recording, after a service execution, the parameters values used by any service defined in IEC 61850-7-2. It is thus possible to read, report or log these values for system behaviour analysis.

Figure 30 shows an example of a logical node LTRK used to track the execution of control services. In the example, an operate service with the value close is successfully executed on the object XCBR.Pos, i.e. the position of a breaker[2]. The service uses the parameters defined in IEC 61850-7-2 and is tracked by the logical node LTRK which contains the data class DpcTrk. The data class DpcTrk is used to record the last applied parameter values on any controllable object within the IED whose CDC is of type DPC (controllable double point).

The logical node LTRK contains data classes needed for any service defined in IEC 61850-7-2. The data classes are based on common data classes (CDC) also defined in IEC 61850-7-2. In the example, DpcTrk contains attributes inherited from the CDC CTS, common service tracking common data class. These attributes are common to all service tracking data classes. DpcTrk also contains attributes which are specific to the control service tracing data classes. The attributes have the functional constraint FC=SR.

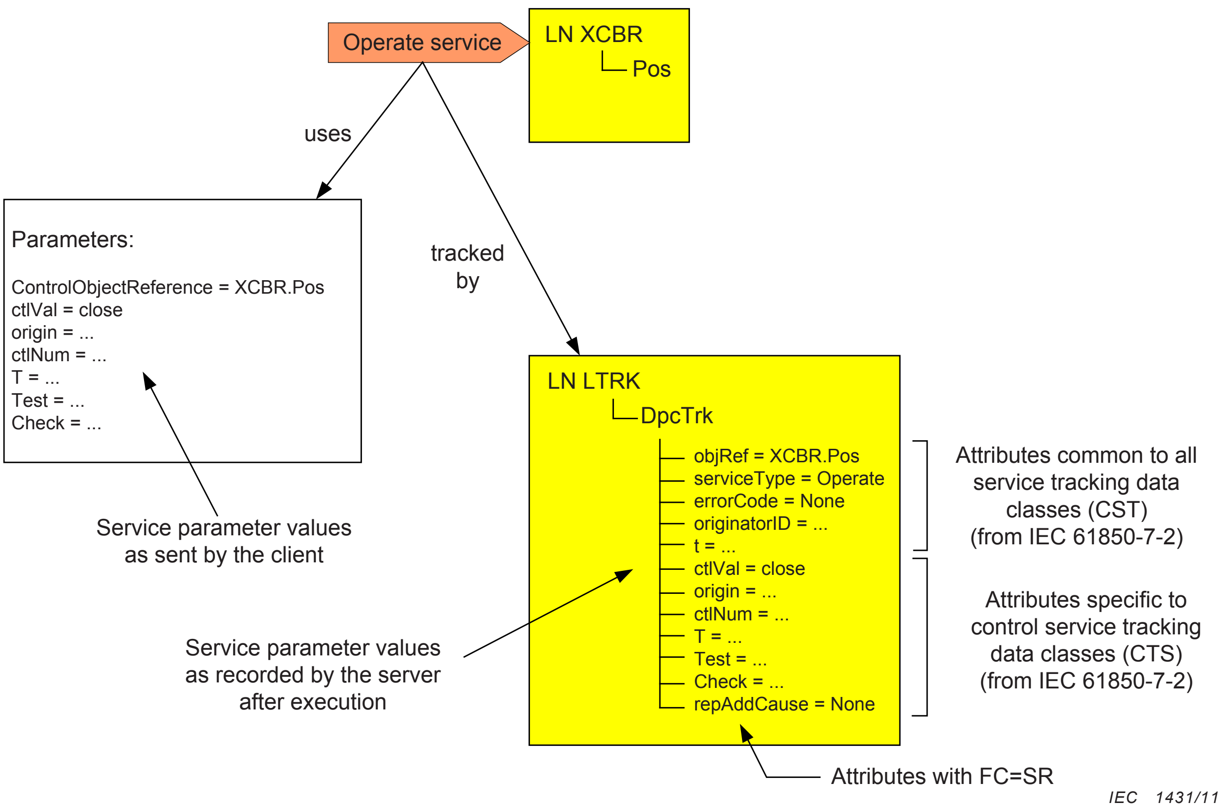


Figure 30 – Concept of the service tracking model – Example: control service tracking

---

2 The example shown in Figure 30 is conceptual and XCBR.Pos is used for convenience as the control object reference, which is incomplete. The control object reference of an instance shall contain the complete reference to an objectname including the logical device name as explained later in this document.

7 Application view

7.1 General

A sample operation, illustrated in Figure 31 is to switch a circuit breaker. An operator at a remote HMI wants to remotely switch the circuit breaker. The HMI computer and the circuit breaker have to operate together (interoperate). First, the computer needs to know what information it has to transmit to the IED representing the circuit breaker (normally called the “process interface”). Secondly, it has also to know the name of this IED (for example “Circuitbreaker1”) and how to address the IED. Both the HMI computer on the left side and the IED “circuitbreaker1” on the right side are connected to a common communication network. The HMI sends a control command to the “Circuitbreaker1” to switch the position of the breaker (close the breaker). After switching is completed, the interface IED may (if configured) send a report to the HMI computer indicating that the switch position has changed.

Different users may name the circuit breaker differently: one may use “Circuitbreaker1”, another may chose “CBK-2”. IEC 61850-7-4 based on the approach described in IEC 61850-5 standardises many abbreviated names for substation functions and related equipment. The standardised name for a circuit breaker is XCBR. This name may be accompanied by a suffix and a prefix: "Q1XCBR1" (for naming conventions, see 12.4, Clause A.1, Clause A.2 and IEC 61850-7-2).

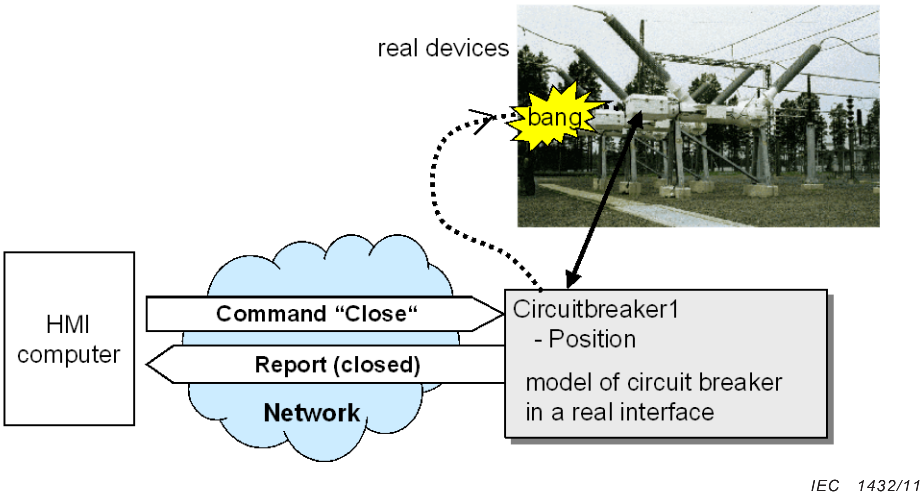


Figure 31 – Real world devices

Applications in the computer on the left-hand side of Figure 31 may also query information about the:

– real physical circuit breaker (nameplate, health, ratings, etc.),

– real device (IED) that hosts the process interface (nameplate, health, operation mode, etc.),

– behaviour of the reporting services that determines the transmission of status reports.

In addition, the operator (or the computer if it is in some type of automatic mode) may change the active setting group of a protection function to a different setting group, may configure the reporting behaviour remotely, or may request the substitution of a fixed value for a process value. Alternatively, the operator may want to receive a sequence of events.

All these and many other functions supported by the controller have three major aspects in common that are standardised in the IEC 61850 series:

– what functions and what information is network-visible and how are they named and described (see IEC 61850-7-4, IEC 61850-7-3, and IEC 61850-7-2),
– how functions can be accessed and how (more generally speaking) information can be exchanged (see IEC 61850-7-2), and
– how devices can be connected to communication networks (IEC 61850-8-x and IEC 61850-9-x series).

IEC 61850-7-4 comprises a list of more than 2 000 named and well-defined information elements which enable creation of the information model of a real substation device (for example, complete power transformer, circuit switch, or measuring unit). This static information model inherits the type information from IEC 61850-7-3 and the required (dynamic) communication services from IEC 61850-7-2. Functions in the context of the IEC 61850-7-x series are those that are needed to exchange all information of the information model in the manner that is required for the substation domain or other domains within the scope of IEC technical committee 57. Functions (for example, substation functions like bus bar protection, or point of wave switching) make use of the data and functions provided by IEC 61850-7-x series.

From the point of view of the IEC 61850-7-x series, the internal processes involved to accomplish functions represented by logical nodes within an IED are outside the scope of IEC 61850-7-x series. Only the interactions between logical nodes located in different IEDs have to use the defined services and data. Examples of interaction of logical nodes for complex functions such as synchronised switching including the basic sequence of exchanged messages can be found in IEC 61850-5.

The dynamic behaviour of a real device is established through the configurable attributes of the implemented information model, and by changing its (changeable) values. The effects resulting from any change of value in the information model is defined in the standard. As a result of a control of the “Circuitbreaker1”, the real circuit switch opens or closes. The controller may immediately send a report (value, quality, and timestamp) to the initiator and may additionally write this event in the log of the device for later retrieval. Various dynamic behaviours of the controller may be pre-configured by an engineering tool. The behaviour may be changed by configuring the controller using specific services for remote configuration (set), for example, reporting values can be remotely enabled or disabled.

7.2 First modelling step – Logical nodes and data

IEC 61850-7-4 defines a list of tens of logical nodes. Examples are circuit breaker (abbreviated "XCBR") and distance protection ("PDIS"). Each logical node (as illustrated in Figure 32) is composed of several data that represent some application-specific meaning (see Annex A for an overview of logical nodes and data).

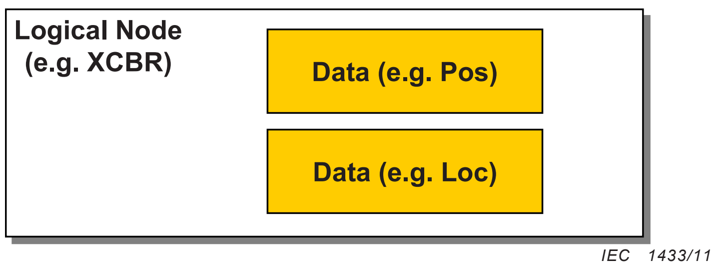


Figure 32 – Logical nodes and data (IEC 61850-7-2)

The data Pos as part of a circuit breaker is used to control the position and to report the status of the position; the data "Loc" stands for “local operation” and is a status indicating that the operation of the breaker from a remote location is enabled or not.

These data constitute the basis of most information exchanges over the network. Most interactions with a device are through data in logical nodes and services. What type of application information a specific data represents is defined in IEC 61850-7-3 (common data class), for example, double point status or measured value. Each common data class has services assigned to it that define the possible services that are allowed to be operated on this data. Some information may be writeable and readable while other information may be readable only. The so-called functional constraint (FC) defines this characteristic for each information of a specific data class. The information of a data is defined to be mandatory or optional. All services (for example, GetDataValues, Operate) are defined in IEC 61850-7-2.

The logical node names (for example XCBR for circuit breaker) and the names of data (for example Pos for the position of the real switch) define the standardised meaning (semantic) of the substation device. These abbreviated terms are standardised names that are used for communication (independent of the communication system used). The information model comprises many logical nodes, data, and data attributes.

This model is also used as a basis for the already mentioned substation configuration language (SCL) according to IEC 61850-6. The substation configuration describes which of the optional information is used in a specific device, what the instance names of all logical nodes are, what communication links exist, what the relation of the IEDs to the single line diagram is, and all information which is needed for the system engineering. The instance inherits everything from its class and assigns a unique name to it.

This standard makes use of a hierarchical organisation of data. Figure 33 shows an example of a real device “BayUnit”, protection function functions such as “Distance protection” (PDIS) and “Time Overcurrent” (PTOC) and “Trip conditioning” (PTRC). The process data of these basic functions, the basic functions and other important aspects of the bay unit are modelled as data in a tree-structure. Each element of this tree is a data: the data at the top level is “Bay Unit” which contains “Distance protection”, “Time Overcurrent” and “Trip conditioning”. The “Distance protection” contains, for example, the data “Start” (Str) with different attributes such as “general” (general) and “Phase A” (phsA).

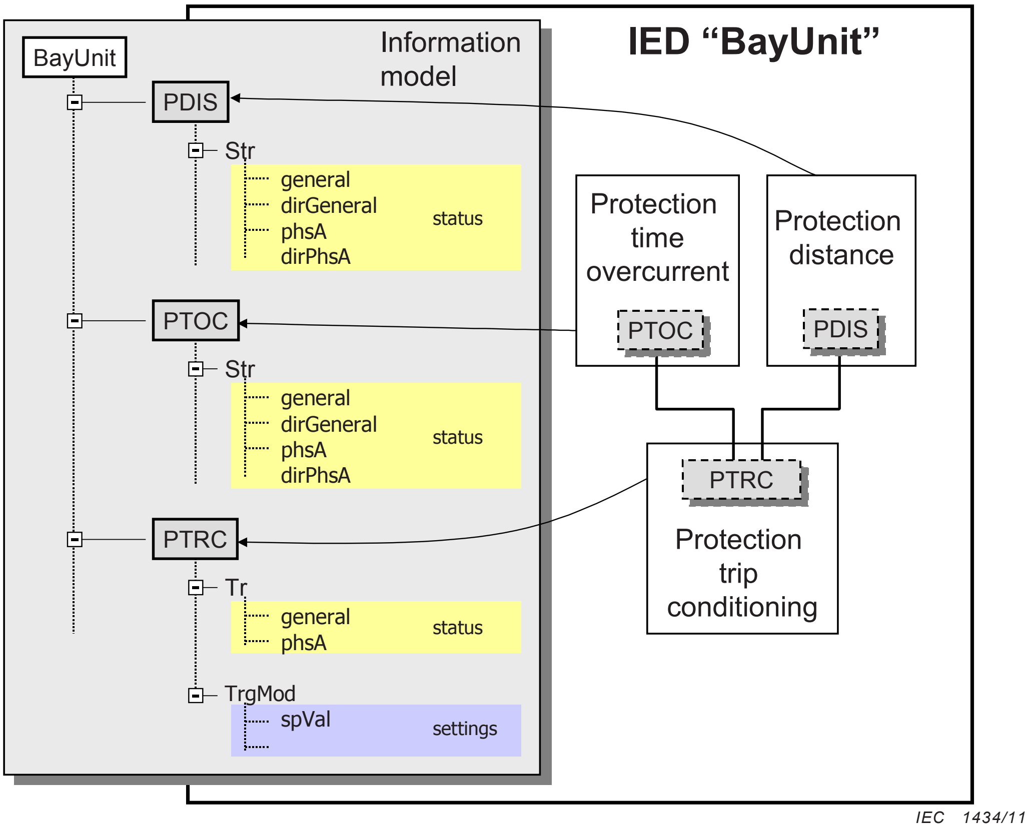


Figure 33 – Simple example of modelling

The elements of a logical node are illustrated in Figure 34. A control service represents the ability to control something in a device. This is modelled as data. To reset, for example, all LEDs in a device, only the value of the data "LEDRs" has to be set to True. Data can be grouped into data sets and be reported immediately or logged for later query.

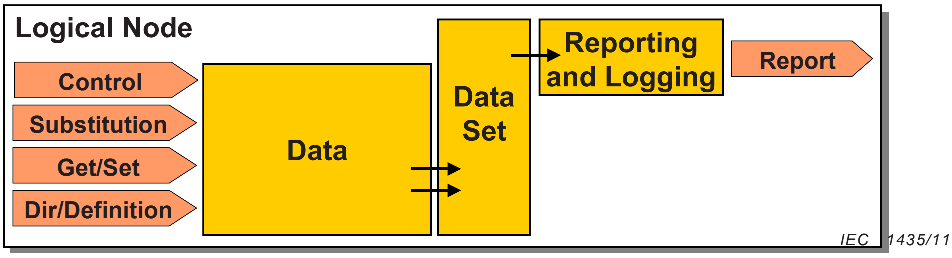


Figure 34 – Basic building blocks

Control and report build one part of the interface of a logical node. Other services that operate on data are: substitution, for replacing a data value with a fixed one; get and set, for reading and writing data and data set values; Dir and Definition (GetDataDirectory, GetDataDefinition) retrieve the directory information of a data instance and the definition of a data instance. From an abstract viewpoint, the interfaces of a logical node can be summarised as illustrated in Figure 35. The services can be understood as carrying the information defined by PICOMs (Pieces of communication) as introduced in IEC 61850-5.

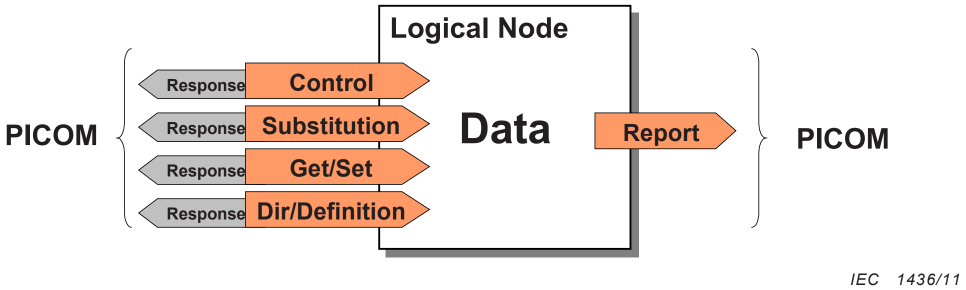


Figure 35 – Logical nodes and PICOM

Logical nodes and data (contained in logical nodes) are the fundamental concepts that are used to describe real systems and their functions. Logical nodes function mostly as a container for data and can be placed anywhere in an IED. Each data defined in IEC 61850-7-4 has a specific meaning assigned to it. The data interact with their environment through their services. The concepts of logical nodes and data in the IEC 61850-7-x series defines the information that can be accessed in a logical node. The device that issues for example the request to retrieve data from a logical node can be modelled as a logical node, too. The information flow can be viewed between logical nodes (see Figure 36 and 9.5).

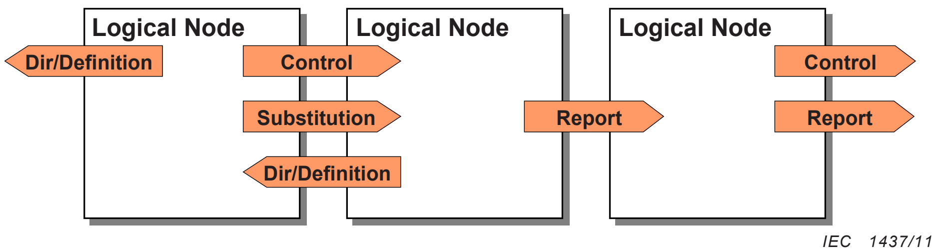


Figure 36 – Logical nodes connected (outside view in IEC 61850-7-x series)

From this point of view, the information flow is abstracted from any communication related information, for example, request/response notation.

Further building blocks and their services are explained in Clause 8.

Domain experts, for example of switchgear devices or power transformers, should primarily read and understand the logical nodes for switchgear devices (XCBR, XSWI, etc.) or for power transformers (YPTR, YLTC, etc.) and the data that belong to these logical nodes as defined in IEC 61850-7-4. IEC 61850-7-3 needs to be understood to see all the detailed information that may be exchanged with a device.

7.3 Mode and behaviour of a logical node

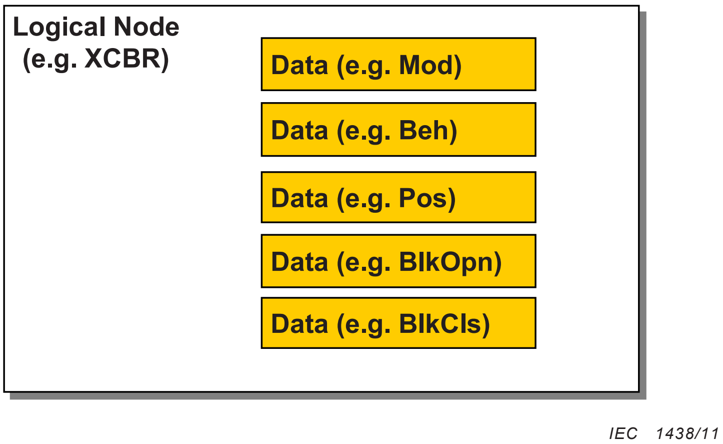


Figure 37 – Mode and behaviour data (IEC 61850-7-4)

Mod and Beh are data which carry specific meanings in the context of this standard. In Figure 37, Mod is a controllable data representing the current operation mode of the circuit breaker logical node (on, on-blocked, test, test/blocked, off).

Switching the Mod to different states has an impact at the application level. However, the change of the Mod value has no effect on the communication related functions like reporting and GOOSE messages. The behaviour of those functions shall be handled by the appropriate services defined in IEC 61850-7-2.

Beh is a non-controllable data and has the same values as Mod. It represents the combination status of the mode of a logical node and the mode of the containing logical device (see 8.2).

The main requirements with regard to the different states of Mod/Beh (e.g. test) are defined in IEC 61850-5. The detailed requirements and application examples can be found in IEC 61850-7-4.

7.4 Use of measurement ranges and alarms for supervision functions

The logical nodes from the S group, Supervision, may be used to monitor physical parameters of a primary device. Each of these logical nodes has the following data :

– one or several measurements (usually from the common data class MV) of the physical parameters,
– one or several alarms related to the measured process.

The measurement values are used to monitor a trend. The range of values (see 6.4.3.2) gives an indication of the limits the values are crossing. These limits only have a generic meaning as their values “high, low, high-high…” may have different meanings depending upon the type of measurement. For example, “high” may mean “take an action” or just “wait and see”.

On the other hand, the alarms give immediate information as to when some fast changes occur. Moreover, these alarms have specific names, like InsTr (insulation trip for device isolation), in order to standardize the semantic meaning of process variable supervision.

7.5 Data used for limiting the access to control actions

Control actions may be performed by an operator at different functional levels in the substation: station level, bay level and process level. As described in IEC 61850-5, the logical nodes including those involved in control actions may be allocated to different functional levels.

In order to limit the access to control actions at any level inside the substation and also from outside the substation, special data shall be used in controllable logical nodes (see Figure 38). For example, a breaker function (logical node XCBR) shall contain the mandatory data Loc indicating that the control authority for this function has been switched from remote to local by certain means. The logical node XCBR may also contain the data LocSta indicating that the control authority has been switched from the station level to an upper level, e.g. the NCC level. The data LocKey is intended to be used to signal the state of a physical local/remote key or toggle switch. As an example, the position of a key switching simultaneously several logical nodes in local mode, e.g. at the entire IED level, will be indicated by the data LocKey in the logical node LLN0. MltLev is a setting which allows for the choice of different modes of switching authority. MltLev is intended to be applicable to a whole logical device (see 8.2.1) and therefore shall only be present in the logical node LLN0. The details concerning the values of MltLev and the related checking conditions a real device shall perform when receiving a control request can be found in IEC 61850-7-4.

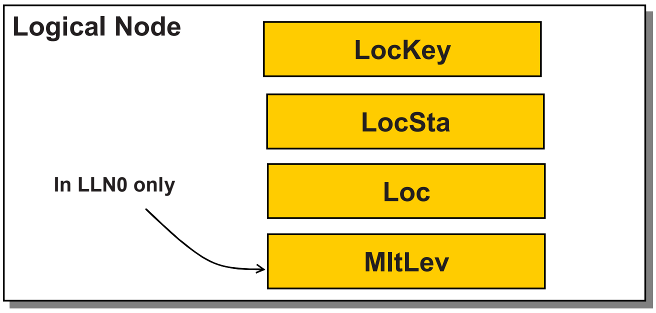


Figure 38 – Data used for limiting the access to control actions (IEC 61850-7-4)

7.6 Data used for blocking functions described by logical nodes

A specialized LN input is described by the data BlkRef defined in the common logical node class (see IEC 61850-7-4). BlkRef can be used to reference an incoming blocking signal. A typical application is the dynamic blocking of the operation of a protection function or a recloser function. BlkRef is associated with the data Blk intended to inform about the blocked status of the function represented by the concerned LN.

7.7 Data used for logical node inputs/outputs blocking (operational blocking)

7.7.1 General

It may be required by an operator to manually block inputs/outputs of a function. For example, it shall be possible to block incoming signals having an effect on control actions or on process output activation. As shown on the left side of Figure 39, some of those blockings are realized by the use of control services (see IEC 61850-7-2) and dedicated data defined in the common logical node class (see IEC 61850-7-4). Other blocking mechanisms use the “Set” service and dedicated data attributes as shown on the right side of the figure.

s

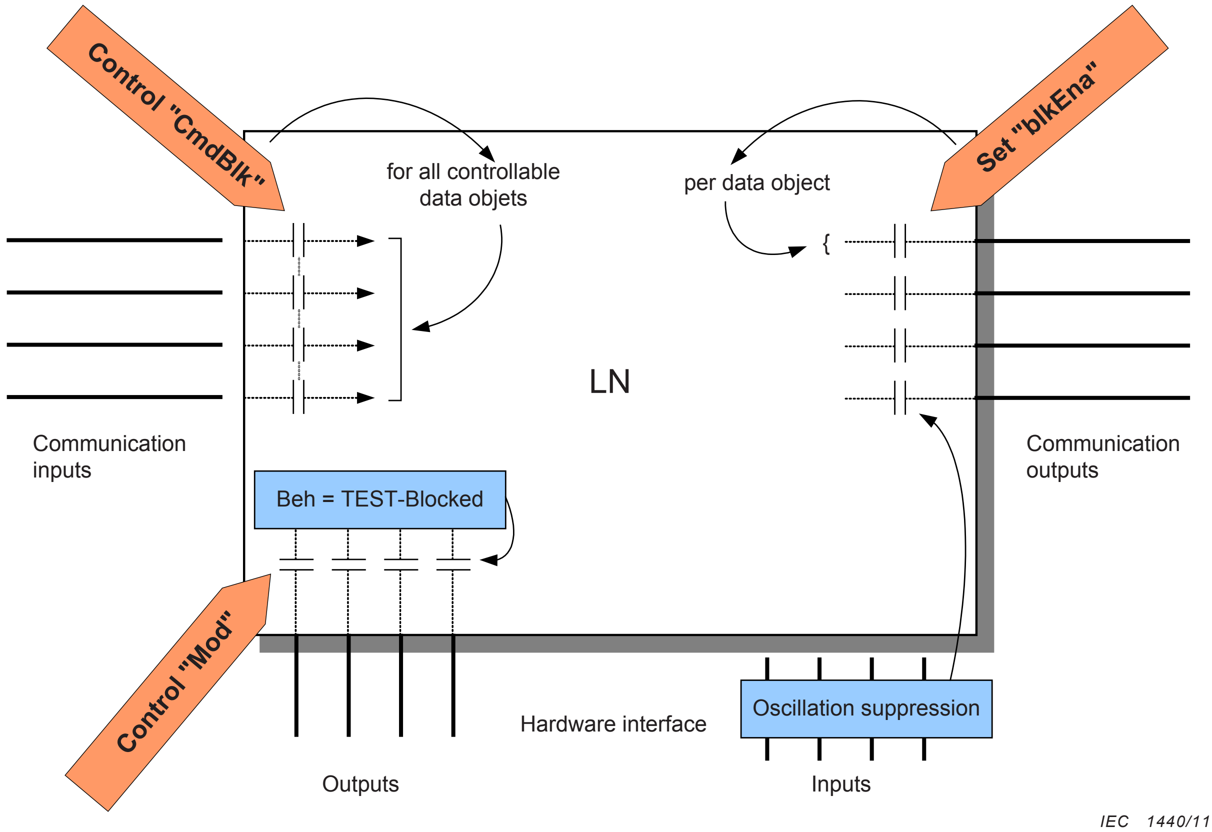


Figure 39 – Data used for logical node inputs/outputs blocking (IEC 61850-7-4)

7.7.2 Blocking incoming commands

The data CmdBlk (command blocked) is used to block incoming commands at LN level. It shall be applied to controllable data of any LN, i.e. to the common data classes SPC, DPC, INC, etc. When the blocking is activated, the command shall not be accepted and an appropriate control negative response shall be sent to the client who initiated the command.

The data CmdBlk can also block commands coming in via GOOSE messages (e.g. the data OpOpn/OpCls in the LN CSWI), thus disabling any action which could activate the outputs to the process.

The data BlkOpn/BlkCls in the LN XCBR are similar to the data CmdBlk but have a more precise semantic attached to them.

The data CmdBlk shall have no effect on the controllable data Mod.

7.7.3 Blocking process outputs

The data Mod (Mode) is used to put an LN in different modes. The mode TEST-Blocked (Beh=TEST-Blocked) shall be interpreted as the mode which explicitly disables any physical outputs having an effect on the process. For example, when XCBR.Beh = TEST-Blocked, the physical outputs for closing/opening the breaker are blocked.

7.7.4 Blocking oscillating inputs

An operator, or an automatic function as well, may freeze the value of communication outputs related to oscillating inputs. The blocking of the update of the value shall be available at data level and thus is realized by the data attribute blkEna (blocking enabled). When the data attribute is set, as a result, the quality of the associated data becomes operatorBlocked and oldData (see IEC 61850-7-3).

7.8 Data used for testing

7.8.1 General

In order to carry out functional, commissioning or maintenance tests, a communications network-based SAS that supports testing functions should offer some of the following facilities:

– at the IED level, the option of receiving multicast simulation signals instead of actual signals (see 7.8.2);
– at the LN (function) level, the option of receiving test input signals instead of actual signals (see 7.8.3);
– at the LN (function) level, the option of setting a function or a group of functions of the system in test mode (see 7.8.4).

7.8.2 Multicast signals used for simulation

Figure 40 shows an IED (IED1) receiving simultaneously simulation and actual signals.

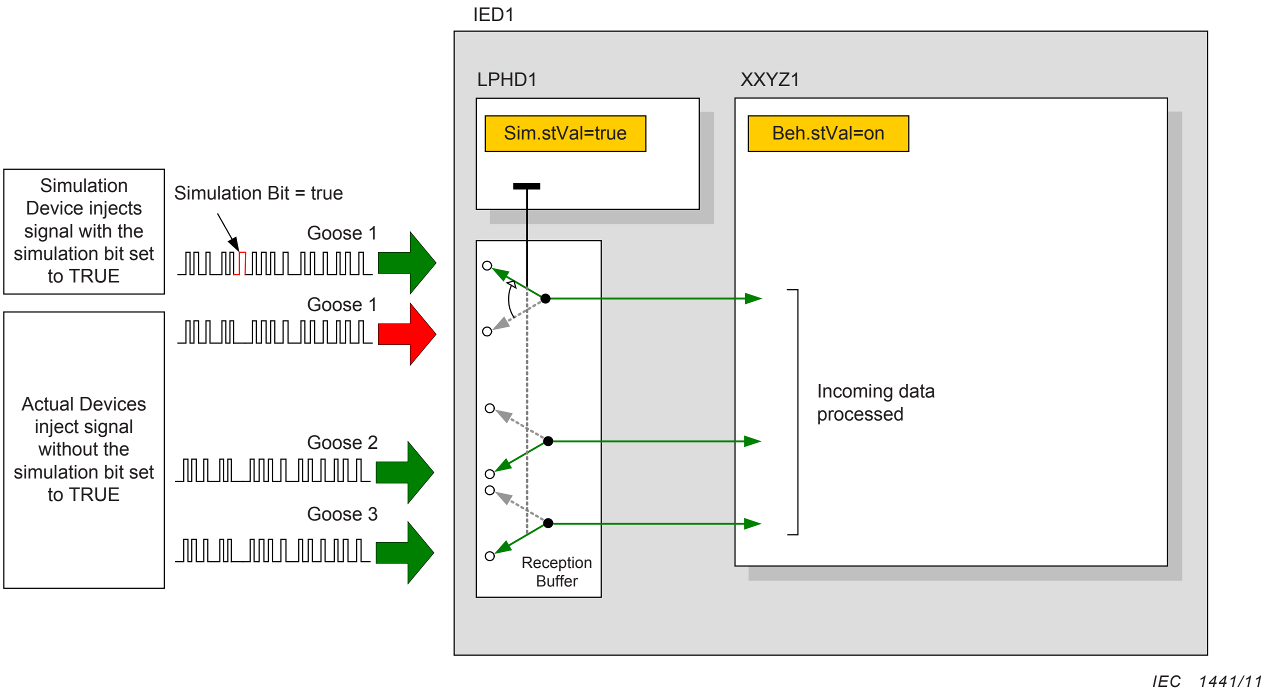


Figure 40 – Data used for receiving simulation signals

On the left side of the figure, actual devices are sending three Goose messages (Goose 1, Goose 2 and Goose 3) to the function represented by the LN XXYZ1 in the IED1. Additionally, a simulation device is sending the same Goose 1 message, but this time with the simulation bit being set to TRUE.

