

DIN EN 61850-6/A1

ICS 33.040.40

Einsprüche bis 2018-06-27
Vorgesehen als Änderung von
DIN EN 61850-6:2010-06

Entwurf

**Kommunikationsnetze und -systeme für die Automatisierung in der elektrischen Energieversorgung –
Teil 6: Sprache für die Beschreibung der Konfiguration für die Kommunikation in Stationen mit intelligenten elektronischen Geräten (IED)
(IEC 57/1918/FDIS:2017);
Englische Fassung EN 61850-6:2010/FprA1:2017**

Communication networks and systems for power utility automation –
Part 6: Configuration description language for communication in power utility automation systems related to IEDs
(IEC 57/1918/FDIS:2017);
English version EN 61850-6:2010/FprA1:2017

Réseaux et systèmes de communication pour l'automatisation des systèmes électriques –
Partie 6: Langage pour la description de configuration pour la communication dans les systèmes électriques d'automation, entre les dispositifs électroniques intelligents (IED)
(IEC 57/1918/FDIS:2017);
Version anglaise EN 61850-6:2010/FprA1:2017

Anwendungswarnvermerk

Dieser Norm-Entwurf mit Erscheinungsdatum 2018-04-27 wird der Öffentlichkeit zur Prüfung und Stellungnahme vorgelegt.

Weil die beabsichtigte Norm von der vorliegenden Fassung abweichen kann, ist die Anwendung dieses Entwurfs besonders zu vereinbaren.

Stellungnahmen werden erbeten

- vorzugsweise online im Norm-Entwurfs-Portal von DIN unter www.din.de/go/entwuerfe bzw. für Norm-Entwürfe der DKE auch im Norm-Entwurfs-Portal der DKE unter www.entwuerfe.normenbibliothek.de, sofern dort wiedergegeben;
- oder als Datei per E-Mail an dke@vde.com möglichst in Form einer Tabelle. Die Vorlage dieser Tabelle kann im Internet unter www.din.de/go/stellungnahmen-norm-entwuerfe oder für Stellungnahmen zu Norm-Entwürfen der DKE unter www.dke.de/stellungnahme abgerufen werden;
- oder in Papierform an die DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE, Stresemannallee 15, 60596 Frankfurt am Main.

Die Empfänger dieses Norm-Entwurfs werden gebeten, mit ihren Kommentaren jegliche relevanten Patentrechte, die sie kennen, mitzuteilen und unterstützende Dokumentationen zur Verfügung zu stellen.

Gesamtumfang 143 Seiten

DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE

Nationales Vorwort

Die Englische Fassung des europäischen Dokuments EN 61850-6:2010/FprA1:2017 „Kommunikationsnetze und -systeme für die Automatisierung in der elektrischen Energieversorgung – Teil 6: Sprache für die Beschreibung der Konfiguration für die Kommunikation in Stationen mit intelligenten elektronischen Geräten (IED)“, (Schluss-Entwurf) ist unverändert in diesen Norm-Entwurf übernommen worden.

Die Internationale Elektrotechnische Kommission (IEC) und das Europäische Komitee für Elektrotechnische Normung (CENELEC) haben vereinbart, dass ein auf IEC-Ebene erarbeiteter Entwurf für eine Internationale Norm zeitgleich (parallel) bei IEC und CENELEC zur Abstimmung als FDIS (en: Final Draft International Standard) bzw. Schluss-Entwurf für eine Europäische Norm gestellt wird, um eine Beschleunigung und Straffung der Normungsarbeit zu erreichen. Dem entsprechend ist das internationale Dokument IEC 57/1918/FDIS:2017 „Communication networks and systems for power utility automation – Part 6: Configuration description language for communication in power utility automation systems related to IEDs“ unverändert in den Schluss-Entwurf EN 61850-6:2010/FprA1:2017 übernommen worden.

Das internationale Dokument wurde vom TC 57 „Power systems management and associated information exchange“ der Internationalen Elektrotechnischen Kommission (IEC) erarbeitet.

Dokumente, die bei CENELEC als Europäische Norm angenommen und ratifiziert werden, sind unverändert als Deutsche Normen zu übernehmen.

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

Für den Fall einer undatierten Verweisung im normativen Text (Verweisung auf ein Dokument ohne Angabe des Ausgabedatums und ohne Hinweis auf eine Abschnitsnummer, 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.

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 den vorliegenden Normentwurf erfüllt.

Da sich die Benutzer des vorliegenden Normentwurfs der englischen Sprache als Fachsprache bedienen, wird die Englische Fassung der EN 61850-6 veröffentlicht. Zu deren Abschnitt 1, der den Anwendungsbereich festlegt, wurde eine Übersetzung angefertigt und als informativer Nationaler Anhang NA des vorliegenden Normentwurfs hinzugefügt. Für viele der verwendeten Begriffe existieren keine gebräuchlichen deutschen Benennungen, da sich die deutschen Anwender in der Regel ebenfalls der englischen Benennungen bedienen. Dieser Normentwurf steht nicht in unmittelbarem Zusammenhang mit Rechtsvorschriften und ist nicht als Sicherheitsnorm anzusehen.

Es ist vorgesehen, auch bei der entsprechenden zukünftigen Deutschen Norm auf die deutsche Sprachfassung zu verzichten und diese in der englischsprachigen Fassung zu veröffentlichen.

Es wird auf die Möglichkeit hingewiesen, dass einige Elemente dieses Dokuments Patentrechte berühren können. IEC ist nicht dafür verantwortlich, einige oder alle diesbezüglichen Patentrechte zu identifizieren.

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.

Kommentare fließen in die Überarbeitung der nächsten Edition ein.

Nationaler Anhang NA (informativ)

Auszugsweise Übersetzung^{N1)}

1 Anwendungsbereich

Der Text des Anwendungsbereichs ist durch den folgenden neuen Text zu ersetzen:

1.1 Allgemeines

Dieser Teil der IEC 61850 legt ein Dateiformat zur Beschreibung kommunikationsbezogener IED-Konfigurationen (en: Intelligent Electronic Device, de: intelligentes elektronisches Gerät) und IED-Parameter, Konfigurationen von Kommunikationssystemen, Strukturen von Schaltanlagen(-funktionen) und deren Beziehungen untereinander fest. Dieses Format soll hauptsächlich dem Zweck dienen, IED-Funktionsbeschreibungen (en: IED Capability Description) und SA-Systembeschreibungen zwischen IED-Engineeringtools und den System-Engineeringtools verschiedener Hersteller kompatibel auszutauschen.

Die definierte Sprache wird als „System Configuration Description Language“ (SCL, de: Sprache zur Beschreibung von Systemkonfigurationen) bezeichnet. Das IED- und Kommunikationssystemmodell in SCL entspricht IEC 61850-5 und IEC 61850-7-x. In den entsprechenden Teilen können SCSM-spezifische Erweiterungen oder Anwendungsregeln erforderlich sein.

Die Konfigurationssprache beruht auf XML (en: Extensible Markup Language), Version 1.0 (siehe XML-Verweisungen in Abschnitt 2).

Diese Norm legt weder einzelne Implementierungen oder Produkte fest, die diese Sprache verwenden, noch enthält sie Beschränkungen hinsichtlich der Implementierung von Einheiten und Schnittstellen innerhalb eines Computersystems. Dieser Teil der Norm legt nicht das Format zum Herunterladen von Konfigurationsdaten auf ein IED fest, obwohl er für einen Teil der Konfigurationsdaten verwendet werden könnte.

1.2 Name und Version des Namensraums

Dieser neue Abschnitt ist für alle Namensräume nach IEC 61850 obligatorisch (wie in IEC 61850-7-1:2011 festgelegt).

Die Parameter zur Identifikation dieser neuen Freigabe des SCL-Namensraums `xmlns:scl="http://www.iec.ch/61850/2003/SCL"` sind:

- Version des Namensraums: 2007
- Überarbeitung des Namensraums: B
- Freigabe des Namensraums: 4
- Freigabedatum des Namensraums: 2017/07/05

Ausgabe	Datum der Veröffentlichung	Onlineshop	Namensraum
Ausgabe 1.0	2004-03	IEC 61850-6:2004	IEC 61850-6:2003
Ausgabe 2.0	2009-12	IEC 61850-6:2009	IEC 61850-6:2007B
Änderung 1 von Ausgabe 2.0	2017-??	IEC 61850-6:2009/AMD1:2017	IEC 61850-6:2007B4
Ausgabe 2.1	2017-??	IEC 61850-6:2009+AMD1:2017 CSV	IEC 61850-6:2007B4

^{N1)} Nationale Fußnote: Die Nummerierung aus IEC 57/1918/FDIS:2017-09 wurde beibehalten.

1.3 Verbreitung der Code-Komponenten

Die in dieser IEC-Norm enthaltenen Code-Komponenten sind auch als maschinenlesbare Datei unter folgender Adresse erhältlich:

<http://www.iec.ch/public/TC57/supportdocuments/IEC 61850-6.2015.SCL.2007B4.full.zip>

Die in dieser IEC-Norm enthaltene(n) Code-Komponente(n) wird (werden) möglicherweise derzeit überarbeitet, sodass Anwender die neueste Freigabe in der Datenbank unter folgender Adresse auswählen müssen: <http://www.iec.ch/TC57/supportdocuments>

Die neueste Version/Freigabe des Dokuments wird durch Auswahl der Datei „IEC 61850-6.2015.SCL.{VersionStateInfo}.full.zip“ gefunden, wobei das Feld „VersionStateInfo“ den höchsten Wert enthält.

Jede Code-Komponente ist als ZIP-Paket verfügbar und enthält die elektronische Darstellung der Code-Komponente selbst sowie eine Datei, die den Inhalt des Pakets beschreibt („IECManifest.xml“).

„IECManifest.xml“ enthält verschiedene Abschnitte mit Informationen zu folgenden Punkten:

- Urheberrechtsvermerk;
- Identifikation der Code-Komponente;
- Veröffentlichung im Zusammenhang mit der Code-Komponente;
- Liste der elektronischen Dateien, die die Code-Komponente bilden;
- optionale Liste der historischen Dateien zur Nachverfolgung der Änderungen während des Entwicklungsprozesses der Code-Komponente.

Die zu dieser Veröffentlichung gehörende „IECManifest“-Datei ist folgende:

```
<IECManifest xmlns="http://www.iec.ch/CC/2017/IECManifest" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.iec.ch/CC/2017/IECManifest IECManifest.xsd">
  <Copyright>
    <Notice>
      COPYRIGHT (c) IEC, 2017. This version of this XSD is part of IEC 61850-6:2009/AMD1:2016; see
the IEC 61850-6:2009/AMD1:2016 for full legal notices. In case of any differences between the here-below code and the IEC
published content, the here-below definition supersedes the IEC publication; it may contain updates. See history files. The whole
document has to be taken into account to have a full description of this code component.
      See www.iec.ch/CCv1 for copyright details.
    </Notice>
    <License uri="www.iec.ch/CCv1">IEC License</License>
  </Copyright>
  <CodeComponent id="IEC 61850-6.2007B4.SCL.XSD" name="IEC 61850-6 SCL schema V2007B4" content="full"
date="2017-07-05">
    <Publication name="IEC 61850-6.2015_ed2.1" comment="Configuration description language for communication
in power utility automation systems related to IEDs"/>
    <File name="SCL.xsd" category="normative" content="full"/>
    <File name="SCL_Substation.xsd" category="normative" content="full"/>
    <File name="SCL_Communication.xsd" category="normative" content="full"/>
    <File name="SCL_IED.xsd" category="normative" content="full"/>
    <File name="SCL_DataTypeTemplates.xsd" category="normative" content="full"/>
    <File name="SCL_BaseTypes.xsd" category="normative" content="full"/>
    <File name="SCL_BaseSimpleTypes.xsd" category="normative" content="full"/>
    <File name="SCL_Enums.xsd" category="normative" content="full"/>
    <File name="SCL.Doc.HTML.zip" category="normative" content="full" comment="Zip archive containing the HTML
documentation of the SCL. Contains the 'SCL.html' file and all related pictures"/>
    <HistoryFile name="history.2007B1.txt" startingDate="2014-06-11" endingDate="2014-07-18"
startingVersion="IEC 61850-6.2007B.SCL.XSD" endingVersion="IEC 61850-6.2007B1.SCL.XSD"/>
    <HistoryFile name="history.2007B2.txt" startingDate="2015-04-02" endingDate="2015-12-02"
startingVersion="IEC 61850-6.2007B1.SCL.XSD" endingVersion="IEC 61850-6.2007B2.SCL.XSD"/>
    <HistoryFile name="history.2007B3.txt" startingDate="2016-02-26" endingDate="2017-03-01"
startingVersion="IEC 61850-6.2007B2.SCL.XSD" endingVersion="IEC 61850-6.2007B3.SCL.XSD"/>
    <HistoryFile name="history.2007B4.txt" startingDate="2017-07-05" endingDate="2017-07-05"
startingVersion="SCL.XSD.v2007B3" endingVersion="SCL.XSD.v2007B4"/>
  </CodeComponent>
</IECManifest>
```

Das Paket wird mithilfe der folgenden Namensregel identifiziert:

{RefStandard}.{CodeComponentName}.{VersionRevision}.{LightFull}{PublicationStage}.zip

Für die vorliegende Veröffentlichung lautet der Paketname der Code-Komponente:

IEC 61850-6.2015.SCL.2007B4.full.zip

Der Lebenszyklus einer Code-Komponente hängt nicht vom Lebenszyklus der zugehörigen Veröffentlichung ab. Der Lebenszyklus der Veröffentlichung umfasst zwei Stufen: Version (entsprechend einer Ausgabe) und Überarbeitung (entsprechend einer Änderung). Mithilfe einer dritten Veröffentlichungsstufe (Freigabe) kann eine Code-Komponente veröffentlicht werden, ohne dass eine Änderung herausgegeben werden muss.

