

DIN EN 61850-7-2

ICS 33.040.40

Ersatz für
DIN EN 61850-7-2:2011-04
Siehe Anwendungsbeginn

**Kommunikationsnetze und -systeme für die Automatisierung in der elektrischen Energieversorgung –
Teil 7-2: Grundlegende Informations- und Kommunikationsstruktur –
Abstrakte Schnittstelle für Kommunikationsdienste (ACSI)
(IEC 61850-7-2:2010 + A1:2020);
Englische Fassung EN 61850-7-2:2010 + A1:2020**

Communication networks and systems for power utility automation –
Part 7-2: Basic information and communication structure –
Abstract communication service interface (ACSI)
(IEC 61850-7-2:2010 + A1:2020);
English version EN 61850-7-2:2010 + A1:2020

Réseaux et systèmes de communication pour l'automatisation des systèmes électriques –
Partie 7-2: Informations de base et structure de communication –
Interface abstraite pour les services de communication (ACSI)
(IEC 61850-7-2:2010 + A1:2020);
Version anglaise EN 61850-7-2:2010 + A1:2020

Gesamtumfang 240 Seiten

DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE

Anwendungsbeginn

Anwendungsbeginn für die von CENELEC am 2010-10-01 angenommene Europäische Norm und die am 2020-03-16 angenommene Änderung A1 als DIN-Norm ist 2022-08-01.

Für DIN EN 61850-7-2:2011-04 besteht eine Übergangsfrist bis 2023-03-16.

Nationales Vorwort

Vorausgegangener Norm-Entwurf: E DIN EN 61850-7-2/A1:2018-07.

Für dieses Dokument ist das nationale Arbeitsgremium DKE/K 952 „Netzleittechnik“ der DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE (www.dke.de) zuständig.

Aktuelle Informationen zu diesem Dokument können über die Internetseiten von DKE (www.dke.de) und DIN (www.din.de) durch eine Suche nach der Dokumentennummer aufgerufen werden.

Die enthaltene IEC-Publikation wurde vom IEC/TC 57 „Power systems management and associated information exchange“ erarbeitet und unterliegt dem Copyright der IEC .

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

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

Für den Fall einer undatierten Verweisung im normativen Text (Verweisung auf ein Dokument ohne Angabe des Ausgabedatums und ohne Hinweis auf eine Abschnittsnummer, eine Tabelle, ein Bild usw.) bezieht sich die Verweisung auf die jeweils aktuellste Ausgabe des in Bezug genommenen Dokuments.

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

Der Zusammenhang der zitierten Dokumente mit den entsprechenden deutschen Dokumenten 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.

Diese Norm ist die Übernahme der konsolidierten Edition 2.1 von IEC 61850-7-2:2010 + A1:2020. Durch das Amendment sind sehr umfangreiche Änderungen fast jeden Abschnittes durchgeführt worden. Deshalb wurde auf die Kennzeichnung der einzelnen Änderungen verzichtet.

Die Inhalte von Teilen des Abschnitts 6, einige UML-Diagramme sowie der Anhang B werden auf Grundlage des UML-Modells automatisch erzeugt.

Abweichend von dem üblichen Vorgehen der IEC wurden im Sinne einer besseren Nutzungsfreundlichkeit in diesem Falle alle Tabellen und Bilder (einschließlich jener, die seit Ausgabe 2 hinzugefügt wurden) in der Änderung und der konsolidierten Fassung fortlaufend nummeriert.

Diese IEC-Norm beinhaltet Code-Elemente, d. h. Elemente, die für die direkte Verarbeitung durch einen Rechner vorgesehen sind. Dies betrifft jeglichen Text, der sich zwischen den Kennzeichnungen <CODE BEGINS> und <CODE ENDS> befindet oder in dieser Norm anderweitig eindeutig als Code-Element gekennzeichnet ist.

In diesem Dokument werden die folgenden Drucktypen verwendet:

- Fettdruck wird verwendet, um bestimmte Begriffe hervorzuheben,
- Tahoma bold wird verwendet, wenn der Unterschied zwischen einem großen i (I) und einem kleinen L (l) wichtig ist,
- Tabellenummerierungen mit dem zusätzlichen Zeichen „N“ (z. B. Tabelle 16N) sind Tabellen, die der Nummerierung der Ed 2.1 folgen.

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 EN 61850-7-2 veröffentlicht. Zu deren Abschnitt 1, der den Anwendungsbereich festlegt und Abschnitt 3, der die Begriffe festlegt, sowie Abschnitt 4, der die Abkürzungen festlegt, wurde eine Übersetzung angefertigt und als informativer Nationaler Anhang NA der vorliegenden Norm hinzugefügt. 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.

Das Original-Dokument enthält Bilder in Farbe, die in der Papierversion in einer Graustufen-Darstellung wiedergegeben werden. Elektronische Versionen dieses Dokuments enthalten die Bilder in der originalen Farbdarstellung.

Änderungen

Gegenüber DIN EN 61850-7-2:2011-04 wurden folgende Änderungen vorgenommen:

- a) die Klassendiagramme wurden aktualisiert;
- b) die Datentypen wurden in einem Dokument zusammengeführt und mit IEC 61850-6:2007B harmonisiert;
- c) Fehler und Tippfehler wurden korrigiert;
- d) die Klassen gemeinsamer Daten (CDC, en: common data class) wurden in IEC 61850-7-3:2007B verschoben, damit alle CDCs in einem Dokument zu finden sind;
- e) mehrere Begriffe wurden mit Begriffen aus anderen Teilen harmonisiert;
- f) die Definitionen und Erläuterungen der Werte der einzelnen Attribute wurden in die Klassendefinitionstabellen verschoben;
- g) die Definitionen und Erläuterungen der Werte der einzelnen Dienstparameter wurden in die Dienstparameter-Definitionstabellen verschoben;
- h) das Benennungsschema für Enumeration- und CodedEnum-Typen wurde zu <EnumTitel>Art bzw. <CodedEnumTitel>Art harmonisiert – vollständige Rückwärtskompatibilität, da sich der Wert des Literals nicht geändert hat;
- i) das USVCB-Modell wurde zurückgewiesen;
- j) Erläuterungen und Korrekturen zu/an der zweiten Ausgabe von IEC 61850-7-2 auf Grundlage der technischen Problempunkte (Tissue) = {728, 730, 778, 780, 783, 786, 813, 820, 850, 852, 858, 860, 861, 869, 875, 876, 943, 970, 1038, 1050, 1061, 1062, 1071, 1091, 1092, 1116, 1122, 1127, 1145, 1154, 1194, 1202, 1232, 1242, 1252, 1276, 1283, 1307, 1308, 1319, 1338, 1341, 1356, 1377, 1386, 1428, 1432, 1433, 1435, 1439, 1455, 1569, 1589, 1622, 1630, 1650, 1652}.

Frühere Ausgaben

DIN EN 61850-7-2: 2004-02, 2011-04

Nationaler Anhang NA (informativ)

Auszugsweise Übersetzung

Reihenfolge und Inhalt der folgenden Abschnitte sind identisch mit denen in den jeweiligen Abschnitten der Englischen Fassung.

1 Anwendungsbereich

1.1 Allgemeines

Dieses Dokument ist anwendbar für die ACSI-Kommunikation für die Versorgungsautomation. ACSI bietet die folgenden abstrakten Schnittstellen für Kommunikationsdienste.

- a) Abstrakte Schnittstelle zur Beschreibung der Kommunikation zwischen einem Client und einem entfernten Server für
 - Echtzeit-Datenzugriff und -abruf,
 - Gerätesteuerung,
 - Ereignisberichterstattung und Protokollierung,
 - Steuerung von Einstellungsgruppen,
 - Selbstbeschreibung von Geräten (Gerätedatenwörterbuch),
 - Datentypisierung und Erkennung von Datentypen sowie
 - Dateiübertragung.
- b) Abstrakte Schnittstelle zur schnellen und zuverlässigen systemweiten Ereignisverteilung zwischen einer Anwendung in einem Gerät und vielen entfernten Anwendungen in verschiedenen Geräten (Publisher/Sub-Scriber) und zur Übertragung von abgetasteten Messwerten (Publisher/Subscriber).

1.2 Namespace name and version (Name und Version des Namensraums)

Dieser neue Abschnitt ist obligatorisch für alle Namensräume nach IEC 61850 (entsprechend der Definition in IEC 61850-7-1, Ausgabe 2).

Die folgenden Parameter identifizieren diese neue Version dieses Namensraums:

- Version des Namensraums: 2007
- Überarbeitung des Namensraums: B
- Name des Namensraums: „IEC 61850-7-2:2007B“
- Veröffentlichung des Namensraums: 3
- Veröffentlichungsdatum des Namensraums: 2019-10-02

In der untenstehenden Tabelle werden alle veröffentlichten Versionen dieses Namensraums aufgeführt.

Ausgabe	Veröffentlichungsdatum	Webstore	Namensraum
Ausgabe 1.0	2003-05	IEC 61850-7-2:2003	IEC 61850-7-2:2003
Ausgabe 2.0	2010-10	IEC 61850-7-2:2010	IEC 61850-7-2:2007
Ausgabe 2.0, Änderung 1	2020-02	IEC 61850-7-2:2010/AMD1:2020	IEC 61850-7-2:2007B
Ausgabe 2.1	2020-02	IEC 61850-7-2:2010+AMD1:2020 CSV	IEC 61850-7-2:2007B

1.3 Code Component distribution (Verteilung der Code-Elemente)

Die Code-Elemente (en: code component) werden in einer eingeschränkten Version („Light“) und einer Vollversion („Full“) zur Verfügung stehen.

- Die Vollversion enthält die Definition der gesamten in dieser Norm definierten Basistypen (en: basic type) sowie die zugehörige Dokumentation; der Zugriff wird nur Personen gewährt, die diesen Teil erworben haben.
- Die eingeschränkte Version enthält ebenfalls die Definition der gesamten in dieser Norm definierten Basistypen, nicht aber die zugehörige Dokumentation; diese eingeschränkte Version wird kostenfrei auf der IEC-Website zum Download bereitstehen, die Verwendung ist jedoch an entsprechende Lizenzierungsbedingungen geknüpft.

Die eingeschränkte Version dieser Code-Elemente kann unter dem folgenden Link heruntergeladen werden:

http://www.iec.ch/public/TC57/supportdocuments/IEC_61850-7-2.NSD.2007B3.light.zip^{N1}

Die Code-Elemente für Datenmodelle nach IEC 61850 (wie die Definition der Basistypen, Anwesenheitsbedingungen usw. in dieser IEC-Norm) stehen in dem Dateiformat NSD nach der Norm IEC 61850-7-7 zur Verfügung.

Bei den in dieser IEC-Norm enthaltenen Code-Elementen besteht stets die Möglichkeit, dass sie überarbeitet wurden; Anwender müssen deshalb stets die neueste Veröffentlichung aus dem unter dem folgenden Link erreichbaren Verzeichnis auswählen:

<http://www.iec.ch/TC57/supportdocuments>

Die neueste Version/Veröffentlichung des Dokuments ist diejenige IEC 61850-7-2.NSD.{Versionsinfo}.light.zip-Datei mit dem höchsten Wert unter Versionsinfo.

Jedes Code-Element ist ein ZIP-Paket mit der elektronischen Wiedergabe des Code-Elements selbst und einer Datei, die den Inhalt des Pakets beschreibt (IECManifest.xml).

Das IECManifest umfasst mehrere Abschnitte mit Informationen zu den folgenden Aspekten:

- Urheberrechtsvermerk
- Kennung des Code-Elements
- Publikation, zu der das Code-Element gehört
- Liste der elektronischen Dateien, aus denen sich das Code-Element zusammensetzt
- optionale Liste der früheren Dateien, damit der Änderungsverlauf des Code-Elements nachvollzogen werden kann

Der Lebenszyklus eines Code-Elements ist nicht auf den Lebenszyklus der zugehörigen Publikation beschränkt. Der Lebenszyklus einer Publikation umfasst zwei Stufen, die Version (die einer Ausgabe entspricht) und die Überarbeitung (die einer Änderung entspricht). Mittels einer dritten Publikationsstufe (Veröffentlichung (en: release)) kann das Code-Element veröffentlicht werden, ohne dass eine Änderung veröffentlicht werden muss.

^{N1} Nationale Fußnote: Der auch in IEC 61850-7-2 Ed. 2.1:2020-02 nicht funktionierende Link wurde so übernommen.

Dies ist nützlich, wenn InterOp-TIssues zu beheben sind. In diesem Falle wird eine neue Veröffentlichung des Code-Elements veröffentlicht, die die vorangehende Veröffentlichung ersetzt, und über die Website des IEC TC57 verteilt.

3 Begriffe

Für die Anwendung dieses Dokuments gelten die Begriffe nach IEC 61850-2^{N2} und die folgenden Begriffe.