To allow the IED1 to process the simulated Goose1 message instead of the actual Goose 1 message, the data Sim.stVal in the LN LPHD1 shall be set to TRUE. As of this moment, the simulated signal shall replace the actual signal. This will remain as such until Sim.stVal is set back to 'false'. During this time, the other signals, Goose 2 and Goose 3 are normally processed by the IED1. Note that the switching between a normal signal and a simulated signal shall be done for the entire IED. In the example, this means that if a new simulated Goose 2 signal appears, it will replace the actual Goose 2 signal.

NOTE The preceding example doesn't imply any specific sequence with regards to connecting the simulation device and setting the data Sim.stVal to TRUE. Different sequences may be used.

The example in Figure 40 not only applies to Goose messages but to sampled values (SMV) as well.

7.8.3 Input signals used for testing

Each LN may have one or several data named InRef1 to InRefn and used as input signals to the function represented by the LN (see 9.6). The common data class of InRef1 is ORG, "object reference setting group". The data can be used to switch between actual and test incoming signals as shown in the example of Figure 41.

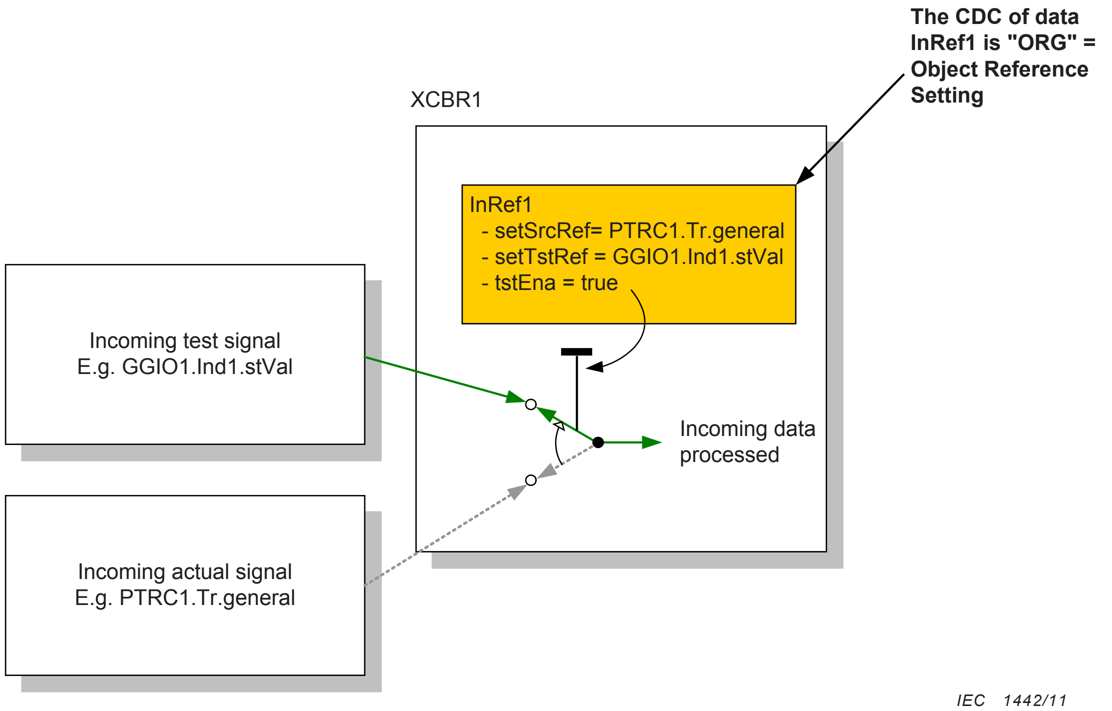


Figure 41 – Example of input signals used for testing

In normal operation, the breaker function (XCBR1) receives as an input the signal PTRC1.Tr.general intended to trip the breaker. For functional testing, this signal can be changed for another incoming signal, GGIO1.Ind1.stVal in the example. This change is made by setting the value of the attribute InRef1.tstEna to TRUE. The common data class ORG also has the attributes setSrcRef and setTstRef used to indicates the origin of the incoming actual and test signals. The values of the attributes setSrcRef and setTstRef shall also contain the logical device names of the incoming signals. In the example, they have intentionally not been specified as the concept of the logical device is introduced further in this document.

7.8.4 Test mode

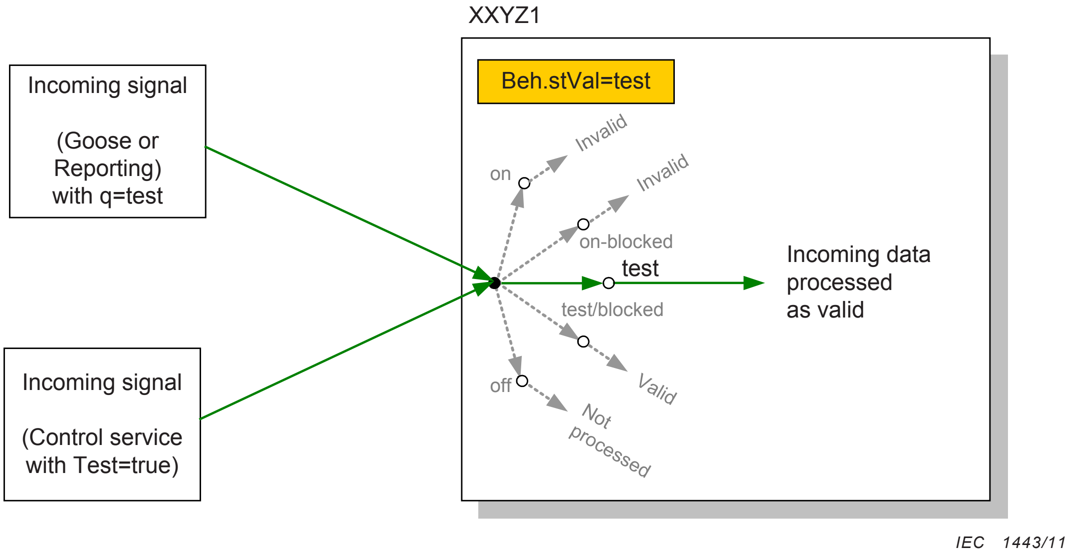


Figure 42 – Test mode example

The Figure 42 shows an example of a function (LN XXYZ1) in test mode. In order to make the function accept an incoming signal as valid for testing purposes, the following conditions shall be observed:

– the mode of the function (Beh.stval) being part of the test shall be set to 'test' or 'test/blocked',

– in case of Goose or reporting services, the quality value of the incoming data shall be 'test',

– in case of control services, the control service parameter Test value shall be TRUE.

Otherwise, the signal will be treated as invalid. The detailed requirements for the different states of Mod/Beh can be found in IEC 61850-7-4.

7.9 Logical node used for extended logging functions

Some applications require the recording of sequential events whose events are not only defined by the data that triggered the event but also by other useful information that needs to be captured simultaneously when the event occurred. The function used to set the additional data to be captured is represented by the LN GLOG: Generic Log, as shown in Figure 43.

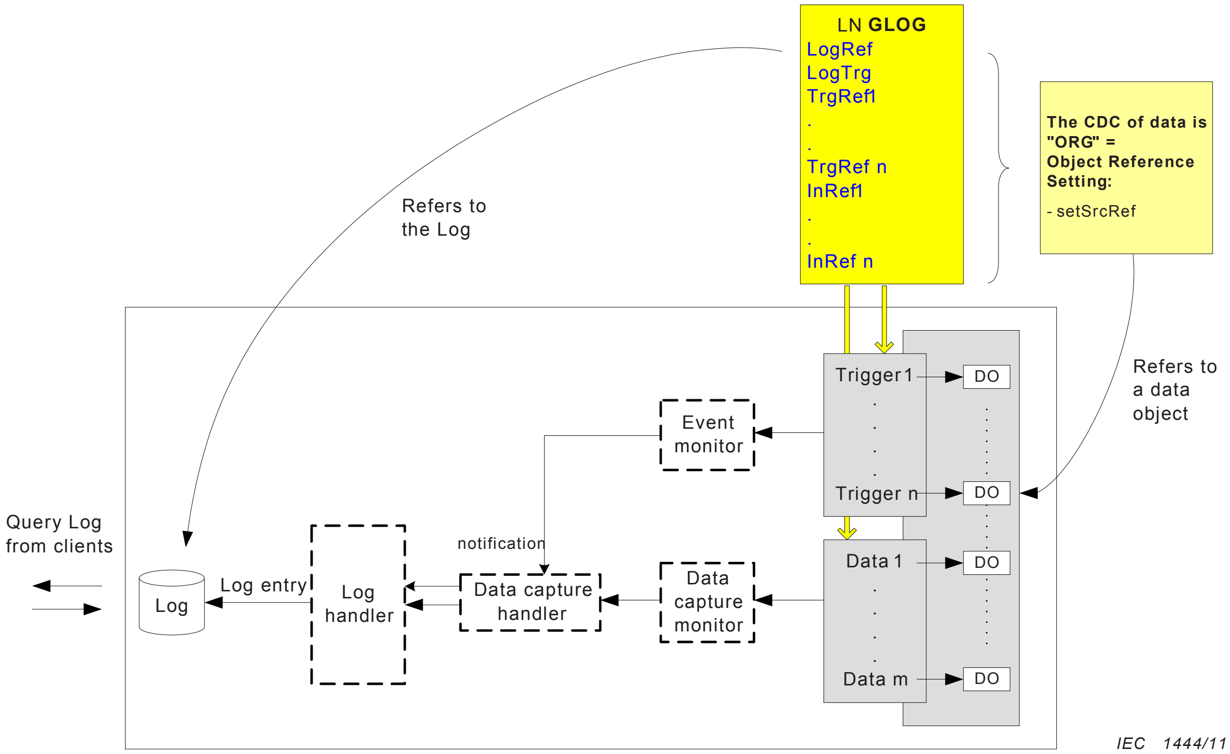


Figure 43 – Logical node used for extended logging functions (GLOG)

The LN GLOG is associated to a Log (see 6.4.3.3.3) and contains settings (CDC ORG) which are used to reference event triggers and the additional data to be captured. In Figure 43, it can be seen that the LN GLOG contains the information about the triggers (TrgRef1 to TrgRefn) and the additional data (InRef1 to InRefn).

Once an event is detected and notified, the captured data is added to the Log. The reason for inclusion, as defined in IEC 61850-7-2, shall be application-trigger.

GLOG only represents additional data to be logged when a trigger occurs. The usage of GLOG is application specific and therefore doesn't need to be controlled by a LCB (see 6.4.3.3.3).

If needed, the triggers themselves may be included into the log by two means:

– the use of a log control block and a dataset which includes the trigger references;
– by fixing the settings InRefx equal to the trigger references.

There shall be only one Log per LN GLOG.

8 Device view

8.1 General

Real devices host mainly:

– logical nodes and data – representing the real application functions and associated information visible from the communication network (data defined in IEC 61850-7-4),
– information about real devices – representing information about the resources of the host itself and (if applicable) about the real equipment connected to the host device (specific logical nodes and data defined in IEC 61850-7-4),

– communication services and mapping to specific communication systems – representing the supported information exchange services (defined in IEC 61850-7-2 and SCSMs).

The second and third items require further components in the model. To define information about the devices and to model those communication aspects that are applicable to more than one logical node requires a model that comprises the logical nodes and the further information and service models.

8.2 Second modelling step – logical device model

8.2.1 The logical device concept

For communication purposes (beyond a logical node), the concept of a logical device has been introduced. A logical device is mainly a composition of logical nodes and additional services (for example GOOSE, sampled value exchange, setting groups) as illustrated in Figure 44. The grouping of logical nodes in logical devices is based on common features of these logical nodes. For example, the modes of all these nodes are normally switched on and off together, or in the test mode.

NOTE GOOSE is used to very rapidly exchange input and output data mainly of relays.

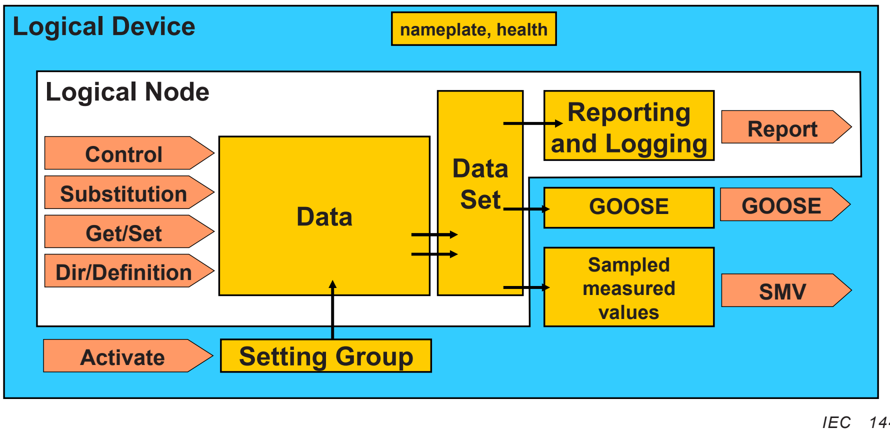


Figure 44 – Logical device building block

Logical devices provide information about the physical devices they use as host (nameplate and health) or about external devices that are controlled by the logical device (external equipment nameplate and health). Logical devices reside in physical devices as shown in the example in Figure 45. Only those aspects of physical devices that are defined as visible to the network are of interest in this standard.

In the example of Figure 45, logical device “LD1” contains three logical nodes: LLN0, LPHD and LN, the latter representing an application specific LN as defined in the IEC 61850-7-x series. The logical node zero (LLN0) represents common data of the logical device. For example, the mode of LLN0 is used to control the mode of the whole logical device and, as a result, the mode of every logical node being part of the logical device. The logical node physical device (LPHD) represents common data of the physical device hosting the logical device. LLN0 shall be defined in any logical device while LPHD shall be defined in at least one logical device. With the exception of LPHD, all system logical nodes (Group L) belonging to the same IED shall be defined in the same logical device (e.g. logical device “SYS” in Figure 45).

On the right-hand side, the representation of for example nameplate information about the primary equipment is defined as data of the logical node that represents the primary equipment.

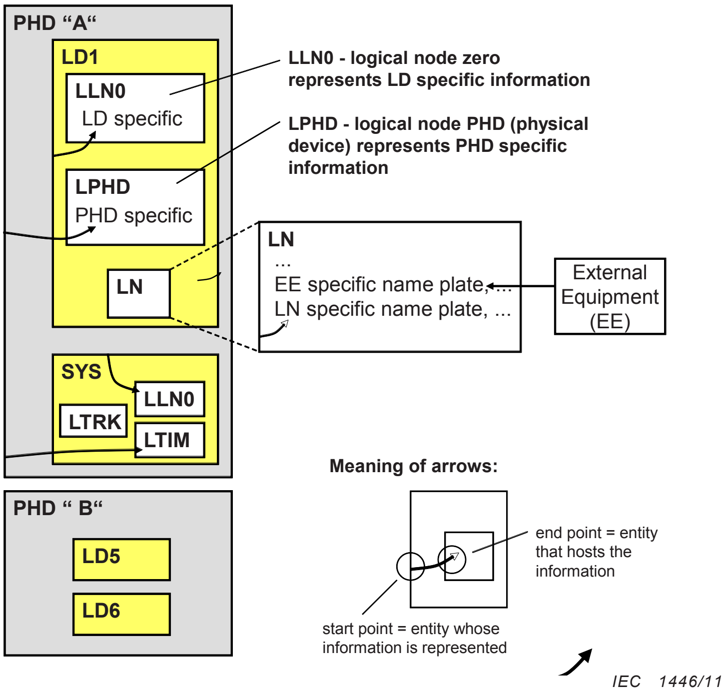


Figure 45 – Logical devices and LLN0/LPHD

The functional content of logical devices, e.g. logical nodes, is not standardized by the series of documents IEC 61850-7-x, neither the name of logical devices.

8.2.2 The device nameplate

Physical devices and affiliated primary equipment are identified by the device nameplate information. This information comprises asset management related data like the serial number, the location, the owner, the electric power system to which the primary equipment is connected, the operator of the device, etc. The device nameplate information is part of the common data class DPL in IEC 61850-7-3. The DPL attributes apply to physical devices and to primary equipment as well. Of course, some attributes only make sense when they are used for physical devices.

Figure 46 shows the DPL common data class used in two different logical nodes. The data PhyNam in logical LPHD represent the nameplate of a physical device while the data EEName is the nameplate of primary equipment represented by the logical node XCBR.

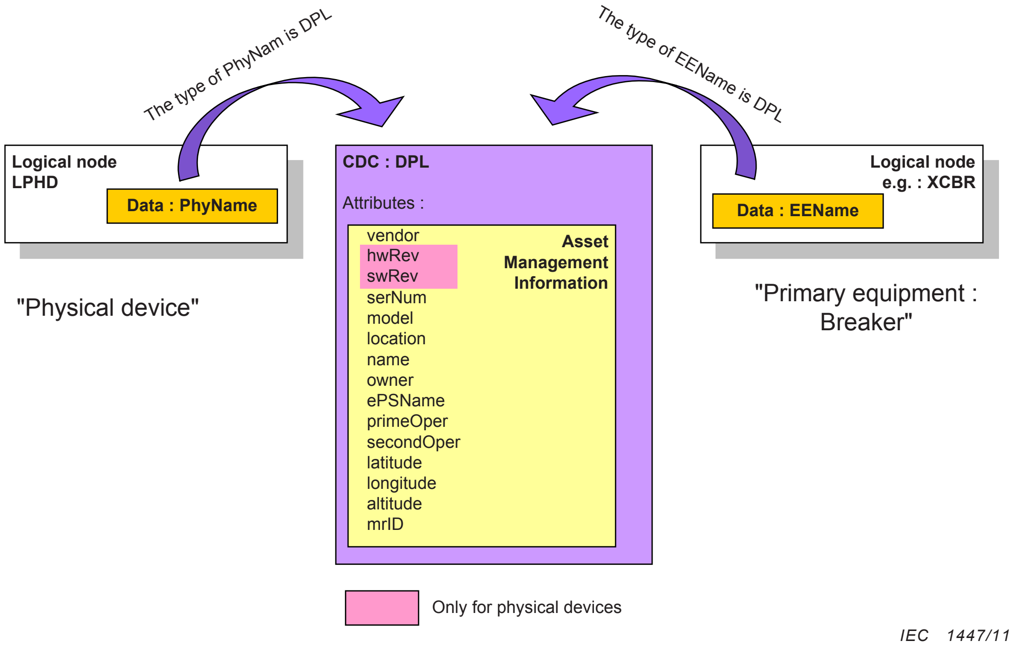


Figure 46 – The common data class DPL

8.2.3 Gateways and proxies

Gateways are network interconnection devices that translate protocols to other protocols. For example, gateways may convert non IEC 61850 data into IEC 61850 data.

Proxies are special devices that mirror logical devices located in other IEC 61850 physical devices. Hence, these logical devices are, from a functional point of view, transparent. They can be identified independently of their location (in a separate device connected to the network or in a proxy device).

Logical devices allows for the building of proxies and/or gateways.

Figure 47 shows how a physical device is mapped to a device acting as a proxy and/or a gateway. In the figure, LN refers to an application specific LN as defined in the IEC 61850-7-x series.

The logical devices A_LD1 and A_SYS are "copied" to the proxy/gateway. The LPHD of A_LD1 in the proxy/gateway represents the physical device PHD "A".

Logical devices that mirror logical devices of other physical devices shall provide a LPHD that represents the remote physical device on which the original LD resides (e.g., A_LD1). These logical devices shall have the data LPHD.Proxy.stVal of the LPHD set to TRUE.

Logical devices that do not mirror logical devices of other physical devices shall provide a LPHD that represents the physical device on which they reside. In Figure 47, this logical device is PXY_GTW_LD1, which shall be implemented to represent the information about the proxy/gateway itself. The logical nodes LLN0 and LPHD of this logical device shall represent information about the proxy/gateway device. The logical device may also contain domain specific logical nodes.

Figure 47 also shows how non IEC 61850 external physical devices are mapped to the proxy/gateway device. C_LD2 is used to represent the information related to the non IEC 61850 physical device PHD "C". The LPHD of C_LD2 represents the physical device PHD "C". The data supplied by PHD "C" is mapped to different logical nodes into C_LD2.

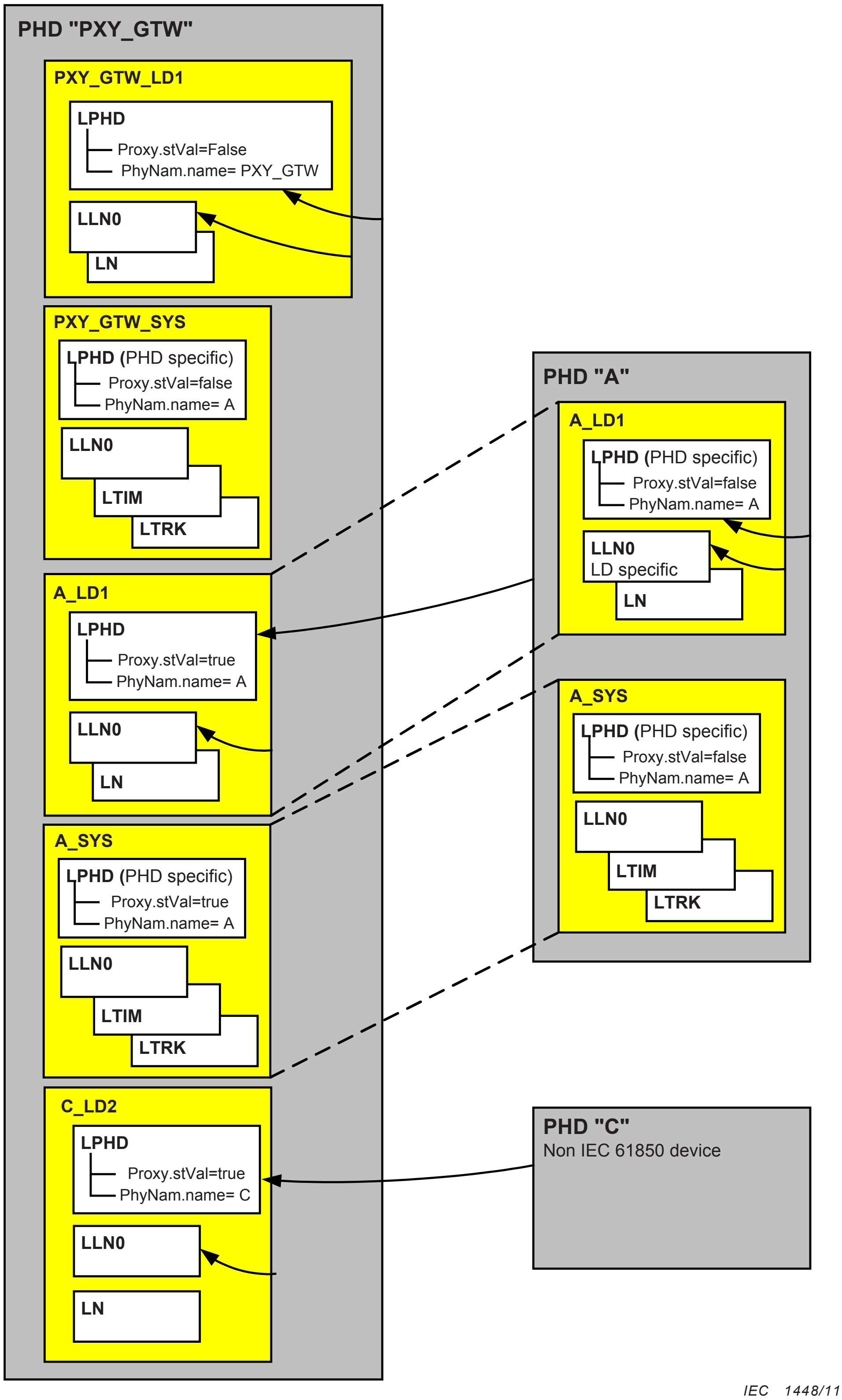


Figure 47 – Logical devices in proxies or gateways

8.2.4 Logical devices for monitoring external device health

The logical device concept may be used to supervise the health of an external device. Figure 48 shows an example where the gateway approach is reused. PHD “B” is a measuring device connected to a sensor and communicating with PHD “A” using a non-IEC 61850 means of communication. The T node is used to represent the sensor health while the LPHD node is used to represent the health of the external device.

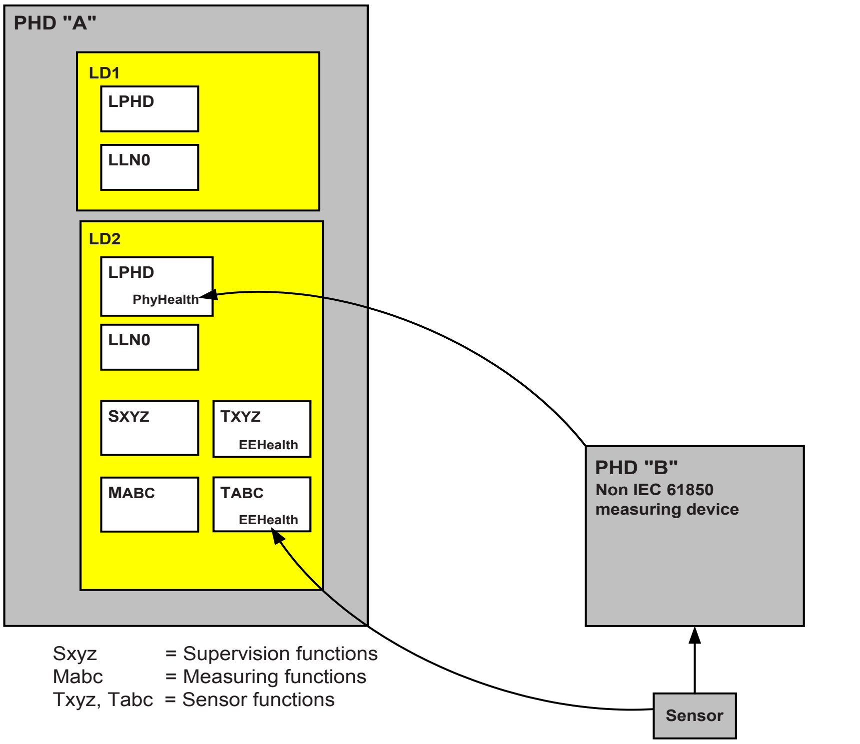


Figure 48 – Logical devices for monitoring external device health

8.2.5 Logical devices management hierarchy

Logical devices are used to represent a group of typical automation, protection or other functions. The functions are defined as logical nodes contained and managed in logical devices. This simple hierarchy may not be sufficient to model complex functions, e.g. distance protection. These functions are usually made of nested functions and sub-functions, represented by logical nodes, which shall be grouped together according to a management hierarchy. The hierarchy determines how the mode (e.g. On, Off, Test, …) or the health of these functions and sub-functions is managed. The concept is shown in Figure 49 where LD1 is called the “root” logical device.

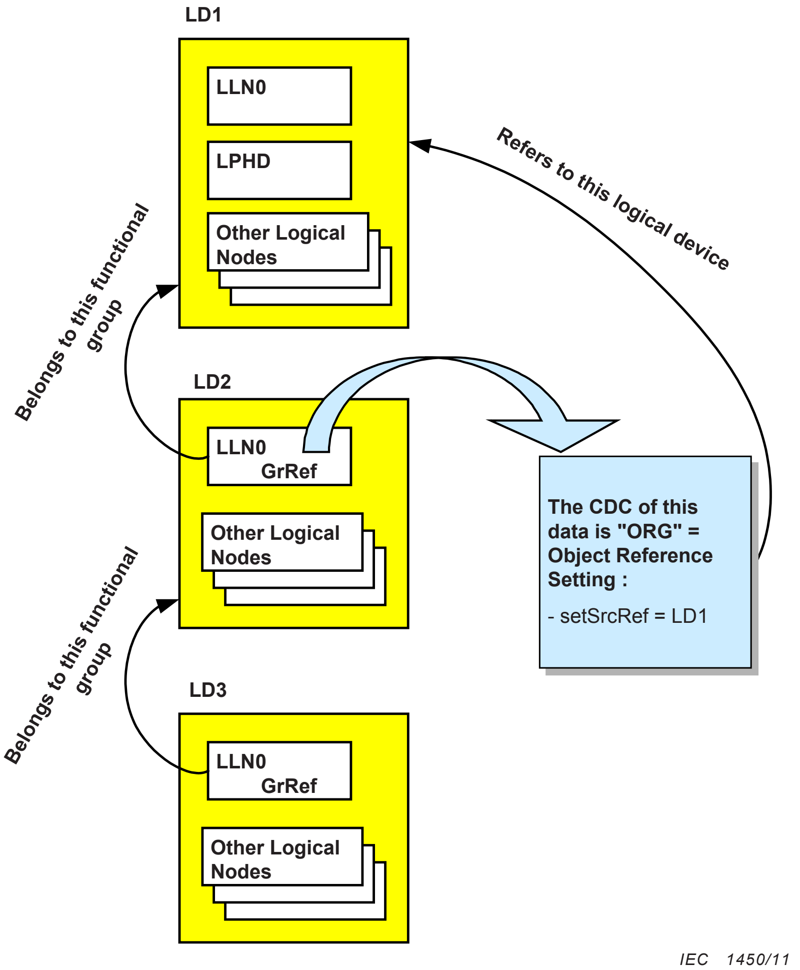


Figure 49 – Logical devices management hierarchy

LLN0 of the logical device LD2 contains a setting data named GrRef whose common data class is ORG, "object reference setting group". The referenced value of GrRef is LD1, meaning that the logical device function refers to the functional group represented by the logical device LD1. Likewise, LLN0 of LD3 refers to the functional group represented by the logical device LD2. In other words, LNs in LD3 are sub-functions of LD2.

