

DIN EN 61850-7-3

ICS 33.040.40

Ersatz für
DIN EN 61850-7-3:2004-02
Siehe Anwendungsbeginn

**Kommunikationsnetze und -systeme für die Automatisierung in der
elektrischen Energieversorgung –
Teil 7-3: Grundlegende Kommunikationsstruktur –
Gemeinsame Datenklassen (IEC 61850-7-3:2010);
Englische Fassung EN 61850-7-3:2011**

Communication networks and systems for power utility automation –
Part 7-3: Basic communication structure –
Common data classes (IEC 61850-7-3:2010);
English version EN 61850-7-3:2011

Réseaux et systèmes de communication pour l'automatisation des systèmes électriques –
Partie 7-3: Structure de communication de base –
Classes de données communes (CEI 61850-7-3:2010);
Version anglaise EN 61850-7-3:2011

Gesamtumfang 85 Seiten

DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik im DIN und VDE

Anwendungsbeginn

Anwendungsbeginn für die von CENELEC am 2011-01-20 angenommene Europäische Norm als DIN-Norm ist 2011-08-01.

Daneben darf DIN EN 61850-7-3:2004-02 noch bis 2013-12-07 angewendet werden.

Nationales Vorwort

Vorausgegangener Norm-Entwurf: E DIN IEC 61850-7-3/A1:2006-09.

Für diese Norm ist das nationale Arbeitsgremium K 952 „Netzleittechnik“ der DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik im DIN und VDE (www.dke.de) zuständig.

Die enthaltene IEC-Publikation wurde vom TC 57 „Power systems management and associated information exchange“ erarbeitet.

Das IEC-Komitee hat entschieden, dass der Inhalt dieser Publikation bis zu dem Datum (stability date) unverändert bleiben soll, das auf der IEC-Website unter „<http://webstore.iec.ch>“ zu dieser Publikation angegeben ist. Zu diesem Zeitpunkt wird entsprechend der Entscheidung des Komitees die Publikation

- bestätigt,
- zurückgezogen,
- durch eine Folgeausgabe ersetzt oder
- geändert.

Für den Fall einer undatierten Verweisung im normativen Text (Verweisung auf eine Norm ohne Angabe des Ausgabedatums und ohne Hinweis auf eine Abschnichtsnummer, eine Tabelle, ein Bild usw.) bezieht sich die Verweisung auf die jeweils neueste gültige Ausgabe der in Bezug genommenen Norm.

Für den Fall einer datierten Verweisung im normativen Text bezieht sich die Verweisung immer auf die in Bezug genommene Ausgabe der Norm.

Der Zusammenhang der zitierten Normen mit den entsprechenden Deutschen Normen ergibt sich, soweit ein Zusammenhang besteht, grundsätzlich über die Nummer der entsprechenden IEC-Publikation. Beispiel: IEC 60068 ist als EN 60068 als Europäische Norm durch CENELEC übernommen und als DIN EN 60068 ins Deutsche Normenwerk aufgenommen.

Das Präsidium des DIN hat mit Präsidialbeschluss 1/2004 festgelegt, dass DIN-Normen, deren Inhalt sich auf internationale Arbeitsergebnisse der Informationsverarbeitung gründet, unter bestimmten Bedingungen allein in englischer Sprache veröffentlicht werden dürfen. Diese Bedingungen sind für die vorliegende Norm erfüllt.

Da sich die Benutzer der vorliegenden Norm der englischen Sprache als Fachsprache bedienen, wird die Englische Fassung der IEC 61850-7-3 veröffentlicht. Für die meisten der verwendeten Begriffe existieren keine gebräuchlichen deutschen Benennungen, da sich die deutschen Anwender in der Regel ebenfalls der englischen Benennungen bedienen. Diese Norm steht nicht in unmittelbarem Zusammenhang mit Rechtsvorschriften und ist nicht als Sicherheitsnorm anzusehen.

Änderungen

Gegenüber DIN EN 61850-7-3:2004-02 wurden folgende Änderungen vorgenommen:

- a) Definition neuer allgemeiner Datenklassen für neue Standards zur Definition von Objekt-Modellen für Anwendungsbereiche, die auf DIN EN 61850 basieren. Sie werden auch für die Darstellung von statistischen und historischen Daten benutzt.
- b) Klarstellungen und Berichtigungen.

Frühere Ausgaben

DIN EN 61850-7-3: 2004-02

English version

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

Réseaux et systèmes de communication
pour l'automatisation des
systèmes électriques -
Partie 7-3: Structure de
communication de base -
Classes de données communes
(CEI 61850-7-3:2010)

Kommunikationsnetze und -systeme für
die Automatisierung in
der elektrischen Energieversorgung -
Teil 7-3: Grundlegende
Kommunikationsstruktur -
Gemeinsame Datenklassen
(IEC 61850-7-3:2010)

This European Standard was approved by CENELEC on 2011-01-20. 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 Central Secretariat 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 Central Secretariat 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

Foreword

The text of document 57/1087/FDIS, future edition 2 of IEC 61850-7-3, prepared by IEC TC 57, Power systems management and associated information exchange, was submitted to the IEC-CENELEC parallel vote and was approved by CENELEC as EN 61850-7-3 on 2010-12-07.

This European Standard supersedes EN 61850-7-3:2003.

Compared to EN 61850-7-3:2003, this edition:

- defines new common data classes used for new standards defining object models for other domains based on EN 61850 and for the representation of statistical and historical data;
- provides clarifications and corrections to EN 61850-7-3:2003.

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

The following dates were fixed:

- | | | |
|--|-------|------------|
| – latest date by which the EN has to be implemented at national level by publication of an identical national standard or by endorsement | (dop) | 2011-09-07 |
| – latest date by which the national standards conflicting with the EN have to be withdrawn | (dow) | 2013-12-07 |

Annex ZA has been added by CENELEC.