Das ist hilfreich, wenn die Interoperabilität betreffende Fehler („InterOp Tissues“) behoben werden müssen. Dann wird eine neue Freigabe der Code-Komponente veröffentlicht, welche die vorhergehende Freigabe ersetzt, und über die Website des IEC TC 57 verbreitet.

– Leerseite –

FOREWORD

This amendment has been prepared by IEC technical committee 57: Power systems management and associated information exchange.

The text of this amendment is based on the following documents:

FDIS	Report on voting
57/XX/FDIS	57/XX/RVD

Full information on the voting for the approval of this amendment can be found in the report on voting indicated in the above table.

This IEC standard includes Code Components, i.e. components that are intended to be directly processed by a computer. Such content is any text found between the markers <CODE BEGINS> and <CODE ENDS>, or otherwise is clearly labeled in this standard as a Code Component.

The purchase of this IEC standard carries a copyright license for the purchaser to sell software containing Code Components from this standard directly to end users and to end users via distributors, subject to IEC software licensing conditions, which can be found at: <http://www.iec.ch/CCv1>.

If any updates are required to the published code component that needs to apply immediately and can not wait for an amendment (i.e. fixing a major problem), a new release of the Code Component will be issued and distributed through the IEC WebSite. Any new release of the Code Component related to this part will supersede any previously published Code Component including the one published within the current document.

This publication contains attached.xsd files which compose the Code Component of this part. These files are intended to be used as a complement and do not form an integral part of this standard.

The committee has decided that the contents of this amendment and the base publication will remain unchanged until the stability date indicated on the IEC web site under "<http://webstore.iec.ch>" in the data related to the specific publication. At this date, the publication will be

- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

The National Committees are requested to note that for this publication the stability date is 2022.

THIS TEXT IS INCLUDED FOR THE INFORMATION OF THE NATIONAL COMMITTEES AND WILL BE DELETED AT THE PUBLICATION STAGE.

IMPORTANT – The 'colour inside' logo on the cover page of this publication indicates that it contains colours which are considered to be useful for the correct understanding of its contents. Users should therefore print this document using a colour printer.

INTRODUCTION

This Amendment 1 aims to amend the original content of IEC 61850-6:2009 and to correct its content in order ensure a better interoperability of the components implementing it.

This Amendment 1 brings two distinct sets of changes

- 1) Resolved Interop Issues (covered by Clause 3 of this Amendment 1) which have already followed the technical issues (Tissues) process as described in IEC 61850-1 and have reached the green "status".
- 2) Resolved Editorial Tissues (Technical Issues) (covered by Clause 4 of this Amendment 1) which may have lead to interoperability issues.

The resolutions of these issues which lead to these changes are described in greater detail in the Tissue database hosted at <http://tissue.iec61850.com>.

The only new features compared to the original IEC 61850-6:2009 are the inclusion of the Process and Line elements supporting other application areas than substations, and necessary enhancements to fully support the amended communication related parts. Apart from this this Amendment 1 strictly respects the scope of the original edition.

Because some of the corrected issues may have a widespread impact on many pages of the original standard, a complete consolidation of these changes into IEC 61850-6:2009 is circulated through the reference 57/xxxx/INF, which reflects the content of IEC 61850-6:2009 as amended by this Amendment 1.

Technical issues summary

N°, Subject, Cat, Clause and Paragraph are as they appear on the Tissue database hosted at <http://tissue.iec61850.com> where all technical issues have been stored from the origin of IEC 61850.

"Subject" defines very briefly the topic under focus.

"Cat" defines whether this has been considered as an Interoperability Issue (IntOp2) or just a need for amending IEC 61850-6:2009.

The Tissues which have been considered in this Amendment 1 are:

N°	Subject	Cat	Clause	Paragraph
658	Tracking related features	IntOp2	Annex A	
660	XML encoding header repeat	Ed2	Annex A.4	
663	FCDA element cannot be a "functionally constrained logical node"	IntOp2	9.3.7	Table 22
668	Modeling of autotransformer	IntOp2	9.2.4	
678, 699	DO type description table	Ed2		Table 43
687	ResVTms attribute at the SGCB	Ed2	9.3.12	
719	ConfDataSet maxAttributes defineds FCDA elements in data set	IntOp2	9.3.2	Table 11
721	Log element name	IntOp2	9.3.5	
731	SCL example inconsistent	Ed2	9.3.4§	
752	Input section naming	Ed2	9.3.13	
768	bType VisString65 is missing	IntOp2	Annex A	
779	Relative object references	IntOp2	9.5.4.1	

N°	Subject	Cat	Clause	Paragraph
787	SICS I45 inconsistency	Ed2	Annex G	
788	SICS S56 from optional to mandatory	IntOp2	Annex G	
789 (822)	ConfLdName for services applies to client and server	IntOp2	9.3.2	Table 11
804	valKind and IED versus system configurator	IntOp2	9.5.4.1	
806	Max length of log name incosnsten to part 7-2	Ed2	Annex A	SCL_BaseSimpleTypes
807	Indicate if 'owner' is present in RCB	Ed2	9.3.2	Table 11
823	valKind for structured data attributes	IntOp2	9.5.4.1	
824	Short addresses at structured data attributes	IntOp2	9.3.6, 9.5.4.1	
825	Floating point values	IntOp2	9.5.4.1	Table 45
845	SGCB ResvTms	IntOp2	9.3.2	Table 11
853	SBO and ProtNs	IntOp2	9.5.5	
855	recursive SubFunction	Ed2	9.2.7	
856	Voltage level frequency and phases	Ed2	9.2.2	
857	Function/SubFunction for primary equipment	Ed2	9.2.4	
873	Examples for "curvpts"	Ed2	9.3.6	end of paragraph
886	Missing 8-1 P-Types	Ed2	Annex A	
901	tServices at Ap and at IED	Ed2	9.3.2	below Table 12
936	SupSubscription parameter usage is difficult	IntOp2	9.3.2	Table 11
948	Enumeration (string) value format	IntOp2	9.5.6	
949	type of LN inst is ambiguous	Ed2	9.3.5	Table 17
1118	RptEnabled definition	Ed2	9.3.8	
1147	Filehandling service capability	Ed2	9.3.2	Table 11
1195	Typographical error	Ed2	9.3.2	
1208	IP V6 address format	IntOp2	Annex A.5	
1284	SCSM mapping may require a communication section in an ICD file	Ed2	7	
1298	How to differentiate preconfigured Report data sets from those generated by the system tool	Ed2	9.3.2	Table 11
1304	Error in SCL object model	Ed2		Figure 6
1318	SSD will not validate against XSD	Ed2	9.2.6	Note 2
1328	Limitation of the size of identifiers	IntOp2	9.5.2, 9.5.3, 9.5.6	
1354	Changes to SICS	Ed2	Annex G	
1359	Replace "c37_238" with "61850-9-3"	Ed2	9.3.2	
1365	Need to tighten up the XSD in regards to IED name usage	IntOp2	9.2.6	3
1395	Client LN attributes	IntOp2	9.3.8	
1396	The use and configuration flow of LGOS and LSVS is Unclear	Ed2	9.3.2	Table 11
1397	Subscription limitation visible in IxD file	Ed2	Annex G	
1398	originalSciVersion management in SCT	Ed2	9.3.2	
1402	Extref during engineering	Ed2	9.3.13, Annex H(new)	

N°	Subject	Cat	Clause	Paragraph
1415	SICS-S110 IID import mandatory	Ed2	Annex G	Table G.2
1419	Support of IdName on other IEDs	Ed2	Annex G	Table G.1
1434	Add capability to change nofASDU is missing	Ed2	9.3.2	Table 11
1444	Need to support fixed and SCT controlled data sets	Ed2		
1445	ConfReportControl and a fixed reportSettings	Ed2	9.3.2	Table 11
1447	Restriction on EnumTypes in SCL	Ed2		
1448	Not possible to determine the restriction on number of BRCBs if rptMod=both	Ed2		
1450	originalSclXxx computation rules	IntOp2	9.3.2	
1457	Multiple DOI nodes with the same name	Ed2		
1458	New ExtRef attributes for later binding	Ed2		

1 Scope

Replace the text of the Scope with the following new text:

1.1 General

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

The defined language is called System Configuration description Language (SCL). The IED and communication system model in SCL is according to IEC 61850-5 and IEC 61850-7-x. SCSM specific extensions or usage rules may be required in the appropriate parts.

The configuration language is based on the Extensible Markup Language (XML) version 1.0 (see XML references in Clause 2).

This standard does not specify individual implementations or products using the language, nor does it constrain the implementation of entities and interfaces within a computer system. This part of the standard does not specify the download format of configuration data to an IED, although it could be used for part of the configuration data.

1.2 Namespace name and version

This new section is mandatory for any IEC 61850 namespace (as defined by IEC 61850-7-1:2011).

The parameters which are identifying this new release of the SCL namespace `xmlns:scl="http://www.iec.ch/61850/2003/SCL"` are:

- Namespace Version: 2007
- Namespace Revision: B
- Namespace Release: 4
- Namespace release date: 2017/07/05

Edition	Publication date	Webstore	Namespace
Edition 1.0	2004-03	IEC 61850-6:2004	IEC 61850-6:2003
Edition 2.0	2009-12	IEC 61850-6:2009	IEC 61850-6:2007B
Amendment 1 of Edition 2.0	2017-??	IEC 61850-6:2009/AMD1:2017	IEC 61850-6:2007B4
Edition 2.1	2017-??	IEC 61850-6:2009+AMD1:2017 CSV	IEC 61850-6:2007B4

1.3 Code Component distribution

The Code Components included in this IEC standard are also available as electronic machine readable file at:

http://www.iec.ch/public/TC57/supportdocuments/IEC_61850-6.2015.SCL.2007B4.full.zip

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 61850-6.2015.SCL.{VersionStateInfo}.full.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 IECManifest related to this publication is:

```
<IECManifest xmlns="http://www.iec.ch/CC/2017/IECManifest" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.iec.ch/CC/2017/IECManifest IECManifest.xsd">
  <Copyright>
    <Notice>
      COPYRIGHT (c) IEC, 2017. This version of this XSD is part of IEC 61850-6:2009/AMD1:2016;
      see the IEC 61850-6:2009/AMD1:2016 for full legal notices. In case of any differences between the here-below code and the
      IEC published content, the here-below definition supersedes the IEC publication; it may contain updates. See history files. The
      whole document has to be taken into account to have a full description of this code component.
      See www.iec.ch/CCv1 for copyright details.
    </Notice>
    <License uri="www.iec.ch/CCv1">IEC License</License>
  </Copyright>
  <CodeComponent id="IEC 61850-6.2007B4.SCL.XSD" name="IEC 61850-6 SCL schema V2007B4" content="full"
date="2017-07-05">
    <Publication name="IEC 61850-6.2015_ed2.1" comment="Configuration description language for
communication in power utility automation systems related to IEDs"/>
    <File name="SCL.xsd" category="normative" content="full"/>
    <File name="SCL_Substation.xsd" category="normative" content="full"/>
    <File name="SCL_Communication.xsd" category="normative" content="full"/>
    <File name="SCL_IED.xsd" category="normative" content="full"/>
    <File name="SCL_DataTypeTemplates.xsd" category="normative" content="full"/>
    <File name="SCL_BaseTypes.xsd" category="normative" content="full"/>
    <File name="SCL_BaseSimpleTypes.xsd" category="normative" content="full"/>
    <File name="SCL_Enums.xsd" category="normative" content="full"/>
    <File name="SCL.Doc.HTML.zip" category="normative" content="full" comment="Zip archive containing the
HTML documentation of the SCL. Contains the 'SCL.html' file and all related pictures"/>
    <HistoryFile name="history.2007B1.txt" startDate="2014-06-11" endDate="2014-07-18"
startingVersion="IEC 61850-6.2007B.SCL.XSD" endingVersion="IEC 61850-6.2007B1.SCL.XSD"/>
    <HistoryFile name="history.2007B2.txt" startDate="2015-04-02" endDate="2015-12-02"
startingVersion="IEC 61850-6.2007B1.SCL.XSD" endingVersion="IEC 61850-6.2007B2.SCL.XSD"/>
    <HistoryFile name="history.2007B3.txt" startDate="2016-02-26" endDate="2017-03-01"
startingVersion="IEC 61850-6.2007B2.SCL.XSD" endingVersion="IEC 61850-6.2007B3.SCL.XSD"/>
    <HistoryFile name="history.2007B4.txt" startDate="2017-07-05" endDate="2017-07-05"
startingVersion="SCL.XSD.v2007B3" endingVersion="SCL.XSD.v2007B4"/>
  </CodeComponent>
</IECManifest>
```

The package is identified using the following naming rule:

{RefStandard}.{CodeComponentName}.{VersionRevision}.{LightFull}{PublicationStage}.zip

For current publication, the Code Component package name is:

IEC 61850-6.2015.SCL.2007B4.full.zip

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) allow 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.