In Figure 49, the mode of the LNs in LD3 may be changed individually or globally by means of LLN0 of LD3. Their mode may also be changed either by means of LLN0 of LD2 or by means of LLN0 of LD1. For example, if the mode of the functional group LD2 is set to "Off", it not only set the behaviour of all logical nodes in LD2 to "Off" but also the behaviour of all logical nodes in LD3. Switching the mode of LD1 will affect the behaviour of all logical devices and logical nodes belonging to the functional group LD1, i.e. all logical nodes in LD1, LD2 and LD3.

The health (see Figure 63) of the various LN and LD is also managed within the hierarchy. In Figure 49, the health of LLN0 in LD1 reflects the worst value of "Health" of the logical nodes that are part of the logical devices LD2 and LD3.

The rules with regard to the concept of the logical device management hierarchy are the followings:

– a logical device management hierarchy shall only be defined in a single physical device,
– within a logical device management hierarchy, only the logical node class LLN0 can refer to another logical device,
– the logical node class LPHD shall be present in the root logical device, i.e. the LD where the LLN0 does not contain a data object GrRef. It may also be present at other levels of the hierarchy,
– if a proxy LD exists within the hierarchy, it shall only be present at the root logical device,
– the behavior of any logical node or logical device in a branch of the logical device management hierarchy is a function of the modes along the branch from the leaf to the root logical device (see example above),
– if setting group control blocks are present, they shall appear only once in any branch of the logical device management hierarchy. The settings in a branch shall belong to the setting group control block of that branch.

9 Communication view

9.1 General

Communication systems abstract mainly from the application and device view. On the other side, the communication system, the devices, and the application are closely related. Clause 9 introduces the communication models. After that, the relations between these views are discussed.

9.2 The service models of the IEC 61850 series

The services are defined using an object-modelling technique. The service interface uses an abstract modelling method. Abstract means that the definition is focused on the description of what the services provide. The concrete messages (and their encoding) to be exchanged between devices (how the services are built) are not defined in this part of the standard. These concrete messages are specified in the specific communication service mappings (SCSMs in IEC 61850-8-x and IEC 61850-9-x series).

NOTE This abstraction allows various mappings appropriate for different requirements and following state-of-the-art in communication technology without in such a case changing the model and, therefore, databases, etc.

The ACSI (Abstract communication service interface) defines common utility services for substation devices. The two groups of communication services are depicted in Figure 50. One group uses a client-server model with services like control or get data values. A second group comprises a peer-to-peer model with GSE services (used for time-critical purposes, for example, fast and reliable transmission of data between protection IEDs, from one IED to many remote IEDs) and with the sampled value services for transmissions based on a periodic basis.

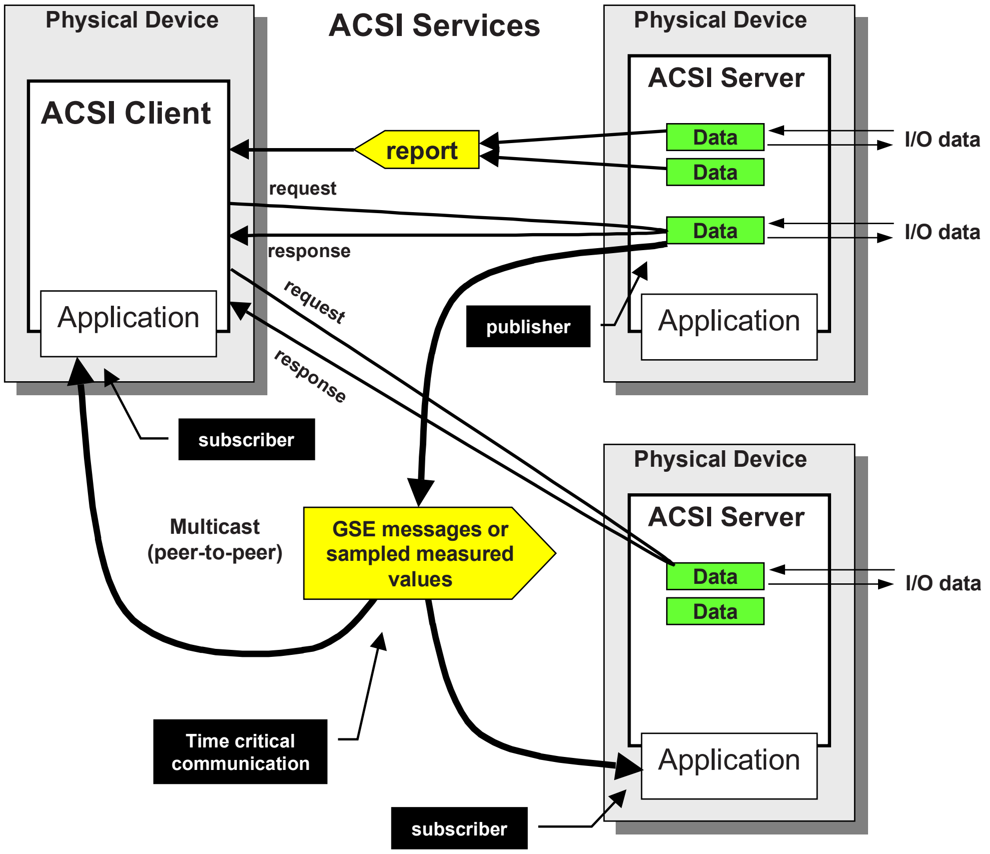


Figure 50 – ACSI communication methods

Real clients and servers can be connected by a variety of communication systems. Communication media may have geographic and utilisation constraints, such as limited bit rates, proprietary data link layers, restricted times for use, and satellite hop delays. Systems may be hierarchical, with a few central sites authorising and managing the interactions with a large number of "field" sites, or it may be networked with peer-to-peer interactions. Communication media may have varying configurations, such as point to multi-point, multi-drop, meshed, hierarchical, WAN-to-LAN, intermediate nodes acting as routers, as gateways, or as data concentrator databases, etc.

Table 5 lists the ACSI service models and services.

Table 5 – ACSI models and services

Service model	Description	Services
Server	Represents the external visible behaviour of a device. All other ACSI models are part of the server.	GetServerDirectory
Application association	Provision of how two devices can be connected. Provides different views to a device: restricted access to the server's information and functions.	Associate Abort Release
Logical device	Represents a group of functions; each function is defined as a logical node.	GetLogicalDeviceDirectory
Logical node	Represents a specific function of the substation system, for example, overvoltage protection.	GetLogicalNodeDirectory GetAllDataValues
Data	Provides a means to specify typed information, for example, position of a switch with quality information, and timestamp.	GetDataValues SetDataValues GetDataDefinition GetDataDirectory
Data set	Allow to group various data together.	GetDataSetValues SetDataSetValues

Service model	Description	Services
		CreateDataSet DeleteDataSet GetDataSetDirectory
Setting group control	Defines how to switch from one set of setting values to another one and how to edit setting groups.	SelectActiveSG SelectEditSG SetEditSGValues ConfirmEditSGValues GetEditSGValues GetSGCBValues
Reporting and logging	Describes the conditions for generating reports and logs based on parameters set by the client. Reports may be triggered by changes of process data values (for example, state change or deadband) or by quality changes. Logs can be queried for later retrieval. Reports may be sent immediately or deferred (buffered). Reports provide change-of-state and sequence-of-events information exchange.	Buffered RCB: Report GetBRCBValues SetBRCBValues Unbuffered RCB: Report GetURCBValues SetURCBValues Log CB: GetLCBValues SetLCBValues QueryLogByTime QueryLogAfter GetLogStatusValues
Generic substation events (GSE)	Provides fast and reliable system-wide distribution of data; peer-to-peer exchange of IED binary status information. GOOSE means Generic object oriented substation event and supports the exchange of a wide range of possible common data organised by a DATA-SET	GOOSE CB: SendGOOSEMessage GetGoReference GetGOOSEElementNumber GetGoCBValues SetGoCBValues
Transmission of sampled values	Fast and cyclic transfer of samples, for example, of instrument transformers.	Multicast SVC: SendMSVMessage GetMSVCBValues SetMSVCBValues Unicast SVC: SendUSVMessage GetUSVCBValues SetUSVCBValues
Control	Describes the services to control, for example, devices or parameter setting groups.	Select SelectWithValue Cancel Operate CommandTermination TimeActivatedOperate
Time and time synchronisation	Provides the time base for the device and system.	TimeSynchronization
File transfer	Defines the exchange of huge data blocks such as programs.	GetFile SetFile DeleteFile GetFileAttributeValues

9.3 The virtualisation

The ACSI provides access to the real data and real devices through a virtual image as depicted in Figure 51. A virtual image that represents the real data of devices is made visible and accessible through ACSI services. A computer may request services, for example, get data values, or may receive spontaneously reported values from the controller.

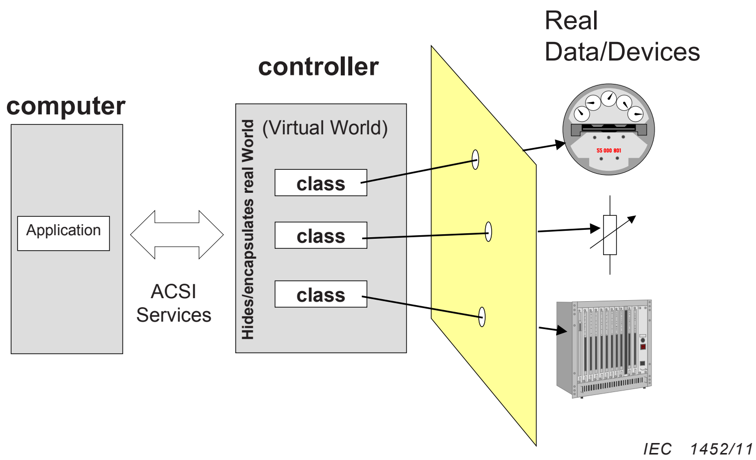


Figure 51 – Virtualisation

The virtual view can be used (as shown in Figure 52) to describe and represent the complete behaviour of a device. Any other device, another controller or even a SCADA system, a maintenance system, or an engineering system may use the ACSI services to inter-operate with that device. A service request received is independent of the device that has requested the service.

The communication system provides means to prevent that every single computer in the whole network is able to connect to any device and see and modify all information of any device. There are diverse access schemes defined that restrict the “visibility” of a device or particular data of a device. An operator may for example not be allowed to change protection settings.

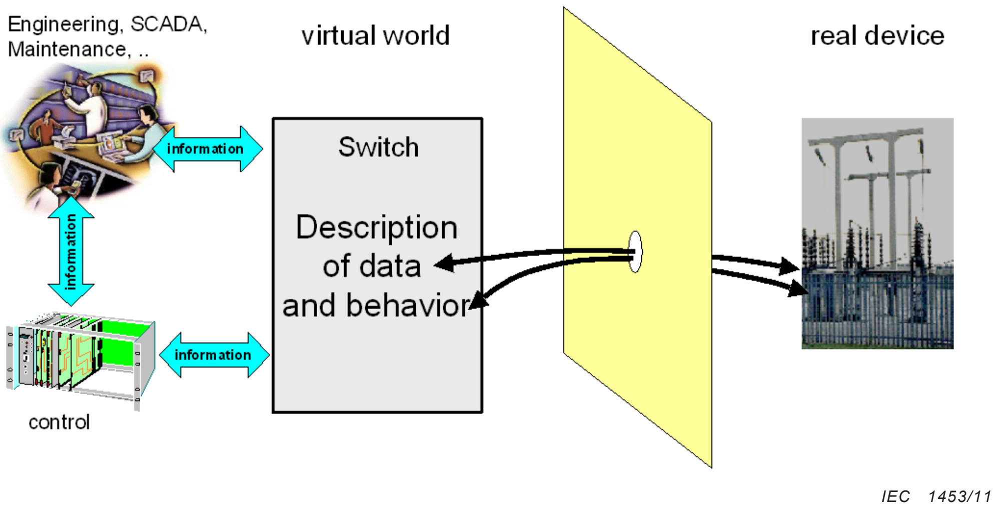


Figure 52 – Virtualisation and usage

9.4 Basic information exchange mechanisms

The ACSI model basically provides the methods of exchanging information between devices as shown in Figure 53.

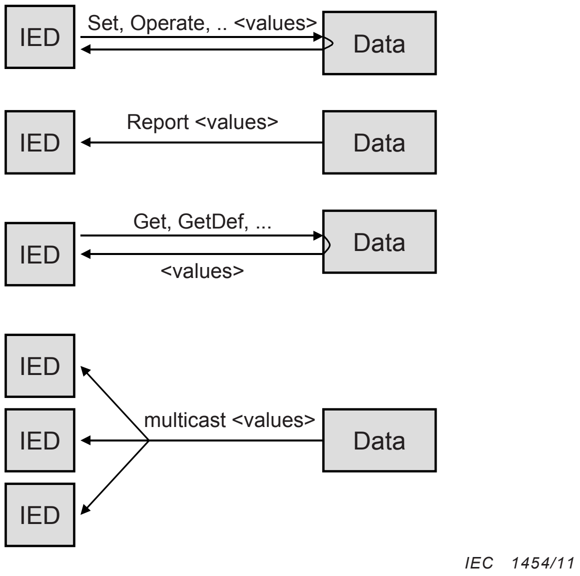


Figure 53 – Information flow and modelling

The use of the generic substation event model (GSE) is quite important because this model supports the implementation of real-time applications. Figure 54 shows an example of an application of the GSE model.

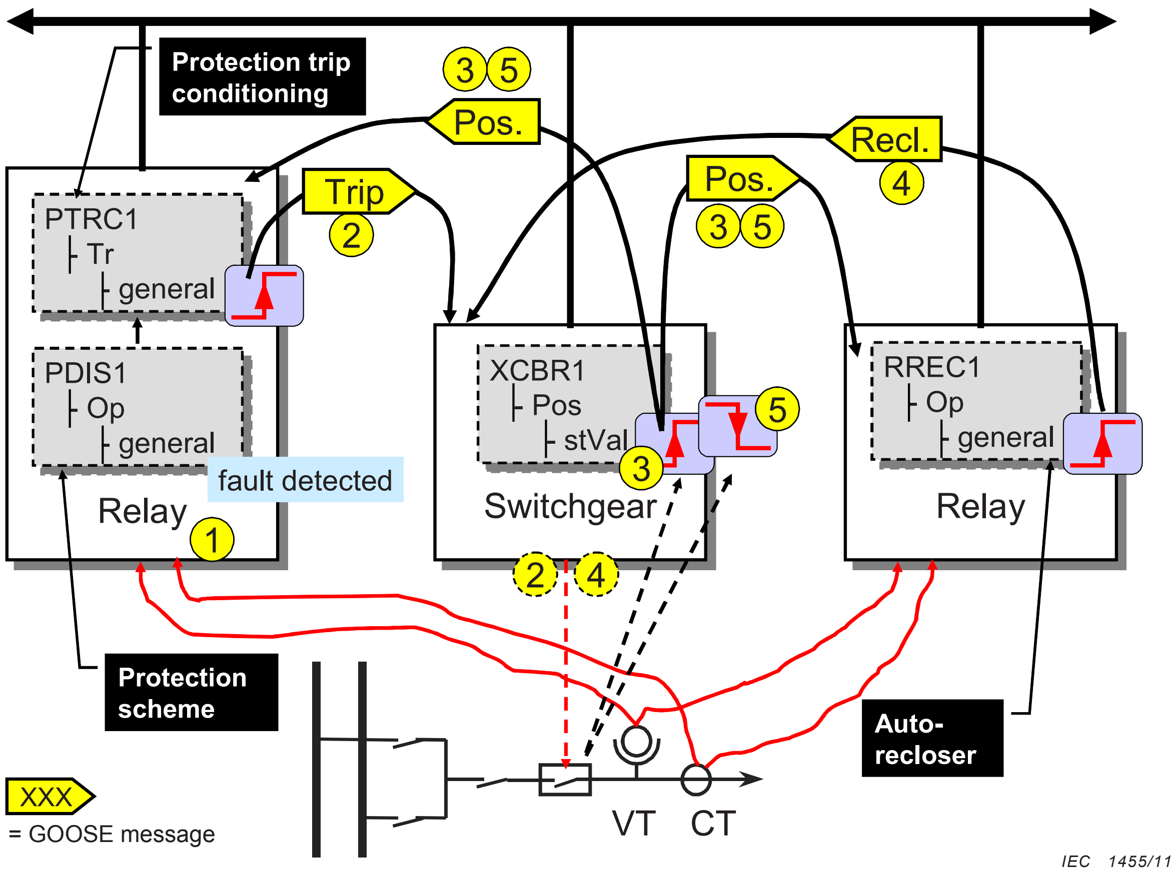


Figure 54 – Application of the GSE model

Five logical nodes are involved in the example. The sequence of actions and GOOSE messages is as follows.

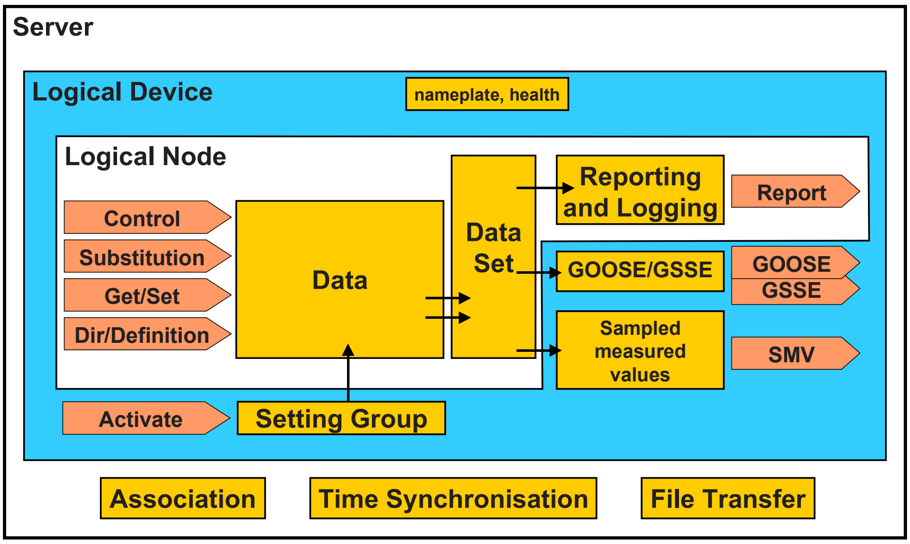
a) The logical node “protection scheme” (PDIS1) detects a fault, this results in a decision to issue a trip.
b) The logical node “protection trip conditioning” (PTRC1) issues a trip message (applying a GOOSE message), the “circuit breaker” (XCBR1) has been configured to receive the trip message. After additional processing, the switchgear opens the circuit breaker.
c) The status information of the “circuit breaker” (XCBR1.Pos.stVal) changes from ON to OFF. This new state is immediately sent by a GOOSE message with the indication: <new position of switch = open>. In addition, the reporting model may report the change.
d) The logical node “autoreclosing” (RREC1) receives a GOOSE message from the XCBR1 with the value <open>. According to the configured behaviour, the RREC decides to re-close the circuit breaker and sends a GOOSE message with the value <reclose>.
e) The “circuit breaker” (XCBR1) receives the GOOSE message with the value <reclose>. After additional processing, the switchgear closes the circuit breaker. The XCBR1 issues sending another GOOSE message < new position of switch = close>.

The sequence is an example only. The IEC 61850 series provides the basic mechanisms for exchanging GOOSE messages under real-time conditions. Applications of GOOSE messaging may be as simple as described in the example. But it may be used in more sophisticated schemes. All these schemes are outside the scope of the IEC 61850 series.

9.5 The client-server building blocks

9.5.1 Server

Additional common building blocks provided by the communication system are depicted in Figure 55. The association model provides mechanisms for establishing and maintaining connections between devices and to implement access control mechanisms. Time synchronisation provides the accurate time for time tagging (ms range) in applications such as reporting and logging or for applications such as synchronised sampling (µs range).



IEC 1456/11

Figure 55 – Server building blocks

The server contains everything that is defined to be visible and accessible from the communication network. A physical device may host one or more server.

Client-server Figure 56 illustrates the client/server role. Clients issue service requests and receive confirmations of the service that has been processed in the server. A client may also receive report indications from a server. All service requests and responses are communicated by the protocol stack that is being used by a specific communication service mapping.

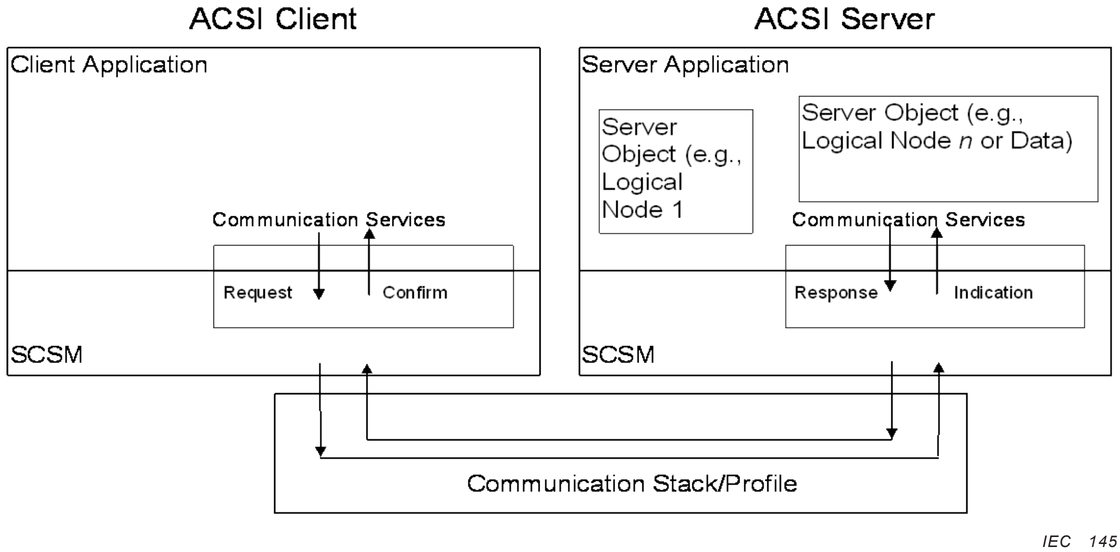


Figure 56 – Interaction between application process and application layer (client/server)

Figure 57 shows an example of a get service that enables a client to retrieve the values of the data inside the server.

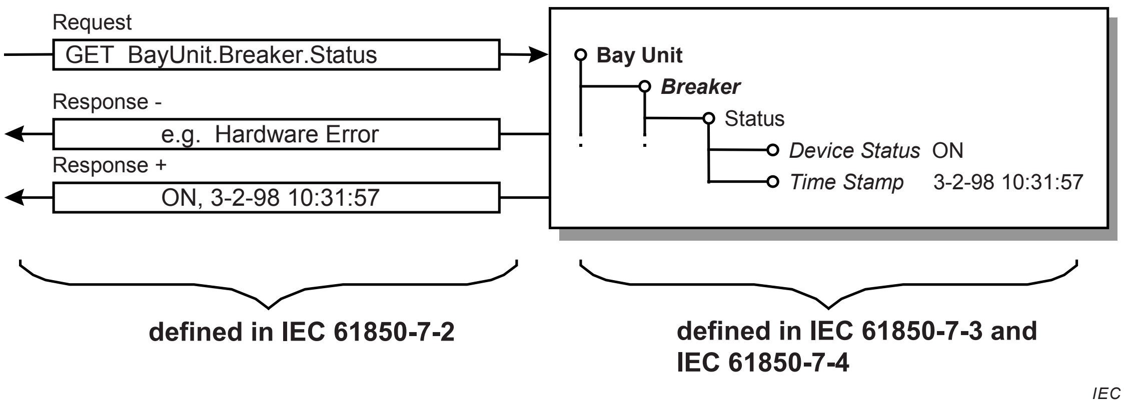


Figure 57 – Example for a service

9.5.2 Client-server roles

According to Figure 58 one server “serves” various logical nodes and clients.

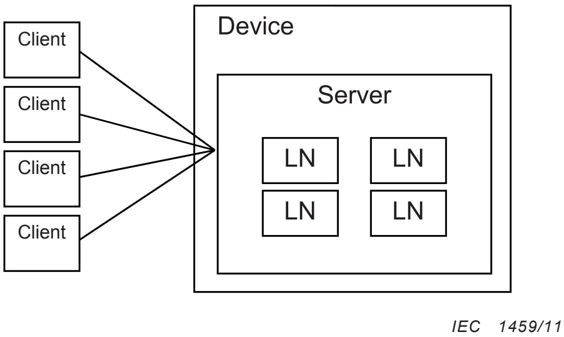


Figure 58 – Client/server and logical nodes

The standard defines just the server role: the logical nodes, data, control, etc. located in the server, and the service request exchanged. The client role is complementary.

NOTE Clients and their internal structure and functions are not defined in this standard.

As shown in Figure 59, the devices may implement the client and the server role.

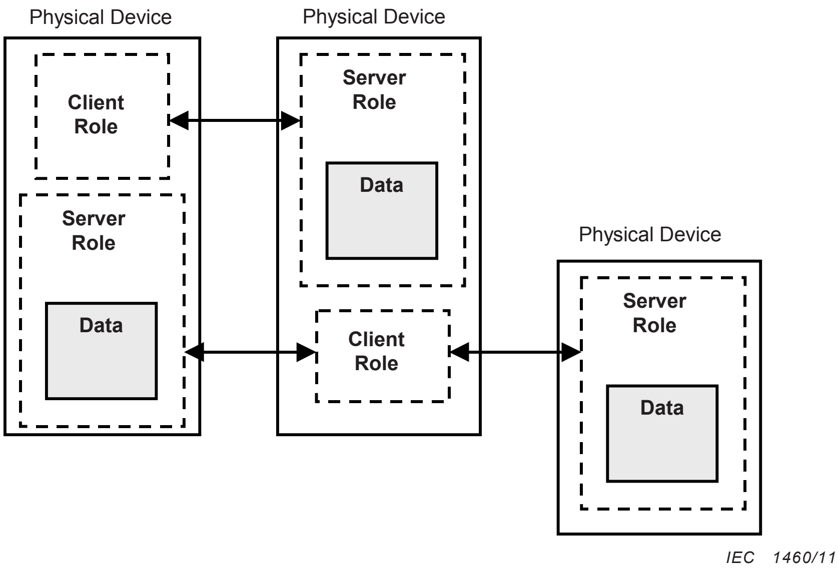


Figure 59 – Client and server roles

9.6 Logical nodes communicate with logical nodes

Logical nodes communicate with other logical nodes by means of PICOMs as described in IEC 61850-5. The logical nodes in that sense comprise the data and control as well as the client/server and publisher/subscriber roles (see Figure 59). The client and server are communication-specific entities. From an application viewpoint, they are not required. Therefore, the logical nodes (and only the logical nodes) can be understood as being in communication with one another. The logical node view and the communication view are two different views of the very same real subject.

As previously mentioned, the standard defines the server roles, i.e. the data model and the services used to exchange information. It means that only the server side of the communication view is standardized. Nevertheless, in order to describe the dataflow between logical nodes, the logical node class definition allows for the description of LN input references on the client/subscriber side as shown in Figure 60.

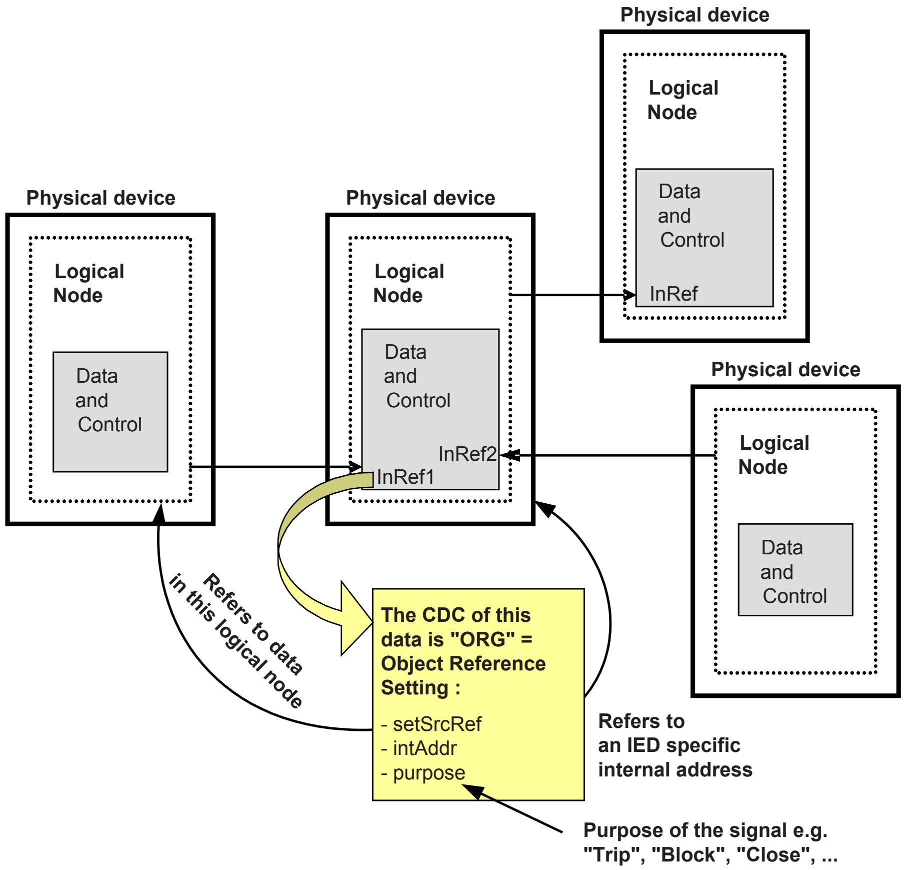


Figure 60 – Logical nodes communicate with logical nodes

Generic LN input references are used to describe incoming signals from an external source, i.e. reference to data coming from an LN hosted by the same IED or by another IED and its binding to an IED specific internal address. It is also possible to describe the intended purpose of the incoming signal. LN input references are defined by the instantiation of data named InRef and the use of the common data class ORG, “object reference setting group”.

NOTE Being a setting, the value of the CDC ORG attributes may be changed online.

Additionally, the SCL configuration language (see IEC 61850-6), allows to statically describe the data flow between LNs using special elements called LN inputs.

9.7 Interfaces inside and between devices

Real substation systems have many interfaces – interfaces for different purposes (see Figure 61). IEC 61850-7-x and IEC 61850-8-x series as well as IEC 61850-9-x series define interfaces between devices (between two devices in a client/server relationship and between many devices in a peer-to-peer relationship). IEC 61850-7-x series define abstract interfaces, while IEC 61850-8-x and IEC 61850-9-x series define concrete interfaces.

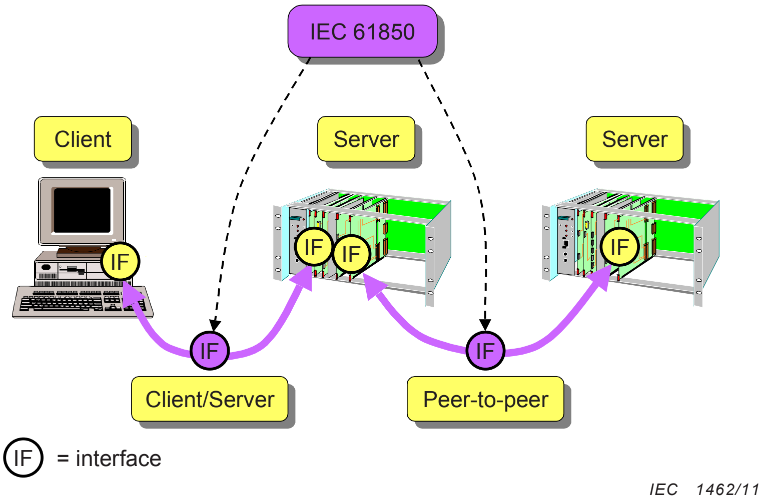


Figure 61 – Interfaces inside and between devices

Any other interfaces (especially APIs within client or server devices) are outside the scope of this standard. On the other hand, the information model and services defined have an impact on the software and the concrete interfaces in real devices.