Endorsement notice

The text of the International Standard IEC 61850-7-3:2010 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 61850-8 series | NOTE | Harmonized in EN 61850-8 series (not modified). |
| IEC 61850-9 series | NOTE | Harmonized EN 61850-9 series (not modified). |
-

CONTENTS

INTRODUCTION.....	7
1 Scope.....	8
2 Normative references	8
3 Terms and definitions	9
4 Abbreviated terms	9
5 Conditions for attribute inclusion	9
6 Constructed attribute classes	10
6.1 General.....	10
6.2 Quality	10
6.2.1 Overview	10
6.2.2 Validity	11
6.2.3 Detail quality	12
6.2.4 Source.....	13
6.2.5 Test.....	13
6.2.6 Frozen by operator	13
6.2.7 Quality in the client server context.....	14
6.2.8 Relation between quality identifiers	15
6.3 Analogue value	16
6.4 Configuration of analogue value	17
6.5 Range configuration	18
6.6 Step position with transient indication.....	18
6.7 Pulse configuration.....	19
6.8 Originator	19
6.9 Unit definition	20
6.10 Vector definition	20
6.11 Point definition	21
6.12 CtlModels definition.....	21
6.13 SboClasses definition.....	21
6.14 Cell	21
6.15 CalendarTime definition.....	22
7 Common data class specifications	23
7.1 General.....	23
7.2 Name spaces	23
7.3 Common data class specifications for status information	24
7.3.1 Application of services.....	24
7.3.2 Single point status (SPS).....	25
7.3.3 Double point status (DPS)	25
7.3.4 Integer status (INS)	26
7.3.5 Enumerated status (ENS)	26
7.3.6 Protection activation information (ACT)	27
7.3.7 Directional protection activation information (ACD)	28
7.3.8 Security violation counting (SEC)	28
7.3.9 Binary counter reading (BCR).....	29
7.3.10 Histogram (HST).....	29
7.3.11 Visible string status (VSS).....	30

7.4	Common data class specifications for measurand information	30
7.4.1	Application of services.....	30
7.4.2	Measured value (MV)	31
7.4.3	Complex measured value (CMV)	32
7.4.4	Sampled value (SAV)	33
7.4.5	Phase to ground/neutral related measured values of a three-phase system (WYE)	33
7.4.6	Phase to phase related measured values of a three-phase system (DEL) ..	34
7.4.7	Sequence (SEQ)	34
7.4.8	Harmonic value (HMF).....	35
7.4.9	Harmonic value for WYE (HWYE)	36
7.4.10	Harmonic value for DEL (HDEL)	36
7.5	Common data class specifications for controls	37
7.5.1	Application of services.....	37
7.5.2	Controllable single point (SPC).....	38
7.5.3	Controllable double point (DPC)	39
7.5.4	Controllable integer status (INC).....	40
7.5.5	Controllable enumerated status (ENC).....	40
7.5.6	Binary controlled step position information (BSC)	41
7.5.7	Integer controlled step position information (ISC)	42
7.5.8	Controllable analogue process value (APC).....	43
7.5.9	Binary controlled analog process value (BAC)	44
7.6	Common data class specifications for status settings	45
7.6.1	Application of services.....	45
7.6.2	Single point setting (SPG)	45
7.6.3	Integer status setting (ING)	46
7.6.4	Enumerated status setting (ENG)	46
7.6.5	Object reference setting (ORG)	47
7.6.6	Time setting group (TSG)	47
7.6.7	Currency setting group (CUG)	48
7.6.8	Visible string setting (VSG).....	48
7.7	Common data class specifications for analogue settings	49
7.7.1	Application of services.....	49
7.7.2	Analogue setting (ASG)	49
7.7.3	Setting curve (CURVE).....	50
7.7.4	Curve shape setting (CSG).....	51
7.8	Common data class specifications for description information	52
7.8.1	Application of services.....	52
7.8.2	Device name plate (DPL).....	53
7.8.3	Logical node name plate (LPL)	53
7.8.4	Curve shape description (CSD)	54
8	Data attribute semantic.....	55
	Annex A (normative) Value range for units and multiplier	70
	Annex B (informative) Functional constraints	73
	Annex C (normative) Tracking of configuration revisions	75
	Annex D (normative) SCL enumerations	76
	Bibliography.....	82

Annex ZA (normative) Normative references to international publications with their corresponding European publications	83
Figure 1 – Quality identifiers in a single client-server relationship	14
Figure 2 – Quality identifiers in a multiple client-server relationship	14
Figure 3 – Interaction of substitution and validity	16
Figure 4 – Configuration of command output pulse	19
Figure 5 – Cell definition	22
Figure 6 – Two-dimensional curve represented by CSG	52
Figure 7 – Two-dimensional shape created by multiple CSG	52
Table 1 – Conditions for presence of attributes	9
Table 2 – Quality	11
Table 3 – Relation of the detailed quality identifiers with invalid or questionable quality	12
Table 4 – Analogue value	17
Table 5 – Configuration of analogue value	17
Table 6 – Range configuration	18
Table 7 – Step position with transient indication	18
Table 8 – Pulse configuration	19
Table 9 – Originator	19
Table 10 – Values for orCat	20
Table 11 – Unit	20
Table 12 – Vector	20
Table 13 – Point	21
Table 14 – Cell	21
Table 15 – CalendarTime	22
Table 16 – Semantic interpretation of calendar time settings	23
Table 17 – Name space attributes	24
Table 18 – Basic status information template	24
Table 19 – Single point status common data class definition	25
Table 20 – Double point status common data class specification	25
Table 21 – Integer status common data class specification	26
Table 22 – Enumerated status common data class specification	26
Table 23 – Protection activation information common data class specification	27
Table 24 – Directional protection activation information common data class specification	28
Table 25 – Security violation counting common data class specification	28
Table 26 – Binary counter reading common data class specification	29
Table 27 – Histogram common data class specification	29
Table 28 – Visible string status common data class definition	30
Table 29 – Basic measurand information template	31
Table 30 – Measured value	31
Table 31 – Complex measured value	32
Table 32 – Sampled value	33
Table 33 – WYE	33