ISO und IEC stellen terminologische Datenbanken für die Verwendung in der Normung unter den folgenden Adressen bereit:

- IEC Electropedia: verfügbar unter <http://www.electropedia.org/>
- ISO Online Browsing Platform: verfügbar unter <http://www.iso.org/obp>

3.1

Klasse

en **class**

Beschreibung einer Gruppe von Objekten, die dieselben Attribute, Dienste, Beziehungen und Semantiken aufweisen

3.2

Kunde

en **client**

Einheit, die einen Dienst von einem Server anfordert und die unaufgeforderte Nachrichten von einem Server empfängt

3.3

Gerät

en **device**

Einheit, die Steuerungs-, Betätigungs- und/oder Erfassungsfunktionen ausführt und Schnittstellen zu anderen derartigen Einheiten innerhalb eines Automatisierungssystems bildet

Anmerkung 1 zum Begriff: Geräte allein übernehmen keine Funktionen der Energieerzeugung, -übertragung oder -verteilung

3.4

Instanz (einer Klasse)

en **instance (of a class)**

Einheit, die eine eindeutige Identität hat, auf die eine Reihe von Diensten angewendet werden kann und die einen Zustand hat, der die Auswirkungen der Dienste speichert

Anmerkung 1 zum Begriff: Instanz ist ein Synonym für den Begriff Objekt.

3.5

Logisches Gerät

en **logical device**

Einheit, die eine Reihe typischer Automatisierungs-, Schutz- oder anderer Funktionen darstellt

^{N2} Nationale Fußnote: Fehler in IEC 61850-7-2 Ed. 2.1:2020-02, korrekt ist IEC TS 61850-2.

3.6**Logischer Knoten**en **logical node**

Einheit, die eine typische Automatisierungs-, Schutz- oder andere Funktion darstellt

3.7**Physisches Gerät**en **physical device**

Einheit, die die physischen Teile eines Geräts darstellt (Hardware und Betriebssystem usw.)

Anmerkung 1 zum Begriff: Physische Geräte enthalten logische Geräte.

4 Abkürzungen

AA	Anwendungsassoziation (en: application association)
ACSI	abstrakte Kommunikationsdienstschnittstelle (en: abstract communication service interface)
BRCB	Steuerblock für gepufferte Berichte (en: buffered report control block)
CB	Steuerblock (en: control block)
CDC	Gemeinsame Datenklasse (IEC 61850-7-3) (en: common data class (IEC 61850-7-3))
CT	Stromwandler (en: current transformer)
DA	Datenattribut (en: data attribute)
DataRef	Datenreferenz (en: data reference)
dchg	Triggeroption für Datenänderungen (en: data change trigger option)
DS	Datensatz (en: DataSet)
dupd	Trigger-Option für Datenaktualisierung (en: data-update trigger option)
FC	funktionale Einschränkung (en: functional constraint)
FCD	funktionale Datenbeschränkung (en: functional constrained data)
FCDA	Attribut für funktionale Beschränkung der Daten (en: functional constrained data attribute)
GI	allgemeine Abfrage (en: general interrogation)
GoCB	GOOSE-Kontrollblock (en: GOOSE control block)
GOOSE	generische objektorientierte Unterstationsereignisse (en: generic object oriented substation events)
GSE	generisches Unterstationsereignis (en: generic substation event)
GsCB	GSSE-Steuerblock (en: GSSE control block)
GSSE	generisches Unterstationsstatus-Ereignis (en: generic substation status event)
ICD	Beschreibungsdatei der IED-Konfiguration (en: IED configuration description file)
IED	intelligentes elektronisches Gerät (en: intelligent electronic device)
IID	instanzierte IED-Beschreibungsdatei (en: instantiated IED description file)
IntgPd	Integritätsperiode (en: integrity period)
LCB	Log Control Block (en: log control block)
LD	logisches Gerät (in diesem Dokument wird die generische logische Geräteklasse definiert (en: genLogicalDevice))
LN	logischer Knoten (in diesem Dokument wird die generische Klasse des logischen Knotens definiert (en: genLogicalNode))
MC	Multicast
MCAA	Multicast-Anwendungsverband (en: multicast application association)

MMS	Spezifikation der Fertigungsmeldung (ISO 9506) (en: manufacturing message specification (ISO 9506))
MSVCB	multicast sampled value control block (en: multicast sampled value control block)
PresCond	Anwesenheitsbedingung (en: presence condition)
PDU	Protokoll-Dateneinheit (en: protocol data unit)
PICS	Konformitätserklärung zur Protokollimplementierung (en: protocol implementation conformance statement)
PIXIT	Zusatzinformationen zur Protokollimplementierung für Prüfungen (en: protocol implementation extra information for testing)
qchg	Auslöseoption für Qualitätsänderungen (en: quality change trigger option)
SBO	vor dem Betrieb auswählen (en: select before operate)
SCL	Systemkonfigurationssprache (en: system configuration language) (IEC 61850-6)
SCSM	spezifische Abbildung von Kommunikationsdiensten (definiert in IEC 61850-8-x und IEC 61850-9-x)
SGCB	Einstellung des Gruppensteuerblocks (en: setting group control block)
SoE	Ereignisfolge (en: sequence-of-events)
SVC	Steuerung von Abtastwerten
TAI	internationale Atomzeit (fr: temps atomique international)
TP	Zweiergruppe/Zweiparteien (en: two party)
TPAA	Zweiparteien-Anwendungsverband (en: two party application association)
TrgOp	Trigger-Option (en: trigger option)
UCA ^{TM1}	Architektur der Dienstleistungskommunikation (en: Utility Communication Architecture)
URCB	ungepufferter Berichtskontrollblock (en: unbuffered report control block)
UTC	Koordinierte Weltzeit (en: coordinated universal time)
SV	abgetasteter Wert/Musterwert (en: sampled value)
USVCB	Unicast-Abtastwert-Steuerblock (en: unicast sampled value control block)
VT	Spannungswandler (en: voltage transformer)

¹ UCATM ist eine eingetragene Marke von EPRI, Palo Alto (USA). Diese Informationen dienen der Benutzerfreundlichkeit und stellen keine Befürwortung des genannten Produkts durch die IEC dar.

EUROPEAN STANDARD

EN 61850-7-2:2010-10

+ A1:2020-04

NORME EUROPÉENNE

EUROPÄISCHE NORM

ICS 33.200

Supersedes EN 61850-7-2:2003

English version

Communication networks and systems for power utility
automation – Part 7-2: Basic information and communication
structure – Abstract communication service interface (ACSI)
(IEC 61850-7-2:2010 + A1:2020)

Réseaux et systèmes de communication pour
l'automatisation des systèmes électriques –
Partie 7-2: Informations de base et structure
de communication – Interface abstraite
pour les services de communication (ACSI)
(IEC 61850-7-2:2010 + A1:2020)

Kommunikationsnetze und -systeme für die Automatisierung in
der elektrischen Energieversorgung – Teil 7-2: Grundlegende
Informations- und Kommunikationsstruktur – Abstrakte
Schnittstelle für Kommunikationsdienste (ACSI)
(IEC 61850-7-2:2010 + A1:2020)

This European Standard was approved by CENELEC on 2010-10-01 and A1 on 2020-03-16. 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, Republic of North Macedonia, Romania, Serbia, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.



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

CEN-CENELEC Management Centre: Rue de la Science 23, B-1040 Brüssel

Foreword

The text of document 57/1065/FDIS, future edition 2 of IEC 61850-7-2, 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 61850-7-2 on 2010-10-01.

This European Standard supersedes EN 61850-7-2:2003.

The major technical changes with regard to EN 61850-7-2:2003 are as follows:

- class diagrams have been updated;
- data types not required have been removed;
- errors and typos have been corrected;
- substitution model has been moved to EN 61850-7-3;
- service tracking for control blocks have been added;
- the view concept will be according to the new work on role based access (RBA);
- security issues are solved by the IEC 62351 series; and
- several terms have been harmonized with those in the other parts.

This publication has been drafted in accordance with the ISO/IEC Directives, Part 2.

In this document, the following print types are used:

- **bold** is used to highlight defined terms;
- Tahoma is used where the difference between a capital i (I) and a small L (l) is important to see.

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) 2011-07-01
- latest date by which the national standards conflicting
with the EN have to be withdrawn (dow) 2013-10-01

Annex ZA has been added by CENELEC.

Endorsement notice

The text of the International Standard IEC 61850-7-2:2010 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 61400-25-2	NOTE	Harmonized as EN 61400-25-2.
IEC 61850-8 series	NOTE	Harmonized in EN 61850-8 series (not modified).
IEC 61850-9 series	NOTE	Harmonized in EN 61850-9 series (not modified).
IEC 61850-9-1	NOTE	Harmonized as EN 61850-9-1.

European foreword A1

The text of document 57/2100/FDIS, future IEC 61850-7-2/A1, 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 61850-7-2:2010/A1:2020.

The following dates are fixed:

- latest date by which the document has to be implemented at national level by (dop) 2020-12-16
publication of an identical national standard or by endorsement
- latest date by which the national standards conflicting with the (dow) 2023-03-16
document have to be withdrawn

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

This document has been prepared under a mandate given to CENELEC by the European Commission and the European Free Trade Association.

Endorsement notice

The text of the International Standard IEC 61850-7-2:2010/A1:2020 was approved by CENELEC as a European Standard without any modification.

Annex ZA (normative)

Normative references to international publications with their corresponding European publications

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements 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 1 Where an International Publication has been modified by common modifications, indicated by (mod), the relevant EN/HD applies.

NOTE 2 Up-to-date information on the latest versions of the European Standards listed in this annex is available here: www.cenelec.eu.

<u>Publication</u>	<u>Year</u>	<u>Title</u>	<u>EN/HD</u>	<u>Year</u>
IEC/TS 61850-2	-	Communication networks and systems in substations - Part 2: Glossary	-	-
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 61850-7-1	2020	Communication networks and systems for power utility automation - Part 7-1: Basic communication structure - Principles and models	-	-
IEC 61850-7-3	2020	Communication networks and systems for power utility automation - Part 7-3: Basic communication structure - Common data classes	-	-
IEC 61850-7-4	2020	Communication networks and systems for power utility automation - Part 7-4: Basic communication structure - Compatible logical node classes and data object classes	-	-
ISO 4217	-	Codes for the representation of currencies and funds	-	-
IEEE 754	-	Binary floating-point arithmetic	-	-

CONTENTS

Foreword.....	10
European foreword A1.....	11
Annex ZA (normative) Normative references to international publications with their corresponding European publications.....	12
INTRODUCTION	24
1 Scope	25
1.1 General.....	25
1.2 Namespace name and version	25
1.3 Code Component distribution.....	26
2 Normative references	26
3 Terms and definitions	27
4 Abbreviated terms	28
5 ACSI overview and basic concepts.....	29
5.1 Conceptual model of IEC 61850	29
5.2 The meta-meta model	30
5.3 The meta model	30
5.3.1 General.....	30
5.3.2 Information modelling classes.....	30
5.3.3 Information exchange modelling classes	31
5.3.4 Relations between classes.....	35
5.4 The domain type model.....	36
5.5 The data instance model	36
5.6 Conditions for element presence	37
5.7 Functional constraints (FcKind).....	39
5.8 Trigger options	43
6 Type definition	43
6.1 General.....	43
6.2 Data model types	45
6.2.1 General.....	45
6.2.2 Basic types	45
6.2.3 Domain types (general)	47
6.2.4 Domain types (enumerated).....	58
6.2.5 Domain types (coded enumerated).....	63
6.3 ACSI – service parameter types	65
6.3.1 General.....	65
6.3.2 Basic types	65
6.3.3 Domain types	65

6.3.4	ACSI enumerated types	66
7	GenServerClass model	67
7.1	GenServerClass definition	67
7.2	Server class services	68
7.2.1	Overview of directory and GetDefinition services	68
7.2.2	GetServerDirectory	68
8	Application association model	69
8.1	Introduction	69
8.2	Concept of application associations	69
8.3	TWO-PARTY-APPLICATION-ASSOCIATION (TPAA) class model	70
8.3.1	TWO-PARTY-APPLICATION-ASSOCIATION (TPAA) class definition	70
8.3.2	Two-party application association services	71
8.4	MULTICAST-APPLICATION-ASSOCIATION (MCAA) class	74
9	GenLogicalDeviceClass model	75
9.1	GenLogicalDeviceClass definition	75
9.2	GenLogicalDeviceClass services	76
10	GenLogicalNodeClass model	76
10.1	GenLogicalNodeClass definition	76
10.1.1	GenLogicalNodeClass diagram	76
10.1.2	GenLogicalNodeClass attributes	77
10.2	GenLogicalNodeClass services	79
10.2.1	Overview	79
10.2.2	GetLogicalNodeDirectory	79
10.2.3	GetAllDataValues	80
11	Generic data object class model	80
11.1	GenDataObjectClass diagram	80
11.2	GenDataObjectClass syntax	81
11.3	GenDataObjectClass services	82
11.3.1	General definitions and overview	82
11.3.2	GetDataValues	83
11.3.3	SetDataValues	84
11.3.4	GetDataDirectory	85
11.3.5	GetDataDefinition	85
12	Generic common data class model	86
12.1	General	86
12.2	GenCommonDataClass	86
12.2.1	GenCommonDataClass diagram	86
12.2.2	GenCommonDataClass syntax	87
12.3	GenDataAttributeClass	88