10 Where physical devices, application models and communication meet

Physical devices are placed in the centre of a hierarchy of components as shown in Figure 62. All views "meet" in the server. Each view has a relation to the other views inside the physical device. The various views are shown here to demonstrate that in addition to the IEC 61850 series (which describes just one view of a real automation system), many other aspects have to be taken into account when real devices are implemented.

The server is the key component. It is important to differentiate the following aspects:

a) the server represents the application data modelling view (the IEC 61850 series) to the outside network,

b) the server represents all aspects of the communication network and the process I/Os to the application of the physical device,

c) a SCSM maps the IEC 61850 series view to the communication network visible objects,

d) the server, the SCSM and the application functional view are mapped to the resource of a physical device.

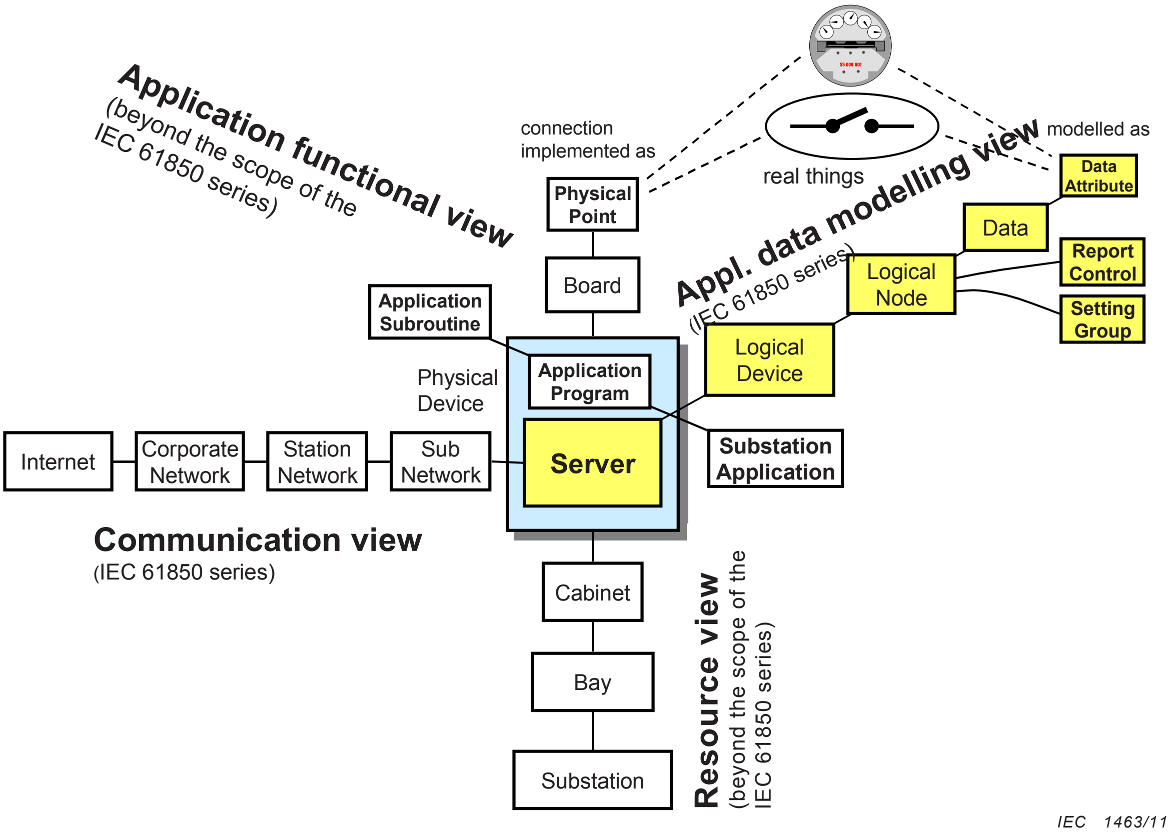


Figure 62 – Component hierarchy of different views (excerpt)

For real devices, all aspects (applications, APIs, views, mappings, relations) have to be implemented. Devices conforming to the IEC 61850 series make the IEC 61850 series view visible to any other device connected to the network for interoperability with applications running in these devices. Anything that is not modelled as a service, logical device, logical node, data, data attribute, setting group, report control, etc. is not visible to the network.

NOTE 1 The standard covers those definitions (information models and service models) that are defined to be compatible. Real devices usually require also vendor and user specific definitions that go beyond the standard. These specific definitions (outside of the scope of this standard) may be implemented as well.

NOTE 2 The engineering and configuration of real devices and systems deal with (1) the compatible definitions (mainly information models) of this standard which are covered by the SCL and (2) the application, vendor, and user specific definitions which require special attention (the extensions of the information model may be partly specified in an SCL extension).

Additional views such as the configuration view are outside the scope of this part of IEC 61850. The network management view and the system management view are not covered by this standard. A lot of information required for device management is modelled in IEC 61850-7-4 as data classes in logical node zero (LLN0). For details on the configuration view, refer to IEC 61850-6.

11 Relationships between IEC 61850-7-2, IEC 61850-7-3 and IEC 61850-7-4

11.1 Refinements of class definitions

One major building block is the DATA class defined in IEC 61850-7-2. The DATA class is used in the definition of almost all information that is defined in logical nodes. The DATA class as defined in IEC 61850-7-2 is on the left hand side of Figure 63. The DATA class defines three data attributes and four services. The services are defined in IEC 61850-7-2. The content of the data attributes are not specified in IEC 61850-7-2. The DATA class therefore is very generic. It must become more specific if it is to be used in an application domain. This could require definition of all DATA needed to model substation specific functions inside the logical nodes. It is common practice to analyse an application domain to find common properties and terms applicable to many data classes. These common definitions are provided by the common data classes (CDC) specified in IEC 61850-7-3.

Common data classes are built on DATA classes. In the middle of the figure, the example common data class "ENS" (enumerated status) is shown as a refinement of the DATA class. The "ENS" refines the DataAttributes that are left empty in IEC 61850-7-2. Four attributes are defined: "stVal" (Status value), "q" (Quality), "t" (Timestamp), and "d" (Description). This common definition is used in many data definitions throughout IEC 61850-7-4.

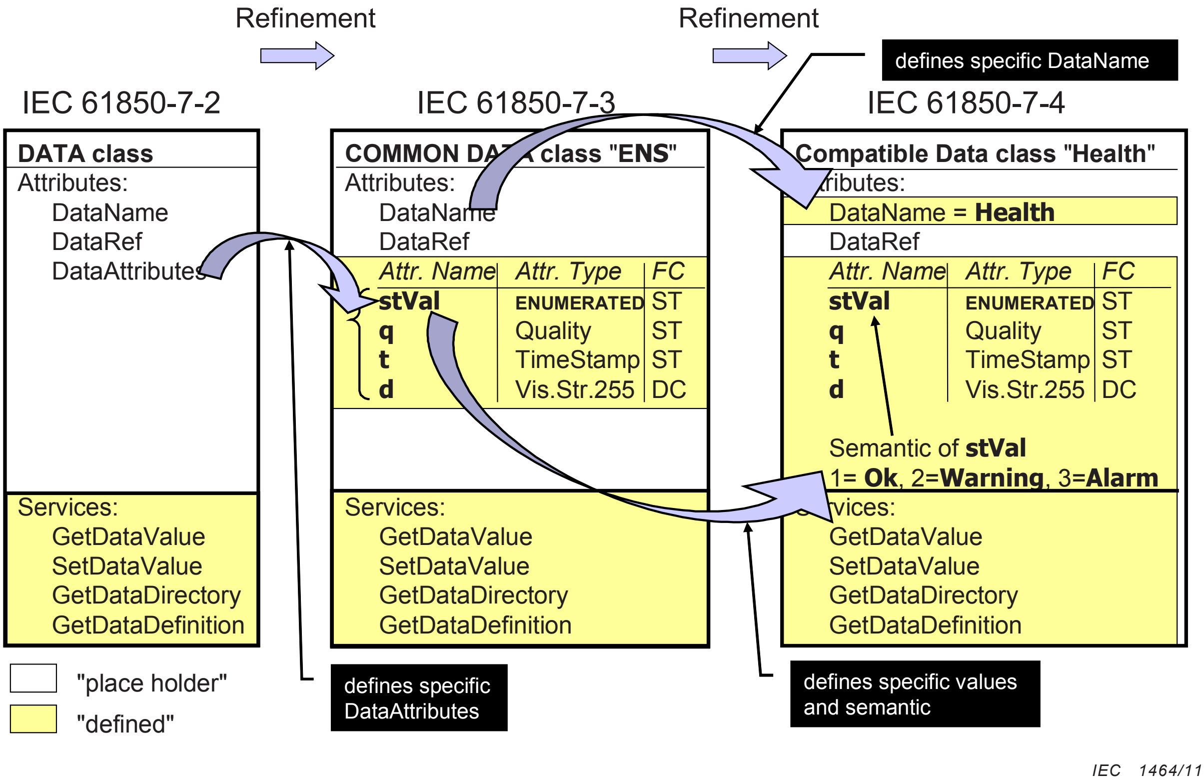


Figure 63 – Refinement of the DATA class

The DATA so far does not tell much about its use or the semantics of the data attributes derived from "ENS". The class on the right-hand side of Figure 63 defines exactly this “use”. The "Health" class defines the name "Health". That name will be used in all instances derived from this class. In addition, the status value "stVal" is an enumerated value defined as having three values: “Ok” (=1), “Warning” (=2), and “Alarm” (=3).

The standardised names and the semantic definitions associated with the names contribute essentially to the requested interoperability.

The final definition as to what the names "OK", "Warning", and "Alarm" really mean, depends on the context in which this class is used. In a circuit breaker, it may have a slightly different meaning than in a measurement unit.

11.2 Example 1 – Logical node and data class

Table 6 shows an example of a list of DATA classes for a circuit breaker. The name of the circuit breaker class is "XCBR". The DATA classes that make up the circuit breaker are grouped into three categories (basic LN information, controllable data, and status information). Each category comprises some DATA classes, for example, "Mode" and “Switch position”. These DATA classes are referenced by their DataName: "Mode" and "Pos". To be more precise, each DATA class also has a common data class, defining the details, i.e. the attributes of the DATA class. The last column specifies whether this data class is mandatory (M) or optional (O).

Table 6 – Logical node circuit breaker

Logical Node: Circuit breaker		**Name: XCBR**	
Data-Class	**DataName**	**Common Data Class (CDC)**	**M/O**
Basic Logical Node information			
Mode	Mod	ENC - Controllable integer status	O
Behaviour	Beh	ENS - Integer status	M
Health	Health	ENS - Integer status	O
Name plate	NamPlt	LPL - Logical node name plate	O
Local control behaviour	Loc	SPS - Single point status	M
External equipment health	EEHealth	ENS - Integer status	O
External equipment name plate	EEName	DPL - Device name plate	O
Operation counter	OpCnt	INS - Integer status	M
Controllable Data			
Switch position	Pos	**DPC - Controllable Double Point**	M
Block opening	BlkOpn	SPC - Controllable single point	M
Block closing	BlkCls	SPC - Controllable single point	M
Charger motor enabled	ChMotEna	SPC - Controllable single point	O
Control authority at station level. Switches between station and higher level.	LocSta	SPC - Controllable single point	O
Metered Values			
Sum of switched amperes, resettable	SumSwARs	BCR - Binary counter reading	O
Status information			
Local operation. Indicates the switchover between local and remote operation; local = TRUE, remote = FALSE.	LocKey	SPS - Single point status	M
Circuit breaker operating capability	CBOpCap	ENS - Integer status	O
Point on wave switching capability	POWCap	ENS - Integer status	O
Circuit breaker operating capability when fully charged	MaxOpCap	ENS - Integer status	O
Setting information			
Select mode of authority for local control	MltLev	SPG - Single point setting	O

Since many DATA classes use the same details (ATTRIBUTES), these details are therefore collected for re-use in common data classes (common to many DATA classes). The common data classes are defined in IEC 61850-7-3. As an example, the “controllable double point” (DPC) common data class for "Pos" is shown in Table 7.

Table 7 – Controllable double point (DPC)

DPC class					
Data attribute name	**Type**	**FC**	**TrgOp**	**Value/Value range**	**M/O/C**
DataName	Inherited from GenDataObject Class or from GenSubDataObject Class (see IEC 61850-7-2)				
DataAttribute					
		status and control mirror			
origin	Originator	ST			AC_CO_O
ctlNum	INT8U	ST		0..255	AC_CO_O
stVal	CODED ENUM	ST	dchg	intermediate-state | off | on | bad-state	M
q	Quality	ST	qchg		M
t	TimeStamp	ST			M
stSeld	BOOLEAN	ST	dchg		O
opRcvd	BOOLEAN	OR	dchg		O
opOk	BOOLEAN	OR	dchg		O
tOpOk	TimeStamp	OR			O
		substitution and blocked			
subEna	BOOLEAN	SV			PICS_SUBST
subVal	CODED ENUM	SV		intermediate-state | off | on | bad-state	PICS_SUBST
subQ	Quality	SV			PICS_SUBST
subID	VISIBLE STRING64	SV			PICS_SUBST
blkEna	BOOLEAN	BL			O
		configuration, description and extension			
pulseConfig	PulseConfig	CF	dchg		AC_CO_O
ctlModel	CtlModels	CF	dchg		M
sboTimeout	INT32U	CF	dchg		AC_CO_O
sboClass	SboClasses	CF	dchg		AC_CO_O
operTimeout	INT32U	CF	dchg		AC_CO_O
d	VISIBLE STRING255	DC		Text	O
dU	UNICODE STRING255	DC			O
cdcNs	VISIBLE STRING255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING255	EX			AC_DLN_M

Services (see IEC 61850-7-2)			
The following services are inherited from IEC 61850-7-2. They are specialised by restricting the service to attributes with a functional constraint as specified below.			
Service model of IEC 61850-7-2	**Service**	**Service applies to Attr with FC**	**Remark**
GenCommonDataClass model	SetDataValues GetDataValues GetDataDefinition GetDataDirectory	DC, CF, SV, BL ALL ALL ALL	
Data set model	GetDataSetValues SetDataSetValues	ALL DC, CF, SV, BL	
Reporting model GSE model Sampled values model	Report SendGOOSEMessage SendGSSEMessage SendMSVMessage SendUSVMessage	ALL ST, MX ST ST, MX ST, MX	As specified within the data set that is used to define the content of the message
Control model	Select SelectWithValue Cancel Operate CommandTermination TimeActivatedOperate		

	Parameters for control services	
Service parameter name	**Service parameter type**	**Value/Value range**
ctlVal	BOOLEAN	off (FALSE) | on (TRUE)

The "DPC" common data class is composed of a list of about 20 data attributes. Each attribute has a name, type, functional constraint, trigger option, value/value range, and an indication of whether the attribute is mandatory or optional.

At least all the mandatory attributes of all mandatory DATA classes of the "XCBR" in Table 6 make up the attributes of the "XCBR". Optional DATA classes (for example, Point on wave switching capability – POWCap) and optional data attributes (for example, origin – Originator) shall be used if required by an application.

All (possible) data attributes of the DATA Pos derived from the common data class DPC are shown on the left hand side of Figure 64. An instance containing all data attributes is depicted in the middle. The DATA class Pos is contained in the logical device MyLD and in the logical node XCBR1. The second instance has just the five mandatory data attributes.

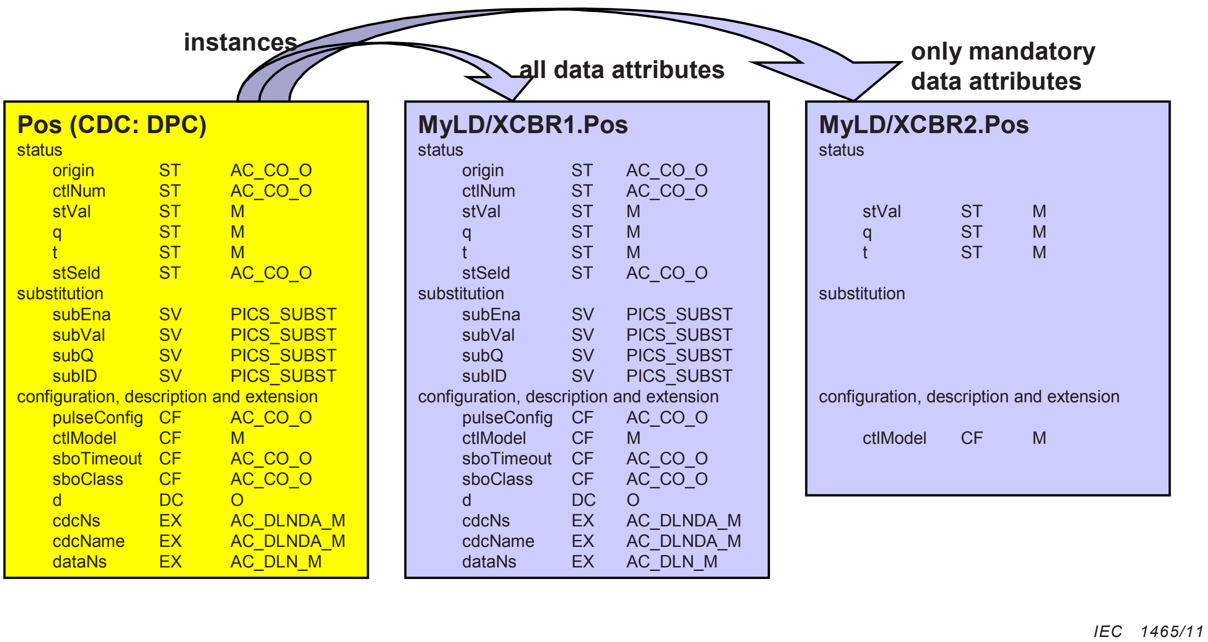


Figure 64 – Instances of a DATA class (conceptual)

During system design, the designer shall decide which data attributes are required to meet the required functionality of a logical node.

The conditions in the last column of the DATA class Pos are as follows (excerpt of IEC 61850-7-3 shown):

Abbreviation	Condition
M	Attribute is mandatory
O	Attribute is optional
PICS_SUBST	Attribute is mandatory, if substitution is supported for this object (for substitution, see IEC 61850-7-2)
AC_DLN_M	Applies to dataNs in all CDCs, dataNs shall be present if the name space of the DATA deviates from the name space defined in ldNs/lnNs
AC_DLNDA_M	The attribute shall be present, if the name space of the CDC deviates from either the name space defined in ldNs/lnNs or the name space defined in dataNs, or both
…	…

All DATA classes of the circuit breaker class together contain (i.e. when the common data classes are expanded) a total of more than 100 simple data attributes (all mandatory and optional data attributes counted).

11.3 Example 2 – Relationship of IEC 61850-7-2, IEC 61850-7-3, and IEC 61850-7-4

IEC 61850-7-4 specifies the application-specific semantic for logical node classes and the data classes that belong to logical node classes. The data classes represent structured information, for example, status, quality, or timestamp. A set of common simple and complex structures applicable in most applications are defined in IEC 61850-7-3 (common data classes).

Figure 65 depicts an example of the relation between the three parts. On the level of IEC 61850-7-4, two logical node classes "XCBR" and "XSWI" are exposed. Each logical node has a data class representing the “controllable double point position” (common data class: "DPC"). IEC 61850-7-3 defines a list of some 20 common data classes that can be used to describe the common functionality of data. One logical node instance XCBR1 is shown at the bottom of Figure 65. This instance can be accessed by services.

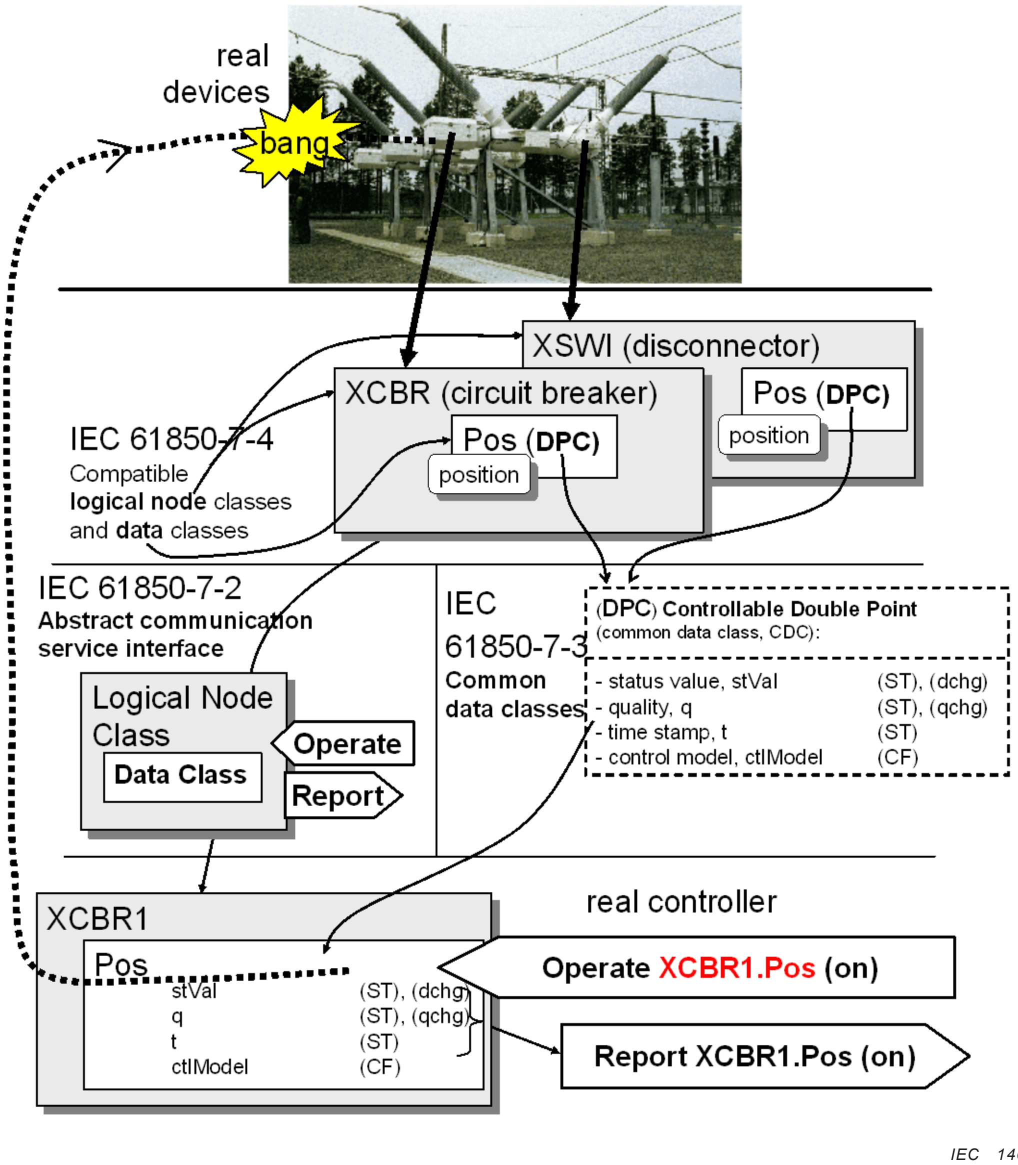


Figure 65 – Relation between parts of the IEC 61850 series

The common data class "DPC" comprises a re-usable list of attributes such as status value, quality or time stamp (values that can be reported: ST), and control model (value that can be configured: CF). The attributes of the common data classes have standardised data attribute names, for example, "stVal", or "q". These names are used in the communication (independent of the SCSMs) and in the substation configuration (language) according to IEC 61850-6.

The status value stVal has an additional information about when to trigger a report (dchg - trigger to send a report on value change of the status value). Reports can also be triggered by changes of the quality q attribute qchg.

An application of the logical node class and data class (see IEC 61850-7-4), common data class (see IEC 61850-7-3), and the common logical node, data class, and services (see IEC 61850-7-2) in a real system is shown at the bottom of Figure 65. The service (Operate "XCBR1.Pos" = on) closes the circuit breaker. The service (Report "XCBR1.Pos") informs the receiver about the current position change with time stamp and quality information. After a successful switching process, the "stVal" data attribute has the new state information.

12 Formal specification method

12.1 Notation of ACSI classes

IEC 61850-7-2 shall use the class notation as depicted in Table 8.

Table 8 – ACSI class definition

ABC class		
Attribute name	**Attribute type**	**Value/value range/explanation**
Attribute1 [1..n]	Type1	
Attribute2 [0..n]	Type2	
Services Service1 Service2		

The class name in the tables shall be written in CAPITAL letters of Tahoma format. The attributes of a class shall have an attribute name and an attribute type. The multiplicity of the attributes shall be:

– [0..n] – attribute may be available zero to *n* times (for example DataSet [0..n] in the GenLogicalNodeClass);
– [1..n] – attribute may be available one to *n* times (for example DataObject [1..n] in the GenLogicalNodeClass);
– [0..1] – attribute may be available zero or one times (for example SettingGroupControlBlock [0..1] in the GenLogicalNodeClass).

NOTE The service parameter multiplicity (in the service tables) applies the same syntax.

The services defined for a class shall be contained in the last row.

In the text, all class names shall be bold letters of Tahoma format (for example **GenLogicalNodeClass**) to differentiate “normal” text and standardised (reserved) names. In addition, other key words such as attribute names etc. shall be in bold letters of Tahoma format (for example **LNRef**).

12.2 Class modelling

12.2.1 Overview

IEC 61850-7-x series uses an approach based on object modelling to describe the object and service models. In this modelling technique, object classes, the characteristics of such classes, and services (methods) on those classes are described. The classes defined aid in the understanding of the intent of IEC 61850-7-2 service procedures and their effects. In implementing IEC 61850-7-2, these classes are mapped to specific communication systems (SCSMs). A real system maps the concepts described in the model to the real device. Hence, as viewed externally, a device that conforms to IEC 61850-7-2 and a specific SCSM shall exhibit the characteristics described in these class models.

Figure 66 depicts the class model of IEC 61850-7-x series. The SERVER class, which represents the external behaviour of a device, is composed of (1 to *n*) LOGICAL-DEVICE classes. LOGICAL-DEVICE classes comprise two or more LOGICAL-NODE classes (2 to n).

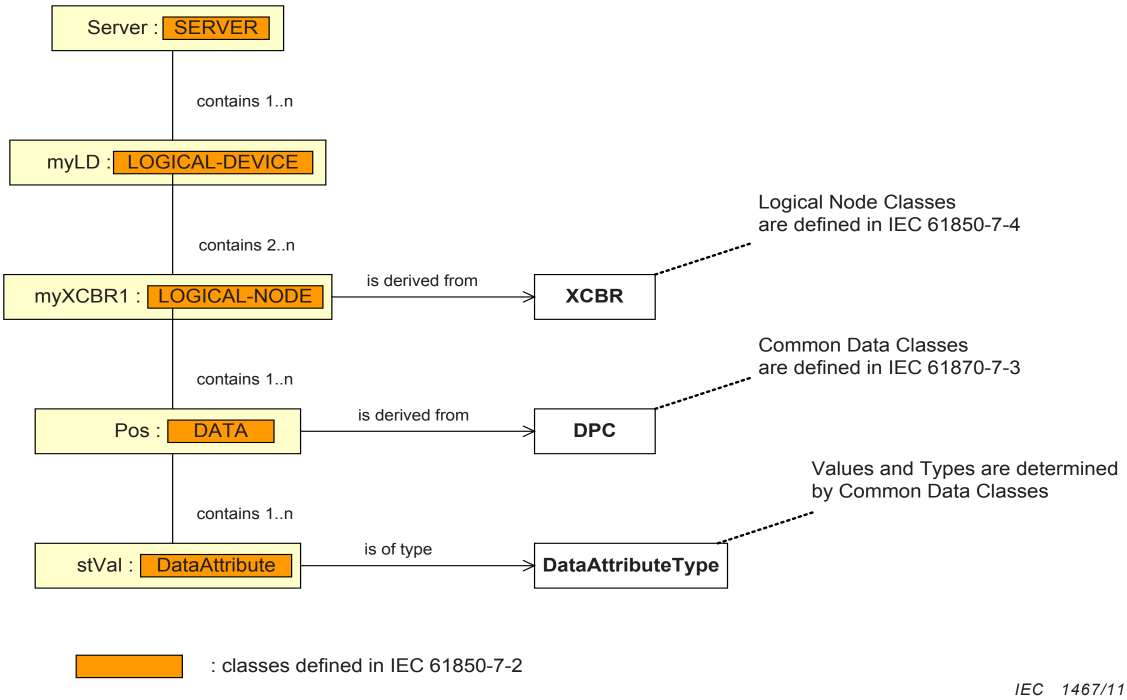


Figure 66 – Abstract data model example for IEC 61850-7-x

The logical node instance myXCBR1 is derived from the class XCBR which is defined in IEC 61860-7-4. XCBR has a lot of data. myXCBR1 may implement just a subset of XCBR. The data instance Pos is derived from the Common Data Class DPC which is defined in IEC 61860-7-3. Again, here, DPC has many DataAttributes and Pos may implement just a part of the whole set. Finally, the DataAttribute instance stVal has a defined type (CODED ENUM). Those DataAttributeTypes are determined by the common data class involved and are mainly defined in IEC 61860-7-3 but also in IEC 61860-7-2.

Each class is characterised by a number of attributes that describe the externally visible feature(s) of all instances of this class. Each instance of a class uses the same attribute types, but has specific values (the instance-specific values) for the attributes. The values of these attributes are defined by IEC 61850-7-x series or may be established by IEC 61850-7-x services; hence a change in the device may be modelled by a change in one or more attribute values of an instance.

The following subclauses discuss examples of the structure of a class defined in the IEC 61850 series.

12.2.2 Common data class

The attributes of the single point status class are (according to IEC 61850-7-3) defined as depicted in Table 9.

Table 9 – Single point status common data class (SPS)

SPS class					
Attribute name	**Attribute type**	**FC**	**TrgOp**	**Value/value range**	**M/O/C**
DataName	Inherited from GenDataObject Class or from GenSubDataObject Class (see IEC 61850-7-2)				
DataAttribute					
				status	
stVal	BOOLEAN	ST	dchg	TRUE | FALSE	M
q	**Quality**	ST	qchg		M
t	TimeStamp	ST			M
				substitution	
subEna	BOOLEAN	SV			PICS_SUBST
subVal	BOOLEAN	SV		TRUE | FALSE	PICS_SUBST
subQ	Quality	SV			PICS_SUBST
subID	VISIBLE STRING64	SV			PICS_SUBST
blkEna	BOOLEAN	BL			O
				configuration, description and extension	
d	VISIBLE STRING255	DC		Text	O
dU	UNICODE STRING255	DC			O
cdcNs	VISIBLE STRING255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING255	EX			AC_DLN_M
Services					
As defined in Table 11.					

The first column represents the name of the attribute, the second specifies the attribute type. An attribute that is composed of several components is defined as depicted in the example of Table 10 (excerpt of Quality type of IEC 61850-7-3).