Table 34 – Delta	34
Table 35 – Sequence	35
Table 36 – Harmonic value	35
Table 37 – Harmonic values for WYE.....	36
Table 38 – Harmonic values for delta.....	37
Table 39 – Basic controllable status information template	37
Table 40 – Controllable single point.....	38
Table 41 – Controllable double point.....	39
Table 42 – Controllable integer status.....	40
Table 43 – Controllable enumerated status	41
Table 44 – Binary controlled step position information	41
Table 45 – Integer controlled step position information	42
Table 46 – Controllable analogue process value	43
Table 47 – Binary controlled analog process value	44
Table 48 – Basic status setting template	45
Table 49 – Single point setting.....	45
Table 50 – Integer status setting	46
Table 51 – Enumerated status setting	46
Table 52 – Object reference setting common data class specification	47
Table 53 – Time setting group common data class specification	47
Table 54 – Currency setting group common data class specification	48
Table 55 – Visible string setting group common data class specification	48
Table 56 – Basic analogue setting template	49
Table 57 – Analogue setting	50
Table 58 – Setting curve	50
Table 59 – Curve shape setting	51
Table 60 – Basic description information template	52
Table 61 – Device name plate common data class specification.....	53
Table 62 – Logical node name plate common data class specification	54
Table 63 – Curve shape description common data class specification.....	54
Table 64 – Semantics of data attributes and data	55
Table A.1 – SI units: base units	70
Table A.2 – SI units: derived units	70
Table A.3 – SI units: extended units.....	71
Table A.4 – SI units: industry specific units.....	71
Table A.5 – Multiplier	72
Table B.1 – Functional constraints	73

INTRODUCTION

This document is part of a set of specifications, which details layered substation communication architecture. This architecture has been chosen to provide abstract definitions of classes and services such that the specifications are independent of specific protocol stacks and objects. The mapping of these abstract classes and services to communication stacks is outside the scope of IEC 61850-7-x and may be found in IEC 61850-8-x (station bus) and IEC 61850-9-x (process bus).

IEC 61850-7-1 gives an overview of this communication architecture. This part of IEC 61850 defines constructed attributed classes and common data classes related to applications in the power system using IEC 61850 modeling concepts like substations, hydro power or distributed energy resources. These common data classes are used in IEC 61850-7-4 to define compatible dataObject classes. The SubDataObjects, DataAttributes or SubAttributes of the instances of dataObject are accessed using services defined in IEC 61850-7-2.

This part of IEC 61850 is used to specify the abstract common data class and constructed attribute class definitions. These abstract definitions are mapped into concrete object definitions that are to be used for a particular protocol (for example MMS, ISO 9506 series).

Note that there are common data classes used for service tracking, that are defined in IEC 61850-7-2.

COMMUNICATION NETWORKS AND SYSTEMS FOR POWER UTILITY AUTOMATION –

Part 7-3: Basic communication structure – Common data classes

1 Scope

This part of IEC 61850 specifies constructed attribute classes and common data classes related to substation applications. In particular, it specifies:

- common data classes for status information,
- common data classes for measured information,
- common data classes for control,
- common data classes for status settings,
- common data classes for analogue settings and
- attribute types used in these common data classes.

This International Standard is applicable to the description of device models and functions of substations and feeder equipment.

This International Standard may also be applied, for example, to describe device models and functions for:

- substation to substation information exchange,
- substation to control centre information exchange,
- power plant to control centre information exchange,
- information exchange for distributed generation, or
- information exchange for metering.

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/TS 61850-2, *Communication networks and systems in substations – Part 2: Glossary*

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

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

IEEE C37.118:2005, *IEEE Standard for Synchrophasors for Power Systems*

ISO 4217, *Codes for the representation of currencies and funds*

¹ To be published.

3 Terms and definitions

For the purposes of this document, the terms and definitions given in IEC/TS 61850-2 and IEC 61850-7-2 apply.

4 Abbreviated terms

CDC	common data class
dchg	trigger option for data-change
dupd	trigger option for data-update
FC	functional constraint
qchg	trigger option for quality-change
TrgOp	trigger option

NOTE Abbreviations used for the identification of the common data classes and as names of the attributes are specified in the specific clauses of this document and are not repeated here.

5 Conditions for attribute inclusion

This clause lists general conditions that specify the presence of an attribute. Table 1 gives the conditions for presence of attributes.

Table 1 – Conditions for presence of attributes

Abbreviation	Condition
M	Attribute is mandatory. Attribute shall exist on any CDC type instance.
O	Attribute is optional. Attribute may or may not exist on any CDC type instance.
PICS_SUBST	Attribute is mandatory, if substitution is supported (for substitution, see IEC 61850-7-2), otherwise forbidden.
GC_1	At least one of the attributes shall be present for a given instance of DataObject / SubDataObject.
GC_2_n	All or none of the data attributes belonging to the same group (n) shall be present for a given instance of DataObject / SubDataObject.
GC_1_EXCL	At most one of the data objects shall be present for a given instance.
GC_CON_attr	A configuration data attribute shall only be present, if the (optional) specific data attribute (attr) to which this configuration relates is also present.
GC_2_XOR_n	All or none of a group (n) shall be present. Groups are exclusive, but one group shall be present.
AC_LN0_M	The attribute shall be present if the DataObject NamPIt belongs to LLN0; otherwise it may be optional.
AC_LN0_EX	The attribute shall be present only if the DataObject NamPIt belongs to LLN0 (applies to IdNs in CDC LPL only).
AC_DLD_M	The attribute shall be present, if LN name space of this LN deviates from the LN name space referenced by IdNs of the logical device in which this LN is contained (applies to InNs in CDC LPL only).
AC_DLN_M	The attribute shall be present, if the data name space of this data deviates from the data name space referenced by either InNs of the logical node in which the data is contained or, if there is no InNs, IdNs of the logical device in which the data is contained (applies to dataNs in all CDCs only).
AC_DLND_A_M	The attribute shall be present, if CDC name space of this data deviates from the CDC name space referenced by either the dataNs of the data, the InNs of the logical node in which the data is defined or IdNs of the logical device in which the data is contained (applies to cdcNs and cdcName in all CDCs only).

Abbreviation	Condition
AC_SCAV	The presence of the configuration data attribute depends on the presence of <i>i</i> and <i>f</i> of the Analog Value of the data attribute to which this configuration attribute relates. For a given data object, that attribute 1) shall be present, if both <i>i</i> and <i>f</i> are present, 2) shall be optional if only <i>i</i> is present, and 3) is not required if only <i>f</i> is present. NOTE If only <i>i</i> is present in a device without floating point capabilities, the configuration parameter may be exchanged offline.
AC_ST	The attribute is mandatory, if the controllable status class supports status information.
AC_CO_O	If the controllable status class supports control, this attribute is available and an optional attribute.
AC_CO_SBO	If the controllable status class supports control and if the control model supports the values "sbo-with-normal-security" or "sbo-with-enhanced-security" or both, that attribute shall be mandatory.
AC_SG_M	The attribute is mandatory, if this data shall be member of a setting group.
AC_SG_O	The attribute is optional, if this data shall be member of a setting group.
AC_SG_C1	One of the attributes is mandatory, if this data shall be member of a setting group.
AC_NS_G_M	The attribute is mandatory, if this data shall be a setting outside a setting group.
AC_NS_G_O	The attribute is optional, if this data shall be a setting outside a setting group.
AC_NS_G_C1	One of the attributes is mandatory, if this data shall be a setting outside a setting group.
AC_RMS_M	The attribute is mandatory when the harmonics reference type is rms.
AC_CLC_O	The attribute shall be optional, when the calculation type (according to data ClcMth) for this LN is Peak fundamental or RMS fundamental. The attribute shall not be available, if ClcMth is TRUE RMS.