12.3.1	GenDataAttributeClass diagram	88
12.3.2	GenDataAttributeClass syntax	88
12.3.3	Details of some GenDataAttributeClass attributes	89
12.4	GenConstructedAttributeClass	91
12.4.1	GenConstructedAttributeClass diagram	91
12.4.2	GenConstructedAttributeClass syntax	91
12.5	GenSubDataAttributeClass	91
12.5.1	SubDataAttributeClass diagram	91
12.5.2	SubDataAttributeClass syntax	92
12.6	Referencing data objects and their components	92
12.6.1	General	92
12.6.2	Reference syntax	92
12.6.3	Base types and their relation	93
12.6.4	Example of using references	94
13	DATA-SET class model	95
13.1	General	95
13.2	DATA-SET class definition	96
13.2.1	DATA-SET class syntax	96
13.2.2	DATA-SET class attributes	97
13.3	DATA-SET class services	98
13.3.1	Overview	98
13.3.2	GetDataSetValues	98
13.3.3	SetDataSetValues	99
13.3.4	CreateDataSet	99
13.3.5	DeleteDataSet	100
13.3.6	GetDataSetDirectory	100
14	Service tracking	101
14.1	General service tracking	101
14.2	Control block service tracking	101
14.3	Control service tracking	102
15	Modelling of control block classes	103
15.1	General	103
15.2	Control block class models	103
16	SETTING-GROUP-CONTROL-BLOCK class model	103
16.1	General	103
16.2	SGCB class definition	105
16.3	SGCB class services	106
16.3.1	Overview	106
16.3.2	SelectActiveSG	107

16.3.3	SelectEditSG	107
16.3.4	SetEditSGValue.....	108
16.3.5	ConfirmEditSGValues.....	109
16.3.6	GetEditSGValue	109
16.3.7	GetSGCBValues.....	110
17	REPORT-CONTROL-BLOCK and LOG-CONTROL-BLOCK class models	110
17.1	Overview.....	110
17.2	REPORT-CONTROL-BLOCK class model	112
17.2.1	Basic concepts	112
17.2.2	BUFFERED-REPORT-CONTROL-BLOCK (BRCB) class definition	113
17.2.3	BRCB class services	123
17.2.4	UNBUFFERED-REPORT-CONTROL-BLOCK (URCB) class definition	133
17.2.5	URCB class services	135
17.3	LOG-CONTROL-BLOCK class model.....	136
17.3.1	General.....	136
17.3.2	LCB class definition	137
17.3.3	LCB services	138
17.3.4	LOG class definition	140
17.3.5	Reason code for log entries.....	142
17.3.6	LOG services.....	142
18	Generic substation event class model (GSE).....	145
18.1	Overview.....	145
18.2	GOOSE-CONTROL-BLOCK (GoCB) class.....	146
18.2.1	GoCB definition	146
18.2.2	GOOSE service definitions.....	147
18.2.3	Generic object oriented substation event (GOOSE) message.....	150
19	Transmission of sampled value class model	151
19.1	Overview.....	151
19.2	Transmission of sampled values using multicast	153
19.2.1	General.....	153
19.2.2	MSVCB class definition	153
19.2.3	MSVCB class services	154
19.3	Transmission of sampled values using unicast	157
19.3.1	General.....	157
19.3.2	USVCB class definition.....	158
19.3.3	USVCB services	159
19.4	Sampled value format.....	161
20	CONTROL class model	162
20.1	General.....	162

20.2	Control with normal security	165
20.2.1	General.....	165
20.2.2	Direct control with normal security	165
20.2.3	SBO control with normal security	168
20.3	Control with enhanced security	171
20.3.1	General.....	171
20.3.2	Direct control with enhanced security	171
20.3.3	SBO control with enhanced security	173
20.4	Time-activated operate	177
20.5	CONTROL class service definitions	178
20.5.1	Overview	178
20.5.2	Service parameter definition	179
20.5.3	Service specification	180
21	Time and time-synchronization model	186
21.1	General.....	186
21.2	Information requirements	186
22	Naming conventions.....	187
22.1	Class naming and class specializations.....	187
22.2	Referencing an instance of a class	188
22.3	Scopes in IEC 61850	189
23	File transfer model.....	190
23.1	File class	190
23.2	File services	191
23.2.1	Overview	191
23.2.2	GetFile.....	191
23.2.3	SetFile	191
23.2.4	DeleteFile	192
23.2.5	GetFileAttributeValues	192
Annex A (normative)	ACSI conformance statement	193
A.1	General.....	193
A.2	ACSI basic conformance statement.....	193
A.3	ACSI models conformance statement.....	194
A.4	ACSI service conformance statement.....	196
Annex B (normative)	SCL enumerations	201
Annex C (informative)	Generic substation state event (GSSE) control block (GsCB)	202
C.1	General.....	202
C.2	GsCB class definition	202
C.3	Generic substation state event (GSSE) control block class attributes.....	203
C.3.1	GsCBName – GSSE control name	203

C.3.2	GsCBRef – GSSE control reference	203
C.3.3	GsEna – GSSE enable.....	203
C.3.4	GsID – GSSE identification	203
C.3.5	DataLabel [1..n]	204
C.3.6	LSentData [1..n] – last sent data values.....	204
C.3.7	DstAddress	204
C.4	GSSE service definitions	204
C.4.1	Overview.....	204
C.4.2	SendGSSEMessage.....	205
C.4.2.1	SendGSSEMessage parameter table	205
C.4.2.2	Request	205
C.4.2.3	GetGsReference.....	205
C.4.2.4	GetGSSEDataOffset.....	206
C.4.2.5	GetGsCBValues	207
C.4.2.6	SetGsCBValues.....	208
C.4.2.7	Generic substation state event (GSSE) message.....	209
Annex D (normative)	Clarification on usage of quality.....	211
D.1	Relation of the detailed quality identifiers with invalid or questionable quality	211
D.2	Quality in the client server context	211
D.3	Relation between quality identifiers.....	212
Annex E (normative)	Clarification on RCB reservation.....	215
E.1	General	215
E.2	Buffered Report Control Block.....	215
E.2.1	Instantiation of the buffered report control block from the SCL definition	215
E.2.2	Reservation of a BRCB	217
E.2.3	Un-reservation of a BRCB	219
E.2.4	End of association and BRCB reservation scheme	221
E.3	Unbuffered Report Control Block	224
E.3.1	Instantiation of the unbuffered report control block from the SCL definition	224
E.3.2	Reservation of a URCB	225
E.3.3	Un-reservation of a URCB.....	228
E.3.4	End of association and URCB reservation scheme	229
Annex F (normative)	Compatibility of the different revisions of the standard	232
F.1	General	232
F.2	List of the modifications to consider for backward / forward compatibility	232
F.3	Special compatibility rules	238
F.3.1	Leap second by server conformant to previous version of the standard	238
F.3.2	TimeAccuracy by server conformant to previous version of the standard	238
F.3.3	RCB reservation	238

F.3.4 Name Length restriction	238
Bibliography	239
Index	240

Figures

Figure 1 – Excerpt of conceptual model of IEC 61850.....	30
Figure 2 – Basic conceptual class model of the ACSI	31
Figure 3 – Conceptual service model of the ACSI	33
Figure 4 – Core of the conceptual meta model and relationship.....	35
Figure 5 – Data instance model (conceptual).....	36
Figure 6 – P_Timestamp	49
Figure 7 – Time accuracy	51
Figure 8 – Overview about GetDirectory and GetDefinition services.....	68
Figure 9 – Normal operation.....	70
Figure 10 – Aborting association.....	70
Figure 11 – Principle of multicast application association.....	74
Figure 12 – Basic conceptual model of the GenLogicalNodeClass	77
Figure 13 – Basic conceptual class model of the GenDataObjectClass	81
Figure 14 – Excerpt of GenDataObjectClass services	83
Figure 15 – Class diagram of the GenCommonDataClass	87
Figure 16 – Conceptual Class diagram of the GenCommonDataClass.....	87
Figure 17 – Class diagram of the GenDataAttributeClass	88
Figure 18 – Relation of TrgOp and Reporting	90
Figure 19 – Class diagram of the GenConstructedAttributeClass	91
Figure 20 – Relation of types (example)	94
Figure 21 – Example of a data object.....	95
Figure 22 – Dynamic creation of DataSet instances	96
Figure 23 – Control block service mapping	102
Figure 24 – Basic model of the settings model	104
Figure 25 – Basic building blocks for reporting and logging.....	111
Figure 26 – BRCB state machine	114
Figure 27 – General queue of entries for report handler	115
Figure 28 – Buffer time	117
Figure 29 – State Machine for Sequence Number Generation	118
Figure 30 – Logical state machine for general interrogation	120
Figure 31 – Report example on the use of sequence number	125
Figure 32 – Entry discard that does not cause indication of loss of information in enabled state	126
Figure 33 – Indication of loss of information due to resource constraints in enable state	127
Figure 34 – DataSet members and reporting	128
Figure 35 – Report example	129

Figure 36 – Log model overview.....	137
Figure 37 – GoCB model	145
Figure 38 – Model for transmission of sampled values	152
Figure 39 – Principle of the control model	163
Figure 40 – State machine of direct control with normal security	167
Figure 41 – Direct control with normal security	168
Figure 42 – State machine of SBO control with normal security	170
Figure 43 – State machine of direct control with enhanced security	173
Figure 44 – State machine SBO control with enhanced security	175
Figure 45 – Select before operate with enhanced security – positive case	176
Figure 46 – Select before operate with enhanced security – negative case (no status change)	176
Figure 47 – Time-activated operate	178
Figure 48 – Time model and time synchronization (principle)	186
Figure 49 – Specializations	188
Figure 50 – Object names and object reference	189
Figure C.1 – GsCB model	202
Figure D.1 – Quality identifiers in a single client-server relationship	211
Figure D.2 – Quality identifiers in a multiple client-server relationship	212
Figure D.3 – Interaction of substitution and validity (CL=client, Se=server) when used in a proxy model environnement	214
Figure E.1 – BRCB instantiation from SCL and reservation scheme	217
Figure E.2 – BRCB reservation scheme	219
Figure E.3 – BRCB un-reservation scheme	221
Figure E.4 – BRCB reservation and end of association	223
Figure E.5 – URCB instantiation from SCL and reservation scheme	225
Figure E.6 – URCB reservation scheme	227
Figure E.7 – URCB un-reservation scheme	229
Figure E.8 – URCB reservation and end of association	231

Tables

Table 1 – Example of class instantiation	31
Table 2 – ACSI model classes with related services	34
Table 3 – Conditions for presence of elements within a context	37
Table 4 – Functional constraints (FcKind)	39
Table 5 – Trigger options	43
Table 6 – Type definition overview	43
Table 7 – Use of IEC 61850-6 bType	45
Table 8 – Attributes of P_Timestamp	50
Table 9 – Attributes of TimeQuality	51
Table 10 – Attributes of P_Quality	53

Table 11 – Attributes of DetailQual	54
Table 12 – Attributes of P_TriggerConditions	55
Table 13 – Attributes of P_RCBReportOptions	56
Table 14 – Attributes of P_LCBLogEntryOptions	57
Table 15 – Attributes of P_SVMessageOptions	57
Table 16 – Attributes of P_CheckConditions	57
Table 17 – Attributes of S_Originator	58
Table 18 – Literals of ServiceNameKind	58
Table 19 – Literals of ServiceStatusKind	60
Table 20 – Literals of OriginatorCategoryKind	61
Table 21 – Literals of ControlServiceStatusKind	61
Table 22 – Literals of SamplingModeKind	63
Table 23 – Literals of StepControlKind	64
Table 24 – Literals of DpStatusKind	64
Table 25 – Literals of SourceKind	64
Table 26 – Literals of ValidityKind	65
Table 27 – Attributes of ReasonForInclusionInReport	66
Table 28 – Attributes of ReasonForInclusionInLog	66
Table 29 – Literals of ACSIClassKind	67
Table 30 – GenServerClass definition	67
Table 31 – GetServerDirectory service parameters	69
Table 32 – TWO-PARTY-APPLICATION-ASSOCIATION (TPAA) class definition	71
Table 33 – Two-party-application-association services	71
Table 34 – Associate service parameters	72
Table 35 – Abort service parameters	73
Table 36 – Release service parameters	74
Table 37 – MULTICAST-APPLICATION-ASSOCIATION (MCAA) class definition	75
Table 38 – GenLogicalDeviceClass (GenLD) class definition	75
Table 39 – GetLogicalDeviceDirectory service parameters	76
Table 40 – GenLogicalNodeClass definition	78
Table 41 – GenLogicalNodeClass services	79
Table 42 – GetLogicalNodeDirectory service parameters	79
Table 43 – GetAllDataValues service parameters	80
Table 44 – GenDataObjectClass definition	82
Table 45 – GenDataObjectClass services	82
Table 46 – GetDataValues service parameters	84
Table 47 – SetDataValue service parameters	84
Table 48 – GetDataDirectory service parameters	85
Table 49 – GetDataDefinition service parameters	86