The components of the Quality (for example, validity or detailQual) are data attribute components. The attribute types of the data attribute components (for example, PACKED LIST or CODED ENUM) are defined in IEC 61850-7-2.

Table 10 – Quality components attribute definition

Quality type definition			
Attribute name	**Attribute type**	**Value/value range**	**M/O/C**
	PACKED LIST		
validity	CODED ENUM	good | invalid | reserved | questionable	M
detailQual	PACKED LIST		M
overflow	BOOLEAN		M
outOfRange	BOOLEAN		M
badReference	BOOLEAN		M
oscillatory	BOOLEAN		M
failure	BOOLEAN		M
oldData	BOOLEAN		M
inconsistent	BOOLEAN		M
inaccurate	BOOLEAN		M
source	CODED ENUM	process | substituted DEFAULT process	M
test	BOOLEAN	DEFAULT FALSE	M
operatorBlocked	BOOLEAN	DEFAULT FALSE	M

The FC column specifies the functional constraint, if applicable. The functional constraint indicates which services can be used to access the values of the data attributes. Table 11 shows which services are allowed for the status information.

Table 11 – Basic status information template (excerpt)

Basic status information template			
Services (see IEC 61850-7-2)			
The following services are inherited from IEC 61850-7-2. They are specialised by restricting the service to attributes with a functional constraint as specified below.			
Service model of IEC 61850-7-2	**Service**	**Service applies to Attr with FC**	**Remark**
Data model	SetDataValues GetDataValues GetDataDefinition GetDataDirectory	DC, CF, SV **ALL** **ALL** **ALL**	
Data set model	GetDataSetValues SetDataSetValues	**ALL** DC, CF, SV	
Reporting model GSE model Sampled values model	Report SendGOOSEMessage SendGSSEMessage SendMSVMessage SendUSVMessage	**ALL** ST ST ST ST	As specified within the data set that is used to define the report content

The services applicable are listed in the third column. For all data that inherit the attributes from the common data class SPS (see Table 11), the attributes with FC=**ST** can be accessed by the following services (indicated by the key word **ALL** and ST):

GetDataValues

GetDataDefinition

GetDataDirectory

GetDataSetValues

Report

SendGOOSEMessage

SendGSSEMessage

SendMSVMessage

SendUSVSEMessage

Each group of common data classes defined in IEC 61850-7-3 has a table like Table 11 to specify the services supported (or allowed).

The trigger options TrgOp specify the possible trigger conditions that may cause to send a report or to log events. The service procedures shall be as specified in Table 12.

Table 12 – Trigger option

TrgOp	Semantics	Services allowed
dchg	data-change	A report or a log entry shall be generated due to a change of the value of the data attribute.
qchg	quality-change	A report or a log entry shall be generated due to a change of the value of the quality attribute.
dupd	data value update	A report or a log entry shall be generated due to freezing the value of a freezable attribute or updating the value of any other attribute. An updated value may have the same value as the old value.
Empty field	No trigger option is defined	No report or log entry shall be generated.

As depicted in Figure 67, the value of a data attribute that provides a specific TrgOp (trigger option) shall be monitored for reporting and logging if the report control block has enabled the specific trigger option (TrgOps). In the example below, the TrgOps is dchg; the TrgOp of the data attribute is dchg for the first, dupd for the second, and qchg for the last data attribute. Reports are sent on data changes only, because only dchg is enabled in the report control block. In the second example, all changes will be reported. In addition, a report will be sent on the expiration of the integrity period.

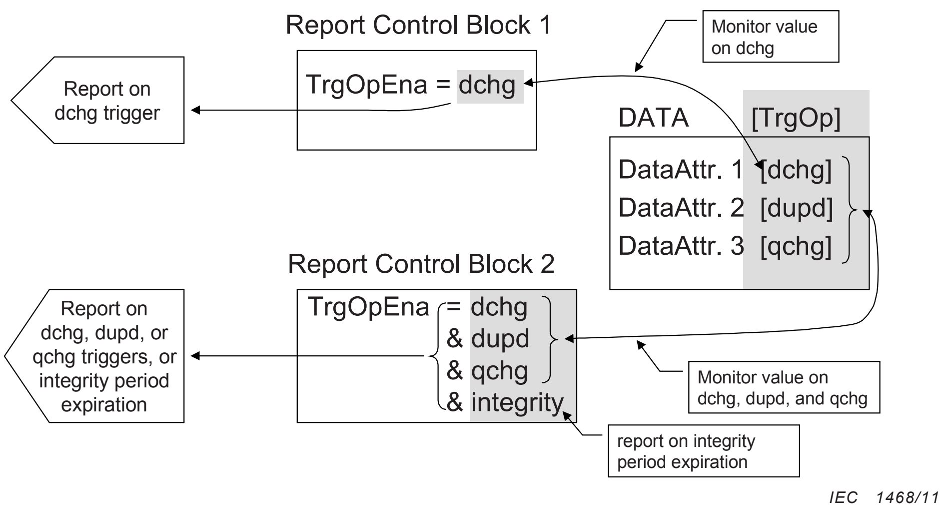


Figure 67 – Relation of TrgOp and Reporting

The column "value/value range" may contain enumerations (for example, stop | lower | higher | reserved); where "|" separates the values. The last column indicates if the attribute is mandatory, optional, conditional mandatory, or conditional optional.

12.2.3 Logical node class

Table 13 shows the table of a basic logical node class defined in IEC 61850-7-2: the GenLogicalNodeClass. The logical nodes contained in IEC 61850-7-4 inherit all definitions from this basic logical node class.

Table 13 – GenLogicalNodeClass definition

GenLogicalNodeClass		
Attribute name	**Attribute type**	**Explanation**
LNName	ObjectName	Instance name of an instance of LOGICAL-NODE
LNRef	ObjectReference	Path-name of an instance of LOGICAL-NODE
DataObject [1..n]	GenCommonDataClass	
DataSet [0..n]	DATA-SET	
BufferedReportControlBlock [0..n]	BRCB	
UnbufferedReportControlBlock [0..n]	URCB	
Log [0..n]	LOG	
LogControlBlock [0..n]	LCB	
The following attributes shall only be available if their support is explicitly stated in the definition of a compatible LN class, e.g., in IEC 61850-7-4.		
SettingGroupControlBlock [0..1]	SGCB	
GOOSEControlBlock [0..n]	GoCB	
GSSEControlBlock [0..n]	GsCB	
MulticastSampledValueControlBlock [0..n]	MSVCB	
UnicastSampledValueControlBlock [0..n]	USVCB	
Services GetLogicalNodeDirectory GetAllDataValues		

The columns of the class table are: attribute name, attribute type, and explanation.

The lines represent the attributes of the logical node.

Each logical node class has a logical node name (LNName). IEC 61850-7-4 defines many logical node class names, for example, XCBR for the logical node “circuit breaker”.

The logical node reference (LNRef) is used to reference an instance of a logical node. An example is MyLD/XCBR1. That means there is an instance with the name XCBR1 of class XCBR that is contained in the logical device MyLD.

The logical node contains one or more data (DataObject). Data represent the function (and semantic) of the logical node. The logical nodes defined in IEC 61850-7-4 each contain a list of a few to many data.

Data sets (DataSet) contained in a logical node may reference data and data attributes included defined in the same logical node or contained in any other logical node of any logical device.

Report and log control blocks may be contained in a logical node as well. IEC 61850-7-4 does not define any common report or log control blocks nor any common data sets. It is up to the system design to define specific data sets and report and log control blocks.

The presence of the last five optional attributes is explicitly stated in the definition of a compatible LN class, e.g., in IEC 61850-7-4.

The services that operate on the logical node are the two services listed at the end of the table (GetLogicalNodeDirectory and GetAllDataValues) and ALL services defined with the classes listed in the column "Attribute type". All classes that are used as types have their own services. The class DATA has several services, for example, GetDataValues and SetDataValues.

From this point of view, the logical node comprises all services of all classes that are used to build up the logical node class.

12.3 Service tables

IEC 61850-7-2 provides unconfirmed and confirmed services. The mapping of the confirmed services requires that the used application layer provides a method that serves to identify the request and the accompanying response within an association. The service tables summarise the parameters that are required for the processing of a particular service:

Parameter name
Request
Parameter 1…
Parameter n
Response+
Parameter 1…
Parameter n
Response−

NOTE The service tables of the services defined in IEC 61850-7-2 do not show all parameters required in concrete interface implementations; for example the parameter "association" or "retransmission time" are not depicted in the abstract service tables. These tables are abstract – local issues and concrete protocol issues are not shown. These specific issues are not required to understand the semantic and behaviour of the service.

Usually the service table provides the request and response parameters of a specific service. Each parameter and the effect this parameter has on the processing of the service is described in IEC 61850-7-2 in an abstract way. The sequences of the service primitives of confirmed services are depicted in Figure 68.

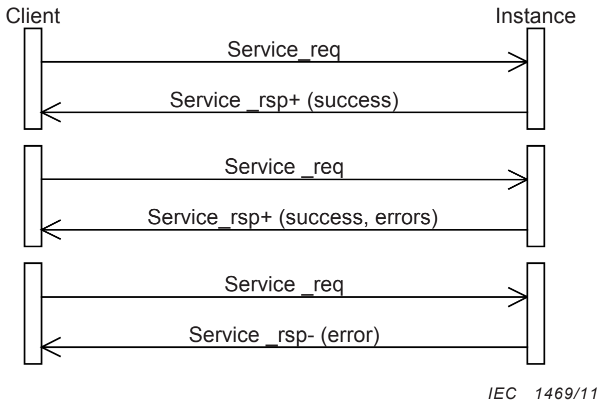


Figure 68 – Sequence diagram

12.4 Referencing instances

The standard differentiates between object names and object references. Object names (see Figure 69) identify an instance of a class at one hierarchy level (for example, "Mod" at the Data level or "Q0XCBR1" at the logical node level). "Q0" is a prefix and the "1" is a suffix to the name "XCBR". The concatenation of all the object names forms the object reference (for example, "MyLD/Q0XCBR1.Mod.stVal").

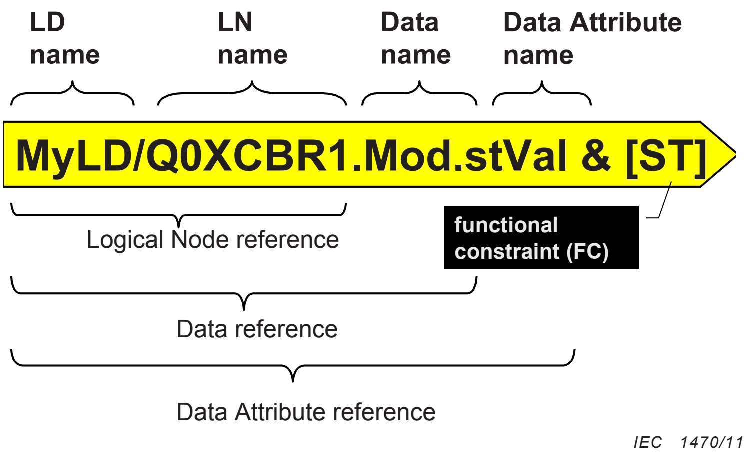


Figure 69 – References

The data attribute reference identifies a specific data attribute of a data instance. Data reference identifies a complete data instance with all its data attributes.

The logical node name "XCBR" may be enhanced by a prefix (for example "Q0"). It shall also contain a logical node instance number (for example "1"). Those together with the logical node class name build the object name Q0XCBR1. The standardisation of prefixes and logical node instance numbers is outside the scope of this standard. All data names from IEC 61850-7-4 and data attribute names from IEC 61850-7-3 shall be used unchanged for instances. Extension rules for logical nodes, data and common data classes are specified in Clause 14 of this document.

Functional constraints (FC) play a crucial role in the definition of the information models and in the services to access the various parts of the information model. To simplify the description of service parameters, the following definitions (short cuts) have been defined to reduce the amount of parameters in a service request or response:

– functionally constrained data (FCD),
– functionally constrained data attribute (FCDA), and
– control block service parameters.

The conceptual use of FCD and FCDA is shown in Figure 70 using the MMS mapping.

NOTE MMS uses the notation “$” and map the functional constraint FC between LNName and DataName.

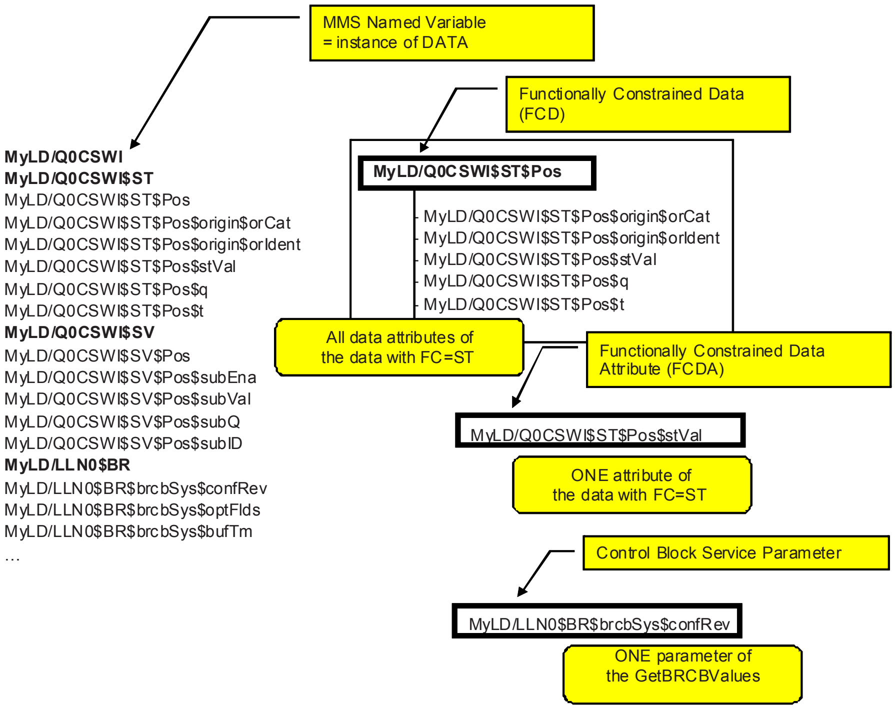


Figure 70 – Use of FCD and FCDA

As for the logical device names (MyLD in Figure 69), they are not standardized and there are no reserved names either.

The logical device names introduced in a project are to be described by the substation configuration language (SCL).

According to IEC 61860-6, SCL allows for two main options for signal naming: product-related naming and function-related naming. The former method is mandatory while the latter is optional.

For example, in case of product related naming, the name of a logical device is the name of the IED concatenated with the IED LD instance name:

S1E1Q1SB1LD2/Q0XCBR1 where:

– S1E1Q1SB1[3] is the name of the IED,
– LD2 is the name of the IED LD instance,
– XCBR1 is the name of the LN instance,
– Q0 is the LN prefix.

With function related naming, the name of the LD can freely be edited. The example below shows a name based on the functional structure of the substation. In this example, the name of the logical device makes no reference to the underlying SAS.

---

3 SI, E1, Q1 and SB1 are names chosen according to IEC 61346-1 and IEC 61346-2 whose application is defined in IEC 61850-6.

S1E1Q1/QA1XCBR2 where:

- – S1 is the name of the substation,
- – E1 is the name of the voltage level,
- – Q1 is the name of the bay,
- – QA1 is the LN prefix corresponding to the primary equipment name[4].

In both cases, the name of the logical device shall be unique in a subnetwork, which is easily achieved in case of product related naming.

The function-related naming convention could be used in the specification phase of the SAS. However, when implementing the specification in a real SAS, it may not be possible to apply the naming convention of the specification to the SAS. In the preceding example, Q0 will not be changed to QA1 if the former value is fixed (not configurable) in the IED S1E1Q1SB1.

Almost all services of IEC 61850-7-2 use the object references as a service parameter. These object references shall not be changed by a SCSM. They may be mapped in a SCSM to unique numbers.

Figure 71 shows examples of object names and object references. The example at the top (first four lines) can be just four class definitions (not yet instantiated) or four instances of the classes "E1QA5/XCBR.Pos.stVal", "...q", "...t", "...ctlModel". The object references do not in this case indicate if object references refer to classes or instances. The context where these references are used has to provide sufficient information to know what is meant (class or instance).

The other examples refer to instances only.

The LD name "E1QA5" and its structure are outside the scope of the IEC 61850 series. The functional constraint (FC) is not shown in the object reference. The FC information may be mapped into the object reference in a SCSM; IEC 61850-8-1 maps the FC between logical node name and the data name ("XCBRSTPos").

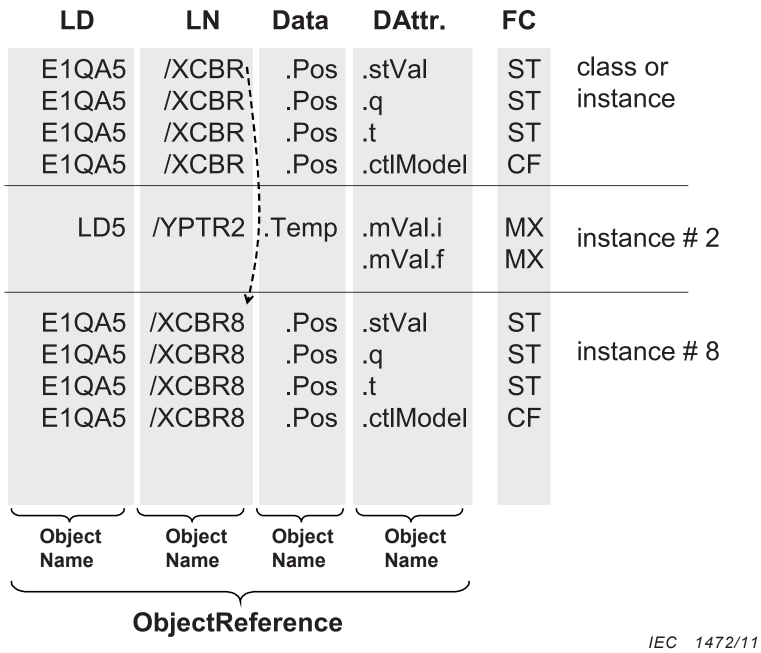


Figure 71 – Object names and object reference

4 SI, E1, Q1 and QA1 are names chosen according to IEC 61346-1 and IEC 61346-2.

13 Name spaces

13.1 General

Each class defined in IEC 61850-7-4 (for example circuit breaker LOGICAL-NODE, XCBR), IEC 61850-7-3 (for example double point status, DPS), and IEC 61850-7-2 (for example buffered report control block, BRCB) has a class name and a specific meaning associated with the class. Almost all classes are made up of other classes. The hierarchical class model of IEC 61850-7-x provides a naming hierarchy. Each name in the hierarchy has a semantic in the context where the name is used.

Figure 72 shows an example of the definition of names and their semantic in the context of a substation. The application to be modelled and defined in the IEC 61850 series may be the circuit breaker sketched in the middle at the top. The standardized circuit breaker is modelled as a LOGICAL-NODE with the specific class name XCBR. The circuit breaker is part of the LOGICAL-DEVICE with the name SUBST2. Among other attributes, the circuit breaker has the information (modelled as DATA class) that represents the position named Pos. Among other information, the position has a status (modelled as DATA-ATTRIBUTE) named stVal. The status value stVal has four values that represent the possible status of the real breaker.

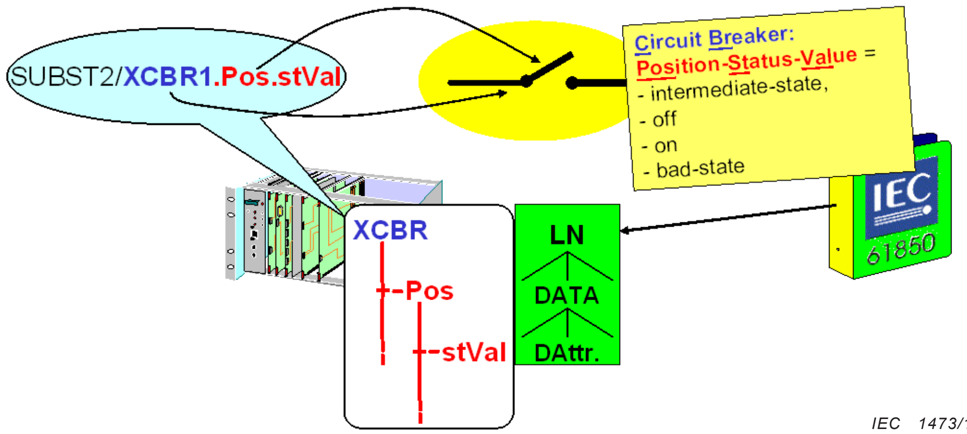


Figure 72 – Definition of names and semantics

If only the classes defined in the IEC 61850-7-x series are used to build a LOGICAL-DEVICE, then the semantic is as defined in the IEC 61850-7-x series.

For applications that need additional LOGICAL-NODEs, DATA or DATA-ATTRIBUTE rules are provided to unambiguously interpret the names, i.e., to understand the content and meaning of an instance of a class in a specific context. Especially in the case where the same name, for example Pos, has been defined carrying different meanings, the standard needs to prevent a conflict with a single name having multiple definitions. Two meanings of a name of a DATA item are shown in Figure 73. The instance name Pos of a DATA item is used in the circuit breaker and in the nacelle of a wind turbine (LOGICAL-NODE with the name WNAC). In the context of a wind turbine, the position is defined as the plain angle of the nacelle. The value is a measured analogue in degrees (SI unit).

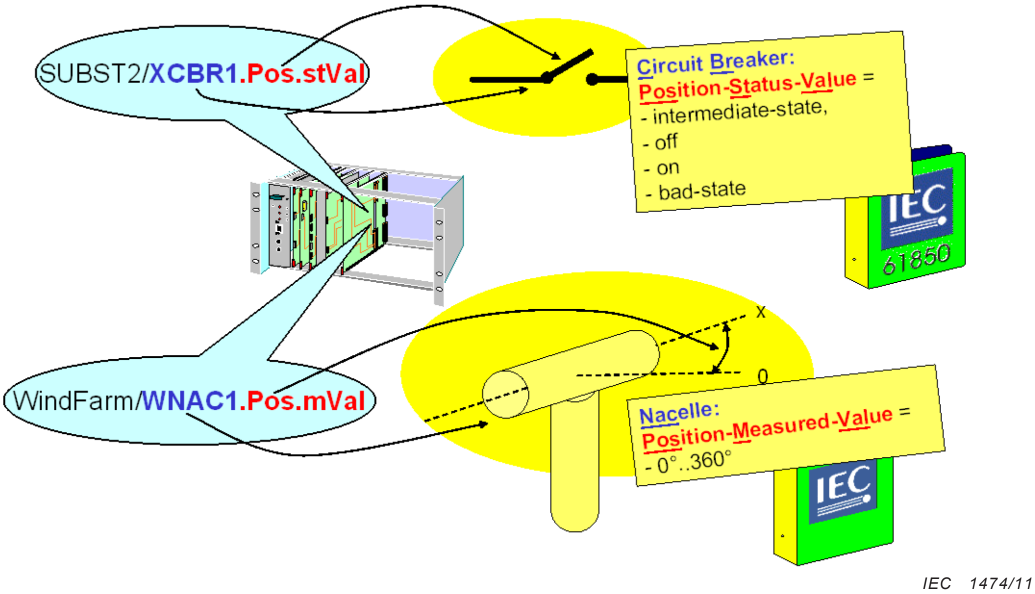


Figure 73 – One name with two meanings

The name Pos is used in two different contexts: substation (IEC 61850 series) and wind turbine (IEC 61400 series).

Using a reference to the defined context of the data, i.e., the concept of the name space provides a means to uniquely identify the complete semantic of an instance of a LOGICAL-DEVICE, i.e., the semantic of all its LOGICAL-NODEs, DATA, DATA-ATTRIBUTEs, and all other instances in the context of its use.

The concept of the name space allows the distinction of classes defined by different groups – as long as the name spaces have unique identifiers.

Any instance of a class of the classes defined in the IEC 61850 series and any instances of a class defined as extension to the classes of the IEC 61850 series shall provide sufficient name space information to allow the unambiguous interpretation of the semantic of the instance. The instances of classes are marked to identify the name space.

13.2 Name spaces defined in the IEC 61850-7-x series

IEC 61850-7-4 and IEC 61850-7-3 define name spaces for application specific classes. IEC 61850-7-2 defines a name space for communication related (service) classes such as BUFFERED-REPORT-CONTROL-BLOCK, LOG-CONTROL-BLOCK, LOGICAL-NODE, DATA, DATA-SET.

NOTE The mixed use of data with name spaces from other communication standards or from private definitions always implies that the conceptual approach of the data model is the same.

As depicted in Figure 74, a name space is conceptually speaking a class repository containing various classes. A logical device is built up by instances derived from the classes of the repository. The standard class repository that comes with the IEC 61850 series is shown on the right-hand side of the figure. An example of an additional name space is shown on the left-hand side.

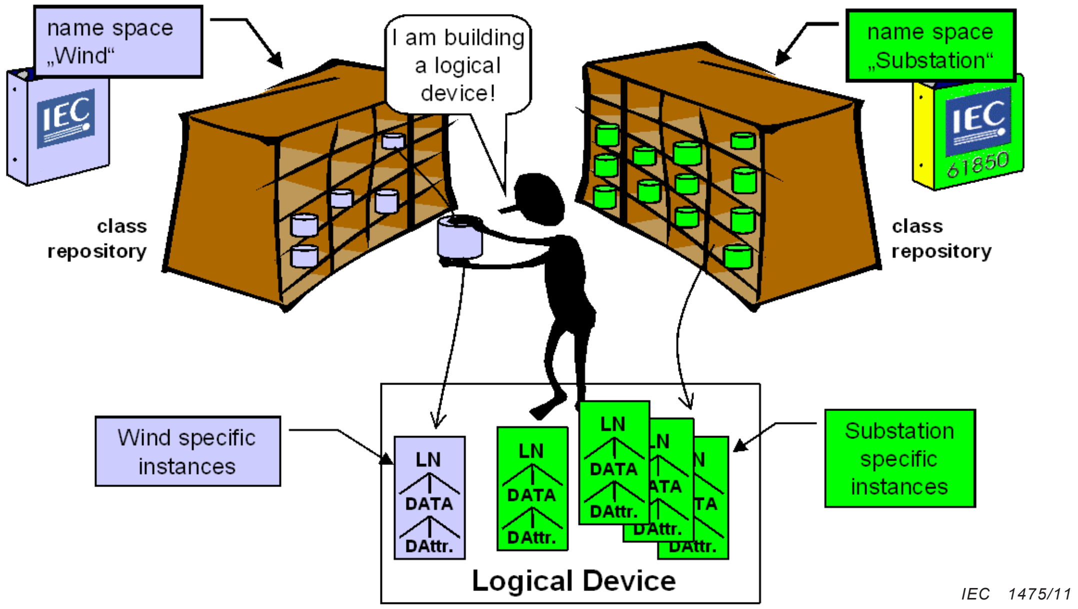


Figure 74 – Name space as class repository

The instances that are part of the logical device are coloured differently. The instances derived from the IEC 61850 series are green and the instance derived from the other standard is blue. As shown in Figure 75, the designation of the name space is represented by the attribute “logical device name space”:

ldNs = IEC 61850-7-4:2003

In this example, the name space “IEC 61850-7-4:2003” indicates that ALL instances within this logical device are derived from the 2003 editions of IEC 61850-7-4, IEC 61850-7-3, and IEC 61850-7-2. The logical device name space could be understood as the prime name space; even if there is just one name. The attribute ldNs is an attribute contained in the name plate of the logical node zero (LLN0).

As long as all three documents (see IEC 61850-7-4, IEC 61850-7-3, and IEC 61850-7-2) are of the same edition, it is sufficient to refer only to the IEC 61850-7-4. IEC 61850-7-4 has normative references which apply to the other two documents. The underlying instances of LOGICAL-NODEs, DATA, and DATA-ATTRIBUTEs have an implicit name space (i.e., IEC 61850-7-4, IEC 61850-7-3, and IEC 61850-7-2) which is derived by the normative references in IEC 61850-7-4.

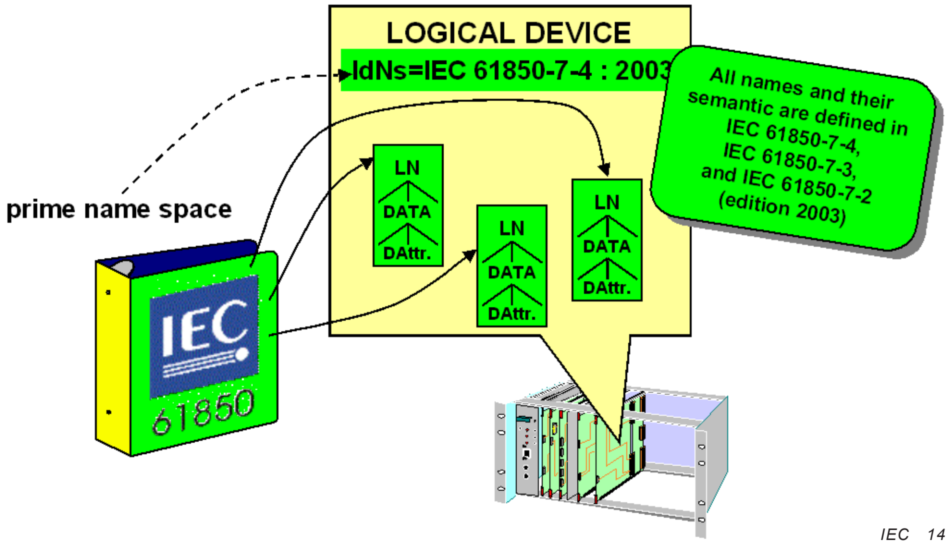


Figure 75 – All instances derived from classes in a single name space

As long as only optional additions are defined by a next version of a standard, the standard's name space IdNs is kept, although a version identification should be added, so that a human user can find the meaning. This is needed for backward compatibility between applications. This means however, that all applications use the 'may ignore' principle, i.e. ignore what they do not understand, which should normally be the case, because an automatic application will just use what it needs and knows. Problems can only come with 'generic' applications like pure communication drivers, which suddenly see new CDCs or FCs - and then have to 'ignore' them. For mandatory additions as well as 'private' extensions and additional application areas, a new name space shall always be defined.

The example instances of Figure 76 are derived from three different name spaces: IEC 61850-7-4, the other standard document, and a private name space (Vestas). Since the majority of instances are derived from IEC 61850-7-4, the logical device name space IdNs is still IEC 61850-7-4 (prime name space).

The logical node at the bottom has been derived from the other specification. This needs to be indicated with an explicit logical node name space lnNs with the value, for example, “Other Specification: year of publication”. The instances below that logical node are defined in the name space. The instances of data have implicitly the same name space. The third name space applied is a private name space: lnNs with the value “Vestas”.