6 Constructed attribute classes

6.1 General

Constructed attribute classes are defined for the use in common data classes (CDC) in Clause 7.

IEC 61850-7-1 provides an overview of all IEC 61850-7 documents (IEC 61850-7-2, IEC 61850-7-3, and IEC 61850-7-4). IEC 61850-7-1 also describes the basic notation used in IEC 61850-7-3 and the description of the relations between the IEC 61850-7 documents.

NOTE The common ACSI type "TimeStamp" is specified in IEC 61850-7-2.

6.2 Quality

6.2.1 Overview

Quality type shall be as defined in Table 2.

Table 2 – Quality

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	DEFAULT FALSE	M
outOfRange	BOOLEAN	DEFAULT FALSE	M
badReference	BOOLEAN	DEFAULT FALSE	M
oscillatory	BOOLEAN	DEFAULT FALSE	M
failure	BOOLEAN	DEFAULT FALSE	M
oldData	BOOLEAN	DEFAULT FALSE	M
inconsistent	BOOLEAN	DEFAULT FALSE	M
inaccurate	BOOLEAN	DEFAULT FALSE	M
source	CODED ENUM	process substituted DEFAULT process	M
test	BOOLEAN	DEFAULT FALSE	M
operatorBlocked	BOOLEAN	DEFAULT FALSE	M

The DEFAULT value shall be applied, if the functionality of the related attribute is not supported. The mapping may specify to exclude the attribute from the message, if it is not supported or if the DEFAULT value applies.

Quality shall be an attribute that contains information on the quality of the information from the server. Quality of the data is also related to the mode of a logical node. Further details can be found in IEC 61850-7-4. The different quality identifiers are not independent. Basically, there are the following quality identifiers:

- validity;
- detail quality;
- source;
- test;
- frozen by operator.

6.2.2 Validity

Validity shall be good, questionable or invalid.

good: The value shall be marked good if no abnormal condition of the acquisition function or the information source is detected.

invalid: The value shall be marked invalid when an abnormal condition of the acquisition function or the information source (missing or non-operating updating devices) is detected. The value shall not be defined under this condition. The mark invalid shall be used to indicate to the client that the value may be incorrect and shall not be used.

EXAMPLE If an input unit detects an oscillation of one input, it will mark the related information as invalid.

questionable: The value shall be marked questionable if a supervision function detects an abnormal behaviour, however the value could still be valid. The client shall be responsible for determining whether or not values marked "questionable" should be used.

6.2.3 Detail quality

The reason for an invalid or questionable value of an attribute may be specified in more detail with further quality identifiers. If one of these identifiers is set then validity shall be set to invalid or questionable. Table 3 shows the relation of the detailed quality identifiers with invalid or questionable quality.

Table 3 – Relation of the detailed quality identifiers with invalid or questionable quality

detailQual	invalid	questionable
overflow	X	
outOfRange	X	X
badReference	X	X
oscillatory	X	X
failure	X	
OldData		X
inconsistent		X
inaccurate		X

overflow: this identifier shall indicate a quality issue that the value of the attribute to which the quality has been associated is beyond the capability of being represented properly (used for measurand information only).

EXAMPLE A measured value may exceed the range that may be represented by the selected data type, for example the data type is a 16-bit unsigned integer and the value exceeds 65 535.

outOfRange: this identifier shall indicate a quality issue that the attribute to which the quality has been associated is beyond a predefined range of values. The server shall decide if validity shall be set to invalid or questionable (used for measurand information only).

EXAMPLE A measured value may exceed a predefined range, however the selected data type can still represent the value, for example the data type is a 16-bit unsigned integer, the predefined range is 0 to 40 000, if the value is between 40 001 and 65 535 it is considered to be out of range.

badReference: this identifier shall indicate that the value may not be a correct value due to a reference being out of calibration. The server shall decide if validity shall be set to invalid or questionable (used for measurand information and binary counter information only).

oscillatory: to prevent overloading of event driven communication channels, it is desirable to detect and suppress oscillating (fast changing) binary inputs. If a signal changes in a defined time (t_{osc}) twice in the same direction (from 0 to 1 or from 1 to 0) then it shall be defined as an oscillation and the detail quality identifier "oscillatory" shall be set. If a configured number of transient changes is detected, they shall be suppressed. In this time, the validity status "questionable" shall be set. If the signal is still in the oscillating state after the defined number of changes, the value shall be left in the state it was in when the oscillatory flag was set. In this case, the validity status "questionable" shall be reset and "invalid" shall be set as long as the signal is oscillating. If the configuration is such that all transient changes should be suppressed, the validity status "invalid" shall be set immediately in addition to the detail quality identifier "oscillatory" (used for status information only).

failure: this identifier shall indicate that a supervision function has detected an internal or external failure.

oldData: a value shall be oldData if an update is not made during a specific time interval. The value may be an old value that may have changed in the meantime. This specific time interval may be defined by an allowed-age attribute.

NOTE "Fail silent" errors, where the equipment stops sending data, will cause an oldData condition. In this case, the last received information was correct.

inconsistent: this identifier shall indicate that an evaluation function has detected an inconsistency.

inaccurate: this identifier shall indicate that the value does not meet the stated accuracy of the source.

EXAMPLE The measured value of power factor may be noisy (inaccurate) when the current is very small.

6.2.4 Source

Source shall give information related to the origin of a value. The value may be acquired from the process or be a substituted value.

process: the value is provided by an input function from the process I/O or is calculated from some application function.

substituted: the value is provided by input of an operator or by an automatic source.

NOTE 1 Substitution may be done locally or via the communication services. In the second case, specific attributes with a FC SV are used.