Table 50 – GenCommonDataClass definition	88
Table 51 – GenDataAttributeClass definition	89
Table 52 – GenConstructedAttributeClass definition	91
Table 53 – GenSubDataAttributeClass definition	92
Table 54 – DATA-SET (DS) class definition	97
Table 55 – DATA-SET services	98
Table 56 – GetDataSetValues service parameters	98
Table 57 – SetDataSetValues service parameters	99
Table 58 – CreateDataSet service parameters	99
Table 59 – DeleteDataSet service parameters	100
Table 60 – GetDataSetDirectory service parameters	100
Table 61 – CB class definition	103
Table 62 – SGCb class definition	105
Table 63 – SGCb services	107
Table 64 – SelectActiveSG service parameters	107
Table 65 – SelectEditSG service parameters	108
Table 66 – SetEditSGValue service parameters	108
Table 67 – ConfirmEditSGValues service parameters	109
Table 68 – GetEditSGValue service parameters	109
Table 69 – GetSGCbValues service parameters	110
Table 70 – BRCb class definition	113
Table 71 – BRCb services	123
Table 72 – Report service parameters	123
Table 73 – Report segment format specification	123
Table 74 – GetBRCbValues service parameters	132
Table 75 – SetBRCbValues service parameters	133
Table 76 – URCb class definition	133
Table 77 – URCb services	135
Table 78 – LCB class definition	138
Table 79 – LCB services	139
Table 80 – GetLCbValues service parameters	139
Table 81 – SetLCbValues service parameters	140
Table 82 – LOG class definition	140
Table 83 – LOG services	142
Table 84 – QueryLogByTime service parameters	143
Table 85 – QueryLogAfter service parameters	144
Table 86 – GetLogStatusValues service parameters	144
Table 87 – GOOSE control block class definition	146
Table 88 – GoCb services	148

Table 89 – SendGOOSEMessage service parameters	148
Table 90 – GetGoReference service parameters	148
Table 91 – GetGOOSEElementNumber service parameters.....	149
Table 92 – GetGoCBValues service parameters	150
Table 93 – SetGoCBValues service parameters.....	150
Table 94 – GOOSE message definition	151
Table 95 – MSVCB class definition	153
Table 96 – MSVCB services.....	154
Table 97 – SendMSVMessage service parameters.....	155
Table 98 – GetMSVCBValues service parameters	155
Table 99 – SetMSVCBValues service parameters	156
Table 100 – GetMsvReference service parameters.....	156
Table 101 – GetMSVElementNumber service parameters	157
Table 102 – USVCB class definition	158
Table 103 – USVCB services.....	159
Table 104 – SendUSVMessage service parameters	159
Table 105 – GetUSVCBValues service parameters	160
Table 106 – SetUSVCBValues service parameters.....	160
Table 107 – Sampled value (SV) format definition.....	161
Table 108 – Generic behavior and negative responses.....	164
Table 109 – Control services.....	178
Table 110 – Control service parameters	179
Table 111 – FILE class definition	190
Table 112 – File services.....	191
Table 113 – GetFile service parameters	191
Table 114 – SetFile service parameters.....	191
Table 115 – DeleteFile service parameters.....	192
Table 116 – GetFileAttributeValues service parameters.....	193
Table A.1 – Basic conformance statement.....	193
Table A.2 – ACSI models conformance statement	194
Table A.3 – ACSI service conformance statement	196
Table C.1 – GSSE control block class definition	203
Table C.2 – GsCB Services.....	204
Table C.3 – SendGSSEMessage parameter table	205
Table C.4 – GetGsReference parameter table.....	205
Table C.5 – GetGSSEDataOffset parameter table	206
Table C.6 – GetGsCBValues parameter table	207
Table C.7 – SetGsCBValues parameter table.....	208
Table C.8 – GSSE message definition.....	209
Table D.1 – Relation of the detailed quality identifiers with invalid or questionable quality	211

INTRODUCTION

This document is part of a set of definitions which details a layered utility communication architecture. This architecture has been chosen to provide abstract definitions of classes and services such that the definitions are independent of specific protocol stacks, implementations, and operating systems.

The IEC 61850 series is intended to provide interoperability between a variety of devices. Communication between these devices is achieved by the definition of a hierarchical class model (for example, logical device, logical node, data, data set, report control, or log) and services provided by these classes (for example, get, set, report, define, delete) in IEC 61850-7-x.

This document defines the abstract communication service interface (ACSI) for use in the utility application domain, which requires real-time cooperation of intelligent electronic devices. The ACSI has been defined so as to be independent of the underlying communication systems. Specific communication service mappings² (SCSM) are specified in IEC 61850-8-x and IEC 61850-9-x.

This document defines the abstract communication service interface in terms of

- a hierarchical class model of all information that can be accessed via a communication network,
- services that operate on these classes, and
- parameters associated with each service.

The ACSI description technique abstracts away from all the different approaches to implement the cooperation of the various devices.

This document does not provide comprehensive tutorial material. It is recommended that IEC 61850-5 and IEC 61850-7-1 be read first in conjunction with IEC 61850-7-2 and IEC 61850-7-3.

NOTE 1 Refer to International Electrotechnical Vocabulary, IEC 60050, for general glossary definitions.

NOTE 2 Abstraction in ACSI has two meanings. First, only those aspects of a real device (for example, a breaker) or a real function that are visible and accessible over a communication network are modelled. This abstraction leads to the hierarchical class models and their behaviour defined in IEC 61850-7-2, IEC 61850-7-3, and IEC 61850-7-4. Second, the ACSI abstracts from the aspect of concrete definitions on how the devices exchange information; only a conceptual cooperation is defined. The concrete information exchange is defined in the SCSMs.

NOTE 3 Examples use names of classes (for example XCBR for a class of a logical node) defined in IEC 61850-7-4 and IEC 61850-7-3. The normative names are defined in IEC 61850-7-4 and IEC 61850-7-3 only.

² The ACSI is independent of the specific mapping. Mappings to standard application layers or middle ware technologies are possible.

COMMUNICATION NETWORKS AND SYSTEMS FOR POWER UTILITY AUTOMATION –

Part 7-2: Basic information and communication structure – Abstract communication service interface (ACSI)

1 Scope

1.1 General

This document applies to the ACSI communication for utility automation. The ACSI provides the following abstract communication service interfaces.

- a) Abstract interface describing communications between a client and a remote server for
 - real-time data access and retrieval,
 - device control,
 - event reporting and logging,
 - setting group control,
 - self-description of devices (device data dictionary),
 - data typing and discovery of data types, and
 - file transfer.
- b) Abstract interface for fast and reliable system-wide event distribution between an application in one device and many remote applications in different devices (publisher/sub-scriber) and for transmission of sampled measured values (publisher/subscriber).

1.2 Namespace name and version

This new clause is mandatory for any IEC 61850 namespace (as defined by the part 7-1 of IEC 61850 Edition 2).

The parameters which identify this new release of this namespace are:

- Namespace version: 2007
- Namespace revision: B
- Namespace name: “IEC 61850-7-2:2007B”
- Namespace release: 3
- Namespace release date: 2019-10-02

The table below provides an overview of all published versions of this namespace.

Edition	Publication date	Webstore	Namespace
Edition 1.0	2003-05	IEC 61850-7-2:2003	IEC 61850-7-2:2003
Edition 2.0	2010-10	IEC 61850-7-2:2010	IEC 61850-7-2:2007
Amendment 1 of Edition 2.0	2020-02	IEC 61850-7-2:2010/AMD1:2020	IEC 61850-7-2:2007B
Edition 2.1	2020-02	IEC 61850-7-2:2010+AMD1:2020 CSV	IEC 61850-7-2:2007B

1.3 Code Component distribution

The Code Component will be available in light and full version:

- Full version will contain definition of the whole Basic Types defined in this standard with the documentation associated and access will be restricted to purchaser of this part
- Light version will not contain the documentation but will contain the whole definition of the Basic Types as per full version, and this light version will be freely accessible on the IEC website for download, but the usage remains under the licensing conditions.

The link for downloading the light version of this code component is:

http://www.iec.ch/public/TC57/supportdocuments/IEC_61850-7-2.NSD.2007B3.light.zip

The Code Components for IEC 61850 data models (like basic types, presence conditions, ... definition in this IEC standard) are available as the file format NSD defined by standard IEC 61850-7-7.

The Code Component(s) included in this IEC standard are potentially subject to maintenance works and user shall select the latest release in the repository located at:

<http://www.iec.ch/TC57/supportdocuments>

The latest version/release of the document will be found by selecting the file IEC_61850-7-2.NSD.{VersionStateInfo}.light.zip with the filed VersionStateInfo of the highest value.

Each Code Component is a ZIP package containing the electronic representation of the Code Component itself, with a file describing the content of the package (IECManifest.xml).

The IECManifest contains different sections giving information on:

- The copyright notice
- The identification of the code component
- The publication related to the code component
- The list of the electronic files which compose the code component
- An optional list of history files to track changes during the evolution process of the code component

The life cycle of a code component is not restricted to the life cycle of the related publication. The publication life cycle goes through two stages, Version (corresponding to an edition) and Revision (corresponding to an amendment). A third publication stage (Release) allows publication of Code Component without need to publish an amendment.

This is useful when InterOp Tissues need to be fixed. Then a new release of the Code Component will be released, which supersedes the previous release, and distributed through the IEC TC57 web site.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements 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 for power utility automation - Part 2: Glossary*

IEC 61850-5, *Communication networks and systems for power utility automation - 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 61850-7-1:2020, *Communication networks and systems for power utility automation - Part 7-1: Basic communication structure - Principles and models*

IEC 61850-7-3:2020, *Communication networks and systems for power utility automation - Part 7-3: Basic communication structure - Common data classes*

IEC 61850-7-4:2020, *Communication networks and systems for power utility automation - Part 7-4: Basic communication structure - Compatible logical node classes and data object classes*

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

IEEE 754, *Standard for Floating-Point Arithmetic*

3 Terms and definitions

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

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- IEC Electropedia available at <http://www.electropedia.org>
- ISO Online browsing platform available at <http://www.iso.org/obp>

3.1

class

description of a set of objects that share the same attributes, services, relationships, and semantics

3.2

client

entity that requests a service from a server and that receives unsolicited messages from a server

3.3

device

entity that performs control, actuating and/or sensing functions and interfaces to other such entities within an automation system

Note 1 to entry: Devices alone do not perform energy generation, transport, or distribution functions.

3.4

instance (of a class)

entity that has unique identity, to which a set of services can be applied, and which has a state that stores the effects of the services

Note 1 to entry: Instance is a synonym for the term object.

3.5

logical device

entity that represents a set of typical automation, protection or other functions

3.6

logical node

entity that represents a typical automation, protection or other function

3.7

physical device

entity that represents the physical parts of a device (hardware and operating system, etc.)

Note 1 to entry: Physical devices host logical devices.

4 Abbreviated terms

AA	application association
ACSI	abstract communication service interface
BRCB	buffered report control block
CB	control block
CDC	common data class (IEC 61850-7-3)
CT	current transformer
DA	data attribute
DataRef	data reference
dchg	data change trigger option
DS	DataSet – set of data
dupd	data-update trigger option
FC	functional constraint
FCD	functional constrained data
FCDA	functional constrained data attribute
GI	general interrogation
GoCB	GOOSE control block
GOOSE	generic object oriented substation events
GSE	generic substation event
GsCB	GSSE control block
GSSE	generic substation status event
ICD	IED configuration description file
IED	intelligent electronic device
IID	instantiated IED Description file
IntgPd	integrity period
LCB	log control block
LD	logical device (in this document, the generic logical device class is defined (genLogicalDevice))
LN	logical node (in this document, the generic logical node class is defined (genLogicalNode))
MC	multicast
MCAA	multicast application association
MMS	manufacturing message specification (ISO 9506)
MSVCB	multicast sampled value control block
PresCond	presence condition
PDU	protocol data unit
PICS	protocol implementation conformance statement
PIXIT	protocol Implementation extra information for testing
qchg	quality change trigger option
SBO	select before operate
SCL	system configuration language (IEC 61850-6)

SCSM	specific communication service mapping (defined in IEC 61850-8-x and IEC 61850-9-x)
SGCB	setting group control block
SoE	sequence-of-events
SVC	sampled value control
TAI	international atomic time
TP	two party
TPAA	two party application association
TrgOp	trigger option
UCA ^{TM3}	Utility Communication Architecture
URCB	unbuffered report control block
UTC	coordinated universal time
SV	sampled value
USVCB	unicast sampled value control block
VT	voltage transformer

5 ACSI overview and basic concepts

5.1 Conceptual model of IEC 61850

The models of the ACSI provide

- the definition of the basic model of the utility information models contained in IEC 61850-7-3 (common data classes for utility automation applications), IEC 61850-7-4 (compatible logical node classes and compatible data classes for utility automation applications) and IEC 61850-6 (substation configuration language), and
- the definition of information exchange service models.