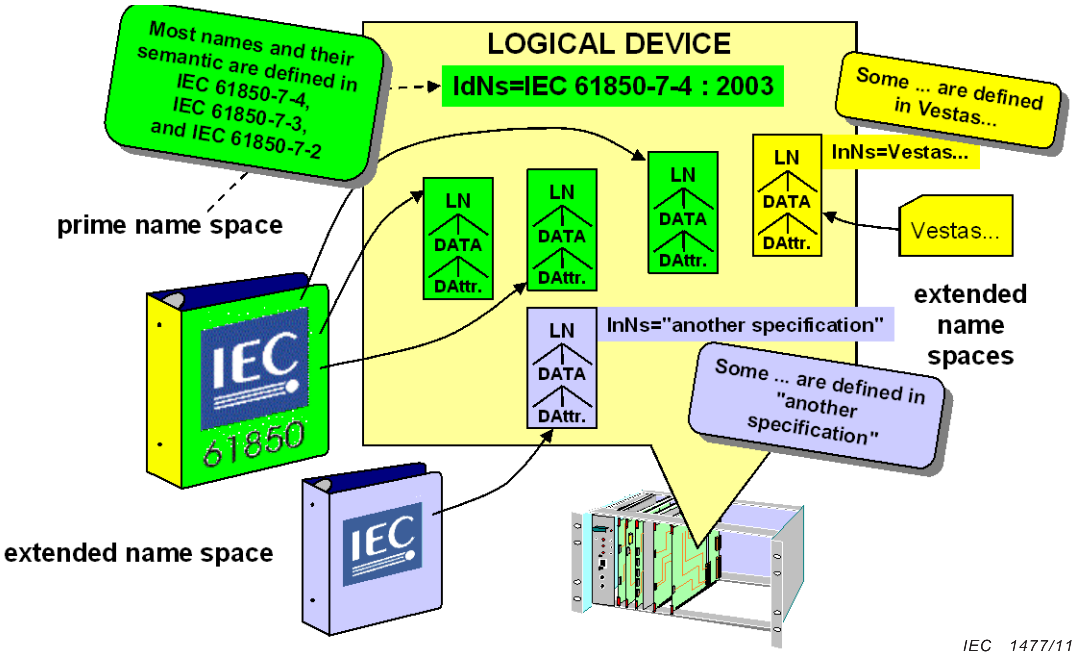


Figure 76 – Instances derived from multiple name spaces

The logical devices name space could be “IEC 61850-7-4:2003” or any other name space depending on the context in which a logical device is defined and used. The example given in Figure 77 shows how another specification could for example inherit all classes (LOGICAL-NODE, DATA, and common DATA classes) from IEC 61850-7-4 and IEC 61850-7-3. In that case, the other specification would define a superset name space. Since the basic sets from different standards or other definitions are maintained completely independently from each other, superset name spaces shall be avoided to minimise the risk of inconsistencies and not to endanger interoperability.

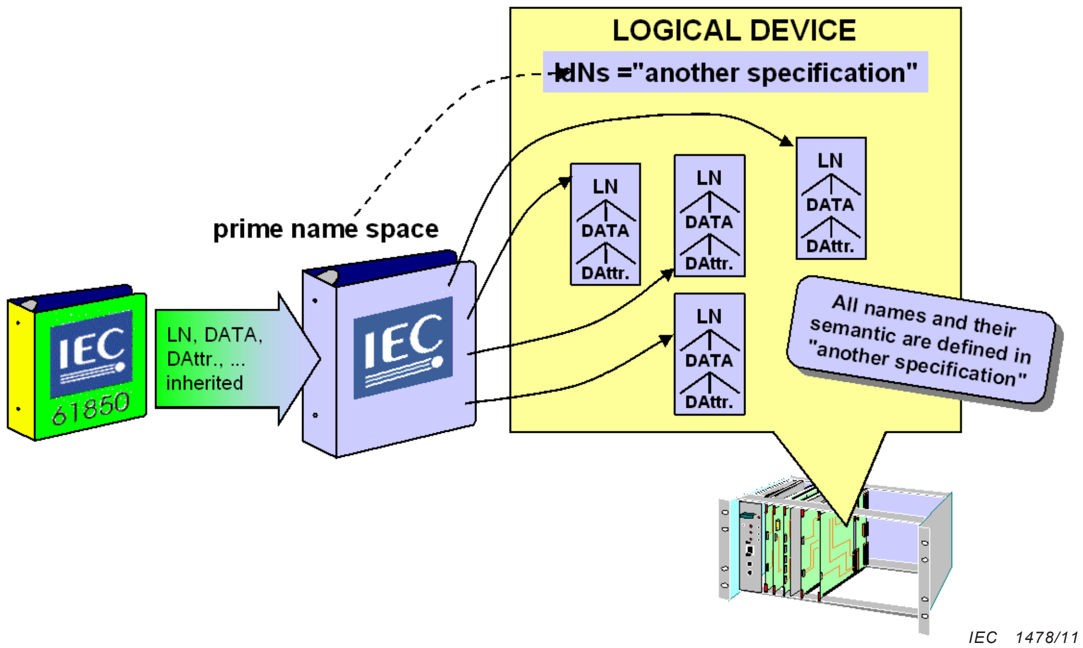


Figure 77 – Inherited name spaces

The ldNs has the value “other specification: year of publication”. Since all classes are contained in that single name space, the underlying instances have the implicitly the same name space. They do not need to have an explicit value for their name spaces.

13.3 Specification of name spaces

13.3.1 General

Within the IEC, name spaces are defined by entities (e.g. IEC TC 57) responsible for vocabularies used by specific domain applications. These entities are called owners of a name space. Entities may own several name spaces.

In the context of the IEC 61850 series, name spaces are defined as basic name spaces and domain application name spaces. Name spaces for domain applications shall import the definitions from basic name spaces.

The basic name spaces covered by the IEC 61850 series are:

– IEC 61850-7-3: specifies common data classes and enumerations;
– IEC 61850-7-2: specifies basic types, enumerations, common data classes, dataset classes, control block classes and the control class model.

The name spaces for domain applications covered by the IEC 61850 series are:

- IEC 61850-7-4: specifies LN and enumerations for substation automation applications;
- IEC 61850-7-410: specifies LN and enumerations for hydroelectric power plant applications;
- IEC 61850-7-420: specifies LN and enumerations for distributed energy resource applications;
- IEC 61850-7-XXX for future applications;
- IEC 61850-8-1: specifies control objects and their references used in the mapping of the control class model;
- IEC 61850-90-XXX: specifies extensions foreseen for IEC 61850 domain applications. The extensions are intended to be published in dedicated Technical Reports.

As name spaces evolve with time, either for enhancements of their content or for error corrections, some rules have to be defined by owners in order to avoid compatibility issues between different versions of a name space. Also, some rules for extensibility shall be provided by owners to allow third parties to make extensions to a name space in a way that do not jeopardize interoperability.

Clause 14 specifies rules for class version control and extension of classes, including the use of name spaces.

13.3.2 Specification

The following three name spaces shall be defined to provide sufficient information to allow the unambiguous interpretation of the instances of logical node classes, data object classes, DATA-SET classes, and the various CONTROL BLOCK classes:

a) Logical node name space shall be a technical specification containing the definition of the logical node classes (including the underlying classes and services) defined for a specific application domain (for example, Power utility automation),
b) Data name space shall be a technical specification containing the definition of the data object classes (including the underlying definitions and services) defined for a specific application domain (for example, Power utility automation), and
c) Common data class name space shall be a technical specification containing the definition of the common data classes (including the underlying definitions and services) defined for a specific application domain (for example, Power utility automation).

Name space names shall be structured as follows:

NamespaceID:VersionRevision

where

– NamespaceID is the identification of the name space. NamespaceID shall refer to a technical specification (paper specification, schema (XML, database), UML description, …), e.g. IEC 61850-7-4,

– In case of name spaces specified in Technical Reports (such as in the 61850-90-XXX series), the NamespaceID shall start with the prefix (Tr) to indicate that the name space shall be considered as a transitional name space, e.g. (Tr)IEC 61850-90-4:2011,

– Version refers to the year of publication of the specification. For the domain covered by Edition 1 of IEC 61850-7-4, IEC 61850-7-3, and IEC 61850-7-2, the default value is 2003. For Edition 2, the value shall be 2007,

– Revision refers to a revision indicator. This indicator is required to differentiate revisions of a specification between two versions. For Edition 2, the value shall be A. Subsequent revisions values shall be B, C, D, …

Name space names shall be explicitly defined within the medium describing the content (paper specification, schema (XML, database), UML description, …).

In addition, for Technical Reports, the following accompanying text shall be added to the definition of the name space name:

‘The content of this name space shall be considered as a transitional name space. This means that the content may be moved to an IS status with some changes, thus making possible backward incompatibilities with existing implementations.’

13.4 Attributes for references to name spaces

13.4.1 General

The following four attributes that reference name spaces are defined:

– Attribute logical device name space (ldNs) shall reference the prime technical specification used for the whole logical device.

– Attribute logical node name space (lnNs) shall reference the logical node name space of a single instance of LOGICAL-NODE.

– Attribute data name space (dataNs) shall reference the data name space of a single instance of DATA.

– Attribute common data class name space (cdcNs) shall reference the CDC name space of the CDC used for the definition of a single instance of DATA.

The common data classes contain the DataAttributes ldNs and lnNs as shown in the excerpt of Table 14.

NOTE 1 The conditions in the last column of Table 14 are defined in IEC 61850-7-3.

Table 14 – Excerpt of logical node name plate common data class (LPL)

LPL class					
Attribute name	**Attribute type**	**FC**	**TrgOp**	**Value/value range**	**M/O/C**
DataName	Inherited from Data Class (see IEC 61850-7-2)				
DataAttribute					
configuration, description and extension					
...	...	...	...	...	...
ldNs	VISIBLE STRING255	EX		shall be included in **LLN0** only; for example "IEC 61850-7-4:2003"	AC_LN0_M
lnNs	VISIBLE STRING255	EX			AC_DLD_M

The common data classes contain the DataAttributes cdcNs and dataNs as shown in the excerpt of Table 15.

Table 15 – Excerpt of common data class

Applied by all common data classes					
Attribute name	**Attribute type**	**FC**	**TrgOp**	**Value/value range**	**M/O/C**
DataName	Inherited from Data Class (see IEC 61850-7-2)				
DataAttribute					
...	...	...	...	...	...
configuration, description and extension					
cdcNs	VISIBLE STRING255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING255	EX			AC_DLN_M

NOTE 2 The conditions in the last column of Table 15 are defined in IEC 61850-7-3.

13.4.2 Attribute for logical device name space (ldNs)

The DataAttribute logical device name space ldNs shall be used to indicate the domain application name space used as the prime name space for the whole logical device, e.g. IEC 61850-7-4, IEC 61850-7-410, IEC 61850-7-420, etc.

The attribute ldNs shall be a DataAttribute of the name plate NamPlt of the logical node class LLN0. The attribute ldNs shall be as defined in the common data class LPL (logical node name plate) of IEC 61850-7-3. Its value shall be as specified in 13.3.2.

The attribute ldNs shall be available in every instance of the logical node class LLN0.

The ObjectReference for the DataAttribute ldNs shall be:

LDName/LLN0.NamPlt.ldNs

13.4.3 Attribute for logical node name space (lnNs)

The DataAttribute logical node name space lnNs shall be used to indicate the used name space for a specific logical node. The attribute shall only be available in a LN instance if its value differs from the ldNs value.

The attribute lnNs shall be a DataAttribute of the name plate NamPlt of a logical node. The attribute lnNs shall be as defined in the common data class LPL (logical node name plate) of IEC 61850-7-3.

The ObjectReference for the DataAttribute lnNs shall be:

LDName/LNName.NamPlt.lnNs

13.4.4 Attribute for data name space (dataNs)

The DataAttribute data name space dataNs shall be used to indicate the used name space for a specific data. The attribute shall only be available when the data class is added by extension to standardized LNs.

The attribute dataNs shall be a DataAttribute of the data. The attribute dataNs shall be as defined in the common data classes of IEC 61850-7-3.

The ObjectReference for the DataAttribute dataNs shall be:

LDName/LNName.DataName[.DataName[. ...]].dataNs

13.4.5 Attribute for common data class name space (cdcNs)

The DataAttribute common data class name space cdcNs shall be used to indicate the common data class name space used for the creation of a specific data.

Starting from Edition 2 of this standard, new versions of standardized common data classes shall only be made by the owner of the name space. As a consequence, private extension of an existing CDC is not encouraged.

The attribute cdcNs shall be a DataAttribute of the data. The attribute cdcNs shall be as defined in the common data classes of IEC 61850-7-3.

The ObjectReference for the DataAttribute cdcNs shall be:

LDName/LNName.DataName[.DataName[. ...]].cdcNs

14 Common rules for new version of classes and for extension of classes

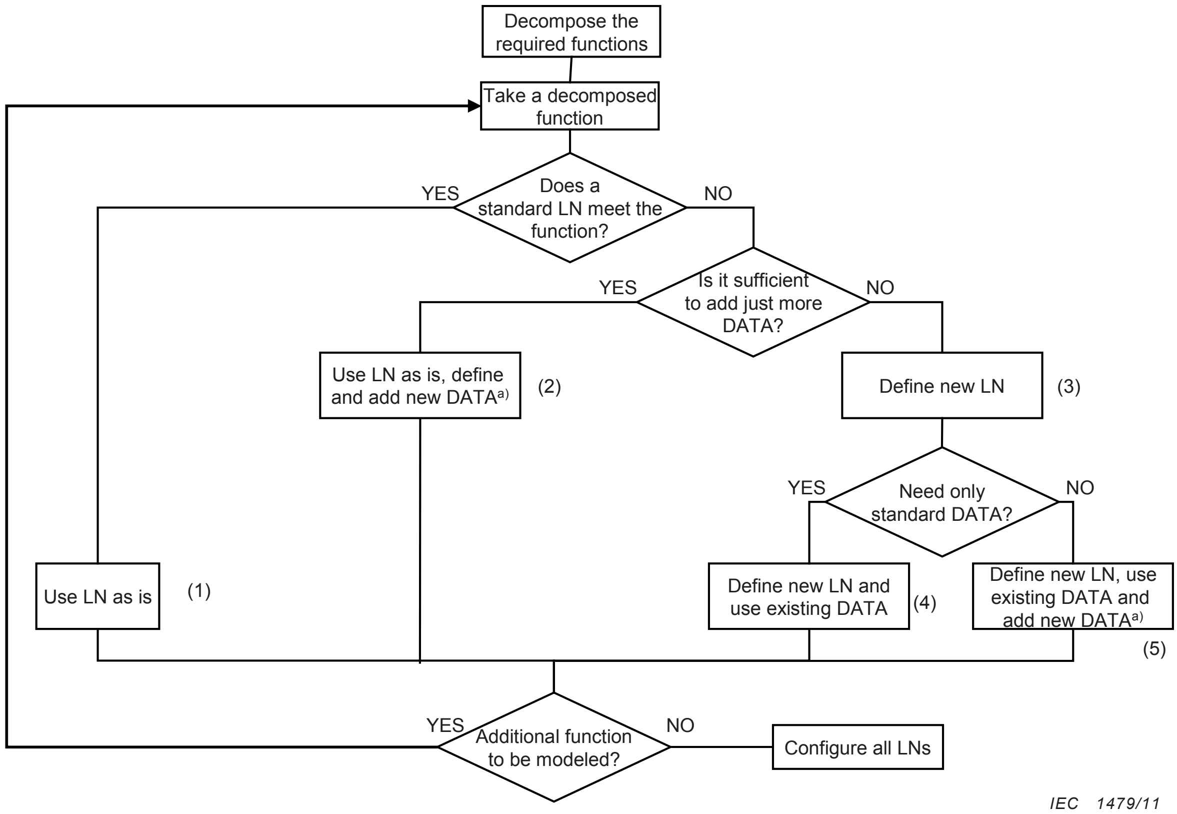
14.1 General

The rules defined in this clause apply to the following classes: logical node (LN) classes, data classes and common data classes (CDC). They shall be used for defining new versions of classes and for making extensions to existing classes.

In the following, classes which are part of a owner name space (e.g. IEC 61850-7-4) are called standardized classes.

14.2 Basic rules

The basic rules are depicted in the diagram shown in Figure 78. After having decomposed the required application function to a degree of granularity of the existing logical node classes of the application domain, e.g. substation in case of the IEC 61850 series that are listed in IEC 61850-7-4, the rules described in the following subclauses shall be applied:



[a] New DATA based on existing CDC.

Figure 78 – Basic extension rules diagram

14.3 Rules for LN classes

14.3.1 Use of standardized LN classes

The use of standardized LN classes shall follow the following rules:

- If there is any logical node class which fits the function to be modelled, an instance of this logical node shall be used with all its mandatory data (M). The rules of a unique instantiation can be found in IEC 61850-7-2. Step (1) in the diagram.
- If, in addition to the mandatory data (M), there are also optional data (O), which fit the function to be modelled, these optional data shall be used. Step (1) in the diagram.
- If there are the same data (M or O) which need to be instantiated several times, additional data with number extensions shall be used according to the rules for specialisation of data by use of number extensions, see 14.6.
- If there are dedicated versions of the function to be modelled with the same basic data (for example ground, phase, zone A, zone B, etc.), different instances of this logical node class shall be used, see 14.4.3.

14.3.2 Extensions to standardized LN classes made by third parties

Extensions to standardized LN classes may be made by third parties. However, third parties cannot change the semantics of a LN and therefore cannot change the name space name when making extensions. To not jeopardize interoperability, the rules for LN extensions are the following.

– The nature of LN extensions shall only be the addition of new optional data. Step (2) in the diagram.

– The new data object class name shall follow the naming conventions defined in IEC 61850-7-2 and:

  • be one of the common LN data object class defined in IEC 61850-7-4 or,

  • be a new data object class name.

– For building the new data object class name, the abbreviated terms of IEC 61850-7-4 shall be used if applicable. Only in other cases are new abbreviations out of the English name for the data allowed.

– The new data object class shall be assigned to one and only one of the common data classes defined in IEC 61850-7-3 or by other domains like IEC 61850-7-410 or IEC 61850-7-420.

– The new data object class shall be marked by a namespaceID different from the owner's namespaceID,

– The description of the new data object class shall be added to the IEC documentation of the provided specific system or the customer specific project.

14.3.3 New LN classes

For new LN classes, step (3) in the diagram, the following rules shall be applied:

– New LN classes within a name space shall only be defined by the owner of the name space.

– Third parties may expand the owner LN class set by defining new LNs outside the owner's name space. LNs defined by third parties shall be marked by a namespaceID different from the owner's namespaceID.

– The LN class name shall follow the naming conventions defined in IEC 61850-7-2 and be categorized according to the LN groups defined in IEC 61850-7-4.

– Data object class names within a LN class shall follow the naming conventions defined in IEC 61850-7-2.

– Data object class names defined by a third party shall:

  • be one of the LN data object class names defined in IEC 61850-7-4, step (4) in the diagram, or,

  • be a new data object class name, step (5) in the diagram.

– For building the new data object class name, the abbreviated terms of IEC 61850-7-4 shall be used if applicable. Only in other cases are new abbreviations out of the English name for the data allowed.

– Data object classes shall be assigned to one and only one of the common data classes defined in IEC 61850-7-3 or by other domains like IEC 61850-7-410 or IEC 61850-7-420.

– The description of the new LN class shall be added to the IEC documentation of the provided specific system or the customer specific project.

14.3.4 New versions of standardized LN classes made by name space owners

New versions of standardized LN classes shall only be made by the owner of the name space. The owner should not change the namespaceID for new versions. However, the name space should contain a new version number and a new revision number. For backward compatibility, definitions of new LN versions shall follow the following rules:

– new data shall be added as optional data,
– existing data which are no longer required, shall not be removed. Instead, their use should be discouraged,
– changing the semantics of existing LNs and data shall not be allowed. New LNs and data shall be added instead and the use of the older ones should be discouraged.

14.4 Rules for common data classes and control block classes

14.4.1 New common data classes and control block classes

New standardized common data classes and control block classes within a name space shall only be defined by the owner of the name space.

The naming conventions for common data class names and the notation for control block classes defined in IEC 61850-7-2 shall be used.

14.4.2 New versions of standardized common data classes

New versions of standardized common data classes shall only be made by the owner of the name space. The owner should not change the namespaceID for new versions. However, the name space should contain a new version number and a new revision number. For backward compatibility, new common data class versions shall follow the following rules:

– new attributes shall be added as optional attributes;
– existing attributes which are no longer required, shall not be removed. Instead, their use should be discouraged;
– changing the semantics of existing common data classes and attributes is not allowed. New common data classes and attributes shall be added instead and the use of the older ones should be discouraged.

14.4.3 New versions of control block classes

New versions of control block classes shall only be made by the owner of the name space. For backward compatibility, new control block class versions shall follow the following rules:

– new attributes shall be added as optional attributes;
– existing attributes which are no longer required shall not be removed. Instead, their use should be discouraged;
– changing the semantics of existing control block class attributes is not allowed. New control block classes shall be added instead and the use of the older ones should be discouraged.

14.5 Multiple instances of LN classes for dedicated and complex functions[5]

14.5.1 Example for time overcurrent

Logical node class name: PTOC (time overcurrent)		
LN instance name (with LN-Prefix)	Meaning	Meaning of “Start value” StrVal
GndPTOC1	Ground fault detection	“Ground Start Value”
PhsPTOC1	Phase fault detection	“Phase Start Value”

The semantics of the different instances may be given in the description attribute “d” of data NamPlt (name plate).

14.5.2 Example for PDIS

Logical node class name: PDIS (distance)	
LN instance name (without LN-Prefix)	Meaning
PDIS1	Zone 1 of the distance protection
PDIS2	Zone 2 of the distance protection
PDIS3	Zone 3 of the distance protection
Etc.	Etc.

The semantics of the different instances may be given in the description attribute “d” of data NamPlt (name plate).

14.5.3 Example for power transformer

Logical node class name: YPTR (power transformer)	
LN instance name (without LN-Prefix)	Meaning
YPTR1	Transformer unit phase L1
YPTR2	Transformer unit phase L2
YPTR3	Transformer unit phase L3

The semantics of the different instances may be given in the description attribute “d” of data NamPlt (name plate).

14.5.4 Example for auxiliary network

Logical node class name: ZAXN (auxiliary network)	
LN instance name (without LN-Prefix)	Meaning
ZAXN1	220 V DC
ZAXN2	60 V DC
ZAXN3	380 V DC

The semantics of the different instances may be given in the description attribute “d” of data NamPlt (name plate).

---

5 The examples supplied in this clause are for informative purpose only and are therefore non normative.

14.6 Specialisation of data by use of number extensions

Standardised data names in logical nodes provide a unique identification. If the same data (i.e. data with the same semantics) are needed several times as defined, additional data with number extensions shall be used. The rules for number extensions shall follow the naming conventions defined in IEC 61850-7-2 and be as follows:

– the number extension usage shall only be defined by the owner of the data namespace. This shall be done by adding the number extension 1 to a data object name (e.g. data1),
– data with no number extension shall not be extended by third parties,
– data with the number extension 1 can be extended. Number extensions may be ordered or not (1,2,3,4, or, 1,2,19,25),
– if only one instance of an extendable data is present in an LN, it shall have the number extension "1".

Examples are given hereafter:

Logical node class name: GGIO (generic process I)	
Data name: Ind1 (General indication - binary input)	
Ind1	General indication -binary input 1
Ind2	General indication -binary input 2
Ind3	General indication -binary input 3
Ind4	General indication -binary input 4

14.7 Examples for new LNs

New LN "Automatic door entrance control"

1st character Logical node group indication	2nd character	3rd character	4th character	New LN	Value of ADEC.NamPlt.lnNs
A for "Automatic Control"	**D**oor	**E**ntrance	**C**ontrol	**ADEC** "Automatic door entrance control"	**myVendor1Extension:2010A**

New LN "Fire protection"

1st character Logical node group indication	2nd charact er	3rd character	4th character	New LN	Value of ZFPT.NamPlt.lnNs
Z for " Further equipment"	**F**ire	**P**rotection	**T**ransformer	**ZFPT** "Fire protection of a power transformer"	**myVendor2Extension:2010A**

14.8 Example for new Data

New Data "Colour of Transformer Oil"

Data name	Attribute type (CDC)	Value of ColrTOil.dataNs
ColrTOil	INS	**myVendor1Extension:2010A**

Annex A
(informative)

Overview of logical nodes and data

A.1 Compatible logical node classes and data classes (IEC 61850-7-4)

A.1.1 List of LN groups (IEC 61850-7-4)

A list of all groups of logical nodes is contained in Table 1 of this part of IEC 61850.

A.1.2 LN classes (IEC 61850-7-4)

An excerpt of groups of logical nodes is contained in 5.4 of this part of IEC 61850.

A.1.3 Data classes (IEC 61850-7-4)

A total number of some 500 data classes are defined in IEC 61850-7-4. Table A.1 shows an excerpt of a few DATA classes with their names and semantic definitions.

The data names are composed using standardised abbreviations listed in a table at the beginning of IEC 61850-7-4 (some 260 abbreviations are defined), for example:

Term	Description
A	Current
Acs	Access
Acu	Acoustic
Age	Ageing
Alm	Alarm
Amp	Current non phase related
An	Analogue
Ang	Angle
…	…

These abbreviations should be used in the creation of new data names.

Table A.1 – Excerpt of data classes for measurands

Data name	Semantics
AcsCtlFail	Number of access control failures detected.
AcuPaDsch	Acoustic level of partial discharge in db.
AgeRat	Ageing rate, for example of transformer.
Alm	General single alarm.
AlmLstOv	TRUE = indication that the alarm list has overflowed.
AlmThm	Thermal alarm.
AlmVal	Alarm value is the pre-set value for a measurand that when reached will result in an alarm.
Amp	Current of a non-three-phase circuit.
Ang	Angle between phase voltage and current.
AngCor	Phase angle correction of a phasor (used for example for instrument transformers).
AngInd	This data indicates the check result of the differences between the angles of the busbar and line voltages. FALSE indicates that the angle difference is below the required limit. The angle difference criteria for the synchronising are fulfilled. TRUE indicates the angle difference exceeds the limit. The synchronising process shall be aborted because the angle criteria are not fulfilled (synchrocheck) or shall be continued with turbine control activities (synchronising).
...	...

The common data class to be used with a specific DATA item is defined in the logical nodes.

A.2 Common data class specifications (IEC 61850-7-3)

Table A.2 shows an excerpt of the list of the common data classes defined in IEC 61850-7-3. All common data classes are used by one or the other logical node.

Table A.2 – List of common data classes (excerpt)

Common data class specifications for status information
Single point status (SPS)
Double point status (DPS)
Integer status (INS)
Enumerated status (ENS)
Protection activation information (ACT)
Directional protection activation information (ACD)
Security violation counting (SEC)
Binary counter reading (BCR)
Histogram (HST)
…
Common data class specifications for measurand information
Measured value (MV)
Complex measured value (CMV)
Sampled value (SAV)
WYE
Delta (DEL)
Sequence (SEQ)
Harmonic value for WYE (HWYE)
…
Common data class specifications for controllable information
Controllable single point (SPC)
Controllable double point (DPC)
Controllable integer status (INC)
Controllable enumerated status (ENC)
Binary controlled step position information (BSC)
Integer controlled step position information (ISC)
…
Common data class specifications for status settings
Single point setting (SPG)
Integer status setting (ING)
Enumerated status setting (ENG)
Object reference setting group (ORG)
…
Common data class specifications for analogue settings
Analogue setting (ASG)
Setting curve (CURVE)
…
Common data class specifications for description information
Device name plate (DPL)
Logical node name plate (LPL)
…

Annex B
(informative)

Allocation of data to logical nodes

Figure B.1 illustrates an example of the assignment of data to logical nodes.

A data is assigned to that logical node that produces or consumes the values of this data object, this means for example:

– data consisting of instantaneous values of current and voltage are assigned to the logical nodes “current transformer” and “voltage transformer” respectively;
– data consisting of calculated values of current and voltage, for example root-mean-square r.m.s., are assigned to the logical node “measurement unit”;
– data consisting of voltage and step position are assigned to the logical node “tap changer controller”. Same holds for the tap change commands;
– data consisting of a (fault) impedance Z are assigned to the logical node “distance protection”. The same applies to the protection trip.

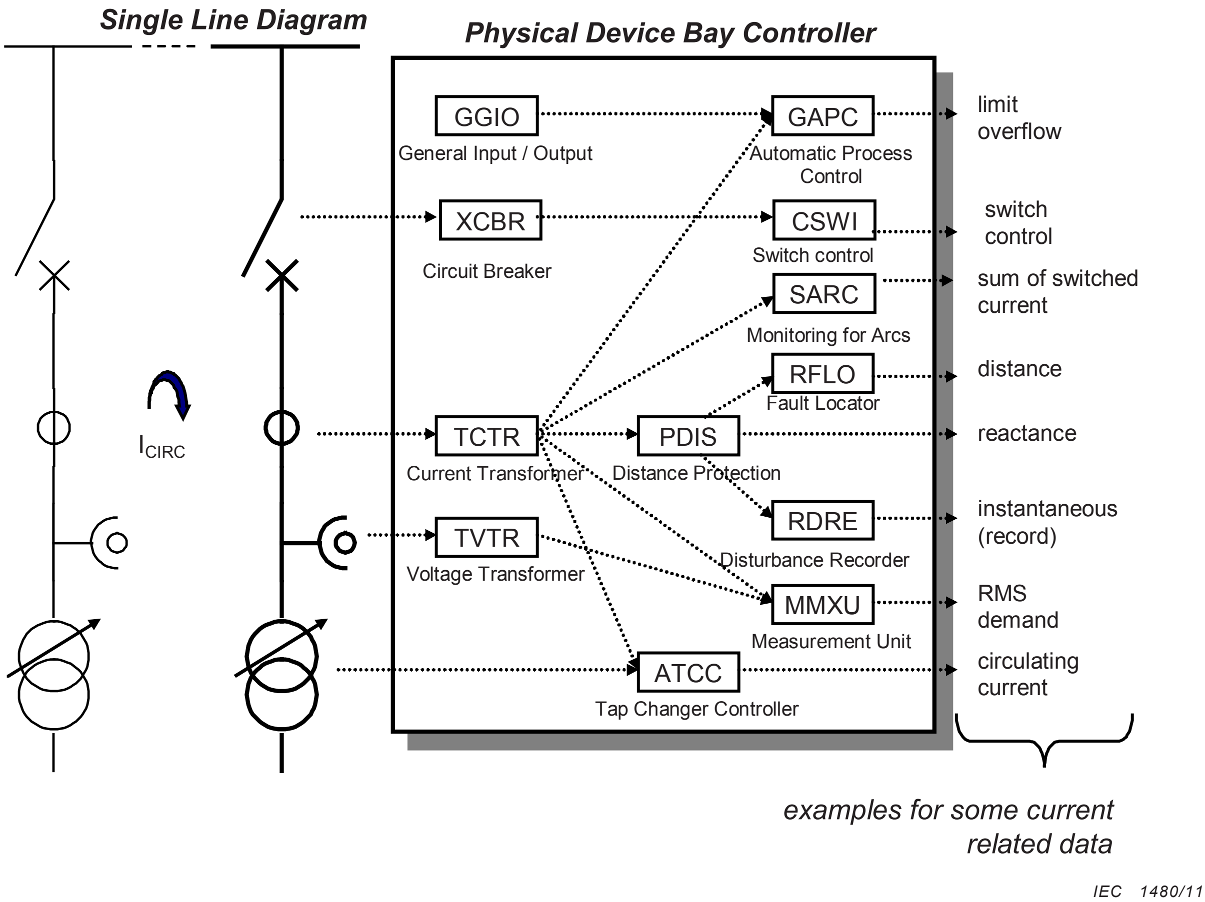


Figure B.1 – Example for control and protection LNs combined in one physical device

In any case where a compatible data has been defined in the standard for a specific application, this compatible data shall be used instead of defining a new one.