NOTE 2 There are various means to clear a substitution. As an example, a substitution that was done following an invalid condition may be cleared automatically if the invalid condition is cleared. However, this is a local issue and therefore not within the scope of this standard.

6.2.5 Test

Test shall be an additional identifier that may be used to classify a value being a test value and not to be used for operational purposes. The processing of the test quality in the client shall be as described in IEC 61850-7-4. The bit shall be completely independent from the other bits within the quality descriptor.

6.2.6 Frozen by operator

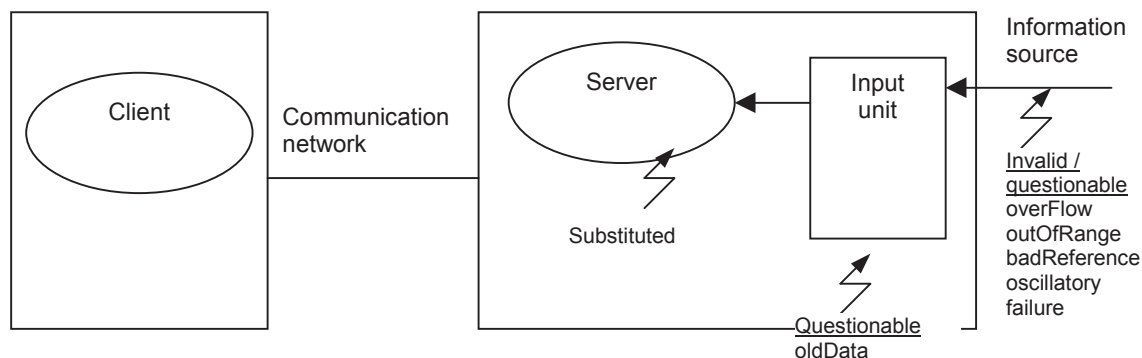
operatorBlocked: this identifier shall be set if further update of the value has been blocked by an operator. The value shall be the information that was acquired before blocking. If this identifier is set, then the identifier oldData of detailQual shall also be set.

The operator shall use the data attribute blkEna to block the update of the value.

NOTE Both an operator as well as an automatic function may freeze communication updating as well as input updating. In both cases, detailQual.oldData will be set. If the blocking is done by an operator, then the identifier operatorBlocked is set additionally. In that case, an operator activity is required to clear the condition.

EXAMPLE An operator may freeze the update of an input, to save the old value before the auxiliary supply is switched off.

6.2.7 Quality in the client server context

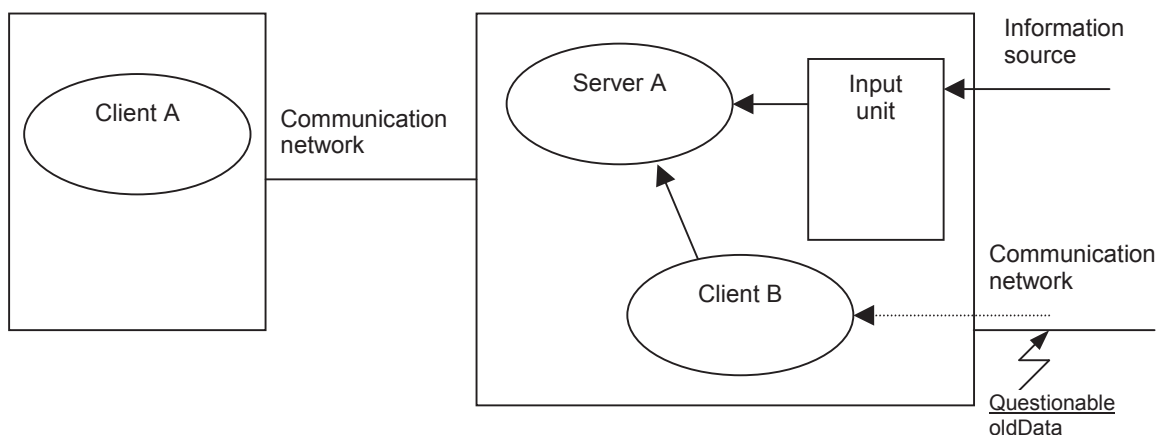


IEC 808/03

Figure 1 – Quality identifiers in a single client-server relationship

The quality identifier shall reflect the quality of the information in the server, as it is supplied to the client. Figure 1 shows potential sources that may influence the quality in a single client-server relationship. "Information source" is the (hardwired) connection of the process information to the system. The information may be invalid or questionable as indicated in Figure 1. Further abnormal behaviour of the information source may be detected by the input unit. In that case, the input unit may keep the old data and flag it accordingly.

In a multiple client-server relationship, as shown in Figure 2, information may be acquired over a communication link (with client B). If that communication link is broken, client B will detect that error situation and qualify the information as questionable/old data.



IEC 809/03

Figure 2 – Quality identifiers in a multiple client-server relationship

In the multiple client-server relationship, the quality of the data received from server A shall reflect both the quality of the server B (acquired with client B) as well as its own quality. Therefore, handling of prioritisation of quality from different levels may require further specification beyond that included in this standard. For the identifier validity, the value invalid shall dominate over the value questionable, since this is the worst case. For the identifier source, the higher level of the multiple client-server relationship shall dominate over the lower level.

EXAMPLE Let A be the higher level and B the lower level. The quality from server B is invalid. If now the communication fails (questionable, oldData) between server B and client B, the quality will remain invalid and not become questionable, since the last information was not correct. Server A therefore will report the information as invalid.

6.2.8 Relation between quality identifiers

Validity and source have a prioritized relation. If source is in the “process” state, then validity shall determine the quality of the origin value. If source is in the “substitute” state, then validity shall be overruled by the definition of the substituted value. This is an important feature, since substitution is used to replace invalid values with substituted values that may be used by the client such as good values.

EXAMPLE 1 If both questionable and substituted are set, this means that the substituted value is questionable. This may happen if, in a hierarchical configuration, a substitution is performed at the lowest level and the communication fails on a higher level.

EXAMPLE 2 If an invalid value is substituted, the invalid field will be cleared and the substituted field will be set to indicate the substitution.

The quality identifier operatorBlocked is independent of the other quality identifiers.

EXAMPLE 3 An oscillating input may cause the invalid field to be set. Due to the continuing changes in the value, many reports are generated, loading the communication network. An operator may block the update of the input. In this case, the field operatorBlocked will also be set.