The information models and information exchange services are interwoven. From a descriptive point of view, the two aspects are separated to some degree (see the excerpt shown in Figure 1).

The first level of the definitions is a list of base types and rules how to build hierarchical structures (meta-meta model) defined in 5.2.

The generic models (meta model) for example, for logical nodes and data classes including their services, are defined 5.3 and applied in IEC 61850-7-3 and IEC 61850-7-4 to define many specialized information models for utility automation models or in IEC 61400-25-2 to define specialized models for wind power plant applications (domain type models; see 5.4).

³ UCATM is a registered trademark of EPRI, Palo Alto (USA). This information is given for the convenience of users and does not constitute an endorsement by the IEC of the product named.

IEC 61850-6 (SCL) defines the instances to be implemented (configured) in real devices (data instance model; see 5.5).

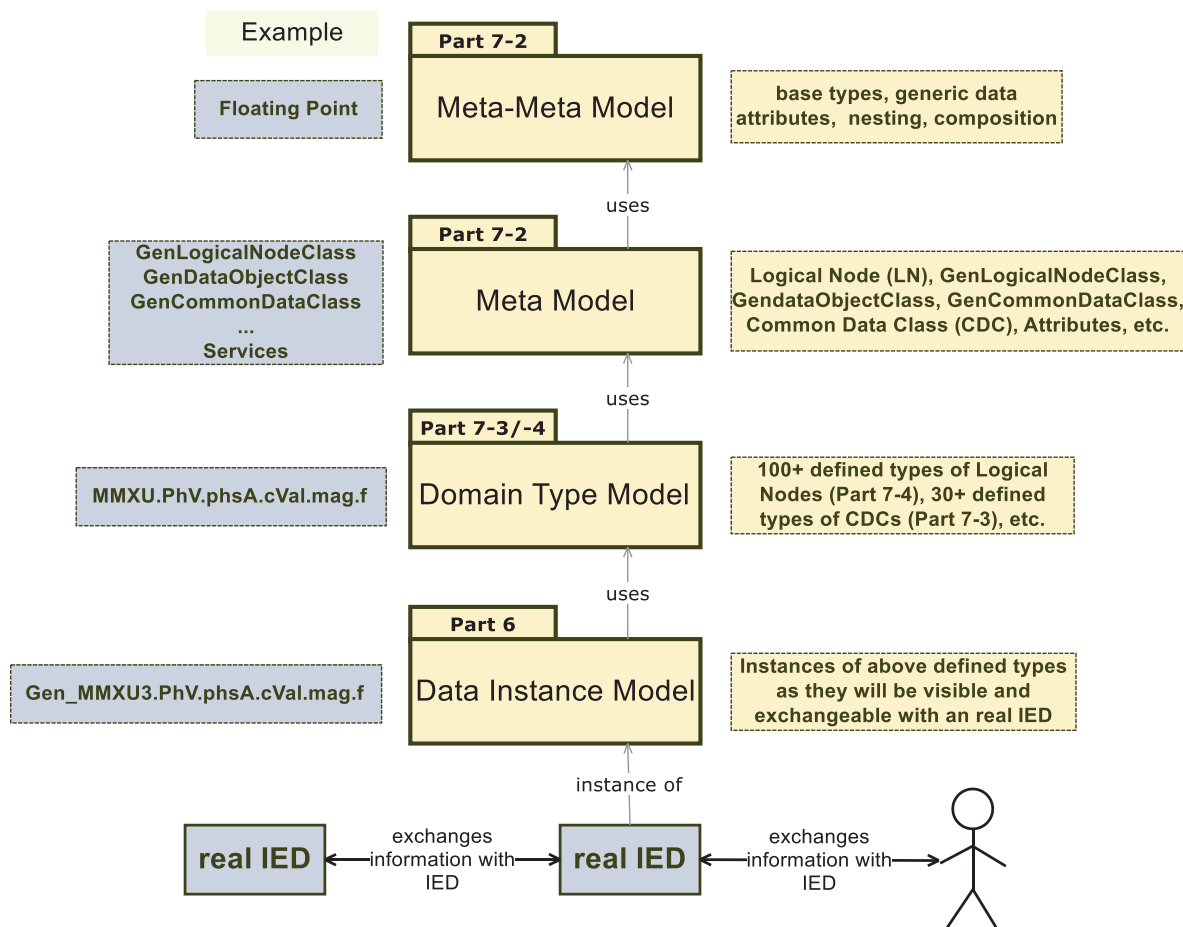


Figure 1 – Excerpt of conceptual model of IEC 61850

5.2 The meta-meta model

The meta-meta model is defined in Clause 6. It defines the basic data type classes to be used in the meta model with the exception of the recursion (nesting) of components to define hierarchical data models. The recursion is defined in the meta model.

5.3 The meta model

5.3.1 General

This document defines mainly the meta model for the whole IEC 61850 standard series. The meta model comprises classes for the description of a device with regard to data models and information exchange.

5.3.2 Information modelling classes

The following overall classes are defined:

- Server** – represents the external visible behaviour of a device. All other ACSI models are part of the server.

NOTE A server has two roles: to communicate with a client (most service models in IEC 61850 provide communication with client devices) and to send information to peer devices (for example, for sampled values).

- b) **Logical device (LD)** – represents the information produced and consumed by a group of domain-specific application functions.
- c) **Logical node (LN)** – contains the information produced and consumed by a single domain-specific application function, for example, overvoltage protection or circuit-breaker.
- d) **Data objects** – provide means to define typed information, for example, position of a switch with quality information and timestamp, contained in logical nodes.

Each of these models is defined as a class. The classes comprise attributes and services. The conceptual class diagram of the ACSI is depicted in Figure 2.

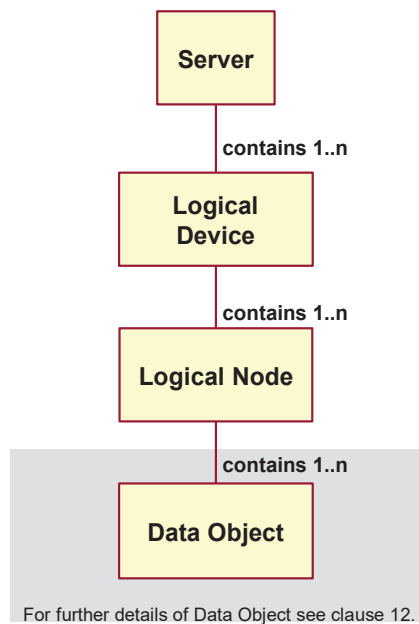


Figure 2 – Basic conceptual class model of the ACSI

Each of the following classes has a name and a reference: logical device, logical node, and data object.

EXAMPLE In an implementation, the logical device, logical node, data objects, and data attribute have each an object name (instance name) which is a unique name among classes of the same container to which they belong. In addition, each of the four has an ObjectReference (path name) which is a concatenation of all object names from each container. The four object names (one per column) can be concatenated as shown in Table 1.

Table 1 – Example of class instantiation

	Logical device	Logical node	Data object	Data attribute
Object name	"Atlanta_HV5"	"XCBR1"	"Pos"	"stVal"
Description	High-voltage station 5	Circuit-breaker 1	Position	Status value

5.3.3 Information exchange modelling classes

In addition to the models listed above, the ACSI comprises the following models that provide services operating on data objects, data attributes, and data sets.

- a) **DATA-SET** – permits the grouping of data objects and data attributes. Used for direct access, for reporting, logging, **GOOSE** messaging and sampled value exchange.

- b) **Setting group control** – defines how to switch from one set of setting values to another one and how to edit setting groups.
- c) **Report control and logging** – describe the conditions for generating reports and logs based on parameters set by configuration or by a client. Reports may be triggered by changes of process data values (for example, state change or dead band) or by quality changes. Logs can be queried for later retrieval. Reports may be sent immediately or deferred. Reports provide change-of-state and sequence-of-events information exchange.
- d) **Control blocks for generic substation event (GSE)** – supports a fast and reliable system-wide distribution of input and output data values; peer-to-peer exchange of IED binary status information, for example, a trip signal.
- e) **Control blocks for transmission of sampled values** – fast and cyclic transfer of samples, for example, of instrument transformers.
- f) **Control** – describes the services to control, for example, devices.
- g) **Time and time synchronization** – provides the time base for the device and system.
- h) **File system** – defines the exchange of large data blocks such as programs.
- i) **Tracking** – provides a diagnosis interface to track services (control, configuration, exchange).

An overview of the conceptual service model of the ACSI is shown in Figure 3.

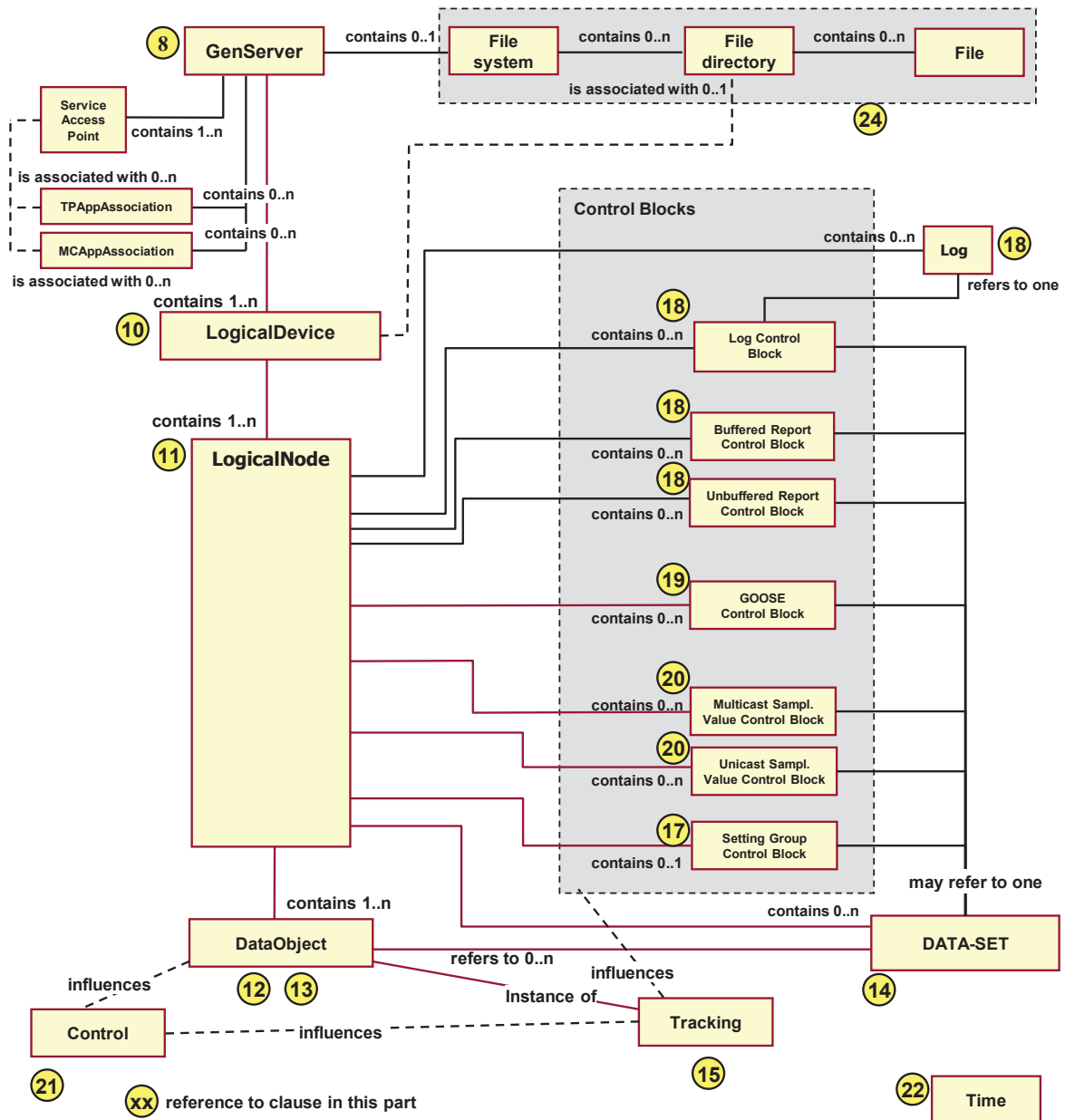


Figure 3 – Conceptual service model of the ACSI

NOTE 1 The numbers in the circles indicate the respective clauses in this document.