A second example application is shown in Figure B.2. The merging unit receives the current and voltage values directly from the instrument transformers. This unit may exist integrated per instrument transformer. These implementations are outside the scope of the standard. The sources of the samples are always instances of the LN classes TVTR and TCTR. In the example with the merging unit, the samples of all three phases and the neutral are collected and send out as multicast messages. Several applications receive the sampled values.

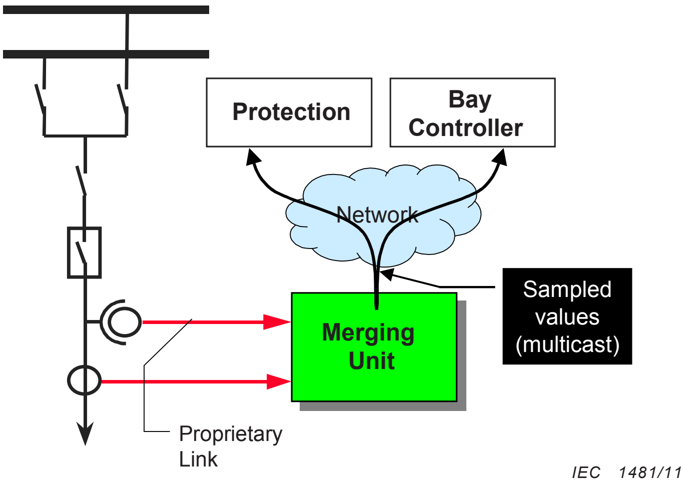


Figure B.2 – Merging unit and sampled value exchange (topology)

The merging unit is modelled as a single logical device named “MergingUnit” (see Figure B.3).

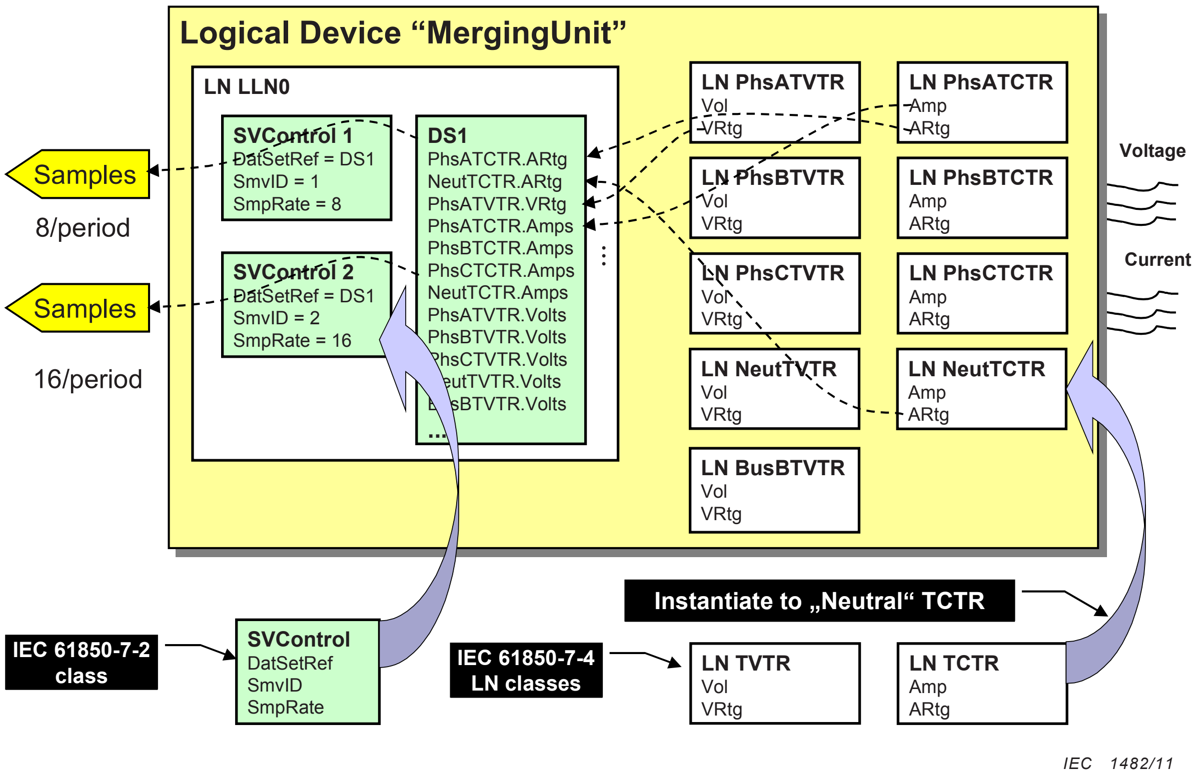


Figure B.3 – Merging unit and sampled value exchange (data)

The current and voltage values are received at the right-hand side. The three-phase values for current and voltage are modelled in the following logical nodes:

– Current transformer – TCTR class in IEC 61850-7-4 instantiated for three phases and neutral:

  PhsATCTR, PhsBTCTR, PhsCTCTR, NeutTCTR,

– Voltage transformer – TVTR class in IEC 61850-7-4 instantiated for three phases, neutral, and bus bar:

  PhsATVTR, PhsBTVTR, PhsCTVTR, NeutTVTR, BusBTVTR,

The sampled values (Amp and Vol) and the corresponding ratings are used in the example. These data are referenced by the data set "DS1".

Two instances of the sampled value control blocks (SVControl1 and SVControl2) are defined to control the exchange of the samples. The two control blocks just realise two different sample rates (8 and 16 samples per nominal period – 400/800 samples per second in the case of a 50 Hertz system).

Annex C
(informative)

Use of the substation configuration language (SCL)

C.1 General

This annex explains only the application of the SCL to define the use of optional definitions contained in class definitions in IEC 61850-7-4 and IEC 61850-7-3.

C.2 SCL and options in logical nodes

Figure C.1 shows the logical node class XCBR as it is defined in IEC 61850-7-4. There are several data items defined as being mandatory (M) other data are defined as being optional (O).

A logical node (XCBR) of a device model is specified with a SCL file. By definition, all mandatory data defined in the class defined in IEC 61850-7-4 are used by the logical node in the device model.

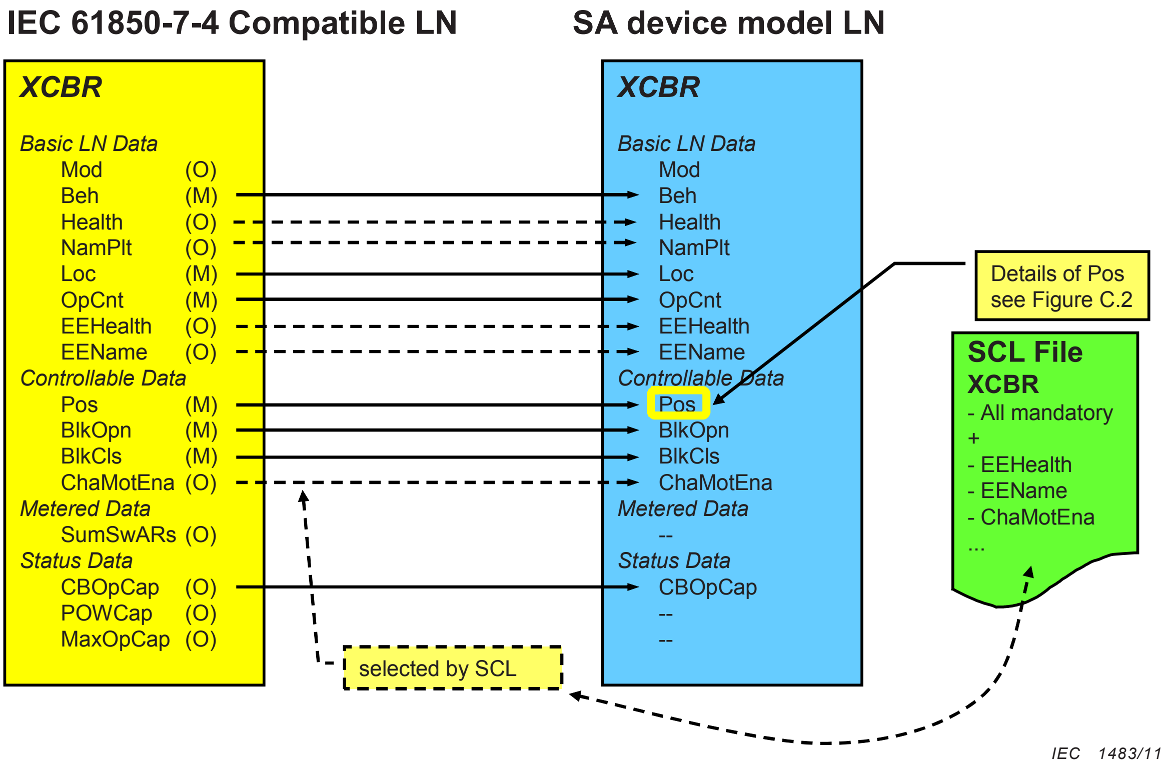


Figure C.1 – Application of SCL for LNs (conceptual)

The SCL needs to list all data to be used in the device model. Three optional data items are selected in the example (EEHealth, EEName, and ChaMotEna).

The SCL also needs to list the optional data attributes of each data selected. The marked data Pos is detailed in Clause C.3.

C.3 SCL and options in data

At the logical node level, the SCL shall list the names of the mandatory and optional data. The SCL for the data requires the list of mandatory and optional data attributes as well as the initialisation (configuration) values for several data attributes.

The SCL file in Figure C.2 shows which optional data attributes are selected. In addition, the SCL file assigns values to three data attributes.

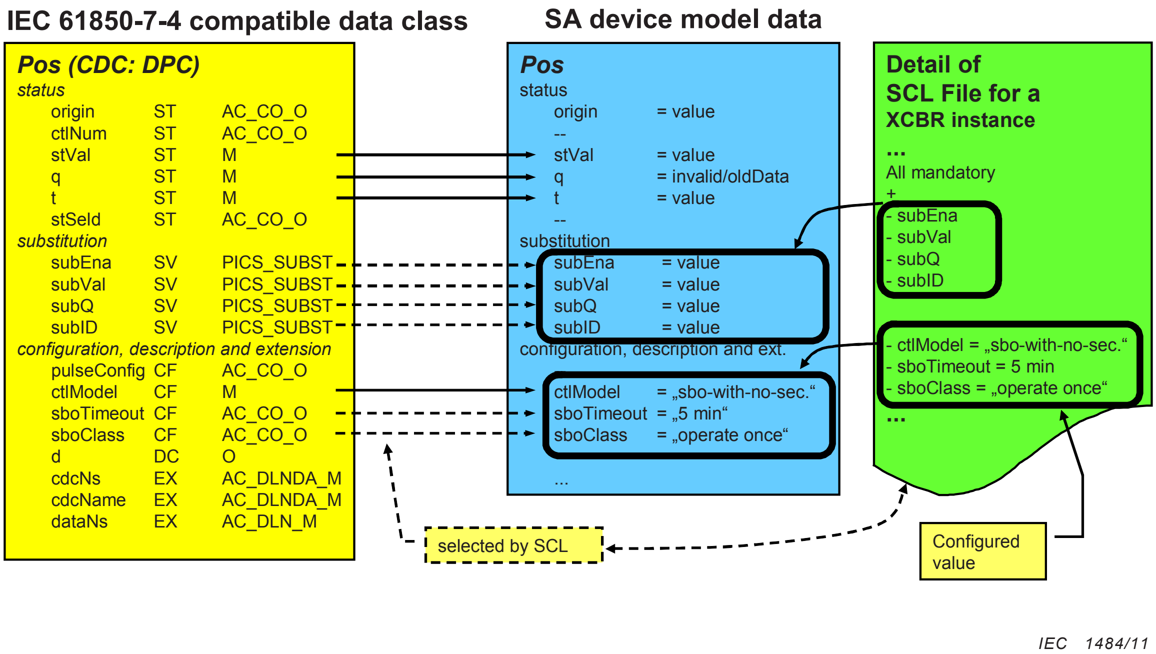


Figure C.2 – Application of SCL for data (conceptual)

The configured values for ctlModel, sboTimeout, and sboClass become effective as soon as the real device has been configured. The values may be overwritten (if the device allows overwriting these values at all) by a service request from a specific client.

Annex D
(informative)

Applying the LN concept to options for future extensions

D.1 General – Seamless telecontrol communication architecture

The seamless communication as depicted in Figure D.1 allows the modelling of the control center view (CC view) of a substation and the communication between substations and control centers.

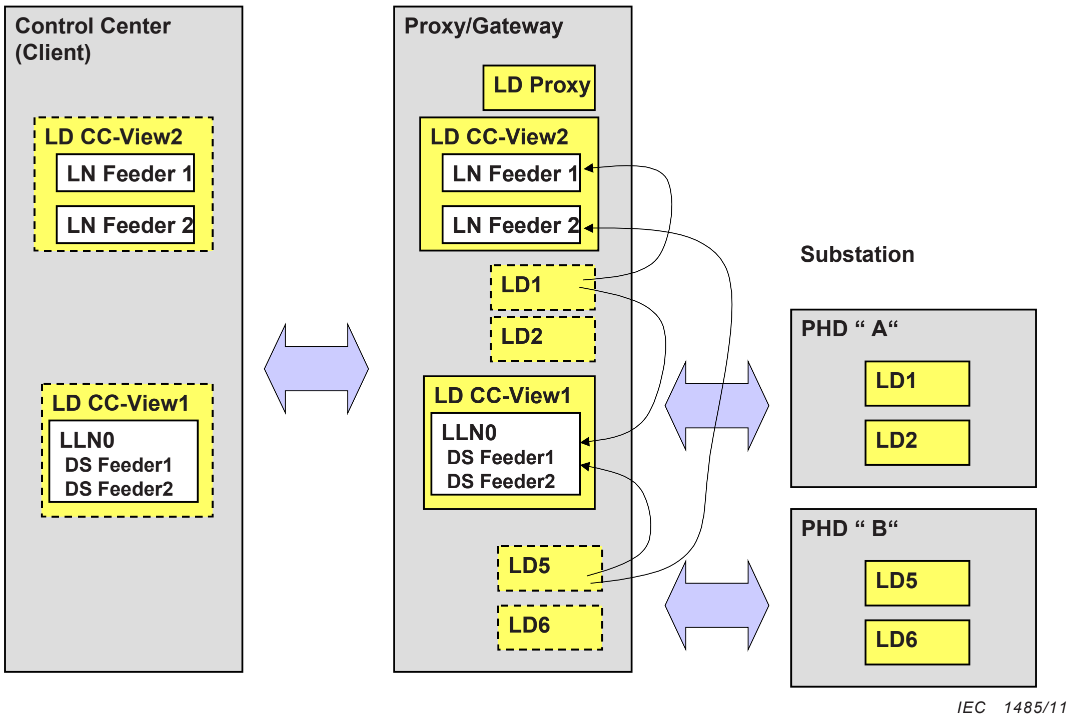


Figure D.1 – Seamless communication (simplified)

The control center can access the substation through a physical device serving as a gateway. This provides several options to access the data of the substation:

a) The gateway/proxy provides the capability for a direct access to the physical devices in the substation.

b) Within the gateway/proxy, a logical device (for example CC-View1) may define data sets that collect the information required in the control center.

c) As an alternative solution, new logical node classes and data classes may be defined to be used in the gateway/proxy that provide the control center view of the substation (in the example above, instantiated in the logical device CC-View2).

If new logical node classes and data classes are to be defined to reflect the control center view, this shall be harmonized with the CIM model in particular with regard to the names of the data classes.

This provides three options to access the data of the substation:

Option a) is most valuable for maintenance issues.

Option b) could be used for operational issues, but requires expensive engineering effort.

Option c) seems to be a promising solution for operational issues, since the same engineering concept as used within the substation could be applied for the whole control system of a utility. Therefore, this option was further investigated.

Combinations of options are possible. For example it may be suitable to use option c) for operational purposes and option a) for engineering and maintenance purposes.

Example for the requirement of a new logical node:

For the control center, the building blocks of the substation are bays (for example feeders or transformer bays). Therefore, a new logical node group "bay" might be defined with logical nodes for different bay types. These logical nodes with their data classes will provide the control center view of the substation.

An example is given in Figure D.2.

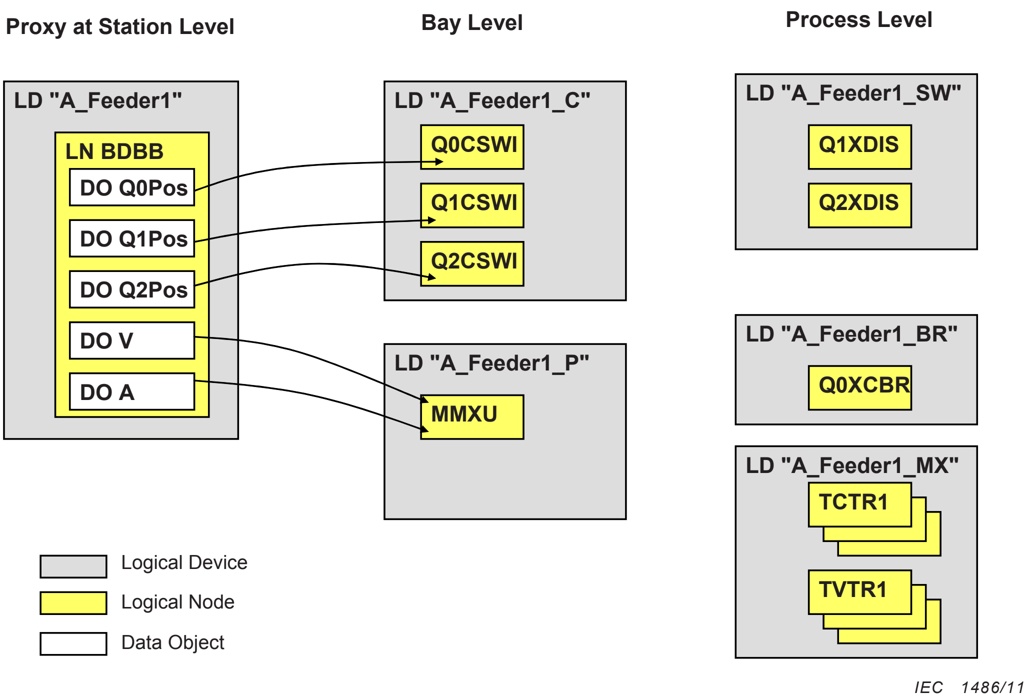


Figure D.2 – Example for new logical nodes

This supports a similar modelling approach for the control center as for the substation. As a consequence, the same engineering concepts and tools as well as the same communication software can be used.

EXAMPLE

Logical node of group "bay":

BDBB double busbar
BHCB one and a half CB

Logical devices defined, for example, per voltage level:

SSAtlanta_110
SSAtlanta_380

Some data objects:

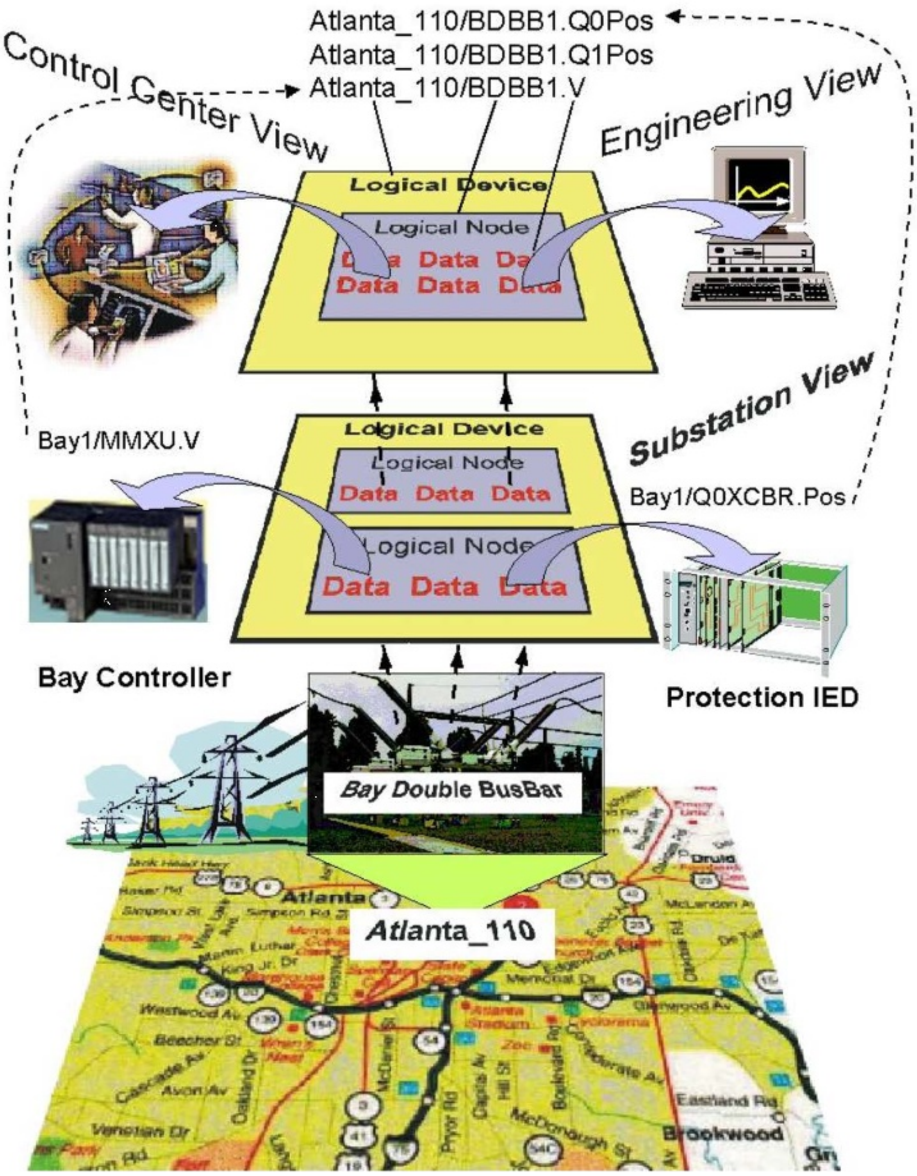
SSAtlanta_110/BDBB1.Q0Pos
SSAtlanta_110/BDBB1.Q1Pos
SSAtlanta_110/BDBB1.V
SSAtlanta_110/BDBB2.Q0Pos
SSAtlanta_110/BDBB2.Q1Pos
SSAtlanta_110/BDBB2.V
SSAtlanta_380/BHCB1.QAPos
SSAtlanta_380/BHCB1.QBPos
SSAtlanta_380/BHCB1.QCPos

Basically, the LN Bxxx is another virtual view on the same real world object. How the LN Bxxx receives the information is an implementation issue. It may for example subscribe to reports from LN Q0CSWI or it may directly forward a control command to Q0CSWI.

The data objects in the new LNs will be of the same common data classes (including important metadata) as the original data objects. However, in a first approach only the mandatory attributes may be supported for the seamless communication to the control center. By creating new logical device and logical node instances dedicated for a specific control center view, the names may be defined to meet the system operator's preferences. The name translation is done in the gateway/proxy of the substation.

The new LNs may also define new data classes such as summary alarms representing a logical combination of individual alarms.

An example is depicted in Figure D.3. The substation view can be mapped to one or more control center views. In the example above, the logical device SSAtlanta_110 could be the view of the control center A and the logical device SSAtlanta_380, the view of control center B.



IEC 1487/11

Figure D.3 – Example for control center view and mapping to substation view

D.2 Teleprotection

Teleprotection functions like line distance protection or line differential protection shall use IEC 61850 as the basis for the needed information exchange. For a description of the use of IEC 61850 for the communication between substations, the reader may refer to IEC/TR 61850-90-1.

The technical report provides a comprehensive overview on the different aspects that need to be considered while using IEC 61850 for information exchange between substations. In particular, the report:

– defines use cases that require an information exchange between substations;
– describes the communication requirements;
– gives guidelines for the communication services and communication architecture to be used;
– defines data as a prerequisite for interoperable applications;
– does not define implementations which guarantee interoperability between different IEDs;
– describes the usage and enhancements of the configuration language SCL.

Annex E
(informative)

Relation between logical nodes and PICOMs

IEC 61850-5 describes functions in a substation automation system that are divided into sub-functions called logical nodes. The content of the exchanged data between the LNs are (in IEC 61850-5) called PICOMs (pieces of communication), see Figure E.1. This view is independent of any model that is used to define the semantic and syntax of exchanged data – such as the client/server model.

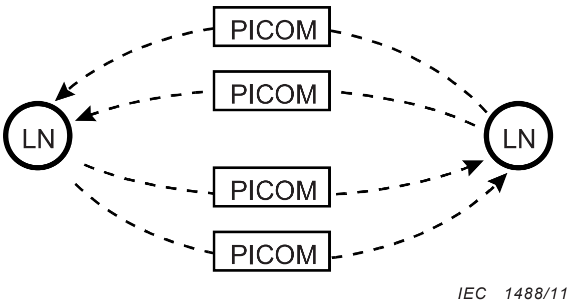


Figure E.1 – Exchanged data between subfunctions (logical nodes)

In the client/server model, there are defined services that determine the semantics and syntax of exchanged data. In this sense, the exchanged data is called protocol data unit (PDU) that determines the “bits on the wire”. The content and the semantics of the exchanged data is determined by the objects inside the server. In this model, the logical nodes are objects. Their subcomponents – the data objects – comprise all the process information that is related to the content of the PICOMs, see Figure E.2.

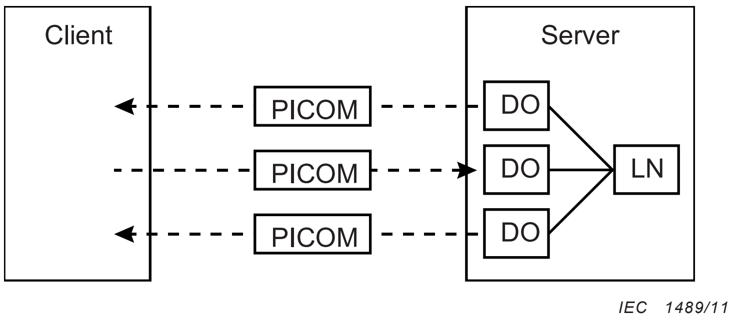


Figure E.2 – Relationship between PICOMS and client/server model

Since the logical nodes are objects in the client/server model, the subfunctions (LNs) inside a client are not of interest to describe the communication with the server.

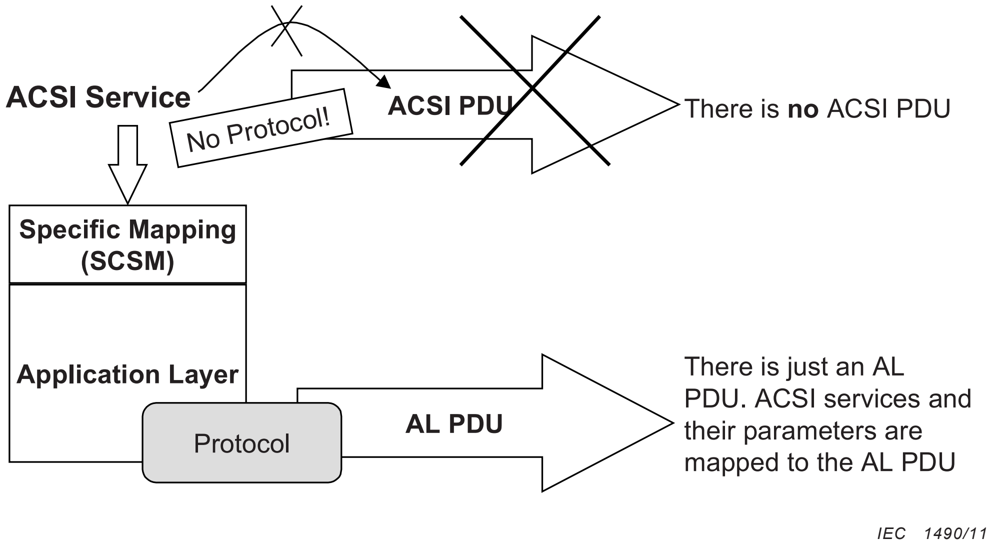
One PDU may comprise the content of several data – therefore the content of several PICOMs.

Annex F
(informative)

Mapping the ACSI to real communication systems

F.1 General

Figure F.1 depicts the relation of the ACSI to an underlying application layer. The ACSI does not define concrete ACSI messages. The ACSI services are mapped to a series of one or more application layer messages (AL PDU – protocol data unit) of the underlying application layer.



PDU: Protocol data unit (**encoded** message containing the service parameter, etc.).

Figure F.1 – ACSI mapping to an application layer

The mapping of ACSI services to specific application layer messages is beyond the scope of IEC 61850-7-2. This mapping is specified by a specific communication service mapping (SCSM) in the IEC 61850-8-x and the IEC 61850-9-x series.

NOTE 1 This mapping allows the ACSI to be applied to different application layers. As these application layers provide different features, the mapping within the SCSM may be simple or complex, and more or less efficient.

IEC 61850-7-4, IEC 61850-7-3, and IEC 61850-7-2 define abstract information and service models for the application domain substation. Even so, the IEC 61850 series in general allows discrete devices to share data and services. For this to occur, the devices shall agree on the concrete form of the services and data that will be exchanged.

The form of the service and data is of no consequence to the transport, network, and media protocols, i.e. to the lower layers of the communication stack and they are invariant to it. Conversely, the application that is sending and receiving data has no real procedure describing how this is achieved and it is therefore largely invariant of the mechanisms used.

This separation of roles is important as it allows many different technologies to be employed in a relatively transparent manner. As consequence, these lower layers may be exchanged. For example,

– networks with different types of physical media may be used;

– more than one application layer protocol may exist and use the same physical network and protocols.

Standardised mappings of the abstract services to different communication stacks are defined in the IEC 61850-8-x and the IEC 61850-9-x series, so that common utility functions will be performed consistently across all field devices independently of the underlying communication systems. Figure F.2 summarises the mappings defined in IEC 61850-8-1 and IEC 61850-9-2.