An example for the interaction between the quality identifiers and the impact of multiple client-server relation is shown in Figure 3. In this example, it is assumed that a bay level device acts as a client of the process level server and as a server to the station level client.

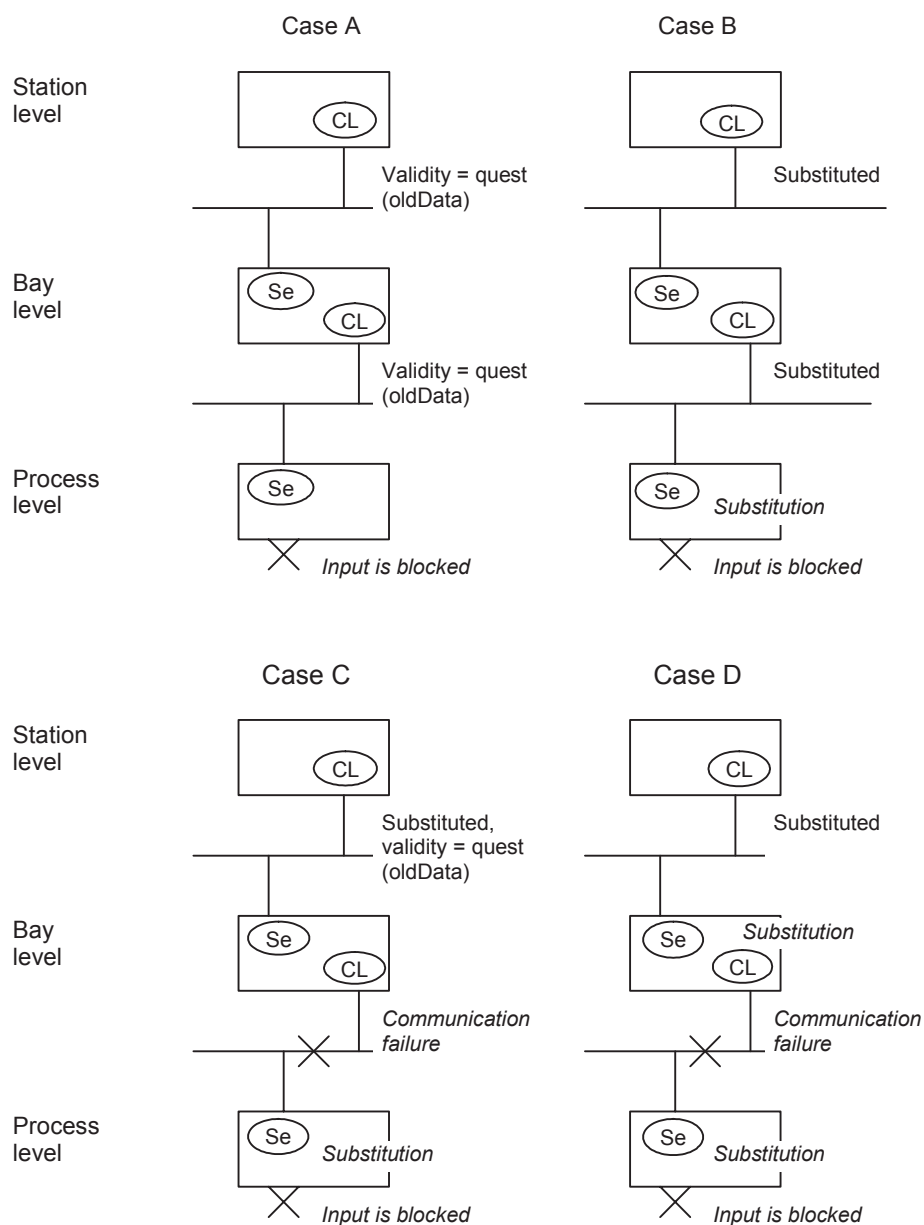
NOTE This is one example of a multiple client-server relationship; other multiple client-server relationships may exist, but the behaviour will not change.

In case A, the input is blocked, the quality of the information is marked as questionable and oldData.

In case B, a substitution is done at process level. Now, the quality of the information to the next higher level (the bay level) is marked as substituted (but good).

In case C, the communication between process and bay level fails. Between bay level and station level, the information is still marked as substituted. In addition, questionable and oldData is set to indicate that the (substituted) information may be old.

In case D, a new substitution is made at bay level. Now the quality of the information to the next higher level is marked as substituted (and good) and is independent from the first substitution.



IEC 2550/10

Key

CL client
Se server

Figure 3 – Interaction of substitution and validity

6.3 Analogue value

Analogue value type shall be as defined in Table 4.

Table 4 – Analogue value

AnalogueValue type definition			
Attribute name	Attribute type	Value/Value range	M/O/C
<i>i</i>	INT32	integer value	GC_1
<i>f</i>	FLOAT32	floating point value	GC_1

Analogue values may be represented as a basic type INTEGER (attribute *i*) or as FLOATING POINT (attribute *f*). At least one of the attributes shall be used. If both *i* and *f* exist, the application in the server shall insure that both values remain consistent. The latest value set by the communication service shall be used to update the other value. As an example, if xxx.*f* is written, the application shall update xxx.*i* accordingly.

The measured values represent primary process values.

i: The value of *i* shall be an integer representation of the measured value. The formula to convert between *i* and the process value (*pVal*) shall be:

$$pVal = (i \times scaleFactor) + offset$$

It shall be true within acceptable error when *i*, *scaleFactor*, *offset* and *f* are all present.

f: The value of *f* shall be the floating point representation of the measured value. The formula to convert between *f* and the process value shall be:

$$pVal = f \times 10^{units.multiplier}$$

NOTE The reason for both integer and floating point representation is so that IEDs without FLOATING POINT capabilities are enabled to support analogue values. In this case, the *scaleFactor* and *offset* may be exchanged offline between clients and servers.

6.4 Configuration of analogue value

Configuration of analogue value type shall be as defined in Table 5.

Table 5 – Configuration of analogue value

ScaledValueConfig type definition			
Attribute name	Attribute type	Value/value range	M/O/C
<i>scaleFactor</i>	FLOAT32		M
<i>offset</i>	FLOAT32		M

This constructed attribute class shall be used to configure the INTEGER value representation of the analogue value. The formula for conversion between integer and floating point value is given in 6.3.

scaleFactor: the value of *scaleFactor* shall be the scaling factor.

offset: the value of *offset* shall be the offset.