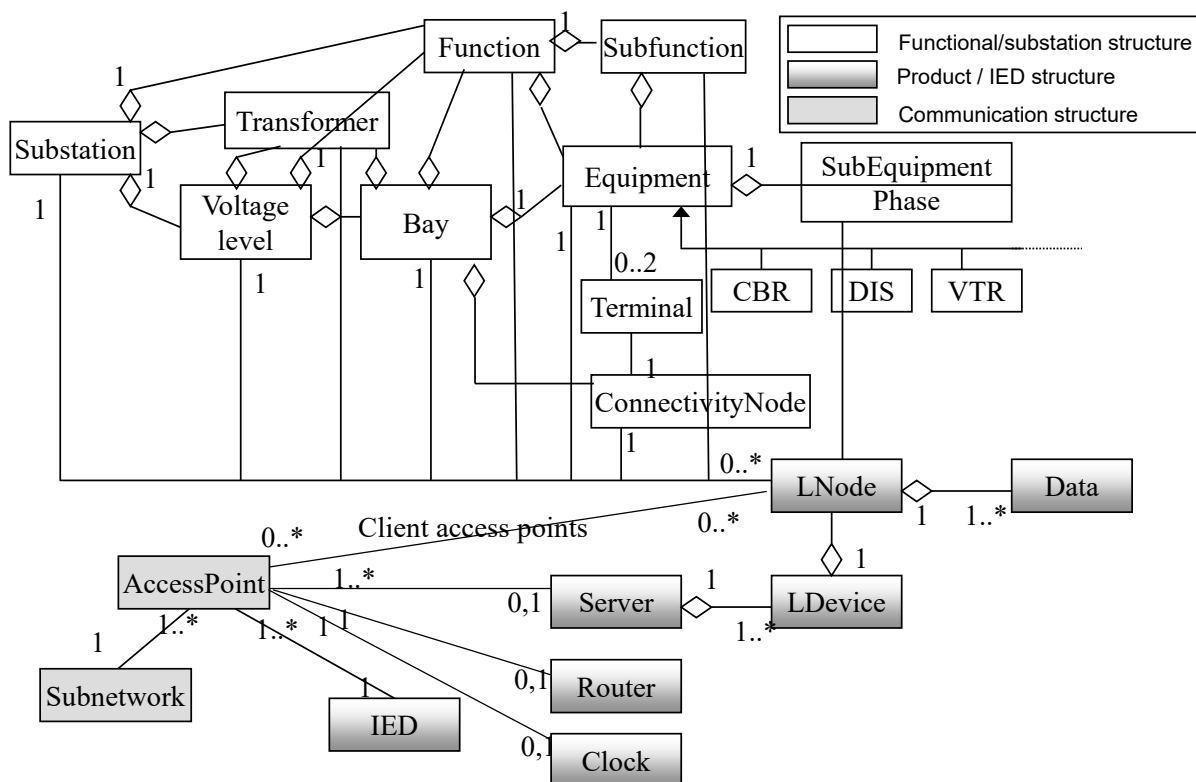
6 The SCL object model

6.1 General

Replace the text of the first numbered point in the fifth paragraph by the following:

- 1) Substation / Line / Process: this part describes the primary process related functions and devices like switch yard, respectively any primary process in the functional view according to IEC 81346-1, electrical connections on single line level (topology), and the designation of equipment and functions;

Replace Figure 6 with the following new Figure 6:



IEC

Figure 6 – SCL Substation object model

6.2 The substation model

Replace the existing title and text of Subclause 6.2 with the following new title and text:

6.2 The process model

The process model (upper part of Figure 6 for substations) is an object hierarchy based on the functional structure of the primary process. A special primary process is the Substation, which has its own SCL element. Another is the Line, which connects electrically bays of substations. The generic *Process* element shall be used to model any additionally needed structuring levels of a power grid above the substation and for all other primary processes. Although each object is self-contained, its reference designation is derived from its name.

the hierarchy. Because LNs perform functions within the complete context of the Process / Substation respective Line hierarchy, they can be attached as functional objects at each function level. Typically, a switch controller LN is attached to a switching device, while a measuring LN is attached to the bay, which delivers the measurands, and transformer-related LNs are attached to the appropriate transformer.

NOTE 1 In the CIM model measurands are allocated to primary device terminals. This is a topological allocation, while the allocation in SCL in first line serves functional naming. However, if the single line topology is modelled completely, by means of the transformers (VTR, CTR) and their data acquisition nodes (TVTR, TCTR) also some primary device terminal in the topology can be found to which the measurands belong according to the CIM model.

The purpose of the process model is

- to relate a logical node and its function to a function of or at the primary process (process part or substation part or line part or equipment or subequipment);
- to derive a functional designation for the logical node from the process structure.

The following objects of the functional structure (in hierarchical order) are used in the SCL model, for power networks analogue to the CIM model for energy management systems. More background information on these terms can be found in IEC 61850-2:

Process	the whole or part of the primary process handled by the automation system, which is no substation. Can be parts of the power grid containing several substations, or complete other primary processes like power plants.
Line	a line connecting several substations.
Substation	the object identifying a whole substation.
VoltageLevel	an identifiable, electrically connected substation part having an identical voltage level.
Bay	an identifiable part or subfunction of the switch yard (substation) within one voltage level.
Equipment	an apparatus within the switch yard, for example circuit breaker, disconnecter, voltage transformer, power transformer winding etc. The single line diagram of a switch yard shows the electrical connections between these primary devices. Connectivity node objects model these connections. Therefore, each primary device can contain at its terminals references to the connectivity nodes to which it is connected. At single line level, one or two terminals (connections) per equipment are normally sufficient.
SubEquipment	a part of an Equipment, which might especially be one phase of a three-phase equipment.
ConnectivityNode	the (electrical) connectivity node object connecting different primary devices. Typical connectivity node examples are: connecting nodes within a bay, bus bars connecting several bays in the same voltage level, lines connecting bays in different substations. See also Equipment above.
Terminal	an electrical connection point of a primary apparatus at single line level. A terminal can be connected to a ConnectivityNode. Within SCL terminals can be explicitly named, or exist implicitly.
Function	allows additional functions at substation, voltage level or bay level, either independent from the basic switch yard functionality like fire fighting or building supervision, or as part of the switch yard like main 1 protection and main 2 protection.
SubFunction	a hierarchical subpart of a Function or SubFunction, e.g. earth fault protection as subpart of the main 1 function.
EqFunction	allows additional functions at or below Equipment level, e.g. redundant functions on the same equipment.
EqSubFunction	a hierarchical subpart of an EqFunction or EqSubFunction.

The *PowerTransformer* is a special equipment, which can hierarchically be located below Substation, VoltageLevel or Bay. It contains Transformer windings as equipment, which might again have a relation to a tap changer.

NOTE 2 Observe that the hierarchical structure is used for functional designations. If substructures of bays are needed, this can be introduced by appropriate structured bay names. If, for example, a bay B1 is structured into sub-bays SB1 and SB2, this would in the SCL model lead to two bays named B1.SB1 and B1.SB2. If logical nodes are also attached to the B1 structure level, then B1 can be introduced as a third bay.

NOTE 3 In the CIM model the bay level is optional, while in SCL it is mandatory. However, if the bay level structuring is not needed, a whole voltage level can be considered to be one bay. The only restriction here is that the SCL syntax demands at least one character as name on each level, so that in this case the voltage level name needs at least 2 characters, from which within the SCL substation structure the first character is taken as the voltage level name, and the last character is taken as the name for the one bay element.

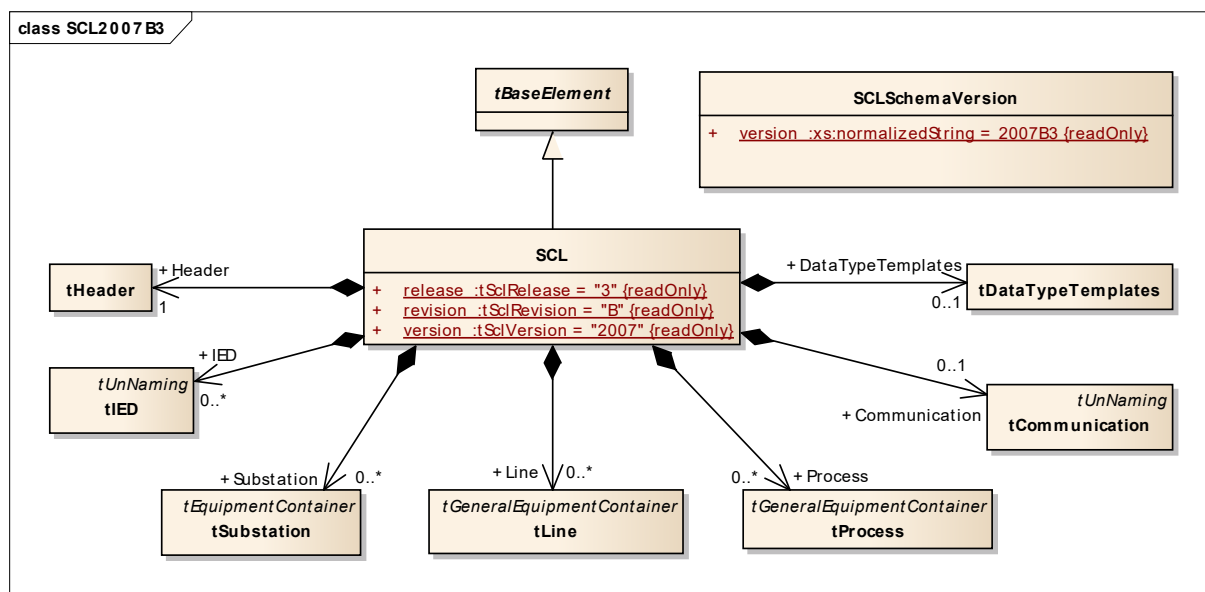
7 SCL description file types

Add the following new text at the end of the first paragraph of the first bullet (ICD file):

- A specific SCSM might make this mandatory for some address parts.

8.1 Specification method

Replace Figure 9 by the following new Figure 9:



8.2.1 MustUnderstand rules

Replace the existing first paragraph of 8.2.1 with the following new text:

The MustUnderstand / MayIgnore rules have to be followed when reading an SCL file. When producing SCL output, this should conform to the claimed supported SCL version/revision/release.

Elements, which a tool or an IED must understand to produce interoperable results, shall be declared as *mustUnderstand* and marked with the *mustUnderstand* attribute with value *true*, so that the tool processing the instance knows if it can ignore the element or not. All elements which the tool does not understand and which do not have the *mustUnderstand* property, can safely be ignored. The 'may ignore all' strategy for elements (tags) is taken, i.e. ignore the element and all its contained contents. If a known element contains directly below it an

element with `mustUnderstand` property which a tool does not understand, then it must also ignore the known containing element.

For attributes just the attribute not understood is ignored. This means especially, that there is no 'mustUnderstand' possibility for attributes, only for elements. Therefore adding of attributes to the language is done only as optional attributes with a defined default value in the newer version, which is backward compatible to 'not knowing this attribute'. For later compatibility it is good practice to use these default values from the schema by not explicitly writing them into the SCL instance. This is possible because once released default values are not changed in the schema, as long as the attribute itself is needed.

Add, at the end of 8.2.1, the following new text:

The following example illustrates the use of the *mustUnderstand* feature.

An IED defines a GOOSE control block as follows:

```
<GSEControl name="GoCB02" appID="GoCB02" datSet="MyDatSet">
  <IEDName>AA1_D1_Q11A1</IEDName>
  <IEDName>AA1_D1_Q10A1</IEDName>
  <IEDName>AA1_D1_Q07A1</IEDName>
  <Protocol mustUnderstand="true">R-GOOSE</Protocol>
</GSEControl>
```

If the tool importing this does not understand the *Protocol* element, then it is also not allowed to use (e.g. add IED names) the *GSEControl* element containing it, although *GSEControl* is known to it.

8.2.3 Incompatibilities to earlier versions

Replace the text of the fifth bullet point with the following new text:

- The introduction of the *mustUnderstand* attribute; it is currently used for the *Protocol* element contained in GOOSE and SV control block definitions.

Add the following new text after the eighth bullet point:

- The newly introduced *LDevice* attribute *IdName* leads to incompatibilities with edition 1 (2003A) tools, as ignoring this attribute leads to wrong communication level configuration
- The attribute *ReportControl.rptId* shall no longer have the empty string value
- It is clarified that the meaning of the *maxAttributes* attribute of the *ConfDataSet* element denotes FCDAs and not basic attributes.
- The *max* attribute of the *SupSubscription* element has been replaced by two special attributes for GOOSE and SV supervision.
- The modeling of the transformer neutral point has been corrected.
- The length of *DataTypeTemplate* identifiers is restricted to 255 characters
- *EqFunction* and *EqSubFunction* are used below Equipment elements instead of *Function* / *SubFunction*

8.3.5 XML name spaces

Replace the existing first paragraph with the following new text:

For all tag elements inheriting from *tBaseElement*, private or standardized extensions may be added by means of (sub-)tags and attributes. These added parts shall belong to a defined XML name space with defined semantics for all these elements. It is recommended to define the used name spaces at the main tag (SCL), although standard XML allows to add it also

below at elements from this name space. Observe that standard XML rules allow to replace this definition.

This namespace shall not be the same as the target namespace of the SCL schema (see below). For private name spaces, the used internal name space abbreviation should start with the character **e**. IED configurator tools shall be aware that the used name space abbreviation might be changed by the system configurator, if its relation to the referenced URI is not unique across all used ICD/IID files. An example of a standard extension for single line or communication diagram layouts is given in Annex C. The name space URI of this version of the SCL, which shall be used as default name space in all SCL files, is:

```
xmlns:scl="http://www.iec.ch/61850/2003/SCL"
```

8.4 General Structure

Replace the first paragraph with the following new text:

An SCL – XML document starts with the XML *prolog*, and then continues with elements as defined later. The *prolog* shall contain the identification of the XML version and the character coding used. UTF-8 coding is the preferred coding and shall be supported if standard conformance is claimed. The XML encoding attribute shall be processed in a case-insensitive way. SCL generators shall always use upper case encoding.

The whole SCL definition part is contained in the SCL element:

9.1 Header

Add the following new text after the Note after Table 3:

The revision history is optional. It describes the modification history of the containing file. This means e.g. that in principle the history of an SCD file is independent from that of IID and ICD files it is based on, although the SCD file history might contain the inclusion of a specific IID file version and revision into a project as comment.