NOTE 2 The class diagrams are conceptual. Details are defined in the respective clauses. Comprehensive diagrams are contained in IEC 61850-7-1.

The logical node is one of the major building blocks that have associations to most of the other information exchange models, for example, report control, log control, and setting control. In this document, the generic logical node class is defined (**GenLogicalNode**).

NOTE 3 The class models and services are defined using an object-oriented approach allowing for the mapping of class models and services to different application layer and middle ware solutions.

The complete list of ACSI classes and their services is shown in Table 2.

Table 2 – ACSI model classes with related services

<u>GenServer model</u> GetServerDirectory <u>Association model</u> Associate Abort Release <u>GenLogicalDeviceClass model</u> GetLogicalDeviceDirectory <u>GenLogicalNodeClass model</u> GetLogicalNodeDirectory GetAllDataValues <u>GenDataObjectClass model</u> GetDataValues SetDataValues GetDataDirectory GetDataDefinition <u>DATA-SET model</u> GetDataSetValues SetDataSetValues CreateDataSet DeleteDataSet GetDataSetDirectory <u>SETTING-GROUP-CONTROL-BLOCK model</u> SelectActiveSG SelectEditSG SetEditSGValue ConfirmEditSGValues GetEditSGValue GetSGCBValues <u>REPORT-CONTROL-BLOCK and LOG-CONTROL-BLOCK model</u> BUFFERED-REPORT-CONTROL-BLOCK: Report ^a GetBRCBValues SetBRCBValues UNBUFFERED-REPORT-CONTROL-BLOCK:	LOG-CONTROL-BLOCK model: GetLCBValues SetLCBValues QueryLogByTime QueryLogAfter GetLogStatusValues <u>Generic substation event model – GSE</u> GOOSE SendGOOSEMessage ^a GetGoReference GetGOOSEElementNumber GetGoCBValues SetGoCBValues <u>Transmission of sampled values model</u> MULTICAST-SAMPLE-VALUE-CONTROL-BLOCK: SendMSVMessage ^a GetMSVCBValues SetMSVCBValues GetMsvReference GetMSVElementNumber UNICAST-SAMPLE-VALUE-CONTROL-BLOCK (deprecated): SendUSVMessage ^a GetUSVCBValues SetUSVCBValues <u>Control model</u> Select SelectWithValue Cancel Operate CommandTermination TimeActivatedOperate <u>Time and time synchronization</u> TimeSynchronization <u>FILE transfer model</u> GetFile SetFile DeleteFile
---	--

Report ^a	GetFileAttributeValues
GetURCBValues	
SetURCBValues	
^a All the services for spontaneous sending are limited to one access point per control block instance.	

NOTE 4 All services defined in this document operate on instances of classes only. The service GetDataValues for example operates on an instantiated data object class implemented in a real device. The service parameters in the abstract service tables and the definition of the services refer to those instances and not to the generic classes defined in this document.

5.3.4 Relations between classes

The crucial relations of the meta model classes are shown in Figure 4. The figure shows the crucial building rule for data objects (the recursions). The abstract model in the ACSI uses the generic common data class model to define any hierarchical model of domain specific information. The class diagram uses two recursions: the **GenCommonDataClass** and the **GenConstructedAttributeClass**. These two recursions allow the definition of any common data class defined in IEC 61850-7-3.

Figure 4 shows some examples of the definitions of IEC 61850-7-3 (common data classes **WYE**, **CMV**; attributes like **cVal**, etc.) and IEC 61850-7-4 (logical node **MMXU** and data object **Phv**). The various types, names and identifiers (IDs) are shown in coloured boxes with the part number indicated as “6” for IEC 61850-6, “7-3” for names and identifiers defined in IEC 61850-7-3, and “7-4” for logical node and data object names defined in IEC 61850-7-4.

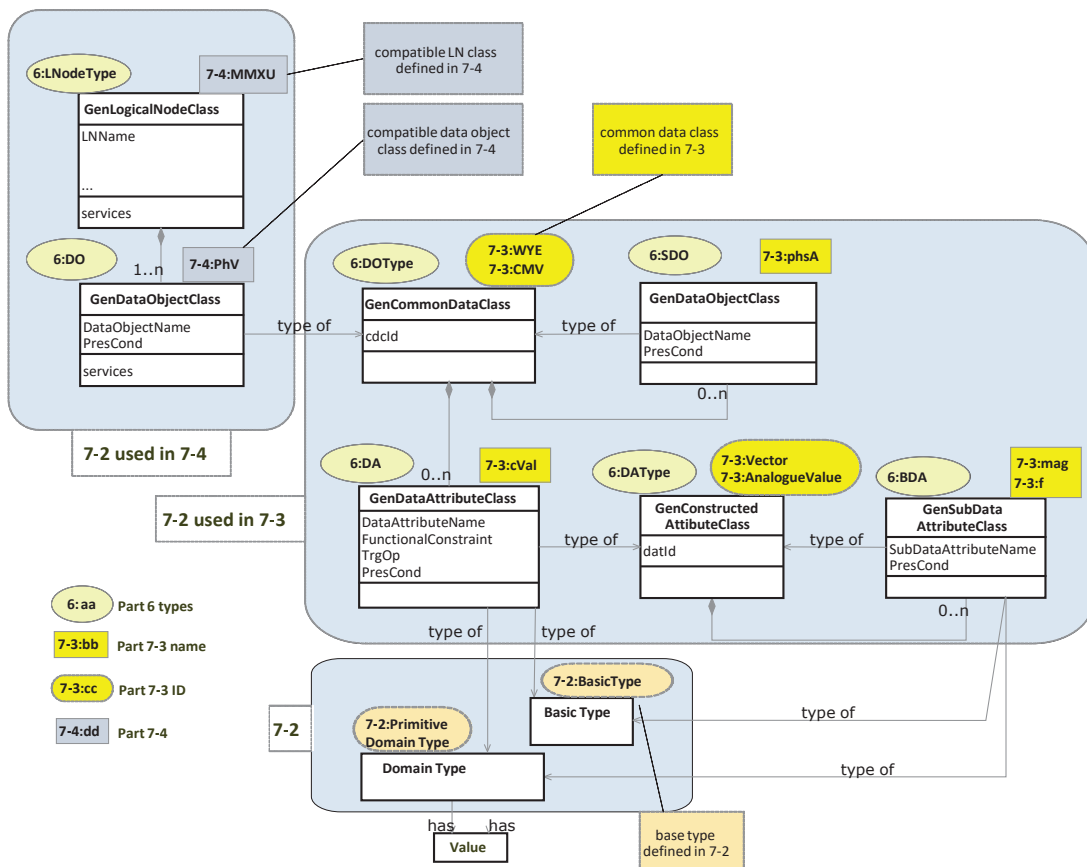


Figure 4 – Core of the conceptual meta model and relationship

The **GenCommonDataClass** is one of the crucial models needed to build the information models. The **GenCommonDataClass** model is used as a rule to define (build) common data classes (common for many domains like **SPC** for single point status or specific for a domain like **WYE** for electrical applications).

5.4 The domain type model

The domain type model of IEC 61850 defines lists of common data classes (**CDC**, IEC 61850-7-3), data objects (typed by common data classes) and logical node classes (IEC 61850-7-4) aggregating data objects. These classes are used to build data models for real IEDs.

IEC 61850-6 defines a method to describe the data instance model based on these classes, that can be used to describe the complete model implemented (programmed or configured) in a real IED. The data instance model is introduced in 0.

5.5 The data instance model

The data instance model describes instances of the classes defined in IEC 61850-7-x (see Figure 5). IEC 61850-6 defines by means of an XML schema (the SCL schema) a language to describe the configuration of IEDs. The SCL schema uses the element **DOType** to describe the common data class instantiation of a specific data object (**DO** element) in the logical node type (**LNType**). IEC 61850-6 defines an IED element that has logical devices (**LD**) composed of logical nodes (**LN**). The logical nodes are typed by instantiable **LNTypes** listed in the **DataTypeTemplate** section of an SCL document. Data objects in a **LNType** become a data object instance (**DOI**) in the corresponding logical node.

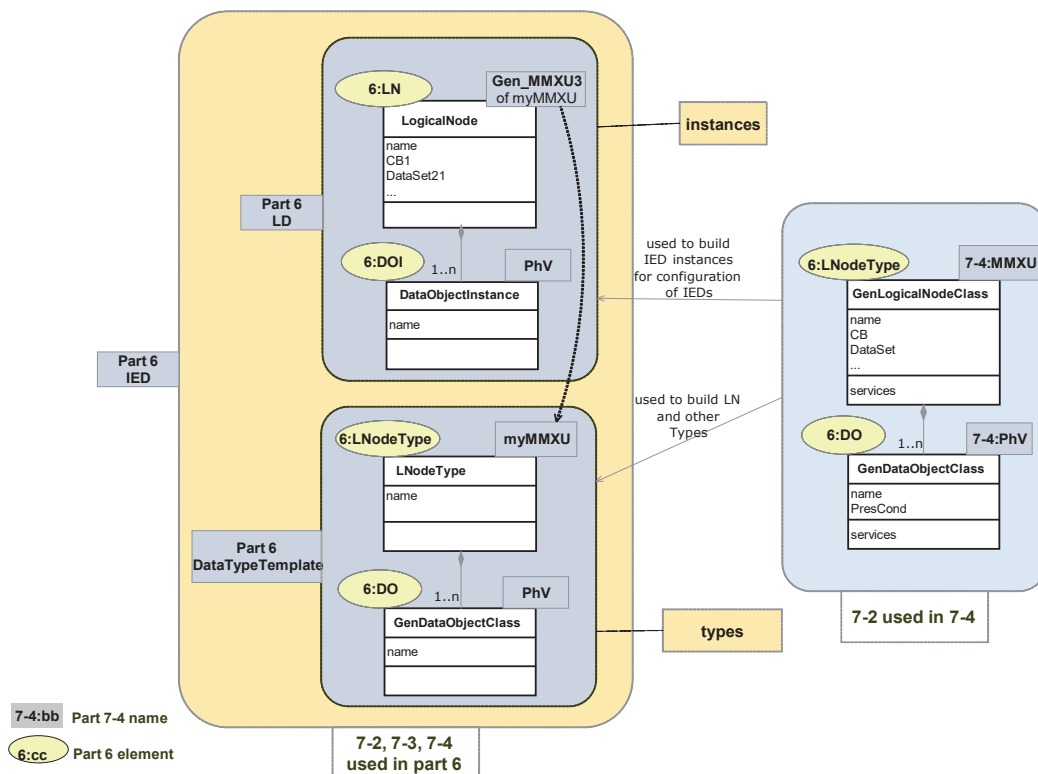


Figure 5 – Data instance model (conceptual)

NOTE The mapping of these instances to application layer protocols like MMS (manufacturing message specification, ISO 9506) are defined in the SCSMs. The logical device is mapped in MMS to a MMS domain class and the logical node and data objects are mapped to MMS NamedVariables as part of a domain.

5.6 Conditions for element presence

This subclause introduces conditions that specify the presence of elements in a given context (one LN, or one CDC, or one data attribute type, or one data object for dataNs). The name of the attribute type of presence is PresenceCondition.

The presence conditions are part of the Code Component of this document and are available as an electronic machine readable file in the related NSD file.

Table 3 shows the presence conditions.

Table 3 – Conditions for presence of elements within a context

Condition name	Definition
M	Element is mandatory.
O	Element is optional.
F	Element is forbidden.
na	Element is not applicable.
Mmulti	At least one element shall be present; all instances have an instance number > 0.
Omulti	Zero or more elements may be present; all instances have an instance number > 0.
AtLeastOne(n)	Parameter n: group number (> 0). At least one of marked elements of a group n shall be present.
AtMostOne	At most one of marked elements shall be present.
AllOrNonePerGroup(n)	Parameter n: group number (> 0). All or none of the elements of a group n shall be present.
AllOnlyOneGroup(n)	Parameter n: group number (> 0). All elements of only one group n shall be present.
AllAtLeastOneGroup(n)	Parameter n: group number (> 0). All elements of at least one group n shall be present.
MF(sibling)	Parameter sibling: sibling element name. Mandatory if sibling element is present, otherwise forbidden.
MO(sibling)	Parameter sibling: sibling element name. Mandatory if sibling element is present, otherwise optional.
OM(sibling)	Parameter sibling: sibling element name. Optional if sibling element is present, otherwise mandatory.
FM(sibling)	Parameter sibling: sibling element name. Forbidden if sibling element is present, otherwise mandatory.
MOcond(condID)	Parameter condID: condition number (> 0). Textual presence condition (non-machine processable) with reference condID to context specific text. If satisfied, the element is mandatory, otherwise optional.
MFcond(condID)	Parameter condID: condition number (> 0). Textual presence condition (non-machine processable) with reference condID to context specific text. If satisfied, the element is mandatory, otherwise forbidden.
OFcond(condID)	Parameter condID: condition number (> 0). Textual presence condition (non-machine processable) with reference condID to context specific text. If satisfied, the element is optional, otherwise forbidden.