NOTE If a server does not support transmission of FLOAT32 values, the client may retrieve these values from the SCL file.

6.5 Range configuration

Range configuration type is used to configure the limits that define the range of a measured value and shall be as defined in Table 6.

Table 6 – Range configuration

RangeConfig type definition			
Attribute name	Attribute type	Value/Value range	M/O/C
hhLim	AnalogueValue		M
hLim	AnalogueValue		M
lLim	AnalogueValue		M
llLim	AnalogueValue		M
min	AnalogueValue		M
max	AnalogueValue		M
limDb	INT32U	0 ... 100 000	O

hhLim, hLim, lLim, llLim: These attributes shall be the configuration parameters used in the context with the range attribute as defined in Clause 8.

min: the min (minimum) attribute shall represent the minimum process measurement for which values of *i* or *f* are considered within process limits. If the value is lower, q shall be set accordingly (validity = questionable, detailQual = outOfRange).

max: the max (maximum) attribute shall represent the maximum process measurement for which values of *i* or *f* are considered within process limits. If the value is higher, q shall be set accordingly (validity = questionable, detailQual = outOfRange).

limDb: The value is used to introduce a hysteresis in the calculation of range. Range is immediately set to the higher value, when a high limit has been crossed (to the lower value, when a low limit has been crossed). However, range is only set back to the lower value, when the value of the high limit minus limDb has been crossed (to the higher value when the value of the low limit plus limDb has been crossed). The value shall represent the percentage between max and min in units of 0,001 %. If limDb is not present, no hysteresis calculation is made.

6.6 Step position with transient indication

Step position with transient indication type is for example used to indicate the position of tap changers and shall be as defined in Table 7.

Table 7 – Step position with transient indication

ValWithTrans type definition			
Attribute name	Attribute type	Value/Value range	M/O/C
posVal	INT8	-64 ... 63	M
transInd	BOOLEAN		O

The posVal shall contain the step position, the transInd shall indicate that the equipment is in a transient state.

6.7 Pulse configuration

Pulse configuration type is used to configure the output pulse generated with a command and shall be as defined in Table 8.

Table 8 – Pulse configuration

PulseConfig type definition			
Attribute name	Attribute type	Value/Value range	M/O/C
cmdQual	ENUMERATED	pulse persistent	M
onDur	INT32U		M
offDur	INT32U		M
numPls	INT32U		M

cmdQual: this identifier shall define if the control output is a pulse output or if it is a persistent output. If it is set to pulse, then the duration of the pulse shall be defined with the identifiers onDur, offDur and numPls. If it is set to persistent, the output stays in the state indicated in the operate service.

onDur, offDur, numPls: as the result of receiving an Operate service, a pulsed output may be generated to the on or off input of a switching device. The shape of this output is defined by onDur, offDur and numPls according to Figure 4. NumPls shall specify the number of pulses that are generated. onDur shall specify the on duration of the pulse, offDur specifies the duration between two pulses. onDur and offDur shall be specified in ms; a value of 0 ms shall specify that the duration is locally defined.

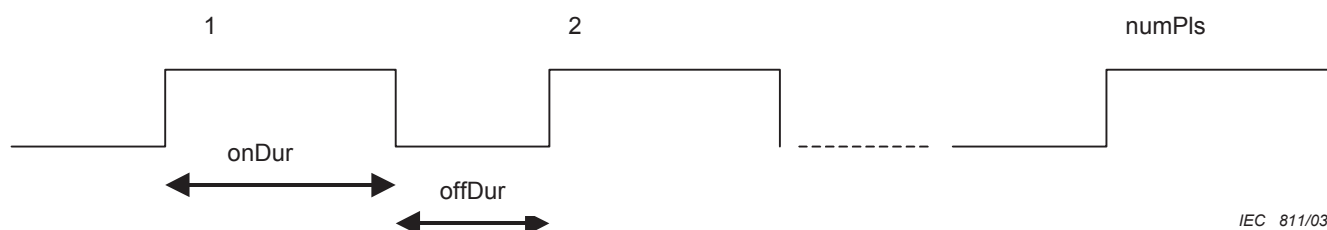


Figure 4 – Configuration of command output pulse

6.8 Originator

Originator type shall be as defined in Table 9.

Table 9 – Originator

Originator type definition			
Attribute name	Attribute type	Value/Value range	M/O/C
orCat	ENUMERATED	not-supported bay-control station-control remote-control automatic-bay automatic-station automatic-remote maintenance process	M
orIdent	OCTET STRING64		M

orCat: The originator category shall specify the category of the originator. An explanation of the values for orCat is given in Table 10.

Table 10 – Values for orCat

Value	Explanation
not-supported	That value shall not be used
bay-control	Control operation issued from an operator using a client located at bay level
station-control	Control operation issued from an operator using a client located at station level
remote-control	Control operation from a remote operator outside the substation (for example network control center)
automatic-bay	Control operation issued from an automatic function at bay level
automatic-station	Control operation issued from an automatic function at station level
automatic-remote	Control operation issued from a automatic function outside of the substation
maintenance	Control operation issued from a maintenance/service tool
process	Status change occurred without control action (for example external trip of a circuit breaker or failure inside the breaker)

orIdent: the originator identification shall show the identification of the originator. The value of NULL shall be reserved to indicate that the originator of a particular action is not known.

6.9 Unit definition

Unit type shall be as defined in Table 11.

Table 11 – Unit

Unit type definition			
Attribute name	Attribute type	Value/Value range	M/O/C
SIUnit	ENUMERATED	According to Tables A.1 to A.4 in Annex A	M
multiplier	ENUMERATED	According to Table A.5 in Annex A	O

SIUnit: shall define the SI unit according to Annex A.

multiplier: shall define the multiplier value according to Annex A. The default value is 0 (i.e. multiplier = 1).

6.10 Vector definition

Vector type shall be as defined in Table 12.

Table 12 – Vector

Vector type definition			
Attribute name	Attribute type	Value/Value range	M/O/C
mag	AnalogueValue		M
ang	AnalogueValue	$-180 < n \leq +180$	AC_CLC_O

mag: the magnitude of the complex value.

ang: the angle of the complex value. The SIUnit shall be degrees and the unit multiplier is 1. The angle reference is defined in the context where the Vector type is used.