9.2 Substation description

Replace the existing title of Subclause 9.2 by the following new title:

9.2 Process description

9.2.1 General

Replace the first five paragraphs before Figure 15 with the following new text:

The process section serves to describe the functional structure of a primary process like a power plant or a substation, and to identify the primary devices and their electrical connections inside substations and power networks. For an industrial process or to describe whole power networks, it is possible to have several substation sections, one for each substation served by the automation system. Sets of substations can be structured by means of the Process element. Substations can be electrically connected by lines modelled with the Line element. By means of logical nodes attached to the primary process elements, this clause defines additionally the system functionality (for example, in an SSD file), or, in the case where the logical nodes are already allocated to IEDs (SCD file), the relation of IED functions to the power system.

Note that the *name* attribute is always mandatory and shall not be the empty string. If the substation, line or process section is used as a template within an ICD file, then the name shall be TEMPLATE. The name value is also a global identification of the process element,

because it shall be unique for all substation / line / process elements contained in the SCL file.

If the *desc* attribute is missing, its default value is an empty string.

The *Process* and *Function* related elements have additionally an optional *type* attribute, which allows to define the type of the process / function object. it is recommended that other application areas standardize on these type values.

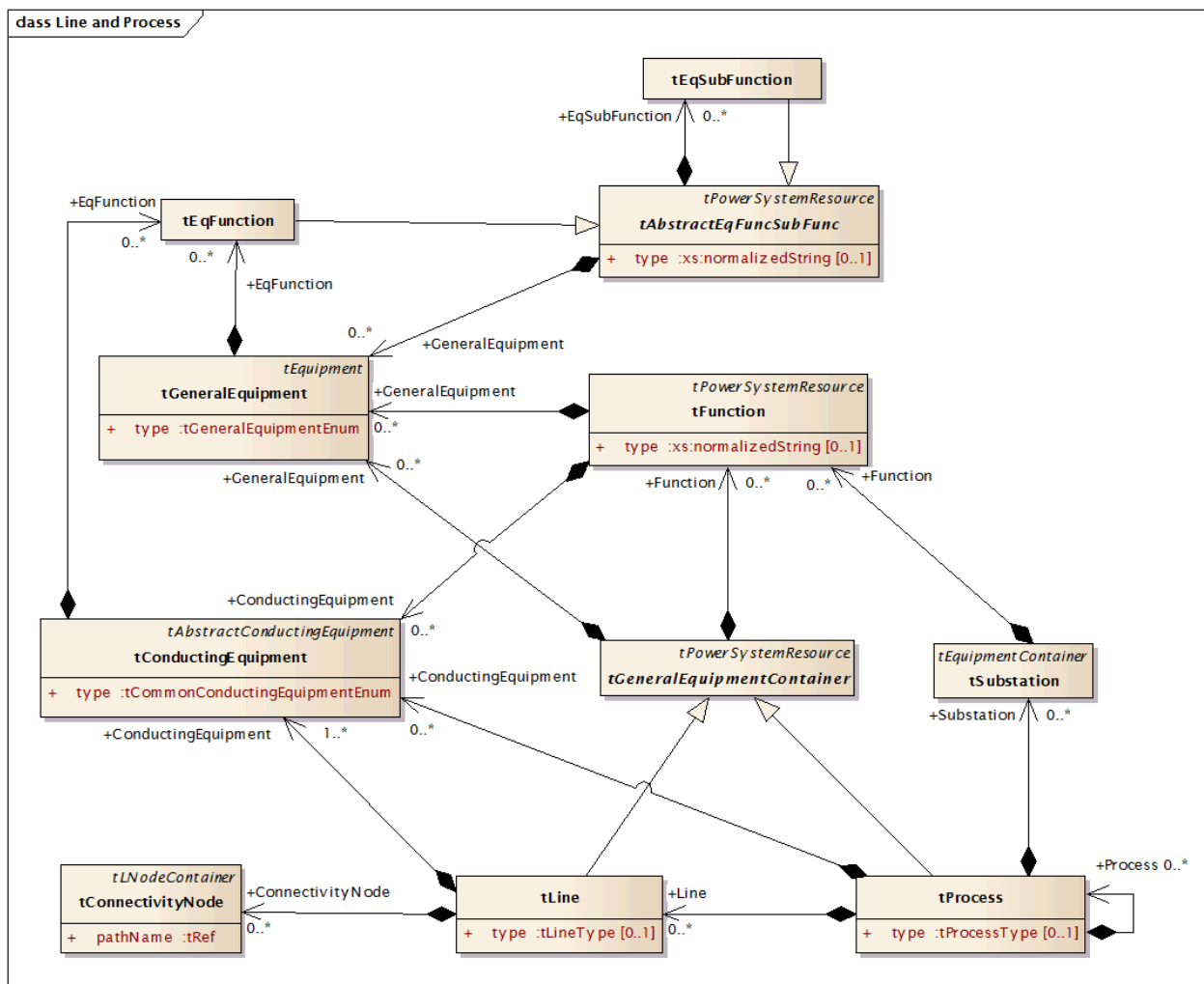
Logical nodes (LNode) can be attached at each level of the structure (i.e., substation, voltage level, bay, equipment, subequipment respective function, subfunction). Power transformers (*PowerTransformer*) can also be attached at the structure levels substation, voltage level and bay. Conducting equipments (*ConductingEquipment*) can only be attached to the bay level. Logical node instances at the same level shall have different identifications.

The UML diagram of Figure 15 gives an overview on the substation section without the *Process* element:

Add the following new text to the end of Subclause 9.2.1:

The *Process* element is a logical node container, which can be used for other processes than substations, or to group several substations into parts of a power grid. For the first purpose it can also contain equipment elements. It can be recursively used, and its type attribute can indicate the type of the process part / process object identified by the name attribute. It is recommended that the standards for the application areas standardize the type attribute values.

The structure is shown as UML diagram in Figure 24.



IEC

Figure 24 – UML diagram of Process and Line elements

The general *Process* element (of type *tProcess*, see *Figur 15a*), which is referred to by the *SCL* element, includes additionally several identity constraints:

- Within a *Process*, there cannot be two *Substation* elements with the same name.
- Within a *Process*, there cannot be two direct *ConductingEquipment* or *GeneralEquipment* elements with the same *name*.
- Within a *Process*, there cannot be two *Function* elements with the same *name*.
- Within a *Process*, there cannot be two *LNode* elements with the same combination of *InInst*, *InClass*, *iedName*, *IdInst*, and *prefix*.
- Further, in order to avoid any ambiguities, within a *Process* there cannot be two direct child elements with the same *name*.
- In general, at each hierarchy level within the *Process* section all names shall be unique, leading to unique object references (path names) of all objects defined by the substation naming hierarchy.

Further, inside system specifications, the following rule holds:

- For a primary system template within an ICD file, the *Process* name shall be *TEMPLATE*. There can be a maximum of one *Process* element named *TEMPLATE* per level in one *SCL* file.
- *Process* elements at different levels in a hierarchy branch, which have a type defined, should have different types. Any exceptions shall be explicitly defined for this usage.

The *Line* element is a logical node container, which can be used to model lines between substations of a power grid. It can contain equipment elements modelling line segments, general equipment and connectivity nodes.

The *Line* element has similar restrictions as all other process elements, namely that

- each contained element shall have a unique name.
- if a *Line* element is used as template in an ICD file, its name shall be TEMPLATE.

Terminals inside Line elements are allowed to reference connectivity nodes in substations. Substations shall never reference connectivity nodes inside Line elements.

9.2.2 Voltage level

Replace the existing XML schema with the following new schema, and add the following new text after this schema segment:

```
<xs:complexType name="tVoltageLevel">
  <xs:complexContent>
    <xs:extension base="tEquipmentContainer">
      <xs:sequence>
        <xs:element name="Voltage" type="tVoltage" minOccurs="0"/>
        <xs:element name="Bay" type="tBay" maxOccurs="unbounded">
          <xs:unique name="uniqueChildNameInBay">
            <xs:selector xpath="/*"/>
            <xs:field xpath="@name"/>
          </xs:unique>
          <xs:unique name="uniqueLNodeInBay">
            <xs:selector xpath="/scl:LNode"/>
            <xs:field xpath="@lnInst"/>
            <xs:field xpath="@lnClass"/>
            <xs:field xpath="@jedName"/>
            <xs:field xpath="@ldInst"/>
            <xs:field xpath="@prefix"/>
          </xs:unique>
        </xs:element>
        <xs:element name="Function" type="scl:tFunction" minOccurs="0" maxOccurs="unbounded">
          <xs:unique name="uniqueLNodeInFunctionVL">
            <xs:selector xpath="/scl:LNode"/>
            <xs:field xpath="@lnInst"/>
            <xs:field xpath="@lnClass"/>
            <xs:field xpath="@jedName"/>
            <xs:field xpath="@ldInst"/>
            <xs:field xpath="@prefix"/>
          </xs:unique>
          <xs:unique name="uniqueChildNameInVoltageLevelFunc">
            <xs:selector xpath="/*"/>
            <xs:field xpath="@name"/>
          </xs:unique>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="nomFreq" use="optional">
        <xs:simpleType>
          <xs:restriction base="xs:decimal">
            <xs:minInclusive value="0"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
      <xs:attribute name="numPhases" use="optional">
        <xs:simpleType>
          <xs:restriction base="xs:unsignedByte">
            <xs:minExclusive value="0"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Beneath *name* and *desc* like any equipment container the voltage level has the following additional attributes:

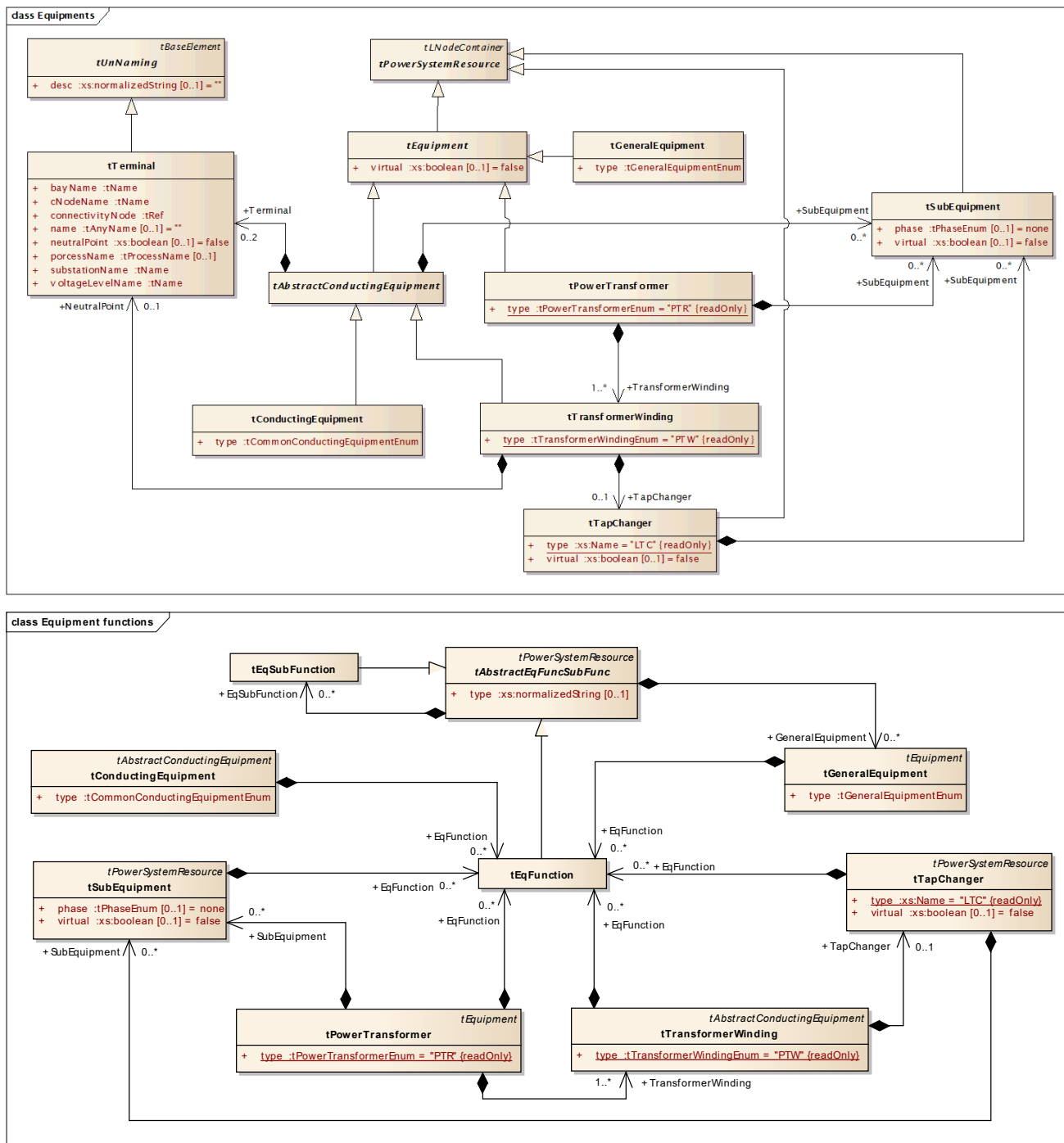
- *nomFreq*: the nominal frequency in Hz, 0 for DC systems. If missing, the nominal frequency is not known.
- *numPhases*: the number of phases of the single line considered, typically 1, 2 or 3. If missing, not known.

9.2.4 Power equipment

Replace the existing first paragraph of 9.2.4 with the following new text:

The power equipment is subdivided into the *PowerTransformer* and *ConductingEquipment*. The *PowerTransformer* might appear in each equipment container, and contains the transformer windings as special *ConductingEquipment*. To each transformer winding, a tap changer and a neutral point can be allocated. All other *ConductingEquipment* might appear in the bays only. All equipment is derived from the *tEquipment* base type, and the *ConductingEquipment* from the *tAbstractConductingEquipment* type.

Replace Figure 16 with the following new figure:



IEC

Figure 16 – UML diagrams for equipment type inheritance and relations

Replace the existing XML schema with the following new schema:

```
<xs:complexType name="tEquipment" abstract="true">
  <xs:complexContent>
    <xs:extension base="tPowerSystemResource">
      <xs:attributeGroup ref="agVirtual"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tAbstractConductingEquipment" abstract="true">
  <xs:complexContent>
    <xs:extension base="tEquipment">
      <xs:sequence>
        <xs:element name="Terminal" type="tTerminal" minOccurs="0" maxOccurs="2"/>
        <xs:element name="SubEquipment" type="tSubEquipment" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

```

<xs:unique name="uniqueLNodeInSubEquipment">
  <xs:selector xpath="/scl:LNode"/>
  <xs:field xpath="@lnInst"/>
  <xs:field xpath="@lnClass"/>
  <xs:field xpath="@iedName"/>
  <xs:field xpath="@ldInst"/>
  <xs:field xpath="@prefix"/>
</xs:unique>
<xs:unique name="uniqueChildNameInACESubEquipment">
  <xs:selector xpath="/*"/>
  <xs:field xpath="@name"/>
</xs:unique>
</xs:element>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tConductingEquipment">
  <xs:complexContent>
    <xs:extension base="tAbstractConductingEquipment">
      <xs:sequence>
        <xs:element name="EqFunction" type="scl:tEqFunction" minOccurs="0" maxOccurs="unbounded">
          <xs:unique name="uniqueLNodeInFuncForCE">
            <xs:selector xpath="/scl:LNode"/>
            <xs:field xpath="@lnInst"/>
            <xs:field xpath="@lnClass"/>
            <xs:field xpath="@iedName"/>
            <xs:field xpath="@ldInst"/>
            <xs:field xpath="@prefix"/>
          </xs:unique>
          <xs:unique name="uniqueChildNameInFuncForCE">
            <xs:selector xpath="/*"/>
            <xs:field xpath="@name"/>
          </xs:unique>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="type" type="tCommonConductingEquipmentEnum" use="required"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tSubEquipment">
  <xs:complexContent>
    <xs:extension base="tPowerSystemResource">
      <xs:sequence>
        <xs:element name="EqFunction" type="scl:tEqFunction" minOccurs="0" maxOccurs="unbounded">
          <xs:unique name="uniqueLNodeInFuncForSubEq">
            <xs:selector xpath="/scl:LNode"/>
            <xs:field xpath="@lnInst"/>
            <xs:field xpath="@lnClass"/>
            <xs:field xpath="@iedName"/>
            <xs:field xpath="@ldInst"/>
            <xs:field xpath="@prefix"/>
          </xs:unique>
          <xs:unique name="uniqueChildNameInFuncForSubEq">
            <xs:selector xpath="/*"/>
            <xs:field xpath="@name"/>
          </xs:unique>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="phase" type="tPhaseEnum" use="optional" default="none"/>
      <xs:attributeGroup ref="agVirtual"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tPowerTransformer">
  <xs:complexContent>
    <xs:extension base="tEquipment">
      <xs:sequence>
        <xs:element name="TransformerWinding" type="tTransformerWinding" maxOccurs="unbounded">
          <xs:unique name="uniqueLNodeInTransformerWinding">
            <xs:selector xpath="/scl:LNode"/>
            <xs:field xpath="@lnInst"/>
            <xs:field xpath="@lnClass"/>

```

```
<xs:field xpath="@iedName"/>
<xs:field xpath="@ldInst"/>
<xs:field xpath="@prefix"/>
</xs:unique>
<xs:unique name="uniqueChildNameInPTW">
  <xs:selector xpath="./scl:SubEquipment|./scl:TapChanger|./scl:EqFunction"/>
  <xs:field xpath="@name"/>
</xs:unique>
</xs:element>
<xs:element name="SubEquipment" type="scl:tSubEquipment" minOccurs="0" maxOccurs="unbounded">
  <xs:unique name="uniqueLNodeInSubEquipmentPTR">
    <xs:selector xpath="./scl:LNode"/>
    <xs:field xpath="@lnInst"/>
    <xs:field xpath="@lnClass"/>
    <xs:field xpath="@iedName"/>
    <xs:field xpath="@ldInst"/>
    <xs:field xpath="@prefix"/>
  </xs:unique>
  <xs:unique name="uniqueChildNameInPTRSubEquipment">
    <xs:selector xpath=".*"/>
    <xs:field xpath="@name"/>
  </xs:unique>
</xs:element>
<xs:element name="EqFunction" type="scl:tEqFunction" minOccurs="0" maxOccurs="unbounded">
  <xs:unique name="uniqueLNodeInFuncForPTR">
    <xs:selector xpath="./scl:LNode"/>
    <xs:field xpath="@lnInst"/>
    <xs:field xpath="@lnClass"/>
    <xs:field xpath="@iedName"/>
    <xs:field xpath="@ldInst"/>
    <xs:field xpath="@prefix"/>
  </xs:unique>
  <xs:unique name="uniqueChildNameInFuncForPTR">
    <xs:selector xpath=".*"/>
    <xs:field xpath="@name"/>
  </xs:unique>
</xs:element>
</xs:sequence>
<xs:attribute name="type" type="tPowerTransformerEnum" use="required" fixed="PTR"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tTransformerWinding">
  <xs:complexContent>
    <xs:extension base="tAbstractConductingEquipment">
      <xs:sequence>
        <xs:element name="TapChanger" type="tTapChanger" minOccurs="0">
          <xs:unique name="uniqueLNodeInTapChanger">
            <xs:selector xpath="./scl:LNode"/>
            <xs:field xpath="@lnInst"/>
            <xs:field xpath="@lnClass"/>
            <xs:field xpath="@iedName"/>
            <xs:field xpath="@ldInst"/>
            <xs:field xpath="@prefix"/>
          </xs:unique>
          <xs:unique name="uniqueChildNameInLTC">
            <xs:selector xpath=".*"/>
            <xs:field xpath="@name"/>
          </xs:unique>
        </xs:element>
        <xs:element name="NeutralPoint" type="tTerminal" minOccurs="0"/>
        <xs:element name="EqFunction" type="scl:tEqFunction" minOccurs="0" maxOccurs="unbounded">
          <xs:unique name="uniqueLNodeInFuncForPTW">
            <xs:selector xpath="./scl:LNode"/>
            <xs:field xpath="@lnInst"/>
            <xs:field xpath="@lnClass"/>
            <xs:field xpath="@iedName"/>
            <xs:field xpath="@ldInst"/>
            <xs:field xpath="@prefix"/>
          </xs:unique>
          <xs:unique name="uniqueChildNameInFuncForPTW">
            <xs:selector xpath=".*"/>
            <xs:field xpath="@name"/>
          </xs:unique>
        </xs:element>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

```

        </xs:unique>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="type" type="tTransformerWindingEnum" use="required" fixed="PTW"/>
  </xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tTapChanger">
  <xs:complexContent>
    <xs:extension base="tPowerSystemResource">
      <xs:sequence>
        <xs:element name="SubEquipment" type="scl:tSubEquipment" minOccurs="0" maxOccurs="unbounded">
          <xs:unique name="uniqueLNodeInSubEquipmentLTC">
            <xs:selector xpath="/scl:LNode"/>
            <xs:field xpath="@lnInst"/>
            <xs:field xpath="@lnClass"/>
            <xs:field xpath="@iedName"/>
            <xs:field xpath="@ldInst"/>
            <xs:field xpath="@prefix"/>
          </xs:unique>
          <xs:unique name="uniqueChildNameInLTCSubEquipment">
            <xs:selector xpath="."/>
            <xs:field xpath="@name"/>
          </xs:unique>
        </xs:element>
        <xs:element name="EqFunction" type="scl:tEqFunction" minOccurs="0" maxOccurs="unbounded">
          <xs:unique name="uniqueLNodeInFuncForLTC">
            <xs:selector xpath="/scl:LNode"/>
            <xs:field xpath="@lnInst"/>
            <xs:field xpath="@lnClass"/>
            <xs:field xpath="@iedName"/>
            <xs:field xpath="@ldInst"/>
            <xs:field xpath="@prefix"/>
          </xs:unique>
          <xs:unique name="uniqueChildNameInFuncForLTC">
            <xs:selector xpath="."/>
            <xs:field xpath="@name"/>
          </xs:unique>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="type" type="xs:Name" use="required" fixed="LTC"/>
      <xs:attributeGroup ref="agVirtual"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tGeneralEquipment">
  <xs:complexContent>
    <xs:extension base="tEquipment">
      <xs:sequence>
        <xs:element name="EqFunction" type="scl:tEqFunction" minOccurs="0" maxOccurs="unbounded">
          <xs:unique name="uniqueLNodeInFuncForGE">
            <xs:selector xpath="/scl:LNode"/>
            <xs:field xpath="@lnInst"/>
            <xs:field xpath="@lnClass"/>
            <xs:field xpath="@iedName"/>
            <xs:field xpath="@ldInst"/>
            <xs:field xpath="@prefix"/>
          </xs:unique>
          <xs:unique name="uniqueChildNameInFuncForGE">
            <xs:selector xpath="."/>
            <xs:field xpath="@name"/>
          </xs:unique>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="type" type="tGeneralEquipmentEnum" use="required"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

Add the following new text after the first paragraph following the XML schema:

To allow functional identification also below equipment level, all equipment elements allow to contain EqFunction definitions. These are essentially the same as the Function / SubFunction structures above equipment, however do not allow another equipment of the same class inside it.

Replace the existing XML schema part before Table 6 as follows. Note that a neutral point connection is now a special terminal element at the transformer winding:

```
<xs:complexType name="tTerminal">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:attribute name="name" type="tAnyName" use="optional" default=""/>
      <xs:attribute name="connectivityNode" type="tRef" use="required"/>
      <xs:attribute name="processName" type="tProcessName" use="optional"/>
      <xs:attribute name="substationName" type="tName" use="required"/>
      <xs:attribute name="voltageLevelName" type="tName" use="required"/>
      <xs:attribute name="bayName" type="tName" use="required"/>
      <xs:attribute name="cNodeName" type="tName" use="required"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Replace existing Table 6 with the following new table and text:

Table 6 – Attributes of the Terminal element

Attribute name	Description
name	The optional relative name of the terminal at this Equipment. The default is the empty string, which means that the name of the ConnectivityNode is also the terminal identification.
desc	Descriptive text to the terminal
connectivityNode	The pathname of the connectivity node to which this terminal connects. If the Equipment shall not be connected, then the whole Terminal element shall be removed.
processName	The name of the Process element containing the connectivityNode (if any – else missing)
lineName	The name of a line element containing the connectivityNode (only used inside Line element)
substationName	The name of the substation containing the connectivityNode (only used to reference a substation, not a line)
voltageLevelName	The name of the voltage level containing the connectivityNode (only used to reference a substation, not a line)
bayName	The name of the bay containing the connectivityNode (only used to reference a substation, not a line)
cNodeName	The (relative) name of the connectivityNode within its bay

References to connectivity nodes inside a line are only allowed inside Line elements. In this case the lineName attribute is required. For references to connectivity nodes in substations from Substation or Line elements lineName shall not be used, instead substationName, voltageLevelName and bayName are required.

9.2.6 Process function logical nodes

Replace the existing text of Subclause 9.2.6 with the following new text:

All equipment and equipment containers are also logical node containers. The logical node (abbreviated here as LN) defines the function part performed at the appropriate level of the hierarchy. The LNode element identifies the function by specifying a logical node as defined in IEC 61850-5, IEC 61850-7-x or other domain standards. The optional attribute desc may contain some operator-related text describing the LN and its usage.

```

<xs:complexType name="tLNode">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:attribute name="iedName" type="tIEDName" use="optional" default="None"/>
      <xs:attribute name="IdInst" type="tLDInstOrEmpty" use="optional" default=""/>
      <xs:attribute name="prefix" type="tPrefix" use="optional" default=""/>
      <xs:attribute name="InClass" type="tLNClassEnum" use="required"/>
      <xs:attribute name="InInst" type="tLNInstOrEmpty" use="optional" default=""/>
      <xs:attribute name="InType" type="tName" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

The logical node and its function is identified by the element attributes. The *LNode* element can be used within an SSD for functional specification, without allocation to an IED. In this case the *iedName* shall be **None**. This name **None** is forbidden for an IED definition in the IED section. For more detailed specification *InType* may refer to a logical node type definition (9.5.2), which then also defines the optional data objects required to exist in this special case, or defines certain values, which some (configuration) parameters shall have. If the logical node is later allocated to an IED within an SCD, then the value of this *InType* attribute can be ignored, or may be used to check if the logical node type used on the IED fulfills the requirements.

The attributes of the *LNode* element are described in Table 8.

Table 8 – Attributes of the *LNode* element

Attribute name	Description
InInst	The LN instance identification. Can only be missing for InClass=LLN0, meaning as value here the empty string
InClass	The LN class as defined in IEC 61850-7-x or other domain standards
iedName	The name of the IED which contains the LN, None if used for specification (default if attribute is not specified)
IdInst	The LD instance on the IED which contains the LN. Within a specification (SSD file), where iedName=None, this shall result in unique LN instance identification, i.e. may contain the LD name
prefix	The LN prefix used in the IED (if needed; default, if not specified, is the empty string). Can be used for more detailed function specification than possible by LN class alone, if the LN is not allocated to an IED
InType	The logical node type definition containing more detailed functional specification. Might be missing, especially if the LN is allocated to an IED.

NOTE For LLN0, the value of inst is the empty string. In all other cases, it is an unsigned integer.

The *iedName* identifies the IED on which the LN resides, the *IdInst* the LD within this IED to which the LN belongs. The attributes *prefix*, *InClass* and *inst* (meaning the LN instance identification according to IEC 61850-7-x or other domain standards) then identify the logical node within that LD. In this way, the binding between the substation function and the SA system is defined.

Restrictions

- A logical node instance within an IED can only be referenced once within all process / substation sections.
- Therefore, the combination of *iedName*, *IdInst*, *prefix*, *InClass* and *InInst* shall be unique within all process / substation sections if *iedName* is not None.
- The naming conventions for all these name parts shall be followed, even if used within a specification.