Condition name	Definition
MmultiRange(min, max)	Parameters min, max: limits for instance number (> 0). One or more elements shall be present; all instances have an instance number within range [min, max] (see IEC 61850-7-1).
OmultiRange(min, max)	Parameters min, max: limits for instance number (> 0). Zero or more elements may be present; all instances have an instance number within range [min, max] (see IEC 61850-7-1).
MFsubst	Element is mandatory if substitution is supported (for substitution, see IEC 61850-7-3), otherwise forbidden.
MOIn0	Element is mandatory in the context of LLN0; otherwise optional.
MFIn0	Element is mandatory in the context of LLN0; otherwise forbidden.
MOInNs	Element is mandatory if the name space of its logical node deviates from the name space of the containing logical device, otherwise optional. See IEC 61850-7-1 for use of name space.
MOdataNs	Element is mandatory if the name space of its data object deviates from the name space of its logical node, otherwise optional. See IEC 61850-7-1 for use of name space.
MFscaledAV	Element is mandatory* if any sibling elements of type AnalogueValue include 'i' as a child, otherwise forbidden. *Even though devices without floating point capability cannot exchange floating point values through ACSI services, the description of scaling remains mandatory for their (SCL) configuration.
MFscaledMagV	Element is mandatory* if any sibling elements of type Vector include 'i' as a child of their 'mag' attribute, otherwise forbidden. *See MFscaledAV.
MFscaledAngV	Element is mandatory* if any sibling elements of type Vector include 'i' as a child of their 'ang' attribute, otherwise forbidden. *See MFscaledAV.
MOrms	Element is mandatory if the harmonic values in the context are calculated as a ratio to RMS value (value of data attribute 'hvRef' is 'rms'), optional otherwise.
MOrootLD	Element is mandatory in the context of a root logical device; otherwise it is optional.
MOoperTm	Element is mandatory if at least one controlled object on the IED supports time activation service; otherwise it is optional.
MmultiF(sibling)	Parameter sibling: sibling element name. One or more elements must be present if sibling element is present, otherwise forbidden.
MOsbo	Element is mandatory if declared control model supports 'sbo-with-normal-security' or 'sbo-with-enhanced-security', otherwise optional and value is of no impact.
MOenhanced	Element is mandatory if declared control model supports 'direct-with-enhanced-security' or 'sbo-with-enhanced-security', otherwise optional and value is of no impact.
MONamPIt	Element is mandatory if the name space of its logical node deviates from the name space of the containing logical device, otherwise optional. See IEC 61850-7-1 for use of name space.
OF(sibling)	Parameter sibling: sibling element name. Optional if sibling element is present, otherwise forbidden.
MORange	Element is mandatory if the measured value associated (amplitude respectively angle) exposes the range eventing (with the attribute range respectively rangeAng).
OMSynPh	This attribute is optional if value of 'phsRef' is Synchrophasor otherwise Mandatory.
MAIOrNonePerGroup(n)	Parameter n: group number (> 0). Element is mandatory if declared control model supports 'direct-with-enhanced-security' or 'sbo-

Condition name	Definition
	with-enhanced-security', otherwise all or none of the elements of a group n shall be present.

5.7 Functional constraints (FcKind)

Functional constraints indicates services applicable to a specific data attribute.

From an application point of view, functional constraints classify data attributes according to their specific use, e.g. status information, measurement, setting or description.

FC serve as a data filter in the sense of defining the services applicable to specific data attributes of common data classes (defined in IEC 61850-7-3).

Functional constraints and applicable services are part of the Code Component of this IEC standard and are available as electronic machine readable file in related NSD file.

NOTE The possibility to access a data attribute can be further constrained by a view, access control or an implementation.

EXAMPLE The common data class single point status (SPS) according to IEC 61850-7-3 has the following data attributes related to process state: stVal (status value), q (quality), and t (time stamp) with the functional constraint ST (status information). The write service is not allowed for these attributes.

Table 4 shows all functional constraints.

Table 4 – Functional constraints (FcKind)

Functional constraints (FcKind)		
FC	Semantic	Description (services allowed, initial values, storage)
ST	Status information	Data attribute shall represent status information. Initial value shall be taken from the process. Modelling note: Applicable ACSI services: – GetDataValues – GetDataDefinition – GetDataDirectory – GetDataSetValues – GetAllDataValues – may be a DataSetMember of a DataSet referred to by any of: GOOSE control block, report control block, log control block, sampled value control block.
MX	Measurands (analogue values)	Data attribute shall represent measurand information. Initial value shall be taken from the process. Modelling note: Applicable ACSI services: – GetDataValues – GetDataDefinition – GetDataDirectory – GetDataSetValues – GetAllDataValues

Functional constraints (FcKind)		
FC	Semantic	Description (services allowed, initial values, storage)
		– may be a DataSetMember of a DataSet referred to by any of: GOOSE control block, report control block, log control block, sampled value control block.
SP	Setting (outside setting group)	Data attribute shall represent setting parameter information. Initial value shall be as configured; value shall be non-volatile. Modelling note: Applicable ACSI services: – GetDataValues – SetDataValues – GetDataDefinition – GetDataDirectory – GetDataSetValues – SetDataSetValues – GetAllDataValues – may be a DataSetMember of a DataSet referred to by any of: GOOSE control block, report control block, log control block.
SV	Substitution	Data attribute shall be used to handle substitution (see IEC 61850-7-3). Initial behaviour of substitution shall be substitution disabled. If the substitution handling relies on non-volatile DataAttributes, then the behaviour of substitution at restart of the IED shall be as set before the restart. Modelling note: Applicable ACSI services: – GetDataValues – SetDataValues – GetDataDefinition – GetDataDirectory – GetDataSetValues – SetDataSetValues – GetAllDataValues – may be a DataSetMember of a DataSet referred to by any of: report control block, log control block.
CF	Configuration	Data attribute shall represent configuration information. Initial value shall be as configured; value shall be non-volatile. Modelling note: Applicable ACSI services: – GetDataValues – SetDataValues – GetDataDefinition – GetDataDirectory – GetDataSetValues – SetDataSetValues – GetAllDataValues

Functional constraints (FcKind)		
FC	Semantic	Description (services allowed, initial values, storage)
		– may be a DataSetMember of a DataSet referred to by any of: report control block, log control block.
DC	Description	Data attribute shall represent description (intended for humans) information. Initial value shall be as configured; value shall be non-volatile. Modelling note: Applicable ACSI services: – GetDataValues – SetDataValues – GetDataDefinition – GetDataDirectory – GetDataSetValues – SetDataSetValues – GetAllDataValues – may be a DataSetMember of a DataSet referred to by any of: report control block, log control block.
SG	Setting group	Data attribute shall represent the current active value of a setting member of a setting. See SETTING GROUP CONTROL BLOCK model. Initial value shall be as configured; value shall be non-volatile. Modelling note: Applicable ACSI services: – GetDataValues – GetDataDefinition – GetDataDirectory – GetDataSetValues – GetAllDataValues – may be a DataSetMember of a DataSet referred to by any of: report control block, log control block.
SE	Setting group editable	Data attribute shall belong to the editing services associated to a setting group. See SETTING GROUP CONTROL BLOCK model. Modelling note: Applicable ACSI services: – GetDataDefinition – GetDataDirectory – GetEditSGValues – SetEditSGValues.
SR	Service response	Data attribute shall represent data from different process objects with the same tracking object. These attributes are used for service tracking. Initial value of the data attribute is a private issue, e.g., all zero. Modelling note: Applicable ACSI services: – GetDataValues – GetDataDefinition – GetDataDirectory

Functional constraints (FcKind)		
FC	Semantic	Description (services allowed, initial values, storage)
		– GetDataSetValues – GetAllDataValues – may be a DataSetMember of a DataSet referred to by any of: report control block, log control block.
OR	Operate received	Data attribute shall represent the result of an Operate request at the data object receiving the Operate request, even if the execution of the Operate is blocked. Initial value is irrelevant / arbitrary. Modelling note: Applicable ACSI services: – GetDataValues – GetDataDefinition – GetDataDirectory – GetDataSetValues – GetAllDataValues – may be a DataSetMember of a DataSet referred to by any of: GOOSE control block, report control block, log control block.
BL	Blocking	Data attribute shall be used for blocking value updates. If the value of the data attribute is volatile then the initial value shall be false, otherwise the value should be as set or configured. Modelling note: Applicable ACSI services: – GetDataValues – SetDataValues – GetDataDefinition – GetDataDirectory – GetDataSetValues – SetDataSetValues – GetAllDataValues – may be a DataSetMember of a DataSet referred to by any of: report control block, log control block.
EX	Extended definition (application name space)	Data attribute shall represent an application name space. See IEC 61850-7-1. Value of the data attribute shall be as configured; value shall be non-volatile. Modelling note: Applicable ACSI services: – GetDataValues – GetDataDefinition – GetDataDirectory – GetDataSetValues – GetAllDataValues – may be a DataSetMember of a DataSet referred to by any of: report control block, log control block.

5.8 Trigger options

To support reporting and logging control logic (see Clause 17) related to process data changes, one of these values can be associated as a trigger option, TrgOp, to a data attribute within a data object or sub-data object.

Table 5 shows all trigger options.

Table 5 – Trigger options

Trigger options		
TrgOp	Semantic	Description
dchg	data-change	The change of the value of the associated data attribute can trigger the generation of a report or a log entry.
qchg	quality-change	The change of the value of the quality data attribute can trigger the generation of a report or a log entry.
dupd	data-update	The update of the value of the associated data attribute can trigger the generation of a report or a log entry. The updated value may be the same as the old value. An example is freezing the value of a freezable data attribute updating the value of another data attribute, which could lead to the same value it already has. Another example would be the distance to a fault that would persist, or a calculation of a minimum value over a period and the same minimum is reached in two consecutive statistical calculation windows.
dchg, dupd	dchg or dupd	Either 'dchg' or 'dupd', to be chosen at data object definition in the scope of a logical node, or at implementation.

6 Type definition

6.1 General

This subclause introduces the type concept used within IEC 61850-7 series. An overview of the types is provided in Table 6.

Table 6 – Type definition overview

IEC 61850-7-2 type definition	Reference to the type	clause of 61850-7-2	Example
Basic Type	<type name>	6.2.2	INT16
Domain Type			
Abstract	<type name>	6.2.3	PhyComAddr
Non-structured	<type name>	6.2.3, 6.3.3	ObjectReference
Enumerations	<type name>	6.2.4, 6.3.4	ServiceNameKind
<i>Coded enum</i>	<type name>	6.2.5	StepControlKind
Structured	<type name>	6.2.3	Origin
<i>Packed List</i>	<type name>	6.2.3, 6.3.3	Timestamp
Arrays	ARRAY 0..m OF p		ARRAY 0...maxPts-1 OF INT32

On a first level, IEC 61850 differentiates between basic types and domain types.

- Basic types are non-structured and non-domain specific types as they are typically known by any compiler (e.g. integer 16, unsigned integer 8).
- Domain types can be non-structured as well as structured types that have a domain specific semantic.

Among the domain types, we have the following classifications:

- Abstract types: types that cannot be fully defined in part 7 of IEC 61850 because they are SCSM dependent.
- Non structured domain types: types that have a domain specific semantic as compared to basic types.
- Enumeration types: types that are an ordered set of values where custom extensions are allowed. Each value is associated to an enumeration item, which is limited to a length of 127 characters and to a character set of Basic-Latin and Latin-1-supplement. The naming of the Enumeration types has been harmonized over all IEC 61850-7-x parts to follow the naming scheme <EnumTitle>Kind. The renaming is fully backward compatible since the value of the literal have not changed. The name of the enumeration type is not a standardized name that shall be used by the implementation, while the supported positive value of the enumeration items shall be used by the implementation. The following rules apply for the extensions:
 - the positive range of values is reserved for standardized value of enumerations, except for the IEC 61850-7-3 multiplierKind that standardizes also values in the negative range,
 - the extension of the enumeration type can be done in the negative range of the values, only if none of the standardized value is providing the proper semantic,
 - private enumeration types can be done both in the positive and in the negative range of the values,
 - an implementation can decide to implement/support only part of the standardized set of the enumeration.
- Coded enum types: variant of the enumeration types where custom extensions are not allowed and that shall be mapped on an efficient coding in the SCSM. The naming of the coded enum types has been harmonized over all IEC 61850-7-x parts to follow the naming scheme <CodedEnumTitle>Kind. The renaming is fully backward compatible since the value of the literal have not changed.
- Structured types: types composed of components that themselves can be of any type.
- Packed list types: variant of structured types where the components shall be mapped on an efficient coding in the SCSM and no individual access to the components shall be possible. The components can be of any type other than arrays.
- Array types: types defined as ARRAY 0..m OF p and are a list of elements of type “p” numbered from 0 to m.