6.11 Point definition

Point type shall be as defined in Table 13 and is used to represent points in a two- or three-dimensional coordinates system.

Table 13 – Point

Point type definition			
Attribute name	Attribute type	Value/Value range	M/O/C
xVal	FLOAT32		M
yVal	FLOAT32		M
zVal	FLOAT32		O

xVal: the x value of a point.

yVal: the y value of a point.

zVal: the z value of a point.

6.12 CtlModels definition

CtlModels type is defined as follows:

ENUMERATED (status-only | direct-with-normal-security | sbo-with-normal-security | direct-with-enhanced-security | sbo-with-enhanced-security)

Details are provided in Clause 8.

6.13 SboClasses definition

SboClasses type is defined as follows:

ENUMERATED (operate-once | operate-many)

Details are provided in Clause 8.

6.14 Cell

Cell type is used to define a rectangle area in a two-dimensional environment and shall be defined as in Table 14. Cell type can as well be used to describe a range within a one-dimensional environment. For details, see Figure 5.

Table 14 – Cell

Cell type definition			
Attribute name	Attribute type	Value/Value range	M/O/C
xStart	FLOAT32		M
xEnd	FLOAT32		O
yStart	FLOAT32		O
yEnd	FLOAT32		O

xStart: the x value of the lower left corner of the square.

xEnd: the x value of the upper right corner of the square. That component shall not be present to indicate infinity in the direction of the x axis.

yStart: The y value of the lower left corner of the square. That component shall not be present, if only a one-dimensional range needs to be described.

yEnd: The y value of the upper right corner of the square. That component shall not be present, if only a one-dimensional range needs to be described or to indicate infinity in the direction of the y axis.

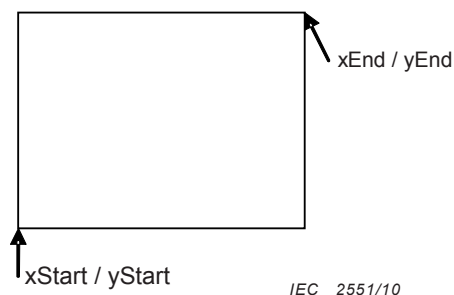


Figure 5 – Cell definition

6.15 CalendarTime definition

CalendarTime type is used to define a time setting in reference to the calendar and shall be as defined in Table 15. That constructed attribute class allows the specification of times like the last day of the month or the second Sunday in March at 03.00h.

Table 15 – CalendarTime

CalendarTime type definition			
Attribute name	Attribute type	Value/Value range	M/O/C
occ	INT16U		M
occType	ENUMERATED	Time, WeekDay, WeekOfYear, DayOfMonth, DayOfYear	M
occPer	ENUMERATED	Hour, Day, Week, Month, Year	M
weekDay	ENUMERATED	reserved, Monday, Tuesday, ... Sunday	M
month	ENUMERATED	reserved, January, February, ... December	M
day	INT8U	1..31	M
hr	INT8U	0..23	M
mn	INT8U	0..59	M

occ: Occurrence of a calendar element. The value 0 is used to indicate the last. For the identification of week numbers, week number 01 shall always be the first week in January (according to definition of UN / CEFAC).

occType: the kind of calendar element that is used for the occurrence.

occPer: the repetition period of a calendar-based time setting.

weekDay: the weekday.

month: the month.

day: the day.

hr: the hour.

mn: the minute.

The semantic interpretation of the attributes is given in Table 16.

Table 16 – Semantic interpretation of calendar time settings

occPer	occType	
Hour	Time	At <mn> minute every hour
Day	Time	At <hr>, <mn> every day
Week	WeekDay	At <weekDay>, <hr>, <mn> every week
Month	WeekDay	At <occ>, <weekDay>, <hr>, <mn> every month
Month	DayOfMonth	At <occ>, <hr>, <mn> every month
Year	Time	At <month>, <day>, <hr>, <mn> every year
Year	WeekDay	At <occ>, <weekDay>, <month>, <hr>, <mn> every year
Year	WeekOfYear	At week <occ>, <weekDay>, <hr>, <mn>
Year	DayOfYear	At <occ>, <hr>, <mn> every year

7 Common data class specifications

7.1 General

Common data classes are defined for use in IEC 61850-7-4. Common data classes are composed of constructed attribute classes defined in Clause 6 of this document or of types defined in IEC 61850-7-2 or of common data classes defined in this clause. IEC 61850-7-1 provides the basic notation used in this clause.

The common data classes define the relation between their attributes and the functional constraint as well as the possible trigger options. If two trigger options are stated, then a concrete implementation shall select one of them. The selection is based on the purpose of the data object of this common data class and is fix for the data object within a LN class.

The semantic of the SubDataObjects and DataAttributes is defined in Clause 8.

7.2 Name spaces

Name spaces are defined to identify extensions to the present definitions of IEC 61850-7-3 and IEC 61850-7-4. The name space is based on a hierarchical structure from logical node zero LLN0 at the top down to the common data class CDC. See Table 17.

Table 17 – Name space attributes

Attribute	Application	Scope of the standard specified with the attribute
IdNs	The DataAttribute IdNs shall be included in the logical node LLN0 .	LN class definition (CDC definition by reference)
InNs	The DataAttribute InNs shall be included if the name space of the LN deviates from the name space of the logical device in which the LN is defined.	LN Class definition (CDC definition by reference)
cdcNs	The DataAttribute cdcNs shall be included if the definition of at least one SubDataObject, DataAttribute, or SubAttribute of the CDC deviates from the definition in the specification in which the CDC of the DataObject is defined. In that case, the name of the new CDC is provided in the data attribute cdcName.	CDC definition
dataNs	The DataAttribute dataNs shall be included if the name space of the DataObject deviates from the name space of the logical node in which the DataObject is defined.	LN class definition (CDC definition by reference)

7.3 Common data class specifications for status information

7.3.1 Application of services

Table 18 defines the basic status information template. In particular, it defines the inheritance and specialization of services defined in IEC 61850-7-2.

Table 18 – Basic status information template

Basic status information template					
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					
substitution and blocked					
configuration, description and extension					
Services (see IEC 61850-7-2)					
The following services are inherited from IEC 61850-7-2. They are specialized 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 ST ST ST		As specified within the data set that is used to define the content of the message	