For specifications where *iedName*="None" everywhere, the combination of the other attributes must be unique within the same level. This means e.g. that the *prefix* or *InInst* should be

different if several LNs with identical InClass are used within the same substation part (i.e. same bay). This should also be the case for IED names, if application related naming (see 8.5.4) shall be used. This is NOT checked by the SCL schema, therefore it is the responsibility of the project engineer or system tool, if application related naming shall be used additionally to product (IED) related naming.

9.2.7 Non power equipment

Replace the existing text of Subclause 9.2.7 with the following new text:

To be able to model the connection of IED hosted logical nodes to functions other than power system-related ones such as fire fighting equipment or door supervision, the Process section as well as the *Substation* section contains the element *Function*, which again contains an arbitrary number of recursive *SubFunction* elements. Both elements are logical node containers and may also contain *GeneralEquipment*, if necessary. Both *Function* and *Subfunction* have the *name*, *type* and *desc* attributes, and might also contain the *Text* and *Private* elements. However, there are no connections defined between the equipment. It is possible to have *ConductingEquipment* within these elements to model relations between electrical and non-electrical parts. The *ConductingEquipment* then might contain (electrical) connection definitions to the electrical part. The *SubFunction* element can be used recursively to allow arbitrary function name levels.

```
<xs:complexType name="tFunction">
  <xs:complexContent>
    <xs:extension base="tPowerSystemResource">
      <xs:sequence>
        <xs:element name="SubFunction" type="tSubFunction" minOccurs="0" maxOccurs="unbounded">
          <xs:unique name="uniqueGeneralEquipmentInSubFunction">
            <xs:selector xpath="/scl:GeneralEquipment"/>
            <xs:field xpath="@name"/>
          </xs:unique>
        </xs:element>
        <xs:element name="GeneralEquipment" type="tGeneralEquipment" minOccurs="0"
maxOccurs="unbounded"/>
        <xs:element name="ConductingEquipment" type="tConductingEquipment" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attribute name="type" type="xs:normalizedString" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

```
<xs:complexType name="tSubFunction">
  <xs:complexContent>
    <xs:extension base="tPowerSystemResource">
      <xs:sequence>
        <xs:element name="GeneralEquipment" type="tGeneralEquipment" minOccurs="0"
maxOccurs="unbounded"/>
        <xs:element name="ConductingEquipment" type="tConductingEquipment" minOccurs="0"
maxOccurs="unbounded"/>
        <xs:element name="SubFunction" type="tSubFunction" minOccurs="0" maxOccurs="unbounded">
          <xs:sequence>
            <xs:attribute name="type" type="xs:normalizedString" use="optional"/>
          </xs:sequence>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="type" type="xs:normalizedString" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

The equipment type allowed within Function and Subfunction is termed GeneralEquipment.

```
<xs:complexType name="tGeneralEquipment">
  <xs:complexContent>
    <xs:extension base="tEquipment">
      <xs:attribute name="type" type="tGeneralEquipmentEnum" use="required"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

From the conducting equipment type list (Table 5) this is AXN, BAT, MOT, FAN, additionally to the codes defined in Table 9. Furthermore, private codes (containing only capital letters, starting with “E”) can be used. Other parts of this standard or other standards will define more type codes.

Table 9 – General Equipment codes from IEC 61850-7-4

Type code	Meaning
FIL	Filters
PMP	Pumps
VLV	Valves
AXN	Auxilliary power network

9.2.8 Substation section example

Replace the existing text of Subclause 9.2.8 with the following new text:

The following example for a system specification SSD, as shown in Figure 17, contains a substation section for substation Baden220_132 with one transformer T1 between voltage levels D1 and E1, and three bays D1Q1, E1Q2 and E1W1 as bus bar.

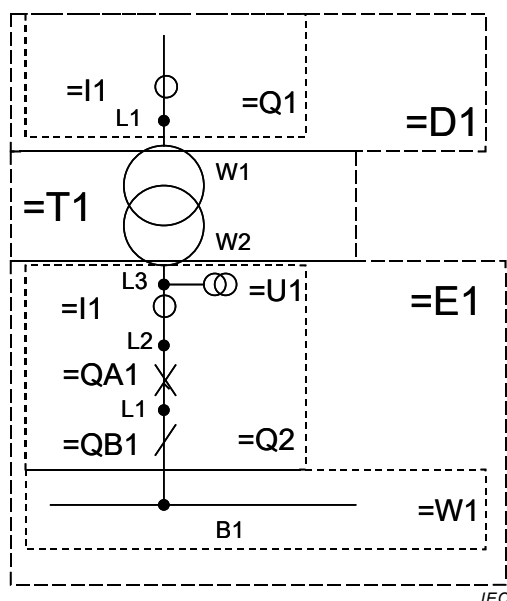


Figure 17 – Substation section example

The transformer T1 has two windings W1 and W2. Winding W1 is connected to a 220 kV voltage level D1 at bay Q1, connectivity node L1. Winding W2 is connected to the bay Q2 in 132 kV voltage level E1. From the attachment of logical nodes in the SSD file it can be seen that there is the measurement of a current transformer at the transformer, and a differential protection. At the 220 kV side (bay D1Q1) there is a distance protection.

The 132 kV bay E1Q2 contains a circuit breaker QA1 and a bus bar disconnector QB1, both electrically connected together at connectivity node L1, as well as a voltage transformer U1 at connectivity node L3, and current transformer I1 between the connectivity nodes L3 and L2. The connectivity node within the same bay is explicitly defined. A logical node of type CSWI controls each switch, and the LN CILO handles the interlocking. No association to IEDs is defined, as this is a functional specification only, so the *iedName* is per default None. In addition, the possibility of defining more details by *InType* references has not been used here.