Beyond the above concept, this clause first groups the type definitions according to the usage:

- Clause 6.2 “Data model types” describes the types that are used by the data model;
- Clause 6.3 “ACSI service parameter types” describes the types that not used by the data model but used by the parameters of ACSI services.

Beyond the types defined in this clause of IEC 61850-7-2, further domain types that are not used within IEC 61850-7-2 are defined in other parts of IEC 61850:

- IEC 61850-7-3 defines additional enumeration types as “Enumerated data attribute types” in Clause 8.
- IEC 61850-7-3 defines additional structured types as “Constructed attribute classes” in Clause 6.
- IEC 61850-7-4xx defines additional enumerations that are used in the common data classes ENC, ENG and ENS as “Enumerated data attribute types”. Within the specification of the common data class in IEC 61850-7-3, for these types the placeholder “EnumDA” is used. In IEC 61850-7-4xx, reference to these types is made through the name of the common data class and the type name in parenthesis (e.g. **ENS (HealthKind)**).

The SCSM (IEC 61850-8-x, IEC 61850-9-x) define the details of the abstract types.

Arrays are used within the specification of the common data classes in IEC 61850-7-3. For the arrays, the following constraints apply:

- $m \geq 0$
- p is of one of the following:
 - any type except an **ARRAY**
 - a common data class that contains no arrays

The above type categories from the data model are referred to in IEC 61850-6 as bTypes as shown in Table 7.

Table 7 – Use of IEC 61850-6 bType

IEC 61850-7-2 Type definition	bType used in IEC 61850-6
Basic type	Specific bType (e.g. INT8)
Abstract type	Specific bType (e.g. PhyComAddr)
Non-structured domain type	Specific bType (e.g. ObjRef)
Enumerations	Generic bType Enum
Coded enum type	Specific bType (e.g. DbPos)
Structured types	Generic bType Struct
Packed list type	Specific bType (e.g. Timestamp)
Arrays	With the bType of the element and the number of members in count

The different types are part of the Code Component of this IEC standard and are available as an electronic machine readable file in the related NSD file.

6.2 Data model types

6.2.1 General

This subclause defines core types, mainly primitive, common to the IEC 61850 both data and service models.

6.2.2 Basic types

6.2.2.1 General

This subclause defines basic types common to the IEC 61850 both data and service models.

NOTE The basic types: VisString65 and INT128 used in the first edition of this standard have been removed.

6.2.2.2 BOOLEAN (P_BOOLEAN basic)

Boolean: {false, true}.

6.2.2.3 INT8 (P_INT8 basic)

Signed integer: [-128, 127].

6.2.2.4 INT16 (P_INT16 basic)

Signed integer: [-32 768, 32 767].

6.2.2.5 INT32 (P_INT32 basic)

Signed integer: [-2 147 483 648, 2 147 483 647].

6.2.2.6 INT64 (P_INT64 basic)

Signed integer: $[-2^{63}, (2^{63})-1]$.

6.2.2.7 INT8U (P_INT8U basic)

Unsigned integer: [0, 255].

6.2.2.8 INT16U (P_INT16U basic)

Unsigned integer: [0, 65 535].

6.2.2.9 INT24U (P_INT24U basic)

Unsigned integer: [0, 16 777 215].

Only used for TimeStamp type.

6.2.2.10 INT32U (P_INT32U basic)

Unsigned integer: [0, 4 294 967 295].

6.2.2.11 FLOAT32 (P_FLOAT32 basic)

Single-precision floating point according to IEEE 754).

6.2.2.12 Octet64 (P_Octet64 basic)

(old name = OCTET_STRING64) Should be able to hold up to 64 bytes. NULL string has length 0.

6.2.2.13 VisString64 (P_VisString64 basic)

(old name = VISIBLE_STRING64) Should be able to hold up to 64 characters. NULL string has length 0.

6.2.2.14 VisString129 (P_VisString129 basic)

(old name = VISIBLE_STRING129) Should be able to hold up to 129 characters. NULL string has length 0.

6.2.2.15 VisString255 (P_VisString255 basic)

(old name = VISIBLE_STRING255) Should be able to hold up to 255 characters. NULL string has length 0.

6.2.2.16 Unicode255 (P_Unicode255 basic)

(old name = UNICODE_STRING255) Should be able to hold up to 255 Unicode characters. NULL string has length 0.

6.2.3 Domain types (general)

6.2.3.1 General

This subclause defines general domain types common to the IEC 61850 both data and service models.

6.2.3.2 PhyComAddr (P_PhyComAddr basic)

(old name = PHYCOMADDR) Type used for physical communication address (e.g. media access address, priority, and other information) as defined by a SCSM.

6.2.3.3 ObjectName (ObjectName basic)

Type used for unique instance name among instances of a class owned by the same parent.

This type is a case sensitive VisString64, starts with a letter and is constrained to the following character set:

- upper and lower case 26 letter of US English ("A"|"a"|"B"|"b"|...|"Y"|"y"|"Z"|"z")
- numbers ("0"|"1"|"2"|"3"|"4"|"5"|"6"|"7"|"8"|"9"), and,
- underscore ("_").

6.2.3.4 ObjectReference (P_ObjectReference basic)

Instances of classes in the hierarchical information model (ACSI class hierarchy of logical device, logical node, data, data attributes) shall be constructed by the concatenation of all instance names comprising the whole path-name of an instance of a class that identifies the instance uniquely.

This type is a case sensitive VisString129 and follows the character set of the ObjectName extended with the "/", "\$", ".", "(", ")", "@".

The syntax of an object reference within the scope of a logical device shall be: LDName/LNName[.Name[...]], where the "/" shall separate the LDName from the rest of the object reference, and the "." shall separate the further names in the hierarchy.

- The "[.]" indicates an option.
- The "[. ...]" indicates further names of recursively nested definitions.
- The "(...)" shall indicate an array element.

The NULL ObjectReference is an empty ObjectReference whose length is zero(0).

The syntax within the scope of a TPAA shall be: @ObjectName.

The constraints defined in Clause 22 on the use of this type shall be applied.

When this type is used in SCL, there is an additional possibility to represent references relative to the IED (instead of the full references). However the appropriate on-line values shall always be absolute references.

NOTE The first edition of this standard defined the ObjectReference as a VisString65; this basic type has been removed as the VisString129 replaces it.

6.2.3.5 EntryID (P_EntryID basic)

An arbitrary octet string used to identify an entry in a sequence of events such as a log or a buffered report as specified by an SCSM.

NOTE The EntryID (handle) allows a client to re-synchronize, for example, with the sequence of the events stored in the IED. The syntax of the value is a local issue outside the scope of this standard. However the NULL EntryID used in the standard must be the octet string whose octets have all the value 0 (zero) and is reserved to indicate an unassigned ID.

6.2.3.6 Currency (P_Currency basic)

A currency identification code based on ISO 4217 3-character currency code. The NULL currency is represented by the 3-characters "XXX". The concrete coding shall be defined by the SCSMs.

6.2.3.7 Timestamp (P_Timestamp packed list)

(old name = TimeStamp) UTC time with the epoch of midnight (00:00:00) of 1970-01-01. The presentation is defined in the SCSMs.

The NULL time stamp has all fields set to 0 (zero).

The relation between a timestamp value, the synchronization of an internal time with an external time source (for example, UTC or TAI time), and other information related to time model are available as requirements in Clause 21.

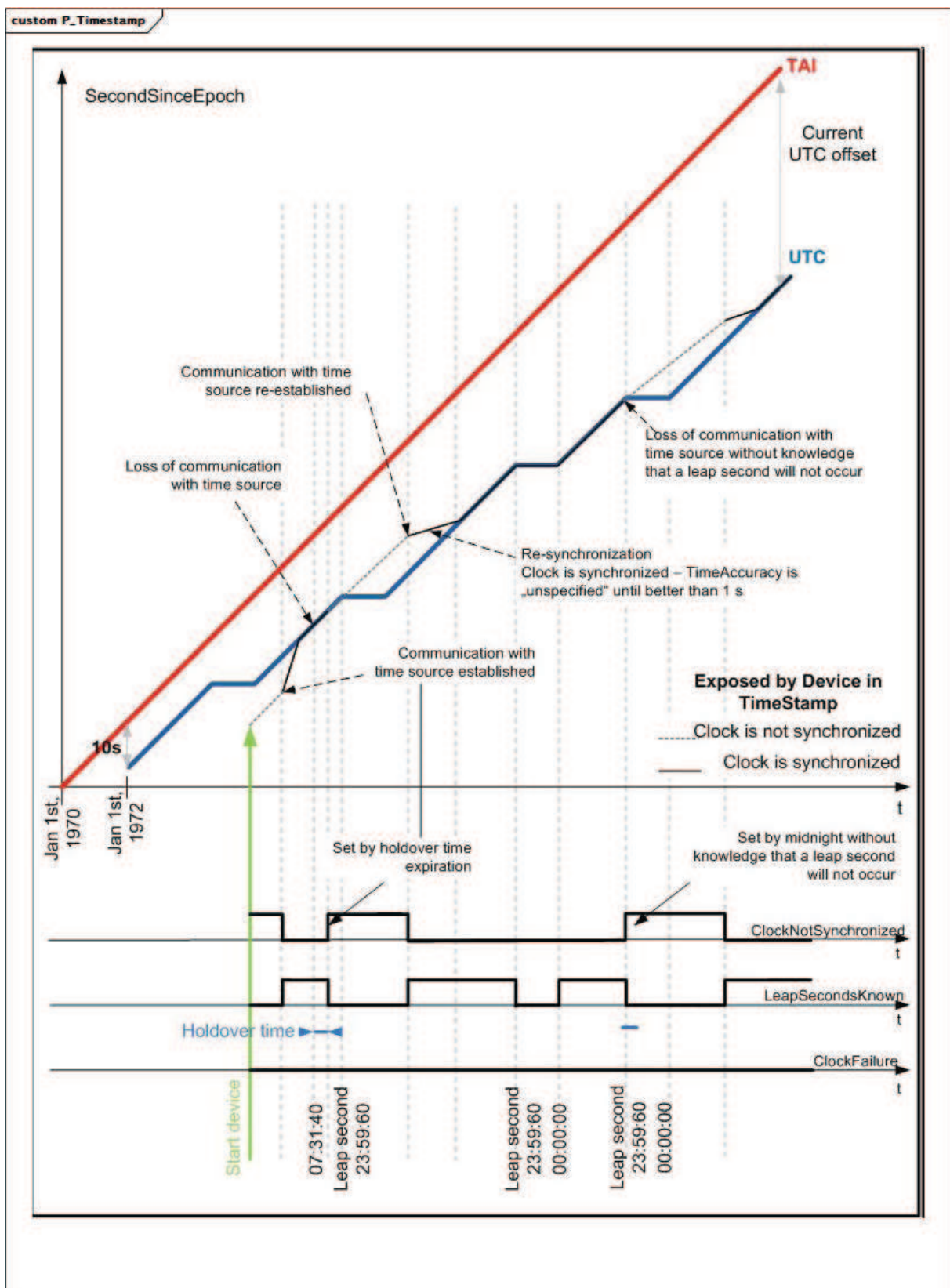


Figure 6 – P_Timestamp

Figure 6: This diagram illustrates the values of the different attributes of a Timestamp over time when the synchronization with the external time source is lost, when a positive leap second occurs during the

communication outage with the external time source or when a positive leap second occurs while the internal clock is connected to the external time source.

Table 8 shows all attributes of P_Timestamp.

Table 8 – Attributes of P_Timestamp

datId = P_Timestamp			
Attribute name	Attribute type	(Value/Value range) Description	PresCond
SecondSinceEpoch	INT32U	Interval in seconds counted from the epoch 1970-01-01 00:00:00 UTC.	M
FractionOfSecond	INT24U	Fraction of the current second when the value of time stamp has been determined. It shall be calculated as (SUM from $i = 0$ to 23 of $bi \cdot 2^{23-i}$) [s]. NOTE 1 The resolution is the smallest unit by which the time stamp is updated. The 24 bits of the integer provide 1 out of 16 777 216 counts as the smallest unit; calculated by $1/2^{24}$ which equals approximately 60 ns. NOTE 2 The resolution of a time stamp may be $1/2^{*1}$ (= 0.5 s) if only the first bit is used; or may be $1/2^{*2}$ (= 0.25 s) if the first two bits are used; or may be approximately 60 ns if all 24 bits are used. The resolution provided by an IED is outside the scope of this standard and an internal clock that is not capable of providing 24 bits of resolution will limit the range of the supported TimeAccuracy. Example: a time stamping function using a resolution of 11 bits cannot expose a TimeAccuracy higher than 10. A time stamping function using a resolution of 24 bits can expose any TimeAccuracy.	M
TimeQuality	TimeQuality	Information about the quality of the time source of the sending IED.	M

6.2.3.8 TimeQuality (TimeQuality packed list)

Information about the quality of the time source of the sending IED.