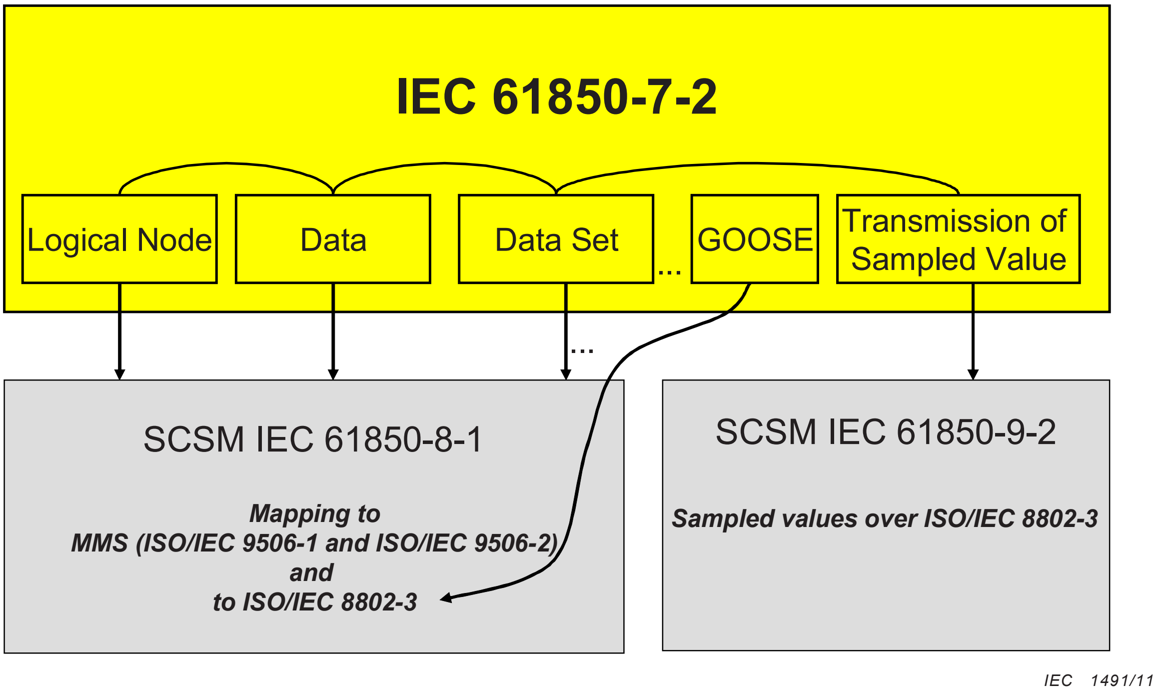


Figure F.2 – ACSI mappings (conceptual)

All but GOOSE and transmission of sampled values are mapped to MMS, TCP/IP, and ISO/IEC 8802-3. GOOSE is mapped directly to ISO/IEC 8802-3. The transmission of sampled values is mapped in IEC 61850-9-2.

The specific communication service mapping defines how the services and the models (server, logical devices, logical nodes, data, data sets, report controls, log controls, setting groups, etc.) are implemented using a specific communication stack, i.e. a complete profile. The mappings and the used application layer define the syntax (concrete encoding) for the data exchanged over the network.

NOTE 2 The concept of the SCSM has been introduced to be independent from communication stacks including application protocols. One objective of the IEC 61850 series is the interoperability of devices. This requires that all communicating devices use the same communication stack. Therefore, the goal of this independence is not to have many mappings in parallel, but to be able to follow the state of the art in communication technology.

According to Figure F.3, the SCSM maps the abstract communication services, objects and parameters to the specific application layers. These application layers provide the concrete coding. Depending on the technology of the communication network, these mappings may have different complexities, and some ACSI services may not be supported directly in all mappings, but equivalent services shall be provided (see example below). An application layer may use one or more stacks (layer 1 to 6).

EXAMPLE The ACSI service “GetDataSetValues” may have different mappings for different application layers (AL). For example, a specific AL may support this service directly while another AL provides “Get of single data” only. In the last case the mapping has to issue several “Get of single data”.

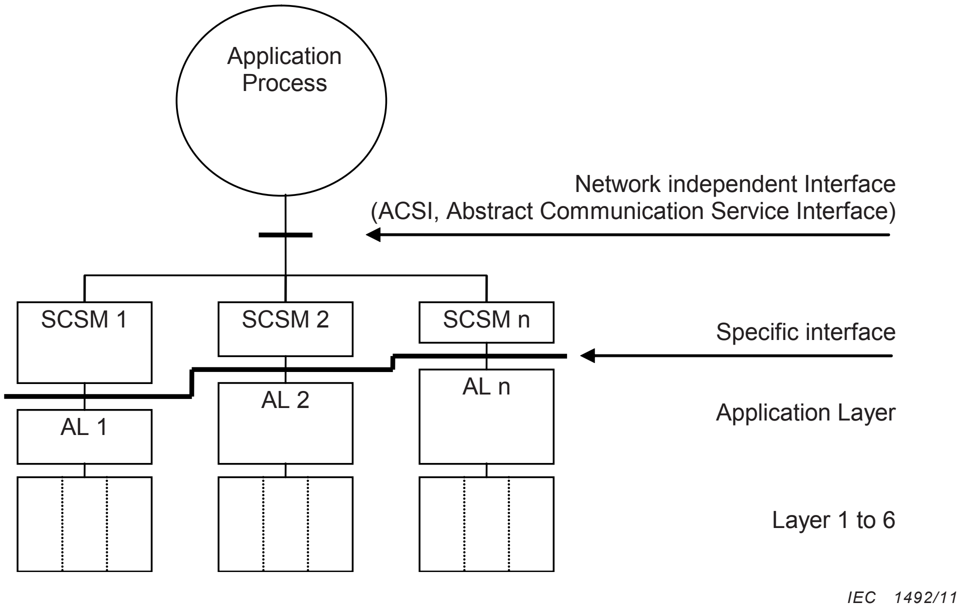


Figure F.3 – ACSI mapping to communication stacks/profiles

F.2 Mapping example (IEC 61850-8-1)

The information models (logical device, logical node, data, and data attributes) are defined in an abstract way in IEC 61850-7-4 and IEC 61850-7-3. In addition, the service models are defined as abstract services (ACSI – abstract communication service interface) defined in IEC 61850-7-2.

NOTE The names of logical nodes, data and data attributes are used as they are defined. The name XCBR is kept as the name XCBR. Mappings that do not support names in their protocols may map the names to unique numbers.

The abstract models of IEC 61850-7-4, IEC 61850-7-3 and IEC 61850-7-2 need to be mapped to an application layer (see Figure F.4).

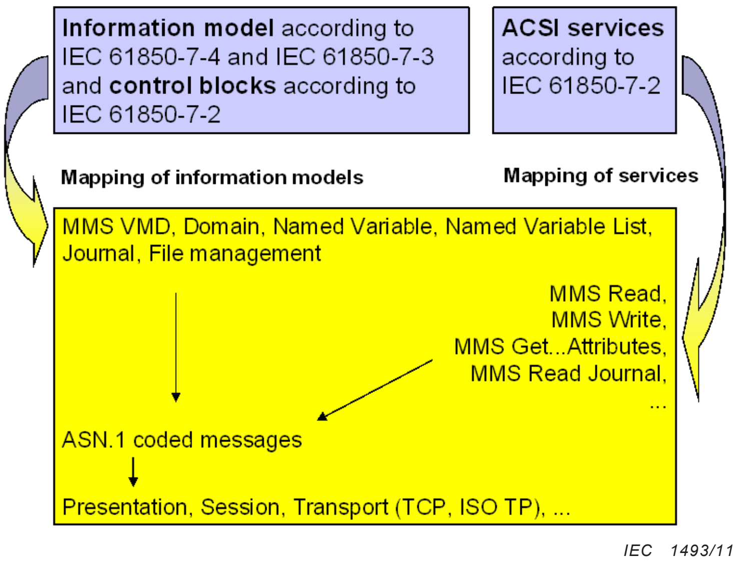


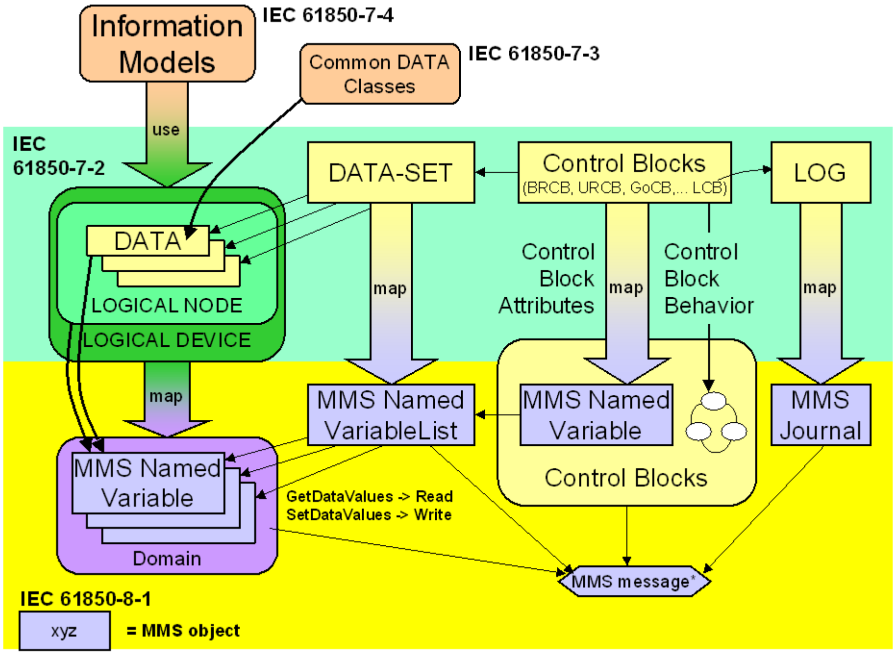
Figure F.4 – Mapping to MMS (conceptual)

The information model and the various control blocks are mapped in this example to the manufacturing message specification (MMS), i.e., to the virtual manufacturing device (VMD), domain, named variable, named variable list, journal, and file management. The services are mapped to the corresponding services of the MMS classes.

A more detailed mapping is shown in Figure F.5. The abstract information models defined in IEC 61850-7-4, IEC 61850-7-3 and IEC 61850-7-2 map as follows:

What to be mapped?	Maps to
Logical device (contains logical nodes); IEC 61850-7-2	MMS domain
Logical nodes (contains data); IEC 61850-7-4	MMS named variable
Data; IEC 61850-7-4	MMS named variable (and structured components of the named variable representing the “logical node data”)
Data attribute; IEC 61850-7-3	MMS named variable (and structured components of the named variable representing the “data”)
Data set: IEC 61850-7-2	MMS named variable list
Control blocks (attributes); IEC 61850-7-2	MMS named variable
Control blocks (behaviour); IEC 61850-7-2	Needs to be programmed as defined in IEC 61850-7-2
Log; IEC 61850-7-2	MMS journal

The messages carrying the information are mapped to MMS messages except for the GOOSE and SV messages.

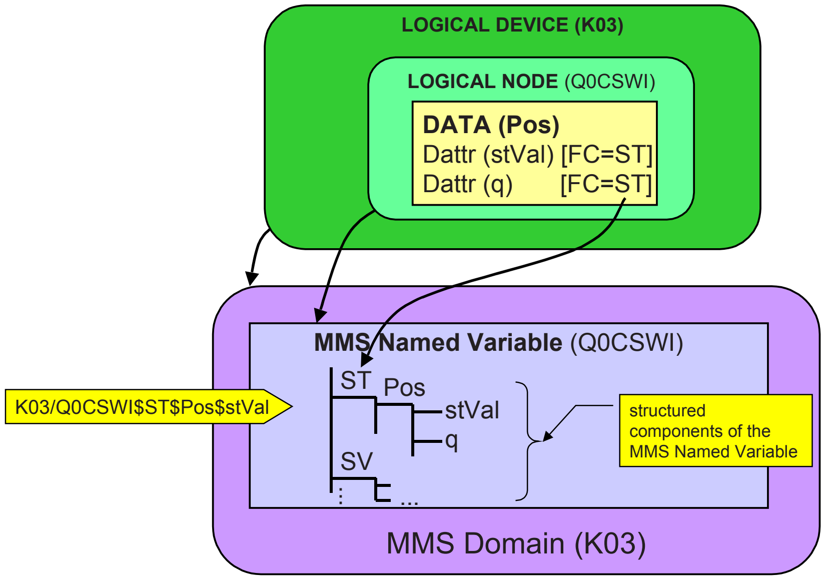


IEC 1494/11

* GOOSE/SMV messages map directly to ISO/IEC 8802-3.

Figure F.5 – Mapping approach

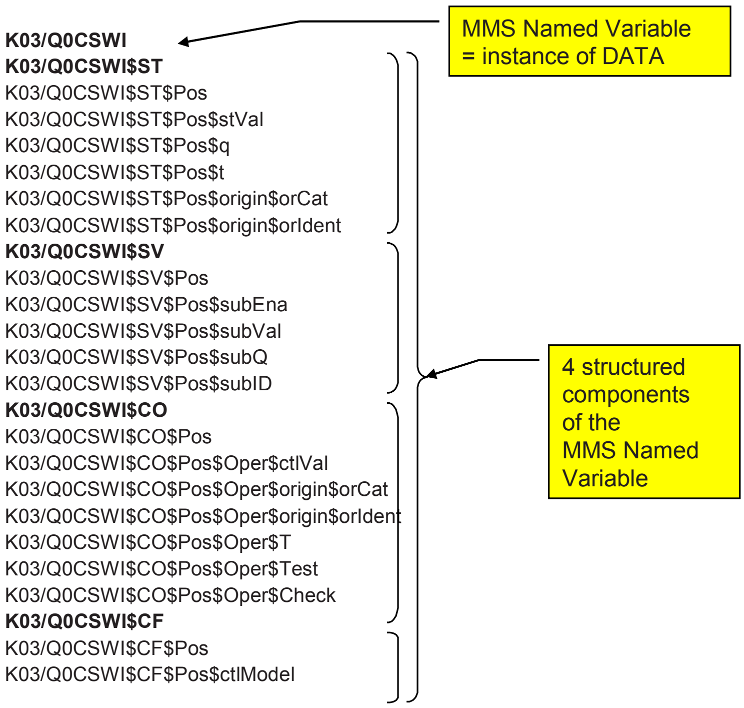
The details of the mapping to the MMS named variable are sketched in Figure F.6.



IEC 1495/11

Figure F.6 – Mapping detail of mapping to a MMS named variable

The MMS Domain (with the name K03) contains named variables. The named variable shown in Figure F.7 has the name "Q0CSWI". The components of this named variable are constructed by selecting all data attributes with the same functional constraint (FC), for example, the value FC=ST (all status data attributes). The first component of the named variable has the component name "ST". The DATA (for example, Pos) are placed at the next nesting level. The data attributes (for example, stVal, q, t, etc.) are at the next level below. The dots "." in the hierarchical name have been replaced by "$" in the MMS mapping.



IEC 1496/11

Figure F.7 – Example of MMS named variable (process values)

The process values (from the position) of several switch controllers Q0CSWI$Pos, Q1CSWI$Pos, and Q2CSWI$Pos are mapped to MMS named variables (see right lower corner of Figure F.8). The position information is grouped by the named variable list (data set) with the name K03/LLN0$AllRpts. The attributes of the unbuffered report control block are mapped to the MMS named variable K03/LLN0RPAllRpts.The components of this named variable can be written (configured). The control block references the data set.

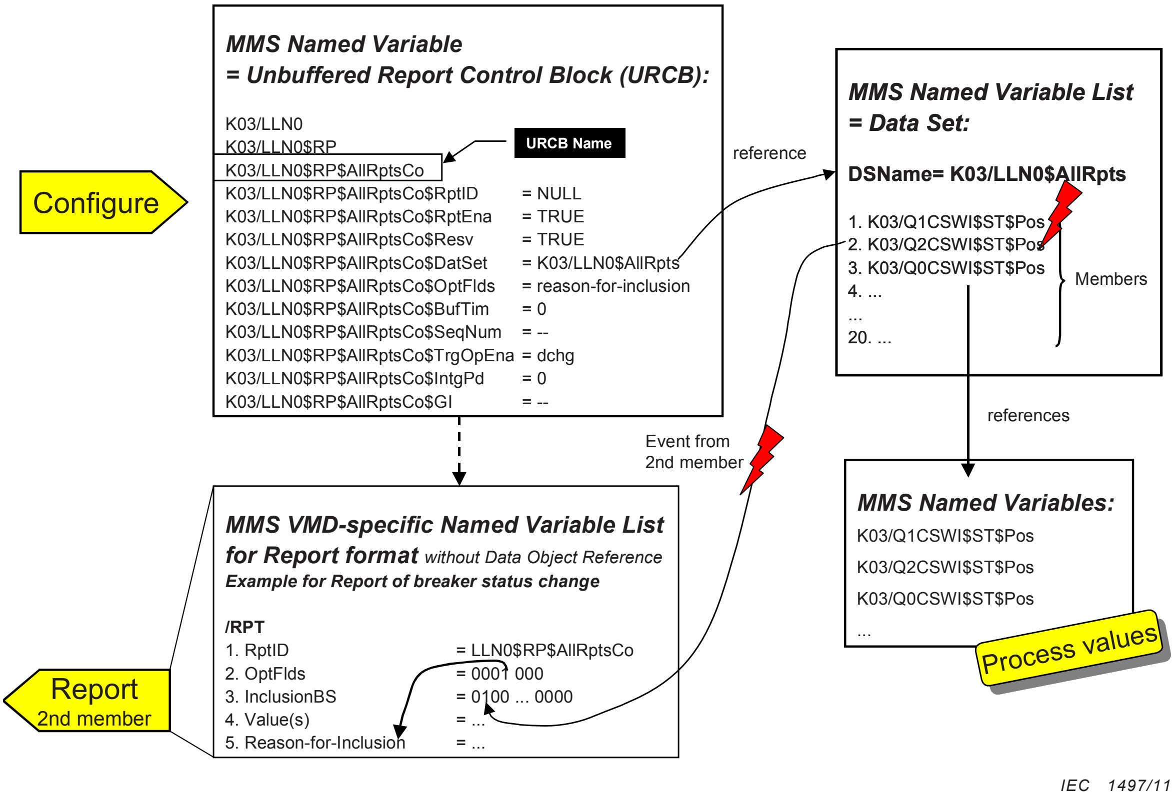


Figure F.8 – Use of MMS named variables and named variable list

A change in one of the members of the data set (for example, in member 2) issues sending a report with the status of the position of Q2CSWI. The report message is generated using another MMS named variable list (left lower corner). The report will be sent immediately.

The report is mapped to an MMS information report (see Figure F.9). The figure shows the concrete encoding according to ASN.1 BER (the basic encoding rule for abstract syntax notation number one – ISO 8825).

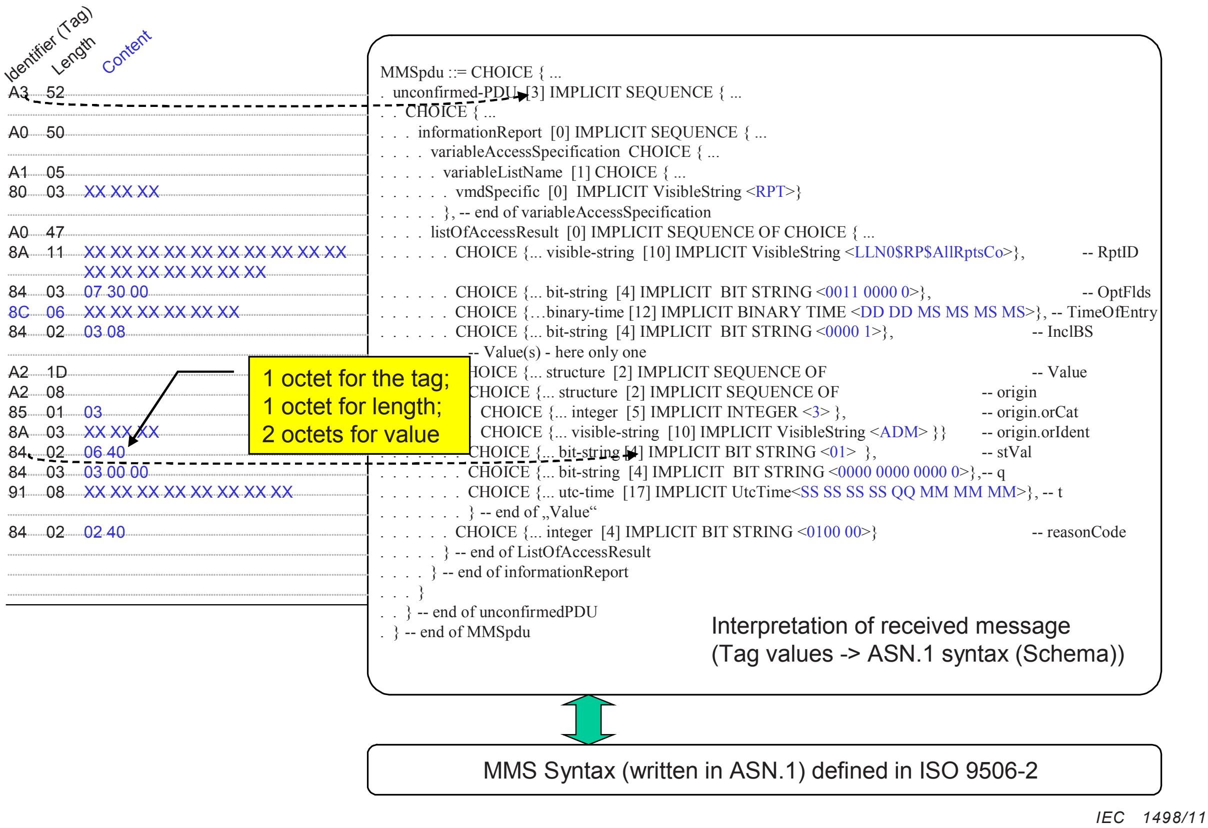


Figure F.9 – MMS information report message

These octets are packed into further messages that add lower layer-specific control and address information, for example, the TCP header and IP header.

The receiving IED is able to interpret the report message according to the identifier, lengths, names, and other values. The interpretation of the message requires the same stack, i.e., knowledge of all layers involved – including the definitions of IEC 61850-7-4, IEC 61850-7-3, IEC 61850-7-2, and IEC 61850-7-1.

NOTE 1 Implementations are expected to realise the layers in a way that hides the assembly, encoding, transmission, decoding, and interpretation of the messages. The application programs on both ends are expected to not be involved in these communication issues.

Figure F.10 illustrates a model excerpt of XCBR1 representing a real device. The complete hierarchical model may be mapped, for example, to MMS applying the SCSM according to IEC 61850-8-1. As a result, many MMS named variables have to be implemented in a real server. The services of the ACSI are mapped to MMS services.

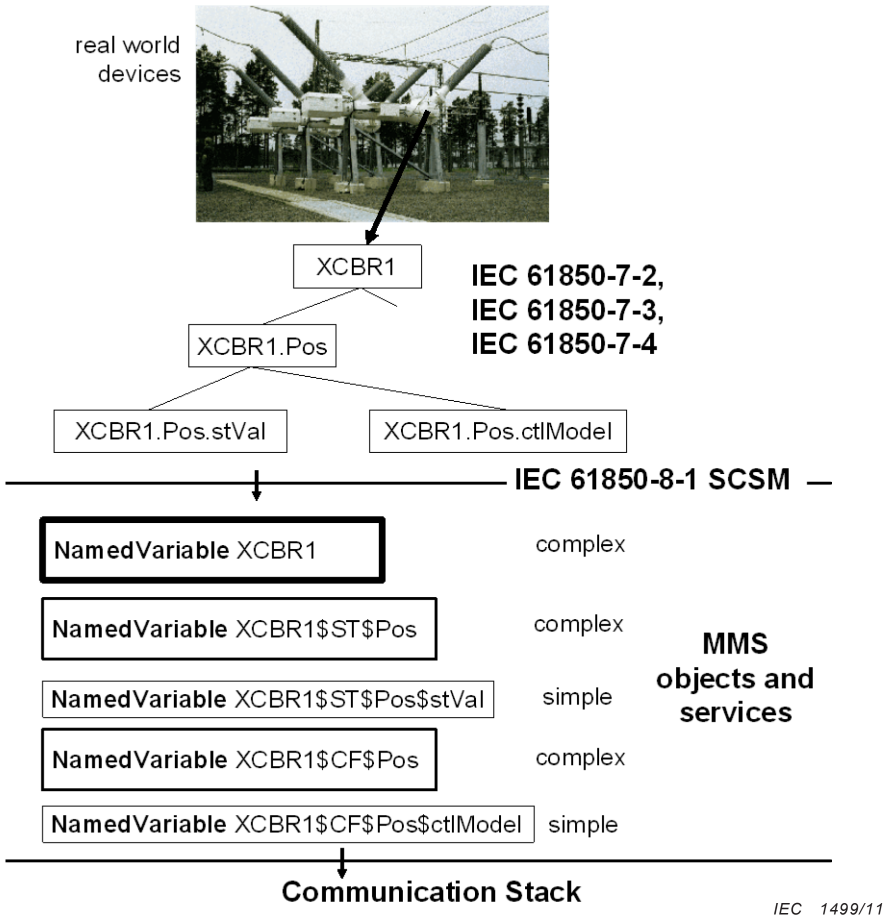


Figure F.10 – Mapping example

This example shows that the named variable XCBR1 represents the logical node (including all DATA as components of this named variable). Each component has been mapped to a less complex named variable, for example Pos (with components stVal and ctlModel). These components are mapped to two even less complex named variables: XCBR1STPos$stVal and XCBR1$CFPosctlModel.

NOTE 2 This multi-mapping does not require multiple storages of one value (for example, for ctlModel). The tree with all components and sub-(sub-)components is implemented once. The named variable XCBR1CFPos$ctlModel is just the "address" of the leaf in this tree.

MMS services support to read in one request many "full" or partial "trees". The partial tree is described in the request message with the MMS alternate access.

Bibliography

IEC 60050-101:1998, *International Electrotechnical Vocabulary – Part 1: Mathematics*

IEC 61346-1, *Industrial systems, installations and equipment and industrial products – Structuring principles and reference designations – Part 1: Basic rules* (withdrawn)

IEC 61346-2, *Industrial systems, installations and equipment and industrial products – Structuring principles and reference designations – Part 2: Classification of objects and codes for classes* (withdrawn)

IEC 61400 (all parts), *Wind turbines*

IEC/TR 61850-90-1, *Communication networks and systems for power utility automation – Part 90-1: Use of IEC 61850 for the communication between substations*

IEEE-SA TR 1550, 1999: Utility Communications Architecture (UCA®) Version 2.

This page deliberately left blank

This page deliberately left blank

British Standards Institution (BSI)

BSI is the independent national body responsible for preparing British Standards and other standards-related publications, information and services.
It presents the UK view on standards in Europe and at the international level.

It is incorporated by Royal Charter.

Revisions

British Standards are updated by amendment or revision. Users of British Standards should make sure that they possess the latest amendments or editions.

It is the constant aim of BSI to improve the quality of our products and services. We would be grateful if anyone finding an inaccuracy or ambiguity while using this British Standard would inform the Secretary of the technical committee responsible, the identity of which can be found on the inside front cover.

Tel: +44 (0)20 8996 9001 Fax: +44 (0)20 8996 7001

BSI offers Members an individual updating service called PLUS which ensures that subscribers automatically receive the latest editions of standards.

Tel: +44 (0)20 8996 7669 Fax: +44 (0)20 8996 7001
Email: plus@bsigroup.com

Buying standards

You may buy PDF and hard copy versions of standards directly using a credit card from the BSI Shop on the website **www.bsigroup.com/shop.** In addition all orders for BSI, international and foreign standards publications can be addressed to BSI Customer Services.

Tel: +44 (0)20 8996 9001 Fax: +44 (0)20 8996 7001
Email: orders@bsigroup.com

In response to orders for international standards, it is BSI policy to supply the BSI implementation of those that have been published as British Standards, unless otherwise requested.

Information on standards

BSI provides a wide range of information on national, European and international standards through its Knowledge Centre.

Tel: +44 (0)20 8996 7004 Fax: +44 (0)20 8996 7005
Email: knowledgecentre@bsigroup.com

Various BSI electronic information services are also available which give details on all its products and services.

Tel: +44 (0)20 8996 7111 Fax: +44 (0)20 8996 7048
Email: info@bsigroup.com

BSI Subscribing Members are kept up to date with standards developments and receive substantial discounts on the purchase price of standards. For details of these and other benefits contact Membership Administration.

Tel: +44 (0)20 8996 7002 Fax: +44 (0)20 8996 7001
Email: membership@bsigroup.com

Information regarding online access to British Standards via British Standards Online can be found at **www.bsigroup.com/BSOL**

Further information about BSI is available on the BSI website at **www.bsigroup.com/standards**

Copyright

Copyright subsists in all BSI publications. BSI also holds the copyright, in the UK, of the publications of the international standardization bodies. Except as permitted under the Copyright, Designs and Patents Act 1988 no extract may be reproduced, stored in a retrieval system or transmitted in any form or by any means – electronic, photocopying, recording or otherwise – without prior written permission from BSI. This does not preclude the free use, in the course of implementing the standard of necessary details such as symbols, and size, type or grade designations. If these details are to be used for any other purpose than implementation then the prior written permission of BSI must be obtained. Details and advice can be obtained from the Copyright & Licensing Manager.

Tel: +44 (0)20 8996 7070
Email: copyright@bsigroup.com

BSI Group Headquarters

389 Chiswick High Road London W4 4AL UK

Tel +44 (0)20 8996 9001
Fax +44 (0)20 8996 7001
www.bsigroup.com/standards

raising standards worldwide™

This page deliberately left blank

British Standards Institution (BSI)

BSI is the independent national body responsible for preparing British Standards and other standards-related publications, information and services.
It presents the UK view on standards in Europe and at the international level.

It is incorporated by Royal Charter.

Revisions

British Standards are updated by amendment or revision. Users of British Standards should make sure that they possess the latest amendments or editions.

It is the constant aim of BSI to improve the quality of our products and services. We would be grateful if anyone finding an inaccuracy or ambiguity while using this British Standard would inform the Secretary of the technical committee responsible, the identity of which can be found on the inside front cover.

Tel: +44 (0)20 8996 9001 Fax: +44 (0)20 8996 7001

BSI offers Members an individual updating service called PLUS which ensures that subscribers automatically receive the latest editions of standards.

Tel: +44 (0)20 8996 7669 Fax: +44 (0)20 8996 7001
Email: plus@bsigroup.com

Buying standards

You may buy PDF and hard copy versions of standards directly using a credit card from the BSI Shop on the website **www.bsigroup.com/shop.** In addition all orders for BSI, international and foreign standards publications can be addressed to BSI Customer Services.

Tel: +44 (0)20 8996 9001 Fax: +44 (0)20 8996 7001
Email: orders@bsigroup.com

In response to orders for international standards, it is BSI policy to supply the BSI implementation of those that have been published as British Standards, unless otherwise requested.

Information on standards

BSI provides a wide range of information on national, European and international standards through its Knowledge Centre.

Tel: +44 (0)20 8996 7004 Fax: +44 (0)20 8996 7005
Email: knowledgecentre@bsigroup.com

Various BSI electronic information services are also available which give details on all its products and services.

Tel: +44 (0)20 8996 7111 Fax: +44 (0)20 8996 7048
Email: info@bsigroup.com

BSI Subscribing Members are kept up to date with standards developments and receive substantial discounts on the purchase price of standards. For details of these and other benefits contact Membership Administration.

Tel: +44 (0)20 8996 7002 Fax: +44 (0)20 8996 7001
Email: membership@bsigroup.com

Information regarding online access to British Standards via British Standards Online can be found at **www.bsigroup.com/BSOL**

Further information about BSI is available on the BSI website at **www.bsigroup.com/standards**

Copyright

Copyright subsists in all BSI publications. BSI also holds the copyright, in the UK, of the publications of the international standardization bodies. Except as permitted under the Copyright, Designs and Patents Act 1988 no extract may be reproduced, stored in a retrieval system or transmitted in any form or by any means – electronic, photocopying, recording or otherwise – without prior written permission from BSI. This does not preclude the free use, in the course of implementing the standard of necessary details such as symbols, and size, type or grade designations. If these details are to be used for any other purpose than implementation then the prior written permission of BSI must be obtained. Details and advice can be obtained from the Copyright & Licensing Manager.

Tel: +44 (0)20 8996 7070
Email: copyright@bsigroup.com

BSI Group Headquarters

389 Chiswick High Road London W4 4AL UK

Tel +44 (0)20 8996 9001
Fax +44 (0)20 8996 7001
www.bsigroup.com/standards

raising standards worldwide™