```

<?xml version="1.0"?>
<SCL xmlns="http://www.iec.ch/61850/2003/SCL" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
version="2007" revision="B" release="3">
  <Header id="SSD Example " />
  <Substation name="Baden220_132">
    <PowerTransformer name="T1" type="PTR">
      <LNode lnInst="1" lnClass="PDIF" lnInst="F1"/>
      <LNode lnInst="1" lnClass="TCTR" lnInst="C1"/>
      <TransformerWinding name="W1" type="PTW">
        <Terminal connectivityNode="baden220_132/D1/Q1/L1" substationName="baden220_132"
voltageLevelName="D1" bayName="Q1" cNodeName="L1"/>
      </TransformerWinding>
      <TransformerWinding name="W2" type="PTW">
        <Terminal connectivityNode="baden220_132/E1/Q2/L3" substationName="baden220_132"
voltageLevelName="E1" bayName="Q2" cNodeName="L3"/>
      </TransformerWinding>
    </PowerTransformer>
    <VoltageLevel name="D1" nomFreq="50" numPhases="3">
      <Voltage multiplier="k" unit="V">220</Voltage>
      <Bay name="Q1">
        <LNode lnInst="1" lnClass="PDIS" lnInst="F1"/>
        <ConductingEquipment name="I1" type="CTR">
          <Terminal connectivityNode="baden220_132/D1/Q1/L1" substationName="baden220_132"
voltageLevelName="D1" bayName="Q1" cNodeName="L1"/>
        </ConductingEquipment>
        <ConnectivityNode name="L1" pathName="baden220_132/D1/Q1/L1"/>
      </Bay>
    </VoltageLevel>
    <VoltageLevel name="E1">
      <Voltage multiplier="k" unit="V">132</Voltage>
      <Bay name="Q2">
        <ConductingEquipment name="QA1" type="CBR">
          <LNode lnInst="1" lnClass="CILO" lnInst="C1"/>
          <Terminal connectivityNode="baden220_132/E1/Q2/L1" substationName="baden220_132"
voltageLevelName="E1" bayName="Q2" cNodeName="L1"/>
          <Terminal connectivityNode="baden220_132/E1/Q2/L2" substationName="baden220_132"
voltageLevelName="E1" bayName="Q2" cNodeName="L2"/>
        </ConductingEquipment>
        <ConductingEquipment name="QB1" type="DIS">
          <LNode lnInst="2" lnClass="CSWI" lnInst="C1"/>
          <LNode lnInst="2" lnClass="CILO" lnInst="C1"/>
          <Terminal connectivityNode="baden220_132/E1/W1/B1" substationName="baden220_132"
voltageLevelName="E1" bayName="W1" cNodeName="B1"/>
          <Terminal connectivityNode="baden220_132/E1/Q2/L1" substationName="baden220_132"
voltageLevelName="E1" bayName="Q2" cNodeName="L1"/>
        </ConductingEquipment>
        <ConductingEquipment name="I1" type="CTR">
          <Terminal connectivityNode="baden220_132/E1/Q2/L2" substationName="baden220_132"
voltageLevelName="E1" bayName="Q2" cNodeName="L2"/>
          <Terminal connectivityNode="baden220_132/E1/Q2/L3" substationName="baden220_132"
voltageLevelName="E1" bayName="Q2" cNodeName="L3"/>
        </ConductingEquipment>
        <ConductingEquipment name="U1" type="VTR">
          <Terminal connectivityNode="baden220_132/E1/Q2/L3" substationName="baden220_132"
voltageLevelName="E1" bayName="Q2" cNodeName="L3"/>
        </ConductingEquipment>
        <ConnectivityNode name="L1" pathName="baden220_132/E1/Q2/L1"/>
        <ConnectivityNode name="L2" pathName="baden220_132/E1/Q2/L2"/>
        <ConnectivityNode name="L3" pathName="baden220_132/E1/Q2/L3"/>
      </Bay>
      <Bay name="W1">
        <ConnectivityNode name="B1" pathName="baden220_132/E1/W1/B1"/>
      </Bay>
    </VoltageLevel>
  </Substation>
</SCL>

```

Figure 19 – UML description of IED related schema part – Base

Replace existing Figure 19 with the following new figure:

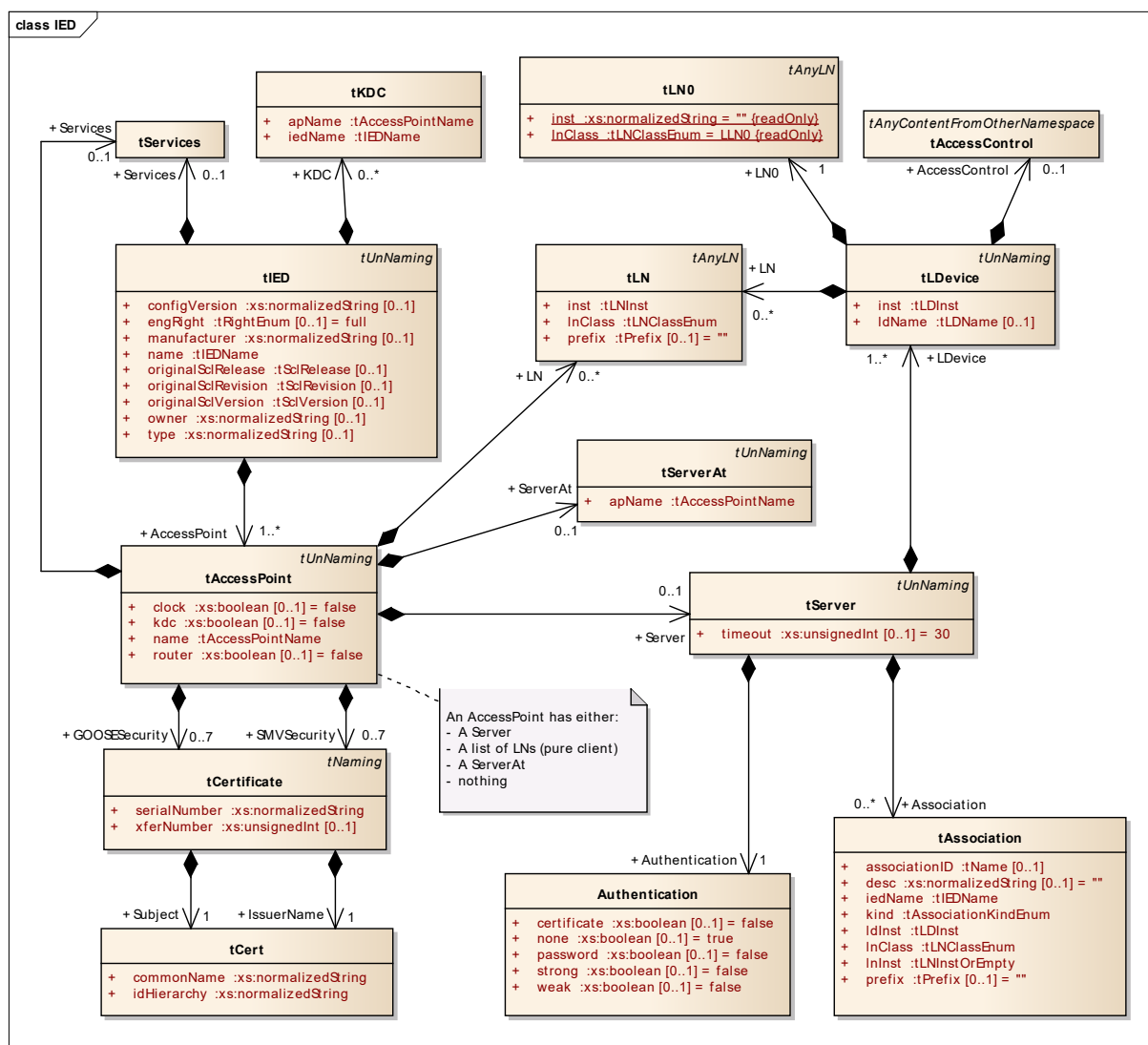


Figure 19 – UML description of IED related schema part – Base

Replace Table 10 and the following two paragraphs with the following new table and text:

Table 10 – Attributes of the IED element

Attribute name	Description
name	The identification of the IED. Within an ICD file describing a device type, the name shall be TEMPLATE. The IED name shall not be an empty string and not None, and shall be unique within an SCL file
desc	The description text
type	The (manufacturer specific) IED product type
manufacturer	The manufacturer's name
configVersion	The basic configuration version of this IED configuration
originalScIVersion	The original SCL schema version of the IEDs ICD file; optional, default "2003"
originalScIRevision	The original SCL schema revision of the IEDs ICD file; optional, default "A"
originalScIRelease	The original SCL schema release of the IEDs ICD file; optional, default '1': Observe that 2003A had no release at all
engRight	The engineering right transferred by a SED file (only fix, dataflow), or the current state in an SCD file. Values are full, dataflow, fix, the default is full
owner	The owner project of this IED, i.e. the Header id of that SCD file of that project which has the right to use the IED tool for this IED. The default is the Header id of the SCD file containing the IED

The IED configVersion above only identifies the IED basic configuration (IED type capabilities as defined/delivered by the manufacturer e.g. after preengineering of a flexibly engineerable IED type), and not its individual configuration after instantiation into a project. The version of the project-specific IED is a parameter of the IED instance, or of its logical nodes. It shall be contained in an SCL file as an attribute value of the attribute *LLN0.NamPIt.configRev*. This means that the value of the *LLN0.NamPIt.configRev* attribute must appear in a Val element of a DAI or DA element.

The *originalScIVersion* states the SCL version of the originally generated and imported ICD file of this IED, and *originalScIRevision* the SCL revision of this ICD file. Both attributes are set by the IED tool when creating the ICD or IID file, and shall be kept within an SCD file. If they are missing in the ICD / IID file, the system configuration tool might deduce them from the first imported ICD or IID file for this IED type. In case that they are missing, the default 2003A (SCL Edition 1) applies.

Replace existing Table 11 and the text from Table 11 onwards with the following new table and text:

Table 11 – List of service capabilities and setting elements and attributes

Service capability	Description
ClientServices	Indicates which general service classes this IED can use as a client / subscriber: <i>goose</i>, <i>gsse</i>, sampled values (<i>sv</i>), unbuffered reporting (<i>unbufReport</i>), buffered reporting (<i>bufReport</i>), reading logs (<i>readLog</i>). Default (missing element): supported client services not known (except possibly subscribers from GOOSE/GSSE elements) – look into PICS. Required for 2007 version. A pure client shall set at least one of the options to true. The attribute <i>supportsLdName</i> indicates that the client/subscriber understands explicit LD name setting at servers (mandatory for clients from Ed2 onwards). Default for an attribute missing: false The following attributes allow to describe further client/subscriber limits. If they are missing at IED and access point level, the limits are not known: <i>maxAttributes</i>: the maximal number of data set entries (as sum across all received data sets) supported by the client. <i>maxReports</i>: the maximal number of report control blocks the client can subscribe to. <i>maxGOOSE</i>: the maximum number of GOOSE messages receivable. <i>maxSMV</i>: the maximum number of SV messages receivable. <i>rGOOSE</i>: if true, GOOSE subscription at network level (layer 3) is supported; default = false. <i>rSV</i>: if true, SV message subscription at network level (layer 3) is supported; default = false. <i>noIctBinding</i>: if true, the IED configuration tool cannot bind incoming signals to internal addresses; instead it provides a template with supported internal addresses for possible incoming data. The binding task must be performed by the system configuration tool. The contained element <i>TimeSyncProt</i> allows to specify the supported time synchronization protocols as a client – details see at TimeSynchProt element below. If it is missing, time synchronization is not supported. The contained <i>McSecurity</i> element describes the supported security features at client side – details see at McSecurity element below. If it is missing here, the client supports no multicast related security features. As GOOSE or SV subscriber it can receive signed messages but ignores the signature. If the client security features are different for GOOSE and SV, the capabilities shall be declared at different access points.
DynAssociation	All services for dynamic building of associations. The <i>max</i> attribute indicates the guaranteed number of dynamic associations which are possible under all circumstances.
SettingGroups: SGEdit ConfSG	Setting group services belong to the setting group control block. If this control block is available, then the setting group service SelectActiveSG for activating a setting group is also available. The capability of online editing (IEC 61850-7-2 services SelectEditSG, ConfirmEditSGValues, SetSGValues) is decided with the SGEdit element. The capability to configure the (number of) setting groups by SCL can be also available (ConfSG). In this case the configured number must not be higher than the number in the ICD file. Both have the following attribute: <i>resvTms</i>: the value true at SGEdit means that this attribute is online visible at the SGCB; at ConfSG it means that the IED tool accepts configured values from an SCD file. The default value is false.
GetDirectory	Services for reading the contents of a server, i.e. the LD and LN directories (all LDs, LNs and DATA of the LNs). This is an option without attributes. Includes the IEC 61850-7-2 services GetServerDirectory, GetLogicalDeviceDirectory, GetLogicalNodeDirectory
GetDataObjectDefinition	Service to retrieve the complete list of all DA definitions of the referenced data that are visible and thus accessible to the requesting client by the referenced LN. It is a service without attributes. Refers to IEC 61850-7-2 service GetDataDefinition

Service capability	Description
DataObjectDirectory	Service to get the DATA defined in a LN. It is a service without attributes. Refers to IEC 61850-7-2 service GetDataDirectory
GetDataSetValue	Service to retrieve all values of data referenced by the members of the data set. It is a service without attributes. Refers to IEC 61850-7-2 service GetDataSetValues
SetDataSetValue	Service to write all values of data referenced by the members of the data set. It is a service without attributes. Refers to IEC 61850-7-2 service SetDataSetValues
DataSetDirectory	Service to retrieve FCD/FCDA of all members referenced in the data set. It is a service without attributes. Refers to IEC 61850-7-2 service GetDataSetDirectory
ConfDataSet	If ConfDataSet is not specified, then the default value of its <i>max</i> attribute is equal to the number of configured data sets, and they may only be modified. If it is specified, it is possible to configure new data sets up to the defined max, or modify existing ones at configuration time via SCL. The attribute meaning is: <i>max</i> – the maximum number of data sets <i>maxAttributes</i> – the maximum number of FCDA elements (data set entries) allowed in a data set. If an ICD file contains data sets with more elements, the system configurator shall not modify it. <i>modify</i> – TRUE means that configured data sets may be modified; default: true Note that the control block settings might also restrict data set modifications even if <i>modify</i>=TRUE.
DynDataSet	Services to dynamically create and delete data sets. Refers to IEC 61850-7-2 services CreateDataSet and DeleteDataSet. The attribute meaning is: <i>max</i> – the maximum number of creatable data sets (including any predefined or preconfigured data sets) <i>maxAttributes</i> – the maximum number of FCDA elements (data set entries) allowed in a dynamically created data set
ReadWrite	Basic data read and write facility; includes the IEC 61850-7-2 services GetData, SetData, and the Operate service, if appropriate data exist. It is a capability without attributes.
TimerActivatedControl	This element specifies that timer activated control services are supported. All other control-related services are specified directly at a DO with the ctlModel attribute. It is a service without attributes.
ConfReportControl	Capability of static (by configuration via SCL) creation of report control blocks. The system configurator can add ReportControls up to the defined max value and delete them, if no value of ReportSettings is fix. The attribute's meaning is: <i>max</i> – the maximum number of instantiable report control blocks. If this is equal to the number of configured instances, then no new instances can be created. If it is higher than the number of configured instances, the project engineer is allowed to create more instances for existing and new types up to this limit. <i>bufMode</i> – unbuffered, buffered, both; the buffer mode allowed to configure for new control block types. If it is not both and <i>bufConf</i>=false, existing RCBs of the opposite type shall not be deleted and their mode not be changed. <i>bufConf</i> – boolean. TRUE means, the <i>buffered</i> attribute of report control blocks can be changed via SCL by a system configurator. <i>maxBuf</i> – the maximum number of instantiable buffered control blocks. If it is missing, its value is equal to the max value. If supplied, its value shall be smaller than the max value. All RCBs instantiable by the system configurator are indexed.
GetCBValues	Read values of control blocks. It is a service without attributes
ConfLogControl	Capability of static (by configuration via SCL) creation of log control blocks. The system configurator can add LogControls up to the defined max value and delete them, if no value of LogSettings is fix. The attribute's meaning is: <i>max</i> – maximum number of instantiable log control blocks

Service capability	Description
ReportSettings	The report control block attributes for which a change of setting is possible at engineering time or online with services SetURCBValues respective SetBRCBValues: The attribute's meaning is: cbName – control block name (Fix, Conf; default = Fix) datSet – data set reference (Fix, Conf, Dyn, default=Fix; Fix means that both the datSet value and the structure of the referenced data set are fix) rptID – report identifier optFields – optional fields to include in report bufTime – buffer time trgOps – trigger options enable intgPd – integrity period resvTms – if true, the ResvTms attribute exists at all buffered control blocks. In this case, if the BRCB instance is allocated to a client, it should be configured as -1 (reserved), else as 0 (free). owner – if true, the report control block possesses an owner attribute. Default is false
LogSettings	The log control block attributes for which a change of setting is possible at engineering time or with service SetLCBValues: The attribute's meaning is: cbName – control block name (Fix, Conf; default = Fix) datSet – data set reference (Fix, Conf, Dyn, default=Fix; Fix means that both the datSet value and the structure of the referenced data set are fix) logEna – log enable trgOps – trigger options intgPd – integrity period
GSESettings	The GSE control block attributes for which a change of setting is possible at engineering time or with service SetGoCBValues: The attribute's meaning is: cbName – control block name(Fix, Conf; default = Fix) datSet – data set reference (Fix, Conf, Dyn, default=Fix; Fix means that both the datSet value and the structure of the referenced data set are fix) appID – application identifier dataLabel – value for the object reference if the corresponding element ist being sent (applies only to GSSE control blocks, which are deprecated) <i>kdaParticipant</i>: if <i>true</i>, the server access point supports KDA (Key Delivery Assurance) for GOOSE as described in IEC 62351, and the McSecurity element for the server shall also be specified. GSESettings allows the following subelement: A contained <i>McSecurity</i> element describes the supported security options available at each GOOSE control block – details see at McSecurity element below. If <i>kdaParticipant</i> is true, at least one of this options shall be true.

Service capability	Description
SMVSettings	The SMV control block attributes for which a change of setting is possible at engineering time or with service SetMSVCBValues respective SetUSVCBValues: The attribute's meaning is: cbName – control block name (Fix, Conf; default = Fix) datSet – data set reference (Fix, Conf, Dyn, default=Fix; Fix means that both the datSet value and the structure of the referenced data set are fix) svID – sample value identifier optFields – optional fields to include in sample value message smpRate – sample rate configuration capabilities (fix, conf, dyn) samplesPerSec – samples per second resp. seconds per samples are supported synchrSrcId – inclusion of grand master clock Id (IEC 61850-9-3); default = false nofASDU – the number of ASDUs in the SV message (Fix, Conf, default=Fix) pdvTimeStamp – indicates if the PDC time stamp can be included into the message; default is false kdaParticipant: if <i>true</i>, the server access point supports KDA (Key Delivery Assurance) as described in IEC 62351, and the McSecurity element for the server shall also be specified. SMVSettings allows the following (sub-)elements: SmpRate – defines the implemented sample rate(s) per period SamplesPerSec – defines the implemented sample rate(s) per second SecPerSamples – defines the implemented seconds between samples If no appropriate elements are defined, the sample rate per period or per second as defined by above attributes is assumed to be freely settable A contained subelement <i>McSecurity</i> describes the supported security options available at each SMV control block – details see at McSecurity element below. If kdaParticipant is true, at least one of the options shall be true.
ConfLNs	Describes what can be configured for LNs defined in an ICD file The attribute meanings are: fixPrefix – if false, prefixes can be set/changed; default = false fixLnInst – if false, LN instance numbers can be changed; default = false
ConfLdName	If this element is present, the IED allows as a server to define functional LD names (by means of the LDevice IdName attribute)
GSSEDir	GSE directory services according to IEC 61850-7-2. This capability has no attributes.
GOOSE	This element shows that the IED can be a GOOSE publisher or subscriber according to IEC 61850-7-2. The system configurator can add GoCBs up to the defined max value and delete them, if no value of GSSESetting is fix. The value max=0 is deprecated; instead ClientServices shall be used. The attributes meaning is: <i>max</i> = maximum number of GOOSE control blocks, which are configurable for publishing (max= 0 at an access point means that this access point does not support GOOSE sending). Shall be greater or equal to the number of defined GoCBs. <i>fixedOffs</i> = true indicates that fixed offsets are supported and can be set individually per GoCB. Default: false. <i>goose</i>: if true, layer 2 GOOSE sending is supported. Default: true. <i>rGOOSE</i>: if true, layer 3 (network level) GOOSE sending is supported. Default: false.
GSSE	This element shows that the IED can be a binary data GSSE publisher or subscriber according to IEC 61850-7-2. The attributes meaning is: <i>max</i> – maximum number of GSSE control blocks, which are configurable. Max=0 means only GSSE client. GSSE is deprecated since Edition 2.

Service capability	Description
SMVsc	This element shows that the IED can be a Sampled Value publisher or subscriber according to IEC 61850-7-2. The system configurator can add SVCBs up to the given max value or remove them, if no SMVSetting attribute has the value Fix. The value max=0 is deprecated; instead ClientServices shall be used. The attributes meaning is: max = maximum number of SMV control blocks, which are configurable for publishing (max=0 at an access point means that this access point does not support SV sending).delivery: indicates supported SV types (multicast, unicast, both); default. multicast. If the value is not both and deliveryConf=false, any existing SVCBs of the opposite type shall not be removed. deliveryConf: indicates if above type (multicast, unicast) can be configured by the system configurator; default: false. sv: if true, layer 2 SV is supported; default: true. rSV: if true, layer 3 (network level; multicast only) SV is supported; default false.
FileHandling	Indicates that file handling services are supported. The supported services additionally to all Get... services are defined in the PICS. The following attributes indicate on which protocol the file handling is based. If file handling is supplied at all, at least one of the attributes must be true. mms: MMS based file handling, default=true ftp: FTP based file handling, default=false ftps: FTP with SSL based file handling, default = false
SupSubscription	This element shows the capability to supervise GOOSE or SMV subscriptions. The attribute meaning is: maxGo – maximum number of GOOSE subscription supervision LNs (LN class LGOS) to be instantiated by a system configurator on the IED. Default: only preconfigured LNs can be used. If >0, at least one instance shall exist in the data model. maxSv – maximum number of SV subscription supervision LNs (LN class LSVS) to be instantiated on the IED. Default: only preconfigured LNs can be used. If >0, at least one instance shall exist in the data model. If the actually instantiated number of any category is less, the system configurator is allowed to add more as needed up to the defined max. The created instances shall reference the same LNodeType as an existing instance, have a modified unique LN instance number and be in the same logical device. If this element is missing, only preconfigured supervision LNs are allowed to be used
ConfSigRef	This element shows the IED capability to include input references into logical nodes. The attribute meaning is: max – maximum number of input references (e.g. data objects InRef and BlkRef, having CDC ORG) which can be instantiated on the IED. If the actually instantiated number is less and more are needed, they can be created by the ICT up to this limit and a new IID file included by the SCT. If this element is missing, only existing preconfigured input reference elements are allowed to be used.
CommProt	Specifies additional lower layer protocols. Observe that IP V4 always needs to be supported, if IP is used at all. ipv6: if true, IP V6 is supported. Default is false
TimeSyncProt	Declares the time synchronization protocols supported by the IED / access point either as time server or as time client if contained inside the ClientServices element, dependent also on the access point role. At least one of them shall be true, if time synchronization is supported at all.: sntp: the SNTP protocol; default is true c37_238: the IEEE 1588 method as defined in C37.238; default is false (deprecated) iec61850_9_3: the IEEE 1588 method as defined in IEC 61850-9-3; default is false other: any other method, e.g. PPS (although no protocol); default is false

Service capability	Description
RedProt	The types of communication redundancy which are supported on the IED. Default is false (none) hsr: indicates that HSR is supported by the IED prp: Indicates that PRP is supported by the IED rstp: indicates that RSTP is supported by the IED.
ValueHandling	Defines the handling allowances of valKind modifications for the system configurator. setToRo: if true, valKind=Set for fc=CF, fc=DC or fc=SP can be modified to RO; default is false.
McSecurity	Defines the supported multicast message security handling options at the server connected to this access point. The actually used option is configured at the control block instances. <i>signature</i> : if <i>true</i> , calculation of a signature is supported. Default: false. <i>encryption</i> : if <i>true</i> , message encryption is supported. Default: false.

NOTE Within an IED capability description, the maximum numbers specified above shall be a guaranteed (minimal) maximum, i.e. this number of elements shall be possible to instantiate respective use under all circumstances, for example even if some dynamic memory allocation allows sometimes to have more elements (than maximum) of one type at the cost of another element type (always at least maximum). The value 0 means that none such element can be used / created.

There are some setting and configuration capabilities which may be performed online, per SCL configuration, or just have fix values. These are indicated by the appropriate attribute values of *Dyn* (dynamically settable by IEC 61850 communication services), *Conf* (configurable via an SCL file), and *Fix* (only one fix value, typically documented in the SCL file). The *Dyn* option in this case always includes the *Conf* option, i.e. if online setting is possible, also setting via SCL shall be possible. If *Fix* applies to a data set reference, also the structure and definition of the referenced data set is fix. For consistency in this case the appropriate ConfReportControl, ConfLogControl, GOOSE and SMVsc elements shall not be supplied. If they are supplied, their max attribute shall be ignored for compatibility with earlier versions of the standard, and no control block instance of the appropriate class shall be removed or added. Additionally any IED configurator supplied control block having values which are NOT allowed to be configured by the system configurator according to the settings, shall keep these values and shall not be removed.

The Access point element of the IED defines the available communication access points.

```
<xs:complexType name="tAccessPoint">
  <xs:complexContent>
    <xs:extension base="tNaming">
      <xs:sequence>
        <xs:choice minOccurs="0">
          <xs:element name="Server" type="scl:tServer">
            <xs:unique name="uniqueAssociationInServer">
              <xs:selector xpath="/scl:Association"/>
              <xs:field xpath="@associationID"/>
            </xs:unique>
          </xs:element>
          <xs:element ref="scl:LN" maxOccurs="unbounded"/>
          <xs:element name="ServerAt" type="scl:tServerAt"/>
        </xs:choice>
        <xs:element name="Services" type="tServices" minOccurs="0" />
        <xs:element name="GOOSESecurity" type="tCertificate" minOccurs="0" maxOccurs="7"/>
        <xs:element name="SMVSecurity" type="tCertificate" minOccurs="0" maxOccurs="7"/>
      </xs:sequence>
      <xs:attribute name="router" type="xs:boolean" use="optional" default="false">
      </xs:attribute>
      <xs:attribute name="clock" type="xs:boolean" use="optional" default="false">
      </xs:attribute>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

The Access point is described by one of the elements: *Server*, *ServerAt* or *LN* list. It may optionally contain the security-related elements *GOOSESecurity* and *SMVSecurity*, stating the certificate information to be used for GOOSE sending and Sampled Value sending. The detailed meaning of these certificate descriptions can be found in IEC 62351-6.

The attributes of the Access point element are defined in Table 12.

Table 12 – Attributes of the Access point element

Attribute name	Description
name	Reference identifying this access point within the IED
desc	The description text
router	The presence and setting to true defines this IED to have a router function. By default, its value is false (no router function).
clock	The presence and setting to true defines this IED to be a master clock at this bus. By default, its value is false (no master clock).

The name attribute of the access point together with the name of the IED gives a unique reference for the access point within the SA system.

If neither a router, nor a clock, nor a server, nor a LN list is specified, the access point may only be used by client LNs in the same IED to access the bus to which it is connected. This is typical for a process bus access point of a bay level device, where the LNs offer their data via a server to the station bus only.

The Services element specifies the service capabilities of this access point additionally to those already stated at the IED level. It shall not contain IED general configuration or engineering-related capabilities like *ConfLNs*, *ConfLdName* or *ValueHandling* – these shall be defined at IED level. As general rule each element of the Services section shall only be used either at IED level, being common for all servers and access points, or at access point level, being then valid only for this access point. If it is not defined at any level, the defaults as specified above have to be taken. Observe that a *Server* access point and a *ServerAt* element at the same server shall not define conflicting service capabilities to the server, i.e. they might sometimes have to repeat the same options. If different access points define different numbers (e.g. number of control blocks accessible via the access point), then for the server the sum is valid. The following table makes this rule visible for each service element and attribute. The column IED marks those elements which are always valid for the whole IED. The column *Server* versus *ServerAt* defines the consequences if several access points to the same server exist. The basic idea is that they might have different service classes (client server, GOOSE, SV), however if they have the same server class, the capabilities must be identical.

It should be noted that the meaning of the max attribute of the *ConfReportControl*, *GOOSE* and *SMVsc* elements at an access point, especially for multiple access points at the same server, changes its semantics from ‘configurable’ to ‘capable of sending’. To make it simple, the ‘capability of sending’ can either be 0 (i.e. no sending) or all control blocks defined. Therefore these elements, if defined at IED level to indicate configuration capabilities, are additionally allowed at AP level, but only with value max=0.

Table 50 shows which service capabilities can be defined at which level. The IED column declares if only at IED (y) or also at access point level (n), and the *Server* / *ServerAt* column, which one can be different to the referred server at a *ServerAt* element.

Table 50 – Usage of Service element at IED level and Server / ServerAt level

Services			IED	Server + ServerAt
nameLength			y	
DynAssociation	max		n	may be different
SettingGroup	SGEdit		n	identical
	SGEdit	resvTms	n	identical
	ConfSG		n	identical
	ConfSG	resvTms	n	identical
GetDirectory			n	identical
GetDataObjectDefinition			n	identical
DataObjectDirectory			n	identical
GetDataSetValue			n	identical
SetDataSetValue			n	identical
DataSetDirectory			n	identical
ConfDataSet	max		n	identical
	maxAttributes		n	identical
	modify		n	identical
DynDataSet	max		n	may be different
	maxAttributes		n	may be different
ReadWrite			n	may be different
TimerActivatedControl			n	identical
ConfReportControl	max		n	identical or 0
	bufMode		n	identical
	bufConf		n	identical
GetCBValues			n	identical
ConfLogControl	max		n	identical
ReportSettings	cbName		n	identical
	datSet		n	identical
	rptID		n	identical
	optFields		n	identical
	bufTime		n	identical
	trgOps		n	identical
	intgPd		n	identical
	resvTms		n	identical
	owner		n	identical
LogSettings	cbName		n	identical
	datSet		n	identical
	logEna		n	identical
	trgOps		n	identical
	intgPd		n	identical
GSESettings	cbName		n	identical
	datSet		n	identical
	appID		n	identical
	dataLabel		n	identical

Services			IED	Server + ServerAt
	kdaParticipant		n	may be different
	McSecurity	signature	n	may be different
	McSecurity	encryption	n	may be different
SMVSettings	cbName		n	identical
	datSet		n	identical
	svID		n	identical
	optFields		n	identical
	smpRate		n	identical
	samplesPerSec		n	identical
	synchSrcId		n	identical
	SmpRate		n	identical
	samplesPerSec		n	identical
	SecPerSamples		n	identical
	kdaParticipant		n	may be different
	McSecurity	signature	n	may be different
	McSecurity	encryption	n	may be different
GSEDir			n	identical
GOOSE	max		n	identical or 0
	fixedOffs		n	may be different
SMVsc	max		n	identical or 0
	delivery		n	may be different
	deliveryConf		n	may be different
FileHandling			n	
	mms		n	may be different
	ftp		n	may be different
	ftps		n	may be different
ConfLNs	fixPrefix		y	
	fixLnInst		y	
ClientServices	goose		n	may be different
	gsse		n	may be different
	bufReport		n	may be different
	unbufReport		n	may be different
	readLog		n	may be different
	sv		n	may be different
	rGOOSE		n	may be different
	rSV		n	may be different
	noIctBinding		y	
	supportsLdName		n	may be different
	maxAttributes		n	may be different
	maxReports		n	may be different
	maxGOOSE		n	may be different
	maxSMV		n	may be different
	TimeSyncProt	sntp	n	may be different

Services			IED	Server + ServerAt
	TimeSyncProt	iec61850_9_3	n	may be different
	TimeSyncProt	other	n	may be different
	McSecurity	signature	n	may be different
	McSecurity	encryption	n	may be different
ConfLdName			y	
SupSubscription	maxGo		n	identical
SupSubscription	maxSv		n	identical
ConfSigRef	max		n	identical
ValueHandling	setToRO		y	
RedProt	hsr		n	may be different
	prp		n	may be different
	rstp		n	may be different
TimeSyncProt	sntp		n	may be different
	iec61850_9_3		n	may be different
	other		n	may be different
CommProt	ipv6		n	may be different
McSecurity	signature		n	may be different
	encryption		n	may be different

Project-specific access point attributes, such as the address within a communication system, are contained in the SCL Communication section.

The *ServerAt* element references an existing access point, which shall contain a server. It can be used to define another access point to the same server. It has to be taken in mind, that this other access point shall be connected to Subnetworks other than all other access points of this server, and that all access points share all control block instances of the defined server. This means especially, that if a GOOSE message shall be sent to different Subnetworks, then another GOOSE control block instance shall be used.

```
<xs:complexType name="tServerAt">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:attribute name="apName" type="tName" use="required"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

The *ServerAt* element has only the attribute *apName*, which references the *AccessPoint* on the same IED, which hosts / defines the server's data model.

The LN list allows a list of client LNs to be defined for a pure client access point. These client LNs can then be referenced to define the data flow between IEDs and to reserve control block instances, e.g. as reporting client to a report control block instance. For an example see IED A1KA1 in D.2.

9.3.5 LN0 and other Logical Nodes

Replace the second paragraph after the XML schema with the following new text:

The *Log* element indicates that the logical device, which is controlled by this LN0, contains a log, and its instance name *IdInst* can be used as log name within a log control block.

Replace the two paragraphs before Table 17 with the following new text:

The LN contains the following elements: DataSet (see 9.3.7), ReportControl(see 9.3.8), LogControl (see 9.3.9), DOI (see 9.3.6) and Inputs (see 9.3.13). Further it contains Log element(s), if the LN contains one or more logs. The Log element has as its only attribute its name, which shall be unique within the LN. If this log name is missing, the default name value is the LD instance name (LD inst value).

The attributes of the LN are defined as shown in Table 17.

Table 17 – Attributes of the LN element

Replace existing Table 17 with the following new table:

Table 17 – Attributes of the LN element

Attribute name	Description
desc	The description text for the logical node
InType	The instantiable type definition of this logical node, reference to a LNodeType definition
InClass	The LN class according to IEC 61850-7-x or other domain specific standards
inst	The LN instance number identifying this LN – an unsigned integer; leading zeros are not recommended, as they formally lead to another instance identification.
prefix	The LN prefix part

9.3.6 Data object (DOI) definition

Replace existing Table 18 with the following new table:

Table 18 – Attributes of the DOI element

Attribute name	Description
desc	The description text for the data
name	A standardized DO name for example from IEC 61850-7-4. It is the root name part as defined in the LNodeType definition. Its value must be unique at this level, i.e. there shall be at maximum one DOI element for the same data object.
ix	Index of a data element in case of an array type; shall not be used if DOI has no array type
accessControl	Access control definition for this data. The empty string (default) means that the higher-level access control definition applies. Possible values are SCSSM dependent.